Samba

Robert Eckstein, David Collier-Brown i Peter Kelly

Samba

Robert Eckstein, David Collier-Brown, Peter Kelly

Tytuł oryginału amerykańskiego: **Using Samba**
Tłumaczenie: Grzegorz Werner

Printed in Poland

Wydawnictwo RM, 00-987 Warszawa, skr. poczt. 144
e-mail: rm@rm.com.pl
WWW: http://www.rm.com.pl

ISBN 83-7243-098-5

Redaktor prowadzący: Mirosława Szymańska
Redakcja: Irmina Wala-Pęgierska
Korekta: Maria Najder
Redakcja CD-ROM-u: Wiesław Buszman
Opracowanie graficzne okładki według oryginału: Grażyna Jędrzejec
Redaktor techniczny: Beata Donner
Skład: Marcin Fabijański
Druk i oprawa: Oficyna Wydawnicza READ ME - Drukarnia w Łodzi

Wydanie I
10 9 8 7 6 5 4 3 2 1

Spis treści

Przedmowa

Jest dziewiąta rano, a ty właśnie przyszedłeś do centrum komputerowego po odświeżającym śnie. Twój pager milczy od miesięcy. Życie administratora systemu jest godne pozazdroszczenia – zresztą, czemu miałoby być inaczej, skoro zarządzasz taką siecią? Dwieście identycznych komputerów z tym samym systemem operacyjnym. Wszystkie drukarki są podłączone do sieci i dostępne z dowolnego miejsca budynku, a dzięki dostarczonym przez producenta skryptom konfiguracyjnym wszyscy pracownicy firmy dokładnie tak samo widzą przygotowane przez ciebie udziały dyskowe. Żyć nie umierać. Rozsiadasz się w fotelu i bierzesz pierwszy łyk porannej kawy...

I wtedy budzik wyrywa cię ze słodkich marzeń. Jeśli jesteś taki jak większość administratorów sieci, to opisany scenariusz zdarza ci się tylko w snach. Twój dzień prawdopodobnie zaczyna się od rozpaczliwej bieganiny, aby nakłonić do współpracy cztery różne platformy sprzętowe wyposażone w trzy odmienne systemy operacyjne – co może ci się nawet udać pod warunkiem, że telefon choć na chwilę przestanie dzwonić. Większość użytkowników nie rozumie, czemu tak trudno uzyskać dostęp do pliku w innym komputerze albo wydrukować coś na zdalnej drukarce. Dzienniki systemowe alarmują, że spóźniasz się ze sporządzeniem kopii zapasowej. Z nieznanej przyczyny pecety na drugim piętrze nie mogą znaleźć serwera taśm. Co może na to poradzić administrator sieci?

To proste: wziąć dzień wolnego, przeczytać tę książkę i poznać Sambę!

Pakiet Samba

Samba to pakiet narzędzi umożliwiających współdzielenie zasobów, takich jak drukarki i pliki, w sieci. Być może jest to pewne uproszczenie, ale Sambę naprawdę opracowano po to, aby ułatwić ci życie. Samba korzysta z protokołu Server Message Block (SMB), wspólnego produktu Microsoftu i IBM-u, w celu przesyłania danych między klientami Windows i serwerami uniksowymi w sieci TCP/IP.

Samba jest szczególnie atrakcyjna z następujących powodów:

- Używa tego samego protokołu, z którego standardowo korzystają systemy operacyjne IBM-u i Microsoftu, począwszy od DOS-a 3.0. Oznacza to, że rozumieją go

niemal wszystkie komputery Windows i nie ma potrzeby instalowania w nich dodatkowego oprogramowania klienckiego.

- Działa na różnych platformach, w tym w większości odmian Uniksa, OpenVMS, OS/2, AmigaDOS i NetWare. Dzięki temu jeden program na serwerze może zapewnić dostęp do plików i drukarek całej gromadzie pecetów.
- Jest bezpłatna. Istnieje kilka produktów komercyjnych, które odpowiadają funkcjonalnością Sambie, ale niektóre z nich są dość drogie. Samba stanowi alternatywę wobec pakietów, które mogłyby poważnie nadszarpnąć budżet działu informatycznego. Jest dystrybuowana na zasadach Powszechnej Licencji Publicznej GNU (GNU General Public License, GPL) i jest uważana przez autorów za oprogramowanie o *otwartym źródle* (*open source*). Oznacza to, że można bezpłatnie pobrać do swojego komputera zarówno aplikację, jak i jej kod źródłowy, a w razie potrzeby nawet poprawić oryginalny program.
- Administrowanie Sambą jest scentralizowane. Nie musisz biegać od komputera do komputera z dyskietką lub CD-ROM-em w ręku, chcąc uaktualnić oprogramowanie klienckie.

Samba jest kompletnym rozwiązaniem dla sieci lokalnych każdej wielkości – od domowej sieci z dwoma komputerami do korporacyjnego monstrum z setkami węzłów. Samba jest łatwa do zainstalowania i administrowania, zapewnia też przezroczyste środowisko sieciowe, które udostępnia użytkownikom wszystkie zasoby potrzebne do pracy. Po skonfigurowaniu Samba pozwala na:

- dostarczanie uniksowych plików klientom działającym pod kontrolą Windows, OS/2 i innych systemów operacyjnych,
- dostęp uniksowych komputerów do plików PC,
- udostępnianie drukarek sieciowych klientom Windows,
- świadczenie usług nazewniczych (rozgłoszeniowych i WINS),
- przeglądanie zasobów sieciowych przez klientów Windows,
- tworzenie grup roboczych lub domen Windows,
- wymuszanie uwierzytelniania nazw użytkowników i haseł klientów.

Dla kogo przeznaczona jest ta książka?

W założeniu odbiorcami tej książki są administratorzy Uniksa, którzy muszą udostępniać zasoby sieciowe komputerom PC, oraz wszyscy, którzy chcą umieścić uniksowy serwer w pecetowym środowisku. Nie zamierzamy jednak obciążać czytelników nauką niezliczonych narzędzi administracyjnych i fachowego słownictwa. Choć zakładamy, że znają oni podstawy administrowania uniksowym systemem, nie kierujemy tej książki do sieciowych ekspertów. Uczyniliśmy wszystko, co było w naszej mocy, aby rozwiać wątpliwości dotyczące mniej znanych definicji i terminów.

Ponieważ nie zakładamy dogłębnej znajomości systemu Windows, opiszemy czynności instalacyjne po stronie Windows bardzo szczegółowo, podając przykłady zarówno dla Windows 95/98, jak i Windows NT (między którymi występują pewne

subtelne różnice). Jeśli chodzi o stronę uniksową, podamy przykłady dla dwóch popularnych odmian tego systemu – Linuksa 2.0 i Solarisa 2.6.

Przed instalacją Samby

Zanim zaczniesz, powinieneś dysponować:

- CD-ROM-em z tej książki (który zawiera źródłową i binarną dystrybucję Samby 2.0.5) albo najnowszą dystrybucją pakietu, którą możesz pobrać bezpośrednio z Internetu pod adresem *http://www.samba.org*.
- Nazwami i adresami IP serwerów i klientów, które będą korzystać z Samby, maską używaną w sieci oraz nazwami i adresami IP swoich serwerów nazw domenowych (DNS).

Układ książki

Książkę tę można z grubsza podzielić na dwie części: instalowanie Samby (rozdziały od 1 do 3) oraz konfigurowanie i optymalizowanie Samby (rozdziały od 4 do 9). Oto zagadnienia omawiane w poszczególnych rozdziałach:

Rozdział 1, *Poznajemy Sambę*

Rozdział ten przedstawia wszystkie składniki pakietu i skrótowo omawia sieci NetBIOS i Windows.

Rozdział 2, *Instalowanie Samby w Uniksie*

Ten rozdział omawia konfigurowanie, kompilowanie, instalowanie i testowanie serwera Samby na platformie uniksowej.

Rozdział 3, *Konfigurowanie klientów Windows*

W tym rozdziale opisano konfigurowanie klientów Windows 95/98 i NT 4.0 do współpracy w sieci SMB. Znajduje się tu także krótki opis funkcjonowania protokołu SMB.

Rozdział 4, *Udziały dyskowe*

Rozdział ten opisuje poszczególne części pliku konfiguracyjnego Samby i omawia konfigurowanie usług dyskowych.

Rozdział 5, *Przeglądanie i zaawansowane udziały dyskowe*

Ten rozdział przedstawia opcje dyskowe i wyjaśnia usługi przeglądania w Sambie.

Rozdział 6, *Użytkownicy, bezpieczeństwo i domeny*

Rozdział omawia dodawanie kont użytkowników i mechanizmy bezpieczeństwa Samby oraz pokazuje, jak pracować z zaszyfrowanymi i niezaszyfrowanymi hasłami. Przedstawia także sposób konfigurowania Samby jako podstawowego kontrolera domeny dla klientów Windows 95/98 i NT.

Rozdział 7, *Drukowanie i odwzorowywanie nazw*

Ten rozdział opisuje konfigurowanie drukarek i usługi Windows Internet Name Service (WINS) w Sambie.

Rozdział 8, *Dodatkowe informacje o Sambie*
Rozdział ten zawiera omówienie innych czynności konfiguracyjnych i administracyjnych, jak na przykład tworzenie udziałów dyskowych na potrzeby programistów, kwestie obsługi języków narodowych i tworzenie kopii zapasowych za pomocą programu *smbtar*.

Rozdział 9, *Rozwiązywanie problemów*
Jeśli Samba sprawia problemy, w tym dość obszernym rozdziale znajdziesz wskazówki i rady dotyczące radzenia sobie z nimi.

Dodatek A, *Konfigurowanie Samby do obsługi SSL*
Ten dodatek szczegółowo omawia konfigurowanie Samby do obsługi połączeń Secure Sockets Layer (SSL) między serwerem a klientami.

Dodatek B, *Optymalizowanie wydajności Samby*
Opisano tu różne techniki mające na celu zwiększenie wydajności Samby.

Dodatek C, *Spis opcji konfiguracyjnych Samby*
Ten dodatek opisuje wszystkie opcje konfiguracyjne, których można użyć w pliku *smb.conf*.

Dodatek D, *Spis demonów i poleceń Samby*
Zawiera opis wszystkich demonów i narzędzi wchodzących w skład pakietu Samba. Oprócz tego zamieszczamy listę serwerów bliźniaczych (mirrorów) na całym świecie, z których można pobrać Sambę.

Dodatek E, *Pobieranie Samby za pomocą systemu CVS*
Ten dodatek omawia pobieranie najnowszej wersji Samby za pomocą systemu CVS.

Dodatek F, *Przykładowy plik konfiguracyjny*
Znajduje się tu obszerny plik konfiguracyjny Samby, który mógłby znaleźć zastosowanie w dużej firmie. Opatrzyliśmy go komentarzami, aby wyjaśnić bardziej zaawansowane opcje.

Konwencje typograficzne

W książce użyto 4 rodzajów czcionek o specjalnym przeznaczeniu:

Kursywa
Występuje w nazwach plików, rozszerzeniach plików, adresach URL, adresach internetowych, plikach wykonywalnych, poleceniach oraz miejscach, na które chciano zwrócić szczególną uwagę.

`Sta□a szeroko□□ liter`
Występuje w opcjach konfiguracyjnych Samby i pozostałym kodzie w tekście książki oraz w tekstach pojawiających się na ekranie komputera.

`Sta□a szeroko□□ liter, pogrubiona`
Występuje w poleceniach wpisywanych przez użytkownika oraz nowych opcjach konfiguracyjnych, na które chcemy zwrócić uwagę.

Sta□a szeroko□□ liter, kursywa
Występuje w zastępowalnych parametrach w kodzie i liniach poleceń.

Ikona sowy oznacza notę z ważnym komentarzem do pobliskiego tekstu.

Ikona indyka oznacza ostrzeżenie dotyczące pobliskiego tekstu.

Prośba o komentarze

Jako czytelnik tej książki możesz pomóc nam w poprawieniu jej następnego wydania. Jeśli znajdziesz w niej jakieś błędy lub nieścisłości, poinformuj o nich wydawnictwo O'Reilly. Daj nam także znać o wszystkich mylnych stwierdzeniach lub niezrozumiałych wyjaśnieniach. Korespondencję prosimy kierować na adres:

O'Reilly & Associates
101 Morris Street
Sebastopol, CA 95472
1-800-998-9938 (w Stanach Zjednoczonych lub Kanadzie)
1-707-829-0515 (numer międzynarodowy/lokalny)
1-707-829-0104 (fax)
bookquestions@ora.com

Powiedz, co możemy zrobić, aby ta książka stała się bardziej użyteczna. Potraktujemy serio wszystkie komentarze i uczynimy wszystko, aby ta książka była tak przydatna, jak to możliwe.

Podziękowania

Robert Eckstein

Przede wszystkim chciałbym złożyć podziękowanie Dave'owi Collierowi-Brownowi i Peterowi Kelly'emu za pomoc w tworzeniu tej książki. Chciałbym także podziękować redaktorom technicznym, którzy pomogli nadać tej książce ostateczny kształt w jakże krótkim terminie: Matthew Temple'owi, Jeremy'emu Allisonowi i oczywiście Andrew Tridgellowi. Andrew i Jeremy zasługują na szczególne uznanie, nie tylko za stworzenie tak wspaniałego programu, ale także za nieocenione wsparcie w końcowym etapie pisania tej książki – kapelusze z głów! Ciepły uścisk dla mojej żony, Michelle, która znów musiała znosić męża przesiąkniętego kofeiną i starającego się nadgonić terminy. Dziękuję również Dave'owi Sifry i personelowi LinuxCare z San Francisco za ugoszczenie mnie na czas wizyty Andrew Tridgella. Wreszcie chciałbym wyrazić najgłębszą wdzięczność głównemu redaktorowi, Andy'emu Oramowi, który bardzo cierpliwie prowadził nas przez poszczególne etapy pisania tej książki, aż stała się taka, jaką być powinna.

David Collier-Brown

Przede wszystkim chciałbym podziękować Joyce, która zdołała wytrzymać ze mną, kiedy byłem pochłonięty współtworzeniem tej książki. Dziękuję Andy'emu Oramowi za krytykę i zaproszenie mnie do udziału w tym przedsięwzięciu; personelowi Opcomu, który tolerował jawnego wariata w swoich szeregach, oraz Ianowi MacMillanowi, który z własnej inicjatywy przełożył moją pisaninę z komputerowego na angielski. Specjalne podziękowania składam Perry'emu Donhamowi, Drew Sullivan i Jerry'emu DeRoo.

Peter Kelly

Książka ta zawdzięcza swoje powstanie kilku osobom, przed którymi chcę się tu nisko pokłonić. Dave Collier-Brown, a następnie Bob Eckstein przejęli moją część projektu i naznaczyli ją doskonałym stylem i profesjonalizmem; trudno mi wyrazić wdzięczność za kształt, jaki nadali tej książce. Redaktor Andy Oram to zdecydowanie najcierpliwszy i najsympatyczniejszy człowiek, jakiego miałem przyjemność spotkać. Nie sądzę, że bym uczestniczył w pisaniu tej książki bez Xaviera Cazina z wydawnictwa O'Reilly, który złożył mi tę propozycję po przeczytaniu mojego artykułu w „Linux Journal". Chciałbym też podziękować wszystkim konsultantom z JDP.COM (Jerry'emu, Peggyann, Drew, Gordowi, Jerome'owi, Markowi, Rickowi – nie zdołam wymienić wszystkich!) za to, że pozwolili mi dalej ze sobą pracować. Dziękuję wszystkim z wydawnictwa O'Reilly, z którymi współpracowałem. Nie mogę też pominąć zespołu Samby, który stworzył program wart opisania w książce. I najważniejsze – dziękuję ci, Kate McKay, że jesteś ze mną od tak dawna!

Wszyscy chcielibyśmy serdecznie podziękować Perry'emu Donhamowi, który pomógł stworzyć pierwszy szkic tej książki. Choć inne obowiązki uniemożliwiły mu dalszą współpracę, jego sugestie w znacznej mierze ukształtowały tę książkę. Czujemy się w obowiązku nadmienić, że fragmenty rozdziału o przeglądaniu pochodzą z tekstu napisanego dla wydawnictwa O'Reilly przez Dana Shearera.

Mamy wielki dług wdzięczności u pracowników działu produkcyjnego wydawnictwa za znakomicie wykonaną pracę. Sarah Jane Shangraw pracowała przez długie godziny, aby uwzględnić nasze niekończące się poprawki, a Rob Romano niestrudzenie edytował rysunki, doprowadzając je do doskonałości. Specjalne podziękowania za pomoc należą się Claire Cloutier LeBlanc, Rhonowi Porterowi i Mike'owi Sierrze – bez nich ta książka zapewne nie ukazałaby się w ogóle, a jeśli nawet, to spóźniona przynajmniej o rok.

1

Poznajemy Sambę

Jeśli jesteś typowym administratorem systemu, doskonale wiesz, co to znaczy być *zawalonym* pracą. Twój dzień wypełniają kwestie niezgodności sprzętu i oprogramowania, awarie systemu, kłopoty z kopiami zapasowymi i niekończące się skargi podirytowanych użytkowników. Pomysł dodawania kolejnego programu do mieszanki narzędzi, z którymi musisz codziennie się zmagać, może wydać się nie na miejscu. Jeśli jednak jesteś zdecydowany uprościć swoje środowisko robocze i zmniejszyć nakład pracy potrzebny do utrzymania go w ruchu, Samba może okazać się właśnie tym, na co czekałeś.

Oto przykład: jeden z autorów tej książki opiekował się 70 programistami uniksowymi, którzy korzystali z 5 uniksowych serwerów. Jego sąsiad nadzorował 20 użytkowników Windows 3.1 i 5 serwerów Windows NT i OS/2. Mówiąc oględnie, administrator Windows 3.1 miał pełne ręce roboty. Kiedy wreszcie odszedł z pracy, a kontroler domeny odmówił posłuszeństwa, na pomoc przyszła Samba. Nasz autor szybko zastąpił serwery NT i OS/2 Sambą działającą na serwerze uniksowym, a z czasem zakupił komputery PC dla większości programistów. Okazało się przy tym, że nie trzeba zatrudniać nowego pracownika – administrator zarządza teraz jednym scentralizowanym serwerem uniksowym, zamiast 50 rozproszonymi komputerami PC.

Jeśli zdajesz sobie sprawę, że twoja sieć sprawia problemy, i szukasz lepszych rozwiązań, zachęcamy cię do przeczytania tej książki. Jeśli zaś słyszałeś już o Sambie i zastanawiasz się, czy może ci się do czegoś przydać, odpowiedź znajdziesz właśnie tutaj. Będziemy towarzyszyć ci w drodze do poznania Samby i jej pełnego potencjału. Już niedługo będziesz umiał zapewnić uniksowe usługi wszystkim komputerom Windows – bez tracenia czasu i pieniędzy. Brzmi zachęcająco? Doskonale, zatem zaczynamy.

Co to jest Samba?

Samba to zbiór uniksowych aplikacji rozumiejących protokół SMB (*Server Message Block*). Wiele systemów operacyjnych, w tym Windows i OS/2, używa SMB do komunikacji sieciowej między klientami i serwerami. Samba umożliwia uniksowym serwerom porozumiewanie się za pomocą tego samego protokołu, którego używają

systemy Microsoftu. Zatem uniksowy komputer z Sambą może udawać serwer w sieci Microsoftu i udostępniać następujące usługi:

- współdzielić systemy plików,
- współdzielić drukarki podłączone do serwera i klientów,
- wspomagać klientów w przeglądaniu Otoczenia sieciowego,
- uwierzytelniać klientów logujących się do domeny Windows,
- wspomagać odwzorowywanie nazw jako serwer WINS.

Samba jest dziełem Andrew Tridgella, który obecnie kieruje grupą programistów Samby ze swojego domu w Canberrze w Australii. Projekt ruszył w 1991 roku, kiedy na potrzeby swojej lokalnej sieci Andrew napisał program serwera plików, który obsługiwał protokół DEC firmy Digital Pathworks. Choć wówczas Andrew nie był tego świadom, protokół ten okazał się później protokołem SMB. Po kilku latach Andrew rozwinął swój serwer SMB i zaczął dystrybuować go w Internecie pod nazwą SMB Server. Nazwę tę trzeba było jednak zmienić – nosił ją produkt innej firmy – więc Andrew spróbował uniksowego podejścia do zmiany nazw plików:

```
grep -i 's.*m.*b' /usr/dict/words
```

Odpowiedź brzmiała:

```
salmonberry samba sawtimber scramble
```

I tak narodziła się nazwa „Samba".

Dziś głównymi składnikami pakietu Samba są dwa uniksowe demony, które udostępniają współdzielone zasoby – zwane *udziałami* – sieciowym klientom SMB. (Udziały bywają nazywane także *usługami*). Tymi demonami są:

smbd
: Demon umożliwiający współdzielenie plików i drukarek w sieci SMB i zapewniający uwierzytelnianie klientów SMB.

nmbd
: Demon świadczący usługi Windows Internet Name Service (WINS) i wspomagający przeglądanie zasobów sieci.

Samba jest obecnie poprawiana i rozwijana przez grupę ochotników pod przewodnictwem Andrew Tridgella. Podobnie jak system operacyjny Linux, Samba jest uważana przez autorów za *oprogramowanie o otwartym źródle* (*Open Source Software*, OSS) i rozpowszechniana na warunkach Powszechnej Licencji Publicznej GNU (*GNU General Public License*, GPL). Od samego początku prace nad Sambą są w części sponsorowane przez Narodowy Uniwersytet Australii, gdzie Andrew uzyskał swój tytuł naukowy. Część prac wsparły niezależne firmy, takie jak Whistle i SGI. Właśnie fakt, że zarówno instytucje komercyjne, jak i niekomercyjne wydają pieniądze na projekt Open Source, stanowi najważniejsze świadectwo popularności Samby.

Microsoft również przyczynił się do rozwoju pakietu, upubliczniając definicję protokołu SMB i jego internetowej odmiany, Common Internet File System (CIFS), w standardowym dokumencie Request For Comments (RFC). CIFS to nowa nazwa dla przyszłych wersji protokołu SMB, który będzie używany w nowych produktach

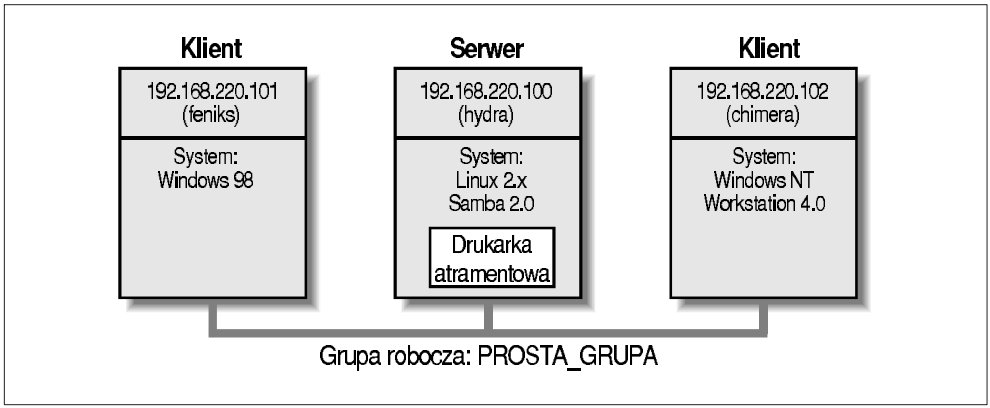
Microsoftu – oba terminy będą w tej książce używane zamiennie. Zatem często nazwę protokołu będziemy zapisywać jako „SMB/CIFS".

Do czego może się przydać Samba?

Jak wyjaśniono wcześniej, Samba umożliwia koegzystencję komputerów z Uniksem i Windows w ramach jednej sieci. Istnieje jednak kilka bardziej konkretnych powodów, dla których warto umieścić w sieci serwer Samby:

- Nie chcesz – lub nie możesz – płacić za pełny serwer Windows NT, ale przydałoby ci się narzędzie o podobnych funkcjach.
- Chcesz stworzyć wspólny obszar dla danych lub katalogów użytkowników w trakcie migracji z Windows do Uniksa lub *vice versa*.
- Chcesz, żeby uniksowe i windowsowe stacje robocze współdzieliły drukarki.
- Chcesz mieć dostęp do plików NT z uniksowego serwera.

Przyjrzyjmy się Sambie w działaniu. Załóżmy, że mamy następującą konfigurację sieci: komputer uniksowy z zainstalowaną Sambą, któremu nadamy nazwę `hydra`, oraz dwa klienty Windows, którym nadamy nazwy `feniks` i `chimera`, wszystkie połączone siecią lokalną (*Local Area Network*, LAN). Załóżmy także, że do `hydry` podłączone są lokalna drukarka atramentowa, `lp`, oraz udział dyskowy o nazwie `siec` i że oba te zasoby będą udostępniane pozostałym komputerom. Sieć ta jest przedstawiona na rysunku 1.1.



Rysunek 1.1. Prosta konfiguracja sieciowa z serwerem Samby

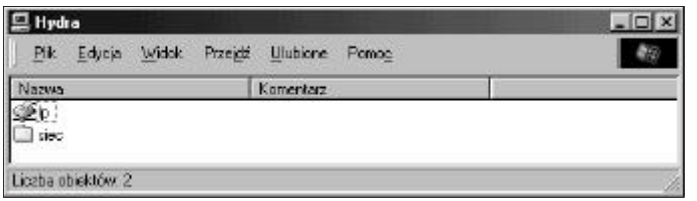
W tej sieci wszystkie komputery są w tej samej *grupie roboczej*. Grupa robocza to po prostu etykieta identyfikująca arbitralny zbiór komputerów i ich zasobów w sieci SMB. W sieci może jednocześnie istnieć kilka grup roboczych, ale w naszej przykładowej sieci będziemy mieli tylko jedną grupę o nazwie PROSTA_GRUPA.

Współdzielenie dysków

Jeśli wszystko jest poprawnie skonfigurowane, powinniśmy zobaczyć serwer Samby, `hydr`, w oknie Otoczenia sieciowego na pulpicie windowsowego komputera `feniks`. Widać to na rysunku 1.2, który przedstawia Otoczenie sieciowe `feniksa`, w tym `hydr` i pozostałe komputery należące do grupy roboczej PROSTA_GRUPA. Zauważ, że u góry listy widoczna jest ikona Cała sieć. Jak przed chwilą wspomnieliśmy, w sieci SMB może jednocześnie istnieć kilka grup roboczych. Jeśli użytkownik kliknie ikonę Cała sieć, zobaczy listę wszystkich grup istniejących w sieci.

Rysunek 1.2. Okno Otoczenia sieciowego

Możemy przyjrzeć się bliżej `hydrze`, klikając dwukrotnie jej ikonę. Tym samym połączymy się z `hydr` i zażądamy przesłania listy jej *udziałów* – zasobów dyskowych i drukarek – udostępnianych klientom. W tym przypadku widzimy drukarkę o nazwie `lp` oraz udział dyskowy o nazwie `siec`, co przedstawia rysunek 1.3. Zwróć uwagę, że w oknie Windows nazwy hostów są pisane małymi literami z pierwszą dużą (Hydra). Wielkość liter w nazwach hostów nie ma znaczenia, więc w różnych oknach i wynikach poleceń będziesz trafiał na nazwy pisane różnie: hydra, Hydra i HYDRA, odnoszące się jednak do tego samego komputera. Dzięki Sambie Windows 98 rozpoznaje serwer uniksowy jako serwer SMB i może otworzyć folder `siec` zupełnie tak samo, jak inny folder systemowy.



Rysunek 1.3. Udziały udostępniane przez serwer hydra widziane z komputera feniks

Systemy Windows 95/98/NT umożliwiają mapowanie litery dysku na katalog sieciowy za pomocą opcji Mapuj dysk sieciowy w Eksploratorze Windows*. Kiedy to zrobisz, twoje aplikacje będą mogły korzystać z folderu sieciowego, używając standardowej litery dysku. Dzięki temu będziesz mógł zapisywać w nim dane, instalo-

* Można także kliknąć prawym przyciskiem myszy współdzielony zasób w oknie Otoczenia sieciowego i wybrać z menu polecenie Mapuj dysk sieciowy.

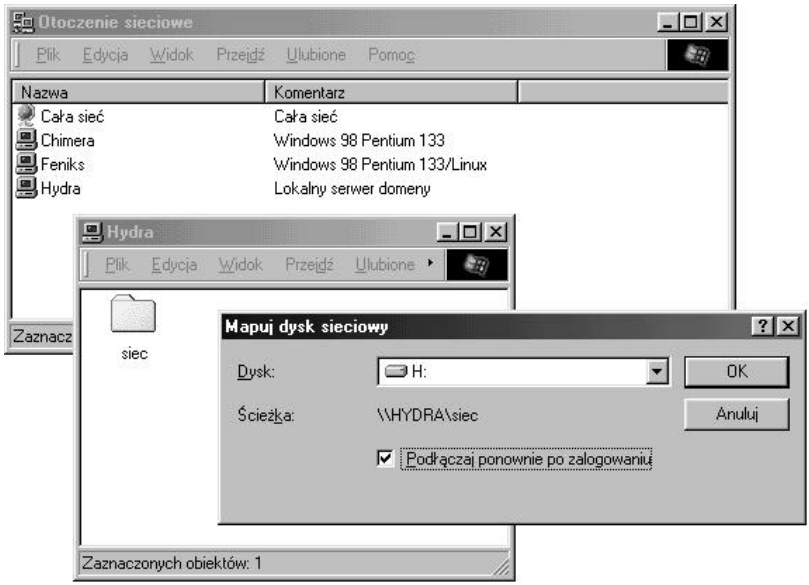
wać i uruchamiać z niego programy, a nawet zabezpieczyć dostęp do niego za pomocą hasła. Rysunek 1.4 przedstawia przykład mapowania litery dysku na katalog sieciowy.

Przyjrzyj się uważnie polu Ścieżka: w oknie dialogowym z rysunku 1.4. Katalog w zdalnym komputerze jest reprezentowany przez dwa odwrotne ukośniki, po których następuje nazwa zdalnego komputera, kolejny odwrotny ukośnik oraz nazwa sieciowego katalogu, jak niżej:

```
\\zdalny-komputer\katalog
```

Zapis ten znany jest w świecie Windows jako *uniwersalna konwencja nazewnicza* (*Universal Naming Convention*, UNC). Na przykład okno dialogowe z rysunku 1.4 reprezentuje katalog `siec` w serwerze `hydra` jako:

```
\\HYDRA\siec
```



Rysunek 1.4. Mapowanie katalogu sieciowego na windowsową literę dysku

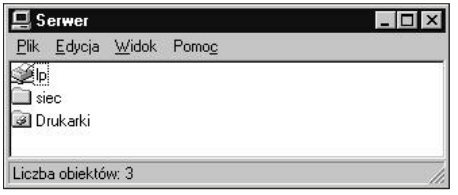
Zapewne przypomina ci to *jednolity lokalizator zasobów* (*Uniform Resource Locator*, URL), używany przez przeglądarki WWW, takie jak Microsoft Explorer i Netscape Navigator, do nawiązywania połączeń z serwerami internetowymi. Nie myl tych dwóch notacji: przeglądarki WWW zwykle używają zwykłych, a nie odwrotnych ukośników, a przed pierwszymi dwoma ukośnikami znajduje się nazwa protokołu transmisji danych (na przykład ftp, http) oraz dwukropek. Prawdę mówiąc, adresy URL i UNC to dwie zupełnie różne rzeczy.

Kiedy skonfigurujesz stację sieciową, system Windows i jego programy będą zachowywały się tak, jakby katalog sieciowy był lokalnym dyskiem. Jeśli masz aplikacje, które umożliwiają współpracę wielu użytkowników w sieci, możesz zainstalować je

na stacji sieciowej*. Rysunek 1.5 pokazuje, jak taka stacja sieciowa wygląda na tle innych urządzeń pamięciowych Windows 95. Zwróć uwagę na ikonę dysku połączonego z przewodem, która wskazuje, że jest to dysk sieciowy, a nie stały.

Rysunek 1.5. Katalog sieciowy mapowany przez klienta na literę dysku G

Z naszego komputera Windows NT Workstation, `chimery`, Samba niemal niczym nie różni się od Windows 98. Rysunek 1.6 przedstawia ten sam widok `hydry` z okna Otoczenia sieciowego w Windows NT 4.0. Skonfigurowanie stacji sieciowej za pomocą opcji Mapuj dysk sieciowy w Windows NT Workstation 4.0 dałoby identyczny rezultat.



Rysunek 1.6. Udziały udostępniane przez hydrę (oglądane z chimery)

Współdzielenie drukarki

Prawdopodobnie zauważyłeś, że wśród udziałów udostępnianych przez `hydr` (patrz rysunek 1.3) pojawia się drukarka `lp`. Oznacza to, że uniksowy serwer ma drukarkę, której mogą używać wszystkie klienty SMB w grupie roboczej. Dane wysyłane do drukarki przez różne klienty będą buforowane na uniksowym serwerze i drukowane w takiej kolejności, w jakiej zostały odebrane.

Konfigurowanie udostępnianej przez Sambę drukarki po stronie Windows jest jeszcze łatwiejsze, niż konfigurowanie udziału dyskowego. Klikając dwukrotnie ikonę drukarki i określając jej producenta i model, możemy zainstalować jej sterownik w kliencie. System Windows będzie mógł wówczas poprawnie formatować informacje przesyłane do drukarki sieciowej i korzystać z niej tak, jakby była podłączona lokalnie (jak to zrobić, wyjaśnimy dalej w tym rozdziale). Rysunek 1.7 przedstawia

* Trzeba zaznaczyć, że wiele umów licencyjnych zabrania instalowania programu w sieci tak, aby mogło z niego korzystać wielu użytkowników. Należy sprawdzić dołączoną do produktu licencję i upewnić się, czy zezwala ona na taką instalację.

skonfigurowaną drukarkę sieciową w oknie Drukarki w Windows 98. Podobnie jak w przypadku dysku, ikona przedstawia drukarkę podłączoną do przewodu, identyfikując ją tym samym jako drukarkę sieciową.

Rysunek 1.7. Drukarka sieciowa udostępniana przez hydrę (oglądana z chimery)

Oglądanie zasobów od strony Uniksa

Jak wspomniano wcześniej, Samba z perspektywy Uniksa jest zbiorem demonów. Możesz obejrzeć je za pomocą uniksowych poleceń `ps` i `netstat`, możesz przeczytać generowane przez nie komunikaty w ich własnych plikach diagnostycznych albo za pośrednictwem dziennika `syslog` (w zależności od konfiguracji Samby) i możesz skonfigurować je, modyfikując plik właściwości Samby: *smb.conf*. Oprócz tego, jeśli chcesz wiedzieć, co w danej chwili robią poszczególne demony, Samba dysponuje programem *smbstatus*, który wyświetla wszystkie niezbędne informacje. Oto przykład:

```
# smbstatus
Samba version 2.0.4
Service      uid      gid      pid     machine
----------------------------------------------
siec         davecb   davecb   7470    feniks  (192.168.220.101) Sun May 16
siec         davecb   davecb   7589    chimera (192.168.220.102) Sun May 16

Locked files:
Pid    DenyMode   R/W        Oplock           Name
--------------------------------------------------
7589   DENY_NONE  RDONLY     EXCLUSIVE+BATCH  /home/samba/quicken/inet/common/
system/help.bmp   Sun May 16 21:23:40 1999
7470   DENY_WRITE RDONLY     NONE             /home/samba/word/office/findfast.exe
Sun May 16 20:51:08 1999
7589   DENY_WRITE RDONLY     EXCLUSIVE+BATCH  /home/samba/quicken/lfbmp70n.dll
Sun May 16 21:23:39 1999
7589   DENY_WRITE RDWR       EXCLUSIVE+BATCH  /home/samba/quicken/inet/qdata/
runtime.dat   Sun May 16 21:23:41 1999
7470   DENY_WRITE RDONLY     EXCLUSIVE+BATCH  /home/samba/word/office/osa.exe
Sun May 16 20:51:09 1999
7589   DENY_WRITE RDONLY     NONE             /home/samba/quicken/qversion.dll
Sun May 16 21:20:33 1999
7470   DENY_WRITE RDONLY     NONE             /home/samba/quicken/
qversion.dll   Sun May 16 20:51:11 1999
```

```
Share mode memory usage (bytes):
   1043432(99%) free + 4312(0%) used + 832(0%) overhead = 1048576(100%) total
```

Status Samby jest wyrażony trzema zbiorami danych, z których każdy zajmuje oddzielną sekcję. Pierwsza sekcja informuje, jakie komputery są połączone z serwerem Samby, identyfikując każdy z nich za pomocą nazwy komputera (`feniks` i `chimera`) oraz adresu IP. Druga sekcja zawiera nazwy i stany współdzielonych plików, które są obecnie w użyciu, w tym tryb dostępu (zapis i odczyt) oraz ewentualne blokady plików. Wreszcie Samba raportuje ilość pamięci przydzielonej zarządzanym przez siebie udziałom, w tym część aktywnie używaną przez udziały i pewną część nadmiarową (zwróć uwagę, że to nie to samo, co całkowita ilość pamięci używana przez procesy *smbd* i *nmbd*).

Nie przejmuj się, jeśli ta statystyka jest dla ciebie niejasna. W miarę czytania tej książki stanie się bardziej zrozumiała.

Poznajemy sieć SMB/CIFS

Po tym krótkim wprowadzeniu zapoznamy się ze środowiskiem zaadoptowanym przez Sambę: siecią SMB/CIFS. Operacje sieciowe w SMB znacznie różnią się od pracy w uniksowej sieci TCP/IP – trzeba nauczyć się kilku nowych pojęć i przyswoić sobie mnóstwo informacji. Zaczniemy od podstawowych koncepcji dotyczących sieci SMB, później opiszemy, jak zaimplementował ją Microsoft, a wreszcie pokażemy, gdzie na tym tle plasuje się Samba.

NetBIOS

Cofnijmy się nieco w czasie. W roku 1984 w IBM-ie opracowano prosty interfejs programowy aplikacji (*Application Programming Interface*, API) do łączenia komputerów w sieć, który nazwano *Network Basic Input/Output System* (NetBIOS). Interfejs NetBIOS zapewniał fundamentalny mechanizm łączenia się aplikacji z innymi komputerami i współdzielenia z nimi danych.

Interfejs NetBIOS można uważać za sieciowe rozszerzenie standardowych wywołań interfejsu BIOS. W BIOS-ie każde niskopoziomowe wywołanie jest sprzężone ze sprzętem lokalnego komputera i nie musi podróżować przez sieć do miejsca przeznaczenia. Jednakże NetBIOS pierwotnie musiał wymieniać instrukcje z innymi komputerami połączonymi sieciami IBM PC lub Token Ring. Wymagał zatem niskopoziomowego protokołu transportowego, który przenosiłby żądania między komputerami.

Pod koniec roku 1985 w IBM-ie opracowano taki protokół, który po scaleniu z interfejsem NetBIOS otrzymał nazwę *NetBIOS Extended User Interface* (NetBEUI). Protokół NetBEUI zaprojektowano na potrzeby małych sieci lokalnych. Pozwalał on komputerom na zarezerwowanie nazwy (nie dłuższej niż 15 znaków), która w danej chwili nie była używana w sieci. Przez małą sieć lokalną rozumiemy sieć o liczbie węzłów mniejszej niż 255, co już w 1985 uważano za praktyczne ograniczenie!

Protokół NetBEUI był powszechnie wykorzystywany w aplikacjach sieciowych, w tym tych pisanych dla Windows for Workgroups. Później pojawiły się również implementacje NetBIOS-u ponad protokołami sieciowymi IPX Novella, konkurując z rozwiązaniami opartymi na NetBEUI. Jednakże społeczność internetowa preferowała protokoły TCP/IP i UDP/IP, więc oparcie interfejsu NetBIOS o te protokoły wkrótce stało się koniecznością.

Jak wiadomo, TCP/IP używa liczb do reprezentowania adresów komputerów (na przykład 192.169.220.100), natomiast NetBIOS tylko nazw. Pogodzenie tych wymagań było kluczowym problemem na drodze do połączenia obu protokołów. W 1987 roku grupa Internet Engineering Task Force (IETF) opublikowała dwa dokumenty standaryzacyjne, zatytułowane RFC 1001 i 1002, które definiowały sposób działania NetBIOS-u ponad siecią TCP/UDP. Dokumenty te do dziś są wyrocznią dla istniejących implementacji NetBIOS-u, łącznie z tymi obecnymi w systemach operacyjnych Windows Microsoftu, a także w pakiecie Samba.

Obecnie standard definiowany przez te dokumenty znany jest jako *NetBIOS over TCP/IP*, w skrócie NBT. Standard NBT (RFC 1001/1002) definiuje trzy usługi sieciowe:

- usługę nazewniczą,
- dwie usługi komunikacyjne: datagramy i sesje.

Usługa nazewnicza rozwiązuje wspomniany wyżej problem z tłumaczeniem adresów na nazwy: pozwala każdemu komputerowi zadeklarować własną nazwę w sieci, która może zostać przetłumaczona na czytelny dla komputerów adres IP, podobnie jak usługa DNS w Internecie. Usługi datagramowe i sesyjne to wtórne protokoły komunikacyjne, służące do transmitowania danych między NetBIOS-owymi komputerami.

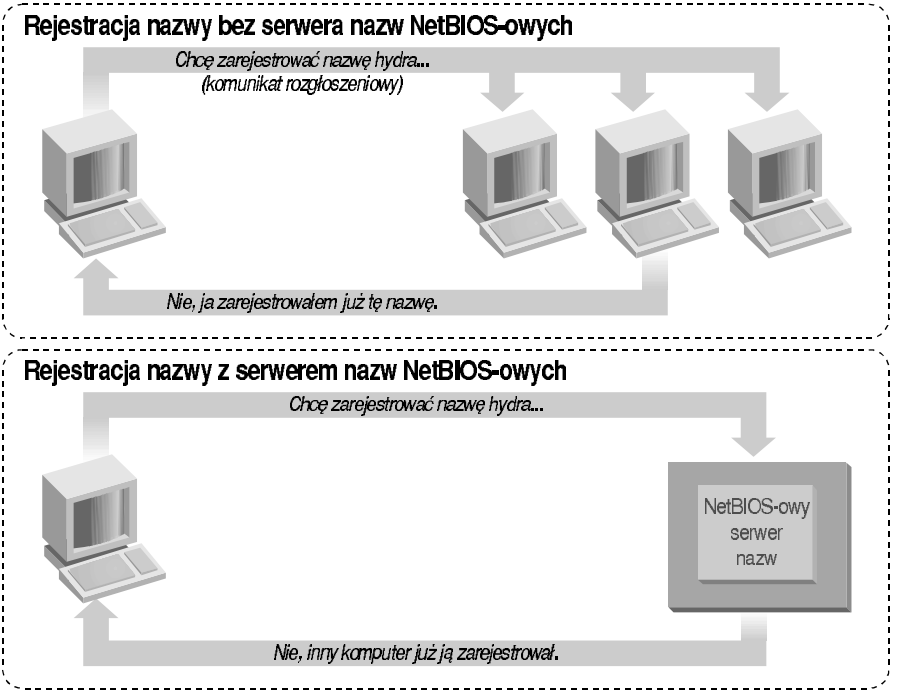
Uzyskiwanie nazwy

Nadanie imienia człowiekowi jest proste. Komputer w sieci NetBIOS ma z tym nieco więcej kłopotów. Przyjrzyjmy się kilku zagadnieniom.

W świecie NetBIOS-u każdy włączający się do sieci komputer próbuje określić własną nazwę; nazywamy to *rejestrowaniem nazwy*. Jednakże dwa komputery w tej samej sieci nie mogą zarejestrować takiej samej nazwy; myliłoby to pozostałe komputery, które chciałyby nawiązać komunikację z którymkolwiek z nich. Istnieją dwa sposoby rozwiązania tego problemu:

- Wykorzystanie serwera nazw NetBIOS-owych (*NetBIOS Name Server*, NBNS), który będzie zapamiętywał wszystkie nazwy zarejestrowane przez hosty.
- Pozwolenie każdemu komputerowi w sieci na obronę swojej nazwy, kiedy inny komputer spróbuje jej użyć.

Rysunek 1.8 przedstawia (nieudaną) rejestrację nazwy w obecności serwera NBNS i bez niego.



Rysunek 1.8. Rejestracja nazw z serwerem NBNS i bez niego

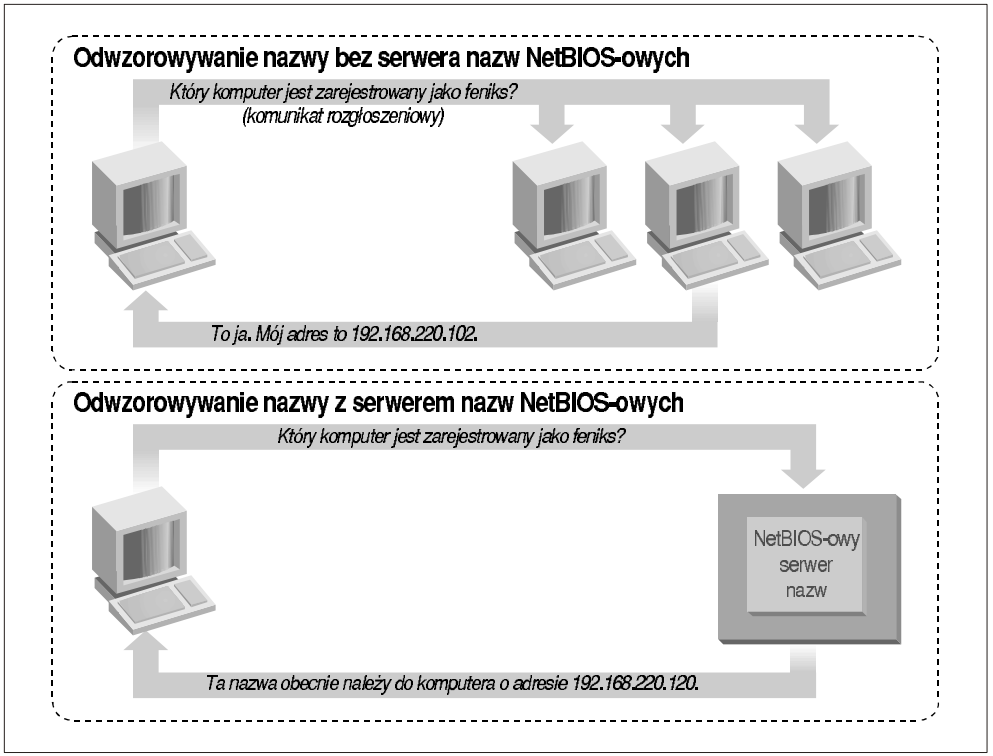
Oprócz tego musi istnieć metoda przetłumaczenia nazwy NetBIOS-owej na konkretny adres IP, czyli *odwzorowania nazwy*. Także i tutaj możliwe są dwa podejścia:

- Każdy komputer raportuje swój adres IP, kiedy „usłyszy" rozgłoszeniowe pytanie o swoją nazwę.
- Serwer NBNS tłumaczy nazwy NetBIOS-owe na adresy IP.

Rysunek 1.9 ilustruje te dwie metody odwzorowywania nazw.

Użycie serwera NBNS może przynieść spore korzyści. Najlepiej wyjaśnić to na przykładzie sieci bez takiego serwera.

W takim przypadku uruchomiony komputer rozgłasza komunikat, próbując zarejestrować pewną nazwę NetBIOS-ową jako swoją własną. Jeśli po kilku próbach rejestracji nikt nie zgłasza zastrzeżeń, komputer może zatrzymać nazwę. Jeśli jednak inny komputer w lokalnej podsieci używa już tej nazwy, wyśle on do próbującego się zarejestrować klienta komunikat, że nazwa jest już zajęta. Proces ten znany jest jako *obrona* nazwy hosta. Metoda ta przydaje się wtedy, kiedy jeden z klientów niespodziewanie odłączy się od sieci (wówczas inny host może bez problemów przejąć jego nazwę), ale wprowadza do sieci nadmierną ilość ruchu spowodowanego czymś tak banalnym jak rejestracja nazwy.



Rysunek 1.9. Odwzorowywanie nazw z serwerem NBNS i bez niego

W obecności serwera NBNS rzeczy mają się podobnie, ale komunikacja jest ograniczona do zgłaszającego żądanie klienta i serwera NBNS. Podczas występowania o zarejestrowanie nazwy nie są stosowane rozgłoszenia; komunikat jest wysyłany bezpośrednio od klienta do serwera, a ten udziela odpowiedzi, czy dana nazwa jest już zajęta. Taki model komunikacji nazywamy *dwupunktowym* – ma on niekwestionowane zalety w sieciach składających się z kilku podsieci, ponieważ rutery są często skonfigurowane tak, że blokują pakiety rozgłoszeniowe kierowane do wszystkich komputerów w podsieci.

Te same uwagi odnoszą się do odwzorowywania nazw. Bez serwera NBNS odwzorowywanie nazw również wymaga użycia mechanizmu rozgłoszeniowego. Każdy pakiet zapytania jest kierowany do wszystkich komputerów w sieci, w nadziei, że ten właściwy bezpośrednio odpowie pytającemu. Jak łatwo zauważyć, użycie serwera NBNS i komunikacji dwupunktowej znacznie zmniejsza obciążenie sieci w porównaniu z zalewaniem sieci pakietami rozgłoszeniowymi przy każdym zapytaniu o nazwę.

Typy węzłów

Jak ustalić strategię klientów sieciowych podczas rejestrowania i odwzorowywania nazw? Każdy komputer w sieci NBT można określić jednym z poniższych oznaczeń, w zależności od używanej przez niego metody rejestrowania i odwzorowywania

nazw: b-węzeł, p-węzeł, m-węzeł i h-węzeł. Zachowanie węzłów różnego typu opisuje tabela 1.1.

Tabela 1.1. Typy węzłów NetBIOS-owych

Rola	*Działanie*
b-węzeł	Używa tylko rozgłoszeniowego rejestrowania i odwzorowywania nazw
p-węzeł	Używa tylko dwupunktowego rejestrowania i odwzorowywania nazw
m-węzeł	Rejestruje nazwę rozgłoszeniowo. Jeśli rejestracja się powiedzie, informuje o tym serwer NBNS. Odwzorowuje nazwy rozgłoszeniowo. Używa serwera NBNS, jeśli rozgłoszenie się nie powiedzie
h-węzeł (hybrydowy)	Używa serwera NBNS do rejestrowania i odwzorowywania nazw. Używa rozgłoszeń, jeśli serwer NBNS nie odpowiada lub nie działa

Klienty Windows są zwykle skonfigurowane jako h-węzły (węzły hybrydowe). Tak na marginesie – h-węzły są późniejszym wynalazkiem Microsoftu i nie pojawiają się w dokumentach RFC 1001/1002.

Można ustalić typ węzła każdego komputera Windows, wpisując polecenie `ipconfig /all` i szukając linii zaczynającej się od słów `Node Type`.

```
C:\>ipconfig /all
Windows 98 IP Configuration
...
  Node Type . . . . . . . . . . . : Hybrid
...
```

Co się kryje w nazwie?

Nazwy używane przez NetBIOS różnią się znacznie od nazw hostów DNS, które być może są ci lepiej znane. Po pierwsze, nazwy NetBIOS-owe istnieją w płaskiej przestrzeni nazw. Innymi słowy, nie występują w nich kwalifikatory, takie jak *ora.com* lub *samba.org*, które hierarchizowałyby nazwy hostów; każdy komputer jest reprezentowany przez jedną, niepowtarzalną nazwę. Po drugie, nazwy NetBIOS-owe nie mogą być dłuższe niż 15 znaków, nie mogą zaczynać się od gwiazdki (*) i mogą składać się tylko ze standardowych znaków alfanumerycznych oraz następujących znaków:

```
! @ # $ % ^ & ( ) - ' { } . ~
```

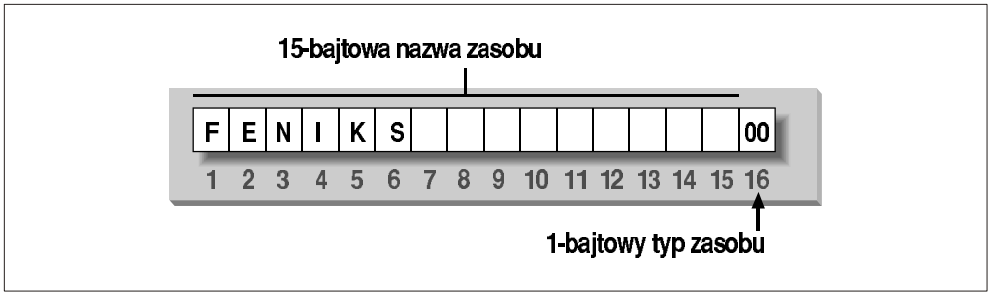
Choć w nazwach NetBIOS-owych można używać kropek (.), nie zalecamy tego, ponieważ mogą one nie działać w przyszłych wersjach NetBIOS-u ponad TCP/IP.

Poprawne nazwy DNS są zarazem poprawnymi nazwami NetBIOS-owymi i nie jest to przypadek. W rzeczywistości nazwa DNS serwera Samby bywa zwykle powielana jako jego nazwa NetBIOS-owa. Jeśli na przykład masz komputer o nazwie `feniks.ora.com`, jego nazwą NetBIOS-ową przypuszczalnie będzie FENIKS (plus dziewięć spacji).

Nazwy i typy zasobów

W NetBIOS-ie komputer nie tylko ogłasza swoją obecność, ale także oferowane przez siebie usługi. Na przykład komputer `feniks` może ogłosić, że nie jest zwykłą

stacją roboczą, ale że jest serwerem plików i może odbierać komunikaty WinPopup. Chcąc to uczynić, dodaje do nazwy komputera szesnasty bajt, nazywany *typem zasobu*, i rejestruje nazwę więcej niż jeden raz (patrz rysunek 1.10).



Rysunek 1.10. Struktura nazw NetBIOS-owych

Jednobajtowy typ zasobu oznacza usługę świadczoną przez komputer. W tej książce często spotkasz się z typem zasobu ujętym w nawiasy ostrokątne (<>) i umieszczonym za nazwą NetBIOS-ową, na przykład:

```
FENIKS<00>
```

Możesz sprawdzić, które nazwy są zarejestrowane przez konkretny komputer NBT, używając narzędzia Windows o nazwie NBTSTAT, uruchamianego z linii poleceń. Ponieważ usługi te są niepowtarzalne (żaden komputer nie może zarejestrować więcej niż jednej), w wynikach polecenia zostaną one oznaczone słowem UNIQUE. Poniższe skrócone wyniki polecenia opisują serwer `hydra`:

```
D:\>NBTSTAT -a hydra

       NetBIOS Remote Machine Name Table
   Name               Type         Status
---------------------------------------------
HYDRA          <00>  UNIQUE      Registered
HYDRA          <03>  UNIQUE      Registered
HYDRA          <20>  UNIQUE      Registered
...
```

Wyniki te informują, że serwer zarejestrował NetBIOS-ową nazwę `hydra` jako nazwę komputera (stacji roboczej), jako odbiorcę komunikatów WinPopup i jako serwer plików. Niektóre atrybuty, które może mieć nazwa, są wymienione w tabeli 1.2.

Tabela 1.2. Typy niepowtarzalnych zasobów NetBIOS-u

Nazwany zasób	*Szesnastkowa wartość bajtu*
Standardowa usługa stacji roboczej	00
Usługa posłańca (WinPopup)	03
Usługa serwera zdalnego dostępu	06
Usługa głównej przeglądarki domeny (związana z podstawowym kontrolerem domeny)	1B

Dokończenie tabeli na str. 14

Dokończenie tabeli ze str. 13

Tabela 1.2. Typy niepowtarzalnych zasobów NetBIOS-u

Nazwany zasób	*Szesnastkowa wartość bajtu*
Nazwa głównej przeglądarki	1D
Usługa NetDDE	1F
Serwer plików (i drukarek)	20
Usługa klienta zdalnego dostępu	21
Agent monitorowania sieci	BE
Narzędzie monitorowania sieci	BF

Zauważ, że ponieważ z nazwami DNS nie są związane typy zasobów, twórcy NetBIOS-u celowo ustalili, że szesnastkowa wartość 20 (spacja w kodzie ASCII) będzie odpowiadać typowi serwera plików.

Nazwy i typy grup

W protokole SMB istnieje pojęcie grup, w których komputery mogą się rejestrować. Wcześniej wspomnieliśmy, że komputery w naszej przykładowej sieci należą do *grupy roboczej*, czyli oddzielnego zbioru komputerów w tej samej sieci. Na przykład w firmie mogą istnieć grupy robocze KSIEGOWOSC i SPRZEDAZ, dysponujące odrębnymi serwerami i drukarkami. W świecie Windows grupa robocza i grupa SMB to jedno i to samo.

Wracając do przykładowego polecenia NBTSTAT, serwer Samby `hydra` jest także członkiem grupy roboczej PROSTA_GRUPA (atrybut GROUP równy 00 szesnastkowo) i może kandydować w wyborach na główną przeglądarkę (atrybut GROUP równy 1E szesnastkowo). Oto pozostała część wyników polecenia NBTSTAT:

```
      NetBIOS Remote Machine Name Table (kontynuacja)
   Name               Type        Status
---------------------------------------------
PROSTA_GRUPA    <00>   GROUP       Registered
PROSTA_GRUPA    <1E>   GROUP       Registered
..__MSBROWSE__.<01>   GROUP       Registered
```

Dostępne atrybuty grup są wymienione w tabeli 1.3. Więcej informacji można znaleźć w książce *Windows NT in a Nutshell* opublikowanej przez wydawnictwo O'Reilly.

Tabela 1.3. Typy grupowych zasobów NetBIOS-u

Nazwany zasób	*Szesnastkowa wartość bajtu*
Standardowa grupa stacji roboczych	00
Serwer logowania	1C
Nazwa głównej przeglądarki	1D
Zwykła nazwa grupy (używana podczas wyborów przeglądarki)	1E
Internetowa nazwa grupy (administracyjna)	20
`<01><02>__MSBROWSE__ <02>`	01

Ostatni wpis, __MSBROWSE__, służy do ogłaszania grupy innym głównym przeglądarkom. Niedrukowalne znaki wchodzące w skład nazwy są wyświetlane w wynikach polecenia NBTSTAT jako kropki. Nie przejmuj się, jeśli nie rozumiesz wszystkich typów zasobów i grup. Niektóre nie będą potrzebne do pracy z Sambą, a o innych będziemy jeszcze mówić w tym rozdziale. Warto jednak zapamiętać ogólne zasady działania mechanizmu nazewniczego.

Datagramy i sesje

W tym momencie odejdziemy nieco od tematu i omówimy inne zadanie NBT: zapewnianie usług komunikacyjnych między dwoma komputerami NetBIOS-owymi. NetBIOS ponad TCP/IP oferuje w rzeczywistości dwa typu usług: *usługi sesyjne* i *usługi datagramowe*. Zrozumienie ich działania nie jest konieczne do posługiwania się Sambą, ale da ci pewne pojęcie o pracy protokołu NBT i pomoże w rozwiązywaniu problemów, kiedy Samba odmówi posłuszeństwa.

Usługa datagramowa nie opiera się na stabilnych połączeniach między komputerami. Pakiety danych są po prostu wysyłane do drugiego komputera lub rozgłaszane w sieci, bez gwarancji, że dotrą do miejsca przeznaczenia we właściwej kolejności i że w ogóle tam dotrą. Korzystanie z datagramów obciąża sieć w mniejszym stopniu, niż otwieranie sesji, choć one także mogą spowolnić pracę sieci (czy pamiętasz rozgłoszeniowe odwzorowywanie nazw, o którym mówiliśmy wcześniej?). Datagramy służą zatem do szybkiego przesyłania prostych bloków danych do jednego lub wielu komputerów. Usługa datagramowa zapewnia komunikację za pomocą prostych symbolicznych funkcji wymienionych w tabeli 1.4.

Tabela 1.4. Symboliczne funkcje datagramowe

Funkcja	*Opis*
Wysłać datagram	Wysyła datagram do komputera lub grupy komputerów
Wysłać datagram rozgłoszeniowy	Wysyła datagram do wszystkich komputerów, które czekają na niego w następstwie wywołania funkcji „Odebrać datagram rozgłoszeniowy"
Odebrać datagram	Odbiera datagram od innego komputera
Odebrać datagram rozgłoszeniowy	Czeka na datagram rozgłoszeniowy

Usługi sesyjne są bardziej skomplikowane. Sesje są metodą komunikacji, która – przynajmniej w założeniu – umożliwia wykrycie wadliwych lub nie działających połączeń między dwoma aplikacjami NetBIOS-owymi. Sesję NBT można porównać do rozmowy telefonicznej*. Między komputerem wywołującym a wywoływanym otwierane jest pełnodupleksowe połączenie, które musi pozostać otwarte na czas trwania komunikacji. Obie strony wiedzą, który komputer nawiązał połączenie,

* Czytając dokument RFC 1001 łatwo zauważyć, że analogia „telefoniczna" miała spory udział w procesie opracowywania protokołu NBT.

a który je odebrał, i mogą porozumiewać się za pomocą prostych symbolicznych funkcji wymienionych w tabeli 1.5.

Tabela 1.5. Symboliczne funkcje sesyjne

Funkcja	*Opis*
Wywołać	Nawiązać łączność z nasłuchującym komputerem o określonej nazwie
Nasłuchiwać	Czekać na wywołanie przez określony lub dowolny komputer
Rozłączyć się	Zakończyć sesję
Wysłać	Wysłać dane do drugiego komputera
Odebrać	Odebrać dane od drugiego komputera
Stan sesji	Uzyskać informacje o określonych sesjach

Sesje to szkielet, na którym opiera się dzielenie zasobów w sieci NBT. Zwykle służą do nawiązania stabilnych połączeń między klientami a udziałami dyskowymi lub drukarkami serwera. Klient wywołuje serwer i zaczyna wymieniać z nim informacje, na przykład żądając otwarcia określonych plików, przesłania i odebrania danych i tak dalej. Połączenia takie mogą trwać dość długo (przez kilka godzin, a nawet dni), a wszystkie operacje są wykonywane w ramach jednego połączenia. Jeśli wystąpi błąd, oprogramowanie sesji (korzystające z TCP) będzie ponownie przesyłać dane aż do chwili, kiedy zostaną one odebrane poprawnie, inaczej niż ma to miejsce w usługach datagramowych (korzystających z UDP).

W praktyce sesje nie są tak niezawodne i nie zawsze radzą sobie z błędami w komunikacji. Wielu użytkowników sieci Windows z pewnością dostrzega tę poważną wadę sesji NBT. Jeśli z jakiegoś powodu połączenie zostanie przerwane, informacje o sesji otwartej między dwoma komputerami szybko tracą ważność. Jeśli tak się stanie, jedynym sposobem odtworzenia sesji jest ponowne wywołanie drugiego komputera i rozpoczęcie wszystkiego od nowa.

Jeśli jesteś zainteresowany dodatkowymi informacjami o tych usługach, polecamy lekturę dokumentu RFC 1001. Warto jednak zapamiętać dwie rzeczy:

- Sesje są otwierane zawsze między dwoma i tylko dwoma komputerami NetBIOS-owymi. Jeśli sesja zostanie przerwana, klient powinien zachować wystarczająco wiele informacji, aby możliwe było odtworzenie połączenia. W praktyce rzadko się to udaje.
- Datagramy mogą być rozgłaszane do wielu komputerów, ale są zawodne. Innymi słowy, komputer źródłowy nie ma pewności, że wysłane przez niego datagramy rzeczywiście dotarły do miejsca przeznaczenia.

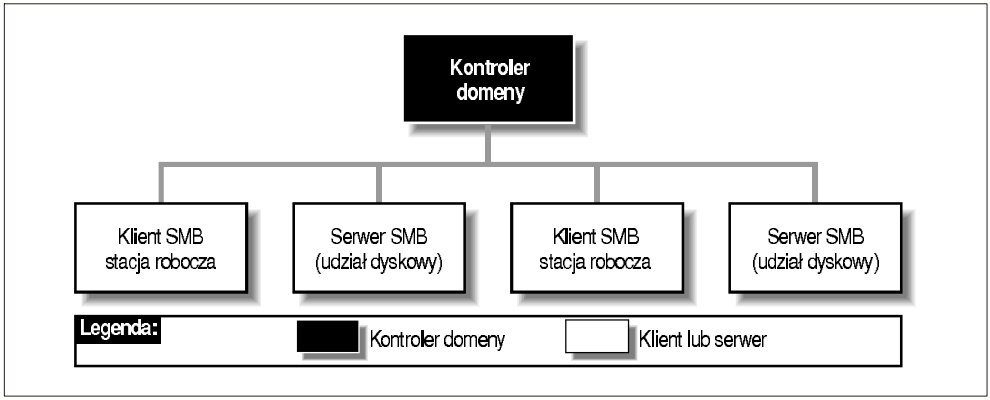
Implementacje Microsoftu

Uzbrojeni w nową wiedzę możemy przyjrzeć się, w jaki sposób Microsoft zrealizował powyższe koncepcje w sieciach CIFS/SMB. Jak łatwo było przewidzieć, trzeba będzie zapoznać się z pewnymi rozszerzeniami protokołu.

Domeny Windows

Przypomnijmy, że grupa robocza to zbiór komputerów SMB rezydujących w tej samej podsieci i zarejestrowanych w tej samej grupie SMB. *Domena Windows* to grupa robocza z jednym dodatkiem: serwerem pełniącym funkcję *kontrolera domeny*. Aby w sieci powstała domena Windows*, niezbędny jest kontroler domeny – w przeciwnym wypadku możemy mówić tylko o grupie roboczej (patrz rysunek 1.11).

Obecnie istnieją dwa odrębne protokoły używane przez kontroler domeny (serwer logowania): jeden do komunikacji z komputerami Windows 95/98, a drugi do komunikacji z komputerami Windows NT. Na dzień dzisiejszy Samba obsługuje protokół kontrolera domeny dla Windows 95/98 (dzięki czemu może działać jako kontroler domeny dla komputerów Windows 9*x*), ale nie wspiera jeszcze w pełni protokołu dla Windows NT. Zespół programistów Samby obiecuje jednak obsługę protokołu kontrolera domeny dla Windows NT w wersji 2.1 Samby.



Rysunek 1.11. Prosta domena Windows

W czym tkwi problem? Protokół używany przez kontrolery domeny Windows do komunikacji z klientami i innymi kontrolerami jest protokołem firmowym, a Microsoft nie opublikował jego specyfikacji. Zmusiło to zespół programistów Samby do wstecznego analizowania protokołu i ustalania, które kody realizują określone funkcje.

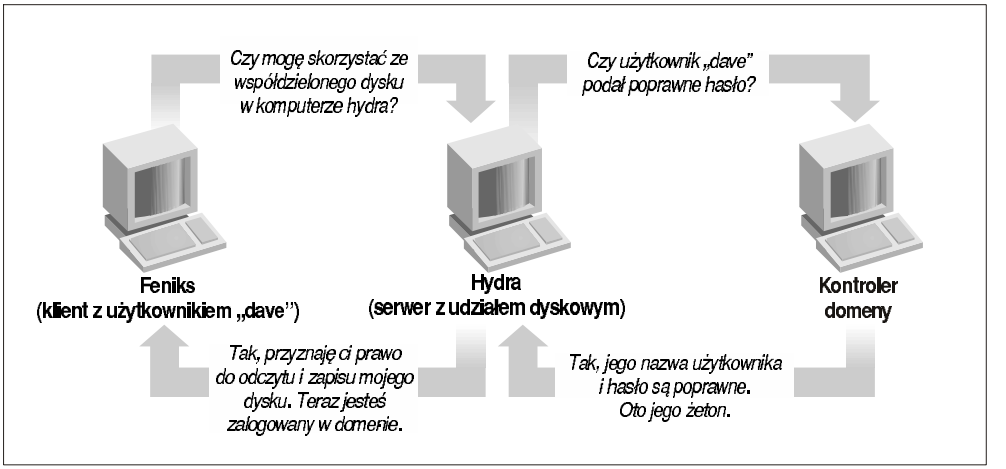
Kontrolery domeny

Kontroler to mózg domeny Windows, podobnie jak serwer NIS jest centralnym zarządcą uniksowych usług informacyjnych. Kontrolery domeny są odpowiedzialne za wiele różnych zadań, z których interesować nas będzie zwłaszcza *uwierzytelnianie*. Uwierzytelnianie to proces zezwalania użytkownikowi na dostęp (lub odmawiania dostępu) do współdzielonego zasobu, zwykle na podstawie hasła.

* Microsoft nazywa domeny Windows „domenami Windows NT", zakładając, że kontrolerem domeny musi być komputer z Windows NT. Ponieważ jednak funkcję tę może spełniać także Samba, będziemy je nazywać po prostu „domenami Windows", aby nie wprowadzać niepotrzebnego zamieszania.

Każdy kontroler domeny używa menedżera bezpieczeństwa (*security account manager*, SAM), który zarządza listą kombinacji „nazwa użytkownika – hasło". Kontroler domeny jest zatem centralną składnicą haseł związanych z nazwami użytkowników (jedno hasło na jednego użytkownika), co jest wydajniejsze, niż utrzymywanie setek haseł do wszystkich współdzielonych zasobów przez każdego klienta.

W domenie Windows, kiedy nieuwierzytelniony klient żąda dostępu do udziałów serwera, serwer pyta kontroler domeny, czy można zaufać temu klientowi. Jeśli tak, serwer nawiązuje połączenie, nadając użytkownikowi ściśle określone prawa dostępu do danej usługi. Jeśli nie, klientowi odmawia się połączenia. Gdy użytkownik zostanie uwierzytelniony przez kontroler domeny, klient otrzymuje specjalny żeton uwierzytelniający, aby użytkownik nie musiał się ponownie logować, kiedy zechce skorzystać z innych zasobów domeny. W tym momencie użytkownika uważa się za „zalogowanego" w samej domenie (patrz rysunek 1.12).



Rysunek 1.12. Uwierzytelnianie za pośrednictwem kontrolera domeny

Podstawowe i zapasowe kontrolery domeny

Nadmiarowość jest jednym z podstawowych założeń domeny Windows. Aktywny kontroler domeny jest nazywany *podstawowym kontrolerem domeny* (*Primary Domain Controller*, PDC). W sieci może być też jeden lub więcej zapasowych kontrolerów domeny (*Backup Domain Controller*, BDC), które przejmują zadania podstawowego kontrolera, gdy ten ulega awarii lub jest niedostępny. Kontrolery BDC regularnie synchronizują dane w swoich bazach SAM z kontrolerem PDC, aby w razie potrzeby podjąć świadczenie usług kontrolera domeny w sposób niezauważalny dla klientów. Zwróćmy jednak uwagę, że kopie bazy danych SAM w kontrolerach BDC są przeznaczone tylko do odczytu, a ich uaktualnienie jest możliwe tylko przez synchronizację z kontrolerem PDC. Serwer w domenie Windows może użyć bazy SAM dowolnego kontrolera podstawowego lub zapasowego w celu uwierzytelnienia użytkownika, który próbuje skorzystać z jego zasobów i zalogować się w domenie.

Zauważmy, że pod wieloma względami domeny i grupy robocze Windows są do siebie podobne. Podobieństwo to nie jest przypadkowe. Wynika z faktu, że mechanizm domen wprowadzono dopiero w Windows NT 3.5 i trzeba było zapewnić jego zgodność z wcześniejszymi grupami roboczymi Windows for Workgroups 3.1. Należy zapamiętać, że domena Windows to po prostu grupa robocza, do której dodano kontroler domeny.

Samba bez żadnych problemów może pełnić funkcję kontrolera domeny dla komputerów Windows 95/98. Jednakże działanie Samby 2.0 w charakterze podstawowego kontrolera domeny ogranicza się do uwierzytelniania użytkowników – nie może spełniać innych zadań (gdy ukaże się ta książka, być może będzie już dostępna wersja 2.1 Samby, która będzie mogła działać jako kontroler PDC dla klientów NT). Natomiast ze względu na zamkniętą naturę protokołu Microsoftu używanego do synchronizowania danych SAM, Samba nie może obecnie działać jako zapasowy kontroler domeny.

Przeglądanie

Przeglądanie to odpowiedź na pytanie użytkownika: „Jakie komputery są w mojej sieci Windows?". Warto zaznaczyć, że nie ma to nic wspólnego z przeglądarkami WWW, może oprócz ogólnej koncepcji odkrywania zasobów sieci. Podobnie jak w WWW, zasoby sieci Windows mogą się zmieniać bez żadnego ostrzeżenia.

Zanim wymyślono mechanizm przeglądania, użytkownicy musieli znać nazwę komputera, z którym chcieli połączyć się przez sieć i samodzielnie wpisywać w aplikacji lub menedżerze plików adresy UNC podobne do poniższego:

```
\\HYDRA\siec
```

Dzięki przeglądaniu można natomiast zbadać zasoby komputera za pomocą standardowego interfejsu graficznego typu „wskaż i kliknij" – ściślej, za pomocą okna Otoczenia sieciowego w kliencie Windows.

Poziomy przeglądania

Jak nadmieniliśmy na początku tego rozdziału, w sieci SMB/CIFS mamy do czynienia z dwoma typami przeglądania:

- przeglądanie listy komputerów (ze współdzielonymi zasobami),
- przeglądanie współdzielonych zasobów konkretnego komputera.

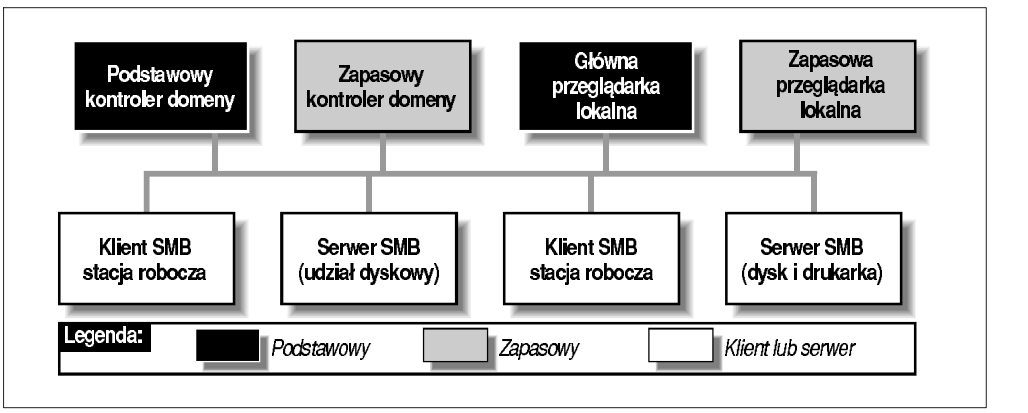
Przyjrzyjmy się pierwszemu typowi. W każdej podsieci grupy roboczej Windows (lub domeny) jeden z komputerów jest odpowiedzialny za utrzymywanie listy komputerów dostępnych w sieci. Ten komputer jest nazywany *główną przeglądarką lokalną*, a zarządzana przez niego lista – *listą przeglądania*. Komputery w danej podsieci używają listy przeglądania, aby zmniejszyć ruch sieciowy generowany przez operację przeglądania. Zamiast używać dynamicznych zapytań mających na celu ustalenie listy maszyn dostępnych w sieci, komputer może po prostu skontaktować się z główną przeglądarką lokalną i uzyskać kompletną, aktualną listę.

Aby przejrzeć zasoby w konkretnym komputerze, użytkownik musi się z nim połączyć; takich informacji nie ma na liście przeglądania. Listę zasobów komputera można przejrzeć, jeśli kliknie się jego ikonę w oknie Otoczenia sieciowego w Windows 95/98 lub NT. Na początku rozdziału pokazaliśmy, że komputer zwraca wówczas listę współdzielonych zasobów, do których użytkownik będzie mógł uzyskać dostęp, jeśli zostanie uwierzytelniony.

Każdy serwer w grupie roboczej Windows musi poinformować o swojej obecności główną przeglądarkę lokalną zaraz po zarejestrowaniu nazwy NetBIOS-owej, i (teoretycznie) ogłosić, że opuszcza grupę, kiedy jest wyłączany. Za rejestrowanie ogłoszeń serwerów odpowiedzialna jest główna przeglądarka lokalna. Zauważ, że lokalna przeglądarka to niekoniecznie ten sam komputer, co serwer nazw NetBIOS-owych (NBNS), o którym mówiliśmy poprzednio.

Otoczenie sieciowe Windows może zachowywać się dość dziwnie: dopóki nie wybierzesz komputera w celu przejrzenia jego zasobów, okno programu może zawierać nieaktualne dane. Oznacza to, że w oknie Otoczenia sieciowego mogą widnieć komputery, które uległy awarii, a może brakować tych, które jeszcze nie zostały zauważone. Mówiąc krótko, dopiero po wybraniu serwera i połączeniu się z nim możesz mieć pewność, że udziały dyskowe i drukarki rzeczywiście istnieją w sieci.

W przeciwieństwie do funkcji, o których mówiliśmy wcześniej, niemal każdy komputer Windows (NT Server, NT Workstation, 98, 95 lub Windows 3.1 for Workgroups) może działać jako główna przeglądarka lokalna. Podobnie jak kontroler domeny, główna przeglądarka lokalna może mieć w lokalnej podsieci jedną lub więcej *przeglądarek zapasowych*, które przejmą jej funkcje, gdyby uległa awarii lub stała się niedostępna. Aby wszystko działało płynnie, zapasowe przeglądarki lokalne często synchronizują swoje listy przeglądania z przeglądarką główną. Uaktualnijmy teraz nasz diagram domeny Windows tak, aby uwzględniał główną i zapasową przeglądarkę lokalną. Rezultat widać na rysunku 1.13.



Rysunek 1.13. Domena Windows z główną i zapasową przeglądarką lokalną

Oto, jak można obliczyć minimalną liczbę zapasowych przeglądarek działających w grupie roboczej:

- Jeśli w sieci jest od 1 do 32 stacji roboczych Windows NT albo od 1 do 16 komputerów Windows 95/98, główna przeglądarka lokalna wyznacza jedną przeglądarkę zapasową.
- Jeśli liczba stacji roboczych Windows NT wynosi od 33 do 64 albo liczba stacji roboczych Windows 95/98 wynosi od 17 do 32, główna przeglądarka lokalna wyznacza dwie przeglądarki zapasowe.
- Dla każdej następnej grupy 32 stacji roboczych Windows NT lub 16 komputerów Windows 95/98 główna przeglądarka lokalna wyznacza następną przeglądarkę zapasową.

Obecnie nie ma górnego limitu przeglądarek zapasowych, które mogą zostać wyznaczone przez przeglądarkę główną.

Wybory przeglądarek

Przeglądanie to bardzo istotny aspekt każdej grupy roboczej Windows. Niestety, żadna sieć nie działa idealnie. Przypuśćmy, że serwer Windows NT na biurku dyrektora niewielkiej firmy jest główną przeglądarką lokalną – dopóki dyrektor nie wyłączy go z gniazdka, żeby podłączyć swój aparat do masażu. W tym momencie stacja robocza Windows NT w dziale części zamiennych mogłyby przejąć jego zadanie. Komputer ten wykonuje jednak duży, kiepsko napisany program, który pochłania cały czas procesora. Morał: przeglądanie musi być bardzo odporne na włączanie i wyłączanie serwerów. Ponieważ niemal każdy komputer Windows może działać jako przeglądarka, musi istnieć sposób ustalenia, który przyjmie na siebie tę odpowiedzialność. Proces podejmowania tej decyzji nosi nazwę *wyborów*.

Algorytm wyborów jest wbudowany w niemal wszystkie systemy operacyjne Windows, dzięki czemu mogą one uzgodnić, który komputer stanie się przeglądarką główną, a który zapasową. Wybory mogą zostać wymuszone w dowolnej chwili. Załóżmy, że dyrektor skończył masaż i ponownie uruchomił serwer. Kiedy serwer włączy się do sieci, ogłosi swoją obecność i zostaną przeprowadzone wybory, które ustalą, czy komputer PC w dziale części zamiennych powinien nadal być przeglądarką główną.

W czasie trwania wyborów wszystkie komputery rozgłaszają następujące informacje:

- wersja używanego protokołu elekcyjnego,
- system operacyjny komputera,
- jak długo komputer jest podłączony do sieci,
- nazwa hosta.

Wartości te pozwalają ustalić, który system będzie pełnił funkcję głównej przeglądarki lokalnej (proces wyborów opisano bardziej szczegółowo w rozdziale 5, *Przeglądanie i zaawansowane udziały dyskowe*. Algorytmy odpowiedzialne za ten proces są jednak mało eleganckie i stanowią zagrożenie bezpieczeństwa sieci. Choć przeglądanie domeny można zintegrować z domenowymi funkcjami bezpieczeństwa, al-

gorytm elekcyjny nie określa jasno, który komputer zostanie przeglądarką. Tak więc dowolny komputer z usługą przeglądania może zgłosić swój udział w wyborach i (po ich wygraniu) dowolnie zmieniać listę przeglądania. Niemniej jednak przeglądanie jest kluczowym aspektem sieci Windows, a konieczność zapewnienia wstecznej zgodności wróży mu długi żywot.

Czy grupa robocza Windows może obejmować wiele podsieci?

Tak, ale administratorzy, którzy się na to decydują, powinni mieć pod ręką tabletki od bólu głowy. Rozciąganie grupy roboczej na kilka podsieci nie było brane pod uwagę podczas opracowywania Windows NT 3.5 i Windows for Workgroups. W rezultacie domena Windows obejmująca kilka podsieci to „sklejenie" kilku grup roboczych o identycznych nazwach. Co prawda nadal można używać podstawowego kontrolera domeny do uwierzytelniania klientów we wszystkich podsieciach, ale przeglądanie staje się bardziej skomplikowane.

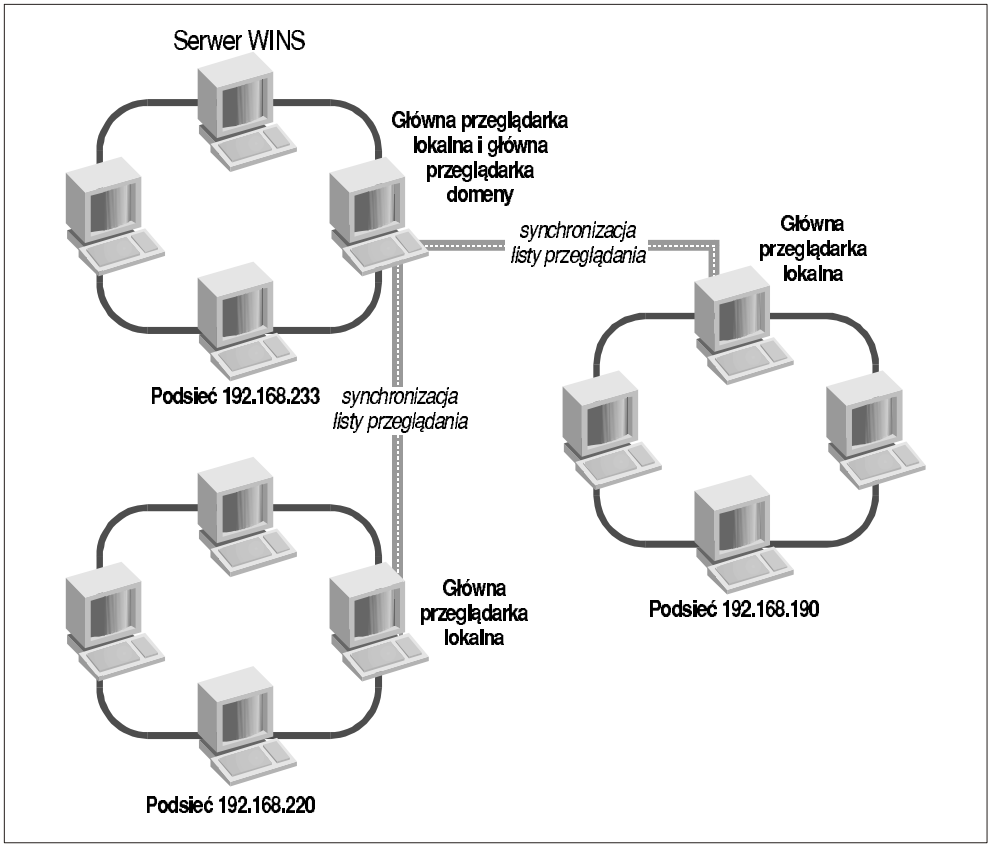
Jak już wspomniano, każda podsieć musi mieć własną główną przeglądarkę lokalną. Jeśli domena Windows rozciąga się na kilka podsieci, administrator musi wyznaczyć jeden z komputerów na *główną przeglądarkę domeny*. Główna przeglądarka domeny będzie przechowywać listę przeglądania dla całej domeny Windows. Lista ta jest tworzona dzięki regularnej synchronizacji list przeglądania wszystkich głównych przeglądarek lokalnych z listą przeglądania głównej przeglądarki domeny. Po synchronizacji główna przeglądarka lokalna i główna przeglądarka domeny powinny dysponować identycznymi listami (patrz rysunek 1.14).

Brzmi nieźle? Cóż, rozwiązanie to nie jest doskonałe, z następujących przyczyn:

- Podstawowy kontroler domeny, jeśli istnieje, zawsze pełni funkcję głównej przeglądarki domeny. Microsoft zdecydował, że obie usługi będą dzielić ten sam typ zasobu NetBIOS-u <1B>, więc (niestety) nie można ich rozdzielić.
- Komputery Windows 95/98 nie mogą zostać głównymi przeglądarkami domeny *ani nawet się z nimi kontaktować*. Zespół programistów Samby uważa, że jest to marketingowa decyzja Microsoftu, zmuszająca klientów do kupowania przynajmniej jednej kopii Windows NT Workstation dla każdej podsieci (na szczęście z pomocą przychodzi tu Samba).

Główna przeglądarka lokalna każdej podsieci nadal utrzymuje listę przeglądania dla tej podsieci i sprawuje nad nią kontrolę. Jeśli więc komputer chce uzyskać listę serwerów we własnej podsieci, skieruje zapytanie do głównej przeglądarki lokalnej tej podsieci. Jeśli chce uzyskać listę serwerów spoza podsieci, też musi skontaktować się z przeglądarką lokalną. Jest to możliwe, ponieważ w regularnych odstępach czasu autorytatywna lista przeglądania głównej przeglądarki lokalnej jest synchronizowana z listą głównej przeglądarki domeny, która z kolei jest zsynchronizowana z przeglądarkami lokalnymi pozostałych podsieci. Proces ten nosi nazwę *rozpowszechniania list przeglądania*.

Samba w razie potrzeby może pełnić funkcję głównej przeglądarki domeny Windows. Oprócz tego może działać jako główna przeglądarka lokalna podsieci Windows, synchronizując swoją listę przeglądania z główną przeglądarką domeny.



Rysunek 1.14. Grupa robocza rozciągająca się na kilka podsieci

Windows Internet Name Service (WINS)

Windows Internet Name Service to implementacja serwera nazw NetBIOS-owych (*NetBIOS Name Server*, NBNS) i jako taka dziedziczy wiele charakterystyk NetBIOS-u. Po pierwsze, usługa nazewnicza WINS jest płaska; konieczne jest stosowanie nazw komputerów typu `franek` albo nazw grup roboczych typu KANADA lub USA. Po drugie, WINS jest usługą dynamiczną: kiedy klient włącza się do sieci, musi zgłosić swoją nazwę hosta, adres i grupę roboczą lokalnemu serwerowi WINS. Serwer zachowuje te informacje tak długo, jak długo klient okresowo odnawia swoją rejestrację, co oznacza, że jest nadal podłączony do sieci. Warto zauważyć, że serwery WINS nie są związane z grupą roboczą ani domeną; mogą znajdować się w dowolnym miejscu sieci i obsługiwać wszystkie komputery.

Można skonfigurować kilka serwerów WINS tak, aby synchronizowały swoje dane w określonych odstępach czasu. Dzięki temu informacje o komputerach włączających się do sieci i odłączających się mogą być rozpowszechniane wśród wszystkich serwerów WINS. Choć teoretycznie rozwiązanie takie wydaje się efektywne, to

szybko staje się kłopotliwe, jeśli sieć obsługuje kilka serwerów WINS. Ponieważ serwery WINS mogą świadczyć usługi dla kilku podsieci (trzeba albo ręcznie skonfigurować adres serwera WINS w każdym kliencie, albo dostarczać go za pośrednictwem DHCP), często wydajniejszym rozwiązaniem jest skierowanie wszystkich klientów do tego samego serwera WINS, niezależnie od liczby domen Windows w sieci. Dzięki temu będziemy dysponować jednym autorytatywnym serwerem WINS zamiast zmagać się z kilkoma serwerami, bezustannie usiłującymi zsynchronizować często zmieniające się informacje.

Aktywny serwer WINS nazywany jest *podstawowym serwerem WINS*. Można zainstalować także zapasowy serwer WINS, który w razie awarii lub niedostępności serwera podstawowego przejmie jego funkcję. Komputery nie wybierają spomiędzy siebie podstawowego i zapasowego serwera WINS – decyzja należy tu do administratora systemu. Podstawowy i zapasowy serwer WINS synchronizują swoje informacje w regularnych odstępach czasu.

W rodzinie systemów operacyjnych Windows serwerem WINS może być tylko NT Workstation lub NT Server. Samba może działać jako podstawowy, ale nie jako zapasowy serwer WINS.

Co może Samba?

Uff! Pewnie nigdy nie sądziłeś, że sieci Microsoftu są tak skomplikowane. Teraz podsumujemy wszystkie informacje i przypomnimy, do czego można wykorzystać Sambę. W tabeli 1.6 wymienione są funkcje, które Samba może i których nie może pełnić w domenie lub grupie roboczej Windows. Ponieważ wiele protokołów domeny NT to firmowe, nieudokumentowane opracowania Microsoftu, Samba nie potrafi poprawnie synchronizować danych z serwerami Microsoftu i w większości przypadków nie może działać jako serwer zapasowy. Od wersji 2.0.*x* Samba potrafi jednak w ograniczonym stopniu obsługiwać protokoły uwierzytelniające podstawowego kontrolera domeny, a jej funkcjonalność zwiększa się z każdym dniem.

Tabela 1.6. Funkcje Samby (w wersji 2.0.4b)

Funkcja	*Potrafi pełnić?*
Serwer plików	Tak
Serwer wydruku	Tak
Podstawowy kontroler domeny	Tak (zaleca się wersję 2.1 lub nowszą)
Zapasowy kontroler domeny	Nie
Uwierzytelnianie klientów Windows 95/98	Tak
Główna przeglądarka lokalna	Tak
Zapasowa przeglądarka lokalna	Nie
Główna przeglądarka domeny	Tak
Podstawowy serwer WINS	Tak
Zapasowy serwer WINS	Nie

Przegląd dystrybucji Samby

Jak już wspomniano, Samba to w rzeczywistości kilka programów spełniających różne, ale komplementarne funkcje. Omówimy krótko każdy z nich i wyjaśnimy, jak ze sobą współpracują. Większość programów wchodzących w skład dystrybucji Samby koncentruje się na jej dwóch demonach. Oto podstawowe zadania demonów Samby:

smbd
: Demon *smbd* jest odpowiedzialny za zarządzanie zasobami współdzielonymi przez serwer Samby i jego klientów. Zapewnia on klientom SMB (w jednej lub kilku sieciach) usługi dostępu do plików i drukarek oraz usługi przeglądania. Demon *smbd* przekazuje wszystkie powiadomienia między serwerem Samby a klientami sieciowymi. Oprócz tego odpowiada za uwierzytelnianie użytkowników, blokowanie zasobów i współdzielenie danych przez protokół SMB.

nmbd
: Demon *nmbd* to prosty serwer nazw, naśladujący działanie WINS i serwera nazw NetBIOS-owych w stylu pakietu LAN Manager. Demon ten oczekuje na żądania klientów i dostarcza im odpowiednich informacji. Obsługuje także listy przeglądania na potrzeby Otoczenia sieciowego i bierze udział w wyborach przeglądarek.

Dystrybucja Samby zawiera także niewielki zbiór narzędzi uruchamianych z uniksowej linii poleceń:

smbclient
: Uniksowy klient, przypominający klienta FTP, który umożliwia połączenie z udziałami Samby.

smbtar
: Program służący do sporządzania zapasowych kopii udziałów, podobny do uniksowego polecenia *tar*.

nmblookup
: Program umożliwiający sprawdzanie nazw NetBIOS-owych.

smbpasswd
: Program umożliwiający administratorowi zmianę zaszyfrowanych haseł używanych przez Sambę.

smbstatus
: Program wyświetlający bieżące połączenia sieciowe z udziałami w serwerze Samby.

testparm
: Prosty program sprawdzający poprawność pliku konfiguracyjnego Samby.

testprns
: Program umożliwiający sprawdzenie, czy różne drukarki są rozpoznawane przez demona *smbd*.

Każde znaczące wydanie Samby przechodzi przez szczegółowe testy, zanim zostanie oficjalnie udostępnione. Jest też szybko uaktualniane w razie odkrycia problemów lub efektów ubocznych. Najnowszą stabilną dystrybucją w czasie pisania tej książki była Samba 2.0.5, długo oczekiwana wersja produkcyjna Samby 2.0. Niniejsza książka skupia się na funkcjach oferowanych przez Sambę 2.0, a nie przez wersje 1.9.*x*, które obecnie są przestarzałe.

Skąd wziąć Sambę?

Samba jest dostępna zarówno w postaci binarnej, jak i źródłowej w wielu witrynach internetowych. Główna witryna Samby znajduje się pod adresem *http://www.samba.org*. Jeśli jednak nie chcesz czekać na pakiety podróżujące aż z Australii, możesz skorzystać z kilku witryn bliźniaczych, których listę można znaleźć na stronie głównej Samby.

Jeśli wcześniej nie używałeś Samby, zachęcamy do skorzystania z CD-ROM-u dołączonego do tej książki. Zamieściliśmy na nim programy binarne i kody źródłowe aż do wersji 2.0.5 Samby, a oprócz tego kilka wygodnie spakowanych narzędzi diagnostycznych, o których jest mowa w książce.

Co nowego w Sambie 2.0?

Samba 2.0 była bardzo oczekiwanym pakietem. Najważniejsze dodatki w wersji 2.0 to lepsza obsługa domen NT oraz nowe narzędzie administracyjne – Samba Web Administration Tool (SWAT), które umożliwia konfigurowanie Samby za pomocą przeglądarki WWW. Oprócz tego pakiet zawiera wiele drobnych usprawnień, wprowadzonych w drugiej połowie 1998 roku.

Domeny NT

Obsługa domen NT w Sambie (od wersji 2.0.*x*) jest istotnym ulepszeniem: pozwala serwerom SMB na używanie mechanizmów uwierzytelniania, co jest niezbędne do zapewnienia przyszłej zgodności z Windows NT, oraz na stosowanie *logowań domenowych NT*. Dzięki temu użytkownik może zalogować się w domenie Windows NT i używać wszystkich komputerów bez indywidualnego logowania się na każdym z nich. We wcześniejszych wersjach Samba obsługiwała usługi logowania Windows 95/98, ale nie logowania domenowe NT. Obsługa logowań domenowych w wersji 2.0 Samby jest już zaimplementowana, choć jeszcze niekompletna.

Łatwość administrowania

SWAT, czyli Samba Web Based Administration Tool, ułatwia ustawianie serwera i zmienianie jego konfiguracji, bez zmuszania użytkownika do rezygnacji z prostego, tekstowego pliku konfiguracyjnego. SWAT to interfejs graficzny umożliwiający konfigurowanie zasobów, które Samba dzieli z klientami. SWAT oszczędza administratorowi eksperymentowania i pracy pamięciowej podczas ustalania i zmieniania

konfiguracji sieci. Można nawet utworzyć początkową konfigurację za pomocą programu SWAT, a następnie zmodyfikować plik ręcznie lub *vice versa*. Samba nie będzie się skarżyć.

Jeśli chodzi o kompilację, obecnie stosowany jest program GNU *autoconf*, który ułatwia wstępne kompilowanie i konfigurowanie programu, dzięki czemu szybciej można skorzystać z pomocy oferowanej przez program SWAT.

Wydajność

Znacznie wzrosła wydajność i skalowalność Samby. Kod programu został zreorganizowany, a *nmbd* (demon usług nazewniczych Samby) – w dużej mierze przepisany:

- Usługi nazewnicze i przeglądania mogą teraz jednocześnie obsługiwać nawet 35 000 klientów.
- Usługi plikowe i drukowania mogą obsługiwać 500 użytkowników jednocześnie (działając w serwerze o średniej wydajności) bez zauważalnego pogorszenia sprawności.
- Linux z Sambą na identycznym sprzęcie działa teraz wydajniej od serwera NT, a co najlepsze, Samba wciąż jest udoskonalana.
- Ulepszone „oportunistyczne" blokady pozwalają klientom na lokalne buforowanie plików, co znacznie przyspiesza operacje dyskowe bez ryzyka przypadkowego nadpisania buforowanych plików.

Inne cechy

Samba 2.0 ma kilka dodatkowych funkcji. Obecnie można skonfigurować kilka aliasów dla Samby na jednym komputerze, z których każdy udaje osobny serwer – ta funkcja przypomina wirtualne hosty znane z nowoczesnych serwerów WWW. Dzięki temu serwer może obsługiwać kilka wydziałów firmy lub grup albo udostępniać udziały plikowe ze zwykłymi zabezpieczeniami przez nazwę użytkownika i hasło, natomiast drukarki bez żadnych zabezpieczeń. Zmieniono obsługę drukowania, aby ułatwić życie użytkownikom Uniksów typu System V: Samba może teraz automatycznie odszukać dostępne drukarki, podobnie jak czyni to w Uniksach typu Berkeley. Oprócz tego Samba może teraz używać wielu stron kodowych, dzięki czemu obsługuje języki nieeuropejskie, oraz protokołu Secure Sockets Layer (SSL) do szyfrowania wszystkich przesyłanych danych, a nie tylko haseł*.

Lepsza kompatybilność

Wraz ze wzrostem funkcjonalności Samba staje się także coraz bardziej zgodna z Windows NT. Samba od początku obsługiwała szyfrowanie haseł metodą Microsoftu. Obecnie zawiera narzędzia i opcje umożliwiające przejście na szyfrowanie

* W Stanach Zjednoczonych istnieją pewne prawa i przepisy federalne dotyczące zaawansowanej kryptografii. Omówimy je w dodatku A, *Konfigurowanie Samby do obsługi SSL*.

Microsoftu z synchronizacją plików haseł Uniksa i Microsoftu. Wreszcie, Samba jako główna przeglądarka może teraz wyszukiwać serwery SMB w innych sieciach lokalnych i synchronizować się z nimi, dzięki czemu SMB działa bez problemów w środowisku wielosieciowym. Samba używa w tym celu innej metody niż nieudokumentowany protokół Microsoftu.

Smbwrapper

Istnieje też zupełnie nowa wersja uniksowego klienta o nazwie *smbwrapper*. Zamiast modułu jądra, dzięki któremu Linux może działać jako klient Samby, obecnie można użyć polecenia ładującego bibliotekę, która udostępnia pełny system plików SMB w niektórych odmianach Uniksa. Po załadowaniu tej biblioteki polecenie `ls /smb` wyświetli listę wszystkich komputerów w grupie roboczej, a polecenie `cd /smb/nazwa_serwera/nazwa_udziału` przejdzie do określonego udziału (współdzielonego katalogu), co przypomina działaniem Network File System (NFS). Kiedy pisaliśmy tę książkę, program *smbwrapper* działał w Linuksie, Solarisie, SunOS 4, IRIX-ie i OSF/1, a w niedalekiej przyszłości miał pracować także w kilku innych systemach operacyjnych.

To nie wszystko...

Samba jest doskonałym narzędziem, wartym zastosowania nawet w najmniejszej sieci SMB/CIFS. Ten rozdział szczegółowo pokazał czym jest Samba, a co ważniejsze – jakie jest jej miejsce w sieci Windows. W następnych rozdziałach zajmiemy się zarówno ustawianiem Samby po stronie uniksowego serwera, w którym rezydują jej dwa demony, jak i konfigurowaniem klientów Windows 95, 98 i NT do współpracy z Sambą. Już niedługo wszelkie problemy z heterogeniczną siecią odejdą w przeszłość. Witaj we wspaniałym świecie Samby!

2

Instalowanie Samby w Uniksie

Kiedy wiesz już, co Samba ma do zaoferowania tobie i użytkownikom, możesz przystąpić do skonfigurowania swojej sieci. Zacznijmy od zainstalowania Samby w Uniksie. Żeby zatańczyć sambę, trzeba uczyć się jej krok po kroku. To samo odnosi się do instalowania Samby, a ten rozdział pomoże ci zacząć od właściwej nogi.

W celach demonstracyjnych będziemy instalować wersję 2.0.4 serwera Samby w Linuksie* z jądrem w wersji 2.0.31, ale poszczególne etapy instalacji są takie same na wszystkich platformach, na których może działać Samba. Typowa instalacja zajmuje około godziny, wliczając w to pobieranie plików źródłowych, kompilowanie ich, tworzenie plików konfiguracyjnych i testowanie serwera.

Oto przegląd etapów instalacji:

1. Pobieranie plików źródłowych lub binarnych.
2. Czytanie dokumentacji instalacyjnej.
3. Konfigurowanie pliku *makefile*.
4. Kompilowanie kodu serwera.
5. Instalowanie plików serwera.
6. Tworzenie pliku konfiguracyjnego Samby.
7. Testowanie pliku konfiguracyjnego.
8. Uruchamianie demonów Samby.
9. Testowanie demonów Samby.

Pobieranie dystrybucji Samby

Jeśli chcesz szybko przystąpić do rzeczy, na dołączonym CD-ROM-ie znajdują się źródła i binaria Samby, które były dostępne w momencie oddawania książki do druku. CD-ROM to dokładny obraz plików i katalogów serwera dystrybucyjnego Samby: *ftp.samba.org*.

* Jeśli jeszcze nie słyszałeś o Linuksie, czeka cię miła niespodzianka. Linux to rozpowszechniany bezpłatnie, uniksopodobny system operacyjny, działający na platformach x86 Intela, PowerPC Motoroli i Sparc Suna. Linux jest w miarę łatwy w konfiguracji, naprawdę niezawodny i zdobywa coraz większą popularność. Więcej informacji o Linuksie znajdziesz pod adresem *http://www.linux.org/*.

Jeśli jednak wolałbyś pobrać najnowszą wersję pakietu, zajrzyj na stronę główną Samby pod adresem *http://www.samba.org*. Po otworzeniu tej strony zobaczysz łącza do kilku witryn bliźniaczych na całym świecie – zarówno standardowych stron Samby, jak i serwisów poświęconych wyłącznie pobieraniu pakietu. Aby pobieranie przebiegło sprawnie, wybierz serwis położony najbliżej ciebie.

Na standardowych stronach WWW Samby znajduje się dokumentacja i samouczki, archiwa list wysyłkowych, najnowsze informacje o Sambie oraz źródłowe i binarne dystrybucje pakietu. Serwisy poświęcone pobieraniu Samby (nazywane *serwisami FTP*) zawierają tylko dystrybucje binarne i źródłowe. Jeśli nie masz specjalnych powodów do pobierania starszej wersji pakietu i nie zamierzasz instalować dystrybucji binarnej, pobierz najnowszą dystrybucję źródłową z najbliższej witryny bliźniaczej. Dystrybucja ta zawsze nosi nazwę:

```
samba-latest.tar.gz
```

Jeśli zechcesz użyć wersji Samby umieszczonej na CD-ROM-ie dołączonym do tej książki, najnowszą wersję pakietu znajdziesz w katalogu głównym.

Pakiet binarny czy źródłowy?

Dla wielu platform uniksowych dostępne są także prekompilowane pakiety. Pakiety te zawierają wszystkie pliki wykonywalne Samby w postaci binarnej oraz standardową dokumentację programów. Choć zainstalowanie dystrybucji binarnej może oszczędzić ci wielu problemów i sporo czasu, podejmując decyzję powinieneś mieć na uwadze następujące kwestie:

- Pakiety binarne mogą pozostawać w tyle za najnowszymi wersjami oprogramowania o jedno lub dwa (czasem więcej) pomniejsze wydania, zwłaszcza po serii drobnych zmian i dla mniej popularnych platform. Porównaj noty wydawnicze w pakiecie źródłowym i binarnym, aby upewnić się, czy nie pojawiły się nowe funkcje, które mogłyby się okazać przydatne na twojej platformie. Dotyczy to zwłaszcza pakietów źródłowych i binarnych na CD-ROM-ie: kiedy książkę oddawano do druku, oba pochodziły z najnowszego wydania produkcyjnego Samby. Rozwój jednak wciąż trwa, więc w Internecie będzie można znaleźć nowsze wersje beta.
- Jeśli używasz prekompilowanych programów, będziesz musiał sprawdzić, czy masz właściwe wersje bibliotek wymaganych przez pliki wykonywalne. Dla niektórych platform pliki te są konsolidowane statycznie, więc nie stanowi to problemu, ale w nowoczesnych odmianach Uniksa (jak na przykład Linux, SGI Irix, Solaris, HP-UX) biblioteki są często konsolidowane dynamicznie. Oznacza to, że plik binarny szuka w systemie odpowiedniej wersji biblioteki, którą być może będziesz musiał zainstalować. Plik *README* lub *makefile* towarzyszący wersji binarnej powinien wymieniać tego rodzaju wymagania*.

* Dotyczy to zwłaszcza programów używających biblioteki *glibc-2.1* (standardowej w Linuksie Red Hat 6.0). Biblioteka ta wywołała sporą konsternację w środowisku programistów, kiedy po jej wydaniu okazało się, że jest niezgodna z poprzednimi wersjami *glibc*.

Wiele systemów ze współdzielonymi bibliotekami jest wyposażonych w przydatne narzędzie o nazwie *ldd*. Program ten informuje, jakich bibliotek wymaga dany plik binarny i które biblioteki zainstalowane w systemie spełniają te wymagania. Na przykład sprawdzenie programu *smbd* na testowym komputerze dało następujące wyniki:

```
$ ldd smbd
libreadline.so.3 => /usr/lib/libreadline.so.3
libdl.so.2 => /lib/libdl.so.2
libcrypt.so.1 => /lib/libcrypt.so.1
libc.so.6 => /lib/libc.so.6
libtermcap.so.2 => /lib/libtermcap.so.2
/lib/ld-linux.so.2 => /lib/ld-linux.so.2
```

Jeśli istnieją niezgodności między Sambą i pewnymi bibliotekami w twoim systemie, powinna je wyjaśnić dokumentacja dołączona do dystrybucji systemu.

- Pamiętaj, że każda dystrybucja binarna przyjmuje pewne założenia co do docelowej platformy, dotyczące na przykład domyślnych katalogów i wartości opcji konfiguracyjnych. Warto zajrzeć więc do dokumentacji i pliku *makefile* w katalogu źródłowym, aby sprawdzić, jakich dyrektyw i zmiennych użyto podczas kompilacji. W niektórych przypadkach wartości te będą nieadekwatne do twojej konfiguracji systemu.

  Niektóre parametry konfiguracyjne można ustalić za pomocą opcji linii polecenia w czasie wykonywania, a nie kompilacji programu. Jeśli na przykład twój plik binarny próbuje umieścić pliki dziennika, blokady i statusu w „złym" miejscu (na przykład w katalogu */usr/local*), możesz to zmienić bez rekompilowania programu.

Warto tu wspomnieć, że źródła Samby wymagają kompilatora ANSI C. Jeśli używasz systemu, którego kompilator jest niezgodny z tym standardem (jak na przykład kompilator *cc* z Sun OS 4), będziesz musiał najpierw zainstalować kompilator zgodny z ANSI, na przykład *gcc**. Jeśli nie pociąga cię perspektywa instalowania kompilatora, możesz zacząć od pakietu binarnego. Jednak dla zapewniania największej elastyczności i zgodności ze swoim systemem powinieneś zawsze skompilować najnowsze pliki źródłowe.

Przeczytaj dokumentację

Wydaje się to dość oczywistą radą, ale prawdopodobnie nie raz zdarzyło ci się zdekompresować pakiet, wpisać na chybcika `configure`, `make` oraz `make install` i udać się po kolejną filiżankę kawy. Przyznajemy, że sami tak robimy, dużo częściej niż powinniśmy. To jednak kiepski pomysł – zwłaszcza wtedy, gdy planujesz oprzeć na Sambie swoją sieć.

Samba 2.0 sama konfiguruje się przed instalacją. Zmniejsza to ryzyko wystąpienia problemów zależnych od konfiguracji komputera, ale w pliku *README* może być wspomniana jakaś opcja, której będzie ci brakować po zainstalowaniu Samby. Zarówno w pakiecie binarnym, jak i źródłowym znajdziesz w katalogu *docs* wiele dokumentów w różnych formatach. Dwa najważniejsze to:

* Kompilator *gcc* jest dostępny w postaci binarnej dla niemal wszystkich współczesnych systemów. Listę witryn z *gcc* i innymi programi GNU znajdziesz pod adresem *http://www.gnu.org/*.

```
WHATSNEW.txt
docs/textdocs/UNIX_INSTALL.txt
```

Pliki te informują, jakich funkcji możesz oczekiwać w danej dystrybucji Samby i zwracają uwagę na częste problemy instalacyjne, z którymi przyjdzie ci się zmierzyć. Koniecznie przeczytaj oba te pliki, zanim przystąpisz do instalacji.

Konfigurowanie Samby

· ródłowa instalacja Samby 2.0 i nowszych wersji początkowo nie zawiera pliku *makefile*. Należy wygenerować go za pomocą skryptu GNU *configure*, który jest ulokowany w katalogu *samba-2.0.x/source*. Skrypt *configure*, który należy uruchomić z uprawnieniami użytkownika root, konfiguruje opcje instalacyjne zależne od systemu. Jeśli jednak chciałbyś mieć wpływ na pewne parametry globalne, możesz ustawić je za pomocą opcji linii polecenia:

```
# ./configure --with-ssl
```

Powyższy przykład konfiguruje plik *makefile* Samby do obsługi protokołu szyfrującego Secure Sockets Layer (SSL). Jeśli chciałbyś wyświetlić pełną listę opcji, wydaj następujące polecenie:

```
# ./configure --help
```

Każda z opcji włącza lub wyłącza różne funkcje Samby. Włączenie danej funkcji polega zwykle na dodaniu opcji typu `--with-`*`funkcja`*, która powoduje skompilowanie i zainstalowanie funkcji. Podobnie, jeśli podasz opcję `--without-`*`funkcja`*, określona funkcja zostanie wyłączona. W wersji 2.0.5 Samby poniższe funkcje są domyślnie wyłączone:

`--with-smbwrapper`
: Dołącza obsługę nakładki SMB, dzięki której uniksowe programy mogą korzystać z systemów plików SMB/CIFS tak, jakby były to zwykłe systemy uniksowe. Zalecamy włączenie tej opcji. Jednakże w czasie oddawania tej książki do druku istniały pewne niezgodności między pakietem *smbwrapper* i biblioteką GNU *glibc* w wersji 2.1, co uniemożliwiało jego kompilację w Red Hacie 6.0. Więcej informacji o tych niezgodnościach można znaleźć na stronie głównej Samby.

`--with-afs`
: Dołącza obsługę systemu plików Andrew Filesystem, opracowanego na Uniwersytecie Carnegie Mellon. Jeśli nawet zamierzasz udostępniać pliki AFS za pośrednictwem Samby, radzimy wstępnie skompilować Sambę bez tej opcji i upewnić się, że wszystko działa poprawnie. Następnie można ponownie skompilować Sambę z włączoną obsługą AFS i porównać ewentualne błędy z wcześniejszą konfiguracją.

`--with-dfs`
: Dołącza obsługę DFS, nowszej wersji AFS używanej przez OSF/1 (Digital Unix). Warto podkreślić, że jest to zupełnie inny system plików niż Microsoft DFS. I tym razem radzimy wstępnie skompilować Sambę bez tej opcji i upewnić się, że wszystko działa poprawnie, a następnie ponownie skompilować Sambę z tą opcją i porównać ewentualne błędy z wcześniejszą konfiguracją.

`--with-krb4=`*`katalog-podstawowy`*

Dołącza obsługę protokołu Kerberos w wersji 4.0, jawnie określając podstawowy katalog dystrybucji. Kerberos to sieciowy protokół bezpieczeństwa opracowany w MIT, który wykorzystuje kryptografię z kluczem prywatnym do zapewnienia ścisłej ochrony węzłów sieci. Tak na marginesie, Microsoft zapowiedział, że Kerberos w wersji 5.0 będzie standardowym mechanizmem uwierzytelniającym w Windows 2000 (NT 5.0). Mechanizmy uwierzytelniania Kerberosa 5.0 różnią się jednak znacznie od mechanizmów bezpieczeństwa Kerberosa 4.0. Jeśli używasz w swoim systemie Kerberosa 4, zespół programistów Samby zaleca uaktualnienie i użycie opcji `--with-krb5` (patrz następny punkt). Więcej informacji o Kerberosie znajdziesz pod adresem *http://web.mit.edu/kerberos/www*.

`--with-krb5=`*`katalog-podstawowy`*

Dołącza obsługę protokołu Kerberos w wersji 5.0, jawnie określając podstawowy katalog dystrybucji. Microsoft ogłosił, że Kerberos 5.0 będzie standardowym mechanizmem uwierzytelniającym w Windows 2000 (NT 5.0). Jednakże nie ma gwarancji, że w przyszłości Microsoft nie wprowadzi do Kerberosa własnych poprawek. Obecnie obsługa Kerberosa w Sambie uwzględnia tylko hasła przesyłane otwartym tekstem, a nie zaszyfrowane. Więcej informacji o Kerberosie znajdziesz na jego stronie głównej pod adresem *http://web.mit.edu/kerberos/www*.

`--with-automount`

Dołącza obsługę automatycznego programu montującego, często używanego w sieciach wykorzystujących NFS.

`--with-smbmount`

Dołącza obsługę programu *smbmount*, działającego tylko w Linuksie. Kod tego programu nie jest już rozwijany, więc programiści Samby zdecydowali, że opcja ta będzie domyślnie wyłączona i w zamian udostępnili program *smbwrapper*. Ten ostatni działa także w innych odmianach Uniksa, więc prawdopodobnie lepiej będzie użyć opcji `--with-smbwrapper`.

`--with-pam`

Dołącza obsługę wymiennych modułów uwierzytelniających (*Pluggable Authentication Modules*, PAM), mechanizmu uwierzytelniającego używanego w wielu dystrybucjach Linuksa.

`--with-ldap`

Dołącza obsługę protokołu Lightweight Directory Access Protocol (LDAP). Przyszła wersja LDAP będzie używana w systemie operacyjnym Windows 2000 (NT 5.0); obsługa tego protokołu w Sambie jest eksperymentalna. LDAP to elastyczny protokół katalogowy* typu klient-serwer, który umożliwia przenoszenie informacji o certyfikatach i przynależności do grup.

`--with-nis`

Umożliwia pozyskiwanie informacji z pliku haseł NIS (sieciowych „żółtych stron").

---

* Przez *katalog* nie rozumiemy tutaj katalogu w systemie plików, ale poindeksowany spis (taki jak książka telefoniczna). Publiczny system LDAP umożliwia łatwe składowanie i udostępnianie informacji.

`--with-nisplus`
Umożliwia pozyskiwanie informacji z pliku haseł NIS+, następcy NIS.

`--with-ssl`
Dołącza eksperymentalną obsługę protokołu Secure Sockets Layer (SSL), używanego do tworzenia szyfrowanych połączeń między klientem i serwerem. Więcej informacji na ten temat znajdziesz w dodatku A, *Konfigurowanie Samby do obsługi SSL*.

`--with-nisplus-home`
Umożliwia lokalizowanie serwera z katalogiem macierzystym użytkownika i łączenie klienta z tym serwerem. Wymaga użycia opcji `--with-nis` i zwykle `--with-automounter`.

`--with-mmap`
Uaktywnia eksperymentalny kod mapowania plików w pamięci. Opcja ta nie jest wymagana do zapewnienia szybkich blokad, które i tak używają mapowania plików w pamięci lub pamięci dzielonej typu System V.

`--with-syslog`
Włącza obsługę rejestrowania przez mechanizm SYSLOG informacji generowanych przez serwer Samby. Istnieje kilka opcji konfiguracyjnych Samby, dzięki którym można włączyć rejestrowanie komunikatów serwera przez SYSLOG; opcje te omówimy w rozdziale 4, *Udziały dyskowe*.

`--with-netatalk`
Włącza eksperymentalną współpracę z (macintoshowym) serwerem plików Netatalk.

`--with-quotas`
Dołącza obsługę limitów dyskowych.

Opcje te są domyślnie wyłączone, ponieważ żadna z nich nie ma zasadniczego wpływu na działanie Samby. Możesz jednak zawsze wrócić do tego punktu i skompilować zmodyfikowaną wersję Samby, gdyby okazało się, że któraś z nich jest ci potrzebna.

W tabeli 2.1 wymienione są dodatkowe parametry, które możesz przekazać skryptowi *configure*, jeśli chciałbyś umieścić różne części dystrybucji Samby w niestandardowych katalogach, na przykład w celu wykorzystania kilku dysków lub partycji. Zauważ, że ścieżki domyślne czasem odwołują się do przedrostka określonego za pomocą opcji `--prefix`.

Tabela 2.1. Dodatkowe opcje konfiguracyjne

Opcja	*Opis*	*Wartość domyślna*
`--prefix=`*`katalog`*	Instaluje pliki niezależne od architektury w określonym katalogu podstawowym	*/usr/local/samba*
`--eprefix=`*`katalog`*	Instaluje pliki zależne od architektury w określonym katalogu podstawowym	*/usr/local/samba*

Opcja	*Opis*	*Wartość domyślna*
`--bindir=`*katalog*	Instaluje wykonywalne pliki użytkownika w określonym katalogu	*eprefix*/*bin*
`--sbindir=`*katalog*	Instaluje wykonywalne pliki administratora w określonym katalogu	*eprefix*/*bin*
`--libexecdir=`*katalog*	Instaluje wykonywalne pliki programów w określonym katalogu	*eprefix*/*libexec*
`--datadir=`*katalog*	Instaluje pliki przeznaczone tylko do odczytu i niezależne od architektury w określonym katalogu	*prefix*/*share*
`--libdir=`*katalog*	Instaluje biblioteki programów w określonym katalogu	*eprefix*/*lib*
`--includedir=`*katalog*	Instaluje pliki nagłówkowe pakietu w określonym katalogu	*prefix*/*include*
`--infodir=`*katalog*	Instaluje dodatkowe pliki konfiguracyjne w określonym katalogu	*prefix*/*info*
`--mandir=`*katalog*	Instaluje strony man w określonym katalogu	*prefix*/*man*

Zanim uruchomisz skrypt *configure*, powinieneś zalogować się w systemie jako root. W przeciwnym wypadku prawdopodobnie otrzymasz następujące ostrzeżenie:

```
configure: warning: running as non-root will disable some tests
```

(ostrzeżenie: uruchomienie skryptu bez uprawnień roota spowoduje pominięcie niektórych testów).

Lepiej nie pomijać żadnych testów podczas tworzenia pliku *makefile* Samby; mogłoby to spowodować błędy podczas kompilowania lub uruchamiania już skompilowanej Samby.

Oto przykładowe wyniki wykonania skryptu *configure*, który tworzy plik *makefile* Samby 2.0.4 do kompilacji w Linuksie. Skrypt *configure* należy uruchomić w katalogu *source*; w wynikach pominięto część linii ze środka:

```
# cd samba-2.0.4b/source
# ./configure | tee mojdziennik

loading cache ./config.cache
checking for gcc... (cached) gcc
checking whether the C compiler (gcc -O ) works... yes
checking whether the C compiler (gcc -O ) is a cross-compiler... no
checking whether we are using GNU C... (cached) yes
checking whether gcc accepts -g... (cached) yes
checking for a BSD compatible install... (cached) /usr/bin/install -c
```

...(*pominięty fragment*)...

```
checking configure summary
configure OK
creating ./config.status
creating include/stamp-h
creating Makefile
creating include/config.h
```

Ogólnie rzecz biorąc, każdy komunikat skryptu *configure* nie zaczynający się od słów `checking` lub `creating` jest błędem; często pomaga przeadresowanie wyjścia skryptu do pliku, jak uczyniliśmy to powyżej za pomocą polecenia `tee`. Jeśli podczas konfiguracji wystąpi błąd, bardziej szczegółowe informacje o nim można znaleźć w pliku *config.log*, zapisanym w bieżącym katalogu przez skrypt *configure*.

Jeśli konfiguracja jest poprawna, na ekranie pojawi się komunikat `checking configure summary`, a po nim `configure OK` i cztery lub pięć komunikatów o tworzeniu plików. Jak dotąd, wszystko idzie gładko... Następny etap: kompilacja.

Kompilowanie i instalowanie Samby

W tym momencie powinieneś być gotowy do skompilowania plików wykonywalnych Samby. Kompilowanie jest również łatwe: wystarczy wpisać polecenie `make` w katalogu *source*. Narzędzie *make* wyświetli mnóstwo informacji i komunikatów o pomyślnej kompilacji, zaczynając od:

```
Using FLAGS = -O -Iinclude ...
```

Budowanie programów zaczyna się od kompilacji *smbd* i *nmbd*, a kończy na poleceniu konsolidującym *bin/make_printerdef*. Oto przykładowa kompilacja Samby 2.0.4 w serwerze linuksowym:

```
# make
Using FLAGS =  -O -Iinclude -I./include -I./ubiqx -I./smbwrapper -DSMBLOGFILE="/
usr/local/samba/var/log.smb" -DNMBLOGFILE="/usr/local/samba/var/log.nmb" -
DCONFIGFILE="/usr/local/samba/lib/smb.conf" -DLMHOSTSFILE="/usr/local/samba/lib/
lmhosts"   -DSWATDIR="/usr/local/samba/swat" -DSBINDIR="/usr/local/samba/bin" -
DLOCKDIR="/usr/local/samba/var/locks" -DSMBRUN="/usr/local/samba/bin/smbrun" -
DCODEPAGEDIR="/usr/local/samba/lib/codepages" -DDRIVERFILE="/usr/local/samba/lib/
printers.def" -DBINDIR="/usr/local/samba/bin" -DHAVE_INCLUDES_H -DPASSWD_
PROGRAM="/bin/passwd" -DSMB_PASSWD_FILE="/usr/local/samba/private/smbpasswd"
Using FLAGS32 =  -O -Iinclude -I./include -I./ubiqx -I./smbwrapper -
DSMBLOGFILE="/usr/local/samba/var/log.smb" -DNMBLOGFILE="/usr/local/samba/var/log.
nmb" -DCONFIGFILE="/usr/local/samba/lib/smb.conf" -DLMHOSTSFILE="/usr/local/samba/
lib/lmhosts"   -DSWATDIR="/usr/local/samba/swat" -DSBINDIR="/usr/local/samba/bin"
-DLOCKDIR="/usr/local/samba/var/locks" -DSMBRUN="/usr/local/samba/bin/smbrun" -
DCODEPAGEDIR="/usr/local/samba/lib/codepages" -DDRIVERFILE="/usr/local/samba/lib/
printers.def" -DBINDIR="/usr/local/samba/bin" -DHAVE_INCLUDES_H -DPASSWD_
PROGRAM="/bin/passwd" -DSMB_PASSWD_FILE="/usr/local/samba/private/smbpasswd"
Using LIBS = -lreadline -ldl  -lcrypt -lpam
Compiling smbd/server.c
Compiling smbd/files.c
Compiling smbd/chgpasswd.c
```

...(*pominięty fragment*)...

```
Compiling rpcclient/cmd_samr.c
Compiling rpcclient/cmd_reg.c
Compiling rpcclient/cmd_srvsvc.c
Compiling rpcclient/cmd_netlogon.c
Linking bin/rpcclient
Compiling utils/smbpasswd.c
Linking bin/smbpasswd
Compiling utils/make_smbcodepage.c
Linking bin/make_smbcodepage
```

```
Compiling utils/nmblookup.c
Linking bin/nmblookup
Compiling utils/make_printerdef.c
Linking bin/make_printerdef
```

Jeśli napotkasz problemy podczas kompilacji, zajrzyj do dokumentacji Samby – być może ich rozwiązanie okaże się łatwe. Możesz także poprosić o pomoc w listach wysyłkowych Samby, których adresy podajemy na końcu dodatku D, *Spis demonów i poleceń Samby*, lub poszukać informacji na stronie głównej Samby. Większość problemów z kompilacją ma związek z systemem i z reguły łatwo się z nimi uporać.

Po skompilowaniu plików możesz zainstalować je w określonych wcześniej katalogach za pomocą polecenia:

```
# make install
```

Jeśli uaktualniasz program, starsze wersje plików Samby zostaną zapisane z rozszerzeniem *.old* i będziesz mógł przywrócić poprzednią wersję za pomocą polecenia `make revert`. Po wydaniu polecenia `make install`, powinieneś skopiować pliki *.old* (jeśli takie istnieją) do innego katalogu. W przeciwnym wypadku podczas następnego instalowania Samby zostaną one nadpisane bez ostrzeżenia i utracisz poprzednią wersję programu. Jeśli skonfigurowałeś Sambę tak, aby korzystała z domyślnych lokacji dla plików, nowe pliki zostaną zainstalowane w katalogach wymienionych w tabeli 2.2. Pamiętaj, że musisz przeprowadzić instalację z konta, które ma prawo do zapisu w docelowych katalogach; zwykle jest to konto użytkownika root.

Tabela 2.2. Katalogi instalacyjne Samby

Katalog	*Opis*
/usr/local/samba	Główne drzewo
/usr/local/samba/bin	Pliki binarne
/usr/local/samba/lib	smb.conf, *lmhosts*, pliki konfiguracyjne itp.
/usr/local/samba/man	Dokumentacja Samby
/usr/local/samba/private	Plik zaszyfrowanych haseł Samby
/usr/local/samba/swat	Pliki programu SWAT
/usr/local/samba/var	Pliki dziennika Samby, pliki blokady, informacje o liście przeglądania, pliki pamięci dzielonej, pliki z identyfikatorami procesów

W następnych rozdziałach będziemy od czasu do czasu używać skrótu *`katalog_samby`* na oznaczenie głównego drzewa katalogów Samby. W większości konfiguracji podstawowym katalogiem zainstalowanego pakietu Samby jest */usr/local/samba*.

Czy nie skonfigurowałeś partycji */usr* jako przeznaczonej tylko do odczytu? W takim przypadku będziesz musiał umieścić gdzie indziej pliki dziennika, blokady i haseł.

Oto zapis instalacji przeprowadzonej w naszym komputerze. Jak widzisz, wybraliśmy */usr/local/samba* (to jest *katalog_samby*) na podstawowy katalog dystrybucji.

```
# make install
Using FLAGS =  -O -Iinclude -I./include -I./ubiqx -I./smbwrapper - DSMBLOGFILE="/
usr/local/samba/var/log.smb" -DNMBLOGFILE="/usr/local/samba/var/log.nmb" -
DCONFIGFILE="/usr/local/samba/lib/smb.conf" -

...(pominięty fragment)...

The binaries are installed. You may restore the old binaries (if there
were any) using the command "make revert". You may uninstall the binaries
using the command "make uninstallbin" or "make uninstall" to uninstall
binaries, man pages and shell scripts.

...(pominięty fragment)...

======================================================================
The SWAT files have been installed. Remember to read the swat/README
for information on enabling and using SWAT.
======================================================================
```

Jeśli ostatni komunikat dotyczył programu SWAT, udało ci się zainstalować wszystkie pliki. Gratulacje! Samba jest teraz w twoim systemie.

Końcowe czynności instalacyjne

Do wykonania zostało jeszcze kilka końcowych czynności. Należy dodać narzędzie Samba Web Administration Tool (SWAT) do plików konfiguracyjnych */etc/services* i */etc/inetd.conf*. Program SWAT działa jako demon pod kontrolą *inetd* i udostępnia w przeglądarkach WWW formularze służące do tworzenia i modyfikowania plików konfiguracyjnych SMB.

1. Aby dodać obsługę SWAT, dopisz następującą linię na końcu pliku */etc/services*:

   ```
   swat   901/tcp
   ```

2. Dopisz poniższą linię do pliku */etc/inetd.conf* (zajrzyj na stronę podręcznika man dla pliku *inetd.conf*, jeśli jego format w twoim systemie jest inny niż pokazany w przykładzie poniżej). Nie zapomnij zmienić ścieżki do pliku binarnego SWAT, jeśli zainstalowałeś go w katalogu innym niż domyślny (*/usr/local/samba*).

   ```
   swat  stream tcp  nowait.400 root /usr/local/samba/bin/swat  swat
   ```

Na tym instalacja dobiega końca. Zanim jednak uruchomisz Sambę, musisz utworzyć jej plik konfiguracyjny.

Podstawowy plik konfiguracyjny Samby

Kluczem do konfigurowania Samby jest jej pojedynczy plik konfiguracyjny: *smb.conf*. Plik ten może być zupełnie prosty albo niezwykle skomplikowany (resztę książki poświęciliśmy temu, abyś nabrał do niego bardzo osobistego stosunku). Na razie pokażemy ci, jak skonfigurować jedną usługę plikową, abyś mógł uruchomić demony Samby i upewnić się, że wszystko działa tak, jak powinno. W następnych roz-

działach dowiesz się, jak skonfigurować Sambę do bardziej skomplikowanych i interesujących zadań.

Plik konfiguracyjny *smb.conf* nie jest automatycznie tworzony w procesie instalacji, choć kilka przykładowych plików jest dołączonych do dystrybucji Samby. Aby przetestować programy serwera, użyjemy jednak poniższego pliku. Powinien on nazywać się *smb.conf* i znajdować w katalogu */usr/local/samba/lib**.

```
[global]
    workgroup = PROSTA_GRUPA

[test]
    comment = Tylko w celach testowych, je□li □aska
    path = /export/samba/test
    read only = no
    guest ok = yes
```

Ten krótki plik konfiguracyjny mówi serwerowi Samby, aby udostępnił katalog */export/samba/test* jako udział SMB/CIFS o nazwie `test`. Serwer staje się także częścią grupy roboczej o nazwie PROSTA_GRUPA, której członkiem musi być także każdy klient (użyj zamiast tego nazwy własnej grupy roboczej, jeśli wiesz już, jak będzie brzmiała). Udziału `[test]` użyjemy w następnym rozdziale, demonstrując konfigurowanie klientów Windows. Teraz dokończ konfigurowanie Samby, wprowadzając następujące polecenia w uniksowym serwerze jako użytkownik root:

```
# mkdir /export/samba/test
# chmod 777 /export/samba/test
```

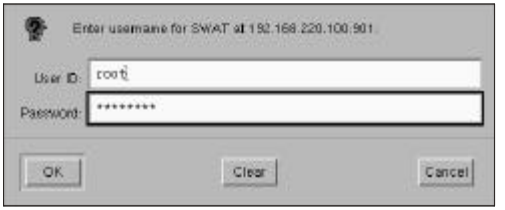
Winniśmy nadmienić, że z punktu widzenia bezpieczeństwa systemu jest to najgorsza z możliwych konfiguracji. Na razie chcemy jednak tylko przetestować Sambę, więc tymczasowo pominiemy kwestie bezpieczeństwa. Co więcej, niebawem zapoznamy się z pewnymi zagadnieniami dotyczącymi szyfrowania haseł przez klientów Windows, a taka konfiguracja przysporzy nam najmniej kłopotów.

Jeśli używasz Windows 98 lub Windows NT z dodatkiem Service Pack 3 lub nowszym, musisz dodać następujący wpis do sekcji `[global]` pliku konfiguracyjnego Samby: `encrypt passwords = yes`. Oprócz tego musisz użyć programu *smbpassword* (zwykle umieszczonego w katalogu */usr/local/samba/bin*), aby wprowadzić do zaszyfrowanej bazy danych klientów Samby kombinacje „nazwa użytkownika-hasło" dla tych użytkowników uniksowego serwera, którzy powinni mieć dostęp do udziałów. Jeśli chciałbyś na przykład pozwolić uniksowemu użytkownikowi `stefan` na dostęp do udziałów serwera z klienta SMB, wpisałbyś polecenie `smbpassword -a stefan`. Kiedy dodasz pierwszego użytkownika, program wyświetli komunikat o błędzie, informując, że baza haseł nie istnieje. Nie martw się – program automatycznie utworzy tę bazę danych. Upewnij się, że kombinacje „nazwa użytkownika-hasło" dodawane do zaszyfrowanej bazy danych odpowiadają nazwom i hasłom, które będą używane w klientach Windows.

* Jeśli nie kompilowałeś Samby, lecz pobrałeś pakiet binarny, sprawdź w dokumentacji, gdzie należy umieścić plik *smb.conf*. Jeśli Samba została zainstalowana wraz z systemem, plik ten prawdopodobnie znajduje się już w którymś katalogu.

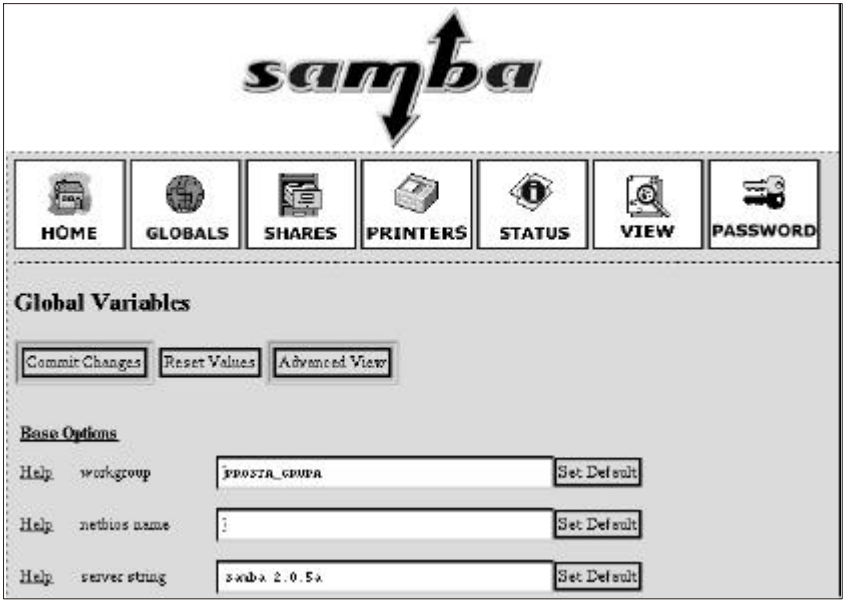
Korzystanie z programu SWAT

Od wersji 2.0 Samby nie trzeba pisać ręcznie pliku konfiguracyjnego. Możesz uruchomić przeglądarkę WWW, połączyć się z adresem *http://localhost:901* i zalogować się na koncie roota, co przedstawia rysunek 2.1.



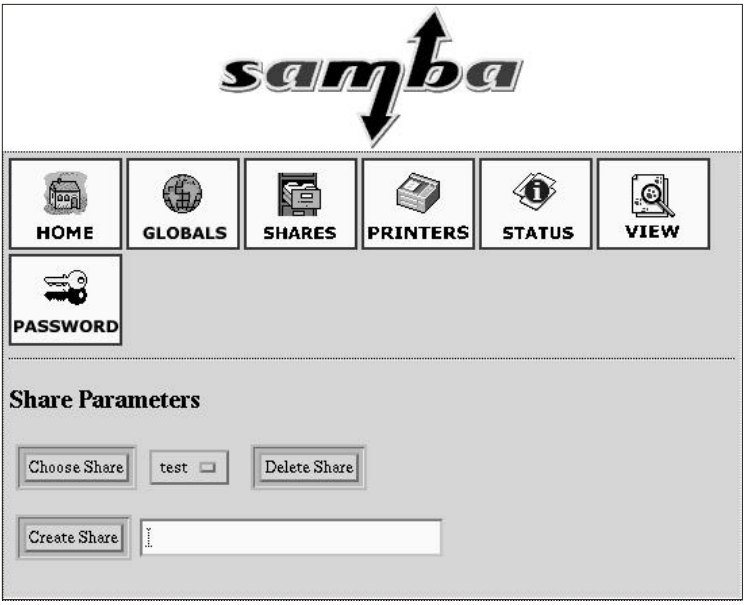
Rysunek 2.1. Logowanie się w programie SWAT

Po zalogowaniu się kliknij przycisk GLOBALS na górze ekranu. Powinieneś zobaczyć stronę Global Variables, przedstawioną na rysunku 2.2.



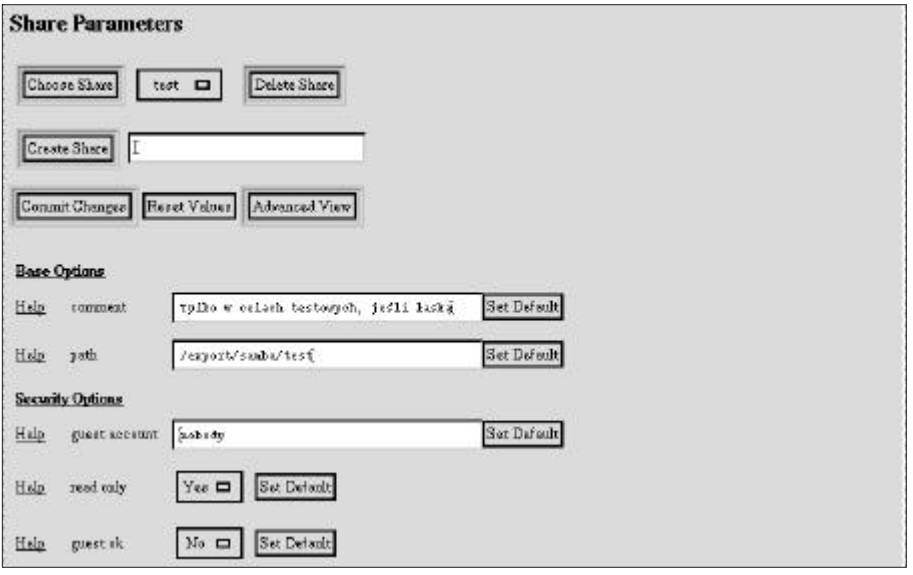
Rysunek 2.2. Strona Global Variables w programie SWAT

Wpisz PROSTA_GRUPA w pole workgroup i ustaw pole security na USER. W tym menu musisz zmienić jeszcze jedną opcję, określającą, który system w sieci lokalnej odwzorowuje adresy NetBIOS-u; system ten nazywany jest *serwerem WINS*. Na samym dole strony ustaw pole WINS support na Yes, chyba że w twojej sieci jest już serwer WINS. W takim przypadku wpisz adres IP serwera WINS w pole wins server. Następnie wróć na górę ekranu i kliknij przycisk Commit Changes, aby zapisać zmiany w pliku *smb.conf*.



Rysunek 2.3. Strona Share Creation programu SWAT

Następnie kliknij ikonę SHARES. Powinieneś zobaczyć stronę taką jak na rysunku 2.3. Wybierz nazwę test z listy obok przycisku Choose Share. Zobaczysz stronę Share Parameters, przypominającą tę z rysunku 2.4. Wcześniej dopisaliśmy w pliku *smb.conf* komentarz, który przypominał, że jest to tylko udział testowy. SWAT skopiował wszystkie te informacje.



Rysunek 2.4. Strona Share Parameters programu SWAT

Jeśli klikniesz przycisk View, SWAT wyświetli następujący plik *smb.conf*:

```
# Samba config file created using SWAT
# from localhost (127.0.0.1)
# Date: 1998/11/27 15:42:40

# Global parameters
        workgroup = PROSTA_GRUPA

[test]
        comment = Tylko w celach testowych, je□li □aska
        path = /export/samba/test
        read only = No
        guest ok = Yes
```

Kiedy plik konfiguracyjny będzie gotowy, możesz pominąć następny etap, ponieważ dane wyjściowe programu SWAT są na pewno poprawne pod względem syntaktycznym.

Testowanie pliku konfiguracyjnego

Jeśli nie tworzyłeś pliku konfiguracyjnego za pomocą programu SWAT, prawdopodobnie powinieneś przetestować go i upewnić się, że jest poprawny pod względem syntaktycznym. Pomysł sprawdzania ośmiowierszowego pliku konfiguracyjnego programem testującym może wydawać się śmieszny, ale robimy to, aby nabrać wprawy przed sprawdzaniem prawdziwych plików konfiguracyjnych, które będziemy pisać później.

Analizator składniowy, program *testparm*, sprawdza plik *smb.conf* pod kątem błędów syntaktycznych i informuje o wszystkich znalezionych uchybieniach, wyświetlając zarazem listę usług udostępnianych przez serwer. Poniżej podajemy przykład; jak zauważysz, tak nam było spieszno do uruchomienia serwera, że w pośpiechu zamiast `workgroup` napisaliśmy `workgrp` (wyniki programu są często dość długie, więc radzimy przechwycić końcówkę za pomocą polecenia `tee`):

```
Load smb config files from smb.conf
Unknown parameter encountered: "workgrp"
Ignoring unknown parameter "workgrp"
Processing section "[test]"
Loaded services file OK.
Press enter to see a dump of your service definitions

# Global parameters
[global]
        workgroup = WORKGROUP
        netbios name =
        netbios aliases =
        server string = Samba 2.0.5a
        interfaces =
        bind interfaces only = No
```

...(*pominięty fragment*)...

```
[test]
        comment = Tylko w celach testowych, je□li □aska
        path = /export/samba/test
        read only = No
        guest ok = Yes
```

Interesujące informacje znajdują się na początku i na końcu. W początkowej części wyników zaznaczone są wszystkie błędy składni, a w części końcowej widnieje lista usług, które serwer zamierza udostępnić. Dobra rada: upewnij się, że ty i serwer macie takie same poglądy.

Jeśli wszystko jest w porządku, możesz przystąpić do uruchomienia demonów serwera!

Uruchamianie demonów Samby

Aby Samba pracowała poprawnie, w systemie muszą działać dwa procesy: *smbd* i *nmbd*. Istnieją trzy sposoby ich uruchomienia:

- ręcznie,
- jako demony autonomiczne,
- za pośrednictwem *inetd*.

Ręczne uruchamianie demonów

Jeśli się spieszysz, możesz uruchomić demony Samby ręcznie. Wpisz po prostu jako root następujące dwa polecenia:

```
# /usr/local/samba/bin/smbd -D
# /usr/local/samba/bin/nmbd -D
```

Od tego momentu Samba zacznie działać w systemie i będzie gotowa na przyjmowanie połączeń.

Demony autonomiczne

Aby uruchomić procesy Samby jako demony autonomiczne, musisz dodać powyższe polecenia do standardowych skryptów startowych Uniksa. Procedura ta różni się w zależności od tego, czy używasz Uniksa typu BSD, czy też typu System V.

Unix BSD

W Uniksach typu BSD musisz dopisać poniższy kod do pliku *rc.local*, przechowywanego zwykle w katalogu */etc* lub */etc/rc.d*:

```
if [ -x /usr/local/samba/bin/smbd]; then
    echo "Uruchamiam smbd..."
    /usr/local/samba/bin/smbd -D
    echo "Uruchamiam nmbd..."
    /usr/local/samba/bin/nmbd -D
fi
```

Kod jest bardzo prosty: sprawdza, czy plik *smbd* ma ustawione prawo do wykonania, a jeśli tak, to uruchamia oba demony Samby podczas startu systemu.

Unix System V

W Uniksach typu System V sprawy nieco się komplikują. System V zwykle używa oddzielnych skryptów do uruchamiania i zatrzymywania systemowych demonów. Dlatego musisz poinstruować Sambę, co ma robić, kiedy zaczyna lub kończy pracę.

Możesz w tym celu zmodyfikować zawartość katalogu */etc/rc.d/init.d*, umieszczając w nim program o nazwie *smb* podobny do poniższego:

```
#!/bin/sh

# Zawiera funkcj□ "killproc" w Linuksie Red Hat
./etc/rc.d/init.d/functions

PATH="/usr/local/samba/bin:$PATH"

case $1 in
    'start')
        echo "Uruchamiam smbd..."
        smbd -D
        echo "Uruchamiam nmbd..."
        nmbd -D
        ;;
    'stop')
        echo "Zatrzymuj□ smbd i nmbd..."
        killproc smbd
        killproc nmbd
        rm -f /usr/local/samba/var/locks/smbd.pid
        rm -f /usr/local/samba/var/locks/nmbd.pid
        ;;
    *)
        echo "sk□adnia: smb {start|stop}"
        ;;
esac
```

Dzięki temu skryptowi możesz uruchamiać i zatrzymywać usługi SMB za pomocą następujących poleceń:

```
# /etc/rc.d/init.d/smb start
Uruchamiam smbd...
Uruchamiam nmbd...
# /etc/rc.d/init.d/smb stop
Zatrzymuj□ smbd i nmbd...
```

Uruchamianie demonów za pośrednictwem inetd

Demon *inetd* to internetowy „superdemon" systemów uniksowych, który monitoruje porty TCP zdefiniowane w pliku */etc/services* i wykonuje odpowiedni program dla każdego portu, co jest zdefiniowane w pliku */etc/inetd.conf*. Rozwiązanie takie ma tę zaletę, że możesz dysponować znaczną liczbą demonów gotowych do świadczenia usług, które nie muszą być uruchomione – demon *inetd* odbiera żądania w ich imieniu. Wadą jest niewielki narzut związany z tworzeniem nowego procesu demona oraz konieczność edytowania dwóch plików konfiguracyjnych zamiast jednego. Metoda ta przydaje się, jeśli twój komputer ma jednego lub dwóch użytkowników albo działa w nim zbyt wiele demonów. Łatwiej jest też przeprowadzić uaktualnienie oprogramowania bez zrywania istniejących połączeń.

Jeśli chcesz uruchamiać Sambę za pośrednictwem *inetd*, najpierw otwórz w edytorze plik */etc/services*. Jeśli nie ma w nim jeszcze poniższych linii, dopisz je:

```
netbios-ssn    139/tcp
netbios-ns     137/udp
```

Następnie zmodyfikuj plik */etc/inetd.conf*. Poszukaj w nim poniższych dwóch linii, a jeśli ich nie znajdziesz, dopisz je. Jeśli linie dla demonów `smbd` i `nmbd` są już w pliku, zmodyfikuj je tak, aby wskazywały na katalog, w którym zainstalowałeś oba demony. Twoja odmiana Uniksa może używać nieco innej składni tego pliku; kieruj się już istniejącymi wpisami i stroną podręcznika man dla pliku *inetd.conf*:

```
netbios-ssn stream tcp nowait root /usr/local/samba/bin/smbd smbd
netbios-ns  dgram  udp wait   root /usr/local/samba/bin/nmbd nmbd
```

Wreszcie usuń wszystkie procesy *smbd* i *nmbd* i wyślij sygnał HUP procesowi *inetd* (demon *inetd* ponownie wczytuje swój plik konfiguracyjny po otrzymaniu tego sygnału). W tym celu użyj polecenia `ps`, aby ustalić identyfikator procesu, i wyślij sygnał za pomocą polecenia:

```
# kill -HUP identyfikator_procesu
```

Od tego momentu Samba powinna działać.

Testowanie demonów Samby

Trudno w to uwierzyć, ale niemal skończyliśmy już konfigurowanie serwera Samby. Pozostało tylko sprawdzić, czy wszystko działa tak, jak powinno. Najwygodniejszym sposobem będzie użycie programu *smbclient* w celu sprawdzenia, jakie usługi serwer oferuje w sieci. Jeśli wszystko jest skonfigurowane prawidłowo, powinieneś móc wykonać następujące polecenie:

```
# smbclient -U% -l localhost
Added interface ip=192.168.220.100 bcast=192.168.220.255 nmask=255.255.255.0
Domain=[PROSTA_GRUPA] OS=[Unix] Server=[Samba 2.0.5a]

        Sharename      Type      Comment
        ---------      ----      -------
        test           Disk      Tylko w celach testowych, je□li □aska
        IPC$           IPC       IPC Service (Samba 2.0.5a)

        Server                   Comment
        ---------                -------
        HYDRA                    Samba 2.0.5a

        Workgroup                Master
        ---------                -------
        PROSTA_GRUPA             HYDRA
```

Jeśli pojawił się jakiś problem, nie panikuj! Spróbuj uruchomić demony ręcznie i sprawdź komunikaty systemowe lub plik diagnostyczny */usr/local/samba/var/log.smb*, aby ustalić przyczynę niepowodzenia. Jeśli uważasz, że problem jest poważniejszy, zajrzyj do rozdziału 7, *Drukowanie i odwzorowywanie nazw*, gdzie znajdziesz wskazówki dotyczące diagnozowania demonów Samby.

Jeśli wszystko działa, gratulujemy! Udało ci się skonfigurować serwer Samby z udziałem dyskowym. Jest to dość prosty serwer, ale możemy go wykorzystać do skonfigurowania i przetestowania klientów Windows 95 i NT w następnym rozdziale. Będziemy stopniowo rozszerzać jego możliwości, dodając usługi, takie jak dostęp do katalogów macierzystych, drukowanie i mechanizmy bezpieczeństwa, a wreszcie integrując go z domeną Windows.

3

Konfigurowanie klientów Windows

Z pewnością ucieszy cię wiadomość, że konfigurowanie klientów Windows do współpracy z serwerem Samby jest bardzo proste. Protokół SMB to rodzimy język Microsoftu używany do współdzielenia zasobów w sieci lokalnej, więc większa część instalacji i konfiguracji po stronie Windows została już przeprowadzona. Podstawowe kwestie, którymi zajmiemy się w tym rozdziale, to komunikacja i koordynacja między Windows a Uniksem, dwoma zupełnie różnymi systemami operacyjnymi.

Samba używa TCP/IP do porozumiewania się z klientami w sieci. Jeśli nie korzystasz jeszcze z protokołu TCP/IP w komputerach Windows, w tym rozdziale wyjaśnimy, jak go zainstalować. Następnie będziesz musiał skonfigurować komputery Windows do pracy w sieci TCP/IP. Kiedy te dwa wymagania będą spełnione, pokażemy, jak uzyskać dostęp do współdzielonego dysku w serwerze Samby.

Rozdział ten podzielony jest na trzy części. W pierwszej zajmiemy się konfigurowaniem komputerów Windows 95/98, a w drugiej – Windows NT 4.0. Ostatnia część zawiera pewne wstępne informacje o nawiązywaniu połączeń SMB przez klienty Windows i serwery. Wiadomości te przydadzą się w dalszych rozdziałach książki.

Konfigurowanie komputerów Windows 95/98

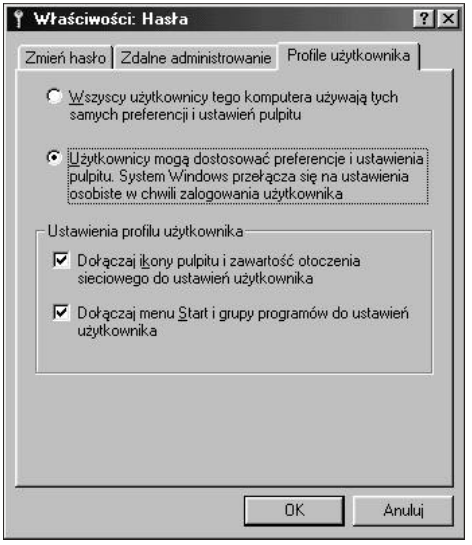
Niestety, systemy Windows 95/98 nie zostały zaprojektowane jako platformy dla wielu użytkowników, w przeciwieństwie do Uniksa albo Windows NT. Systemy te mogą jednak do pewnego stopnia obsługiwać wielu użytkowników: jeśli zechcesz, system operacyjny będzie przechowywał oddzielny profil (układ pulpitu) oraz plik haseł dla każdego użytkownika, choć jest to tylko blade odbicie środowiska prawdziwie wielodostępnego. Innymi słowy, Windows 95/98 nie przeszkodzi ci w zniszczeniu pracy innego użytkownika zapisanej na lokalnym dysku, jak to czyni Unix, ale od czegoś trzeba zacząć.

Konta i hasła

Najpierw musimy poinformować Windows, że należy przechowywać oddzielne profile użytkowników i gromadzić nazwy użytkowników i hasła w celu uwierzytelniania osób próbujących skorzystać z zasobów Samby. Służy do tego aplet Hasła

w Panelu sterowania. Jeśli jeszcze nie posługiwałeś się Panelem sterowania Windows, możesz go otworzyć, wybierając pozycję Ustawienia z menu otwieranego przez kliknięcie przycisku Start w lewym dolnym rogu ekranu. Znajdziesz go też jako folder w oknie, które otworzy się po kliknięciu ikony reprezentującej komputer, umieszczonej zwykle w lewym górnym rogu ekranu i oznaczonej napisem Mój komputer.

Po wybraniu ikony Hasła w Panelu sterowania, kliknij kartę Profile użytkownika. Powinieneś zobaczyć okno dialogowe pokazane na rysunku 3.1. Następnie kliknij pole opcji oznaczone napisem zaczynającym się od słów „Użytkownicy mogą dostosować preferencje...". Spowoduje to, że system Windows będzie przechowywał oddzielny profil dla każdego użytkownika i zapisze wprowadzoną przez ciebie nazwę użytkownika i hasło, których użyje podczas łączenia się z serwerem SMB/CIFS. Wreszcie zaznacz *oba* pola wyboru w ramce Ustawienia profilu użytkownika, jak pokazano na rysunku.



Rysunek 3.1. Okno dialogowe Właściwości: Hasła

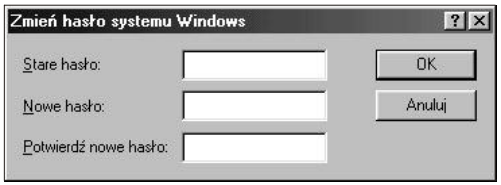
Teraz wybierz kartę Zmień hasło po lewej stronie okna dialogowego. Aby Samba pozwoliła ci na dostęp do swoich zasobów, nazwa użytkownika i hasło, które podajesz podczas logowania się w Windows, muszą odpowiadać nazwie konta i hasłu na serwerze Samby. Jeśli w oknie dialogowym nie ma tej zakładki, nie martw się – prawdopodobnie nie nadałeś sobie jeszcze nazwy użytkownika Windows i hasła. Po prostu kliknij przycisk OK na dole okna i odpowiedz Tak na pytanie o ponowne uruchomienie systemu. Pomiń następny podrozdział i przeczytaj ten zatytułowany „Pierwsze logowanie".

Zmiana hasła Windows

Po wybraniu karty Zmień hasło ukaże się okno dialogowe z rysunku 3.2.

Rysunek 3.2. Karta Zmień hasło

Kliknij przycisk Zmień hasło systemu Windows. Powinno ukazać się okno dialogowe Zmień hasło systemu Windows z rysunku 3.3. Tutaj możesz zmienić swoje hasło tak, aby odpowiadało hasłu konta w serwerze Samby, przez które zamierzasz się logować.



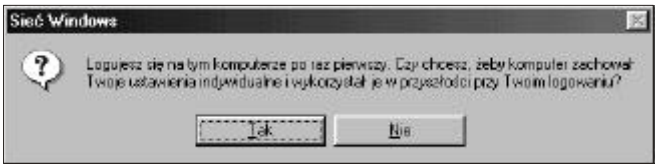
Rysunek 3.3. Okno dialogowe Zmień hasło systemu Windows

Pierwsze logowanie

Jeśli w twoim oknie Właściwości: Hasła nie było karty Zmień hasło, to po ponownym uruchomieniu Windows zostaniesz poproszony o wprowadzenie nazwy użytkownika i hasła. Skorzystaj z nazwy i hasła, które masz na serwerze Samby. Następnie je potwierdź. Teraz, po potwierdzeniu, lub od razu, jeśli miałeś już wprowadzoną nazwę i hasło, Windows powinien zapytać, czy chcesz mieć swój profil, wyświetlając okno dialogowe z rysunku 3.4.

Odpowiedz Tak. Windows utworzy dla ciebie oddzielny profil i plik haseł, w którym zapisze kopię twojego hasła. Kiedy będziesz łączył się z Sambą, Windows wyśle do niej hasło, które posłuży do uwierzytelnienia twojego dostępu do wszystkich

udziałów. Chwilowo nie będziemy zajmować się profilami; wrócimy do nich w rozdziale 6, *Użytkownicy, bezpieczeństwo i domeny*. Powinniśmy jednak ostrzec, że istnieje tu pewne niebezpieczeństwo: ktoś może wykraść plik z hasłami i rozszyfrować je, ponieważ nie są one silnie zaszyfrowane. Niestety, problemu tego nie da się rozwiązać w Windows 95/98. W Windows 2000 (NT 5.0) szyfrowanie haseł ma opierać się na znacznie lepszym algorytmie.



Rysunek 3.4. Profile sieci Windows

Konfigurowanie sieci

Teraz powinieneś sprawdzić, czy protokół sieciowy TCP/IP jest prawidłowo skonfigurowany. W tym celu kliknij dwukrotnie ikonę Sieć w Panelu sterowania. Powinieneś zobaczyć okno dialogowe służące do konfigurowania sieci, pokazane na rysunku 3.5.

Rysunek 3.5. Okno dialogowe Sieć w Windows 95/98

Działanie sieci Microsoftu opiera się na wiązaniu określonych protokołów, takich jak IPX lub TCP/IP, z konkretnymi urządzeniami sprzętowymi, na przykład z kartą Ethernetu lub połączeniem dial-up. Przesyłając dane określonego protokołu przez

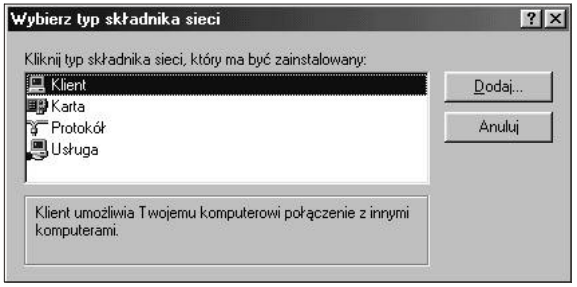
wskazane urządzenie sprzętowe, komputer może działać jako klient lub serwer dla określonego typu sieci. W przypadku Samby powinniśmy powiązać protokół TCP/IP z urządzeniem sieciowym, czyniąc komputer klientem sieci Microsoft. Kiedy więc ukaże się okno dialogowe, w komputerze powinien być zainstalowany przynajmniej Klient sieci Microsoft Networks, a także urządzenie sieciowe (najlepiej karta Ethernetu) powiązane z protokołem TCP/IP. Jeśli w komputerze jest tylko jedno urządzenie sieciowe, protokół TCP/IP będzie wymieniony pod nim. Jeśli twoje okno przypomina to z rysunku 3.5, protokół jest powiązany z urządzeniem.

Być może zobaczysz też przydatną usługę udostępniania plików i drukarek w sieciach Microsoft Networks. Oprócz tego może pojawić się protokół NetBEUI lub protokoły Novella, które są standardowo instalowane w Windows, ale niepożądane podczas pracy z TCP/IP. Usuń protokół NetBEUI, jeśli to możliwe – jest niepotrzebny i utrudnia diagnozowanie przeglądania zasobów sieci. Jeśli w twojej sieci nie ma żadnych serwerów Novella, możesz usunąć również obsługę protokołów Novella (IPX/SPX).

Dodawanie protokołu TCP/IP

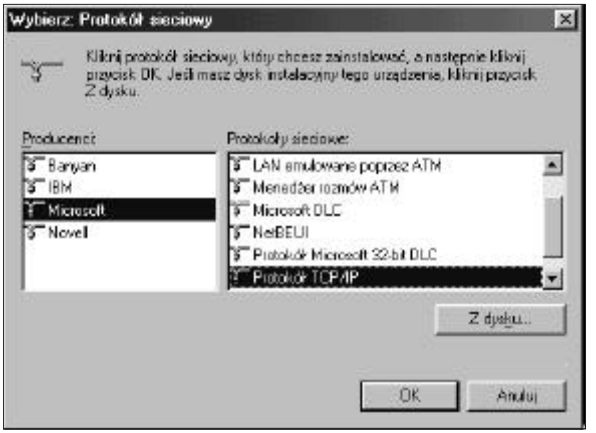
Jeśli protokół TCP/IP w ogóle nie jest wymieniony, będziesz musiał go zainstalować. Jeśli masz już TCP/IP, pomiń ten etap i przystąp do lektury podrozdziału „Ustawianie nazwy komputera i grupy roboczej".

Instalowanie TCP/IP nie jest trudne, ponieważ Microsoft dystrybuuje własną wersję protokołu na instalacyjnym CD-ROM-ie. Protokół możesz dodać, klikając przycisk Dodaj w oknie składników sieci. Zaznacz, że chcesz dodać nowy protokół, wybierając z listy pozycję Protokół i klikając przycisk Dodaj... w wyświetlonym oknie dialogowym, które powinno wyglądać tak jak na rysunku 3.6.



Rysunek 3.6. Wybieranie protokołu do zainstalowania

Następnie zaznacz protokół TCP/IP producenta Microsoft, jak pokazano na rysunku 3.7 i kliknij przycisk OK. Wrócisz do okna dialogowego Sieć. Kliknij OK, aby zamknąć okno dialogowe, po czym Windows zainstaluje niezbędne składniki i ponownie uruchomi komputer.



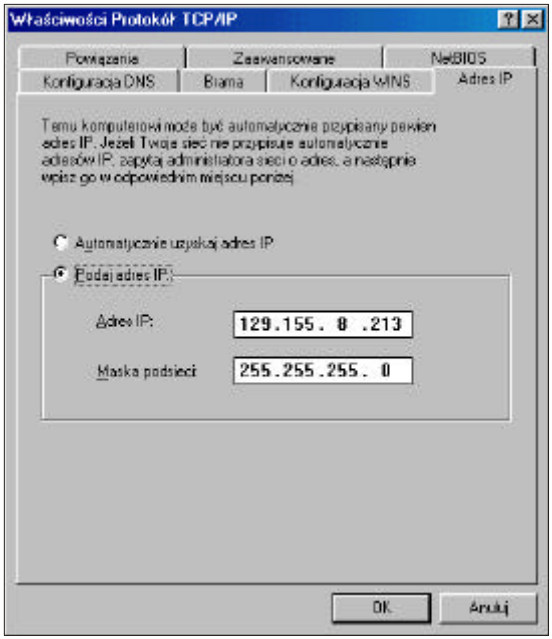
Rysunek 3.7. Wybieranie protokołu do zainstalowania

Konfigurowanie TCP/IP

Jeśli masz więcej niż jedno urządzenie sieciowe (na przykład kartę Ethernetu i modem), powinny być one powiązane z protokołem TCP/IP, co jest oznaczone strzałką (patrz rysunek 3.8). Zaznacz protokół TCP/IP powiązany z urządzeniem sieciowym, przez które będziesz łączył się z siecią Samby, i kliknij przycisk Właściwości.

Rysunek 3.8. Zaznaczanie właściwego protokołu TCP/IP

Powinno ukazać się okno właściwości tego urządzenia, pokazane na rysunku 3.9.



Rysunek 3.9. Okno Właściwości Protokół TCP/IP

Na górze okna widnieje siedem kart. Będziesz musiał zmienić parametry na czterech następujących kartach:

- Adres IP,
- Konfiguracja DNS,
- Konfiguracja WINS,
- Powiązania.

Karta Adres IP

Karta Adres IP jest pokazana na rysunku 3.9. Kliknij pole opcji Podaj adres IP i wprowadź adres klienta oraz maskę podsieci w odpowiednie pola. Ty albo administrator sieci powinniście wcześniej wybrać adres komputera. Wartości należy dobrać tak, aby komputer znajdował się w tej samej podsieci, co serwer Samby. Jeśli na przykład adres serwera to 192.168.236.86, a jego maska podsieci to 255.255.255.0, mógłbyś przypisać komputerowi Windows 98 adres 192.168.236.10 (jeśli jest wolny) i taką samą maskę podsieci, jaką ma serwer Samby. Jeśli w swojej sieci używasz serwera DHCP do przydzielania adresów IP komputerom Windows, zaznacz pole opcji Automatycznie uzyskaj adres IP z serwera DHCP.

Karta Konfiguracja DNS

Usługa Domain Name Service (DNS) jest odpowiedzialna za tłumaczenie internetowych nazw komputerów, takich jak *finta.przyklad.com*, na czytelne dla komputerów adresy IP, takie jak 192.168.236.10. Istnieją dwa sposoby na dokonanie tego w komputerze Windows: możesz określić serwer, który zajmie się tłumaczeniem, albo przechowywać lokalną listę par nazwa-adres, do której będą odwoływać się programy.

Sieci podłączone do Internetu używają zwykle serwera DNS, ponieważ plik z adresami hostów miałby ogromne rozmiary. W sieci lokalnej (bez dostępu do Internetu) lista hostów jest krótka i dobrze znana; w komputerze uniksowym można ją przechowywać w pliku */etc/hosts*. Jeśli nie jesteś pewien, czy sieć korzysta z serwera DNS albo jaki jest jego adres, zajrzyj do pliku */etc/resolv.conf* w serwerze uniksowym. Wszystkie komputery używające DNS będą miały ten plik. Wygląda on następująco:

```
#resolv.conf
domain przyklad.com
nameserver 127.0.0.1
nameserver 192.168.236.20
```

W podanym wyżej przykładzie druga linia `nameserver` zawiera adres IP innego komputera w sieci lokalnej: 192.168.236.20. To dobry kandydat na serwer DNS*.

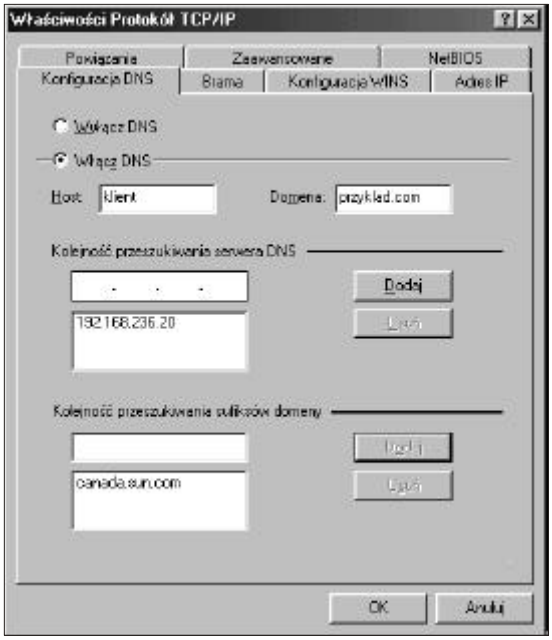
Musisz wpisać właściwy adres IP jednego lub kilku serwerów DNS (zauważ, że *nie możesz* użyć jego internetowej nazwy, takiej jak *dns.oreilly.com*) w odpowiednie pole (patrz rysunek 3.10). Nie wpisuj adresu 127.0.0.1 – to z pewnością nie jest poprawny adres serwera DNS!

Spróbuj wybrać adres z własnej sieci. Działać będą wszystkie serwery wymienione w pliku */etc/resolv.conf*, ale korzystając z pobliskiego serwera osiągniesz lepszą wydajność (jeśli nie znalazłeś pliku */etc/resolv.conf* w uniksowych komputerach, po prostu wyłącz DNS, dopóki nie poznasz adresu przynajmniej jednego serwera). Załóżmy, że masz tylko jeden serwer DNS, a jego adres to 192.168.236.20. Kliknij pole opcji Włącz DNS, jak pokazano na rysunku 3.10, i wpisz adres serwera w górne pole Kolejność przeszukiwania serwera DNS.

Wpisz też nazwę komputera Windows 95/98 i swojej domeny internetowej. Jeśli chodzi o współpracę z Sambą, możesz zignorować pole Kolejność przeszukiwania sufiksów domeny.

---

* Możemy pominąć drugi adres, ponieważ każdy uniksowy komputer ma adres lokalnego hosta 127.0.0.1, czy jest podłączony do sieci, czy też nie. Adres ten jest potrzebny do poprawnego działania niektórych narzędzi systemowych.



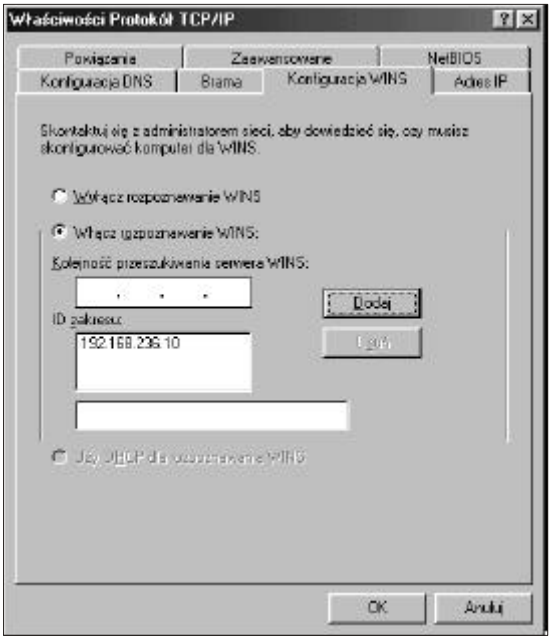
Rysunek 3.10. Karta Konfiguracja DNS

Karta Konfiguracja WINS

Windows Internet Name Service (WINS) to wersja serwera nazw NetBIOS-owych opracowana przez Microsoft. Jeśli włączyłeś usługę WINS w Sambie, musisz poinformować Windows o adresie serwera Samby. Jeśli używasz również serwerów WINS działających w komputerach Windows NT, podaj także ich adresy. Okno dialogowe wyświetlane po wybraniu karty Konfiguracja WINS jest pokazane na rysunku 3.11.

Nie podawaj serwera WINS Samby i serwera Windows NT jako kombinacji serwer podstawowy-serwer zapasowy w oknie dialogowym Konfiguracja WINS. Powstałyby błędy w odwzorowywaniu nazw, ponieważ serwery te nie mogą replikować swoich baz danych.

Zaznacz pole opcji Włącz rozpoznawanie WINS i wprowadź adres serwera WINS w odpowiednie pole, a następnie kliknij przycisk Dodaj. Nie wpisuj niczego w pole Identyfikator zakresu.



Rysunek 3.11. Karta Konfiguracja WINS

Plik hostów

Jeśli nie używasz DNS ani WINS i nie chcesz korzystać z rozgłoszeniowego odwzorowywania nazw, będziesz musiał utworzyć tabelę adresów IP i nazw hostów w standardowym formacie uniksowego pliku */etc/hosts*. W komputerach Windows tabelę tę należy umieścić w pliku \WINDOWS\HOSTS na tym dysku, na którym zainstalowałeś Windows (zwykle C:\). Oto przykładowy plik hostów:

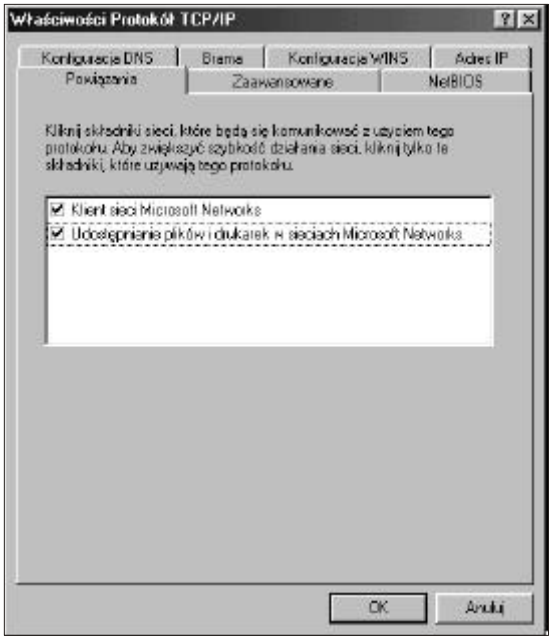
```
# 127.0.0.1             localhost
192.168.236.1       sztych.przyklad.com sztych
192.168.236.2      riposta.przyklad.com riposta
192.168.236.3       garda.przyklad.com garda
192.168.236.4       touche.przyklad.com touche
192.168.236.10     finta.przyklad.com finta
```

Możesz skopiować ten plik bezpośrednio z pliku */etc/hosts* dowolnego komputera uniksowego – format jest identyczny. Jednak do odwzorowywania nazw w Windows za pomocą pliku hostów powinieneś uciekać się *tylko w ostateczności*.

Sprawdzenie powiązań

Do sprawdzenia pozostała jeszcze karta Powiązania (patrz rysunek 3.12).

Pole obok pozycji Klient sieci Microsoft Networks powinno być zaznaczone, wskazując, że usługa klienta korzysta z TCP/IP. Jeśli w oknie dialogowym znajduje się pozycja Udostępnianie plików i drukarek w sieciach Microsoft Networks, również ona powinna być zaznaczona, jak pokazano na rysunku.



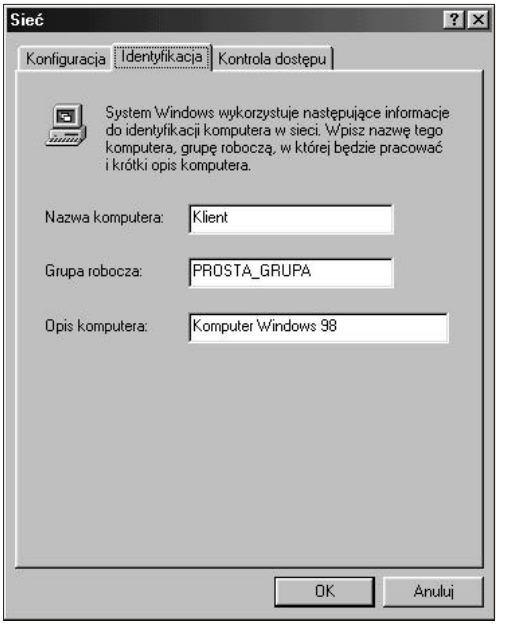
Rysunek 3.12. Karta Powiązania

Ustawianie nazwy komputera i grupy roboczej

Naciśnij teraz przycisk OK w oknie Właściwości Protokół TCP/IP, a wrócisz do okna Sieć. Wybierz kartę Identyfikacja, co spowoduje wyświetlenie okna dialogowego z rysunku 3.13.

Tutaj po raz drugi wprowadzisz nazwę komputera. Tym razem jednak określasz nie nazwę hosta i domenę DNS, ale NetBIOS-ową nazwę komputera. Najlepiej będzie wybrać taką samą nazwę, jak nazwa hosta DNS. Uważaj, żeby nie zrobić błędu w pisowni: konfigurowanie komputera może być kłopotliwe, kiedy TCP/IP myśli, że ma do czynienia z `fredem`, a SMB sądzi, że komputer nosi nazwę `ferd`!

W tym oknie określasz także nazwę grupy roboczej. W naszym przypadku będzie to PROSTA_GRUPA, ale jeśli w rozdziale 2, *Instalowanie Samby w Uniksie*, wybrałeś inną nazwę podczas tworzenia pliku konfiguracyjnego Samby, użyj jej także tutaj. Nie wybieraj nazwy WORKGROUP, gdyż w takim przypadku znajdziesz się w tej samej grupie, co wszystkie nieskonfigurowane (lub źle skonfigurowane) komputery na świecie.



Rysunek 3.13. Karta Identyfikacja

Uzyskiwanie dostępu do serwera Samby

Kliknij przycisk OK, aby zakończyć konfigurację. Będziesz musiał ponownie uruchomić komputer, aby zmiany zostały uwzględnione.

Nadeszła wielka chwila. Serwer Samby działa, a ty skonfigurowałeś klienta Windows 95/98 do współpracy z nim. Po ponownym uruchomieniu Windows zaloguj się i kliknij dwukrotnie ikonę Otoczenia sieciowego na pulpicie. Powinieneś zobaczyć serwer Samby jako członka swojej grupy roboczej (patrz rysunek 3.14).

Rysunek 3.14. Otoczenie sieciowe w Windows

Dwukrotne kliknięcie nazwy serwera spowoduje wyświetlenie zasobów udostępnianych przez niego w sieci, co pokazano na rysunku 3.15 (w tym przypadku jest to drukarka i katalog *test*).

Rysunek 3.15. Udziały w serwerze

Jeśli pojawi się okno dialogowe z prośbą o podanie hasła dostępu do zasobu `IPC$`, oznacza to, że Samba nie zaakceptowała hasła przesłanego przez klienta. Pamiętaj, że nazwa użytkownika i hasło ustawione w kliencie *musi* odpowiadać nazwie użytkownika i hasłu w serwerze Samby. Jeśli używasz Windows 98 lub Windows NT z Service Pack 3 lub nowszą wersją poprawek, prawdopodobnie dzieje się tak dlatego, że klient wysyła hasło w postaci zaszyfrowanej, a nie otwartym tekstem. Możesz uporać się z tym problemem, wykonując dwie czynności w serwerze Samby. Po pierwsze, do sekcji `[global]` w pliku konfiguracyjnym Samby dodaj wpis `encrypt passwords=yes`. Po drugie, znajdź program *smbpasswd* w serwerze Samby (domyślnie jest on przechowywany w katalogu */usr/local/samba/bin*) i dodaj za jego pomocą nowy wpis do bazy zaszyfrowanych haseł Samby. Aby na przykład dodać użytkownika `stefan`, do bazy wpisz `smbpasswd -a stefan`. Kiedy będziesz wprowadzał pierwsze hasło, program wyświetli komunikat o błędzie, informując, że baza haseł nie istnieje, a następnie utworzy bazę danych (zwykle w katalogu */usr/local/samba/private/smbpasswd*).

Jeśli serwer nie pojawi się na liście, uruchom Eksploratora Windows (nie Internet Explorera!) i wybierz polecenie Mapuj dysk sieciowy z menu Narzędzia. Pojawi się okno dialogowe, w którym będziesz mógł wpisać nazwę serwera i udziału `test` w notacji UNC Windows: `\\serwer\test`, jak robiliśmy to w pierwszym rozdziale. Powinno to doprowadzić do połączenia z serwerem Samby i naszym tymczasowym udziałem. Jeśli to również nie zadziała, zajrzyj do rozdziału 9, *Rozwiązywanie problemów*, gdzie znajdziesz wskazówki dotyczące diagnozowania błędów.

Konfigurowanie komputerów Windows NT 4.0

Konfigurowanie Windows NT przebiega nieco odmiennie od konfigurowania Windows 95/98. Do używania Samby z Windows NT będziesz potrzebował zarówno usługi stacji roboczej, jak i protokołu TCP/IP. Oba te składniki są standardowymi elementami Windows NT, ale opiszemy ich instalowanie i konfigurowanie, ponieważ mogą być skonfigurowane nieprawidłowo.

Oto sześć głównych etapów:

1. Nadanie nazwy komputerowi.
2. Zainstalowanie usługi stacji roboczej.
3. Zainstalowanie protokołu TCP/IP.
4. Ustawienie nazwy i adresu IP komputera.
5. Skonfigurowanie usług DNS i WINS.
6. Powiązanie protokołu i usług.

Podstawowa konfiguracja

W tym podrozdziale przedstawimy czynności, które umożliwią współpracę Windows NT i Samby. Jeśli potrzebujesz więcej informacji o administrowaniu systemem Windows NT, zajrzyj do doskonałej książki Craiga Hunta i Roberta Bruce'a Thompsona *Windows NT TCP/IP – administracja systemu*, wydanej przez Wydawnictwo RM. Wszystkie opisane czynności powinieneś wykonać jako użytkownik „Administrator".

Nadawanie nazwy komputerowi

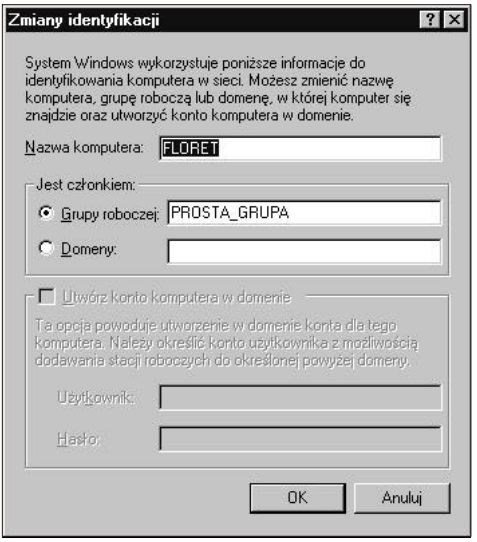
Najpierw musisz nadać komputerowi NetBIOS-ową nazwę. Kliknij dwukrotnie ikonę Sieć w Panelu sterowania. Na ekranie pojawi się okno dialogowe Sieć. Pierwszą kartą w tym oknie powinna być karta Identyfikacja, jak na rysunku 3.16.

Rysunek 3.16. Karta Identyfikacja okna dialogowego Sieć

Tutaj musisz zidentyfikować swój komputer, nadając mu nazwę (my użyliśmy nazwy Floret) i zmienić domyślną nazwę grupy roboczej na taką, jaką podałeś w pliku *smb.conf* w serwerze Samby. W tym przypadku nazwa grupy roboczej to PROSTA_GRUPA. Nie możesz jednak zmienić obu nazw na tej karcie (jak w Windows 95/98). Musisz najpierw kliknąć przycisk Zmień pod polami tekstowymi, co spowoduje wyświetlenie okna dialogowego Zmiany identyfikacji, w którym możesz podać nową nazwę komputera i grupy roboczej (patrz rysunek 3.17).

Słowo ostrzeżenia: będziesz musiał ponownie ustalić nazwę komputera podczas konfigurowania TCP/IP, więc upewnij się, że obie nazwy są takie same. Nazwa, którą ustawiasz tutaj, to nazwa NetBIOS-owa. Może ona się różnić od nazwy hosta TCP/IP, ale z reguły nie jest to pożądane. Nie przejmuj się, że Windows NT wymu-

sza użycie dużych liter w nazwie komputera i grupy; system będzie wiedział, o co ci chodziło, kiedy włączy się do sieci.



Rysunek 3.17. Zmienianie danych identyfikacyjnych

Instalowanie protokołu TCP/IP

Następnie wybierz kartę Protokoły w oknie dialogowym Sieć i sprawdź, czy masz zainstalowany protokół TCP/IP (patrz rysunek 3.18).

Rysunek 3.18. Karta Protokoły

Jeśli protokół nie jest zainstalowany, będziesz musiał go dodać. Kliknij przycisk Dodaj, co spowoduje wyświetlenie okna dialogowego Wybierz: Protokół sieciowy (patrz rysunek 3.19). W przeciwieństwie do Windows 95/98, powinieneś od razu zobaczyć protokół TCP/IP na dole listy.

Rysunek 3.19. Okno dialogowe Wybierz: Protokół sieciowy

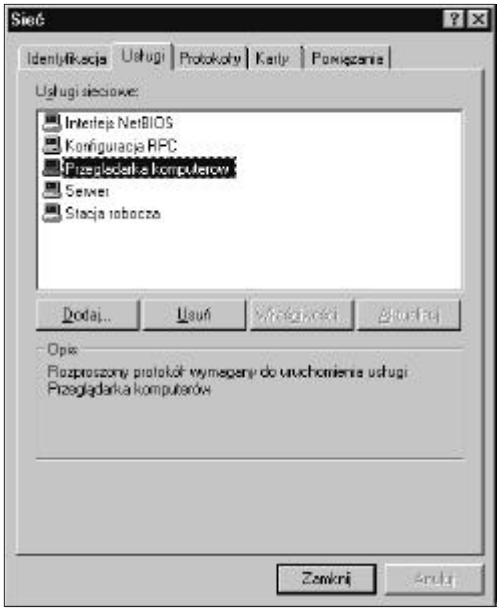
Zaznacz protokół TCP/IP i potwierdź swój wybór. Jeśli to możliwe, zainstaluj tylko TCP/IP. Zwykle nie ma sensu instalować NetBEUI, ponieważ w takim przypadku komputer szuka usług, korzystając z dwóch różnych protokołów, z których zapewne tylko jeden jest używany*.

Instalowanie usługi stacji roboczej

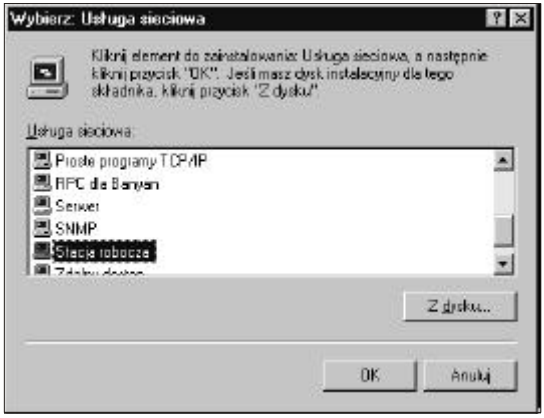
Po zainstalowaniu TCP/IP kliknij kartę Usługi w oknie dialogowym Sieć i sprawdź, czy masz zainstalowaną usługę stacji roboczej, którą widać na dole listy z rysunku 3.20.

Ta usługa to w istocie klient sieci Microsoft, który pozwala komputerowi na korzystanie z zasobów SMB. Usługa stacji roboczej jest niezbędna. Jest ona instalowana domyślnie zarówno w Windows NT Workstation 4.0, jak i w Server 4.0. Jeśli nie ma jej na liście, możesz zainstalować ją niemal tak samo, jak protokół TCP/IP. Tym razem będziesz musiał kliknąć przycisk Dodaj i zaznaczyć usługę Stacja robocza, jak pokazano na rysunku 3.21.

* Często spotykany objaw: po chwili sprawdzania nieużywanego protokołu upływa limit czasu, po czym komputer próbuje użyć tego właściwego. To bezproduktywne wyszukiwanie jest przyczyną obniżenia wydajności i tajemniczych opóźnień.



Rysunek 3.20. Karta Usługi w oknie dialogowym Sieć



Rysunek 3.21. Okno dialogowe Wybierz: Usługa sieciowa

Konfigurowanie TCP/IP

Po zainstalowaniu usługi stacji roboczej wróć do karty Protokoły i zaznacz na liście pozycję Protokół TCP/IP. Następnie kliknij umieszczony pod listą przycisk Właściwości. Pojawi się okno dialogowe Właściwości: Microsoft TCP/IP. Okno to zawiera pięć kart, z których trzy wymagają zmodyfikowania:

- Adres IP,
- DNS,
- Adres WINS.

Karta Adres IP

Karta Adres IP jest pokazana na rysunku 3.22.

Rysunek 3.22. Okno dialogowe Właściwości: Microsoft TCP/IP w Windows NT

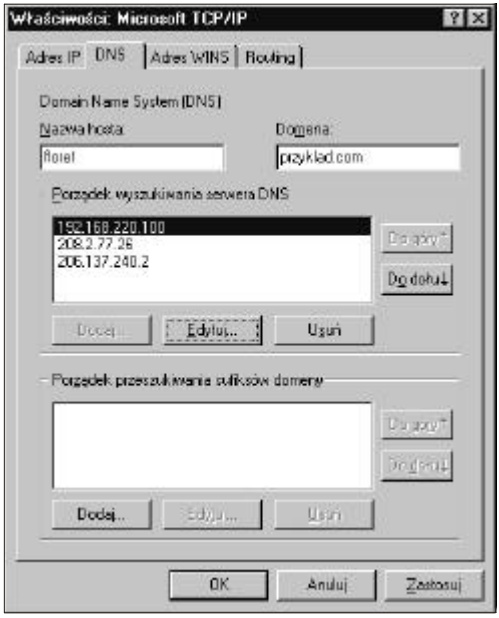
Zaznacz pole opcji Podaj adres IP i wpisz adres oraz maskę podsieci dla odpowiedniego interfejsu sieciowego (karty Ethernetu). Ty albo administrator sieci powinniście wcześniej wybrać adres komputera tak, aby znajdował się w tej samej podsieci, co serwer Samby. Jeśli na przykład adres serwera to 192.168.236.86, a jego maska podsieci to 255.255.255.0, mógłbyś przypisać komputerowi Windows NT Workstation adres 192.168.236.10 (jeśli jest wolny) i taką samą maskę podsieci. Jeśli w swojej sieci używasz serwera DHCP, zaznacz pole opcji Uzyskaj adres IP z serwera DHCP.

Jeśli nie masz przydzielonego adresu IP, a twoja sieć nie jest połączona z Internetem, użyj naszego adresu. Podsieć 192.168.*x.x* jest specjalnie zarezerwowana przez organizację InterNIC na potrzeby prywatnych sieci lokalnych. W przeciwnym wypadku skontaktuj się z administratorem sieci i dowiedz się, jakie adresy są dostępne.

Pole Brama domyślna odnosi się do komputera nazywanego zwykle *ruterem*. Jeśli masz pod opieką kilka sieci połączonych ruterami, powinieneś umieścić tu adres tego rutera, który znajduje się w twojej podsieci.

Karta DNS

Teraz przejdź do karty DNS, pokazanej na rysunku 3.23.



Rysunek 3.23. Karta DNS

Domain Name System (DNS) to usługa odpowiedzialna za tłumaczenie używanych przez ludzi nazw komputerów, takich jak *floret.przyklad.com*, na adresy IP, takie jak 192.168.236.10. Komputer Windows NT może dokonywać takiego tłumaczenia na dwa sposoby. Możesz podać adres serwera DNS, który będzie zajmował się odwzorowywaniem nazw na adresy, albo utworzyć lokalną listę par nazwa-adres, do których będzie się odwoływać twoja stacja robocza.

W przypadku sieci lokalnej nie podłączonej do Internetu, lista jest zwykle krótka i może być przechowywana w lokalnym pliku. Sieci podłączone do Internetu korzystają zwykle z usług DNS, ponieważ nie da się z góry przewidzieć, z którymi adresami internetowymi będą łączyć się użytkownicy. Jeśli nie jesteś pewien, czy sieć korzysta z serwera DNS albo jaki jest jego adres, zajrzyj do pliku */etc/resolv.conf* w serwerze Samby: wszystkie komputery używające DNS będą miały taki plik. Wygląda on następująco:

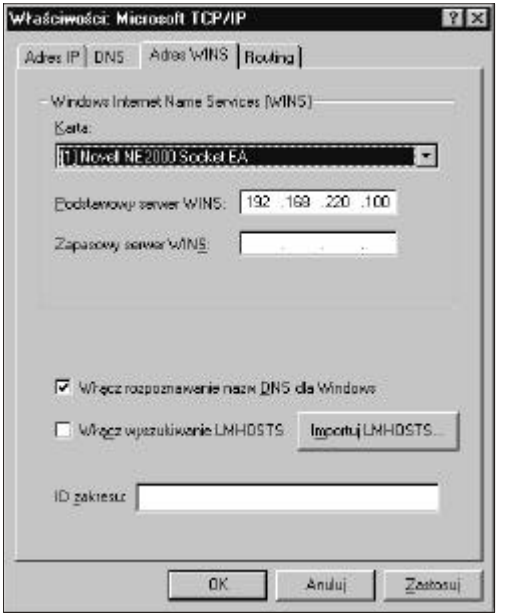
```
#resolv.conf
domain przyklad.com
nameserver 127.0.0.1
nameserver 192.168.236.20
```

W tym przykładzie pierwsza linia `nameserver` zawiera adres 127.0.0.1, co wskazuje, że serwer Samby jest zarazem serwerem DNS dla tej sieci lokalnej*. W takim przypadku powinieneś użyć sieciowego adresu serwera Samby (a nie jego adresu lokalnego hosta, 127.0.0.1) podczas wypełniania karty DNS. Możesz użyć także innych adresów wymienionych w liniach zaczynających się od słowa `nameserver`. Spróbuj wybrać takie, które znajdują się w twojej sieci. Działać będą wszystkie serwery wymienione w pliku */etc/resolv.conf*, ale korzystając z pobliskiego serwera osiągniesz lepszą wydajność.

Teraz musisz wprowadzić jeszcze raz nazwę komputera, upewniając się, że jest ona taka sama jak ta, która widnieje na karcie Identyfikacja okna dialogowego Sieć (nazwa NetBIOS-owa). Wpisz także nazwę domeny DNS, w której rezyduje komputer. Jeśli nazwą domeny jest *przyklad.com*, wpisz ją tutaj. Możesz zignorować pozostałe opcje.

Karta Adres WINS

Jeśli nie używasz serwera DNS, będziesz i tak potrzebował mechanizmu tłumaczenia nazw NetBIOS-owych na adresy i odwrotnie. Zalecamy skonfigurowanie zarówno DNS, jak i WINS. Windows NT preferuje użycie WINS, a WINS może korzystać z DNS w sytuacji awaryjnej, kiedy nie potrafi odwzorować nazwy któregoś komputera. Karta Adres WINS jest pokazana na rysunku 3.24.



Rysunek 3.24. Karta Adres WINS

* Adres 127.0.0.1 jest znany jako adres *lokalnego hosta*, a za jego pomocą komputer zawsze odwołuje się do samego siebie. Jeśli na przykład wpiszesz `ping 127.0.0.1` na uniksowym serwerze, powinieneś zawsze otrzymać odpowiedź, ponieważ sprawdzasz własny komputer.

Jeśli w twojej sieci jest serwer WINS, wpisz jego adres w pole oznaczone napisem Podstawowy serwer WINS. Jeśli usługi WINS świadczy Samba (innymi słowy, w pliku *smb.conf* serwera Samby znajduje się linia `wins service = yes`), wpisz tutaj adres IP serwera Samby. W przeciwnym wypadku podaj adres innego serwera WINS w twojej sieci.

Prawdopodobnie zauważyłeś pole umożliwiające wybór karty sieciowej. Musi w nim widnieć nazwa tej karty, z którą powiązany jest protokół TCP/IP, aby komputer korzystał z usług nazewniczych WINS we właściwej sieci. Jeśli masz zarówno kartę sieciową, jak i modemową, upewnij się, że w polu tym widnieje nazwa karty sieciowej.

Na zakończenie zaznacz pole wyboru Włącz rozpoznawanie nazw DNS dla Windows, aby usługa WINS korzystała z pomocy DNS, jeśli nie zdoła odszukać nazwy. Możesz zignorować pozostałe opcje.

Plik hostów

Jeśli nie używasz DNS ani WINS i nie chcesz korzystać z rozgłoszeniowego odwzorowywania nazw, będziesz musiał utworzyć tabelę adresów IP i nazw hostów w standardowym formacie uniksowego pliku */etc/hosts*. Odradzamy takie rozwiązanie, ponieważ uaktualnianie takiego pliku w dynamicznej sieci jest mocno kłopotliwe, ale mimo to je opiszemy. Plik HOSTS w Windows powinien znajdować się w katalogu \WINDOWS na tym dysku, na którym zainstalowano system. Oto przykład:

```
127.0.0.1       localhost
192.168.236.1    sztych     sztych.przyklad.com
192.168.236.2    riposta    riposta.przyklad.com
192.168.236.3    garda      garda.przyklad.com
192.168.236.4    touche     touche.przyklad.com
192.168.236.5    floret     floret.przyklad.com
192.168.236.6    szabla     szabla.przyklad.com
192.168.236.7    rapier     rapier.przyklad.com
```

Możesz skopiować ten plik bezpośrednio z pliku */etc/hosts* serwera Samby – format jest identyczny. Będzie on spełniał tę samą funkcję, co plik hostów w uniksowym serwerze. Jednakże pliku hostów w Windows należy używać tylko w ostateczności.

Powiązania

Termin *powiązany* oznacza mniej więcej tyle, co „połączony z czymś w czasie konfiguracji". Oznacza on, że protokół TCP/IP będzie przepływał przez kartę Ethernetu (a nie na przykład przez połączenie modemowe) i że jest poprawnie skonfigurowany. Jeśli klikniesz kartę Powiązania okna dialogowego Sieć, wybierzesz z listy opcję wszystkie usługi i klikniesz znaki plus w drzewie usług, powinieneś zobaczyć konfigurację taką jak na rysunku 3.25.

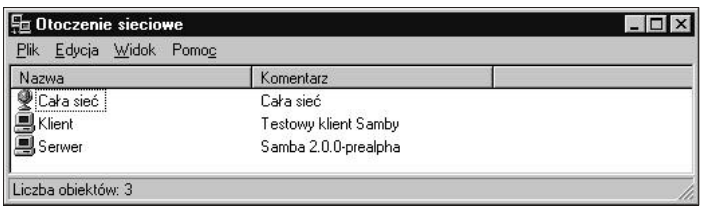
Oznacza ona, że usługi stacji roboczej, serwera i interfejsu NetBIOS są połączone z klientem WINS. Są to poprawne powiązania dla protokołu Microsoft TCP/IP.

Rysunek 3.25. Powiązania usług

Łączenie się z serwerem Samby

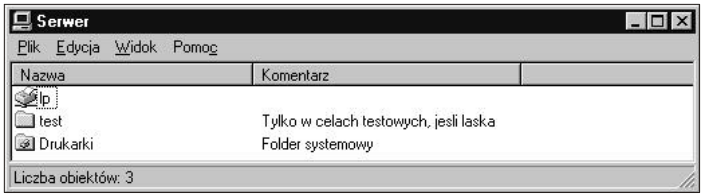
Na innych kartach okna dialogowego Sieć możesz pozostawić wartości domyślne. Kliknij przycisk OK, aby zakończyć konfigurację. Po załadowaniu potrzebnych plików (jeśli będzie to konieczne) będziesz musiał ponownie uruchomić komputer, aby zmiany zostały uwzględnione.

Nadeszła wielka chwila. Serwer Samby działa, a ty skonfigurowałeś klienta Windows NT do współpracy z nim. Po ponownym uruchomieniu Windows zaloguj się i kliknij dwukrotnie ikonę Otoczenia sieciowego na pulpicie. Powinieneś zobaczyć serwer Samby jako członka swojej grupy roboczej (patrz rysunek 3.26).



Rysunek 3.26. Otoczenie sieciowe w Windows NT

Dwukrotne kliknięcie nazwy serwera spowoduje wyświetlenie zasobów udostępnianych przez niego w sieci, co pokazano na rysunku 3.27 (w tym przypadku jest to drukarka i katalog *test*). Więcej informacji znajdziesz w ostrzeżeniu w podrozdziale „Uzyskiwanie dostępu do serwera Samby" wcześniej w tym rozdziale.



Rysunek 3.27. Udziały w serwerze

Jeśli pojawi się okno dialogowe z prośbą o podanie hasła dostępu do zasobu `IPC$`, oznacza to, że Samba nie zaakceptowała hasła przesłanego przez klienta. Pamiętaj, że nazwa użytkownika i hasło ustawione w kliencie *musi* odpowiadać nazwie użytkownika i hasłu w serwerze Samby. Jeśli używasz Windows 98 lub Windows NT z Service Pack 3 lub nowszą wersją poprawek, prawdopodobnie dzieje się tak dlatego, że klient wysyła hasło w postaci zaszyfrowanej, a nie otwartym tekstem. Możesz uporać się z tym problemem, wykonując dwie czynności w serwerze Samby. Po pierwsze, do sekcji `[global]` w pliku konfiguracyjnym Samby dodaj wpis `encrypt passwords=yes`. Po drugie, znajdź program *smbpasswd* w serwerze Samby (domyślnie jest on przechowywany w katalogu */usr/local/samba/bin*) i dodaj za jego pomocą nowy wpis do bazy zaszyfrowanych haseł Samby. Aby na przykład dodać do bazy użytkownika `stefan`, wpisz `smbpasswd -a stefan`. Kiedy będziesz wprowadzał pierwsze hasło, program wyświetli komunikat o błędzie, informując, że baza haseł nie istnieje, a następnie utworzy bazę danych (zwykle w katalogu */usr/local/samba/private/smbpasswd*).

Jeśli serwer nie pojawi się na liście, nie wpadaj w panikę. Uruchom Eksploratora Windows NT (nie Internet Explorera!) i wybierz polecenie Mapuj dysk sieciowy z menu Narzędzia. Pojawi się okno dialogowe, w którym będziesz mógł wpisać w notacji UNC nazwę serwera i jego udziału. Jeśli twój serwer ma nazwę „serwer", wpisałbyś tutaj `\\serwer\test`. Jeśli to również nie zadziała, zajrzyj do podrozdziału „Drzewo błędów" w rozdziale 9 i sprawdź, czy pomoże ci on w zdiagnozowaniu problemu.

Jeśli operacja się powiodła, gratulujemy! Spróbuj zapisać plik w serwerze i przesłać dane do drukarki sieciowej. Będziesz miło zdziwiony, że wszystko działa tak gładko. Teraz, kiedy skonfigurowałeś już Sambę i jej klienty, możemy omówić działanie Samby i zacząć dostosowywać ją do twoich potrzeb.

Wprowadzenie do protokołu SMB/CIFS

Zakończymy ten rozdział krótkim przewodnikiem po protokole SMB/CIFS. Właśnie tego protokołu używają komputery Windows 95/98 i NT do porozumiewania się z serwerem Samby i między sobą. Z perspektywy korzystających z niego aplikacji zestaw protokołów SMB jest względnie prosty. Zawiera polecenia dla wszystkich operacji, które wykonuje się na lokalnych plikach i drukarkach, między innymi:

- otwierania i zamykania plików,
- tworzenia i usuwania plików i katalogów,
- czytania i zapisywania plików,
- wyszukiwania plików,
- kolejkowania plików w buforze wydruku i usuwania ich z kolejki.

Każdą z tych operacji można zakodować w komunikacie SMB i przesłać do serwera. Sama nazwa protokołu pochodzi od formatu komunikatów: są one sieciową wersją standardowych struktur danych używanych w wywołaniach systemowych DOS-a, czyli *blokami komunikatów serwera*, przystosowanymi do transmisji między komputerami w sieci.

Format komunikatu SMB

Richard Sharpe z zespołu Samby definiuje SMB jako protokół typu żądanie-odpowiedź*. Oznacza to, że klient wysyła żądanie do serwera, a serwer odsyła odpowiedź SMB klientowi. Serwer rzadko wysyła komunikaty nie będące odpowiedzią na żądanie klienta.

Komunikat SMB nie jest szczególnie skomplikowany. Przyjrzyjmy się bliżej jego wewnętrznej strukturze. Można podzielić go na dwie części: *nagłówek* o stałej wielkości i *łańcuch polecenia*, którego wielkość może być bardzo różna, w zależności od treści komunikatu.

Format nagłówka SMB

Tabela 3.1 przedstawia format nagłówka SMB. Polecenia SMB nie muszą wykorzystywać wszystkich pól nagłówka. Kiedy na przykład klient próbuje połączyć się z serwerem, nie dysponuje jeszcze tak zwanym identyfikatorem drzewa (*tree identifier*, TID) – zostanie on mu przyznany dopiero po nawiązaniu połączenia – więc w odpowiednim polu nagłówka klient umieszcza pusty TID (0xFFFF). Inne pola, kiedy nie są używane, mogą być wypełnione zerami.

Pola nagłówka SMB są wymienione w tabeli 3.1.

Tabela 3.1. Pola nagłówka SMB

Pole	*Rozmiar (bajty)*	*Opis*
`0xFF 'SMB'`	1	Identyfikator protokołu
`COM`	1	Kod polecenia, od 0x00 do 0xFF
`RCLS`	1	Klasa błędu
`REH`	1	Zarezerwowane
`ERR`	2	Kod błędu
`REB`	1	Zarezerwowane
`RES`	14	Zarezerwowane
`TID`	2	Identyfikator drzewa; niepowtarzalny identyfikator zasobu używanego przez klienta
`PID`	2	Identyfikator wywołującego procesu
`UID`	2	Identyfikator użytkownika
`MID`	2	Identyfikator multipleksowy; używany do kojarzenia różnych żądań i odpowiedzi w ramach jednego procesu

* Pod adresem *http://anu.samba.org/cifs/docs/what-is-smb.html* znajduje się doskonały opis SMB autorstwa Richarda.

Format poleceń SMB

Zaraz za nagłówkiem następuje zmienna liczba bajtów tworzących polecenie lub odpowiedź SMB. Każde polecenie, takie jak Open File (otwórz plik, identyfikator w polu COM: `SMBopen`) albo Get Print Queue (pobierz kolejkę wydruku, `SMBsplretq`), ma własny zestaw parametrów i danych. Podobnie jak pola nagłówka SMB, nie wszystkie pola polecenia muszą być wypełnione, zależy to od konkretnego polecenia. Na przykład polecenie Get Server Attributes (pobierz atrybuty serwera, `SMBdskattr`) ustawia pola WCT i BCC na zero. Pola segmentu polecenia są wymienione w tabeli 3.2.

Tabela 3.2. Zawartość poleceń SMB

Pole	*Rozmiar (bajty)*	*Opis*
`WCT`	1	Liczba słów
`VWV`	zmienny	Słowa parametrów (ich łączny rozmiar określa pole WCT)
`BCC`	2	Liczba bajtów danych
`DATA`	zmienny	Dane (ich łączny rozmiar określa pole BCC)

Nie martw się, jeśli nie rozumiesz znaczenia niektórych pól; nie jest to potrzebne do używania Samby na poziomie administracyjnym. Ich znajomość może się jednak przydać podczas przeglądania komunikatów systemowych. Dalej w tym rozdziale pokażemy komunikaty SMB wysyłane przez klienty i serwery, przechwycone za pomocą zmodyfikowanej wersji programu *tcpdump*. (Jeśli wolałbyś używać sniffera z interfejsem graficznym, wypróbuj program „ethereal", korzystający z bibliotek GTK. Więcej informacji o tym narzędziu znajduje się na stronie głównej Samby).

Jeśli chciałbyś dowiedzieć się więcej o poszczególnych poleceniach protokołu SMB, zajrzyj do dokumentacji SMB/CIFS pod adresem *ftp://ftp.microsoft.com/developr/drg/CIFS/*.

Odmiany SMB

Od swojego powstania protokół SMB był kilkakrotnie rozszerzany o nowe polecenia. Każda nowa wersja jest zgodna wstecz ze starszymi, co sprawia, że w sieciach lokalnych często można spotkać różne klienty i serwery posługujące się odmiennymi wersjami protokołu SMB.

W tabeli 3.3 wymienione są główne wersje protokołu SMB. Każdy „dialekt" SMB może mieć wersje poboczne, które zawierają polecenia obsługujące poszczególne wydania systemu operacyjnego. Łańcuch identyfikacyjny jest stosowany przez klienty i serwery do uzgodnienia wariantu protokołu, który będzie używany do wzajemnej komunikacji.

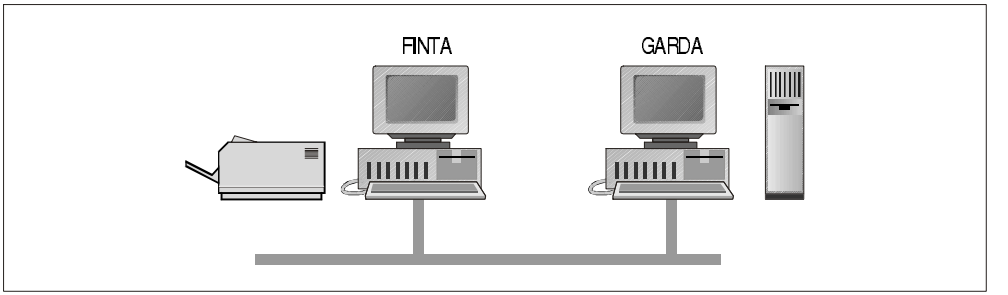
Samba używa specyfikacji `NT LM 0.12` dla NT LAN Managera 1.0. Jest ona zgodna z wszystkimi wcześniejszymi wersjami SMB. Specyfikacja CIFS to w istocie LAN Manager 0.12 z kilkoma specyficznymi dodatkami.

Tabela 3.3. Dialekty protokołu SMB

Nazwa protokołu	*Łańcuch identyfikacyjny*	*Używany przez*
Core	`PC NETWORK PROGRAM 1.0`	
Core Plus	`MICROSOFT NETWORKS 1.03`	
LAN Manager 1.0	`LANMAN1.0`	
LAN Manager 2.0	`LM1.2X002`	
LAN Manager 2.1	`LANMAN2.1`	
NT LAN Manager 1.0	`NT LM 0.12`	Windows NT 4.0
Samba NT LM 0.12	`Samba`	Sambę
Common Internet File System	`CIFS 1.0`	Windows 2000

Klienty i serwery SMB

Jak już wspomniano, SMB to protokół typu klient-serwer. Zgodnie z klasyczną definicją oznacza to, że klient wysyła żądanie do serwera, który przetwarza żądanie i zwraca odpowiedź klientowi. Jednak role klienta i serwera mogą zostać odwrócone, czasem w kontekście jednej sesji SMB. Rozważmy na przykład dwa komputery Windows 95/98 z rysunku 3.28. Komputer o nazwie FINTA udostępnia w sieci drukarkę, a komputer o nazwie GARDA udostępnia katalog dysku. FINTA występuje w roli klienta, kiedy korzysta ze stacji sieciowej GARDY, a w roli serwera, kiedy wykonuje zadanie wydruku na rzecz FINTY.



Rysunek 3.28. Dwa komputery udostępniające swoje zasoby

Doszliśmy w ten sposób do ważnej kwestii w terminologii Samby:

- *Serwer* to komputer udostępniający współdzielony zasób.
- *Klient* to komputer, który używa tego zasobu.
- Serwer może w każdym momencie być klientem (używać zasobu innego komputera).

Zauważ, że nie można określić żadnej granicy, od której komputer stawałby się serwerem – nieistotna jest ilość udostępnianych zasobów, wielkość dysku czy szybkość procesora. Serwerem może więc być stary komputer 486 z dołączoną drukarką albo nowoczesna stacja UltraSparc z dziesięcioma gigabajtami udostępnianej przestrzeni dyskowej.

Systemy operacyjne Microsoft Windows zawierają wbudowane programy klienta i serwera SMB. Windows NT 4.0 używa nowszego protokołu SMB niż Windows for Workgroups i zawiera ulepszone mechanizmy bezpieczeństwa sieciowego, o których będziemy mówić w rozdziale 6. Oprócz tego istnieje wiele komercyjnych wersji serwera SMB, opracowanych przez firmy, takie jak Sun, Compaq, SCO, Hewlett-Packard, Syntax i IBM. Niestety, po stronie klienta wybór jest dużo mniejszy – w grę wchodzi właściwie tylko pakiet Pathworks firmy Digital Equipment i, oczywiście, Samba.

Proste połączenie SMB

Zanim przejdziemy do następnego rozdziału, przyjrzyjmy się prostemu połączeniu SMB. Są to zagadnienia natury technicznej – właściwie niepotrzebne do zarządzania Sambą – więc jeśli chcesz, możesz pominąć ten fragment. Zamieszczamy tutaj te informacje głównie po to, aby zaznajomić cię ze sposobem, w jaki protokół SMB negocjuje połączenia z innymi komputerami w sieci.

Połączenie między klientem a zasobem serwera odbywa się w czterech etapach:

1. Nawiązanie połączenia wirtualnego.
2. Negocjacja odmiany protokołu.
3. Ustawienie parametrów sesji.
4. Nawiązanie połączenia z zasobem.

Zbadamy wszystkie cztery etapy za pomocą przydatnego narzędzia, o którym wspomnieliśmy wcześniej: zmodyfikowanej wersji programu *tcpdump* dostępnej na witrynie WWW Samby.

Program ten znajdziesz w serwerze *samba.org* w katalogu *samba/ftp/tcpdumb-smb*. W czasie pisania tej książki najnowsza wersja miała numer 3.4-5. Możesz korzystać z niego tak, jak ze zwykłego programu *tcpdump*, ale dodaj opcję `-s 1500`, aby przechwytywać całe pakiety, a nie tylko kilka pierwszych bajtów.

Nawiązywanie połączenia wirtualnego

Kiedy użytkownik wysyła pierwsze żądanie dostępu do dysku sieciowego lub zdalnej drukarki, NetBIOS jest odpowiedzialny za nawiązanie połączenia w warstwie sesji. W rezultacie powstaje dwukierunkowy kanał komunikacyjny między klientem i serwerem. W rzeczywistości klient i serwer potrzebują tylko dwóch komunikatów do nawiązania połączenia. Widać to w poniższym przykładzie żądania nawiązania sesji i odpowiedzi na to żądanie, przechwyconych przez program *tcpdump*:

```
>>> NBT Packet
NBT Session Request
Flags=0x81000044
Destination=FINTA       NameType=0x20 (Server)
Source=GARDA            NameType=0x00 (Workstation)

>>> NBT Packet
NBT Session Granted
Flags=0x82000000
```

Negocjowanie wariantu protokołu

W tym momencie istnieje już otwarty kanał między klientem i serwerem. Następnie klient wysyła do serwera komunikat negocjujący wariant protokołu SMB. Jak wspomniano wcześniej, klient ustawia swój identyfikator drzewa (TID) na zero, ponieważ nie wie jeszcze, jakiej wartości TID powinien użyć. *Identyfikator drzewa* to numer reprezentujący połączenie z udziałem w serwerze.

Polecenie zawarte w komunikacie to `SMBnegprot`, żądanie negocjacji wariantu protokołu, który będzie używany w trakcie całej sesji. Zauważ, że to klient wysyła do serwera listę wszystkich rozumianych przez siebie wariantów protokołu, a nie odwrotnie.

Serwer odpowiada na żądanie `SMBnegprot` indeksem do listy wariantów nadesłanych przez klienta (pierwszy wariant ma indeks 0) albo wartością 0xFF, jeśli nie akceptuje żadnego wariantu. W tym przykładzie serwer zwraca indeks 5, co oznacza, że podczas tej sesji będzie używany dialekt `NT LM 0.12`.

```
>>> NBT Packet
NBT Session Packet
Flags=0x0
Length=154

SMB PACKET: SMBnegprot (REQUEST)
SMB Command    =  0x72
Error class    =  0x0
Error code     =  0
Flags1         =  0x0
Flags2         =  0x0
Tree ID        =  0
Proc ID        =  5371
UID            =  0
MID            =  385
Word Count     =  0
Dialect=PC NETWORK PROGRAM 1.0
Dialect=MICROSOFT NETWORKS 3.0
Dialect=DOS LM1.2X002
Dialect=DOS LANMAN2.1
Dialect=Windows for Workgroups 3.1a
Dialect=NT LM 0.12

>>> NBT Packet
NBT Session Packet
Flags=0x0
Length=69

SMB PACKET: SMBnegprot (REPLY)
SMB Command    =  0x72
Error class    =  0x0
Error code     =  0
Flags1         =  0x0
Flags2         =  0x1
Tree ID        =  0
Proc ID        =  5371
UID            =  0
MID            =  385
Word Count     =  02
[000] 05 00
```

Ustawianie parametrów sesji i logowania

W kolejnym etapie przesyłane są parametry sesji i logowania. Jest to nazwa konta i hasło (jeśli jakieś jest), nazwa grupy roboczej, maksymalny rozmiar transmitowanych danych i liczba zaległych żądań, które jednocześnie mogą być buforowane w kolejce.

Przedstawione w poniższym przykładzie polecenie Session Setup (konfiguruj sesję) pozwala na dołączenie do niego kolejnego polecenia SMB. Wskazuje na to litera X na końcu nazwy polecenia; szesnastkowy kod drugiego polecenia jest podany w polu `Com2`. W tym przypadku drugim poleceniem jest `0x75`, czyli Tree Connect and X (połącz z drzewem i X). Polecenie `SMBtconX` szuka nazwy zasobu w buforze *smb_buf* (jest to ostatnie pole żądania). W tym przykładzie bufor *smb_buf* zawiera łańcuch `\\FINTA\PUBL`, co jest pełną ścieżką do współdzielonego katalogu w komputerze FINTA. Takie użycie poleceń „i X" przyspiesza wszystkie transakcje, ponieważ serwer nie musi czekać, aż klient wyśle drugie żądanie.

Zauważ, że pole TID nadal zawiera zero. Serwer dostarczy identyfikator klientowi dopiero po otwarciu sesji i nawiązaniu połączenia z żądanym zasobem. Zwróć też uwagę, że hasło jest przesyłane otwartym tekstem. Zmienimy to później, włączając obsługę zaszyfrowanych haseł.

```
>>> NBT Packet
NBT Session Packet
Flags=0x0
Length=139

SMB PACKET: SMBsesssetupX (REQUEST)
SMB Command   =  0x73
Error class   =  0x0
Error code    =  0
Flags1        =  0x10
Flags2        =  0x0
Tree ID       =  0
Proc ID       =  5371
UID           =  1
MID           =  385
Word Count    =  13
Com2=0x75
Res1=0x0
Off2=106
MaxBuffer=2920
MaxMpx=2
VcNumber=0
SessionKey=0x1FF2
CaseInsensitivePasswordLength=1
CaseSensitivePasswordLength=1
Res=0x0
Capabilities=0x1
Pass1&Pass2&Account&Domain&OS&LanMan=
KRISTIN PARKSTR Windows 4.0 Windows 4.0

PassLen=2
Passwd&Path&Device=
smb_bcc=22
smb_buf[]=\\FINTA\PUBL
```

Nawiązywanie połączenia z zasobem

W końcowym etapie serwer przesyła klientowi identyfikator TID, wskazując tym samym, że klient został uwierzytelniony, a zasób jest gotowy do użycia. Ustawia także pole *ServiceType* na „A", co oznacza, że jest to usługa plikowa. Dostępne typy usług to:

- „A": dysk lub plik,
- „LPT1": buforowana drukarka,
- „COMM": bezpośrednio dołączona drukarka lub modem,
- „IPC": nazwany potok.

Pakiet wygląda następująco:

```
>>> NBT Packet
NBT Session Packet
Flags=0x0
Length=78

SMB PACKET: SMBsesssetupX (REPLY)
SMB Command   =  0x73
Error class   =  0x0
Error code    =  0
Flags1        =  0x80
Flags2        =  0x1
Tree ID       =  121
Proc ID       =  5371
UID           =  1
MID           =  385
Word Count    =  3
Com2=0x75
Off2=68
Action=0x0
[000] Unix Samba 1.9.1
[010] PARKSTR

SMB PACKET: SMBtconX (REPLY) (CHAINED)
smbvwv[]=
Com2=0xFF
Off2=78
smbbuf[]=
ServiceType=A:
```

Po przydzieleniu identyfikatora TID klient może używać wszystkich poleceń, których użyłby w odniesieniu do lokalnego dysku. Może otwierać pliki, czytać je, zapisywać, usuwać i tworzyć, wyszukiwać pliki o określonych nazwach i tak dalej.

4

Udziały dyskowe

W poprzednich trzech rozdziałach pokazaliśmy, jak zainstalować Sambę na uniksowym serwerze i jak skonfigurować klienty Windows do korzystania z prostego udziału dyskowego. W tym rozdziale spróbujemy przydzielić Sambie więcej funkcji do pełnienia w twojej sieci.

Demony Samby, *smbd* i *nmbd*, są kontrolowane przez jeden plik ASCII, *smb.conf*, który może zawierać ponad 200 różnych opcji. Opcje te definiują reakcje Samby na otaczającą ją sieć, zaczynając od prostych praw dostępu, a kończąc na zaszyfrowanych połączeniach i domenach NT. W następnych pięciu rozdziałach zapoznasz się z tym plikiem i używanymi w nim opcjami. Niektóre opcje będziesz wykorzystywał i modyfikował bardzo często, innych zapewne nie użyjesz nigdy – wszystko zależy od tego, jak dalece funkcjonalne mają być klienty Samby.

W tym rozdziale opiszemy strukturę pliku konfiguracyjnego Samby i przedstawimy opcje służące do tworzenia i modyfikowania udziałów dyskowych. W kolejnych rozdziałach omówimy przeglądanie, konfigurowanie użytkowników, bezpieczeństwa, domen i drukarek oraz mnóstwa innych usług, które dzięki Sambie udostępnisz w swojej sieci.

Poznajemy plik konfiguracyjny Samby

Oto przykładowy plik konfiguracyjny Samby. Jeśli pracowałeś z plikami .INI w Windows, jego struktura zapewne wyda ci się znajoma:

```
[global]
    log level = 1
    max log size = 1000
    socket options = TCP_NODELAY IPTOS_LOWDELAY
    guest ok = no
[homes]
    browseable = no
    map archive = yes
[printers]
    path = /usr/tmp
    guest ok = yes
    printable = yes
    min print space = 2000
```

```
[test]
   browseable = yes
   read only = yes
   guest ok = yes
   path = /export/samba/test
```

Na razie zapewne nie rozumiesz wszystkich opcji, ale możesz wykorzystać ten plik, jeśli naprawdę ci się spieszy (jeśli nie, za chwilę będziemy tworzyć nowy plik od podstaw). Mówiąc w skrócie, plik ten ustawia podstawowy poziom komunikatów diagnostycznych, które będą rejestrowane w domyślnym pliku dziennika o rozmiarze nie przekraczającym 1 megabajta, optymalizuje połączenia TCP/IP między serwerem Samby i klientami SMB oraz pozwala Sambie na dynamiczne tworzenie udziałów dyskowych dla wszystkich użytkowników, którzy mają standardowe konta uniksowe na serwerze. Udostępnia też publicznie wszystkie drukarki zarejestrowane w serwerze oraz przeznaczony tylko do odczytu udział plikowy, odpowiadający katalogowi */export/samba/test*. Ostatnia część tego pliku przypomina udział dyskowy, którego użyliśmy do przetestowania Samby w rozdziale 2, *Instalowanie Samby w Uniksie*.

Struktura pliku konfiguracyjnego

Spójrzmy jeszcze raz na ten plik konfiguracyjny, tym razem z wyższego poziomu:

```
[global]
    ...
[homes]
    ...
[printers]
    ...
[test]
    ...
```

Nazwy wewnątrz nawiasów kwadratowych wydzielają sekcje pliku *smb.conf*; każda sekcja przyjmuje nazwę *udziału* (lub usługi), którego parametry są opisane w tej sekcji. Na przykład sekcje `[test]` i `[homes]` to niepowtarzalne udziały dyskowe – zawierają opcje odwołujące się do konkretnych katalogów w serwerze Samby. Udział `[printers]` zawiera opcje odwołujące się do drukarek serwera. Wszystkie sekcje zdefiniowane w pliku *smb.conf*, z wyjątkiem sekcji `[globals]`, będą dostępne jako udział dyskowy lub drukarka dla każdego klienta łączącego się z serwerem Samby.

Pozostałe linie zawierają indywidualne opcje konfiguracyjne odnoszące się do określonego udziału. Opcje te rozciągają się aż do początku następnej sekcji albo do końca pliku. Wszystkie opcje konfiguracyjne mają prosty format:

opcja = warto□□

Opcje w pliku *smb.conf* ustawia się przez przypisanie im wartości. Powinniśmy z góry cię ostrzec, że nazwy niektórych opcji są dobrane nie najlepiej. Na przykład opcja `read only` (tylko do odczytu) jest oczywista, podobnie jak wiele nowszych opcji Samby, ale starsza opcja `public` ma dość niejasne znaczenie, dlatego obecnie można używać jej synonimu `guest ok` (z udziału mogą korzystać goście). Niektóre często używane, choć już przestarzałe nazwy, opiszemy w podrozdziałach poświę-

conych głównym czynnościom konfiguracyjnym. W dodatku C, *Spis opcji konfiguracyjnych Samby*, znajdziesz alfabetyczny indeks wszystkich opcji konfiguracyjnych i ich znaczenia.

Odstępy, cudzysłowy i przecinki

Należy pamiętać, że wszystkie odstępy w *warto□ciach* opcji konfiguracyjnych mają znaczenie. Rozważmy na przykład następującą opcję:

```
volume = Wielki Grozny Dysk Twardy Numer 3543
```

Samba usuwa spacje między końcową literą `e` w słowie `volume` i początkową literą `W` w słowie `Wielki`. Te odstępy nie są znaczące. Inne odstępy mają jednak znaczenie i zostaną rozpoznane oraz zachowane przez Sambę podczas wczytywania pliku. Spacje nie mają znaczenia w nazwach opcji (takich jak `guest ok`), ale zalecamy stosowanie się do konwencji i zachowywanie spacji między poszczególnymi słowami opcji.

Jeśli czujesz się bezpieczniej, wpisując znak cudzysłowu na początku i na końcu wartości opcji, możesz tak robić – Samba zignoruje te znaki. Nie umieszczaj jednak w cudzysłowie nazwy opcji, ponieważ Samba potraktuje to jako błąd składni.

Możesz też używać zamiennie odstępów i przecinków do oddzielania poszczególnych wartości na liście. Te dwie opcje są równoważne:

```
netbios aliases = sprzedaz, ksiegowosc, place
netbios aliases = sprzedaz ksiegowosc place
```

W pewnych wartościach wolno ci jednak używać tylko jednego rodzaju separatora – albo spacji, albo przecinków.

Wielkość liter

Wielkość liter w pliku konfiguracyjnym Samby nie ma znaczenia, oprócz tych miejsc, w których mogłaby kolidować z zasadami obowiązującymi w systemie operacyjnym Samby. Przypuśćmy, że wpisałeś następującą opcję konfiguracyjną w udziale dyskowym wskazującym na katalog */export/samba/test*:

```
PATH = /EXPORT/SAMBA/TEST
```

Samba zaakceptowałaby opcję konfiguracyjną zapisaną dużymi literami. Kiedy jednak spróbowałaby połączyć się ze wskazanym katalogiem, próba byłaby nieudana, ponieważ system plików Uniksa rozróżnia wielkość liter. W rezultacie katalog nie zostałby znaleziony, a klienci nie mogliby łączyć się z tym udziałem.

Kontynuowanie linii

Możesz przenosić tekst do następnej linii pliku konfiguracyjnego, używając odwrotnego ukośnika, jak w przykładzie poniżej:

```
comment = Pierwszy udzia□ przechowuj□cy podstawowe kopie \
          nowego programu firmy Teamworks.
```

Ze względu na użycie odwrotnego ukośnika Samba traktuje te dwie linie jak jedną. Druga linia zaczyna się od pierwszego znaku nie będącego odstępem; w tym przypadku od litery `n` w słowie `nowego`.

Komentarze

Do pliku *smb.conf* możesz wstawiać komentarze, zaczynając linię od znaku hash (#) lub średnika (;). Oba znaki są równoważne. W poniższym przykładzie trzy pierwsze linie zostaną uznane za komentarz:

```
# To jest sekcja drukarek. Przydzielili□my minimaln□ przestrze□
; drukowania rówN□ 2000, aby zapobiec pewnym b□□dom, które
; pojawiaj□ si□, kiedy programowi buforuj□cemu zabraknie pami□ci

[printers]
     public = yes
     min print space = 2000
```

Samba ignoruje wszystkie linie komentarza w pliku konfiguracyjnym. Nie ma żadnych ograniczeń co do znaków, które można wpisać w linii komentarza po początkowym hashu lub średniku. Zauważ, że znak kontynuacji linii (\) w komentarzu nie zostanie rozpoznany. Jest ignorowany, podobnie jak cała reszta linii.

Zmiany w czasie wykonania

Można modyfikować plik konfiguracyjny *smb.conf* i wszystkie zawarte w nim opcje nawet podczas pracy demonów Samby. Domyślnie Samba sprawdza co 60 sekund, czy plik konfiguracyjny nie został zmodyfikowany. Jeśli wykryje jakieś zmiany, natychmiast wprowadza je w życie. Jeśli nie chcesz czekać tak długo, możesz wymusić przeładowanie pliku, wysyłając sygnał SIGHUP do procesów *smnd* i *nmbd* albo po prostu restartując demony.

Jeśli na przykład proces *smbd* ma identyfikator 893, możesz wymusić przeładowanie pliku konfiguracyjnego za pomocą następującego polecenia:

```
# kill -SIGHUP 893
```

Nie wszystkie zmiany zostaną natychmiast zauważone przez klienty. Na przykład zmiany dotyczące używanego udziału nie zostaną zarejestrowane, dopóki klient nie odłączy się od zasobu i nie połączy się z nim ponownie. Również pewne zmiany parametrów serwera, na przykład jego nazwy NetBIOS-owej i nazwy grupy roboczej, nie zostaną od razu zarejestrowane. Dzięki temu w trakcie danej sesji klienci nie zostaną nagle odłączeni i nie spotkają się z nieoczekiwaną odmową dostępu.

Zmienne

Samba przechowuje obszerny zbiór zmiennych określających charakterystykę serwera i połączonych z nim klientów. Każda zmienna zaczyna się od znaku procentu, po którym następuje pojedyncza duża lub mała litera. Zmiennych można używać tylko po prawej stronie opcji konfiguracyjnej (to znaczy po znaku równości).

```
[pub]
    path=/home/ftp/pub/%a
```

Zmienna `%a` oznacza architekturę klienta (na przykład `WinNT` dla Windows NT, `Win95` dla Windows 95 lub 98 lub `WfWg` dla Windows for Workgroups). Z tej przyczyny Samba udostępni inną ścieżkę do udziału `[pub]` klientom pracującym pod kontrolą Windows NT, inną dla klientów Windows 95, a jeszcze inną dla klientów

Windows for Workgroups. Innymi słowy, ścieżki do udziału widziane przez poszczególne klienty będą różnić się w zależności od architektury:

```
/home/ftp/pub/WinNT
/home/ftp/pub/Win95
/home/ftp/pub/WfWg
```

Takie wykorzystanie zmiennych przydaje się wtedy, gdy chcesz, aby różni użytkownicy otrzymywali odmienną konfigurację, uzależnioną od ich cech lub innych okoliczności. Samba dysponuje 19 zmiennymi, wymienionymi w tabeli 4.1.

Tabela 4.1. Zmienne Samby

Zmienna	*Definicja*
Zmienne klienta	
`%a`	Architektura klienta (na przykład Samba, WfWg, WinNT, Win95 lub UNKNOWN)
`%I`	Adres IP klienta (na przykład 192.168.220.100)
`%m`	NetBIOS-owa nazwa klienta
`%M`	Nazwa DNS klienta
Zmienne użytkownika	
`%g`	Podstawowa grupa `%u`
`%G`	Podstawowa grupa `%U`
`%H`	Katalog macierzysty `%u`
`%u`	Bieżąca uniksowa nazwa użytkownika
`%U`	Żądana nazwa użytkownika (nie zawsze używana przez Sambę)
Zmienne udziału	
`%p`	Ścieżka do głównego katalogu udziału używana przez program montujący, jeśli różni się od `%P`
`%P`	Katalog główny bieżącego udziału
`%S`	Nazwa bieżącego udziału
Zmienne serwera	
`%d`	Bieżący identyfikator procesu serwera
`%h`	Nazwa DNS serwera Samby
`%L`	Nazwa NetBIOS-owa serwera Samby
`%N`	Serwer katalogów macierzystych, ustalony na podstawie mapy programu montującego
`%v`	Wersja Samby
Pozostałe zmienne	
`%R`	Wynegocjowany wariant protokołu SMB
`%T`	Bieżąca data i czas

Oto kolejny przykład użycia zmiennych: przypuśćmy, że w twojej sieci jest pięć klientów, ale jeden z nich, `fred`, wymaga załadowania nieco innej konfiguracji udziału `[homes]`, kiedy łączy się z serwerem Samby. Samba pozwala łatwo uporać się z tym problemem:

```
[homes]
    ...
    include = /usr/local/samba/lib/smb.conf.%m
    ...
```

Użyta tu opcja `include` powoduje wczytanie pliku konfiguracyjnego dla komputera o określonej nazwie NetBIOS-owej (`%m`). Jeśli nazwa klienta to `fred`, a w katalogu *`katalog_samby`*`/lib/` (lub tym, którego nazwę podałeś w pliku konfiguracyjnym) znajduje się plik *smb.conf.fred*, Samba dołączy jego zawartość do domyślnego pliku konfiguracyjnego. Jeśli plik *smb.conf.fred* modyfikuje którąś zmienną konfiguracyjną, to zmieniona wartość będzie miała pierwszeństwo przed wartością zdefiniowaną wcześniej. Zwróć uwagę, że piszemy „wcześniej". Jeśli któraś opcja konfiguracyjna zostanie zmodyfikowana w głównym pliku już po opcji `include`, Samba przyjmie jej nową wartość dla udziału, w którym została zdefiniowana.

Co najważniejsze: jeśli nie ma takiego pliku, Samba nie zgłosi żadnego błędu. W rzeczywistości nie podejmie w ogóle żadnego działania. Dzięki temu wystarczy utworzyć tylko jeden plik konfiguracyjny dla komputera `fred`, a nie po jednym dla każdego NetBIOS-owego komputera w sieci.

Pliki konfiguracyjne przeznaczone dla konkretnych komputerów mogą posłużyć nie tylko do dostosowania usług Samby, ale i do diagnozowania jej działania. Jeśli mamy na przykład jednego klienta, który sprawia problemy, możemy utworzyć dla niego oddzielny plik dziennika o bardziej szczegółowym poziomie diagnostycznym. Możemy dzięki temu sprawdzić, co robi Samba, nie spowalniając innych klientów i nie wypełniając dysku bezużytecznymi dziennikami. Pamiętaj, że w dużych sieciach restartowanie Samby w celach diagnostycznych nie zawsze będzie możliwe!

Możesz użyć dowolnej zmiennej z tabeli 4.1 do nadania dostosowanych wartości różnym opcjom Samby. W następnych rozdziałach wspomnimy o kilku takich opcjach.

Sekcje specjalne

Kiedy wiemy już co nieco o zmiennych, powinniśmy zająć się kilkoma specjalnymi sekcjami pliku konfiguracyjnego Samby. Nie przejmuj się, jeśli i tutaj nie wszystkie opcje będą dla ciebie zrozumiałe; omówimy każdą z nich w następnych rozdziałach.

Sekcja [globals]

Sekcja `[globals]`* pojawia się w niemal każdym pliku konfiguracyjnym Samby, mimo że jej definiowanie nie jest obowiązkowe. Każda opcja zdefiniowana w tej sekcji pliku będzie odnosić się do wszystkich pozostałych udziałów – tak, jakby zawartość sekcji została skopiowana do samego udziału. Jeśli jednak w innej sekcji opcja zostanie zdefiniowana ponownie, nowa wartość będzie miała pierwszeństwo przed wartością z sekcji `[globals]`.

Zilustrujmy to przykładem, który pojawił się już na początku rozdziału:

---

* Można zamiennie stosować nazwy [globals] i [global] (–przyp. tłum).

```
[global]
    log level = 1
    max log size = 1000
    socket options = TCP_NODELAY IPTOS_LOWDELAY
    guest ok = no
[homes]
    browseable = no
    map archive = yes
[printers]
    path = /usr/tmp
    guest ok = yes
    printable = yes
    min print space = 2000
    [test]
    browseable = yes
    read only = yes
    guest ok = yes
    path = /export/samba/test
```

Jeśli chcielibyśmy połączyć klienta z udziałem `[test]`, Samba najpierw wczytałaby sekcję `[globals]`. W tym momencie ustawiłaby opcję `guest ok = no` dla wszystkich udziałów zdefiniowanych w pliku konfiguracyjnym. Dotyczy to udziałów `[homes]` i `[printers]`. Kiedy jednak Samba wczyta sekcję `[test]`, znajdzie opcję `guest ok = yes` i zastąpi domyślną wartość `no` z sekcji `[globals]` wartością `yes`, ale tylko w kontekście udziału `[test]`.

Każda opcja, która występuje poza sekcją (przed pierwszą z oznaczonych sekcji), jest również uważana za opcję globalną.

Sekcja [homes]

Jeśli klient stara się połączyć z udziałem, który nie figuruje w pliku *smb.conf*, Samba spróbuje odszukać sekcję `[homes]`. Jeśli taka sekcja istnieje, niezidentyfikowana nazwa udziału zostanie uznana za uniksową nazwę użytkownika, a Samba sprawdzi, czy taka nazwa występuje w bazie haseł serwera. Jeśli tak jest w istocie, Samba założy, że klient jest uniksowym użytkownikiem, próbującym połączyć się ze swoim katalogiem macierzystym w serwerze.

Przypuśćmy, że klient łączy się po raz pierwszy z serwerem Samby o nazwie `hydra` i próbuje uzyskać dostęp do udziału o nazwie `[alicja]`. W pliku *smb.conf* nie ma zdefiniowanego udziału o takiej nazwie, ale jest sekcja `[homes]`, więc Samba przeszukuje bazę haseł i ustala, że w systemie jest konto użytkownika o nazwie `alicja`. Samba sprawdza następnie hasło dostarczone przez klienta i porównuje je z uniksowym hasłem `alicji`, korzystając albo z systemowego pliku haseł (jeśli w użyciu są niezaszyfrowane hasła), albo z pliku *smbpasswd* (jeśli w użyciu są hasła zaszyfrowane). Jeśli hasła są zgodne, Samba wie, że odgadła poprawnie: `alicja` próbuje połączyć się ze swoim katalogiem macierzystym. Samba tworzy więc dla niej specjalny udział o nazwie `[alicja]`.

Proces tworzenia udziałów użytkowników za pomocą sekcji `[homes]` jest omówiony szczegółowo w rozdziale 6, *Użytkownicy, bezpieczeństwo i domeny*.

Sekcja [printers]

Trzecia sekcja specjalna nosi nazwę `[printers]` i przypomina sekcję `[homes]`. Jeśli klient próbuje połączyć się z udziałem, który nie jest zdefiniowany w pliku *smb.conf*, a jego nazwy nie można znaleźć w pliku haseł, Samba sprawdza, czy nie chodzi o udział drukarki. Odczytuje w tym celu plik parametrów drukarek (zwykle */etc/printcap*) i sprawdza, czy występuje w nim nazwa udziału*. Jeśli tak jest, Samba tworzy udział o nazwie drukarki.

Podobnie jak w przypadku sekcji `[homes]`, oznacza to, że nie musisz tworzyć w pliku *smb.conf* udziału dla każdej z systemowych drukarek. Jeśli odpowiednio poinstruujesz Sambę, będzie ona odwoływać się do uniksowego rejestru drukarek i udostępniać je klientom. Istnieje jednak oczywiste ograniczenie: jeśli masz konto użytkownika `fred` i drukarkę o takiej samej nazwie, Samba zawsze znajdzie najpierw konto użytkownika, nawet wtedy, gdy klient próbuje połączyć się z drukarką.

Proces konfigurowania udziału `[printers]` jest szczegółowo opisany w rozdziale 7, *Drukowanie i odwzorowywanie nazw*.

Opcje konfiguracyjne

Opcje w pliku konfiguracyjnym Samby dzielą się na dwie kategorie: *globalne* i *dotyczące udziałów*. Przynależność do kategorii warunkuje miejsca, w których może pojawić się dana opcja.

Opcje globalne
: Opcje globalne mogą występować tylko w sekcji `[global]` i nigdzie indziej. Są to z reguły opcje, które wpływają na zachowanie samego serwera Samby, a nie udostępnianych przez niego udziałów.

Opcje udziałów
: Opcje udziałów mogą występować w poszczególnych udziałach albo w sekcji `[global]`. Jeśli są umieszczone w sekcji `[global]`, wówczas definiują domyślnie zachowanie wszystkich udziałów, chyba że udział przypisze danej opcji nową wartość.

Można też wyróżnić cztery kategorie wartości, które mogą przyjmować opcje konfiguracyjne. Są to:

Wartości logiczne (boole'owskie)
: Są to po prostu wartości typu „tak-nie". Można je reprezentować za pomocą symboli: `yes`, `no`, `true`, `false`, `0`, `1`. Wielkość liter nie ma znaczenia: `YES` i `yes` to ta sama wartość.

Wartości liczbowe
: Liczby dziesiętne, szesnastkowe lub ósemkowe. Standardowa składnia `0x`*nn* oznacza liczbę szesnastkową, a `0`*nnn* – ósemkową.

---

* W twoim systemie plik ten może mieć inne położenie lub nazwę. Możesz użyć dostarczanego wraz z Sambą polecenia *testparm*, aby sprawdzić wartość opcji konfiguracyjnej `printcap name`; jest to domyślna wartość, wybrana podczas kompilacji Samby.

Łańcuchy
Łańcuchy znaków, takie jak nazwy plików lub użytkowników, w których wielkość liter jest rozróżniana.

Listy wyliczane
Skończone listy znanych wartości. Wartość logiczna jest w istocie listą wyliczaną tylko o dwóch wartościach.

Opcje pliku konfiguracyjnego

Samba oddaje do dyspozycji użytkownika ponad 200 opcji. Zacznijmy od tych, które pozwalają na modyfikowanie samego pliku konfiguracyjnego.

Jak już wspomnieliśmy w tym rozdziale, pliki konfiguracyjne wcale nie muszą być statyczne. Możesz nakazać Sambie dołączenie, a nawet zastąpienie opcji konfiguracyjnych w trakcie ich przetwarzania. Służące do tego opcje są wymienione w tabeli 4.2.

Tabela 4.2. Opcje pliku konfiguracyjnego

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`config file`	Łańcuch (pełna nazwa ze ścieżką)	Określa położenie pliku konfiguracyjnego, który należy wczytać zamiast bieżącego	Brak	Globalny
`include`	Łańcuch (pełna nazwa ze ścieżką)	Określa dodatkowy plik z opcjami, który należy dołączyć w tym punkcie pliku konfiguracyjnego	Brak	Globalny
`copy`	Łańcuch (nazwa udziału)	Umożliwia powielenie opcji innego udziału w bieżącym udziale	Brak	Udział

config file

Globalna opcja `config file` określa zastępczy plik konfiguracyjny, który należy załadować w miejsce przetwarzanego obecnie. Jeśli istnieje plik docelowy, pozostała część bieżącego pliku konfiguracyjnego i wszystkie wczytane dotąd opcje zostaną odrzucone; Samba skonfiguruje się wyłącznie na podstawie opcji z nowego pliku. W opcji `config file` można korzystać z opisanych wyżej zmiennych, co bywa przydatne w sytuacji, gdy chcesz załadować specjalny plik konfiguracyjny w oparciu o nazwę komputera lub użytkownika, który łączy się z serwerem.

Na przykład poniższa linia nakazuje Sambie użycie pliku konfiguracyjnego wyznaczonego przez NetBIOS-ową nazwę łączącego się klienta, jeśli taki plik istnieje. W takim przypadku opcje określone w pierwotnym pliku konfiguracyjnym zostaną zi-

gnorowane. Poniższy przykład próbuje wczytać nowy plik konfiguracyjny w oparciu o NetBIOS-ową nazwę klienta:

```
[global]
    config file = /usr/local/samba/lib/smb.conf.%m
```

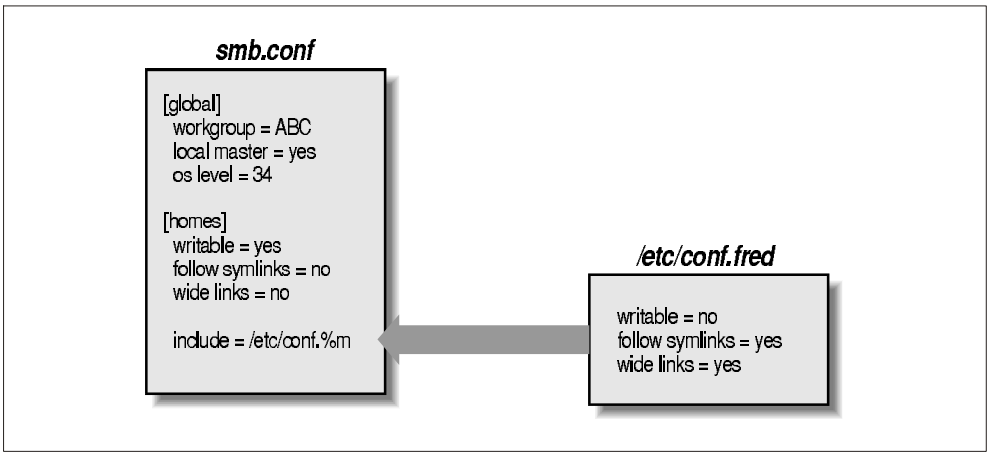
Jeśli określony plik konfiguracyjny nie istnieje, opcja zostanie zignorowana, a Samba skonfiguruje się na podstawie bieżącego pliku.

include

Opcja ta, opisana szczegółowo nieco wcześniej, dołącza docelowy plik do bieżącego pliku konfiguracyjnego w określonym punkcie, co ilustruje rysunek 4.1. W opcji tej również można korzystać z opisanych wcześniej zmiennych, co bywa przydatne w sytuacji, gdy chcesz załadować opcje konfiguracyjne w oparciu o nazwę komputera lub użytkownika, który łączy się z serwerem. Opcję tę możesz wykorzystać następująco:

```
[global]
    include = /usr/local/samba/lib/smb.conf.%m
```

Jeśli wskazany plik konfiguracyjny nie istnieje, opcja zostanie zignorowana. Pamiętaj, że zdefiniowane wcześniej wartości opcji zostaną zastąpione nowymi. W przykładzie z rysunku 4.1 zmienione zostaną wartości wszystkich trzech opcji.



Rysunek 4.1. Opcja include w pliku konfiguracyjnym Samby

Opcja `include` nie rozpoznaje zmiennych `%u` (użytkownik), `%p` (katalog główny bieżącego udziału) ani `%s` (bieżący udział), ponieważ zmienne te nie są jeszcze ustawione w momencie odczytywania pliku.

copy

Opcja konfiguracyjna `copy` umożliwia powielenie opcji innego udziału z bieżącego pliku konfiguracyjnego. Wskazany udział musi występować w pliku konfiguracyjnym wcześniej niż ten, do którego kopiowane są opcje. Na przykład:

```
[szablon]
    writeable = yes
```

```
    browsable = yes
    valid users = andrzej, dawid, piotr

[dane]
    path = /usr/local/samba
    copy = szablon
```

Pamiętaj, że wszystkie opcje w udziale, w którym umieszczono dyrektywę `copy`, będą miały pierwszeństwo przed powielanymi opcjami, niezależnie od tego, czy występują przed, czy za tą dyrektywą.

Konfiguracja serwera

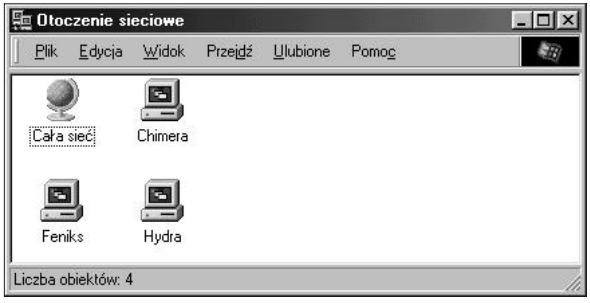
Możemy teraz przystąpić do konfigurowania serwera Samby. Zacznijmy od trzech podstawowych opcji konfiguracyjnych, które mogą pojawić się w sekcji `[global]` pliku *smb.conf*:

```
[global]
    # Parametry konfiguracyjne serwera
    netbios name = HYDRA
    server string = Samba %v w serwerze (%L)
    workgroup = PROSTA_GRUPA
```

Ten plik konfiguracyjny jest bardzo prosty: ogłasza w sieci NBT obecność serwera Samby o NetBIOS-owej nazwie `hydra`. Serwer należy do grupy roboczej PROSTA_GRUPA i zwraca swoim klientom komunikat zawierający numer wersji Samby i NetBIOS-ową nazwę serwera.

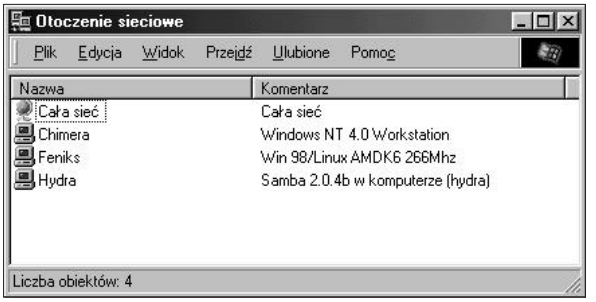
Jeśli poprzednio musiałeś wprowadzić do pliku konfiguracyjnego opcję `encrypt passwords = yes`, musisz wpisać ją także tutaj.

Wypróbuj ten plik konfiguracyjny. Utwórz plik *smb.conf* w katalogu */usr/local/samba/lib* i wpisz do niego powyższy tekst. Następnie zresetuj serwer Samby i zweryfikuj wyniki za pomocą klienta Windows. Upewnij się, że klienty Windows również należą do grupy PROSTA_GRUPA. Po kliknięciu ikony Otoczenia sieciowego w kliencie Windows powinieneś zobaczyć okno podobne do tego z rysunku 4.2 (komputery `feniks` i `chimera` to klienty Windows).



Rysunek 4.2. Otoczenie sieciowe z serwerem Samby

Możesz zweryfikować opcję `server string`, wybierając szczegółowy widok w oknie Otoczenia sieciowego (zaznacz pozycję Szczegóły w menu Widok). Powinieneś teraz zobaczyć okno podobne do tego z rysunku 4.3.



Rysunek 4.3. Szczegółowy widok Otoczenia sieciowego

Jeśli klikniesz ikonę Hydry, powinno ukazać się okno z oferowanymi przez nią usługami. Na razie okno to jest puste, ponieważ w serwerze nie ma jeszcze żadnych udziałów.

Opcje konfiguracyjne serwera

W tabeli 4.3 zebraliśmy opisane dotychczas opcje konfiguracyjne serwera. Zauważ, że wszystkie trzy opcje mają zasięg globalny. Oznacza to, że muszą występować w sekcji `[global]` pliku konfiguracyjnego.

Tabela 4.3. Opcje konfiguracyjne serwera

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`netbios name`	Łańcuch	Ustawia podstawową nazwę NetBIOS-ową serwera Samby	Nazwa hosta DNS	Globalny
`server string`	Łańcuch	Ustawia tekst opisujący serwer Samby	`Samba %v`	Globalny
`workgroup`	Łańcuch	Ustawia NetBIOS-ową grupę komputerów, do której należy serwer	Definiowana podczas kompilacji	Globalny

netbios name

Opcja `netbios name` umożliwia ustawienie NetBIOS-owej nazwy serwera. Na przykład:

```
netbios name = KRAKOWWM1
```

Domyślna wartość tej opcji konfiguracyjnej to nazwa hosta DNS, czyli pierwsza część pełnej nazwy DNS komputera. Na przykład komputer o nazwie DNS `rubin.ora.com` domyślnie otrzymałby NetBIOS-ową nazwę `RUBIN`. Choć można

użyć tej opcji do zmiany NetBIOS-owej nazwy komputera w pliku konfiguracyjnym (jak zrobiliśmy to wcześniej), częściej służy ona do przypisania serwerowi nazwy NetBIOS-owej różnej od bieżącej nazwy DNS. Pamiętaj, że podana tu nazwa musi odpowiadać regułom poprawności określonym w rozdziale 1, *Poznajemy Sambę*.

Nie zaleca się zmieniania NetBIOS-owej nazwy komputera, jeśli nie ma po temu ważnego powodu. Może się zdarzyć, że nazwa hosta nie jest niepowtarzalna, ponieważ sieć lokalna jest podzielona na dwie lub więcej domen DNS. Na przykład nazwa KRAKOWWM1 może odróżniać komputer `wm1.krakow.przyklad.com` od komputera `wm1.katowice.przyklad.com`, który ma tę samą nazwę hosta, ale jest w innej domenie DNS.

Inne zastosowanie tej opcji to przenoszenie usług SMB z uszkodzonego lub wyłączonego komputera. Jeśli na przykład serwer działu sprzedaży ma nazwę SPRZEDAZ i nagle ulegnie awarii, możesz natychmiast ustawić opcję `netbios name = SPRZEDAZ` w zapasowym serwerze Samby, który przejmuje funkcje tamtego. Dzięki temu użytkownicy nie będą musieli zmieniać mapowania dysków – nowe połączenia z serwerem `SPRZEDAZ` po prostu trafią do nowego komputera.

server string

Parametr `server string` określa łańcuch komentarza, który pojawia się obok nazwy serwera w oknie Otoczenia sieciowego (kiedy włączona jest opcja Szczegóły w menu Widok) oraz w menedżerze wydruku Windows. W komentarzu tym możesz użyć standardowych zmiennych Samby. W naszym wcześniejszym przykładzie komentarz miał postać:

```
server string = Samba %v w serwerze (%L)
```

Domyślna wartość tej opcji to po prostu numer wersji Samby; odpowiada to ustawieniu:

```
server string = Samba %v
```

workgroup

Parametr `workgroup` ustawia grupę roboczą, w której serwer będzie ogłaszał swoją obecność. Klienty chcące skorzystać z udziałów serwera powinny być w tej samej grupie roboczej NetBIOS-u. Pamiętaj, że grupy robocze to po prostu NetBIOS-owe nazwy grup, więc muszą stosować się do konwencji nazewniczej opisanej w rozdziale 1. Na przykład:

```
[global]
    workgroup = PROSTA_GRUPA
```

Domyślna wartość tej opcji jest ustawiana w czasie kompilacji. Jeśli nie zmieniono odpowiedniego wpisu w pliku *makefile*, nazwą grupy roboczej będzie `WORKGROUP`. Ponieważ jest to nazwa grupy roboczej w niemal każdej nieskonfigurowanej NetBIOS-owej sieci, zalecamy ustawienie tej opcji w pliku konfiguracyjnym Samby*.

---

* Powinniśmy także nadmienić, że nadawanie grupie roboczej takiej samej nazwy, jaką nosi serwer, nie jest najlepszym pomysłem.

Konfiguracja udziałów dyskowych

W poprzednim podrozdziale wspomnieliśmy, że w serwerze `hydra` nie ma jeszcze żadnych udziałów dyskowych. Utwórzmy zatem pusty udział dyskowy o nazwie `[dane]`. Oto niezbędne uzupełnienia:

```
[global]
    netbios name = HYDRA
    server string = Samba %v w serwerze (%L)
    workgroup = PROSTA_GRUPA

[dane]
    path = /export/samba/dane
    comment = Dysk z danymi
    volume = Stacja-Sieciowa
    writeable = yes
    guest ok = yes
```

Zasób `[dane]` to typowy udział dyskowy Samby. Udział ten odpowiada katalogowi */export/samba/dane* w serwerze Samby. Dołączyliśmy także komentarz, który opisuje udział jako `Dysk z danymi`, a także nazwę wolumenu dla samego udziału.

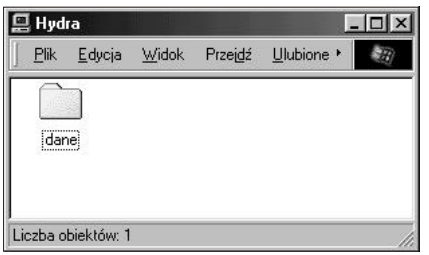
Udział jest zapisywalny, dzięki czemu użytkownicy mogą zapisywać na nim dane; domyślnie Samba tworzy udziały przeznaczone tylko do odczytu. Musisz zatem jawnie ustawić tę opcję dla każdego udziału dyskowego, jeśli chcesz zezwolić na zapis.

Prawdopodobnie zauważyłeś, że ustawiliśmy opcję `guest ok` na `yes`. Nie jest to zbyt bezpieczne, ale musimy najpierw wyjaśnić pewne zagadnienia związane z hasłami, zanim zaczniemy konfigurować konta użytkowników i uwierzytelnianie. Póki co, dzięki tej opcji pominiemy kwestię haseł i pozwolimy na łączenie się z udziałem wszystkim użytkownikom.

Dopisz teraz powyższe opcje do swojego pliku konfiguracyjnego. Oprócz tego jako root utwórz katalog */export/samba/dane* w serwerze Samby za pomocą poleceń:

```
# mkdir /export/samba/dane
# chmod 777 /export/samba/dane
```

Jeśli znów połączysz się z serwerem `hydra` (klikając jego ikonę w oknie Otoczenia sieciowego), powinieneś zobaczyć udział o nazwie `dane`, jak na rysunku 4.4. Udział ten powinien umożliwiać zarówno odczyt, jak i zapis. Spróbuj skopiować do niego plik lub utworzyć nowy. Jeśli nie brak ci odwagi, możesz nawet przypisać mu literę dysku!



Rysunek 4.4. Udział dane w serwerze Samby

Opcje konfiguracyjne udziałów dyskowych

Opisane poprzednio opcje konfiguracyjne udziałów dyskowych są wymienione w tabeli 4.4.

Tabela 4.4. Podstawowe opcje konfiguracyjne udziałów

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`path (directory)`	Łańcuch (pełna nazwa ze ścieżką)	Ustawia nazwę uniksowego katalogu, który będzie udostępniany jako udział dyskowy albo używany jako katalog buforowy drukarki	`/tmp`	Udział
`guest ok (public)`	Wartość logiczna	Jeśli jest ustawiona na `yes`, dostęp do tego udziału nie wymaga uwierzytelnienia	`no`	Udział
`comment`	Łańcuch	Ustawia komentarz pojawiający się przy nazwie udziału	Brak	Udział
`volume`	Łańcuch	Ustawia nazwę wolumenu (DOS-ową nazwę fizycznego dysku)	Nazwa udziału	Udział
`read only`	Wartość logiczna	Jeśli jest ustawiona na `yes`, umożliwia tylko odczyt udziału	`yes`	Udział
`writeable (write ok)`	Wartość logiczna	Jeśli jest ustawiona na `no`, umożliwia tylko odczyt udziału	`no`	Udział

path

Opcja ta (i jej synonim `directory`) określa ścieżkę do głównego katalogu udziału plikowego lub drukarki. Możesz wybrać dowolny katalog w serwerze Samby, pod warunkiem, że użytkownik łączącego się procesu ma prawo do odczytu i zapisu w tym katalogu. Jeśli ścieżka odnosi się do drukarki, powinna wskazywać na katalog tymczasowy, w którym można zapisać pliki na serwerze przed ich wysłaniem do bufora docelowej drukarki (często używa się w tym celu katalogów */tmp* lub */var/spool*). Jeśli ścieżka odnosi się do udziału dyskowego, zawartość folderu reprezentującego udział w kliencie będzie odpowiadać zawartości katalogu w serwerze Samby. Jeśli na przykład zdefiniujemy poniższy udział dyskowy w naszym pliku konfiguracyjnym:

```
[siec]
     path = /export/samba/siec
     writable* = yes
     guest ok = yes
```

* Obie formy zapisu tej opcji: `writeable` i `writable` są równoważne; analogicznie: `browseable` i `browsable` (–przyp. tłum.)

a zawartość uniksowego katalogu */export/samba/siec* wygląda następująco:

```
$ ls -al /export/samba/siec
drwxrwxrwx   9  root   nobody  1024  Feb 16 17:17  .
drwxr-xr-x   9  nobody nobody  1024  Feb 16 17:17  ..
drwxr-xr-x   9  nobody nobody  1024  Feb 16 17:17  quicken
drwxr-xr-x   9  nobody nobody  1024  Feb 16 17:17  podatki98
drwxr-xr-x   9  nobody nobody  1024  Feb 16 17:17  dokumenty
```

to po stronie klienta powinniśmy zobaczyć mniej więcej to, co na rysunku 4.5.

Rysunek 4.5. Widok systemu plików określonego opcją path w kliencie Windows

guest ok

Ta opcja (i jej starszy synonim `public`) umożliwia lub uniemożliwia „gościnny" dostęp do udziału. Jej domyślna wartość to `no`. Jeśli jest ustawiona na `yes`, oznacza to, że w celu uzyskania dostępu do udziału nie trzeba będzie podawać nazwy użytkownika ani hasła. Kiedy użytkownik połączy się z serwerem, jego prawa dostępu będą równoważne prawom wyznaczonego użytkownika-gościa. Domyślne konto gościa to `nobody`, co można zmienić za pomocą opcji konfiguracyjnej `guest account`. Na przykład poniższe linie zezwalają gościom na dostęp do udziału `[ksiegowosc]` z prawami konta `ftp`:

```
[global]
    guest account = ftp
[ksiegowosc]
    path = /usr/local/ksieg
    guest ok = yes
```

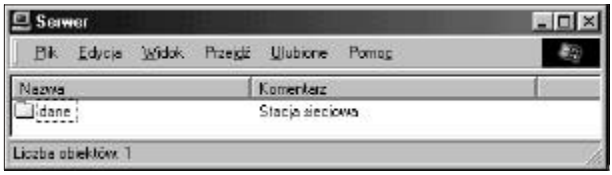
Użytkownicy mogą nadal łączyć się z udziałem za pomocą poprawnej kombinacji nazwa użytkownika-hasło. Jeśli zostaną pomyślnie uwierzytelnieni, uzyskają prawa dostępu wynikające z uprawnień własnego konta, a nie konta gościa. Jeśli jednak próba zalogowania się nie powiedzie, użytkownik otrzyma prawa dostępu wynikające z uprawnień konta gościa. Używając opcji `guest only = yes` możesz sprawić, że każdy użytkownik łączący się z udziałem będzie korzystał z konta gościa (i wobec tego miał prawa gościa).

comment

Opcja `comment` pozwala na wpisanie komentarza, który będzie wysyłany do klientów próbujących przejrzeć zawartość udziału. Użytkownik może obejrzeć ten komentarz wybierając szczegółowy widok folderu odpowiedniego komputera w oknie Otoczenia sieciowego albo wpisując `NET VIEW` w oknie MS-DOS. Oto, jak można dodać komentarz do udziału `[siec]`:

```
[siec]
    comment = Stacja sieciowa
    path = /export/samba/siec
```

Po stronie klienta będzie można wówczas zobaczyć folder podobny do tego z rysunku 4.6. W obecnej konfiguracji Windows komentarz ten nie jest wyświetlany dla udziałów, które są mapowane na dysk sieciowy.



Rysunek 4.6. Komentarz udziału w kliencie Windows

Nie myl opcji `comment`, która dokumentuje udziały serwera Samby, z opcją `server string`, która opisuje sam serwer.

volume

Opcja ta umożliwia określenie nazwy wolumenu udziału zgłaszanej przez SMB. Zwykle jest ona taka sama, jak nazwa udziału zdefiniowana w pliku *smb.conf*. Jeśli jednak chciałbyś nadać jej inną nazwę (z dowolnej przyczyny), możesz to zrobić za pomocą tej opcji.

Na przykład program instalacyjny może sprawdzać nazwę wolumenu CD-ROM-u, aby przed przystąpieniem do instalacji upewnić się, że w stacji jest właściwy CD-ROM. Jeśli chciałbyś skopiować zawartość CD-ROM-u do udziału dyskowego i instalować oprogramowanie z tego udziału, opcja ta pozwoli obejść zabezpieczenie:

```
[siec]
    comment = Stacja sieciowa
    volume = ASVP-102-RTYUIKA
    path = /home/samba/siec
```

read only i writeable

Opcje `read only` i `writeable` (lub `write ok`) to w gruncie rzeczy dwa sposoby powiedzenia tej samej rzeczy, choć z przeciwnych pozycji. Możesz więc ustawić dowolną z poniższych dwóch opcji w sekcji `[global]` albo w którymś z udziałów:

```
read only = yes
writeable = no
```

Jeśli dowolna z tych opcji jest ustawiona w taki sposób, można będzie odczytywać dane z udziału, ale nie zapisywać. Być może myślisz, że opcje te będą potrzebne tylko podczas tworzenia udziału przeznaczonego tylko do odczytu. Zauważ jednak, że ów tryb tylko do odczytu jest *domyślnym* trybem udziałów; jeśli chcesz zapisywać dane na udziale, musisz jawnie podać w pliku konfiguracyjnym jedną z poniższych opcji dla każdego udziału:

```
read only = no
writeable = yes
```

Jeśli użyjesz którejś z tych opcji więcej niż jeden raz, Samba przyjmie ostatnią zdefiniowaną wartość.

Sieciowe opcje Samby

Gdy Samba działa w serwerze z wieloma kartami sieciowymi (połączonym z różnymi podsieciami), a nawet wtedy, gdy chcesz wdrożyć politykę bezpieczeństwa we własnej podsieci, powinieneś bliżej zainteresować się sieciowymi opcjami konfiguracyjnymi.

W tym przykładzie założymy, że serwer Samby jest podłączony do więcej niż jednej podsieci. Mówiąc ściślej, komputer ma dostęp do podsieci 192.168.220.* i 134.213.233.*. Oto, jak można uzupełnić tworzony przez nas plik o opcje konfiguracji sieci:

```
[global]
    netbios name = HYDRA
    server string = Samba %v w serwerze (%L)
    workgroup = PROSTA_GRUPA

    # Opcje konfiguracji sieci
    hosts allow = 192.168.220. 134.213.233. localhost
    hosts deny = 192.168.220.102
    interfaces = 192.168.220.100/255.255.255.0 \
                    134.213.233.110/255.255.255.0
    bind interfaces only = yes

[dane]
    path = /export/samba/dane
    guest ok = yes
    comment = Dysk z danymi
    volume = Stacja-Sieciowa
    writeable = yes
```

Zajmijmy się najpierw opcjami `hosts allow` i `hosts deny`. Jeśli ich nazwy wydają ci się znajome, prawdopodobnie myślisz o plikach *hosts.allow* i *hosts.deny* znajdujących się w katalogu */etc* wielu systemów uniksowych. Opcje te mają takie samo zastosowanie: zapewniają bezpieczeństwo, pozwalając hostom o określonych adresach IP na nawiązywanie połączeń lub odmawiając im dostępu. Czemu nie można po prostu użyć plików *hosts.allow* i *hosts.deny*? Ponieważ w serwerze mogą znajdować się inne usługi, które chciałbyś oddać do dyspozycji innym użytkownikom, nie dając im dostępu do udziałów dyskowych i drukarek Samby.

W opcji `hosts allow` powyżej podaliśmy obcięty adres IP: 192.168.220. (zauważ, że jest w nim trzecia kropka; brakuje tylko czwartej liczby). Jest to równoważne stwierdzeniu: „wszystkie hosty w podsieci 192.168.220". Zarazem jawnie określiliśmy, że host o adresie 192.168.220.102 nie ma zezwolenia na dostęp.

Być może zastanawiasz się, dlaczego host 192.168.220.102 nie uzyska zezwolenia na dostęp, skoro jest w podsieci określonej opcją `hosts allow`? Oto, w jaki sposób Samba przetwarza reguły zawarte w opcjach `hosts allow` i `hosts deny`:

1. Jeśli w pliku *smb.conf* nie ma zdefiniowanych opcji `hosts allow` i `hosts deny`, Samba będzie przyjmować połączenia od dowolnego komputera, któremu pozwala na to sam system.

2. Jeśli opcje `hosts allow` lub `hosts deny` są zdefiniowane w sekcji `[global]` pliku *smb.conf*, będą odnosiły się do wszystkich udziałów, nawet wtedy, gdy w udziale znajdują się przedefiniowujące je opcje.
3. Jeśli w udziale zdefiniowana jest tylko opcja `hosts allow`, tylko wymienione hosty będą miały dostęp do udziału. Wszystkie inne spotkają się z odmową dostępu.
4. Jeśli w udziale zdefiniowana jest tylko opcja `hosts deny`, wszystkie hosty nie wymienione na liście będą mogły korzystać z udziału.
5. Jeśli zdefiniowane są opcje `hosts allow` i `hosts deny`, host musi występować na liście `allow` i nie występować na liście `deny` (w dowolnej formie), aby mógł korzystać z udziału. W przeciwnym wypadku spotka się z odmową dostępu.

Uważaj, aby po jawnym zezwoleniu hostowi na dostęp nie odmówić dostępu całej podsieci, której ten host jej częścią.

Spójrzmy na przykład ilustrujący ostatni punkt. Rozważmy następujące opcje:

```
hosts allow = 111.222.
hosts deny = 111.222.333.
```

W tym przypadku tylko hosty należące do podsieci 111.222.*.* będą miały dostęp do udziałów Samby. Jeśli jednak klient należy do podsieci 111.222.333.*, spotka się z odmową dostępu, choć spełnia wymagania opcji `hosts allow`. Klient musi występować na liście `hosts allow` i *nie może* występować na liście `hosts deny`, aby uzyskać dostęp do udziału Samby. Jeśli komputer spróbuje połączyć się z udziałem, do którego nie ma prawa dostępu, otrzyma komunikat o błędzie.

Pozostałe dwie opcje to `interfaces` i `bind interfaces only`. Przyjrzyjmy się najpierw opcji `interfaces`. Domyślnie Samba wysyła dane tylko przez podstawowy interfejs sieciowy, który w naszym przypadku znajduje się w podsieci 192.168.220.100. Jeśli chcielibyśmy, aby dane były wysyłane przez więcej niż jeden interfejs, musielibyśmy podać ich pełną listę w opcji `interfaces`. W poprzednim przykładzie nakazaliśmy Sambie współpracę z oboma podsieciami (192.168.220 i 134.213.233), w których działa komputer, podając adres drugiego interfejsu sieciowego: 134.213.233.100. Jeśli w twoim komputerze jest więcej niż jeden interfejs, powinieneś zawsze użyć tej opcji, ponieważ nie można zagwarantować, że podstawowy interfejs wybrany przez Sambę będzie tym właściwym.

Wreszcie opcja `bind interfaces only` instruuje proces *nmbd*, aby nie akceptował komunikatów rozgłoszeniowych pochodzących z podsieci innych niż te wymienione w opcji `interfaces`. Zauważ, że różni się to od działania opcji `hosts allow` i `hosts deny`, które uniemożliwiają nawiązanie połączenia z usługami, ale nie zapobiegają przyjmowaniu komunikatów rozgłoszeniowych. Opcja `bind interfaces only` sprawia, że serwer Samby nie będzie odbierał nawet datagramów z obcych podsieci. Oprócz tego instruuje proces *smbd*, aby wiązał się tylko z interfejsami wymienionymi w opcji `interfaces`, co ogranicza liczbę podsieci obsługiwanych przez Sambę.

Opcje sieciowe

Opisane wyżej opcje sieciowe są wymienione w tabeli 4.5.

Tabela 4.5. Opcje konfiguracji sieci

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`hosts allow` (`allow hosts`)	Łańcuch (lista nazw hostów)	Określa komputery, które mogą łączyć się z Sambą	Brak	Udział
`hosts deny` (`deny hosts`)	Łańcuch (lista nazw hostów)	Określa komputery, które nie mogą łączyć się z Sambą	Brak	Udział
`interfaces`	Łańcuch (lista kombinacji adres IP/maska sieciowa)	Określa interfejsy sieciowe obsługiwane przez Sambę. Umożliwia skorygowanie domyślnych ustawień	Zależna od systemu	Globalny
`bind interfaces only`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba wiąże się tylko z tymi interfejsami, które podano w opcji `interfaces`	`no`	Globalny
`socket address`	Łańcuch (adres IP)	Ustawia adres IP, pod którym należy czekać na żądania; używana w serwerach z wieloma wirtualnymi interfejsami	Brak	Globalny

hosts allow

Opcja `hosts allow` (czasem zapisywana także w postaci `allow hosts`) określa komputery, które mogą korzystać z udziałów w serwerze Samby. Jej parametrem jest lista oddzielonych przecinkami nazw komputerów lub adresów IP. Możesz znacznie podnieść poziom bezpieczeństwa, po prostu umieszczając w tej opcji adres swojej lokalnej podsieci. W naszym przykładzie użyliśmy następującej opcji:

```
hosts allow = 192.168.220. localhost
```

Zauważ, że oprócz adresu podsieci uwzględniliśmy także lokalnego hosta. Jedna z najczęstszych pomyłek w użyciu opcji `hosts allow` polega na tym, że serwerowi Samby można przypadkowo uniemożliwić komunikację z samym sobą. Program *smbpasswd* od czasu do czasu musi połączyć się z serwerem Samby jako klient, aby zmienić zaszyfrowane hasło użytkownika. Ponadto dostęp z lokalnego hosta jest

niezbędny, aby rozpowszechniać informacje o przeglądaniu. Jeśli opcja ta jest włączona, a nie zawarto w niej adresu lokalnego hosta, wówczas wygenerowane lokalnie pakiety z żądaniem zmiany hasła zostaną odrzucone przez Sambę, a rozpowszechnianie informacji o przeglądaniu będzie działać błędnie. Aby tego uniknąć, należy jawnie zezwolić na korzystanie z adresu pętli zwrotnej (pisząc albo `localhost`, albo `127.0.0.1`)*.

W opcji tej można używać dowolnego z poniższych formatów:

- Nazwy hostów, takie jak `ftp.przyklad.com`.
- Adresy IP, takie jak `130.63.9.252`.
- Nazwy domen, które można odróżnić od nazw hostów, ponieważ zaczynają się od kropki. Zapis `.ora.com` oznacza wszystkie komputery w domenie *ora.com*.
- Grupy sieciowe, których nazwy zaczynają się od znaku at, na przykład `@hosty-wydruku`. Z grup sieciowych można korzystać praktycznie tylko w systemach używających NIS lub NIS+. Jeśli twój system obsługuje grupy sieciowe, powinna być w nim dostępna strona podręcznika man `netgroups`, opisująca je bardziej szczegółowo.
- Podsieci, które kończą się kropką. Na przykład `136.63.9.` oznacza wszystkie komputery, których adres IP zaczyna się od 130.63.9.
- Słowo kluczowe `ALL`, które zezwala na dostęp wszystkim klientom.
- Słowo kluczowe `EXCEPT`, po którym następuje jedna lub więcej nazw komputerów, adresów IP, nazw domen, grup sieciowych albo podsieci. Mógłbyś na przykład nakazać, aby Samba obsługiwała wszystkie hosty oprócz tych w podsieci 192.168.110, używając opcji `hosts allow = ALL EXCEPT 192.168.110.` (pamiętaj o ostatniej kropce).

Używanie słowa kluczowego `ALL` jest niemal zawsze niewskazane, ponieważ umożliwia dowolnemu użytkownikowi dowolnej sieci przeglądanie twoich plików, jeśli tylko zdoła on odgadnąć nazwę serwera.

Zauważ, że opcja konfiguracyjna `hosts allow` nie ma wartości domyślnej, choć domyślne działanie w razie nieobecności obu opcji `hosts` polega na zezwoleniu na dostęp ze wszystkich źródeł. Oprócz tego, jeśli umieścisz opcję `hosts allow` w sekcji `[global]` pliku konfiguracyjnego, będzie ona miała pierwszeństwo przed opcjami `hosts allow` zdefiniowanymi w udziałach.

hosts deny

Opcja `hosts deny` (także `deny hosts`) określa komputery, które nie mają zezwolenia na dostęp do udziałów. Jej parametrem jest lista oddzielonych przecinkami nazw komputerów lub adresów IP. Do określania klientów można użyć tego samego formatu co w opisanej wyżej opcji `hosts allow`. Aby na przykład zezwolić na dostęp do serwera tylko z domeny *przyklad.com*, mógłbyś użyć opcji:

```
hosts deny = ALL EXCEPT .przyklad.com
```

* Od wersji 2.0.5 Samby lokalny host automatycznie uzyskuje prawo dostępu, o ile jawnie mu się go nie odmówi.

Podobnie jak `hosts allow`, opcja ta nie ma wartości domyślnej, choć domyślne działanie w razie nieobecności obu opcji `hosts` polega na zezwoleniu na dostęp ze wszystkich źródeł. Oprócz tego, jeśli umieścisz opcję `hosts deny` w sekcji `[global]` pliku konfiguracyjnego, będzie ona miała pierwszeństwo przed opcjami `hosts deny` zdefiniowanymi w udziałach. Jeśli chcesz ograniczyć hostom dostęp do konkretnych udziałów, nie umieszczaj opcji `hosts allow` i `hosts deny` w sekcji `[global]`.

interfaces

Opcja `interfaces` wymienia interfejsy sieciowe, które Samba będzie rozpoznawać i przez które będzie odpowiadać. Opcja ta jest przydatna, jeśli twój komputer należy do więcej niż jednej podsieci. Jeśli opcja ta nie jest ustawiona, Samba w trakcie uruchamiania wyszukuje podstawowy interfejs serwera (zwykle pierwszą kartę Ethernetu) i konfiguruje się do obsługi tej jednej podsieci. Jeśli serwer działa nie tylko w jednej podsieci, a ty nie podasz tej opcji, Samba będzie pracować wyłącznie w pierwszej wykrytej podsieci. Musisz posłużyć się tą opcją, aby Samba obsługiwała pozostałe podsieci.

Wartością tej opcji jest jedna lub więcej par adres IP-maska sieciowa, jak w poniższym przykładzie:

```
interfaces = 192.168.220.100/255.255.255.0 192.168.210.30/255.255.255.0
```

Można także podać maskę bitową w formacie CIDR, jak niżej:

```
interfaces = 192.168.220.100/24 192.168.210.30/24
```

Maska bitowa to liczba początkowych bitów, które zostaną włączone w masce sieciowej. Na przykład liczba 24 oznacza, że aktywne będą pierwsze 24 bity (z 32), co jest równoważne notacji 255.255.255.0. Podobnie, liczba 16 odpowiada masce 255.255.0.0, a 8 – 255.0.0.0.

Jeśli używasz DHCP, opcja ta może działać nieprawidłowo.

bind interfaces only

Opcja `bind interfaces only` sprawia, że procesy *smbd* i *nmbd* obsługują żądania pochodzące tylko z tych podsieci, które są wymienione w opcji `interfaces`. Zwykle proces *nmbd* jest powiązany z interfejsem wszystkich adresów (0.0.0.0) na portach 137 i 138, co pozwala mu na odbieranie rozgłoszeń z dowolnego źródła. Możesz jednak to zmienić za pomocą następującej opcji:

```
bind interfaces only = yes
```

Spowoduje to, że oba procesy Samby będą ignorować wszystkie pakiety, których adres źródłowy nie odpowiada adresom rozgłoszeniowym określonym w opcji `interfaces`, łącznie z pakietami rozgłoszeniowymi. Jeśli chodzi o proces *smbd*, opcja ta sprawi, że Samba nie będzie honorować żądań dostępu do plików pochodzących z podsieci innych niż wymienione w opcji `interfaces`. Nie powinieneś używać tej opcji, jeśli chcesz zezwolić na tymczasowe połączenia sieciowe, na przykład tworzo-

ne za pośrednictwem SLIP lub PPP. Opcja ta jest potrzebna bardzo rzadko i powinni jej używać tylko eksperci.

Jeśli ustawisz opcję `bind interfaces only` na `yes`, powinieneś dodać adres lokalnego hosta (127.0.0.1) do listy `interfaces`. W przeciwnym wypadku program *smbpasswd* nie będzie mógł połączyć się z serwerem w celu zmiany hasła.

socket address

Opcja `socket address` określa, pod którymi adresami podanymi w opcji `interfaces` Samba będzie czekać na połączenia. Samba domyślnie akceptuje połączenia kierowane pod wszystkie podane adresy. Jeśli użyjesz tej opcji w pliku *smb.conf*, wymusi ona czekanie pod tylko jednym adresem IP. Na przykład:

```
interfaces = 192.168.220.100/24 192.168.210.30/24
socket address = 192.168.210.30
```

Opcja ta jest narzędziem programistycznym, więc radzimy jej nie używać.

Serwery wirtualne

Serwery wirtualne tworzą iluzję obecności wielu serwerów NetBIOS-u w sieci, choć w rzeczywistości jest tylko jeden taki serwer. Uzyskanie takiego efektu nie jest trudne: komputer po prostu rejestruje więcej niż jedną NetBIOS-ową nazwę w połączeniu ze swoim adresem IP. Metoda ta przynosi wymierne korzyści.

Dział księgowości mógłby na przykład dysponować serwerem `ksiegowosc`, którego klienci widzieliby tylko dyski i drukarki należące do działu księgowości. Dział marketingu mógłby mieć własny serwer `marketing`, przechowujący sprawozdania tworzone w tym dziale – i tak dalej. Jednakże wszystkie usługi byłyby świadczone przez uniksową stację roboczą (z pomocą jednego zrelaksowanego administratora), zamiast przez kilka niewielkich serwerów obsługiwanych przez niezależnych administratorów.

Samba pozwala przypisać kilka nazw NetBIOS-owych uniksowemu serwerowi za pomocą opcji `netbios aliases` (patrz tabela 4.6).

Tabela 4.6. Opcje konfiguracji wirtualnych serwerów

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`netbios aliases`	Lista nazw NetBIOS-owych	Dodatkowe nazwy NetBIOS-owe, na które powinien reagować serwer, używane do utworzenia „wirtualnych" serwerów Samby	Brak	Globalny

netbios aliases

Dzięki opcji `netbios aliases` można nadać serwerowi Samby więcej niż jedną nazwę NetBIOS-ową. Każda podana nazwa NetBIOS-owa zostanie wyświetlona

w oknie Otoczenia sieciowego komputera przeglądającego zasoby sieci. Kiedy jednak klient spróbuje nawiązać połączenie, połączy się z tym samym serwerem Samby.

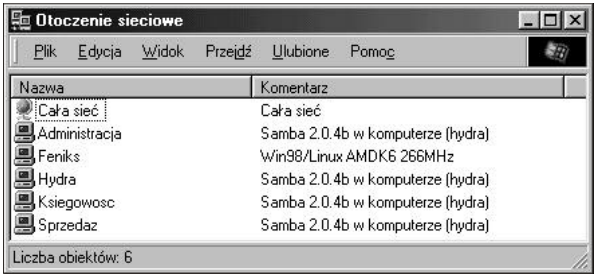
Opcja ta może się przydać na przykład wtedy, gdy przenosisz dane trzech oddziałów firmy na jeden serwer uniksowy z nowoczesnym dużym dyskiem, rezygnując ze starych serwerów NT lub przeznaczając je do innych zadań. Jeśli serwery te nosiły nazwy `sprzedaz`, `ksiegowosc` i `administracja`, Samba może reprezentować wszystkie trzy dzięki opcjom:

```
[global]
    netbios aliases = sprzedaz ksiegowosc administracja
    include = /usr/local/samba/lib/smb.conf.%L
```

To, co pojawiłoby się w Otoczeniu sieciowym klientów, pokazuje rysunek 4.7. Kiedy klient spróbuje połączyć się z Sambą, poda nazwę serwera, z którym chce nawiązać połączenie – nazwa ta będzie dostępna w zmiennej `%L`. Jeśli żądany serwer to `sprzedaz`, Samba dołączy plik */usr/local/samba/lib/smb.conf.sprzedaz*. Plik ten może zawierać deklaracje globalne i definicje udziałów przeznaczone wyłącznie dla działu sprzedaży, na przykład:

```
[global]
    workgroup = SPRZEDAZ
    hosts allow = 192.168.10.255

[sprzedaz1998]
    path = /usr/local/samba/sprzedaz/sprzedaz1998/
...
```

W tym przykładzie ustawiamy także nazwę grupy roboczej na SPRZEDAZ i określamy adres IP, który zezwala na przyjmowanie połączeń tylko z podsieci SPRZEDAZ (192.168.10). Definiujemy także udziały specyficzne dla działu sprzedaży.



Rysunek 4.7. Stosowanie aliasów NetBIOS-owych w serwerze Samby

Opcje konfiguracji rejestrowania

Od czasu do czasu musimy sprawdzić, co właściwie robi Samba, zwłaszcza wtedy, gdy przeprowadzi ona nieoczekiwaną operację lub w ogóle przestanie działać. Aby uzyskać niezbędne informacje, musimy przejrzeć pliki dziennika Samby i sprawdzić, czemu zrobiła to, co zrobiła.

Pliki dziennika Samby mogą być tak szczegółowe albo tak zdawkowe, jak sobie tylko zażyczysz. Oto przykład typowego pliku dziennika Samby:

```
[1999/07/21 13:23:25, 3] smbd/service.c:close_cnum(514)
  feniks (192.168.220.101) closed connection to service IPC$
[1999/07/21 13:23:25, 3] smbd/connection.c:yield_connection(40)
  Yielding connection to IPC$
[1999/07/21 13:23:25, 3] smbd/process.c:process_smb(615)
  Transaction 923 of length 49
[1999/07/21 13:23:25, 3] smbd/process.c:switch_message(448)
  switch message SMBread (pid 467)
[1999/07/21 13:23:25, 3] lib/doscalls.c:dos_ChDir(336)
  dos_ChDir to /home/samba
[1999/07/21 13:23:25, 3] smbd/reply.c:reply_read(2199)
  read fnum=4207 num=2820 nread=2820
[1999/07/21 13:23:25, 3] smbd/process.c:process_smb(615)
  Transaction 924 of length 55
[1999/07/21 13:23:25, 3] smbd/process.c:switch_message(448)
  switch message SMBreadbraw (pid 467)
[1999/07/21 13:23:25, 3] smbd/reply.c:reply_readbraw(2053)
  readbraw fnum=4207 start=130820 max=1276 min=0 nread=1276
[1999/07/21 13:23:25, 3] smbd/process.c:process_smb(615)
  Transaction 925 of length 55
[1999/07/21 13:23:25, 3] smbd/process.c:switch_message(448)
  switch message SMBreadbraw (pid 467)
```

Wiele z tych zapisów przydaje się tylko programistom, ale znaczenie niektórych omówimy nieco dokładniej w rozdziale 9, *Rozwiązywanie problemów*.

Samba zawiera sześć opcji umożliwiających określenie sposobu i miejsca rejestrowania informacji. Każda z nich ma zasięg globalny i nie może pojawić się w definicji udziału. Oto uaktualniony plik konfiguracyjny z omówionymi do tej pory opcjami udziałów i dodanymi opcjami rejestrowania:

```
[global]
    netbios name = HYDRA
    server string = Samba %v w serwerze (%L)
    workgroup = PROSTA_GRUPA

    # Opcje konfiguracji sieci
    hosts allow = 192.168.220. 134.213.233. localhost
    hosts deny  = 192.168.220.102
    interfaces  = 192.168.220.100/255.255.255.0 \
                  134.213.233.110/255.255.255.0
    bind interfaces only = yes

    # Rejestrowanie informacji diagnostycznych
    log level = 2
    log file = /var/log/samba.log.%m
    max log size = 15
    debug timestamp = yes

[dane]
    path = /export/samba/dane
    browseable = yes
    guest ok = yes
    comment = Dysk z danymi
    volume = Stacja-Sieciowa
    writeable = yes
```

Dodaliśmy tutaj własny plik dziennika, który uwzględnia informacje diagnostyczne aż do poziomu 2. Jest to raczej mało szczegółowy poziom diagnostyczny – poziom

rejestrowania można zmieniać od 1 do 10, przy czym poziom 1 zapewnia niewiele informacji, a poziom 10 – bardzo dużo i bardzo szczegółowych. Poziom 2 dostarczy nam pożytecznych informacji diagnostycznych bez marnotrawienia przestrzeni dyskowej serwera. W praktyce powinieneś unikać poziomów rejestrowania wyższych niż 3, jeśli nie programujesz Samby.

Plik zostanie umieszczony w katalogu */var/log* dzięki opcji konfiguracyjnej `log file`. Możemy jednak skorzystać z mechanizmu podstawiania zmiennych, aby utworzyć pliki konfiguracyjne dla poszczególnych klientów lub użytkowników, jak w poniższej linii ze zmienną `%m`:

```
log file = /usr/local/logs/samba.log.%m
```

Wyizolowanie komunikatów może być niezwykle pomocne podczas tropienia błędów w sieci, jeśli wiesz, że problem powoduje konkretny komputer lub użytkownik.

Narzuciliśmy też pewne ograniczenie na pliki dziennika: żaden z nich nie może mieć więcej niż 50 kilobajtów, co określa opcja `max log size`. Jeśli plik dziennika przekroczy ten rozmiar, jego zawartość zostanie przeniesiona do pliku o takiej samej nazwie, ale z przyrostkiem *.old*. Jeśli plik *.old* już istnieje, zostanie nadpisany, a jego zawartość utracona. Pierwotny plik zostanie opróżniony i będzie czekał na nowe komunikaty. Dzięki temu demony Samby nie przepełnią dysku twardego.

Dla wygody zdecydowaliśmy się na umieszczanie znaczników czasowych w dziennikach, używając opcji `debug timestamp`. Spowoduje to zapisywanie daty i czasu obok każdego komunikatu w pliku dziennika. Jeśli informacje te nie byłyby nam potrzebne, moglibyśmy nadać tej opcji wartość `no`.

Korzystanie z rejestratora systemowego

Jeśli oprócz lub zamiast standardowego dziennika Samby chcesz używać systemowego programu rejestrującego (*syslog*), możesz skorzystać z przeznaczonych do tego opcji. Zanim jednak zdecydujesz się na użycie programu *syslog*, musisz upewnić się, że Samba została skompilowana z opcją `configure --with-syslog`. Więcej informacji o konfigurowaniu i kompilowaniu Samby znajdziesz w rozdziale 2.

Następnie musisz zmodyfikować plik */etc/syslog.conf* tak, aby komunikaty Samby były akceptowane przez *syslog*. Jeśli w pliku */etc/syslog.conf* nie ma jeszcze wpisu `daemon.*`, dopisz poniższą linię:

```
daemon.*      /var/log/daemon.log
```

Dzięki temu wszystkie komunikaty od demonów systemowych będą zapisywane w pliku */var/log/daemon.log*. Tam również trafią komunikaty Samby. Teraz możesz umieścić następującą opcję globalną w pliku konfiguracyjnym:

```
syslog = 2
```

Określa ona, że komunikaty o poziomie 1 będą wysyłane zarazem do programu *syslog*, jak i do plików dziennika Samby (odwzorowania na priorytety programu *syslog* są opisane w podrozdziale „syslog"). Załóżmy, że ustawiliśmy opisaną wcześniej opcję `log level` na 4. Wszystkie komunikaty o poziomie 2, 3 i 4 będą wysyłane do pliku dziennika Samby, ale nie do dziennika systemowego. Do obu trafią tylko ko-

munikaty o poziomie 1. Jeśli wartość opcji `syslog` jest większa niż wartość `log level`, program *syslog* niczego nie zarejestruje.

Jeśli chcesz, aby komunikaty były wysyłane tylko do programu *syslog* – a nie do standardowych plików dziennika Samby – możesz umieścić w pliku konfiguracyjnym następującą opcję:

```
syslog only = yes
```

Opcje konfiguracji rejestrowania

W tabeli 4.7 wymienione są opcje konfiguracji rejestrowania używane przez Sambę.

Tabela 4.7. Opcje konfiguracji rejestrowania

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`log file`	Łańcuch (nazwa pliku wraz z pełną ścieżką)	Ustawia nazwę i położenie pliku dziennika używanego przez Sambę. Rozpoznaje standardowe zmienne	Określona w pliku *makefile* Samby	Globalny
`log level` (`debug level`)	Wartość liczbowa (od 0 do 10)	Określa ilość komunikatów informacyjnych i diagnostycznych wysyłanych do pliku dziennika. Wartość 0 oznacza brak komunikatów, 3 – znaczną ilość	`1`	Globalny
`max log size`	Wartość liczbowa (rozmiar w KB)	Ustawia maksymalny rozmiar pliku dziennika. Kiedy dziennik przekroczy ten rozmiar, otrzyma rozszerzenie *.old* i utworzony zostanie nowy plik dziennika	`5000`	Globalny
`debug timestamp` (`timestamp logs`)	Wartość logiczna	Jeśli jest ustawiona na `no`, nie umieszcza znaczników czasowych w plikach dziennika, co ułatwia ich czytanie podczas diagnozowania problemów	`yes`	Globalny
	Wartość liczbowa (0-10)	Ustawia poziom komunikatów wysyłanych do programu *syslog*. Komunikaty o poziomie nie przekraczającym wartości tej opcji zostaną zapisane w dzienniku systemowym	1	Globalny
`syslog only`	Wartość logiczna	Jeśli jest ustawiona na `yes`, używany jest tylko program *syslog*, a w standardowych plikach dziennika Samby nie są zapisywane żadne dane	`no`	Globalny

Wówczas komunikaty o poziomie przekraczającym wartość opcji `syslog` zostaną odrzucone, podobnie jak w przypadku opcji `log level`.

log file

W naszym serwerze Samba zapisuje komunikaty informacyjne w plikach tekstowych umieszczonych w podkatalogu *var* głównego katalogu Samby, zgodnie z ustawieniami w pliku *makefile* dokonanymi podczas kompilacji. Dzięki opcji `log file` można określić inną nazwę i położenie pliku dziennika. Aby na przykład zmienić jego ścieżkę i nazwę na */usr/local/logs/samba.log*, można użyć następującej opcji:

```
[global]
    log file = /usr/local/logs/samba.log
```

Aby utworzyć indywidualne pliki dziennika dla poszczególnych użytkowników lub klientów, można skorzystać z podstawiania zmiennych.

Domyślne położenie pliku dziennika można zmienić za pomocą opcji linii polecenia `-l` podczas uruchamiania obu demonów, która jednak nie ma pierwszeństwa przed opcją `log file`. Jeśli podasz ten parametr, wówczas początkowe komunikaty zostaną zapisane w pliku określonym opcją `-l` (albo w pliku domyślnym, określonym w pliku *makefile* Samby), dopóki demony nie przetworzą pliku *smb.conf* i nie zaczną zapisywać swoich komunikatów w nowym pliku dziennika.

log level

Opcja `log level` określa ilość rejestrowanych danych. Zwykle nadaje się jej wartość 0 lub 1. Jeśli jednak masz konkretny problem, możesz ustawić ją na 3, co spowoduje zapisanie informacji diagnostycznych przydatnych podczas jego rozwiązywania. Poziomy powyżej 3 zapisują informacje przydatne głównie dla programistów szukających wewnętrznych usterek w programie i znacznie spowalniają serwer. Dlatego nie zalecamy ustawiania tej opcji na wartość większą niż 3.

```
[global]
    log file = /usr/local/logs/samba.log.%m
    log level = 3
```

max log size

Opcja `max log size` określa maksymalny rozmiar (w kilobajtach) diagnostycznego pliku dziennika Samby. Kiedy plik dziennika przekroczy ten rozmiar, zostanie do niego dołączone rozszerzenie *.old* (jeśli istnieje już starszy plik o tej nazwie, zostanie usunięty). Następnie zostanie utworzony nowy plik dziennika o pierwotnej nazwie. Na przykład:

```
[global]
    log file = /usr/local/logs/samba.log.%m
    max log size = 1000
```

W tym przykładzie, jeśli plik dziennika przekroczy rozmiar jednego megabajta, Samba zmieni jego nazwę na *samba.log.nazwa-komputera.old* i utworzy nowy plik dziennika. Jeśli istnieje już plik o rozszerzeniu *.old*, Samba go usunie. Zalecamy ustawienie tej opcji w pliku konfiguracyjnym, ponieważ rejestrowanie informacji dia-

gnostycznych (nawet o niższych poziomach) może niezauważalnie pochłonąć znaczną część przestrzeni dyskowej. Dzięki tej opcji nieuważny administrator nie odkryje nagle, że jego dysk jest niemal całkiem zajęty przez wielki plik dziennika Samby.

debug timestamp lub timestamp logs

Jeśli akurat diagnozujesz jakiś problem sieciowy i przeszkadzają ci w tym daty i czasy zapisywane w plikach dziennika Samby, możesz wyłączyć ich rejestrowanie, nadając opcjom `debug timestamp` lub `timestamp logs` (synonimy) wartość `no`. W zwykłym dzienniku Samby informacje są rejestrowane w następujący sposób:

```
12/31/98 12:03:34 hydra (192.168.220.101) connect to server siec as daveb
```

Jeśli nadasz tej opcji wartość `no`, komunikat zostanie zarejestrowany bez daty i czasu:

```
hydra (192.168.220.101) connect to server siec as daveb
```

syslog

Opcja `syslog` sprawia, że komunikaty Samby są wysyłane do systemowego programu rejestrującego. Typ wysyłanych informacji jest zdefiniowany przez parametr tej opcji. Podobnie jak w przypadku opcji `log level`, może to być liczba od 0 do 10. Informacje o poziomie niższym od podanej liczby będą wysyłane do rejestratora systemowego, natomiast informacje o poziomie równym lub większym od wartości opcji `syslog` będą nadal zapisywane w standardowych dziennikach Samby. Aby temu zapobiec, możesz użyć opcji `syslog only`. Na przykład:

```
[global]
    log level = 3
    syslog = 1
```

Przy takich opcjach komunikaty o poziomie 0 będą wysyłane do standardowych dzienników Samby i rejestratora systemowego, natomiast komunikaty o poziomach 1, 2 i 3 będą wysyłane tylko do standardowych dzienników Samby. Komunikaty o poziomie większym niż 3 w ogóle nie będą rejestrowane. Zauważ, że wszystkie komunikaty wysyłane do rejestratora są odwzorowywane na priorytet rozpoznawany przez proces *syslog*, jak w tabeli 4.8. Domyślny poziom to 1.

Tabela 4.8. Konwersja na priorytety programu syslog

Poziom komunikatu	*Priorytet syslog*
0	`LOG_ERR`
1	`LOG_WARNING`
2	`LOG_NOTICE`
3	`LOG_INFO`
4 i wyższe	`LOG_DEBUG`

Jeśli chcesz używać programu *syslog*, będziesz musiał wpisać `configure --with-syslog` podczas kompilowania Samby i odpowiednio skonfigurować plik */etc/syslog.conf* (patrz podrozdział „Korzystanie z rejestratora systemowego").

syslog only

Opcja `syslog only` informuje Sambę, że nie chcesz korzystać ze zwykłych plików dziennika, a tylko z rejestratora systemowego. W tym celu wpisz poniższą opcję w sekcji `[global]` pliku konfiguracyjnego Samby:

```
[global]
    syslog only = yes
```

5

Przeglądanie i zaawansowane udziały dyskowe

W tym rozdziale będziemy nadal zajmować się udziałami dyskowymi. Omówimy różnice między systemami plików Windows i Uniksa oraz sposób ich niwelowania przez Sambę. Między systemami plików DOS-a i Uniksa istnieje zadziwiająco wiele rozbieżności. Oprócz tego krótko opiszemy przekształcanie nazw, blokowanie plików i względnie nową funkcję Samby: blokowanie oportunistyczne. Zanim jednak zaczniemy zgłębiać te zagadnienia, przyjrzymy się dość skomplikowanemu tematowi: przeglądaniu zasobów sieci.

Przeglądanie

Dzięki przeglądaniu można stwierdzić, jakie serwery i udziały są obecnie dostępne w sieci. Użytkownik klienta Windows NT 4.0 lub 95/98 może przeglądać serwery sieciowe w folderze Otoczenia sieciowego. Po dwukrotnym kliknięciu ikony reprezentującej serwer, użytkownik powinien zobaczyć udziały dyskowe i drukarki udostępniane przez serwer (jeśli masz Windows NT 3.*x*, możesz użyć opcji Disk⇨Connect Network Drive w menedżerze plików w celu wyświetlenia udziałów udostępnianych przez serwer).

W linii poleceń Windows możesz także wpisać polecenie `net view`, które wyświetla serwery obecnie dostępne w sieci. Oto przykład użycia polecenia `net view`:

```
C:\>net view
Serwery dost□pne dla grupy PROSTA_GRUPA.
Nazwa serwera          Opis
-------------------------------------------------------------
\\CHIMERA              Windows NT 4.0
\\HYDRA                Samba 2.0.4 w serwerze (hydra)
\\FENIKS               Windows 98
```

Zapobieganie przeglądaniu

Możesz zapobiec umieszczaniu udziału na liście przeglądania za pomocą opcji `browseable`. Opcja ta sprawia, że udział w ogóle nie pojawia się w oknie Otoczenia sieciowego. Aby na przykład zapobiec wyświetlaniu udziału `[dane]` z poprzedniego rozdziału, moglibyśmy napisać:

```
[dane]
    path = /home/samba/dane
    browseable = no
    guest ok = yes
    comment = Dysk z danymi
    volume = Stacja-Sieciowa
    writeable = yes
```

Choć najczęściej takie ustawienie opcji `browseable` nie jest wskazane w zwykłych udziałach dyskowych, to bywa pożyteczne, kiedy zawartość udziału nie powinna być widoczna dla użytkowników, jak w przypadku udziału `[netlogon]` ze skryptami logowania do domeny Windows (więcej informacji o skryptach logowania znajdziesz w rozdziale 6, *Użytkownicy, bezpieczeństwo i domeny*).

Innym przykładem jest udział `[homes]`. Jest on często oznaczany jako „nieprzeglądalny", dzięki czemu podczas przeglądania zasobów komputera udział `[homes]` nie pojawia się na liście. Jeśli jednak użytkownik `alicja` zaloguje się i spojrzy na zasoby komputera, zobaczy udział o nazwie `[alicja]`. A jeśli chcielibyśmy, aby udział `alicji` był widoczny dla innych użytkowników jeszcze przed jej zalogowaniem? Umożliwia to globalna opcja `auto services`. Opcja ta wstępnie umieszcza udziały na liście przeglądania, dzięki czemu są one zawsze widoczne:

```
[global]
    ...
    auto services = alicja
    ...
```

Usługi domyślne

Możesz zdefiniować udział, z którym połączy się użytkownik, jeśli nie zdoła się połączyć z żądanym udziałem. Ponieważ nie da się przewidzieć, kim będzie pechowy użytkownik, prawdopodobnie powinieneś ustawić opcję `guest ok` dla tego awaryjnego udziału. Opcja `default service` może przydać się do odsyłania skonfundowanych użytkowników do katalogu z plikami pomocy. Na przykład:

```
[global]
    ...
    default service = pomoc
    ...

[pomoc]
    path = /home/samba/pomoc/%S
    browseable = yes
    guest ok = yes
    comment = Domy□lny udzia□ dla nieudanych po□□cze□
    volume = Stacja-Sieciowa
    writeable = no
```

Zauważ, że użyliśmy zmiennej `%S` w opcji `path`. Jeśli użyjesz zmiennej `%S`, będzie się ona odnosiła do żądanego, nieistniejącego udziału (tego, z którym użytkownik próbował połączyć się pierwotnie), a nie do nazwy udziału domyślnego. Możemy zatem utworzyć różne ścieżki z nazwami poszczególnych serwerów, udostępniając w ten sposób użytkownikom bardziej dostosowane pliki pomocy. Oprócz tego, kie-

dy używana jest zmienna %S, znaki podkreślenia (_) w żądanym zasobie zostaną zamienione na ukośniki (/).

Wybory przeglądarki

Jak wspomnieliśmy w rozdziale 1, *Poznajemy Sambę*, jeden z komputerów w każdej podsieci przechowuje listę wszystkich aktywnych komputerów. Lista ta nosi nazwę *listy przeglądania*, a przechowujący ją serwer jest nazywany *główną przeglądarką lokalną*. W miarę dołączania i odłączania komputerów główna przeglądarka lokalna uaktualnia informacje na liście przeglądania i udostępnia je żądającym tego komputerom.

Komputer staje się główną przeglądarką lokalną w wyniku wyborów ogłaszanych w lokalnej podsieci. Wybory przeglądarki mogą zostać ogłoszone w dowolnym momencie. Samba może dowolnie fałszować wybory, na przykład po to, aby zawsze zostawać główną przeglądarką lokalną lub nigdy nie przejmować tej funkcji. Na przykład poniższe opcje dodane do pliku konfiguracyjnego z rozdziału 4, *Udziały dyskowe*, zapewnią Sambie wygraną w wyborach na główną przeglądarkę lokalną:

```
[global]
    netbios name = HYDRA
    server string = Samba %v w serwerze (%L)
    workgroup = PROSTA_GRUPA

    # Opcje wyborów przegl□darki
    os level = 34
    local master = yes

    # Opcje konfiguracji sieci
    hosts allow = 192.168.220. 134.213.233. localhost
    hosts deny  = 192.168.220.102
    interfaces  = 192.168.220.100/255.255.255.0 \
                    134.213.233.110/255.255.255.0

    # Rejestrowanie informacji diagnostycznych
    log level = 2
    log file = /var/log/samba.log.%m
    max log size = 30
    debug timestamp = yes

[dane]
    path = /export/samba/dane
    browseable = yes
    guest ok = yes
    comment = Dysk z danymi
    volume = Stacja-Sieciowa
    writeable = yes
```

Załóżmy jednak, że nie chcemy zawsze wygrywać wyborów, ponieważ wolimy ustąpić pola obecnemu w sieci serwerowi Windows NT. Aby to zrobić, musimy wiedzieć, jak odbywają się wybory przeglądarki. Jak już wiesz, każdy komputer biorący udział w wyborach rozgłasza informacje o sobie. Informacje te obejmują:

- wersję używanego protokołu elekcyjnego,
- system operacyjny komputera,

- czas, od którego komputer jest w sieci,
- nazwę hosta.

A oto sposób wyłaniania zwycięzcy. Systemom operacyjnym przypisuje się wartość zależną od ich wersji, jak w tabeli 5.1.

Tabela 5.1. Wartości przypisywane systemom operacyjnym w wyborach przeglądarki

System operacyjny	*Wartość*
Windows NT Server 4.0	33
Windows NT Server 3.51	32
Windows NT Workstation 4.0	17
Windows NT Workstation 3.51	16
Windows 98	2
Windows 95	1
Windows 3.1 for Workgroups	1

Następnie każdemu komputerowi w sieci przypisuje się odrębną wartość zależną od jego funkcji, jak w tabeli 5.2.

Tabela 5.2. Wartości zależne od funkcji komputera

Funkcja	*Wartość*
Podstawowy kontroler domeny	128
Klient WINS	32
Preferowana przeglądarka główna	8
Aktywna przeglądarka główna	4
Przeglądarka awaryjna	2
Aktywna przeglądarka zapasowa	1

Wybory są rozstrzygane w następujący sposób:

1. Zwycięża komputer z najwyższą wersją protokołu elekcyjnego (obecnie nie ma to znaczenia, ponieważ wszystkie klienty Windows posługują się 1. wersją protokołu).
2. Zwycięża komputer z najwyższą wartością systemu operacyjnego.
3. Jeśli jest remis, zwycięża komputer będący preferowaną przeglądarką główną (funkcja 8).
4. Jeśli nadal jest remis, zwycięża komputer będący najdłużej w sieci.
5. Wreszcie, jeśli wciąż jest remis, zwycięża komputer, którego nazwa jest pierwsza w porządku alfabetycznym.
6. Komputer, który zajmie „drugie miejsce", może zostać przeglądarką zapasową.

Tak więc, jeśli chcesz, żeby Samba przejęła funkcję głównej przeglądarki lokalnej, ale tylko wtedy, gdy w sieci nie ma serwera Windows NT 4.0 lub 3.51, powinieneś zmienić parametr `os level` w poprzednim przykładzie na:

```
os level = 31
```

Spowoduje to, że Samba przegra wybory na rzecz serwera Windows NT 4.0 lub 3.51, które mają wyższą wartość systemu operacyjnego. Z drugiej strony, jeśli chciałbyś wyłaniać główną przeglądarkę lokalną na podstawie funkcji komputera (na przykład według tego, który komputer jest podstawowym kontrolerem domeny), mógłbyś ustawić `os level` na wartość odpowiadającą najwyższej wartości systemu operacyjnego obecnego w sieci, i pozwolić, aby protokół elekcyjny rozstrzygnął wybory w kolejnych etapach.

Jak można stwierdzić, czy komputer jest główną przeglądarką lokalną? Służy do tego polecenie `nbtstat`. Umieść NetBIOS-ową nazwę sprawdzanego komputera za opcją `-a`:

```
C:\NC>nbtstat -a hydra

       NetBIOS Remote Machine Name Table

   Name               Type         Status
---------------------------------------------
HYDRA           <00>  UNIQUE      Registered
HYDRA           <03>  UNIQUE      Registered
HYDRA           <20>  UNIQUE      Registered
..__MSBROWSE__.<01>  GROUP       Registered
PROSTA_GRUPA    <00>  GROUP       Registered
PROSTA_GRUPA    <1D>  UNIQUE      Registered
PROSTA_GRUPA    <1E>  GROUP       Registered

MAC Address = 00-00-00-00-00-00
```

Wpis, którego szukasz to `..__MSBROWSE__.<01>`. Wskazuje on, że serwer obecnie działa jako główna przeglądarka lokalna dla bieżącej podsieci. Ponadto, jeśli komputer jest serwerem Samby, możesz poszukać w pliku dziennika *nmbd* wpisu podobnego do poniższego:

```
nmbd/nmbd_become_lmb.c:become_local_master_stage2(406)
*****
Samba name server HYDRA is now a local master browser for
workgroup PROSTA_GRUPA on subnet 192.168.220.100
*****
```

Wreszcie, serwery Windows NT działające jako podstawowe kontrolery domeny dysponują mechanizmem, który w pewnych okolicznościach pozwala im przejąć funkcję głównej przeglądarki lokalnej; jest to bit *preferowanej przeglądarki głównej*. Wcześniej wspomnieliśmy, że Samba również może ustawić ten bit. Włącza się go za pomocą opcji `preferred master`:

```
# Opcje wyborów przegl□darki
os level = 33
local master = yes
preferred master = yes
```

Jeśli ustawiony jest bit preferowanej przeglądarki głównej, komputer wymusi wybory zaraz po uruchomieniu. Oczywiście, jest to potrzebne tylko wtedy, jeśli opcja `os level` odpowiada komputerowi Windows NT. Nie zalecamy korzystania z tej opcji, jeśli inny komputer – na przykład sewer Windows NT – również jest preferowaną przeglądarką.

Główna przeglądarka domeny

W rozdziale 1 wspomnieliśmy, że w celu rozciągnięcia grupy roboczej lub domeny Windows na kilka podsieci jeden z komputerów musi podjąć funkcję *głównej przeglądarki domeny*. Główna przeglądarka domeny rozpowszechnia listy przeglądania między wszystkimi podsieciami w grupie roboczej. Każda główna przeglądarka lokalna okresowo synchronizuje swoją listę przeglądania z główną przeglądarką domeny. Podczas synchronizacji przeglądarka lokalna przekazuje przeglądarce domeny informacje o wszystkich serwerach, których ta nie ma na swojej liście przeglądania, i vice versa. Gdyby świat był doskonały, w końcu każda główna przeglądarka lokalna dysponowałaby listą przeglądania dla całej domeny.

Inaczej niż w przypadku przeglądarek lokalnych, nie przeprowadza się wyborów w celu wyłonienia głównej przeglądarki domeny. Musi wyznaczyć ją administrator. Jednakże Microsoft zdecydował, że główna przeglądarka domeny i podstawowy kontroler domeny (PDC) będą rejestrować typ zasobu <1B>, więc te dwie funkcje są nierozdzielne.

Jeśli w twojej sieci znajduje się serwer Windows NT pełniący funkcję PDC, odradzamy wykorzystywanie Samby jako głównej przeglądarki domeny. To samo dotyczy sytuacji odwrotnej: jeśli Samba wykonuje zadania PDC, powinna być także główną przeglądarką domeny. Choć Samba umożliwia oddzielenie obu funkcji, nie jest to dobry pomysł. Jeśli dwa różne komputery działają jako PDC i główna przeglądarka domeny, w grupie roboczej Windows mogą występować losowe błędy.

Samba może przejąć funkcję głównej przeglądarki domeny we wszystkich podsieciach grupy roboczej dzięki następującej opcji:

```
domain master = yes
```

Możesz upewnić się, że serwer Samby rzeczywiście został główną przeglądarką domeny, zaglądając do pliku dziennika *nmbd*:

```
nmbd/nmbd_become_dmb.c:become_domain_master_stage2(118)
*****
Samba name server HYDRA is now a domain master browser for
workgroup PROSTA_GRUPA on subnet 192.168.220.100
*****
```

Możesz także użyć polecenia `nmblookup` wchodzącego w skład dystrybucji Samby, aby wyszukać niepowtarzalny typ zasobu <1B> w grupie roboczej:

```
# nmblookup PROSTA_GRUPA#1B
Sending queries to 192.168.220.255
192.168.220.100 PROSTA_GRUPA<1b>
```

Grupa robocza w wielu podsieciach

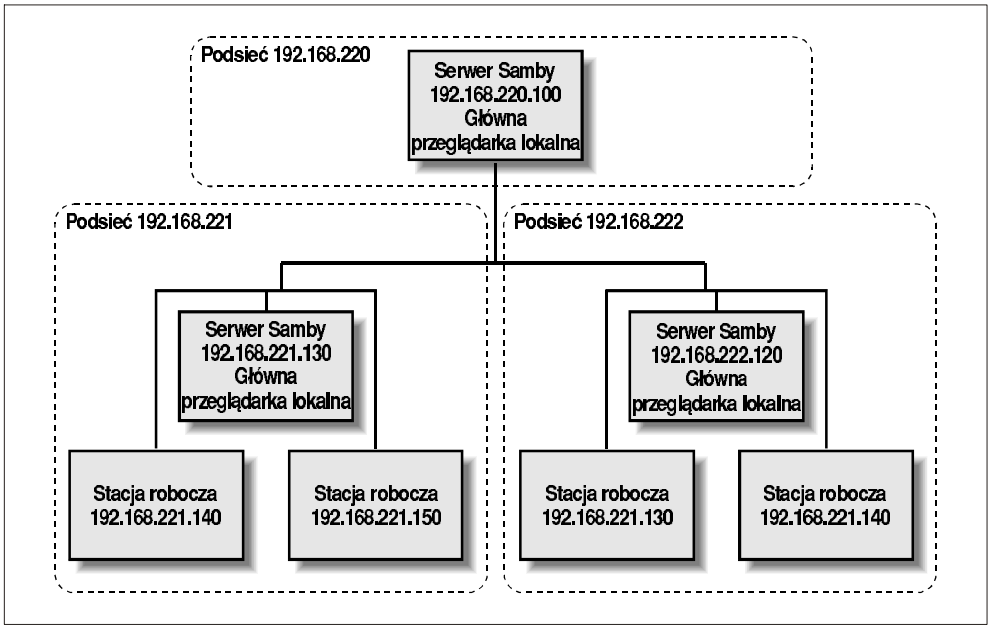
Kiedy tworzysz grupę roboczą lub domenę rozciągającą się na kilka podsieci, pamiętaj o trzech regułach:

- W każdej podsieci grupy roboczej lub domeny musi znajdować się komputer z Windows NT lub Sambą, działający jako główna przeglądarka lokalna (jeśli

w podsieci znajduje się główna przeglądarka domeny, główna przeglądarka lokalna jest niepotrzebna).

- W dowolnym miejscu grupy roboczej musi znajdować się komputer z Windows NT lub Sambą, działający jako główna przeglądarka domeny.
- Każda główna przeglądarka lokalna musi zostać poinstruowana, aby synchronizowała swoje informacje z główną przeglądarką domeny.

Samba dysponuje kilkoma innymi funkcjami z tej dziedziny, które są przydatne, jeśli nie masz albo nie chcesz mieć głównej przeglądarki domeny w swojej sieci. Rozważmy podsieci przedstawione na rysunku 5.1.



Rysunek 5.1. Wiele podsieci z serwerami Samby

Po pierwsze, serwer Samby będący główną przeglądarką lokalną może użyć opcji konfiguracyjnej `remote announce`, dzięki czemu komputery w innych podsieciach będą otrzymywać rozgłaszane komunikaty o obecności serwera. W ten sposób serwer Samby pojawi się na listach przeglądania zdalnych podsieci. Żeby było to możliwe, ukierunkowane rozgłoszenia muszą osiągnąć główną przeglądarkę lokalną w innej podsieci. Pamiętaj, że wiele ruterów domyślnie blokuje ukierunkowane rozgłoszenia; być może będziesz musiał zmienić konfigurację rutera, aby ukierunkowane rozgłoszenia przedostawały się do innych podsieci.

W opcji `remote announce` należy podać podsieci i grupę roboczą, które powinny otrzymywać komunikaty rozgłoszeniowe. Jeśli na przykład komputery znajdujące się w podsieciach 192.168.221 i 192.168.222 oraz w grupie roboczej PROSTA_GRUPA powinny otrzymywać informacje rozgłaszane przez serwer Samby, należy użyć następujących opcji:

```
# Opcje wyborów przegl□darki
os level = 34
local master = yes
remote announce = 192.168.221.255/PROSTA_GRUPA \
    192.168.222.255/PROSTA_GRUPA
```

Oprócz tego możesz podać dokładny adres, na który będą wysyłane komunikaty, pod warunkiem, że główna przeglądarka lokalna w zdalnej podsieci zawsze ma ten sam adres IP.

Serwer Samby działający jako główna przeglądarka lokalna może synchronizować swoją listę przeglądania z innym serwerem Samby, będącym przeglądarką lokalną dla innej podsieci. Załóżmy, że Samba jest skonfigurowana jako główna przeglądarka lokalna, a dwie inne przeglądarki lokalne mają adresy 192.168.221.130 oraz 192.168.222.120. Możemy użyć opcji `remote browse sync`, aby synchronizować listę przeglądania bezpośrednio z innymi serwerami Samby, jak w poniższym przykładzie:

```
# Opcje wyborów przegl□darki
os level = 34
local master = yes
remote browse sync = 192.168.221.130 192.168.220.120
```

Aby opcja ta zadziałała, pozostałe serwery Samby muszą również być głównymi przeglądarkami lokalnymi. W opcji tej możesz użyć także ukierunkowanych rozgłoszeń, jeśli nie znasz dokładnych adresów IP głównych przeglądarek lokalnych.

Opcje przeglądania

W tabeli 5.3. wymieniono czternaście opcji określających sposób, w jaki Samba wykonuje czynności związane z przeglądaniem. Jeśli chcesz, żeby użytkownicy nie mieli problemów ze znalezieniem swoich udziałów dyskowych i drukarek, powinieneś pozostawić wartości domyślne.

Tabela 5.3. Opcje konfiguracji przeglądania

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`announce as`	`NT` lub `Win95` lub `WfW`	Ustawia system operacyjny, za który będzie podawać się Samba	`NT`	Globalny
`announce version`	Wartość liczbowa	Ustawia wersję systemu operacyjnego ogłaszaną przez Sambę	`4.2`	Globalny
`browseable` (`browsable`)	Wartość logiczna	Umożliwia wyświetlanie udziału na liście zasobów komputera	`yes`	Udział
`browse list`	Wartość logiczna	Jeśli jest ustawiona na `yes`, serwer Samby będzie udostępniał listę przeglądania	`yes`	Globalny
`auto services` (`preload`)	Łańcuch (lista udziałów)	Określa listę udziałów, które będą zawsze obecne na liście przeglądania	Brak	Globalny

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`default service (default)`	Łańcuch (nazwa udziału)	Określa udział (usługę), który zostanie udostępniony, jeśli klient zażąda dostępu do udziału nie zdefiniowanego w pliku *smb.conf*	Brak	Globalny
`local master`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba spróbuje zostać główną przeglądarką w lokalnej podsieci	`yes`	Globalny
`lm announce`	`yes`, `no` lub `auto`	Włącza lub wyłącza ogłoszenia o hoście w formacie LAN Managera	`auto`	Globalny
`lm interval`	Wartość liczbowa	Określa częstotliwość (w sekundach), z jaką będą ponawiane ogłoszenia LAN Managera	`60`	Globalny
`preferred master (prefered master)`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba użyje bitu preferowanej przeglądarki głównej, próbując zostać główną przeglądarką lokalną	`no`	Globalny
`domain master`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba spróbuje zostać główną przeglądarką domeny w swojej grupie roboczej	`no`	Globalny
`os level`	Wartość liczbowa	Ustawia poziom systemu operacyjnego Samby podczas wyborów głównej przeglądarki lokalnej	`0`	Globalny
`remote browse sync`	Łańcuch (lista adresów IP)	Wymienia serwery Samby, z którymi należy synchronizować listy przeglądania	Brak	Globalny
`remote announce`	Łańcuch (pary adres/grupa robocza)	Wymienia podsieci i grupy robocze, do których należy wysyłać ukierunkowane rozgłoszenia, aby umieścić serwer Samby na listach przeglądania	Brak	Globalny

announce as

Ta globalna opcja konfiguracyjna określa typ systemu operacyjnego, który Samba ogłasza innym komputerom w sieci. Jej domyślna wartość to `NT`, co oznacza system operacyjny Windows NT. Inne możliwe wartości to `Win95`, oznaczająca system Windows 95, oraz `WfW`, oznaczająca Windows for Workgroups. Możesz zmienić domyślną wartość za pomocą opcji:

```
[global]
    announce as = Win95
```

Nie zalecamy zmieniania domyślnej wartości tej opcji.

announce version

Ta globalna opcja jest często używana w połączeniu z opcją `announce as`; określa wersję systemu operacyjnego, którą Samba będzie ogłaszać innym komputerom w sieci. Jej domyślna wartość to `4.2`, co umieszcza Sambę ponad bieżącą wersją Windows NT 4.0. Możesz podać nową wartość za pomocą następującego wpisu w sekcji globalnej:

```
[global]
    announce version = 4.3
```

Nie zalecamy zmieniania domyślnej wartości tej opcji.

browseable

Opcja `browseable` (z alternatywną pisownią `browsable`) określa, czy wskazany udział powinien pojawiać się na liście zasobów udostępniającego go komputera. Opcja ta ma domyślną wartość `yes`. Jeśli chcesz, aby udział nie był wyświetlany przez klienty, możesz nadać jej wartość `no`.

Zauważ, że nie zapobiegnie to dostępowi do udziału za pomocą innych środków, na przykład przez podanie położenia UNC (`\\serwer\ksiegowosc`) w Eksploratorze Windows. Opcja ta sprawia tylko, że udział nie pojawia się na liście zasobów serwera, gdy użytkownik przegląda tę listę.

browse list

Prawdopodobnie nigdy nie zajdzie potrzeba zmieniania domyślnej wartości tej opcji: `yes`. Jeśli serwer Samby działa jako główna przeglądarka lokalna (to znaczy wygrał wybory przeglądarki), za pomocą globalnej opcji `browse list` możesz poinstruować Sambę, aby nie udostępniała swojej listy przeglądania klientom. Domyślnie Samba zawsze udostępnia swoją listę przeglądania. Możesz temu zapobiec, używając następującej opcji:

```
[global]
    browse list = no
```

Jeśli wyłączysz listę przeglądania, klienty nie będą mogły przeglądać nazw innych komputerów, świadczonych przez nie usług i innych domen dostępnych w sieci. Zauważ, że nie zapobiegnie to dostępowi do innych komputerów; jeśli ktoś zna poprawną nazwę i adres komputera oraz nazwę udziału w tym komputerze, może nadal się z nim połączyć, używając polecenia NET USE lub mapując literę dysku w Eksploratorze Windows. Opcja ta po prostu uniemożliwia klientom uzyskanie informacji z listy przeglądania.

auto services

Globalna opcja `auto services`, nazywana także `preload`, sprawia, że określone udziały są zawsze widoczne na liście przeglądania. Najczęściej używa się jej do ogłaszania konkretnych udziałów dyskowych lub drukarek, tworzonych automatycznie przez udziały `[homes]` lub `[printers]`, ale normalnie nie widniejących na listach przeglądania.

Opcja ta nadaje się zwłaszcza do ogłaszania udziałów dyskowych. Jeśli chcesz umieścić na liście przeglądania wszystkie drukarki systemowe (to znaczy te, które są zdefiniowane w pliku parametrów drukarek), powinieneś skorzystać raczej z opcji `load printers`. Jeśli opcja `browse list` jest ustawiona na `no`, nie zostanie wyświetlony żaden z udziałów wymienionych w opcji `auto services`.

default service

Globalna opcja `default service` (czasem nazywana `default`) określa udział „awaryjny". Jeśli zostanie ustawiona na nazwę istniejącego udziału, a klient zażąda dostępu do nieistniejącego udziału lub drukarki, Samba spróbuje połączyć go z udziałem określonym w tej opcji. Używa się jej następująco:

```
default service = pomoc
```

Zauważ, że nazwy udziału `pomoc` nie umieszcza się w nawiasach kwadratowych, choć podczas definiowania tego udziału w dalszej części pliku konfiguracyjnego nawiasy zostaną użyte. Oprócz tego, jeśli użyjesz zmiennej `%S` w udziale wskazanym za pomocą tej opcji, będzie ona reprezentować żądany, nieistniejący udział, a nie usługę domyślną. Wszystkie znaki podkreślenia (_) w żądanym udziale zostaną zamienione na ukośniki (/).

local master

Ta globalna opcja określa, czy Samba zaraz po uruchomieniu spróbuje zostać główną przeglądarką lokalną w swojej podsieci. Jeśli opcja ta jest ustawiona na `yes`, Samba weźmie udział w wyborach. Jednakże włączenie tylko tej opcji nie gwarantuje zwycięstwa (pomogą w tym inne parametry, takie jak `preferred master` i `os level`). Jeśli opcja ta jest ustawiona na `no`, Samba zawsze przegra wybory przeglądarek, niezależnie od ustawień innych opcji konfiguracyjnych. Jej domyślna wartość to `yes`.

lm announce

Globalna opcja `lm announce` informuje proces *nmbd*, czy w imieniu serwera należy wysyłać ogłoszenia LAN Managera o hoście. Takich ogłoszeń o hoście mogą wymagać starsze klienty, na przykład system operacyjny OS/2 IBM-a. Pozwalają one na dodanie serwera do listy przeglądania klienta. Jeśli włączysz tę opcję, Samba będzie ogłaszać swoją obecność w okresach ustalonych opcją `lm interval`.

Opcja ta może przyjmować standardowe wartości logiczne, `yes` i `no`, które włączają i wyłączają ogłoszenia LAN Managera. Do dyspozycji jest także trzecie ustawienie, `auto`, które sprawia, że proces *nmbd* pasywnie czeka na ogłoszenia LAN Managera, ale początkowo sam ich nie wysyła. Gdy wykryje ogłoszenia LAN Managera wysyłane przez inny komputer w sieci, zaczyna wysyłać własne, aby być widzianym przez tamten komputer. Opcji tej używa się następująco:

```
[global]
    lm announce = yes
```

Domyślna wartość tej opcji to `auto`. Prawdopodobnie nie będziesz musiał jej zmieniać.

lm interval

Ta opcja, używana w połączeniu z opcją `lm announce`, określa liczbę sekund między wysyłaniem kolejnych ogłoszeń LAN Managera przez proces *nmbd*. Żeby użyć tej opcji, trzeba najpierw uaktywnić ogłoszenia LAN Managera. Jej domyślna wartość to 60 sekund. Jeśli ustawisz ją na 0, Samba nie będzie wysyłać żadnych ogłoszeń LAN Managera, niezależnie od ustawienia opcji `lm announce`. Możesz zmienić jej wartość w następujący sposób:

```
[global]
    lm interval = 90
```

preferred master

Opcja `preferred master` nakazuje Sambie ustawić bit preferowanej przeglądarki głównej podczas udziału w wyborach. Dzięki temu komputer uzyskuje w grupie roboczej wyższą preferencję niż inne komputery o tym samym poziomie systemu operacyjnego. Jeśli chcesz, żeby komputer z Sambą został główną przeglądarką lokalną, powinieneś użyć poniższej opcji:

```
[global]
    preferred master = yes
```

W przeciwnym wypadku powinieneś pozostawić wartość domyślną `no`. Jeśli Samba zostanie skonfigurowana jako preferowana przeglądarka główna, wymusi wybory podczas włączania się do sieci.

os level

Globalna opcja `os level` określa poziom systemu operacyjnego, który Samba będzie pozorować podczas wyborów przeglądarki. Jeśli chcesz, żeby Samba wygrała wybory i została główną przeglądarką, powinieneś ustawić tę opcję na wartość wyższą od najwyższego poziomu systemu operacyjnego znajdującego się aktualnie w sieci. Wartości podano w tabeli 5.1. Domyślny poziom jest równy 0, co sprawia, że Samba zawsze przegrywa wybory. Jeśli chcesz, żeby Samba zawsze wygrywała wybory, powinieneś ustawić tę opcję jak niżej:

```
os level = 34
```

Oznacza to, że serwer podczas wyborów będzie głosował na siebie 34 razy, co zapewni mu zwycięstwo.

domain master

Jeśli Samba jest podstawowym kontrolerem domeny w twojej grupie roboczej lub domenie NT, powinna być także główną przeglądarką domeny. Główna przeglądarka domeny to specjalny komputer z typem zasobu NetBIOS-u <1B>, który wymienia listy przeglądania z przeglądarkami lokalnymi w poszczególnych podsieciach domeny. Aby Samba przejęła funkcję głównej przeglądarki domeny, należy ustawić następującą opcję w sekcji `[global]` pliku *smb.conf*:

```
[global]
    domain master = yes
```

Jeśli w twojej sieci jest serwer Windows NT działający jako podstawowy kontroler domeny (PDC), odradzamy wykorzystanie Samby jako głównej przeglądarki domeny. To samo dotyczy sytuacji odwrotnej: jeśli Samba wykonuje zadania PDC, powinna być także główną przeglądarką domeny. Oddzielenie funkcji PDC i głównej przeglądarki domeny może spowodować występowanie nieprzewidywalnych błędów w sieci.

remote browse sync

Globalna opcja `remote browse sync` informuje Sambę, że powinna synchronizować swoją listę przeglądania z głównymi przeglądarkami lokalnymi w innych podsieciach. Listy przeglądania mogą być synchronizowane tylko z innymi serwerami Samby, a nie z komputerami Windows. Jeśli na przykład twój serwer Samby jest główną przeglądarką w podsieci 192.168.235, a pod adresami 192.168.234.92 i 192.168.236.2 w pozostałych podsieciach znajdują się dwie inne główne przeglądarki lokalne, mógłbyś użyć następującej opcji:

```
remote browse sync = 192.168.234.92 192.168.236.2
```

Serwer Samby kontaktowałby się wówczas bezpośrednio z komputerami o wskazanych adresach i synchronizował z nimi listy przeglądania. Mógłbyś też napisać:

```
remote browse sync = 192.168.234.255 192.168.236.255
```

Samba użyłaby wówczas zapytań rozgłoszeniowych w celu ustalenia adresów IP głównych przeglądarek lokalnych w obu podsieciach, z którymi następnie synchronizowałaby listy przeglądania. Takie rozwiązanie jest jednak możliwe tylko wtedy, jeśli twój ruter nie blokuje ukierunkowanych rozgłoszeń wysyłanych na adres kończący się liczbą 255.

remote announce

Serwery Samby mogą udostępniać listy przeglądania w zdalnych podsieciach dzięki opcji `remote announce`. Są one zwykle wysyłane do głównej przeglądarki lokalnej w zdalnej podsieci. Jeśli jednak nie znasz adresu głównej przeglądarki lokalnej, możesz skorzystać z następującej opcji:

```
[global]
    remote announce = 192.168.234.255/KSIEGOWOSC \
                      192.168.236.255/KSIEGOWOSC
```

Samba będzie wówczas wysyłać komunikaty rozgłoszeniowe o swojej obecności do wszystkich komputerów w podsieciach 192.168.234 i 192.168.236, aby skontaktować się z ich głównymi przeglądarkami lokalnymi. Możesz też podać dokładne adresy IP przeglądarek, jeśli je znasz.

Różnice w systemach plików

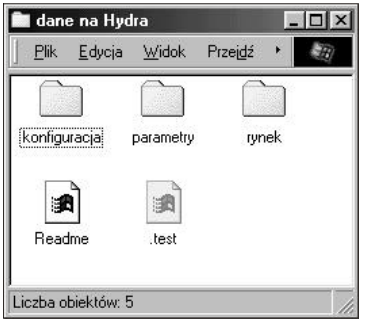
Jednym z najpoważniejszych zadań stojących przed Sambą jest korygowanie różnic między uniksowymi i nieuniksowymi systemami plików. Chodzi tu o obsługę dowiązań symbolicznych, plików ukrytych i plików „z kropką". Przy niepoprawnej

konfiguracji problemem mogą być też prawa dostępu do plików. W tym podrozdziale opiszemy, jak Samba radzi sobie z tymi irytującymi różnicami i jak dodaje własne funkcje.

Ukrywanie i wetowanie plików

W pewnych okolicznościach zależy nam na tym, żeby użytkownik nie mógł zobaczyć ani otworzyć pliku. W innych wypadkach nie chcemy odmawiać użytkownikowi dostępu do pliku – chcemy tylko ukryć plik przed użytkownikiem, kiedy ten przegląda zawartość katalogu. W systemach Windows pliki mają atrybut, który pozwala na ich ukrycie podczas wyświetlania zawartości katalogu. W Uniksie tradycyjną metodą ukrywania plików w katalogu jest nadanie im nazw zaczynających się od kropki (.). Dzięki temu podczas wykonywania zwykłego polecenia `ls` nie są wyświetlane pliki konfiguracyjne lub pliki z parametrami domyślnymi. Jeśli jednak chcemy w ogóle pozbawić użytkownika możliwości dostępu do pliku, musimy posługiwać się zezwoleniami plikowymi i katalogowymi.

Pierwsza opcja, o której powinniśmy wspomnieć, to `hide dot files`. Opcja ta robi dokładnie to, na co wskazuje jej nazwa – jeśli jest ustawiona na `yes`, pliki z kropką są traktowane jako ukryte. Jeśli jest ustawiona na `no`, pliki te są zawsze wyświetlane. Należy pamiętać, że pliki te nie są niedostępne, a tylko ukryte. Jeśli użytkownik zechce obejrzeć ukryte pliki podczas przeglądania katalogu (na przykład używając polecenia Opcje z menu Widok w folderze Windows 98), wówczas pliki te zostaną wyświetlone (patrz rysunek 5.2).

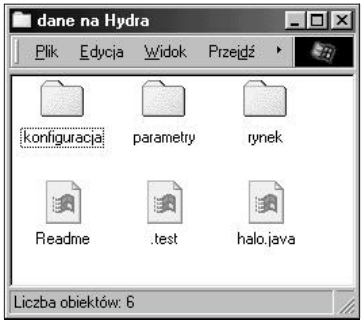


Rysunek 5.2. Pliki ukryte w udziale [dane]

Jeśli chcesz ukrywać nie tylko pliki zaczynające się od kropki, dzięki opcji `hide files` możesz podać wzorzec nazw, które Samba powinna ukrywać. Przypuśćmy, że w przykładowm udziale `[dane]` wpisaliśmy poniższą opcję:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    case sensitive = no
    hide files = /*.java/*README*/
```

Każdy wpis w tej opcji musi zaczynać się, kończyć lub być oddzielony od innego znakiem ukośnika (/), nawet wtedy, gdy podany jest tylko jeden wzorzec. Konwencja ta pozwala na używanie spacji w nazwach plików. W tym przykładzie katalog udziału wyglądałby tak, jak na rysunku 5.3. Zauważ, że i tym razem wybraliśmy opcję wyświetlania w oknie plików ukrytych.



Rysunek 5.3. Ukrywanie plików według wzorca nazwy

Jeśli chcielibyśmy całkowicie zapobiec wyświetlaniu plików przez użytkownika, moglibyśmy zamiast tego użyć opcji `veto files`. Opcja ta ma taką samą składnię jak `hide files` i określa pliki, których użytkownik nigdy nie powinien widzieć. Zmieńmy na przykład udział `[dane]` w następujący sposób:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    case sensitive = no
    veto files = /*.java/*README*/
```

Składnia tej opcji jest taka sama jak opcji `hide files`: każdy wpis musi zaczynać się, kończyć lub być oddzielony od innego znakiem ukośnika (/), nawet wtedy, gdy podany jest tylko jeden wzorzec. W tym przypadku pliki `hello.java` i `README` po prostu znikną z katalogu i użytkownik nie będzie miał do nich dostępu przez protokół SMB.

Pozostała jeszcze jedna kwestia. Co się stanie, jeśli użytkownik spróbuje usunąć katalog z zawetowanymi plikami? Określa to opcja `delete veto files`. Jeśli ta opcja jest ustawiona na `yes`, użytkownik będzie mógł usunąć z katalogu zarówno zwykłe, jak i zawetowane pliki, więc katalog zostanie usunięty. Jeśli jest ustawiona na `no`, użytkownik nie będzie mógł usunąć zawetowanych plików, a w konsekwencji także samego katalogu. Z perspektywy użytkownika katalog będzie wydawał się pusty, ale niemożliwy do usunięcia.

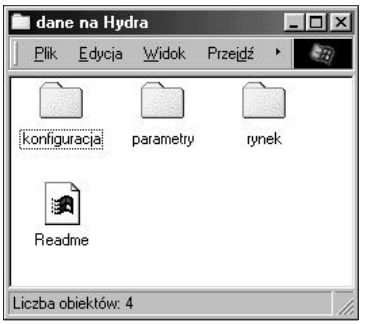
Dyrektywa `dont descend` określa listę katalogów, których zawartość powinna być niewidoczna dla użytkowników. Zauważ, że mówimy tu o *zawartości*, a nie o samym katalogu. Użytkownicy będą mogli przejść do oznaczonego w ten sposób katalogu, ale nie będą mogli zejść na niższe poziomy drzewa katalogów – zobaczą tylko pusty folder. Spróbujmy użyć tej opcji w udziale zdefiniowanym wcześniej w tym rozdziale:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    case sensitive = no
    dont descend = konfiguracja parametry
```

Oprócz tego załóżmy, że katalog */home/samba/data* ma następującą zawartość:

```
drwxr-xr-x  6 tom      users    1024 Jun 13 09:24 .
drwxr-xr-x  8 root     root     1024 Jun 10 17:53 ..
-rw-r--r--  2 tom      users    1024 Jun  9 11:43 README
drwxr-xr-x  3 tom      users    1024 Jun 13 09:28 konfiguracja
drwxr-xr-x  3 tom      users    1024 Jun 13 09:28 parametry
drwxr-xr-x  3 tom      users    1024 Jun 13 09:28 rynek
```

Jeśli użytkownik połączy się z udziałem, zobaczy katalogi pokazane na rysunku 5.4. Jednakże katalogi */konfiguracja* i */parametry* będą wyglądały na puste, jeśli nawet istniałyby w nich pliki lub podkatalogi. Użytkownik nie będzie też mógł zapisać żadnych danych w folderze (dzięki czemu nie utworzy pliku lub katalogu o nazwie identycznej z już istniejącym, choć niewidocznym). Jeśli spróbuje to zrobić, zobaczy komunikat odmowy dostępu. Opcja `dont descend` to opcja administracyjna, a nie opcja bezpieczeństwa, i nie należy traktować jej jak substytutu praw dostępu.



Rysunek 5.4. Zawartość udziału [dane] z opcją dont descend

Dowiązania

W systemach plików DOS-a i Windows NT nie ma dowiązań symbolicznych. Systemy Windows 95/98/NT używają zamiast nich „skrótów". Kiedy więc klient spróbuje otworzyć dowiązanie symboliczne w udziale serwera Samby, Samba podąża za dowiązaniem, aby znaleźć prawdziwy plik i umożliwić klientowi jego otworzenie, zupełnie tak, jakby użytkownik pracował na komputerze uniksowym. Jeśli nie chcesz na to pozwolić, ustaw opcję `follow symlinks`:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    case sensitive = no
    follow symlinks = no
```

Możesz przetestować tę konfigurację, tworząc katalog w serwerze uniksowym wewnątrz udziału, do którego się logujesz. Wpisz następujące polecenia:

```
% mkdir witaj; cd witaj
% cat "To jest test" > halo
% ln -s halo "Dowiazanie do pliku halo"
```

Dzięki temu powstaną dwa pliki pokazane w oknie na rysunku 5.5. Zwykle, jeśli klikniesz którykolwiek z nich, powinieneś uzyskać dostęp do pliku z tekstem „To jest test". Jeśli jednak opcja `follow symlinks` jest ustawiona na `no`, po kliknięciu dowiązania do pliku *witaj.txt* powinieneś otrzymać komunikat o błędzie, jak na rysunku 5.5.

Rysunek 5.5. Okno dialogowe wyświetlane po próbie otwarcia dowiązania symbolicznego, kiedy jest to zabronione przez Sambę

Przyjrzyjmy się jeszcze opcji `wide links`. Opcja ta, jeśli jest ustawiona na `yes`, pozwala użytkownikowi na podążanie za dowiązaniami symbolicznymi wskazującymi na zewnątrz współdzielonego drzewa katalogów. Załóżmy, że zmodyfikowalibyśmy udział `[dane]` w następujący sposób:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    case sensitive = no
    follow symlinks = no
    wide links = yes
```

Jeśli włączona jest także opcja `follow symlinks`, konfiguracja ta pozwoli Sambie na podążanie za dowiązaniami symbolicznymi wskazującymi poza drzewo katalogów bieżącego udziału. Jeśli utworzymy plik na zewnątrz udziału (na przykład w czyimś katalogu macierzystym), a następnie w udziale utworzymy dowiązanie do niego:

```
ln -s ~tomek/plikdanych ./plikdanych
```

wówczas będziemy mogli otworzyć plik w katalogu Tomka, jeśli tylko pozwalają na to prawa dostępu.

Opcje systemu plików

W tabeli 5.4 znajduje się podsumowanie omówionych opcji. Zalecamy użycie wartości domyślnych, z wyjątkiem tych przypadków, o których mowa w zamieszczonych niżej opisach.

Tabela 5.4. Opcje konfiguracji systemu plików

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`unix realname`	Wartość logiczna	Udostępnia klientowi nazwisko uniksowego użytkownika	`no`	Globalny
`dont descend`	Łańcuch (lista katalogów)	Określa listę katalogów, których zawartość powinna być niewidoczna dla klientów	Brak	Udział
`follow symlinks`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba nie pozwala używać dowiązań symbolicznych	`yes`	Udział
`getwd cache`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba będzie buforować wywołania `getwd()`	`yes`	Globalny
`wide links`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba będzie podążać za dowiązaniami na zewnątrz udziału	`yes`	Udział
`hide dot files`	Wartość logiczna	Jeśli jest ustawiona na `yes`, ukryte pliki uniksowe są traktowane jako ukryte także w Windows	`yes`	Udział
`hide files`	Łańcuch (lista plików)	Lista wzorców nazw plików, które należy traktować jako ukryte	Brak	Udział
`veto files`	Łańcuch (lista plików)	Lista wzorców nazw plików, które nigdy nie są wyświetlane	Brak	Udział
`delete veto files`	Wartość logiczna	Jeśli jest ustawiona na `yes`, pliki określone opcją `veto files` zostaną usunięte podczas usuwania katalogu, w którym się znajdują	`no`	Udział

unix realname

Niektóre programy potrzebują do działania pełnego nazwiska użytkownika. Na przykład programy e-mail dla Windows często muszą skojarzyć nazwę użytkownika z jego prawdziwym nazwiskiem. Jeśli twój systemowy plik haseł zawiera nazwiska użytkowników w polu GECOS, opcja `unix realname` nakazuje Sambie udostępnianie tych informacji klientom. Bez nich nazwą użytkownika będzie po prostu jego identyfikator używany przy logowaniu. Jeśli na przykład uniksowy plik haseł zawiera linię:

```
jkowalski:/KaBfco47Rer5:500:500:Jan Kowalski:/home/jkowalski:/bin/sh
```

a w pliku konfiguracyjnym znajduje się opcja:

```
[global]
    unix realname = yes
```

wówczas każdemu klientowi, który poprosi o podanie prawdziwego nazwiska użytkownika `jkowalski` zostanie przesłane nazwisko Jan Kowalski. Zwykle nie ma potrzeby korzystania z tej opcji.

dont descend

Opcja `dont descend` służy do określania katalogów, które z perspektywy klienta powinny wydawać się puste. Pamiętaj, że sam katalog będzie widoczny, ale Samba nie wyświetli jego zawartości. Opcja ta nie jest pomyślana jako mechanizm bezpieczeństwa (użytkownik prawdopodobnie zdołałby ją obejść), ale raczej jako udogodnienie, które ma powstrzymać klientów od zaglądania do katalogów z plikami, które nie powinny być modyfikowane. Przykład użycia tej opcji znajduje się wcześniej w tym podrozdziale.

follow symlinks

Opcja ta, omówiona szczegółowo wcześniej, określa, czy Samba powinna podążać za uniksowymi dowiązaniami symbolicznymi do docelowego pliku, czy też zgłosić błąd użytkownikowi. Jeśli jest ustawiona na `yes`, dowiązanie zostanie zinterpretowane jako plik.

getwd cache

Ta globalna opcja określa, czy Samba będzie używać lokalnego bufora dla uniksowego wywołania systemowego `getwd()` (zwracającego nazwę bieżącego katalogu roboczego). Możesz zmienić jej domyślną wartość (`yes`) w następujący sposób:

```
[global]
    getwd cache = no
```

Ustawienie tej opcji na `yes` może znacznie wydłużyć czas określania bieżącego katalogu roboczego, zwłaszcza wtedy, kiedy opcja `wide links` jest ustawiona na `no`. Zwykle nie ma potrzeby zmieniania tej opcji.

wide links

Opcja ta określa, czy klient może używać dowiązań symbolicznych wskazujących na zewnątrz współdzielonego drzewa katalogów. Dotyczy to wszystkich plików i katalogów na drugim końcu dowiązania, jeśli tylko użytkownik ma odpowiednie prawa dostępu. Domyślna wartość tej opcji to `yes`. Opcja ta nie będzie honorowana, jeśli opcja `follow symlinks` jest ustawiona na `no`. Ustawienie wartości `no` powoduje znaczne spowolnienie demona *smbd*.

hide files

W opcji `hide files` określa się wzorce nazw katalogów i plików. Wszystkie pliki pasujące do tego wzorca będą traktowane przez klienta jako ukryte. Odpowiada to ustawieniu dosowego atrybutu ukrycia pliku, co niekoniecznie znaczy, że użytkownik nie zobaczy pliku podczas przeglądania katalogu.

Każdy wpis na liście musi zaczynać się, kończyć lub być oddzielony od innego wpisu znakiem ukośnika (/), nawet wtedy, gdy podany jest tylko jeden wzorzec. Dzięki

temu nazwy na liście mogą zawierać spacje. Można używać gwiazdek, reprezentujących zero lub więcej znaków, oraz znaków zapytania, reprezentujących dokładnie jeden znak. Na przykład:

```
hide files = /.jav*/README.???/
```

hide dot files

Opcja `hide dot files` ukrywa wszystkie pliki, których nazwa zaczyna się od kropki (.), naśladując w ten sposób działanie kilku poleceń powłoki w Uniksie. Podobnie jak w przypadku opcji `hide files`, pliki zaczynające się od kropki mają ustawiony dosowy atrybut ukrycia, co nie gwarantuje, że klient ich nie zobaczy. Domyślna wartość tej opcji to `yes`.

veto files

Pliki wymienione w opcji `veto files` są bardziej chronione niż pliki ukryte. Samba nie informuje nawet o ich istnieniu, a klienty nie mogą ich listować ani otwierać. Nie jest to jednak godna zaufania opcja bezpieczeństwa. Zapobiega ona raczej usuwaniu przez programy PC pewnych specjalnych plików, na przykład tych służących do przechowywania rozwidlenia zasobów pliku Macintosha w uniksowym systemie plików. Jeśli Windows i Macintosh dzielą te same pliki, opcja `veto files` może powstrzymać domorosłych ekspertów przed usunięciem plików potrzebnych użytkownikom Macintoshy.

Składnia tej opcji jest dokładnie taka sama, jak opcji `hide files`: każdy wpis na liście musi zaczynać się, kończyć lub być oddzielony od innego znakiem ukośnika (/), nawet wtedy, gdy podany jest tylko jeden wzorzec. Można używać gwiazdek, reprezentujących zero lub więcej znaków, oraz znaków zapytania, reprezentujących dokładnie jeden znak. Na przykład:

```
veto files = /*konfig*/*parametr?/
```

Opcja ta jest narzędziem administracyjnym i nie może zastąpić odpowiednio dobranych praw dostępu.

delete veto files

Opcja ta zezwala Sambie na pozbycie się zawetowanych plików podczas usuwania przechowującego je katalogu. Jej domyślna wartość to `no`. Oznacza to, że próba usunięcia katalogu z zawetowanym plikiem zakończy się niepowodzeniem. Katalog pozostanie na dysku, choć z perspektywy użytkownika będzie wyglądał na pusty. Jeśli opcja jest ustawiona na `yes`, katalog i zawetowane pliki zostaną usunięte.

Prawa dostępu i atrybuty plików w systemach MS-DOS i Unix

DOS nigdy nie miał być wielodostępnym, sieciowym systemem operacyjnym, natomiast Unix był zaprojektowany w ten sposób od samego początku. W obsłudze ich systemów plików występuje więc sporo rozbieżności i luk, których Samba nie tylko musi być świadoma, ale którym musi także zaradzić. Jedną z największych różnic między Uniksem i DOS-em jest obsługa praw dostępu do plików.

Zobaczmy, w jaki sposób Unix przypisuje plikom prawa dostępu. Wszystkie pliki Uniksa mają zezwolenia na odczyt, zapis i wykonanie dla trzech kategorii użytkowników: właściciela, grupy i „reszty świata". Zezwolenia te można obejrzeć w skrajnej lewej kolumnie wyników polecenia `ls -al` wydanego w uniksowym katalogu. Na przykład:

```
-rwxr--r--  1 tomek    users    2014 Apr 13 14:11 access.conf
```

Windows dysponuje natomiast czterema bitami, które przypisuje wszystkim plikom: tylko do odczytu, systemowy, ukryty i archiwalny. Można obejrzeć ustawienia tych bitów, klikając plik prawym przyciskiem myszy i wybierając z menu polecenie Właściwości. Powinno ukazać się okno dialogowe podobne do tego z rysunku 5.6*.

Rysunek 5.6. Atrybuty plików DOS-a i Windows

Oto definicje poszczególnych bitów:

Tylko do odczytu
: Użytkownik może odczytać zawartość pliku, ale nie może nic w nim zapisać.

Systemowy
: Ten plik pełni specjalną funkcję w systemie operacyjnym.

Ukryty
: Plik został oznaczony jako niewidoczny dla użytkownika, chyba że ten jawnie zażąda jego wyświetlenia.

* Pole wyboru Systemowy prawdopodobnie będzie szare i nieaktywne. Nie przejmuj się tym – powinieneś bez trudu dostrzec, czy jest ono zaznaczone, czy też nie.

Archiwalny
Plik został zmodyfikowany od czasu sporządzenia jego dosowej kopii zapasowej.

Jak widać, żaden z bitów nie wskazuje, że plik jest wykonywalny. Systemy plików DOS-a i Windows NT identyfikują pliki wykonywalne na podstawie rozszerzeń .EXE, .COM, .CMD i .BAT.

Z tego wynika, że trzy uniksowe bity wykonywalności nie mają żadnego zastosowania w odniesieniu do pliku przechowywanego w udziale dyskowym Samby. Pliki dosowe mają jednak własne atrybuty, które powinny zostać zachowane podczas przechowywania ich w środowisku Uniksa: archiwalne, systemowe i ukryte. Samba może zachować te atrybuty, wykorzystując bity wykonywalności pliku uniksowego – jeśli tego się od niej zażąda. Odwzorowywanie bitów wiąże się jednak z niepożądanym efektem ubocznym: jeśli użytkownik Windows zapisze plik w udziale Samby, a ty obejrzysz go w Uniksie za pomocą polecenia `ls -al`, niektóre bity wykonywalności będą oznaczać coś innego niż to, do czego przywykłeś.

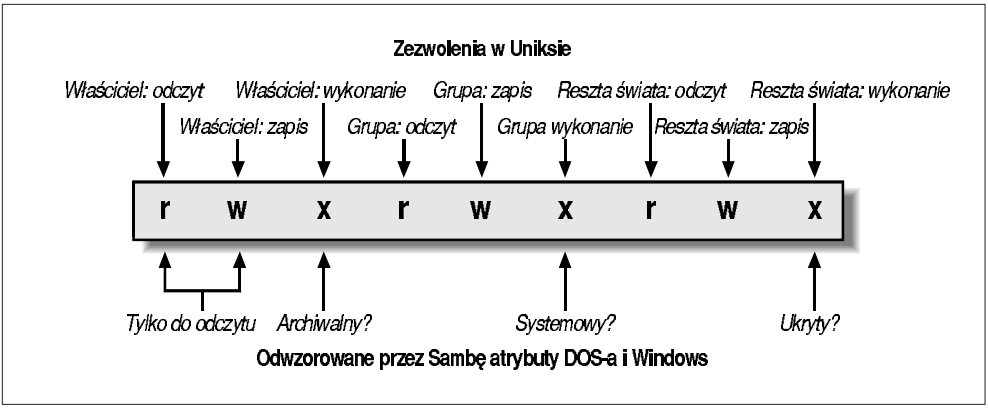
O odwzorowywaniu bitów decydują trzy opcje Samby: `map archive`, `map system` i `map hidden`. Opcje te umożliwiają odwzorowywanie atrybutów archiwalnego, systemowego i ukrytego na bity wykonywalności dla – odpowiednio – właściciela, grupy i reszty świata. Możesz dodać te opcje do udziału `[dane]`, ustawiając ich wartości w następujący sposób:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    map archive = yes
    map system = yes
    map hidden = yes
```

Następnie spróbuj utworzyć plik w Uniksie, na przykład o nazwie `witaj.java`, i zmienić jego zezwolenia na 755. Jeśli ustawiłeś opcje jak w przykładzie powyżej, po sprawdzeniu atrybutów pliku w Windows powinieneś zauważyć, że w oknie Właściwości zaznaczone są trzy pola atrybutów pliku. A co z atrybutem Tylko do odczytu? Domyślnie Samba 2.0 ustawia ten atrybut zawsze wtedy, gdy plik nie ma ustawionego uniksowego zezwolenia na odczyt dla właściciela. Innymi słowy, możesz ustawić ten atrybut, zmieniając zezwolenia pliku na 555.

Powinniśmy cię ostrzec, że domyślną wartością opcji `map archive` jest `yes`, natomiast pozostałe dwie mają domyślną wartość `no`. Wynika to z faktu, że wiele programów nie działa poprawnie, jeśli z plikami DOS-a i Windows nie jest zapisany właściwy bit archiwalny. Atrybuty systemowy i ukryty nie mają natomiast większego wpływu na działanie programów, więc decyzję o ich użyciu pozostawia się administratorowi.

Rysunek 5.7 przedstawia bity zezwoleń uniksowych i ich odwzorowanie na atrybuty DOS-a. Zauważ, że bity odczytu i zapisu dla grupy i reszty świata nie przekładają się bezpośrednio na atrybuty DOS-a, ale zachowują swoje pierwotne uniksowe znaczenie w serwerze Samby.



Rysunek 5.7. Znaczenie praw dostępu do pliku w Uniksie i Sambie

Maski tworzenia plików

Samba ma kilka opcji pomagających w obsłudze masek tworzenia plików. Maski te określają początkowe prawa dostępu do nowo tworzonych plików i katalogów. W Uniksie daje to możliwość selektywnego wyłączania praw dostępu do tworzonych plików i katalogów. W przypadku plików używanych przez Windows oznacza to, że możesz wyłączać również atrybuty: tylko do odczytu, archiwalny, systemowy i ukryty.

Na przykład poniższa opcja `create mask` stanowi, że prawa dostępu do plików tworzonych przez klienty Windows będą równe najwyżej 744:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    create mask = 744
```

natomiast użyta poniżej opcja `directory mask` sprawi, że prawa dostępu do nowo tworzonych katalogów będą równe najwyżej 755:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes
    directory mask = 755
```

Możesz także wymusić ustawianie różnych bitów za pomocą opcji `force create mode` i `force directory mode`. Wartości określone w tych opcjach zostaną poddane logicznej operacji OR z maskami tworzenia plików i katalogów, dzięki czemu wskazane bity będą zawsze ustawione. Opcje te mają zwykle zasięg globalny, zapewniając właściwe ustawianie zezwoleń na zapis i odczyt dla grupy oraz reszty świata w plikach lub katalogach tworzonych we wszystkich udziałach.

Podobnie, jeśli chciałbyś jawnie określić uniksowego użytkownika i grupę plików tworzonych przez klienty Windows, możesz posłużyć się opcjami `force user` i `force group`. Na przykład:

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes

    create mask = 744
    directory mask = 755
    force user = jacek
    force group = ksiegowosc
```

Opcje te w istocie statycznie przypisują uniksowego użytkownika i grupę każdemu połączeniu z udziałem. Dzieje się to jednak już po uwierzytelnieniu klienta, więc swobodny dostęp do udziału jest nadal niemożliwy. Opcje te są często używane ze względu na efekty uboczne: przypisywanie określonego użytkownika i grupy wszystkim plikom tworzonym w udziale. Korzystaj z nich według własnego uznania.

Kolejną cechą Uniksa, której brakuje DOS-owi, jest możliwość usunięcia z zapisywalnego katalogu pliku przeznaczonego tylko do odczytu. Jeśli uniksowy katalog jest zapisywalny, można usuwać z niego pliki przeznaczone tylko do odczytu. Dzięki temu użytkownicy mogą usuwać dowolne pliki ze swoich katalogów, nawet wtedy, gdy umieścił je tam ktoś inny.

System plików DOS-a nie został zaprojektowany pod kątem wielu użytkowników, więc jego twórcy zdecydowali, że „przeznaczony tylko do odczytu" będzie znaczyć „chroniony przed przypadkową zmianą, w tym usunięciem", a nie „chroniony przed innym użytkownikiem tego samego komputera", i uniemożliwili usuwanie takich plików. Nawet dzisiaj systemy plików Windows zachowują się w taki sposób.

Zwykle nie ma w tym nic złego. Programy Windows nie próbują usuwać plików przeznaczonych tylko do odczytu, ponieważ wiedzą, że to kiepski pomysł. Jednakże niektóre narzędzia służące do kontroli kodu źródłowego – napisane początkowo dla Uniksa i przeniesione do Windows – muszą mieć możliwość usuwania plików przeznaczonych tylko do odczytu. Samba będzie na to zezwalać, jeśli ustawisz opcję `delete readonly` na `yes`.

```
[dane]
    path = /home/samba/dane
    browseable = yes
    guest ok = yes
    writeable = yes

    create mask = 744
    directory mask = 755
    force user = jacek
    force group = ksiegowosc
    delete readonly = yes
```

Opcje praw dostępu do plików i katalogów

Opcje praw dostępu do plików i katalogów zebrano w tabeli 5.5. Poniżej każda opcja jest szczegółowo opisana.

Tabela 5.5. Opcje praw dostępu do plików i katalogów

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`map archive`	Wartość logiczna	Zachowuje dosowy atrybut archiwalny w bicie wykonywalności dla użytkownika (0100)	`yes`	Udział
`map system`	Wartość logiczna	Zachowuje dosowy atrybut systemowy w bicie wykonywalności dla grupy (0010)	`no`	Udział
`map hidden`	Wartość logiczna	Zachowuje dosowy atrybut ukrycia w bicie wykonywalności dla reszty świata (0001)	`no`	Udział
`create mask` (`create mode`)	Wartość liczbowa	Ustawia maksymalną wartość zezwoleń dla plików tworzonych przez Sambę	`0744`	Udział
`directory mask` (`directory mode`)	Wartość liczbowa	Ustawia maksymalną wartość zezwoleń dla katalogów tworzonych przez Sambę	`0755`	Udział
`force create mode`	Wartość liczbowa	Wymusza ustawienie określonych zezwoleń (bitowa suma logiczna OR) dla plików tworzonych przez Sambę	`0000`	Udział
`force directory mode`	Wartość liczbowa	Wymusza ustawienie określonych zezwoleń (bitowa suma logiczna OR) dla katalogów tworzonych przez Sambę	`0000`	Udział
`force group` (`group`)	Łańcuch (nazwa grupy)	Ustawia obowiązującą grupę dla użytkownika korzystającego z udziału	Brak	Udział
`force user`	Łańcuch (nazwa użytkownika)	Ustawia obowiązującą nazwę użytkownika korzystającego z udziału	Brak	Udział
`delete readonly`	Wartość logiczna	Pozwala użytkownikowi usuwać z zapisywalnego katalogu pliki przeznaczone tylko do odczytu	`no`	Udział

create mask

Argumentem tej opcji jest ósemkowa liczba określająca zezwolenia, które mogą zostać ustawione podczas tworzenia pliku przez klienta. Jej domyślna wartość to 0744, co oznacza, że uniksowy użytkownik może odczytywać, zapisywać lub wykonywać pliki, natomiast członkowie jego grupy i inni mogą je co najwyżej odczytywać. Jeśli chcesz zmienić te zezwolenia dla plików niewykonywalnych, zalecamy wartość 0644, czyli `rw-r--r--`. Pamiętaj, że bity wykonywalności mogą być używane przez serwer do odwzorowywania dosowych atrybutów plików, co opisano wcześniej. Jeśli zmieniasz maskę tworzenia plików, te bity również muszą być częścią maski.

directory mask

Argumentem tej opcji jest ósemkowa liczba określająca zezwolenia, które mogą zostać ustawione podczas tworzenia katalogu przez klienta. Jej domyślna wartość to 0755, co oznacza, że inni użytkownicy uniksowi mogą co najwyżej odczytywać katalogi i przechodzić do nich, a tylko właściciel ma prawo do ich modyfikacji. Zalecamy maskę 0750, która uniemożliwia dostęp reszcie świata.

force create mode

Opcja ta określa bity zezwoleń, które zostaną ustawione przez Sambę podczas zmiany praw dostępu do pliku. Często używa się jej do ustawiania zezwoleń grupowych, o czym była mowa wcześniej. Można wykorzystać ją także do wstępnego ustawiania atrybutów dosowych: archiwalnego (0100), systemowego (0010) i ukrytego (0001). Opcja ta jest zawsze uwzględniana po opcjach `map archive`, `map system`, `map hidden` i `create mask`.

Wiele aplikacji Windows zmienia nazwę swoich plików danych na *plikdanych.bak* i tworzy nowe pliki, zmieniając tym samym ich właściciela i prawa dostępu, co uniemożliwia ich edytowanie przez członków tej samej uniksowej grupy. Ustawienie opcji `force create mode = 0660` sprawia, że nowy plik nadal może być edytowany przez członków grupy.

force directory mode

Opcja ta określa bity zezwoleń, które Samba ustawia podczas zmiany praw dostępu do istniejącego katalogu lub podczas tworzenia nowego. Często używa się jej do ustawiania zezwoleń grupowych, o czym była mowa wcześniej. Domyślna wartość tej opcji to 0000. Można używać jej dokładnie tak samo, jak opcji `force create mode`, aby w razie potrzeby dodawać zezwolenia grupowe lub inne. Opcja ta jest zawsze uwzględniana po opcjach `map archive`, `map system`, `map hidden` i `directory mask`.

force group

Opcja ta, czasem zapisywana w postaci `group`, określa statyczny identyfikator grupy, który po uwierzytelnieniu klienta będzie używany we wszystkich połączeniach z udziałem. Opcja ta przypisuje określoną grupę każdemu plikowi i katalogowi utworzonemu przez klienta SMB.

force user

Opcja ta określa statyczny identyfikator użytkownika, który po uwierzytelnieniu klienta będzie używany we wszystkich połączeniach z udziałem. Opcja ta przypisuje określonego użytkownika każdemu plikowi i katalogowi utworzonemu przez klienta SMB.

delete readonly

Ta opcja pozwala użytkownikowi usunąć katalog zawierający pliki przeznaczone tylko do odczytu. Domyślnie DOS i Windows nie pozwalają na taką operację. Zwy-

kle powinieneś pozostawić tę opcję wyłączoną, chyba że używasz programu, który wymaga takiej możliwości; wielu użytkowników Windows byłoby niepocieszonych, gdyby okazało się, że pomyłkowo usunęli plik, który był przeznaczony tylko do odczytu. Prawdę mówiąc, nawet uniksowe polecenie `rm` pyta, czy użytkownik rzeczywiście chce obejść zabezpieczenie i usunąć pliki przeznaczone tylko do odczytu. Samba powinna być równie ostrożna w tym względzie.

map archive

Dosowy bit archiwalny wskazuje, że plik został zmieniony od czasu ostatniej archiwizacji (to znaczy sporządzenia jego kopii zapasowej przez dosowy program archiwizujący). Ustawienie opcji Samby `map archive = yes` sprawia, że dosowy bit archiwalny jest odwzorowywany na uniksowy bit wykonywalności dla właściciela (0100). Lepiej pozostawić tę opcję włączoną, jeśli użytkownicy Windows sami sporządzają kopie zapasowe lub używają programów posługujących się bitem archiwalnym. W Uniksie nie istnieje pojęcie bitu archiwalnego. Programy do sporządzania kopii zapasowych zwykle przechowują plik z listą plików i datą ich skopiowania, więc mogą osiągnąć ten sam cel, porównując daty modyfikacji plików.

Ustawienie tej opcji na `yes` może od czasu do czasu wzbudzić zdziwienie użytkownika Uniksa, który zauważy, że plik danych jest oznaczony jako wykonywalny, ale rzadko prowadzi do problemów. Jeśli użytkownik spróbuje wykonać taki plik, zwykle zobaczy tylko ciąg komunikatów o błędzie, kiedy powłoka będzie próbowała zinterpretować kilka pierwszych linii jako polecenia. Możliwa jest też sytuacja odwrotna: wykonywalny program uniksowy będzie z perspektywy Windows stwarzał wrażenie niezarchiwizowanego, ale zdarza się to rzadko i jest raczej nieszkodliwe.

map system

Dosowy atrybut systemowy wskazuje, że plik jest wymagany przez system operacyjny i nie należy go usuwać ani przenosić bez podjęcia specjalnych kroków. Ustaw tę opcję tylko wtedy, gdy musisz przechowywać systemowe pliki Windows na uniksowym serwerze plików. Wykonywalne programy uniksowe sprawiają wrażenie nieusuwalnych, specjalnych plików Windows, gdy ogląda się je z klienta Windows. Może to być nieco niedogodne, jeśli zechcesz je przenieść lub usunąć. W większości przypadków jest to jednak nieszkodliwe.

map hidden

Dosowy atrybut ukrycia wskazuje, że plik nie powinien być zwykle widoczny w listingach katalogów. Unix nie dysponuje takim mechanizmem; o tym, co ma zostać wyświetlone, a co nie, decydują poszczególne programy (a zwłaszcza powłoka). Zwykle nie będziesz posługiwał się żadnymi plikami dosowymi, które powinny być ukryte, więc lepiej pozostaw tę opcję wyłączoną.

Ustawienie tej opcji na `yes` sprawia, że bit ukrycia jest odwzorowywany na uniksowy bit wykonywalności dla innych (0001). Może to prowadzić do dość nieoczekiwanych rezultatów: każdy program uniksowy wykonywalny dla „reszty świata" jest niewidoczny z perspektywy klienta Windows. Jednakże, jeśli opcja ta nie jest usta-

wiona, a użytkownik Windows spróbuje oznaczyć plik w udziale Samby jako ukryty, operacja się nie powiedzie – nie będzie gdzie zapisać atrybutu ukrycia.

Przekształcanie nazw i wielkość liter

W czasach DOS-a i Windows 3.1 każda nazwa pliku mogła składać się co najwyżej z ośmiu dużych liter, kropki i kolejnych trzech dużych liter. Posługiwanie się nazwami w tak zwanym *formacie 8.3* było mocno kłopotliwe. W Windows 95/98, Windows NT i Uniksie nazwy plików mogą liczyć dużo więcej znaków, a ich wielkość jest rozróżniana. Tabela 5.6 przedstawia obecne możliwości nazewnicze kilku popularnych systemów operacyjnych.

Tabela 5.6. Ograniczenia długości nazw plików w różnych systemach operacyjnych

System operacyjny	*Reguły nazywania plików*
DOS 6.22 i wcześniejsze wersje	Osiem znaków, po których następuje kropka i trzy znaki rozszerzenia (format 8.3); wielkość liter nie ma znaczenia
Windows 3.1 for Workgroups	Osiem znaków, po których następuje kropka i trzy znaki rozszerzenia (format 8.3); wielkość liter nie ma znaczenia
Windows 95/98	127 znaków; wielkość liter jest rozróżniana
Windows NT	127 znaków; wielkość liter jest rozróżniana
Unix	255 znaków; wielkość liter jest rozróżniana

Samba musi zachować zgodność wstecz z klientami sieciowymi, które przechowują pliki tylko w formacie 8.3, jak Windows for Workgroups. Jeśli użytkownik utworzy w udziale plik o nazwie *antygonadotropistyczność.txt*, klient Windows for Workgroups nie będzie umiał go odróżnić od znajdującego się w tym samym katalogu pliku o nazwie *antygona.txt*. Podobnie jak Windows 95/98 i Windows NT, Samba musi stosować specjalną metodę tłumaczenia długich nazw plików na nazwy 8.3, w taki sposób, aby podobne nazwy nie powodowały kolizji. Metodę tę nazywamy *przekształcaniem nazw*; Samba radzi sobie z tym podobnie, choć nie identycznie, jak Windows 95 i jego następcy.

Przekształcanie nazw przez Sambę

Oto jak Samba przekształca długą nazwę pliku na nazwę w formacie 8.3:

- Jeśli pierwotna nazwa pliku nie zaczyna się od kropki, najwyżej pięć pierwszych znaków alfanumerycznych występujących przed ostatnią kropką (jeśli kropka występuje w nazwie) jest przekształcanych na duże litery. Stanowią one pierwsze pięć znaków przekształconej nazwy 8.3.
- Jeśli pierwotna nazwa pliku zaczyna się od kropki, kropka jest usuwana, a najwyżej pięć pierwszych znaków alfanumerycznych występujących przed ostatnią kropką (jeśli kropka występuje w nazwie) jest przekształcanych na duże litery. Stanowią one pierwsze pięć znaków przekształconej nazwy 8.3.
- Za tymi znakami umieszczany jest specjalny znak przekształcenia: domyślnie jest to tylda (~), choć Samba pozwala na użycie innego znaku.

- Podstawowa część długiej nazwy pliku przed ostatnią kropką jest zamieniana za pomocą funkcji mieszającej na dwuznakowy kod; w razie potrzeby wykorzystywane są części nazwy znajdujące się za ostatnią kropką. Ten dwuznakowy kod jest dołączany do nazwy 8.3 za znakiem przekształcenia.
- Pierwsze trzy znaki pierwotnej nazwy pliku znajdujące się za ostatnią kropką (jeśli kropka występuje w nazwie) są zamieniane na duże litery i dołączane do przekształconej nazwy jako rozszerzenie. Jeśli pierwotna nazwa pliku zaczynała się od kropki, zamiast rozszerzenia zostaną użyte trzy znaki podkreślenia (_ _ _).

Oto kilka przykładów:

```
wirtuozeria.dat                    WIRTU~NZ.DAT
.htaccess                          HTACC~46.___
witaj.java                         WITAJ~FG.JAV
grupa.konf.txt                     GRUPA~3Q.TXT
antygonawn.txt                     ANTYG~3O.TXT
antygonadotropiczno□□.txt          ANTYG~N6.TXT
```

Dzięki tym regułom Windows for Workgroups będzie rozróżniać pliki na potrzeby użytkowników, którzy mają wątpliwe szczęście oglądać sieć z perspektywy tego systemu operacyjnego. Zauważ, że określona długa nazwa pliku powinna zostać przekształcona przez Sambę zawsze na taką samą nazwę 8.3; w Windows nie zawsze tak się dzieje. Kolizje nadal mogą występować, ale znacznie zmniejsza się ich prawdopodobieństwo.

Opcje konfiguracji przekształcania przydają się właściwie tylko do współpracy z najstarszymi klientami. Jeśli zdecydujesz się je wykorzystać, powinieneś zrobić to bez zakłócania pracy innych klientów, za pomocą następującej dyrektywy `include` w pliku konfiguracyjnym:

```
[global]
    include = /usr/local/samba/lib/smb.conf.%m
```

Spowoduje to dołączenie pliku konfiguracyjnego *smb.conf.WfWg*, kiedy połączenie z serwerem nawiąże klient Windows for Workgroups. Teraz możesz utworzyć plik */usr/local/samba/lib/smb.conf.WfWg* na przykład z takimi opcjami:

```
[global]
    case sensitive = no
    default case = upper
    preserve case = no
    short preserve case = no
    mangle case = yes
    mangled names = yes
```

Jeśli nie używasz Windows for Workgroups, zapewne nie będziesz musiał zmieniać domyślnych wartości tych opcji.

Reprezentowanie i ustalanie nazw plików

Powinniśmy jeszcze nadmienić, że systemy operacyjne mogą inaczej *reprezentować* nazwy plików, a inaczej je *ustalać*. Jeśli zdarzyło ci się korzystać z Windows 95/98/NT, prawdopodobnie trafiłeś na plik o nazwie *README.TXT*. Plik może być reprezentowany przez system operacyjny wyłącznie za pomocą dużych liter. Jeśli

jednak uruchomisz tryb MS-DOS i wpiszesz polecenie `edit readme.txt`, wówczas do edytora zostanie załadowany plik o nazwie złożonej z dużych liter, choć ty wpisałeś ją małymi!

Dzieje się tak dlatego, że rodzina systemów operacyjnych Windows 95/98/NT ustala nazwy plików bez uwzględniania wielkości liter, choć reprezentuje je z rozróżnieniem małych i dużych liter. Systemy uniksowe zawsze ustalają nazwy plików z rozróżnieniem wielkości liter; jeśli spróbujesz otworzyć plik *README.TXT* za pomocą polecenia `vi readme.txt`, najprawdopodobniej będziesz edytował pusty bufor nowego pliku.

Oto jak Samba obsługuje wielkość liter: jeśli opcja `preserve case` jest ustawiona na `yes`, Samba będzie zawsze posługiwać się literami takiej wielkości, której system operacyjny używa do reprezentowania (nie ustalania) nazw plików. Jeśli jest ustawiona na `no`, Samba będzie posługiwać się literami wielkości określonej opcją `default case`. Tak samo działa opcja `short preserve case`. Jeśli jest ustawiona na `yes`, Samba będzie używać takiej wielkości liter do reprezentowania nazw 8.3, jaka jest domyślna dla systemu operacyjnego; w przeciwnym wypadku użyje wielkości określonej opcją `default case`. Wreszcie, Samba będzie zawsze ustalać nazwy plików w swoich udziałach według wartości opcji `case sensitive`.

Opcje przekształcania

Samba pozwala na dostosowanie przekształcania nazw do potrzeb użytkownika, w tym na kontrolowanie rozróżniania wielkości liter, na określenie znaku używanego do tworzenia przekształconych nazw oraz na ręczne odwzorowywanie nazw plików między dwoma formatami. Opcje te są wymienione w tabeli 5.7.

Tabela 5.7. Opcje przekształcania nazw

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`case sensitive (casesignames)`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba będzie rozróżniać wielkość liter w nazwach plików (w przeciwieństwie do Windows)	`no`	Udział
`default case`	(`upper` lub `lower`)	Wielkość liter przyjmowana za domyślną (używana tylko wtedy, gdy opcja `preserve case` ma wartość `no`)	`lower`	Udział
`preserve case`	Wartość logiczna	Jeśli jest ustawiona na `yes`, zachowuje wielkość liter w nazwach przekazanych przez klienta (to znaczy nie konwertuje ich na wielkość domyślną, określoną za pomocą opcji `default case`)	`yes`	Udział
`short preserve case`	Wartość logiczna	Jeśli jest ustawiona na `yes`, zachowuje wielkość liter w nazwach 8.3 przekazanych przez klienta	`yes`	Udział

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`mangle case`	Wartość logiczna	Przekształca nazwę, jeśli w jej skład wchodzą litery różnej wielkości	`no`	Udział
`mangled names`	Wartość logiczna	Przekształca długie nazwy na dosowy format 8.3	`yes`	Udział
`mangling char`	Łańcuch (pojedynczy znak)	Określa znak przekształcania	~	Udział
`mangled stack`	Wartość liczbowa	Liczba przekształconych nazw przechowywanych na lokalnym stosie przekształcania	`50`	Globalny
`mangled map`	Łańcuch (lista wzorców)	Pozwala na odwzorowywanie nazw plików między dwoma formatami	Brak	Udział

case sensitive

Ta opcja (o dość mało mówiącym synonimie `casesignames`) określa, czy Samba powinna zachowywać wielkość liter podczas ustalania nazw w konkretnym udziale. Jej domyślna wartość to `no`, co odpowiada ustalaniu nazw w Windows. Jeśli klienty używają systemu operacyjnego, który rozróżnia wielkość liter w nazwach plików, możesz ustawić tę opcję na `yes`, jak w przykładzie poniżej:

```
[ksiegowosc]
    case sensitive = yes
```

W przeciwnym wypadku zalecamy pozostawienie domyślnej wartości tej opcji.

default case

Opcji `default case` używa się w połączeniu z opcją `preserve case`. Określa ona domyślną wielkość liter (duże, `upper`, lub małe, `lower`), której Samba użyje podczas tworzenia pliku w imieniu klienta. Domyślna wielkość liter to `lower`, co oznacza, że w nazwach nowo tworzonych plików będą występować litery o takiej wielkości, jaką podał klient. Jeśli zachodzi taka potrzeba, możesz zmienić tę globalną opcję, pisząc:

```
[global]
    default case = upper
```

Jeśli podasz taką wartość, nazwy nowo tworzonych plików będą zawsze przekształcane na duże litery. Zalecamy pozostawienie wartości domyślnej, o ile nie masz do czynienia z klientami Windows for Workgroups lub innymi używającymi formatu 8.3 – wtedy powinieneś nadać tej opcji wartość `upper`.

preserve case

Opcja ta definiuje, czy w nazwach plików tworzonych przez Sambę na zlecenie klienta będą występowały litery o wielkości określonej przez system operacyjny klienta, czy też o wielkości określonej za pomocą omówionej wyżej opcji `default`

case. Domyślna wartość tej opcji to yes, co oznacza, że wielkość liter ustala system operacyjny klienta. Jeśli ustawisz ją na no, wielkość liter zostanie ustalona na podstawie opcji default case.

Warto wspomnieć, że opcja ta nie dotyczy żądań dostępu do plików w formacie 8.3 wysyłanych przez klientów – patrz opcja short preserve case poniżej. Możesz ustawić tę opcję na yes, jeśli aplikacje tworzące pliki w serwerze Samby rozróżniają wielkość liter. Jeśli chcesz, żeby Samba naśladowała działanie systemu plików Windows NT, pozostaw domyślną wartość tej opcji, yes.

short preserve case

Opcja ta określa, czy w nazwach plików formatu 8.3 tworzonych przez Sambę na zlecenie klienta będą występowały litery o wielkości określonej przez system operacyjny klienta, czy też o wielkości określonej za pomocą opcji default case. Domyślna wartość tej opcji to yes, co oznacza, że wielkość liter ustala system operacyjny klienta. Możesz zezwolić Sambie na wybieranie wielkości liter według ustawienia opcji default case, pisząc co następuje:

```
[global]
    short preserve case = no
```

Jeśli chcesz, żeby Samba naśladowała działanie systemu plików Windows NT, pozostaw domyślną wartość tej opcji, yes.

mangled names

Ta opcja określa, czy Samba będzie przekształcać nazwy plików w swoich udziałach na potrzeby klientów posługujących się formatem 8.3. Jeśli jest ustawiona na no, Samba nie będzie przekształcać nazw, co sprawi, że dla systemów operacyjnych pracujących z nazwami w formacie 8.3 będą one niewidoczne lub będą się wydawać obcięte. Domyślna wartość tej opcji to yes. Możesz zmienić jej wartość dla pojedynczego udziału:

```
[dane]
    mangled names = no
```

mangle case

Opcja ta informuje Sambę, czy należy przekształcać nazwy plików nie składające się w całości z liter o wielkości określonej opcją konfiguracyjną default case. Domyślna wartość tej opcji to no. Jeśli ustawisz ją na yes, powinieneś upewnić się, czy wszystkie klienty będą zdolne do posługiwania się przekształconymi nazwami plików. Możesz zmienić jej wartość dla pojedynczego udziału w następujący sposób:

```
[dane]
    mangled case = yes
```

Odradzamy zmienianie wartości tej opcji, jeśli nie masz po temu ważnego powodu.

mangling char

Ta opcja określa znak używany podczas przekształcania nazw plików na format 8.3. Domyślnym znakiem przekształcania jest tylda (~). Zamiast niej możesz użyć dowolnego innego znaku, na przykład:

```
[dane]
    mangling char = #
```

mangled stack

Samba używa lokalnego stosu niedawno przekształconych nazw 8.3. Stos ten można wykorzystać do przywracania pierwotnej postaci przekształconych nazw plików; często wymagają tego aplikacje, które tworzą i zapisują plik, zamykają go, a później muszą go zmodyfikować. Stos ten domyślnie przechowuje 50 par: długa nazwa pliku – przekształcona nazwa pliku. Jeśli chcesz odciążyć procesor od zadań związanych z przekształcaniem nazw, możesz zwiększyć rozmiar stosu kosztem większego zużycia pamięci i nieco wolniejszego dostępu do plików.

```
[global]
    mangled stack = 100
```

mangled map

Jeśli domyślny algorytm przekształcania nazw jest niewystarczający, możesz sprecyzować jego działanie za pomocą opcji `mangled map`. Opcja ta pozwala na określenie własnych wzorców przekształcania nazw, stosowanych przed przekształceniem nazwy przez Sambę. Na przykład:

```
[dane]
    mangled map = (*.database *.db) (*.class *.cls)
```

Samba będzie szukać we wszystkich nazwach plików znaków odpowiadających pierwszemu wzorcowi określonemu w nawiasie i przekształcać je na drugi wzorzec w celu wyświetlenia ich w kliencie 8.3. Jest to przydatne wtedy, gdy domyślny algorytm przekształca nazwy nieprawidłowo lub na format, którego klient nie może rozpoznać. Poszczególne wzorce oddziela się znakami odstępu.

Blokady i blokady oportunistyczne

Jednoczesne zapisy w jednym pliku są w każdym systemie operacyjnym rzeczą niepożądaną. Aby temu zapobiec, większość systemów używa *blokad*, które gwarantują, że w danej chwili w pliku może pisać tylko jeden proces. Tradycyjne systemy operacyjne blokowały całe pliki, ale te nowocześniejsze potrafią zablokować także zakres bajtów w pliku. Jeśli inny proces spróbuje zapisać coś w zablokowanym już pliku (albo w jego części), system operacyjny zgłosi błąd i proces będzie musiał poczekać na zwolnienie blokady.

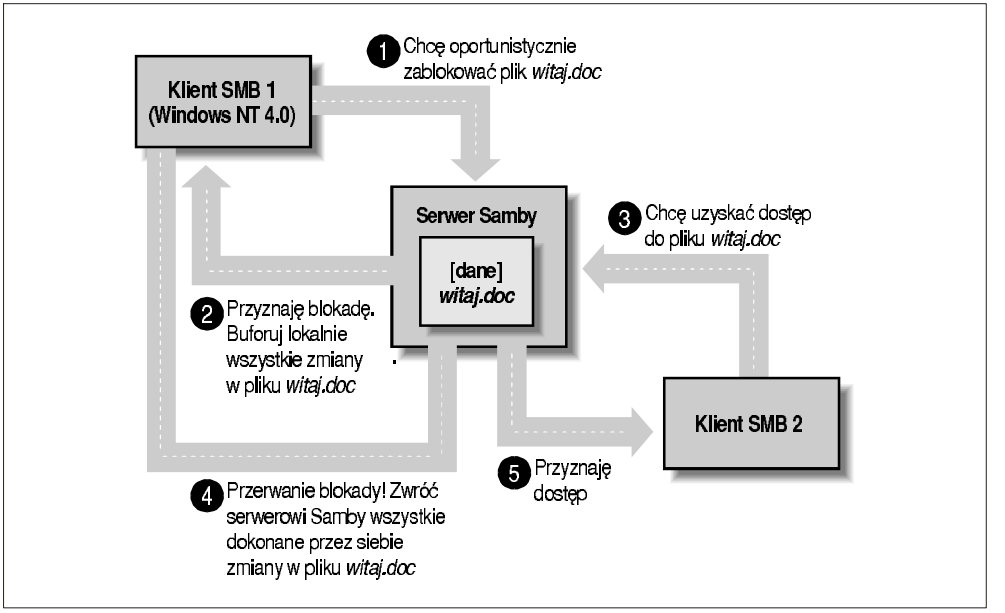
Samba obsługuje standardowe żądania blokady DOS-a i systemu plików NT (w trybie odmowy, ang. *deny-mode*), które pozwalają na pisanie w całym pliku tylko jednemu procesowi w danej chwili. Samba nakłada także blokady zakresów bajtów. Ponadto Samba obsługuje nowy mechanizm blokowania, w terminologii Windows NT nazywany *blokadą oportunistyczną*.

Blokady oportunistyczne

Dzięki blokadom oportunistycznym klient może poinformować serwer Samby, że nie tylko będzie jedynym uprawnionym do pisania w pliku, ale że będzie buforował

wszystkie zmiany lokalnie (a nie w serwerze Samby) w celu przyspieszenia dostępu do pliku. Kiedy Samba wie, że plik został oportunistycznie zablokowany przez klienta, zaznacza swoją wersję tego pliku jako obłożoną blokadą i czeka, aż klient zakończy operacje na pliku i odeśle jego ostateczną wersję w celu wzajemnej synchronizacji.

Jeśli inny klient zażąda dostępu do tego pliku, zanim pierwszy klient zakończy pracę, Samba może wysłać żądanie *przerwania blokady* do pierwszego klienta. Jest to informacja dla klienta, że powinien zaprzestać lokalnego buforowania i zwrócić informacje o bieżącym stanie pliku, aby nowy klient mógł użyć go wedle swego uznania. Blokada oportunistyczna nie jest jednak zamiennikiem standardowej blokady w trybie odmowy. Często zdarza się, że serwer przyznaje klientowi prawo do przerwania blokady oportunistycznej, a ten odkrywa, że pierwotny proces nałożył na plik także blokadę w trybie odmowy. Proces blokowania oportunistycznego przedstawiono na rysunku 5.8.



Rysunek 5.8. Zakładanie blokady oportunistycznej

Jeśli chodzi o opcje blokad, zalecamy poprzestanie na wartościach domyślnych Samby: standardowych blokadach DOS/Windows w trybie odmowy, które gwarantują zachowanie zgodności, i blokadach oportunistycznych, które zwiększają wydajność dzięki lokalnemu buforowaniu. Jeśli twój system operacyjny potrafi korzystać z blokad oportunistycznych, zapewne uda ci się znacznie poprawić wydajność. Jeśli nie masz ważnego powodu do zmieniania którejś z opcji, powinieneś pozostawić je takimi, jakie są.

Unix i blokowanie

Systemy Windows potrafią ze sobą współpracować tak, aby uniknąć nadpisania zmian dokonanych w pliku przez innego klienta. Jeśli jednak do pliku przechowywanego w serwerze Samby uzyska dostęp proces uniksowy, nie będzie on miał pojęcia o oportunistycznym blokowaniu używanym w Windows i bez żadnych ceregieli naruszy blokadę. Niektóre systemy uniksowe przystosowano do współpracy z oportunistycznymi blokadami Windows utrzymywanymi przez Sambę. Obecnie potrafi je obsługiwać system SGI Irix 6.5.2f i jego nowsze wersje, a niebawem zdolność tę będą miały także Linux i FreeBSD.

Jeśli twój system rozpoznaje blokady oportunistyczne, ustaw opcję `kernel oplocks = yes` w pliku konfiguracyjnym Samby. Powinno to wyeliminować konflikty między procesami Uniksa a użytkownikami Windows.

Jeśli twój system nie obsługuje blokad oportunistycznych na poziomie jądra, może się zdarzyć, że ktoś uruchomi uniksowy proces czytający lub piszący w pliku, z którego korzysta również Windows, co spowoduje uszkodzenie danych. Samba udostępnia jednak prowizoryczny mechanizm ochronny, z którego można skorzystać w razie nieobecności oportunistycznych blokad: opcję `veto oplock files`. Jeśli możesz przewidzieć, z których plików będą korzystać zarówno użytkownicy Windows, jak i Uniksa, możesz podać ich nazwy w tej opcji. Uniemożliwi to zakładanie oportunistycznych blokad na pliki o takich nazwach, a więc i lokalne buforowanie zmian, dzięki czemu programy działające w Windows i Uniksie będą mogły skorzystać z blokowania systemowego lub czasów uaktualnienia, aby rozstrzygnąć rywalizację o ten sam plik. Oto przykład:

```
veto oplock files = /*.dbm/
```

Opcja ta pozwoli na bezpieczne edytowanie plików o rozszerzeniu *.dbm* zarówno przez użytkowników Windows, jak i Uniksa. Zauważ, że składnia jest podobna jak w opcji `veto files`.

Opcje blokad zwykłych i oportunistycznych są podane w tabeli 5.8.

Tabela 5.8. Opcje konfiguracji blokad zwykłych i oportunistycznych

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`share modes`	Wartość logiczna	Jeśli jest ustawiona na `yes`, włącza obsługę dosowych blokad całych plików	`yes`	Udział
`locking`	Wartość logiczna	Jeśli jest ustawiona na `yes`, włącza blokady zakresów bajtów	`yes`	Udział
`strict locking`	Wartość logiczna	Jeśli jest ustawiona na `yes`, a na plik nałożona jest blokada zakresu bajtów, odmawia dostępu do całego pliku	`no`	Udział
`oplocks`	Wartość logiczna	Jeśli jest ustawiona na `yes`, włącza lokalne buforowanie plików w tym udziale	`yes`	Udział

Dokończenie tabeli na str. 142

Dokończenie tabeli ze str. 141

Tabela 5.8. Opcje konfiguracji blokad zwykłych i oportunistycznych

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`kernel oplocks`	Wartość logiczna	Jeśli jest ustawiona na `yes`, informuje Sambę, że jądro obsługuje blokady oportunistyczne	`yes`	Globalny
`fake oplocks`	Wartość logiczna	Jeśli jest ustawiona na `yes`, informuje klienta, że przyznano blokadę, choć w rzeczywistości plik nie jest blokowany	`no`	Udział
`blocking locks`	Wartość logiczna	Pozwala klientowi żądającemu blokady zaczekać na jej przyznanie	`yes`	Udział
`veto oplock files`	Łańcuch (lista nazw plików)	Nie zezwala na oportunistyczne blokowanie wskazanych plików	Brak	Udział
`lock directory`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa położenie katalogu, w którym przechowywane są różne pliki Samby, w tym pliki blokady	Taka jak w pliku *makefile* Samby	Globalny

share modes

Najprostsze blokady udostępniane przez Sambę to blokady w trybie odmowy, tak zwane *tryby współdzielenia*, z których korzystają programy, takie jak edytor tekstów, aby uniknąć przypadkowego nadpisania plików. Różne typy blokady w trybie odmowy są zebrane w tabeli 5.9.

Tabela 5.9. Blokady SMB w trybie odmowy

Blokada	*Opis*
`DENY_NONE`	Nie odrzuca innych żądań dostępu do pliku
`DENY_ALL`	Odrzuca wszystkie żądania otwarcia pliku
`DENY_READ`	Odrzuca wszystkie żądania otwarcia pliku w trybie tylko do odczytu
`DENY_WRITE`	Odrzuca wszystkie żądania otwarcia pliku w trybie tylko do zapisu
`DENY_DOS`	Jeśli plik jest otwarty do odczytu, inni mogą czytać plik, ale nie mogą w nim pisać. Jeśli plik jest otwarty do zapisu, inni w ogóle nie mogą otworzyć pliku
`DENY_FCB`	Przestarzała

Opcja `share modes`, która wymusza użycie tych blokad, jest domyślnie włączona. Aby ją wyłączyć, użyj następującej dyrektywy:

```
[ksiegowosc]
    share modes = no
```

Stanowczo odradzamy wyłączanie domyślnego mechanizmu blokowania, jeśli nie ma po temu ważnej przyczyny. Większość aplikacji Windows i DOS-a wymaga mechanizmów blokowania do poprawnej pracy i poskarży się, jeśli będą one niedostępne.

locking

Opcja `locking` informuje Sambę, czy należy włączyć, czy też wyłączyć blokowanie zakresów bajtów na zlecenie klienta. Samba implementuje blokady zakresów bajtów po stronie serwera za pomocą zwykłych uniksowych blokad doradczych, co zapobiega nadpisaniu zablokowanego zakresu bajtów przez poprawnie działające procesy uniksowe.

Opcję tę można określić dla każdego udziału z osobna, jak w poniższym przykładzie:

```
[ksiegowosc]
    locking = yes
```

Jeśli opcja `locking` zostanie ustawiona na `yes`, klient proszący o przyznanie blokady zostanie wstrzymany aż do czasu, kiedy aktualny posiadacz zwolni ją (lub załamie się). Jeśli zaś opcja ta jest ustawiona na `no`, Samba nie będzie przechowywać blokad zakresów bajtów, choć żądania zablokowania i odblokowania pliku będą pozornie spełniane. Opcja ta jest domyślnie ustawiona na `yes`, możesz jednak ustawić ją na `no`, jeśli pliki znajdują się na nośniku przeznaczonym tylko do odczytu.

strict locking

Opcja ta sprawia, że przy każdym dostępie do pewnego zakresu bajtów Samba sprawdza, czy nie jest na niego nałożona blokada. Zwykle nie jest to potrzebne, jeśli klient czyni zadość wszystkim stosowanym mechanizmom blokowania. Opcja ta ma domyślną wartość `no`, możesz jednak zmienić ją dla każdego udziału z osobna, jak w przykładzie poniżej:

```
[ksiegowosc]
    strict locking = yes
```

Jeśli opcja ta jest ustawiona na `yes`, wówczas na każdy plik z blokadą zakresu bajtów nakładana jest blokada obowiązkowa.

blocking locks

Samba obsługuje także *blokady wstrzymujące*, które są pewną odmianą blokad zakresu bajtów. Jeśli żądany zakres bajtów jest niedostępny, klient może określić, ile czasu jest gotowy czekać. Serwer buforuje żądanie przyznania blokady i okresowo sprawdza, czy plik stał się dostępny. Jeśli tak jest, informuje o tym klienta, a jeśli upłynie limit czasu, Samba informuje klienta, że żądanie nie zostało spełnione. Strategia ta zapobiega ciągłemu odpytywaniu serwera przez klienta w celu sprawdzenia, czy blokada może zostać przyznana.

Możesz wyłączyć tę opcję dla każdego udziału z osobna, jak w przykładzie poniżej:

```
[ksiegowosc]
    blocking locks = no
```

Jeśli opcja ta jest ustawiona na `yes`, na plik nakładane są blokady wstrzymujące. Jeśli jest ustawiona na `no`, Samba zachowuje się tak, jakby na plik były nałożone zwykłe blokady. Domyślna wartość tej opcji to `yes`.

oplocks

Opcja ta włącza lub wyłącza obsługę blokad oportunistycznych. Domyślnie jest włączona. Możesz ją jednak wyłączyć za pomocą dyrektywy:

```
[dane]
    oplocks = no
```

Jeśli twoje środowisko sieciowe jest bardzo niestabilne albo zarządzasz wieloma klientami, które nie obsługują blokad oportunistycznych, warto wyłączyć tę opcję. Powinieneś wyłączyć blokady oportunistyczne także wtedy, gdy z tych samych plików korzystają zarówno aplikacje uniksowe (na przykład edytor *vi*), jak i klienty SMB (chyba że twój system operacyjny obsługuje blokady oportunistyczne na poziomie jądra, o czym była mowa wcześniej).

fake oplocks

Zanim w Sambie pojawiła się obsługa blokad oportunistycznych, dzięki opcji `fake oplocks` demony Samby mogły udawać, że przyznają takie blokady. Jeśli opcja ta była włączona, wszystkie klienty były informowane, że plik może zostać zablokowany oportunistycznie i nigdy nie ostrzegano ich przed jednoczesnym dostępem. Opcja ta jest obecnie przestarzała, ponieważ Samba obsługuje prawdziwe blokady oportunistyczne.

kernel oplocks

Jeśli nie związana z Sambą aplikacja uniksowa spróbuje uaktualnić plik zablokowany oportunistycznie przez klienta Windows, prawdopodobnie zdoła to zrobić (w zależności od systemu operacyjnego), a klient i Samba nie będą tego świadome. Jeśli jednak używana odmiana Uniksa na to pozwala, Samba może ostrzegać system operacyjny o zablokowanych oportunistycznie plikach, co spowoduje wstrzymanie procesu uniksowego, powiadomienie klienta za pośrednictwem Samby o konieczności zapisania jego wersji i – dopiero w tym momencie – nastąpi zezwolenie na otwarcie pliku. Zasadniczo oznacza to, że jądro systemu operacyjnego w serwerze Samby potrafi posługiwać się oportunistycznymi blokadami nie gorzej od Samby.

Możesz włączyć tę funkcję za pomocą opcji `kernel oplocks`, jak w poniższym przykładzie:

```
[global]
    kernel oplocks = yes
```

Samba może automatycznie wykryć obsługę blokad oportunistycznych w jądrze i skorzystać z nich, jeśli są dostępne. W czasie pisania książki mechanizm ten funkcjonował tylko w systemie SGI Irix 6.5.2f i jego nowszych wersjach. Jednakże w najbliższej przyszłości obsługa blokad oportunistycznych ma pojawić się w systemach Linux i FreeBSD. System bez blokad oportunistycznych w jądrze zezwoli procesowi uniksowemu na uaktualnienie pliku, co programy klienckie zauważą dopiero później – jeśli w ogóle.

veto oplock files

Dzięki opcji `veto oplock files` możesz podać listę plików, na które nigdy nie będą nakładane blokady oportunistyczne. Opcję tę można ustalić globalnie lub dla każdego udziału z osobna. Na przykład:

```
veto oplock files = /*.bat/*.htm/
```

Wartością tej opcji jest ciąg wzorców. Każdy wzorzec musi zaczynać się, kończyć lub być oddzielony od innego znakiem ukośnika (/), nawet wtedy, gdy używa się tylko jednego wzorca. Można używać gwiazdek, reprezentujących zero lub więcej znaków, oraz znaków zapytania, reprezentujących dokładnie jeden znak.

Zalecamy wyłączenie oportunistycznego blokowania wszystkich plików, które mogą być uaktualniane z Uniksa lub są przewidziane do współdzielenia przez kilka różnych procesów.

lock directory

Opcja ta (czasem używana w postaci `lock dir`) określa położenie katalogu, w którym Samba będzie przechowywać pliki blokad SMB (przyznanych w trybie odmowy dostępu). Samba przechowuje w nim także inne pliki, na przykład listy przeglądania i plik pamięci dzielonej. Jeśli włączona jest obsługa WINS, w katalogu tym zapisana zostanie również baza danych WINS. Domyślne położenie tego katalogu określa się w pliku *makefile* Samby, najczęściej jest to */usr/local/samba/var/locks*. Możesz zmienić położenie tego katalogu w następujący sposób:

```
[global]
    lock directory = /usr/local/samba/locks
```

Zwykle nie ma potrzeby zmieniania tej opcji, chyba że chcesz przenieść pliki blokad w bardziej standardowe położenie, na przykład */var/spool/locks*.

6

Użytkownicy, bezpieczeństwo i domeny

W tym rozdziale omówimy konfigurowanie kont użytkowników serwera Samby. Początkowo zagadnienie to wydaje się raczej proste, ale niebawem przekonasz się, że może pojawić się kilka pomniejszych problemów. Administratorzy Samby często miewają kłopoty z uwierzytelnianiem użytkowników – znakomita większość pytań pojawiających się na listach wysyłkowych Samby dotyczy problemów z hasłami i bezpieczeństwem. Jeśli będziesz wiedział, dlaczego niektóre mechanizmy bezpieczeństwa działają w jednych systemach, a w innych nie, w przyszłości oszczędzisz mnóstwo czasu, który musiałbyś poświęcić na testowanie i diagnozowanie kont użytkowników Samby.

Użytkownicy i grupy

Już na wstępie winniśmy cię ostrzec, że jeśli łączysz się z Sambą z Windows 98 lub NT 4.0 Workstation SP3, musisz skonfigurować swój serwer do obsługi zaszyfrowanych haseł, zanim nawiążesz połączenie. W przeciwnym wypadku klienty odmówią połączenia się z serwerem Samby. Dzieje się tak dlatego, że klienty te posługują się zaszyfrowanymi hasłami, a Samba musi być odpowiednio skonfigurowana, aby je szyfrować i deszyfrować. W tym rozdziale pokażemy ci, jak to zrobić, zakładając, że nie uporałeś się z tym problemem już w rozdziale 2, *Instalowanie Samby w Uniksie*.

Zacznijmy od jednego użytkownika. Najłatwiejszym sposobem skonfigurowania użytkownika jest założenie uniksowego konta (i katalogu macierzystego) w serwerze i powiadomienie Samby o istnieniu tego użytkownika. To ostatnie można zrobić, tworząc w pliku konfiguracyjnym Samby udział dyskowy odpowiadający katalogowi macierzystemu użytkownika i ograniczając dostęp do tego udziału za pomocą opcji `valid users`. Na przykład:

```
[dawid]
    path = /home/dawid
    comment = Katalog macierzysty Dawida
    writeable = yes
    valid users = dawid
```

Opcja `valid users` określa użytkowników, którzy będą mogli korzystać z udziału. W tym przypadku dostęp do udziału będzie miał tylko użytkownik `dawid`.

W poprzednich rozdziałach dowiedziałeś się, że dzięki opcji `guest ok` możesz zezwolić na dostęp do udziału dowolnemu użytkownikowi. Ponieważ nie chcemy pozwolić na gościnny dostęp, nie używamy tej opcji. W razie potrzeby moglibyśmy przyznać dostęp do udziału zarówno uwierzytelnionym, jak i nieuwierzytelnionym użytkownikom. Różnica między nimi polega zwykle na różnych prawach dostępu do plików udziału.

Pamiętaj, że zamiast nazwy macierzystego katalogu użytkownika, możesz użyć zmiennej `%H`. Oprócz tego, w opcjach możesz skorzystać ze zmiennej `%u`, odpowiadającej uniksowej nazwie użytkownika, oraz zmiennej `%U`, odpowiadającej nazwie klienta. Na przykład:

```
[dawid]
    comment = Katalog macierzysty %U
    writeable = yes
    valid users = dawid
    path = %H
```

Oba te przykłady zadziałają, jeśli tylko uniksowy użytkownik, którego Samba używa do reprezentowania klienta, ma prawo do odczytu i zapisu w katalogu określonym opcją `path`. Innymi słowy, zanim klient uzyska prawo do zapisu i odczytu udziału, musi zostać zweryfikowany przez mechanizmy bezpieczeństwa Samby (jak na przykład zaszyfrowane hasła, opcja `valid users` i tak dalej), a zarazem mieć odpowiednie prawa dostępu do plików i katalogów wynikające z uniksowych uprawnień użytkownika.

W przypadku jednego użytkownika korzystającego ze swojego katalogu macierzystego, prawa dostępu są ustalane podczas zakładania konta użytkownika przez system operacyjny. Jeśli jednak tworzysz współdzielony katalog, z którego będzie korzystać grupa użytkowników, będziesz musiał wykonać kilka dodatkowych czynności. Spróbujmy utworzyć w pliku *smb.conf* udział dla działu księgowości:

```
[ksiegowosc]
    comment = Katalog dzia□u ksi□gowo□ci
    writeable = yes
    valid users = @ksieg
    path = /home/samba/ksiegowosc
    create mode = 0660
    directory mode = 0770
```

Zapewne zauważyłeś, że podaliśmy `@ksieg` jako użytkownika udziału, zamiast jednej lub wielu nazw użytkowników. Jest to skrótowy sposób na określenie, że uprawnionych użytkowników reprezentuje uniksowa grupa `ksieg`. Użytkowników tych trzeba będzie dopisać do definicji grupy `ksieg` w systemowym pliku grup (*/etc/group* lub jego odpowiedniku), aby byli rozpoznawani jako członkowie grupy. Wtedy Samba będzie uznawać ich za uprawnionych do korzystania z udziału.

Ponadto będziesz musiał utworzyć współdzielony katalog dostępny dla członków grupy, na który będzie wskazywać opcja konfiguracyjna `path`. Oto polecenia uniksowe, które tworzą współdzielony katalog dla działu księgowości (zakładając, że katalog */home/samba* już istnieje):

```
# mkdir /home/samba/ksiegowosc
# chgrp ksieg /home/samba/ksiegowosc
# chmod 770 /home/samba/ksiegowosc
```

W omawianym przykładzie pojawiają się dwa inne wpisy, o których była już mowa w poprzednich rozdziałach: `create mode` i `directory mode`. Opcje te ustawiają najwyższe dozwolone prawa dostępu do nowo tworzonych plików i katalogów. W tym przypadku uniemożliwiliśmy „reszcie świata" dostęp do zawartości udziału (co podkreśla jeszcze wydane wcześniej polecenie *chmod*).

Udział [homes]

Wróćmy na chwilę do udziałów użytkowników. Jeśli mamy kilku użytkowników, którym chcemy założyć katalogi macierzyste, prawdopodobnie najłatwiej będzie nam użyć specjalnego udziału `[homes]`, o którym wspomnieliśmy w rozdziale 4, *Udziały dyskowe*. Wystarczy wówczas napisać:

```
[homes]
    browsable = no
    writable = yes
```

Udział `[homes]` jest specjalną sekcją pliku konfiguracyjnego Samby. Jeśli użytkownik spróbuje połączyć się ze zwykłym udziałem, który nie figuruje w pliku *smb.conf* (na przykład podając jego położenie UNC w Eksploratorze Windows), Samba spróbuje odszukać udział `[homes]`. Jeśli taki udział istnieje, Samba przyjmuje, że nazwa żądanego udziału jest nazwą użytkownika i próbuje wyszukać jego wpis w pliku haseł (*/etc/passwd* lub odpowiedniku) serwera Samby. Jeśli Samba odnajdzie taki wpis, wówczas zakłada, że klient jest użytkownikiem uniksowym, który próbuje połączyć się ze swoim katalogiem macierzystym.

Załóżmy, że użytkownik `zofia` próbuje połączyć się z udziałem o nazwie `[zofia]` w serwerze Samby. W pliku konfiguracyjnym nie ma udziału o takiej nazwie, ale istnieje udział `[homes]`, a w pliku haseł znajduje się wpis dla `zofii`, więc Samba podejmuje następujące czynności:

1. Samba tworzy nowy udział dyskowy o nazwie `[zofia]` ze ścieżką określoną zmienną `path` w sekcji `[homes]`. Jeśli w sekcji `[homes]` nie ma opcji `path`, Samba ustawia ścieżkę na katalog macierzysty `zofii`.
2. Samba inicjuje nowy udział wartościami domyślnymi z sekcji `[globals]` oraz zmieniającymi je opcjami z sekcji `[homes]`, z wyjątkiem opcji `browseable`.
3. Samba łączy klienta `zofii` z tym udziałem.

Sekcja `[homes]` to szybki i niekłopotliwy sposób tworzenia udziałów dla społeczności użytkowników, nie wymagający powielania informacji z pliku haseł w pliku *smb.conf*. Wiąże się on jednak z pewnymi osobliwościami, o których wypada tutaj wspomnieć:

- Sekcja `[homes]` może reprezentować dowolne konto w komputerze, co nie zawsze jest pożądane. Potencjalnie może utworzyć udziały dla użytkowników *root*, *bin*, *sys*, *uucp* i podobnych (możesz temu zapobiec, ustawiając globalną opcję `invalid users`).

- Znaczenie opcji konfiguracyjnej `browseable` jest tu inne niż w pozostałych udziałach. Wskazuje ona tylko, że sekcja `[homes]` nie pojawi się na lokalnej liście przeglądania, natomiast nie odnosi się to do udziału `[zofia]`. Kiedy (po wstępnym nawiązaniu połączenia) zostanie utworzony udział `[zofia]`, wykorzysta on wartość opcji `browseable` z sekcji `[globals]`, a nie `[homes]`.

Jak wspomnieliśmy, nie ma potrzeby umieszczania opcji `path` w sekcji `[homes]`, jeśli użytkownicy mają uniksowe katalogi macierzyste wyszczególnione w pliku */etc/passwd* serwera. Powinieneś jednak upewnić się, że katalog użytkownika rzeczywiście istnieje, ponieważ Samba nie utworzy go automatycznie i nie połączy klienta z zasobem, jeśli katalog nie istnieje lub jest niedostępny.

Kontrolowanie dostępu do udziałów

Ze względów bezpieczeństwa często będziesz musiał ograniczać dostęp do konkretnych udziałów. W Sambie jest to bardzo łatwe, ponieważ masz do dyspozycji mnóstwo opcji, które pozwalają na utworzenie praktycznie dowolnej konfiguracji zabezpieczeń. Przedstawimy teraz kilka konfiguracji, które być może zechcesz wykorzystać we własnej sieci.

Przypominamy jeszcze raz, że jeśli używasz Windows 98 lub NT 4.0 Service Pack 3 (lub nowszych wersji), klienty te będą wysyłać Sambie zaszyfrowane hasła. Jeśli Samba nie zostanie odpowiednio skonfigurowana, będzie odmawiać połączenia. W tym rozdziale opiszemy konfigurowanie Samby do obsługi zaszyfrowanych haseł (patrz podrozdział „Hasła").

Wiemy już, co się stanie, kiedy podasz listę uprawnionych użytkowników. Możesz też stworzyć listę użytkowników nieuprawnionych – takich, którzy nie będą mogli uzyskać dostępu do Samby i jej udziałów. Służy do tego opcja `invalid users`. Zwracaliśmy już uwagę na najczęstsze zastosowanie tej opcji: w razie korzystania z sekcji `[homes]` warto określić jej globalną wartość, aby uniemożliwić podszywanie się pod różnych użytkowników systemowych i uprzywilejowanych. Na przykład:

```
[global]
    invalid users = root bin daemon adm sync shutdown \
                    halt mail news uucp operator gopher
    auto services = dawid piotr robert

[homes]
    browsable = no
    writable = yes
```

W opcji `invalid users`, podobnie jak w `valid users`, można podać nazwy grup zamiast nazw użytkowników. Jeśli użytkownik lub grupa figuruje na obu listach, wówczas opcja `invalid users` ma pierwszeństwo i użytkownik lub grupa nie uzyska dostępu do udziału.

Przeciwieństwem tej opcji jest opcja `admin users`, która umożliwia jawne określenie użytkowników mających uprzywilejowany dostęp do udziału (z prawami roota). Oto przykład:

```
[sprzedaz]
    path = /home/sprzedaz
    comment = Dane sprzeda□y w firmie Fikcja Sp. z o. o.
    writeable = yes
    valid users = tomek ryszard henryk
    admin users = marek
```

W opcji tej można podawać zarówno nazwy użytkowników, jak i grup. Możesz także podać nazwę grupy sieciowej NIS, poprzedzając ją znakiem `@`; jeśli Samba nie znajdzie takiej grupy sieciowej, założy, że jest to standardowa grupa uniksowa.

Przypisywanie przywilejów administracyjnych całej grupie jest dość ryzykowne. Zespół Samby stanowczo odradza używanie tej opcji, ponieważ w gruncie rzeczy nadaje ona korzystającym z udziału użytkownikom lub grupom uprawnienia roota.

Jeśli chcesz zezwolić tylko na odczyt lub na odczyt i zapis udziału, możesz podać uprawnionych do tego użytkowników w opcjach `read list` i `write list`. Opcje te można określać dla każdego udziału z osobna w celu ograniczenia dostępu do udziału zapisywalnego lub nadania uprawnień do zapisu w udziale przeznaczonym tylko do odczytu. Na przykład:

```
[sprzedaz]
    path = /home/sprzeda□
    comment =  Dane sprzeda□y w firmie Fikcja Sp. z o. o.
    read only = yes
    write list = tomek ryszard
```

Opcja `write list` nie ma wpływu na uniksowe prawa dostępu. Jeśli tworząc udział, nie nadasz użytkownikowi prawa zapisu w uniksowym katalogu, wówczas nie będzie mógł nic zapisać w udziale, niezależnie od ustawienia opcji `write list`.

Dostęp gościnny

Jak wspomnieliśmy wcześniej, możesz wyznaczyć użytkowników, którzy będą mieli gościnny dostęp do udziału. Opcje kontrolujące dostęp gościnny są nieskomplikowane. Pierwsza, `guest account`, określa konto uniksowe, które będzie przypisywane gościom łączącym się z serwerem Samby. Domyślną wartość tej opcji określa się podczas kompilacji; zwykle jest to `nobody`. Jeśli jednak masz problemy z dostępem do niektórych usług systemowych, możesz zmienić konto gościa na `ftp`.

Jeśli chcesz pozwolić na dostęp do udziału tylko gościom – innymi słowy, wszystkie klienty łączące się z udziałem będą korzystać z konta gościnnego – możesz użyć opcji `guest only` w połączeniu z opcją `guest ok`, jak w poniższym przykładzie:

```
[sprzedaz]
    path = /home/sprzeda□
    comment =  Dane sprzeda□y w firmie Fikcja Sp. z o. o.
    writeable = yes
    guest ok = yes
    guest account = ftp
    guest only = yes
```

Jeśli zastosujesz taką konfigurację, upewnij się, że wpisałeś `yes` zarówno w opcji `guest ok`, jak i `guest only`. Inaczej Samba nie użyje podanego przez ciebie konta gościnnego.

Opcje kontroli dostępu

W tabeli 6.1 zebrano opcje, dzięki którym można kontrolować dostęp do udziałów.

Tabela 6.1. Opcje kontroli dostępu do udziału

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`admin users`	Łańcuch (lista nazw użytkowników)	Określa listę użytkowników, którzy mogą wykonywać operacje z przywilejami roota	Brak	Udział
`valid users`	Łańcuch (lista nazw użytkowników)	Określa listę użytkowników, którzy mogą łączyć się z udziałem	Brak	Udział
`invalid users`	Łańcuch (lista nazw użytkowników)	Określa listę użytkowników, którzy nie mogą łączyć się z udziałem	Brak	Udział
`read list`	Łańcuch (lista nazw użytkowników)	Podaje listę użytkowników, którzy mogą tylko odczytywać pliki z udziału	Brak	Udział
`write list`	Łańcuch (lista nazw użytkowników)	Podaje listę użytkowników, którzy mogą zapisywać pliki w udziale przeznaczonym tylko do odczytu	Brak	Udział
`max connections`	Wartość liczbowa	Określa maksymalną liczbę jednoczesnych połączeń z udziałem	`0`	Udział
`guest only` (`only guest`)	Wartość logiczna	Określa, że dany udział zezwala tylko na dostęp gościnny	`no`	Udział
`guest account`	Łańcuch (nazwa konta)	Określa uniksowe konto, które będzie używane do dostępu gościnnego	`nobody`	Udział

admin users

Opcja ta określa użytkowników, którzy mogą wykonywać operacje na plikach tak, jakby mieli uprawnienia roota. Oznacza to, że mogą zmodyfikować lub zniszczyć pracę innych użytkowników, niezależnie od obowiązujących praw dostępu. Wszystkie tworzone przez nich pliki będą własnością roota i domyślnej grupy administratorów. Opcja `admin users` pozwala użytkownikom komputerów PC na zarządzanie udziałami. Odradzamy używanie tej opcji.

valid users i invalid users

Te dwie opcje pozwalają wymienić użytkowników, którym chcesz przyznać dostęp do konkretnego udziału albo odmówić dostępu. Możesz wpisać listę nazw użytkowników, oddzielając je przecinkami, albo podać nazwę grupy uniksowej lub grupy NIS, poprzedzając ją znakiem „at" (`@`).

Należy pamiętać o ważnej regule: każdy użytkownik lub grupa z listy `invalid users` *zawsze* spotka się z odmową dostępu, nawet jeśli figuruje (w dowolnej for-

mie) na liście `valid users`. Domyślnie obie opcje nie mają przypisanej żadnej wartości; w takim wypadku dostęp do udziału mają dowolni użytkownicy.

read list i write list

Podobnie jak opcje `valid users` i `invalid users`, ta para opcji określa użytkowników, którym wolno tylko odczytywać udziały zapisywalne albo zapisywać udziały przeznaczone tylko do odczytu. Wartością każdej z nich jest lista użytkowników. Opcja `read list` ma pierwszeństwo przed innymi uprawnieniami przyznanymi przez Sambę – a także prawami dostępu do plików w systemie uniksowym – i uniemożliwia zapisywanie plików w udziale. Opcja `write list` umożliwia zapis nawet wbrew ograniczeniom nałożonym przez Sambę, ale nie w sytuacji, gdy użytkownik nie ma prawa do zapisu pliku w Uniksie. Możesz podać nazwę grupy NIS lub grupy uniksowej, poprzedzając ją znakiem „at" (na przykład `@users`). Żadna z tych opcji nie ma wartości domyślnej.

max connections

Opcja ta określa maksymalną liczbę jednoczesnych połączeń z udziałem. Po przekroczeniu tego progu wszystkie następne połączenia zostaną odrzucone. Domyślną wartością tej opcji jest `0`, co oznacza, że liczba jednoczesnych połączeń jest nielimitowana. Możesz określić wartość tej opcji dla każdego udziału z osobna, jak w przykładzie poniżej:

```
[ksiegowosc]
    max connections = 30
```

Opcja ta przydaje się w sytuacji, gdy chcesz ograniczyć liczbę użytkowników korzystających z licencjonowanego programu lub zbioru danych.

guest only

Ta opcja udziału (czasem zapisywana w postaci `only guest`) wymusza połączenie z udziałem za pośrednictwem konta użytkownika określonego opcją `guest account`. Udział, do którego stosuje się opcja `guest only`, musi zawierać jawny wpis `guest ok = yes`, gdyż w przeciwnym razie Samba ją zignoruje. Domyślna wartość tej opcji to `no`.

guest account

Opcja ta określa nazwę konta, które będzie używane podczas gościnnego dostępu do udziałów Samby. Domyślna wartość tej opcji zależy od systemu, ale często używa się konta `nobody`. Niektóre domyślne konta użytkowników mają problemy z łączeniem się jako goście. W takim przypadku zespół Samby zaleca zmianę konta gościnnego na `ftp`.

Opcje nazw użytkowników

W tabeli 6.2 zamieściliśmy dwie dodatkowe opcje, dzięki którym Samba może skorygować niezgodności między nazwami użytkowników w Uniksie i Windows.

Tabela 6.2. Opcje nazw użytkowników

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`username map`	Łańcuch (nazwa pliku wraz ze ścieżką)	Ustawia nazwę pliku z odwzorowaniami nazw użytkowników	Brak	Globalny
`username level`	Wartość liczbowa	Określa liczbę dużych liter używanych podczas dopasowywania nazwy użytkownika	`0`	Globalny

username map

Nazwy użytkowników klientów w sieci SMB mogą być dość długie (do 255 znaków), natomiast nazwy użytkowników w sieci Uniksa często nie mogą przekraczać 8 znaków. Oznacza to, że użytkownik może mieć inną nazwę w kliencie, a inną (krótszą) w serwerze Samby. Możesz uporać się z tym problemem, odwzorowując nazwę użytkownika klienta na uniksową nazwę użytkownika, składającą się co najwyżej z ośmiu znaków. Należy w tym celu utworzyć standardowy plik tekstowy, którego format omówimy niebawem, a następnie podać jego nazwę wraz ze ścieżką w globalnej opcji `username map`. Powinieneś koniecznie ograniczyć dostęp do tego pliku; jego właścicielem powinien być root, a inni nie mogą mieć prawa do zapisu. W przeciwnym wypadku każdy użytkownik mający dostęp do tego pliku mógłby odwzorować swoją nazwę w kliencie na nazwę uprzywilejowanego użytkownika serwera Samby.

Opcji tej możesz użyć w następujący sposób:

```
[global]
    username map = /etc/samba/usermap.txt
```

Każdy wpis w pliku odwzorowań powinien składać się z następujących elementów: uniksowej nazwy użytkownika, znaku równości i jednej lub wielu nazw użytkowników klientów SMB, oddzielonych znakami odstępu. Zauważ, że przy braku odpowiednich instrukcji (na przykład podczas połączenia gościnnego), Samba będzie się spodziewała, że użytkownik serwera i użytkownik klienta mają to samo hasło. Możesz także odwzorowywać grupy NT na jedną lub więcej grup uniksowych, używając znaku `@`. Oto kilka przykładów:

```
jkowal = JanKowalski
markwiat = MarekKwiatkowski
users = @ksieg
```

W pliku odwzorowań nazw możesz też użyć gwiazdki, która odpowiada dowolnej nazwie użytkownika klienta:

```
nobody = *
```

Komentarze można wpisać, zaczynając linię od znaku (`#`) lub (`;`).

Zauważ, że dzięki temu plikowi możesz także odwzorować jednego użytkownika uniksowego na drugiego. Uważaj z tym jednak, gdyż w takim przypadku Samba

i klient mogą nie powiadomić użytkownika o dokonaniu odwzorowania, a Samba będzie oczekiwać innego hasła.

username level

Klienty SMB (takie jak Windows) często wysyłają nazwy użytkowników w żądaniach SMB zapisane dużymi literami; innymi słowy, wielkość liter w nazwach użytkowników klientów nie ma znaczenia. W serwerze uniksowym wielkość liter jest jednak rozróżniana: użytkownik `ANDRZEJ` różni się od użytkownika `andrzej`. Domyślnie Samba rozwiązuje ten problem w następujący sposób:

1. Sprawdza, czy istnieje konto użytkownika o nazwie identycznej z tą przysłaną przez klienta.
2. Sprawdza nazwę użytkownika zapisaną samymi małymi literami.
3. Sprawdza nazwę użytkownika zapisaną małymi literami, ale rozpoczynającą się dużą literą.

Jeśli chcesz, aby Samba wypróbowywała więcej kombinacji liter dużych i małych, możesz posłużyć się opcją konfiguracyjną `username level`. Opcja ta przyjmuje liczbę całkowitą, która określa, ile liter w nazwie użytkownika należy zamienić na duże litery podczas próby połączenia z udziałem. Opcji tej używa się następująco:

```
[global]
    username level = 3
```

W tym przykładzie Samba wypróbuje wszystkie permutacje nazw użytkownika, które da się utworzyć z wykorzystaniem trzech dużych liter. Im większa wartość opcji, tym więcej obliczeń będzie musiała wykonać Samba w celu dopasowania nazwy użytkownika i tym dłużej będzie trwało uwierzytelnianie.

Uwierzytelnianie użytkowników

Teraz pora na omówienie uwierzytelniania użytkowników przez Sambę. Każdy użytkownik, który próbuje połączyć się z udziałem nie dopuszczającym gościnnego dostępu, musi podać hasło, aby pomyślnie nawiązać połączenie. Operacje, które Samba przeprowadza na tym haśle – a w konsekwencji strategia uwierzytelniania użytkowników – zależą od opcji konfiguracyjnej `security`. Obecnie istnieją cztery poziomy zabezpieczeń, które Samba może obsługiwać w swojej sieci: poziom *udziału*, *użytkownika*, *serwera* i *domeny*.

Zabezpieczenia na poziomie udziału
: Każdy udział w grupie roboczej jest chroniony oddzielnym hasłem. Każdy, kto zna hasło dostępu do udziału, może z tego udziału korzystać.

Zabezpieczenia na poziomie użytkownika
: Każdy udział w grupie roboczej jest skonfigurowany tak, aby pozwalać na dostęp tylko określonym użytkownikom. Po wstępnym nawiązaniu połączenia z zasobem, serwer Samby weryfikuje użytkowników i ich hasła, zanim przyzna im dostęp do udziału.

Zabezpieczenia na poziomie serwera
Podobne do zabezpieczeń na poziomie użytkownika, ale Samba używa oddzielnego serwera SMB, który zatwierdza użytkowników i ich hasła przed przyznaniem dostępu do udziału.

Zabezpieczenia na poziomie domeny
Samba staje się członkiem domeny Windows i uwierzytelnia klienty za pośrednictwem podstawowego kontrolera domeny (PDC). Po uwierzytelnieniu użytkownik otrzymuje specjalny żeton, który umożliwia mu korzystanie ze wszystkich udziałów, do których ma odpowiednie uprawnienia. Dzięki temu żetonowi, PDC nie będzie musiał ponownie weryfikować hasła użytkownika za każdym razem, kiedy ten spróbuje połączyć się z innym udziałem w domenie.

Omówione wyżej mechanizmy bezpieczeństwa można włączyć za pomocą globalnej opcji `security` (patrz tabela 6.3).

Tabela 6.3. Opcja security

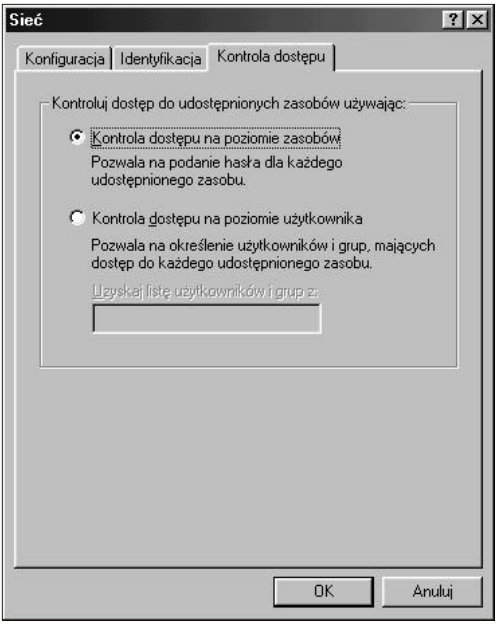
Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`security`	`domain`, `server`, `share` lub `user`	Określa poziom bezpieczeństwa serwera Samby	`user` (Samba 2.0) lub `share` (Samba 1.9)	Globalny

Zabezpieczenia na poziomie udziału

Jeśli korzystasz z zabezpieczeń na poziomie udziału, każdy udział ma jedno lub kilka związanych z nim haseł. W stosunku do innych trybów bezpieczeństwa nie ma tutaj żadnych dodatkowych ograniczeń co do tego, kto może korzystać z udziału, jeśli tylko ta osoba zna poprawne hasło. Udziały często mają kilka różnych haseł. Jedno z nich może umożliwiać dostęp tylko do odczytu, inne dawać prawo do zapisu i tak dalej. Bezpieczeństwo jest zachowywane dopóty, dopóki nieautoryzowani użytkownicy nie odkryją hasła udziału, do którego nie powinni mieć dostępu.

Systemy OS/2 i Windows 95/98 umożliwiają zabezpieczanie swoich zasobów na poziomie udziału. W Windows 95/98 trzeba otworzyć okno dialogowe Sieć z Panelu sterowania i kliknąć kartę Kontrola dostępu. Następnie należy zaznaczyć pole opcji Kontrola dostępu na poziomie zasobów (co spowoduje usunięcie zaznaczenia z pola Kontrola dostępu na poziomie użytkownika) i kliknąć przycisk OK (patrz rysunek 6.1).

Następnie kliknij prawym przyciskiem myszy dowolny zasób – na przykład dysk twardy lub CD-ROM – i wybierz z menu polecenie Właściwości. Ukaże się okno dialogowe z właściwościami zasobu. Wybierz kartę Udostępnianie na górze okna dialogowego i zaznacz pole opcji Udostępniony jako. Tutaj możesz określić, jak współdzielony zasób będzie widziany przez różnych użytkowników. Możesz także zdecydować, czy zasób będzie przeznaczony tylko do odczytu, do odczytu i zapisu, czy też będzie pozwalał na zapis w zależności od podanego hasła.



Rysunek 6.1. Ustawianie zabezpieczeń na poziomie udziału

Być może sądzisz, że taki model bezpieczeństwa niezbyt przystaje do Samby. I masz rację. W rzeczywistości, jeśli ustawisz opcję `security = share` w pliku konfiguracyjnym, Samba nadal będzie uwierzytelniać użytkowników na podstawie ich nazw i haseł zapisanych w systemowych plikach haseł. Ściślej rzecz biorąc, kiedy klient zażąda połączenia przy zabezpieczeniu na poziomie udziału, Samba podejmie następujące czynności:

1. Kiedy nadejdzie żądanie połączenia, Samba przyjmie hasło i nazwę użytkownika klienta (jeśli zostanie przesłana).
2. Jeśli udział jest przeznaczony tylko dla gości (opcja `guest only`), użytkownik natychmiast uzyska dostęp z uprawnieniami określonymi przez opcję `guest account`; Samba nie będzie weryfikować hasła.
3. W przypadku innych udziałów Samba dołączy nazwę użytkownika do listy użytkowników, którzy mogą korzystać z udziału. Następnie spróbuje zweryfikować hasło przesłane wraz z nazwą użytkownika. Jeśli operacja ta się powiedzie, Samba przyzna dostęp do udziału z uprawnieniami przypisanymi danemu użytkownikowi. Użytkownik nie będzie ponownie uwierzytelniany, chyba że w definicji udziału znajduje się opcja `revalidate = yes`.
4. Jeśli uwierzytelnianie się nie powiedzie, Samba spróbuje zweryfikować hasło, korzystając z listy skompilowanej podczas wcześniejszych prób nawiązania połączenia, a także z opcji określonych w definicjach udziałów w pliku konfiguracyjnym. Jeśli hasło nie odpowiada żadnej nazwie użytkownika (określonej w systemowym pliku haseł, zwykle */etc/passwd*), użytkownik nie uzyska dostępu do udziału pod tą nazwą użytkownika.

5. Jeżeli jednak w definicji udziału ustawiono opcje `guest ok` lub `public`, użytkownik uzyska dostęp z uprawnieniami określonymi opcją `guest account`.

W pliku konfiguracyjnym możesz podać użytkowników, którzy powinni zostać wstępnie wpisani na listę kontroli dostępu na poziomie udziału, używając opcji `username`, jak w poniższym przykładzie:

```
[global]
    security = share
[ksiegowosc1]
    path = /home/samba/ksiegowosc1
    guest ok = no
    writable = yes
    username = dawid, piotr, andrzej
```

W takim przypadku, gdy użytkownik spróbuje połączyć się z udziałem, Samba zweryfikuje przesłane hasło w oparciu o własną listę użytkowników, a także o hasła użytkowników `dawid`, `piotr` i `andrzej`. Jeśli znajdzie pasujące hasło, połączenie zostanie zatwierdzone i użytkownik uzyska dostęp do udziału. W przeciwnym razie połączenie z tym zasobem nie powiedzie się.

Opcje zabezpieczeń na poziomie udziału

W tabeli 6.4 wymienione są opcje zwykle kojarzone z zabezpieczeniami na poziomie udziału.

Tabela 6.4. Opcje kontroli dostępu na poziomie udziału

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`only user`	Wartość logiczna	Określa, czy jedynymi dopuszczalnymi nazwami użytkownika będą te określone w opcji `username`	`no`	Udział
`username` (`user` lub `users`)	Łańcuch (lista nazw użytkowników)	Podaje listę użytkowników, których hasła będą porównywane z hasłem dostarczonym przez klienta	Brak	Udział

only user

Ta opcja określa, czy Samba zezwoli na połączenia z udziałem (jeśli wykorzystywane są zabezpieczenia na poziomie udziału) tylko tym użytkownikom, którzy zostali określeni w opcji `username`, a nie tym zgromadzonym na wewnętrznej liście Samby. Domyślna wartość tej opcji to `no`. Możesz zmienić ją dla każdego udziału z osobna, jak w poniższym przykładzie:

```
[global]
    security = share
[dane]
    username = andrzej, piotr, weronika
    only user = yes
```

username

Opcja ta wymienia użytkowników, których hasła Samba porówna z hasłem dostarczonym przez klienta. Zwykle używa się jej po to, aby pozwolić klientom (korzystającym z zabezpieczeń na poziomie udziału) na połączenie się z konkretnym zasobem tylko na podstawie odpowiedniego hasła – takiego, które odpowiada hasłu określonego użytkownika.

```
[global]
    security = share
[dane]
    username = andrzej, piotr, anna
```

Nie zalecamy używania tej opcji, chyba że konfigurujesz serwer Samby do pracy z zabezpieczeniami na poziomie udziału.

Zabezpieczenia na poziomie użytkownika

Preferowany model bezpieczeństwa w Sambie to *zabezpieczenia na poziomie użytkownika*. W tym rodzaju zabezpieczenia każdemu udziałowi przypisuje się użytkowników, którzy mogą z niego korzystać. Kiedy użytkownik zażąda połączenia z udziałem, Samba uwierzytelnia go, porównując jego nazwę i hasło z listą autoryzowanych użytkowników w pliku konfiguracyjnym oraz z plikiem haseł serwera Samby. Jak wspomniano wcześniej w tym rozdziale, jednym ze sposobów ograniczenia dostępności udziału jest użycie opcji `valid users`:

```
[global]
    security = user
[ksiegowosc1]
    writable = yes
    valid users = robert, jacek, sylwia
```

Każdy z wymienionych użytkowników będzie mógł połączyć się z udziałem, jeśli dostarczone przez niego hasło będzie odpowiadać temu przechowywanemu w bazie haseł serwera. Jeśli wstępne uwierzytelnienie powiedzie się, użytkownik nie będzie musiał ponownie wpisywać hasła, aby połączyć się z udziałem, o ile nie ustawiono opcji `revalidate = yes`.

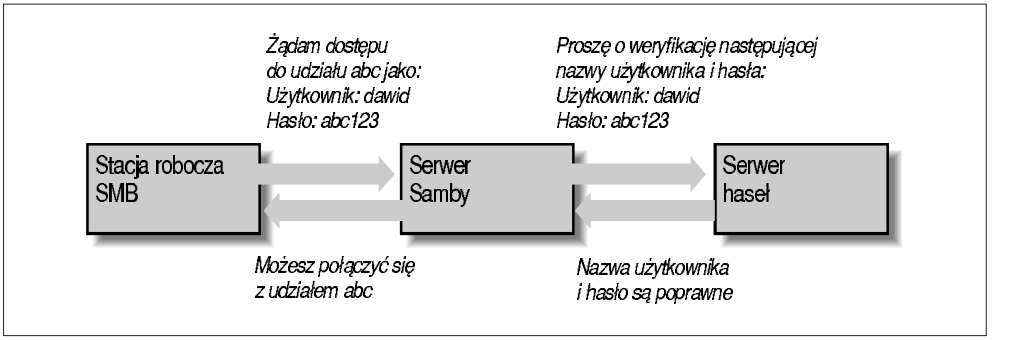
Hasła mogą być przesyłane do serwera Samby w postaci zaszyfrowanej lub niezaszyfrowanej. Jeśli w twojej sieci znajdują się systemy obu typów, powinieneś upewnić się, że hasła poszczególnych użytkowników są przechowywane zarówno w tradycyjnej bazie haseł, jak i w bazie zaszyfrowanych haseł Samby. Dzięki temu autoryzowani użytkownicy będą mogli uzyskać dostęp do swoich udziałów z klienta dowolnego typu*. Jeśli jednak zależy ci szczególnie na bezpieczeństwie sieci, zalecamy stosowanie haseł zaszyfrowanych i rezygnację z haseł niezaszyfrowanych. W podrozdziale „Hasła" wyjaśnimy, jak używać haseł obu typów.

---

* To, że w sieci mogą znajdować się zarówno klienty używające haseł zaszyfrowanych, jak i niezaszyfrowanych, jest jeszcze jednym powodem, dla którego Samba pozwala dołączać (lub nie) różne opcje do pliku konfiguracyjnego w zależności od systemu operacyjnego klienta lub nazwy komputera.

Zabezpieczenia na poziomie serwera

Kontrola dostępu na poziomie serwera przypomina kontrolę na poziomie użytkownika. Jednakże w tym przypadku Samba deleguje uwierzytelnianie haseł do serwera haseł SMB, zwykle innego serwera Samby lub serwera Windows NT działającego jako podstawowy kontroler domeny. Zauważ, że Samba nadal przechowuje swoją listę udziałów i ich konfiguracji w pliku *smb.conf*. Kiedy klient próbuje nawiązać połączenie z udziałem, Samba sprawdza, czy użytkownik jest rzeczywiście uprawniony do korzystania z udziału. Następnie próbuje zweryfikować hasło, łącząc się z serwerem haseł SMB za pośrednictwem znanego protokołu i przedstawiając do zatwierdzenia nazwę użytkownika i hasło. Jeśli hasło zostanie zaakceptowane, Samba nawiąże sesję z klientem. Konfigurację tę przedstawia rysunek 6.2.



Rysunek 6.2. Typowa konfiguracja zabezpieczeń na poziomie serwera

Możesz skonfigurować Sambę do współpracy z serwerem haseł (kiedy używasz zabezpieczeń na poziomie serwera) za pomocą globalnej opcji `password server`, jak w poniższym przykładzie:

```
[global]
    security = server
    password server = FENIKS120 HYDRA134
```

Zauważ, że w opcji `password server` możesz podać nazwy kilku komputerów. Samba będzie łączyć się z kolejnym serwerem na liście, jeśli pierwszy wybrany będzie niedostępny. Serwery w opcji `password server` określa się za pomocą nazw NetBIOS-owych, a nie nazw DNS lub równoważnych im adresów IP. Jeśli zaś któryś z serwerów odrzuci podane hasło, połączenie nie powiedzie się – Samba nie będzie łączyć się z następnym serwerem.

Jedno zastrzeżenie: mimo że korzystasz z tej opcji, nadal musisz mieć konto reprezentujące użytkownika w serwerze Samby. A to dlatego, że Unix wymaga nazwy użytkownika podczas wykonywania różnych operacji wejścia-wyjścia. Aby ominąć ten problem, najczęściej zakłada się konto użytkownika w serwerze Samby i wyłącza hasło tego konta przez zastąpienie go gwiazdką (*) w systemowym pliku haseł (na przykład */etc/passwd*).

Zabezpieczenia na poziomie domeny

Kontrola dostępu na poziomie domeny przypomina kontrolę na poziomie użytkownika. Jednakże w tym wypadku Samba działa jako członek domeny Windows. Jak dowiedziałeś się w rozdziale 1, w każdej domenie znajduje się *kontroler domeny* – zwykle serwer Windows NT świadczący usługi uwierzytelniania haseł. Dzięki kontrolerowi domeny grupa robocza dysponuje autorytatywnym serwerem haseł. Kontrolery domeny przechowują nazwy użytkowników i hasła we własnym module bezpieczeństwa (*Security Authentication Module*, SAM) i uwierzytelniają użytkowników, gdy ci logują się po raz pierwszy lub chcą skorzystać z udziałów innego komputera.

Jak wspomniano już w tym rozdziale, Samba może obsługiwać zabezpieczenia na poziomie użytkownika, ale opcja ta jest uniksocentryczna i zakłada uwierzytelnianie użytkowników z wykorzystaniem uniksowych plików haseł. Jeśli uniksowy komputer wchodzi w skład domeny NIS lub NIS+, Samba będzie uwierzytelniać użytkowników w oparciu o współdzielony plik haseł, na sposób typowo uniksowy. Samba umożliwia zatem dostęp do domeny NIS lub NIS+ z poziomu Windows. Oczywiście, pojęcie domeny NIS i domeny Windows to dwie różne rzeczy.

Dzięki zabezpieczeniom na poziomie domeny możemy wykorzystać rodzimy mechanizm bezpieczeństwa Windows NT. Metoda taka ma wiele zalet:

- Umożliwia znacznie ściślejszą integrację z NT: w opcjach pliku *smb.conf* mających związek z obsługą domen jest znacznie mniej „prowizorki" niż w innych opcjach odpowiadających za współpracę z Windows. Dzięki temu można wykorzystać w szerszym zakresie narzędzia do zarządzania systemem NT, na przykład program User Manager for Domains, który pozwala personelowi wsparcia technicznego traktować serwery Samby jak komputery NT.
- Lepsza integracja oznacza poprawki w protokole i kodzie, dzięki którym zespół programistów Samby może łatwiej śledzić ewoluujące implementacje systemu NT. NT Service Pack 4 koryguje pewne błędy protokołu, a ściślejsza integracja ułatwia wyśledzenie tych zmian i zaadaptowanie się do nich.
- Zmniejsza się obciążenie podstawowego kontrolera domeny, ponieważ niepotrzebne jest jedno ze stałych połączeń między nim a serwerem Samby. W przeciwieństwie do protokołu wykorzystywanego w razie użycia opcji `security = server`, serwer Samby może korzystać ze zdalnych wywołań procedury (*Remote Procedure Call*, RPC) tylko wtedy, gdy chce uzyskać informacje uwierzytelniające. Nie musi w tym celu utrzymywać stałego połączenia.
- Wreszcie, mechanizm domenowego uwierzytelniania NT zwraca pełny zbiór atrybutów użytkownika, nie tylko informację o powodzeniu lub błędzie. W skład tych atrybutów wchodzą dłuższe, sieciowe wersje uniksowego identyfikatora użytkownika, grupy NT i inne informacje, w tym:
  - nazwa użytkownika,
  - nazwisko,
  - opis,

- identyfikator bezpieczeństwa (domenowe rozszerzenie uniksowego identyfikatora użytkownika),
- przynależność do grup NT,
- dozwolone godziny logowania oraz informacja o tym, czy użytkownik będzie zmuszony do natychmiastowego wylogowania się,
- stacje robocze, z których może korzystać użytkownik,
- data wygaśnięcia konta,
- katalog macierzysty,
- skrypt logowania,
- profil,
- typ konta.

- Twórcy Samby wykorzystali zabezpieczenia na poziomie domeny w wersji 2.0.4 Samby do półautomatycznego dodawania i usuwania użytkowników domenowych. Model ten umożliwia też stosowanie dodatkowych funkcji znanych z NT, takich jak obsługa list kontroli dostępu i zmienianie praw dostępu do plików z klienta.

Zaletą takiego podejścia jest zmniejszenie nakładu pracy administratora: wystarczy synchronizować jedną bazę informacji uwierzytelniających. Lokalne administrowanie serwerem Samby sprowadza się do tworzenia katalogów roboczych dla użytkowników oraz dodawania wpisów z identyfikatorami i grupami użytkowników do pliku */etc/passwd*.

Dodawanie serwera Samby do domeny Windows NT

Jeśli masz już domenę Windows NT, możesz łatwo dodać do niej serwer Samby. Najpierw musisz zatrzymać demony Samby. Następnie dodaj serwer Samby do domeny NT z podstawowego kontrolera domeny, używając programu Windows NT Server Manager for Domains. Kiedy program zapyta o typ komputera, wybierz „Windows NT Workstation or Server" i podaj NetBIOS-ową nazwę serwera Samby. W ten sposób utworzysz konto na serwerze NT.

Następnie wygeneruj hasło dla komputera za pomocą narzędzia *smbpasswd*, które omówiono dokładnie w następnym podrozdziale. Jeśli na przykład nasza domena ma nazwę PROSTA, a podstawowy kontroler domeny Windows NT to `beowulf`, powinieneś wydać następujące polecenie na serwerze Samby:

```
smbpasswd -j PROSTA -r beowulf
```

Teraz dopisz poniższe opcje do sekcji `[global]` pliku *smb.conf* i ponownie uruchom demony Samby.

```
[global]
    security = domain
    domain logins = yes
    workgroup = PROSTA
    password server = beowulf
```

Samba powinna teraz korzystać z zabezpieczeń na poziomie domeny. Opcję `domain logins` wyjaśnimy szczegółowo dalej w tym rozdziale.

Hasła

Hasła w Sambie to sprawa najeżona trudnościami – do tego stopnia, że niemal zawsze stanowią pierwszy poważny problem po zainstalowaniu Samby i ich właśnie dotyczy zdecydowana większość pytań w grupach dyskusyjnych poświęconych Sambie. W poprzednich rozdziałach unikaliśmy stosowania haseł, umieszczając w każdym pliku konfiguracyjnym opcję `guest ok`, co pozwalało na nawiązywanie połączeń bez uwierzytelniania haseł. W tym momencie musimy jednak przyjrzeć się bliżej działaniu Samby, aby dowiedzieć się, co się dzieje w sieci.

Hasła wysyłane przez poszczególne klienty mogą być albo zaszyfrowane, albo niezaszyfrowane. Zaszyfrowane hasła są oczywiście dużo bezpieczniejsze. Niezaszyfrowane hasło można łatwo odczytać za pomocą programu przechwytującego pakiety, na przykład zmodyfikowanej wersji programu *tcpdump*, której użyliśmy w rozdziale 3, *Konfigurowanie klientów Windows*. Szyfrowanie haseł jest zależne od systemu operacyjnego klienta łączącego się z serwerem Samby. Tabela 6.5 pokazuje, które odmiany systemu Windows szyfrują swoje hasła przed wysłaniem ich do uwierzytelnienia przez podstawowy kontroler domeny. Jeśli twój klient nie działa pod kontrolą Windows, sprawdź w dokumentacji, czy szyfruje hasła SMB.

Tabela 6.5. Systemy operacyjne Windows i szyfrowanie haseł

System operacyjny	*Hasła*
Windows 95	Niezaszyfrowane
Windows 95 z uaktualnieniem SMB	Zaszyfrowane
Windows 98	Zaszyfrowane
Windows NT 3.*x*	Niezaszyfrowane
Windows NT 4.0 przed SP3	Niezaszyfrowane
Windows NT 4.0 po SP3	Zaszyfrowane

W użyciu są dwie różne metody szyfrowania: jedna dla klientów Windows 95 i 98, wykorzystująca algorytm LAN Managera Microsoftu, a druga dla klientów i serwerów Windows NT. Windows 95 i 98 używają starszej metody szyfrowania, odziedziczonej po oprogramowaniu sieciowym LAN Manager, natomiast serwery i klienty Windows NT korzystają z nowszych algorytmów szyfrowania.

Jeśli Samba obsługuje zaszyfrowane hasła, wówczas przechowuje je w pliku o nazwie *smbpasswd*. Domyślnie plik ten znajduje się w katalogu *private* dystrybucji Samby (*/usr/local/samba/private*). Klient również przechowuje zaszyfrowane hasło użytkownika w swoim systemie. Hasło pisane jawnym tekstem nie jest przechowywane w żadnym systemie. Oba systemy automatycznie szyfrują ustawione lub zmienione hasło za pomocą tego samego algorytmu.

Kiedy klient zażąda dostępu do serwera SMB obsługującego zaszyfrowane hasła (jak na przykład Samba lub Windows NT), dwa komputery prowadzą negocjacje następująco:

1. Klient próbuje wynegocjować protokół z serwerem.
2. Serwer określa protokół i zaznacza, że obsługuje zaszyfrowane hasła. W tym momencie przesyła losowo wygenerowany, 8-bitowy łańcuch wyzwania.
3. Klient używa łańcucha wyzwania jako klucza do wtórnego zaszyfrowania swojego (już zaszyfrowanego) hasła, korzystając z algorytmu zdefiniowanego przez wynegocjowany protokół. Następnie wysyła wynik do serwera.
4. Serwer robi to samo z zaszyfrowanym hasłem przechowywanym w swojej bazie danych. Jeśli wyniki są zgodne, oznacza to, że hasła są równoważne, więc użytkownik zostaje uwierzytelniony.

Zauważ, że choć pierwotne hasła nie są używane w procesie uwierzytelniania, musisz strzec zaszyfrowanych haseł z pliku *smbpasswd* przed nieautoryzowanymi użytkownikami. Jeśli hasła te zostaną ujawnione, nieautoryzowany użytkownik może włamać się do systemu, odtwarzając etapy opisanego algorytmu. Zaszyfrowane hasła są równie podatne na przejęcie, jak hasła niezaszyfrowane – w świecie kryptografii o takich danych mówi się, że są *równoważne jawnemu tekstowi*. Oczywiście, powinieneś upewnić się, że klienty również chronią swoje hasła równoważne jawnemu tekstowi.

Aby skonfigurować Sambę do obsługi zaszyfrowanych haseł, powinieneś dopisać następujące linie do pliku *smb.conf*. Zauważ, że jawnie podajemy położenie pliku haseł Samby:

```
[global]
    security = user
    encrypt passwords = yes
    smb passwd file = /usr/local/samba/private/smbpasswd
```

Samba nie zaakceptuje jednak żadnych użytkowników, dopóki nie zainicjujesz pliku *smbpasswd*.

Wyłączanie zaszyfrowanych haseł w kliencie

Choć już od dziesięcioleci stosuje się uniksowe metody uwierzytelniania, w tym zdalny dostęp przez Internet za pomocą poleceń *telnet* i *rlogin*, dobrze znane są ich słabe punkty. Hasła są wysyłane przez Internet jawnym tekstem i mogą zostać przechwycone z pakietów TCP. Jeśli jednak uważasz, że twoja sieć jest bezpieczna i chcesz używać standardowego uwierzytelniania wszystkich klientów z wykorzystaniem pliku */etc/passwd*, możesz tak zrobić, ale będziesz musiał wyłączyć szyfrowanie haseł w tych klientach Windows, które domyślnie z niego korzystają.

W tym celu musisz zmodyfikować rejestr Windows, instalując dwa pliki w każdym systemie. W zależności od platformy będą to pliki *NT4_PlainPassword.reg* lub *Win95_PlainPassword.reg*. Instalacja polega na skopiowaniu odpowiednich plików *.reg* z katalogu */docs* dystrybucji Samby na dyskietkę dosową i uruchomieniu ich za pomocą polecenia Uruchom w menu Start. Plik *.reg* przeznaczony dla Windows 95 działa także w Windows 98.

Po ponownym uruchomieniu komputera klient nie będzie szyfrował swoich haseł przed wysłaniem ich do serwera. Oznacza to, że hasła równoważne jawnemu teksto-

wi będą widoczne w pakietach TCP rozgłaszanych w sieci. Podkreślmy jeszcze raz, że jest to niewskazane, jeśli nie jesteś całkowicie pewien, że twoja sieć jest bezpieczna.

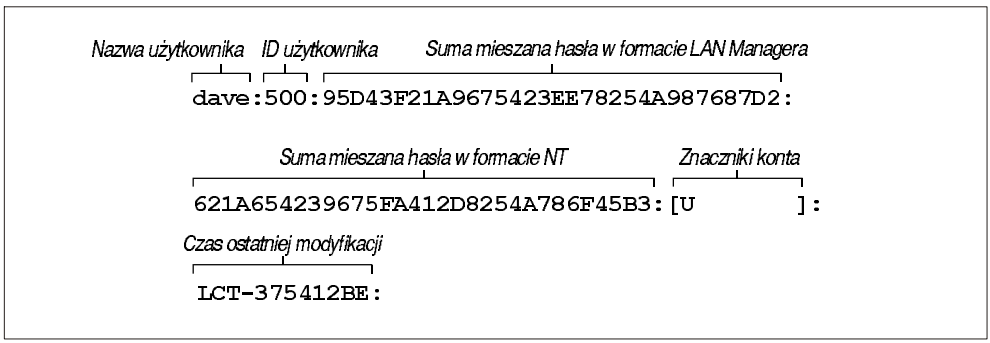
Jeśli hasła nie są szyfrowane, powinieneś zaznaczyć to w pliku konfiguracyjnym Samby:

```
[global]
    security = user
    encrypt passwords = no
```

Plik smbpasswd

Samba przechowuje zaszyfrowane hasła w pliku o nazwie *smbpasswd*, który domyślnie znajduje się w katalogu */usr/local/samba/private*. Plik *smbpasswd* należy chronić dokładnie tak samo, jak plik *passwd*; powinien znajdować się w katalogu, w którym prawo do odczytu i zapisu ma tylko użytkownik root. Inni użytkownicy w ogóle nie powinni mieć prawa do odczytu tego katalogu. Oprócz tego sam plik *smbpasswd* powinien być dostępny tylko dla roota.

Zanim zaczniesz korzystać z zaszyfrowanych haseł, musisz utworzyć wpis dla każdego uniksowego użytkownika w pliku *smbpasswd*. Struktura tego pliku przypomina nieco uniksowy plik *passwd*, ale ma inne pola. Na rysunku 6.3 przedstawiono składnię wpisu w pliku *smbpasswd*; w rzeczywistości wpis ten zajmuje tylko jedną linię pliku, którą złamano na potrzeby rysunku.



Rysunek 6.3. Struktura wpisu w pliku smbpasswd (w rzeczywistości jedna linia)

Oto opis poszczególnych pól:

Nazwa użytkownika
: Jest to nazwa użytkownika konta. Jest pobierana bezpośrednio z systemowego pliku haseł.

Identyfikator użytkownika
: Jest to identyfikator użytkownika konta. Podobnie jak nazwa, jest pobierany bezpośrednio z systemowego pliku haseł i musi odpowiadać użytkownikowi, którego tam reprezentuje.

Suma mieszana hasła w formacie LAN Managera
Jest to 32-bitowa sekwencja szesnastkowa reprezentująca hasło używane przez klienty Windows 95 i 98. Otrzymuje się ją przez zaszyfrowanie łańcucha `KGS!@#$%` za pomocą 56-bitowego algorytmu DES, z wykorzystaniem klucza w postaci dwukrotnie powtórzonego hasła użytkownika (które wyrównuje się do długości 14 bajtów i przekształca na duże litery). Jeśli użytkownik obecnie nie ma hasła, suma mieszana będzie składać się z tekstu `NO PASSWORD` uzupełnionego znakami `X` aż do końca pola. Jeśli zaś hasło zostanie wyłączone, suma będzie się składać z 32 znaków `X`. Samba nie zezwoli na dostęp użytkownikowi bez hasła, chyba że ustawiona jest opcja `null passwords`.

Suma mieszana hasła w formacie NT
Jest to 32-bitowa sekwencja szesnastkowa reprezentująca hasło używane przez klienty Windows NT. Otrzymuje się ją przez zaszyfrowanie hasła użytkownika (reprezentowanego przez 16-bitową sekwencję Unikodu w formacie „młodszy bajt – starszy bajt") za pomocą sumy MD4. Hasło nie jest wstępnie zamieniane na duże litery.

Znaczniki konta
Pole to składa się z 11 znaków ujętych w nawiasy kwadratowe (`[]`). Może zawierać poniższe znaki, występujące w dowolnej kolejności, i powinno być uzupełnione spacjami.

- *U* To konto jest standardowym kontem użytkownika.
- *D* To konto jest obecnie wyłączone i Samba nie powinna zezwalać na logowanie.
- *N* To konto nie ma przypisanego hasła.
- *W* Jest to konto zaufanej stacji roboczej, którego można użyć do skonfigurowania Samby jako podstawowego kontrolera domeny, gdy chcemy zezwolić komputerom NT na dołączanie się do jej domeny.

Czas ostatniej modyfikacji
To pole składa się ze znaków `LCT-`, po których następuje szesnastkowa reprezentacja liczby sekund, które upłynęły od początku epoki (północ 1 stycznia 1970 roku) do momentu ostatniej zmiany hasła.

Dodawanie wpisów do pliku smbpasswd

Istnieje kilka sposobów na dodanie nowego wpisu do pliku *smbpasswd*:

- Możesz użyć programu *smbpasswd* z opcją `-a`, aby automatycznie dodać dowolnego użytkownika, który obecnie ma standardowe konto uniksowe w serwerze. Program ten znajduje się w katalogu */usr/local/samba/bin*.
- Możesz użyć pliku wykonywalnego *addtosmbpass*, znajdującego się w katalogu */usr/local/samba/bin*. Jest to prosty skrypt *awk*, który analizuje systemowy plik haseł i wydobywa z niego nazwę i identyfikator użytkownika dla każdego wpisu, który chcesz dodać do pliku haseł SMB. Następnie umieszcza w pozostałych polach wpisu domyślne wartości, które możesz później zmienić za pomocą programu *smbpasswd*. Aby skorzystać z tego skryptu, prawdopodobnie będziesz musiał

zmienić jego pierwszą linię, która musi wskazywać położenie programu *awk* w twoim systemie.

- Gdyby oba powyższe sposoby zawiodły, możesz ręcznie utworzyć domyślny wpis w pliku *smbpasswd*. Wpis powinien mieścić się w jednej linii. Każde pole powinno być oddzielone dwukropkiem, a cały wpis powinien wyglądać mniej więcej tak:

```
dawid:500:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:
[U          ]:LCT-00000000:
```

  Wpis składa się z nazwy i identyfikatora użytkownika (jak w systemowym pliku haseł), dwóch zbiorów zawierających po 32 znaki X, znaczników konta i czasu ostatniej modyfikacji. Kiedy dodasz ten wpis, będziesz musiał zmienić hasło użytkownika za pomocą programu *smbpasswd*.

Zmienianie zaszyfrowanego hasła

Programu *smbpasswd* możesz użyć także wtedy, gdy chcesz zmienić zaszyfrowane hasło w pliku *smbpasswd*. Zauważ, że program do zmieniania haseł ma taką samą nazwę, co plik haseł, więc nie pomyl jednego z drugim.

Program *smbpasswd* działa niemal identycznie jak program *passwd*, służący do zmieniania haseł uniksowych. Program prosi po prostu o wprowadzenie starego hasła (o ile nie jesteś użytkownikiem uprzywilejowanym), a następnie o dwukrotne wpisanie nowego. Znaki składające się na hasła nie są wyświetlane na ekranie.

```
# smbpasswd dawid
Old SMB password:
New SMB password:
Retype new SMB password:
Password changed for user dawid
```

Po wydaniu tego polecenia możesz zajrzeć do pliku *smbpasswd* i upewnić się, czy sumy mieszane hasła w formacie LAN Managera i NT zostały zapisane w odpowiednich polach. Kiedy w bazie danych znajdą się hasła użytkowników, będą oni mogli łączyć się z udziałami, używając zaszyfrowanych haseł!

Synchronizowanie haseł

Gdy musisz posługiwać się zarówno zwykłą, jak i zaszyfrowaną wersją tego samego hasła, wprowadzanie w nich jakichkolwiek zmian może być kłopotliwe. Na szczęście Samba ma pewne możliwości synchronizowania haseł. Istnieją opcje konfiguracyjne, które pozwalają na automatyczne uaktualnianie zwykłego hasła uniksowego, kiedy ulegnie zmianie jego zaszyfrowana wersja. Aby uaktywnić tę funkcję, należy ustawić globalną opcję konfiguracyjną `unix password sync`:

```
[global]
    encrypt passwords = yes
    smb passwd file = /usr/local/samba/private/smbpasswd

    unix password sync = yes
```

Po włączeniu tej opcji Samba będzie próbowała zmienić zwykłe hasło użytkownika (jako `root`), kiedy program *smbpasswd* będzie zmieniał jego zaszyfrowaną wersję. Żeby to zadziałało, należy jednak odpowiednio ustawić dwie inne opcje.

Łatwiejsza w użyciu jest opcja `passwd program`. W opcji tej wystarczy podać polecenie Uniksa, które służy do zmieniania standardowego hasła systemowego. Jej domyślna wartość to `/bin/passwd %u`. W niektórych wersjach Uniksa nie trzeba nic zmieniać, inne – jak Linux Red Hat – używają ścieżki */usr/bin/passwd*. Co więcej, być może w przyszłości zechcesz wykorzystać inny program lub skrypt. Załóżmy, że do zmiany hasła użytkownika chcesz użyć skryptu `zmienhaslo`. Jak pamiętasz, zmienna `%u` reprezentuje bieżącą uniksową nazwę użytkownika. Oto przykład:

```
[global]
    encrypt passwords = yes
    smb passwd file = /usr/local/samba/private/smbpasswd

    unix password sync = yes
    passwd program = zmienhaslo %u
```

Warto zaznaczyć, że ten program zostanie wywołany z przywilejami roota, jeśli opcja `unix password sync` zostanie ustawiona na `yes`, ponieważ Samba nie zawsze dysponuje poprzednim hasłem użytkownika w postaci niezaszyfrowanej.

Nieco trudniejsza do skonfigurowania jest opcja `passwd chat`. Opcja ta działa podobnie jak uniksowy skrypt *chat*. Określa ona ciąg wysyłanych łańcuchów i odpowiedzi programu wskazanego przez opcję `passwd program`. Oto domyślna wartość opcji `passwd chat`; ogranicznikami są spacje między poszczególnymi grupami znaków:

```
passwd chat = *old*password* %o\n *new*password* %n\n *new*password* %n\n *changed*
```

Pierwsza grupa reprezentuje odpowiedź, której powinien udzielić program służący do zmieniania hasła. Zauważ, że może ona zawierać wieloznaczniki (*), które umożliwiają uogólnienie sekwencji wymiany danych w celu obsługi programów generujących podobne wyniki. Zapis `*old*password*` oznacza, że Samba będzie czekać, aż program do zmieniania hasła wygeneruje dowolną linię z literami `old`, po których następują litery `password`, bez uwzględniania żadnych znaków po lewej lub prawej stronie ani w środku. Odpowiednio poinstruowana, Samba będzie nieprzerwanie czekać na pasujący łańcuch znaków. Jeśli nie otrzyma oczekiwanej odpowiedzi, zmiana hasła się nie powiedzie.

Druga grupa informuje, co Samba powinna odesłać, kiedy otrzyma łańcuch znaków pasujący do pierwszej grupy. W tym przypadku jest to `%o\n`. Odpowiedź ta w rzeczywistości składa się z dwóch elementów: zmienna `%o` reprezentuje stare hasło, a `\n` to znak nowej linii. W rezultacie spowoduje to „wpisanie" starego hasła na standardowym wejściu programu do zmieniania hasła i „naciśnięcie" klawisza [Enter].

Trzecia grupa reprezentuje kolejną odpowiedź, a czwarta dane odsyłane programowi do zmieniania hasła (taki wzorzec odpowiedź-odesłanie danych ciągnie się nieprzerwanie w każdym standardowym skrypcie *chat*). Skrypt kończy działanie wtedy, kiedy dopasowany zostanie ostatni wzorzec*.

---

* Ta wersja nie będzie działać w Linuksie Red Hat, w którym program do zmieniania hasła zwykle odpowiada „All authentication tokens updated successfully" zamiast „Password changed". Dalej w tym podrozdziale zamieścimy odpowiednią poprawkę.

Dopasowanie łańcuchów przesyłanych przez program do zmieniania hasła ułatwiają znaki wymienione w tabeli 6.6. Do sformułowania swojej odpowiedzi możesz natomiast użyć znaków z tabeli 6.7.

Tabela 6.6. Znaki używane w definiowaniu oczekiwanych odpowiedzi

Znak	*Definicja*
*	Zero lub więcej wystąpień dowolnego znaku
" "	Pozwala na dopasowywanie łańcuchów zawierających spacje. Gwiazdki są nadal uznawane za wieloznaczniki, nawet wtedy, gdy znajdują się w cudzysłowie, a pustą odpowiedź reprezentuje pusty cudzysłów

Tabela 6.7. Znaki używane w definiowaniu odsyłanych danych

Znak	*Definicja*
%o	Stare hasło użytkownika
%n	Nowe hasło użytkownika
\n	Znak nowej linii
\r	Znak powrotu karetki
\t	Znak tabulacji
\s	Spacja

Możesz na przykład ustawić swój skrypt zmiany hasła na podany poniżej. Obsługuje on scenariusze, w których nie musisz wpisywać starego hasła. Rozpoznaje też łańcuch `all tokens updated successfully` wysyłany w Linuksie Red Hat:

```
passwd chat = *new*password* %n\n *new*password* %n\n *success*
```

Domyślna sekwencja wymiany danych powinna być wystarczająca w wielu odmianach Uniksa. Jeśli jest inaczej, podczas tworzenia nowego skryptu wymiany danych z programem do zmieniania hasła może ci się przydać opcja `passwd chat debug`. Opcja ta powoduje rejestrowanie wszystkich wymian danych w trakcie zmieniania hasła. Przyjmuje ona wartości logiczne, jak w przykładzie poniżej:

```
[global]
    encrypt passwords = yes
    smb passwd file = /usr/local/samba/private/smbpasswd

    unix password sync = yes
    passwd chat debug = yes
    log level = 100
```

Kiedy uaktywnisz diagnozowanie wymiany danych podczas zmieniania hasła, wszystkie operacje wejścia-wyjścia wykonywane wówczas przez Sambę będą zapisywane w dziennikach Samby z poziomem diagnostycznym równym 100 (dlatego ustawiliśmy nową wartość opcji `log level`). Ponieważ takie ustawienie może znacznie zwiększyć ilość wpisów w dziennikach, bardziej efektywne może okazać się napisanie własnego skryptu (i wskazanie go w opcji `passwd program` zamiast programu */bin/passwd*), który będzie rejestrował zdarzenia zachodzące podczas wymiany danych. Należy też zabezpieczyć pliki dziennika za pomocą odpowiednich

praw dostępu i usunąć je natychmiast po uzyskaniu poszukiwanych informacji, ponieważ będą one zawierać hasła pisane jawnym tekstem.

System operacyjny, w którym działa Samba, może mieć określone wymagania co do poprawności haseł, aby uodpornić je na ataki słownikowe i inne podobne. Użytkownicy zmieniający hasło powinni być informowani o takich ograniczeniach.

Wcześniej stwierdziliśmy, że synchronizacja haseł jest niepełna, ponieważ zaszyfrowane hasła w pliku *smbpasswd* nie są uaktualniane, kiedy użytkownicy zmieniają swoje standardowe hasła uniksowe. Istnieją różne metody obejścia tego ograniczenia, w tym użycie NIS oraz bezpłatnych implementacji standardu wymiennych modułów uwierzytelniających (*pluggable authentication modules*, PAM), ale żadna z nich nie rozwiązuje wszystkich problemów. Po ukazaniu się Windows 2000, Samba będzie dążyć do zgodności z Ligthweight Directory Access Protocol (LDAP), dzięki czemu kwestie synchronizacji haseł odejdą w przeszłość.

Opcje konfiguracji haseł

Opcje wymienione w tabeli 6.8 umożliwiają skonfigurowanie obsługi haseł w Sambie.

Tabela 6.8. Opcje konfiguracji haseł

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`encrypt passwords`	Wartość logiczna	Włącza obsługę zaszyfrowanych haseł	`no`	Globalny
`unix password sync`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba uaktualnia standardową uniksową bazę haseł, kiedy użytkownik zmieni swoje zaszyfrowane hasło	`no`	Globalny
`passwd chat`	Łańcuch (polecenia wymiany danych)	Ustawia sekwencję poleceń, które będą wysyłane do programu zmieniającego hasło	Patrz podrozdział poświęcony tej opcji	Globalny
`passwd chat debug`	Wartość logiczna	Wysyła komunikaty diagnostyczne dotyczące procesu zmiany hasła do plików dziennika z poziomem równym 100	`no`	Globalny
`passwd program`	Łańcuch (polecenie Uniksa)	Określa program służący do zmieniania haseł	`/bin/passwd %u`	Globalny
`password level`	Wartość liczbowa	Ustawia liczbę dużych liter permutowanych podczas dopasowywania hasła przesłanego przez klienta	Brak	Globalny
`update encrypted`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba uaktualnia plik zaszyfrowanych haseł, kiedy klient połączy się z udziałem za pomocą jawnego hasła	`no`	Globalny

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`null passwords`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba zezwala na dostęp użytkownikom mającym puste hasło	`no`	Globalny
`smb passwd file`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa nazwę pliku z zaszyfrowanymi hasłami	`/usr local/ samba/ private/ smbpasswd`	Globalny
`hosts equiv`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa nazwę pliku zawierającego nazwy hostów i użytkowników, którzy mogą się łączyć bez podawania hasła	Brak	Globalny
`use rhosts`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa nazwę pliku *.rhosts*, który pozwala użytkownikom na łączenie się bez podawania hasła	Brak	Globalny

unix password sync

Globalna opcja `unix password sync` pozwala Sambie na uaktualnianie standardowego uniksowego pliku haseł, kiedy użytkownik zmieni swoje zaszyfrowane hasło. Zaszyfrowane hasła są przechowywane w serwerze Samby w pliku *smbpasswd*, który domyślnie znajduje się w katalogu */usr/local/samba/private*. Możesz uaktywnić tę funkcję w następujący sposób:

```
[global]
    unix password sync = yes
```

Jeśli opcja ta jest włączona, Samba zmienia zaszyfrowane hasło, a oprócz tego próbuje zmienić standardowe hasło uniksowe, przekazując nazwę użytkownika i nowe hasło programowi określonemu opcją `passwd program` (opisaną wcześniej). Zauważ, że Samba nie zawsze ma dostęp do jawnego hasła użytkownika, więc program do zmieniania hasła musi zostać wywoływany z konta roota*. Jeśli z jakiekolwiek przyczyny zmiana hasła uniksowego nie powiedzie się, hasło SMB również nie ulegnie zmianie.

encrypt passwords

Globalna opcja `encrypt passwords` sprawia, że Samba uwierzytelnia klientów za pomocą haseł zaszyfrowanych, a nie pisanych jawnym tekstem. Samba będzie spodziewać się zaszyfrowanych haseł, jeśli opcja ta zostanie ustawiona na `yes`:

```
encrypt passwords = yes
```

* Dzieje się tak dlatego, że uniksowy program *passwd*, który jest najczęstszym podmiotem tej operacji, pozwala rootowi na zmianę hasła użytkownika bez wpisywania obecnego hasła tego użytkownika.

Domyślnie Windows NT 4.0 z Service Pack 3 oraz Windows 98 transmitują hasła w postaci zaszyfrowanej. Jeśli chcesz włączyć obsługę zaszyfrowanych haseł, musisz przygotować plik *smbpasswd*, wypełniając go nazwami użytkowników, którzy będą uwierzytelniani z wykorzystaniem zaszyfrowanych haseł (patrz wcześniejszy podrozdział „Plik smbpasswd"). Oprócz tego Samba musi znać położenie pliku *smbpasswd*; jeśli nie znajduje się on w domyślnym katalogu (zwykle */usr/local/samba/private/smbpasswd*), możesz jawnie podać jego położenie za pomocą opcji `smb passwd file`.

Jeśli chcesz, możesz użyć opcji `update encrypted`, aby wymusić uaktualnianie zaszyfrowanych haseł w pliku *smbpasswd* za każdym razem, kiedy klient połączy się za pomocą niezaszyfrowanego hasła.

Aby zapewnić uwierzytelnianie z wykorzystaniem zaszyfrowanych haseł wszystkim hostom, które tego wymagają, można skorzystać z opcji `include`. Dzięki niej możesz utworzyć odrębne pliki konfiguracyjne, które będą wczytywane na podstawie typu systemu operacyjnego (`%a`) lub nazwy klienta (`%m`). Takie pliki konfiguracyjne zależne od hosta lub systemu operacyjnego mogą zawierać opcję `encrypted passwords = yes`, która będzie uaktywniana tylko podczas łączenia się tych klientów z serwerem.

passwd program

Opcja `passwd program` służy do określenia programu w uniksowym serwerze Samby, który będzie uaktualniał standardowy systemowy plik haseł, kiedy Samba uaktualnia plik haseł zaszyfrowanych. Opcja ta domyślnie wskazuje na standardowy program *passwd*, zwykle położony w katalogu */bin*. Zwykle używa się tu zmiennej `%u`, oznaczającej użytkownika zmieniającego hasło. Obsługa wejścia i wyjścia programu jest sterowana przez opcję `passwd chat`. Opcję tę opisano szczegółowo wcześniej w podrozdziale „Synchronizowanie haseł".

passwd chat

Ta opcja określa serię łańcuchów: wysyłane dane-odpowiedź, przypominającą uniksowy skrypt *chat* i używaną do współpracy z programem zmieniającym hasła w serwerze Samby. Opcję tę opisano szczegółowo w podrozdziale „Synchronizowanie haseł".

passwd chat debug

Jeśli opcja `passwd chat debug` jest ustawiona na `yes`, podczas wymiany danych przy zmianie hasła wszystkie informacje wysłane lub otrzymane przez Sambę są zapisywane w pliku dziennika. Ponieważ poziom diagnostyczny tych informacji jest równy 100, będziesz musiał użyć opcji `log level = 100`, aby zostały one zarejestrowane. Opcję tę opisano szczegółowo w podrozdziale „Synchronizowanie haseł". Pamiętaj, że jeśli ustawisz tę opcję, hasła pisane jawnym tekstem będą widoczne w plikach dziennika, co może stanowić zagrożenie bezpieczeństwa, jeśli odpowiednio nie zabezpieczysz tych plików.

password level

W protokole SMB hasła niezaszyfrowane (pisane jawnym tekstem) są wysyłane dużymi literami, podobnie jak nazwy użytkowników. Wielu użytkowników Uniksa wybiera jednak hasła składające się zarówno z małych, jak i dużych liter. Samba domyślnie próbuje dopasować hasła pisane tylko małymi literami, nie zamieniając pierwszej litery na dużą.

Podobnie jak `username level`, opcja `password level` służy do wypróbowania różnych permutacji hasła z dużymi literami. Przyjmuje ona parametr w postaci liczby całkowitej, który określa, ile liter hasła należy zamienić na duże podczas próby nawiązania połączenia z udziałem. Możesz użyć jej w następujący sposób:

```
[global]
    password level = 3
```

W tym przykładzie Samba wypróbuje wszystkie permutacje hasła, które można uzyskać przy użyciu trzech dużych liter. Im większa liczba, tym więcej obliczeń, które Samba będzie musiała wykonać podczas dopasowywania hasła i tym dłuższy czas nawiązywania połączenia z udziałem.

update encrypted

Dla sieci przechodzących na zaszyfrowane hasła Samba oferuje opcję, która powinna pomóc w tym procesie. Opcję `update encrypted` możesz uaktywnić w następujący sposób:

```
[global]
    update encrypted = yes
```

Opcja ta informuje Sambę, że należy utworzyć zaszyfrowaną wersję uniksowego hasła każdego użytkownika, kiedy ten łączy się z udziałem. Jeśli włączysz tę opcję, musisz ustawić opcję `encrypt passwords` na `no`, aby klienty mogły przekazywać Sambie jawne hasła służące do uaktualnienia plików haseł. Kiedy każdy użytkownik co najmniej raz połączy się z serwerem, możesz ustawić opcję `encrypt passwords = yes` i od tej chwili używać tylko zaszyfrowanych haseł. Aby opcja ta zadziałała, użytkownik musi mieć już poprawny wpis w pliku *smbpasswd*.

null passwords

Ta globalna opcja informuje Sambę, czy należy zezwolić na dostęp użytkownikom, których konta mają ustawione puste hasło (zaszyfrowane bądź nie). Jej domyślna wartość to `no`. Możesz zmienić ją następująco:

```
null passwords = yes
```

Stanowczo odradzamy włączanie tej opcji, chyba że zdajesz sobie sprawę z zagrożeń, które stwarza. Na przykład umożliwia ona dostęp do kont systemowych (takich jak *bin*), które mają ustawione puste hasła w systemowym pliku haseł.

smb passwd file

Ta globalna opcja określa położenie bazy danych z zaszyfrowanymi hasłami. Domyślnie jest to */usr/local/samba/private/smbpasswd*. Możesz zmienić ją w następujący sposób:

```
[global]
    smb passwd file = /etc/smbpasswd
```

Takie położenie jest charakterystyczne dla wielu dystrybucji Linuksa Red Hat.

hosts equiv

Ta globalna opcja określa nazwę standardowego uniksowego pliku *hosts.equiv*, który pozwala hostom i użytkownikom na dostęp do udziału bez podawania hasła. Możesz określić położenie tego pliku w następujący sposób:

```
[global]
    hosts equiv = /etc/hosts.equiv
```

Opcja ta domyślnie nie określa żadnego pliku *hosts.equiv*. Ponieważ korzystanie z takiego pliku wiąże się z ogromnym zagrożeniem bezpieczeństwa, odradzamy stosowanie tej opcji, o ile nie jesteś absolutnie pewien, że twoja sieć jest zabezpieczona.

use rhosts

Ta globalna opcja określa nazwę standardowego uniksowego pliku *.rhosts*, który pozwala zdalnym hostom na dostęp do udziału bez podawania hasła. Możesz określić położenie tego pliku w następujący sposób:

```
[global]
    use rhosts = /home/dawid/.rhosts
```

Opcja ta domyślnie nie określa żadnego pliku *.rhosts*. Podobnie jak w przypadku omówionej wyżej opcji `hosts equiv`, korzystanie z takiego pliku zagraża bezpieczeństwu sieci. Odradzamy użycie tej opcji, o ile nie jesteś absolutnie pewien, że twoja sieć jest dobrze zabezpieczona.

Domeny Windows

Kiedy wiesz już wszystko o użytkownikach i hasłach w serwerze Samby, możemy pokazać ci, jak skonfigurować Sambę jako podstawowy kontroler domeny dla komputerów Windows 95/98 i NT. Do czego przydają się domeny? Odpowiedź prawdopodobnie nie będzie oczywista, dopóki nie zajrzysz za kulisy, zwłaszcza w przypadku Windows 95/98.

Przypomnij sobie, że w tradycyjnych grupach roboczych Windows 95/98 po prostu akceptuje wszystkie wpisywane podczas logowania nazwy użytkownika i hasła. W Windows 95/98 nie ma nieautoryzowanych użytkowników; kiedy loguje się nowy użytkownik, system operacyjny prosi go o wprowadzenie nowego hasła i od tego momentu uwierzytelnia go za pomocą tego hasła. Windows 95/98 używa wprowadzonego hasła tylko wtedy, gdy próbujesz połączyć się z udziałem.

Logowania domenowe przypominają natomiast uniksowe mechanizmy bezpieczeństwa. Aby zalogować się w domenie, musisz podać poprawną nazwę użytkow-

nika i hasło, które są następnie weryfikowane w bazie haseł podstawowego kontrolera domeny. Jeśli hasło jest błędne, użytkownik jest o tym natychmiast informowany i nie może zalogować się w domenie.

Oto dodatkowe zalety takiego rozwiązania: kiedy pomyślnie zalogujesz się w domenie, będziesz miał dostęp do wszystkich udziałów domeny, do których ci go przyznano, bez potrzeby ponownego uwierzytelniania się. Mówiąc ściślej, podstawowy kontroler domeny przyznaje klientowi żeton, który upoważnia do korzystania ze wszystkich udziałów bez ponownego kontaktowania się z kontrolerem. Choć prawdopodobnie nie zauważysz żadnej zmiany, może to dobroczynnie wpłynąć na ruch w sieci (jeśli zechcesz, możesz wyłączyć tę funkcję za pomocą opcji `revalidate`).

Konfigurowanie Samby do obsługi logowania w domenach Windows

Jeśli chcesz, aby Samba działała jako podstawowy kontroler domeny, użyj poniższych sekcji w celu skonfigurowania Samby i jej klientów do obsługi logowań domenowych.

Jeśli chciałbyś dowiedzieć się więcej o konfigurowaniu domen, przeczytaj plik *DOMAINS.TXT* z dystrybucji Samby.

Klienty Windows 95/98

Konfigurowanie Samby jako podstawowego kontrolera domeny dla klientów Windows 95/98 rozczarowuje swą prostotą. Po stronie serwera wystarczy, aby spełnione były następujące warunki:

- Samba jest jedynym podstawowym kontrolerem domeny w bieżącej grupie roboczej.
- W sieci dostępny jest serwer WINS – albo serwer Samby, albo serwer Windows NT (więcej informacji o WINS znajdziesz w rozdziale 7, *Drukowanie i odwzorowywanie nazw*).
- Samba korzysta z zabezpieczeń na poziomie użytkownika (to znaczy nie ceduje uwierzytelniania haseł na inny komputer). Nie powinieneś używać zabezpieczeń na poziomie domeny, jeśli sama Samba działa jako PDC.

Teraz powinieneś dopisać następujące opcje do pliku konfiguracyjnego Samby:

```
[global]
    workgroup = PROSTA_GRUPA
    domain logons = yes

# Koniecznie ustaw zabezpieczenia na poziomie u□ytkownika!

    security = user

# Samba musi zosta□ podstawowym kontrolerem domeny!

    os level = 34
    local master = yes
    preferred master = yes
    domain master = yes
```

Opcja `domain logons` umożliwia Sambie przeprowadzanie domenowego uwierzytelniania w imieniu tych klientów, które tego zażądają. Nazwa domeny będzie taka sama, jak nazwa grupy wymienionej w pliku konfiguracyjnym Samby, w tym przypadku: `PROSTA_GRUPA`.

Następnie musisz utworzyć niezapisywalny, niepubliczny, nieprzeglądalny udział dyskowy o nazwie `[netlogon]` (nieważne, na który katalog będzie wskazywał ten udział, jeśli tylko będą mogły się z nim połączyć wszystkie klienty Windows):

```
[netlogon]
    comment = Us□uga logowania domenowego
    path = /export/samba/logon
    public = no
    writeable = no
    browsable = no
```

Klienty Windows NT

Jeśli w twojej sieci są klienty Windows NT, musisz wykonać kilka dodatkowych czynności, aby Samba mogła zostać ich podstawowym kontrolerem domeny.

Aby korzystać z funkcji PDC dla klientów Windows NT, będziesz musiał użyć Samby w wersji 2.1 lub nowszej. Wcześniejsze wersje umożliwiały tylko ograniczone uwierzytelnianie klientów NT. Kiedy oddawaliśmy tę książkę do druku, najnowszą wersją Samby była wersja 2.0.5, ale wersję 2.1 można było pobrać za pomocą CVS. Instrukcje dotyczące pobierania wersji alfa Samby znajdziesz w dodatku E, *Pobieranie Samby za pomocą systemu CVS*.

Jak poprzednio, musisz upewnić się, że Samba jest podstawowym kontrolerem domeny w bieżącej grupie roboczej i korzysta z zabezpieczeń na poziomie użytkownika. Musisz jednak także włączyć obsługę zaszyfrowanych haseł. Innymi słowy, zmień sekcję `[global]` z poprzedniego przykładu tak, aby zawierała opcję `encrypted passwords = yes`, jak pokazano niżej:

```
[global]
    workgroup = PROSTA_GRUPA
    encrypted passwords = yes
    domain logons = yes

    security = user
```

Tworzenie kont zaufania dla klientów NT

Ten etap obowiązuje tylko klienty Windows NT. Wszystkie klienty NT łączące się z podstawowym kontrolerem domeny korzystają z *kont zaufania* (*trust accounts*). Konta ta pozwalają komputerowi na logowanie się w samym kontrolerze domeny (a nie w jednym z jego udziałów), co oznacza, że PDC może zaufać wszystkim przyszłym połączeniom nawiązywanym przez użytkowników tego klienta. Praktycznie rzecz biorąc, konto zaufania jest równoważne kontu użytkownika. W rzeczywistości będziemy używać standardowych uniksowych kont użytkownika do emulowania kont zaufania w serwerze Samby.

Nazwa używana przy logowaniu się na koncie zaufania to nazwa komputera z dołączonym znakiem dolara. Jeśli na przykład nasz komputer Windows NT nosi nazwę `chimera`, wówczas konto zaufania będzie miało nazwę `chimera$`. Początkowo hasłem konta jest po prostu nazwa komputera pisana małymi literami. Aby zasymulować konto zaufania w serwerze Samby, musisz utworzyć konto użytkownika o nazwie odpowiadającej nazwie komputera oraz dodać zaszyfrowane hasło do pliku *smbpasswd*.

Uporajmy się z pierwszym zadaniem. Musimy tylko zmodyfikować plik */etc/passwd* w sposób pozwalający na obsługę konta zaufania; nie musimy tworzyć katalogu macierzystego ani wyznaczać powłoki użytkownika, ponieważ zależy nam tylko na umożliwieniu logowania. Możemy więc utworzyć „fikcyjne" konto za pomocą następującego wpisu:

```
chimera$:*:1000:900:Konto zaufania:/dev/null:/dev/null
```

Zauważ, że wyłączyliśmy hasło, umieszczając w jego polu gwiazdkę. Zrobiliśmy tak dlatego, że właściwe hasło będzie przechowywane w pliku *smbpasswd*, a nie chcemy, aby ktoś zalogował się na tym koncie za pośrednictwem telnetu. W rzeczywistości, jedyną wartością użytą tutaj oprócz nazwy konta jest identyfikator użytkownika na potrzeby bazy zaszyfrowanych haseł (1000). Liczba ta musi odpowiadać niepowtarzalnemu identyfikatorowi zasobu w serwerze NT i nie może kolidować z innym identyfikatorem zasobu. Zatem żaden inny użytkownik ani grupa NT nie powinna odpowiadać tej liczbie, gdyż spowodowałoby to wystąpienie błędu sieciowego.

Następnie dodaj zaszyfrowane hasło za pomocą polecenia *smbpasswd*, jak niżej:

```
# smbpasswd -a -m chimera
Added user chimera$
Password changed for user chimera$
```

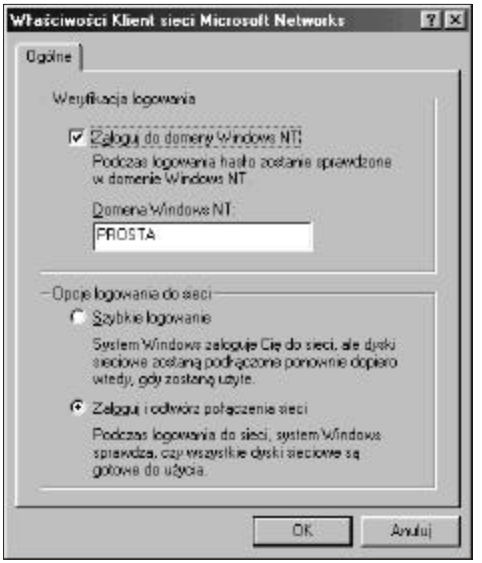
Opcja `-m` informuje, że generowane jest komputerowe konto zaufania. Program *smbpasswd* automatycznie ustawi początkowe zaszyfrowane hasło na nazwę komputera pisaną małymi literami – nie musisz go wpisywać. Kiedy używasz tej opcji w linii polecenia, nie wpisuj znaku dolara za nazwą komputera – zostanie on dodany automatycznie. Po dodaniu zaszyfrowanego hasła Samba będzie mogła obsługiwać logowania domenowe klientów NT.

Konfigurowanie logowania domenowego w klientach Windows

Kiedy Samba jest już skonfigurowana do obsługi logowania domenowego, musisz ustawić klienty Windows tak, aby po uruchomieniu logowały się w domenie.

Windows 95/98

W Windows 95/98 można to zrobić następująco: otwórz okno dialogowe Sieć z Panelu sterowania, zaznacz pozycję Klient sieci Microsoft Networks i kliknij przycisk Właściwości. Powinno ukazać się okno dialogowe podobne do tego z rysunku 6.4. Zaznacz pole wyboru Zaloguj do domeny Windows NT na górze okna dialogowego i wpisz nazwę grupy roboczej, która jest wymieniona w pliku konfiguracyjnym Samby jako domena Windows NT. Następnie kliknij przycisk OK i ponownie uruchom komputer.



Rysunek 6.4. Konfigurowanie logowania domenowego w kliencie Windows 95/98

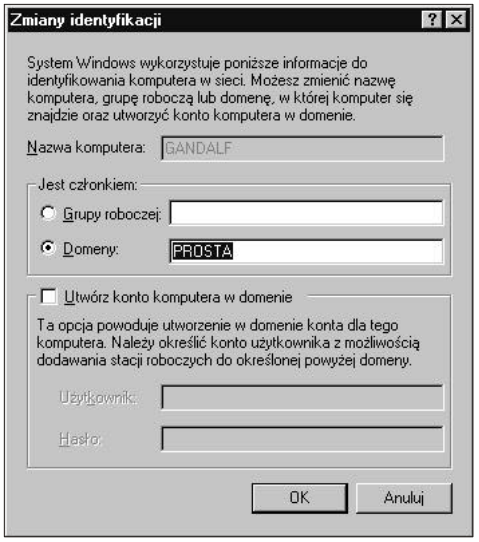
Jeśli Windows poskarży się, że jesteś już zalogowany w domenie, prawdopodobnie masz aktywne połączenie z udziałem w grupie roboczej (na przykład mapowanym dyskiem sieciowym). Po prostu odłącz tymczasowo zasób, klikając jego ikonę prawym przyciskiem myszy i wybierając z menu polecenie Odłącz.

Kiedy Windows ponownie się uruchomi, powinieneś zobaczyć standardowe okno logowania z dodatkowym polem na nazwę domeny. Nazwa ta powinna być już wpisana, więc wystarczy wprowadzić hasło i kliknąć przycisk OK. W tym momencie Windows skontaktuje się z podstawowym kontrolerem domeny (Sambą), aby sprawdzić, czy hasło jest poprawne (możesz zajrzeć do plików dziennika, aby sprawdzić przebieg tego procesu). Jeśli wszystko działa, gratulujemy! Poprawnie skonfigurowałeś Sambę jako podstawowy kontroler domeny dla komputerów Windows 95/98, a twój klient pomyślnie nawiązał połączenie.

Windows NT 4.0

Aby skonfigurować logowanie domenowe w Windows NT, otwórz okno dialogowe Sieć w Panelu sterowania Windows NT. Pierwsza wyświetlona karta umożliwia zmianę identyfikacji komputera.

Kliknij przycisk Zmień, a ukaże się okno dialogowe z rysunku 6.5. W tym oknie możesz skonfigurować klienta Windows NT jako członka domeny, zaznaczając opcję Domeny w ramce Jest członkiem. Następnie wpisz nazwę domeny, w której powinien logować się klient; powinna to być nazwa grupy roboczej, która jest wymieniona w pliku konfiguracyjnym Samby. Nie zaznaczaj pola wyboru Utwórz konto komputera w domenie – Samba nie obsługuje jeszcze tej funkcji.



Rysunek 6.5. Konfigurowanie logowania domenowego w kliencie Windows NT

Podobnie jak w Windows 95/98, jeśli Windows NT poskarży się, że jesteś już zalogowany w domenie, prawdopodobnie masz aktywne połączenie z udziałem w grupie roboczej (na przykład mapowanym dyskiem sieciowym). Odłącz tymczasowo zasób, klikając jego ikonę prawym przyciskiem myszy i wybierając z menu polecenie Odłącz.

Kiedy klikniesz przycisk OK, Windows powinno wyświetlić niewielkie okno dialogowe witające cię w domenie. W tym momencie będziesz musiał ponownie uruchomić komputer, który następnie powinien wyświetlić ekran logowania podobny do tego w klientach Windows 95/98. Możesz teraz zalogować się na dowolne istniejące już konto w serwerze Samby, które jest skonfigurowane tak, że akceptuje logowanie.

Sprawdź, czy wybrałeś właściwą domenę w oknie logowania Windows NT. Zbudowanie listy dostępnych domen może chwilę potrwać.

Kiedy już wprowadzisz hasło, Windows NT skontaktuje się z podstawowym kontrolerem domeny (Sambą), aby sprawdzić, czy hasło jest poprawne. Możesz też zajrzeć do pliku dziennika, aby sprawdzić przebieg tego procesu. Jeśli wszystko działa, jest to znak, że poprawnie skonfigurowałeś Sambę jako podstawowy kontroler domeny dla komputerów Windows NT.

Opcje domenowe

W tabeli 6.9 zebrano opcje używane do konfigurowania logowania domenowego.

Tabela 6.9. Opcje logowania domenowego Windows 95/98

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`domain logons`	Wartość logiczna	Informuje, czy będzie używane logowanie domenowe Windows	`no`	Globalny
`domain group map`	Łańcuch (nazwa pliku wraz ze ścieżką)	Nazwa pliku z odwzorowaniami grup uniksowych na grupy domenowe Windows NT	Brak	Globalny
`domain user map`	Łańcuch (nazwa pliku wraz ze ścieżką)	Nazwa pliku z odwzorowaniami użytkowników uniksowych na użytkowników domeny Windows NT	Brak	Globalny
`local group map`	Łańcuch (nazwa pliku wraz ze ścieżką)	Nazwa pliku z odwzorowaniami grup uniksowych na grupy lokalne Windows NT	Brak	Globalny
`revalidate`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba wymusza uwierzytelnianie użytkowników przy każdym łączeniu się z udziałem	`no`	Udział

domain logons

Ta opcja sprawia, że Samba akceptuje logowania domenowe jako podstawowy kontroler domeny. Kiedy klient pomyślnie zaloguje się w domenie, Samba przesyła mu specjalny żeton, który pozwala na dostęp do udziałów w domenie bez ponownego uwierzytelniania przez PDC. Pamiętaj, że Samba musi używać zabezpieczeń na poziomie użytkownika (`security = user`) i być podstawowym kontrolerem domeny, aby opcja ta zadziałała. Oprócz tego, komputery Windows będą spodziewać się, że w serwerze Samby istnieje udział `[netlogon]` (patrz wcześniejszy podrozdział „Konfigurowanie Samby do obsługi logowania w domenach Windows").

domain group map

Ta opcja określa położenie pliku odwzorowań, służącego do tłumaczenia nazw grup domenowych Windows NT na nazwy grup uniksowych. Plik ten powinien znajdować się w serwerze Samby, na przykład:

```
/usr/local/samba/private/grupy.domenowe
```

Plik ten ma prosty format:

```
GrupaUniksowa = GrupaNT
```

Oto przykładowy wpis:

```
admin = Administratorzy
```

Podana grupa uniksowa powinna być wymieniona w pliku */etc/group*. Grupa NT to nazwa, na którą powinna zostać odwzorowana grupa uniksowa w kliencie NT. Opcja ta działa tylko z klientami Windows NT.

domain user map

Ta opcja określa położenie pliku odwzorowań, służącego do tłumaczenia nazw użytkowników uniksowych na nazwy użytkowników domen Windows NT. Plik ten powinien znajdować się w serwerze Samby, na przykład:

```
/usr/local/samba/private/uzytkownicy.domeny
```

Plik ten ma prosty format:

```
UniksowaNazwaU□ytkownika = [\\Domena\\]NazwaU□ytkownikaNT
```

Oto przykładowy wpis:

```
rysiek = Ryszard Kwiatkowski
```

Podany użytkownik uniksowy powinien być wymieniony w pliku */etc/passwd*. Użytkownik NT to nazwa, na którą powinien zostać odwzorowany użytkownik uniksowy w kliencie NT. Opcja ta działa tylko z klientami Windows NT.

Jeśli chciałbyś dowiedzieć się więcej o użyciu domenowych nazw użytkowników i grup lokalnych w Windows NT, polecamy książkę Erica Pearce'a *Windows NT in a Nutshell*, opublikowaną przez wydawnictwo O'Reilly.

local group map

Ta opcja określa położenie pliku odwzorowań, służącego do tłumaczenia nazw grup lokalnych Windows NT na nazwy grup uniksowych; grupy lokalne to między innymi Administratorzy i Użytkownicy. Plik ten powinien znajdować się w serwerze Samby, na przykład:

```
/usr/local/samba/private/grupy.lokalne
```

Plik ten ma prosty format:

```
GrupaUniksowa = [BUILTIN\]GrupaNT
```

Oto przykładowy wpis:

```
root = BUILTIN\Administratorzy
```

Opcja ta działa tylko z klientami Windows NT. Więcej informacji znajdziesz we wspomnianej książce Erica Pearce'a *Windows NT in a Nutshell* (wydawnictwo O'Reilly).

revalidate

Ta opcja udziału informuje Sambę, że należy wymusić uwierzytelnianie użytkowników przy każdym łączeniu się z innym udziałem w komputerze, niezależnie od poziomu bezpieczeństwa używanego przez serwer Samby. Jej domyślna wartość to `no`, co oznacza, że Samba ufa użytkownikom po ich pomyślnym uwierzytelnieniu. Możesz zmienić ją w następujący sposób:

```
revalidate = yes
```

Opcja ta może poprawić bezpieczeństwo systemu. Pamiętaj jednak o niewygodzie, która wiąże się z wpisywaniem hasła przy dostępie do każdego udziału.

Skrypty logowania

Samba umożliwia wykonywanie skryptów logowania Windows, czyli plików wsadowych (.BAT lub .CMD) uruchamianych w kliencie podczas logowania się w domenie Windows. Zauważ, że skrypty te są przechowywane w serwerze uniksowym, a następnie przesyłane przez sieć do klienta i wykonywane wtedy, gdy użytkownik się zaloguje. Skrypty logowania są niezwykle przydatnym narzędziem do dynamicznego konfigurowania parametrów sieciowych dla logujących się użytkowników. Ponieważ jednak wykonują się w systemie Windows, muszą zawierać polecenia konfiguracji sieci używane w Windows.

Jeśli chciałbyś dowiedzieć się więcej o grupie poleceń NET, polecamy następujące podręczniki wydawnictwa O'Reilly: *Windows NT in a Nutshell*, *Windows 95 in a Nutshell* i *Windows 98 in a Nutshell*.

Możesz poinformować Sambę o chęci korzystania ze skryptów logowania za pomocą opcji `logon script`, jak w poniższym przykładzie:

```
[global]
    domain logons = yes
    security = user
    workgroup = PROSTA_GRUPA

    os level = 34
    local master = yes
    preferred master = yes
    domain master = yes
    logon script = %U.bat

[netlogon]
    comment = Us□uga logowania domenowego
    path = /export/samba/logon
    public = no
    writeable = no
    browsable = no
```

Zwróć uwagę, że w tym przykładzie użyliśmy zmiennej `%U`, która pozwala zindywidualizować skrypt w zależności od logującego się użytkownika. Skrypty logowania są często dostosowywane do komputera lub użytkownika logującego się w domenie. Dzięki temu mogą ustawiać indywidualne parametry dla użytkowników lub klientów.

Wszystkie skrypty logowania powinny być przechowywane w głównym katalogu udziału `[netlogon]`. Jeśli na przykład katalogiem udziału `[netlogon]` jest */export/samba/logon*, a skrypt logowania nosi nazwę *jacek.bat*, to pełną ścieżką do skryptu będzie */export/samba/logon/jacek.bat*. Kiedy użytkownik zaloguje się w domenie, która zawiera skrypt startowy, zobaczy małe okno dialogowe informujące o działaniu skryptu oraz okno podobne do trybu MS-DOS z wynikami pracy skryptu.

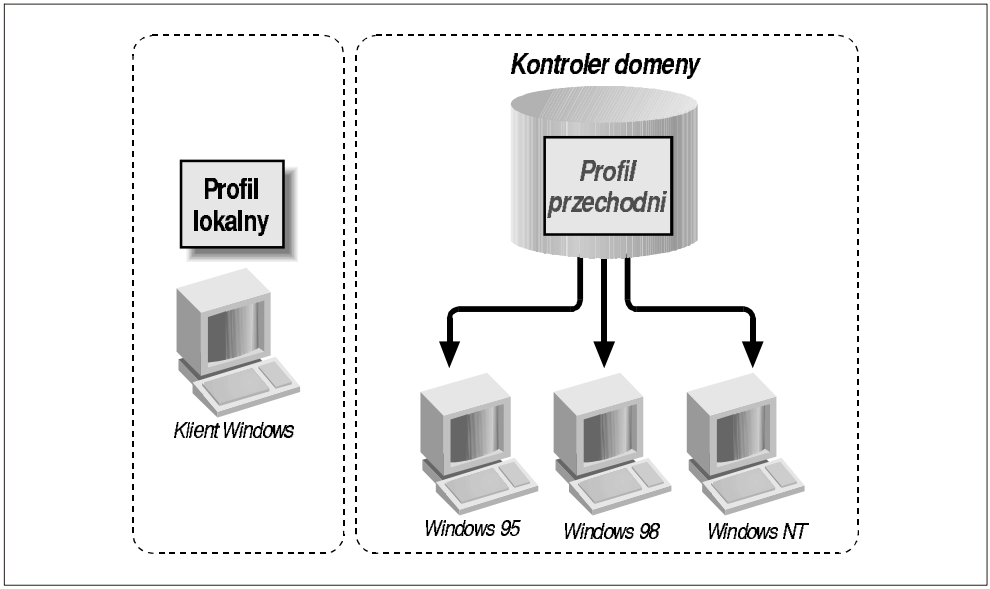
Słowo ostrzeżenia: ponieważ skrypty logowania są pobierane przez Windows i tam też wykonywane, muszą być sformatowane na sposób dosowy, czyli zawierać kombinację znaków powrotu karetki i nowej linii zamiast samych nowych linii charakterystycznych dla Uniksa. Najwygodniej będzie utworzyć je w edytorze DOS-a lub Windows.

Oto przykład skryptu logowania, który synchronizuje lokalny czas z czasem serwera Samby i mapuje dwie stacje sieciowe, `h` oraz `i`, na udziały serwera:

```
# Synchronizujemy czas lokalny z czasem serwera.
# Aby polecenie to zadzia□a□o, w pliku smb.conf
# musi znajdowa□ si□ opcja "time server = yes"

echo Ustawiam aktualny czas...
net time \\hydra /set /yes

# Mapujemy stacje sieciowe na udzia□y serwera Samby

echo Mapuj□ stacje sieciowe na udzia□y serwera Hydra...
net use h: \\hydra\dane
net use i: \\hydra\siec
```

Profile przechodnie

W Windows 95 i NT każdy użytkownik może mieć swój własny *profil*. W profilu gromadzone są informacje takie jak: wygląd pulpitu użytkownika, aplikacje widoczne w menu Start, tło i inne. Jeśli profil jest przechowywany na lokalnym dysku, nazywamy go *profilem lokalnym*, ponieważ opisuje on środowisko pracy użytkownika w jednym komputerze. Jeśli jednak profil jest przechowywany w serwerze, użytkownik może pobrać go do każdego komputera, który jest połączony z serwerem. Taki profil nazywamy *przechodnim* (*roaming profile*), ponieważ użytkownik może korzystać z tego samego profilu podczas pracy na różnych komputerach. Jest to szczególnie przydatne w sytuacji, gdy ktoś jednego dnia pracuje przy własnym biurku, a następnego loguje się z komputera przenośnego spoza miejsca pracy. Rysunek 6.6 ilustruje profile lokalne i przechodnie.



Rysunek 6.6. Profile lokalne i przechodnie

Samba będzie udostępniać profile przechodnie, jeśli zostanie skonfigurowana do obsługi logowania domenowego, a ty wskażesz jej za pomocą opcji `logon path` drzewo katalogów używane do ich przechowywania. Opcja ta jest zwykle używana w połączeniu z jedną ze zmiennych określających użytkownika, jak w poniższym przykładzie:

```
[global]
    domain logons = yes
    security = user
    workgroup = PROSTA_GRUPA

    os level = 34
    local master = yes
    preferred master = yes
    domain master = yes

    logon path = \\hydra\profile\%U
```

Do obsługi profili będziemy musieli utworzyć nowy udział, który będzie prostym udziałem dyskowym dostępnym tylko dla użytkownika procesu Samby (roota). Udział ten musi być zapisywalny, ale nie może być przeglądalny. Oprócz tego musimy utworzyć katalog dla każdego logującego się użytkownika (w miejscu, które wskazuje opcja `logon path` z przykładu powyżej), dostępny tylko dla tego użytkownika. Aby zwiększyć bezpieczeństwo, użyjemy opcji `directory mode` i `create mode`, które uniemożliwią użytkownikom łączącym się z tym udziałem przeglądanie lub zmienianie tworzonych w nim plików.

```
[profile]
    comment = Profile u□ytkowników
    path = /export/samba/profile
    create mode = 0600
    directory mode = 0700
    writable = yes
    browsable = no
```

Kiedy użytkownik zaloguje się po raz pierwszy, klient Windows utworzy plik *user.dat* lub *ntuser.dat* – w zależności od systemu operacyjnego klienta. Klient następnie zapisze w odpowiednim katalogu udziału zawartość pulpitu, menu Start, Otoczenia sieciowego oraz folderów programów. Przy ponownym logowaniu informacje te zostaną pobrane z serwera i wykorzystane do skonfigurowania komputera, z którego zalogował się użytkownik. Kiedy użytkownik wyloguje się, informacje zostaną ponownie zapisane na serwerze aż do czasu następnego logowania. Jeśli spojrzysz na listing katalogu z treścią profilu, zobaczysz następujące pliki:

```
# ls -al

total 321
drwxrwxr-x  9 root   prosta     Jul 21 20:44 .
drwxrwxr-x  4 root   prosta     Jul 22 14:32 ..
drwxrwx---  3 jacek  program    Jul 12 07:15 Dane aplikacji
drwxrwx---  3 jacek  program    Jul 12 07:15 Menu Start
drwxrwx---  2 jacek  program    Jul 12 07:15 cookies
drwxrwx---  2 jacek  program    Jul 12 07:15 pulpit
drwxrwx---  7 jacek  program    Jul 12 07:15 historia
drwxrwx---  2 jacek  program    Jul 12 07:15 nethood
drwxrwx---  2 jacek  program    Jul 19 21:05 recent
-rw-------  1 jacek  program    Jul 21 21:59 user.dat
```

Pliki *user.dat* to binarne pliki konfiguracyjne, tworzone automatycznie przez Windows. Można je edytować za pomocą edytora profili w kliencie Windows, ale zapewnienie ich poprawności czasem przysparza kłopotów. Samba obsługuje je poprawnie aż do wersji NT 5.0 beta systemu Windows, ale funkcja ta jest względnie nowa.

Wskazówki i dokumenty HOWTO dotyczące obsługi skryptów logowania można znaleźć w dokumentacji Samby, w plikach *docs/textdocs/DOMAIN.txt* oraz *docs/textdocs/PROFILES.txt*.

Profile obowiązkowe

Użytkownicy mogą mieć także *profile obowiązkowe*, czyli profile przechodnie, których nie mogą zmienić. Jeśli użytkownik korzystający z profilu obowiązkowego doda polecenie do menu Start we wtorek, nie znajdzie go tam, kiedy zaloguje się ponownie w środę. Profil obowiązkowy to po prostu plik *user.dat*, którego nazwa została zmieniona na *user.man* i który jest przeznaczony tylko do odczytu. Zwykle zawiera on ustawienia, które administrator uznał za obowiązujące dla wszystkich użytkowników. Jeśli administrator chce utworzyć stałą konfigurację dla użytkowników, powinien wykonać następujące czynności:

1. Utworzyć w serwerze Samby katalog przeznaczony do odczytu i zapisu.
2. Ustawić opcję `logon path` w pliku *smb.conf* tak, aby wskazywała ten katalog.
3. Zalogować się jako użytkownik z Windows 95/98, aby klient zapisał pliki w tym katalogu.
4. Zmienić nazwę pliku *user.dat* na *user.man*.
5. Ustawić katalog wraz z zawartością jako przeznaczony tylko do odczytu.

Profili obowiązkowych używa się dość rzadko, natomiast profile przechodnie są jedną z najprzydatniejszych cech Windows, które Samba potrafi obsługiwać.

Opcje skryptów logowania

Tabela 6.10 podsumowuje opcje używane do konfigurowania skryptów logowania w domenach Windows.

Tabela 6.10. Opcje skryptów logowania

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`logon script`	Łańcuch (ścieżka dosowa)	Nazwa pliku wsadowego DOS-a/NT	Brak	Globalny
`logon path`	Łańcuch (nazwa UNC serwera i udziału)	Położenie przechodniego profilu użytkownika	`\\%N\%U\profile`	Globalny

Dokończenie tabeli na str. 186

Dokończenie tabeli ze str. 185

Tabela 6.10. Opcje skryptów logowania

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`logon drive`	Łańcuch (litera dysku)	Określa stację dysków z katalogiem macierzystym (tylko NT)	`Z:`	Globalny
`logon home`	Łańcuch (nazwa UNC serwera i udziału)	Określa położenie macierzystych katalogów klientów logujących się w domenie	`\\%N\%U`	Globalny

logon script

Ta opcja określa położenie pliku wykonywalnego Windows o rozszerzeniu .BAT lub .CMD (z liniami zakończonymi sekwencjami znaków powrót karetki – nowa linia), który zostanie uruchomiony w kliencie po zalogowaniu się użytkownika w domenie. Każdy skrypt logowania powinien być przechowywany w głównym katalogu udziału o nazwie `[netlogon]` (szczegóły w podrozdziale „Konfigurowanie Samby do obsługi logowania w domenach Windows"). W opcji tej często używa się zmiennych `%U` lub `%m` (nazwa użytkownika lub NetBIOS-owa nazwa komputera), które wskazują na konkretny skrypt. Na przykład opcja:

```
logon script = %U.bat
```

spowoduje wykonanie skryptu przechowywanego w głównym katalogu udziału `[netlogon]`, o nazwie odpowiadającej nazwie użytkownika. Jeśli łączący się użytkownik to `franek`, a ścieżka udziału `[netlogon]` odpowiada katalogowi */export/samba/netlogon*, klient wykona skrypt */export/samba/netlogon/franek.bat*. Ponieważ skrypty te są pobierane przez klienta i wykonywane po stronie Windows, muszą zawierać dosowe sekwencje znaków powrót karetki – nowa linia zamiast, (jak w Uniksie), samych znaków nowej linii.

logon path

Opcja ta określa położenie profili przechodnich. Kiedy użytkownik się loguje, profil przechodni jest przekazywany z serwera do klienta i uaktywniany dla tego użytkownika. Kiedy użytkownik się wylogowuje, zawartość profilu jest składowana z powrotem na serwerze aż do następnego zalogowania.

Często bezpieczniej jest utworzyć osobny udział przeznaczony specjalnie na profil użytkownika:

```
logon path = \\hydra\profile\%U
```

Więcej informacji o tej opcji znajdziesz we wcześniejszym podrozdziale „Skrypty logowania".

logon drive

Opcja ta określa literę dysku w kliencie NT, na którą będzie mapowany katalog macierzysty podany w opcji `logon home`. Opcja ta działa tylko z klientami Windows NT. Na przykład:

```
logon home = I:
```

Powinieneś zawsze używać takich liter dysku, które nie wejdą w konflikt z literami stałych dysków klienta. Domyślna litera to `Z:`, która jest położona najdalej od liter `A:`, `C:` i `D:`.

logon home

Ta opcja określa położenie katalogu macierzystego użytkownika na potrzeby dosowych poleceń NET. Aby na przykład określić katalog macierzysty jako udział w serwerze Samby, należy napisać:

```
logon home = \\hydra\%U
```

Warto podkreślić, że opcja ta dobrze działa z usługą `[homes]`, chociaż można podać dowolny katalog. Katalogi macierzyste można mapować w skrypcie logowania za pomocą następującego polecenia:

```
NET USE I: /HOME
```

Możesz także sprawdzić w oknie Profil środowiska użytkownika, otwieranym z okna Właściwości użytkownika w Menedżerze użytkowników Windows NT, czy katalog macierzysty został automatycznie ustawiony.

Inne skrypty połączeniowe

Kiedy użytkownik pomyślnie połączy się z udziałem Samby, serwer Samby może wykonać po swojej stronie program, który przygotuje udział do użycia. Samba pozwala na wykonywanie skryptów zarówno przed nawiązaniem połączenia, jak i po rozłączeniu z udziałem. Nie trzeba używać domen Windows, aby móc skorzystać z tych opcji. W tabeli 6.11 przedstawiamy niektóre opcje używane do konfigurowania środowiska użytkownika.

Tabela 6.11. Opcje skryptów połączeniowych

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`root preexec`	Łańcuch (polecenie Uniksa)	Określa polecenie, które ma zostać wykonane z przywilejami roota przed nawiązaniem połączenia z udziałem	Brak	Udział
`preexec (exec)`	Łańcuch (polecenie Uniksa)	Określa polecenie, które ma zostać wykonane z przywilejami użytkownika przed nawiązaniem połączenia z udziałem	Brak	Udział
`postexec`	Łańcuch (polecenie Uniksa)	Określa polecenie, które ma zostać wykonane z przywilejami użytkownika po rozłączeniu się z udziałem	Brak	Udział
`root postexec`	Łańcuch (polecenie Uniksa)	Określa polecenie, które ma zostać wykonane z przywilejami roota po rozłączeniu się z udziałem	Brak	Udział

root preexec

Pierwsza forma polecenia logowania nosi nazwę `root preexec`. Opcja ta określa polecenie Uniksa, które zostanie wykonane *z przywilejami roota*, zanim zostanie nawiązane połączenie z udziałem. Możesz użyć go w celu wykonania poleceń, które wymagają przywilejów roota, na przykład do montowania CD-ROM-ów w celu udostępnienia ich klientom albo do tworzenia katalogów. Jeśli nie określisz opcji `root preexec`, nie zostanie podjęta żadna domyślna czynność. Oto przykład użycia tej opcji w celu zamontowania CD-ROM-u:

```
[homes]
    browseable = no
    writeable = yes
    root preexec = /etc/mount /dev/cdrom2
```

Pamiętaj, że podane polecenie zostanie wykonane z przywilejami roota. Ze względów bezpieczeństwa użytkownicy nie powinni mieć możliwości zmiany polecenia w opcji `root preexec`.

preexec

Następne polecenie uruchamiane przed zalogowaniem użytkownika określa się za pomocą opcji `preexec` (czasem nazywanej po prostu `exec`). Jest to zwykłe, nieuprzywilejowane polecenie, wykonywane przez Sambę z przywilejami użytkownika określonego zmienną `%u`. Opcji tej często używa się do rejestrowania połączeń, jak w poniższym przykładzie:

```
[homes]
    preexec = echo "%u po□□czy□ si□ z %S z komputera %m (%I)" >> /tmp/.log
```

Pamiętaj, że dane wysyłane przez to polecenie na standardowe wyjście nie będą widoczne dla użytkownika, lecz zostaną odrzucone. Jeśli chcesz używać skryptu `preexec`, upewnij się, że działa on poprawnie, zanim zlecisz Sambie jego wykonywanie.

postexec

Kiedy użytkownik rozłączy się z udziałem, polecenie określone opcją `postexec` zostanie wykonane w serwerze Samby z przywilejami użytkownika, na przykład w celu uporządkowania systemu. Opcja ta bardzo przypomina opcję `preexec`. Także tutaj polecenie jest wykonywane z przywilejami użytkownika określonego zmienną `%u`, a dane wysyłane na standardowe wyjście są ignorowane.

root postexec

Po poleceniu `postexec` wykonywane jest polecenie `root postexec`, jeśli je określono. Opcja ta również przyjmuje polecenie Uniksa i wykonuje je *z przywilejami roota* przed odłączeniem udziału. Powinieneś jej używać tylko w celu wykonania poleceń, które wymagają przywilejów roota.

Współpraca z NIS i NFS

Samba potrafi także współpracować z NIS i NIS+. Jeśli w sieci znajduje się więcej niż jeden serwer plików, a w każdym działa Samba, byłoby dobrze, gdyby klient SMB łączył się z tym serwerem, który przechowuje macierzysty katalog użytkownika. Zwykle nie ma sensu przesyłać plików siecią za pomocą NFS do serwera Samby tylko po to, żeby za chwilę odesłać je klientowi SMB (operacja taka jest wolna – około 30 procent normalnej prędkości Samby). Dlatego istnieją dwie opcje, które informują Sambę, że NIS zna nazwę właściwego serwera, i wskazują, w której mapie NIS można znaleźć odpowiednie informacje.

W tabeli 6.12 przedstawiamy kolejne opcje, które służą do konfigurowania środowiska użytkownika.

Tabela 6.12. Opcje NIS

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`nis homedir`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba sprawdza ścieżkę do macierzystego katalogu użytkownika w NIS, a nie w pliku */etc/passwd*	`no`	Globalny
`homedir map`	Łańcuch (nazwa mapy NIS)	Określa nazwę mapy NIS, w której należy wyszukać macierzysty katalog użytkownika	Brak	Globalny

nis homedir i homedir map

Opcje `nis homedir` i `homedir map` są przeznaczone do użycia w sieciach, w których uniksowe katalogi macierzyste są udostępniane za pomocą NFS, programu montującego i NIS (Yellow Pages).

Opcja `nis homedir` informuje, że serwer z katalogiem macierzystym użytkownika należy wyszukać za pomocą NIS. Opcja `homedir map` informuje Sambę, w której mapie NIS należy szukać serwera z katalogiem macierzystym użytkownika. Musi to

być serwer Samby, aby klient mógł nawiązać z nim połączenie za pomocą SMB, a w pierwotnym serwerze Samby musi być zainstalowany NIS, aby można było przeprowadzić wyszukiwanie.

Jeśli na przykład użytkownik `jacek` poprosi o połączenie z udziałem `[jacek]`, a opcja `nis homedir` jest ustawiona na `yes`, Samba sprawdzi, czy w pliku określonym opcją `homedir map` znajduje się informacja o katalogu macierzystym tego użytkownika. Jeśli Samba znajdzie taką informację, wówczas zwróci klientowi związaną z nią nazwę komputera. Klient spróbuje wówczas połączyć się z drugim komputerem i tam uzyskać dostęp do udziału. Wyszukiwanie NIS włącza się w następujący sposób:

```
[global]
    nis homedir = yes
    homedir map = amd.map
```

7

Drukowanie i odwzorowywanie nazw

W rozdziale tym zajmiemy się dwoma zagadnieniami: konfigurowaniem drukarek w serwerze Samby i konfigurowaniem Samby do współpracy z serwerem Windows Internet Name Service (WINS) lub do pracy w roli tego serwera. Samba umożliwia klientom wysyłanie dokumentów do drukarek połączonych z serwerem, a także drukowanie dokumentów uniksowych na drukarkach podłączonych do komputerów Windows. W pierwszej części rozdziału omówimy konfigurowanie drukarek do pracy po obu stronach połączenia.

W drugiej części rozdziału omówimy Windows Internet Name Service, implementację serwera nazw NetBIOS-owych (*NetBIOS Name Server*, NBNS) Microsoftu. Jak wspomniano w rozdziale 1, *Poznajemy Sambę*, serwer NBNS pozwala komputerom w sieci NetBIOS-owej na odwzorowywanie nazw bez uciekania się do rozgłoszeń. Każdy komputer wie, gdzie znajduje się serwer WINS, i może pytać go o adresy IP innych komputerów w sieci.

Wysyłanie zleceń wydruku do Samby

Drukarka podłączona do serwera Samby pojawia się na liście udziałów w Otoczeniu sieciowym. Jeśli drukarka jest zarejestrowana w kliencie, a klient dysponuje jej sterownikami, wówczas klient może bez problemu wysyłać zlecenia wydruku do drukarki podłączonej do serwera Samby. Rysunek 7.1 przedstawia drukarkę Samby w oknie Otoczenia sieciowego klienta Windows.

Do zarządzania drukarkami Samby będzie ci potrzebna znajomość procesu, który umożliwia drukowanie w sieci. Wysyłanie zleceń wydruku do drukarki podłączonej do serwera Samby składa się z czterech etapów:

1. Otwarcie i uwierzytelnienie połączenia z udziałem drukarki.
2. Skopiowanie pliku przez sieć.
3. Zamknięcie połączenia.
4. Wydrukowanie i usunięcie kopii pliku.

Rysunek 7.1. Drukarka Samby w Otoczeniu sieciowym

Kiedy Samba odbierze zlecenie wydruku, dane są tymczasowo zapisywane na dysku w katalogu określonym opcją `path` udziału drukarki. Samba wykonuje następnie uniksowe polecenie wydruku, aby przesłać dane do drukarki. Zlecenie jest drukowane z przywilejami uwierzytelnionego użytkownika udziału. Warto zauważyć, że może to być użytkownik-gość, w zależności od tego, jak skonfigurowany jest udział.

Polecenia wydruku

Aby wydrukować dokument, musisz poinformować Sambę, jakie polecenie służy do drukowania i usuwania plików. W Linuksie poleceniem tym jest:

```
lpr -r -Pdrukarka plik
```

Polecenie to nakazuje programowi `lpr` skopiować plik do katalogu buforowego, zwykle */var/spool*, wyszukać nazwę drukarki w systemowym pliku konfiguracyjnym (*/etc/printcap*) i zinterpretować znalezione tam informacje w celu odpowiedniego przetworzenia danych i wysłania ich do właściwego urządzenia fizycznego. Ze względu na użycie opcji `-r` plik podany w linii polecenia zostanie usunięty po wydrukowaniu. Oczywiście, usuwana jest tylko kopia pliku z serwera Samby, pierwotny plik w kliencie pozostaje nienaruszony.

Linux używa mechanizmów wydruku Uniksa typu Berkeley (BSD), ale w Systemie V proces przebiega podobnie. Tutaj do wydrukowania i usunięcia pliku potrzebne jest polecenie złożone:

```
lp -ddrukarka -s plik; rm plik
```

W Systemie V plik */etc/printcap* jest zastąpiony zbiorem plików konfiguracyjnych położonych w katalogu */usr/spool/lp* i nie ma opcji umożliwiającej usunięcie pliku po wydrukowaniu. Musisz zrobić to sam i właśnie dlatego dołączyliśmy polecenie `rm`.

Zmienne związane z drukowaniem

Samba udostępnia cztery zmienne przeznaczone do użycia w opcjach konfiguracji drukowania. Wymieniono je w tabeli 7.1.

Tabela 7.1. Zmienne związane z drukowaniem

Zmienna	*Definicja*
`%s`	Pełna ścieżka do drukowanego pliku w serwerze Samby
`%f`	Sama nazwa drukowanego pliku (bez ścieżki) w serwerze Samby
`%p`	Nazwa uniksowej drukarki, na której zostanie wydrukowany plik
`%j`	Numer zlecenia wydruku (do użycia w poleceniach `lprm`, `lppause` i `lpresume`)

Minimalna konfiguracja drukarki

Zacznijmy od prostego, ale pouczającego przykładu konfiguracji udziału drukarki. Zakładając, że Samba działa w Linuksie, a w pliku parametrów drukarek znajduje się drukarka o nazwie `lp`, dopisanie poniższych linii do pliku *smb.conf* udostępni tę drukarkę w sieci:

```
[drukarka1]
    printable = yes
    print command = /usr/bin/lpr -r %s
    printer = lp
    printing = BSD
    read only = yes
    guest ok = yes
```

Ta konfiguracja pozwala na wysyłanie danych do drukarki wszystkim użytkownikom, co później zapewne zechcesz zmienić. W tym momencie wystarczy wiedzieć, że zmienna `%s` w opcji `print command` zostanie zastąpiona nazwą drukowanego pliku, kiedy Samba wykona to polecenie. Zmiana używanego mechanizmu drukowania polega zwykle na zmodyfikowaniu prawej strony opcji `print command` oraz określeniu innego typu mechanizmu w opcji `printing`.

Spójrzmy na opcje używane w Uniksie typu System V. Wykorzystując zmienne, możemy zapisać polecenie Systemu V w następujący sposób:

```
print command = lp -d%p -s %s; rm %s
```

Jak wspomniano wcześniej, zmienna `%p` oznacza nazwę drukarki, a zmienna `%s` – nazwę drukowanego pliku. Powinieneś także zmienić opcję `printing`, aby poinformować, że używasz mechanizmu drukowania z Systemu V:

```
printing = SYSV
```

Jeśli używasz zabezpieczeń na poziomie użytkownika, zwróć szczególną uwagę na konto gościnne używane przez Sambę. Typowe ustawienie, `nobody`, może nie pozwalać na drukowanie w twoim systemie. Jeśli rzeczywiście tak jest, powinieneś dopisać opcję `guest account` w udziale drukarki (a nawet w sekcji globalnej), która będzie określać konto z zezwoleniem na wydruk. Twórcy Samby polecają konto `ftp`, które często jest skonfigurowane tak, aby mogli z niego bezpiecznie korzystać niezaufani użytkownicy-goście. Możesz ustawić je za pomocą poniższej opcji:

```
guest account = ftp
```

Z drukowaniem wiąże się też inna kwestia: klienty czasem muszą dowiedzieć się, jaki jest status zlecenia wysłanego do serwera Samby. Samba nie zapobiega wy-

słaniu dokumentu do już zajętego udziału drukarki, więc musi informować klienty nie tylko o statusie bieżącego zlecenia wydruku, ale także o innych dokumentach czekających na wydrukowanie. Samba musi także udostępniać klientom możliwość wstrzymywania, wznawiania i usuwania zleceń z kolejki wydruku. Istnieją opcje kontrolujące każdą z tych czynności. Jak łatwo się domyślić, są one realizowane za pomocą poleceń Uniksa. Odpowiednie opcje to:

- `lpq command`,
- `lprm command`,
- `lppause command`,
- `lpresume command`.

Nieco dalej znajdziesz szczegółowy opis tych opcji, jednakże w większości przypadków za ich ustawienia jest odpowiedzialna opcja `printing` i nie ma potrzeby zmieniania wartości domyślnych.

Oto kilka ważnych wskazówek dotyczących udziałów drukarek:

- We wszystkich udziałach drukarek (nawet `[printers]`) musisz ustawić opcję `printable = yes`, aby Samba wiedziała, że definiują one drukarki. Jeśli o tym zapomnisz, udziały te zostaną uznane nie za drukarki, ale za udziały dyskowe.
- Jeśli ustawisz opcję konfiguracyjną `path` w udziale drukarki, wszystkie pliki wysyłane do drukarki będą kopiowane do podanego przez ciebie katalogu, a nie do domyślnego katalogu */tmp*. Ponieważ w niektórych systemach uniksowych przestrzeń dyskowa przydzielona katalogowi */tmp* jest relatywnie niewielka, wielu administratorów woli używać katalogu */var/spool* lub innego.
- Opcja `read only` w udziałach drukarek jest ignorowana.
- Jeśli ustawisz opcję `guest ok = yes` w udziale drukarki, a Samba jest skonfigurowana do pracy z zabezpieczeniami na poziomie udziału, każdy będzie mógł wysłać dane do drukarki, z przywilejami użytkownika określonego opcją `guest account`.

Użycie jednego lub kilku komputerów z Sambą jako serwerów wydruku pozwala na elastyczne zarządzanie zasobami sieci lokalnej. Możesz łatwo rozdzielić drukarki, na przykład udostępniając część z nich tylko pracownikom jednego działu lub prowadząc bank drukarek dostępnych dla wszystkich. Możesz także ograniczyć dostęp do drukarki za pomocą opcji `valid users`, zezwalając na wydruk tylko wybranym użytkownikom:

```
[deskjet]
    printable = yes
    path = /var/spool/samba/wydruk
    valid users = agata wojtek
```

Wszystkie inne opcje udostępniania udziałów omówione w poprzednich rozdziałach powinny działać także z udziałami drukarek. Ponieważ dostęp do drukarek Samby uzyskuje się za pomocą ich nazw, łatwo jest rozdzielić zadania wydruku między kilka serwerów, używając poleceń Uniksa przeznaczonych do równoważenia obciążenia lub konserwacji systemu.

Udział [printers]

W rozdziale 4, *Udziały dyskowe*, krótko przedstawiono specjalny udział `[printers]`, służący do automatycznego tworzenia usług wydruku. Przypomnijmy, jak to się odbywa: jeśli w pliku konfiguracyjnym Samby utworzysz udział o nazwie `[printers]`, Samba automatycznie wczyta plik z parametrami drukarek i utworzy udział dla każdej drukarki, która jest wymieniona w tym pliku. Jeśli na przykład w pliku z parametrami drukarek w serwerze Samby znajdują się definicje drukarek `lp`, `pcl` i `ps`, Samba udostępni trzy udziały drukarek o takich nazwach, konfigurując je zgodnie z opcjami zawartymi w sekcji `[printers]`.

Jak pamiętasz, kiedy klient zażąda dostępu do udziału niezdefiniowanego w pliku *smb.conf*, Samba postępuje według poniższych reguł:

- Jeśli nazwa żądanego udziału odpowiada nazwie użytkownika w systemowym pliku haseł i istnieje udział `[homes]`, tworzony jest nowy udział o nazwie odpowiadającej nazwie użytkownika i parametrach zdefiniowanych w sekcjach `[homes]` i `[global]`.
- W innym przypadku, jeśli nazwa żądanego udziału odpowiada nazwie drukarki w systemowym pliku z parametrami drukarek, tworzony jest nowy udział o nazwie odpowiadającej nazwie drukarki i parametrach zdefiniowanych w sekcji `[printers]` (sekcja `[global]` nie jest tu brana pod uwagę).
- Jeśli Samba nie znajdzie żadnego z powyższych, wówczas szuka udziału określonego za pomocą opcji `default service`. Jeśli go nie znajdzie, zwraca błąd.

Rzecz, która wymaga podkreślenia: nie nadawaj drukarce nazwy, którą ma użytkownik. Mogłoby się bowiem zdarzyć, że połączyłbyś się z udziałem dyskowym, zamiast z drukarką.

Oto przykładowy udział `[printers]` dla Linuksa (mechanizm wydruku BSD). Niektóre z tych opcji powielają wartości domyślne; zamieściliśmy je tu jednak dla pełnego obrazu:

```
[global]
    printing = BSD
    print command = /usr/bin/lpr -P%p -r %s
    printcap file = /etc/printcap
    min print space = 2000

[printers]
    path = /usr/spool/public
    printable = true
    guest ok = true
    guest account = pcguest
```

W przykładzie tym podaliśmy opcje globalne, które określają: typ wydruku (BSD), polecenie wysyłające dane do drukarki i usuwające plik tymczasowy, nazwę pliku z parametrami drukarek i minimalną przestrzeń buforowania wydruku równą 2 megabajtom.

Oprócz tego utworzyliśmy udział `[printers]`, udostępniający wszystkie drukarki systemowe. Nasz tymczasowy katalog buforowy jest określony opcją `path`: */usr/spool/public*. Poszczególne udziały będą umożliwiały drukowanie, o czym informuje

opcja `printable` – jej użycie jest konieczne, nawet w sekcji `[printers]`. Dwie opcje `guest` są przydatne, jeśli Samba używa zabezpieczeń na poziomie użytkownika: zezwalamy na gościnny dostęp do drukarki i określamy użytkownika, z którego przywilejami Samba będzie wykonywać polecenia wydruku.

Test drukowania

Oto, jak przetestować wydruk na drukarce Samby. Przyjmijmy najbardziej skomplikowany scenariusz i użyjmy konta gościa. Najpierw wykonaj polecenie Samby *testparm* na pliku konfiguracyjnym zawierającym udziały drukarek, jak robiliśmy to w rozdziale 2, *Instalowanie Samby w Uniksie*. Dzięki temu dowiesz się, czy w pliku nie ma żadnych błędów składni. Oto, co byś zobaczył, gdybyś pominął opcję `path` w poprzednim przykładzie:

```
# testparm
Load smb config files from /usr/local/samba/lib/smb.conf
Processing configuration file "/usr/local/samba/lib/smb.conf"
Processing section "[global]"
Processing section "[homes]"
Processing section "[dane]"
Processing section "[printers]"
No path in service printers - using /tmp
Loaded services file OK.
Press enter to see a dump of your service definitions
Global parameters:
    load printers: Yes
    printcap name: /etc/printcap
Default service parameters:
    guest account: ftp
    min print space: 0
    print command: lpr -r -P%p %s
    lpq command: lpq -P%p
    lprm command: lprm -P%p %j
    lppause command:
    lpresume command:
Service parameters [printers]:
    path: /tmp
    print ok: Yes
    read only: true
    public: true
```

Następnie wypróbuj polecenie `testprns` *`nazwadrukarki`*. Jest to prosty program, który sprawdza, czy określona nazwa drukarki jest obecna w pliku *printcap*. Jeśli twój plik *printcap* nie znajduje się w zwykłym miejscu, możesz podać jego nazwę wraz ze ścieżką jako drugi argument polecenia *testprns*:

```
# testprns lp /etc/printcap
Looking for printer lp in printcap file /etc/printcap
Printer name lp is valid.
```

Następnie zaloguj się jako gość, przejdź do katalogu buforowego i sprawdź, czy możesz drukować za pomocą tego samego polecenia, którego według programu *testprns* będzie używać Samba. Jak wspomniano wcześniej, dzięki temu dowiesz się, czy musisz zmodyfikować konto gościnne, ponieważ domyślne konto często nie ma uprawnień do drukowania.

Następnie wydrukuj coś na serwerze Samby za pomocą programu `smbclient` i sprawdź, czy zachodzą następujące zdarzenia:

- zlecenie wydruku pojawia się (na krótko) w katalogu buforowym Samby określonym opcją `path`,
- zlecenie pojawia się w katalogu buforowym systemu wydruku,
- zlecenie znika z katalogu buforowego używanego przez Sambę.

Jeśli *smbclient* nie może drukować, możesz zmienić opcję `print command` tak, aby rejestrowała informacje diagnostyczne:

```
print command = /bin/cat %s >> /tmp/dziennikdruku; rm %s
```

albo

```
print command = echo "Wydrukowano %s na drukarce %p" >> /tmp/dziennikdruku
```

Problemy z konfiguracją drukarek w Sambie często są spowodowane pominięciem pełnych ścieżek do programów; proste polecenia często nie działają, ponieważ zmienna PATH konta gościnnego nie zawiera odpowiednich katalogów. Innym często występującym problemem jest brak odpowiednich praw dostępu do katalogu buforowego.

Więcej informacji o diagnozowaniu drukarek znajdziesz w dokumentacji Samby (plik *Printing.txt*). Uniksowe systemy wydruku są też omówione w książce Æeleen Frisch *Unix – administracja systemu* (opublikowanej przez Wydawnictwo RM).

Konfigurowanie i testowanie klienta Windows

Kiedy Samba udostępnia już działającą drukarkę, musisz odpowiednio skonfigurować klienta Windows. Spójrz na serwer Samby w Otoczeniu sieciowym. Powinien on obecnie wyświetlać wszystkie dostępne drukarki. Na rysunku 7.1 widzimy drukarkę o nazwie `lp`.

Klient Windows musi teraz rozpoznać drukarkę. Zacznij od dwukrotnego kliknięcia jej ikony. Jeśli spróbujesz wybrać niezainstalowaną drukarkę (co właśnie zrobiłeś), Windows zapyta, czy ma ci dopomóc w skonfigurowaniu jej do użycia przez system Windows. Odpowiedz Tak, a pojawi się okno Kreatora dodawania drukarki.

Kreator zapyta najpierw, czy chcesz drukować z DOS-a. Załóżmy, że nie chcesz, więc wybierz Nie i kliknij przycisk Dalej, aby otworzyć okno dialogowe pokazane na rysunku 7.3.

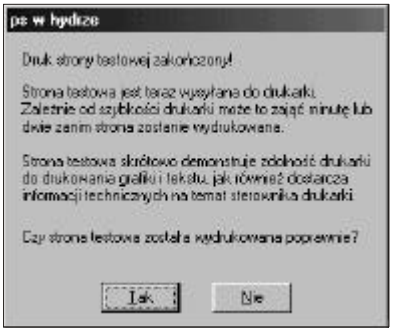
Rysunek 7.2. Drukarka w Otoczeniu sieciowym

W tym oknie dialogowym powinieneś zobaczyć długą listę producentów i modeli drukarek. Jeśli nie znajdziesz na niej swojej drukarki, ale wiesz, że jest to drukarka postscriptowa, wybierz Apple jako producenta i Apple LaserWriter jako model. Otrzymasz w ten sposób dostęp do podstawowych funkcji drukarki postscriptowej, a konfiguracja taka jest przez wielu uważana za najbardziej niezawodną. Jeśli do komputera dołączone są już jakieś drukarki postscriptowe, pojawi się pytanie, czy chcesz zastąpić istniejący sterownik, czy wykorzystać go do obsługi nowej drukarki. Pamiętaj, że jeśli zastąpisz sterownik, inne drukarki mogą przestać działać. Dlatego zalecamy pozostawienie istniejących sterowników, jeśli działają poprawnie.

Następnie Kreator poprosi o podanie nazwy drukarki. W przykładzie z rysunku 7.3 system Windows nadał jej domyślną nazwę „Apple LaserWriter (Kopia 2)", a my zmieniliśmy ją na „drukarka ps w serwerze Samby", aby wiedzieć, gdzie szukać gotowych wydruków. Możesz jednak nadać drukarce dowolną nazwę.

Rysunek 7.3. Producenci i modele drukarek

Wreszcie Kreator dodawania drukarki zapyta, czy chcesz wydrukować stronę testową. Kliknij Tak, a powinieneś zobaczyć okno dialogowe z rysunku 7.4.



Rysunek 7.4. Informacja o pomyślnym zakończeniu drukowania

Jeśli testowy wydruk nie powiedzie się, kliknij przycisk Nie w oknie dialogowym z rysunku 7.4, a Kreator dodawania drukarki przeprowadzi cię przez kilka kroków diagnozujących proces drukowania po stronie klienta. Jeśli testowy wydruk udał się, gratulujemy! Zdalna drukarka będzie odtąd dostępna dla aplikacji PC w menu Plik⇨Drukuj.

Automatyczne konfigurowanie sterowników drukarki

W poprzednim podrozdziale opisaliśmy ręczne instalowanie sterownika drukarki w systemie Windows. Jako administrator systemu nie zawsze możesz zagwarantować, że użytkownicy bezbłędnie wykonają wszystkie wymagane czynności. Na szczęście możesz poinstruować Sambę, aby automatycznie konfigurowała sterowniki dla określonych drukarek.

Samba ma trzy opcje służące do automatycznego konfigurowania sterowników drukarki w klientach łączących się z nią po raz pierwszy. Są to: `printer driver`, `printer driver file` oraz `printer driver location`. W podrozdziale tym opiszemy, jak ustawić te opcje, aby umożliwić użytkownikom pominięcie okna wyboru producenta w Kreatorze dodawania drukarki.

Więcej informacji na ten temat znajdziesz w pliku *PRINTER_DRIVER.TXT* w dokumentacji Samby.

Procedura składa się z czterech zasadniczych etapów:

1. Zainstaluj sterowniki drukarki w kliencie Windows (drukarka nie musi być do niego podłączona).
2. Utwórz plik definicji drukarki, używając informacji z komputera Windows.
3. Utwórz udział `PRINTER$`, w którym umieścisz uzyskane pliki sterowników.
4. Odpowiednio zmodyfikuj plik konfiguracyjny Samby.

Teraz szczegółowo opiszemy wszystkie etapy.

Instalowanie sterowników w kliencie Windows

Na tym etapie skorzystasz z klienta Windows 95/98. Nie ma znaczenia, który to będzie klient, jeśli tylko będzie można zainstalować w nim odpowiednie sterowniki drukarki. Nie musisz nawet podłączać do niego drukarki – chodzi tylko o to, aby skopiować właściwe pliki sterowników do katalogu Windows. Najpierw otwórz okno Drukarki w folderze Mój komputer i kliknij dwukrotnie ikonę Dodaj drukarkę (patrz rysunek 7.5).

Teraz możesz użyć okien dialogowych Kreatora dodawania drukarki, aby wybrać jej producenta i model. Jeśli pojawi się pytanie, czy chcesz drukować z DOS-a, odpowiedz przecząco. System Windows załaduje odpowiednie sterowniki z CD-ROM-u i zapyta, czy chcesz wydrukować stronę testową. Ponownie odpowiedz przecząco i zamknij okno dialogowe Kreatora.

Rysunek 7.5. Okno Drukarki

Tworzenie pliku definicji drukarki

Plik definicji drukarki możesz utworzyć za pomocą skryptu *make_printerdef* znajdującego się w katalogu */usr/local/samba/bin*. Aby skorzystać z tego skryptu, musisz skopiować następujące cztery pliki z klienta Windows*:

C:\WINDOWS\INF\MSPRINT.INF

C:\WINDOWS\INF\MSPRINT2.INF

C:\WINDOWS\INF\MSPRINT3.INF

C:\WINDOWS\INF\MSPRINT4.INF

Kiedy już skopiujesz te pliki, możesz utworzyć plik definicji drukarki, używając odpowiedniego sterownika drukarki i jego pliku .INF. Jeśli nazwa sterownika zaczyna się od litery z zakresu od A do K, użyj albo pliku *MSPRINT.INF*, albo *MSPRINT3.INF*. Jeśli zaczyna się od litery z zakresu od L do Z, użyj albo pliku *MSPRINT2.INF*, albo *MSPRINT4.INF*. Być może będziesz musiał przeszukać te pliki poleceniem *grep*, aby znaleźć swój sterownik. W poniższym przykładzie zlokalizowaliśmy nasz sterownik w pliku *MSPRINT3.INF* i utworzyliśmy plik definicji drukarki HP DeskJet 560C:

```
$grep "HP Deskjet 560C Printer" MSPRINT.INF MSPRINT3.INF
MSPRINT3.INF: "HP DeskJet 560C Printer"=DESKJETC.DRV,HP_DeskJet_ ...

$make_printerdef MSPRINT3.INF "HP DeskJet 560C Printer" >printers.def
FOUND:DESKJETC.DRV
End of section found
CopyFiles: DESKJETC,COLOR_DESKJETC
Datasection: (null)
Datafile: DESKJETC.DRV
Driverfile: DESKJETC.DRV
Helpfile: HPVDJC.HLP
LanguageMonitor: (null)

Copy the following files to your printer$ share location:
DESKJETC.DRV
HPVCM.HPM
HPVIOL.DLL
```

* W starszych klientach Windows 95 mogą być tylko dwa pierwsze pliki.

```
HPVMON.DLL
HPVRES.DLL
HPCOLOR.DLL
HPVUI.DLL
HPVDJCC.HLP
color\HPDESK.ICM
```

Zapamiętaj pliki, o których skopiowanie prosi skrypt. Będziesz ich potrzebował w następnym etapie.

Tworzenie udziału PRINTER$

Ten etap jest względnie łatwy. Utwórz w pliku *smb.conf* udział o nazwie `[PRINTER$]`, wskazujący na pusty katalog w serwerze Samby. Kiedy to zrobisz, skopiuj pliki wskazane przez skrypt *make_printerdef* w położenie określone opcją konfiguracyjną `path` udziału `[PRINTER$]`. Do pliku konfiguracyjnego możesz dopisać na przykład następujące linie:

```
[PRINTER$]
    path = /usr/local/samba/druk
    read only = yes
    browsable = no
    guest ok = yes
```

Pliki wskazane przez skrypt *make_printerdef* znajdują się zwykle w katalogu *C:\WINDOWS\SYSTEM*, ale możesz poznać ich dokładne położenie za pomocą poleceń:

```
cd C:\WINDOWS
dir nazwapliku /s
```

W naszym przykładzie każdy plik musi zostać skopiowany do katalogu */usr/local/samba/druk* w serwerze Samby. Oprócz tego skopiuj do udziału również utworzony przez siebie plik *printers.def*. Kiedy to zrobisz, wszystko będzie niemal gotowe.

Modyfikowanie pliku konfiguracyjnego Samby

Ostatni etap polega na zmodyfikowaniu pliku konfiguracyjnego Samby przez dopisanie poniższych opcji:

- `printer driver`,
- `printer driver file`,
- `printer driver location`.

Globalna opcja `printer driver file` powinna wskazywać na plik *printers.def*; umieść ją w sekcji `[global]`. Pozostałe dwie opcje należy umieścić w udziale, dla którego chcesz automatycznie instalować sterowniki. Wartość opcji `printer driver` powinna odpowiadać nazwie, która pojawia się w Kreatorze dodawania drukarki w systemie Windows. Wartością opcji `printer driver location` powinna być ścieżka UNC do utworzonego przez ciebie udziału `[PRINTER$]`, a nie uniksowa ścieżka do katalogu serwera. Mógłbyś zatem użyć następujących opcji:

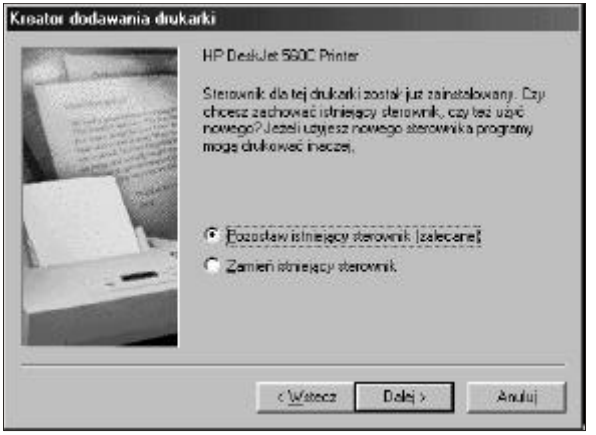
```
[global]
    printer driver file = /usr/local/samba/print/printers.def
[hpdeskjet]
```

```
path = /var/spool/samba/drukarki
printable = yes

printer driver = HP DeskJet 560C Printer
printer driver location = \\%L\PRINTER$
```

Teraz możesz wypróbować swoją konfigurację. Powinieneś usunąć drukarkę Windows, którą „zainstalowałeś" na pierwszym etapie, gdy wybrałeś ją z listy w oknie dialogowym Drukarki. Jeśli Windows zapyta, czy chcesz usunąć niepotrzebne pliki, zrób to. Niebawem zostaną one zastąpione plikami, które obecnie są przechowywane w serwerze Samby.

Testowanie konfiguracji

Zrestartuj demony Samby i poszukaj udziału `[hpdeskjet]` pod ikoną serwera w Otoczeniu sieciowym. Jeśli teraz klikniesz ikonę drukarki, powinieneś zapoczątkować proces jej instalacji i dojść do okna dialogowego pokazanego na rysunku 7.6.



Rysunek 7.6. Automatyczne konfigurowanie sterownika drukarki

Różni się ono od okna, które pojawiło się wcześniej podczas konfigurowania drukarki. Zasadniczo, Windows pyta, czy chcesz zaakceptować „już zainstalowany" sterownik – czyli ten, który udostępnia Samba. Wybierz więc opcję zachowania istniejącego sterownika i kliknij przycisk Dalej. Teraz będziesz mógł nadać drukarce nazwę i wydrukować stronę testową. Jeśli wszystko działa, konfiguracja jest poprawna. Będziesz mógł powtórzyć tę procedurę w każdym kliencie Windows.

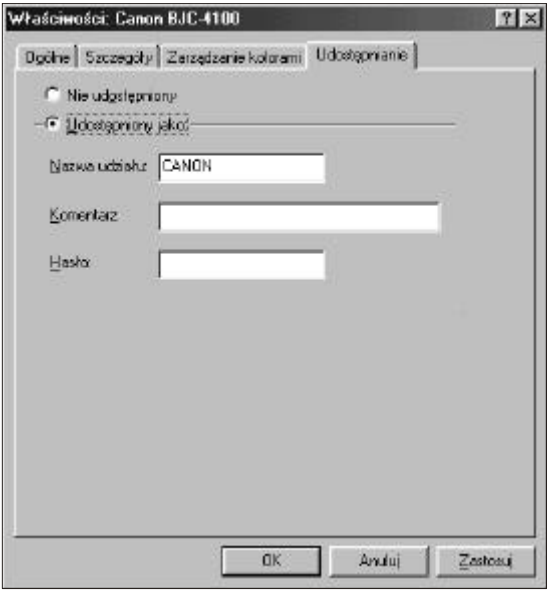
Drukowanie na drukarkach udostępnianych przez klienty Windows

Jeśli masz drukarki podłączone do klientów pracujących pod kontrolą Windows 95/98 lub NT 4.0, możesz uzyskać do nich dostęp z Samby. Samba wyposażona jest w narzędzie o nazwie *smbprint*, które służy do buforowania zleceń wydruku kierowanych do drukarek Windows. Aby jednak z niego skorzystać, będziesz musiał

skonfigurować drukarkę klienta jako współdzielony zasób. Jeśli jeszcze tego nie zrobiłeś, możesz użyć przycisku Start i otworzyć okno Drukarki (patrz rysunek 7.7).

Rysunek 7.7. Okno Drukarki

Wybierz drukarkę podłączoną lokalnie (w naszym przykładzie jest to drukarka Canona), kliknij ją prawym przyciskiem myszy i wybierz z menu pozycję Udostępnianie. Dostaniesz się w ten sposób do karty Udostępnianie w oknie właściwości drukarki (patrz rysunek 7.8). Jeśli chcesz udostępnić ją gościnnie wszystkim użytkownikom sieci lokalnej, pozostaw puste pole hasła.



Rysunek 7.8. Karta Udostępnianie w oknie właściwości drukarki

Kiedy już to zrobisz, możesz dodać swoją drukarkę do listy standardowych drukarek, a Samba może ją udostępnić wszystkim komputerom PC w grupie roboczej. Aby ułatwić instalację w Uniksie, dystrybucja Samby zawiera dwa przykładowe skrypty: *smbprint* i *smbprint.sysv*. Pierwszy działa z drukarkami typu BSD, drugi jest przeznaczony dla drukarek typu System V.

Drukarki BSD

Aby Unix typu BSD rozpoznał zdalną drukarkę, muszą być spełnione dwa warunki:

1. Umieszczenie wpisu dla drukarki w pliku */etc/printcap* (lub jego odpowiedniku).
2. Umieszczenie pliku konfiguracyjnego w katalogu */var/spool* drukarki.

Najpierw otwórz plik */etc/printcap* i dodaj wpis dla zdalnej drukarki. Pamiętaj, że pole filtra wejściowego (`if`) musi wskazywać na program *smbprint*, jeśli drukarka jest podłączona do komputera Windows 95/98. W Linuksie możesz dopisać poniższe linie:

```
laserjet:\
  :sd=/var/spool/lpd/laser:\          # katalog buforowy
  :mx#0:\                             # maksymalny rozmiar pliku (brak)
  :sh:\                               # brak nag□ówka (nie)
  :if=/usr/local/samba/bin/smbprint: # filtr tekstowy
```

Następnie musisz utworzyć plik konfiguracyjny w katalogu buforowym, który określiłeś wyżej parametrem `sd` (być może będziesz musiał utworzyć ten katalog). Plik musi mieć nazwę *.config* i powinien zawierać następujące informacje:

- NetBIOS-ową nazwę komputera Windows z drukarką,
- nazwę usługi reprezentującej drukarkę,
- hasło dostępu do tej usługi.

Ostatnie dwa parametry zostały wpisane na karcie Udostępnianie dla zasobu klienta Windows. W tym przypadku plik *.config* zawierałby trzy linie:

```
server = feniks
service = CANON
password = ""
```

Kiedy to zrobisz, zresetuj demony Samby i spróbuj wydrukować coś z dowolnego standardowego programu Uniksa.

Drukarki Systemu V

Wysyłanie zleceń wydruku w Uniksach typu System V jest nieco łatwiejsze. Tutaj musisz użyć skryptu *smbprint.sysv* z katalogu */usr/local/samba/examples/printing* i zrobić co następuje:

1. Zmień parametry `server`, `service` i `password` w skrypcie tak, aby odpowiadały NetBIOS-owej nazwie komputera, nazwie udostępnianej drukarki i hasłu. Dla usługi z poprzedniego przykładu użyłbyś następujących parametrów:

   ```
   server = feniks
   service = CANON
   password = ""
   ```

2. Wykonaj następujące polecenia, które utworzą odwołanie do drukarki w pliku z parametrami drukarek. Zauważ, że wpis dla nowej uniksowej drukarki nosi nazwę `canon`:

```
# lpadmin -p canon -v /dev/null -i./smbprint.sysv
# enable canon
# accept canon
```

Kiedy to zrobisz, zresetuj demony Samby i spróbuj wydrukować coś z dowolnego standardowego programu Uniksa. Teraz będziesz mógł wysyłać dane do drukarki podłączonej do klienta Windows.

Opcje drukowania w Sambie

W tabeli 7.2 zebrano opcje drukowania w Sambie.

Tabela 7.2. Opcje konfiguracji drukowania

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`printing`	`bsd`, `sysv`, `hpux`, `aix`, `qnx`, `plp`, `softq` lub `lprng`	Ustawia typ systemu drukowania w twojej odmianie Uniksa	Zależna od systemu	Udział
`printable` (`print ok`)	Wartość logiczna	Oznacza udział jako udział drukarki	`no`	Udział
`printer` (`printer name`)	Łańcuch (uniksowa nazwa drukarki)	Ustawia nazwę drukarki wysyłaną do klientów	Zależna od systemu	Udział
`printer driver`	Łańcuch (nazwa sterownika drukarki)	Ustawia nazwę sterownika, z którego powinny korzystać klienty podczas wysyłania danych do drukarki	Brak	Udział
`printer driver file`	Łańcuch (nazwa pliku wraz ze ścieżką)	Ustawia nazwę pliku sterownika drukarki	Brak	Globalny
`printer driver location`	Łańcuch (ścieżka sieciowa)	Określa sieciową ścieżkę do udziału z plikiem sterownika drukarki	Brak	Udział
`lpq cache time`	Wartość liczbowa (czas w sekundach)	Ustawia czas, przez który Samba będzie pamiętała status drukarki	`10`	Globalny
`postscript`	Wartość logiczna	Traktuje wszystkie zlecenia wydruku jak postscriptowe, dołączając znaki `%!` na początku każdego pliku	`no`	Udział

Dokończenie tabeli na str. 206

Dokończenie tabeli ze str. 205

Tabela 7.2. Opcje konfiguracji drukowania

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`load printers`	Wartość logiczna	Automatycznie ładuje wszystkie drukarki z pliku *printcap* na listę udziałów	`no`	Globalny
`print command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które zapoczątkowuje wydruk	Patrz niżej	Udział
`lpq command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które informuje o statusie kolejki wydruku	Patrz niżej	Udział
`lprm command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które usuwa zlecenie z kolejki wydruku	Patrz niżej	Udział
`lppause command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które wstrzymuje drukowanie zlecenia	Patrz niżej	Udział
`lpresume command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które wznawia drukowanie zlecenia	Patrz niżej	Udział
`printcap name` (`printcap`)	Łańcuch (nazwa pliku wraz ze ścieżką)	Określa położenie pliku z parametrami drukarek	Zależna od systemu	Globalny
`min print space`	Wartość liczbowa (rozmiar w kilobajtach)	Określa minimalny rozmiar wolnej przestrzeni dyskowej koniecznej do rozpoczęcia drukowania	`0`	Udział
`queuepause command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które wstrzymuje przetwarzanie kolejki wydruku	Patrz niżej	Udział
`queueresume command`	Łańcuch (polecenie powłoki)	Ustawia polecenie Uniksa, które wznawia przetwarzanie kolejki wydruku	Patrz niżej	Udział

printing

Opcja konfiguracyjna `printing` informuje Sambę o systemie druku używanym przez serwer. W Uniksie istnieje kilka zbiorów poleceń służących do sterowania wydrukiem i określania statusu zleceń wydruku. Samba obsługuje siedem różnych typów, wymienionych w tabeli 7.3.

Tabela 7.3. Typy systemów druku

Zmienna	*Definicja*
BSD	System Berkeley Unix
SYSV	System V
AIX	System operacyjny AIX (IBM)
HPUX	Unix Hewlett-Packard
QNX	System operacyjny czasu rzeczywistego QNX
LPRNG	LPR Next Generation (Powell)
SOFTQ	System SOFTQ
PLP	Portable Line Printer (Powell)

Opcja ta może przyjmować jedną z siedmiu wartości wymienionych w tabeli. Na przykład:

```
printing = SYSV
```

Domyślna wartość tej opcji jest zależna od systemu i konfigurowana podczas kompilacji Samby. W większości systemów skrypt *configure* automatycznie wykrywa używany system druku i odpowiednio konfiguruje go w pliku *makefile* Samby. Jeśli jednak używasz systemów druku PLP, LPRNG lub QNX, będziesz musiał to jawnie określić w pliku *makefile* lub udziale drukarki.

Najczęściej spotykane systemy druku to BSD i SYSV. Każda drukarka serwera Uniksa BSD jest opisana w pliku parametrów drukarek – zwykle */etc/printcap*.

Ustawiając opcję `printing`, automatycznie ustawiasz wartości przynajmniej trzech innych opcji danego udziału: `print command`, `lpq command` i `lprm command`. Jeśli Samba działa w serwerze, który nie obsługuje żadnej z powyższych metod druku, po prostu określ ręcznie wartości tych opcji.

printable

Opcja `printable` musi zostać ustawiona na `yes`, aby oznaczyć udział jako usługę drukarki. Jeśli opcja ta nie zostanie ustawiona, udział zostanie wzięty za udział dyskowy. Możesz ustawić tę opcję w następujący sposób:

```
[drukarka1]
    printable = yes
```

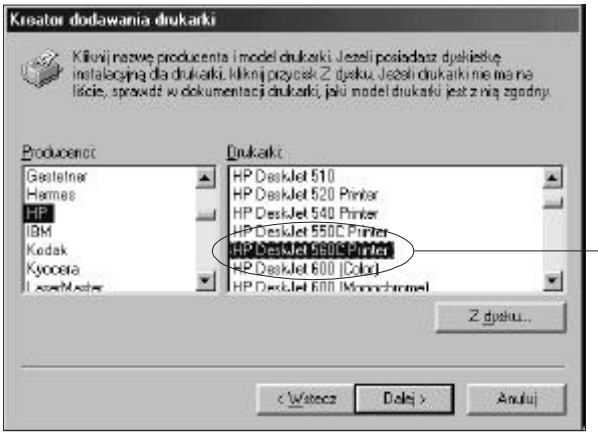
printer

Ta opcja (czasem używana w postaci `printer name`) określa nazwę drukarki serwera, której odpowiada udział. Opcja ta nie ma wartości domyślnej i powinna zostać jawnie ustawiona w pliku konfiguracyjnym, choć systemy uniksowe często same rozpoznają nazwę `lp` jako domyślną nazwę drukarki. Na przykład:

```
[deskjet]
    printer = hpdkjet1
```

printer driver

Opcja `printer driver` ustawia łańcuch, za pomocą którego Samba informuje Windows, jaką drukarką dysponuje. Jeśli opcja ta zostanie ustawiona poprawnie, Kreator dodawania drukarki w Windows będzie wiedział, jaką drukarkę instaluje, co ułatwi pracę użytkownikom i pozwoli im na pominięcie jednego okna dialogowego. Łańcuch podany w tej opcji powinien odpowiadać łańcuchowi wyświetlanemu w oknie Kreatora dodawania drukarki (patrz rysunek 7.9). Na przykład drukarka Apple LaserWriter jest opisana łańcuchem `Apple LaserWriter`, a drukarka Hewlett Packard Deskjet 560C – łańcuchem `HP DeskJet 560C Printer`.



Rysunek 7.9. Okno dialogowe Kreatora dodawania drukarki w Windows 98

Automatyczne konfigurowanie sterowników drukarek przez Sambę zostało szczegółowo opisane we wcześniejszym podrozdziale „Automatyczne konfigurowanie sterowników drukarki".

printer driver file

Ta globalna opcja określa położenie pliku definicji drukarki Windows 95/98, który jest niezbędny do wysłania plików sterownika do klientów korzystających z drukarki Samby. Domyślna wartość tej opcji to */usr/local/samba/lib/printer.def*. Możesz zmienić tę wartość w następujący sposób:

```
[deskjet]
    printer driver file = /var/drukarki/printers.def
```

Opcja ta jest opisana szczegółowo we wcześniejszym podrozdziale „Automatyczne konfigurowanie sterowników drukarki".

printer driver location

Ta opcja określa udział, który zawiera pliki sterowników i definicji drukarek Windows 95/98. Nie ma ona wartości domyślnej. Możesz określić położenie udziału za pomocą ścieżki sieciowej. Często używa się udziału w tym samym komputerze:

```
[deskjet]
    printer driver location = \\%L\PRINTER$
```

Opcja ta jest opisana szczegółowo we wcześniejszym podrozdziale „Automatyczne konfigurowanie sterowników drukarki".

lpq cache time

Globalna opcja `lpq cache time` pozwala na określenie czasu (w sekundach), przez który Samba będzie pamiętała bieżący status drukarki. Po upłynięciu tego czasu Samba wyda polecenie `lpq` (lub dowolne inne, które określiłeś opcją `lpq command`), aby uzyskać bardziej aktualny status. Domyślny czas to 10 sekund, ale możesz go zwiększyć, jeśli twoje polecenie określone opcją `lpq command` wykonuje się bardzo długo lub jeśli masz wiele klientów. Poniższy przykład ustawia czas na 30 sekund:

```
[deskjet]
    lpq cache time = 30
```

postscript

Opcja `postscript` sprawia, że drukarka traktuje wszystkie wysyłane do niej dane jako PostScript. Osiąga się to przez dołączenie znaków `%!` na początku pierwszej linii każdego zlecenia. Opcji tej zwykle używa się z klientami PC, które wstawiają znak `^D` ([Ctrl+D], czyli znak końca pliku) na początek pierwszej linii postscriptowego pliku. Oczywiście nie zamieni ona drukarki niepostscriptowej w postscriptową. Domyślna wartość tej opcji to `no`. Możesz zmienić ją następująco:

```
[deskjet]
    postscript = yes
```

print command, lpq command, lprm command, lppause command, lpresume command

Te opcje informują Sambę, jakich poleceń uniksowych należy użyć do sterowania wydrukiem i wysyłania danych do drukarki. Te polecenia Uniksa to: *lpr* (wysyła dane do drukarki), *lpq* (wyświetla kolejkę wydruku), *lprm* (usuwa zlecenia z kolejki) i, opcjonalnie, *lppause* oraz *lpresume*. Samba udostępnia opcje o nazwach odpowiadających każdemu z tych poleceń na wypadek, gdybyś musiał zmienić domyślne parametry. Oto przykład:

```
lpq command = /usr/ucb/lpq %p
```

Dzięki temu poleceniem `lpq command` byłoby */usr/ucb/lpq*. Podobnie opcja:

```
lprm command = /usr/local/lprm -P%p %j
```

ustawiłaby polecenie usunięcia zlecenia na */usr/local/lprm* i dostarczyła mu numer zlecenia wydruku za pomocą zmiennej `%j`.

Domyślne wartości tych opcji zależą od wartości opcji `printing`. W tabeli 7.4 znajdziesz domyślne polecenia dla każdego systemu druku. Najpopularniejszym systemem druku jest BSD.

Tabela 7.4. Domyślne polecenia dla różnych systemów druku

Opcja	*BSD, AIX, PLP, LPRNG*	*SYSV, HPUX*	*QNX*	*SOFTQ*
`print command`	`lpr -r -P%p %s`	`lp -c -d%p %s; rm %s`	`lpr -r -P%p %s`	`lp -d%p -s %s; rm %s`
`lpq command`	`lpq -P%p`	`lpstat -o%p`	`lpq -P%p`	`lpstat -o%p`
`lprm command`	`lprm -P%p %j`	`cancel %p-%j`	`cancel %p-%j`	`cancel %p-%j`
`lppause command`	`lp -i %p-%j -H hold (tylko SYSV)`	Brak	Brak	Brak
`lpresume command`	`lp -i %p-%j -H resume (tylko SYSV)`	Brak	Brak	`qstat -s -j%j -r`

Zwykle nie ma potrzeby modyfikowania tych opcji; wyjątkiem bywa `print command`. Być może będziesz musiał ustawić jawnie tę opcję, jeśli twój system nie przyjmuje opcji `-r` (usunąć po wydrukowaniu) w poleceniu wydruku. Na przykład:

```
/usr/local/lpr -P%p %s; /bin/rm %s
```

Odrobina programowania może zamienić te opcje w narzędzie diagnostyczne:

```
print command = cat %s >>/tmp/dziennikdruku; lpr -r -P%p %s
```

Ta przykładowa opcja pozwoli stwierdzić, czy pliki są rzeczywiście dostarczane do serwera Samby. Jeśli tak, ich zawartość pojawi się w pliku */tmp/dziennikdruku*.

Drugim najpopularniejszym systemem druku po BSD jest SYSV (lub System V) i jego odmiany, jak AIX IBM-a lub HP-UX Hewletta-Packarda. W systemach tych nie ma pliku */etc/printcap*. Opcję `printcap name` można tu ustawić na odpowiednie polecenie *lpstat*, co informuje Sambę, że powinna uzyskać listę drukarek z wyników polecenia *lpstat*. Można też ustawić globalną opcję konfiguracyjną `printcap name` na nazwę samodzielnie utworzonego pliku *printcap*. Plik taki powinien zawierać linie podobne do poniższych:

```
lp|druk1|Moja drukarka 1
druk2|Moja drukarka 2
druk3|Moja drukarka 3
```

Każda linia nazywa drukarkę i określa jej aliasy. W tym przykładzie pierwsza drukarka nosi nazwy `lp`, `druk1` lub `Moja drukarka 1`, a użytkownik może posługiwać się dowolną z nich. Pierwsza nazwa będzie użyta w miejsce zmiennej `%p` w każdym poleceniu, które Samba wykona w odniesieniu do tej drukarki.

Samba obsługuje również dwa inne systemy druku: LPRNG (LPR New Generation) oraz PLP (Public Line Printer). Są to systemy typu public domain i Open Source, używane w wielu sieciach w celu rozwiązania problemów związanych z oprogramowaniem firmowym. Samba obsługuje także systemy operacyjne czasu rzeczywistego SOFTQ i QNX.

load printers

Opcja `load printers` nakazuje Sambie utworzenie udziałów dla wszystkich dostępnych drukarek i załadowanie ich na listę przeglądania. Samba utworzy i umieści na liście udział dla każdej drukarki, której nazwa występuje w pliku */etc/printcap* (lub jego systemowym odpowiedniku). Jeśli na przykład twój plik *printcap* wygląda następująco* :

```
lp:\
  :sd=/var/spool/lpd/lp:\          # katalog buforowy
  :mx#0:\                          # maksymalny rozmiar pliku (brak)
  :sh:\                            # pomijanie nag□ówków (nie)
  :lp=/dev/lp1:\                   # nazwa urz□dzenia wyj□ciowego
  :if=/var/spool/lpd/lp/filter:    # filtr tekstowy

laser:\
  :sd=/var/spool/lpd/laser:\       # katalog buforowy
  :mx#0:\                          # maksymalny rozmiar pliku (brak)
  :sh:\                            # pomijanie nag□ówków (nie)
  :lp=/dev/laser:\                 # nazwa urz□dzenia wyj□ciowego
  :if=/var/spool/lpd/lp/filter:    # filtr tekstowy
```

i użyjesz opcji:

```
load printers = yes
```

wówczas podczas uruchamiania Samby zostaną utworzone udziały drukarek `[lp]` i `[laser]`. Oba udziały zostaną skonfigurowane według opcji z sekcji `[printers]` i zostaną umieszczone na liście przeglądania serwera Samby.

printcap name

Jeśli w udziale drukarki występuje opcja `printcap name` (czasem nazywana `printcap`), Samba użyje określonego pliku jako pliku parametrów drukarek. Zwykle jest to plik */etc/printcap*. Możesz jednak ustawić tę opcję tak, aby wskazywała plik tylko z tymi drukarkami, które chcesz udostępnić w sieci. Wartością tej opcji musi być pełna ścieżka i nazwa pliku parametrów drukarek w serwerze:

```
[deskjet]
    printcap name = /usr/local/printcap
```

min print space

Opcja `min print space` określa minimalną ilość przestrzeni buforowej, która musi być dostępna na dysku, aby drukowanie było możliwe. Ustawienie jej na zero (wartość domyślna) wyłącza test; ustawienie jej na dowolną inną wartość określa ilość wymaganej przestrzeni w kilobajtach. Dzięki tej opcji można uniknąć zapełnienia przestrzeni dyskowej przez zlecenia wydruku, które mogłoby doprowadzić do błędnej pracy innych procesów.

```
[deskjet]
    min print space = 4000
```

---

* Poszczególne linie opatrzyliśmy komentarzami, aby wyjaśnić ich przeznaczenie czytelnikom, którzy dotąd nie mieli do czynienia z tym plikiem.

queuepause command

Ta opcja konfiguracyjna określa polecenie, które wstrzymuje przetwarzanie całej kolejki wydruku (w przeciwieństwie do wstrzymywania pojedynczego zlecenia). Domyślna wartość zależy od wybranego typu systemu druku. Prawdopodobnie nie będziesz musiał zmieniać tej opcji.

queueresume command

Ta opcja konfiguracyjna określa polecenie, które wznawia przetwarzanie kolejki wydruku (w przeciwieństwie do wznawiania pojedynczego zlecenia). Domyślna wartość zależy od wybranego typu systemu druku. Prawdopodobnie nie będziesz musiał zmieniać tej opcji.

Odwzorowywanie nazw w Sambie

Przed opracowaniem serwerów nazw NetBIOS-owych (NBNS), odwzorowywanie nazw opierało się wyłącznie na rozgłoszeniach. Jeśli potrzebny był adres komputera, wystarczyło rozgłosić jego nazwę w całej sieci i czekać na odpowiedź od tego komputera. Taka metoda jest nadal dostępna: szukając komputera o nazwie `franek`, można rozgłosić zapytanie i dowiedzieć się, czy taki komputer istnieje i jaki jest jego adres (użyjemy tej metody do diagnozowania usług nazewniczych Samby za pomocą polecenia `nmblookup` w rozdziale 9, *Rozwiązywanie problemów*).

Jak dowiedziałeś się w rozdziale 1, pakiety rozgłoszeniowe – czy to związane z przeglądaniem, czy rejestracją i odwzorowywaniem nazw – nie przechodzą przez granice podsieci. Co gorsze, częste rozgłoszenia obniżają efektywność sieci. Aby rozwiązać ten problem, Microsoft opracował usługę Windows Internet Name Service (WINS), czyli serwer NBNS dostępny z wielu podsieci, który Samba potrafi emulować. Dzięki temu administrator może wyznaczyć jeden z komputerów na serwer WINS, a następnie podać jego adres każdemu z klientów, które korzystają z usług odwzorowywania nazw. W rezultacie żądania rejestracji i odwzorowania nazwy mogą być kierowane do jednego komputera z dowolnego punktu sieci, a nie rozgłaszane.

WINS i rozgłoszenia nie są jednak jedynymi metodami odwzorowywania nazw. W istocie Samba dysponuje czterema różnymi mechanizmami:

- WINS,
- rozgłoszenia,
- plik */etc/hosts* lub usługi NIS/NIS+,
- plik *LMHOSTS*.

Samba może korzystać z dowolnego albo ze wszystkich tych mechanizmów, w kolejności określonej opcją `name resolve order`. Zanim jednak przedstawimy dostępne opcje konfiguracyjne, omówimy plik, z którym zapewne nie miałeś nigdy do czynienia: *LMHOSTS*.

Plik LMHOSTS

LMHOSTS to standardowy plik hostów LAN Managera, używany do odwzorowywania nazw na adresy IP w lokalnym systemie. W NBT jest on odpowiednikiem standardowego uniksowego pliku */etc/hosts*. Domyślnie plik ten jest przechowywany w katalogu */usr/local/samba/lib* i ma format podobny do pliku */etc/hosts*, na przykład:

```
192.168.220.100   hydra
192.168.220.101   feniks
```

Jedyna różnica polega na tym, że nazwy po prawej stronie wpisów są nazwami NetBIOS-owymi, a nie nazwami DNS. Ponieważ są to nazwy NetBIOS-owe, możesz przypisać im także typ zasobu:

```
192.168.220.100   hydra#20
192.168.220.100   prosta#1b
192.168.220.101   feniks#20
```

W przykładzie tym określiliśmy komputer `hydra` jako podstawowy kontroler domeny `PROSTA`, o czym informuje typ zasobu <1B> przypisany nazwie związanej z adresem IP `hydry` w drugiej linii. Pozostałe dwa wpisy określają standardowe stacje robocze.

Jeśli chcesz umieścić plik *LMHOSTS* w niestandardowym położeniu, będziesz musiał poinformować o tym proces *nmbd* podczas startu:

```
nmbd -H /etc/samba/lmhosts -D
```

Konfigurowanie Samby do współpracy z innym serwerem WINS

Możesz skonfigurować Sambę do współpracy z serwerem WINS znajdującym się w innym miejscu sieci, podając po prostu adres tego serwera. Służy do tego globalna opcja konfiguracyjna `wins server`:

```
[global]
    wins server = 192.168.200.122
```

Po dopisaniu tej opcji Samba będzie kierować wszystkie żądania WINS do serwera znajdującego się pod adresem 192.168.200.122. Ponieważ żądanie jest kierowane bezpośrednio do innego komputera, nie występują tu problemy nieodłącznie związane z metodą rozgłoszeniową. Jednakże, mimo podania adresu serwera WINS w pliku konfiguracyjnym, Samba niekoniecznie użyje go przed wypróbowaniem innych metod odwzorowywania nazw. Kolejność, w której Samba próbuje różnych metod odwzorowywania nazw, jest określona opcją `name resolve order`, o której będzie mowa za chwilę.

Jeśli serwer Samby znajduje się w podsieci używającej rozgłoszeń i zna położenie serwera WINS w innej podsieci, za pomocą opcji `wins proxy` można skonfigurować go tak, aby przekazywał wszystkie żądania odwzorowania nazwy:

```
[global]
    wins server = 192.168.200.12
    wins proxy = yes
```

Opcji tej powinieneś użyć tylko wtedy, gdy serwer WINS znajduje się w innej podsieci. W przeciwnym wypadku rozgłoszenie zostanie odebrane przez serwer WINS bez pośrednictwa Samby.

Konfigurowanie Samby jako serwera WINS

Możesz skonfigurować Sambę jako serwer WINS, ustawiając dwie globalne opcje w pliku konfiguracyjnym:

```
[global]
    wins support = yes
    name resolve order = wins lmhosts hosts bcast
```

Opcja `wins support` zamienia Sambę w serwer WINS. Choć trudno w to uwierzyć, to jest wszystko, co trzeba zrobić. Samba zajmie się wszystkimi szczegółami, dając administratorowi chwilę wytchnienia. Opcje `wins support=yes` oraz `wins server` wzajemnie się wykluczają; nie możesz uczynić Samby serwerem WINS i jednocześnie podać innego komputera jako serwera WINS.

Jeśli Samba działa jako serwer WINS, prawdopodobnie powinieneś zaznajomić się ze wspomnianą wcześniej opcją `name resolve order`. Opcja ta określa kolejność metod, których Samba używa w celu odwzorowania NetBIOS-owej nazwy. Możesz podać w niej cztery wartości:

lmhosts
: Używa pliku *LMHOSTS* LAN Managera.

hosts
: Używa standardowych metod odwzorowywania nazw w Uniksie: pliku */etc/hosts*, DNS, NIS lub ich kombinacji (zgodnie z konfiguracją systemu).

wins
: Używa serwera WINS.

bcast
: Używa metody rozgłoszeniowej.

Kolejność, w jakiej wpiszesz te wartości, będzie odpowiadać kolejności, w jakiej Samba wypróbuje różne metody odwzorowania nazwy podczas pracy w charakterze serwera WINS. Przyjrzyjmy się wartościom podanym w poprzednim przykładzie:

```
name resolve order = wins lmhosts hosts bcast
```

Oznacza to, że Samba najpierw spróbuje odwzorować nazwę na podstawie swoich wpisów WINS. Następnie skorzysta z lokalnego pliku LAN Managera, *LMHOSTS*. Później wypróbuje uniksowe metody odwzorowywania nazw, o czym informuje wartość `hosts`. Słowo to może być mylące: obejmuje nie tylko użycie pliku */etc/hosts*, ale także usług DNS lub NIS (zgodnie z konfiguracją uniksowego systemu). Wreszcie, jeśli te trzy metody nie przyniosą rezultatu, Samba spróbuje zlokalizować komputer metodą rozgłoszeniową.

Możesz też poinstruować serwer Samby działający jako serwer WINS, aby skonsultował się z systemowym serwerem DNS, jeśli nie odnajdzie żądanego hosta w swojej

bazie danych WINS. W typowym systemie linuksowym możesz znaleźć adres serwera DNS w pliku */etc/resolv.conf*. Zobaczysz tam wpis podobny do poniższego:

```
nameserver 127.0.0.1
nameserver 192.168.200.192
```

Informuje on, że serwer DNS znajduje się pod adresem 192.168.200.192 (127.0.0.1 to adres lokalnego hosta, który nigdy nie jest poprawnym adresem serwera DNS).

Użyj globalnej opcji `dns proxy`, aby Samba korzystała ze skonfigurowanego w systemie serwera DNS:

```
[global]
    wins support = yes
    name resolve order = wins lmhosts hosts bcast
    dns proxy = yes
```

Opcje konfiguracji odwzorowywania nazw

Opcje Samby związane z WINS są zebrane w tabeli 7.5.

Tabela 7.5. Opcje WINS

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`wins support`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba będzie działać jako serwer WINS	`no`	Globalny
`wins server`	Łańcuch (adres IP lub nazwa DNS)	Identyfikuje serwer WINS, którego Samba będzie używać do rejestrowania i odwzorowywania nazw	Brak	Globalny
`wins proxy`	Wartość logiczna	Umożliwia Sambie pośredniczenie w zapytaniach do serwera WINS znajdującego się w innej podsieci	`no`	Globalny
`dns proxy`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba przeszuka DNS, jeśli nie znajdzie nazwy w WINS	`no`	Globalny
`name resolve order`	`lmhosts`, `hosts`, `wins` lub `bcast`	Określa kolejność metod używanych do odwzorowywania nazw NetBIOS-owych	`lmhosts hosts wins bcast`	Globalny
`max ttl`	Wartość liczbowa	Określa maksymalny „czas życia” (w sekundach) odwzorowanych nazw NetBIOS-owych	`259200` (3 dni)	Globalny
`max wins ttl`	Wartość liczbowa	Określa maksymalny „czas życia” (w sekundach) nazw NetBIOS-owych zwracanych przez Sambę działającą jako serwer WINS	`518400` (6 dni)	Globalny
`min wins ttl`	Wartość liczbowa	Określa minimalny „czas życia” (w sekundach) nazw NetBIOS-owych zwracanych przez Sambę działającą jako serwer WINS	`21600` (6 godzin)	Globalny

wins support

Samba będzie świadczyć usługi WINS na rzecz wszystkich komputerów w sieci, jeśli dopiszesz poniższą linię do sekcji `[global]` pliku *smb.conf*:

```
[global]
    wins support = yes
```

Domyślna wartość tej opcji to `no`, co pozwala innemu komputerowi (zwykle serwerowi Windows NT) na pełnienie funkcji serwera WINS. Jeśli włączysz tę opcję, pamiętaj, że Samba działająca jako serwer WINS obecnie nie może wymieniać danych z zapasowymi serwerami WINS. Opcja ta wzajemnie wyklucza się z opcją `wins server`; jeśli obie te opcje ustawisz jednocześnie na `yes`, Samba zgłosi błąd.

wins server

Samba użyje innego dostępnego w sieci serwera WINS, jeśli umieścisz w pliku konfiguracyjnym opcję `wins server`. Wartością tej opcji może być albo adres IP, albo nazwa DNS (nie NetBIOS-owa) serwera WINS. Na przykład:

```
[global]
    wins server = 192.168.220.110
```

lub

```
[global]
    wins server = wins.przyklad.com
```

Aby opcja ta zadziałała, opcja `wins support` musi być ustawiona na `no` (wartość domyślną). W przeciwnym wypadku Samba zgłosi błąd. Za pomocą tej opcji możesz określić adres tylko jednego serwera WINS.

wins proxy

Ta opcja pozwala Sambie działać jako pośrednik innego serwera WINS, a więc przekazywać kierowane do siebie żądania rejestracji i odwzorowywania nazw do rzeczywistego serwera WINS, często znajdującego się w innej podsieci. Adres tego serwera można określić za pomocą opcji `wins server`. Samba będzie zwracać klientom odpowiedzi serwera WINS. Możesz włączyć tę opcję, dopisując poniższą linię w sekcji `[global]`:

```
[global]
    wins proxy = yes
```

dns proxy

Jeśli chcesz używać usług Domain Name Service (DNS) wtedy, gdy nazwa nie zostanie znaleziona w WINS, możesz ustawić następującą opcję:

```
[global]
    dns proxy = yes
```

Sprawi ona, że proces *nmbd* będzie wyszukiwał nazwy komputerów za pomocą standardowych usług DNS serwera. Zapewne zechcesz wyłączyć tę opcję, jeśli nie masz stałego połączenia z serwerem DNS. Mimo istnienia tej opcji, zalecamy raczej użycie serwera WINS. Jeśli w twojej sieci nie ma jeszcze serwera WINS, skonfiguruj

jeden z komputerów z Sambą jako serwer WINS. Nie możesz jednak skonfigurować dwóch komputerów z Sambą jako serwerów WINS (podstawowego i zapasowego), ponieważ obecnie Samba nie może wymieniać baz danych WINS.

name resolve order

Globalna opcja `name resolve order` określa kolejność metod używanych przez Sambę podczas odwzorowywania nazw. Domyślnie w celu ustalenia adresu komputera Samba najpierw sprawdza plik *LMHOSTS*, następnie korzysta ze standardowych metod odwzorowywania nazw w Uniksie (kombinacja */etc/hosts*, DNS i NIS), później odpytuje serwer WINS, a wreszcie używa rozgłoszeń. Możesz zmienić wartość tej opcji w następujący sposób:

```
[global]
    name resolve order = lmhosts wins hosts bcast
```

W tym przykładzie nazwy będą odwzorowywane przez sprawdzanie pliku *LMHOSTS*, odpytywanie serwera WINS, sprawdzanie systemowego pliku hostów, a wreszcie rozgłoszenia. Jeśli nie chcesz, nie musisz używać wszystkich czterech metod. Opcja ta jest opisana szczegółowo we wcześniejszym podrozdziale „Konfigurowanie Samby jako serwera WINS".

max ttl

Ta opcja określa maksymalny „czas życia" (*time to live*, TTL), przez który nazwa NetBIOS-owa zarejestrowana w serwerze Samby pozostanie aktywna. Prawdopodobnie nie będziesz musiał modyfikować tej wartości.

max wins ttl

Ta opcja określa maksymalny „czas życia" (TTL), przez który nazwa NetBIOS-owa odwzorowana przez serwer WINS pozostanie aktywna. Prawdopodobnie nie będziesz musiał modyfikować tej wartości.

min wins ttl

Ta opcja określa minimalny „czas życia" (TTL), przez który nazwa NetBIOS-owa odwzorowana przez serwer WINS pozostanie aktywna. Prawdopodobnie nie będziesz musiał modyfikować tej wartości.

8

Dodatkowe informacje o Sambie

W tym rozdziale zakończymy opis pliku konfiguracyjnego Samby, podając różnorodne opcje o najrozmaitszym przeznaczeniu. Omówimy krótko opcje służące do wspierania programistów, internacjonalizacji, przesyłania komunikatów i usuwania usterek Windows. Zwykle opcji tych będziesz używał tylko w szczególnych okolicznościach. Pod koniec rozdziału omówimy także automatyczne wykonywanie kopii zapasowych za pomocą polecenia *smbtar*. Bez dalszych wstępów zajmijmy się więc pierwszym zagadnieniem: opcjami pomocnymi przy programowaniu.

Pomoc dla programistów

Jeśli z twojego serwera Samby korzystają programiści, powinieneś zainteresować się opcjami wymienionymi w tabeli 8.1.

Tabela 8.1. Opcje konfiguracji wsparcia dla programistów

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`time server`	Wartość logiczna	Jeśli jest ustawiona na `yes`, *nmbd* będzie ogłaszał się jako serwer czasu SMB dla klientów Windows	`no`	Globalny
`time offset`	Wartość liczbowa (liczba minut)	Dodaje określoną liczbę minut do zgłaszanego czasu	`0`	Globalny
`dos filetimes`	Wartość logiczna	Pozwala zmieniać czas modyfikacji pliku użytkownikom innym niż właściciel pliku, jeśli tylko mają prawo do zapisu w tym pliku	`no`	Udział
`dos filetime resolution`	Wartość logiczna	Sprawia, że czasy modyfikacji pliku są zaokrąglane do następnej parzystej sekundy	`no`	Udział
`fake directory create times`	Wartość logiczna	Fałszuje czasy utworzenia katalogów, aby uporać się z usterką w programie *nmake* Microsoftu	`no`	Udział

Synchronizacja czasu

Synchronizacja czasu może mieć duże znaczenie dla programistów. Rozważ następujące opcje:

```
time server = yes
dos filetimes = yes
fake directory create times = yes
dos filetime resolution = yes
delete readonly = yes
```

Jeśli ustawisz te opcje, Samba będzie zwracała czasy zgodne z wymogami Visual C++, *nmake* i innych narzędzi programistycznych Microsoftu. W przeciwnym wypadku programy typu *make* dla komputerów PC będą uważały, że za każdym razem trzeba ponownie skompilować wszystkie pliki w katalogu. Oczywiście, nie jest to pożądane działanie.

time server

Jeśli twój serwer Samby ma dokładny zegar lub jeśli jest klientem jednego z uniksowych serwerów czasu, możesz poinstruować go, aby ogłaszał się jako serwer czasu SMB, ustawiając opcję `time server`:

```
[global]
    time server = yes
```

Klient będzie musiał zażądać dostarczenia mu czasu, używając poniższego polecenia DOS-a (w odpowiednim miejscu należy podstawić nazwę serwera):

```
C:\NET TIME \\serwer /YES /SET
```

Polecenie to można umieścić w skrypcie logowania Windows (patrz rozdział 6, *Użytkownicy, bezpieczeństwo i domeny*).

Domyślnie opcja `time server` jest ustawiona na `no`. Jeśli włączysz tę usługę, będziesz mógł użyć powyższego polecenia, aby zsynchronizować zegary klientów. Synchronizacja czasu jest ważna w klientach używających programów takich jak *make*, które kompilują pliki na podstawie czasu ostatniej modyfikacji. · le zsynchronizowany czas może spowodować, że albo wszystkie pliki w katalogu zostaną przekompilowane, co jest stratą czasu, albo nie zostanie skompilowany plik źródłowy, który został zmodyfikowany przed chwilą.

time offset

Aby obsłużyć klienty, które błędnie posługują się czasem letnim, Samba udostępnia opcję `time offset`. Jeśli opcja ta jest ustawiona, do bieżącego czasu dodawana jest określona liczba minut. Przydaje się to, jeśli pracujesz w Nowej Fundlandii, a Windows nie wie o obowiązującej tam 30-minutowej różnicy czasu:

```
[global]
    time offset = 30
```

dos filetimes

W systemach uniksowych tylko root i właściciel pliku mogą zmienić czas ostatniej modyfikacji pliku. Opcja udziału `dos filetimes` umożliwia Sambie naśladowa-

nie DOS-a i Windows: każdy użytkownik może zmienić czas ostatniej modyfikacji pliku, jeśli tylko ma prawo do zapisu w tym udziale. Aby to zrobić, Samba korzysta z przywilejów roota w celu zmiany znacznika czasowego pliku.

Domyślnie opcja ta jest wyłączona. Ustawienie jej na `yes` bywa niezbędne do zapewnienia poprawnej pracy programów *make* dla komputerów PC. Inaczej programy te nie będą mogły zmieniać czasów ostatniej modyfikacji plików, przez co często będą rekompilować *wszystkie* pliki, choćby nie było to potrzebne.

dos filetime resolution

Jest to opcja udziału. Jeśli jest ustawiona na `yes`, Samba będzie zaokrąglać czasy modyfikacji pliku do najbliższej wielokrotności dwóch sekund. Opcja ta służy przede wszystkim do uporania się z usterką w Windows, przez którą Visual C++ nie dostrzega, że plik nie został zmodyfikowany. Możesz włączyć ją w następujący sposób:

```
[dane]
    dos filetime resolution = yes
```

Zalecamy użycie tej opcji tylko wtedy, gdy Visual C++ używa plików w udziale korzystającym z blokad oportunistycznych.

fake directory create times

Opcja `fake directory create times` pozwala na poprawną pracę programów *make* dla komputerów PC. Systemy VFAT i NTFS zapisują czas utworzenia katalogu, a Unix nie. Bez tej opcji Samba pobiera najwcześniejszą zarejestrowaną datę dla katalogu (często jest to data ostatniej modyfikacji pliku) i zwraca ją klientowi. Jeśli to nie wystarcza, ustaw poniższą opcję w definicji udziału:

```
[dane]
    fake directory create times = yes
```

Jeśli opcja ta jest ustawiona, Samba będzie zgłaszać czas utworzenia katalogu równy zakodowanej na stałe wartości 1 stycznia 1980 roku. Służy głównie do przekonania programu *nmake* z pakietu Visual C++, że pliki obiektowe w katalogach budowy zostały rzeczywiście utworzone później niż sam katalog i wymagają ponownej kompilacji.

Magiczne skrypty

Poniższe opcje służą do obsługi *magicznych skryptów* w serwerze Samby. Magiczne skrypty umożliwiają wykonywanie programów w Uniksie i zwracanie ich wyników klientom SMB. Mają one charakter zarazem eksperymentalny i prowizoryczny. Mimo to niektórzy użytkownicy wciąż potrzebują tych dwóch opcji do zapewnienia poprawnej pracy swoich programów. Magiczne skrypty nie cieszą się większym zaufaniem, a zespół twórców Samby stanowczo odradza ich użycie. Więcej informacji znajdziesz w tabeli 8.2.

Tabela 8.2. Opcje konfiguracji magicznych skryptów

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`magic script`	Łańcuch (nazwa pliku wraz ze ścieżką)	Ustawia nazwę pliku, który zaraz po zamknięciu powinien zostać wykonany przez Sambę z przywilejami zalogowanego użytkownika.	Brak	Udział
`magic output`	Łańcuch (nazwa pliku wraz ze ścieżką)	Ustawia nazwę pliku, w którym zostaną zapisane wyniki magicznego skryptu	*nazwaskryptu.out*	Udział

magic script

Jeśli w opcji `magic script` podasz nazwę pliku, a klient utworzy w udziale plik o takiej nazwie, Samba wykona ten plik po jego otwarciu i zamknięciu przez użytkownika. Załóżmy, że w udziale `[ksiegowosc]` umieszczono poniższą opcję:

```
[ksiegowosc]
    magic script = rejestr.sh
```

Samba będzie nieprzerwanie monitorować pliki w tym udziale. Gdy plik o nazwie `rejestr.sh` zostanie zamknięty przez użytkownika (po uprzednim otwarciu), Samba wykona lokalnie polecenia zawarte w tym pliku. Plik zostanie przekazany do wykonania powłoce; musi zatem być poprawnym skryptem powłoki Uniksa. Oznacza to, że linie skryptu powinny kończyć się znakiem nowej linii, a nie używaną w Windows kombinacją powrót karetki – nowa linia. Na początku pliku warto też dodać dyrektywę `#!`, aby wskazać powłokę, która powinna wykonać skrypt.

magic output

Ta opcja określa plik wyjściowy, do którego skrypt wymieniony w opcji `magic script` będzie wysyłał swoje dane. Plik ten musi znajdować się w zapisywalnym katalogu:

```
[ksiegowosc]
    magic script = rejestr.sh
    magic output = /var/log/magicznewyniki
```

Jeśli pominiesz tę opcję, domyślny plik wyjściowy będzie miał tę samą nazwę co skrypt (określony opcją `magic script`), ale z dołączonym rozszerzeniem *.out*.

Internacjonalizacja

Samba posiada ograniczoną zdolność posługiwania się obcymi językami: jeśli musisz posługiwać się znakami spoza standardowego zbioru ASCII, pomogą ci w tym opcje wymienione w tabeli 8.3. Jeśli nie masz takiej potrzeby, możesz pominąć ten podrozdział.

Tabela 8.3. Opcje obsługi języków narodowych

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`client code page`	Opisane w tym podrozdziale	Ustawia stronę kodową, którą posługują się klienty	850	Globalny
`character set`	Opisane w tym podrozdziale	Tłumaczy strony kodowe na alternatywne zestawy znaków Uniksa	Brak	Globalny
`coding system`	Opisane w tym podrozdziale	Tłumaczy stronę kodową 932 na azjatycki zestaw znaków	Brak	Globalny
`valid chars`	Łańcuch (zbiór znaków)	Przestarzała; poprzednio dodawała pojedyncze znaki do strony kodowej i należało jej używać już po ustawieniu strony kodowej klienta	Brak	Globalny

client code page

Zestawy znaków w systemach Windows biorą swój początek z pierwotnej koncepcji *stron kodowych*. Strony kodowe są używane przez klienty DOS-a i Windows w celu ustalenia reguł odwzorowywania liter małych na duże. Samba może używać różnych stron kodowych dzięki opcji `client code page`, którą należy ustawić tak, aby odpowiadała stronie kodowej klienta. Opcja ta powoduje załadowanie pliku z definicją strony kodowej i może przybierać wartości wymienione w tabeli 8.4.

Tabela 8.4. Strony kodowe dostępne w Sambie 2.0

Strona kodowa	*Definicja*
`437`	MS-DOS Latin (Stany Zjednoczone)
`737`	Windows 95 grecki
`850`	MS-DOS Latin 1 (zachodnioeuropejski)
`852`	MS-DOS Latin 2 (wschodnioeuropejski)
`861`	MS-DOS islandzki
`866`	MS-DOS cyrylica (rosyjski)
`932`	MS-DOS japoński Shift-JIS
`936`	MS-DOS uproszczony chiński
`949`	MS-DOS koreański Hangul
`950`	MS-DOS tradycyjny chiński

Możesz ustawić stronę kodową klienta w następujący sposób:

```
[global]
    client code page = 852
```

Domyślna wartość tej opcji to 850. Możesz użyć narzędzia *make_smbcodepage* dostarczanego wraz z Sambą (domyślnie w katalogu */usr/local/samba/bin*) w celu utworzenia własnych stron kodowych SMB, gdyby te wymienione w tabeli 8.4 okazały się niewystarczające.

character set

Globalna opcja `character set` służy do przekształcania nazw plików udostępnianych przez dosową stronę kodową (patrz poprzedni podrozdział, „client code page") na odpowiedniki reprezentowane przez zestawy znaków Uniksa inne niż te, które są używane w Stanach Zjednoczonych. Jeśli na przykład chciałbyś tłumaczyć zachodnioeuropejskie znaki MS-DOS-a na zachodnioeuropejskie znaki Uniksa, mógłbyś dopisać poniższe opcje do pliku konfiguracyjnego:

```
[global]
    client code page = 850
    character set = IOS8859-1
```

Zauważ, że musisz użyć opcji `client code page`, aby określić przekształcany zestaw znaków. Zestawy znaków (i odpowiadające im strony kodowe) dostępne w Sambie 2.0 są wymienione w tabeli 8.5:

Tabela 8.5. Zestawy znaków dostępne w Sambie 2.0

Zestaw znaków	*Strona kodowa*	*Opis*
`ISO8859-1`	`850`	Zachodnioeuropejski zestaw znaków Uniksa
`ISO8859-2`	`852`	Wschodnioeuropejski zestaw znaków Uniksa
`ISO8859-5`	`866`	Rosyjski zestaw znaków Uniksa (cyrylica)
`KOI8-R`	`866`	Alternatywny rosyjski zestaw znaków Uniksa (cyrylica)

Opcja `character set` jest domyślnie wyłączona.

coding system

Opcja `coding system` przypomina opcję `character set`. Służy ona jednak do określenia, jak należy przekształcić japońską stronę kodową Shift JIS na odpowiedni zestaw znaków Uniksa. Aby użyć tej opcji, musisz najpierw ustawić opisaną wcześniej opcję `client code page` na 932. Systemy kodowania akceptowane przez Sambę 2.0 są wymienione w tabeli 8.6.

Tabela 8.6. Parametry systemu kodowania w Sambie 2.0

Zestaw znaków	*Opis*
`SJIS`	Standardowy system Shift JIS
`JIS8`	Ośmiobitowe kody JIS
`J8BB`	Ośmiobitowe kody JIS
`J8BH`	Ośmiobitowe kody JIS
`J8@B`	Ośmiobitowe kody JIS
`J8@J`	Ośmiobitowe kody JIS
`J8@H`	Ośmiobitowe kody JIS
`JIS7`	Siedmiobitowe kody JIS

Zestaw znaków	*Opis*
J7BB	Siedmiobitowe kody JIS
J7BH	Siedmiobitowe kody JIS
J7@B	Siedmiobitowe kody JIS
J7@J	Siedmiobitowe kody JIS
J7@H	Siedmiobitowe kody JIS
JUNET	Kody JUNET
JUBB	Kody JUNET
JUBH	Kody JUNET
JU@B	Kody JUNET
JU@J	Kody JUNET
JU@H	Kody JUNET
EUC	Kody EUC
HEX	Trzybajtowe kody szesnastkowe
CAP	Trzybajtowe kody szesnastkowe (Columbia Appletalk Program)

valid chars

Opcja `valid chars` to starsza funkcja Samby, pozwalająca na dodanie pojedynczych znaków do strony kodowej. Opcja ta jednak jest wypierana przez nowocześniejsze systemy kodowania. Możesz używać jej następująco:

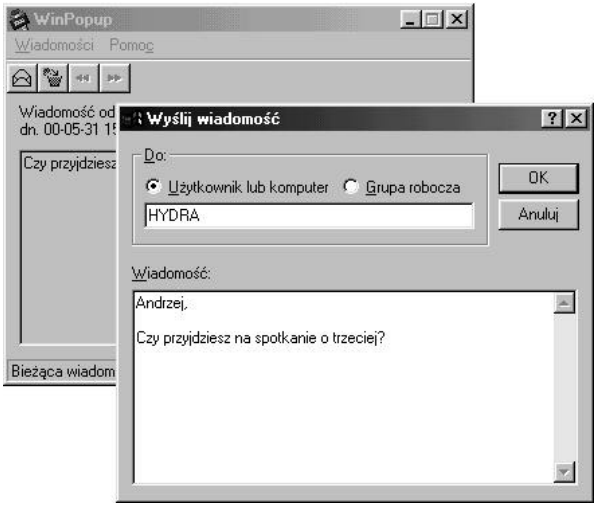
```
valid chars = Î
valid chars = 0450:0420 0x0A20:0x0A00
valid chars = A:a
```

Wszystkie znaki na liście powinny być oddzielone spacjami. Jeśli między dwoma znakami lub ich liczbowymi odpowiednikami znajduje się dwukropek, znak po lewej stronie jest uważany za duży, a po prawej – za mały. Możesz podać znaki albo dosłownie (jeśli możesz je wpisać z klawiatury), albo używając ich odpowiedników ósemkowych, szesnastkowych lub dziesiętnych w Unikodzie.

Nie zalecamy używania tej opcji. Zamiast niej lepiej stosować jedną z wymienionych wcześniej standardowych stron kodowych. Jeśli jednak korzystasz z tej opcji, musisz wpisać ją po definicji strony kodowej klienta (`client code page`), do której chcesz dodać znak. W przeciwnym wypadku znaki nie zostaną dodane.

Komunikaty WinPopup

Narzędzie WinPopup (*WINPOPUP.EXE*) w Windows służy do wysyłania komunikatów do użytkowników, komputerów lub całych grup roboczych. Narzędzie to wchodzi w skład Windows 95 OSR2 i jest standardowym elementem Windows 98. Zarówno w Windows 95, jak i 98 musisz uruchomić program WinPopup, aby wysyłać i odbierać komunikaty. W Windows NT możesz odbierać je bez uruchamiania żadnego narzędzia; będą pojawiać się automatycznie w niewielkim oknie dialogowym na ekranie. Okno aplikacji WinPopup jest pokazane na rysunku 8.1.



Rysunek 8.1. Aplikacja WinPopup

Samba dysponuje tylko jedną opcją związaną z komunikatami WinPopup, `message command` (patrz tabela 8.7).

Tabela 8.7. Opcja konfiguracji WinPopup

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`message command`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa polecenie, które należy wykonać w Uniksie po odebraniu komunikatu WinPopup	Brak	Globalny

message command

Opcja `message command` określa ścieżkę do polecenia, które zostanie wykonane w serwerze Samby po odebraniu komunikatu WinPopup. Polecenie to zostanie wykonane z przywilejami użytkownika określonego opcją `guest account`. Prawdę mówiąc, nie bardzo wiadomo, co robić z komunikatami, ponieważ prawdopodobnie są przeznaczone dla administratora Samby, a Samba nie zna jego nazwiska. Jeśli wiesz, że przy konsoli serwera siedzi użytkownik, możesz użyć polecenia sugerowanego przez zespół twórców Samby:

```
[global]
    message command = /bin/csh -c 'xedit %s; rm %s' &
```

Zwróć uwagę na użycie zmiennych. Zmienna `%s` określa plik, w którym znajduje się komunikat. Plik ten powinien zostać usunięty, kiedy polecenie zakończy pracę, ponieważ w przeciwnym razie w serwerze gromadziłyby się pliki z komunikatami. Oprócz tego polecenie musi rozwidlić swój proces (zauważ, że na końcu polecenia jest znak `&`); w przeciwnym wypadku klient mógłby się zawiesić i czekać na powiadomienie o pomyślnym przesłaniu wiadomości przed wznowieniem pracy.

Oprócz standardowych zmiennych w opcji `message command` możesz użyć trzech dodatkowych zmiennych, wymienionych w tabeli 8.8.

Tabela 8.8. Zmienne w opcji message command

Zmienna	*Opis*
`%s`	Nazwa pliku z komunikatem
`%f`	Nazwa klienta, który przesłał komunikat
`%t`	Nazwa komputera, który jest adresatem komunikatu

Nowe opcje Samby

Samba ma kilka opcji, które pojawiły się w wersji 2.0 i których działanie nie jest w pełni gwarantowane. W tym podrozdziale omówimy krótko ich przeznaczenie. Opcje te zebrano w tabeli 8.9.

Tabela 8.9. Nowe opcje Samby

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`change notify timeout`	Wartość liczbowa (sekundy)	Ustawia odstęp czasu między kolejnymi testami, wykonywanymi wtedy, gdy klient prosi o powiadomienie o zmianie zawartości określonego katalogu	`60`	Globalny
`machine password timeout`	Wartość liczbowa (sekundy)	Ustawia okres odnowienia hasła komputera w domenie NT	`604 800` (1 tydzień)	Globalny
`stat cache`	Wartość logiczna	Jeśli jest ustawiona na `yes`, Samba będzie buforować odwzorowania nazw	`yes`	Globalny
`stat cache size`	Wartość liczbowa	Ustawia rozmiar bufora odwzorowań	`50`	Globalny

change notify timeout

Globalna opcja `change notify timeout` emuluje mechanizm SMB Windows NT znany jako *powiadamianie o zmianie*. Dzięki niemu klienty mogą zażądać, aby serwer NT okresowo monitorował wskazany katalog w udziale i sprawdzał, czy nie zaszły w nim jakieś zmiany. Jeśli coś by się zmieniło, serwer ma powiadomić o tym klienta.

Od wersji 2.0 Samba może świadczyć taką usługę swoim klientom. Zbyt częste testy mogą jednak znacznie spowolnić działanie serwera. Opcja ta ustawia odstęp między kolejnymi testami. Jej domyślna wartość to jedna minuta (60 sekund), możesz jednak zmienić czas oczekiwania:

```
[global]
    change notify timeout = 30
```

machine password timeout

Globalna opcja `machine password timeout` ustawia czas ważności hasła komputera w domenie NT. Jej domyślna wartość jest obecnie równa tej używanej przez

Windows NT: 604 800 sekund (1 tydzień). Samba okresowo próbuje zmienić *hasło konta komputera*, czyli hasło używane przez inny serwer specjalnie w celu zgłaszania zmian. Opcja ta określa liczbę sekund, po upłynięciu których Samba spróbuje zmienić to hasło. W poniższym przykładzie czas oczekiwania na zmianę hasła jest ustawiany na jeden dzień:

```
[global]
    machine password timeout = 86400
```

stat cache

Globalna opcja `stat cache` włącza buforowanie odwzorowań nazw bez odróżniania małych i dużych liter. Jej domyślna wartość to `yes`. Zespół programistów Samby odradza zmienianie tego parametru.

stat cache size

Globalna opcja `stat cache size` ustawia liczbę wpisów w buforze odwzorowań. Jej domyślna wartość to 50. Zespół programistów Samby odradza zmienianie tego parametru.

Opcje różne

Wiele opcji Samby jest związanych ze specyfiką systemu operacyjnego – Uniksa albo Windows. Opcje wymienione w tabeli 8.10 rozwiązują niektóre ze znanych problemów. Zwykle nie zmieniamy ich domyślnych wartości i to samo zalecamy tobie.

Tabela 8.10. Opcje różne

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`deadtime`	Wartość liczbowa (liczba minut)	Określa w minutach czas bezczynności, po którym połączenie powinno zostać zakończone	`0`	Globalny
`dfree command`	Łańcuch (polecenie)	Określa polecenie, które zwraca ilość wolnej przestrzeni dyskowej w formacie rozpoznawanym przez Sambę	Brak	Globalny
`fstype`	`NTFS`, `FAT` lub `Samba`	Ustawia typ systemu plików, który serwer zgłasza klientom	`NTFS`	Globalny
`keep alive`	Wartość liczbowa (sekundy)	Ustawia liczbę sekund między kolejnymi testami nieaktywności klienta	`0` (brak)	Globalny
`max disk size`	Wartość liczbowa (rozmiar w MB)	Ustawia największy rozmiar dysku zwracany klientom (które mogą mieć ograniczenia). Nie ma wpływu na rzeczywiste operacje dyskowe	`0` (nieskończoność)	Globalny

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`max mux`	Wartość liczbowa	Ustawia maksymalną liczbę jednoczesnych operacji SMB, które mogą wykonywać klienty	`50`	Globalny
`max open files`	Wartość liczbowa	Ustawia maksymalną liczbę otwartych plików poniżej uniksowego limitu	`10 000`	Globalny
`max xmit`	Wartość liczbowa	Określa maksymalny rozmiar pakietów wysyłanych przez Sambę	`65 535`	Globalny
`nt pipe support`	Wartość logiczna	Wyłącza eksperymentalną funkcję NT; używana w celach pomiarowych albo w razie wystąpienia błędu	`yes`	Globalny
`nt smb support`	Wartość logiczna	Wyłącza eksperymentalną funkcję NT; używana w celach pomiarowych albo w razie wystąpienia błędu	`yes`	Globalny
`ole locking compatibility`	Wartość logiczna	Przekształca wychodzące poza zakres żądania blokady używane w Windows tak, aby w Uniksie mieściły się w dozwolonym zakresie. Wyłączenie tej opcji powoduje błędy blokad w Uniksie	`yes`	Globalny
`panic action`	Łańcuch (polecenie)	Program uruchamiany w razie błędu serwera Samby; używana w celach diagnostycznych	Brak	Globalny
`set directory`	Wartość logiczna	Jeśli jest ustawiona na `yes`, pozwala klientom VMS na wydawanie poleceń `set dir`	`no`	Globalny
`smbrun`	Łańcuch (nazwa polecenia wraz ze ścieżką)	Ustawia polecenie, za pośrednictwem którego Samba wywołuje polecenia powłoki	Brak	Globalny
`status`	Wartość logiczna	Jeśli jest ustawiona na `yes`, pozwala Sambie na monitorowanie statusu na potrzeby polecenia `smbstatus`	`yes`	Globalny
`strict sync`	Wartość logiczna	Jeśli jest ustawiona na `no`, ignoruje prośby aplikacji Windows o natychmiastowe zapisanie danych na dysku	`no`	Globalny
`sync always`	Wartość logiczna	Jeśli jest ustawiona na `yes`, wymusza zapisanie wszystkich danych klienta na dysku przed powrotem z wywołania	`no`	Globalny
`strip dot`	Wartość logiczna	Jeśli jest ustawiona na `yes`, usuwa końcowe kropki z uniksowych nazw plików	`no`	Globalny

deadtime

Ta globalna opcja ustawia liczbę minut, przez które Samba będzie czekać na bezczynnego klienta, zanim zamknie jego sesję z serwerem. Klienta uważa się za bezczynnego, jeśli nie ma on żadnych otwartych plików i nie są do niego przesyłane żadne dane. Domyślna wartość tej opcji to `0`, co oznacza, że Samba nigdy nie zamyka żadnych połączeń, bez względu na to, jak długo pozostają bezczynne. Możesz zmienić to w następujący sposób:

```
[global]
    deadtime = 10
```

Opcja ta nakazuje Sambie zakończenie bezczynnych sesji z klientem po 10 minutach. W większości sieci takie ustawienie tej opcji będzie odpowiednie, ponieważ ponowne połączenia klientów z serwerem są zwykle wykonywane w sposób niewidoczny dla użytkownika.

dfree command

Ta opcja globalna jest używana w systemach, które niepoprawnie określają wolną przestrzeń pozostałą na dysku. Jak dotąd, jedynym systemem, który wymaga użycia tej opcji, jest Ultrix. Opcja ta nie ma wartości domyślnej, co oznacza, że Samba wie sama, jak obliczyć rozmiar wolnej przestrzeni dyskowej, a uzyskane przez nią wyniki są uważane za wiarygodne. Możesz ustawić wartość tej opcji następująco:

```
[global]
    dfree command = /usr/local/bin/dfree
```

Opcja ta powinna wskazywać na skrypt, który zwraca całkowitą ilość przestrzeni dyskowej zajmowanej przez blok oraz liczbę dostępnych bloków. Dokumentacja Samby zaleca użycie następującego skryptu:

```
#!/bin/sh
df $1 | tail -1 | awk '{print $2" "$4}'
```

W Uniksach typu System V zadziała poniższy skrypt:

```
#!/bin/sh
/usr/bin/df $1 | tail -1 | awk '{print $3" "$5}'
```

fstype

Ta opcja udziału ustawia typ systemu plików, który zgłasza Samba pytającym o to klientom. Istnieją trzy łańcuchy, które mogą stanowić wartość tej opcji konfiguracyjnej (patrz tabela 8.11).

Tabela 8.11. Typy systemów plików

Zmienna	*Opis*
NTFS	System plików NT Microsoft Windows
FAT	System plików FAT DOS-a
Samba	System plików Samby

Domyślna wartość tej opcji to NTFS, co reprezentuje system plików Windows NT. Prawdopodobnie nie ma potrzeby określania innego typu systemu plików. Jeśli jednak zechcesz, możesz zmienić go dla każdego udziału z osobna, jak w przykładzie poniżej:

```
[dane]
    fstype = FAT
```

keep alive

Ta opcja globalna określa liczbę sekund, przez które Samba czeka między wysyłaniem NetBIOS-owych *pakietów podtrzymujących połączenie* (*keep-alive packets*). Domyślna wartość tej opcji to `0`, co oznacza, że Samba w ogóle nie wysyła takich pakietów. Możesz zmienić to w następujący sposób:

```
[global]
    keep alive = 10
```

max disk size

Ta opcja globalna określa iluzoryczny limit rozmiaru (w megabajtach) każdego udziału używanego przez Sambę. Zwykle opcję tę ustawia się po to, aby klienty ze starszymi systemami operacyjnymi nie miały kłopotów z przetwarzaniem dużych rozmiarów dysków, na przykład przekraczających jeden gigabajt.

Domyślna wartość tej opcji to `0`, co oznacza, że nie ma żadnego górnego limitu. Możesz zmienić to w następujący sposób:

```
[global]
    max disk size = 1000
```

max mux

Ta opcja globalna określa maksymalną liczbę jednoczesnych operacji SMB dozwolonych przez Sambę. Domyślna wartość tej opcji to `50`. Możesz zmienić ją w następujący sposób:

```
[global]
    max mux = 100
```

max open files

Ta opcja globalna określa maksymalną liczbę plików, które mogą być jednocześnie otwarte przez wszystkie procesy Samby. Wartość ta musi być mniejsza lub równa limitowi narzucanemu przez system operacyjny. Domyślna wartość tej opcji to `10 000`. Możesz zmienić ją w następujący sposób:

```
[global]
     max open files = 8000
```

max xmit

Ta opcja globalna określa maksymalny rozmiar pakietów, które Samba wymienia z klientami. W niektórych przypadkach ustawienie mniejszego rozmiaru pakietu

może zwiększyć wydajność, zwłaszcza podczas współpracy z Windows for Workgroups. Domyślna wartość tej opcji to `65 535`. Możesz zmienić ją następująco:

```
[global]
    max xmit = 4096
```

Niektóre zastosowania tej opcji znajdziesz w podrozdziale „Okno odbioru TCP" w dodatku B, *Optymalizowanie wydajności Samby*.

nt pipe support

Ta opcja globalna jest używana przez twórców Samby, aby zabronić klientom Windows NT łączenia się z charakterystycznymi dla NT potokami IPC$ lub na to zezwolić. Jako użytkownik Samby nie powinieneś zmieniać wartości domyślnej:

```
[global]
    nt pipe support = yes
```

nt smb support

Ta opcja globalna jest używana przez twórców Samby, aby negocjować charakterystyczne dla NT opcje SMB z klientami Windows NT. Zespół Samby odkrył, że ustawienie tej opcji na `no` daje odrobinę lepszą wydajność, ale jako użytkownik Samby raczej nie powinieneś zmieniać wartości domyślnej:

```
[global]
    nt smb support = yes
```

ole locking compatibility

Ta opcja globalna włącza lub wyłącza wewnętrzne manipulacje blokadami zakresów bajtów w Sambie, które dają zgodność z aplikacjami OLE (*Object Linking and Embedding*), używającymi blokad wysokich zakresów bajtów jako metody komunikacji międzyprocesowej. Domyślna wartość tej opcji to `yes`. Jeśli ufasz uniksowym mechanizmom blokującym, możesz zmienić wartość tej opcji w następujący sposób:

```
[global]
    ole locking compatibility = no
```

panic action

Ta opcja globalna określa polecenie, które należy wykonać, kiedy podczas pracy lub uruchamiania Samby wystąpi poważny błąd. Opcja ta nie ma wartości domyślnej. Możesz zdefiniować czynność wykonywaną w takiej sytuacji:

```
[global]
    panic action = /bin/sh -c
          'xedit < "Samba nieoczekiwanie zako□czy□a prac□!'
```

set directory

Ta opcja udziału umożliwia klientom Digital Pathworks korzystanie z polecenia `setdir` w celu zmiany katalogu w serwerze. Jeśli nie używasz klientów Digital Pathworks, nie powinieneś zmieniać tej opcji. Jej domyślna wartość to `no`. Możesz zmienić ją dla każdego udziału z osobna w następujący sposób:

```
[dane]
    set directory = yes
```

smbrun

Ta opcja ustala położenie pliku wykonywalnego *smbrun*, za pośrednictwem którego Samba wywołuje polecenia powłoki. Jej domyślna wartość jest automatycznie ustalana podczas kompilacji Samby. Jeśli nie zainstalowałeś Samby w standardowym katalogu, możesz określić położenie pliku wykonywalnego *smbrun* w następujący sposób:

```
[global]
    smbrun = /usr/local/bin/smbrun
```

status

Ta opcja globalna wskazuje, czy Samba powinna rejestrować wszystkie aktywne połączenia w pliku statusu. Plik ten jest używany tylko przez polecenie *smbstatus*. Jeśli nie zamierzasz używać tego polecenia, możesz ustawić tę opcję na `no`, co nieznacznie zwiększy szybkość serwera. Jej domyślna wartość to `yes`. Możesz zmienić ją następująco:

```
[global]
    status = no
```

strict sync

Ta opcja udziału określa, czy Samba będzie spełniać wszystkie żądania zsynchronizowania dysku zgłaszane przez klienta. Klienty często żądają synchronizacji dysku, kiedy po prostu próbują wymusić zapisanie danych we własnych otwartych plikach, przez co znacznie spowalniają pracę serwera Samby. Domyślna wartość tej opcji to `no`. Możesz zmienić ją następująco:

```
[dane]
    strict sync = yes
```

sync always

Ta opcja udziału określa, czy po każdym zapisie należy wykonać synchronizację dysku, zanim wywołanie zapisu zwróci sterowanie klientowi. Jeśli nawet opcja ta jest ustawiona na `no`, klient może zażądać synchronizacji dysku (patrz opcja `strict sync` powyżej). Domyślna wartość tej opcji to `no`. Możesz zmienić ją dla każdego udziału z osobna w następujący sposób:

```
[dane]
    sync always = yes
```

strip dot

Ta opcja globalna określa, czy należy usuwać końcową kropkę z uniksowych nazw plików. Domyślna wartość tej opcji to `no`. Możesz zmienić ją dla każdego udziału z osobna w następujący sposób:

```
[global]
    strip dot = yes
```

Opcja ta jest uważana za przestarzałą; powinieneś zamiast niej używać opcji `mangled map` (patrz rozdział 5, podrozdział „Przekształcanie nazw i wielkość liter").

Tworzenie kopii zapasowych za pomocą programu smbtar

Ostatnim punktem tego rozdziału jest narzędzie *smbtar*. W nowoczesnych komputerach PC istnieje pewna niedogodność: dyskietki, a często także CD-ROM-y nie są dość pojemne, by pomieścić zapasowe kopie dysków, a kupowanie stacji taśm dla każdego komputera nie jest najlepszym pomysłem. W rezultacie często w ogóle nie sporządza się kopii dysków komputerów PC, a w razie awarii ponownie instaluje się system z dyskietek i CD-ROM-ów.

Na szczęście Samba oferuje nam inną opcję: sporządzanie zapasowych kopii danych za pomocą narzędzia *smbtar*. Możesz to robić regularnie, jeśli przechowujesz dane użytkowników w serwerze Samby, albo tylko okazjonalnie, aby zapisać lokalne aplikacje oraz pliki konfiguracyjne i w ten sposób przyspieszyć ponowną instalację.

Aby sporządzić zapasową kopię danych komputera PC z uniksowego serwera, musisz wykonać następujące czynności:

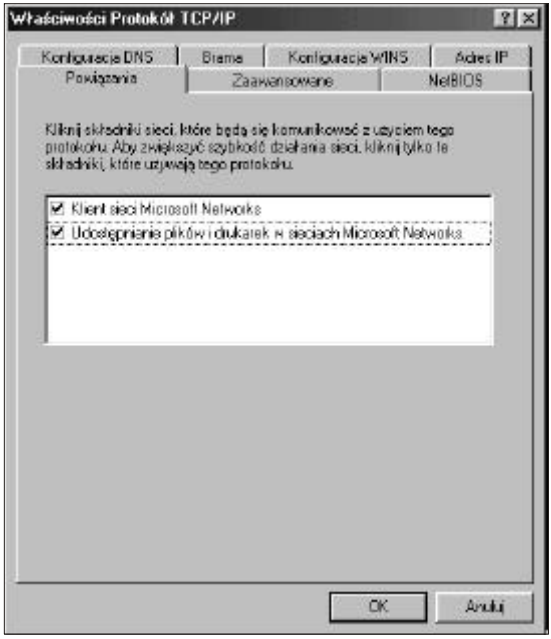
1. Upewnić się, że w komputerze PC zainstalowana jest usługa Udostępnianie plików i drukarek w sieciach Microsoft Networks i że jest ona powiązana z protokołem TCP/IP.
2. Jawnie udostępnić dysk w komputerze PC, aby serwer mógł go odczytać.
3. Utworzyć w serwerze skrypty kopiujące dane.

Pierwsze dwie czynności przeprowadzimy w Windows 95/98. Otwórz z Panelu sterowania okno Sieć i sprawdź, czy w górnym okienku widnieje usługa Udostępnianie plików i drukarek w sieciach Microsoft Networks (patrz rysunek 8.2).

Rysunek 8.2. Okno Sieć

Jeśli usługa ta nie jest zainstalowana, możesz to zrobić, klikając przycisk Dodaj w oknie Sieć. Będziesz musiał wybrać typ instalowanego składnika sieci. Wybierz pozycję Usługa i kliknij przycisk Dodaj. Zostaniesz poproszony o wskazanie producenta i usługi. Zaznacz pozycję Udostępnianie plików i drukarek w sieciach Microsoft Networks i kliknij przycisk Zakończ, aby zainstalować usługę.

Kiedy zainstalujesz usługę udostępniania plików i drukarek, wróć do okna Sieć i zaznacz protokół TCP/IP powiązany z kartą sieciową. Następnie kliknij przycisk Właściwości i wybierz kartę Powiązania. Powinieneś zobaczyć okno dialogowe podobne do tego z rysunku 8.3. Musisz upewnić się, że zaznaczone jest pole Udostępnianie plików i drukarek w sieciach Microsoft Networks, które daje usłudze dostęp do protokołu TCP/IP. Od tej chwili możesz współdzielić pliki i drukarki z innymi komputerami w sieci.

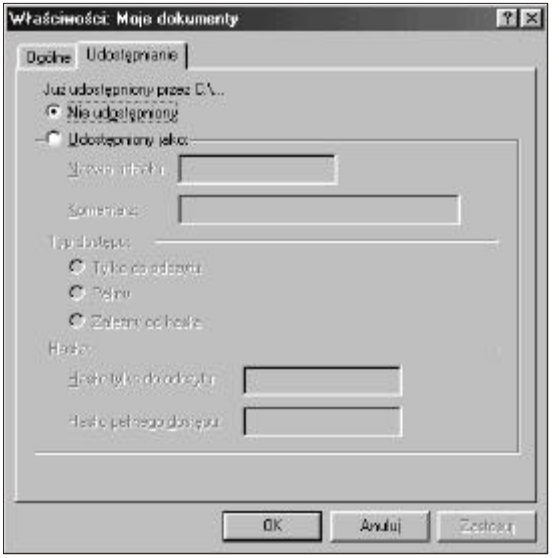


Rysunek 8.3. Powiązania TCP/IP

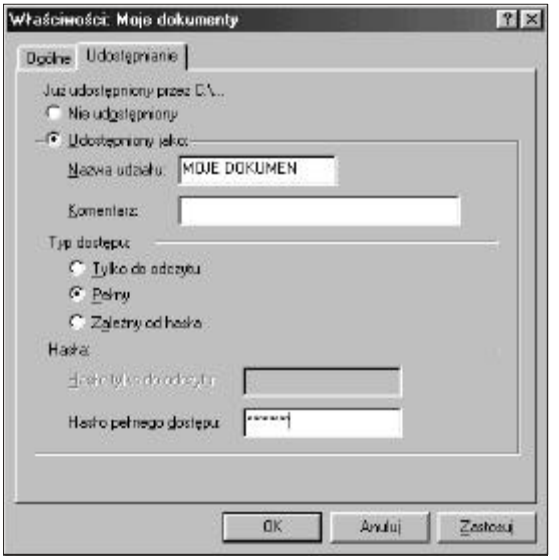
Następnie serwerowi taśm należy udostępnić dysk z przeznaczonymi do skopiowania danymi. Otwórz okno Mój komputer i wybierz na przykład katalog Moje dokumenty. Następnie kliknij jego ikonę prawym przyciskiem myszy i wybierz z menu polecenie Właściwości. Powinieneś zobaczyć okno dialogowe podobne do tego z rysunku 8.4.

Wybierz kartę Udostępnianie i włącz udostępnianie plików. Możesz wybrać, czy chcesz udostępnić dysk w trybie tylko do odczytu, do odczytu i zapisu (pełnym), czy też zależnym od hasła. To okno dialogowe pochodzi z Windows 95/98, więc oferuje tylko zabezpieczenia na poziomie udziału. W tym przykładzie wybraliśmy pełny typ dostępu i ustawiliśmy hasło (patrz rysunek 8.5). Kiedy wpiszesz hasło i klikniesz

OK, zostaniesz poproszony o jego ponowne wprowadzenie. Na tym kończy się drugi etap.



Rysunek 8.4. Okno Właściwości: Moje dokumenty



Rysunek 8.5. Udostępniony folder Moje dokumenty

Ostatnim etapem jest napisanie w serwerze taśm skryptu wykorzystującego program *smbtar*. Najprostszy skrypt składa się z jednej linii i wygląda mniej więcej tak:

```
smbtar -s klient -t /dev/rst0 -x "Moje dokumenty" -p hasło
```

Powoduje on bezwarunkowe zapisanie udziału *//klient/Moje dokumenty* w urządzeniu */dev/rst0*. Oczywiście, skrypt ten jest zbyt uproszczony i nie dość bezpieczny. Czynności wykonywane przez rzeczywisty skrypt zależą od obowiązującej procedury tworzenia kopii zapasowych.

Aby jednak zaostrzyć twój apetyt, podajemy kilka możliwości programu *smbtar*:

- Przyrostowe kopiowanie plików z użyciem dosowego bitu archiwalnego (opcja `-i`). Wymaga to dostępu do udziału klienta w trybie do odczytu i zapisu, aby program *smbtar* mógł zerować bity archiwalne.
- Kopiowanie tylko tych plików, które zostały zmodyfikowane po określonej dacie (opcja `-N` *`nazwapliku`*).
- Kopiowanie całych dysków komputerów PC, na przykład przez udostępnienie całego dysku C: lub D: i sporządzenie jego kopii.

Z wyjątkiem pierwszego przykładu, czynności te można wykonać w trybie tylko do odczytu, co zmniejsza ryzyko związane z przechowywaniem haseł w skryptach i przekazywaniem ich w linii polecenia.

9

Rozwiązywanie problemów

Samba jest niezwykle niezawodna. Kiedy już odpowiednio ją skonfigurujesz, prawdopodobnie zapomnisz, że w ogóle działa. Problemy występują najczęściej podczas instalacji lub dodawania nowych funkcji do serwera. Na szczęście istnieje wiele narzędzi, które umożliwiają szybkie postawienie diagnozy. Nie możemy szczegółowo opisać rozwiązania każdego możliwego problemu, ale wskazówki zawarte w tym rozdziale z pewnością nie raz ci pomogą.

W pierwszym podrozdziale opisujemy „skrzynkę z narzędziami" – zbiór zasobów ułatwiających diagnozowanie Samby. W drugim podrozdziale zamieszczamy szczegółowy poradnik, a w trzecim wymieniamy dodatkowe źródła informacji, które mogą pomóc w rozwiązaniu szczególnie dotkliwych problemów.

Skrzynka z narzędziami

Unix to w gruncie rzeczy zbiór aplikacji i narzędzi. Niektóre narzędzia służą do diagnozowania innych; oczywiście, zawsze istnieje kilka sposobów na osiągnięcie tego samego celu. Kiedy próbujesz rozwiązać problem związany z Sambą, możesz skorzystać z następujących zasobów:

- dzienniki Samby,
- drzewo błędów,
- narzędzia uniksowe,
- testowe narzędzia Samby,
- dokumentacja i dokumenty FAQ,
- archiwa list wysyłkowych,
- grupy dyskusyjne.

W kolejnych podpunktach omówimy szczegółowo każdy z nich.

Dzienniki Samby

Pierwszą czynnością powinno być zawsze sprawdzenie dzienników Samby. Zawierają one odpowiedzi na znakomitą większość pytań, które mogą sobie zadawać

początkujący i średnio zaawansowani administratorzy Samby. Samba jest bardzo elastyczna, jeśli chodzi o rejestrowanie podejmowanych przez siebie czynności. Możesz skonfigurować serwer tak, aby rejestrował dokładnie tyle informacji, ile sobie zażyczysz. Podstawianie zmiennych pozwala ci na wyodrębnienie danych związanych z poszczególnymi komputerami, udziałami lub ich kombinacją.

Domyślnie dzienniki są przechowywane w plikach *katalog_samby/var/smbd.log* oraz *katalog_samby/var/nmbd.log*, gdzie *katalog_samby* to katalog, w którym zainstalowano Sambę (zwykle */usr/local/samba*). Jak wspomnieliśmy w rozdziale 4, *Udziały dyskowe*, możesz zmienić ich położenie i nazwę za pomocą opcji `log file` pliku *smb.conf*. Opcja ta rozpoznaje wszystkie zmienne opisane w rozdziale 2, *Instalowanie Samby w Uniksie*, więc możesz prowadzić oddzielny dziennik dla każdego klienta, dopisując poniższą opcję do sekcji `[global]` pliku *smb.conf*:

```
log file = %m.log
```

Możesz także określić katalog dzienników za pomocą opcji `-l` w linii polecenia, na przykład:

```
smbd -l /usr/local/var/samba
```

Innym przydatnym rozwiązaniem jest prowadzenie dzienników dla każdego udostępnianego zasobu (udziału), zwłaszcza wtedy, gdy podejrzewasz, że sprawcą kłopotów jest konkretny udział. Aby to skonfigurować, użyj zmiennej `%S` w sekcji `[global]` pliku konfiguracyjnego:

```
log file = %S.log
```

Poziomy rejestrowania

Poziom rejestrowania można ustawić w pliku *smb.conf* za pomocą opcji `log level` lub `debug level` (są równoważne). Poziom rejestrowania to liczba całkowita: wartość 0 oznacza brak rejestrowania, a przy wartości 3 rejestrowane są już znaczne ilości danych. Załóżmy, że klient Windows ma przejrzeć katalog w serwerze Samby. Aby w dziennikach pojawiła się niewielka liczba danych, mógłbyś użyć opcji `log level = 1`, która nakazuje Sambie rejestrowanie tylko pobieżnych informacji, w tym przypadku po prostu komunikatu o nawiązaniu połączenia:

```
05/25/98 22:02:11 serwer (192.168.236.86) connect to service publ as user
pcguest (uid=503,gid=100) (pid 3377)
```

Wyższe poziomy rejestrowania powodują zapisywanie bardziej szczegółowych informacji. Zwykle nie będziesz potrzebował poziomów wyższych niż 3; uzyskane dane i tak będą więcej niż wystarczające dla większości administratorów Samby. Poziomy powyżej 3 są przeznaczone dla programistów Samby i rejestrują ogromne ilości niezrozumiałych informacji.

Oto przykładowe wyniki tej samej operacji na poziomie rejestrowania 2 i 3. Nie przejmuj się, jeśli nie rozumiesz wszystkich zawiłości połączenia SMB; chcemy tylko pokazać, jakie informacje są zapisywane na różnych poziomach rejestrowania.

```
/* Poziom 2 */
Got SIGHUP
Processing section "[homes]"
```

```
Processing section "[publ]"
Processing section "[tymcz]"
Allowed connection from 192.168.236.86 (192.168.236.86) to IPC$
Allowed connection from 192.168.236.86 (192.168.236.86) to IPC/

/* Poziom 3 */
05/25/98 22:15:09 Transaction 63 of length 67
switch message SMBtconX (pid 3377)
Allowed connection from 192.168.236.86 (192.168.236.86) to IPC$
ACCEPTED: guest account and guest ok
found free connection number 105
Connect path is /tmp
chdir to /tmp
chdir to /
05/25/98 22:15:09 serwer (192.168.236.86) connect to service IPC$ as user
pcguest (uid=503,gid=100) (pid 3377)
05/25/98 22:15:09 tconX service=ipc$ user=pcguest cnum=105
05/25/98 22:15:09 Transaction 64 of length 99
switch message SMBtrans (pid 3377)
chdir to /tmp
trans <\PIPE\LANMAN> data=0 params=19 setup=0
Got API command 0 of form <WrLeh> <B13BWz>
(tdscnt=0,tpscnt=19,mdrcnt=4096,mprcnt=8)
Doing RNetShareEnum
RNetShareEnum gave 4 entries of 4 (1 4096 126 4096)
05/25/98 22:15:11 Transaction 65 of length 99
switch message SMBtrans (pid 3377)
chdir to /
chdir to /tmp
trans <\PIPE\LANMAN> data=0 params=19 setup=0
Got API command 0 of form <WrLeh> <B13BWz>
(tdscnt=0,tpscnt=19,mdrcnt=4096,mprcnt=8)
Doing RNetShareEnum
RNetShareEnum gave 4 entries of 4 (1 4096 126 4096)
05/25/98 22:15:11 Transaction 66 of length 95
switch message SMBtrans2 (pid 3377)
chdir to /
chdir to /dyskpc/publ
call_trans2findfirst: dirtype = 0, maxentries = 6, close_after_first=0,
close_if_end = 0 requires_resume_key = 0 level = 260, max_data_bytes = 2432
unix_clean_name [./DESKTOP.INI]
unix_clean_name [desktop.ini]
unix_clean_name [./]
creating new dirptr 1 for path ./, expect_close = 1
05/25/98 22:15:11 Transaction 67 of length 53
switch message SMBgetatr (pid 3377)
chdir to /

[...]
```

Obcięliśmy ten listing po pierwszym pakiecie, ponieważ ciągnąłby się przez wiele stron. Powinieneś zdawać sobie sprawę, że poziomy rejestrowania powyżej 3 szybko wypełnią dysk megabajtami nieznośnie szczegółowych informacji o wewnętrznych operacjach Samby. Poziom rejestrowania 3 jest niezwykle przydatny do dokładnego śledzenia pracy serwera i w większości przypadków pozwala na wykrycie miejsca wystąpienia błędu przez przejrzenie pliku dziennika.

Słowo ostrzeżenia: ustawienie wysokiego poziomu rejestrowania (3 lub powyżej) *znacznie* spowolni pracę serwera Samby. Pamiętaj, że każdy wygenerowany komunikat powoduje zapis na dysku (co jest operacją z natury powolną), a poziomy rejestrowania wyższe niż trzy produkują ogromne ilości danych. Zasadniczo, poziom 3 powinieneś włączać tylko wtedy, gdy próbujesz wyśledzić źródło problemu w serwerze Samby.

Włączanie i wyłączanie rejestrowania

Aby włączyć lub wyłączyć rejestrowanie, musisz ustawić odpowiednią opcję w sekcji `[global]` pliku *smb.conf*. Następnie możesz albo zrestartować Sambę, albo wymusić ponowne przetworzenie pliku konfiguracyjnego przez właśnie działającego demona. Możesz również wysłać procesowi *smbd* sygnał SIGUSR1, aby zwiększyć poziom rejestrowania o 1, albo SIGUSR2, aby zmniejszyć go o 1:

```
# Zwi□kszamy poziom rejestrowania o 1
kill -SIGUSR1 1234

# Zmniejszamy poziom rejestrowania o 1
kill -SIGUSR2 1234
```

Rejestrowanie z podziałem na poszczególne komputery i użytkowników

Wydajną metodą diagnozowania problemów bez przeszkadzania innym użytkownikom jest przypisanie różnym komputerom różnych poziomów rejestrowania w sekcji `[global]` pliku *smb.conf*. Możemy to zrobić, rozszerzając omówioną uprzednio metodę:

```
[global]
    log level = 0
    log file = /usr/local/samba/lib/log.%m
    include = /usr/local/samba/lib/smb.conf.%m
```

Te opcje nakazują Sambie użycie odrębnych plików dziennika i plików konfiguracyjnych dla wszystkich łączących się z nią klientów. Teraz wystarczy utworzyć plik *smb.conf* dla konkretnego klienta, umieścić w nim opcję `log level = 3` (pozostałe klienty przyjmą domyślną wartość poziomu rejestrowania równą 0) i użyć pliku dziennika do ustalenia przyczyny problemu.

Podobnie, jeśli problem dotyczy tylko niektórych użytkowników i wędruje za nimi od komputera do komputera, możesz utworzyć odrębne pliki dziennika dla poszczególnych użytkowników, dopisując do pliku *smb.conf* poniższe opcje:

```
[global]
    log level = 0
    log file = /usr/local/samba/lib/log.%u
    include = /usr/local/samba/lib/smb.conf.%u
```

Następnie możesz utworzyć oddzielny plik *smb.conf* dla danego użytkownika (na przykład */usr/local/samba/lib/smb.conf.tomek*), umieścić w nim opcję `log level = 3` i rejestrować szczegółowo tylko połączenia tego użytkownika.

Narzędzia testowe Samby

Zbiór rygorystycznych testów, którym można poddać główne elementy Samby, jest opisany w różnych plikach zawartych w katalogu */docs/textdocs* dystrybucji Samby, przede wszystkim w pliku *DIAGNOSIS.TXT*. Drzewo błędów zamieszczone w tym rozdziale jest bardziej szczegółową wersją podstawowych testów proponowanych przez zespół twórców Samby, ale obejmuje tylko diagnozowanie instalacji i rekonfiguracji, podobnie jak *DIAGNOSIS.TXT*. Inne pliki w podkatalogach */docs* omawiają specyficzne problemy (na przykład z klientami Windows NT) i podają sposoby usuwania usterek, których nie przedstawiliśmy w tej książce. Jeśli drzewo błędów okaże się niewystarczające, zajrzyj do *DIAGNOSIS.TXT* i pokrewnych plików.

Narzędzia uniksowe

Czasem warto użyć narzędzia spoza pakietu Samby, aby sprawdzić, co dzieje się w serwerze. Unix zawsze był systemem operacyjnym przypominającym kuchenny zlew. Do tropienia błędów Samby szczególnie przydatne są dwa narzędzia diagnostyczne: *trace* i *tcpdump*.

Korzystanie z programu trace

Polecenie *trace* kryje się pod kilkoma różnymi nazwami, w zależności od używanego systemu operacyjnego. W Linuksie będzie to *strace*, w Solarisie – *truss*, a w SGI – *padc* i *par*. Wszystkie mają zasadniczo tę samą funkcję: wyświetlają wywołania wszystkich funkcji systemu operacyjnego w momencie ich wykonywania. Dzięki temu można prześledzić wykonanie programu, na przykład serwera Samby, i wychwycić wywołanie funkcji, która sprawia problemy.

Jednym z problemów, które dają się łatwo wykryć za pomocą *trace*, jest niepoprawna wersja biblioteki dołączanej dynamicznie. Może się to zdarzyć, jeśli pobierzesz z Internetu skompilowane pliki binarne Samby. Wywołanie odpowiedzialne za błąd zwykle znajdziesz na końcu wyników polecenia *trace*, tuż przed przerwaniem działania programu.

Poniżej zamieszczamy przykładowe wyniki polecenia `strace` w Linuksie. Jest to niewielka część pliku utworzonego podczas otwierania katalogu w serwerze Samby. Każda linia zaczyna się od nazwy wywołania systemowego i zawiera jego parametry oraz wartość zwracaną. Jeśli nastąpił błąd, wyświetlany jest kod błędu (na przykład `ENOENT`) i jego wyjaśnienie. Możesz zapoznać się z typami parametrów i możliwymi błędami na stronie podręcznika man dla polecenia `trace` lub jego odpowiednika w twoim systemie operacyjnym.

```
chdir("/dyskpc/publ")                   = 0
stat("mini/desktop.ini", 0xbffff7ec)    = -1 ENOENT (No such file or directory)
stat("mini", {st_mode=S_IFDIR|0755, st_size=1024, ...}) = 0
stat("mini/desktop.ini", 0xbffff7ec)    = -1 ENOENT (No such file or directory)
open("mini", O_RDONLY)                  = 5
fcntl(5, F_SETFD, FD_CLOEXEC)           = 0
fstat(5, {st_mode=S_IFDIR|0755, st_size=1024, ...}) = 0
lseek(5, 0, SEEK_CUR)                   = 0
SYS_141(0x5, 0xbfffdbbc, 0xedc, 0xbfffdbbc, 0x80ba708) = 196
```

```
lseek(5, 0, SEEK_CUR)                   = 1024
SYS_141(0x5, 0xbfffdbbc, 0xedc, 0xbfffdbbc, 0x80ba708) = 0
close (5)                               = 0
stat("mini/desktop.ini", 0xbffff86c)    = -1 ENOENT (No such file or directory)
write(3, "\0\0\0#\377SMB\10\1\0\2\0\200\1\0"..., 39) = 39
SYS_142(0xff, 0xbffffc3c, 0, 0, 0xbffffc08) = 1
read(3, "\0\0\0?", 4)                   = 4
read(3, "\377SMBu\0\0\0\0\0\0\0\0\0\0\0"..., 63) = 63
time(NULL)                              = 896143871
```

Ten przykład pokazuje, jak kilka wywołań `stat` nie znajduje plików w oczekiwanym miejscu. Nie trzeba być ekspertem, aby domyślić się, że w katalogu nie ma pliku *desktop.ini*. Na co dzień wiele trudnych problemów można zidentyfikować, szukając oczywistych, powtarzalnych błędów w wynikach polecenia *trace*. Często wystarczy spojrzeć na ostatni komunikat przed załamaniem programu.

Korzystanie z programu tcpdump

Program *tcpdump*, napisany przez Van Jacobsona, Craiga Leresa i Stevena McCanne'a oraz rozszerzony przez Andrew Tridgella, pozwala na monitorowanie ruchu sieciowego w czasie rzeczywistym. Program udostępnia wiele różnorodnych formatów wyjściowych i umożliwia filtrowanie danych w celu wyizolowania konkretnego typu ruchu. Program *tcpdump* umożliwia zbadanie wszystkich konwersacji między klientem i serwerem, w tym rozgłoszeniowych komunikatów SMB i NMB. Choć funkcje diagnostyczne programu ograniczają się w zasadzie do warstwy sieci w modelu OSI, jego wyniki pozwalają ogólnie zorientować się w celach, które klient i serwer próbują osiągnąć.

Poniżej zamieszczamy przykładowe wyniki programu *tcpdump*. W tym przykładzie klient poprosił o listing katalogu, a serwer spełnił żądanie, podając nazwy katalogów: `homes`, `publ`, `IPC$` i `tymcz` (po prawej stronie listingu dołączyliśmy kilka komentarzy):

```
$tcpdump -v -s 255 -i eth0 port not telnet
SMB PACKET: SMBtrans (REQUEST)              Pakiet □□dania
SMB Command   =  0x25                       □□danie to ls lub dir

[000] 01 00 00 10                           ....

>>> NBT Packet                              Zewn□trzna otoczka pakietu SMB
NBT Session Packet
Flags=0x0
Length=226

[pomini□te linie]
SMB PACKET: SMBtrans (REPLY)                Pocz□tek odpowiedzi na □□danie
SMB Command   =  0x25                       Polecenie to ls lub dir
Error class   =  0x0
Error code    =  0                          Nie by□o b□□dów
Flags1        =  0x80
Flags2        =  0x1
Tree ID       =  105
Proc ID       =  6075
UID           =  100
MID           =  30337
Word Count    =  10
```

```
TotParamCnt=8
TotDataCnt=163
Res1=0
ParamCnt=8
ParamOff=55
Res2=0
DataCnt=163
DataOff=63
Res3=0
Lsetup=0
Param Data: (8 bytes)
[000] 00 00 00 00 05 00 05 00                           ........
Data Data: (135 bytes)
Rzeczywista zawarto□□ katalogu:
[000] 70 75 62 6C 00 00 00 00  00 00 00 00 00 00 00 00  publ.... ........
[010] 64 00 00 00 74 79 6D 63  7A 00 00 00 00 00 00 00  d...tymc z.......
[020] 00 00 00 00 65 00 00 00  68 6F 6D 65 73 00 00 00  ....e... homes...
[030] 00 00 00 00 00 00 00 00  66 00 00 00 49 50 43 24  ........ f...IPC$
[040] 00 00 00 00 00 00 00 00  00 00 03 00 67 00 00 00  ........ ....g...
[050] 77 65 72 00 00 00 00 00  00 00 00 00 00 00 00 00  wer..... ........
[060] 81 00 00 00 00 00 00 49  50 43 20 53 65 72 76 69  .......I PC Servi
[070] 63 65 20 28 53 61 6D 62  61 20 32 2E 30 2E 35 29  ce (Samb a 2.0.5)
[080] 00 48 6F 6D 65 20 64                              .Home d
```

Jest to dalsza część tej samej sesji diagnostycznej co w przykładzie z poleceniem *trace*: listing katalogu. Opcje, których użyliśmy to: `-v` (szczegółowe wyniki), `-i eth0` (określa interfejs, który ma być monitorowany przez program *tcpdump*, w tym przypadku port Ethernetu) oraz `-s 255` (aby program zapisał pierwsze 255 bajtów każdego pakietu, zamiast domyślnych 68). Opcja `port not telnet` zapobiegała wyświetlaniu ruchu związanego z telnetem, ponieważ byliśmy zalogowani zdalnie. Program *tcpdump* ma znaczną liczbę opcji, które umożliwiają odfiltrowanie tylko tego ruchu, któremu chcesz się przyjrzeć. Jeśli używałeś programów *snoop* lub *etherdump*, to wyniki zapewne wydają ci się znajome.

Możesz pobrać zmodyfikowaną wersję programu *tcpdump* z serwera FTP Samby pod adresem *ftp://samba.anu.edu.au/pub/samba/tcpdump-smb*. Inne wersje nie obsługują protokołu SMB; jeśli uzyskane przez ciebie wyniki różnią się od tych z przykładu, będziesz musiał zaopatrzyć się w wersję z obsługą SMB.

Drzewo błędów

Drzewo błędów służy do diagnozowania i rozwiązywania problemów podczas instalowania i rekonfigurowania Samby. Jest to rozszerzona wersja dokumentu diagnostycznego wchodzącego w skład dystrybucji Samby.

Zanim przystąpisz do tropienia błędów w którymkolwiek elemencie pakietu Samby, powinieneś dysponować następującymi informacjami:

- adresem IP klienta (my używamy adresu 192.168.236.10),
- adresem IP serwera (my używamy adresu 192.168.236.86),
- maską używaną w sieci (zwykle 255.255.255.0),
- czy wszystkie komputery znajdują się w tej samej podsieci (tak jest u nas).

Dla jasności zmieniliśmy nazwę serwera w poniższych przykładach na *serwer.przyklad.com*, a klienta – na *klient.przyklad.com*.

Jak korzystać z drzewa błędów

Zacznij testy od tego miejsca i nie przeskakuj do przodu; dana procedura testowa nie powinna trwać zbyt długo (około 5 minut), a ty unikniesz cofania się po własnych śladach. Jeśli test zakończy się pomyślnie, podamy ci tytuł podrozdziału i numer strony, do której powinieneś przejść.

Diagnozowanie problemów z niskopoziomowym ruchem IP

Pierwsza seria testów bada niskopoziomowe usługi, które są niezbędne do pracy Samby. Testy w tym podrozdziale pozwolą ci upewnić się, że:

- oprogramowanie IP działa poprawnie,
- sprzętowe elementy Ethernetu działają poprawnie,
- funkcjonują podstawowe usługi nazewnicze.

W następnych sekcjach zajmiemy się oprogramowaniem TCP/IP, demonami Samby: *smbd* i *nmbd*, kontrolą dostępu na poziomie hostów, uwierzytelnianiem i kontrolą dostępu na poziomie użytkownika, usługami plikowymi i przeglądaniem. Testy opisaliśmy dość szczegółowo, aby były zarozumiałe nie tylko dla doświadczonych administratorów systemu i sieci, ale i dla zwykłych użytkowników (którym jednak nieobce są kwestie techniczne).

Testowanie oprogramowania sieciowego za pomocą polecenia ping

Pierwsze polecenie, które należy wpisać zarówno w serwerze, jak i w kliencie, to `ping 127.0.0.1`. Jest to *adres pętli zwrotnej*, a przetestowanie go pozwoli stwierdzić, czy w komputerze jest zainstalowana obsługa sieci. W Uniksie możesz wpisać polecenie `ping 127.0.0.1` z opcją statystyczną i przerwać je po wyświetleniu kilku linii. W stacjach roboczych Suna można użyć polecenia `/usr/etc/ping -s 127.0.0.1`; w Linuksie wystarczy po prostu `ping 127.0.0.1`. W kliencie Windows powinieneś uruchomić polecenie `ping 127.0.0.1` w oknie MS-DOS-a i zaczekać, aż samo przerwie działanie po wyświetleniu czterech linii.

Oto przykład z serwera linuksowego:

```
serwer% ping 127.0.0.1
PING localhost: 56 data bytes
64 bytes from localhost: icmp_seq=0. time=1. ms
64 bytes from localhost: icmp_seq=1. time=0. ms
64 bytes from localhost: icmp_seq=2. time=1. ms ^C
---- 127.0.0.1 PING Statistics ----
3 packets transmitted, 3 packets received, 0% packet loss round-trip (ms)
min/avg/max = 0/0/1
```

Jeśli otrzymasz komunikat „ping: no answer from...” lub „100% packet loss”, oznacza to, że obsługa sieci IP w ogóle nie jest zainstalowana w komputerze. Adres `127.0.0.1` to wewnętrzny adres pętli zwrotnej, który funkcjonuje bez względu na

to, czy komputer jest podłączony do sieci, czy nie. Jeśli ten test się nie powiedzie, masz poważny problem z lokalnym komputerem. Protokół TCP/IP albo nie jest w ogóle zainstalowany, albo jest źle skonfigurowany. Jeśli problemy sprawia serwer uniksowy, zajrzyj do dokumentacji systemu operacyjnego, jeśli zaś klient Windows – postępuj według instrukcji instalowania obsługi sieci zamieszczonej w rozdziale 3, *Konfigurowanie klientów Windows*.

Jeśli zarządzasz siecią, możesz sięgnąć po książkę Craiga Hunta *TCP/IP – administracja sieci* (rozdział 11) albo nową książkę Craiga Hunta i Roberta Bruce'a Thompsona *Windows NT TCP/IP Administration* – obie zostały opublikowane przez wydawnictwo O'Reilly.

Testowanie lokalnych usług nazewniczych za pomocą polecenia ping

Następnie przetestuj poleceniem *ping* nazwę `localhost`. Jest to konwencjonalna nazwa interfejsu pętli zwrotnej (127.0.0.1) i powinna zostać odwzorowana właśnie na ten adres. Po wpisaniu `ping localhost` powinieneś zobaczyć następujące wyniki:

```
serwer% ping localhost
PING localhost: 56 data bytes
64 bytes from localhost: icmp_seq=0. time=1. ms
64 bytes from localhost: icmp_seq=1. time=0. ms
64 bytes from localhost: icmp_seq=2. time=1. ms ^C
```

Jeśli ten test się powiedzie, przeprowadź go także w kliencie. W przeciwnym wypadku:

- Jeśli otrzymasz komunikat „unknown host: localhost", oznacza to, że występuje problem z odwzorowaniem nazwy „localhost" na adres IP (być może po prostu brakuje odpowiedniego wpisu w pliku *hosts*). Przejdź do podrozdziału „Rozwiązywanie problemów z usługami nazewniczymi".
- Jeśli otrzymasz komunikat „ping: no answer" lub „100% packet loss", a testowanie adresu 127.0.0.1 przebiegło pomyślnie, oznacza to, że usługi nazewnicze odwzorowują nazwę „localhost" na adres, ale adres ten jest niepoprawny. Sprawdź plik lub bazę danych (w systemach uniksowych jest to zwykle plik */etc/hosts*) używaną przez usługi nazewnicze do odwzorowywania nazw i popraw błędny wpis.

Testowanie sprzętu sieciowego za pomocą polecenia ping

Następnie przetestuj poleceniem *ping* własny adres IP serwera. Powinno to dać dokładnie ten sam wynik, co testowanie adresu 127.0.0.1:

```
serwer% ping 192.168.236.86
PING 192.168.236.86: 56 data bytes
64 bytes from 192.168.236.86: icmp_seq=0. time=1. ms
64 bytes from 192.168.236.86: icmp_seq=1. time=0. ms
64 bytes from 192.168.236.86: icmp_seq=2. time=1. ms ^C
--- 192.168.236.86 PING Statistics ---
3 packets transmitted, 3 packets received, 0% packet loss round-trip (ms)
min/avg/max = 0/0/1
```

Jeśli ten test powiedzie się w serwerze, przeprowadź go także w kliencie. W przeciwnym wypadku:

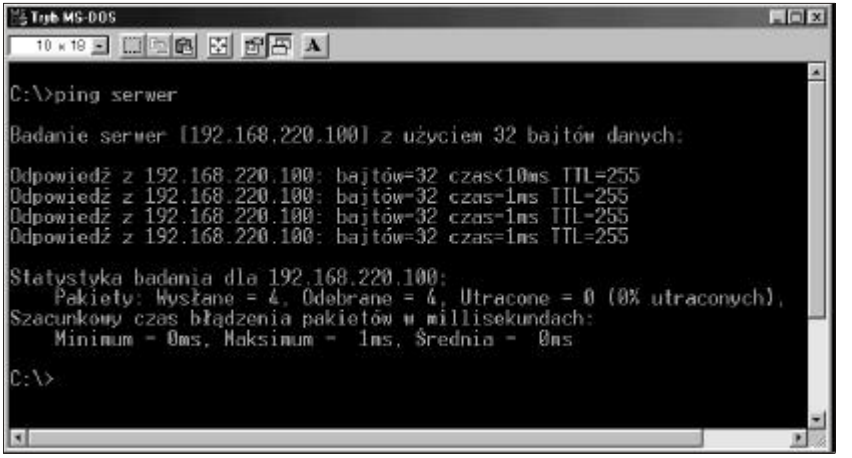
- Jeśli polecenie `ping adres_sieciowy` nie powiedzie się w serwerze lub kliencie, a polecenie `ping 127.0.0.1` zadziała, problem jest związany z konfiguracją TCP/IP karty sieciowej w tym komputerze. Zajrzyj do dokumentacji karty lub systemu operacyjnego i sprawdź, jak poprawnie skonfigurować kartę. Musisz jednak wiedzieć, że w niektórych systemach operacyjnych polecenie *ping* działa nawet wtedy, gdy komputer jest odłączony od sieci, więc ten test może nie wykryć wszystkich problemów sprzętowych.

Testowanie połączeń za pomocą polecenia ping

Teraz przetestuj poleceniem *ping* nazwę serwera (a nie jego adres IP) – raz z serwera, a raz z klienta. Jest to ogólny test sprawności sprzętu sieciowego:

```
serwer% ping serwer
PING serwer.przyklad.com: 56 data bytes
64 bytes from serwer.przyklad.com: icmp_seq=0. time=1. ms
64 bytes from serwer.przyklad.com: icmp_seq=1. time=0. ms
64 bytes from serwer.przyklad.com: icmp_seq=2. time=1. ms ^C
--- serwer.przyklad.com PING Statistics ---
3 packets transmitted, 3 packets received, 0% packet loss round-trip (ms)
min/avg/max = 0/0/1
```

W Microsoft Windows testowanie serwera wyglądałoby tak, jak na rysunku 9.1.



Rysunek 9.1. Testowanie serwera Samby z klienta Windows za pomocą polecenia ping

Jeśli test się powiedzie, będziemy wiedzieli, że:

1. Nazwa hosta (na przykład „serwer") jest rozpoznawana przez lokalny serwer nazw.
2. Nazwa hosta została rozwinięta do pełnej postaci (na przykład *serwer.przyklad.com*).
3. Zwracany jest adres hosta (192.168.236.86).
4. Klient przesłał serwerowi Samby cztery pakiety UDP/IP z 56 bajtami danych.
5. Serwer Samby odpowiedział na wszystkie cztery pakiety.

Jeśli test się nie powiedzie, w sieci występuje jeden z poniższych problemów:

- Po pierwsze, jeśli otrzymasz komunikat „ping: no answer" lub „100% packet loss", oznacza to, że nie jesteś podłączony do sieci, drugi komputer nie jest podłączony do sieci albo jeden z adresów jest niepoprawny. Sprawdź adresy zgłaszane przez polecenie `ping` w obu komputerach i upewnij się, że odpowiadają tym, które wstępnie skonfigurowałeś.

  Jeśli nie, w obu komputerach jest przynajmniej jeden niedopasowany adres. Spróbuj wpisać `arp -a` i zobacz, czy zostanie wyświetlony wpis dla drugiego komputera. Polecenie `arp` bierze swoją nazwę od protokołu Address Resolution Protocol. Opcja `-a` powoduje wyświetlenie wszystkich adresów znanych lokalnemu komputerowi. Oto kilka możliwości:

  - Jeśli otrzymasz komunikat typu „192.168.236.86 at (incomplete)", oznacza to, że ethernetowy odpowiednik adresu 192.168.236.86 jest nieznany. Wskazuje to na zupełny brak łączności – prawdopodobnie problem występuje na samym dole stosu protokołów TCP/IP, w warstwie interfejsu Ethernetu. Zagadnienie to jest omówione w rozdziałach 5 i 6 książki *TCP/IP – administracja sieci* (wydanej przez Wydawnictwo RM).
  - Jeśli otrzymasz komunikat typu „serwer (192.168.236.86) at 8:0:20:12:7c:94", oznacza to, że z serwerem skontaktowano się już wcześniej albo że inny komputer odpowiada w jego imieniu. Jednakże w takim razie polecenie *ping* powinno działać: albo sieć działa nieregularnie, albo masz problem z ARP.
  - Jeśli adres IP wyświetlony przez polecenie `arp` jest inny niż oczekiwałeś, sprawdź, gdzie kryje się błąd i popraw adres ręcznie.

- Jeśli oba komputery mogą sprawdzić poleceniem *ping* same siebie, ale nie drugi komputer, błąd kryje się w łączącej je sieci.
- Jeśli otrzymasz komunikat „ping: network unreachable" lub „ICMP Host Unreachable", oznacza to, że nie otrzymujesz odpowiedzi i najprawdopodobniej w grę wchodzi więcej niż jedna sieć.

  W zasadzie nie powinieneś próbować diagnozowania klientów i serwerów SMB znajdujących się w różnych sieciach. Spróbuj przetestować serwer i klienty w tej samej sieci. W trzech opisanych poniżej testach zakładamy jednak, że diagnozujesz łączność między dwoma sieciami:

  a. Po pierwsze, przeprowadź opisane wcześniej testy ustalające przyczynę braku odpowiedzi. Jeśli nie uda ci się zidentyfikować problemu, pozostają następujące możliwości: adres jest błędny, maska sieciowa jest błędna, sieć jest niedostępna albo po prostu zostałeś zablokowany przez zaporę sieciową.
  b. Sprawdź adresy i maski sieciowe komputera źródłowego i docelowego i zastanów się, czy błąd nie jest oczywisty. Jeśli dwa komputery rzeczywiście są w tej samej sieci, powinny mieć takie same maski sieciowe, a polecenie *ping* powinno zgłaszać poprawne adresy. Jeśli adresy są błędne, będziesz musiał je skorygować. Jeśli są poprawne, być może programy mylą się przez błędne maski sieciowe. Patrz podrozdział „Maski sieciowe" dalej w tym rozdziale.

c. Jeśli polecenia nadal zgłaszają, że sieć jest nieosiągalna, a spełnione są dwa poprzednie warunki, jedna sieć może być rzeczywiście nieosiągalna z drugiej. Rozwiązanie tego problemu leży w gestii administratora sieci.

- Jeśli otrzymasz komunikat „ICMP Administratively Prohibited", oznacza to, że natknąłeś się na zaporę sieciową lub błędnie skonfigurowany ruter. Będziesz musiał porozmawiać z osobą odpowiedzialną za bezpieczeństwo sieci.
- Jeśli otrzymasz komunikat „ICMP Host redirect", ale *ping* informuje, że pakiety trafiają do celu, z reguły jest to nieszkodliwe: po prostu ruch w sieci jest przekierowywany w inne miejsce.
- Jeśli otrzymasz komunikat „ICMP Host redirect", a polecenie *ping* nie odbiera odpowiedzi, oznacza to, że ruch jest przekierowywany, ale żaden komputer nie odpowiada. Potraktuj to tak samo jak przypadek „Network unreachable" i sprawdź adresy oraz maski sieciowe.
- Jeśli otrzymasz komunikat „ICMP Host Unreachable from gateway *nazwa_bramki*", pakiety są przekazywane do innej sieci, ale drugi komputer nie odpowiada, a ruter zgłasza problem w jego imieniu. Potraktuj to tak samo jak przypadek „Network unreachable" i sprawdź adresy oraz maski sieciowe.
- Jeśli otrzymasz komunikat „ping: unkown host *nazwa_hosta*", oznacza to, że nazwa komputera jest nieznana. Najczęściej jest to symptom problemu z usługami nazewniczymi, który nie wystąpił w przypadku nazwy `localhost`. Zajrzyj do podrozdziału „Rozwiązywanie problemów z usługami nazewniczymi" dalej w tym rozdziale.
- Jeśli osiągniesz częściowy sukces – niektóre „pingi" powiodą się, a inne nie – oznacza to sporadyczne zakłócenia w łączności między komputerami lub przeciążenie sieci. Przedłuż test i sprawdź, czy odnotujesz brak odpowiedzi dla więcej niż 3 procent pakietów. Jeśli tak jest w istocie, skontaktuj się z administratorem sieci: problem być może właśnie się zaczyna. Jeśli jednak zawiodą tylko nieliczne próby albo wiesz, że akurat działa program mocno obciążający sieć, nie ma powodu do niepokoju. Protokoły ICMP (i UDP) używane przez polecenie *ping* z założenia mogą gubić niektóre pakiety.
- Jeśli otrzymasz komunikat typu „smtsvr.antares.net is alive" podczas testowania hosta *klient.przyklad.com*, oznacza to, że używasz adresu IP innego komputera albo komputer ma kilka nazw i adresów. Jeśli adres jest błędny, odpowiedzialność za to ponoszą usługi nazewnicze; będziesz musiał zmienić adres w bazie danych usług nazewniczych tak, aby odnosił się do właściwego komputera. Omawiamy to w dalszym podrozdziale „Rozwiązywanie problemów z usługami nazewniczymi".

  Komputery pełniące funkcje serwera są często połączone z wieloma sieciami i mają inną nazwę w każdej z nich. Jeśli otrzymasz odpowiedź od nieoczekiwanej nazwy takiego serwera, sprawdź, czy adres rzeczywiście znajduje się w twojej sieci (patrz podrozdział „Maski sieciowe"). Jeśli tak jest, powinieneś używać tego adresu, a nie znajdującego się w innej sieci, ze względu na wydajność i niezawodność.

Serwery mogą mieć także kilka nazw dla jednego adresu ethernetowego, zwłaszcza wtedy, gdy są serwerami WWW. Jest to nieszkodliwe, choć bywa mylące. Prawdopodobnie lepiej będzie użyć oficjalnej (i stałej) nazwy, niż aliasu, który może się zmienić.

- Jeśli wszystko zadziała, ale polecenie zgłosi adres IP 127.0.0.1, za błąd odpowiedzialne są usługi nazewnicze. Najczęściej zdarza się to wtedy, gdy program instalacyjny systemu operacyjnego umieści w pliku */etc/hosts* linię: `127.0.0.1 localhost` *nazwahosta.nazwadomeny*. Linia ta powinna mieć postać `127.0.0.1 localhost` lub `127.0.0.1 localhost loghost`. Popraw ją, gdyż w przeciwnym razie niemożliwe będzie ustalenie, który komputer przechowuje główną listę przeglądania, a który jest główną przeglądarką domeny. Może także spowodować błędy (niejednoznaczne) w późniejszych testach.

Jeśli wszystko funkcjonuje poprawnie w serwerze, powtórz testy w kliencie.

Rozwiązywanie problemów z TCP

Po przetestowaniu poleceniem *ping* protokołów IP, UDP i usług nazewniczych czas na sprawdzenie protokołu TCP. Program *ping* oraz usługi przeglądania korzystają z ICMP i UDP; usługi plikowe i usługi drukowania (udziały) korzystają z TCP. Do działania TCP niezbędna jest poprawna praca IP w niższej warstwie sieci, a wszystkie cztery wymienione usługi opierają się na usługach nazewniczych. Test TCP najwygodniej przeprowadzić za pomocą programu FTP (*File Transfer Protocol*).

Testowanie TCP za pomocą FTP

Spróbuj połączyć za pomocą FTP najpierw serwer z samym sobą, a potem klienta z serwerem:

```
serwer% ftp serwer
Connected to serwer.przyklad.com.
220 serwer.przyklad.com FTP server (Version 6.2/OpenBSD/Linux-0.10) ready.
 Name (serwer:davecb):
331 Password required for davecb.
Password:
230 User davecb logged in.
 ftp> quit
221 Goodbye.
```

Jeśli to zadziała, przejdź do podrozdziału „Rozwiązywanie problemów z demonami serwera". W przeciwnym wypadku:

- Jeśli otrzymałeś komunikat „serwer: unknown host", oznacza to, że zawiodły usługi nazewnicze. Wróć do podrozdziału „Testowanie lokalnych usług nazewniczych za pomocą polecenia ping" i przeprowadź testy jeszcze raz, aby ustalić, czemu nie działa wyszukiwanie nazw.
- Jeśli otrzymałeś komunikat „ftp: connect: Connection refused", w komputerze nie działa demon FTP. Jest to raczej niezwykłe w przypadku uniksowego serwera. Możesz opcjonalnie przeprowadzić ten test, łącząc się z komputerem za pomocą telnetu zamiast FTP; komunikaty są podobne, a telnet również korzysta z TCP.

- Jeśli po długiej pauzie pojawił się komunikat „ftp: connect: Connection timed out", oznacza to, że komputer jest nieosiągalny. Wróć do podrozdziału „Testowanie połączeń za pomocą polecenia ping".
- Jeśli otrzymałeś komunikat „530 Logon Incorrect", oznacza to, że udało ci się nawiązać połączenie, ale pojawił się inny problem. Prawdopodobnie podałeś nieprawidłową nazwę użytkownika lub hasło. Spróbuj jeszcze raz, wpisując swoją nazwę użytkownika z uniksowego serwera i podając poprawne hasło.

Rozwiązywanie problemów z demonami serwera

Kiedy upewnisz się, że protokół TCP działa poprawnie, powinieneś sprawdzić, czy w serwerze działają demony. Wymaga to przeprowadzenia trzech oddzielnych testów, ponieważ żaden z nich sam z siebie nie pozwoli rozstrzygnąć, czy demony pracują prawidłowo.

Aby upewnić się, że demony działają, musisz sprawdzić, czy demony:

1. Uruchomiły się.
2. Są zarejestrowane lub powiązane z portem TCP/IP przez system operacyjny.
3. Rzeczywiście odpowiadają na żądania.

Zanim zaczniesz

Najpierw sprawdź dzienniki. Jeśli uruchomiłeś demony, powinieneś znaleźć komunikat „smbd version *jakaś_liczba* started". Jeśli go nie znajdziesz, będziesz musiał ponownie uruchomić demony.

Jeśli demon zgłasza, że rzeczywiście rozpoczął pracę, poszukaj komunikatu „bind failed on port 139 socket_addr=0 (Address already in use)". Oznacza on, że inny demon został uruchomiony na porcie 139 (*smbd*). Demon *nmbd* zgłosi podobny błąd, jeśli nie zdoła dowiązać się do portu 137. Albo uruchomiłeś demony dwukrotnie, albo serwer *inetd* spróbował uruchomić je w twoim imieniu. Ten drugi przypadek zdiagnozujemy za chwilę.

Wyszukiwanie procesów demona za pomocą polecenia ps

Następnie musisz sprawdzić, czy demony uruchomiły się. Użyj w serwerze polecenia `ps` z opcją wydłużonych wyników (zwykle `ps ax` lub `ps -ef`) i sprawdź, czy zarówno *smbd*, jak i *nmbd* rzeczywiście działają. Zwykle wygląda to następująco:

```
serwer% ps ax
PID TTY STAT TIME  COMMAND
1   ?   S    0:03  init [2]
2   ?   SW   0:00  (kflushd)
(...wiele linii z informacjami o procesach...)
234 ?   S    0:14  nmbd -D3
237 ?   S    0:11  smbd -D3
(...więcej linii, być może kolejne linie smbd...)
```

Ten przykład pokazuje, że procesy *smbd* i *nmbd* zostały już uruchomione jako autonomiczne demony (opcja `-D`) na poziomie rejestrowania 3.

Wyszukiwanie demonów dowiązanych do portów

Demony muszą być zarejestrowane w systemie operacyjnym, aby mogły uzyskać dostęp do portów TCP/IP. Polecenie `netstat` pokaże, czy tak jest w istocie. Wydaj w serwerze polecenie `netstat -a` i poszukaj linii wzmiankujących `netbios`, `137` lub `139`:

```
serwer% netstat -a
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address       (state)
udp   0      0       *.netbios-            *.*
tcp   0      0       *.netbios-            *.*
LISTEN
tcp   8370   8760    serwer.netbios-       klient.1439
ESTABLISHED
```

lub:

```
serwer% netstat -a
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address       (state)
udp   0      0       *.137                 *.*
tcp   0      0       *.139                 *.*
LISTEN
tcp   8370   8760    serwer.139            klient.1439
ESTABLISHED
```

Pomiędzy wieloma podobnymi liniami powinieneś znaleźć przynajmniej jedną linię UDP z portem `*.netbios-` lub `*.137`. Oznacza to, że serwer *nmbd* jest zarejestrowany i (miejmy nadzieję) czeka, aby odpowiedzieć na żądanie. Powinna także pojawić się co najmniej jedna linia TCP wzmiankująca `*.netbios-` lub `*.139`, prawdopodobnie w stanie nasłuchiwania (LISTENING). Oznacza to, że demon *smbd* działa i czeka na połączenia.

Mogą pojawić się także inne linie TCP informujące o połączeniach między *smbd* i klientami, po jednej na każdego klienta. Zwykle znajdują się w stanie ustalonym (ESTABLISHED). Jeśli zobaczysz linie *smbd* w stanie ESTABLISHED, możesz być pewien, że *smbd* działa poprawnie. Jeśli znajdziesz choćby tylko jedną linię *smbd* w stanie LISTENING, nie można tego stwierdzić na pewno. Jeśli nie ma obu linii, oznacza to, że demon nie zdołał się uruchomić, więc trzeba przejrzeć dzienniki i cofnąć się do rozdziału 2.

Jeśli znajdziesz linie dla poszczególnych klientów, mogą one pochodzić od demona Samby albo od nadrzędnego demona IP, *inetd*. Jest całkiem możliwe, że plik startowy demona *inetd* zawiera wpisy uruchamiające demony Samby, a ty nie zdajesz sobie z tego sprawy; wpisy te mogły zostać tam umieszczone na przykład wtedy, gdy zainstalowałeś Sambę jako część dystrybucji Linuksa. Demony uruchamiane przez *inetd* uniemożliwiają pracę naszych demonów. W takim przypadku w dziennikach znajdziesz komunikaty typu „bind failed on port 139 socket_addr=0 (Address already in use)".

Sprawdź swój plik *inetd.conf*; jeśli nie zamierzasz uruchamiać z niego demonów, *nie może* on zawierać wpisów dla serwerów `netbios-ns` (port UDP 137) i `netbios-ssn` (port TCP 139). Demon *inetd* udostępnia liczne usługi i jest sterowany

przez wpisy w pliku *inetd.conf*. Jeśli demon SMB w twoim systemie jest uruchamiany przez *inetd*, w pliku *inetd.conf* znajdziesz wpisy podobne do poniższych:

```
netbios-ssn stream tcp nowait root /usr/local/samba/bin/smbd smbd
netbios-ns dgram udp wait root /usr/local/samba/bin/nmbd nmbd
```

Sprawdzanie demona smbd za pomocą telnetu

Jak na ironię, najłatwiejszym sposobem przetestowania serwera *smbd* jest przesłanie mu bezsensownej wiadomości i sprawdzenie, czy ją odrzuci. Spróbuj wykonać następujące polecenie:

```
echo halo | telnet localhost 139
```

Spowoduje to wysłanie błędnego, ale nieszkodliwego komunikatu do demona *smbd*. Słowo `halo` jest tu istotne. Nie próbuj łączyć się z portem i wpisywać dowolnych znaków; prawdopodobnie zawiesisz tylko proces demona. Komunikat `halo` nie powinien wywołać niepożądanych skutków.

```
serwer% echo "halo" | telnet localhost 139
Trying
Trying 192.168.236.86 ...
Connected to localhost. Escape character is '^]'.
Connection closed by foreign host.
```

Jeśli otrzymasz komunikat „Connected", a zaraz po nim „Connection closed", oznacza to, że test przebiegł pomyślnie. Demon *smbd* nasłuchuje na porcie i odrzuca błędne połączenia. Jeśli jednak otrzymasz komunikat „telnet: connect: Connection refused", najprawdopodobniej demon jest nieobecny. Sprawdź dzienniki i wróć do rozdziału 2.

Niestety, nie ma łatwego sposobu na przetestowanie demona *nmbd*. Jeśli testy `telnet` i `netstat` potwierdzą, że demon *smbd* działa poprawnie, istnieje duże prawdopodobieństwo, że polecenie `netstat` nie myli się także co do demona *nmbd*.

Testowanie demonów za pomocą polecenia testparm

Kiedy upewnisz się, że demony działają, powinieneś koniecznie wykonać polecenie *`testparm`*, licząc na uzyskanie takich oto wyników:

```
serwer% testparm
Load smb config file from /opt/samba/lib/smb.conf
Processing section "[homes]"
Processing section "[printers]"
Processing section "[tymcz]"
Loaded services file OK. ...
```

Program `testparm` zwykle informuje o przetwarzaniu kolejnych sekcji, a jeśli wszystko przebiegnie pomyślnie, wyświetla komunikat „Loaded services file OK". W przeciwnym wypadku wyświetli jeden z poniższych komunikatów, które pojawiają się także w dziennikach:

„Allow/Deny connection from *host* (*n*) to *udział*"
: Komunikat wyświetlany wtedy, gdy w pliku ustawione są opcje uprawnionych i nieuprawnionych użytkowników. Powinieneś upewnić się, że jesteś na liście uprawnionych użytkowników, a konta root, bin i tym podobne są na liście nie-

uprawnionych użytkowników. W przeciwnym wypadku ty nie będziesz mógł się połączyć, a ci, którzy nie powinni mieć takiej możliwości – owszem.

„Warning: You have some share names that are longer than eight chars"
Ostrzeżenie dla użytkowników Windows for Workgroups i starszych klientów. Nie będą one mogły połączyć się z udziałami o długich nazwach (dłuższych niż osiem znaków) i wyświetlą komunikat o przepełnieniu.

„Warning: [*nazwa*] service MUST be printable!"
W udziale drukarki nie ma opcji `printable = yes`.

„No path in service *nazwa* - using /tmp"
Nie wiadomo, który katalog ma być udostępniony przez udział plikowy, albo jaki katalog buforowy ma być wykorzystywany przez udział drukarki. Jeśli nie podasz ścieżki, udział skorzysta ze ścieżki */tmp*, co nie zawsze odpowiada twoim oczekiwaniom.

„Note: Service *nazwa* is flagged unavailable"
Zwykłe przypomnienie, że w udziale użyłeś opcji `available = no`.

„Can't find include file [*nazwa*]"
Plik konfiguracyjny określony opcją `include` nie istnieje. Jeśli dołączasz ten plik bezwarunkowo, jest to błąd, i to zwykle poważny: udział będzie miał konfigurację inną od zamierzonej. Jeśli dołączasz ten plik, używając zmiennych, na przykład `%a` (architektura), będziesz musiał zdecydować, czy brak pliku dla – dajmy na to – Windows for Workgroups stanowi problem. Często nie stanowi.

„Can't copy service *nazwa*, unable to copy self!"
Spróbowałeś skopiować sekcję pliku *smb.conf* do samej siebie.

„Unable to copy service – source not found: *nazwa*"
Informuje o braku sekcji dołączanej opcją `copy =` lub błędzie w pisowni.

„Ignoring unknown parameter *nazwa*"
Zwykle informuje o przestarzałej, błędnie wpisanej lub nieobsługiwanej opcji.

„Global parameter *nazwa* found in service section".
Informuje, że parametr o zasięgu globalnym znajduje się w definicji udziału. Samba zignoruje ten parametr.

Po zakończeniu testu wykonaj go jeszcze raz, podając w poleceniu `testparm` trzy parametry: nazwę swojego pliku *smb.conf*, nazwę klienta i jego adres IP:

```
testparm katalog_samby/lib/smb.conf klient 192.168.236.10
```

Spowoduje to wykonanie dodatkowego testu, który porównuje nazwę i adres hosta z wartościami opcji `host allow` oraz `host deny` i generuje komunikaty „Allow/Deny connection from *host* (*n*) to *udział*" dla danego klienta. Komunikaty te informują o obecności opcji zezwalających na dostęp lub zabraniających go i określa, czy klient będzie miał dostęp do poszczególnych udziałów, czy też nie. Wprowadzenie polecenia `testparm /usr/local/lib/eksperyment.conf` to wygodny sposób na przetestowanie eksperymentalnego pliku *smb.conf* przed wprowadzeniem go do użytku.

Rozwiązywanie problemów z połączeniami SMB

Kiedy wiesz już, że serwery działają, musisz upewnić się, że działają prawidłowo. Zaczniemy od pliku *smb.conf* w katalogu `katalog_samby/lib`.

Najprostszy plik smb.conf

W poniższych testach zakładamy, że masz udział `[tymcz]` nadający się do celów testowych oraz przynajmniej jedno konto. Oto plik uwzględniający wyłącznie te założenia:

```
[global]
    workgroup = PRZYKLAD
    security = user
    browsable = yes
    local master = yes
[homes]
    guest ok = no
    browsable = no
[tymcz]
    path = /tmp
    public = yes
```

Słowo ostrzeżenia: opcja `public = yes` w udziale `[tymcz]` służy tylko celom testowym. Prawdopodobnie nie chcesz, aby użytkownicy bez kont mogli zapisywać pliki w twoim serwerze, więc wykomentuj tę sekcję, kiedy skończysz testy.

Testowanie konfiguracji lokalnej za pomocą programu smbclient

Pierwszy test pozwoli upewnić się, że serwer potrafi listować swoje usługi (udziały). Wydaj polecenie `smbclient` z opcją `-L localhost`, aby serwer połączył się sam ze sobą, oraz opcją `-U%`, aby określić, że użytkownikiem jest gość. Powinieneś zobaczyć co następuje:

```
serwer% smbclient -L localhost -U%
Server time is Wed May 27 17:57:40 1998 Timezone is UTC-4.0
Server=[localhost]
User=[davecb]
Workgroup=[PRZYKLAD]
Domain=[PRZYKLAD]
    Sharename      Type      Comment
    ---------      ----      -------
    tymcz          Disk
    IPC$           IPC       IPC Service (Samba 1.9.18)
    homes          Disk      Home directories
This machine does not have a browse list
```

Jeśli otrzymasz takie wyniki, przejdź do następnego testu, „Testowanie połączeń za pomocą programu smbclient". W przeciwnym wypadku:

- Jeśli otrzymasz komunikat „Get_hostbyname: unknown host localhost", oznacza to, że albo źle wpisałeś nazwę, albo rzeczywiście występuje jakiś problem (który powinien ujawnić się już w podrozdziale „Testowanie lokalnych usług nazewniczych za pomocą polecenia ping"). W tym drugim przypadku przejdź do podrozdziału „Rozwiązywanie problemów z usługami nazewniczymi".

- Jeśli otrzymasz komunikat „Connect error: Connection refused", oznacza to, że znaleziono serwer, ale nie działa w nim demon *nmbd*. Wróć do podrozdziału „Rozwiązywanie problemów z demonami serwera" i ponownie przetestuj demony.
- Jeśli otrzymasz komunikat „Your server software is being unfriendly", oznacza to, że początkowy pakiet sesji wywołał błędną odpowiedź serwera. Być może serwer uległ awarii lub uruchomił się nieprawidłowo. Przyczyny tego błędu można odkryć, szukając w dziennikach informacji o:
  - błędnych parametrach linii polecenia *smbd*; patrz strona podręcznika man dla *smbd*;
  - poważnych problemach z plikiem *smb.conf*, które uniemożliwiają uruchomienie demona *smbd*. Powinieneś zawsze sprawdzać ten plik po dokonaniu w nim zmian, jak zrobiliśmy to w podrozdziale „Testowanie demonów za pomocą polecenia testparm";
  - braku katalogów, w których Samba przechowuje swoje pliki dziennika i blokady;
  - obecności innego serwera na porcie 139 dla *smbd* i 137 dla *nmbd*, co uniemożliwia uruchomienie naszych demonów.
- Jeśli używasz *inetd*, a nie demonów autonomicznych, sprawdź, czy błąd nie kryje się w plikach */etc/inetd.conf* oraz */etc/services* (zajrzyj na odpowiednie strony podręcznika man).
- Jeśli wyświetlone zostanie pytanie o hasło (`Password:`), oznacza to, że konto gościnne jest źle skonfigurowane. Opcja `%U` informuje program *smbclient*, że należy dokonać „pustego" logowania, które wymaga obecności konta gościnnego, ale nie wymaga nadawania mu jakichkolwiek przywilejów.
- Jeśli otrzymasz komunikat „SMBtconX failed: ERRSRV - ERRaccess", oznacza to, że nie masz prawa dostępu do serwera. Zwykle oznacza to, że wpisałeś opcję `hosts allow`, która nie uwzględnia serwera, albo opcję `hosts deny`, która go uwzględnia. Sprawdź jeszcze raz plik konfiguracyjny za pomocą polecenia `testparm smb.conf` *`nazwa_hosta adres_IP`* (patrz podrozdział „Testowanie demonów za pomocą polecenia testparm") i popraw pomyłkowo ustalone zakazy.

Testowanie połączeń za pomocą programu smbclient

Wydaj polecenie `smbclient \\`*`serwer`*`\tymcz`, które połączy cię z katalogiem */tmp* serwera, aby sprawdzić, czy możesz korzystać z udziału plikowego. Powinieneś otrzymać następujące wyniki:

```
serwer% smbclient '\\serwer\tymcz'
Server time is Tue May  5 09:49:32 1998 Timezone is UTC-4.0
Password:
smb: \> quit
```

- Jeśli otrzymasz komunikat „Get_Hostbyname: Unknown host name", „Connect error: Connection refused" lub „Your server software is being unfriendly", możliwe przyczyny tych błędów znajdziesz w podrozdziale „Testowanie konfiguracji lokalnej za pomocą programu smbclient".

- Jeśli otrzymasz komunikat „\serwertymcz: Not enough '\' characters in service", prawdopodobnie nie ujałeś adresu w cudzysłów, więc Unix usunął jeden z ukośników. Możesz zapisać polecenie także w postaci:

```
smbclient \\\\serwer\\tymcz
```

lub

```
smbclient //serwer/tymcz
```

Teraz wpisz swoje hasło uniksowego konta w linii zgłoszenia `Password`. Jeśli zobaczysz zgłoszenie `smb:\>`, oznacza to, że połączenie się powiodło. Wpisz `quit` i przejdź do podrozdziału „Testowanie połączeń za pomocą polecenia NET USE". Jeśli zaś otrzymasz komunikat „SMBtconX failed. ERRSRV - ERRinvnetname", oznacza to, że wystąpił jeden z poniższych problemów:

- Błędna nazwa udziału: wpisałeś ją niepoprawnie, jest za długa, występują w niej duże i małe litery albo udział jest niedostępny. Sprawdź za pomocą polecenia *testparm*, czy dany udział powinien być dostępny (patrz podrozdział „Testowanie demonów za pomocą polecenia testparm").
- Używasz opcji `security = share`. W takim przypadku zapewne będziesz musiał dodać opcję `-U` *`twoje_konto`* do polecenia *smbclient* albo znać hasło uniksowego konta o nazwie „temp".
- Błędna nazwa użytkownika.
- Błędne hasło.
- Opcja `valid users` lub `invalid users` w pliku *smb.conf*, która uniemożliwia ci nawiązanie połączenia. Ponownie wydaj polecenie `testparm smb.conf` *`nazwa_hosta adres_IP`* (patrz podrozdział „Testowanie demonów za pomocą polecenia testparm").
- Opcja `allow hosts`, która nie uwzględnia serwera, albo `deny hosts`, która go uwzględnia. To także można sprawdzić za pomocą polecenia *testparm*.
- Problem z uwierzytelnianiem, na przykład gdy serwer używa haseł ukrytych (*shadow*) lub modułów uwierzytelniających PAM (*Pluggable Authentication Module*), a Samba nie została skompilowana z ich obsługą. Jest to rzadkie, ale czasem się zdarza, na przykład kiedy pliki binarne Samby skompilowane w SunOS 4 (bez haseł ukrytych) zostaną uruchomione bez rekompilacji w Solarisie (z hasłami ukrytymi).
- W pliku konfiguracyjnym jest opcja `enorypted password = yes`, ale w pliku *smbpasswd* nie ma hasła dla twojego konta.
- Masz puste hasło w uniksowym pliku */etc/passwd* lub pliku *smbpasswd*.
- Łączysz się z udziałem `[tymcz]`, a w sekcji `[tymcz]` pliku *smb.conf* nie ma opcji `guest ok = yes`.
- Łączysz się z udziałem `[tymcz]` przed połączeniem się ze swoim katalogiem macierzystym, a twoje konto gościa jest błędnie skonfigurowane. Jeśli możesz połączyć się najpierw ze swoim katalogiem macierzystym, a później z udziałem `[tymcz]`, problem polega właśnie na tym. Zajrzyj do rozdziału 2, w którym znajdziesz więcej informacji o tworzeniu podstawowego pliku konfiguracyjnego Samby.

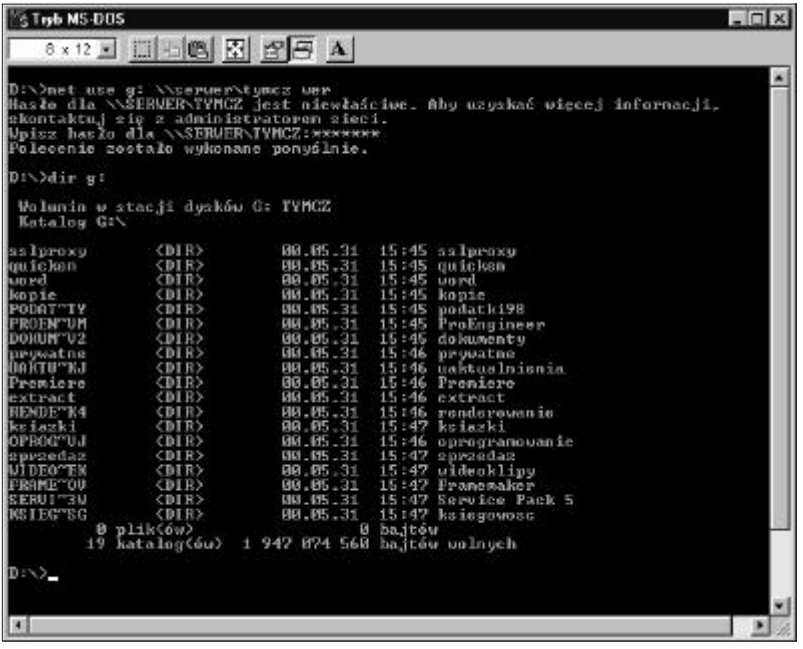
Niepoprawne konto gościa uniemożliwi także drukowanie i przeglądanie, dopóki nie zalogujesz się w swoim katalogu macierzystym.

Istnieje jeszcze jedna możliwa przyczyna takiego błędu, która nie ma nic wspólnego z hasłem: linia `path=` w pliku *smb.conf* wskazuje na katalog, który nie istnieje. Polecenie *testparm* nie wykryje takiego błędu, a większość klientów SMB nie potrafi odróżnić go od innych typów błędów związanych z niepoprawnymi kontami użytkowników. Będziesz musiał sprawdzić to ręcznie.

Kiedy uda ci się połączyć z udziałem `[tymcz]`, powtórz test, tym razem logując się w swoim katalogu macierzystym (na przykład przez mapowanie dysku sieciowego *serwer*`\davecb`) i upewnij się, że wszystko przebiega bezbłędnie. Jeśli będziesz musiał coś skorygować, ponownie przetestuj udział `[tymcz]`.

Testowanie połączeń za pomocą polecenia NET USE

Wydaj polecenie `net use * \`*serwer*`\tymcz` w kliencie DOS-a lub Windows i sprawdź, czy możesz połączyć się z serwerem. Powinieneś zobaczyć pytanie o hasło, a następnie otrzymać komunikat „Polecenie zostało wykonane pomyślnie" (patrz rysunek 9.2).



Rysunek 9.2. Wyniki polecenia NET USE

Jeśli pomyślnie nawiązałeś połączenie, przejdź do podrozdziału „Testowanie połączeń za pomocą Eksploratora Windows". W przeciwnym wypadku:

- Jeśli otrzymasz komunikat „Odnalezienie podanego, udostępnionego katalogu jest niemożliwe", oznacza to, że nazwa udziału została wpisana błędnie lub nie

ma jej w pliku *smb.conf*. Komunikat ten może także ostrzegać o nazwach zawierających mieszane litery oraz spacje lub dłuższych niż osiem znaków.

- Jeśli otrzymasz komunikat „Lokalizacja nazwy komputera określonej w ścieżce sieciowej jest niemożliwa", oznacza to, że błędnie wpisałeś nazwę komputera, zawiodły usługi nazewnicze, występuje problem z siecią, albo opcja `hosts deny =` uwzględnia twojego hosta.
  - Jeśli nie popełniłeś błędu w pisowni, powinieneś cofnąć się przynajmniej do podrozdziału „Testowanie połączeń za pomocą programu smbclient", aby sprawdzić, czemu połączenie się nie powiodło.
  - Jeśli program *smbclient* działa, problem polega na błędnej pracy usług nazewniczych w kliencie. Powinieneś przejść do podrozdziału „Testowanie serwera za pomocą polecenia nmblookup" i sprawdzić, czy możesz wyszukać serwer i klienta.
- Jeśli otrzymasz komunikat „Hasło dla `\\serwer\tymcz` jest niewłaściwe", kopia hasła przechowywana lokalnie w kliencie nie odpowiada hasłu przechowywanemu w serwerze. Zostaniesz poproszony o wpisanie właściwego hasła.

Klienty Windows 95 i 98 przechowują lokalny plik *password*, ale jest to po prostu zapamiętana kopia hasła wysyłanego do Samby i serwerów NT w celu uwierzytelnienia użytkownika i właśnie o to hasło jesteś pytany. W samym komputerze Windows (ale nie NT) możesz zalogować się bez podawania hasła.

Jeśli podane przez ciebie hasło nie zostanie przyjęte, może to znaczyć, że: hasło nie odpowiada temu przechowywanemu w serwerze, opcja `valid users` lub `invalid users` zabrania ci dostępu, komunikację zakłóca protokół NetBEUI albo występuje problem z zaszyfrowanymi hasłami opisany w następnym akapicie.

- Jeśli twoim klientem jest NT 4.0, NT 3.5 z SP3, Windows 95 z SP3, Windows 98 lub dowolny z tych systemów z zainstalowanym Internet Explorerem 4.0, to domyślnie używa on haseł zaszyfrowanych algorytmem Microsoftu (omawiamy to w rozdziale 6, w podrozdziale „Hasła", gdzie podajemy także alternatywne rozwiązania). Ogólnie rzecz biorąc, jeśli niedawno zainstalowałeś którąś z bardziej rozbudowanych aplikacji Microsoftu, być może wraz z nią zainstalowałeś uaktualnienie systemu i włączyłeś szyfrowanie haseł.

Internet Explorer uznaje adresy URL typu *file://pewienhost/pewienplik* i nawiązuje z nimi połączenia SMB, a klienty Windows aż do wersji Windows 95 z poprawką 2 wysyłały hasła jawnym tekstem do serwerów SMB w Internecie. Nie było to najszczęśliwsze rozwiązanie, więc Microsoft dość szybko zdecydował, że w protokole SMB będą używane wyłącznie zaszyfrowane hasła. Odpowiednia poprawka wchodzi w skład kolejnych wersji wszystkich produktów Microsoftu. Jeśli tylko nie używasz Internet Explorera bez zapory sieciowej, zaszyfrowane hasła tak naprawdę nie są potrzebne, więc w prywatnych sieciach możesz śmiało korzystać z haseł niezaszyfrowanych.

- Jeśli hasło uniksowe składa się z mieszanki małych i dużych liter, klient prawdopodobnie zrównuje wielkość wszystkich liter w haśle. Jeśli zmiana liter hasła na jedną wielkość odniesie skutek, problem polega właśnie na tym. Wszystkie klien-

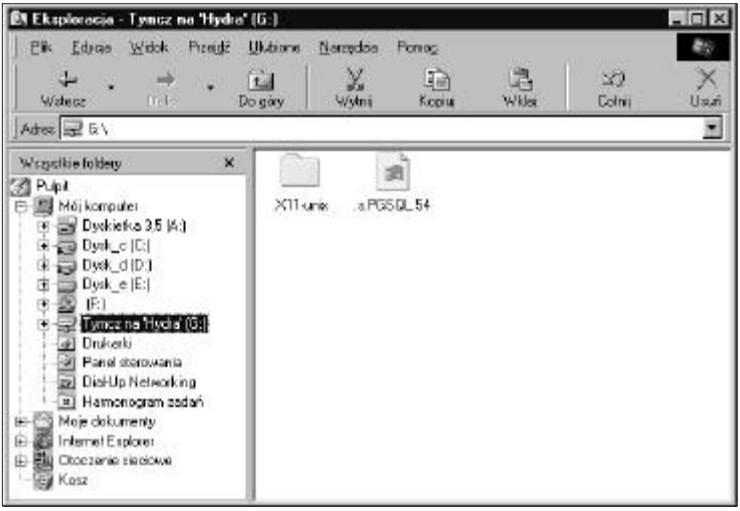
ty (oprócz najstarszych) obsługują hasła pisane dużymi literami, więc Samba próbuje je dopasować, używając raz dużych, a raz małych liter. Jeśli chcesz korzystać z haseł pisanych literami różnej wielkości, możesz skorzystać z opcji `password level`, opisanej w rozdziale 6.

- Możesz mieć problem z opcją `valid users`, co można sprawdzić za pomocą programu *smbclient* (patrz podrozdział „Testowanie połączeń za pomocą programu smbclient").
- Z usługą klienta sieci Microsoft może być powiązany protokół NetBEUI. Taka konfiguracja często powoduje długie opóźnienia i nieoczekiwane błędy, a czasem prowadzi także do odrzucania poprawnych haseł.

Termin „powiązanie" oznacza tutaj połączenie jednego elementu oprogramowania z innym. Klient SMB Microsoftu jest „wiązany" z TCP/IP na karcie Powiązania okna Właściwości: Protokół TCP/IP otwieranego z apletu Sieć Panelu sterowania. TCP/IP jest z kolei powiązany z kartą Ethernetu. Słowo to oznacza tutaj coś innego, niż w przypadku wiązania demona SMB z portem TCP/IP.

Testowanie połączeń za pomocą Eksploratora Windows

Uruchom Eksploratora Windows lub Eksploratora Windows NT (nie Internet Explorera), wybierz polecenie Narzędzia⇨Mapuj dysk sieciowy i wpisz `\\serwer\tymcz`, aby sprawdzić, czy uda ci się skłonić Eksploratora do połączenia z katalogiem */tmp*. Powinieneś zobaczyć okno podobne do tego z rysunku 9.3. Jeśli tak jest, połączenie powiodło się, więc możesz przejść do podrozdziału „Rozwiązywanie problemów z przeglądaniem".



Rysunek 9.3. Dostęp do katalogu */tmp* z Eksploratora Windows

Słowo ostrzeżenia: Eksplorator Windows i Eksplorator Windows NT nie są najlepszymi narzędziami diagnostycznymi. Informują cię, że wystąpił błąd, ale rzadko

wskazują jego przyczynę. Jeśli połączenie się nie powiedzie, powinieneś użyć polecenia NET USE, które znacznie dokładniej opisuje błędy.

- Jeśli otrzymasz komunikat „Hasło dla tego połączenia jest już niepoprawne", przyczyny mogą być następujące:
  - Kopia hasła zapamiętana lokalnie w kliencie nie odpowiada hasłu przechowywanemu w serwerze.
  - Nie podałeś nazwy użytkownika i hasła, kiedy logowałeś się w kliencie. Eksplorator często nadal wysyła pustą nazwę użytkownika i puste hasło, nawet wtedy, gdy w międzyczasie podałeś już hasło.
  - Błędnie wpisałeś hasło.
  - Istnieją opcje `invalid users` lub `valid users`, które zakazują ci dostępu.
  - Twój klient to NT 4.0, NT 3.5 z SP3, Windows 95 z SP3, Windows 98 lub dowolny z tych systemów z zainstalowanym Internet Explorerem 4.0. Wszystkie te klienty korzystają z zaszyfrowanych haseł.
  - Hasło składa się z liter różnej wielkości, a twój klient zrównuje wielkość liter w haśle.
- Jeśli otrzymasz komunikat „Odnalezienie komputera lub udziału jest niemożliwe", przyczyny mogą być następujące:
  - źle wpisana nazwa,
  - błędnie działająca usługa,
  - awaria udziału,
  - problem z siecią,
  - błędna linia `path`,
  - linia `hosts deny`, która uniemożliwia ci dostęp.
- Jeśli otrzymasz komunikat „Aby uzyskać to połączenie, musisz podać hasło", oznacza to, że hasło w kliencie nie jest zsynchronizowane z hasłem w serwerze albo wpisałeś to hasło pierwszy raz i klient jeszcze nie zapamiętał go lokalnie.
- Jeśli otrzymasz komunikat „Odnalezienie podanej nazwy udziału nie jest możliwe", oznacza to, że: podałeś błędną nazwę udziału lub użyłeś złej składni, nazwa udziału ma więcej niż osiem znaków, zawiera spacje lub składa się z liter o różnej wielkości.

Kiedy zdołasz się bezproblemowo połączyć z udziałem `[tymcz]`, spróbuj połączyć się ze swoim katalogiem macierzystym. Jeśli będziesz musiał dokonać zmian, żeby uzyskać dostęp do katalogów macierzystych, wówczas jeszcze raz przetestuj połączenie z udziałem `[tymcz]` i vice versa – o czym była już mowa w podrozdziale „Testowanie połączeń za pomocą polecenia NET USE". Jeśli Eksplorator zawiedzie, wróć do tego podrozdziału i tam spróbuj postawić diagnozę.

Rozwiązywanie problemów z przeglądaniem

Teraz pora zająć się przeglądaniem. Zostawiliśmy je na koniec, nie dlatego, że jest najtrudniejsze, ale dlatego, że jest opcjonalne i częściowo oparte na protokole, który nie gwarantuje dostarczenia pakietów. Przeglądanie bywa trudne do zdiagnozowania, jeśli nie jesteś pewien, czy działają wszystkie inne usługi.

Przeglądanie jest całkowicie opcjonalne: jest to po prostu sposób na wyszukanie w sieci serwerów i oferowanych przez nie usług. Unix nie dysponuje podobnym mechanizmem i jakoś sobie bez niego radzi. Przeglądanie wymaga, aby wszystkie komputery znajdowały się w sieci lokalnej, gdzie dozwolone są rozgłoszenia.

Mechanizm przeglądania najpierw identyfikuje komputer za pomocą zawodnego protokołu UDP, a następnie nawiązuje zwykłe (niezawodne) połączenie TCP/IP, aby uzyskać listę udziałów udostępnianych przez komputer.

Testowanie przeglądania za pomocą programu smbclient

Zaczniemy od przetestowania niezawodnych połączeń. Z serwera spróbuj wylistować jego własne udziały, używając programu *smbclient* z opcją `-L` i nazwą serwera. Powinieneś uzyskać następujące wyniki:

```
serwer% smbclient -L serwer
Added interface ip=192.168.236.86 bcast=192.168.236.255 nmask=255.255.255.0
Server time is Tue Apr 28 09:57:28 1998 Timezone is UTC-4.0
Password:
Domain=[PRZYKLAD]
OS=[Unix]
Server=[Samba 1.9.18]
Server=[serwer]
User=[davecb]
Workgroup=[PRZYKLAD]
Domain=[PRZYKLAD]
        Sharename      Type      Comment
        ---------      ----      -------
        cdrom          Disk      CD-ROM
        cl             Printer   Color Printer 1
        davecb         Disk      Home Directories

This machine has a browse list:
        Server               Comment
        ---------            -------
        SERWER               Samba 1.9.18

This machine has a workgroup list:
        Workgroup            Master
        ---------            -------
        PRZYKLAD             SERWER
```

- Jeśli nie została wyświetlona lista udziałów, oznacza to, że serwer nie pozwala na przeglądanie żadnych udziałów. Coś takiego nie powinno mieć miejsca, jeśli przetestowałeś dowolny udział za pomocą Eksploratora Windows lub polecenia NET USE. Jeśli nie przeprowadziłeś jeszcze testu `smbclient -L localhost -U%` (patrz podrozdział „Testowanie konfiguracji lokalnej za pomocą programu smbclient"), zrób to teraz. Być może udziały nie są wyświetlane ze względu na błędną

konfigurację konta gościa. Sprawdź także plik *smb.conf* i upewnij się, że nie ma w nim opcji `browsable = no`; proponujemy skorzystać z minimalnego pliku *smb.conf* (patrz podrozdział „Najprostszy plik smb.conf"). Opcja `browsable` musi być włączona, abyś mógł zobaczyć przynajmniej udział `[tymcz]`.

- Jeśli nie uzyskałeś listy przeglądania, oznacza to, że serwer nie dostarcza informacji o komputerach w sieci. Przynajmniej jeden komputer musi obsługiwać listy przeglądania. Jeśli chcesz, żeby Samba była główną przeglądarką lokalną, umieść opcję `local master = yes` w pliku *smb.conf*.
- Jeśli uzyskałeś listę przeglądania, ale bez udziału `[tymcz]`, prawdopodobnie problem kryje się w pliku *smb.conf*. Cofnij się do podrozdziału „Testowanie demonów za pomocą polecenia testparm".
- Jeśli nazwa twojej grupy nie pojawiła się na liście grup roboczych, być może grupa jest źle skonfigurowana w pliku *smb.conf*.
- Jeśli w ogóle nie uzyskałeś listy grup roboczych, sprawdź, czy w pliku *smb.conf* jest opcja `workgroup = PRZYKLAD`.
- Jeśli nie otrzymasz żadnych wyników, spróbuj jeszcze raz, używając opcji `-I` *`adres_ip`* `-n` *`nazwa_netbiosowa`* `-W` *`grupa_robocza`* `-d3` i wpisując nazwę NetBIOS-ową i nazwę grupy roboczej dużymi literami (opcja `-d3` ustawia poziom diagnostyczny równy 3).

Jeśli nadal nie otrzymujesz żadnych wyników, nie powinieneś dojść do tego miejsca. Cofnij się przynajmniej do rozdziału „Testowanie TCP za pomocą FTP", a nawet „Testowanie połączeń za pomocą polecenia ping". Natomiast:

- Jeśli otrzymasz komunikat „SMBtconX failed. ERRSRV – ERRaccess", oznacza to, że nie masz prawa dostępu do serwera. Zwykle oznacza to, że istnieje opcja `hosts allow`, która nie uwzględnia serwera, albo opcja `hosts deny`, która go uwzględnia.
- Jeśli otrzymasz komunikat „Bad Password", przyczyną może być:
  - Błędna linia `hosts allow` lub `hosts deny`.
  - Błędna linia `valid users` lub `invalid users`.
  - Hasło składające się z małych liter oraz klienty OS/2 i Windows for Workgroups.
  - Nieistniejące lub błędnie skonfigurowane konto gościnne.

  Sprawdź, które konto jest używane jako gościnne (patrz podrozdział „Testowanie konfiguracji lokalnej za pomocą programu smbclient"), zweryfikuj plik *smb.conf* za pomocą polecenia `testparm smb.conf` *`nazwa_hosta adres_ip`* (patrz podrozdział „Testowanie demonów za pomocą polecenia testparm") i zmień lub wykomentuj wszystkie linie `hosts allow`, `hosts deny`, `valid users` i `invalid users`.
- Jeśli otrzymasz komunikat „Connection refused", oznacza to, że serwer *smbd* nie działa lub uległ awarii. Za pomocą polecenia *netstat* sprawdź, czy serwer jest aktywny i nasłuchuje w sieci (patrz podrozdział „Testowanie demonów za pomocą polecenia testparm").

- Jeśli otrzymasz komunikat „Get_Hostbyname: Unknown host name", przyczyny mogą być następujące: zrobiłeś błąd w pisowni, istnieje niezgodność między uniksową a NetBIOS-ową nazwą hosta albo błędnie działają usługi nazewnicze. Przetestuj usługi nazewnicze według wskazówek z podrozdziału „Testowanie połączeń za pomocą polecenia NET USE". Jeśli to zadziała, możesz podejrzewać niezgodność nazw, więc przejdź do podrozdziału „Rozwiązywanie problemów z nazwami NetBIOS-owymi".
- Jeśli otrzymasz komunikat „Session request failed", oznacza to, że serwer odrzucił połączenie. Zwykle jest to spowodowane błędem wewnętrznym, na przykład ilością pamięci nie wystarczającą do rozwidlenia procesu.
- Jeśli otrzymasz komunikat „Your server software is being unfriendly", oznacza to, że początkowy pakiet sesji wywołał błędną odpowiedź serwera. Być może serwer uległ awarii lub uruchomił się nieprawidłowo. Przejdź do podrozdziału „Testowanie konfiguracji lokalnej za pomocą programu smbclient", gdzie zanalizowano ten problem.
- Jeśli podejrzewasz, że serwer nie działa, przejdź do podrozdziału „Wyszukiwanie procesów demona za pomocą polecenia ps" i upewnij się, że właśnie to jest przyczyną braku odpowiedzi.

Testowanie serwera za pomocą polecenia nmblookup

Ten test zbada system „reklamowy", używany przez mechanizm przeglądania i usługi nazewnicze Windows. Zadaniem tego systemu jest poinformowanie o swojej obecności i gotowości do świadczenia usług. Właśnie ten element przeglądania korzysta z zawodnego protokołu (UDP) i działa tylko w sieciach rozgłoszeniowych, takich jak Ethernet. Program *nmblookup* rozgłasza zapytania o określoną przez ciebie nazwę i zwraca adres IP oraz nazwę komputera, podobnie jak czyni program *nslookup* z nazwami DNS. W poniższych przykładach opcja `-d` określa poziom diagnostyczny, a opcja `-B` kieruje zapytanie do określonego komputera.

Najpierw sprawdzimy sam serwer. Wydaj polecenie *nmblookup* z opcją `-B` i nazwą serwera, aby wysłać zapytanie do serwera Samby, oraz z parametrem `__SAMBA__` jako nazwą do wyszukania. Powinieneś uzyskać następujące wyniki:

```
serwer% nmblookup -B serwer __SAMBA__
Added interface ip=192.168.236.86 bcast=192.168.236.255 nmask =
255.255.255.0
Sending queries to 192.168.236.86 192.168.236.86 __SAMBA__
```

Powinieneś uzyskać adres IP serwera, a po nim nazwę `__SAMBA__`, co oznacza, że serwer ogłosił, że dysponuje usługą o tej nazwie `__SAMBA__`, a zatem poprawnie działa co najmniej część NetBIOS-owych usług nazewniczych.

- Jeśli otrzymasz komunikat „Name_query failed to find name __SAMBA__", być może podałeś zły adres w opcji `-B` albo nie działa demon *nmbd*. Opcja `-B` w istocie przyjmuje adres rozgłoszeniowy; my podaliśmy nazwę komputera, aby określić adres jednostkowy i zapytać serwer, czy rości sobie prawa do nazwy __SAMBA__.

- Spróbuj jeszcze raz, używając opcji `-B adres_ip`, a jeśli to również się nie powiedzie, będzie to znaczyło, że *nmbd* nie uznał nazwy za własną. Wróć na chwilę do podrozdziału „Testowanie demonów za pomocą polecenia testparm" i sprawdź, czy demon *nmbd* działa. Jeśli tak, być może nie przywłaszcza sobie nazw; oznacza to, że Samba nie udostępnia usług przeglądania (zapewne błąd w konfiguracji). W takim przypadku sprawdź, czy w pliku *smb.conf* nie ma opcji `browsing = no`.

Testowanie klienta za pomocą polecenia nmblookup

Następnie przetestuj adres klienta, wpisując polecenie *nmblookup* z opcją `-B` i nazwą klienta oraz parametrem `'*'`, który oznacza „cokolwiek", jak w przykładzie poniżej:

```
serwer% nmblookup -B klient '*'
Sending queries to 192.168.236.10 192.168.236.10 *
Got a positive name query response from 192.168.236.10 (192.168.236.10)
```

- Jeśli otrzymasz komunikat „Name_query failed to find name *", oznacza to, że popełniłeś pomyłkę przy wpisywaniu polecenia lub że oprogramowanie klienckie w komputerze PC nie jest zainstalowane, uruchomione lub powiązane z protokołem TCP/IP. Zajrzyj do rozdziałów 2 i 3, aby upewnić się, że oprogramowanie klienckie jest zainstalowane i nasłuchuje w sieci.

Jeśli wystąpiły jakieś błędy, powtórz to polecenie z następującymi opcjami:

- Jeśli polecenie `nmblookup -B adres_IP_klienta` działa, a `-B nazwa_klienta` nie, oznacza to, że usługi nazewnicze mają problem z nazwą klienta. Przejdź do podrozdziału „Rozwiązywanie problemów z usługami nazewniczymi".
- Jeśli polecenie `nmblookup -B 127.0.0.1 '*'` działa, a `-B adres_IP_klienta` nie, wina leży po stronie sprzętu i polecenie *ping* również nie powinno działać. Skontaktuj się z administratorem sieci.

Testowanie sieci za pomocą polecenia nmblookup

Wydaj polecenie *nmblookup* z opcją `-d 2` (poziom diagnostyczny 2) i parametrem `'*'`. Tym razem sprawdzamy, czy programy (takie jak *nmblookup*) mogą używać rozgłoszeń. Jest to zasadniczo test łączności, korzystający z domyślnego adresu rozgłoszeniowego.

Hosty używające NetBIOS-u i TCP/IP powinny odpowiedzieć komunikatami „got a positive name query response". Samba może nie wyłapać wszystkich odpowiedzi w czasie działania polecenia, więc nie zawsze zobaczysz wszystkie klienty w sieci. Powinieneś jednak uzyskać odpowiedzi od większości z nich:

```
serwer% nmblookup -d 2 '*'
Added interface ip=192.168.236.86 bcast=192.168.236.255 nmask = 255.255.255.0
Sending queries to 192.168.236.255
Got a positive name query response from 192.168.236.191 (192.168.236.191)
Got a positive name query response from 192.168.236.228 (192.168.236.228)
Got a positive name query response from 192.168.236.75 (192.168.236.75)
Got a positive name query response from 192.168.236.79 (192.168.236.79)
Got a positive name query response from 192.168.236.206 (192.168.236.206)
Got a positive name query response from 192.168.236.207 (192.168.236.207)
```

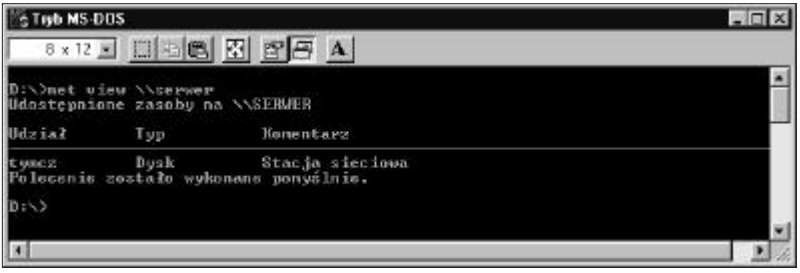
```
Got a positive name query response from 192.168.236.217 (192.168.236.217)
Got a positive name query response from 192.168.236.72 (192.168.236.72) 192.168.
236.86 *
```

Jednakże:

- Jeśli w wynikach nie zobaczysz uprzednio przetestowanego adresu klienta, domyślny adres rozgłoszeniowy jest błędny. Wypróbuj polecenie `nmblookup -B 255.255.255.255 -d 2 '*'` (adres rozgłoszeniowy składający się z samych jedynek). Jeśli pojawią się odpowiedzi, oznacza to, że użyty poprzednio adres rozgłoszeniowy jest niepoprawny. Rozwiązanie tego problemu omówiono w dalszym podrozdziale „Adresy rozgłoszeniowe".
- Jeśli adres 255.255.255.255 również nie zadziała, przypomnij sobie, czy klient i serwer nie znajdują się w innych podsieciach (ustaliłeś to podczas testowania połączeń za pomocą polecenia *ping*). Powinieneś diagnozować przeglądanie, używając serwera i klienta w tej samej podsieci, ale jeśli jest to niemożliwe, spróbuj podać adres rozgłoszeniowy zdalnej podsieci w opcji `-B`. Metodę ustalania tego adresu omawiamy w podrozdziale poświęconym rozwiązywaniu problemów z rozgłoszeniami, „Adresy rozgłoszeniowe". Opcja `-B` może zadziałać, jeśli twój ruter obsługuje ukierunkowane rozgłoszenia; jeśli nie, będziesz musiał wykonać ten test na kliencie z tej samej podsieci.

Testowanie przeglądania w kliencie za pomocą polecenia NET VIEW

W kliencie wpisz polecenie `net view \\serwer` w oknie DOS-a, aby zobaczyć, czy możesz połączyć się z serwerem i zapytać go o udostępniane udziały. Powinieneś zobaczyć listę udostępnianych zasobów, jak na rysunku 9.4.



Rysunek 9.4. Użycie polecenia NET VIEW

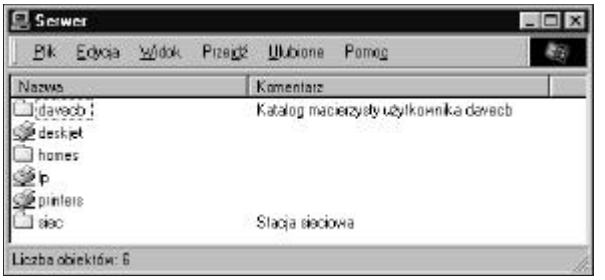
Jeśli uzyskałeś taką listę, przejdź do podrozdziału „Inne problemy".

- Jeśli otrzymasz komunikat „Lokalizacja nazwy komputera określonej w ścieżce sieciowej jest niemożliwa" dla nazwy, którą już przetestowałeś w podrozdziale „Testowanie klienta za pomocą polecenia nmblookup", wina leży po stronie oprogramowania klienta. Upewnij się, że tak jest w istocie, wykonując polecenie *nmblookup* z nazwą klienta; jeśli to zadziała, a polecenie NET VIEW nie, za błąd odpowiedzialny jest klient.

- Oczywiście, jeśli polecenie *nmblookup* nie powiedzie się, oznacza to problem z usługami nazewniczymi NetBIOS-u, co omówiono w podrozdziale „Rozwiązywanie problemów z nazwami NetBIOS-owymi".
- Jeśli otrzymasz komunikat „Brak koniecznych uprawnień do używania zasobu sieciowego" albo „Podany serwer nie jest skonfigurowany do akceptowania wpisanego polecenia", oznacza to, że albo twoje konto gościnne jest źle skonfigurowane (patrz podrozdział „Testowanie konfiguracji lokalnej za pomocą programu smbclient"), albo w pliku konfiguracyjnym istnieje linia `hosts allow` lub `hosts deny`, która uniemożliwia dostęp twojemu komputerowi. Problemy te powinny zostać wykryte podczas testów opisanych w podrozdziale „Testowanie przeglądania za pomocą programu smbclient".
- Jeśli otrzymasz komunikat „Podany komputer nie otrzymuje żądań", może to oznaczać, że: błędnie wpisałeś nazwę, komputer jest nieosiągalny dla transmisji rozgłoszeniowych (co sprawdzaliśmy w podrozdziale „Testowanie sieci za pomocą polecenia nmblookup") lub nie działa w nim demon *nmbd*.
- Jeśli otrzymasz komunikat „Podane hasło sieciowe jest niepoprawne", prawdopodobnie wystąpił problem z zaszyfrowanymi hasłami, który omówiono w rozdziale 6 (tam też zasugerowano rozwiązanie).

Przeglądanie zasobów serwera z klienta

Spróbuj przejrzeć zasoby serwera w oknie Otoczenia sieciowego (w starszych wersjach Windows użyj Menedżera plików). Serwer Samby powinien widnieć na liście przeglądania lokalnej grupy roboczej. Po dwukrotnym kliknięciu nazwy serwera powinieneś uzyskać listę udziałów (patrz rysunek 9.5).



Rysunek 9.5. Lista udziałów serwera

- Jeśli otrzymasz komunikat „Hasło nie jest poprawne" w klientach NT 4.0, NT 3.5 z SP3, Windows 95 z SP3 , Windows 98 lub w dowolnym z tych systemów z zainstalowanym Internet Explorerem 4.0, przyczyną jest zapewne szyfrowanie haseł. Wszystkie te klienty używają algorytmu Microsoftu do szyfrowania haseł (patrz rozdział 6).
- Jeśli otrzymasz komunikat „Przeglądanie sieci nie jest możliwe", przyczyny mogą być następujące:

- Przeglądasz sieć zbyt wcześnie, zanim dobiegły końca transmisje rozgłoszeniowe i uaktualnienia; odczekaj 30 sekund przed kolejną próbą.
- W sieci istnieje problem, którego dotąd nie zdiagnozowaliśmy.
- W sieci nie ma głównej przeglądarki. Dodaj opcję konfiguracyjną `local master = yes` do pliku *smb.conf*.
- Żaden z udziałów w pliku *smb.conf* nie zawiera opcji `browsable = yes`.

- Jeśli otrzymasz komunikat „*serwer* nie jest dostępny", może to oznaczać, że:
  - Występuje problem z zaszyfrowanymi hasłami.
  - Komputer rzeczywiście nie jest dostępny.
  - Komputer nie obsługuje przeglądania.

Inne problemy

Jeśli dotarłeś do tego miejsca, to znaczy, że rozwiązałeś problem, albo trafiłeś na taki, z którym się jeszcze nie spotkaliśmy. W następnych podrozdziałach omawiamy czynności konieczne do zapewnienia infrastruktury dla pracy Samby.

Praca bez zalogowania się

Czasem może się zdarzyć, że zapomnisz zalogować się w kliencie albo zalogujesz się jako nieprawidłowy (nie mający konta) użytkownik. W tym pierwszym przypadku nie otrzymasz żadnego ostrzeżenia: Windows próbuje być przyjazny i pozwala ci korzystać z komputera. Tylko lokalnie! W tym drugim przypadku Windows po prostu powita cię i utworzy nowe konto. W obu przypadkach spotkasz się z odmowami połączenia i powtarzającymi się prośbami o podanie hasła. Jeśli nic nie da się na to poradzić, spróbuj wylogować się lub zamknąć system i zalogować się ponownie.

Rozwiązywanie problemów z usługami nazewniczymi

W tym podrozdziale zajmiemy się rozwiązywaniem problemów z usługami nazewniczymi, ograniczając się do tych, które mają wpływ na pracę Samby.

Diagnozowanie różnych usług nazewniczych jest opisane w kilku dobrych książkach: Domain Name Service (DNS) w książce Paula Albitza i Cricketa Liu *DNS i Bind* (wydanej przez Wydawnictwo RM), NIS („Yellow Pages") w książce Hala Sterna *NFS and NIS* (obie opublikowane przez wydawnictwo O'Reilly), natomiast Windows Internet Name Service (WINS), pliki *hosts* i *LMHOSTS* oraz NIS+ są najdokładniej omówione w podręcznikach producentów.

Problemy, które omówimy w tym podrozdziale, to:

- Identyfikowanie usług nazewniczych.
- Nie można wyszukać nazwy hosta.
- Długa forma nazwy hosta działa, ale krótka nie.
- Krótka forma nazwy hosta działa, ale długa nie.
- Przed uzyskaniem oczekiwanego rezultatu następuje długa zwłoka.

Identyfikowanie usług nazewniczych

Najpierw sprawdź, czy serwer i klient używają DNS, WINS, NIS lub plików *hosts*, aby określić adres IP hosta na podstawie podanej nazwy. Różne typy komputerów mają różne priorytety:

- Komputery Windows 95 i 98 sprawdzają najpierw WINS i plik *LMHOSTS*, następnie używają rozgłoszeń, a na końcu sprawdzają DNS i plik *hosts*.
- Komputery NT najpierw sprawdzają WINS, następnie używają rozgłoszeń, wreszcie sprawdzają plik *hosts* i DNS.
- Programy dla Windows używające standardu Winsock (na przykład klienty NFS dla komputerów PC) sprawdzają plik *hosts*, DNS, WINS, a wreszcie używają rozgłoszeń. Nie zakładaj, że jeśli działa usługa nazewnicza innego programu, to będzie działać również usługa nazewnicza klienta SMB!
- Demony Samby sprawdzają plik *LMHOSTS*, WINS, później uniksowe mechanizmy nazewnicze, a na końcu używają rozgłoszeń.
- Hosty uniksowe można skonfigurować tak, aby używały dowolnej kombinacji DNS, pliku *hosts* oraz NIS i NIS+, praktycznie w dowolnej kolejności.

Zalecamy, aby klienty używały WINS i DNS, podobnie jak demony Samby, natomiast serwer uniksowy powinien używać DNS. Będziesz musiał sprawdzić swoje notatki i poszczególne komputery, aby dowiedzieć się, jakie usługi są w użyciu.

W klientach wszystkie usługi nazewnicze konfiguruje się w oknie Właściwości: Protokół TCP/IP otwieranym z apletu Sieć Panelu sterowania, co omówiono w rozdziale 3. Być może będziesz musiał tam zajrzeć, aby dowiedzieć się, które usługi są włączone. W serwerze sprawdź, czy istnieje plik */etc/resolv.conf*. Jeśli istnieje, oznacza to, że używasz DNS. Być może korzystasz jednak także z innych usług. Będziesz musiał sprawdzić, czy używasz NIS lub pewnej kombinacji usług nazewniczych.

W Solarisie i Systemie V sprawdź, czy istnieje plik */etc/nsswitch.conf*. Jeśli tak, poszukaj w nim linii zaczynającej się od słowa `host:`, po którym następuje jeden lub kilka spośród następujących parametrów: `files`, `bind`, `nis` lub `nis+`. Są to używane usługi nazewnicze, wymienione w kolejności, przy czym w nawiasach kwadratowych mogą znajdować się ich dodatkowe parametry. Parametr *files* oznacza użycie pliku *hosts*, natomiast parametr *bind* (skrót od *Berkeley Internet Name Daemon*) oznacza użycie DNS.

Jeśli klient i serwer stosują inne usługi, najpierw należy je zsynchronizować. Klienty mogą używać tylko DNS, WINS, plików *hosts* i plików *lmhosts*, lecz nie NIS i NIS+. Serwery mogą używać plików *hosts*, DNS, NIS i NIS+, ale nie mogą używać WINS – nawet wtedy, gdy Samba świadczy usługi WINS. Jeśli nie możesz skonfigurować wszystkich systemów tak, aby korzystały z tych samych usług, musisz upewnić się, że serwer i klienty dysponują tymi samymi danymi.

W Sambie 2.0 (i późniejszych wydaniach wersji 1.9) dodano opcję `-R` (kolejność odwzorowywania nazw) do programu *smbclient*. Jeśli chciałbyś na przykład zdiagnozować problem z WINS, mógłbyś wydać polecenie:

```
smbclient -L serwer -R wins
```

Opcja ta może przybierać wartości `hosts` (co oznacza wszystkie mechanizmy używane w Uniksie, nie tylko plik */etc/hosts*), `lmhosts`, `wins` i `bcast` (rozgłoszenia).

W poniższych przykładach używamy terminu *długa nazwa* na określenie pełnej nazwy domenowej (*fully-qualified domain name*, FQDN), jak na przykład `serwer.przyklad.com`, a terminu *krótka nazwa* na określenie części hosta w pełnej nazwie, jak na przykład `serwer`.

Nie można wyszukać nazwy hosta

Spróbuj zrobić co następuje:

- DNS:

  Wydaj polecenie `nslookup` *`nazwa`*. Jeśli to się nie powiedzie, przyczyną może być błąd w pliku *resolv.conf*, awaria serwera DNS albo problem z krótkimi i długimi nazwami (patrz następny podrozdział). W takim przypadku powinieneś postępować w opisany niżej sposób:

  - Plik */etc/resolv.conf* powinien zawierać jedną lub kilka linii zaczynających się od słowa `nameserver`, po którym następuje adres IP. Są to adresy twoich serwerów DNS. Sprawdź każdy z tych adresów za pomocą polecenia *ping*. Jeśli polecenie nie zadziała dla jednego z nich, możesz podejrzewać komputer. Jeśli nie zadziała dla żadnego, możesz podejrzewać sieć.
  - Spróbuj ponownie przeprowadzić wyszukiwanie, tym razem używając pełnej nazwy domenowej (na przykład *serwer.przyklad.com*), jeśli za pierwszym razem użyłeś krótkiej, lub posługując się nazwą krótką, jeśli za pierwszym razem użyłeś długiej. Jeśli rezultaty się różnią, przejdź do następnego podrozdziału.

- Rozgłoszenia i WINS:

  Rozgłoszenia i WINS mogą odwzorowywać tylko nazwy krótkie, takie jak `serwer` (a nie długie, jak na przykład `serwer.przyklad.com`). Wydaj polecenie `nmblookup -S` *`serwer`*. Wyświetli ono wszystkie usługi, które zarejestrowano dla danej nazwy. W naszym przykładzie lista wygląda następująco:

```
Looking up status of 192.168.236.86
received 10 names
        SERWER          <00> -         M <ACTIVE>
        SERWER          <03> -         M <ACTIVE>
        SERWER          <1f> -         M <ACTIVE>
        SERWER          <20> -         M <ACTIVE>
        ..__MSBROWSE__..<01> - <GROUP> M <ACTIVE>
        MOJAGRUPA       <00> - <GROUP> M <ACTIVE>
        MOJAGRUPA       <1b> -         M <ACTIVE>
        MOJAGRUPA       <1c> - <GROUP> M <ACTIVE>
        MOJAGRUPA       <1d> -         M <ACTIVE>
        MOJAGRUPA       <1e> - <GROUP> M <ACTIVE>
```

  Wymagany wpis to `SERWER <00>`, który identyfikuje słowo `SERWER` jako NetBIOS-ową nazwę komputera. Powinieneś także zobaczyć kilkakrotnie wymienioną nazwę swojej grupy roboczej. Jeśli nie ujrzysz tych linii, oznacza to, że rozgłoszenia i WINS nie mogą wyszukiwać nazw i wymagają bliższego sprawdzenia.

Liczby w nawiasach ostrokątnych w poprzednim przykładzie identyfikują nazwy NetBIOS-owe jako grupy robocze, stacje robocze, użytkowników usługi posłańca, główne przeglądarki lokalne, przeglądarki domeny, kontrolery domeny i tak dalej. Najczęściej używane kody to: `<00>`, (który identyfikuje komputer), oraz `<20>`, (który oznacza, że komputer jest serwerem). Pełna lista jest dostępna pod adresem *http://support.microsoft.com/support/kb/articles/q163/4/09.asp*.

- NIS:

  Wydaj polecenie `ypmatch` *`nazwa`* `hosts`. Jeśli to się nie powiedzie, NIS nie działa. Odszukaj nazwę serwera NIS za pomocą polecenia *ypwhich* i za pomocą polecenia *ping* sprawdź, czy jest on dostępny.

- NIS+:

  Jeśli korzystasz z NIS+, wydaj polecenie `nismatch` *`nazwa`* `hosts`. Jeśli to się nie powiedzie, NIS nie działa. Odszukaj nazwę serwera NIS za pomocą polecenia *niswhich* i za pomocą polecenia *ping* sprawdź, czy jest on dostępny.

- Pliki *hosts*:

  Przejrzyj plik *hosts* w kliencie (`C:\WINDOWS\HOSTS`). Każda linia powinna zaczynać się od adresu IP, po którym następuje jedna lub więcej nazw – najpierw nazwa podstawowa, a potem opcjonalne aliasy. Oto przykład:

```
127.0.0.1          localhost
192.168.236.1      dns.srw.przyklad.com
192.168.236.10     klient.przyklad.com klient
192.168.236.11     zapas.przyklad.com loghost
192.168.236.86     serwer.przyklad.com serwer
192.168.236.254    ruter.srw.przyklad.com
```

  W Uniksie nazwa `localhost` powinna być zawsze związana z adresem 127.0.0.1, natomiast w komputerze PC może to być po prostu alias nazwy hosta. W kliencie sprawdź, czy na końcu linii nie ma dyrektyw `#XXX`; są to dyrektywy LAN Managera/NetBIOS-u i powinny znajdować się tylko w plikach *LMHOSTS* (`C:\WINDOWS\LMHOSTS`).

- Pliki *LMHOSTS*:

  Ten plik to lokalne źródło nazw LAN Managera (nazw NetBIOS-owych). Formatem przypomina pliki */etc/hosts*, ale nie obsługuje długich nazw domenowych (takich jak `serwer.przyklad.com`), a po nazwach mogą występować opcjonalne dyrektywy `#XXX`. Zauważ, że w katalogu `C:\WINDOWS` znajduje się zwykle plik *lmhosts.sam* (rozszerzenie pochodzi od słowa *sample*, próbka), ale nie jest on używany, dopóki nie zmienisz jego nazwy na `C:\WINDOWS\LMHOSTS`.

Długie i krótkie nazwy hostów

Jeśli działa długa (FQDN) nazwa hosta, a krótka nie (na przykład można wyszukać nazwę `klient.przyklad.com`, ale nie `klient`), przyczyny mogą być następujące:

- DNS:

  Zwykle oznacza to, że nie ma domyślnej domeny, w której można by wyszukać krótką nazwę. Sprawdź, czy w pliku */etc/resolv.conf* w serwerze Samby znajduje się

linia `default` z nazwą twojej domeny lub linia `search` z jedną lub kilkoma nazwami domen. Aby można było korzystać z krótkich nazw, jedna z tych linii musi być obecna; która to będzie, zależy od producenta i wersji programu odwzorowującego. Spróbuj dodać linię `domain twoja_domena` do pliku *resolv.conf* i zapytaj administratora sieci lub serwera DNS, jaka powinna być zawartość tego pliku.

- Rozgłoszenia i WINS:

  Rozgłoszenia i WINS nie obsługują długich nazw, zatem problem ten tutaj nie występuje.

- NIS:

  Spróbuj wydać polecenie `ypmatch nazwahosta hosts`. Jeśli nie znajdziesz pasującego wpisu, oznacza to, że tabele NIS nie zawierają krótkich nazw. Porozmawiaj z administratorem sieci; być może nieobecność krótkich nazw jest przypadkowa, a być może wynika z założeń administracyjnych. W niektórych sieciach w ogóle nie używa się krótkich (wieloznacznych) nazw.

- NIS+:

  Spróbuj wydać polecenie `nismatch nazwahosta hosts`, a w razie niepowodzenia postępuj tak, jak w przypadku NIS.

- Pliki *hosts*:

  Jeśli krótkiej nazwy nie ma w pliku *hosts*, możesz dodać ją jako alias. Jeśli jest to możliwe, nie stosuj nazw krótkich jako podstawowych (pierwszych w linii). Wpisz je jako aliasy, jeśli system na to pozwala.

- Pliki *LMHOSTS*:

  LAN Manager nie obsługuje długich nazw, zatem problem ten tutaj nie występuje.

Jeśli zaś działa krótka forma nazwy, a długa nie, przyczyny mogą być następujące:

- DNS:

  Jest to dość dziwne; skontaktuj się z administratorem sieci lub DNS, gdyż jest to prawdopodobnie usterka w konfiguracji.

- Rozgłoszenia i WINS:

  Jest to normalne. W rozgłoszeniach i WINS nie można używać długiej formy. Rozważ, czy nie zastosować DNS. Microsoft ogłosił, że przejdzie na DNS, choć usługa ta nie dostarcza typów nazw, jak na przykład <00>.

- NIS:

  Jeśli polecenie `ypmatch` wyszukuje krótką formę, ale nie długą, możesz dodać długą formę do tabeli NIS przynajmniej jako alias.

- NIS+:

  Tak samo jak w przypadku NIS, z tym że do wyszukiwania nazw używa się polecenia `nismatch` zamiast `ypmatch`.

- Pliki *hosts*:

  Dodaj długą formę przynajmniej jako alias, a najlepiej jako formę podstawową. Jeśli jest to możliwe do zastosowania w praktyce, rozważ użycie DNS.

- Pliki *LMHOSTS*:

  Jest to normalne. LAN Manager nie może używać długich form; rozważ zastosowanie DNS lub plików *hosts*.

Niezwykłe opóźnienia

Jeśli przed osiągnięciem zamierzonego rezultatu występuje długa zwłoka:

- DNS:

  Przetestuj tę samą nazwę za pomocą polecenia *nslookup* w komputerze (kliencie lub serwerze), który działa wolno. Jeśli polecenie *nslookup* również pracuje wolno, oznacza to problem z DNS. Jeśli działa wolniej w kliencie, oznacza to, że z kartą Ethernetu powiązanych jest zbyt wiele protokołów. Wyeliminuj NetBEUI, który jest znany z powolności, a opcjonalnie protokół Novela, jeśli go nie potrzebujesz. Jest to szczególnie istotne w systemie Windows 95, który jest bardzo wrażliwy na nadmiarowe protokoły.

- Rozgłoszenia i WINS:

  Przetestuj klienta za pomocą polecenia `nmblookup`, a jeśli zadziała ono szybciej, prawdopodobnie problem stanowią nadmiarowe protokoły, jak opisano w poprzednim punkcie.

- NIS:

  Wypróbuj polecenie `ypmatch`, a jeśli działa ono wolno, zgłoś problem administratorowi sieci.

- NIS+:

  Podobnie, wypróbuj polecenie `nismatch`.

- Pliki *hosts*:

  Pliki *hosts*, jeśli mają rozsądną wielkość, zawsze są szybkie. Prawdopodobnie problem związany jest z protokołami, jak opisano powyżej w punkcie dotyczącym DNS.

- Pliki *LMHOSTS*:

  Problem nie jest związany z wyszukiwaniem nazw; pliki *LMHOSTS* są tak samo szybkie, jak pliki *hosts*.

Problemy z lokalnym hostem

Jeśli nazwie `localhost` nie odpowiada adres 127.0.0.1, spróbuj zrobić co następuje:

- DNS:

  Prawdopodobnie brak rekordu dla lokalnego hosta: `localhost. A 127.0.0.1`. Dodaj ten rekord, a także rekord odwrotny: `1.0.0.127.IN-ADDR.ARPA PTR 127.0.0.1`

- Rozgłoszenia i WINS:

  Nie dotyczy.

- NIS:
  Jeśli w tablicy nie ma wpisu dla nazwy `localhost`, dodaj ją.
- NIS+:
  Jeśli w tablicy nie ma wpisu dla nazwy `localhost`, dodaj ją.
- Pliki *hosts*:
  Dodaj do pliku *hosts* linię o postaci: `127.0.0.1 localhost`.
- Pliki *LMHOSTS*:
  Nie dotyczy.

Rozwiązywanie problemów z adresami sieciowymi

Kilka często spotykanych problemów jest spowodowanych błędnym trasowaniem do adresów internetowych lub niepoprawnym przydziałem adresów. W tym podrozdziale powiemy, jak możesz ustalić swoje adresy.

Maski sieciowe

Maski sieciowe informują komputery, które adresy można osiągnąć bezpośrednio (ponieważ znajdują się w sieci lokalnej), a które wymagają przekazania pakietów przez ruter. Jeśli maska sieciowa jest błędna, komputer popełni jedną z dwóch pomyłek. Pierwsza polega na wysyłaniu lokalnych pakietów do rutera, co obniża wydajność – taka konfiguracja może działać całkiem szybko, może działać wolno, może też zawieść całkowicie. Druga polega na niewysyłaniu pakietów zdalnych do rutera, co uniemożliwi przekazanie ich do zdalnego komputera.

Maska sieciowa to liczba przypominająca adres IP, przy czym bity ustawione na 1 wyznaczają w adresie część sieci, a bity ustawione na 0 – część hosta. Maska sieciowa, jak sama nazwa wskazuje, służy do maskowania części adresu w kodzie TCP/IP. Jeśli maska sieciowa ma wartość 255.255.0.0, wówczas dwa pierwsze bajty adresu stanowią część sieci, a dwa pozostałe – część hosta. Częściej używana jest maska 255.255.255.0, w której trzy pierwsze bajty stanowią część sieci, a ostatni – część hosta.

Załóżmy, że twój adres IP to 192.168.0.10, a adres serwera Samby to 192.168.236.86. Jeśli twoja maska sieciowa ma wartość 255.255.255.0, wówczas sieciową częścią adresu są pierwsze trzy bajty, a część hosta zajmuje ostatni bajt. Oznacza to, że części sieciowe się różnią, a komputery mogą znajdować się w różnych sieciach:

Część sieci	*Część hosta*
192 168 000	10
192 168 236	86

Jeśli twoja maska sieciowa ma wartość 255.255.0.0, wówczas sieciową częścią adresu są tylko dwa pierwsze bajty. W takim przypadku części sieciowe są takie same, a komputery znajdują się w tej samej sieci:

Część sieci	*Część hosta*
192 168	000 10
192 168	236 86

Oczywiście, jeśli twoja maska sieciowa mówi jedno, a administrator sieci co innego, oznacza to, że maska jest błędna.

Adresy rozgłoszeniowe

Adres rozgłoszeniowy to zwykły adres, w którym wszystkie bity w części hosta są ustawione na 1. Oznacza on „wszystkie hosty w sieci". Możesz obliczyć go łatwo na podstawie swojej maski sieciowej i adresu: weź adres i umieść w nim bity o wartości 1 we wszystkich pozycjach, w których maska sieciowa ma bity o wartości 0 (czyli w części hosta). Ilustruje to poniższa tabela:

	Część sieci	*Część hosta*
Adres IP	192 168 236	86
Maska sieciowa	255 255 255	000
Adres rozgłoszeniowy	192 168 236	255

W tym przykładzie adresem rozgłoszeniowym dla sieci 192.168.236 jest adres 192.168.236.255. Rutery nie przekazują dalej tak zaadresowanych pakietów, ale większość komputerów w sieci lokalnej odpowie na rozgłoszenie kierowane na ten adres.

Zakresy adresów sieciowych

Pewne zakresy adresów zostały zarezerwowane do celów testowych oraz na użytek sieci nie podłączonych do Internetu; właśnie takich adresów używamy w książce. Jeśli nie masz jeszcze adresu, możesz zacząć od jednego z nich. Obejmują one jedną sieć klasy A (dużą), 10.*.*.*, oraz 254 sieci klasy C (małe), od 192.168.1.* do 192.168.254.*. W niniejszej książce wybraliśmy jedną z sieci klasy C, 192.168.236.*.

Jeśli rzeczywiście łączysz się z Internetem, powinieneś uzyskać prawdziwy adres sieci oraz nazwę domenową, zwykle za pośrednictwem tej samej firmy, która udostępnia ci połączenie.

Ustalanie adresu sieciowego

Jeśli nie zapisałeś adresu swojego komputera, możesz go wyświetlić za pomocą polecenia *ifconfig* w Uniksie, IPCONFIG w Windows NT oraz WINIPCFG w Windows 95 (sprawdź na stronach podręcznika man, jakich opcji wymaga twoja odmiana Uniksa; w systemie Sun trzeba użyć polecenia `ifconfig -a`). Powinieneś zobaczyć wyniki podobne do poniższych:

```
serwer% ifconfig -a
le0: flags=63<UP,BROADCAST,NOTRAILERS,RUNNING >
      inet 192.168.236.11 netmask ffffff00 broadcast 192.168.236.255
lo0: flags=49<UP,LOOPBACK,RUNNING >
      inet 127.0.0.1 netmask ff000000
```

Jednym z interfejsów będzie pętla zwrotna (w naszym przykładzie `lo0`), a drugim – zwykły interfejs IP. Znaczniki powinny wskazywać, że interfejs działa (RUNNING), a przy interfejsach Ethernetu pojawi się także informacja o obsłudze rozgłoszeń (interfejsy PPP ich nie obsługują). Inne miejsca, w których można poszukać adresów IP, to pliki */etc/hosts*, pliki *HOSTS* w Windows, pliki *LMHOSTS* w Windows, NIS, NIS+ i DNS.

Rozwiązywanie problemów z nazwami NetBIOS-owymi

Protokoły SMB były od początku zależne od systemu nazw NetBIOS-u, nazywanego także systemem nazw LAN Managera. Był to prosty schemat nazewniczy, w którym każdy komputer miał niepowtarzalną nazwę długości 20 znaków i rozgłaszał ją w sieci lokalnej do wiadomości wszystkich innych komputerów. W TCP/IP używamy raczej nazw typu *klient.przyklad.com*, przechowywanych w plikach */etc/hosts* lub udostępnianych przez DNS albo WINS.

Najczęściej spotykane odwzorowywanie nazw domenowych, takich jak *serwer. przyklad.com*, polega po prostu na potraktowaniu części „*serwer*" jako nazwy NetBIOS-owej po uprzednim przekształceniu jej na duże litery. Niestety, podejście takie nie zawsze się sprawdza, zwłaszcza wtedy, gdy nazwa komputera ma więcej niż 20 znaków. Nie każdy komputer używa takiej samej nazwy NetBIOS-owej i DNS; dość często równolegle z nazwą typu *firmavm1* używa się nazwy typu *vm1.firma.com*.

Komputer, którego nazwa DNS różni się od NetBIOS-owej, sprawia kłopoty podczas diagnozowania problemów; jeśli to możliwe, radzimy unikać takiej konfiguracji. Nazwy NetBIOS-owe można ustalić za pomocą programu *smbclient*:

- Jeśli możesz wylistować udziały serwera Samby za pomocą polecenia *smbclient* z opcją `-L krótka_nazwa_serwera`, krótka nazwa jest nazwą NetBIOS-ową.
- Jeśli otrzymasz komunikat „Get_Hostbyname: Unknown host *nazwa*", prawdopodobnie zachodzi niezgodność nazw. Sprawdź, czy nazwa NetBIOS-owa nie jest określona jawnie w pliku *smb.conf*.
- Spróbuj ponownie, tym razem z opcją `-I` i adresem IP serwera Samby (na przykład `smbclient -L serwer -I 192.168.236.86`). W ten sposób pominiesz wyszukiwanie nazwy i wymusisz przesłanie pakietów pod podanym adresem IP. Jeśli to zadziała, zachodzi niezgodność nazw.
- Spróbuj ponownie, tym razem z opcją `-I` i pełną nazwą domenową serwera (na przykład `smbclient -L serwer -I serwer.przyklad.com`). W ten sposób przetestujesz mechanizm wyszukiwania nazw domenowych używany przez serwer Samby (na przykład DNS). Jeśli to się nie powiedzie, masz problem z usługami nazewniczymi. Kiedy uporasz się z nazwami NetBIOS-owymi, powinieneś jeszcze raz przeczytać podrozdział „Rozwiązywanie problemów z usługami nazewniczymi".
- Spróbuj ponownie, tym razem z opcją `-n` (nazwa NetBIOS-owa) i nazwą, która powinna zadziałać (na przykład `smbclient -n serwer -L serwer-12`), ale nie wymuszając adresu opcją `-I`. Jeśli to się powiedzie, nazwa podana po opcji `-n` jest rzeczywistą NetBIOS-ową nazwą serwera. Jeśli otrzymasz komunikat „Get_Hostbyname: Unknown host *nazwa*", to wciąż nie będzie ta nazwa.

- Jeśli do tej pory nic nie zadziałało, powtórz testy, używając opcji `-U nazwa_u□ytkownika` oraz `-W grupa_robocza`, aby upewnić się, że na zawadzie nie stoi niezgodność nazw użytkownika lub grup roboczych.
- Jeśli nadal nic nie działa, a masz dowody na błędne działanie usług nazewniczych, zajrzyj do podrozdziału „Rozwiązywanie problemów z usługami nazewniczymi" i skoryguj ewentualne błędy, a następnie wróć do diagnozowania usług nazewniczych NetBIOS-u.

Dodatkowe zasoby

W którymś punkcie swojej kariery administratora Samby zapewne zechcesz zajrzeć do źródeł internetowych i drukowanych, aby zapoznać się z nowinami, uaktualnieniami i wskazówkami.

Dokumentacja

Nie ma nic złego w czytaniu dokumentacji. Naprawdę. Nikt cię na tym nie nakryje, a i my nikomu nie powiemy. Samba jest dostarczana wraz z obszernym zbiorem plików dokumentacji i warto choćby pobieżnie je przejrzeć. Znajdziesz je w katalogu */docs* dystrybucji Samby w swoim komputerze, albo w witrynie Samby pod adresem *http://samba.anu.edu.au/samba/*. W witrynie tej znajdziesz najbardziej aktualną listę dokumentów FAQ, informacje o usterkach w programie oraz łącza do centrów dystrybucji, stron podręcznika man Samby i dokumentów HOW-TO.

Grupy dyskusyjne Samby

W grupach Usenetu zawsze można było zasięgnąć porady na niemal każdy temat. W ostatnich kilku latach ten niezmierzony ocean wiedzy rozwinął coś, co uczyniło z niego wprost nieoceniony zasób: pamięć. Dzięki witrynom archiwizującym i wyszukiwawczym, takim jak DejaNews (*http://www.dejanews.com*) -kilkoma kliknięciami możesz uzyskać dostęp do gromadzonych przez lata rozwiązań najrozmaitszych problemów.

Podstawowa grupa dyskusyjna Samby to *comp.protocols.smb*. Właśnie tu powinieneś zajrzeć najpierw, jeśli masz jakiś problem. Zwykle kilka minut spędzonych tutaj zaoszczędzi ci długich godzin samodzielnego diagnozowania problemu.

Kiedy przeszukujesz archiwum grupy dyskusyjnej, zadawaj krótkie, precyzyjne pytania. Najlepiej wyszukiwać rzeczywiste komunikaty o błędach. Jeśli nie znajdziesz odpowiedzi natychmiast, odpędź pokusę wysłania prośby o pomoc, dopóki nie spróbujesz sam zgłębić tematu. Być może odkryjesz, że odpowiedź znajduje się w dokumencie FAQ lub jednym z wielu innych plików dokumentacji dostarczanych wraz z Sambą, albo rozwiązanie stanie się oczywiste, kiedy skorzystasz z narzędzi diagnostycznych Samby. Jeśli nic nie wskórasz, wyślij prośbę do grupy *comp.protocols.smb* i opisz możliwie dokładnie swoje próby i ich rezultaty. Dołącz wszystkie wyświetlone komunikaty o błędach. Być może minie kilka dni, zanim uzyskasz pomoc, więc bądź cierpliwy, a czekając na odpowiedź, nie rezygnuj z samodzielnego rozwiązania problemu.

Gdy wyślesz prośbę o pomoc, spróbuj podejść do problemu z innej strony. Większości z nas zdarzyło się wysłać do grupy dyskusyjnej artykuł z setkami zawiłych detali i uporać się z problemem bez żadnej pomocy w godzinę później, gdy artykuł zdążył już przewędrować wszytkie kontynenty. Oto ogólna reguła: im więcej ludzi przeczyta twoją prośbę, tym prostsze okaże się rozwiązanie. Zwykle oznacza to, że kiedy już cała uniksowa społeczność zapozna się z twoim artykułem, odpowiedź będzie brzmiała: „Podłącz komputer do gniazdka w ścianie".

Listy wysyłkowe Samby

Poniżej podajemy adresy list wysyłkowych związanych z Sambą. Informacje o subskrybowaniu tych grup i rezygnowaniu z subskrypcji znajdziesz na stronie głównej Samby pod adresem *http://www.samba.org/*.

samba-binaries@samba.org
: Ta lista wysyłkowa zawiera informacje o prekompilowanych plikach binarnych Samby dla różnych platform.

samba-bugs@samba.org
: Na tej liście wysyłkowej należy zgłaszać zauważone usterki w działaniu Samby.

samba-ntdom@samba.org
: Na tej liście wysyłkowej znajdziesz informacje o obsłudze domen (zwłaszcza Windows NT) w Sambie.

samba-technical@samba.org
: Na tej liście wysyłkowej toczy się dyskusja o przyszłości Samby.

samba@samba.org
: Jest to podstawowa lista wysyłkowa, która zawiera ogólne pytania i informacje HOW-TO („jak to zrobić") dotyczące Samby.

Archiwa list wysyłkowych Samby

Istnieje usługa wyszukiwawcza dla list wysyłkowych Samby. Kiedy pisaliśmy tę książkę, można do niej dotrzeć przez łącze archives na głównej stronie Samby i w witrynach bliźniaczych lub bezpośrednio przez adres *http://us1.samba.org/search/smb-mail.shtml*.

Dalsza lektura

Craig Hunt, *TCP/IP – administracja sieci, wyd. 2.*, Wydawnictwo RM, Warszawa 1998 r.

Craig Hunt i Robert Bruce Thompson, *Windows NT TCP/IP Network Administration*. Sebastopol, CA: O'Reilly and Associates, 1998 (ISBN 1-56592-377-4).

Paul Albitz i Cricket Liu, *DNS i Bind*, Wydawnictwo RM, Warszawa 1999 r.

Hal Stern, *Managing NFS and NIS*. Sebastopol, CA: O'Reilly and Associates, 1997 (ISBN 0-937175-75-7).

A

Konfigurowanie Samby do obsługi SSL

W tym dodatku opisujemy konfigurowanie Samby do obsługi bezpiecznych połączeń między serwerem i klientami. Użyjemy w tym celu protokołu Secure Sockets Layer (SSL) Netscape'a. Spróbujemy utworzyć bezpieczne połączenie między serwerem Samby a stacją roboczą Windows NT.

Zanim przystąpisz do lektury, powinieneś zapoznać się z podstawowymi informacjami na temat kryptografii z kluczem publicznym oraz certyfikatów X.509. Jeśli zagadnienia te nie są ci znajome, gorąco polecamy książkę Bruce'a Schneiera *Applied Cryptography, 2nd Edition* (wydawnictwo Wiley), która jest znakomitym źródłem wiedzy o wielu mało znanych aspektach kryptografii.

Jeśli chciałbyś dowiedzieć się więcej o SSL w Sambie, przeczytaj dokument *SSLeay.txt* w katalogu */docs/textdocs* dystrybucji Samby, na podstawie którego napisano ten rozdział.

Certyfikaty

Oto kilka pytań i odpowiedzi wziętych z pliku *SSLeay.txt*, wchodzącego w skład dystrybucji Samby, i dotyczących korzyści płynących ze stosowania SSL i certyfikatów. Tekst ten został napisany przez Christiana Starkjohanna w ramach projektu Samby.

Co to jest certyfikat?

Certyfikat jest wystawiany przez wydawcę, zwykle przez serwis certyfikacyjny (*Certification Authority*, CA), który potwierdza coś przez wystawienie certyfikatu. Podmiot tego potwierdzenia jest zależny od polityki wydawcy. Serwisy certyfikacyjne dla bezpiecznych serwerów WWW (używanych na przykład przez sklepy internetowe) zwykle po prostu zaświadczają, że dany klucz publiczny należy do danej nazwy domenowej. Firmowe serwisy CA mogą zaświadczać, że jesteś pracownikiem firmy, że masz prawo do korzystania z serwera i tak dalej.

Co to jest certyfikat X.509?

Technicznie rzecz biorąc, certyfikat to blok danych podpisany przez wydawcę certyfikatu (CA). Składa się z następujących pól:

- niepowtarzalny identyfikator (nazwa) wydawcy certyfikatu,
- czas, przez który certyfikat zachowuje ważność,
- niepowtarzalny identyfikator (nazwa) potwierdzanego obiektu,
- klucz publiczny potwierdzanego obiektu,
- podpis wydawcy nałożony na wszystkie te dane.

Jeśli certyfikat ma zostać zweryfikowany, osoba weryfikująca musi dysponować tabelą z nazwami i kluczami publicznymi zaufanych wydawców. Dla ułatwienia tabele te powinny zawierać certyfikaty wydane przez serwisy CA dla samych siebie (certyfikaty podpisywane samodzielnie).

Jakie są implikacje takiej struktury certyfikatu?

Oto cztery zasadnicze implikacje:

- Ponieważ certyfikat zawiera klucz publiczny podmiotu, do szyfrowania i deszyfrowania danych wystarczy certyfikat i klucz prywatny.
- Aby zweryfikować certyfikat, potrzebne są certyfikaty wszystkich zaufanych serwisów CA.
- Najprostszą formą certyfikatu-atrapy jest taki, który został podpisany przez podmiot.
- Konieczne jest istnienie CA. Klient nie może po prostu wystawiać lokalnych certyfikatów dla serwerów, którym ufa, ponieważ to serwer wybiera prezentowany przez siebie certyfikat.

Wymagania

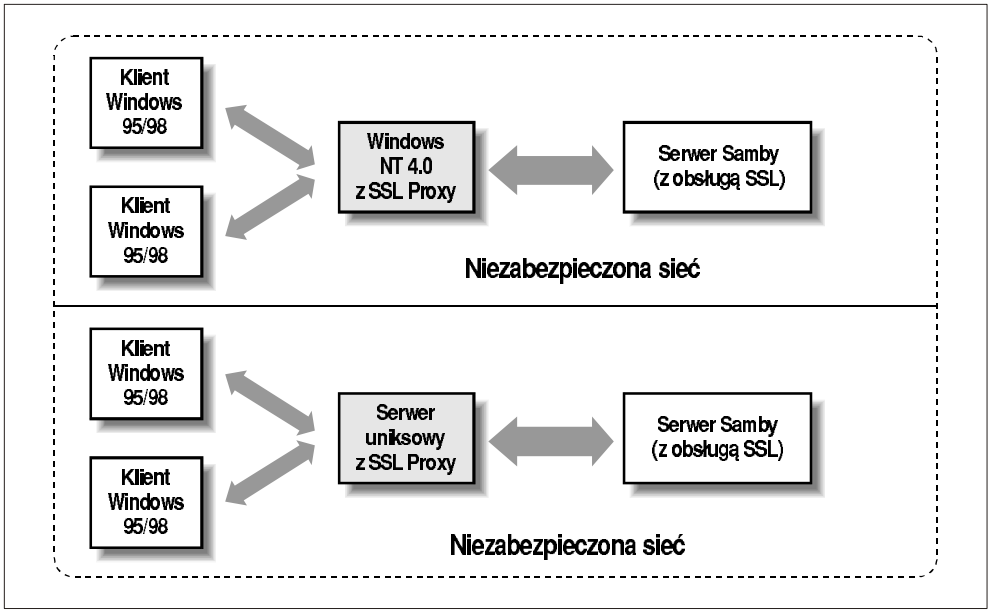
Aby korzystać z połączeń SSL, będziesz musiał pobrać dwa programy oprócz Samby.

SSLeay
: Implementacja protokołu Secure Sockets Layer (SSL) napisana przez Erica Younga w postaci kilku bibliotek uniksowych.

SSL Proxy
: Bezpłatna aplikacja SSL napisana przez grupę programistów Objective Development, która działa w Uniksie oraz Windows NT i może pośredniczyć w nawiązywaniu bezpiecznych połączeń.

Te dwa produkty działają po stronie serwera i klienta w zaszyfrowanym połączeniu SSL. Biblioteki SSLeay kompiluje się i instaluje bezpośrednio w systemie uniksowym, natomiast program SSL Proxy można pobrać w postaci źródłowej lub binarnej i zainstalować po stronie klienta. Jeśli po drugiej stronie połączenia SSL ma znajdować się klient Windows NT lub Samba, nie będzie potrzebna specjalna konfiguracja.

Program SSL Proxy nie działa jednak w komputerach Windows 95/98. Jeśli chcesz zapewnić bezpieczeństwo połączeń między serwerem Samby a klientami Windows 95/98, musisz umieścić serwer uniksowy lub komputer Windows NT w tej samej podsieci co klienty Windows 9*x* i nawiązywać wszystkie połączenia sieciowe za pośrednictwem komputera z zainstalowanym programem SSL Proxy (patrz rysunek A.1).



Rysunek A.1. Dwa możliwe sposoby pośredniczenia w połączeniach nawiązywanych przez klienty Windows 95/98

W tym rozdziale spróbujemy utworzyć proste połączenie SSL między serwerem Samby a klientem Windows NT. Można wykorzystać tę konfigurację do utworzenia bardziej skomplikowanych sieci, według uznania administratora.

Instalowanie pakietu SSLeay

Samba korzysta z pakietu SSLeay napisanego przez (Erica Younga), który zapewnia obsługę SSL po stronie serwera. Ze względu na ograniczenia eksportowe obowiązujące w Stanach Zjednoczonych pakiet SSLeay nie może być dołączany do dystrybucji Samby rozpowszechnianych z USA. Z tej przyczyny twórcy Samby zdecydowali, że SSLeay pozostanie odrębnym pakietem. Możesz pobrać SSLeay z wymienionych poniżej witryn:

- *ftp://ftp.psy.uq.oz.au/pub/Crypto/SSL*
- *ftp://ftp.uni-mainz.de/pub/internet/security/ssl*
- *ftp://ftp.cert.dfn.de/pub/tools/crypt/sslapps*
- *ftp://ftp.funet.fi/pub/crypt/mirrors/ftp.psy.uq.oz.au*
- *ftp://ftp.sunet.se/ftp/pub/security/tools/crypt/ssleay*

Kiedy pisaliśmy tę książkę, najnowsza wersja pakietu miała numer 0.9.0b. Pobierz go do tego serwera, w którym znajduje się dystrybucja Samby, zdekompresuj i rozpakuj poleceniem *tar*. Powinieneś otrzymać katalog o nazwie *SSLeay-0.9.0b*. Przejdź do tego katalogu i skonfiguruj oraz skompiluj pakiet szyfrowania SSL tak samo, jak zrobiłeś to z Sambą.

SSLeay wykorzystuje skrypt *configure* napisany w Perlu. Skrypt ten modyfikuje plik *makefile*, który tworzy narzędzia i biblioteki pakietu SSLeay. Domyślnie skrypt zakłada, że interpreter Perla to */usr/local/bin/perl*. Być może będziesz musiał zmodyfikować go tak, aby wskazywał położenie pliku wykonywalnego Perla w twoim systemie. Możesz ustalić położenie interpretera Perla za pomocą następującego polecenia:

```
# which perl
/usr/bin/perl
```

Następnie zmodyfikuj pierwszą linię skryptu *configure* tak, aby korzystał z właściwego pliku wykonywalnego Perla. W naszym systemie Red Hat zrobiliśmy to w ten sposób:

```
#!/usr/bin/perl
#
# see PROBLEMS for instructions on what sort of things to do
# when tracking a bug -tjh
...
```

Następnie musisz uruchomić skrypt *configure*, określając docelową platformę dla pakietu SSLeay. Oto dostępne platformy:

```
BC-16               BC-32               FreeBSD             NetBSD-m86
NetBSD-sparc        NetBSD-x86          SINIX-N             VC-MSDOS
VC-NT               VC-W31-16           VC-W31-32           VC-WIN16
VC-WIN32            aix-cc              aix-gcc             alpha-cc
alpha-gcc           alpha400-cc         cc                  cray-t90-cc
debug               debug-irix-cc       debug-linux-elf     dgux-R3-gcc
dgux-R4-gcc         dgux-R4-x86-gcc     dist                gcc
hpux-cc             hpux-gcc            hpux-kr-cc          irix-cc
irix-gcc            linux-aout          linux-elf           ncr-scde
nextstep            purify              sco5-cc             solaris-sparc-cc
solaris-sparc-gcc   solaris-sparc-sc4   solaris-usparc-sc4  solaris-x86-gcc
sunos-cc            sunos-gcc           unixware-2.0        unixware
```

W naszym systemie wpisalibyśmy następujące polecenie:

```
# ./Configure linux-elf
CC            =gcc
CFLAG         =-DL_ENDIAN -DTERMIO -DBN_ASM -O3 -fomit-frame-pointer
EX_LIBS       =
BN_MULW       =asm/bn86-elf.o
DES_ENC       =asm/dx86-elf.o asm/yx86-elf.o
BF_ENC        =asm/bx86-elf.o
CAST_ENC      =asm/cx86-elf.o
RC4_ENC       =asm/rx86-elf.o
RC5_ENC       =asm/r586-elf.o
MD5_OBJ_ASM   =asm/mx86-elf.o
SHA1_OBJ_ASM  =asm/sx86-elf.o
RMD160_OBJ_ASM=asm/rm86-elf.o
THIRTY_TWO_BIT mode
```

```
DES_PTR used
DES_RISC1 used
DES_UNROLL used
BN_LLONG mode
RC4_INDEX mode
```

Kiedy pakiet zostanie skonfigurowany, możesz skompilować go poleceniem `make`. Jeśli kompilacja nie zakończy się sukcesem, zajrzyj do dokumentacji dostarczanej wraz z dystrybucją lub do dokumentu FAQ pod adresem *http://www.cryptsoft.com/ssleay*, aby znaleźć więcej informacji o możliwych przyczynach niepowodzenia. Jeśli kompilacja powiedzie się, wpisz `make install`, aby zainstalować biblioteki w systemie. Domyślnie plik *makefile* instaluje pakiet w katalogu */usr/local/ssl*. Jeśli zainstalujesz pakiet w innym katalogu, zapamiętaj jego położenie, ponieważ będziesz musiał je znać podczas konfigurowania Samby.

Konfigurowanie pakietu SSLeay

Najpierw musisz ustawić zmienną środowiskową `PATH` w twoim systemie tak, aby zawierała katalog */bin* dystrybucji SSL. Można to zrobić za pomocą polecenia:

```
PATH=$PATH:/usr/local/ssl/bin
```

To łatwiejsza część zadania. Teraz będziesz musiał utworzyć losową serię znaków, które posłużą do zainicjowania generatora liczb losowych w SSLeay. Generator ten będzie używany do tworzenia par kluczy dla klientów i serwera. Możesz utworzyć losową serię, wypełniając plik tekstowy długim ciągiem dowolnych znaków. W tym celu możesz skorzystać z edytora tekstów albo wydać poniższe polecenie i wpisać losowe znaki na standardowym wejściu:

```
# cat >/tmp/prywatne.txt
```

Dokumentacja Samby zaleca, aby pisać dłużej niż minutę przed zamknięciem strumienia wejściowego sekwencją [Ctrl+D]. Nie wpisuj samych znaków alfabetycznych, dorzuć także trochę symboli i cyfr. Kiedy utworzysz losowy plik, możesz zainicjować generator liczb losowych za pomocą polecenia:

```
# ssleay genrsa -rand /tmp/prywatne.txt >/dev/null
2451 semi-random bytes loaded
Generating RSA private key. 512 bit long modulus
..+++++
.................................+++++
e is 65537 (0x10001)
```

Możesz zignorować wyniki tego polecenia. Kiedy zakończy ono pracę, usuń ciąg znaków użyty do utworzenia klucza, ponieważ mógłby on zostać wykorzystany do odtworzenia kluczy prywatnych utworzonych za pomocą generatora liczb losowych:

```
# rm -f /tmp/prywatne.txt
```

W wyniku działania tego polecenia otrzymasz ukryty plik *.rnd*, który jest zapisany w twoim katalogu macierzystym. SSLeay będzie korzystał z tego pliku podczas tworzenia par kluczy.

Konfigurowanie Samby do korzystania z SSL

W tym momencie możesz skompilować Sambę z dołączoną obsługą SSL. Jak pamiętasz z rozdziału 2, *Instalowanie Samby w Uniksie*, przed kompilacją Samby musisz uruchomić skrypt *configure*, który inicjuje plik *makefile*. Aby Samba mogła korzystać z SSL, musisz zrekonfigurować plik *makefile*:

```
# ./configure --with-ssl
```

Następnie możesz skompilować Sambę za pomocą następujących poleceń:

```
# make clean
# make all
```

Jeśli otrzymasz komunikat o błędzie, który poinformuje cię, że nie można utworzyć pliku wykonywalnego *smbd* ze względu na brak pliku *ssl.h*, prawdopodobnie oznacza to, że nie zainstalowałeś SSLeay w domyślnym katalogu. Użyj opcji konfiguracyjnej `--with-sslinc`, aby wskazać podstawowy katalog dystrybucji – w tym przypadku katalog, który zawiera plik *include/ssl.h*.

Jeśli zaś kompilacja przebiegnie prawidłowo, możesz przejść do następnego etapu: tworzenia certyfikatów.

Własny serwis certyfikacyjny

Protokół SSL wymaga użycia certyfikatów X.509 w fazie negocjacji połączenia. Pozwalają one stwierdzić, że jedna bądź obie strony zaangażowane w komunikację rzeczywiście są tym, za kogo się podają. Rzeczywiste certyfikaty, na przykład te używane w połączeniach SSL przez publiczne serwery WWW, kosztują około 300 dolarów rocznie. Dzieje się tak dlatego, że certyfikat musi być oznaczony cyfrowym podpisem *serwisu certyfikacyjnego*. Serwis certyfikacyjny to jednostka, która gwarantuje autentyczność cyfrowego certyfikatu, podpisując go własnym kluczem prywatnym. W ten sposób każdy, kto chce potwierdzić autentyczność certyfikatu, może po prostu użyć publicznego klucza serwisu certyfikacyjnego, aby sprawdzić podpis.

Pakiet SSLeay pozwala na wykorzystanie publicznych serwisów certyfikacyjnych. Nie musisz jednak tego robić. Zamiast tego możesz zadeklarować własny serwer jako serwis certyfikacyjny – określić, którym klientom ufasz, a którym nie. W tym celu będziesz musiał wykonać kilka czynności, aby odpowiednio skonfigurować pakiet SSLeay.

Najpierw musisz określić bezpieczną lokację, w której będą przechowywane certyfikaty klientów i ewentualnie serwera. My wybraliśmy domyślny katalog */etc/certificates*. Wykonaj poniższe polecenia jako `root`:

```
# cd /etc
# mkdir certificates
# chmod 700 certificates
```

Zauważ, że wszyscy użytkownicy z wyjątkiem roota nie mają żadnych praw dostępu do tego katalogu. Jest to bardzo istotne.

Teraz musisz skonfigurować skrypty i pliki konfiguracyjne SSLeay tak, aby korzystały z plików przechowywanych w tym katalogu. W tym celu zmodyfikuj skrypt

CA.sh (*/usr/local/ssl/bin/CA.sh*) tak, aby określał położenie właśnie utworzonego katalogu. Znajdź linię, która zawiera następujący wpis:

```
CATOP=./demoCA
```

Następnie zmień ją na:

```
CATOP=/etc/certificates
```

Teraz będziesz musiał zmodyfikować plik */usr/local/ssl/lib/ssleay.cnf* tak, aby wskazywał ten sam katalog. Znajdź wpis:

```
[ CA_default ]
dir     = ./demoCA              # Where everything is kept
```

Następnie zmień go na:

```
[ CA_default ]
dir     = /etc/certificates     # Where everything is kept
```

Następnie uruchom skrypt inicjacyjny serwisu certyfikacyjnego, *CA.sh*, aby utworzyć certyfikaty. Zrób to z tego samego konta użytkownika, z którego inicjowałeś generator liczb losowych:

```
# /usr/local/ssl/bin/CA.sh -newca
mkdir: cannot make directory '/etc/certificates': File exists
CA certificate filename (or enter to create)
```

Naciśnij klawisz [Enter], aby utworzyć certyfikat dla serwisu CA. Powinieneś zobaczyć następujące komunikaty:

```
Making CA certificate ...
Using configuration from /usr/local/ssl/lib/ssleay.cnf
Generating a 1024 bit RSA private key
..............................+++++
.........................+++++
writing new private key to /etc/certificates/private/cakey.pem
Enter PEM pass phrase:
```

Wpisz nowe hasło dla certyfikatu. Będziesz musiał zrobić to dwukrotnie, zanim SSLeay je zaakceptuje:

```
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
```

Zapamiętaj wpisane hasło. Będzie ono potrzebne w przyszłości do podpisywania certyfikatów klientów. Kiedy SSLeay zaakceptuje hasło, będziesz musiał odpowiedzieć na kilka pytań dotyczących różnych pól certyfikatu X.509:

```
You are about to be asked to enter information that will be
incorporated into your certificate request.
What you are about to enter is what is called a Distinguished
Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```

Wypełnij resztę pól informacjami o swojej organizacji. Nasz przykładowy certyfikat wygląda następująco:

```
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:California
Locality Name (eg, city) []:Sebastopol
Organization Name (eg, company) []:O'Reilly
Organizational Unit Name (eg, section) []:Books
Common Name (eg, YOUR name) []:John Doe
Email Address []:doe@ora.com
```

Teraz pakiet SSLeay jest skonfigurowany jako serwis certyfikacyjny i będzie mógł podpisywać certyfikaty dla klientów łączących się z serwerem Samby.

Tworzenie certyfikatów dla klientów

Tworzenie certyfikatu dla klienta jest bardzo proste. Najpierw musisz wygenerować parę kluczy publiczny/prywatny dla każdej jednostki, utworzyć plik wniosku o certyfikat, a następnie podpisać ten plik jako zaufany serwis certyfikacyjny.

Dla naszego przykładowego klienta o nazwie `feniks` sprowadza się to do wydania trzech poleceń SSLeay. Pierwsze generuje parę kluczy dla klienta i umieszcza je w pliku *feniks.key*. Prywatny klucz będzie zaszyfrowany, w tym przypadku potrójnym algorytmem DES. Wpisz hasło, kiedy zostaniesz o to poproszony – będzie ci ono potrzebne w następnym etapie:

```
# ssleay genrsa -des3 1024 >feniks.key
1112 semi-random bytes loaded
Generating RSA private key, 1024 bit long modulus
.....................................+++++
.............+++++
e is 65537 (0x10001)
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
```

Kiedy to polecenie zakończy działanie, wpisz poniższe polecenie:

```
# ssleay req -new -key feniks.key -out feniks-csr
Enter PEM pass phrase:
```

Wpisz hasło dla właśnie utworzonego certyfikatu klienta (nie dla serwisu certyfikacyjnego). W tym momencie będziesz musiał jeszcze raz wypełnić kwestionariusz, tym razem dla klienta. Dodatkowo będziesz musiał wpisać hasło wyzwania oraz opcjonalną nazwę firmy – te dane nie mają dla nas znaczenia. Kiedy polecenie zakończy działanie, otrzymasz wniosek o wystawienie certyfikatu w pliku *feniks-csr*.

Następnie musisz podpisać wniosek jako zaufany serwis certyfikacyjny. Wpisz poniższe polecenie:

```
# ssleay ca -days 1000 -inflies feniks-csr >feniks.pem
```

Polecenie to poprosi cię o wprowadzenie hasła PEM dla *serwisu certyfikacyjnego*. Nie wpisuj tu hasła dla właśnie utworzonego certyfikatu klienta. Kiedy wpiszesz poprawne hasło, powinieneś zobaczyć następujące wyniki:

```
Check that the request matches the signature
Signature ok
The Subjects Distinguished Name is as follows:
...
```

Po tym nastąpią dane wprowadzone przed chwilą podczas tworzenia certyfikatu klienta. Jeśli któreś z pól jest błędne, program powiadomi o tym. Jeśli zaś wszystko przebiegnie bezbłędnie, SSLeay potwierdzi, że certyfikat został podpisany i wprowadzony do bazy danych. Rekord certyfikatu zostanie dodany do katalogu */etc/certificates/newcerts*.

Pod koniec tego ćwiczenia powinieneś otrzymać pliki *feniks.key* i *feniks.pem*, zapisane w bieżącym katalogu. Pliki te zostaną przesłane do klienta, z którym będzie współpracował serwer Samby, i będą używane przez program SSL Proxy.

Konfigurowanie serwera Samby

Następnym etapem będzie dopisanie poniższych opcji do pliku konfiguracyjnego Samby. Zakładamy, że utworzyłeś katalog certyfikatów dla serwisu certyfikacyjnego, */etc/certificates*:

```
[global]
    ssl = yes
    ssl server cert = /etc/certificates/cacert.pem
    ssl server key = /etc/certificates/private/cakey.pem
    ssl CA certDir = /etc/certificates
```

W tym momencie będziesz musiał przerwać działanie demonów Samby i uruchomić je ponownie:

```
# nmbd -D
# smbd -D
Enter PEM pass phrase:
```

Będziesz musiał wpisać hasło PEM serwisu certyfikacyjnego, aby uruchomić demony Samby. Zauważ, że może to stanowić pewien problem w normalnym uruchamianiu demonów. Można jednak się z tym uporać za pomocą zaawansowanych języków skryptowych, takich jak Expect lub Python.

Testowanie konfiguracji za pomocą programu smbclient

Dobrym sposobem na sprawdzenie, czy Samba działa prawidłowo, jest użycie programu *smbclient*. W serwerze Samby wprowadź poniższe polecenie, podstawiając odpowiednią nazwę udziału i użytkownika:

```
# smbclient //hydra/dane -U tomek
```

Powinieneś zobaczyć kilka komunikatów diagnostycznych, po których wyświetlona zostanie linia z wynegocjowanym szyfrem, na przykład:

```
SSL: negotiated cipher: DES-CBC3-SHA
```

Następnie możesz wprowadzić swoje hasło i normalnie połączyć się z udziałem. Jeśli to zadziała, możesz być pewien, że Samba poprawnie obsługuje połączenia SSL. Teraz pora na skonfigurowanie klienta.

Konfigurowanie SSL Proxy

Program SSL Proxy jest dostępny w postaci binarnej i źródłowej. Możesz pobrać go pod adresem *http://obdev.at/Products/sslproxy.html*.

Po pobraniu kodu źródłowego możesz skonfigurować go i skompilować tak jak Sambę. W poniższym przykładzie skonfigurujemy go w Windows NT, jednakże instalowanie SSL Proxy w Uniksie przebiega niemal identycznie. Aby wykonać opisane niżej czynności, musisz być użytkownikiem uprzywilejowanym (administratorem).

Jeśli pobrałeś wersję binarną dla Windows NT, w katalogu powinny znajdować się następujące pliki:

- *cygwinb19.dll*,
- README.TXT,
- *sslproxy.exe*,
- *dummyCert.pem*.

Spośród nich będzie nam potrzebny wyłącznie plik wykonywalny programu SSL Proxy. Skopiuj utworzone wcześniej pliki *feniks.pem* i *feniks.key* do katalogu, w którym znajduje się plik wykonywalny SSL Proxy. Upewnij się, że katalog jest zabezpieczony przed ciekawskimi użytkownikami.

Następnie musisz sprawdzić, czy komputer Windows NT może odwzorować NetBIOS-ową nazwę serwera Samby. Oznacza to, że w sieci powinien znajdować się działający serwer WINS (Samba może spełniać tę funkcję z opcją `wins support = yes`), albo serwer Samby powinien być wymieniony w systemowym pliku hostów. Więcej informacji o serwerze WINS znajdziesz w rozdziale 7, *Drukowanie i odwzorowywanie nazw**.

Teraz uruchom SSL Proxy za pomocą poniższego polecenia. Zakładamy tu, że serwer Samby nosi nazwę `hydra`:

```
# C:\SSLProxy>sslproxy -l 139 -R hydra -r 139 -n -c feniks.pem -k feniks.key
```

Polecenie to informuje program SSL Proxy, że należy oczekiwać na połączenia z portem 139 i kierować je do portu 139 w komputerze o NetBIOS-owej nazwie `hydra` oraz że należy użyć plików *feniks.pem* i *feniks.key* do wygenerowania certyfikatów i kluczy potrzebnych do zainicjowania połączenia SSL. SSL Proxy odpowie komunikatem:

```
Enter PEM pass phrase:
```

Wprowadź hasło PEM dla pary kluczy klienta, nie dla serwisu certyfikacyjnego. Powinieneś zobaczyć następujące komunikaty:

```
SSL: No verify locations, trying default
proxy ready, listening for connections
```

---

* Jeśli program SSL Proxy działa w serwerze uniksowym, powinieneś upewnić się, że możliwe jest odwzorowanie nazwy domenowej serwera Samby.

To powinno załatwić sprawę w kliencie. Możesz umieścić powyższe polecenie w sekwencji startowej Uniksa lub Windows NT, jeśli chcesz zapewnić ciągłość usług SSL. Upewnij się, że wszystkie klienty łączące się z serwerem NT (włączając w to sam serwer NT) wskazują na ten serwer, a nie na serwer Samby.

Kiedy wykonasz powyższe czynności, spróbuj połączyć klienta i serwer Samby za pośrednictwem serwera NT. Powinno to działać w sposób niewidoczny dla użytkownika.

Opcje konfiguracji SSL

W tabeli A.1 zebrano opcje konfiguracyjne opisane w poprzednich podrozdziałach i związane z obsługą SSL. Zauważ, że wszystkie te opcje mają zasięg globalny; innymi słowy, muszą występować w sekcji `[global]` pliku konfiguracyjnego.

Tabela A.1. Opcje konfiguracji SSL

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`ssl`	Wartość logiczna	Określa, czy Samba pracuje w trybie SSL	`no`	Globalny
`ssl hosts`	Łańcuch (lista adresów)	Określa listę hostów, które muszą się łączyć za pomocą SSL	Brak	Globalny
`ssl hosts resign`	Łańcuch (lista adresów)	Określa listę hostów, które nigdy nie łączą się za pomocą SSL	Brak	Globalny
`ssl CA certDir`	Łańcuch (pełna ścieżka)	Określa katalog, w którym przechowywane są certyfikaty	Brak	Globalny
`ssl CA certFile`	Łańcuch (nazwa wraz ze ścieżką)	Określa plik, w którym przechowywane są wszystkie certyfikaty Samby	Brak	Globalny
`ssl server cert`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa położenie certyfikatu serwera	Brak	Globalny
`ssl server key`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa położenie prywatnego klucza serwera	Brak	Globalny
`ssl client cert`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa położenie certyfikatu klienta	Brak	Globalny

Dokończenie tabeli na str. 292

Dokończenie tabeli ze str. 291

Tabela A.1. Opcje konfiguracji SSL

Opcja	*Parametry*	*Funkcja*	*Wartość domyślna*	*Zasięg*
`ssl client key`	Łańcuch (pełna nazwa wraz ze ścieżką)	Określa położenie prywatnego klucza klienta	Brak	Globalny
`ssl require clientcert`	Wartość logiczna	Określa, czy Samba powinna wymagać certyfikatów od wszystkich klientów	`no`	Globalny
`ssl require servercert`	Wartość logiczna	Określa, czy sam serwer powinien mieć certyfikat	`no`	Globalny
`ssl ciphers`	Łańcuch	Określa zbiór szyfrów używany podczas negocjacji protokołu	Brak	Globalny
`ssl version`	`ssl2or3`, `ssl3` lub `tls1`	Określa używaną wersję SSL	`ssl2or3`	Globalny
`ssl compatibility`	Wartość logiczna	Określa, czy należy włączyć tryb zgodności z innymi implementacjami SSL	`no`	Globalny

ssl

Ta opcja globalna konfiguruje Sambę do użycia SSL w komunikacji z klientami. Jej domyślna wartość to `no`. Możesz zmienić ją w następujący sposób:

```
[global]
    ssl = yes
```

Aby skorzystać z tej opcji, potrzebujesz pośrednika dla klientów Windows 95/98, co opisano wcześniej w tym rozdziale.

ssl hosts

Ta opcja określa hosty, które będą musiały używać SSL. Składnia używana do określania hostów i adresów jest taka sama jak w opcjach `hosts allow` i `hosts deny`. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
```

W tym przykładzie określamy, że wszystkie hosty z podsieci 192.168.220 muszą używać połączeń SSL. Taki model jest użyteczny, jeśli wiesz, że niektóre połączenia będą pochodzić z podsieci oddzielonej siecią niegodną zaufania, na przykład Internetem. Jeśli nie podasz ani tej opcji, ani opcji `ssl hosts resign`, a opcja `ssl` jest ustawiona na `yes`, Samba będzie akceptować tylko połączenia SSL od wszystkich klientów.

ssl hosts resign

Ta opcja określa hosty, które nie będą musiały używać SSL. Składnia używana do określania hostów i adresów jest taka sama jak w opcjach `hosts allow` i `hosts deny`. Na przykład:

```
[global]
    ssl = yes
    ssl hosts resign = 160.2.310. 160.2.320.
```

W tym przykładzie określamy, że hosty z podsieci 160.2.310 i 160.2.320 nie będą używały połączeń SSL. Jeśli nie podasz ani tej opcji, ani opcji `ssl hosts`, a opcja `ssl` jest ustawiona na `yes`, Samba będzie akceptować tylko połączenia SSL od wszystkich klientów.

ssl CA certDir

Ta opcja określa położenie katalogu z certyfikatami serwisu certyfikacyjnego, których Samba będzie używała do uwierzytelniania klientów. W katalogu tym musi znajdować się jeden plik dla każdego serwisu certyfikacyjnego, nazwany tak, jak opisano wcześniej w tym rozdziale. Wszystkie inne pliki w tym katalogu są ignorowane. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certDir = /usr/local/samba/cert
```

Opcja ta nie ma wartości domyślnej. Możesz zamiast niej użyć opcji `ssl CA certFile`, jeśli chcesz umieścić wszystkie dane certyfikacyjne w jednym pliku.

ssl CA certFile

Ta opcja określa położenie pliku z certyfikatami serwisu certyfikacyjnego, których Samba będzie używała do uwierzytelniania klientów. Różni się od opcji `ssl CA certDir` tym, że dane wszystkich serwisów certyfikacyjnych są przechowywane w jednym pliku. Oto przykład użycia tej opcji:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
```

Opcja ta nie ma wartości domyślnej. Możesz zamiast niej użyć opcji `ssl CA certDir`, jeśli wolisz przechowywać oddzielne pliki dla każdego serwisu certyfikacyjnego.

ssl server cert

Ta opcja określa położenie certyfikatu serwera. Jest obowiązkowa; serwer musi mieć certyfikat, aby mógł używać SSL. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
    ssl server cert = /usr/local/samba/private/server.pem
```

Opcja ta nie ma wartości domyślnej. Zauważ, że certyfikat może zawierać prywatny klucz serwera.

ssl server key

Ta opcja określa położenie prywatnego klucza serwera. Powinieneś upewnić się, że do tego pliku nie ma dostępu nikt z wyjątkiem roota. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
    ssl server key = /usr/local/samba/private/samba.pem
```

Opcja ta nie ma wartości domyślnej. Zauważ, że certyfikat serwera może zawierać jego prywatny klucz.

ssl client cert

Ta opcja określa położenie certyfikatu klienta. Samba może zażądać certyfikatu za pomocą opcji `ssl require clientcert`; certyfikat jest wykorzystywany także przez program *smbclient*. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
    ssl server cert = /usr/local/ssl/private/serwer.pem
    ssl client cert = /usr/local/ssl/private/klient.pem
```

Opcja ta nie ma wartości domyślnej.

ssl client key

Ta opcja określa położenie prywatnego klucza klienta. Powinieneś upewnić się, że do tego pliku nie ma dostępu nikt z wyjątkiem roota. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
    ssl server key = /usr/local/ssl/private/samba.pem
    ssl client key = /usr/local/ssl/private/klienty.pem
```

Opcja ta nie wartości domyślnej. Jest potrzebna tylko wtedy, gdy klient ma certyfikat.

ssl require clientcert

Ta opcja określa, czy klient musi mieć certyfikat. Jeśli jest włączona, Samba będzie przeszukiwać pliki określone opcjami `ssl CA certDir` lub `ssl CA certFile`, aby potwierdzić, że klient ma ważny certyfikat i jest uprawniony do łączenia się z serwerem Samby. Opcja ta przyjmuje wartości logiczne. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
    ssl require clientcert = yes
```

Powinieneś wymagać certyfikatów od wszystkich klientów, które będą się łączyć z serwerem Samby. Domyślna wartość tej opcji to `no`.

ssl require servercert

Opcja ta określa, czy serwer musi mieć certyfikat i jest używana przez program *smbclient*. Opcja ta przyjmuje wartości logiczne. Na przykład:

```
[global]
    ssl = yes
    ssl hosts = 192.168.220.
    ssl CA certFile = /usr/local/samba/cert/certFile
    ssl require clientcert = yes
    ssl require servercert = yes
```

Choć należy wymagać certyfikatów od wszystkich klientów łączących się z serwerem Samby, certyfikat dla serwera nie jest niezbędny. Zalecamy jednak jego stosowanie. Domyślna wartość tej opcji to `no`.

ssl ciphers

Ta opcja określa szyfry, które Samba ustali w fazie negocjacji połączenia SSL. Samba może korzystać z następujących szyfrów:

```
DEFAULT
DES-CFB-M1
NULL-MD5
RC4-MD5
EXP-RC4-MD5
RC2-CBC-MD5
EXP-RC2-CBC-MD5
IDEA-CBC-MD5
DES-CBC-MD5
DES-CBC-SHA
DES-CBC3-MD5
DES-CBC3-SHA
RC4-64-MD5
NULL
```

Nie powinieneś zmieniać tej opcji, chyba że dobrze znasz protokół SSL i chcesz wymusić użycie określonego zbioru szyfrów.

ssl version

Ta globalna opcja określa wersję SSL, której Samba będzie używać podczas obsługi zaszyfrowanych połączeń. Jej domyślna wartość to `ssl2or3`, co oznacza, że można użyć wersji 2 lub 3 protokołu SSL, w zależności od wyniku negocjacji między serwerem i klientem. Jeśli jednak chcesz, aby Samba używała konkretnej wersji protokołu, możesz użyć następującej opcji:

```
[global]
    ssl version = ssl3
```

Nie powinieneś zmieniać tej opcji, chyba że dobrze znasz protokół SSL i chcesz wymusić użycie określonej wersji.

ssl compatibility

Ta globalna opcja określa, czy Samba powinna korzystać z innych wersji SSL. Gdy pisaliśmy tę książkę, nie istniały jednak żadne inne wersje SSL, więc opcja ta chwilowo nie ma zastosowania i powinna zawsze mieć wartość domyślną.

B

Optymalizowanie wydajności Samby

W tym dodatku omówimy różne sposoby optymalizowania wydajności Samby i skalowania systemu. *Optymalizowanie wydajności* to sztuka wykrywania „wąskich gardeł" i ich eliminowania. *Skalowanie* to nieco odmienne podejście do problemu wąskich gardeł, polegające na wydaniu takiej ilości pieniędzy, aby takowe w ogóle nie powstały. Zwykle nie musisz przejmować się ani jednym, ani drugim. Zupełnie niezoptymalizowany serwer Samby bez problemu obsłuży niewielką grupę użytkowników. Jeśli jednak serwer jest odpowiednio dostrojony, będzie mógł służyć grupie przynajmniej dwa razy większej. Niniejszy rozdział jest poświęcony różnym technikom optymalizowania wydajności i skalowania, które mogą okazać się przydatne, jeśli chcesz uzyskać maksymalną wydajność serwera Samby.

Przykładowy pomiar wydajności

Skąd możesz wiedzieć, czy osiągasz rozsądną wydajność? Najprościej będzie porównać Sambę z FTP. W tabeli B.1 przedstawiamy przepustowość dwóch serwerów: średnio wydajnego serwera Sun SPARC Ultra i niewielkiego serwera Pentium z Linuksem. Wyniki są podane w kilobajtach na sekundę (KB/s).

Tabela B.1. Przykładowy pomiar wydajności

Polecenie	*FTP*	*Samba niezoptymalizowana*	*Samba zoptymalizowana*
Sparc get	1014,5	645,3	866,7
Sparc put	379,8	386,1	329,5
Pentium get	973,27	N/D	725
Pentium put	1014,5	N/D	1100

Jeśli przeprowadzisz taki test na własnym serwerze, prawdopodobnie uzyskasz inne wyniki. Powinieneś jednak otrzymać podobny stosunek przepustowości Samby do FTP, zwykle w zakresie od 68 do 80 procent. FTP nie powinno być jedynym miernikiem wydajności Samby, ale zapamiętaj ogólną zasadę: jeśli Samba jest dużo wolniejsza od FTP, należy ją dostroić.

Być może sądzisz, że równie dobrze można porównać Sambę z NFS. W rzeczywistości jednak porównywanie ich prędkości jest dużo mniej użyteczne. W zależności od używanej wersji NFS i poziomu optymalizacji, Samba może być wolniejsza lub szybsza od NFS. Zwykle Samba okazuje się szybsza, ale należy tu zachować ostrożność: NFS korzysta z zupełnie innego algorytmu niż Samba, więc parametry optymalne dla NFS mogą okazać się szkodliwe dla Samby. Jeśli uruchomisz Sambę na dobrze dostrojonym serwerze NFS, jej wydajność może pozostawiać wiele do życzenia.

Jednym z popularnych testów wydajności jest *NetBench* Ziffa-Davisa, symulacja wielu użytkowników pracujących w edytorach tekstów i korzystających z danych serwera SMB. Pomiar ten nie jest doskonały (każdy klient programu NetBench wykonuje dziesięciokrotnie więcej pracy, niż przeciętny użytkownik w naszej sieci), ale umożliwia sprawiedliwe porównanie podobnych serwerów. W testach przeprowadzonych przez Jeremy'ego Allisona w listopadzie 1998 roku Samba 2.0 zainstalowana w wieloprocesorowym komputerze SGI przewyższyła wydajnością serwer NT 4.0 (SP 2) zainstalowany w równoważnym serwerze Compaqa. Wyniki te zostały potwierdzone przez test Linuksa i NT na identycznym sprzęcie, przeprowadzony przez Sm@rt Reseller w lutym w 1999 roku.

W kwietniu 1999 roku laboratorium testowe Mindcraft opublikowało raport z testu, który wykazał, że Linux zainstalowany w czteroprocesorowym komputerze był znacznie wolniejszy w serwowaniu plików niż Windows NT działający na tym samym sprzęcie. Pierwotny raport został zdyskredytowany przez społeczność Open Source, ponieważ był zamówiony przez Microsoft, a system zoptymalizowano na korzyść Windows NT, ale następny test był sprawiedliwszy i wykazał pewne obszary, w których Linux powinien nieco polepszyć swoją wydajność, zwłaszcza w komputerach wieloprocesorowych. Niewiele powiedziano w nim o samej Sambie. Wiadomo, że Samba dobrze się skaluje w takich komputerach i w czteroprocesorowym serwerze SGI O200 osiąga przepustowość ponad 440 MB/s, czyli znacznie większą niż 310 MB/s z testów Mindcraftu.

Oczywiście, stosunek wydajności może się zmienić wraz ze wzrostem prędkości Windows NT i sprzętu PC, ale Samba również się rozwija. Na przykład wersja 1.9.18 Samby była szybsza od NT tylko podczas obsługi więcej niż 35 klientów, a Samba 2.0 jest szybsza niezależnie od liczby klientów. Podsumowując, Samba stanowi poważną konkurencję dla najlepszego komercyjnego oprogramowania sieciowego i wciąż jest ulepszana.

Kiedy oddawaliśmy tę książkę do druku, Andrew Tridgell udostępnił wersję alfa pakietu programów testowych dla Samby i sieci SMB. Zespół twórców Samby nie zamierza rezygnować z dalszego poprawiania wydajności programu.

Optymalizowanie Samby

Po tym wstępie zastanówmy się, jak można przyspieszyć i tak bardzo szybki pakiet sieciowy.

Pomiary

Mierzenie wydajności to sztuka tajemna i graniczącą z czarną magią, ale poziom umiejętności niezbędny do prostej optymalizacji systemu nie jest aż tak wysoki. Ponieważ zasadniczym przeznaczeniem Samby jest transmisja plików, pod mikroskop weźmiemy tylko przepustowość, a nie czasy odpowiedzi na różne zdarzenia. Poza tym, dość łatwo jest zmierzyć prędkość transmisji plików, a Samba nie cierpi na problemy z czasem odpowiedzi, które wymagałaby użycia bardziej zaawansowanych technik pomiarowych.

Nasza strategia przedstawia się następująco:

- Znaleźć plik do skopiowania, który będzie miał rozsądną wielkość, oraz program informujący o czasie kopiowania, taki jak *smbclient*.
- Przeprowadzić test w „cichym" (lub typowym) okresie.
- Wykonać test kilkakrotnie przed przeprowadzeniem właściwego pomiaru, aby wstępnie załadować bufory.
- Przeprowadzić testy kilka razy i sprawdzić, czy nie dają nieoczekiwanych wyników.
- Zanotować szczegółowo wyniki każdego testu.
- Porównać średni wynik testów z oczekiwanym rezultatem.

Wyznaczywszy w ten sposób ogólne ramy eksperymentu, możemy zmieniać pojedyncze parametry i ponownie wykonywać pomiary. Na końcu rozdziału zamieściliśmy tabelę, do której możesz wpisywać wyniki własnych testów.

Co można zmieniać?

W poszukiwaniu doskonałego serwera Samby możesz wypróbować dosłownie tysiące kombinacji różnych parametrów. Jeśli jednak administrowanie systemem nie jest całym twoim życiem, możesz ograniczyć się do modyfikowania opcji opisanych w tym rozdziale, ponieważ ogólna przepustowość zależy głównie od nich. Przedstawione poniżej opcje są uporządkowane według wpływu na wydajność Samby.

Poziom rejestrowania

Tutaj sprawa jest oczywista. Zwiększenie poziomu rejestrowania (opcje konfiguracyjne `log level` lub `debug level`) jest doskonałą metodą diagnozowania problemów, ale nie problemów z wydajnością! Jak wspomniano w rozdziale 4, *Udziały dyskowe*, na poziomie rejestrowania 3 i wyższych Samba produkuje ogromne ilości komunikatów diagnostycznych, a zapisywanie ich na dysku lub w dzienniku systemowym jest wolną operacją. W naszych testach *smbclient/ftp* podniesienie poziomu rejestrowania z 0 na 3 zmniejszyło prędkość niezoptymalizowanej operacji pobierania pliku z 645,3 do 622,2 KB/s, czyli o niespełna 5 procent. Wyższe poziomy rejestrowania były jeszcze gorsze.

Opcje gniazd

Następnym podejrzanym są opcje konfiguracyjne `socket options`. Są to właściwie opcje dostrajania systemu goszczącego Sambę, ale ustawia się je dla każdego połączenia z osobna i mogą być ustawiane przez Sambę w używanych przez nią gniazdach za pomocą opcji `socket options = opcja` w sekcji `[global]` pliku *smb.conf*. Nie wszystkie spośród tych opcji są obsługiwane w każdym systemie; szczegóły znajdziesz na stronach podręcznika man dla funkcji *setsockopt*(1) lub *socket*(5).

Najważniejsze opcje to:

`TCP_NODELAY`
: Opcja ta sprawia, że serwer wysyła tyle pakietów, ile potrzeba do utrzymania zwłoki na niskim poziomie. Używa się jej w połączeniach telnetowych, aby uzyskać krótki czas odpowiedzi, a także – choć jest to nieco sprzeczne z intuicją – aby osiągnąć przyzwoitą prędkość nawet przy niewielkich żądaniach oraz przy opóźnionych potwierdzeniach (co wydaje się być cechą stosu TCP/IP Microsoftu). Już sama ta opcja może przyspieszyć operacje o 30 do 50 procent. Tak na marginesie, w Sambie 2.0.4 `TCP_NODELAY` jest domyślną wartością opcji `socket options`.

`IPTOS_LOWDELAY`
: Jest to kolejna opcja, która zmniejsza zwłokę kosztem przepustowości, ale oddziałuje na rutery i inne systemy, a nie na serwer. Opcje `IPTOS` są nowością i nie są obsługiwane przez wszystkie systemy operacyjne i rutery. Jeśli system je obsługuje, ustaw `IPTOS_LOWDELAY`, gdy ustawiasz `TCP_NODELAY`.

`SO_SNDBUF` i `SO_RCVBUF`
: Bufory nadawczy i odbiorczy można często ustawić na wielkość większą niż domyślna dla systemu operacyjnego. Daje to nieznaczne zwiększenie prędkości (do punktu, w którym zyski stają się niezauważalne).

`SO_KEEPALIVE`
: Opcja ta zapoczątkowuje okresowe (co cztery godziny) sprawdzanie, czy klient nie zniknął z sieci. Z wygasłymi połączeniami lepiej sobie jednak radzić za pomocą opcji Samby `keepalive` i `dead time`. Wszystkie trzy rozwiązania umożliwiają zamknięcie nieaktywnych połączeń i zwrócenie pamięci oraz wpisów z tablicy procesów do puli systemu operacyjnego.

Istnieją inne opcje gniazd, którymi warto się zainteresować (na przykład `SO_SNDLOWAT`), ale nie są one dostępne we wszystkich systemach. Jeśli zamierzasz zbadać ich wpływ na wydajność Samby, powinieneś przeczytać książkę *TCP/IP Illustrated*.

read raw i write raw

Te opcje konfiguracyjne mają istotny wpływ na wydajność; umożliwiają Sambie zapisywanie i odczytywanie dużych bloków danych z sieci, do 64 KB w jednym żądaniu SMB. Wymagają także największych struktur pakietu SMB, `SMBreadraw` i `SMBwriteraw`, od których biorą swoje nazwy. Zauważ, że to nie to samo, co „surowy" odczyt danych (*raw read*) w Uniksie. Termin uniksowy odnosi się zwykle do odczy-

tywania dysków z pominięciem systemu plików, a więc ma całkiem inny sens niż w Sambie.

W przeszłości niektóre programy klienckie odmawiały pracy, jeśli spróbowałeś użyć opcji `read raw`. O ile nam wiadomo, obecnie żaden klient nie cierpi na tę przypadłość. Opcje `read raw` i `write raw` mają domyślną wartość `yes` i nie powinieneś ich zmieniać, chyba że odkryjesz w swojej sieci klienta z taką usterką.

Blokowanie oportunistyczne

Blokady oportunistyczne pozwalają klientom na lokalne buforowanie plików, co zwiększa wydajność o prawie 30 procent. Opcja ta jest obecnie domyślnie włączona. W przypadku plików przeznaczonych tylko do odczytu opcja `fake oplocks` spełnia tę samą funkcję, choć w rzeczywistości wcale nie buforuje danych. Jeśli masz pliki, które nie mogą być buforowane, można wyłączyć blokady oportunistyczne.

Pliki baz danych nigdy nie powinny być buforowane, podobnie jak pliki, które są uaktualniane i w serwerze, i w kliencie, a których zmiany powinny być natychmiast widoczne. Opcja `veto oplock files` umożliwia podanie listy pojedynczych plików lub wzorców nazw plików, które nie powinny być buforowane. Blokady oportunistyczne można wyłączać dla każdego udziału z osobna, jeśli masz duże grupy plików, których klienty nie powinny buforować. Więcej informacji o blokadach oportunistycznych znajdziesz w rozdziale 5, *Przeglądanie i zaawansowane udziały dyskowe*.

Rozmiar pakietu IP (MTU)

Sieci zwykle mają górną granicę rozmiaru pojedynczej transmisji lub pakietu. Nosi ona nazwę maksymalnego rozmiaru segmentu (*Maximum Segment Size*), a jeśli uwzględni się także rozmiar nagłówka pakietu – maksymalnej jednostki transmisji (*Maximum Transport Unit*, MTU). Jednostka MTU nie jest ustawiana przez Sambę, ale Samba powinna używać opcji `max xmit` (rozmiaru zapisu) większej od MTU, gdyż w przeciwnym wypadku zmniejszy się przepustowość. Omówiono to bliżej w poniższej nocie. Jednostka MTU jest równa 1500 bajtom w Ethernecie i 4098 bajtom w FDDI. Ogólnie rzecz biorąc, zbyt niska wartość MTU zmniejsza przepustowość, a zbyt wysoka powoduje nagłe spadki wydajności spowodowane fragmentacją i retransmisjami.

Jeśli komunikacja przebiega przez ruter, niektóre systemy zakładają, że ruter dysponuje łączem szeregowym (na przykład T1) i ustawiają MTU na mniej więcej 536 bajtów. Taki błąd, popełniany na przykład przez Windows 95, powoduje, że pobliskie klienty działają sprawnie, ale klienty po drugiej stronie rutera są zauważalnie wolniejsze. Jeśli klient popełnia odwrotny błąd i używa dużej wartości MTU na łączu wymagającym niższej wartości, pakiety zostaną rozbite na fragmenty. Powoduje to nieznaczne spowolnienie transmisji, a każdy błąd w sieci wiąże się z retransmisją wielu fragmentów, co znacznie pogarsza wydajność Samby. Na szczęście możesz zmienić rozmiar MTU w Windows, aby zapobiec takim błędom. Więcej informacji na ten temat znajdziesz w dokumencie „The Windows 95 Networking Frequently Asked Questions (FAQ)" pod adresem: *http://www.stanford.edu/~llurch/win95netbugs/faq.html*, gdzie wyjaśniono, jak zmienić ustawienia MTU i rozmiaru okna TCP w Windows.

Okno odbiorcze TCP

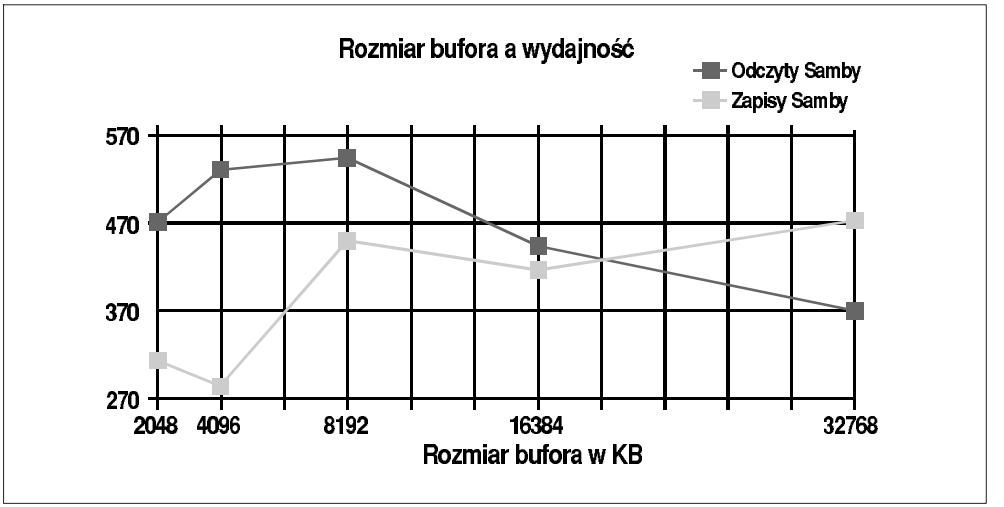
Protokół TCP/IP rozbija dane na małe pakiety, które można przesłać od jednego komputera do drugiego. Każdy wysłany pakiet zawiera sumę kontrolną, dzięki której odbiorca może sprawdzić, czy podczas transmisji dane pakietu nie zostały przekłamane. Teoretycznie, po odebraniu i zweryfikowaniu pakietu do nadawcy powinno zostać wysłane potwierdzenie, które informuje, że dane przybyły nienaruszone, i pozwala na kontynuowanie transmisji.

Aby przyspieszyć przesyłanie danych, TCP akceptuje zakres pakietów (tak zwane okno), które pozwala nadawcy na kontynuowanie transmisji bez czekania na potwierdzenie każdego pakietu (po zgromadzeniu kilku potwierdzeń można je jednocześnie odesłać do nadawcy). Innymi słowy, okno odbiorcze to liczba bajtów, które nadawca może wysłać, zanim będzie musiał wstrzymać transmisję i zaczekać na potwierdzenie od odbiorcy. Podobnie jak MTU, liczba ta jest automatycznie ustawiana w zależności od typu połączenia. Zbyt małe okno powoduje niepotrzebne czekanie na potwierdzenia. Wiele systemów operacyjnych ustawia umiarkowane wielkości bufora dla poszczególnych gniazd, aby zapobiec zmonopolizowaniu pamięci przez jeden program.

Rozmiary buforów określa się w bajtach, na przykład pisząc `SO_SNDBUF=8192` w linii opcji `socket options`. Tak więc przykładowa opcja konfiguracyjna `socket options` może wyglądać następująco:

```
socket options = SO_SNDBUF=8192
```

Zwykle warto ustawić te opcje gniazd na wartości większe od domyślnych: 4098 w SunOS 4.1.3 i SVR4 oraz 8192 lub 16384 w AIX-ie, Solarisie i BSD. Na początek można wypróbować wartość 16384, co sugeruje test (nie Samby) zamieszczony w książce Stevensa – daje ona 40-procentowy wzrost wydajności. Będziesz musiał poeksperymentować, ponieważ wydajność znów spadnie, kiedy ustawisz zbyt duży rozmiar bufora. Ilustruje to rysunek B.1, przedstawiający wyniki testu w systemie linuksowym.



Rysunek B.1. Rozmiar bufora SO_SNDBUF a wydajność

Ustawianie opcji gniazd `SO_SNFBUF` i `SO_RCVBUF` na wartości mniejsze od domyślnych nie jest godne polecenia. Ustawienie ich na wyższą wartość zwiększa wydajność, aż do pewnej granicy, charakterystycznej dla sieci. Po przekroczeniu tej granicy wydajność stabilizuje się na nieco niższym poziomie.

max xmit

Opcją Samby bezpośrednio związaną z MTU i rozmiarem okna jest `max xmit`. Ustawia ona maksymalną wielkość bloku danych, który Samba może zapisać w jednej operacji. Czasem wielkość tę nazywa się *rozmiarem zapisu* (*write size*), choć w Sambie nie istnieje opcja konfiguracyjna o takiej nazwie.

Ponieważ procentowa część każdego bloku przeznaczona na informacje dodatkowe maleje wraz ze wzrostem wielkości bloku, opcję `max xmit` ustawia się zwykle na największą możliwą wartość. Domyślnie jest ona równa limitowi protokołu, czyli 64 kilobajtom. Najmniejsza wartość, która nie powoduje znacznego spowolnienia, to 2048 bajtów. Jeśli ustawi się tę opcję wystarczająco nisko, ograniczy ona maksymalny rozmiar pakietów możliwy do wynegocjowania przez Sambę. Możesz użyć jej do zasymulowania niewielkiej jednostki MTU, jeśli chcesz przetestować zawodne połączenie sieciowe. Podczas normalnej pracy Samby nie powinieneś jednak redukować MTU za pomocą tej opcji.

read size

Ponieważ wielkość zapisywanego bloku danych bywa nazywana rozmiarem zapisu (*write size*), można by się spodziewać, że opcja `read size` określa maksymalną wielkość bloku danych odczytywanych z sieci. Tak jednak nie jest. W istocie opcja ta włącza *zapis z wyprzedzeniem* (*write ahead*). Oznacza to, że jeśli Samba spóźnia się z odczytem dysku względem zapisywania danych w sieci (lub vice versa) o określoną liczbę bajtów, zacznie nakładać zapisy w sieci na odczyty dysku (lub vice versa).

Opcja ta nie ma w Uniksie dużego wpływu na wydajność, chyba że przypiszesz jej bardzo niską wartość. Wtedy spowoduje zauważalne spowolnienie. Z tego względu ma ona domyślną wartość 2048 i nie może być mniejsza niż 1024.

read prediction

Opcja ta jest nie tylko sprzeczna z intuicją, ale także przestarzała. Umożliwia Sambie odczytywanie z wyprzedzeniem plików, które zostały otworzone w trybie tylko do odczytu. Opcja ta jest wyłączona w Sambie 2.0 (i późnych wersjach 1.9), ponieważ przeszkadza w blokowaniu oportunistycznym.

Inne opcje Samby

Poniższe opcje Samby będą miały wpływ na wydajność, jeśli zostaną ustawione nieprawidłowo, podobnie jak omówiliśmy to w przypadku opcji `debug level`. Wspominamy tu o nich, abyś wiedział, czego się wystrzegać.

`hide files`
: Podanie wzorca określającego pliki, które powinny być ukryte dla klientów Windows, powoduje, że każdy plik pasujący do wzorca zostanie przekazany z ustawionym dosowym atrybutem ukrycia. Wymaga to operacji dopasowania wzorca dla każdego pliku w listowanym katalogu, co zauważalnie spowalnia działanie serwera.

`lpq cache time`
: Jeśli twoje polecenie `lpq` (zwracające zawartość kolejki wydruku) wykonuje się długo, powinieneś ustawić opcję `lpq cache time` na czas dłuższy od rzeczywistego czasu wykonania tego polecenia, żeby Samba nie uruchamiała nowego zapytania, gdy poprzednie jest jeszcze w toku. Domyślna – i rozsądna – wartość tej opcji to 10 sekund.

`strict locking`
: Włączenie opcji `strict locking` powoduje, że Samba sprawdza obecność blokad przy każdym dostępie do pliku, a nie tylko wtedy, gdy klient o to poprosi. Opcja ta służy głównie jako zabezpieczenie przed usterkami i może zapobiec uszkodzeniu współdzielonych plików przez błędnie działające aplikacje DOS-a i Windows. Powoduje jednak spowolnienie i powinna być wyłączona.

`strict sync`
: Włączenie opcji `strict sync` powoduje, że Samba zapisuje każdy pakiet na dysku i czeka na dokończenie zapisu, kiedy klient ustawi bit synchronizacji w pakiecie. Eksplorator Windows 98 ustawia ten bit we wszystkich transmitowanych pakietach, więc jeśli włączysz tę opcję, użytkownicy Windows 98 będą uważali serwery Samby za nieznośnie powolne.

`sync always`
: Ustawienie opcji `sync always` na `yes` sprawia, że Samba wymusza fizyczne zapisywanie danych na dysku. Ma to sens, jeśli twój serwer co chwila ulega awarii, ale prowadzi do drastycznego spadku wydajności. Serwery SMB w celu zapobiegania skutkom awarii korzystają z blokad oportunistycznych i automatycznego odnawiania połączeń, więc w normalnych okolicznościach ustawianie tej opcji nie jest konieczne.

`wide links`
: Wyłączenie opcji `wide links` sprawia, że Samba nie podąża za dowiązaniami symbolicznymi znajdującymi się w jednym udziale, a wskazującymi na pliki w innym udziale. Jest ona domyślnie włączona, ponieważ podążanie za dowiązaniami w Uniksie nie stanowi zagrożenia bezpieczeństwa. Wyłączenie jej wymaga dodatkowego przetwarzania każdego otwartego pliku. Jeśli wyłączysz opcję `wide links`, koniecznie włącz `getwd cache`, aby buforować pewną ilość wymaganych danych.

  Istnieje także opcja `follow symlinks`, którą można wyłączyć, aby zapobiec podążaniu za jakimkolwiek dowiązaniem symbolicznym. Ta opcja jednak nie ma wpływu na wydajność.

`getwd cache`
Ta opcja buforuje ścieżkę do bieżącego katalogu, zapobiegając jej czasochłonnemu wyszukiwaniu w drzewie katalogów. Może poprawić wydajność serwera wydruku i przydaje się, gdy wyłączona jest opcja `wide links`.

Nasze zalecenia

Oto plik *smb.conf*, który zawiera omówione do tej pory ulepszenia wydajności. Po prawej stronie zamieszczono komentarze.

```
[global]
    log level = 1             # Domy□lnie 0
    socket options = TCP_NODELAY IPTOS_LOWDELAY
    read raw = yes            # Ustawienie domy□lne
    write raw = yes           # Ustawienie domy□lne
    oplocks = yes             # Ustawienie domy□lne
    max xmit = 65535          # Ustawienie domy□lne
    dead time = 15            # Domy□lnie 0
    getwd cache = yes
    lpq cache = 30
[dobremiejsce]
    veto oplock files = ten/tamten/innyplik
[zlemiejsce]
    oplocks = no
```

Skalowanie serwerów Samby

Skalowanie to sposób na uniknięcie wąskich gardeł, zanim będą miały okazję się objawić. Trzeba w tym celu wiedzieć, ilu żądań na sekundę lub ilu kilobajtów na sekundę będą wymagać klienty, i upewnić się, że wszystkie składniki serwera mogą spełnić te wymagania.

Wąskie gardła

Trzy zasadnicze wąskie gardła, na które powinieneś zwrócić uwagę, to procesor, operacje dyskowe i sieć. W większości komputerów procesor nie stanowi wąskiego gardła. Pojedynczy procesor Sun SPARC 10 może zapoczątkować (i zakończyć) od 700 do 800 operacji wejścia-wyjścia na sekundę, co daje przepustowość między 5 600 a 6 400 KB/s, kiedy rozmiar bloku danych wynosi 8 KB (jest to często używany rozmiar bufora). Pojedynczy procesor Intel Pentium 133 wykonuje ich nieco mniej ze względu na wolniejszą pamięć podręczną i interfejs magistrali, a nie z powodu braku mocy obliczeniowej. Specjalnie zaprojektowane serwery Pentium, na przykład niektóre modele Compaqa, mogą zapoczątkować 700 operacji na procesor, korzystając nawet z czterech procesorów.

Zbyt mała pamięć może natomiast stanowić wąskie gardło; każdy proces Samby zajmuje od 600 do 800 KB w Linuksie na platformie Intela, a nawet więcej w przypadku procesorów RISC. Jeśli pamięci fizycznej jest za mało, wzrośnie częstotliwość stronicowania do pamięci wirtualnej i ucierpi wydajność. W systemie Solaris zmierzono, że *smbd* zajmuje 2,6 MB pamięci na program i współdzielone biblioteki oraz 768 KB

na każdego podłączonego klienta. Proces *nmbd* zajmuje 2,1 MB i dodatkowo 496 KB na swój (pojedynczy) proces pomocniczy.

Dyski twarde zawsze stają się wąskim gardłem przy określonej liczbie operacji wejścia-wyjścia na sekundę. Na przykład każdy dysk SCSI o prędkości 7200 RPM może przeprowadzić 70 operacji na sekundę, co daje przepustowość 560 KB/s; dysk o prędkości 4800 RPM może wykonać do 50 operacji, co daje przepustowość 360 KB/s. Dyski IDE przeprowadzają jeszcze mniej operacji. Jeśli dyski są niezależne lub połączone w paskową macierz RAID 1, każdy z nich będzie miał szczytową przepustowość od 400 do 560 KB/s, a jeśli dołożysz następne dyski, wydajność będzie liniowo rosła. Zauważ, że odnosi się to tylko do macierzy RAID 1. Poziomy RAID inne niż 1 (paskowe) dodają uboczny narzut.

Sieci Ethernet (i inne) są oczywistym wąskim gardłem: Ethernet 10 Mb/s (mega*bity* na sekundę) ma przepustowość około 1100 KB/s (kilo*bajty* na sekundę) przy użyciu pakietów o rozmiarze 1500 bajtów. Fast Ethernet 100 Mb/s staje się wąskim gardłem przy przepustowości 6500 KB/s i takim samym rozmiarze pakietów. Górna granica przepustowości dla FDDI (155 Mb/s) leży w okolicach 6250 KB/s, ale sieć ta pozostaje sprawna nawet przy stuprocentowym obciążeniu i transmituje dużo większe pakiety (4 KB).

Sieć ATM powinna działać dużo lepiej, ale jest to jeszcze zbyt nowa technologia, aby wykorzystywała cały swój potencjał. Obecnie może dostarczać około 7125 Mb/s przy użyciu pakietów o rozmiarze 9 KB.

Oczywiście, mogą istnieć inne wąskie gardła: nie zaleca się dołączania więcej niż jednego dysku IDE do jednego kontrolera, podobnie jak więcej niż trzech dysków 3600 SCSI-I do wolniejszego kontrolera (typu narrow) i niż trzech dysków 7200 SCSI-II do szybszego kontrolera (typu wide). Macierz RAID 5 również jest dość wolna, ponieważ wymaga dwa razy więcej zapisów niż niezależne dyski lub macierz RAID 1.

Po dodaniu drugiej karty sieciowej i drugiego kontrolera dysku warto przyjrzeć się przepustowości magistrali, zwłaszcza jeśli używasz magistral ISA/EISA.

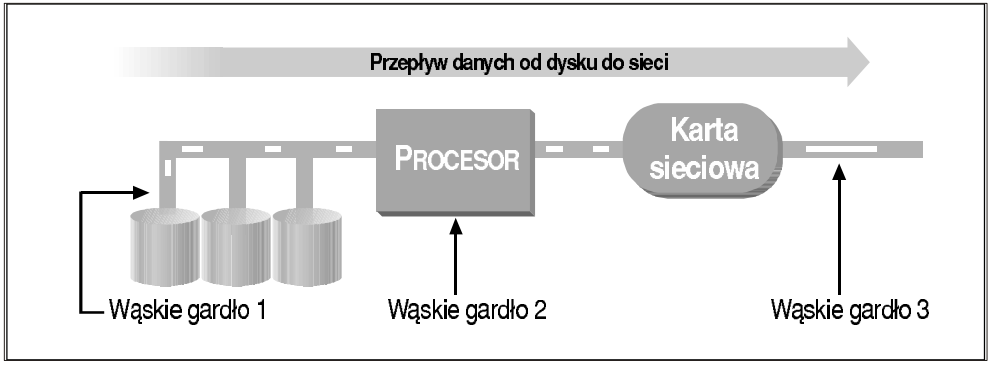
Eliminowanie wąskich gardeł

Na podstawie powyższych informacji możemy wypracować model, który pozwoli nam ustalić maksymalną wydajność konkretnego komputera. Przedstawione tu dane zaczerpnęliśmy głównie z książki Briana Wonga *Configuration and Capacity Planning for Solaris Servers**, która opisuje serwery firmy Sun, co do pewnego stopnia widać w naszych przykładach.

Słowo ostrzeżenia: nie jest to pełny model. Nie powinieneś zakładać, że dzięki niemu zdołasz przewidzieć wszystkie wąskie gardła, a błąd w szacunkach nie przekroczy 10 procent. Model, który przewidywałby wydajność, a nie tylko ostrzegał przed wąskimi gardłami, musiałby być dużo bardziej skomplikowany i zawierać reguły w rodzaju „nie więcej niż trzy dyski SCSI na jeden łańcuch" (dobrą książką o rzeczy-

* Patrz Brian L. Wong, *Configuration and Capacity Planning for Solaris Servers,. Englewood Cliffs, NJ (Sun/Prentice-Hall), 1997, ISBN 0-13-349952-9.*

wistych modelach jest *The Art of Computer Systems Performance Analysis* autorstwa Raja Jaina*). Po tym ostrzeżeniu przedstawiamy system z rysunku B.2.



Rysunek B.2. Przepływ danych w serwerze Samby i potencjalne wąskie gardła

Przepływ danych jest dość oczywisty. Na przykład podczas odczytu dane płyną od dysku, kolejno przez magistralę, procesor (lub obok niego), wreszcie do karty sieciowej. Następnie są dzielone na pakiety i przesyłane przez sieć. Prześledzimy drogę danych przez system i sprawdzimy, w jakich wąskich gardłach mogą utknąć. Choć trudno w to uwierzyć, dość łatwo jest sporządzić zbiór tabel z parametrami popularnych dysków, procesorów i kart sieciowych, więc tak właśnie postąpimy.

Weźmy konkretny przykład: linuksowy komputer Pentium 133 MHz z jednym dyskiem danych o prędkości obrotowej 7200 RPM, magistralą PCI i kartą Ethernetu 10 Mb/s. Jest to całkiem przyzwoity serwer. Zacznijmy od tabeli B.2, która opisuje dysk twardy – pierwsze potencjalne wąskie gardło systemu.

Tabela B.2. Przepustowość dysku

RPM	*Operacje we-wy na sekundę*	*KB na sekundę*
7200	70	560
4800	60	480
3600	40	320

Przepustowość dysku to liczba kilobajtów przesyłanych przez dysk w ciągu jednej sekundy. Można obliczyć ją na podstawie liczby operacji wejścia-wyjścia na 8-kilobajtowych blokach danych, którą dysk może przeprowadzić w ciągu sekundy, a na którą zasadniczy wpływ ma prędkość obrotowa dysku i gęstość zapisu. Należy zatem postawić pytanie: ile danych może przejść pod głowicami dysku w czasie jednej sekundy? W przypadku pojedynczego dysku 7200 RPM przykładowy serwer mógłby wykonać 70 operacji wejścia-wyjścia na sekundę, dając przepustowość około 560 KB/s.

* Patrz Raj Jain, *The Art of Computer Systems Performance Analysis*. New York, NY (John Wiley and Sons), 1991, ISBN 0-47-150336-3.

Drugim potencjalnym wąskim gardłem jest procesor. W nowoczesnych komputerach dane w rzeczywistości nie przepływają przez procesor, więc przepustowość będziemy musieli obliczyć w sposób pośredni.

Procesor musi wydawać polecenia wejścia-wyjścia i obsługiwać nadchodzące przerwania, a następnie przesyłać dane przez magistralę do karty sieciowej. W wielu eksperymentach stwierdzono, że największą część czasu procesora pochłania przetwarzanie kodu systemu plików, więc możemy zignorować inne działające oprogramowanie. Obliczymy przepustowość procesora, po prostu mnożąc (zmierzoną) liczbę operacji wejścia-wyjścia na sekundę, które potrafi przetworzyć procesor, przez ten sam średni rozmiar żądania (8 KB). Wyniki zamieszczono w tabeli B.3.

Tabela B.3. Przepustowość procesora

Procesor	*Operacje we-wy na sekundę*	*KB na sekundę*
Intel Pentium 133	700	5600
Dwa Pentium 133	1200	9600
Sun SPARC II	660	5280
Sun SPARC 10	750	6000
Sun Ultra 200	2650	21200

Porównajmy teraz przepustowość dysku i procesora: w przykładowym komputerze mamy pojedynczy dysk 7200 RPM, który zapewnia przepustowość 560 KB/s, oraz procesor zdolny do zapoczątkowania 700 operacji wejścia-wyjścia na sekundę, co daje przepustowość 5600 KB/s. Jak można było się spodziewać, wąskim gardłem najwyraźniej jest dysk twardy.

Ostatnie potencjalne wąskie gardło to sieć. Jeśli prędkość sieci jest mniejsza niż 100 Mb/s, właśnie ona może stać się wąskim gardłem. Powyżej tej granicy spowolni nas raczej konstrukcja karty sieciowej. Tabela B.4 pokazuje średnią przepustowość różnych typów sieci. Choć prędkość sieci konwencjonalnie mierzy się w bitach na sekundę, w tabeli B.4 podajemy ją w kilobajtach na sekundę, aby ułatwić porównanie z przepustowością dysku i procesora (tabele B.2 i B.3).

Tabela B.4. Przepustowość sieci

Typ sieci	*KB na sekundę*
ISDN	16
T1	197
Ethernet 10m	1113
Token ring	1500
FDDI	6250
Ethernet 100m	6500[a]
ATM 155	7125[a]

[a]Te liczby niebawem wzrosną. Komputery Cray, Sun Ultra i DEC/Compaq Alpha osiągnęły już lepsze wyniki.

W omawianym przykładzie wąskim gardłem jest dysk, który ogranicza przepustowość do 560 KB/s. W tabeli B.4 widać, że standardowy Ethernet 10 Mb/s (1113 KB/s) jest znacznie szybszy niż dysk, zatem winą za ograniczenie wydajności należy obciążyć dysk (jest to zresztą bardzo częsty przypadek). Wystarczy spojrzeć na tabele, aby przewidzieć, że małe serwery nie będą miały problemów z procesorem, a duże serwery z wieloma procesorami będą musiały uciekać się do macierzy paskowych i dodatkowych kart sieciowych, zanim zabraknie im mocy procesora. I tak właśnie się dzieje.

Przykłady praktyczne

Przykład z książki Wonga *Configuration and Capacity Planning for Solaris Servers* pokazuje, że dwuprocesorowy komputer SPARCstation 20/712 z czterema kartami Ethernetu i sześcioma dyskami 2,1 GB będzie spędzał cały czas na czekaniu, aż dyski dostarczą następną porcję danych. Jeśli nawet zapełniono by go dyskami (Wong sugeruje użycie aż 34), jego przepustowość nadal nie przekroczyłaby granicy 1200 KB/s ze względu na karty Ethernetu. Aby osiągnąć pełną wydajność komputera, należałoby użyć wielu kart Ethernetu, Fast Ethernetu (100 Mb/s) lub FDDI.

Ciąg założeń prowadzących do tego wniosku jest przedstawiony w tabeli B.5.

Tabela B.5. Optymalizowanie serwera średniej klasy

Komputer	*Przepustowość jednego procesora*	*Przepustowość dysku*	*Przepustowość sieci*	*Rzeczywista przepustowość*
Dwuprocesorowy SPARC 10, 1 dysk	560	6000	1113	560
Dodatkowe 5 dysków	3360	6000	1113	1113
Dodatkowe 3 karty Ethernetu	3360	6000	4452	3360
Zastosowanie macierzy 20 dysków	11200	6000	4452	4452
Użycie dwóch kart Ethernetu 100 Mb/s	11200	6000	13000	11200

Początkowo wąskim gardłem jest twardy dysk o przepustowości zaledwie 560 KB/s. Rozwiązaniem jest dodanie kolejnych pięciu dysków. Daje to większą przepustowość dysków niż Ethernetu, więc problemem staje się Ethernet. Kontynuując rozbudowę serwera napotykamy problem to z tej, to z tamtej strony. Wąskie gardło przemieszcza się wraz z dodawaniem dysków, procesorów i kart sieciowych. Zasadniczo, nasza strategia polega na dodawaniu sprzętu w celu wyeliminowania kolejnych wąskich gardeł, aż do momentu, kiedy osiągniemy docelową wydajność albo (niestety) nie będziemy mogli dodać więcej elementów czy też skończą się nam fundusze.

Z naszych doświadczeń wynika, że powyższe obliczenia są prawdziwe. Duży dwuprocesorowy serwer plikowy SPARC 10 administrowany przez jednego z autorów

potrafił nasycić danymi Ethernet i około jednej trzeciej pierścienia FDDI. Działał niemal równie wydajnie z jednym procesorem, choć po zastosowaniu szybkiego systemu operacyjnego i dokonaniu gruntownej optymalizacji.

Opisaną powyżej procedurę można zastosować także do innych modeli specjalnie zaprojektowanych serwerów. Te same reguły odnoszą się do komputerów DECstation 2100 oraz najnowszych komputerów Alphy i Compaqa, starszych MIPS 3350 i nowych SGI O2. Ogólnie rzecz biorąc, komputery wieloprocesorowe dysponują taką przepustowością magistrali i mocą procesora, że przy świadczeniu usług plikowych niezawodnie „zatkają" się na dyskowych operacjach wejścia-wyjścia. I całe szczęście, zważywszy na koszty!

Ile klientów może obsłużyć Samba?

Zależy to głównie od ilości danych przetwarzanych przez każdego użytkownika. Mały serwer z trzema dyskami SCSI-1, który może przesyłać około 960 KB danych na sekundę, obsłuży od 36 do 80 klientów w zwykłym środowisku biurowym, gdzie użytkownicy typowo odczytują lub zapisują arkusze kalkulacyjne lub dokumenty tekstowe o podobnych rozmiarach (36 klientów x 2,3 transferów na sekundę x 12 KB danych = 1 MB/s).

Jeśli ten sam serwer jest używany w środowisku programistycznym, gdzie użytkownicy pracują w intensywnym cyklu edycja-kompilacja-testowanie, często pojawiają się żądania rzędu 1 MB/s, co ogranicza serwer do 25 lub mniej klientów. Kontynuując to rozumowanie, system obróbki obrazu, w którym każdy klient wymaga przepustowości 10 MB/s, będzie działał kiepsko, choćby serwer był bardzo wydajny, jeśli wszystkie klienty będą znajdować się w tym samym segmencie Ethernetu. I tak dalej.

Jeśli nie wiesz, ile danych przetwarza statystyczny użytkownik, możesz wyskalować swój serwer na podstawie istniejących konfiguracji serwerów NFS, Netware lub LAN Managera. Powinieneś upewnić się, że nowe serwery mają tyle samo dysków i kontrolerów co te, które skopiowałeś. Technikę tę słusznie określa się terminem „strzel i módl się".

Jeśli wiesz, ile klientów może obsłużyć istniejący serwer, jesteś w znacznie lepszej sytuacji. Możesz ustalić maksymalną wydajność serwera i na tej podstawie oszacować, ile danych przetwarzają klienty. Jeśli na przykład udostępnianie katalogów macierzystych w serwerze z dwoma dyskami IDE jest zbyt wolne przy 30 klientach, a zadowalające przy 25, możesz śmiało założyć, że wąskim gardłem są ethernetowe (około 375 KB), a nie dyskowe operacje wejścia-wyjścia (do 640 KB). W takim przypadku łatwo dojść do wniosku, że klienty średnio wymagają przepustowości 15 KB/s (375/25).

Obsługa nowego laboratorium komputerowego z 75 klientami będzie wymagać przepustowości 1125 KB/s, rozłożonej na kilka (najlepiej trzy) sieci Ethernet, oraz serwera z przynajmniej trzema dyskami 7200 RPM i odpowiednio szybkim procesorem. Te wymagania spełnia Pentium 133 lub szybszy komputer z taką architekturą

magistrali, która pozwala na pracę wszystkich elementów z pełną prędkością (na przykład PCI).

Zbudowany na zamówienie serwer PC albo wieloprocesorowa stacja robocza, taka jak Sun Sparc, DEC/Compaq Alpha, SGI lub podobna, będzie łatwiej się skalować, podobnie jak konfiguracja z Fast Ethernetem i koncentratorem przełączającym, który będzie sterował komputerami klienckimi w poszczególnych Ethernetach o prędkości 10 MB/s.

Zgadujemy

Jeśli zupełnie nie masz pojęcia, jakiej wydajności potrzebujesz, spróbuj odgadnąć ją na podstawie doświadczeń innych użytkowników. Każdy komputer kliencki może potrzebować od 1 operacji wejścia-wyjścia na sekundę (zwykły PC lub Mac używany w dziale sprzedaży lub księgowości) do 4 (szybkie stacje robocze używające dużych aplikacji). Szybka stacja robocza z uruchomionym kompilatorem może średnio żądać transmisji od 3 do 4 MB/s, a system obróbki obrazu nawet więcej.

Jakie są nasze zalecenia? Podpatrz kogoś, kto ma podobną konfigurację, i spróbuj ocenić zapotrzebowanie na przepustowość na podstawie jej wąskich gardeł i głośności krzyków jej użytkowników. Polecamy także książkę Briana Wonga *Configuration and Capacity Planning for Solaris Servers*. Choć w jego przykładach występują głównie serwery Sun Solaris, wąskimi gardłami są tu przede wszystkim dyski i karty sieciowe, podobnie jak u innych czołowych producentów. Jego tabele dla serwerów FTP odpowiadają dość ściśle naszym pomiarom serwerów Samby i są dobrym punktem wyjścia do dalszych badań.

Formularze pomiarowe

Tabele B.6 i B.7 to puste formularze, których możesz użyć do zapisywania własnych danych. Możesz wyznaczyć wąskie gardło za pomocą arkusza kalkulacyjnego albo ręcznie, za pomocą tabeli B.8. Jeśli Samba działa tak dobrze lub lepiej niż FTP, a w niektórych przebiegach testowych nie występują znaczące odchylenia od średniej, oznacza to, że twój system jest dobrze skonfigurowany. Jeśli interfejs pętli zwrotnej nie jest dużo szybszy od innych sposobów komunikacji, oznacza to, że masz problem z oprogramowaniem TCP/IP. Jeśli i FTP, i Samba są powolne, prawdopodobnie masz problem z siecią: może to być spowodowane błędnie działającą kartą sieciową, a także przypadkowym ustawieniem karty Ethernetu na tryb półdupleksowy, gdy nie jest ona podłączona do koncentratora półdupleksowego. Pamiętaj, że prędkości procesora i dysku mierzy się zwykle w bajtach na sekundę, a prędkość sieci w bitach.

W tabelach umieściliśmy kolumny zarówno dla bajtów, jak i bitów. W ostatniej kolumnie porównujemy wyniki z wartością 10 Mb/s, ponieważ jest to prędkość tradycyjnego Ethernetu.

Tabela B.6. Interfejs ethernetowy do tego samego hosta: FTP

Test nr	*Rozmiar w bajtach*	*Czas (s)*	*Bajty/s*	*Bity/s % 10 Mb/s*
1				
2				
3				
4				
5				
Średnia:				
Odchylenie:				

Tabela B.7. Interfejs ethernetowy do tego samego hosta: FTP

Test nr	*Rozmiar w bajtach*	*Czas (s)*	*Bajty/s*	*Bity/s % 10 Mb/s*
1				
2				
3				
4				
5				
Średnia:				
Odchylenie:				

Tabela B.8. Wyznaczanie wąskiego gardła

Liczba procesorów	*Przepustowość procesorów*	*Liczba dysków*	*Przepustowość dysku*	*Liczba sieci*	*Przepustowość sieci*	*Przepustowość całkowita*

W tabeli B.8:

- Przepustowość procesora: (KB/s z tabeli B.3) x (liczba procesorów)
- Przepustowość dysku: (KB/s z tabeli B.2) x (liczba dysków)
- Przepustowość sieci: (KB/s z tabeli B.4) x (liczba sieci)
- Przepustowość całkowita: minimum (przepustowość dysku, procesora i sieci).

Typowy test, w tym przypadku dla polecenia `get` FTP, wygląda tak jak w tabeli B.9:

Tabela B.9. Interfejs ethernetowy do tego samego hosta: FTP

Test nr	*Rozmiar w bajtach*	*Czas (sek.)*	*Bajty/sek*	*Bity/sek*	*% 10 Mb/s*
1	1 812 898	2,3	761 580		
2		2,3	767 820		
3		2,4	747 420		
4		2,3	760 020		
5		2,3	772 700		
Średnia:		2,32	777 310	6 218 480	62
Odchylenie:		0,04			

Przykładowy test serwera Sparc, o którym była mowa wcześniej, wyglądałby tak jak w tabeli B.10.

Tabela B.10. Przykładowy serwer Sparc 20

Liczba procesorów	*Przepustowość procesora*	*Liczba dysków*	*Przepustowość dysku*	*Liczba sieci*	*Przepustowość sieci*	*Przepustowość całkowita*
2	6000	1	560	1 10base2	1113	560
2	6000	6	3360	1	1113	1 113
2	6000	6	3360	4 10base2	4452	3 360
2	6000	20	11 200	4	4452	4 452
2	6000	20	11 200	2 100base2	13 000	11 200

C

Spis opcji konfiguracyjnych Samby

Poniżej znajduje się pełny spis opcji konfiguracyjnych Samby. Jeśli dana opcja może być używana tylko w sekcji globalnej, przed jej nazwą występuje słowo „[global]". We wszystkich wzmiankowanych listach poszczególne elementy należy oddzielać spacjami, chyba że podano inaczej. Na końcu dodatku znajduje się słowniczek terminów używanych w spisie opcji.

admin users = lista użytkowników

dozwolone wartości: lista użytkowników *wartość domyślna*: BRAK

Lista użytkowników, którzy będą mieli przywileje roota podczas korzystania z udziałów Samby.

allow hosts = lista hostów

dozwolone wartości: dowolne *wartość domyślna*: BRAK

Synonim opcji `hosts allow`. Wymienia komputery, które mogą łączyć się z udziałem.

alternate permissions = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Przestarzała. W Sambie 2.0 nie ma żadnego zastosowania. Pliki będą pokazywane jako przeznaczone tylko do odczytu, jeśli właściciel nie może w nich pisać. W Sambie 1.9 włączenie tej opcji powoduje, że każdy plik, którego użytkownik nie mógł zapisać, był oznaczany dosowym atrybutem „tylko do odczytu". To z kolei wymagało włączenia opcji `delete readonly`.

[global] announce as = typ systemu

dozwolone wartości: NT, Win95, WfW *wartość domyślna*: NT

Samba może ogłaszać się jako system inny niż Windows NT. Użycie tej opcji nie jest zalecane, gdyż przeszkadza ona w udostępnianiu list przeglądania.

[global] announce version = liczba.liczba

dozwolone wartości: dowolne *wartość domyślna*: 4.2

Instruuje Sambę, aby ogłaszała się jako starsza wersja serwera SMB. Użycie tej opcji nie jest zalecane.

[global] auto services = lista udziałów

dozwolone wartości: dowolne udziały *wartość domyślna*: BRAK

Lista udziałów, które zawsze będą pojawiać się na listach przeglądania. Synonim opcji `preload`.

available = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na NO, zabrania dostępu do udziału. Nie ma wpływu na przeglądanie.

[global] bind interfaces only = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, udziały i listy przeglądania będą udostępniane tylko przez interfejsy wymienione na liście interfejsów (patrz opcja `interfaces`). Nowość w Sambie 1.9.18. Jeśli ustawisz ją na YES, dodaj 127.0.0.1 do listy interfejsów, aby polecenie *smbpasswd* mogło łączyć się z lokalnym komputerem w celu zmiany hasła. Opcja ta służy tylko wygodzie użytkownika; nie poprawia bezpieczeństwa.

browsable = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Pozwala na ogłaszanie udziałów za pośrednictwem list przeglądania.

blocking locks = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, Samba uwzględnia żądania blokady zakresu bajtów z limitem czasowym, podczas którego żądanie jest kolejkowane i podejmowane są kolejne próby przyznania blokady. Nowość w Sambie 2.0.

[global] browse list = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Włącza lub wyłącza przekazywanie list przeglądania przez serwer. Nie zmieniać.

[global] case sensitive = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba odnajduje pliki o nazwach pisanych literami takiej wielkości, jak zrobił to klient. Jeśli jest ustawiona na NO, dopasowywane są nazwy pisane zarówno dużymi, jak i małymi literami. Nie należy zmieniać tej opcji.

[global] case sig names = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Synonim opcji `case sensitive`.

[global] change notify timeout = liczba

dozwolone wartości: liczba dodatnia *wartość domyślna*: 60

Ustawia liczbę sekund między kolejnymi testami, kiedy klient zażąda powiadamiania o zmianach w katalogu. Wprowadzona w Sambie 2.0, aby zmniejszyć obciążenie testami. Nie należy obniżać jej wartości.

character set = nazwa

dozwolone wartości: ISO8859-1, ISO8859-2, ISO8859-5, KOI8-R *wartość domyślna*: BRAK

Jeśli jest włączona, tłumaczy znaki ze stron kodowych DOS-a na zestawy znaków: zachodnioeuropejski (ISO8859-1), wschodnioeuropejski (ISO8859-2), rosyjską cyrylicę (ISO8859-5) lub alternatywny rosyjski (KOI8-R). Opcja `client code page` musi być ustawiona na 850.

client code page = nazwa

dozwolone wartości: patrz tabela 8.4. *wartość domyślna*: 437 (MS-DOS Stany Zjednoczone)

Ustawia jawnie stronę kodową DOS-a, anulując poprzednie ustawienia opcji `valid chars`. Przykładowe wartości to 850 dla europejskiej strony kodowej, 437 dla standardowej strony kodowej Stanów Zjednoczonych i 932 dla japońskiej strony Shift-JIS. Wprowadzona w Sambie 1.9.19.

coding system = kod

dozwolone wartości: euc, cap, hex, hexN, sjis, j8bb, j8bj, jis8, j8bh, j8@b, j8@j, j8@h, j7bb, j7bj, jis7, j7bh, j7@b, j7@j, j7@h, jubb, jubj, junet, jubh, ju@b, ju@j, ju@h

wartość domyślna: BRAK

Ustawia system kodowania, przede wszystkim dla alfabetu Kanji. Służy do określania nazw plików i powinna odpowiadać używanej stronie kodowej. Opcja `client`

`code page` musi być ustawiona na 932 (japońska strona Shift-JIS). Wprowadzona w Sambie 2.0.

comment = tekst

dozwolone wartości: łańcuch tekstowy lub pusty *wartość domyślna*: BRAK

Ustawia komentarz wyświetlany obok udziału w poleceniu NET VIEW lub w szczegółowym widoku okna katalogu Windows. Patrz także opcja `server string`.

[global] config file = ścieżka i nazwa pliku

dozwolone wartości: uniksowa ścieżka i nazwa pliku *wartość domyślna*: BRAK

Określa plik konfiguracyjny Samby, który należy odczytać zamiast bieżącego. Używana w celu zmiany położenia pliku konfiguracyjnego lub w celu wybrania pliku dostosowanego do potrzeb niektórych komputerów lub użytkowników.

copy = nazwa sekcji

dozwolone wartości: nazwa istniejącej sekcji *wartość domyślna*: BRAK

Kopiuje konfigurację uprzednio napotkanego udziału w inne miejsce. Używana ze zmiennymi (`%`) w celu wybrania konfiguracji dostosowanych do potrzeb różnych komputerów, architektur i użytkowników. Kopiowana sekcja musi występować wcześniej w pliku konfiguracyjnym. Opcje kopiowane mają niższy priorytet niż te, które jawnie podano w docelowej sekcji.

create mask = liczba ósemkowa

dozwolone wartości: bity praw dostępu w postaci ósemkowej, od 0 do 0777 *wartość domyślna*: 0744

Nazywana również `create mode`. Określa najwyższe dopuszczalne prawa dostępu do nowo tworzonych plików (na przykład 0755). Patrz także `directory mask`. Jeśli chcesz ustawiać konkretne prawa dostępu, użyj opcji `force create mask` i `force directory mask`. Opcja ta nie ma wpływu na katalogi od wersji 1.9.17 Samby, a w wersji 2.0 zmieniono jej wartość domyślną.

create mode = liczba ósemkowa

dozwolone wartości: bity praw dostępu w postaci ósemkowej, od 0 do 0777 *wartość domyślna*: 0744

Synonim opcji `create mask`.

[global] deadtime = minuty

dozwolone wartości: minuty *wartość domyślna*: 0

Czas w minutach, po którym nieużywane połączenie zostanie zakończone. Zero oznacza brak limitu czasu. Używana po to, aby klienty niepotrzebnie nie zużywały zasobów serwera. Jeśli z niej skorzystasz, klienty będą musiały automatycznie odnawiać połączenie po określonym czasie braku aktywności. Patrz także `keepalive`.

[global] debug level = liczba

dozwolone wartości: liczba *wartość domyślna*: 0

Ustawia poziom rejestrowania. Wartości większe lub równe 3 znacznie spowalniają Sambę. Synonim opcji `log level`. Zalecana wartość: 1.

[global] debug timestamp = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Umieszcza znacznik czasowy we wszystkich komunikatach zapisywanych w dziennikach Samby. Możesz ją wyłączyć, jeśli nie jest przydatna (na przykład podczas diagnozowania usterek). Nowość w Sambie 2.0.

[global] default = name

dozwolone wartości: nazwa udziału *wartość domyślna*: BRAK

Nazywana także `default service`. Nazwa usługi (udziału), która jest udostępniana, jeśli ktoś zażąda połączenia z udziałem, do którego nie ma uprawnień, albo który nie istnieje. Od wersji 1.9.14 ścieżka jest ustawiana na podstawie nazwy określonej przez klienta, przy czym wszystkie znaki „_" są zamieniane na „/", pozwalając na dostęp do dowolnego katalogu serwera Samby. Stanowczo odradzamy używanie tej opcji.

default case = wielkość liter

dozwolone wartości: LOWER, UPPER *wartość domyślna*: LOWER

Ustawia wielkość liter w nazwach nowo tworzonych plików. LOWER oznacza mieszaną wielkość liter, UPPER oznacza same duże litery.

[global] default service = nazwa udziału

dozwolone wartości: nazwa udziału *wartość domyślna*: BRAK

Synonim opcji `default`.

delete readonly = wartość logiczna

dozwolone wartości: NO, YES *wartość domyślna*: NO

Umożliwia usuwanie plików przeznaczonych tylko do odczytu. Nie jest to dozwolone w DOS-ie ani Windows, ale normalne w Uniksie, gdzie istnieją oddzielne prawa dostępu do katalogów. Do użycia z programami typu RCS lub w połączeniu ze starszą opcją `alternate permissions`.

delete veto files = wartość logiczna

dozwolone wartości: NO, YES *wartość domyślna*: NO

Pozwala na usuwanie katalogów zawierających pliki lub podkatalogi, których użytkownik nie widzi ze względu na ustawienie opcji `veto files`. Jeśli jest ustawiona na NO, katalog nie zostanie usunięty i nadal będzie zawierał niewidoczne pliki.

deny hosts = lista hostów

dozwolone wartości: lista hostów *wartość domyślna*: BRAK

Synonim opcji `hosts deny`. Określa listę komputerów, które nie mogą nawiązywać połączeń z serwerem ani korzystać z jego udziałów.

[global] dfree command = polecenie

dozwolone wartości: polecenie powłoki *wartość domyślna*: różna

Polecenie wykonywane przez serwer, które zwraca ilość wolnej przestrzeni dyskowej. Opcja ta nie jest potrzebna, jeśli systemowe polecenie *dfree* działa poprawnie.

directory = ścieżka

dozwolone wartości: ścieżka *wartość domyślna*: BRAK

Synonim opcji `path`. Katalog udostępniany przez udział dyskowy lub wykorzystywany przez udział drukarki. W udziale `[homes]` ustawiany automatycznie na katalog macierzysty użytkownika, w innych udziałach domyślnie ustawiany na */tmp*.

directory mask = liczba ósemkowa

dozwolone wartości: bity praw dostępu w postaci ósemkowej, od 0 do 0777 *wartość domyślna*: 0755

Nazywana również `directory mode`. Określa najwyższe dopuszczalne prawa dostępu do nowo tworzonych katalogów. Jeśli chcesz ustawiać konkretne prawa dostępu, użyj opcji `force create mask` i `force directory mask`.

directory mode = liczba ósemkowa

dozwolone wartości: bity praw dostępu w postaci ósemkowej, od 0 do 0777 *wartość domyślna*: 0755

Synonim opcji `directory mask`.

[global] dns proxy = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES i opcja `wins server` jest ustawiona na `yes`, Samba wyszukuje nazwy w DNS, jeśli nie znajdzie ich w WINS.

[global] domain logons = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Pozwala klientom Windows 95/98 lub NT na logowanie się w domenie (typu NT).

[global] domain master = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Samba staje się główną przeglądarką domeny i – jeśli to możliwe – gromadzi listy przeglądania z całej grupy roboczej lub domeny.

dont descend = lista przecinkowa

dozwolone wartości: lista ścieżek oddzielonych przecinkami *wartość domyślna*: BRAK

Nie zezwala na przejście do określonych katalogów ani na przeglądanie ich zawartości. Opcja ta ułatwia kontrolowanie przeglądania, ale nie zwiększa bezpieczeństwa.

dos filetimes = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Pozwala użytkownikowi nie będącemu właścicielem pliku na zmianę czasu modyfikacji pliku, jeśli ma prawo do zapisu w tym pliku. Patrz także opcja `dos filetime resolution`.

dos filetime resolution = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Ustawia czasy modyfikacji plików w Uniksie zgodnie ze standardami dosowymi (zaokrąglone do następnej parzystej sekundy). Zalecana w razie korzystania z Visu-

al C++ lub dosowego programu *make* w celu uniknięcia niepotrzebnej rekompilacji programów. Używana w połączeniu z opcją `dos filetimes`.

[global] encrypt passwords = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Włącza szyfrowanie haseł metodą Windows NT. Wymaga użycia programu *smbpasswd* w serwerze Samby.

exec = polecenie

dozwolone wartości: polecenie powłoki *wartość domyślna*: BRAK

Synonim opcji `preexec`. Polecenie, które należy wykonać tuż przed połączeniem użytkownika z udziałem.

fake directory create times = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Obejście usterki w programie *nmake* Microsoftu. Jeśli jest ustawiona na YES, Samba będzie ustawiać takie czasy utworzenia katalogów, aby program *nmake* nie rekompilował za każdym razem wszystkich plików.

fake oplocks = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli klient zapyta, czy może zablokować plik i buforować go lokalnie, Samba zawsze wyrazi zgodę, ale w rzeczywistości nie założy blokady na plik. Obecnie stosowana dla dysków przeznaczonych tylko do odczytu, ponieważ Samba potrafi już obsługiwać prawdziwe blokady oportunistyczne. Patrz także opcje `oplocks` i `veto oplock files`.

follow symlinks = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, Samba będzie podążała za dowiązaniami symbolicznymi w udziałach plikowych. Jeśli chcesz ograniczyć dowiązania symboliczne tylko do bieżącego udziału dyskowego, użyj opcji `wide links`.

force create mask = liczba ósemkowa

dozwolone wartości: liczba ósemkowa od 0 do 0777 *wartość domyślna*: 0

Określa bity, które będą sumowane logicznie z prawami dostępu do nowo tworzonych plików. Używana w połączeniu z opcją `create mode`.

force create mode = liczba ósemkowa

dozwolone wartości: liczba ósemkowa od 0 do 0777 *wartość domyślna*: 0

Synonim opcji `force create mask`.

force directory mask = liczba ósemkowa

dozwolone wartości: liczba ósemkowa od 0 do 0777 *wartość domyślna*: 0

Określa bity, które będą sumowane logicznie z prawami dostępu do nowo tworzonych katalogów, powodując ustawienie odpowiednich bitów. Używana w połączeniu z opcją `directory mode`.

force directory mode = liczba ósemkowa

dozwolone wartości: liczba ósemkowa od 0 do 0777 *wartość domyślna*: 0

Synonim opcji `force directory mask`.

force group = grupa uniksowa

dozwolone wartości: grupa *wartość domyślna*: BRAK

Ustawia obowiązującą nazwę grupy dla użytkowników korzystających z udziału. Zastępuje zwykłą grupę użytkownika.

force user = nazwa

dozwolone wartości: nazwa użytkownika *wartość domyślna*: BRAK

Ustawia obowiązującą nazwę użytkownika korzystającego z udziału. Nie zalecana.

fstype = łańcuch

dozwolone wartości: NTFS, FAT, Samba *wartość domyślna*: NTFS

Ustawia typ systemu plików zgłaszany klientom.

[global] getwd cache = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Buforuje bieżący katalog w celu zwiększenia wydajności. Zalecana w razie korzystania z opcji `wide links`.

group = grupa

dozwolone wartości: grupa uniksowa *wartość domyślna*: BRAK

Przestarzała forma opcji `force group`.

guest account = użytkownik

dozwolone wartości: nazwa użytkownika *wartość domyślna*: BRAK

Określa nazwę nieuprzywilejowanego konta uniksowego, które będzie wykorzystywane podczas drukowania i udostępniania udziałów oznaczonych opcją `guest ok`.

guest ok = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, dostęp do tego udziału nie wymaga podania hasła. Synonim opcji `public`.

guest only = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Użytkownik musi korzystać z udziału jako gość. Wymaga ustawienia na `yes` opcji `guest ok` lub `public`.

hide dot files = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Traktuje wszystkie pliki zaczynające się od kropki tak, jakby miały ustawiony dosowy atrybut ukrycia.

hide files = lista ukośnikowa

dozwolone wartości: lista wzorców oddzielonych znakami „/" *wartość domyślna*: BRAK

Lista nazw plików i katalogów, dla których należy ustawiać dosowy atrybut ukrycia. Nazwy mogą zawierać symbole wieloznaczne ? lub * oraz zmienne %. Patrz także opcje `hide dot files` i `veto files`.

[global] homedir map = nazwa mapy NIS

dozwolone wartości: nazwa mapy NIS *wartość domyślna*: auto.home

Używana w połączeniu z opcją `nis homedir` w celu zlokalizowania macierzystego katalogu użytkownika w usłudze Sun NIS (nie NIS+).

hosts allow = lista hostów

dozwolone wartości: lista nazw hostów *wartość domyślna*: BRAK

Synonim opcji `allow hosts`. Lista komputerów, które mają dostęp do udziału lub udziałów. Jeśli jest pusta (tak jest domyślnie), z udziału może korzystać każdy komputer, który nie widnieje na liście `hosts deny`.

hosts deny = lista hostów

dozwolone wartości: lista nazw hostów *wartość domyślna*: BRAK

Synonim opcji `deny hosts`. Lista komputerów, które nie mogą łączyć się z udziałem lub udziałami.

[global] hosts equiv = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: BRAK

Ścieżka do pliku zawierającego nazwy zaufanych komputerów, które mogą logować się bez podawania hasła. Stanowczo odradzamy jej użycie, ponieważ użytkownicy Windows zawsze mogą zmienić nazwę swojego komputera, a w takiej konfiguracji jest to jedyne zabezpieczenie.

include = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: BRAK

Dołącza wskazany plik do pliku *smb.conf* w miejscu, w którym wpisano tę opcję. Opcja ta nie rozpoznaje zmiennych `%u` (użytkownik), `%P` (katalog główny bieżącego udziału) ani `%S` (nazwa bieżącego udziału), ponieważ w momencie wczytywania pliku zmienne te nie są jeszcze ustawione.

[global] interfaces = lista interfejsów

dozwolone wartości: adresy IP oddzielone spacjami *wartość domyślna*: BRAK

Ustawia interfejsy, przez które Samba będzie odpowiadać na żądania. Domyślnie jest to tylko podstawowy interfejs komputera. Zalecamy użycie tej opcji w komputerach podłączonych do wielu sieci, ponieważ zapobiega ona błędnym przypisaniom adresów i masek sieciowych.

invalid users = lista użytkowników

dozwolone wartości: lista użytkowników *wartość domyślna*: BRAK

Lista użytkowników, którzy nie mogą korzystać z udziału lub udziałów.

[global] keepalive = liczba

dozwolone wartości: liczba sekund *wartość domyślna*: 0

Liczba sekund między kolejnymi kontrolami awarii klienta. Domyślna wartość 0 sprawia, że kontrole takie nie są przeprowadzane. Zalecamy jej użycie, jeśli chcesz przeprowadzać kontrolę częściej niż co 4 godziny. Rozsądną wartością jest 3600 (10 minut). Można też kontrolować klienty w inny sposób – patrz opcja `socket options`.

[global] kernel oplocks = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: ustawiana automatycznie

Przerywa blokadę, kiedy uniksowy proces zamierza skorzystać z zablokowanego pliku, zapobiegając tym samym uszkodzeniu pliku. Ustawiana na YES w systemach operacyjnych, które to umożliwiają, w przeciwnym wypadku ustawiana na NO. Nowość w Sambie 2.0, obsługiwana w systemie SGI, a niebawem także w Linuksie i BSD. Nie należy jej zmieniać.

[global] ldap filters = różne wartości [global] ldap root = różne wartości [global] ldap server = różne wartości [global] ldap suffix = różne wartości

dozwolone wartości: różne *wartość domyślna*: różna

Opcje zaczynające się od słowa `ldap` sterują użyciem rozproszonego katalogu/bazy danych Lightweight Directory Access Protocol (LDAP), zawierającego informacje o użytkownikach i nazwach hostów. Opcje te w Sambie 2.0 są eksperymentalne i zarezerwowane do przyszłego użycia.

[global] load printers = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna* = YES

Umieszcza na liście przeglądania wszystkie drukarki wymienione w systemowym pliku parametrów drukarek. Używa opcji konfiguracyjnych z sekcji `[printers]`.

[global] local master = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Staje do wyborów na główną przeglądarkę lokalną. Patrz także `domain master` i `os level`.

[global] lm announce = wartość

dozwolone wartości: AUTO, YES, NO *wartość domyślna*: AUTO

Generuje rozgłoszenia SMB typu OS/2 z częstotliwością określoną opcją `lm interval`. Wartość YES lub NO włącza lub wyłącza je bezwarunkowo. Opcja AUTO sprawia, że serwer Samby czeka na ogłoszenie LAN Managera od innego komputera, zanim sam je wyśle. Opcja ta jest niezbędna, aby umożliwić przeglądanie klientom OS/2.

[global] lm interval = sekundy

dozwolone wartości: liczba *wartość domyślna*: 60

Ustawia okres (w sekundach) między ogłoszeniami SMB używanymi w OS/2.

[global] lock directory = ścieżka

dozwolone wartości: ścieżka *wartość domyślna*: /usr/local/samba/var/locks

Określa katalog, w którym będą przechowywane pliki blokad. Katalog ten musi umożliwiać zapis Sambie, a odczyt wszystkim użytkownikom.

locking = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Stosuje blokady plików. Jeśli jest ustawiona na NO, Samba będzie akceptowała żądania przyznania blokady, ale w rzeczywistości nie będzie blokowała zasobów, co zaleca się tylko dla systemów plików przeznaczonych tylko do odczytu.

[global] log file = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: różna

Określa nazwę i położenie pliku dziennika. Rozpoznaje wszystkie zmienne %.

[global] log level = liczba

dozwolone wartości: liczba *wartość domyślna*: 0

Synonim opcji `debug level`. Określa poziom rejestrowania. Wartości 3 i większe zauważalnie spowalniają system.

[global] logon drive = stacja

dozwolone wartości: dosowa nazwa stacji *wartość domyślna*: BRAK

Ustawia nazwę stacji, w której przechowywany jest katalog logowania (tylko dla klientów Windows NT).

[global] logon home = ścieżka

dozwolone wartości: uniksowa ścieżka *wartość domyślna*: `\\%N\%U`

Określa katalog macierzysty użytkownika Windows 95/98 lub NT Workstation. Pozwala na użycie polecenia `NET USE H: /HOME` z linii zgłoszenia.

[global] logon path = ścieżka

dozwolone wartości: ścieżka Windows *wartość domyślna*: `\\%N\%U\profile`

Określa ścieżkę do katalogu profili Windows. Katalog ten zawiera pliki *USER.MAN* i *USER.DAT* oraz foldery pulpitu Windows 95, menu Start, Otoczenia sieciowego i programów.

[global] logon script = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: BRAK

Ustawia ścieżkę względem udziału `[netlogon]` do skryptu DOS-a/Windows NT, który jest wykonywany w kliencie podczas logowania. Rozpoznaje zmienne %.

lppause command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie uniksowe wraz z pełną ścieżką *wartość domyślna*: różna

Określa polecenie, które wstrzymuje zlecenie wydruku. Rozpoznaje zmienne `%p` (nazwa drukarki) oraz `%j` (numer zadania).

lpresume command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie uniksowe wraz z pełną ścieżką *wartość domyślna*: różna

Określa polecenie, które wznawia zlecenie wydruku. Rozpoznaje zmienne `%p` (nazwa drukarki) oraz `%j` (numer zadania).

[global] lpq cache time = sekundy

dozwolone wartości: liczba sekund *wartość domyślna*: 10

Określa, jak długo należy buforować status kolejki wydruku uzyskany za pomocą polecenia `lpq`.

lpq command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie uniksowe wraz z pełną ścieżką *wartość domyślna*: różna

Określa polecenie, które zwraca status drukarki. Zwykle jest inicjowana na wartość domyślną przez opcję `printing`. Rozpoznaje zmienne `%p` (nazwa drukarki).

lprm command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie uniksowe wraz z pełną ścieżką *wartość domyślna*: różna

Określa polecenie, które usuwa zadanie wydruku. Zwykle jest inicjowana na wartość domyślną przez opcję `printing`. Rozpoznaje zmienne `%p` (nazwa drukarki) oraz `%j` (numer zadania).

machine password timeout = sekundy

dozwolone wartości: liczba sekund *wartość domyślna*: 604 800

Ustawia okres między zmianami hasła komputera w domenie Windows NT. Wartość domyślna to 1 tydzień, czyli 604 800 sekund.

magic output = ścieżka do pliku

dozwolone wartości: uniksowa ścieżka do pliku *wartość domyślna*: skrypt.out

Określa plik wyjściowy dla niezalecanej opcji `magic scripts`. Domyślnie jest to nazwa skryptu z rozszerzeniem *.out*.

magic script = ścieżka do pliku

dozwolone wartości: uniksowa ścieżka do pliku *wartość domyślna*: BRAK

Ustawia nazwę pliku, który zostanie wykonany przez powłokę po zamknięciu pliku przez klienta, dzięki czemu klienty mogą uruchamiać polecenia w serwerze.

mangle case = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Przekształca nazwę pliku, jeśli występują w niej litery różnej wielkości.

mangled map = lista odwzorowań

dozwolone wartości: lista par „z nazwy – na nazwę" *wartość domyślna*: BRAK

Określa nazwy, które powinny być odwzorowywane (na przykład *.html* na *.htm*).

mangled names = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Nakazuje Sambie przekształcać nazwy zbyt długie lub zawierające niedozwolone znaki na dosowy format 8.3.

mangling char = znak

dozwolone wartości: znak *wartość domyślna*: ~

Ustawia znak, który Samba wykorzystuje w przekształcanych nazwach.

[global] mangled stack = liczba

dozwolone wartości: liczba *wartość domyślna*: 50

Określa rozmiar bufora niedawno przekształconych nazw plików.

map aliasname = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: BRAK

Wskazuje plik, który zawiera pary „grupa uniksowa – grupa NT", po jednej na jedną linię. Plik ten odwzorowuje aliasy Windows NT na uniksowe nazwy grup. Patrz także opcje `username map` i `map groupname`. Wprowadzona w Sambie 2.0.

map archive = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, Samba ustawia bit wykonywalności dla właściciela (0100) w uniksowym pliku, jeśli ustawiony jest dosowy bit „archiwalny". Zalecamy użycie tej opcji; jeśli z niej korzystasz, maska `create mask` musi zawierać bit 0100.

map hidden = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba ustawia bit wykonywalności dla pozostałych użytkowników (0001) w uniksowym pliku, jeśli ustawiony jest dosowy bit „ukryty". Jeśli korzystasz z tej opcji, maska `create mask` musi zawierać bit 0001.

map groupname = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: BRAK

Wskazuje plik, który zawiera pary „grupa uniksowa – grupa NT", po jednej na jedną linię. Plik ten odwzorowuje nazwy grup Windows NT na uniksowe nazwy grup. Patrz także opcje `username map` i `map aliasname`. Wprowadzona w Sambie 2.0.

map system = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba ustawia bit wykonywalności dla grupy (0010) w uniksowym pliku, jeśli ustawiony jest dosowy bit „systemowy". Jeśli korzystasz z tej opcji, maska `create mask` musi zawierać bit 0010.

max connections = liczba

dozwolone wartości: liczba *wartość domyślna*: 0 (nieskończenie wiele)

Określa maksymalną liczbę połączeń z udziałem dla każdego komputera klienckiego.

[global] max disk size = liczba

dozwolone wartości: rozmiar w MB *wartość domyślna*: 0 (bez zmian)

Ustawia maksymalny rozmiar dysku, jaki należy zwracać klientom. Niektóre klienty i aplikacje błędnie interpretują duże rozmiary dysku.

[global] max log size = liczba

dozwolone wartości: rozmiar w KB *wartość domyślna*: 5000

Określa próg (w kilobajtach), po przekroczeniu którego Samba utworzy nowy plik dziennika. Bieżący plik dziennika otrzyma rozszerzenie *.old*, zastępując istniejący plik o takiej nazwie.

[global] max mux = liczba

dozwolone wartości: liczba *wartość domyślna*: 50

Określa liczbę jednoczesnych operacji, które mogą wykonywać klienty Samby. Nie zmieniać.

[global] max packets = liczba

dozwolone wartości: liczba *wartość domyślna*: N/D

Synonim opcji `packet size`. Przestarzała od wersji 1.7 Samby. Zamiast niej należy użyć opcji `max xmit`.

[global] max open files = liczba

dozwolone wartości: liczba *wartość domyślna*: 10 000

Ogranicza liczbę plików, które proces Samby będzie utrzymywał w stanie otwartym. Samba pozwala na ustawienie tej wartości poniżej uniksowego limitu. Opcję tę wprowadzono w Sambie 2.0 w celu obejścia pewnego problemu. Nie należy jej zmieniać.

[global] max ttl = sekundy

dozwolone wartości: czas w sekundach *wartość domyślna*: 14 400 (4 godziny)

Ustawia czas, przez który wyszukane nazwy NetBIOS-owe są przechowywane w buforze. Nie należy zmieniać tej opcji.

[global] max wins ttl = sekundy

dozwolone wartości: czas w sekundach *wartość domyślna*: 259 200 (3 dni)

Ogranicza czas ważności nazwy NetBIOS-owej w buforze WINS procesu *nmbd*. Nie należy zmieniać tej opcji.

[global] max xmit = bajty

dozwolone wartości: rozmiar w bajtach *wartość domyślna*: 65 535

Określa maksymalny rozmiar pakietów, który będzie negocjowany przez Sambę. Jest to parametr służący do optymalizowania wydajności wolnych łączy i korygowania usterek w starszych klientach. Odradzamy używanie wartości mniejszych niż 2048.

[global] message command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki *wartość domyślna*: BRAK

Określa polecenie, które należy wykonać w serwerze po odebraniu od klienta komunikatu WinPopup. Polecenie musi kończyć się znakiem `&`, aby natychmiast zwracało sterowanie. Rozpoznaje wszystkie zmienne, z wyjątkiem `%u` (użytkownik), oraz dodatkowe zmienne: `%s` (nazwa pliku z treścią komunikatu), `%t` (komputer docelowy) oraz `%f` (nadawca komunikatu).

min print space = kilobajty

dozwolone wartości: przestrzeń w KB *wartość domyślna*: 0 (bez ograniczeń)

Określa minimalną ilość wolnej przestrzeni buforowej koniecznej do przyjęcia zlecenia wydruku.

[global] min wins ttl = sekundy

dozwolone wartości: czas w sekundach *wartość domyślna*: 21 600 (6 godzin)

Ustawia minimalny czas, przez który wyszukane nazwy NetBIOS-owe są przechowywane w buforze. Nie należy zmieniać tej opcji.

name resolve order = lista

dozwolone wartości: lista słów lmhosts, wins, hosts i bcast *wartość domyślna*: lmhosts wins hosts bcast

Określa kolejność wyszukiwania podczas prób odwzorowania nazwy na adres IP. Parametr `hosts` powoduje przeprowadzenie wyszukiwania za pomocą zwykłych mechanizmów serwera: */etc/hosts*, DNS, NIS lub ich kombinacji. Wprowadzona w wersji 1.9.18p4 Samby.

[global] netbios aliases = lista

dozwolone wartości: lista nazw NetBIOS-owych *wartość domyślna*: BRAK

Określa dodatkowe nazwy NetBIOS-owe, które serwer Samby będzie ogłaszał jako własne.

netbios name = nazwa hosta

dozwolone wartości: nazwa hosta *wartość domyślna*: różna

Określa nazwę NetBIOS-ową używaną przez serwer lub podstawową nazwę serwera, jeśli istnieją aliasy NetBIOS-owe.

[global] networkstation user login = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Gdy jest ustawiona na NO, klienty nie będą przeprowadzały pełnego logowania, jeśli opcja `security` jest ustawiona na `server`. Opcja ta (wprowadzona w Sambie 1.9.18p3) służyła do tymczasowego obejścia usterki w relacjach zaufania między domenami NT. W Sambie 1.9.18p10 wprowadzono automatyczną korekcję, więc parametr ten można usunąć.

[global] nis homedir = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, nazwa serwera z katalogiem macierzystym użytkownika zostanie wyszukana w mapie NIS określonej opcją `homedir map` i zwrócona klientowi. Klient skontaktuje się następnie z tym serwerem i spróbuje połączyć się z jego udziałem. Dzięki temu można uniknąć montowania katalogu z komputera, który w rzeczywistości nie ma potrzebnego dysku. Komputer z katalogami macierzystymi musi być serwerem SMB.

[global] nt pipe support = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Umożliwia wyłączenie obsługi potoków specyficznych dla Windows NT. Opcja ta jest wykorzystywana podczas rozwijania Samby oraz mierzenia jej wydajności i w przyszłości może zostać usunięta. Nie należy jej zmieniać.

[global] nt smb support = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, umożliwia korzystanie z komunikatów SMB specyficznych dla Windows NT. Opcja ta jest wykorzystywana podczas rozwijania Samby oraz mierzenia jej wydajności i w przyszłości może zostać usunięta. Nie należy jej zmieniać.

[global] null passwords = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, umożliwia dostęp do kont z pustymi hasłami. Stanowczo odradzamy jej użycie.

ole locking compatibility = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, zakresy blokad będą odwzorowywane tak, aby uniknąć błędów w uniksowych blokadach, kiedy Windows używa blokad powyżej 32 KB. Nie należy zmieniać tej opcji. Wprowadzona w wersji 1.9.18p10 Samby.

only guest = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Synonim opcji `guest only`. Wymusza logowanie się na koncie gościnnym.

only user = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Wymaga, aby użytkownik udziału znajdował się na liście `username =`.

oplocks = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, umożliwia lokalne buforowanie oportunistycznie zablokowanych plików. Opcja ta jest zalecana, ponieważ zwiększa wydajność o prawie 30%. Patrz także opcje `fake oplocks` i `veto oplock files`.

[global] os level = liczba

dozwolone wartości: liczba *wartość domyślna*: 0

Ustawia pierwszeństwo serwera w wyborach na główną przeglądarkę. Używana w połączeniu z opcjami `domain master` i `local master`. Możesz ustawić wartość wyższą niż poziom rywalizującego systemu operacyjnego, jeśli chcesz, żeby Samba wygrała wybory. Windows for Workgroups i Windows 95 używają wartości 1, Windows NT Workstation – 17, a Windows NT Server – 33.

[global] packet size = bajty

dozwolone wartości: liczba bajtów *wartość domyślna*: 65 535

Przestarzała. Synonim nie zalecanej opcji `max packet`. Patrz `max xmit`.

[global] passwd chat debug = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Rejestruje całą wymianę komunikatów przy zmianie hasła z poziomem rejestrowania równym 100. Używana tylko do celów diagnostycznych. Wprowadzona w Sambie 1.9.18p5.

[global] passwd chat = sekwencja poleceń

dozwolone wartości: polecenia uniksowego serwera *wartość domyślna*: wartość wkompilowana

Określa polecenia służące do zmiany hasła w serwerze. Rozpoznaje zmienne `%o` (stare hasło) oraz `%n` (nowe hasło) i pozwala na stosowanie znaków specjalnych: `\r`, `\n`, `\t` i `\s` (spacja).

[global] passwd program = program

dozwolone wartości: program w uniksowym serwerze *wartość domyślna*: BRAK

Określa polecenie służące do zmiany hasła użytkownika. Polecenie to będzie wykonane z przywilejami roota. Rozpoznaje zmienną `%u` (użytkownik).

[global] password level = liczba

dozwolone wartości: liczba *wartość domyślna*: 0

Określa liczbę dużych liter w permutacjach używanych podczas dopasowywania hasła. Używana do obsługi klientów, które ustawiają wszystkie litery hasła na jednakową wielkość, zanim prześlą je do serwera Samby. Powoduje ponawianie prób zalogowania z hasłami o różnej wielkości liter, co może doprowadzić do zablokowania konta.

[global] password server = nazwy NetBIOS-owe

dozwolone wartości: lista nazw NetBIOS-owych *wartość domyślna*: BRAK

Lista serwerów SMB, które będą weryfikować hasła. Używana z serwerem haseł NT (podstawowym lub zapasowym kontrolerem domeny) i w połączeniu z opcjami konfiguracyjnymi `security = server` lub `security = domain`. Uwaga: serwer haseł NT musi zezwalać na logowanie się serwera Samby.

panic action = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki Uniksa wraz z pełną ścieżką *wartość domyślna*: BRAK

Ustawia polecenie wykonywane podczas awarii Samby. Twórcy i testerzy Samby używają polecenia `/usr/bin/X11/xterm -display :0 -e gdb /samba/bin/smbd %d`.

path = ścieżka

dozwolone wartości: ścieżka *wartość domyślna*: różna

Katalog udostępniany przez udział dyskowy lub wykorzystywany przez udział drukarki. W udziale `[homes]` ustawiany automatycznie na katalog macierzysty użytkownika, w innych udziałach domyślnie ustawiany na */tmp*. Rozpoznaje zmienne `%u` (użytkownik) oraz `%m` (komputer).

postexec = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki Uniksa wraz z pełną ścieżką — *wartość domyślna*: BRAK

Określa polecenie, które zostanie wykonane z przywilejami użytkownika, kiedy ten odłączy się do udziału. Patrz także opcje `preexec`, `root preexec` i `root postexec`.

postscript = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Traktuje drukarkę jako postscriptową (aby uniknąć błędu Windows), wstawiając znaki `%!` w pierwszej linii wydruku. Działa tylko wtedy, gdy drukarka rzeczywiście obsługuje PostScript.

preexec = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki Uniksa wraz z pełną ścieżką — *wartość domyślna*: BRAK

Określa polecenie, które zostanie wykonane z przywilejami użytkownika, zanim ten podłączy się od udziału. Patrz także opcje `postexec`, `root preexec` i `root postexec`.

[global] preferred master = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba będzie preferowaną przeglądarką główną. W takim przypadku po włączeniu się do sieci Samba zapoczątkuje wybory przeglądarki.

preload = lista udziałów

dozwolone wartości: lista usług — *wartość domyślna*: BRAK

Synonim opcji `auto services`. Określa listę udziałów, które zawsze będą pojawiać się na listach przeglądania.

preserve case = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Jeśli jest ustawiona na YES, nazwy plików będą składać się z liter o takiej wielkości, jaką podał klient. W przeciwnym wypadku litery będą miały wielkość określoną opcją `default case`. Patrz także opcja `short preserve case`.

print command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki Uniksa wraz z pełną ścieżką — *wartość domyślna:* różna

Określa polecenie, które przesyła buforowany plik do drukarki. Zwykle inicjowana na wartość domyślną za pomocą opcji `printing`. Opcja ta rozpoznaje zmienne `%p` (nazwa drukarki), `%s` (plik buforowy) oraz `%f` (plik buforowy wraz ze ścieżką względną). Polecenie będące wartością tej opcji musi usuwać plik buforowy.

print ok = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Synonim opcji `printable`.

printable = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Określa, że dany udział jest udziałem drukarki. Wymagana we wszystkich udziałach drukarek.

[global] printcap name = ścieżka do pliku

dozwolone wartości: ścieżka do pliku — *wartość domyślna*: */etc/printcap*

Określa ścieżkę do pliku z parametrami drukarek, używanego przez udział `[printers]`. Wartość domyślna zmienia się na */etc/qconfig* w Uniksie AIX i *lpstat* w Uniksie typu System V.

printer = nazwa

dozwolone wartości: nazwa drukarki — *wartość domyślna*: `lp`

Ustawia nazwę uniksowej drukarki.

printer driver = nazwa sterownika drukarki

dozwolone wartości: nazwa sterownika drukarki w postaci używanej przez Windows — *wartość domyślna*: BRAK

Ustawia łańcuch, który jest przekazywany klientowi Windows, kiedy ten pyta, jakiego sterownika należy użyć w celu przygotowania pliku do wydruku. Wielkość liter w tym łańcuchu jest istotna.

[global] printer driver file = ścieżka do pliku

dozwolone wartości: uniksowa ścieżka do pliku — *wartość domyślna*: *samba/lib/printers.def*

Określa położenie pliku *msprint.def*, używanego przez Windows 95/98.

printer driver location = ścieżka sieciowa

dozwolone wartości: ścieżka sieciowa Windows *wartość domyślna*: `\\serwer\PRINTER$`

Określa położenie sterownika drukarki. Wartością tej opcji powinien być katalog w udziale, który przechowuje pliki sterowników drukarek.

printer name = nazwa

dozwolone wartości: nazwa drukarki *wartość domyślna*: BRAK

Synonim opcji `printer`.

printing = typ

dozwolone wartości: bsd, sysv, hpux, aix, qnx, plp, lprng *wartość domyślna*: bsd

Określa typ drukowania, którego należy użyć zamiast wkompilowanego typu. Powoduje wstępne ustawienie wartości opcji `print command`, `lpq command` i `lprm command`.

[global] protocol = protokół

dozwolone wartości: NT1, LANMAN2, LANMAN1, COREPLUS, CORE *wartość domyślna*: NT1

Ustawia wersję protokołu SMB na jedną z dopuszczalnych wartości. Stanowczo odradzamy zmienianie tej opcji. Istnieje tylko dla zapewnienia wstecznej zgodności z niektórymi starszymi klientami.

public = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, dostęp do tego udziału nie wymaga podania hasła. Synonim opcji `guest ok`.

queuepause command = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie Uniksa *wartość domyślna*: różna

Określa polecenie, które zatrzymuje przetwarzanie kolejki wydruku. Zwykle inicjowana przez opcję `printing`. Wprowadzona w Sambie 1.9.18p10.

queueresume command = /ścieżka bezwzględna/polecenie

dozwolone wartości: polecenie Uniksa *wartość domyślna*: różna

Określa polecenie, które wznawia przetwarzanie kolejki wydruku. Zwykle inicjowana przez opcję `printing`. Wprowadzona w Sambie 1.9.18p10.

read bmpx = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Przestarzała. Nie zmieniać.

read list = lista przecinkowa

dozwolone wartości: lista użytkowników *wartość domyślna*: BRAK

Określa listę użytkowników, którzy do zapisywalnego udziału mają dostęp w trybie tylko do odczytu.

read only = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Określa, że udział jest przeznaczony tylko do odczytu. Antonim opcji `writable` i `write ok`.

[global] read prediction = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Odczytuje z wyprzedzeniem dane z plików przeznaczonych tylko do odczytu. Przestarzała, usunięta z Samby 2.0.

[global] read raw = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Umożliwia wykonywanie szybkich, strumieniowych odczytów przez TCP przy wykorzystaniu buforów o rozmiarze 64 KB. Zalecana.

[global] read size = bajty

dozwolone wartości: rozmiar w bajtach *wartość domyślna*: 2048

Ustawia opcję buforowania dla serwerów, w których prędkość dysku nie odpowiada prędkości sieci. Wymaga eksperymentowania. Nie należy jej zmieniać. Nie powinna przekraczać 65 536.

[global] remote announce = lista zdalna

dozwolone wartości: lista zdalnych adresów *wartość domyślna*: BRAK

Dodaje grupy robocze do listy, na której serwer Samby będzie się ogłaszać. Definiowana w postaci par: adres IP/grupa robocza (na przykład 192.168.220.215/PROSTA), przy czym różne grupy są oddzielone spacjami. Umożliwia stosowanie ukierunkowanych rozgłoszeń. Serwer będzie się pojawiał na listach przeglądania tych grup roboczych. Nie wymaga użycia WINS.

[global] remote browse sync = lista adresów

dozwolone wartości: lista adresów IP *wartość domyślna*: BRAK

Włącza synchronizację list przeglądania z innymi lokalnymi przeglądarkami (działa tylko między serwerami Samby). Można podać adres konkretnego komputera albo adres rozgłoszeniowy (ukierunkowany, to jest ###.###.###.255). W tym drugim przypadku Samba spróbuje wyszukać główną przeglądarkę lokalną.

revalidate = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, użytkownicy muszą ponownie wprowadzać hasło nawet po pomyślnym wstępnym zalogowaniu się w udziale.

[global] root = ścieżka

dozwolone wartości: ścieżka uniksowa *wartość domyślna*: BRAK

Synonim opcji `root directory`.

[global] root dir = ścieżka

dozwolone wartości: ścieżka uniksowa *wartość domyślna*: BRAK

Synonim opcji `root directory`.

[global] root directory = ścieżka

dozwolone wartości: ścieżka uniksowa *wartość domyślna*: BRAK

Określa katalog, który za pomocą funkcji `chroot()` zostanie zmieniony w katalog główny przed uruchomieniem demonów. Uniemożliwia to jakikolwiek dostęp do katalogów spoza tego drzewa. Patrz także opcja `wide links`.

root postexec = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki Uniksa wraz z pełną ścieżką *wartość domyślna*: BRAK

Określa polecenie, które zostanie wykonane z przywilejami roota, kiedy użytkownik odłączy się od udziału. Patrz także opcje `preexec`, `postexec` i `root preexec`. Wykonywane po poleceniu `postexec`. Używać z zachowaniem niezbędnej ostrożności.

root preexec = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie powłoki Uniksa wraz z pełną ścieżką *wartość domyślna*: BRAK

Określa polecenie, które zostanie wykonane z przywilejami roota, zanim użytkownik połączy się z udziałem. Patrz także opcje `preexec`, `postexec` i `root postexec`. Wykonywane przed poleceniem `preexec`. Używać z zachowaniem niezbędnej ostrożności.

[global] security = wartość

dozwolone wartości: share, user, server, domain *wartość domyślna:* share w Sambie 1.0, user w Sambie 2.0

Określa metodę zabezpieczania zasobów. Jeśli `security = share`, usługi mają wspólne hasło, jedno dla wszystkich użytkowników. Jeśli `security = user`, użytkownicy mają (uniksowe) konta i hasła. Jeśli `security = server`, użytkownicy mają konta i hasła, a uwierzytelnia ich inny komputer. Jeśli `security = domain`, przeprowadza się pełne uwierzytelnianie domenowe. Patrz także opcje `password server` i `encrypted passwords`.

[global] server string = tekst

dozwolone wartości: łańcuch *wartość domyślna*: Samba %v (w wersji 2.0)

Ustawia nazwę, która pojawia się obok serwera na listach przeglądania. Rozpoznaje zmienną `%v` (numer wersji Samby) oraz `%h` (nazwa hosta).

set directory = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Pozwala klientom DEC Pathworks na używanie polecenia *set dir*.

[global] shared file entries = liczba

dozwolone wartości: liczba *wartość domyślna*: 113

Przestarzała, nie używać.

shared mem size = bajty

dozwolone wartości: rozmiar w bajtach *wartość domyślna*: 102 400

Jeśli Samba została skompilowana z opcją FAST_SHARE_MODES (mapowanie plików w pamięci), ustawia rozmiar pamięci dzielonej. Nie należy zmieniać tej opcji.

[global] smb passwd file = ścieżka do pliku

dozwolone wartości: uniksowa ścieżka do pliku *wartość domyślna*: */usr/local/samba/private/smbpasswd*

Zastępuje wkompilowaną ścieżkę do pliku haseł, jeśli opcja `encrypted passwords` jest ustawiona na `yes`.

[global] smbrun = /ścieżka_bezwzględna/polecenie

dozwolone wartości: polecenie smbrun *wartość domyślna*: wartość wkompilowana

Zastępuje wkompilowaną ścieżkę do pliku binarnego *smbrun*. Nie należy zmieniać tej opcji.

share modes = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, Samba obsługuje używane w Windows blokady całych plików (w trybie odmowy).

short preserve case = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, nazwy plików w formacie 8.3 będą składać się z liter o takiej wielkości, jaką podał klient. W przeciwnym wypadku litery będą miały wielkość określoną opcją `default case`. Patrz także opcja `preserve case`.

[global] socket address = adres IP

dozwolone wartości: adres IP *wartość domyślna*: BRAK

Określa adres, pod którym należy czekać na połączenia. Domyślnie Samba czeka na połączenia pod wszystkimi adresami. Używana do obsługi wielu wirtualnych interfejsów w jednym serwerze. Stanowczo odradzamy jej użycie.

[global] socket options = lista opcji gniazd

dozwolone wartości: lista *wartość domyślna*: BRAK

Ustawia opcje gniazd specyficzne dla systemu operacyjnego. Opcja `SO_KEEPALIVE` powoduje, że protokół TCP co cztery godziny sprawdza, czy klient jest wciąż dostępny. Opcja `TCP_NODELAY` powoduje wysyłanie nawet niewielkich pakietów w celu zmniejszenia zwłoki. Te dwie opcje są zalecane, jeśli obsługuje je system operacyjny. Więcej informacji znajdziesz w dodatku B, *Optymalizowanie wydajności Samby*.

[global] status = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Jeśli jest ustawiona na YES, połączenia są rejestrowane w pliku (lub pamięci dzielonej) dostępnym dla polecenia *smbstatus*.

strict sync = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba będzie synchronizować bufory z dyskiem za każdym razem, kiedy klient przyśle pakiet z ustawionym bitem synchronizacji. Jeśli jest ustawiona na NO, Samba zapisze dane dopiero po wypełnieniu buforów. Domyślnie ustawiona na `NO`, ponieważ Eksplorator Windows 98 (błędnie) ustawia ten bit we wszystkich pakietach. Wprowadzona w Sambie 1.9.18p10.

strict locking = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba sprawdza obecność blokad przy każdym dostępie do pliku, a nie tylko na żądanie lub podczas otwierania pliku. Nie zalecana.

[global] strip dot = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Usuwa końcowe kropki z nazw pliku. Zamiast niej należy używać opcji `mangled map`.

[global] syslog = liczba

dozwolone wartości: liczba *wartość domyślna*: 1

Ustawia poziom komunikatów Samby wysyłanych do dziennika systemowego. Wyższe poziomy są bardziej szczegółowe. Rejestrowanie komunikatów musi być także dozwolone przez plik *syslog.conf*.

[global] syslog only = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, komunikaty są rejestrowane tylko w dzienniku systemowym, a nie w standardowych plikach dziennika Samby.

sync always = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba wywołuje funkcję *fsync*(3) po każdym zapisie. Nie należy jej stosować, z wyjątkiem diagnozowania serwera.

[global] time offset = minuty

dozwolone wartości: minuty *wartość domyślna*: 0

Ustawia liczbę minut, które należy dodać do systemowej strefy czasowej. Używana z klientami, które błędnie obliczają czas letni; nie zalecana.

[global] time server = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, proces *nmbd* będzie dostarczał czas klientom.

unix password sync = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba będzie próbowała zmienić hasło uniksowe, kiedy użytkownik zmieni swoje hasło SMB. Służy do ułatwienia synchronizacji między bazami danych Uniksa i Microsoftu. Dodana w wersji 1.9.18p4 Samby. Patrz także `passwd chat`.

unix realname = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Jeśli jest ustawiona na YES, udostępnia klientom pole GECOS z pliku */etc/passwd* jako nazwisko użytkownika.

update encrypted = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: NO

Uaktualnia plik haseł Microsoftu, kiedy użytkownik zaloguje się za pomocą niezaszyfrowanego hasła. Ułatwia wprowadzenie zaszyfrowanych haseł dla użytkowników Windows 95/98 i NT. Wprowadzona w wersji 1.9.18p5 Samby.

user = lista przecinkowa

dozwolone wartości: lista nazw użytkowników oddzielonych przecinkami *wartość domyślna*: BRAK

Synonim opcji `username`.

username = lista przecinkowa

dozwolone wartości: lista nazw użytkowników oddzielonych przecinkami *wartość domyślna*: BRAK

Określa listę nazw użytkowników, na których konta spróbuje zalogować się klient, kiedy stosowane są zabezpieczenia na poziomie udziału. Jej synonimy to `user` i `users`. Nie zalecana. Zamiast niej należy używać w kliencie polecenia `NET USE \\serwer\udzia□%u□ytkownik`.

username level = liczba

dozwolone wartości: liczba *wartość domyślna*: 0

Liczba dużych liter w permutacjach używanych podczas dopasowywania uniksowej nazwy użytkownika. Stosowana ze względu na to, że Windows przesyła nazwy użytkownika literami o takiej samej wielkości. Nie zalecana.

[global] username map = ścieżka do pliku

dozwolone wartości: ścieżka do pliku *wartość domyślna*: BRAK

Określa nazwę i położenie pliku zawierającego pary „nazwa użytkownika w Uniksie – nazwa w Windows". Służy do odwzorowywania nazw o różnej pisowni oraz tych nazw użytkownika Windows, które są dłuższe od ośmiu znaków.

valid chars = lista

dozwolone wartości: lista wartości liczbowych *wartość domyślna*: BRAK

Przestarzała, choć jeszcze nie całkiem. Dodaje znaki narodowe do zestawu znaków. Jej zadania przejęła opcja `client code page`.

valid users = lista użytkowników

dozwolone wartości: lista użytkowników *wartość domyślna*: BRAK (każdy)

Lista użytkowników, którzy mogą korzystać z udziału.

veto files = lista ukośnikowa

dozwolone wartości: lista nazw plików; (elementy oddzielone ukośnikami) *wartość domyślna*: BRAK

Lista plików, które nie są pokazywane klientowi podczas listowania zawartości katalogu. Patrz także opcja `delete veto files`.

veto oplock files = lista ukośnikowa

dozwolone wartości: lista nazw plików; elementy oddzielone ukośnikami — *wartość domyślna*: BRAK

Lista plików, które nie mogą być blokowane oportunistycznie (i buforowane przez klienta). Patrz także opcje `oplocks` i `fake oplocks`.

volume = nazwa udziału

dozwolone wartości: łańcuch — *wartość domyślna*: BRAK

Określa etykietę wolumenu dla udziału dyskowego, zwłaszcza dla CD-ROM-u.

wide links = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: YES

Jeśli jest ustawiona na YES, Samba podąża za dowiązaniami symbolicznymi na zewnątrz bieżącego udziału (lub udziałów). Patrz także opcje `root dir` i `follow symlinks`.

[global] wins proxy = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Jeśli jest ustawiona na YES, demon *nmbd* będzie przekazywał do serwera WINS zapytania o nazwę od starszych klientów, które posługują się rozgłoszeniami. Serwer WINS znajduje się zwykle w innej podsieci.

[global] wins server = host

dozwolone wartości: nazwa hosta — *wartość domyślna*: BRAK

Podaje nazwę DNS lub adres IP serwera WINS.

[global] wins support = wartość logiczna

dozwolone wartości: YES, NO — *wartość domyślna*: NO

Jeśli jest ustawiona na YES, Samba będzie świadczyła usługi WINS. Jeśli `wins support = yes`, opcja `wins server` nie może być ustawiona.

[global] workgroup = nazwa

dozwolone wartości: nazwa grupy roboczej — *wartość domyślna*: wkompilowana

Ustawia nazwę grupy roboczej, której serwer udostępnia swoje zasoby. Zastępuje wkompilowaną nazwę. Zalecamy wybranie nazwy innej niż `WORKGROUP`.

writable = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Antonim opcji `read only`; synonim opcji `write ok`.

write list = lista przecinkowa

dozwolone wartości: lista nazw użytkowników (elementy oddzielone przecinkami) *wartość domyślna:* BRAK (każdy)

Lista użytkowników, którzy mają dostęp w trybie do zapisu i odczytu do udziału przeznaczonego tylko do odczytu. Patrz także opcja `read list`.

write ok = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Synonim opcji `writable`.

[global] write raw = wartość logiczna

dozwolone wartości: YES, NO *wartość domyślna*: YES

Umożliwia wykonywanie szybkich, strumieniowych zapisów przez TCP przy wykorzystaniu buforów o rozmiarze 64 KB. Zalecana.

Słowniczek terminów używanych w spisie opcji

Lista adresów
: Lista adresów IP (oddzielonych spacjami) w formacie ###.###.###.###.

Lista hostów
: Lista hostów, której elementy są oddzielone spacjami. Może zawierać adresy IP, maski adresów, nazwy domenowe oraz słowa ALL i EXCEPT.

Lista interfejsów
: Lista interfejsów (z elementami oddzielonymi spacjami) w formacie adres/maska sieciowa albo adres/liczba bitów, na przykład 192.168.2.10/255.255.255.0 albo 192.168.2.10/24.

Lista odwzorowań
: Lista (z elementami oddzielonymi spacjami) łańcuchów używanych do odwzorowywania nazw plików, na przykład `(*.html *.htm)`.

Lista przecinkowa
: Lista elementów oddzielonych przecinkami.

Lista ukośnikowa
: Lista nazw plików, której elementy są oddzielone znakami „/”, co pozwala na osadzanie spacji. Na przykład `/.*/fred flintstone/*.frk/`.

Lista usług (udziałów)
Lista (z elementami oddzielonymi spacjami) nazw udziałów, bez nawiasów kwadratowych używanych w nagłówku sekcji.

Lista użytkowników
Lista (z elementami oddzielonymi spacjami) nazw użytkowników. W Sambie 1.9 zapis `@nazwa_grupy` oznacza wszystkich członków uniksowej grupy o podanej nazwie. W Sambie 2.0 zapis `@nazwa_grupy` oznacza wszystkich członków grupy sieciowej NIS o podanej nazwie, jeśli taka grupa istnieje, a w przeciwnym razie – wszystkich członków grupy uniksowej. Oprócz tego zapis `+nazwa_grupy` oznacza grupę uniksową, `&nazwa_grupy` – grupę sieciową NIS, a zapisy `&+` i `+&` powodują przeszukanie grup Uniksa i NIS w określonej kolejności.

Lista zdalna
Lista (z elementami oddzielonymi spacjami) par „rozgłoszeniowy adres podsieci – grupa robocza". Na przykład: 192.168.2.255/SERWERY 192.168.4.255/PERSONEL.

Polecenie
Polecenie uniksowe z pełną ścieżką i parametrami.

Tekst
Jedna linia tekstu.

Zmienne pliku konfiguracyjnego

W tabeli C.1 wymienione są zmienne pliku konfiguracyjnego Samby.

Tabela C.1. Zmienne w porządku alfabetycznym

`%a`	Architektura klienta (Samba, WfWg, WinNT, Win95 lub UNKNOWN)
`%d`	Bieżący identyfikator procesu serwera
`%f`	Nazwa pliku buforowego wraz ze ścieżką względną (tylko drukowanie)
`%f`	Użytkownik, który przysłał komunikat (tylko komunikaty)
`%G`	Nazwa podstawowej grupy `%U` (żądanego konta użytkownika)
`%g`	Nazwa podstawowej grupy `%u` (rzeczywistego konta użytkownika)
`%H`	Katalog macierzysty `%u` (rzeczywistego konta użytkownika)
`%h`	Nazwa hosta (internetowa) serwera Samby
`%I`	Adres IP klienta
`%j`	Numer zlecenia wydruku (tylko drukowanie)
`%L`	Nazwa NetBIOS-owa serwera Samby (wirtualne serwery mają wiele nazw)
`%M`	Nazwa hosta (internetowa) klienta
`%m`	Nazwa NetBIOS-owa klienta
`%n`	Nowe hasło (tylko zmiana hasła)
`%N`	Nazwa serwera katalogów macierzystych NIS (bez NIS to samo co `%L`)
`%o`	Stare hasło (tylko zmiana hasła)
`%P`	Katalog główny bieżącego udziału (rzeczywisty)
`%p`	Katalog główny bieżącego udziału (w mapie NIS)

`%p`	Nazwa drukowanego pliku (tylko drukowanie)
`%R`	Używany poziom protokołu (CORE, COREPLUS, LANMAN1, LANMAN2 lub NT1)
`%S`	Nazwa bieżącego udziału
`%s`	Nazwa pliku, w którym jest treść komunikatu (tylko komunikaty)
`%s`	Nazwa pliku buforowego drukarki (tylko drukowanie)
`%T`	Bieżąca data i czas
`%t`	Komputer docelowy (tylko komunikaty)
`%u`	Nazwa użytkownika bieżącego udziału
`%U`	Żądana nazwa użytkownika bieżącego udziału
`%v`	Wersja Samby

D

Spis demonów i poleceń Samby

W tym dodatku zamieszczamy spis opcji linii poleceń i inne informacje, które pomogą ci w używaniu programów wchodzących w skład dystrybucji Samby.

Programy wchodzące w skład dystrybucji Samby

Poniżej zamieszczamy informacje o opcjach linii poleceń używanych przez programy Samby.

smbd

Program *smbd* udostępnia pliki i drukarki, używając jednego strumienia TCP/IP i jednego demona na jednego klienta. Jest sterowany przez domyślny plik konfiguracyjny, `katalog_samby`*/lib/smb.conf*, jednak pierwszeństwo mają opcje linii poleceń.

Plik konfiguracyjny jest automatycznie przetwarzany co minutę. Jeśli w międzyczasie zostanie zmieniony, nowe opcje natychmiast wejdą w życie. Możesz zmusić Sambę do wcześniejszego przeładowania pliku konfiguracyjnego, wysyłając sygnał SIGHUP do procesu *smbd*. Przeładowanie pliku nie będzie jednak miało wpływu na już podłączone klienty. Aby zaprzestać używania tej „odziedziczonej" konfiguracji, klient musi rozłączyć się i połączyć ponownie, albo sam serwer musi zostać zrestartowany, co zmusi wszystkie klienty do ponownego nawiązania połączenia.

Inne sygnały

Aby zamknąć proces *smbd*, wyślij do niego sygnał SIGTERM (-15), który umożliwi mu łagodne zakończenie działania – w przeciwieństwie do sygnału SIGKILL (-9). Aby zwiększyć poziom rejestrowania podczas działania *smbd*, wyślij do programu sygnał SIGUSR1. Aby zmniejszyć poziom rejestrowania, wyślij sygnał SIGUSR2.

Opcje linii poleceń

`-D`

Program *smbd* jest uruchamiany jako demon. Jest to zalecany sposób użycia *smbd* (i zarazem domyślny). Można także uruchamiać *smbd* za pośrednictwem *inetd*.

`-d` *`poziom_diagnostyczny`*
Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może przybierać on wartość od 0 do 10. Podanie wartości w linii polecenia powoduje zastąpienie wartości określonej w pliku *smb.conf*. Poziom diagnostyczny 0 rejestruje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie demona *smbd*.

`-h`
Wyświetla informacje o opcjach linii polecenia programu *smbd*.

Opcje testowe i diagnostyczne

`-a`
Jeśli podasz tę opcję, każde nowe połączenie z serwerem Samby będzie rejestrować komunikaty o logowaniu na końcu pliku dziennika. Jest to przeciwieństwo opcji `-o` i zarazem ustawienie domyślne.

`-i` *`zakres`*
Ta opcja ustawia NetBIOS-owy identyfikator zakresu. Tylko komputery o takim identyfikatorze będą mogły komunikować się z serwerem. Identyfikator zakresu był poprzednikiem grup roboczych; opcja ta jest obsługiwana tylko dla zapewnienia wstecznej zgodności.

`-l` *`plik_dziennika`*
Zapisuje komunikaty diagnostyczne w pliku innym niż wkompilowany lub określony w *smb.conf*. Domyślna nazwa pliku dziennika to często */usr/local/samba/var/log.smb*, */usr/samba/var/log.smb* lub */var/log/log.smb*. Dwa pierwsze położenia nie są zalecane w Linuksie, w którym */usr* często jest systemem plików przeznaczonym tylko do odczytu.

`-O` *`opcje_gniazd`*
Ta opcja linii polecenia ustawia opcje gniazd TCP/IP, używając tych samych parametrów, co opcja konfiguracyjna `socket options`. Często wykorzystuje się ją do optymalizowania wydajności i do testowania.

`-o`
Przeciwieństwo opcji `-a`. Powoduje nadpisywanie plików dziennika po ich otwarciu. Dzięki tej opcji możesz oszczędzić sobie żmudnego wyszukiwania właściwych wpisów, kiedy przeprowadzasz serię testów i za każdym razem sprawdzasz dzienniki.

`-P`
Ta opcja sprawia, że *smbd* nie wysyła żadnych danych do sieci. Zwykle jest używana tylko przez programistów Samby.

`-p` *`numer_portu`*
Opcja ta określa numer portu, z którego serwer będzie odbierał żądania. Obecnie wszystkie klienty Microsoftu wysyłają żądania tylko do portu domyślnego: 139.

`-s plik_konfiguracyjny`
Określa położenie pliku konfiguracyjnego Samby. Domyślne położenie tego pliku to */usr/local/samba/lib/smb.conf*, ale możesz zmienić je z linii polecenia, zwykle w celach diagnostycznych.

nmbd

Program *nmbd* to demon NetBIOS-owych usług nazewniczych i przeglądania. Odpowiada na zapytania o nazwę, rozgłaszane za pomocą protokołu NetBIOS ponad TCP/IP (NBT) przez klienty SMB, a opcjonalnie także na żądania Windows Internet Name Service (WINS). Oba te mechanizmy są używane przez klienty SMB do wyszukiwania adresu IP na podstawie nazwy komputera. Mechanizm rozgłoszeniowy korzysta z rozgłoszeń UDP/IP i ogranicza się do lokalnej podsieci, natomiast WINS używa protokołu TCP/IP, który może być trasowany. Jeśli demon *nmbd* działa jako serwer WINS, wówczas przechowuje aktualną bazę nazw i adresów w pliku *wins.dat* w katalogu `katalog_samby/var/locks`.

Działający program *nmbd* może także odpowiadać na żądania protokołu przeglądania używanego przez Otoczenie sieciowe Windows. Przeglądanie opiera się na protokole łączącym ogłoszenia o serwerach i świadczonych przez nie usługach w aktywny katalog usług. Protokół przeglądania zapewnia więc dynamiczny katalog serwerów oraz udostępnianych przez nie dysków i drukarek. Podobnie jak w przypadku WINS, usługa ta była początkowo realizowana przez wysyłanie rozgłoszeń UDP/IP do lokalnej podsieci. Obecnie, dzięki głównym przeglądarkom lokalnym, przeglądanie polega na nawiązywaniu połączeń TCP/IP z serwerem. Jeśli demon *nmbd* działa jako główna przeglądarka lokalna, wówczas przechowuje bazę danych przeglądania w pliku *browse.dat* w katalogu `katalog_samby/var/locks`.

Sygnały

Podobnie jak *smbd*, program *nmbd* odpowiada na kilka uniksowych sygnałów. Wysłanie do niego sygnału SIGHUP spowoduje zrzucenie znanych programowi nazw do pliku *namelist.debug* w katalogu `katalog_samby/locks`, a bazy danych przeglądania – do pliku *browse.dat* w tym samym katalogu. Aby zamknąć proces *nmbd*, wyślij do niego sygnał SIGTERM (-15) zamiast SIGKILL (-9), co pozwoli mu na łagodne zakończenie pracy. Możesz zwiększyć poziom diagnostyczny programu *nmbd*, wysyłając do niego sygnał SIGUSR1, albo zmniejszyć, wysyłając sygnał SIGUSR2.

Opcje linii polecenia

`-D`
Program *nmbd* jest uruchamiany jako demon. Jest to zalecany sposób użycia *nmbd*. Można także uruchamiać *nmbd* za pośrednictwem *inetd*.

`-d poziom_diagnostyczny`
Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może mieć on wartość od 0 do 10. Podanie wartości w linii polecenia powoduje zastąpienie wartości określonej w pliku *smb.conf*. Poziom diagnostyczny 0 reje-

struje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie demona *nmbd*.

`-h`
Wyświetla informacje o opcjach linii polecenia programu *nmbd* (także `-?`).

Opcje testowe i diagnostyczne

`-a`
Jeśli podasz tę opcję, każde nowe połączenie z serwerem Samby będzie rejestrować komunikaty o logowaniu na końcu pliku dziennika. Jest to przeciwieństwo opcji `-o` i zarazem ustawienie domyślne.

`-H` *`plik_hostów`*
Ta opcja ładuje standardowy plik hostów w celu odwzorowywania nazw.

`-i` *`zakres`*
Ta opcja ustawia NetBIOS-owy identyfikator zakresu. Tylko komputery o takim identyfikatorze będą mogły komunikować się z serwerem. Identyfikator zakresu był poprzednikiem grup roboczych; opcja ta jest obsługiwana tylko dla zapewnienia wstecznej zgodności.

`-l` *`plik_dziennika`*
Zapisuje komunikaty diagnostyczne w pliku innym niż wkompilowany lub określony w *smb.conf*. Domyślna nazwa pliku dziennika to często */usr/local/samba/var/log.nmb*, */usr/samba/var/log.nmb* lub */var/log/log.nmb*. Dwa pierwsze położenia nie są zalecane w Linuksie, w którym */usr* często jest systemem plików przeznaczonym tylko do odczytu.

`-n` *`nazwa_NetBIOS-owa`*
Ta opcja umożliwia zmianę nazwy, pod jaką będzie ogłaszał się demon. Podanie tej opcji w linii polecenia spowoduje zastąpienie wartości określonej w pliku konfiguracyjnym Samby za pomocą opcji `netbios name`.

`-O` *`opcje_gniazd`*
Ta opcja linii polecenia ustawia opcje gniazd TCP/IP, używając tych samych parametrów, co opcja konfiguracyjna `socket options`. Często wykorzystuje się ją do optymalizowania wydajności i do testowania.

`-o`
Przeciwieństwo opcji `-a`. Powoduje nadpisywanie plików dziennika po ich otwarciu. Dzięki tej opcji możesz oszczędzić sobie żmudnego wyszukiwania właściwych wpisów, kiedy przeprowadzasz serię testów i za każdym razem sprawdzasz dzienniki.

`-p` *`numer_portu`*
Opcja ta określa numer portu UDP/IP, z którego serwer będzie odbierał żądania. Obecnie wszystkie klienty Microsoftu wysyłają żądania tylko do portu domyślnego: 137.

`-s plik_konfiguracyjny`
Określa położenie pliku konfiguracyjnego Samby. Domyślne położenie tego pliku to */usr/local/samba/lib/smb.conf*, ale możesz zmienić je z linii polecenia, zwykle w celach diagnostycznych.

`-v`
Ta opcja wyświetla wersję Samby.

Plik startowy Samby

Samba jest zwykle uruchamiana z uniksowych plików *rc* podczas startu systemu. W systemach ze zbiorem katalogów */etc/rcN.d* można to zrobić, umieszczając odpowiednio nazwany skrypt w katalogu */rc*. Zwykle skrypt uruchamiający Sambę nosi nazwę *S91samba*, a zatrzymujący ją – *K91samba*. W Linuksie skrypty te są zazwyczaj przechowywane w podkatalogu */etc/rc2.d*, a w Solarisie – w */etc/rc3.d*. W komputerach z plikiem */etc/rc.local* możesz dopisać do tego pliku poniższe dwie linie:

```
/usr/local/samba/bin/smbd -D
/usr/local/samba/bin/nmbd -D
```

Poniższy przykładowy skrypt obsługuje dwa dodatkowe polecenia, `status` i `restart`, oprócz zwykłych poleceń `start` i `stop` używanych w Uniksach typu System V:

```
#!/bin/sh
#
# /etc/rc2.d/S91samba - zarz□dza serwerem SMB metod□ Systemu V
#
OPCJE="-D"
#DIAGN=-d3
PS="ps ax"
KAT_SAMBY=/usr/local/samba
case "$1" in
'start')
    echo "Uruchamiam Samb□"
    $KAT_SAMBY/bin/smbd $OPCJE $DIAGN
    $KAT_SAMBY/bin/nmbd $OPCJE $DIAGN
    ;;
'stop')
    echo "Zatrzymuj□ Samb□"
    $PS | awk '/usr.local.samba.bin/ { print $1}' |\
    xargs kill
    ;;
'status')
    x='$PS | grep -v grep | grep '$KAT_SAMBY/bin''
    if [ ! "$x" ]; then
        echo "Nie dzia□a □aden proces Samby"
    else
        echo "  PID TT STAT  CZAS POLECENIE"
        echo "$x"
    fi
    ;;
'restart')
    /etc/rc2.d/S91samba stop
    /etc/rc2.d/S91samba start
    /etc/rc2.d/S91samba status
    ;;
```

```
*)
    echo "$0: B□□dne polecenie. Wpisz: $0 start, stop, status lub restart."
    ;;
esac
exit
```

Będziesz musiał podać właściwe ścieżki i opcje polecenia `ps`, aby dostosować je do swojego systemu. Możesz także dołączyć inne polecenia w zależności od swoich potrzeb, na przykład nakazujące Sambie przeładowanie pliku *smb.conf* lub zrzucenie tabel *nmbd*.

smbsh

Program *smbsh* umożliwia korzystanie ze zdalnego udziału Windows tak, jakby był to zwykły katalog uniksowy. Kiedy go uruchomisz, udostępni dodatkowe drzewo katalogów pod katalogiem */smb*. Podkatalogami */smb* są serwery, a podkatalogami serwerów są poszczególne udziały dyskowe i drukarki. Polecenia wykonywane przez *smbsh* traktują system plików */smb* tak, jakby był to system lokalny. Oznacza to, że nie musisz dołączać obsługi *smbmount* do jądra i montować systemów plików Windows w taki sposób, w jaki montujesz systemy plików NFS. Aby włączyć *smbsh*, musisz skonfigurować Sambę z opcją `--with-smbwrappers`.

Opcje

`-d` *`poziom_diagnostyczny`*
: Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może mieć on wartość od 0 do 10. Podanie wartości w linii polecenia powoduje zastąpienie wartości określonej w pliku *smb.conf*. Poziom diagnostyczny 0 rejestruje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie programu *smbsh*.

`-l` *`plik_dziennika`*
: Określa nazwę używanego pliku dziennika.

`-P` *`przedrostek`*
: Określa katalog, w którym należy zamontować system plików SMB. Domyślnie jest to */smb*.

`-R` *`kolejno□□ odwzorowywania`*
: Określa kolejność korzystania z usług nazewniczych. Opcja ta przypomina opcję konfiguracyjną `resolve order` i może przyjmować cztery parametry: `lmhosts`, `host`, `wins` i `bcast`, w dowolnej kolejności.

`-U` *`u□ytkownik`*
: Umożliwia stosowanie opcji *`u□ytkownik%has□o`*.

`-W` *`grupa_robocza`*
: Określa nazwę grupy roboczej, do której podłączy się klient.

smbclient

Program *smbclient* to „złota rączka" w pakiecie Samby. Początkowo napisany jako narzędzie testowe, z czasem stał się pełnym klientem uniksowym z interaktywnym interfejsem przypominającym FTP. Niektóre z jego opcji nadal służą do testowania i optymalizowania serwera; umożliwia też łatwe sprawdzenie, czy Samba działa w serwerze.

Program *smbclient* można uznać za cały pakiet programów:

- interaktywny program do transmisji plików, podobny do FTP,
- interaktywny program drukujący,
- interaktywny program archiwizujący,
- program do wysyłania komunikatów sterowany z linii poleceń,
- program archiwizacyjny sterowany z linii poleceń (patrz też program *smbtar*),
- program do wysyłania zapytań o usługi świadczone przez serwer,
- program diagnostyczny sterowany z linii poleceń.

Ogólne opcje linii poleceń

Program ma zwykły zbiór opcji podobnych do *smbd*, które mają zastosowanie zarówno podczas użycia interaktywnego, jak i sterowania z linii poleceń. Oto składnia polecenia:

```
smbclient //nazwa_serwera/nazwa_udziau [hasło] [-opcje]
```

Oto wszystkie opcje linii poleceń:

-d *poziom_diagnostyczny*
: Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może mieć on wartość od 0 do 10 lub A, co oznacza wszystkie komunikaty. Podanie wartości w linii polecenia powoduje zastąpienie wartości określonej w pliku *smb.conf*. Poziom diagnostyczny 0 rejestruje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie programu *smbclient*.

-h
: Wyświetla informacje o użyciu programu *smbclient* z linii poleceń.

-n *nazwa_NetBIOS-owa*
: Ta opcja umożliwia zmianę nazwy NetBIOS-owej, pod jaką będzie ogłaszał się program.

Działanie programu smbclient

Po wydaniu polecenia `smbclient //nazwa_serwera/udział` zostaniesz poproszony o wpisanie nazwy użytkownika i hasła. Jeśli logowanie przebiegnie pomyślnie, program połączy się z udziałem i zostanie wyświetlone zgłoszenie niemal takie jak w FTP (odwrotny ukośnik w zgłoszeniu zostanie zastąpiony nazwą bieżącego katalogu, kiedy będziesz poruszał się po systemie plików):

```
smb:\>
```

Z tej linii zgłoszenia możesz wydawać polecenia wymienione w tabeli D.1, podobne do poleceń FTP. Argumenty w nawiasach kwadratowych są opcjonalne.

Tabela D.1. Polecenia programu smbclient

Polecenie	*Opis*
`? polecenie`	Wyświetla listę poleceń lub pomoc na temat podanego polecenia
`help [polecenie]`	Wyświetla listę poleceń lub pomoc na temat podanego polecenia
`! [polecenie]`	Jeśli podasz polecenie, zostanie ono wykonane w lokalnej powłoce. Jeśli nie, przejdziesz do lokalnej powłoki klienta
`dir [nazwa_pliku]`	Wyświetla wszystkie pliki w bieżącym katalogu, których nazwa odpowiada podanej *nazwie_pliku*, a w razie pominięcia *nazwy_pliku* wyświetla wszystkie pliki
`ls [nazwa_pliku]`	Wyświetla wszystkie pliki w bieżącym katalogu, których nazwa odpowiada podanej *nazwie_pliku*, a w razie pominięcia *nazwy_pliku* wyświetla wszystkie pliki
`cd [katalog]`	Jeśli podasz *katalog*, przechodzi do określonego katalogu w zdalnym komputerze. Jeśli nie, wyświetla bieżący katalog w zdalnym komputerze
`lcd [katalog]`	Jeśli podasz *katalog*, przechodzi do określonego katalogu w lokalnym komputerze. Jeśli nie, wyświetla bieżący katalog w lokalnym komputerze
`get zdalny_plik [lokalny_plik]`	Kopiuje *zdalny_plik* do lokalnego komputera. Jeśli podasz nazw *lokalnego_pliku*, plik zostanie skopiowany pod tą nazwą. Plik jest traktowany jako binarny; nie dokonuje się konwersji między znakami nowej linii a sekwencjami powrót karetki/nowa linia
`put lokalny_plik [zdalny_plik]`	Kopiuje *lokalny_plik* do zdalnego komputera. Jeśli podasz nazwę *zdalnego_pliku*, plik zostanie skopiowany pod tą nazwą. Plik jest traktowany jako binarny; nie dokonuje się konwersji między znakami nowej linii a sekwencjami powrót karetki/nowa linia
`mget wzorzec`	Pobiera ze zdalnego komputera wszystkie pliki o nazwach pasujących do *wzorca*
`mput wzorzec`	Umieszcza w zdalnym komputerze wszystkie lokalne pliki o nazwach pasujących do *wzorca*
`prompt`	Włącza lub wyłącza interaktywne potwierdzanie operacji w poleceniach `mget` i `mput`
`lowercase ON` (lub `OFF`)	Jeśli opcja `lowercase` jest włączona (`ON`), program *smbclient* będzie przekształcał nazwy plików na małe litery podczas operacji `mget` lub `get` (ale nie `mput` i `put`)
`del nazwa_pliku`	Usuwa plik ze zdalnego komputera
`md katalog`	Tworzy katalog w zdalnym komputerze
`mkdir katalog`	Tworzy katalog w zdalnym komputerze
`rd katalog`	Usuwa katalog ze zdalnego komputera
`rmdir katalog`	Usuwa katalog ze zdalnego komputera

Polecenie	*Opis*
`setmode nazwa_pliku [+|-]rsha`	Ustawia dosowe atrybuty plików, używając uniksowych praw dostępu. Opcja `r` oznacza plik tylko do odczytu, `s` – systemowy, `h` – ukryty, `a` – archiwalny
`exit`	Kończy działanie programu *smbclient*
`quit`	Kończy działanie programu *smbclient*

Istnieją także polecenia maskujące i rekurencyjne przydatne podczas kopiowania większych ilości plików; opis ich użycia znajdziesz na stronie podręcznika man dla programu *smbclient*. Nie licząc masek, rekurencji i braku transmisji w trybie ASCII, *smbclient* działa dokładnie tak, jak FTP. Zauważ, że ze względu na binarny tryb transmisji, pliki Windows skopiowane do Uniksa będą miały linie kończące się znakami powrotu karetki i nowej linii (`\r\n`), a nie samym znakiem nowej linii (`\n`) używanym w Uniksie.

Polecenia wydruku

Program *smbclient* umożliwia także dostęp do drukarek, łącząc się z udziałami drukarek. Kiedy jesteś już połączony, możesz sterować wydrukiem za pomocą poleceń z tabeli D.2.

Tabela D.2. Polecenia wydruku w programie smbclient

Polecenie	*Opis*
`print` *nazwa_pliku*	Drukuje plik, kopiując go z komputera lokalnego do zdalnego i tam zlecając jego wydruk
`printmode` *text* `|` *graphics*	Informuje serwer, że przesłane dane są czystym tekstem (ASCII) albo binarnymi danymi graficznymi w formacie akceptowanym przez drukarkę. Użytkownik sam musi zadbać, aby drukowany plik był właściwego typu
`queue`	Wyświetla kolejkę w udziale drukarki, z którym się połączyłeś, pokazując identyfikator zadania, nazwę, rozmiar i status zlecenia wydruku

Wreszcie, aby wydrukować plik za pomocą programu *smbclient*, użyj opcji `-c`:

```
cat drukowany_plik | smbclient //serwer/nazwa_drukarki -c "print -"
```

Polecenia archiwizujące

Program *smbclient* może archiwizować pliki przechowywane w udziałach. Zwykle robi się to z linii poleceń za pomocą polecenia *smbtar*, ale można także użyć poleceń interaktywnych wymienionych w tabeli D.3.

Tabela D.3. Polecenia archiwizujące

Polecenie	*Opis*
`tar c|x[IXbgNa] operandy`	Tworzy lub rozpakowuje archiwum tar, podobnie jak program sterowany z linii poleceń
`blocksize rozmiar`	Określa rozmiar bloku używany przez *tar*, w jednostkach 512-bajtowych
`tarmode full |inc |reset|noreset`	Sprawia, że polecenie *tar* uzwględnia ustawienie dosowego bitu archiwalnego. W trybie `full` (pełnym, domyślnym) polecenie *tar* archiwizuje wszystkie pliki. W trybie `inc` (przyrostowym) polecenie *tar* kopiuje tylko te pliki, w których ustawiony jest bit archiwalny. W trybie `reset` (zerowania) polecenie *tar* zeruje bit archiwalny we wszystkich kopiowanych plikach (udział musi być zapisywalny), w trybie `noreset` bit archiwalny nie zostanie wyzerowany mimo zarchiwizowania pliku

Opcje linii poleceń służące do wysyłania komunikatów

`-M` *NetBIOS-owa_nazwa_komputera*
: Ta opcja umożliwia wysyłanie komunikatów protokołu WinPopup do innego komputera. Po nawiązaniu połączenia możesz wpisać swój komunikat i zakończyć go przez naciśnięcie [Ctrl+D]. Jeśli w docelowym komputerze nie działa WinPopup, otrzymasz komunikat o błędzie.

`-U` *u□ytkownik*
: Ta opcja pozwala na pośrednie określenie nadawcy komunikatu.

Opcje linii poleceń służące do archiwizacji plików

Do archiwizowania plików można wykorzystać kombinację opcji `-T` (archiwizowanie), `-D` (początkowy katalog) oraz `-c` (polecenie). Lepiej jednak to zrobić za pomocą programu *smbtar*, który omówimy niebawem. Nie zalecamy bezpośredniego użycia programu *smbclient* jako programu archiwizującego.

`-D` *katalog_pocz□tkowy*
: Przechodzi do określonego katalogu przed rozpoczęciem operacji.

`-c` *□a□cuch_polece□*
: Przekazuje *□a□cuch_polece□* do interpretera poleceń, który traktuje go jako listę poleceń (oddzielonych średnikami) do wykonania. Przydaje się to do przekazania poleceń, takich jak `tarmode inc`, które sprawia, że polecenie `smbclient -T` skopiuje tylko pliki z ustawionym bitem archiwalnym.

`-T` *polecenie nazwa_pliku*
: Uruchamia sterownik *tar*, zgodny z programem *gtar*. Dwa najważniejsze polecenia to: `c` (utwórz archiwum) i `x` (rozpakuj pliki), po których mogą następować poniższe opcje:

`a` Zeruje bity archiwalne po skopiowaniu plików.

`b` *rozmiar*
Ustawia rozmiar bloku w jednostkach 512-bajtowych.

`g` Archiwizuje tylko te pliki, w których ustawiony jest bit archiwalny.

`I` *plik*
Dołącza pliki i katalogi (jest to działanie domyślne). Nie wykonuje dopasowań do wzorca.

`N` *nazwa_pliku*
Archiwizuje tylko pliki nowsze od pliku o podanej nazwie.

`q` Nie wyświetla komunikatów diagnostycznych.

`X` *plik*
Wyłącza pliki.

Wysyłanie z linii poleceń zapytań o usługi

Jeśli uruchomisz program *smbclient* w następujący sposób:

```
smbclient -L nazwa_serwera
```

wylistuje on udziały i inne usługi udostępniane przez serwer. Jest to przydatne, jeśli nie masz *smbwrappers*. Bywa także pomocne jako narzędzie diagnostyczne.

Opcje testowe i diagnostyczne

Program *smbclient* można wywołać w różnych trybach działania z następującymi opcjami diagnostycznymi i testowymi:

`-B` *adres_IP*
Określa adres rozgłoszeniowy.

`-d` *poziom_diagnostyczny*
Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może mieć on wartość od 0 do 10 lub A, co oznacza wszystkie opcje diagnostyczne. Poziom diagnostyczny 0 rejestruje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie programu.

`-E`
Wysyła wszystkie komunikaty na standardowe wyjście błędu, zamiast na standardowe wyjście.

`-I` *adres_IP*
Określa adres IP serwera, z którym należy się połączyć.

`-i` *zakres*
Ta opcja ustawia NetBIOS-owy identyfikator zakresu. Tylko komputery o takim identyfikatorze będą mogły komunikować się z serwerem. Identyfikator zakresu był poprzednikiem grup roboczych; opcja ta jest obsługiwana tylko dla zapewnienia wstecznej zgodności.

`-l` *plik_dziennika*
Wysyła komunikaty do określonego pliku.

`-N`
Nie wyświetla pytania o hasło. Jeśli podasz hasło w linii polecenia lub użyjesz tej opcji, klient nie zapyta o hasło.

`-n` *`nazwa_NetBIOS-owa`*
Ta opcja umożliwia zmianę nazwy, pod jaką będzie ogłaszał się program.

`-O` *`opcje_gniazd`*
Ta opcja linii polecenia ustawia opcje gniazd TCP/IP, używając tych samych parametrów, co opcja konfiguracyjna `socket options`. Często wykorzystuje się ją do optymalizowania wydajności i do testowania.

`-p` *`numer_portu`*
Opcja ta określa numer portu, z którego klient będzie odbierał żądania.

`-R` *`kolejno□□_odwzorowywania`*
Określa kolejność korzystania z usług nazewniczych. Opcja ta przypomina opcję konfiguracyjną `resolve order` i może przyjmować cztery parametry: `lmhosts`, `host`, `wins` i `bcast`, w dowolnej kolejności.

`-s` *`plik_konfiguracyjny`*
Określa położenie pliku konfiguracyjnego Samby. Używana w celach diagnostycznych.

`-t` *`kod_terminalowy`*
Ustawia kod terminalowy dla języków azjatyckich.

`-U` *`u□ytkownik`*
Ustawia nazwę użytkownika i opcjonalnie hasło (na przykład `-U` *`janek%tajne`*).

`-W` *`grupa_robocza`*
Określa nazwę grupy roboczej, której członkiem jest klient.

Jeśli chcesz przetestować którąś z usług nazewniczych, uruchom program *smbclient* z opcją `-R` i nazwą tej jednej usługi. Program *smbclient* skorzysta wówczas tylko z tej usługi, którą podałeś w linii polecenia.

smbstatus

Program *smbstatus* listuje bieżące połączenia z serwerem Samby. Jego wyniki dzielą się na trzy sekcje. W pierwszej sekcji wymieniane są różne udziały używane przez konkretnych użytkowników. W drugiej sekcji wymieniane są zablokowane pliki we wszystkich udziałach Samby. Wreszcie, w trzeciej sekcji wyświetlane są informacje o zużyciu pamięci. Na przykład:

```
# smbstatus
Samba version 2.0.3
Service      uid      gid      pid     machine
----------------------------------------------
siec         davecb   davecb   7470   feniks   (192.168.220.101) Tue May 23
wer          wer      wer      7589   chimera  (192.168.220.102) Tue May 23

Locked files:
Pid    DenyMode   R/W        Oplock           Name
--------------------------------------------------
```

```
7589   DENY_NONE  RDONLY   EXCLUSIVE+BATCH  /home/samba/quicken/inet/common/
system/help.bmp   Sun May 16 21:23:40 1999
7470   DENY_WRITE RDONLY   NONE             /home/samba/word/office/findfast.exe
Sun May 16 20:51:08 1999
7589   DENY_WRITE RDONLY   EXCLUSIVE+BATCH  /home/samba/quicken/lfbmp70n.dll
Sun May 16 21:23:39 1999
7589   DENY_WRITE RDWR     EXCLUSIVE+BATCH  /home/samba/quicken/inet/qdata/
runtime.dat   Sun May 16 21:23:40 1999
7470   DENY_WRITE RDONLY   EXCLUSIVE+BATCH  /home/samba/word/office/osa.exe
Sun May 16 20:51:09 1999
7589   DENY_WRITE RDONLY   NONE             /home/samba/quicken/qversion.dll
Sun May 16 21:20:33 1999
7470   DENY_WRITE RDONLY   NONE             /home/samba/quicken/qversion.dll
Sun May 16 20:51:11 1999

Share mode memory usage (bytes):
   1043432(99%) free + 4312(0%) used + 832(0%) overhead = 1048576(100%) total
```

Opcje

`-b`
: Polecenie wyświetla skrócone wyniki. Obejmują one wersję Samby oraz informacje o użytkownikach zalogowanych w serwerze.

`-d`
: Polecenie wyświetla szczegółowe wyniki, obejmujące wszystkie trzy sekcje z powyższego przykładu. Jest to opcja domyślna.

`-L`
: Wyświetla tylko bieżące blokady plików. Odpowiada to drugiej sekcji w szczegółowych wynikach.

`-p`
: Wyświetla tylko listę identyfikatorów procesu *smbd*. Często używana w skryptach.

`-S`
: Wyświetla tylko listę połączeń z udziałami. Odpowiada to pierwszej sekcji w szczegółowych wynikach.

`-s` *`plik_konfiguracyjny`*
: Określa plik konfiguracyjny Samby używany podczas wykonywania polecenia.

`-u` *`nazwa_u□ytkownika`*
: Ogranicza wyniki programu *smbstatus* do aktywności określonego użytkownika.

smbtar

Program *smbtar* to skrypt powłoki, korzystający z polecenia *smbclient*, dzięki któremu podczas archiwizowania plików można posłużyć się bardziej zrozumiałymi opcjami. Funkcjonalnie jest odpowiednikiem uniksowego polecenia *tar*.

Opcje

`-a`
: Zeruje bit archiwalny.

`-b` *rozmiar_bloku*
Rozmiar bloku. Domyślnie 20.

`-d` *katalog*
Przechodzi do wskazanego katalogu przed rozpakowaniem lub zarchiwizowaniem plików.

`-i`
Tryb przyrostowy; pliki zostaną skopiowane tylko wtedy, gdy mają ustawiony bit archiwalny. Bity archiwalne są zerowane po odczytaniu każdego pliku.

`-l` *poziom_rejestrowania*
Określa poziom rejestrowania.

`-N` *nazwa_pliku*
Kopiuje tylko te pliki, które są nowsze od czasu ostatniej modyfikacji pliku o podanej nazwie. Służy do wykonywania przyrostowych kopii zapasowych.

`-p` *has□o*
Określa hasło dostępu do udziału.

`-r`
Odtwarza pliki w udziałach z pliku *tar*.

`-s` *serwer*
Określa nazwę serwera SMB/CIFS, w którym znajduje się udział.

`-t` *ta□ma*
Urządzenie taśmowe lub plik. Domyślnie jest to wartość zmiennej środowiskowej `$TAPE` lub plik o nazwie *tar.out*, jeśli zmienna `$TAPE` nie jest ustawiona.

`-u` *u□ytkownik*
Określa użytkownika udziału. Możesz podać także hasło, w formacie *u□ytkownik%has□o*.

`-v`
Włącza tryb szczegółowy.

`-X` *plik*
Nakazuje wyłączyć podane pliki z operacji archiwizowania lub odtwarzania.

`-x` *udzia□*
Określa nazwę udziału, z którym należy się połączyć. Domyślna nazwa to `backup`, ponieważ kopie zapasowe często wykonuje się z wykorzystaniem udziału o takiej nazwie.

Oto przykładowe polecenie, które archiwizuje dane użytkownika `zuzia`:

```
# smbtar -s pecet -x zuzia -u zuzia -p tajne -t zuzia.tar
```

nmblookup

Program kliencki *nmblookup* sprawdza, czy usługi nazewnicze NetBIOS-u ponad UDP/IP potrafią odwzorowywać nazwy komputerów NBT na adresy IP. Polecenie *nmblookup* rozgłasza zapytanie o nazwę w lokalnej podsieci i czeka, aż komputer o tej nazwie udzieli odpowiedzi. Polecenie to jest odpowiednikiem programów

nslookup(1) lub *dig(1)*, przeznaczonym dla sieci Windows. Może ono wyszukiwać zarówno zwykłe nazwy NetBIOS-owe, jak i te niezwykłe, typu `__MSBROWSE__`, które są używane przez mechanizmy nazewnicze Windows do świadczenia usług katalogowych. Jeśli chcesz wysłać zapytanie o określony typ nazwy, dołącz NetBIOS-owy `<typ>` na końcu nazwy.

Oto składnia polecenia:

```
nmblookup [-opcje] nazwa
```

Oto obsługiwane opcje:

`-A`
: Interpretuje *nazwę* jako adres IP i wysyła zapytanie o status węzła pod tym adresem.

`-B` *adres_rozgłoszeniowy*
: Wysyła zapytanie na określony *adres_rozgłoszeniowy*. Domyślnie zapytanie jest wysyłane pod adresem rozgłoszeniowym podstawowego interfejsu sieciowego.

`-d` *poziom_diagnostyczny*
: Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może mieć on wartość od 0 do 10. Poziom diagnostyczny 0 rejestruje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie programu.

`-h`
: Wyświetla informacje o użyciu opcji programu.

`-i` *zakres*
: Ta opcja ustawia NetBIOS-owy identyfikator zakresu. Tylko komputery o takim identyfikatorze będą mogły komunikować się z serwerem. Identyfikator zakresu był poprzednikiem grup roboczych; opcja ta jest obsługiwana tylko dla zapewnienia wstecznej zgodności.

`-M`
: Znajduje główną przeglądarkę lokalną. Stosuje w tym celu rozgłoszeniowe wyszukiwanie komputera, który odpowie na specjalną nazwę `__MSBROWSE__`, a następnie uzyskuje informacje od tego komputera, zamiast rozgłaszać samo zapytanie.

`-R`
: Ustawia w pakiecie bit zezwolenia na rekurencję. Sprawia on, że komputer udzielający odpowiedzi spróbuje przeprowadzić wyszukiwanie w WINS i zwróci adres oraz inne informacje przechowywane przez serwer WINS.

`-r`
: Używa portu głównego o numerze 137 w komputerach Windows.

`-S`
: Kiedy zapytanie o nazwę zwróci adres IP, wysyła także zapytanie o status węzła. Zapytanie to zwraca wszystkie typy zasobów, które są znane komputerowi, wraz z ich liczbowymi atrybutami. Na przykład:

```
% nmblookup -d 4 -S FENIKS
received 6 names
        FENIKS          <00> - <GROUP> B <ACTIVE>
        FENIKS          <03> -         B <ACTIVE>
        FENIKS          <1d> -         B <ACTIVE>
        FENIKS          <1e> - <GROUP> B <ACTIVE>
        FENIKS          <20> -         B <ACTIVE>
        ..__MSBROWSE__..<01> - <GROUP> B <ACTIVE>
```

`-s` *plik_konfiguracyjny*
: Określa położenie pliku konfiguracyjnego Samby. Domyślne położenie tego pliku to */usr/local/samba/lib/smb.conf*, ale możesz zmienić je z linii polecenia, zwykle w celach diagnostycznych.

`-T`
: Ta opcja służy do tłumaczenia adresów IP na wyszukane nazwy.

`-U` *adres_bezpo□redni*
: Wysyła zapytanie bezpośrednio pod wskazanym adresem. Używana w połączeniu z opcją `-R` do odpytywania serwerów WINS.

Zauważ, że polecenie *nmblookup* nie umożliwia podania nazwy grupy roboczej. Możesz ominąć tę przeszkodę, umieszczając w pliku opcję `workgroup = nazwa_grupy` i wywołując polecenie *nmblookup* z opcją `-s` *plik_konfiguracyjny*.

smbpasswd

Program *smbpasswd* ma dwie różne funkcje. Kiedy jest uruchamiany przez zwykłych użytkowników, zmienia ich zaszyfrowane hasła. Kiedy jest uruchamiany przez roota, uaktualnia plik zaszyfrowanych haseł. Kiedy zostanie uruchomiony przez zwykłego użytkownika bez żadnych opcji, łączy się z podstawowym kontrolerem domeny i zmienia hasło użytkownika Windows.

Program odmówi pracy, jeśli nie działa demon *smbd*, opcje `hosts allow` lub `hosts deny` uniemożliwiają połączenie z lokalnym hostem (127.0.0.1) lub ustawiona jest opcja `encrypted passwords = no`.

Opcje dostępne dla zwykłego użytkownika

`-D` *poziom_diagnostyczny*
: Określa poziom diagnostyczny (nazywany także poziomem rejestrowania). Może mieć on wartość od 0 do 10. Poziom diagnostyczny 0 rejestruje tylko najważniejsze komunikaty, poziom 1 to ustawienie standardowe, natomiast poziomy 3 i wyższe służą głównie do celów diagnostycznych i znacznie spowalniają działanie programu.

`-h`
: Wyświetla informacje o użyciu opcji programu.

`-r` *nazwa_zdalnego_komputera*
: Określa komputer, w którym należy zmienić hasło. Zdalny komputer musi być podstawowym kontrolerem domeny (PDC).

`-R kolejno□□_odwzorowywania`
Określa kolejność korzystania z usług nazewniczych. Opcja ta przypomina opcję konfiguracyjną `resolve order` i może przyjmować cztery parametry: `lmhosts`, `host`, `wins` i `bcast`, w dowolnej kolejności.

`-U nazwa_u□ytkownika`
Używana tylko w połączeniu z opcją `-r`, aby zmodyfikować nazwę użytkownika, która w systemie zdalnym ma inną pisownię.

Opcje dostępne tylko dla roota

`-a nazwa_u□ytkownika`
Dodaje użytkownika do pliku zaszyfrowanych haseł.

`-d nazwa_u□ytkownika`
Wyłącza konto użytkownika w pliku zaszyfrowanych haseł.

`-e nazwa_u□ytkownika`
Włącza konto użytkownika w pliku zaszyfrowanych haseł.

`-m nazwa_komputera`
Zmienia hasło konta komputera. Konta komputerów są używane do uwierzytelniania komputerów łączących się z podstawowym lub zapasowym kontrolerem domeny.

`-j nazwa_domeny`
Dodaje serwer Samby do domeny Windows NT.

`-n`
Ustawia puste hasło użytkownika.

`-s nazwa_u□ytkownika`
Sprawia, że program *smbpasswd* nie wyświetla żadnych komunikatów i odczytuje stare i nowe hasła ze standardowego wejścia, a nie z urządzenia */dev/tty*. Jest to przydatne w skryptach.

testparm

Program *testparm* sprawdza, czy plik *smb.conf* jest wewnętrznie spójny i czy nie występują w nim oczywiste błędy. Oto składnia polecenia:

`testparm [opcje] nazwa_pliku_konfiguracyjnego [nazwa_hosta adres_IP]`

Jeśli nie podasz nazwy pliku konfiguracyjnego, polecenie sprawdzi domyślny plik *katalog_samby/lib/smb.conf*. Jeśli podasz nazwę hosta i adres IP, polecenie przeprowadzi dodatkowy test, aby sprawdzić, czy określony komputer będzie mógł łączyć się z serwerem Samby. Jeśli podasz nazwę hosta, powinieneś podać także jego adres IP.

Opcje

`-h`
Wyświetla informacje o użyciu opcji programu.

`-L nazwa_serwera`
Ustawia zmienną `%L` pliku konfiguracyjnego na podaną nazwę serwera.

`-s`
Ta opcja sprawia, że program *testparm* nie prosi użytkownika o naciśnięcie klawisza [Enter] przed wyświetleniem listy opcji konfiguracyjnych serwera.

testprns

Program *testprns* wyszukuje podaną nazwę drukarki w systemowym pliku z parametrami drukarek (*printcap*). Oto składnia programu:

```
testprns nazwa_drukarki [nazwa_pliku_parametrów]
```

Jeśli nie podasz nazwy pliku z parametrami drukarek, Samba spróbuje użyć nazwy określonej w pliku *smb.conf*. Jeśli i tam nie podano nazwy tego pliku, Samba spróbuje użyć pliku */etc/printcap*. Jeśli to się nie uda, program wyświetli komunikat o błędzie.

rpcclient

Jest to nowy klient, który bada interfejsy RPC (*remote procedure call* – zdalne wywołanie procedury) serwera SMB. Podobnie jak program *smbclient*, program *rpcclient* powstał jako narzędzie testowe na potrzeby programistów Samby i prawdopodobnie nieprędko znajdzie inne zastosowanie. Jego składnia to:

```
rpcclient //serwer/udzia□
```

Opcje linii polecenia są tu takie same, jak w programie *smbclient* z Samby 2.0, a operacje, które możesz wypróbować, są wymienione w tabeli D.4.

Tabela D.4. Polecenia programu rpcclient

Polecenie	*Opis*
`regenum nazwa_klucza`	Listowanie Rejestru (klucze, wartości)
`regdeletekey nazwa_klucza`	Usuwa klucz Rejestru
`regcreatekey nazwa_klucza [warto□□ klucza]`	Tworzy klucz Rejestru
`regquerykey nazwa_klucza`	Sprawdza klucz Rejestru
`regdeleteval nazwa_warto□ci`	Usuwa wartość z Rejestru
`regcreateval nazwa_warto□ci typ_warto□ci warto□□`	Tworzy wartość Rejestru
`reggetsec nazwa_klucza`	Pobiera zabezpieczenia z Rejestru
`regtestsec nazwa_klucza`	Testuje zabezpieczenia w Rejestrze
`ntlogin [nazwa_u□ytkownika] [has□o]`	Testuje logowanie w domenie NT
`wksinfo`	Informacje o stacji roboczej
`srvinfo`	Informacje o serwerze
`srvsessions`	Listuje sesje z serwerem
`srvshares`	Listuje udziały w serwerze

Polecenie	*Opis*
`srvconnections`	Listuje połączenia z serwerem
`srvfiles`	Listuje pliki w serwerze
`lsaquery`	Informacje o założeniach systemowych (członek domeny lub serwer)
`lookupsids`	Określa nazwy na podstawie identyfikatorów SID
`ntpas`	Zmienia hasło w bazie danych SAM

tcpdump

Program *tcpdump*, klasyczne narzędzie administratora systemu, wyświetla wszystkie nagłówki pakietów przechodzących przez interfejs sieciowy, które pasują do podanego wyrażenia. Wersja dołączona do dystrybucji Samby jest rozszerzona o obsługę protokołu SMB. *Wyrażenie* może być wyrażeniem logicznym zawierającym łączniki „and", „or" i „not", choć czasem bywa bardzo proste. Na przykład wyrażenie `host garda` wybrałoby wszystkie pakiety wysyłane lub odbierane od hosta `garda`. Wyrażenie składa się zwykle z jednego lub kilku słów kluczowych:

- `host` *nazwa*,
- `net` *numer_sieci*,
- `port` *numer*,
- `src` *nazwa*,
- `dst` *nazwa*.

Najczęściej używane słowa to `src` (źródło), `dst` (cel) i `port`. Na przykład w tej książce użyliśmy polecenia:

```
tcpdump port not telnet
```

Przechwytuje ono wszystkie pakiety z wyjątkiem telnetu; byliśmy zalogowani przez telnet i chcieliśmy oglądać tylko pakiety SMB.

Poniższy przykład przechwytuje tylko ruch między serwerem a komputerami `zuzia` lub `jacek`:

```
tcpdump host serwer and \(zuzia or jacek \)
```

Zalecamy użycie opcji `-s 1500`, abyś mógł przechwycić całe komunikaty SMB, a nie tylko informacje z nagłówka.

Opcje

Polecenie *tcpdump* ma wiele opcji i rozpoznaje wiele typów wyrażeń. Szczegółowy opis zaawansowanych opcji znajdziesz na stronie podręcznika man. Poniżej przedstawiamy najczęściej używane opcje:

`-c` *liczba*
: Powoduje zakończenie pracy programu po odebraniu określonej liczby pakietów.

`-F` *plik*
Odczytuje wyrażenie z podanego pliku i ignoruje wyrażenia określone w linii polecenia.

`-i` *interfejs*
Wymusza monitorowanie określonego interfejsu.

`-r` *plik*
Odczytuje pakiety z określonego pliku (przechwycone opcją `-w`).

`-s` *długość*
Zapisuje określoną liczbę bajtów z każdego pakietu (zamiast domyślnych 68 bajtów).

`-w` *plik*
Zapisuje pakiety w określonym pliku.

E

Pobieranie Samby za pomocą systemu CVS

Ten dodatek omawia pobieranie najnowszej wersji źródłowej Samby za pomocą systemu CVS (*Concurrent Version System*). System CVS to bezpłatne narzędzie do zarządzania wersjami programu, napisane przez Cyclic Software i rozpowszechniane na zasadach Powszechnej Licencji Publicznej GNU. Najnowszą wersję CVS możesz pobrać pod adresem *http://www.cyclic.com*.

CVS działa w oparciu o system GNU RCS (*Revision Control System*). RCS wchodzi w skład wielu dystrybucji Uniksa. Jeśli jednak chcesz pobrać najnowszą wersję RCS, znajdziesz ją pod adresem *http://ftp.gnu.org/gnu/rcs/*.

Jedną z największych zalet CVS jest obsługa zdalnego logowania. Oznacza to, że użytkownicy Internetu z całego świata mogą pobierać i uaktualniać pliki źródłowe każdego projektu, który korzysta z repozytorium CVS. Tak właśnie jest w przypadku Samby. Kiedy zainstalujesz w swoim systemie RCS i CVS, musisz najpierw zalogować się w serwerze plików źródłowych Samby za pomocą następującego polecenia:

```
cvs -d :pserver:cvs@cvs.samba.org:/cvsroot login
```

Informuje ono system CVS, że chcesz połączyć się z serwerem CVS o adresie *cvs.samba.org*. Kiedy się połączysz, możesz pobrać najnowsze drzewo plików źródłowych za pomocą następującego polecenia:

```
cvs -d :pserver:cvs@cvs.samba.org:/cvsroot co samba
```

Spowoduje ono pobranie całej dystrybucji Samby (plik po pliku) do katalogu o nazwie */samba*, który zostanie utworzony na twoim dysku twardym. Katalog ten będzie miał taką samą strukturę, jak źródłowa dystrybucja Samby omówiona w rozdziale 2, *Instalowanie Samby w Uniksie*. Zawiera ona pliki źródłowe i nagłówkowe, dokumentację i przykładowe pliki konfiguracyjne, które ułatwią ci start. Kiedy pobieranie dobiegnie końca, zajrzyj do instrukcji z rozdziału 2, aby skonfigurować i skompilować Sambę w swoim serwerze.

F

Przykładowy plik konfiguracyjny

W tym dodatku zamieszczamy rzeczywisty plik *smb.conf*, aby pokazać, ile spośród dostępnych opcji jest wykorzystywanych w praktyce. Jest to nieco zmodyfikowana wersja pliku, którego używaliśmy w firmie z pięcioma serwerami linuksowymi, pięcioma klientami Windows for Workgroups i pięcioma klientami Windows NT Workstation:

```
# smb.conf -- Serwer plików dla sieci przyklad.COM
[globals]
    workgroup = 1EG_BSC
    interfaces = 10.10.1.14/24
```

Udostępniamy usługi tylko przez jeden z interfejsów serwera. Opcja `interfaces` określa jego adres i maskę sieciową, przy czym `/24` oznacza to samo, co `255.255.255.0`.

```
    comment = Samba wersja %v
    preexec = csh -c 'echo /usr/samba/bin/smbclient \
                     -M %m -I %I' &
```

Używamy polecenia *preexec* w celu zarejestrowania informacji o połączeniach według nazwy komputera (`%m`) oraz adresu IP (`%I`).

```
    # polecenie smbstatus b□dzie wy□wietla□ informacje o bie□□cym statusie
    Samby
    status = yes
    browseable = yes
    printing = bsd

    # nazwa konta, które b□dzie u□ywane podczas dost□pu do udzia□ów
    # oznaczonych opcj□ 'guest ok = yes'
    guest account = samba
```

Domyślnym kontem gościnnym było `nobody`, o identyfikatorze -1, co w jednym z naszych komputerów powodowało pojawianie się komunikatów „your server is being unfriendly", więc utworzyliśmy specjalne konto gościnne Samby na potrzeby przeglądania i drukowania.

```
    # konto superu□ytkownika - uprzywilejowany dost□p do udzia□ów,
    # bez □adnych ogranicze□
    # OSTRZE□ENIE: u□ywa□ tej opcji ostro□nie, poniewa□ superu□ytkownik
```

```
    # b□dzie mógł zmienia□ pliki niezale□nie od ich praw dost□pu
    admin users = root

    # u□ytkownicy NIE maj□cy dost□pu do □ADNYCH us□ug
    invalid users = @wheel, mail, daemon, adt
```

Demony nie mogą używać Samby, tylko ludzie. Opcja `invalid users` zamyka lukę w bezpieczeństwie – uniemożliwia intruzom włamanie się pod przykrywką procesu demona.

```
    # hosty, które MOG□ lub które NIE MOG□ korzysta□ z us□ug
    hosts allow = 10.10.1.
    hosts deny = 10.10.1.6

    # po□o□enie plików blokady
    lock directory = /var/lock/samba/locks

    # pliki dzienników diagnostycznych
    # %m: oddzielny plik dla ka□dej nazwy NetBIOS-owej (ka□dego komputera)
    log file = /var/log/samba/log.%m

    # Do dziennika systemowego wysy□amy komunikaty o priorytetach 0, 1 i 2
    syslog = 2

    # Je□li do serwera zostanie wys□any komunikat WinPopup, kierujemy go do
    # u□ytkownika za pomoc□ poczty e-mail

    message command = /bin/mail -s 'komunikat od #% z komputera %m' \
                          pkelly < %s; rm %s

# -----------------------------------------------------
# [globals] Optymalizowanie wydajno□ci
# -----------------------------------------------------

    # buforowanie redukuje czas wykonania wywo□a□ getwd()
    getwd cache = yes

    socket options = TCP_NODELAY

    # serwer ma sprawdza□, czy klient jest obecny i aktywny,
    # z podan□ poni□ej cz□stotliwo□ci□
    keep alive = 60

    # liczba minut, po której bezczynne po□□czenie zostanie
    # uznane za przerwane
    dead time = 30

    read prediction = yes
    share modes = yes
    max xmit = 17384
    read size = 512
```

Opcje `share modes`, `max xmit` i `read size` są zależne od komputera (patrz dodatek B, *Optymalizowanie wydajności Samby*).

```
    # serwer obs□uguje blokady plików
    locking = yes

    # dosowe atrybuty plików b□d□ odwzorowywane na uniksowe bity
    # praw do wykonania pliku
```

```
    map hidden = yes
    map archive = yes
    map system = yes
```

Trzy opcje `map` będą działały tylko w udziałach, w których tryb tworzenia plików uwzględnia bity praw do wykonania (0111). Nasze udziały `homes` i `printers` nie będą ich honorować, ale udział `[www]` owszem.

```
# -----------------------------------------------------
# [globals] Bezpiecze□stwo i logowania domenowe
# -----------------------------------------------------

    # w po□□czeniach sprawdzane s□ identyfikatory u□ytkownika
    # i grupy, a nie has□a dost□pu do udzia□u
    security = user

    # zmienna logiczna, która okre□la, czy has□a b□d□ szyfrowane
    encrypt passwords = yes
    passwd chat = "*New password:*" %n\r "*New password (again):*" %n\r \
    "*Password changed*"
    passwd program = /usr/bin/passwd %u

    # Samba zawsze b□dzie g□own□ przegl□dark□ lokaln□
    domain master = yes
    preferred master = yes
    os level = 34

    # Aby logowania domenowe dzia□a□y poprawnie, Samba pracuje
    # jako podstawowy kontroler domeny
    domain logons = yes

    # Skrypty logowania, które b□d□ pobierane z serwera i wykonywane
    # w komputerze u□ytkownika (%U) przy ka□dym logowaniu. Skrypty te
    # ustawiaj□ czas, □□cz□ z udzia□ami, wykonuj□ testy antywirusowe itp.
    logon script = skrypty\%U.bat

[netlogon]
    comment = "Us□ugi logowania domenowego"
    path = /u/netlogon
    writable = yes
    create mode = 444
    guest ok = no
    volume = "Siec"
```

Ten udział, omówiony w rozdziale 6, *Użytkownicy, bezpieczeństwo i domeny*, jest wymagany do bezproblemowej pracy Samby w domenie Windows NT.

```
# -----------------------------------------------------
# [homes] Katalogi macierzyste u□ytkowników
# -----------------------------------------------------
[homes]
    comment = "Katalog macierzysty u□ytkownika %u"
    path = /u/uzytkownicy/%u
```

Plik haseł w serwerze Samby określa katalog macierzysty każdego użytkownika jako */home/nazwa_komputera/osoba*, co NFS przekształca na rzeczywistą lokalizację fizyczną w katalogu */u/uzytkownicy*. Opcja `path` w udziale `[homes]` informuje Sambę o rzeczywistym (a nie widzianym przez NFS) położeniu katalogów.

```
    guest ok = no
```

```
    read only = no
    create mode = 644
    writable = yes
    browseable = no

# -----------------------------------------------------
# [printers] Drukarki systemowe
# -----------------------------------------------------
[printers]
    comment = "Drukarki"
    path = /var/spool/lpd/samba
    printcap name = /etc/printcap
    printable = yes
    public = no
    writable = no

    lpq command = /usr/bin/lpq -P%p
    lprm command = /usr/bin/lprm -P%p %j
    lppause command = /usr/sbin/lpc stop %p
    lpresume command = /usr/sbin/lpc start %p

    create mode = 0700

    browseable = no
    load printers = yes

# -----------------------------------------------------
# Indywidualne udzia□y: [programy] [dane] [www]
# -----------------------------------------------------
[programy]
    comment = "Wspó□dzielone programy %T"
    volume = "programy"
```

Tekst „Współdzielone programy” pojawia się w Otoczeniu sieciowym, a `programy` to nazwa wolumenu, który podajemy po to, aby programy instalacyjne rozpoznały etykietę CD-ROM-u, z którego – jak sądzą – są uruchamiane.

```
    path = /u/programy
    public = yes
    writeable = no
    printable = no
    create mode = 664

[cdrom]
    comment = "Uniksowy CD-ROM"
    path = /u/cdrom
    public = no
    writable = no
    printable = no
    volume = "cdrom"

[dane]
    comment = "Katalogi danych %T"
    path = /u/dane
    public = no
    create mode = 770
    writeable = yes
    volume = "dane"
```

```
[nt4]
    comment = "Serwer NT4"
    path = /u/systemy/nt4
    public = yes
    create mode = 770
    writeable = yes
    volume = "serwer_nt4"

[www]
    comment = "System WWW"
    path = /usr/www/http
    public = yes
    create mode = 775
    writeable = yes
    volume = "system_www"
```

Udział `[www]` to katalog uniksowego serwera, zawierający strony WWW. Samba udostępnia ten katalog komputerom PC, aby pracownicy działu graficznego mogli uaktualniać strony WWW.

O autorach

Robert Eckstein zajmuje się niemal wszystkim, co ma związek z komputerami, zaczynając od renderingu i handlu elektronicznego, a kończąc na konstruowaniu kompilatorów i logice rozmytej. Jego przyjaciele zgodnie twierdzą, że Robert spędza zbyt wiele czasu przed ekranem komputera. W wydawnictwie O'Reilly Robert pracuje głównie nad książkami poświęconymi Javie (w szczególności *Java Swing*). Jest także odpowiedzialny za tytuły *XML Pocket Reference* oraz *Webmaster in a Nutshell, 2nd Edition*. W wolnym czasie zdarzało mu się publikować w Sieci sprawozdania z popularnych konferencji. Pisze także artykuły dla magazynu „JavaWorld". Posiada tytuły licencjata z zakresu informatyki i komunikacji, uzyskane na Uniwersytecie Trinity. Kiedyś pracował w firmie ubezpieczeniowej, a przez ostatnie cztery lata zajmował się oprogramowaniem dla telefonii komórkowej Motoroli. Obecnie mieszka w Austin w Teksasie z niedawno poślubioną żoną, Michelle.

David Collier-Brown jest konsultantem do spraw integracji systemów. Obecnie pracuje w grupie optymalizatorów i inżynierów w oddziale Sun Opcom w Toronto. W wolnym czasie niezmordowanie czyta, podlicza wyniki zespołu baseballowego swojej żony, a latem żegluje w okolicach miejskiego portu.

Peter Kelly pracuje na własną rękę jako konsultant systemowy w Toronto, specjalizując się w Internecie i bezpieczeństwie sieci. Obecnie zdaje ostatnie egzaminy potrzebne do uzyskania dyplomu MCSE, ale kiedy tylko może, woli pracować w Linuksie. Kiedy nie pracuje, lubi grać w golfa oraz czytać o bezpieczeństwie, sieciach i Calvinie & Hobbesie.

Informacje licencyjne

Samby nie ponoszą odpowiedzialności za poprawność modyfikacji nie dołączonych do pierwotnie rozpowszechnianej wersji. Nazwisk pierwotnych autorów i nazw „O'Reilly & Associates" oraz „zespół Samby" nie można wykorzystywać jako gwarancji jakości wynikowego dzieła, o ile wcześniej nie uzyska się odpowiedniego zezwolenia. Każdy, kto rozpowszechnia wersję tej książki ze zmianami w tekście, rysunkach lub dowolnym innym elemencie musi udostępnić zmodyfikowaną wersję w standardowym formacie źródłowym wydawnictwu O'Reilly i zespołowi Samby, na takich samych warunkach, na jakich udostępniana jest wersja oryginalna.

Zwykłe połączenie tego dzieła (lub jego części) z innymi dziełami lub programami na tym samym nośniku nie powoduje objęcia innych dzieł niniejszą licencją. Połączone dzieło musi zawierać tę licencję oraz notę informującą o dołączeniu materiału.

Wydawnictwo O'Reilly zachowuje wszystkie prawa autorskie, dopóki nie zaprzestanie drukowania tej książki. Książka ta będzie jednak aktualizowana przez zespół Samby. Wszystkie zmiany dokonane przez wydawnictwo zostaną przekazane zespołowi Samby i vice versa.

TŁUMACZENIA. W przypadku tłumaczeń wydawnictwo O'Reilly zdecyduje, kiedy uaktualnić i ponownie wydać drukowaną wersję. Jeśli wydawnictwo zaniecha drukowania tłumaczeń na dłużej niż sześć miesięcy, prawa autorskie i inne prawa przechodzą na rzecz zespołu Samby.

ROZŁĄCZNOŚĆ. Jeśli dowolna część tej licencji nie będzie miała zastosowania ze względu na obowiązujące prawo, pozostałe części pozostają w mocy.

BRAK GWARANCJI. To dzieło jest udostępniane „tak, jak jest", bez jakiejkolwiek gwarancji, jawnej czy domyślnej, łącznie, choć nie tylko, z domyślnymi gwarancjami przydatności handlowej, przydatności do konkretnych zastosowań i nienaruszalności.

ZALECANA PRAKTYKA. Niezależnie od wymogów tej licencji, prosimy redystrybutorów o spełnienie poniższych zaleceń:

1. Jeśli wydajesz to dzieło drukiem lub na CD-ROM-ie, powiadom o tym autorów za pomocą e-maila co najmniej trzydzieści dni przed planowanym zamrożeniem manuskryptu lub nośnika, aby autorzy mieli czas na dostarczenie zaktualizowanych dokumentów. W powiadomieniu wspomnij o modyfikacjach, którym poddano dokument.
2. Wszystkie istotne modyfikacje (także te polegające na usunięciu tekstu) powinny być czytelnie oznaczone w samym dokumencie lub opisane w załączniku do dokumentu.
3. Choć licencja tego nie nakazuje, uważamy za stosowne zaoferowanie bezpłatnej kopii drukowanej lub elektronicznej wersji tego dzieła pierwotnym autorom i twórcom oprogramowania.
4. Tłumaczenia powinny zawierać tę licencję w języku przekładu.

Indeks

T

U