

**Oracle® Solaris 11.3 ディレクトリサービス
とネームサービスでの作業: DNS と NIS**

ORACLE®

Part No: E62676
2016 年 11 月

Part No: E62676

Copyright © 2002, 2016, Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバース・エンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクルまでご連絡ください。

このソフトウェアまたは関連ドキュメントを、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供する場合は、次の通知が適用されます。

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

このソフトウェアまたはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアまたはハードウェアは、危険が伴うアプリケーション(人的傷害を発生させる可能性があるアプリケーションを含む)への用途を目的として開発されていません。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用する場合、安全に使用するために、適切な安全装置、バックアップ、冗長性(redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、Oracle Corporationおよびその関連会社は一切の責任を負いかねます。

OracleおよびJavaはオラクル およびその関連会社の登録商標です。その他の社名、商品名等は各社の商標または登録商標である場合があります。

Intel, Intel Xeonは、Intel Corporationの商標または登録商標です。すべてのSPARCの商標はライセンスをもとに使用し、SPARC International, Inc.の商標または登録商標です。AMD, Opteron, AMDロゴ、AMD Opteronロゴは、Advanced Micro Devices, Inc.の商標または登録商標です。UNIXは、The Open Groupの登録商標です。

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。適用されるお客様とOracle Corporationとの間の契約に別段の定めがある場合を除いて、Oracle Corporationおよびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。適用されるお客様とOracle Corporationとの間の契約に定めがある場合を除いて、Oracle Corporationおよびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

ドキュメントのアクセシビリティについて

オラクルのアクセシビリティについての詳細情報は、Oracle Accessibility ProgramのWeb サイト(<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>)を参照してください。

Oracle Supportへのアクセス

サポートをご契約のお客様には、My Oracle Supportを通して電子支援サービスを提供しています。詳細情報は(<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>)か、聴覚に障害のあるお客様は (<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>)を参照してください。

目次

このドキュメントの使用	11
1 ネームサービスとディレクトリサービスについて	13
ネームサービスとは	13
Oracle Solaris ネームサービス	20
ネームサービスおよびサービス管理機能	21
DNS ネームサービス	21
マルチキャスト DNS およびサービス検出	22
/etc ファイルネームサービス	22
NIS ネームサービス	22
LDAP ネームサービス	23
ネームサービススイッチ	23
ネームサービスの比較	24
Oracle Solaris ネームサービスへの IPv6 拡張	24
IPv6 の DNS 拡張機能	24
2 ネームサービススイッチについて	25
ネームサービススイッチの概要	25
ネームサービススイッチのデータベースとソース	26
ネームサービススイッチ内の keysevr および publickey エントリ	29
ネームサービススイッチの構成	30
▼ データベースのソースを変更する方法	30
▼ データベースの検索条件を構成する方法	31
▼ すべてのネームデータベースのソースを変更する方法	32
▼ レガシー nsswitch.conf ファイルの使用方法	32
ネームサービススイッチとパスワード情報	33
3 DNS サーバーとクライアントサービスの管理	35
DNS の概要	35

マルチキャスト DNS	35
マルチキャスト DNS サービス検出	36
DNS についての関連資料	36
DNS とサービス管理機能	36
DNS の管理	38
▼ DNS パッケージをインストールする方法	38
▼ DNS サーバーを構成する方法	38
▼ rndc.conf ファイルを作成する方法	39
▼ DNS サーバーのオプションを構成する方法	40
▼ DNS サービスを代替ユーザーとして実行する方法	40
▼ DNS クライアントを有効にする方法	41
▼ DNS 構成の変更を検証する方法	42
DNS サーバーの起動に関する問題のトラブルシューティング	43
マルチキャスト DNS の管理	44
▼ mDNS および DNS サービス検出を有効にする方法	44
DNS のためのリソースの通知	45
DNS 参照	45
DNS ファイル	46
DNS コマンドおよびデーモン	46
BIND が構築されたときに使用されたコンパイルフラグ	48
4 Oracle Solaris Active Directory クライアントの設定	49
nss_ad ネームサービスモジュールの概要	49
▼ nss_ad モジュールを構成する方法	50
パスワード更新	52
AD からの nss_ad ネームサービスモジュールデータの取得	52
passwd 情報の取得	52
shadow 情報の取得	53
group 情報の取得	53
5 ネットワーク情報サービスについて	55
NIS の概要	55
NIS のアーキテクチャー	56
NIS マシンのタイプ	57
NIS サーバー	57
NIS クライアント	58
NIS の要素	58
NIS ドメイン	58

NIS デーモン	59
NIS コマンド	59
NIS マップ	60
NIS のバインド	64
サーバーリストモード	65
ブロードキャストモード	66
6 ネットワーク情報サービスの設定および構成	67
NIS の構成	67
NIS とサービス管理機能	68
NIS ドメインの設計	70
NIS サーバーとクライアントを特定する	70
マスターサーバーの準備	70
マスターサーバーの準備 (タスクマップ)	71
ソースファイルのディレクトリ	71
passwd ファイルと名前空間のセキュリティー	72
▼ 変換用のソースファイルを準備する方法	72
/var/yp/Makefile の準備	74
▼ NIS マスターサーバーパッケージをインストールする方法	75
▼ マスターサーバーを設定する方法	76
▼ 1つのマスターサーバー上で複数の NIS ドメインをサポートする方 法	78
NIS サーバー上の NIS サービスの起動と停止	78
NIS サーバー上の NIS サービスの起動と停止 (タスクマップ)	79
NIS サービスの自動起動	79
▼ NIS サーバーサービスを手動で有効にする方法	79
▼ NIS サーバーサービスを無効にする方法	80
▼ NIS サーバーサービスをリフレッシュする方法	80
NIS スレーブサーバーの設定	81
NIS スレーブサーバーの設定 (タスクマップ)	81
スレーブサーバーを準備する	81
▼ スレーブサーバーを設定する方法	81
▼ スレーブサーバーで NIS を開始する方法	83
▼ 新しいスレーブサーバーを追加する方法	84
NIS クライアントの管理	86
NIS クライアントの管理 (タスクマップ)	86
▼ ブロードキャストモードで NIS クライアントを構成する方法	87

▼ 特定の NIS サーバーを使用して NIS クライアントを構成する方 法	87
▼ NIS クライアントサービスの無効化	88
7 ネットワーク情報サービスの管理	89
パスワードファイルと名前空間のセキュリティー	89
NIS ユーザーの管理	90
▼ NIS ドメインに新しい NIS ユーザーを追加する方法	90
ユーザーパスワードの設定	92
NIS ネットグループ	92
NIS マップに関する作業	93
マップ情報の取得	94
マップのマスターサーバーの変更	95
構成ファイルの変更	96
/var/yp/Makefile の変更および使用	97
Makefile エントリの変更	98
既存のマップの更新	100
▼ デフォルトセットに付いているマップを更新する方法	101
更新されたマップの管理	101
デフォルト以外のマップの変更	104
デフォルト以外のマップを変更するための <code>makedbm</code> コマンドの使用	104
テキストファイルからの新しいマップの作成	105
ファイルをベースとしたマップにエントリを追加する	105
標準入力からマップを作成する	105
標準入力から作成されたマップを更新する	105
NIS サーバーの操作	106
特定の NIS サーバーへのバインド	106
▼ マシンの NIS ドメイン名を設定する方法	107
▼ NIS と DNS を使用してマシンのホスト名とアドレスの検索を構成す る方法	107
NIS サービスをオフにする	108
8 ネットワーク情報システムのトラブルシューティング	111
NIS のバインドに関する問題	111
NIS のバインドに関する問題の現象	111
単一の NIS クライアントに影響する問題	112
複数の NIS クライアントに影響する問題	116

用語集	121
索引	129

このドキュメントの使用

- **概要** – DNS および NIS ネームサービスの概要、これらの使用を計画する方法、DNS および NIS を実装するための手順について説明します。
- **対象読者** – システム管理者。
- **前提知識** – DNS および NIS に関連するネームサービスの概念および用語に対する深い理解。

製品ドキュメントライブラリ

この製品および関連製品のドキュメントとリソースは <http://www.oracle.com/pls/topic/lookup?ctx=E62101-01> で入手可能です。

フィードバック

このドキュメントに関するフィードバックを <http://www.oracle.com/goto/docfeedback> からお聞かせください。

◆◆◆ 第 1 章

ネームサービスとディレクトリサービスについて

この章では、Oracle Solaris 11 リリースに含まれているネームサービスとディレクトリサービスの概要について説明します。また、DNS、NIS、および LDAP ネームサービスについても簡単に説明します。

この章の内容は、次のとおりです。

- [13 ページの「ネームサービスとは」](#)
- [20 ページの「Oracle Solaris ネームサービス」](#)
- [24 ページの「ネームサービスの比較」](#)

ネームサービスとは

ネームサービスは、次のような、格納されている情報の検索を実行します。

- ホスト名とアドレス
- ユーザー名
- パスワード
- アクセス権
- グループ・メンバーシップ
- オートマウントマップ

これらの情報を使用すると、ユーザーは各自のシステムにログインしたり、リソースにアクセスしたり、アクセス許可を付与されたりできます。ネームサービス情報は、ファイル、マップ、または各種の形式のデータベースファイルに格納できます。これらの情報リポジトリは、システムにローカルに格納することも、中央のネットワークベースのリポジトリまたはデータベースに格納することもできます。

注記 - このドキュメントでは、さまざまなネームサービスの構成について説明しています。アクティブなネームサービスの選択はプロファイルを使用して行われます。詳細は、『[Oracle Solaris 11.3 でのネットワークコンポーネントの構成と管理](#)』の第6章、「[Oracle Solaris でのプロファイルベースのネットワーク構成の管理](#)」を参照してください。

中央のネームサービスが存在しない場合、各システムは、これらの情報の独自のコピーを保持する必要があります。すべてのデータを1カ所で管理すれば、管理がより簡単になります。

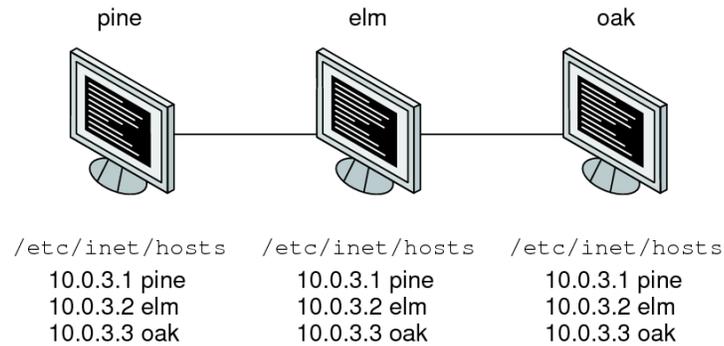
ネームサービスは、どのようなコンピュータネットワークにも欠かせないものです。ネームサービスは、数ある機能の中でも特に、次のアクションを実行する機能を提供します。

- 名前のオブジェクトへの関連付け (バインド)。
- 名前のオブジェクトへの解決
- バインドの削除
- 名前の一覧表示
- 情報の名前の変更

ネットワーク情報サービスを使用すると、システムを数値アドレスではなく、共通名で識別できます。ユーザーが 192.168.0.0 のような煩わしい数値アドレスを覚えたり、入力しようとしたりする必要がなくなるため、通信がより簡単になります。

たとえば、pine、elm、oak という名前の3つのシステムのネットワークがあるとします。pine が elm または oak にメッセージを送信するには、その前に pine がそれらの数値ネットワークアドレスを知っている必要があります。この理由から、pine は、ネットワーク内の (自身を含む) すべてのシステムのネットワークアドレスを格納するファイル /etc/inet/hosts を保持します。同様に、elm と oak が pine と通信したり、互いに通信したりするには、これらのシステムが同様のファイルを保持する必要があります。

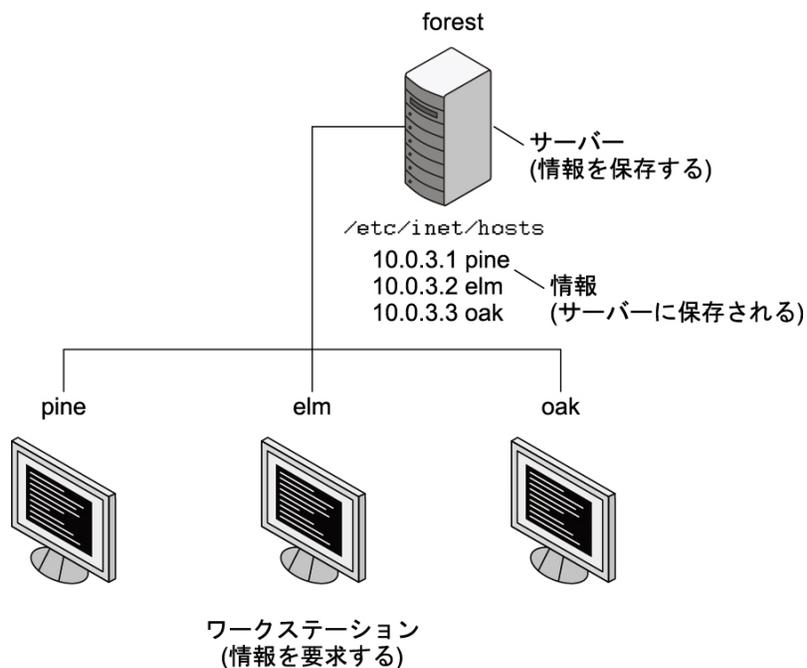
図 1 /etc/inet/hosts のファイルエントリ



アドレスの格納に加えて、システムは、セキュリティー情報、メールデータ、ネットワークサービスの情報なども格納します。ネットワークによって提供されるサービスが増えるにつれ、格納された情報の一覧も増加するため、/etc/inet/hosts のようなファイルのセット全体が各システムに保持される可能性があります。

ネットワーク情報サービスは、どのシステムからでも照会できるサーバー上にネットワーク情報を格納します。これらのシステムは、サーバーのクライアントと呼ばれます。ネットワークについての情報が変更されるたびに、各クライアントのローカルファイルを変更する代わりに、管理者はネットワーク情報サービスが格納する情報だけを更新します。それにより、エラー、クライアント間の不整合、およびタスクのサイズが削減されます。次の図に、クライアントとサーバーの関係を示します。

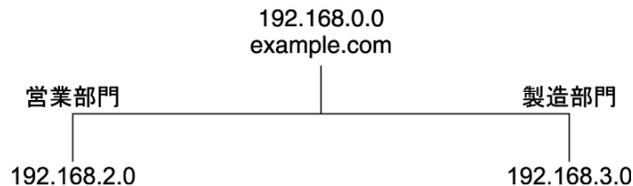
図 2 クライアントサーバーの配置



集中管理されたサービスをネットワーク経由でクライアントに提供しているサーバーのこの配置は、クライアントサーバーコンピューティングと呼ばれます。

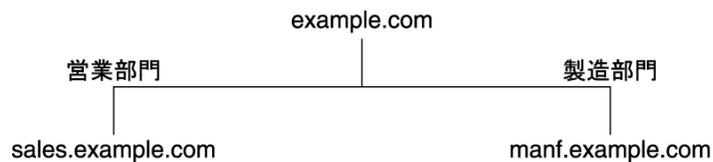
ネットワーク情報サービスの第一の目的は情報の一元管理ですが、もう1つの目的はネットワーク名の簡素化です。たとえば、会社で192.168.0.0のネットワークアドレスと、example.comのドメイン名を持つ、インターネットに接続されたネットワークが設定されているとします。会社には営業(Sales)と製造(Manf)という2つの部門があるため、次の図に示すように、そのネットワークは1つのメインネットワークと部門ごとに1つのサブネットに分割されています。各サブネットには独自のアドレスがあります。

図 3 ドメインと IP アドレスを持つ 2 つのサブネット



各部門はそのネットワークアドレスで識別できますが、ネームサービスによって可能になるわかりやすい名前の方が便利です。

図 4 ドメインとわかりやすい名前を持つ 2 つのサブネット

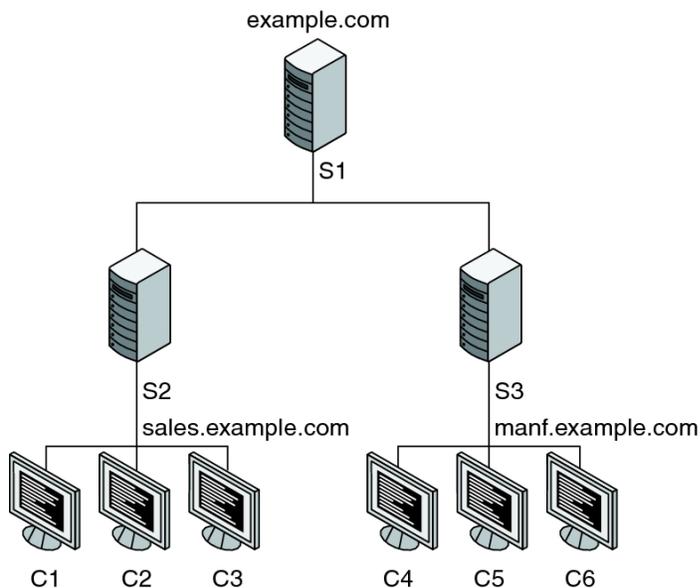


メールやその他のネットワーク通信を `192.168.0.0` にアドレス指定する代わりに、メールを `example.com` にアドレス指定できます。メールを `192.168.2.0` または `192.168.3.0` にアドレス指定する代わりに、メールを `sales.example.com` または `manf.example.com` にアドレス指定できます。

名前はまた、物理アドレスよりもはるかに柔軟です。物理ネットワークは安定したままになる傾向がありますが、会社の組織は変化する傾向があります。

たとえば、次の図に示すように、`example.com` ネットワークが 3 台のサーバー S1、S2、および S3 によってサポートされているとします。それらのサーバーのうちの 2 台 S2 と S3 がクライアントをサポートしているとします。

図 5 2 台のサーバーを含む example.com ドメイン



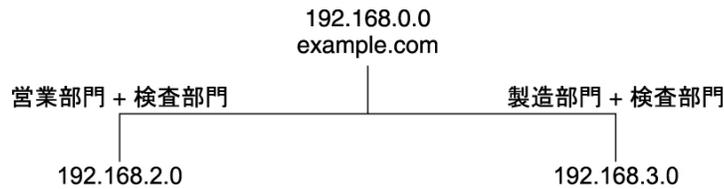
クライアント C1、C2、および C3 は、サーバー S2 から自身のネットワーク情報を取得します。クライアント C4、C5、および C6 は、サーバー S3 から情報を取得します。次の表は、結果として得られるネットワークを要約したものです。この表は、そのネットワークの一般化された表現ですが、実際のネットワーク情報マップとは異なります。

表 1 example.com ネットワークの表現

ネットワークアドレス	ネットワーク名	サーバー	クライアント
192.168.1.0	example.com	S1	
192.168.2.0	sales.example.com	S2	C1、C2、C3
192.168.3.0	manf.example.com	S3	C4、C5、C6

ここで、3 番目の部門 (Test) を作成したとします。この部門はほかの 2 つの部門から一部のリソースを借りましたが、3 番目のサブネットは作成しませんでした。それにより、次の図に示すように、物理ネットワークは会社の組織に対応しなくなります。

図 6 3つのサブネットを含む example.com ドメイン



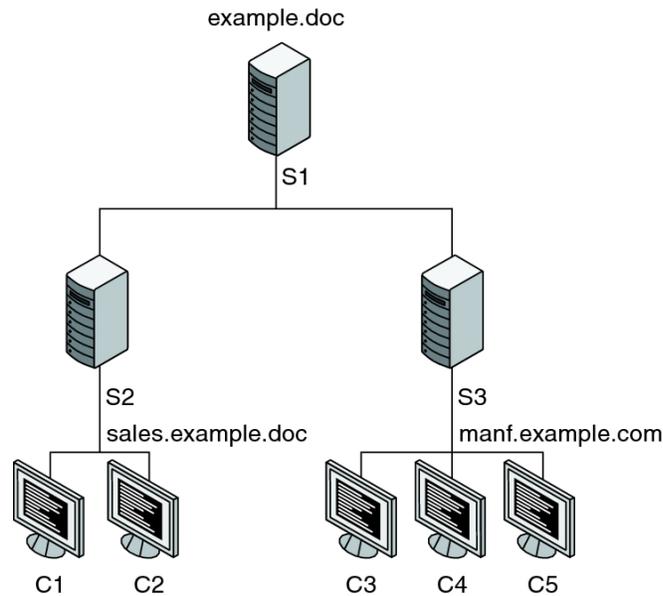
Test 部門のネットワークトラフィックは独自のサブネットが割り当てられず、代わりに 192.168.2.0 と 192.168.3.0 の間で分割されます。ただし、ネットワーク情報サービスでは、Test 部門のトラフィックに独自の専用のネットワークが割り当てられます。

図 7 部門ごとの専用のネットワークを含む example.com ドメイン



このため、次の図に示すように、組織が変化した場合、ネットワーク情報サービスはそのマッピングを変更できます。

図 8 変更された example.com ドメイン



クライアント c1 および c2 は、サーバー s2 から自身の情報を取得します。c3、c4、および c5 は、サーバー s3 から情報を取得します。

組織内での以降の変更には、ネットワーク構造を再編成することなく、ネットワーク情報構造の変更によって対応できます。

Oracle Solaris ネームサービス

Oracle Solaris プラットフォームは、次の各セクションで説明されているネームサービスを提供します。

- [21 ページの「DNS ネームサービス」](#)
- [22 ページの「/etc ファイルネームサービス」](#)
- [22 ページの「NIS ネームサービス」](#)
- 『[Oracle Solaris 11.3 ディレクトリサービスとネームサービスでの作業: LDAP](#)』

最新のほとんどのネットワークは、これらのサービスの2つ以上を組み合わせで使用します。ネームサービススイッチは、特定の検索に使用されるネームサービスを調整します。詳細は、[第2章「ネームサービススイッチについて」](#)を参照してください。

ネームサービスおよびサービス管理機能

Oracle Solaris では、すべてのネームサービスがサービス管理機能 (SMF) によって管理されます。構成情報は構成ファイルではなく、SMF リポジトリに格納されます。SMF が特定のネームサービスと連携して動作する方法の詳細は、この本の個々の章を参照してください。

レガシー構成ファイルは、Oracle Solaris 11 リリースでは、以前の Oracle Solaris リリースとの互換性のためにのみ保持されます。特定のネームサービスに関連する SMF サービスは、レガシー構成ファイルの内容を生成します。ネームサービスの構成のためにこれらのファイルを使用しないでください。代わりに `svcs`、`svcadm`、および `svccfg` などの一般的な SMF コマンドを使用する必要があります。

Oracle Solaris 10 から Oracle Solaris 11 およびそのアップデートリリースにアップグレードすると、システムのネームサービス構成が自動的に SMF に移行されます。ただし、必要に応じて、`nscfg` コマンドを使用して手動での移行を実行できます。詳細は、[nscfg\(1M\)](#) のマニュアルページを参照してください。

DNS ネームサービス

ドメインネームシステム (DNS) は、TCP/IP ネットワーク上に実装された、階層的な分散型データベースです。これは主に、インターネットホスト名に対する IP アドレス、および IP アドレスに対するホスト名を検索するために使用されます。これらのデータはネットワーク全体にわたって分散しており、右から左に読み取られる、ピリオドで区切られた名前を使用して検索されます。DNS はまた、メール交換のルーティング情報、場所のデータ、使用可能なサービスなどの、その他のインターネット関連のホスト情報を格納するためにも使用されます。このサービスの階層的な性質により、ローカルドメインのローカルでの管理が可能になるだけでなく、インターネット、イントラネット、またはその両方に接続されたほかのドメインに国際的に対処できるようになります。

DNS クライアントは、システムに関する情報を 1 つまたは複数のネームサーバーにリクエストし、応答を待ちます。DNS サーバーは、次のソースのいずれかからロードされた情報キャッシュからの要求に応答します。

- DNS マスターサーバー上のファイルまたはサードパーティデータベース

- ネットワーク上で協調している DNS スレーブサーバー上のファイルまたはサードパーティーデータベース
- 以前のクエリーから格納された情報

応答が見つからず、DNS サーバーが問題のドメインを担当していない場合、このサービスは、応答を返すほかの DNS サーバーやキャッシュにホスト名を再帰的にリクエストします (この動作が適切に構成されている場合)。

マルチキャスト DNS およびサービス検出

`svc:network/dns/multicast` サービスは、DNS プロトコルに対する 2 つの拡張機能を管理します。マルチキャスト DNS (mDNS) では、従来の DNS サーバーがインストールされていなかった小規模なネットワーク内に DNS が実装されます。DNS サービス検出 (DNS-SD) は、単純なサービス検出 (ネットワークブラウジング) も提供するようにマルチキャスト DNS を拡張します。詳細は、[35 ページの「マルチキャスト DNS」](#) および [36 ページの「マルチキャスト DNS サービス検出」](#) を参照してください。

mDNS サービスは `.local` のドメイン名を使用するため、競合の可能性を回避するために DNS ではその名前を使用しないでください。

/etc ファイルネームサービス

ホストベースの初期の UNIX ネームシステムは、スタンドアロンの UNIX システム用に開発されたあと、ネットワークで使用するよう適応されました。UNIX オペレーティングシステムの旧版の多くは、現在でも `/etc` 内のローカルファイルのみを使用してすべてのネームデータを管理します。ただし、ローカルファイルによるホスト、ユーザー、その他のネームデータの管理は、大規模で複雑なネットワークには適していません。各ファイルの説明については、該当するマニュアルページを参照してください。たとえば、`/etc/inet/hosts` ファイルは、[hosts\(4\)](#) のマニュアルページで説明されています。

NIS ネームサービス

ネットワーク情報サービス (NIS) は、DNS とは独立して開発されました。DNS は数値 IP アドレスの代わりにホスト名を使うことによって、通信を簡略化します。NIS は、さまざまなネットワーク情報に対する集中管理を可能にすることによって、ネットワーク管理機能を向上させることに焦点を置いています。NIS には、ネットワーク、ホストの名前とアドレス、ユーザー、およびネットワークサービスに関する情報も

格納されます。このようなネットワーク情報の集まりを「NIS の名前空間」と呼びます。

NIS 名前空間情報は NIS マップに格納されています。NIS マップは、UNIX の /etc ファイルやその他の構成ファイルを置き換えます。NIS マップには名前とアドレスよりはるかに多くの情報が格納されるため、NIS 名前空間にはマップの大規模なセットが含まれます。詳細は、[93 ページの「NIS マップに関する作業」](#)を参照してください。

NIS では、DNS と同様のクライアントサーバーの配置が使用されます。複製の NIS サーバーは NIS クライアントへサービスを提供します。主体サーバーはマスターサーバーと呼ばれ、信頼性のために、マスターサーバーにはバックアップ(または、スレーブ)サーバーが割り当てられます。どちらのサーバーも NIS 検索ソフトウェアを使用し、NIS マップを格納します。NIS アーキテクチャーおよび NIS 管理の詳細は、[第6章「ネットワーク情報サービスの設定および構成」](#) および [第7章「ネットワーク情報サービスの管理」](#)を参照してください。

LDAP ネームサービス

LDAP (Lightweight Directory Access Protocol) は、分散型ネームサービスやその他のディレクトリサービスのためにディレクトリサーバーにアクセスするために使用されるセキュアなネットワークプロトコルです。この標準ベースのプロトコルは、階層的なデータベース構造をサポートしています。この同じプロトコルを使用して、UNIX とマルチプラットフォームの両方の環境でネームサービスを提供できます。

Oracle Solaris OS は、Oracle Directory Server Enterprise Edition (従来の Sun Java System Directory Server) およびほかの LDAP Directory Server を使用する場合、LDAP をサポートします。

LDAP と、NIS から LDAP に移行する手順の詳細は、『[Oracle Solaris 11.3 ディレクトリサービスとネームサービスでの作業: LDAP](#)』を参照してください。

シングルサインオンや、Kerberos 認証サービスの設定および保守については、『[Oracle Solaris 11.3 での Kerberos およびその他の認証サービスの管理](#)』の第2章、「[Kerberos サービスについて](#)」を参照してください。

ネームサービススイッチ

ネームサービススイッチは、クライアントが DNS、LDAP、NIS、またはローカルファイルのデータソースを通してネーミング情報を検索できるようにするメカニズムです。このスイッチは、`svc:/system/name-service/switch` サービスによって管理されます。詳細は、[第2章「ネームサービススイッチについて」](#)を参照してください。

ネームサービスの比較

ネームサービス	DNS	NIS	LDAP	ファイル
名前空間	階層	一層	階層	ファイル
データストレージ	ファイル/リソース レコード	2列のマッピング	ディレクトリ (可 変) インデックス化し たデータベース	テキストベースの ファイル
サーバー	マスター/スレーブ	マスター/スレーブ	マスター/複製 複数マスター複製	なし
セキュリティ	DNSSEC (可変)	なし (root またはなし)	Kerberos、 TLS、 SSL (可変)	なし
トランスポート	TCP/IP	RPC	TCP/IP	ファイル入出力
規模	グローバル	LAN	グローバル	ローカルホストの み
データ	ホスト	すべて	すべて	すべて

注記 - DNS は、LDAP やファイルベースのネーミングに対するホストまたはネットワークアドレス検索に使用してください。

Oracle Solaris ネームサービスへの IPv6 拡張

このセクションでは、IPv6 の実装によって導入されたネームサービスの変更について説明します。IPv6 アドレスは、どの Oracle Solaris ネームサービス (NIS、LDAP、DNS、ファイルなど) でも格納できます。また、NIS over IPv6 RPC トランスポートを使用すると、NIS データを検出できます。

IPv6 の DNS 拡張機能

IPv6 固有のリソースレコードである AAAA リソースレコードは、RFC 1886 の IP バージョン 6 をサポートするための DNS 拡張機能で指定されています。AAAA レコードは、ホスト名を 128 ビットの IPv6 アドレスにマッピングします。ポインタレコード (PTR) は、IP アドレスをホスト名にマッピングするために IPv6 で引き続き使用されます。128 ビットアドレスの 32 個の 4 ビットニブルは、IPv6 アドレスでは逆になります。各ニブルは対応する 16 進 ASCII 値に変換されます。その後、ip6.arpa が追加されます。

ネームサービススイッチについて

この章では、ネームサービススイッチについて説明します。ネームサービススイッチは、異なるネームサービスの使用方法を調整するために使います。

この章の内容は、次のとおりです。

- [25 ページの「ネームサービススイッチの概要」](#)
- [30 ページの「ネームサービススイッチの構成」](#)
- [33 ページの「ネームサービススイッチとパスワード情報」](#)

ネームサービススイッチの概要

ネームサービススイッチは、管理者がネットワーク情報のタイプごとに、どのネーム情報サービスまたはソースを使用するかを指定できるようにする、構成可能な選択サービスです。各ネーム情報サービスは、データベースと呼ばれます。ネームサービススイッチは、次のような `getXbyY()` インタフェースのいずれかと呼び出すクライアントアプリケーションによって使用されます。

- `gethostbyname()`
- `getpwuid()`
- `getpwnam()`
- `getaddrinfo()`

各システムは、SMF リポジトリ内に独自の構成を持っています。ネームサービススイッチで定義された各プロパティによって、ホスト、パスワード、グループなどの特定のデータベースが識別されます。各プロパティに割り当てられた値によって、情報をリクエストする先の 1 つまたは複数のソースが一覧表示されます。これらの値にガイダンスまたはオプションが含まれる場合もあります。このガイダンスには、サービスに対して必要な再試行の回数、適用するタイムアウトの種類、サービスが失敗した場合の処理などが含まれている可能性があります。

ネームサービススイッチは、[第3章「DNS サーバーとクライアントサービスの管理」](#)で説明しているように、クライアントの DNS 転送も制御します。DNS 転送によって、クライアントへのインターネットでのアクセスが可能になります。

ネームサービススイッチのデータベースとソース

ネームサービススイッチによってサポートされるデータベースは、SMF サービスを使用して構成します。これらのデータベースのリストを取得するには、`svcfg` コマンドを次の例に示すように使用します。

```
# svcfg -s name-service/switch listprop config
config                application
config/default        astring              files
config/password       astring              "files nis"
config/group           astring              "files nis"
config/host            astring              "files nis"
config/network         astring              "nis [NOTFOUND=return] files"
config/protocol        astring              "nis [NOTFOUND=return] files"
config/rpc             astring              "nis [NOTFOUND=return] files"
config/ether           astring              "nis [NOTFOUND=return] files"
config/netmask         astring              "files nis"
config/bootparam       astring              "nis [NOTFOUND=return] files"
config/publickey       astring              "nis [NOTFOUND=return] files"
config/netgroup        astring              nis
config/automount       astring              "files nis"
config/alias           astring              "files nis"
config/service         astring              "files nis"
config/printer         astring              "user nis"
config/auth_attr       astring              "files nis"
config/prof_attr       astring              "files nis"
config/project         astring              "files nis"
```

注記 - `timezone` テーブルではネームサービススイッチが使用されないため、このテーブルはスイッチのプロパティリストに含まれていません。

SMF の観点から、これらのデータベースは、サービスの構成可能なプロパティであると見なされます。各データベースには、次のタイプの情報が格納されます。

- `alias` – 電子メールアドレスと別名
- `auth_attr` – 承認名と説明
- `automount` – ローカルでマウントされる可能性のあるリモートファイルシステムに関する情報
- `bootparam` – ディスクレスクライアントのブート情報
- `ether` – Ethernet アドレスおよび一致するホスト名
- `group` – ファイルへのアクセスを共有するために使用できるグループに関する情報
- `host` – IP アドレスおよび一致するホスト名
- `netgroup` – 共有 NFS ファイルシステムの情報
- `netmask` – IP サブネットを実装するために使用されるネットワークマスク
- `network` – ネットワークごとの名前と番号
- `password` – ユーザーアカウント情報
- `prof_attr` – 実行プロファイルの名前、説明、およびその他の属性
- `project` – プロジェクト名、一意の識別子、および関連付けられたリソース割り当て

- `protocol` – インターネットプロトコルの名前、プロトコル番号、および別名
- `publickey` – 公開鍵情報
- `rpc` – RPC プログラムの名前と番号
- `service` – インターネットサービスの名前、ポート、およびプロトコル
- `tnrhdb` – Oracle Solaris の Trusted Extensions 機能を使用しているホストのセキュリティ属性
- `tnrhttp` – Trusted Extensions によって使用されるテンプレート

さらに、ネームサービススイッチ内の `default` プロパティによって、ほかでは定義されないすべてのデータベースのソース文字列が定義されます。すべてのデータベースとその情報が `/etc` ディレクトリでローカルに検出されたことを示すため、このプロパティの値は `files` に設定されます。使用可能なソースに基づいて、`default` プロパティの別の構成を設定できます。手順については、[32 ページの「すべてのネームデータベースのソースを変更する方法」](#)を参照してください。

`default` プロパティを使用すると、データベースごとにソースを構成するのではなく、データベースに広く適用されるソースを構成することができます。

データベースのネームサービススイッチで一覧表示できるソースの種類は次のとおりです。

- `ad` – Active Directory サーバー上に格納されているデータベースを識別します。
- `pam_list` – 廃止された `compat` データベースを置き換えます。パスワードやグループ情報に使用すると、`/etc/passwd`、`/etc/shadow`、および `/etc/group` ファイルで旧形式の「+」または「-」構文をサポートできます。
- `dns` – ホスト情報を DNS から取得する必要があることを指定します。
- `files` – クライアントの `/etc` ディレクトリ内に格納されているファイルを指定します (たとえば、`/etc/passwd`)。
- `ldap` – エントリを LDAP ディレクトリから取得する必要があることを指定します。
- `mdns` – mDNS を使用してホスト情報を指定します。
- `nis` – NIS マップを指定します (たとえば、`hosts` マップ)。

注記 - `auto_home` と `auto_master` のテーブルおよびマップのスイッチ検索条件は、`automount` と呼ばれる 1 つのカテゴリに結合されます。

ネームサービススイッチのソース形式

次の検索条件形式を使用すると、1 つまたは複数の情報ソースを選択したり、ソースが使用される順序を指定したりできます。

- **単一ソース** – 情報タイプに 1 つのソースのみが含まれている場合 (`files` など)、スイッチを使用する検索ルーチンは、そのソース内の情報のみを検索します。情報が

見つかった場合、「**success**」というステータスメッセージが返されます。情報が見つからない場合は、検索が停止され、「**success**」以外のステータスメッセージが返されます。ステータスメッセージに基づいて何をするかは、ルーチンによって異なります。

- **複数ソース** — データベースに特定の情報タイプの複数のソースが含まれている場合、スイッチは検索ルーチンに、最初に一覧表示されているソース内を検索するよう指示します。情報が見つかった場合、「**success**」というステータスメッセージが返されます。最初のソースで情報が見つからない場合は、次のソースが検索されます。ルーチンは情報が見つかるか、**return** 処理によって中止されるまで全ソースを検索します。必要な情報がどのソースにもなかったとき、ルーチンは検索を停止し、**non-success** というステータスメッセージを返します。

デフォルトでは、Oracle Solaris 11 リリースで 1 番目のソースは **files** です。この構成によって、一覧表示されている次のソースが使用できない場合でもシステムのフリーズが回避されます。

ネームサービススイッチのステータスメッセージ

ルーチンが情報を見つけた場合、そのルーチンは **success** ステータスメッセージを返します。探している情報が見つからない場合は、3 種類のエラーステータスメッセージのいずれかが返されます。可能性のあるステータスメッセージは次のとおりです。

- **SUCCESS** – 要求されたエントリが指定されたソース内で見つかりました。
- **UNAVAIL** – ソースが応答しないか、または使用できません。つまり、どのデータベースソースも見つからなかったか、またはアクセスできませんでした。
- **NOTFOUND** – ソースが「エントリなし」で応答しました。つまり、データベースにはアクセスしましたが、必要な情報が見つかりませんでした。
- **TRYAGAIN** – ソースはビジー状態にあり、次回は応答する可能性があります。つまり、データベースは見つかりましたが、クエリーに応答できませんでした。

ネームサービススイッチのスイッチアクションオプション

ネームサービススイッチに、次の 2 つのアクションのどちらかでステータスメッセージに応答するよう指示できます。

- **return** – 情報の検索を停止します。
- **continue** – 次のソースを試行します。

さらに、**TRYAGAIN** ステータスメッセージについては、次のアクションを定義できます。

- **forever** – 現在のソースを無期限に再試行します。
- **n** – 現在のソースをさらに *n* 回再試行します。

ネームサービススイッチのデフォルトの検索条件

ネームサービススイッチのステータスメッセージとアクションオプションの組み合わせによって、検索ルーチンが各ステップで何を実行するかが決定されます。ステータスメッセージとアクションオプションの組み合わせによって、検索条件が構成されます。

スイッチのデフォルトの検索条件は、どのソースでも同じです。検索条件のいくつかを次に示します。

- **SUCCESS=return** – 情報の検索を停止します。見つかった情報を使用して処理を続行します。
- **UNAVAIL=continue** – 次のネームサービススイッチソースに進み、検索を続行します。次のソースがなければ、「NOTFOUND」というステータスを返します。
- **NOTFOUND=continue** – 次のネームサービススイッチソースに進み、検索を続行します。次のソースがなければ、「NOTFOUND」というステータスを返します。
- **TRYAGAIN=forever** – 現在のネームサービススイッチソースを無期限に検索します。
- **TRYAGAIN=3** – 現在のソースを3回検索します。TRYAGAIN アクションは、操作を3回再試行したあとで **continue** に遷移し、次のネームサービススイッチソースを検索します。

デフォルトの検索条件は、**STATUS=action** 構文を使用して別の条件を明示的に指定することによって変更できます。手順については、[31 ページの「データベースの検索条件を構成する方法」](#)を参照してください。

注記 - ネームサービススイッチでの検索は、項目が一覧表示されている順序で実行されます。ただし、**passwd -r repository** コマンドを使用して特に指定されていないかぎり、パスワード更新は逆の順序で実行されます。詳細は、[33 ページの「ネームサービススイッチとパスワード情報」](#)を参照してください。

クライアントのライブラリルーチンには、ネームサービススイッチで特定の SMF プロパティまたは **default** の SMF プロパティが定義されていない場合や、プロパティが構文的に正しくない場合に使用される、コンパイル時に組み込まれるデフォルトのエントリが含まれています。通常、これらのコンパイル時に組み込まれるデフォルトは **files** のみです。

ネームサービススイッチ内の **keyserv** および **publickey** エントリ

keyserv デーモンは、**keyserv** が起動されている場合にのみ、ネームサービススイッチ内の **publickey** プロパティを読み取ります。ネームサービススイッチのプロ

パティアーが変更された場合、`keyserv` は、`svcadm refresh svc:/network/rpc/keyserv:default` を使用して `keyserv` デーモンが再起動されるまでその変更を登録しません。プロパティアーの変更が SMF リポジトリにロードされるように、このコマンドはプロパティアーが変更され、`name-service/switch` サービスがリフレッシュされたあとに実行する必要があります。

ネームサービススイッチの構成

ネームサービススイッチを構成する場合は、同時に次のアクションを実行します。

- データベースのソースを示します。
- データベースに複数のソースが存在する場合は、ソースの検索の順序を指定します。
- 対応する検索ステータス用のスイッチアクション (スイッチ条件とも呼ばれます) を定義します。

ネームサービススイッチデータベースまたはプロパティアーは、デフォルト値で構成されます。このセクションの手順では、特定のプロパティアーをさまざまに構成する方法について説明します。

▼ データベースのソースを変更する方法

この手順では、`host` データベースの別のソースを指定する方法について説明します。データベースの元のソース構成がファイルと NIS であるとします。つまり、ホスト検索ではまずローカルファイル、次に NIS が検索されることとなります。`host` 検索で DNS も使用するようにネームサービススイッチを再構成します。

ほかのネームサービススイッチデータベースのソースを構成するためのテンプレートとして、この手順を使用できます。

始める前に ネームサービススイッチの構成がシステムでネームサービスの実際の設定を反映していることを確認します。たとえば、DNS を `host` 検索のソースにする場合は、DNS も構成されている必要があります。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. (オプション) `host` データベースの現在の構成を表示します。

```
# svccfg -s name-service/switch listprop config/host
config                application
config/host          astring          "files nis"
```

3. **host データベースのソース定義を変更します。**

```
# svccfg -s system/name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns nis"
svc:/system/name-service/switch> quit
```

4. **ネームサービススイッチのサービスをリフレッシュします。**

```
# svcadm refresh name-service/switch
```

▼ データベースの検索条件を構成する方法

ネームサービススイッチにはデフォルトの検索条件があります (29 ページの「[ネームサービススイッチのデフォルトの検索条件](#)」を参照)。

この手順では、情報が最初のソースで見つからない場合の `host` データベースの検索メカニズムを再定義する方法について説明します。この検索メカニズムは、次のソースの検索に進むのではなく、停止する必要があります。

1. **管理者になります。**

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. **(オプション) host データベースの現在の構成を表示します。**

```
# svccfg -s name-service/switch listprop config/host
config/host          astring          "files dns nis"
```

3. **情報が最初のソースで見つからないときの host データベース検索の新しい検索条件を作成します。**

```
# svccfg -s system/name-service/switch
svc:/system/name-service/switch> setprop config/host = \
astring: "files [NOTFOUND=return] dns nis"
svc:/system/name-service/switch> quit
```

この構成では、`network` データベースの検索メカニズムでは、`SUCCESS` ステータスおよび `UNAVAILABLE` ステータスの場合にデフォルトの検索条件を使用します。ただし、情報が見つからない場合、検索は即時に停止します。

4. **ネームサービススイッチのサービスをリフレッシュします。**

```
# svcadm refresh name-service/switch
```

▼ すべてのネームデータベースのソースを変更する方法

この手順では、検索のためにネームサービスによって使用されるすべてのデータベースの共通のソースを定義する方法について説明します。デフォルトでは、共通のソースは files です。この手順を使用して別のソースを追加できます。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. (オプション) default プロパティの現在の構成を表示します。

```
# svccfg -s name-service/switch listprop config/default
config/default          astring          files
```

3. NIS をデフォルトのソースとして追加します。

```
# svccfg -s system/name-service/switch
svc:/system/name-service/switch> setprop config/default = astring: "files nis"
svc:/system/name-service/switch> quit
```

4. ネームサービススイッチのサービスをリフレッシュします。

```
# svcadm refresh name-service/switch
```

▼ レガシー nsswitch.conf ファイルの使用方法

既存のネームスイッチ構成で nsswitch.conf ファイルを使用し続けている場合は、この手順を使用します。この手順では、ネームスイッチ構成をファイルから、Oracle Solaris でネームスイッチを構成するためのデフォルトの方法である SMF に移行する方法を示します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. nsswitch.conf ファイルを新しいシステムにコピーします。

ファイル /etc/nsswitch.conf を指定するようにしてください。

3. 情報をファイルから SMF リポジトリにロードします。

```
# nscfg import -f svc:/system/name-service/switch:default
```

4. ネームサービススイッチのサービスをリフレッシュします。

```
# svcadm refresh name-service/switch
```

ネームサービススイッチとパスワード情報

`files` や `nis` などの複数のリポジトリ内にパスワード情報を含め、それらの情報にアクセスできます。パスワード情報の検索順序を確立するには、ネームサービススイッチ内の `config/password` プロパティを使用します。

注記 - システムへのサービス拒否 (DoS) 攻撃の防御に役立てるために、`files` を `passwd` 情報に対するネームサービススイッチ内の最初のソースにしてください。

NIS 環境では、ネームサービススイッチ内の `config/password` プロパティによって、リポジトリが次の順序で一覧表示される必要があります。

```
config/password astring          "files nis"
```

`files` を最初に一覧表示することにより、システムで何らかのネットワークまたはネームサービスの問題が発生した場合でも、`root` ユーザーはほとんどの状況でログインできるようになります。

同じユーザーのために複数のリポジトリを保持しないでください。ほとんどの場合、ネームサービスは、最初の定義のみを検索して返します。重複したエントリは通常、セキュリティーの問題を覆い隠します。

たとえば、`files` とネットワークリポジトリの両方に同じユーザーが含まれている場合は、`config/password` の `name-service/switch` 構成に応じて、あるログイン ID がもう一方より優先して使用されます。特定のシステムの最初に一致した ID が、ログインセッションに使用される ID になります。ある ID が `files` とネットワークリポジトリの両方に含まれていて、セキュリティー上の理由からネットワークリポジトリが無効になっている場合は、その ID が存在して、ネットワーク ID が無効になる前にアクセスされたすべてのシステムがセキュアでなくなり、かつセキュアでない迷惑なアクセスに対して脆弱になっている可能性があります。

◆◆◆ 第 3 章

DNS サーバーとクライアントサービスの管理

この章では、DNS サーバーとクライアントサービスについて説明します。次のトピックが含まれています。

- 35 ページの「DNS の概要」
- 36 ページの「DNS とサービス管理機能」
- 38 ページの「DNS の管理」
- 44 ページの「マルチキャスト DNS の管理」
- 45 ページの「DNS 参照」

DNS の概要

ほとんどのネットワークプロトコルと同様に、DNS には答えを提供するサービスと、そのサービスに照会する DNS クライアントという 2つの部分があります。BIND ソフトウェアとそれに関連付けられたデーモン `named` は、Oracle Solaris オペレーティングシステムのデフォルトの DNS サービスを提供します。BIND は、インターネットシステムコンソーシアム (ISC) によって管理されます。DNS クライアントは、ユーティリティとライブラリの集まりで構成されます。

マルチキャスト DNS

マルチキャスト DNS (mDNS) は、設定が容易であり、かつユーザーがローカルリンク上のシステムを容易に保守できるネームサービスシステムを提供します。同じローカルリンク上の参加しているすべてのネットワークデバイスが、ユニキャストではなく mDNS を使用して標準の DNS 機能を実行するため、ユニキャスト DNS サーバーは必要ありません。管理者にとって、mDNS の主な利点は、ローカルネットワーク上でユニキャスト DNS サーバーを保守する必要がないことです。たとえば、mDNS を使用しているローカルリンク上のシステムへのホスト名から IP アドレスへのリクエストを解決するために、ファイル内のホスト名を更新したり保守したりする必要はありません。

マルチキャスト DNS サービス検出

ネットワークサービスには、出力、ファイル転送、音楽共有や、写真、ドキュメント、その他のファイル共有のためのサーバー、およびほかのローカルデバイスによって提供されるサービスが含まれます。Oracle Solaris での DNS サービス検出のサポートには、アプリケーションがネットワークサービスを通知および検出できるようにするためのオープンソースフレームワークおよびツールが含まれています。

ユーザーにとっては、ネットワークサービス検出により、サービスを手動で検索しなくてもネットワーク上のサービスを参照できるようになるため、コンピューティングがより容易になります。ほかの企業やグループによる既存の標準および作業により、クロスプラットフォームサポートを使用できることが保証されます。

DNS についての関連資料

DNS および BIND の管理については、次のソースを参照してください。

- ISC の Web サイト (<http://www.isc.org>) にある BIND 9 管理者用のリファレンスマニュアル
- /usr/share/doc/bind/migration.txt ファイル内の『BIND 9 Migration Notes』のドキュメント
- BIND の機能、既知のバグと不具合、および ISC の Web サイト (<http://www.isc.org>) 上の資料へのリンク
- 『DNS & BIND 第 5 版』、Paul Albitz および Cricket Liu 著、(オライリー・ジャパン、2008 年)

DNS とサービス管理機能

DNS サーバーデーモン `named` は、SMF を使用して管理されます。SMF の概要については、『Oracle Solaris 11.3 でのシステムサービスの管理』の第 1 章、「サービス管理機能の概要」を参照してください。また、詳細については `svcadm(1M)`、`svcs(1)`、および `svccfg(1M)` の各マニュアルページも参照してください。

SMF サービスを使用して DNS サービスを管理するために必要な次の重要な情報を確認してください。

- このサービスに対する管理操作 (有効化、無効化、再起動など) を実行するには、`svcadm` コマンドを使用します。
- サービス構成に対する何らかの保護を提供するために、`-t` オプションを使用してサービスを一時的に無効にできます。`-t` オプションを使用してサービスを無効に

した場合は、リブートのあと、そのサービスの元の設定が復元されます。-t なしでサービスを無効にした場合は、リブートのあとも、そのサービスは無効のままになります。

- DNS サービスの Fault Managed Resource Identifier (FMRI) は、`svc:/network/dns/server:instance` および `svc:/network/dns/client:instance` です。
- `svcs` コマンドを使用して、DNS サーバーとクライアントのステータスを表示できます。
 - 次の例は、`svcs` コマンドの出力を示しています。

```
# svcs \*dns\*
STATE          STIME    FMRI
disabled       Nov_16   svc:/network/dns/multicast:default
online         Nov_16   svc:/network/dns/server:default
online         Nov_16   svc:/network/dns/client:default
```

- 次の例は、`svcs -l` コマンドの出力を示しています。

```
# svcs -l dns/server
fmri           svc:/network/dns/server:default
name           BIND DNS server
enabled        true
state          online
next_state     none
state_time     Tue Jul 26 19:26:12 2011
logfile        /var/svc/log/network-dns-server:default.log
restarter      svc:/system/svc/restarter:default
contract_id    83
manifest       /lib/svc/manifest/network/dns/server.xml
dependency     require_all/none svc:/system/filesystem/local (online)
dependency     require_any/error svc:/network/loopback (online)
dependency     optional_all/error svc:/network/physical (online)
```

- DNS サービスを異なるオプションで起動するには、`svccfg` コマンドを使用して `svc:/network/dns/server` サービスのプロパティを変更します。例については、[40 ページの「DNS サーバーのオプションを構成する方法」](#)を参照してください。

DNS サーバーデーモン `named` は SMF によって管理されていることから、`named` の異常終了の原因となる予期しないイベントが発生すると、そのサーバーが自動的に再起動されます。さらに、`svcadm` コマンドを使用して、そのサービスを再起動することもできます。`rndc` コマンドを使用することによって可能になる BIND 固有の管理は、SMF では同時に使用できません。

DNS の管理

このセクションでは、DNS を管理するための次のタスクについて説明します。

- 38 ページの「DNS パッケージをインストールする方法」
- 38 ページの「DNS サーバーを構成する方法」
- 39 ページの「rndc.conf ファイルを作成する方法」
- 40 ページの「DNS サーバーのオプションを構成する方法」
- 40 ページの「DNS サービスを代替ユーザーとして実行する方法」
- 41 ページの「DNS クライアントを有効にする方法」
- 43 ページの「DNS サーバーの起動に関する問題のトラブルシューティング」
- 42 ページの「DNS 構成の変更を検証する方法」

▼ DNS パッケージをインストールする方法

Oracle Solaris 11 リリースでは、DNS パッケージは自動的にインストールされます。システムがインストールされたときにこのパッケージが含まれていなかった場合は、この手順を使用してこのパッケージをインストールします。

- DNS パッケージをインストールします。

```
# pkg install pkg:/service/network/dns/bind
```

▼ DNS サーバーを構成する方法

注記 - ルートディレクトリを指定するように named を構成しないでください。よりセキュアなオプションとして、Oracle Solaris ゾーンを作成し、そのゾーン内で動作するように named を構成してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. DNS 構成ファイルを作成して検証します。

named デーモンが起動する前に、有効な構成ファイルが存在する必要があります。このファイルの名前は、デフォルトでは /etc/named.conf です。named の構成は単純です。DNS ルートサーバーにアクセスできる場合は、空のファイルによって、キャッシュのみのサーバーを構成するための十分な情報が提供されます。

```
# touch /etc/named.conf
# named-checkconf -z /etc/named.conf
```

3. (オプション) rndc 構成ファイルを作成します。

このファイルは、DNS サーバーのリモート制御アクセスを構成するために使用します。39 ページの「[rndc.conf ファイルを作成する方法](#)」を参照してください。

4. (オプション) dns/server サービスの構成情報を変更します。

40 ページの「[DNS サーバーのオプションを構成する方法](#)」を参照してください。

5. DNS サービスを起動します。

```
# svcadm enable dns/server
```

▼ rndc.conf ファイルを作成する方法

/etc/rndc.conf ファイルは、rndc コマンドを使用して DNS サーバーデーモン named のリモート制御アクセスを構成するために使用します。この手順では、デフォルトファイルを作成します。詳細は、[rndc.conf\(4\)](#) のマニュアルページを参照してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. rndc 構成ファイルを作成します。

```
# rndc-confgen -a
wrote key file "/etc/rndc.key"
```

3. (オプション) DNS サービスを再起動します。

DNS サーバー構成の一環として rndc.conf ファイルを作成している場合は、DNS サーバー構成が完了するまで DNS サービスの再起動をスキップできます。

```
# svcadm restart dns/server:default
```

▼ DNS サーバーのオプションを構成する方法

svccfg コマンドを使用して、DNS サーバーの構成オプションを設定できます。構成オプションの詳細は、[named\(1M\)](#) のマニュアルページを参照してください。この手順では、named トラフィックの IPv4 トランスポートプロトコルを選択するために options/ip_interfaces オプションを設定する方法について説明します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. dns/server サービスの構成情報を変更します。

```
# svccfg -s dns/server:default
svc:/network/dns/server:default> setprop options/ip_interfaces = "IPv4"
svc:/network/dns/server:default> quit
```

ヒント - 1 つのコマンドで構成情報を変更できます。

```
# svccfg -s dns/server:default setprop options/ip_interfaces=IPv4
```

3. SMF リポジトリを更新し、DNS サービスを有効にします。

```
# svcadm refresh network/dns/server:default
# svcadm enable network/dns/server:default
```

4. (オプション) 変更内容を検証します。

```
# svccfg -s dns/server:default listprop options/ip_interfaces
options/ip_interfaces astring IPv4
```

▼ DNS サービスを代替ユーザーとして実行する方法

この手順では、named デーモンを管理するための関連する承認をユーザーに割り当てる方法について説明します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. 適切な承認を受けている代替ユーザーを指定します。

```
# useradd -c "Trusted DNS administrator user" -s /usr/bin/pfback \
-A solaris.smf.manage.bind user
```

3. ユーザーのサービスプロパティを設定します。

```
# svccfg -s dns/server:default
svc:/network/dns/server:default> setprop start/user = user
svc:/network/dns/server:default> setprop start/group = user
svc:/network/dns/server:default> exit
```

4. 新しいプロセス ID ファイルのためのディレクトリを作成します。

デフォルトのプロセス ID ファイル `/var/run/named/named.pid` を作成するための書き込みアクセス権を持っているのは `root` だけであるため、代わりにファイルを使用するように `named` デーモンを構成する必要があります。

```
# mkdir /var/named/tmp
# chown user /var/named/tmp
```

5. `named.conf` ファイルに次の行を追加することによって、新しいディレクトリを使用するように構成を変更します。

```
# head /etc/named.conf
options {
  directory "/var/named";
  pid-file "/var/named/tmp/named.pid";
};
```

6. SMF リポジトリを更新し、DNS サービスを再起動します。

```
# svcadm refresh svc:/network/dns/server:default
# svcadm restart svc:/network/dns/server:default
```

▼ DNS クライアントを有効にする方法

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. DNS ドメインを構成します。

a. 検索する必要のあるドメインと、DNS ネームサーバーの IP アドレスを一覧表示します。

```
# svccfg -s dns/client
svc:/network/dns/client> setprop config/search = \
astring: ("example.com" "sales.example.com")
svc:/network/dns/client> setprop config/nameserver = \
net_address: (192.168.1.10 192.168.1.11)
```

```
svc:/network/dns/client> quit
```

b. SMF リポジトリを更新します。

```
# svccfg -s dns/client
svc:/network/dns/client> select network/dns/client:default
svc:/network/dns/client:default> refresh
svc:/network/dns/client:default> quit
```

3. DNS を使用するように SMF リポジトリ内のネームサービススイッチ情報を更新します。

```
# svccfg -s system/name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch:default> quit
```

4. DNS クライアントを実行するために必要なサービスを起動します。

```
# svcadm enable network/dns/client
# svcadm enable system/name-service/switch
```

▼ DNS 構成の変更を検証する方法

DNS 構成を変更している場合は、`named-checkzone` コマンドを使用すると `/etc/named.conf` ファイルの構文を検証できます。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. 必要に応じて、構成ファイルを変更します。

この例では、デフォルトディレクトリが変更されます。

```
# echo 'options {directory "/var/named";}' > /etc/named.conf
```

3. ファイルの内容を検証します。

```
# named-checkconf
/etc/named.conf:1: change directory to '/var/named' failed: file not found

/etc/named.conf:1: parsing failed
```

この例では、`/var/named` ディレクトリがまだ作成されていないため、チェックが失敗しました。

4. 報告されたエラーをすべて修正します。

```
# mkdir /var/named
```

5. エラーが報告されなくなるまで、手順 3 と 4 を繰り返します。
6. (オプション) 次のいずれかの方法を使用して、実行中のサービスに変更を反映します。
 - rndc コマンドを使用して、行なった変更に応じて reload または reconfig サブコマンドを使用して構成を更新します。
 - named サービスを再起動します。

```
# svcadm restart svc:/network/dns/server:default
```

DNS サーバーの起動に関する問題のトラブルシューティング

DNS の起動に関する問題が発生している場合は、次のチェックを試してください。まず、管理者になる必要があります。

- DNS サービスのステータスをチェックします。


```
# svcs -x dns/server:default
svc:/network/dns/server:default (BIND DNS server)
State: online since Tue Oct 18 19:35:00 2011
See: named(1M)
See: /var/svc/log/network-dns-server:default.log
Impact: None.
```
- DNS サービスログファイルをチェックします。


```
# tail /var/svc/log/network-dns-server:default.log
```
- システムログメッセージをチェックします。


```
# grep named /var/adm/messages
```
- named デーモンを手動で起動します。named をフォアグラウンドで実行すると、問題が識別しやすくなるように、すべてのログが強制的に標準エラーに出力されます。

```
# named -g
```

問題が解決されたら、保守に必要な状態をクリアします。

```
# svcadm clear dns/server:default
# svcs dns/server:default
STATE          STIME          FMRI
```

```
online          17:59:08 svc:/network/dns/server:default
```

マルチキャスト DNS の管理

mDNS サービスは、システム上で提供されているサービスの可用性を通知するために使用します。mDNS および DNS サービス検出が機能するには、mDNS に参加するすべてのシステム上で mDNS サービスが有効になっている必要があります。次のセクションでは、マルチキャスト DNS (mDNS) および DNS サービス検出を有効にする方法について説明します。

▼ mDNS および DNS サービス検出を有効にする方法

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. mDNS パッケージがインストールされているかどうかをチェックします。

```
# pkginfo -l SUNWdsdu
```

3. mDNS パッケージがまだインストールされていない場合は、インストールします。

```
# pkg install pkg:/service/network/dns/mdns
```

4. name-service/switch サービスの config/host プロパティを、mdns をソースとして含むように変更することによって、ネームサービススイッチ情報を更新します。

例:

```
# /usr/sbin/svccfg -s svc:/system/name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns mdns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch> quit
```

5. mDNS サービスを有効にします。

```
# svcadm enable svc:/network/dns/multicast:default
```

svcadm enable コマンドを使用して mDNS を有効にすると、加えた変更が、アップグレードやリブートを行なった後も維持されるようになります。詳細は、[svcadm\(1M\)](#) のマニュアルページを参照してください。

6. (オプション) mDNS エラーログ `/var/svc/log/network-dns-multicast:default.log` にエラーやメッセージがないかどうかをチェックします。

DNS のためのリソースの通知

`ping` または `traceroute` コマンドと同様に、`dns-sd` コマンドをネットワーク診断ツールとして対話的に使用して、サービスを参照したり、検出したりできます。

`dns-sd` のコマンド行引数やその出力形式は時間とともに変化する場合がありますため、シェルスクリプトからのその起動は予測不可能であり、かつ危険です。さらに、DNS-SD は非同期の性質を持っているため、スクリプト指向のプログラミングにはあまり適していません。

詳細は、[dns-sd\(1M\)](#) のマニュアルページを参照してください。DNS サービスをアプリケーションに組み込むには、[libdns_sd\(3LIB\)](#) のマニュアルページを参照してください。

次の例は、DNS サービス検出を使用してサービスを通知する方法を示しています。

例 1 印刷サービスの通知

この例では、`My Test` という名前のシステムのポート 515 上に LPR 印刷サービスが存在することを通知して、そのサービスを DNS-SD と互換性のある印刷クライアントから使用できるようにします。

```
# dns-sd -R "My Test" _printer._tcp . 515 pdl=application/postscript
```

この登録を有効にするには、LPR サービスがポート 515 上で使用可能である必要があります。

例 2 Web ページの通知

この例では、`My Test` システムのポート 80 上で HTTP サーバーによって処理されている Web ページを通知します。この Web ページは、Safari やその他の DNS-SD と互換性のある Web クライアント内の Bonjour リストに表示されます。

```
# dns-sd -R "My Test" _http._tcp . 80 path=/path-to-page.html
```

DNS 参照

このセクションには、DNS サービスに関連付けられているファイル、デーモン、およびコマンドの表が含まれています。さらに、ISC バージョンの BIND が構築されたときに使用されたいくつかのフラグの表も示されています。

DNS ファイル

次の表では、DNS サービスに関連付けられているファイルについて説明します。

表 2 DNS ファイル

ファイル名	機能
/etc/named.conf	named デーモンの構成情報を提供します。詳細は、 named.conf(4) のマニュアルページを参照してください。
/etc/rndc.conf	rndc コマンドの構成情報を提供します。詳細は、 rndc.conf(4) のマニュアルページを参照してください。

DNS コマンドおよびデーモン

次の表では、DNS サービスに関連付けられているコマンドおよびデーモンについて説明します。詳細は、セクション 1M 内の関連するマニュアルページを参照してください。

表 3 DNS コマンドおよびデーモン

ファイル名	機能
/usr/bin/dns-sd	mDNS サービスによって使用されるリソースを検索または一覧表示します。
/usr/sbin/dig	DNS サーバーに DNS 応答をリクエストします。多くの場合は、トラブルシューティングに使用されます。
/usr/sbin/dnssec-dsfromkey	鍵ファイルから DS RR を生成します。
/usr/sbin/dnssec-keyfromlabel	暗号化デバイスから選択された鍵を取得し、鍵ファイルを構築します。
/usr/sbin/dnssec-keygen	セキュアな DNS およびトランザクション署名 (TSIG) のための鍵と鍵ファイルを作成します。
/usr/sbin/dnssec-signzone	DNS ゾーンに署名します。
/usr/sbin/host	単純な DNS 検索を実行し、多くの場合はホスト名を IP アドレスに、または IP アドレスをホスト名に変換します。
/usr/sbin/named	DNS サーバーデーモン。クライアントからの情報リクエストに応答します。
/usr/sbin/named-checkconf	named.conf ファイルの構文をチェックします。
/usr/sbin/named-checkzone	DNS ゾーンファイルの構文と完全性をチェックします。
/usr/sbin/named-compilezone	DNS ゾーンファイルを変換します。
/usr/sbin/nscfg	レガシーネームサービス構成ファイルと SMF リポジトリの間でネームサービス構成をインポートまたはエクスポートする、レガシーネームサービス構成ユーティリティ。

ファイル名	機能
/usr/sbin/nslookup	非推奨: DNS サーバーに照会します。代わりに <code>dig</code> コマンドを使用してください。
/usr/sbin/nsupdate	DNS 更新リクエストを DNS サーバーに送信します。
/usr/sbin/rndc	DNS サーバーデーモンのリモート制御を提供します。
/usr/sbin/rndc-confgen	<code>rndc</code> コマンドの構成ファイルを生成します。

- /usr/bin/dns-sd – mDNS サービスによって使用されるリソースを検索または一覧表示します。詳細は、[dns-sd\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/dig – DNS サーバーに DNS 応答をリクエストします。多くの場合は、トラブルシューティングに使用されます。詳細は、[dig\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/dnssec-dsfromkey – 鍵ファイルから Delegation Signer リソースレコード (DS RR) を生成します。詳細は、[dnssec-dsfromkey\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/dnssec-keyfromlabel – 暗号化デバイスから選択された鍵を取得し、鍵ファイルを構築します。詳細は、[dnssec-keygen\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/dnssec-keygen – セキュアな DNS およびトランザクション署名 (TSIG) のための鍵と鍵ファイルを作成します。詳細は、[dnssec-keygen\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/dnssec-signzone – DNS ゾーンに署名します。詳細は、[dnssec-signzone\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/host – 単純な DNS 検索を実行し、多くの場合はホスト名を IP アドレスに、または IP アドレスをホスト名に変換します。詳細は、[host\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/named – DNS サーバーデーモン。クライアントからの情報リクエストに応答します。詳細は、[named\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/named-checkconf – `named.conf` ファイルの構文をチェックします。詳細は、[named-checkconf\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/named-checkzone – DNS ゾーンファイルの構文と完全性をチェックします。詳細は、[named-checkzone\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/named-compilezone – DNS ゾーンファイルを変換します。詳細は、[named-compilezone\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/nscfg – レガシーネームサービス構成ファイルと SMF リポジトリの間でネームサービス構成をインポートまたはエクスポートするレガシーネームサービス構成ユーティリティー。詳細は、[nscfg\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/nslookup – 非推奨: DNS サーバーに照会します。代わりに `dig` コマンドを使用してください。
- /usr/sbin/nsupdate – DNS 更新リクエストを DNS サーバーに送信します。詳細は、[nsupdate\(1M\)](#) のマニュアルページを参照してください。

- /usr/sbin/rndc – DNS サーバーデーモンのリモート制御を提供します。詳細は、[rndc\(1M\)](#) のマニュアルページを参照してください。
- /usr/sbin/rndc-confgen – rndc コマンドの構成ファイルを生成します。詳細は、[rndc-confgen\(1M\)](#) のマニュアルページを参照してください。

BIND が構築されたときに使用されたコンパイルフラグ

BIND をコンパイルするために使用されたフラグを表示するには、`named -V` コマンドを使用します。この表は、Oracle Solaris 11 リリース用に ISC バージョンの BIND を構築するときに使用されたいくつかのコンパイルフラグを示しています。

表 4 BIND のコンパイルフラグ

フラグ名	機能
<code>with-openssl</code>	DNSSEC に必要な、暗号化と Secure Sockets Layer (SSL) のサポートを使用して BIND を構築します。
<code>enable-threads</code>	マルチスレッド化を有効にします。
<code>enable-devpoll</code>	多数のファイル記述子に対する高速ポーリングのために <code>/dev/poll</code> ドライバを使用します。
<code>disable-openssl-version-check</code>	OpenSSL が別のダイナミックライブラリによって提供されるため、OpenSSL バージョンのチェックを無効にします。
<code>enable-fixed-rrset</code>	下位互換性に必要な、固定されたリソースレコードセットの順序付けを有効にします。
<code>with-pkcs11</code>	OpenSSL 暗号化ハードウェアサポートの使用を有効にします。

Oracle Solaris Active Directory クライアントの設定

nss_ad ネームサービスモジュールは、passwd、shadow、および group ファイルのバックエンドを提供します。nss_ad モジュールは、Active Directory (AD) とそのネイティブなスキーマをネームサービスとして使用して、AD フォレスト全体にわたるユーザー名やグループ名と ID を解決します。次の項目について説明します。

- 49 ページの「nss_ad ネームサービスモジュールの概要」
- 52 ページの「パスワード更新」
- 52 ページの「AD からの nss_ad ネームサービスモジュールデータの取得」

nss_ad ネームサービスモジュールの概要

いずれかの AD 相互運用性機能 (nss_ad を含む) を使用できるようにするには、その前に Oracle Solaris クライアントが AD ドメインに参加している必要があります。クライアントを AD に参加させるには、kclient ユーティリティを使用します。参加の操作中、kclient は、そのクライアント上に Kerberos v5 を構成します。それ以降は、nss_ad を使用して、ad をサポートされるデータベースの nsswitch.conf ファイル内のソースとして指定することによってネームサービスリクエストを解決できます。nss_ad モジュールは、ホスト資格を使用して AD 内のネームサービス情報を検索します。

nss_ad モジュールは、DNS サーバーのレコードを使用して、ドメインコントローラやグローバルカタログサーバーなどの AD ディレクトリサーバーを自動検出します。そのため、DNS が Oracle Solaris クライアント上で正しく構成されている必要があります。nss_ad モジュールはまた、LDAP v3 プロトコルを使用して AD サーバーのネーミング情報にアクセスします。nss_ad はネイティブな AD スキーマで動作するため、AD サーバーのスキーマに変更は必要ありません。

注記 - nss_ad モジュールは、Windows ユーザーの Oracle Solaris システムへのログインをサポートしていません。このようなログインがサポートされるまで、Windows ユーザーは、引き続き nis や ldap などの従来のバックエンドを使用してログインする必要があります。

nss_ad を使用するには、idmap および svc:/system/name-service/cache サービスを有効にする必要があります。nss_ad モジュールは、idmap サービスを使用して、Windows セキュリティー識別子 (SID)、UNIX ユーザー識別子 (UID)、およびグループ識別子 (GID) の間をマップします。

すべての AD ユーザーおよびグループ名を user@domain や group@domain などのドメイン名で修飾してください。たとえば、abc が domain という名前のドメイン内の有効な Windows ユーザーである場合、getpwnam(abc) は失敗しますが、getpwnam(abc@domain) は成功します。

また、次の追加の規則も nss_ad モジュールに関連します。

- AD と同様に、nss_ad は、ユーザーおよびグループ名の大文字と小文字が区別されないマッチングを実行します。
- nss_ad モジュールは、UTF-8 ロケール、またはユーザーやグループの名前に ASCII 文字のみが存在するドメインでのみ使用してください。
- 既知の SID として、Windows の世界における汎用ユーザーまたは汎用グループを識別する一連の SID があります。これらの SID はドメイン固有ではなく、それらの値はすべての Windows オペレーティングシステムにわたって一定のままです。既知の SID の名前は、文字列 BUILTIN で修飾されます (たとえば、Remote Desktop Users@BUILTIN)。
- nss_ad モジュールは、列挙をサポートしていません。そのため、列挙を使用する getpwent() および getgrent() インタフェースやコマンド (getent passwd や getent group など) は AD から情報を取得できません。
- nss_ad モジュールは現在、passwd および group ファイルのみをサポートしています。nss_ad は、passwd エントリに従うほかのネームサービスデータベース (audit_user や user_attr など) をサポートしていません。ad バックエンドが構成に基づいて処理された場合、そのバックエンドはこれらのデータベースに対して「NOT FOUND」を返します。

▼ nss_ad モジュールを構成する方法

nss_ad モジュールでは、Oracle Solaris クライアントがホスト解決に DNS を使用する必要があります。

1. DNS サービスを構成します。

手順については、[41 ページの「DNS クライアントを有効にする方法」](#)を参照してください。

注記 - AD ドメイン名は、`domain` 指令を使用して、または `search` 指令で指定された一覧内の最初の項目として指定する必要があります。

両方のディレクティブが指定されている場合は、`idmap` 自動検出機能が正しく機能するように、最後に指定されているものが優先されます。

2. **dig** コマンドを使用して、名前と IP アドレスを使用して AD サーバーを解決できることを確認します。

```
# dig -x 192.168.11.22 +short
myserver.ad.example
# dig myserver.ad.example +short
192.168.11.22
```

3. **hosts** に対するネームサービスの一覧に **dns** を追加します。

```
# svccfg -s svc:/system/name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch:default> quit
```

注記 - ホスト解決のために `nis` や `ldap` などの追加のネームサービスを含めるには、それらを `dns` のあとに追加します。

4. **DNS** サービスが有効であり、オンラインになっているかどうかを確認します。

例:

```
# svcs svc:/network/dns/client
STATE STIME FMRI
online Oct_14 svc:/network/dns/client:default
```

5. **kclient** ユーティリティーを使用して、システムを AD ドメインに参加させます。

例:

```
# /usr/sbin/kclient -T ms_ad
```

6. **password** と **group** に対するネームサービスの一覧 **ad** を追加します。

```
# svccfg -s svc:/system/name-service/switch
svc:/system/name-service/switch> setprop config/password = astring: "files nis ad"
svc:/system/name-service/switch> setprop config/group = astring: "files nis ad"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch:default> quit
```

7. **idmap** サービスを有効にします。

```
# svcadm enable idmap
```

8. ネームサービススイッチサービスの **SMF** リポジトリを更新します。

```
# svcadm refresh name-service/switch
```

注記 - ネームサービススイッチがリフレッシュされた場合は常に、nscd モジュールは自動的に再起動します。

9. AD の user や group の情報にアクセスできるかどうかを確認します。

例:

```
# getent passwd 'test_user@example'  
test_user@example:x:2154266625:2154266626:test_user::  
# getent passwd 2154266625  
test_user@example:x:2154266625:2154266626:test_user::
```

パスワード更新

`passwd(4)` のマニュアルページには、ネームサービススイッチ内の `config/passwd` プロパティの有効な形式の一覧が含まれています。これらの構成への `ad` の追加がサポートされています。ただし、`passwd` コマンドを使用した AD ユーザーのパスワードの変更はサポートされていません。パスワード更新中に `passwd` エントリ内に見つかった場合、`ad` はスキップされます。AD ユーザーのパスワードを更新するには、`kpasswd` コマンドを使用します。

ネームサービススイッチ内の既存の有効な `password` および `group` エントリに `ad` の検索順序を追加できます。例:

```
# svccfg -s svc:/system/name-service/switch  
svc:/system/name-service/switch> setprop config/password = astring: "files nis ad"  
svc:/system/name-service/switch> setprop config/group = astring: "files nis ad"  
svc:/system/name-service/switch> select system/name-service/switch:default  
svc:/system/name-service/switch:default> refresh  
svc:/system/name-service/switch:default> quit
```

AD からの `nss_ad` ネームサービスモジュールデータの取得

このセクションでは、`nss_ad` モジュールが AD から対応するデータを取得することによって `passwd`、`shadow`、および `group` ファイルに対するネームサービスリクエストを解決する方法について説明します。

`passwd` 情報の取得

`passwd` エントリには次の構文を使用します。

```
username:password:uid:gid:gecos:home-directory:login-shell
```

詳細は、[passwd\(4\)](#) のマニュアルページを参照してください。

nss_ad モジュールは、AD から passwd 情報を次のように取得します。

- *username* – samAccountName AD 属性の値を使用し、オブジェクトが存在するドメイン名で修飾されます (たとえば、johnd@example.com)。
- *password* – AD オブジェクトではユーザーパスワードが使用できないため、x の値を使用します。
- *uid* – objectSID AD 属性の Windows ユーザーの SID を使用し、それが idmap サービスを使用して UID にマップされます。
- *gid* – Windows ユーザーのプライマリグループ SID を使用し、それが idmap サービスを使用して GID にマップされます。このグループ SID は、primaryGroupID AD 属性の値をドメイン SID に追加することによって取得されます。AD 内のユーザーの場合、primaryGroupID 属性はオプション属性であるため、存在しない可能性があります。この属性が存在しない場合、nss_ad は idmap の対角マッピング機能を使用して、objectSID 属性のユーザー SID をマップします。
- *gecos* – CN AD 属性の値。
- *home-directory* – 値が存在する場合は、homeDirectory AD 属性の値を使用します。それ以外の場合、このフィールドは空のままになります。
- *login-shell* – ネイティブな AD スキーマにはログインシェル属性が存在しないため、このフィールドは空のままになります。

shadow 情報の取得

shadow エントリには次の構文を使用します。

```
username:password:lastchg:min:max:warn:inactive:expire:flag
```

詳細は、[shadow\(4\)](#) のマニュアルページを参照してください。

nss_ad モジュールは、AD から shadow 情報を次のように取得します。

- *username* – samAccountName AD 属性の値を使用し、オブジェクトが存在するドメイン名で修飾されます (たとえば、johnd@example.com)。
- *password* – AD オブジェクトではユーザーパスワードが使用できないため、*NP* の値を使用します。

シャドウフィールドは AD や Kerberos v5 には関係しないため、shadow フィールドの残りは空のままになります。

group 情報の取得

group エントリには次の構文を使用します。

groupname:password:gid:user-list

詳細は、[group\(4\)](#) のマニュアルページを参照してください。

nss_ad モジュールは、AD から情報を次のように取得します。

- *groupname* – samAccountName AD 属性の値を使用し、オブジェクトが存在するドメイン名で修飾されます (たとえば、admins@example.com)。
- *password* – フィールドは、Windows グループにはパスワードがないため、空のままになります。
- *gid* – objectSID AD 属性の Windows グループの SID を使用し、それが idmap サービスを使用して GID にマップされます。
- *user-list* – フィールドは空のままになります。

ネットワーク情報サービスについて

この章では、分散型ネームサービスであるネットワーク情報サービス (NIS) の概要について説明します。NIS は、ネットワークオブジェクトおよびリソースを識別したり、検索したりするためのメカニズムです。NIS は、ネットワーク全体の情報に関する一様なストレージと検索方法を、トランスポートプロトコルやメディアに依存しない形式で提供します。

この章で扱う内容は、次のとおりです。

- 55 ページの「NIS の概要」
- 57 ページの「NIS マシンのタイプ」
- 58 ページの「NIS の要素」
- 64 ページの「NIS のバインド」

NIS の概要

システム管理者は、NIS を実行することによって、マップと呼ばれる管理データベースをさまざまなマスターサーバーおよびスレーブサーバー間に分散させることができます。さらに、これらの管理データベースを一元管理により自動的かつ確実な方法で更新できるため、どの NIS クライアントもネットワーク全体を通して一貫した方法で同じネームサービス情報を共有できます。

NIS は DNS とは独立に開発されており、少し異なるところに焦点を置いています。DNS が、数値 IP アドレスの代わりにマシン名を使用することにより通信をより簡単にすることに焦点を置いているのに対して、NIS はさまざまなネットワーク情報に対する集中管理を可能にすることによって、ネットワーク管理機能を向上させることに焦点を置いています。NIS には、マシン名とアドレスだけでなく、ユーザー、ネットワークそのもの、ネットワークサービスについての情報も格納されます。このネットワーク情報の集まりを、NIS の名前空間と呼びます。

注記 - コンテキストによって、マシン名はホスト名またはマシン名と呼ばれます。この解説では「マシン」名が使われていますが、一部の画面メッセージまたは NIS マップ名では「ホスト」名または「マシン」名が使われています。

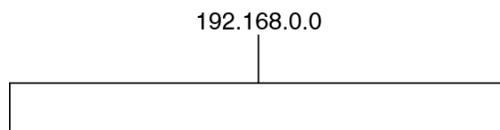
NIS のアーキテクチャー

NIS はクライアントサーバー方式を使用します。NIS サーバーが NIS のクライアントへサービスを提供します。主体サーバーはマスターサーバーと呼ばれ、信頼性のために、複数のバックアップサーバー (または、スレーブサーバー) を割り当てることができます。マスターサーバーとスレーブサーバーはどちらも NIS 情報検索ソフトウェアを使用し、どちらも NIS マップを格納します。

NIS はドメインを使用して、マシン、ユーザー、およびネットワークをその名前空間に配置します。ただし、ドメイン階層は使用しません。NIS の名前空間は一層です。

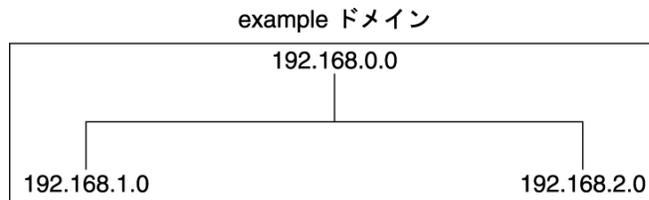
次の図は、階層構造を識別できないドメインを示しています。

図 9 階層構造を識別できないドメイン



次の図は、物理ネットワークを NIS ドメインに配置する方法を示しています。

図 10 NIS 名前空間を持つドメイン



NIS ドメインを、NIS のみを使用してインターネットに直接接続することはできません。ただし、NIS を使用してインターネットへも接続したいと希望する組織では、NIS と DNS を組み合わせることができます。その場合、NIS を使用してすべてのローカル情報を管理し、DNS を使用してインターネットのホストを検索できます。NIS はまた、NIS マップ内で情報が見つからない場合にホスト検索を DNS に転送する転送サービスも提供します。Oracle Solaris システムでは、ホストの検索リクエストを次の方法で転送できるようにネームサービススイッチを設定できます。

- DNS にのみアクセスする
- DNS にアクセスするが、DNS 内にホストが見つからない場合は NIS にアクセスする
- NIS にアクセスするが、NIS によってホストが見つからない場合は DNS にアクセスする

最大の相互運用性を実現するには、ホスト検索のためのサービスとして DNS を推奨します。詳細は、[第2章「ネームサービススイッチについて」](#)を参照してください。

NIS マシンのタイプ

NIS マシンの 3 つのタイプは次のとおりです。

- マスターサーバー
- スレーブサーバー
- NIS サーバーのクライアント

NIS クライアントにはどのマシンでもなれますが、NIS サーバー (マスターまたはスレーブ) となるのはディスクが装備されている物理マシンだけです。一般に NIS サーバーは、多くの場合はそのサーバー自身のクライアントでもあります。

NIS サーバー

NIS サーバーには、マスターサーバーとスレーブサーバーの 2 つの種類があります。マスターサーバーとして指定されているマシンには、NIS 管理者が必要に応じて作成、更新する一群のマップが保存されます。各 NIS ドメインには、マスターサーバーが 1 つだけ存在している必要があります。マスターサーバーは、パフォーマンスの低下を最小限に抑えながら NIS の更新をスレーブサーバーに伝播できます。

ドメイン内の追加の NIS サーバーをスレーブサーバーとして指定できます。スレーブサーバーには、マスターサーバー内の NIS マップの完全なコピーが存在します。マスターサーバー内のマップが更新されると常に、それらの更新がスレーブサーバー間で伝播されます。スレーブサーバーは、マスターサーバーからの要求のオーバーフローに対処して、「サーバー使用不可」エラーを最小限に抑えることができます。

通常、システム管理者はすべての NIS マップに対してマスターサーバーを 1 つ指定します。ただし、各 NIS マップ内ではマスターサーバーのマシン名がエンコードされているので、異なる複数のマップに対して異なる複数のサーバーを、マスターサーバーやスレーブサーバーとして動作するように指定することもできます。管理の複雑さを最小限に抑えるには、1 つのドメイン内で作成されるすべてのマップに対して、マス

ターサーバーを 1 つだけ指定します。この章の例では、1 つのサーバーがドメイン内のすべてのマップのマスターサーバーとなっています。

NIS クライアント

NIS クライアントでは、サーバー上のマップのデータを要求するプロセスが動作します。各 NIS サーバーに保存されている情報は同じであるはずなので、クライアントではマスターサーバーとスレーブサーバーの区別は行われません。

注記 - Oracle Solaris OS は、NIS クライアントとネイティブな LDAP クライアントが同一のクライアントシステム上に共存する構成をサポートしません。

NIS の要素

NIS ネームサービスは、次の要素から構成されています。

- [58 ページの「NIS ドメイン」](#)
- [59 ページの「NIS デーモン」](#)
- [59 ページの「NIS コマンド」](#)
- [60 ページの「NIS マップ」](#)

NIS ドメイン

NIS ドメインは、NIS マップの共通のセットを共有するホストの集まりです。各ドメインにはドメイン名があり、マップの共通のセットを共有する各マシンがそのドメインに属しています。

NIS ドメインと DNS ドメインは、必ずしも同じではありません。一部の環境では、NIS ドメインは、企業全体にわたるネットワークサブネット管理レイアウトに基づいて定義されます。DNS 名と DNS ドメインは、インターネットの DNS ネーミング標準および階層によって定義されます。2 つのネームドメインネームシステムは、同じになるように構成される場合も、されない場合もあります。2 つのサービスのドメイン名は個別に制御されるため、異なった方法で構成される可能性があります。

ある特定のドメインのマップ用のサーバーが同じネットワークまたはサブネット内に存在するかが、どのホストもそのドメインに属することができます。NIS ドメイン

の検索では、リモート手続き呼び出し (RPC) が使用されます。そのため、NIS では、すべてのクライアントと、それらのクライアントに直接サービスを提供するすべてのサーバーマシンが同じアクセス可能なサブネット上に存在する必要があります。各管理サブネットを個別の (企業全体にわたる DNS ドメインとは異なる) NIS ドメインとして、ただし、共通のマスターマシンから管理された共通データベースを使用して管理することは、珍しいことではありません。svc:/network/nis/domain SMF サービスを使用して、NIS ドメイン名およびすべての共有された NIS 構成情報を管理できます。

NIS デーモン

NIS サービスは SMF によって管理されます。このサービスに関する有効化、無効化、再起動などの管理アクションは svcadm コマンドを使用して実行できます。SMF の概要については、『[Oracle Solaris 11.3 でのシステムサービスの管理](#)』の第 1 章、「[サービス管理機能の概要](#)」を参照してください。また、詳細については [svcadm\(1M\)](#) および [svcs\(1\)](#) のマニュアルページも参照してください。次の表では、NIS サービスを提供するデーモンについて説明します。

表 5 NIS デーモン

デーモン	機能
nscd	ほとんどのネームサービスリクエストにキャッシュを提供する、svc:/system/name-service/cache サービスによって管理される NIS クライアントサービス
rpc.yppasswdd	svc:/network/nis/passwd サービスによって管理される NIS パスワード更新デーモン 注記 - rpc.yppasswdd デーモンは、r で始まるすべてのシェルを制限付きと見なします。たとえば、/bin/rksh で作業している場合は、そのシェルから別のシェルへの変更を許可されません。r で始まるシェルを持っているが、そのような制約を受けたくない場合は、 第8章「ネットワーク情報システムのトラブルシューティング」 の回避策を参照してください。
rpc.yppupdated	publickey などのほかのマップを変更する、svc:/network/nis/update サービスによって管理されるデーモン
ybind	svc:/network/nis/client サービスによって管理されるバインドプロセス
ypserv	svc:/network/nis/server サービスによって管理される NIS サーバープロセス
ypxfrd	svc:/network/nis/xfr サービスによって管理される高速マップ転送デーモン

NIS コマンド

次の表では、NIS サービスをサポートするコマンドについて説明します。

表 6 NIS コマンドのサマリー

コマンド	説明
make	このコマンドが /var/yp ディレクトリ内で実行されている場合は、/var/yp/Makefile ファイルを読み取ることによって NIS マップを更新します。make を使うと、入力ファイルに基づいてすべてのマップを更新したり、個々のマップを更新したりできます。NIS の make の機能については、 ypmake(1M) のマニュアルページを参照してください。
makedbm	入力ファイルを取得し、それを dbm.dir および dbm.pag ファイルに変換します。NIS は、有効な dbm ファイルをマップとして使用します。また、マップを構成する鍵と値のペアを表示できるように、makedbm -u を使用してそのマップを分解することもできます。
ypcat	NIS マップの内容を表示します。
ypinit	自動的に入力ファイルから NIS サーバーのマップを作成します。また、クライアント上で初期の /var/yp/binding/domain/ypservers ファイルを作成するためにも使用できます。ypinit は、NIS マスターサーバーと NIS スレーブサーバーをはじめて設定するために使用します。
ypmatch	NIS マップ内の指定された 1 つ以上の鍵の値を出力します。システム管理者は、NIS サーバーマップのバージョンを指定することはできません。
yppoll	指定したサーバー上でどのバージョンの NIS マップが実行されているを示します。yppoll はまた、NIS マップのマスターサーバーを一覧表示します。
yppush	NIS マップの新バージョンを NIS マスターサーバーからそのスレーブサーバーにコピーします。yppush コマンドは、NIS マスターサーバー上で実行できます。
ypset	指定された NIS サーバーにバインドするよう ypbind プロセスに指示します。このコマンドは気軽に使用するものではなく、セキュリティへの影響から、使用しないことが推奨されます。ypbind プロセスの ypset および ypsetme オプションについては、 ypset(1M) および ypbind(1M) のマニュアルページを参照してください。
ypwhich	クライアントが現時点で NIS サービスのためにどの NIS サーバーを使用しているかを示します。-m mapname オプションを指定して起動されると、このコマンドは、どの NIS サーバーが各マップのマスターであるかを示します。-m のみが使用されている場合、このコマンドは、使用可能なすべてのマップの名前と、それらに対応するマスターサーバーを表示します。
ypxfr	NIS 自体をトランスポートメディアとして使用して、NIS マップをリモートサーバーからローカルの /var/yp/domain ディレクトリに取り込みます。ypxfr は対話的に実行するか、または crontab ファイルから定期的に行うことができます。また、ypxfr が ypserv によって呼び出されると、転送が開始されます。

NIS マップ

NIS マップ内の情報は、ndbm 形式で格納されます。マップファイルの形式の詳細は、[ypfiles\(4\)](#) および [ndbm\(3C\)](#) のマニュアルページを参照してください。

NIS マップは、システムのネットワーク間で同じデータを共有できるように、UNIX の /etc データやその他の構成ファイル (passwd、shadow、group など) へのアクセスを拡張します。これらのファイルを共有すると、データファイルの管理上の更新や管理が簡略化されます。NIS は、最小限の作業で配備できます。ただし、より大規模な企業、特にセキュリティ要件を満たす必要のある企業は、代わりに LDAP ネー

ムサービスの使用を考慮するべきです。NIS が動作しているネットワーク上では、各 NIS ドメインの NIS マスターサーバーは、照会されるドメイン内のほかのマシンの NIS マップセットを保持します。NIS スレーブサーバーは、NIS マスターサーバーのマップのコピーを保持します。NIS クライアントマシンは、マスターサーバーまたはスレーブサーバーから名前空間情報を取得できます。

NIS マップは基本的に、2 列のテーブルです。1 つの列は鍵であり、もう 1 つの列は鍵に関連する情報です。NIS は、鍵を検索してクライアントに関する情報を見つけます。各マップでは異なる鍵が使われるので、一部の情報はいくつかのマップに保存されます。たとえば、マシン名とアドレスは、`hosts.byname` および `hosts.byaddr` という 2 つのマップに保存されます。サーバーがマシンの名前を持っており、そのマシンのアドレスを見つける必要がある場合は、サーバーは `hosts.byname` マップを調べます。アドレスがあって、その名前を検索する必要がある場合、サーバーは `hosts.byaddr` マップを調べます。

NIS Makefile は、インストール時に NIS サーバーとして指定されたマシンの `/var/yp` ディレクトリ内に格納されます。そのディレクトリ内で `make` を実行すると、`makedbm` が入力ファイルからデフォルトの NIS マップを作成するか、または変更します。

注記 - マップは必ずマスターサーバー上で作成してください。スレーブサーバーで作成したマップはマスターサーバーに自動的に格納されません。

デフォルトの NIS マップ

Oracle Solaris システムには、NIS マップのデフォルトセットが提供されています。システム管理者は、これらのマップをすべて使用することも一部だけを使用することもできます。また、ほかのソフトウェア製品のインストール時にシステム管理者が作成または追加したマップもすべて NIS で使用できます。

NIS ドメインのデフォルトのマップは、各サーバーの `/var/yp/domain-name` ディレクトリ内にあります。たとえば、`test.com` ドメインに属しているマップは、各サーバーの `/var/yp/test.com` ディレクトリにあります。

次の表では、デフォルトの NIS マップについて説明し、各マップに対応するソースファイル名の一覧を示しています。

表 7 NIS マップの説明

マップ名	対応するソースファイル	説明
<code>audit_user</code>	<code>audit_user</code>	ユーザー監査の事前選択データを含みます。
<code>auth_attr</code>	<code>auth_attr</code>	承認名と説明を含みます。
<code>bootparams</code>	<code>bootparams</code>	ブート中にクライアントに必要なファイルのパス名を含みます (ルート、スワップなど)。

マップ名	対応するソースファイル	説明
ethers.byaddr	ethers	マシン名と Ethernet アドレスを含みます。Ethernet アドレスはマップ内の鍵です。
ethers.byname	ethers	マシン名と Ethernet アドレスを含みます。マシン名はマップ内の鍵です。
exec_attr	exec_attr	プロファイルの実行属性を含みます。
group.bygid	group	グループのセキュリティー情報を含みます。グループ ID はマップ内の鍵です。
group.byname	group	グループのセキュリティー情報を含みます。グループ名はマップ内の鍵です。
hosts.byaddr	hosts	マシン名と IP アドレスを含みます。IP アドレスはマップ内の鍵です。
hosts.byname	hosts	マシン名と IP アドレスを含みます。マシン名はマップ内の鍵です。
mail.aliases	aliases	別名とメールアドレスを含みます。別名はマップ内の鍵です。
mail.byaddr	aliases	メールアドレスと別名を含みます。メールアドレスはマップ内の鍵です。
netgroup.byhost	netgroup	グループ名、ユーザー名、マシン名を含みます。キーはマシン名です。
netgroup.byuser	netgroup	グループ名、ユーザー名、マシン名を含みます。キーはマシン名です。ユーザー名はマップ内の鍵です。
netgroup	netgroup	グループ名、ユーザー名、マシン名を含みます。キーはマシン名です。グループ名はマップ内の鍵です。
netid.byname	passwd、hosts group	マシン名とメールアドレスを含みます (ドメイン名も含む)。netid ファイルがある場合には、ほかのファイルを使用して利用できるデータのほかにそれが参照されます。UNIX スタイルの認証に使用されます。
publickey.byname	publickey	Secure RPC によって使用される公開鍵データベースを含みます。
netmasks.byaddr	netmasks	IP 送信で使用されるネットワークマスクを含みます。アドレスはマップ内の鍵です。
networks.byaddr	networks	システムに認識されているネットワークの名前とその IP アドレスを含みます。アドレスはマップ内の鍵です。
networks.byname	networks	システムに認識されているネットワークの名前とその IP アドレスを含みます。ネットワークの名前はマップ内の鍵です。
passwd.adjunct.byname	passwd および shadow	C2 クライアント用の監査情報と隠蔽されたパスワード情報を含みます。
passwd.byname	passwd および shadow	パスワード情報を含みます。ユーザー名はマップ内の鍵です。
passwd.byuid	passwd および shadow	パスワード情報を含みます。ユーザー ID はマップ内の鍵です。

マップ名	対応するソースファイル	説明
prof_attr	prof_attr	実行プロファイルの属性を含みます。
protocols.byname	protocols	システムに認識されているネットワークプロトコルを含みます。
protocols.bynumber	protocols	システムに認識されているネットワークプロトコルを含みます。プロトコル番号はマップ内の鍵です。
rpc.bynumber	rpc	システムに認識されている RPC のプログラム番号と名前を含みます。RPC プログラム番号はマップ内の鍵です。
services.byname	services	ネットワークに認識されているインターネットサービスを一覧表示します。ポートまたはプロトコルはマップ内の鍵です。
services.byservice	services	ネットワークに認識されているインターネットサービスを一覧表示します。サービス名はマップ内の鍵です。
user_attr	user_attr	ユーザーと役割に関する拡張属性を含みます。
ybservers	なし	ネットワークに認識されている NIS サーバーを一覧表示します。

ageing.byname マッピングには、NIS から LDAP への移行が実装されたときに、yppasswdd デーモンがディレクトリ情報ツリー (DIT) との間のパスワード有効期限情報の読み取りと書き込みを行うために使用する情報が含まれています。パスワード有効期限を使用していない場合は、ageing.byname をマッピングファイルからコメントアウトできます。NIS から LDAP への移行の詳細については、『[Oracle Solaris 11.3 ディレクトリサービスとネームサービスでの作業: LDAP](#)』の第 8 章、「[NIS+ から LDAP への移行](#)」を参照してください。

NIS マップの使用

NIS を使用すると、ネットワークデータベースの更新が /etc ファイルシステムを使用した場合よりはるかに簡単になります。ネットワーク環境を変更するたびに、すべてのマシン上で /etc 管理ファイルを変更する必要はなくなります。

ただし、NIS では、/etc ファイルで提供される以上のセキュリティは追加されません。ネットワークデータベースへのアクセスの制限、SSL を使用したネットワーク経由での検索結果の送信、Kerberos でセキュリティ保護された検索などのより高度な機能の使用といった、追加のセキュリティが必要な場合は、LDAP ネームサービスを使用してください。

たとえば、NIS を実行しているネットワークに新しいユーザーを追加する場合は、マスターサーバー内の入力ファイルを更新し、make コマンドを実行するだけで済みます。このコマンドは、passwd.byname および passwd.byuid マップを自動的に更新します。次に、これらのマップはスレーブサーバーに転送され、NIS ドメインのクライ

アントとそのプログラムのすべてで使用できるようになります。クライアントマシンまたはアプリケーションがユーザー名または UID を使用して情報をリクエストすると、NIS サーバーは必要に応じて `passwd.byname` または `passwd.byuid` マップを参照し、リクエストされた情報をクライアントに送信します。

`ypcat` コマンドを使用して、マップ内の値を表示できます。

```
% ypcat mapname
```

`mapname` は、調べたいマップ名またはその「ニックネーム」です。`ypservers` の場合のように、マップが鍵のみで構成されている場合は、`ypcat -k` を使用します。`ypcat -k` と入力しない場合は、空白行が出力されます。`ypcat` コマンドオプションの詳細は、[ypcat\(1\)](#) のマニュアルページを参照してください。

`ypwhich` コマンドを使用して、どのサーバーが特定のマップのマスターであるかを判定できます。

```
% ypwhich -m mapname
```

ここで、`mapname` は、マスターを検索するマップの名前またはニックネームです。`ypwhich` コマンドの出力には、マスターサーバーの名前が表示されます。詳細は、[ypwhich\(1\)](#) のマニュアルページを参照してください。

NIS マップのニックネーム

ニックネームは、`passwd.byname` に対する `passwd` などの、完全なマップ名に対する別名です。マップの使用可能なニックネームの一覧を取得するには、`ypcat -x` または `ypwhich -x` を入力します。

ニックネームは、`/var/yp/nicknames` ファイル (マップのニックネームのあとにスペースで区切られてマップの完全指定名が含まれる) 内に格納されます。このファイルは変更したり、更新したりできます。ニックネーム数は現在、500 に制限されています。

NIS のバインド

NIS クライアントは、バインドプロセスによって NIS サーバーに接続されます。`svc:/network/nis/client` および `svc:/network/nis/domain` サービスは、このバインドプロセスをサポートしています。いずれかの NIS サービスを動作させるには、これらのサービスを有効にする必要があります。`svc:/network/nis/client` サービスは、サーバーリストまたはブロードキャストの 2 つのモードのどちらかで動作できます。

- サーバーリスト–サーバーリストモードでは、`ypbind` プロセスは、`svc:/network/nis/domain` サービスでドメイン内のすべての NIS サーバーの名前を照

会します。ypbind プロセスは、このファイルに存在するサーバーにだけバインドされます。

svccfg コマンドを使用して、NIS サーバーをいずれかのドメインに追加できます。これらは、svc:/network/nis/domain サービス内の config/ypservers プロパティに追加されます。各プロパティ値が特定の NIS サーバーを表します。

さらに、NIS のバインドが機能するには、svc:/network/nis/domain サービスで指定されているすべてのサーバー名のエントリが /etc/inet/hosts ファイル内に含まれている必要があります。

- ブロードキャスト-ypbind プロセスはまた、RPC ブロードキャストを使用してバインドを開始することもできます。ブロードキャストはそれ以上ルーティングされないローカルサブネットのイベントでしかないため、クライアントと同じサブネット上に少なくとも 1 台のサーバー (マスターまたはスレーブ) が存在する必要があります。マップの伝播はサブネット境界を超えて機能するため、これらのサーバー自体が異なるサブネットワークにわたって存在する可能性があります。サブネット環境での 1 つの一般的な方法は、NIS サーバーとしてサブネットルーターを使用することです。これにより、ドメインサーバーはどちらかのサブネットインタフェース上でクライアントにサービスを提供できます。

ブロードキャストモードは一般に、推奨される動作モードです。ブロードキャストモードでは、追加のホストエントリを指定したり、/etc/inet/hosts を変更したりする必要はありません。

通常、クライアントはサーバーにバインドされたあと、何らかの原因でバインドが変更されるまで、そのサーバーにバインドされたままになります。たとえば、サーバーがサービスを提供できなくなると、このサーバーがサービスを提供していたクライアントは、新しいサーバーにバインドされます。

特定のクライアントに現在、どの NIS サーバーがサービスを提供しているかを判定するには、次のコマンドを使用します。

```
% ypwhich machinename
```

ここで、*machinename* はクライアントの名前です。マシン名が指定されていない場合、ypwhich コマンドはデフォルトで、そのコマンドが実行されているローカルマシンを使用します。

サーバーリストモード

サーバーリストモードでは、バインドプロセスは次のように動作します。

1. NIS マップによって提供される情報を必要としている、NIS クライアントマシン上で実行されているいずれかのプログラムが ypbind デーモンにサーバーの名前をリクエストします。

2. `ypbind` デーモンは、`/var/yp/binding/domainname/ypservers` ファイルでドメインの NIS サーバーのリストを検索します。
3. `ypbind` デーモンが、リスト内の最初のサーバーへのバインドを開始します。先頭サーバーが応答しない場合、`ypbind` はサーバーが見つかるまで、あるいは NIS サーバーリストの最後に達するまで、2 番目以降のサーバーへのバインドを順に試みます。
4. `ypbind` デーモンは、サーバーに関する情報をクライアントプロセスに送信します。次に、クライアントプロセスが直接、サーバーに要求を送信します。
5. NIS サーバー上の `ypserv` デーモンが、該当するマップを調べることによってリクエストを処理します。
6. `ypserv` デーモンは、リクエストされた情報をクライアントに送り返します。

ブロードキャストモード

ブロードキャストモードでは、バインドプロセスは次のように動作します。

1. `ypbind` デーモンを、`broadcast` オプションを設定して起動する必要があります。
2. `ypbind` デーモンが、NIS サーバーを検索するために RPC ブロードキャストを発行します。

注記 - `broadcast` オプションを使用してクライアントをサポートするには、NIS サービスを必要とする各サブネット上に NIS サーバーが存在する必要があります。

3. `ypbind` デーモンが、ブロードキャストに応答した最初のサーバーへのバインドを開始します。
4. `ypbind` デーモンは、サーバーに関する情報をクライアントに送信します。次に、クライアントプロセスが直接、サーバーに要求を送信します。
5. NIS サーバー上の `ypserv` デーモンが、該当するマップを調べることによってリクエストを処理します。
6. `ypserv` デーモンは、リクエストされた情報をクライアントに送り返します。

◆◆◆ 第 6 章

ネットワーク情報サービスの設定および構成

この章では、ネットワーク情報サービス (NIS) の初期の設定と構成について説明します。

この章の内容は、次のとおりです。

- 67 ページの「NIS の構成」
- 70 ページの「NIS ドメインの設計」
- 70 ページの「マスターサーバーの準備」
- 78 ページの「NIS サーバー上の NIS サービスの起動と停止」
- 81 ページの「NIS スレーブサーバーの設定」
- 86 ページの「NIS クライアントの管理」

注記 - 「マシン」名の代わりに「ホスト」名が使われることがあります。この解説では「マシン」名が使われていますが、一部の画面メッセージまたは NIS マップ名では「ホスト」名または「マシン」名が使われています。

NIS の構成

NIS の名前空間を構成する前に、次の操作を行う必要があります。

- NIS ドメインを設計する。詳細は、70 ページの「NIS ドメインの設計」を参照してください。
- NIS を使用する予定のすべてのシステム上に、正しく構成された名前サービススイッチ情報をインストールします。詳細は、第2章「名前サービススイッチについて」を参照してください。

次の表は、NIS を構成するために使用される手順の一覧を示しています。

タスク	説明	説明
変換用のソースファイルを準備します。	ローカルの /etc ファイルから NIS マップを構築する前に、それらの	72 ページの「変換用のソースファイルを準備する方法」

タスク	説明	説明
	ファイルをクリーンアップします。	
マスターサーバーを設定します。	NIS 情報のプライマリソースであるマスターサーバーを作成します。	76 ページの「マスターサーバーを設定する方法」
マスターサーバー上で NIS を起動します。	NIS サーバーからの NIS 情報の提供を開始します。	78 ページの「NIS サーバー上の NIS サービスの起動と停止」
スレーブサーバーを設定します。	NIS 情報のセカンダリソースであるスレーブサーバーを作成します。	81 ページの「スレーブサーバーを設定する方法」
NIS クライアントを設定します。	クライアントが NIS 情報を使用できるようにします。	86 ページの「NIS クライアントの管理」

NIS とサービス管理機能

サービス管理機能 (SMF) は、NIS サービスを管理します。SMF の概要については、『Oracle Solaris 11.3 でのシステムサービスの管理』の第 1 章、「サービス管理機能の概要」を参照してください。詳細は、[svcadm\(1M\)](#) および [svcs\(1\)](#) のマニュアルページを参照してください。

次の一覧は、SMF サービスを使用して NIS を管理するために必要ないくつかの重要な情報の簡単な概要を示しています。

- [svcadm](#) を使用できます。このサービスに対する管理アクション (有効化、無効化、再起動など) は、このコマンドを使用して実行できます。NIS を開始または停止するには、コマンド行から [ypstart](#) および [ypstop](#) も使用できます。詳細については、[ypstart\(1M\)](#) および [ypstop\(1M\)](#) のマニュアルページを参照してください。

ヒント `-t` オプションを使用してサービスを一時的に無効化すると、そのサービス構成に対していくらかの保護を提供できます。`-t` オプションを指定してサービスを無効にした場合、リポート後に元の設定が復元されます。`-t` オプションを指定しないでサービスを無効にした場合、リポート後もそのサービスは無効のままです。

- NIS の障害管理リソース識別子 (FMRI) は次のとおりです。
 - NIS サーバーには `svc:/network/nis/server`
 - NIS クライアントには `svc:/network/nis/client`
 - ドメイン名には `svc:/network/nis/domain`
- [svcs](#) コマンドを使用して、NIS サービスのステータスを照会できます。
 - 次に、[svcs](#) コマンドとその出力の例を示します。

```
$ svcs network/nis/server
STATE      STIME      FMRI
online     Jan_10     svc:/network/nis/server:default
```

```
$ svcs \*nis\*
STATE      STIME      FMRI
online     Oct_09     svc:/network/nis/domain:default
online     Oct_09     svc:/network/nis/client:default
```

- 次に、svcs -l コマンドとその出力の例を示します。

```
$ svcs -l /network/nis/client
fmri       svc:/network/nis/client:default
name       NIS (YP) client
enabled    true
state      online
next_state none
state_time Tue Aug 23 19:23:28 2011
logfile    /var/svc/log/network-nis-client:default.log
restarter  svc:/system/svc/restarter:default
contract_id 88
manifest   /lib/svc/manifest/network/nis/client.xml
manifest   /lib/svc/manifest/network/network-location.xml
manifest   /lib/svc/manifest/system/name-service/upgrade.xml
manifest   /lib/svc/manifest/milestone/config.xml
dependency require_all/none svc:/system/filesystem/minimal (online)
dependency require_all/restart svc:/network/rpc/bind (online)
dependency require_all/restart svc:/network/nis/domain (online)
dependency optional_all/none svc:/network/nis/server (absent)
dependency optional_all/none svc:/network/location:default (online)
dependency optional_all/none svc:/system/name-service/upgrade (online)
dependency optional_all/none svc:/milestone/config (online)
dependency optional_all/none svc:/system/manifest-import (online)
dependency require_all/none svc:/milestone/unconfig (online)
```

- svccfg コーティリティーを使用すると、サービスに関するより詳細な情報を取得できます。svccfg(1M) のマニュアルページを参照してください。
- デーモンの存在は ps コマンドを使用して確認できます。

```
$ ps -ef |grep ypbind
daemon 100813 1 0 Aug 23 ? 0:00 /usr/lib/netsvc/yp/ypbind -broadcast
```

NIS ドメインの設計

NIS サーバーまたはクライアントとしてシステムを構成する前に、NIS ドメインを設計する必要があります。

まず、NIS ドメインに入れるシステムを決めます。NIS ドメインが DNS ドメインをミラー化する必要はありません。DNS ドメインに複数の NIS ドメインを含めることができ、NIS ドメインの外部にあるシステムが DNS ドメイン内に存在できます。

NIS ドメイン名の最大文字数は 256 です。ドメイン名が 32 文字を超えないように制限するとよいでしょう。NIS ドメイン名では大文字と小文字が区別されます。便宜上、NIS ドメイン名の基礎としてインターネットのドメイン名を使用することを選択できます。NIS ドメイン名には大文字が含まれているが、DNS ドメイン名には含まれていない場合、ユーザーが混乱する可能性があるので注意してください。たとえば、インターネットのドメイン名が `example.com` である場合は、NIS ドメインの名前も `example.com` にすることができます。`example.com` を 2 つの NIS ドメインに (たとえば、1 つを営業部門に、もう 1 つを製造部門に) 分割する場合は、1 つのドメインの名前を `sales.example.com` に、もう 1 つのドメインの名前を `manf.example.com` にすることができます。

注記 - 分割された NIS ドメインのマージや管理は非常に困難な場合があるため、NIS ドメインを分割する十分な理由があることを確認してください。

マシンが NIS サービスを使用するには、その前に正しい NIS ドメイン名とマシン名が設定されている必要があります。マシンの名前は、`hostname` コマンドによって設定されます。マシンのドメイン名は、`domainname` コマンドによって設定されます。`hostname` および `domainname` コマンドを使用すると、マシン名と NIS ドメイン名を表示できます。

NIS サーバーとクライアントを特定する

マスターサーバーになる物理マシンを 1 つ選択します。どの物理マシンをスレーブサーバーにするかを決定します。

NIS クライアントになるシステムを決定します。通常は、NIS ドメイン内のすべてのシステムが NIS クライアントとして設定されますが、これは必須ではありません。

マスターサーバーの準備

以降のセクションでは、マスターサーバーのソースファイルと `passwd` ファイルを準備する方法を説明します。

マスターサーバーの準備 (タスクマップ)

次の表に、マスター NIS サーバーを準備するためのタスクを示します。

タスク	説明	説明
変換用のソースファイルを準備します。	NIS マップへの変換用のソースファイルを準備します。	72 ページの「変換用のソースファイルを準備する方法」
NIS マスターサーバーパッケージをインストールします。	NIS マスターサーバーパッケージをインストールします。	75 ページの「NIS マスターサーバーパッケージをインストールする方法」
マスターサーバーを設定します。	NIS マスターサーバーを構成し、マスターサーバーに NIS マップを作成します。	76 ページの「マスターサーバーを設定する方法」
複数の NIS ドメインをサポートします。	複数のドメインをサポートするため、NIS マスターサーバーを設定します。	78 ページの「1つのマスターサーバー上で複数の NIS ドメインをサポートする方法」

ソースファイルのディレクトリ

ソースファイルは通常、マスターサーバー上の `/etc` ディレクトリ内にあります。ただし、マップの内容がマスターサーバー上のローカルファイルの内容と同じになるため、ソースファイルを `/etc` に置いたままにしておくことはお勧めできません。すべてのユーザーがマスターサーバーのマップにアクセスでき、`root` パスワードが `passwd` マップを通してすべての NIS クライアントに渡されるため、これは `passwd` および `shadow` ファイルの特殊な問題です。詳細は、[72 ページの「passwd ファイルと名前空間のセキュリティ」](#)を参照してください。

ただし、ソースファイルをほかのディレクトリに入れた場合は、`/var/yp` 内の `Makefile` の `DIR=/etc` 行を `DIR=/your-choice` に変更する必要があります。*your-choice* はソースファイルを格納するためのディレクトリの名前です。これによって、サーバー上のローカルファイルをクライアント上のファイルのように扱うことができます。(最初に、元の `Makefile` のコピーを保存することをお勧めします。)

さらに、`audit_user`、`auth_attr`、`exec_attr`、および `prof_attr` NIS マップを、デフォルト以外のディレクトリから作成するようにしてください。`RBACDIR=/etc/security` を `RBACDIR=/your-choice` に変更することによって `/var/yp/Makefile` を修正します。

passwd ファイルと名前空間のセキュリティー

セキュリティー上の理由から未承認の root アクセスを防ぐために、NIS のパスワードマップの構築に使用されるファイルには root のエントリを含めないでください。このため、パスワードマップはマスターサーバーの /etc ディレクトリに置かれたファイルから構築しないでください。パスワードマップの構築に使用されるパスワードファイルは、root エントリが削除された上、未承認のアクセスから保護されるディレクトリに置かれている必要があります。

たとえば、マスターサーバーのパスワード入力ファイルは、ファイル自体が別のファイルへのリンクではなく、ファイルの場所が Makefile に指定されている限り、/var/yp/ などのディレクトリに格納されているか、選択したディレクトリに格納されている必要があります。Makefile で指定された構成に従って、正しいディレクトリオプションが自動的に設定されます。



注意 - PWDDIR で指定されたディレクトリ内の passwd ファイルに root のエントリが含まれていないことを確認してください。

ソースファイルが /etc 以外のディレクトリ内に存在する場合は、/var/yp/Makefile 内の PWDIR パスワードマクロを、passwd および shadow ファイルが存在するディレクトリを参照するように変更する必要があります。PWDIR=/etc の行を PWDIR=/your-choice に変更します。ここで、your-choice は、passwd マップのソースファイルを格納するために使用するディレクトリの名前です。

▼ 変換用のソースファイルを準備する方法

この手順では、NIS マップへの変換のためのソースファイルを準備する方法について説明します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. マスターサーバー上のソースファイルをチェックして、そこにシステムが反映されていることを確認します。

次のファイルを確認します。

- audit_user
- auth_attr
- auto.home または auto_home

- auto.master または auto_master
- bootparams
- ethers
- exec_attr
- group
- hosts
- ipnodes
- netgroup
- netmasks
- networks
- passwd
- protocols
- rpc
- service
- shadow
- user_attr

3. **passwd** と **shadow** を除くこれらのすべてのソースファイルを、選択したソースディレクトリにコピーします。

ソースディレクトリは、DIR マクロによって /var/yp/Makefile で定義されています。

4. **passwd** および **shadow** ファイルを、選択したパスワードのソースディレクトリにコピーします。

パスワードのソースディレクトリは、PWDIR マクロによって Makefile で定義されています。

5. **audit_user**、**auth_attr**、**exec_attr**、および **prof_attr** ファイルを、選択した権利ソースディレクトリにコピーします。

権利ソースディレクトリは、RBACDIR マクロによって /var/yp/Makefile に定義されます。必要に応じて、**auth_attr** ファイルをコピーする前に、/etc/security/auth_attr.d ディレクトリ内のファイルの内容をそのファイルのコピーにマージします。同様に、必要に応じて、**exec_attr.d** および **prof_attr.d** ディレクトリ内のファイルを **exec_attr** と **prof_attr** に結合します。



注意 - これらのファイルは、システムがアップグレードされるときは常に再マージする必要があるため、これらのローカルファイルを /etc/security/*.d ディレクトリ内のリリースファイルとは別に保管してください。

6. **/etc/mail/aliases** ファイルを確認します。

ほかのソースファイルと異なり、`/etc/mail/aliases` ファイルは別のディレクトリに移動できません。このファイルは `/etc/mail` ディレクトリに格納されていなければなりません。詳細は、[aliases\(4\)](#) のマニュアルページを参照してください。

注記 - `/var/yp/Makefile` 内の `ALIASES = /etc/mail/aliases` エントリが別の場所を指すようにすることによって、NIS 固有のメールエイリアスファイルを追加できます。そのあとに `make` コマンドを実行すると、`ALIASES` エントリによって `mail.aliases` マップが作成されます。`/etc/nsswitch.conf` ファイルで `files` に加えて `nis` が正しく指定されている場合、`sendmail` サービスは、`/etc/mail/aliases` ファイルに加えてこのマップを使用します。[97 ページの「/var/yp/Makefile の変更および使用」](#) を参照してください。

7. **ソースファイルからすべてのコメントと、その他の余計な行や情報を取り除きます。**
これらの操作は、`sed` または `awk` スクリプト、あるいはテキストエディタを使用して実行できます。`/var/yp/Makefile` は一部のファイルクリーニングを自動的に実行しますが、`make` コマンドを実行する前に、これらのファイルを手動で調べてクリーンアップすることをお勧めします。
8. **すべてのソースファイルのデータが正しい形式になっていることを確認します。**
ソースファイルのデータは、その特定のファイルに適した形式になっている必要があります。該当するマニュアルページを参照して、各ファイルが正しい形式になっていることを確認します。

`/var/yp/Makefile` の準備

ソースファイルをチェックしてソースファイルのディレクトリにコピーしたら、次に、これらのソースファイルを NIS サービスが使用する `ndbm` 形式のマップに変換する必要があります。[How to Set Up the Master Server](#) で説明されているように、これは、マスターサーバー上で呼び出されると [76 ページの「マスターサーバーを設定する方法」](#) によって自動的に実行されます。

`ypinit` スクリプトは、`/var/yp/Makefile` を使用する `make` プログラムを呼び出します。このファイルのデフォルトのコピーが `/var/yp` ディレクトリに用意されており、このコピーには、ソースファイルを目的の `ndbm` 形式のマップに変換するために必要なコマンドが含まれています。

デフォルトの `Makefile` はそのまま使用することも、変更することもできます。デフォルトの `Makefile` を変更する場合は、将来使用する場合に必要に備えて、必ず最初に元のデフォルトの `Makefile` をコピーして保管するようにしてください。次に説明する `Makefile` への修正のうち、必要に応じて 1 つまたは複数を実行します。

- デフォルト以外のマップ

デフォルトではない自分専用のソースファイルを作成して NIS マップに変換する場合は、そのソースファイルを Makefile に追加する必要があります。

- DIR 値

Source Files Directoryで説明しているように、/etc 以外のディレクトリに格納されたソースファイルを 71 ページの「ソースファイルのディレクトリ」で使用する場合は、Makefile の DIR の値を、使用するディレクトリに変更してください。Makefile 内のこの値を変更する場合は、その行をインデントしないでください。

- PWDIR 値

Makefile で、/etc 以外のどこかのディレクトリ内に格納されている passwd、shadow、および adjunct ソースファイルを使用する場合は、Makefile 内の PWDIR の値を、使用するディレクトリに変更する必要があります。Makefile 内のこの値を変更する場合は、その行をインデントしないでください。

- RBACDIR 値

Makefile で、/etc 以外のどこかのディレクトリ内に格納されている audit_user、auth_attr、exec_attr、および prof_attr ソースファイルを使用する場合は、Makefile 内の RBACDIR の値を、使用するディレクトリに変更する必要があります。Makefile 内のこの値を変更する場合は、その行をインデントしないでください。

- ドメインネームリゾルバ

現在のドメインにはないマシンに対して NIS サーバーがドメインネームリゾルバを使用するようにする場合は、Makefile の B= 行をコメントアウトし、B=b 行のコメントを解除します (有効にする)。

Makefile の機能は、all の下にリストされている各データベースの適切な NIS マップを作成することです。データは makedbm を通過したあと、mapname.dir と mapname.pag の 2 つのファイル内に収集されます。この両ファイルは、マスターサーバーの /var/yp/domainname ディレクトリに置かれます。

Makefile は、必要に応じて、/PWDIR/passwd、/PWDIR/shadow、および /PWDIR/security/passwd.adjunct ファイルから passwd マップを構築します。

▼ NIS マスターサーバーパッケージをインストールする方法

通常、NIS マスターサーバーパッケージは、Oracle Solaris リリースの必要に応じてインストールされます。システムをインストールしたときにこのパッケージが含まれていなかった場合は、次の手順を使用してこのパッケージをインストールします。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. NIS マスターサーバーパッケージをインストールします。

```
# pkg install pkg:/service/network/nis
```

▼ マスターサーバーを設定する方法

ypinit スクリプトは、マスターサーバー、スレーブサーバー、およびクライアントを、NIS を使用するように設定します。また、最初に make コマンドを実行して、マスターサーバー上にマップも作成します。

ypinit コマンドを使用して、マスターサーバー上に NIS マップの新しいセットを構築するには、次の手順を完了します。

1. NIS マスターサーバー上の管理者になります。
詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。
2. `/etc/inet/hosts` ファイルを編集します。
各 NIS サーバーのホスト名と IP アドレスを追加します。`IPaddress FQDN-hostname aliases` の形式を使用します。
例:

```
172.16.0.1 master.example.com master
172.16.0.2 slave1.example.com slave1
172.16.0.3 slave2.example.com slave2
```
3. 新しいマップをマスターサーバーに作成します。

```
# /usr/sbin/ypinit -m
```
4. NIS サーバーの名前を入力します。
ypinit から、NIS スレーブサーバーになるほかのマシンのリストを入力するよう求められたら、作業しているサーバーの名前を、`/etc/inet/hosts` ファイルで指定した NIS スレーブサーバーの名前とともに入力します。
5. NIS ドメイン名が設定されていることを確認します。

```
# domainname
example.com
```
6. `y` を入力して、致命的でないエラーが発生した場合にプロセスを停止することを選択します。

致命的でないエラーが発生したときにすぐに処理を終了するか、引き続き処理を継続するかを `ypinit` が尋ねてきたら、**y** と入力します。**y** を選択すると、`ypinit` は最初の問題が発生したときに終了します。その場合は、その問題を解決してから `ypinit` を再起動できます。`ypinit` を初めて実行する場合はこの手順に従うようにしてください。処理を継続する場合は、発生する問題をすべて手動で解決してから `ypinit` を再起動します。

注記 - マップファイルの一部が存在しないと、致命的でないエラーが発生することがあります。これは NIS の機能に影響を与えるエラーではありません。マップが自動的に作成されない場合は、必要に応じて手動で追加します。すべてのデフォルトの NIS マップの詳細については、[61 ページの「デフォルトの NIS マップ」](#) を参照してください。

7. ソースファイルを削除するかどうかを選択します。

`ypinit` コマンドから、`/var/yp/domain-name` ディレクトリ内の既存のファイルを破棄してもよいかどうか尋ねられます。このメッセージは、NIS が以前に設定されている場合にだけ表示されます。通常、以前のインストールのファイルをクリーンアップする場合は、ソースファイルを削除することを選択します。

8. `ypinit` コマンドは、サーバーのリストを作成したあと `make` コマンドを起動します。

このプログラムは、`/var/yp` にある Makefile (デフォルトファイルまたは変更したファイルのどちらか) に含まれている命令を使用します。`make` コマンドは、指定されたファイルから残っているコメント行をすべて削除します。また、指定したファイルに対して `makedbm` を実行して適切なマップを作成し、各マップにマスターサーバー名を設定します。

Makefile によってプッシュされているマップが、マスター上の `domainname` コマンドによって返されたドメイン以外のドメインに対応している場合は、次のように、`ypinit` シェルスクリプト内で変数 `DOM` に正しい識別情報を指定して `make` を起動すれば、それらのマップが正しいドメインに確実にプッシュされます。

```
# make DOM=domain-name passwd
```

このコマンドによって、マスターサーバーが属するドメインではなく目的のドメインに `passwd` マップが転送されます。

9. 必要に応じて、ネームサービススイッチに変更を加えます。

[30 ページの「ネームサービススイッチの構成」](#) を参照してください。

▼ 1つのマスターサーバー上で複数のNISドメインをサポートする方法

通常、NIS マスターサーバーがサポートする NIS ドメインは 1 つだけです。ただし、マスターサーバーを使用して複数のドメインをサポートする場合は、追加のドメインをサポートするようにサーバーを設定するときに、76 ページの「マスターサーバーを設定する方法」で説明されている手順を少し変更する必要があります。

1. NIS マスターサーバー上の管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. NIS ドメイン名を変更します。

```
# domainname sales.example.com
```

3. NIS ファイルを作成します。

```
# make DOM=sales.example.com
```

NIS サーバー上の NIS サービスの起動と停止

これでマスターのマップが作成されたため、マスターサーバー上の NIS デーモンを起動してサービスを開始できます。NIS サービスを有効にすると、NIS サーバー上で `ypserv` および `yplibind` デーモンが起動します。NIS クライアントが NIS サーバーに情報をリクエストした場合、NIS クライアントからの情報リクエストを NIS マップ内で検索したあと、それに応答するデーモンは `ypserv` です。`ypserv` および `yplibind` デーモンは、1 単位として管理されます。

次に、サーバー上で NIS サービスを起動または停止するための 3 つの方法を示します。

- NIS サービスが以前に有効になっていた場合は、ブートプロセス中に SMF サービスが NIS サービスを自動的に起動します。
- `svcadm enable fmri` および `svcadm disable fmri` コマンドの使用は、推奨される手動の方法です。
- `ypstart` および `ypstop` コマンドによって別の手動の方法が提供されますが、SMF を使用して NIS サービスを管理できるように、`svcadm` コマンドを使用することをお勧めします。

NIS サーバー上の NIS サービスの起動と停止 (タスクマップ)

次の表に、`svcadm` コマンドを使用して NIS サービスを起動および停止するためのタスクを示します。

タスク	説明	説明
NIS サーバーサービスを手動で有効にします。	NIS サーバーサービスを有効にするには、 <code>svcadm enable</code> コマンドを使用します。	79 ページの「NIS サーバーサービスを手動で有効にする方法」
NIS サーバーサービスを無効にします。	NIS サーバーサービスを無効にするには、 <code>svcadm disable</code> コマンドを使用します。	80 ページの「NIS サーバーサービスを無効にする方法」
NIS サーバーサービスをリフレッシュします。	NIS サービスをリフレッシュするには、 <code>svcadm refresh</code> コマンドを使用します。	80 ページの「NIS サーバーサービスをリフレッシュする方法」

NIS サービスの自動起動

`svc:/network/nis/server` サービスが有効になっている場合、`ypserv` デーモンはブート時に自動的に起動されます。詳細は、[76 ページの「マスターサーバーを設定する方法」](#)を参照してください。

▼ NIS サーバーサービスを手動で有効にする方法

`svcadm` コマンドを使用するとき、複数のサービスインスタンスを実行している場合のみインスタンス名が必要です。詳細は、[68 ページの「NIS とサービス管理機能」](#)または [svcadm\(1M\)](#) のマニュアルページを参照してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. 必要な NIS サーバーサービスを起動します。

```
# svcadm enable network/nis/domain
```

```
# svcadm enable network/nis/server
```

注記 - NIS サービスはまた、ypstart コマンドを使用して有効にすることもできますが、svcadm コマンドを使用することをお勧めします。

▼ NIS サーバーサービスを無効にする方法

svcadm コマンドを使用するとき、複数のサービスインスタンスを実行している場合のみ特定のインスタンス名が必要です。詳細は、[68 ページの「NIS とサービス管理機能」](#) または [svcadm\(1M\)](#) のマニュアルページを参照してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. 必要な NIS サーバーサービスを無効にします。

```
# svcadm disable network/nis/domain
# svcadm disable network/nis/server
```

注記 - NIS サービスはまた、ypstop コマンドを使用して無効にすることもできます。

▼ NIS サーバーサービスをリフレッシュする方法

この手順では、構成の変更が行われたあとに NIS サーバーサービスをリフレッシュする方法について説明します。

svcadm コマンドを使用するとき、複数のサービスインスタンスを実行している場合のみ特定のインスタンス名が必要です。詳細は、[68 ページの「NIS とサービス管理機能」](#) または [svcadm\(1M\)](#) のマニュアルページを参照してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. 必要な NIS サーバーサービスをリフレッシュします。

```
# svcadm refresh network/nis/domain
# svcadm refresh network/nis/server
```

NIS スレーブサーバーの設定

ネットワークは1つ以上のスレーブサーバーを持つことができます。スレーブサーバーを持つことで、マスターサーバーが利用できない場合にも NIS サービスを継続して利用できます。

NIS スレーブサーバーの設定 (タスクマップ)

次の表に、NIS スレーブサーバーを設定するためのタスクを示します。

タスク	説明	説明
スレーブサーバーを設定します。	システムを NIS スレーブサーバーとして構成します。	81 ページの「スレーブサーバーを設定する方法」
スレーブサーバーで NIS を起動します。	NIS クライアントとサーバーサービスを起動するには、 <code>svcadm</code> コマンドを使用します。	83 ページの「スレーブサーバーで NIS を開始する方法」
新しいスレーブサーバーを追加します。	NIS サービスの起動後に新しいスレーブサーバーを構成します。	84 ページの「新しいスレーブサーバーを追加する方法」

スレーブサーバーを準備する

実際に `ypinit` コマンドを実行してスレーブサーバーを作成する前に、まず `svc:/network/nis/domain` サービスが構成されていることを確認してください。

注記 - NIS ドメイン名は大文字と小文字が区別されますが、DNS ドメイン名は区別されません。

NIS スレーブサーバーを構成する前に、ネットワークが適切に機能していることを確認してください。特に、`sshd` コマンドを使用して NIS マスターサーバーから NIS スレーブにファイルを送信できることを確認してください。

▼ スレーブサーバーを設定する方法

次の手順では、スレーブサーバーを設定する方法について説明します。この手順を、NIS スレーブサーバーとして構成するマシンごとに繰り返してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. /etc/inet/hosts ファイルを編集します。

ほかの各 NIS サーバーの名前と IP アドレスを追加します。IPaddress FQDN-hostname aliases の形式を使用します。

例:

```
172.16.0.1 master.example.com master
172.16.0.2 slave1.example.com slave1
172.16.0.3 slave2.example.com slave2
```

3. スレーブサーバー上の /var/yp ディレクトリに移動します。

注記 - まず、新しいスレーブサーバーを NIS クライアントとして構成して、最初にマスターサーバーから NIS マップを取得できるようにする必要があります。詳細は、[86 ページの「NIS クライアントの管理」](#)を参照してください。NIS マスターマップの変更後に、yppush コマンドを使用して新しいマップを NIS スレーブサーバーに伝搬します。NIS マップの伝播については、[100 ページの「既存のマップの更新」](#)を参照してください。

4. スレーブサーバーを NIS クライアントとして初期化します。

```
# /usr/sbin/ypinit -c
```

ypinit コマンドによって、NIS サーバーのリストを求めるプロンプトが表示されます。まず、作業しているローカルスレーブの名前を入力してから、マスターサーバーの名前のあとに、ドメイン内のほかの NIS スレーブサーバーの名前を入力します。ほかのスレーブサーバーについては、ネットワーク的に物理的にもっとも近いものからもっとも遠いものへの順序に従います。

5. クライアントサービスが実行されているかどうかを判定してから、必要に応じてサービスを起動または再起動します。

```
# svcs \*nis\*
STATE          STIME          FMRI
online         20:32:56      svc:/network/nis/domain:default
online         20:32:56      svc:/network/nis/client:default
```

サービスが online の状態で表示される場合、NIS は実行されています。サービス状態が disabled である場合、NIS は実行されていません。

a. クライアントサービスが実行されている場合は、クライアントサービスを再起動します。

```
# svcadm restart network/nis/domain
```

```
# svcadm restart network/nis/client
```

- b. クライアントサービスが実行されていない場合は、クライアントサービスを起動します。

```
# svcadm enable network/nis/domain  
# svcadm enable network/nis/client
```

6. NIS マスターサーバーが実行されているかどうかを判定してから、必要に応じてサービス起動または再起動します。

```
# svcs network/nis/server  
STATE          STIME      FMRI  
offline        20:32:56  svc:/network/nis/server:default
```

- a. NIS マスターサーバーが実行されている場合は、サービスを再起動します。

```
# svcadm restart network/nis/server
```

- b. NIS マスターサーバーが実行されていない場合は、サービスを起動します。

```
# svcadm enable network/nis/server
```

7. このマシンをスレーブサーバーとして初期化します。

```
# /usr/sbin/ypinit -s master
```

ここで、*master* は既存の NIS マスターサーバーのマシン名です。

▼ スレーブサーバーで NIS を開始する方法

次の手順では、スレーブサーバー上で NIS を起動する方法について説明します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. クライアントサービスを再起動し、すべての NIS サーバードキュメントを起動します。

```
# svcadm restart network/nis/domain  
# svcadm restart network/nis/client  
# svcadm enable network/nis/server
```

▼ 新しいスレーブサーバーを追加する方法

NIS が実行されたあと、ypinit コマンドに指定した最初のリストに含めなかった NIS スレーブサーバーの作成が必要になることがあります。新しい NIS スレーブサーバーを追加するには、この手順を使用します。

1. **NIS マスターサーバー上の管理者になります。**

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. **NIS ドメインディレクトリに移動します。**

```
# cd /var/yp/domainname
```

3. **ypservers ファイルを分解します。**

```
# makedbm -u ypservers >/tmp/temp_file
```

makedbm コマンドは、ypservers を ndbm フォーマットから一時 ASCII ファイル /tmp/temp_file に変換します。

4. **/tmp/temp_file ファイルを編集します。**

つまり、新しいスレーブサーバー名をサーバーリストに追加します。そのあと、このファイルを保存して閉じます。

5. **入力ファイルとして temp_file を、出力ファイルとして ypservers を指定して makedbm コマンドを実行します。**

```
# makedbm /tmp/temp_file ypservers
```

これにより、makedbm コマンドは ypservers を変換して ndbm 形式に戻します。

6. **ypservers マップが正しいことを確認します。**

ypservers の ASCII ファイルは存在しないため、スレーブサーバー上で次のように入力します。

```
slave3# makedbm -u ypservers
```

makedbm コマンドは、画面上に ypservers 内の各エントリを表示します。

注記 - ypservers にマシン名が存在しない場合は、ypservers はマップファイルの更新を受信しません。これは、yppush がこのマップでスレーブサーバーリストを調べるからです。

7. **新しい NIS スレーブサーバー上の管理者になります。**

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

8. NIS ドメイン名が設定されていることを確認します。

```
# domainname
example.com
```

9. 新しいスレーブサーバーの NIS ドメインディレクトリを設定します。

マスターサーバーから NIS マップセットをコピーしてから、NIS クライアントを起動します。ypinit コマンドを実行 (起動) したら、プロンプトに従って、優先順に NIS サーバーを指定してください。

```
slave3# cd /var/yp
slave3# ypinit -c
```

10. このマシンをスレーブサーバーとして初期設定します。

```
slave3# /usr/sbin/ypinit -s ypmaster
```

ypmaster は、既存の NIS マスターサーバーのマシン名です。

11. NIS クライアントとして実行されているマシンを停止します。

```
slave3# svcadm disable network/nis/client
```

12. クライアントサービスが実行されているかどうかを判定してから、必要に応じてサービスを起動または再起動します。

```
# svcs \*nis\*
STATE      STIME      FMRI
online     20:32:56   svc:/network/nis/domain:default
online     20:32:56   svc:/network/nis/client:default
```

サービスが online の状態が表示される場合、NIS は実行されています。サービス状態が disabled である場合、NIS は実行されていません。

- a. クライアントサービスが実行されている場合は、クライアントサービスを再起動します。

```
# svcadm restart network/nis/domain
# svcadm restart network/nis/client
```

- b. クライアントサービスが実行されていない場合は、クライアントサービスを起動します。

```
# svcadm enable network/nis/domain
# svcadm enable network/nis/client
```

13. NIS サーバーが実行されているかどうかを判定してから、必要に応じてサービスを起動または再起動します。

```
# svcs network/nis/server
STATE      STIME      FMRI
offline    20:32:56   svc:/network/nis/server:default
```

- a. NIS サーバーが実行されている場合は、サービスを再起動します。

```
slave3# svcadm restart network/nis/server
```

- b. NIS サーバーが実行されていない場合は、サービスを起動します。

```
slave3# svcadm enable network/nis/server
```

NIS クライアントの管理

このセクションでは、ネームサービスとして NIS を使用するようにクライアントマシンを構成するための 2 つの方法について説明します。

注記 - Oracle Solaris OS は、NIS クライアントとネイティブな LDAP クライアントが同一のクライアントマシン上に共存する構成をサポートしません。

- **ブロードキャストの方法** — NIS を使用するようにクライアントマシンを構成するための推奨される方法です。手順については、[87 ページの「ブロードキャストモードで NIS クライアントを構成する方法」](#)を参照してください。
- **サーバーリストの方法** — クライアントマシンを構成するためのもう 1 つの方法であり、`ypinit` コマンドを使用してサーバーを指定します。手順については、[87 ページの「特定の NIS サーバーを使用して NIS クライアントを構成する方法」](#)を参照してください。

NIS クライアントの管理 (タスクマップ)

次の表に、NIS クライアントの管理のためのタスクを示します。

タスク	説明	説明
ブロードキャストモードで NIS クライアントを構成します。	ローカルサブネット上に存在する NIS サーバーを検索して、NIS クライアントを構成します。	87 ページの「ブロードキャストモードで NIS クライアントを構成する方法」
特定の NIS サーバーを使用して NIS クライアントを構成します。	特定の NIS マスターサーバーとスレーブサーバーを使用して NIS クライアントを構成します。	87 ページの「特定の NIS サーバーを使用して NIS クライアントを構成する方法」
NIS クライアントサービスを無効にします。	NIS クライアントサービスを無効にするには、 <code>svcadm disable</code> コマンドを使用します。	88 ページの「NIS クライアントサービスの無効化」

▼ ブロードキャストモードで NIS クライアントを構成する方法

これは、NIS クライアントを確立するための推奨される方法です。

nis/client サービスを起動すると、このサービスが ypbind コマンドを実行し、これにより NIS サーバーのローカルサブネットが検索されます。サブネットが見つかったら、ypbind がそこにバインドします。この検索を「ブロードキャスト」と呼びます。クライアントのローカルサブネット上に NIS サーバーが存在しない場合は、ypbind がバインドに失敗し、クライアントマシンは NIS サービスから名前空間データを取得できません。手順については、[87 ページの「特定の NIS サーバーを使用して NIS クライアントを構成する方法」](#)を参照してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. NIS ドメイン名を設定します。

```
# domainname
example.com
```

3. 必要に応じて、ネームサービススイッチに変更を加えます。

[30 ページの「ネームサービススイッチの構成」](#)を参照してください。

4. NIS クライアントサービスを起動します。

```
# svcadm enable network/nis/domain
# svcadm enable network/nis/client
```

▼ 特定の NIS サーバーを使用して NIS クライアントを構成する方法

始める前に 次の手順では、手順 3 で入力されるホスト名が解決される必要があります。IP アドレスの代わりにホスト名を指定することを予定している場合は、この手順を実行する前に、クライアント上の /etc/hosts ファイル内に NIS サーバーごとの適切なエントリが存在することを確認してください。詳細は、[ypinit\(1M\)](#) のマニュアルページを参照してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. NIS ドメイン名を設定します。

```
# domainname  
  
example.com  
# svcadm enable network/nis/domain
```

3. クライアント構成スクリプトを実行します。

```
# ypinit -c
```

クライアントがネームサービス情報を取得する元の NIS サーバーを指定するよう求められます。マスターサーバーと、必要な数のスレーブサーバーをリストできます。指定するサーバーはドメイン内のどこにあってもかまいません。最初に、物理マシンに(ネットワーク的に)もっとも近いサーバーをリストし、次にネットワークのより遠い部分に存在するサーバーをリストすることをお勧めします。

▼ NIS クライアントサービスの無効化

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. NIS クライアントサービスを無効にします。

```
# svcadm disable network/nis/domain  
# svcadm disable network/nis/client
```

ネットワーク情報サービスの管理

この章では、NIS の管理方法について説明します。この章の内容は次のとおりです。

- 89 ページの「パスワードファイルと名前空間のセキュリティー」
- 90 ページの「NIS ユーザーの管理」
- 93 ページの「NIS マップに関する作業」
- 100 ページの「既存のマップの更新」
- 106 ページの「NIS サーバーの操作」

注記 - NIS サービスはサービス管理機能によって管理されます。このサービスに関する有効化、無効化、再起動などの管理アクションは `svcadm` コマンドを使用して実行できます。NIS で SMF を使用する場合の詳細については、68 ページの「NIS とサービス管理機能」を参照してください。SMF の概要については、『Oracle Solaris 11.3 でのシステムサービスの管理』の第 1 章、「サービス管理機能の概要」を参照してください。また、詳細については `svcadm(1M)` および `svcs(1)` のマニュアルページも参照してください。

NIS サービスはまた、`ypstart` および `ypstop` コマンドを使用して起動および停止することもできます。詳細については、`ypstart(1M)` および `ypstop(1M)` のマニュアルページを参照してください。

パスワードファイルと名前空間のセキュリティー

セキュリティーの関係上、次のガイドラインに従ってください。

- マスターサーバーの NIS マップへのアクセスは制限します。
- 未承認のアクセスを防ぐために、NIS のパスワードマップの構築に使用されるファイルには `root` のエントリを含めないでください。したがって、`root` エントリをこのパスワードファイルから削除して、このパスワードファイルをマスターサーバーの `/etc` ディレクトリ以外のディレクトリにおく必要があります。このディレクトリへの未許可アクセスは、防止しなければなりません。

たとえば、マスターサーバーのパスワード入力ファイルは、そのファイル自体が別のファイルへのリンクではなく、`Makefile` で指定されているかぎり、`/var/yp` などのディレクトリや、選択した任意のディレクトリ内に格納できます。サービス管理機能または `ypstart` スクリプトのどちらかを使用して NIS サービスを起動すると、`Makefile` で指定された構成に従って、正しいディレクトリオプションが設定されます。

注記 - この NIS 実装では、NIS パスワードマップを作成するための入力として、旧 Solaris 1 バージョンの `passwd` ファイルのフォーマットに加え、Solaris 2 の `passwd` ファイルと `shadow` ファイルのフォーマットも使用できます。

NIS ユーザーの管理

このセクションでは、ユーザーパスワードの設定、NIS ドメインへの新しいユーザーの追加、ネットグループ (`netgroups`) へのユーザーの割り当てについて説明します。

▼ NIS ドメインに新しい NIS ユーザーを追加する方法

1. NIS マスターサーバー上の管理者になります。

詳細は、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. `useradd` コマンドで新しいユーザーのログイン ID を作成します。

```
# useradd userID
```

ここで、`userID` は新しいユーザーのログイン ID です。このコマンドは、NIS マスターサーバー上の `/etc/passwd` および `/etc/shadow` ファイル内にエントリを作成します。

3. 新しいユーザーの初期パスワードを作成します。

新しいユーザーがログインするための初期パスワードを作成するには、`passwd` コマンドを実行します。

```
# passwd userID
```

ここで、`userID` は新しいユーザーのログイン ID です。このユーザーに割り当てるパスワードを入力するようにプロンプトが表示されます。

この手順が必要になるのは、`useradd` コマンドで作成されたパスワードエントリがロックされ、新しいユーザーがログインできないからです。初期パスワードを指定することで、このパスワードエントリのロックが解除されます。

4. 新しいエントリをマスターサーバーの `passwd` マップ入力ファイルにコピーします。

マスターサーバー上のマップソースファイルは、`/etc` 以外のディレクトリにあります。`/etc/passwd` および `/etc/shadow` ファイルの新しい行をサーバー上の `passwd` マップ入力ファイルにコピー & ペーストします。詳細については、[89 ページの「パスワードファイルと名前空間のセキュリティ」](#)を参照してください。

たとえば、新しいユーザー `brown` を追加した場合は、`passwd` 入力ファイルにコピーする `/etc/passwd` の行は次のようになります。

```
brown:x:123:10:User brown:/home/brown:/bin/csh:
```

`/etc/shadow` からコピーする `brown` の行は次のようになります。

```
brown:$5$YiFpYwXb$6jJkG/gKdfkKt1Tbem0RnbeH.qsv09MwBD3u1Tihq9B:6445:::~:~:~
```

5. パスワード入力ファイルが格納されているディレクトリが `Makefile` で正しく指定されていることを確認します。

6. `/etc/passwd` および `/etc/shadow` 入力ファイルから、新しいユーザーのエントリを削除します。

セキュリティ上の理由から、NIS マスターサーバーの `/etc/passwd` および `/etc/shadow` ファイル内にユーザーエントリを保持しないでください。ほかのディレクトリに存在する NIS マップソースファイルに新しいユーザーのエントリをコピーしたあと、マスターサーバー上で `userdel` コマンドを使用して新しいユーザーを削除します。

たとえば、マスターサーバーの `/etc` ファイルから新しいユーザー `brown` を削除するには次のように入力します。

```
# userdel brown
```

`userdel` についての詳細は、[userdel\(1M\)](#) のマニュアルページを参照してください。

7. NIS の `passwd` マップを更新します。

マスターサーバー上の `passwd` 入力ファイルを更新したら、ソースファイルを含むディレクトリ内で `make` を実行することによって `passwd` マップを更新します。

```
# userdel brown
# cd /var/yp
# make passwd
```

8. 新しいユーザーのログイン ID に割り当てられた初期パスワードを新しいユーザーに通知します。

ログイン後、新しいユーザーはいつでも `passwd` を実行して別のパスワードに変更できます。

ユーザーパスワードの設定

ユーザーは、passwd を実行して自分のパスワードを変更します。

```
% passwd username
```

パスワードファイルを更新するために、ユーザーが自分のパスワードを変更する前にマスターサーバー上で rpc.yppasswdd デーモンを起動する必要があります。

rpc.yppasswdd デーモンは、マスターサーバー上で自動的に起動します。rpc.yppasswdd に -m オプションが指定されていると、ファイルが変更されたあと、ただちに /var/yp 内で make コマンドが実行されます。passwd ファイルが変更されるたびに make コマンドが実行されることを回避する場合は、ypstart スクリプト内の rpc.yppasswdd コマンドから -m オプションを削除し、passwd マップのプッシュを crontab ファイルによって制御します。

NIS ネットグループ

NIS ネットグループは、NIS 管理者が管理目的のために定義するユーザーまたはマシンのグループ (集合) です。たとえば、次のようなネットグループを作成できます。

- 特定マシンにアクセスできる一群のユーザーを定義する
- 特定のファイルシステムにアクセスできる一群の NFS クライアントマシンを定義する
- 特定の NIS ドメインのすべてのマシンに対して管理者権限を持つ一群のユーザーを定義する

各ネットグループには、1つのネットグループ名が与えられます。ネットグループはアクセス権を直接設定しません。代わりに、ユーザー名またはマシン名が一般に使用される場所ではネットグループ名がほかの NIS マップで使用されます。たとえば、netadmins という名前のネットワーク管理者のネットグループを作成したとします。netadmins ネットグループのすべてのメンバーに特定のマシンへのアクセス権を付与するには、そのマシンの /etc/passwd ファイルに netadmin エントリを追加するだけで済みます。ネットグループ名を /etc/netgroup ファイルに追加して、NIS グループマップに追加することもできます。ネットグループの使用についての詳細は、[netgroup\(4\)](#) のマニュアルページを参照してください。

NIS を使用しているネットワーク上では、NIS マスターサーバー上の netgroup 入力ファイルを使用して、netgroup、netgroup.byuser、netgroup.byhost の3つのマップが生成されます。netgroup マップには、netgroup 入力ファイル内の基本情報が含まれています。ほかの2つの NIS マップには、マシン名またはユーザー名が指定されると、ネットグループ情報の検索が高速化される形式で情報が含まれています。

`netgroup` 入力ファイル内のエントリの形式は `name ID` です。ここで、`name` はネットグループに付ける名前であり、`ID` はネットグループに属するマシンまたはユーザーを識別します。ネットグループの `ID` (メンバー) は、コンマで区切っていくつでも指定できます。たとえば、3つのメンバーを含むネットグループを作成する場合、`netgroup` 入力ファイルのエントリは `name ID, ID, ID` という形式になります。`netgroup` 入力ファイルのエントリ内のメンバー `ID` の形式は次のようになります。

```
([-|machine], [-|user], [domain])
```

ここで、`machine` はマシン名、`user` はユーザー `ID`、`domain` はマシンまたはユーザーの `NIS` ドメインです。「ドメイン」エレメントは任意指定ですが、ほかの `NIS` ドメインのマシンまたはユーザーを示す場合には必ず指定します。各エントリでは「マシン」エレメントと「ユーザー」エレメントは必須ですが、ダッシュ (-) は空であることを示すために使用されます。エントリでは、「マシン」エレメントと「ユーザー」エレメントの関係を示す必要はありません。

`netgroup` 入力ファイルの2つのサンプルエントリを次に示します。これらの各サンプルエントリでは、`admins` という名前のネットグループが作成されます。これらの各ネットグループは、リモートドメイン `sales` に存在するユーザー `hauri` と `juanita`、およびマシン `altair` と `sirius` で構成されます。

```
admins (altair,hauri,) (sirius,juanita,sales)
admins (altair,-,) (sirius,-,) (-,hauri,) (-,juanita,sales)
```

さまざまなプログラムが、ログイン、リモートマウント、リモートログイン、およびリモートシェル作成中のアクセス許可のチェックのためにネットグループの `NIS` マップを使用します。これらのプログラムには、`mountd` や `login` が含まれます。`login` コマンドは、`passwd` データベース内でネットグループ名を見つけた場合に、ネットグループマップでユーザー分類を調べます。`mountd` デーモンは、`/etc/dfs/dfstab` ファイル内にネットグループ名を検出すると、マシンの分類のためにネットグループマップを参照します。実際、`ruserok` インタフェースを使用するプログラムはすべて、`/etc/hosts.equiv` または `.rhosts` ファイル内にネットグループ名を検出すると、マシンとユーザーの両方の分類のためにネットグループマップをチェックします。

ネットワークに新しい `NIS` ユーザーまたはマシンを追加する場合は、`netgroup` 入力ファイルの該当ネットグループに追加してください。次に、`make` でネットグループマップを作成し、これを `yppush` コマンドですべての `NIS` サーバーに転送してください。ネットグループおよびネットグループ入力ファイルの構文の使用についての詳細は、[netgroup\(4\)](#) のマニュアルページを参照してください。

NIS マップに関する作業

このセクションには次の情報が含まれます。

- 94 ページの「マップ情報の取得」
- 95 ページの「マップのマスターサーバーの変更」
- 96 ページの「構成ファイルの変更」
- 97 ページの「`/var/yp/Makefile` の変更および使用」

マップ情報の取得

ユーザーは、`ypcat`、`ypwhich`、および `ypmatch` コマンドを使用して、マップの情報やマップに関する情報をいつでも取得できます。次の例では、`mapname` はマップの正式名とニックネーム (存在する場合) の両方を意味します。

マップ内のすべての値を一覧表示するには、次のように入力します。

```
% ypcat mapname
```

マップ内の鍵と値 (存在する場合) の両方を一覧表示するには、次のように入力します。

```
% ypcat -k mapname
```

マップのすべてのニックネームを一覧表示するには、次のいずれかのコマンドを入力します。

```
% ypcat -x
% ypmatch -x
% ypwhich -x
```

使用可能なすべてのマップとそのマスターを一覧表示するには、次のように入力します。

```
% ypwhich -m
```

特定のマップのマスターサーバーを一覧表示するには、次のように入力します。

```
% ypwhich -m mapname
```

鍵をマップ内のエン트리と照合するには、次のように入力します。

```
% ypmatch key mapname
```

探している項目がマップ内の鍵でない場合は、次のように入力します。

```
% ypcat mapname | grep item
```

`item` は検索している情報です。ほかのドメインに関する情報を取得するには、これらのコマンドの `-d domainname` オプションを使用します。

デフォルト以外のドメインの情報をリクエストしているマシンに、リクエストしたドメインのバインドがない場合、`ypbind` は、`/var/yp/binding/domainname/ypservers` ファイルを参照してそのドメインのサーバーのリス

トを探します。このファイルが存在しない場合、ypbind は RPC ブロードキャストを送出してサーバーを検索します。この場合、検索先であるドメインのサーバーは要求元マシンと同じサブネットに存在している必要があります。

マップのマスターサーバーの変更

選択したマップのマスターサーバーを変更するには、まず新しい NIS マスター上にマップを構築する必要があります。古いマスターサーバー名は既存のマップにキーと値のペアとして発生するので(このペアは `makedbm` で自動的に挿入される)、`ypxfr` でマップを新しいマスターサーバーにコピーしたり、コピーを新しいマスターサーバーに転送するだけでは不十分です。鍵と新しいマスターサーバー名との対応づけをし直す必要があります。マップに ASCII ソースファイルが存在する場合は、このファイルを新しいマスターサーバーにコピーします。

▼ マップのマスターサーバーを変更する方法

1. **NIS マスターサーバー上の管理者になります。**

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. **ディレクトリを変更します。**

```
newmaster# cd /var/yp
```

3. **作成するマップを指定する前に、`/var/yp/Makefile` に新しいマップのエントリが存在する必要があります。**

そうでない場合は、ここで `Makefile` を編集します。この例では、`sites.byname` という名前のマップのエントリを追加します。

4. **マップを更新または再作成するには、次のように入力します。**

```
newmaster# make sites.byname
```

5. **古いマスターが NIS サーバーとして残っている場合は、古いマスターにリモートログイン (`ssh`) して `/var/yp/Makefile` を編集します。**

`sites.byname` マップを作成した `Makefile` のセクションをコメントアウトして、そのマップがもうそこで作成されないようにしてください。

6. **`sites.byname` が `ndbm` ファイルとしてのみ存在する場合は、新しいマスターサーバー上で作成し直します。**

まず、`ypcat` コマンドを使用して、`sites.byname` ファイルのコピーを分解します。次に、`makedbm` を使用して、分解されたバージョンを実行します。

```
newmaster# cd /var/yp
newmaster# ypcat sites.byname | makedbm domain/sites.byname
```

新しいマスターサーバー上でマップが作成されたら、そのコピーをほかのスレーブサーバーに送信する必要があります。yppushを使用しないでください。これを使用すると、ほかのスレーブは新しいマスターではなく、古いマスターから新しいコピーを取得しようとしています。このような動作を回避するには、一般にマップのコピーを新しいマスターサーバーから古いマスターサーバーに送り返すという方法が使用されます。これを行うには、古いマスターサーバーでスーパーユーザーになるか、同等の役割になり、次のように入力します。

```
oldmaster# /usr/lib/netsvc/yp/ypxfr -h newmaster sites.byname
```

これで、yppushを使用できます。スレーブサーバーは古いマスターサーバーを現行のマスターサーバーとして認識しているので、マップの現行バージョンを古いマスターサーバーから取得しようとしています。クライアントがこの処理を行うときは、新しいマスターサーバーが現行のマスターサーバーとして指定されている新しいマップを取得します。

この方法が失敗した場合は、各 NIS サーバーに root としてログインし、上に示すように ypxfr コマンドを実行できます。

構成ファイルの変更

NIS は設定ファイルを正確に構文解析します。このため NIS 管理は容易になります。設定ファイルおよび構成ファイルにおける変更により、NIS の動作は影響を受けます。

次のいずれかを実行する場合は、このセクションの手順を使用します。

- /var/yp/Makefile による、サポートされるマップの追加または削除
- C2 セキュリティーを許可または拒否するための \$PWDIR/security/passwd.adjunct の追加または削除 (\$PWDIR は /var/yp/Makefile で定義されます)

▼ 構成ファイルを更新する方法

次の点に注意してください。

- NIS マスターサーバーからマップまたはソースファイルを削除しても、スレーブサーバー上の対応するマップまたはソースファイルは自動的に削除されません。スレーブサーバー上の対応するマップまたはソースファイルの削除は、NIS 管理者が手作業で行う必要があります。
- 新しいマップは、自動的に既存のスレーブサーバーに転送されません。新しいマップを既存のスレーブサーバーに転送するには、NIS 管理者がそのスレーブサーバーで ypxfr を実行してください。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の「割り当てられている管理権利の使用」を参照してください。

2. NIS サーバーを停止します。

```
# svcadm disable network/nis/server
```

3. 必要に応じてファイルを変更します。

4. NIS サーバーを起動します。

```
# svcadm enable network/nis/server
```

/var/yp/Makefile の変更および使用

/var/yp で提供されたデフォルトの Makefile を更新することにより、NIS 管理者のニーズを満たすことができます。マップを追加したり削除したり、一部のディレクトリの名前を変更できます。

ヒント - 将来の参照のために、変更していない、元の Makefile のコピーを保存しておいてください。

Makefile での作業

新しい NIS マップを追加するには、マップの ndbm ファイルのコピーをドメインに存在する各 NIS サーバーの /var/yp/domainname ディレクトリに転送する必要があります。通常これは、Makefile によって行われます。どの NIS サーバーがマップのマスターサーバーであるかを決定したら、マップを容易に作成し直せるようにマスターサーバーの Makefile を更新してください。異なるサーバーを異なるマップのマスターサーバーとして設定することも可能ですが、このようにするとたいいていの場合、管理上の混乱を招きます。したがって、1つのサーバーだけをすべてのマップのマスターサーバーとして設定するようにしてください。

通常、人間が読める形式のテキストファイルは、makedbm への入力に適したものにするために awk、sed、または grep によってフィルタリングされます。デフォルトの Makefile を参照してください。make コマンドに関する一般情報については、make(1S) を参照してください。

make が認識する依存性の作成方法を決定する際には、Makefile にすでに備わっているメカニズムを使用してください。make では、依存ルール内の行の始まりにタブが存

在するか否かが重要であることに注意してください。ほかの設定が正しくても、タブが存在しないというだけでエントリが無効になることがあります。

Makefile にエントリを追加する場合は、次の作業を行なってください。

- データベース名を `all` ルールに追加する
- `time` ルールを作成する
- データベースのルールを追加する

たとえば、Makefile をオートマウント入力ファイルで動作させるには、`auto_direct.time` および `auto_home.time` マップを NIS データベースに追加する必要があります。

これらのマップを NIS データベースに追加するには、Makefile を変更する必要があります。

Makefile のマクロおよび変数の変更

Makefile の先頭で定義されている変数の設定は、等号 (=) の右側にある値を変更することによって変更できます。たとえば、マップへの入力として `/etc` にあるファイルを使用せず、代わりに `/var/etc/domainname` などの別のディレクトリにあるファイルを使用する場合は、`DIR` を `DIR=/etc` から `DIR=/var/etc/domainname` に変更するようにしてください。また、`PWDIR` も `PWDIR=/etc` から `PWDIR=/var/etc/domainname` に変更するようにしてください。

これらの変数を次に示します。

- `DIR=passwd` と `shadow` を除くすべての NIS 入力ファイルが存在するディレクトリ。デフォルト値は `/etc` です。マスターサーバーの `/etc` ディレクトリのファイルを NIS 入力ファイルとして使用することは望ましくないため、この値は変更しなければなりません。
- `PWDIR=NIS` 入力ファイル `passwd` と `shadow` が存在するディレクトリ。マスターサーバーの `/etc` ディレクトリのファイルを NIS 入力ファイルとして使用することは望ましくないため、この値は変更しなければなりません。
- `DOM=NIS` ドメイン名。`DOM` のデフォルト値は、`domainname` コマンドを使用して設定できます。

Makefile エントリの変更

次の手順では、Makefile にデータベースを追加したり削除したりする方法を説明します。

▼ 特定のデータベースを使用するように /var/yp/Makefile を変更する方法

この手順を実行するには、NIS マスターサーバーがすでに構成されている必要があります。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の「割り当てられている管理権利の使用」を参照してください。

2. all という単語で始まる行を、追加したいデータベースの名前を追加することによって変更します。

```
all: passwd group hosts ethers networks rpc services protocols \
netgroup bootparams aliases netid netmasks \
audit_user auth_attr exec_attr prof_attr \
auto_direct
```

エントリの順序は任意ですが、継続行の始まりの空白はスペースではなくタブにしてください。

3. Makefile の最後に次の行を追加します。

```
auto_direct: auto_direct.time
auto_home: auto_home.time
```

4. ファイル中央に auto_direct.time エントリを追加します。

```
auto_direct.time: $(DIR)/auto_direct
@(while read L; do echo $$L; done < $(DIR)/auto_direct
$(CHKPIPE) | \ (sed -e "/^#/d" -e "s/#.*$$//" -e "/^ *$$/d"
$(CHKPIPE) | \ $(MAKEDBM) - $(YPBDDIR)/$(DOM)/auto_direct;
@touch auto_direct.time;
@echo "updated auto_direct";
@if [ ! $(NOPUSH) ]; then $(YPPUSH) auto_direct; fi
@if [ ! $(NOPUSH) ]; then echo "pushed auto_direct"; fi
```

次に、各変数について説明します。

- CHKPIPE は、次のコマンドに結果を渡す (パイピングする) 前に、パイプ (|) の左側の動作が正しく行われたことを確認します。パイプの左側の動作が正しく行われなかった場合は、「NIS make terminated」というメッセージが表示されてプロセスは終了します。
- NOPUSH は、makefile が yppush を呼び出して新しいマップをスレーブサーバーに転送することを防止します。NOPUSH が設定されていない場合は、転送は自動的に行われます。

先頭にある while ループは、入力ファイル内のバックスラッシュで拡張された行をすべて削除するように設計されています。sed スクリプトはコメント行と空行を削除します。

ほかのすべてのオートマウントマップ (`auto_home` や、ほかのデフォルト以外のすべてのマップなど) でも、同じ手順に従います。

5. **make** コマンドを実行します。

```
# make mapname
```

ここで、`mapname` は作成するマップの名前です。

▼ データベースを削除するために Makefile を変更する方法

Makefile で特定データベースのマップを作成しない場合は、Makefile を次のように編集してください。

1. **all** 規則からデータベースの名前を削除します。
2. 削除するデータベースのデータベースルールを削除またはコメントアウトします。
たとえば、`hosts` データベースを削除するには、`hosts.time` エントリを削除してください。
3. **time** ルールを削除します。
たとえば、`hosts` データベースを削除するには、`hosts: hosts.time` エントリを削除してください。
4. マスターサーバーとスレーブサーバーからマップを削除します。

既存のマップの更新

NIS のインストール終了後、頻繁に更新しなければならないマップとまったく更新する必要がないマップがあることに気づく場合があります。たとえば、大企業のネットワークでは `passwd.byname` マップが頻繁に変更される場合があるのに対して、`auto_master` マップはほとんど変更されません。

61 ページの「[デフォルトの NIS マップ](#)」で説明されているように、デフォルトの NIS マップのデフォルトの場所は、マスターサーバー上の `/var/yp/domainname` 内にあります。ここで、`domainname` は NIS ドメインの名前です。マップを更新する必要がある場合は、マップがデフォルトのマップか否かによって 2 つの更新手順のどちらかを使用できます。

- デフォルトのマップは、ネットワークデータベースから `ypinit` コマンドによって作成されるデフォルトセット内のマップです。
- デフォルト以外のマップは次のいずれかです。
 - ベンダーから購入したアプリケーションに含まれているマップ

- サイト専用で作成されたマップ
- テキスト以外のファイルから作成されたマップ

このセクションでは、さまざまな更新ツールの使用方法について説明します。これらは実際には、システムがすでに起動され、実行されたあとにデフォルト以外のマップを追加したり、NIS サーバーのセットを変更したりする場合にのみ使用するようになる可能性があります。

▼ デフォルトセットに付いているマップを更新する方法

デフォルトセットに付属のマップを更新するには、次の手順を使用します。

1. **NIS マスターサーバー上の管理者になります。**

詳細は、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. **変更するマップのソースファイルを編集します。**

このファイルは /etc か、または選択したほかのディレクトリ内に存在する可能性があります。

3. **make コマンドを実行します。**

```
# cd /var/yp
# make mapname
```

これにより、make コマンドは、対応するファイルに加えられた変更に従ってマップを更新します。make コマンドはまた、これらの変更をほかのサーバーに伝播します。

更新されたマップの管理

以降のセクションでは、デフォルトセットで提供されているマップの更新完了後に実行する手順について説明します。

NIS マップを伝播する

マップが変更されたあと、Makefile は yppush を使用して、新しいマップをスレーブサーバーに伝播します (Makefile で NOPUSH が設定されていないかぎり)。これは、ypserv デーモンに通知してマップ転送要求を送ることで実行されます。次に、スレーブ上の ypserv デーモンが ypxfr プロセスを起動し、このプロセスがさらにマスターサーバー上の ypxfrd デーモンに接続します。いくつかの基本的なチェック (たとえば、マップが実際に変更されたかどうか) が行われたあと、そのマップが転送され

ます。そのあと、スレーブ上の `ypxfr` が `yppush` プロセスに、転送が成功したかどうかを示す応答を送信します。

`svc:/network/rpc/bind` サービスの `config/local_only` プロパティは `false` に設定する必要があります。このようにしないと、NIS マスターが `yppush` コマンドを使用してマスターマップの更新バージョンを NIS スレーブサーバーに転送することができません。

注記 - 上の手順は、まだスレーブサーバー上に存在しない新しく作成されたマップに対しては機能しません。スレーブサーバー上で `ypxfr` を実行して、新しいマップをスレーブサーバーに転送する必要があります。

ときには、マップの伝播に失敗し、`ypxfr` を使用して新しいマップ情報を手動で送信することが必要になる場合があります。`ypxfr` は、`root` の `crontab` ファイルを通して定期的に使用するか、またはコマンド行で対話的に使用するかの2つの異なる方法を選択できます。これらの方法については、以降のセクションで説明します。

マップ転送のための cron コマンドの使用

マップの更新頻度はマップによってそれぞれ異なります。たとえば、デフォルトのマップにある `protocols.byname` や、デフォルト以外のマップにある `auto_master` など、一部のマップは何か月も変更されないことがあります。一方で、`passwd.byname` は1日に数回変更される場合があります。`crontab` コマンドを使用してマップ転送をスケジュールすると、個々のマップごとに特定の伝播回数を設定できます。

マップに適切な頻度で `ypxfr` を定期的に行うには、各スレーブサーバー上の `root` の `crontab` ファイルに、該当する `ypxfr` エントリを入れる必要があります。`ypxfr` は、マスターサーバー上のコピーがローカルのコピーより新しい場合に限り、マスターサーバーと連絡をとりマップを転送します。

注記 - マスターサーバーでデフォルトの `-m` オプションを指定して `rpc.yppasswdd` が実行されると、だれかが自分の `yp` パスワードを変更するたびに `passwd` デーモンが `make` を実行し、それによって `passwd` マップが再構築されます。

cron と ypxfr でのシェルスクリプトの使用

NIS 管理者は、各マップに対する `crontab` エントリを個々に作成するという方法ではなく、`root` の `crontab` コマンドでシェルスクリプトを実行してすべてのマップを定期的に更新するという方法を使用することもできます。マップ更新シェルスクリプトのサンプルは、`/usr/lib/netsvc/yp` ディレクトリに入っています。スクリプト名は、`ypxfr_1perday`、`ypxfr_1perhour`、`ypxfr_2perday` です。これらのシェルスクリプトは、サイトの要件に合うように変更したり、置き換えたりすることができます。次の例は、デフォルトの `ypxfr_1perday` シェルスクリプトを示しています。

例 3 ypxfr_1perday シェルスクリプト

```
#!/bin/sh
#
# ypxfr_1perday.sh - Do daily yp map check/updates
PATH=/bin:/usr/bin:/usr/lib/netsvc/yp:$PATH
export PATH
# set -xv
ypxfr group.byname
ypxfr group.bygid
ypxfr protocols.byname
ypxfr protocols.bynumber
ypxfr networks.byname
ypxfr networks.byaddr
ypxfr services.byname
ypxfr ypservers
```

このシェルスクリプトは、root の crontab が毎日実行される場合は 1 日に 1 回マップを更新します。また、1 週間に 1 回、1 か月に 1 回、1 時間に 1 回などの頻度でマップを更新するスクリプトを作成することもできます。ただし、マップを頻繁に伝播するとパフォーマンスが低下する可能性があることに注意してください。詳細は、[crontab\(1\)](#) のマニュアルページを参照してください。

NIS ドメイン用に構成された各スレーブサーバー上で root として同じシェルスクリプトを実行します。各サーバー上の実行時間を変更して、マスターサーバーが動作不能にならないようにしてください。

特定のスレーブサーバーからマップを転送する場合は、シェルスクリプト内で ypxfr の `-h machine` オプションを使用します。シェルスクリプトに記述するコマンドの構文は、次のとおりです。

```
# /usr/lib/netsvc/yp/ypxfr -h machine [ -c ] mapname
```

`machine` は転送するマップが存在するサーバー名です。`mapname` は要求されたマップ名です。マシンを指定することなく `-h` オプションを指定すると、ypxfr はマスターサーバーからマップを取得しようとします。ypxfr 実行時に ypserv がローカルに実行されていない場合は、ypxfr がローカル ypserver に現在のマップ要求の取消しを送信しないよう、`-c` フラグを使用してください。

`-s domain` オプションを使用すると、別のドメインからローカルドメインにマップを転送できます。これらのマップは、ドメイン間で同じである必要があります。たとえば、2 つのドメインで同じ `services.byname` および `services.byaddr` マップを共有する可能性があります。あるいは、より細かく制御するために、rcp または rsync を使用してドメイン間でファイルを転送することもできます。

ypxfr コマンドの直接の起動

ypxfr コマンドの 2 番目の起動方法は、コマンドとしての実行です。一般に、ypxfr をコマンドとして実行するのは例外的状況においてだけです。たとえば、一時的に NIS

サーバーを設定して試験環境を作成する場合や、動作不能になっていた NIS サーバーをほかのサーバーと迅速に整合させようとする場合などです。

ypxfr のアクティビティのロギング

ypxfr が試みた転送およびその転送結果は、ログファイルに記録されます。/var/yp/ypxfr.log というファイルが存在する場合は、転送結果はこのファイルに記録されます。このログファイルのサイズには制限がありません。このログファイルのサイズが無限に大きくなることを防止するには、ときどき次のように入力してこのログファイルを空にしてください。

```
# cd /var/yp
# cp ypxfr.log ypxfr.log.old
# cat /dev/null > /var/yp/ypxfr.log
```

これらのコマンドは、crontab で 1 週間に 1 回実行させることができます。記録を取らないようにするには、ログファイルを削除してください。

デフォルト以外のマップの変更

デフォルト以外のマップを更新するには、次の手順を実行する必要があります。

1. 対応するテキストファイルを作成または編集します。
2. 新しいマップまたは更新されたマップを作成 (または再作成) します。マップ作成には 2 つの方法があります。
 - Makefile を使用します。デフォルトでないマップを作成するには、この方法をお勧めします。Makefile にマップのエントリが含まれている場合は、make *name* を実行します。ここで、*name* は構築するマップの名前です。Makefile にマップのエントリが含まれていない場合は、[97 ページの「/var/yp/Makefile の変更および使用」](#)の手順に従って、そのエントリの作成を試みます。
 - /usr/sbin/makedbm プログラムを使用します。このコマンドについては、[makedbm\(1M\)](#) のマニュアルページで詳細に説明されています。

デフォルト以外のマップを変更するための makedbm コマンドの使用

入力ファイルが存在しない場合、makedbm を使用してマップを変更する方法には次の 2 種類があります。

- `makedbm -u` の出力先を一時ファイルに変更し、一時ファイルを更新して更新済みの一時ファイルを `makedbm` の入力として使用します。
- `makedbm -u` の出力を、`makedbm` に入力されるパイプライン内で操作します。分解されたマップを `awk`、`sed`、または `cat` で更新できる場合は、この方法をお勧めします。

テキストファイルからの新しいマップの作成

テキストファイル `/var/yp/mymap.asc` がマスターサーバー上のエディタまたはシェルスクリプトで作成されていると仮定します。このファイルから NIS マップを作成し、それを `home-domain` サブディレクトリ内に配置しようとしています。マスターサーバー上で次のように入力してください。

```
# cd /var/yp
# makedbm mymap.asc home-domain/mymap
```

`mymap` マップは現在、マスターサーバーの `home-domain` ディレクトリに存在しています。この新しいマップをスレーブサーバーに転送するには、`ypxfr` を実行してください。

ファイルをベースとしたマップにエントリを追加する

`mymap` へのエントリの追加は簡単です。まず、テキストファイル `/var/yp/mymap.asc` を更新します。対応するテキストファイルを更新しないで実際の `dbm` ファイルを更新した場合は、更新内容が失われます。次に、上記のように `makedbm` を実行してください。

標準入力からマップを作成する

元のテキストファイルが存在しない場合は、次のように `makedbm` へ入力することによって、キーボードから NIS マップを作成します (Control-D で終了します)。

```
ypmaster# cd /var/yp
ypmaster# makedbm home-domain/mymap key1 value1 key2 value2 key3 value3
```

標準入力から作成されたマップを更新する

あとでマップを変更する必要がある場合は、`makedbm` を使用してマップを分解し、一時的な中間テキストファイルを作成することができます。マップを分解して一時ファイルを作成するには、次のように入力します。

```
% cd /var/yp
% makedbm -u homedomain/mymap > mymap.temp
```

作成される一時ファイル *mymap.temp* には、1 行につき 1 つのエントリが存在します。このファイルは、任意のテキストエディタで必要に応じて編集できます。

マップを更新するには、次のように入力して、変更された一時ファイルの名前を *makedbm* に指定します。

```
% makedbm mymap.temp homedomain/mymap
% rm mymap.temp
```

次に、*root* になり、次のように入力して、マップをスレーブサーバーに伝播します。

```
# yppush mymap
```

前の段落では、*makedbm* を使用してマップを作成する方法について説明しました。ただし、システムがすでに起動され、実行されたあとにデータベースにデフォルト以外のマップを追加したり、NIS サーバーのセットを変更したりしないかぎり、*ypinit* コマンドと */var/yp/Makefile* を使用することによって、実際に必要なほぼすべてのことを実行できます。

/var/yp 内の *Makefile* を使用するか、またはほかの何らかの手順を使用するかにかかわらず、その目的は同じです。最終的には、正しく作成された *dbm* ファイルの新しいペアが、マスターサーバー上の *maps* ディレクトリ内に存在する必要があります。

NIS サーバーの操作

次の手順は、特定の NIS サーバーにバインドし、NIS ドメイン名を設定し、ホスト検索を DNS に転送し、さらに NIS サービスを無効にすることによって NIS 構成を変更する方法を示しています。

特定の NIS サーバーへのバインド

指定した NIS サーバーにバインドするには、次の手順に従います。詳細は、[ypinit\(1M\)](#)、[ypstart\(1M\)](#)、および [svcadm\(1M\)](#) の各マニュアルページを参照してください。

1. NIS サーバーのホスト名とその IP アドレスを */etc/hosts* ファイルに追加します。
2. NIS ドメイン名が設定されていることを確認します。

```
# domainname
example.com
```

3. NIS サーバーホスト名を要求します。

```
# /usr/sbin/ypinit -c
Server name:      NIS サーバーホスト名を入力します
```

4. 次のいずれかの手順を実行することによって、NIS サービスを再起動します。

- リブートのあとも永続するサービスの場合は、svcadm コマンドを実行します。

```
# svcadm enable svc:/network/nis/client
```

- リブートまでしか持続しないサービスの場合は、ypstop および ypstart コマンドを実行します。

```
# /usr/lib/netsvc/yp/ypstop
```

```
# /usr/lib/netsvc/yp/ypstart
```

▼ マシンの NIS ドメイン名を設定する方法

マシンの NIS ドメイン名を変更するには、次の手順を使用します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『[Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. NIS ドメイン名を定義します。

```
# domainname research.example.com
```

3. ドメインネームサービスを更新して実行します。

```
# svccfg -s nis/domain:default refresh
# svcadm enable nis/domain
```

4. マシンを NIS クライアント、スレーブサーバー、またはマスターサーバーとして設定します。

詳細は、[第6章「ネットワーク情報サービスの設定および構成」](#)を参照してください。

▼ NIS と DNS を使用してマシンのホスト名とアドレスの検索を構成する方法

一般に NIS クライアントは、マシン名とアドレスの検索に NIS だけが使用されるように、nsswitch.conf ファイルで構成されます。このような検索が失敗した場合は、NIS サーバーはこれらの結果を DNS に転送します。

1. 管理者になります。

特定のタスクを実行するための適切な権利を取得する方法の詳細については、『Oracle Solaris 11.3 でのユーザーとプロセスのセキュリティ保護』の「割り当てられている管理権利の使用」を参照してください。

2. YP_INTERDOMAIN キーを追加します。

hosts.byname と hosts.byaddr という 2 つのマップファイルには、YP_INTERDOMAIN キーが必要です。このキーをテストするには、次の /var/yp/Makefile を編集します

```
#B=-b
B=
```

上の行を下に示すように変更します。

```
B=-b
#B=
```

これで、マップの作成時に makedbm が -b フラグで起動され、YP_INTERDOMAIN キーが ndbm ファイルに挿入されます。

3. make コマンドを実行してマップを作成し直します。

```
# make hosts
```

4. DNS ネームサーバーが正しく設定されていることを確認します。

次のコマンドは、DNS ネームサーバーのすべての IP アドレスを一覧表示します。

```
# svcprop -p config/nameserver network/dns/client
```

5. DNS 転送を有効にするために、各サーバーを再起動します。

```
# svcadm restart network/nis/server:instance
```

この NIS の実装では、-d オプションで ypserv デーモンが自動的に起動して DNS にリクエストを転送します。

NIS サービスをオフにする

NIS マスター上の ypserv デーモンが無効になっていると、どの NIS マップも更新できなくなります。

- クライアント上の NIS を無効にするには、次のように入力します。

```
# svcadm disable network/nis/domain
# svcadm disable network/nis/client
```

- 特定のスレーブまたはマスターサーバー上の NIS を無効にするには、そのサーバー上で次のように入力します。

```
# svcadm disable network/nis/domain  
# svcadm disable network/nis/server
```


ネットワーク情報システムのトラブルシューティング

この章では、NIS を実行しているネットワーク上で発生する問題の解決方法について説明します。ここでは、NIS クライアントと NIS サーバーの両方で検出される問題を取り上げています。

NIS サーバーまたはクライアントのデバッグを行う前に、NIS 環境について説明している [第5章「ネットワーク情報サービスについて」](#) を参照してください。次に、現在の問題をもっともよく表しているこのセクション内のサブ見出しを探してください。

注記 - NIS サービスはサービス管理機能によって管理されます。このサービスに関する有効化、無効化、再起動などの管理アクションは `svcadm` コマンドを使用して実行できます。NIS で SMF を使用する場合の詳細については、[68 ページの「NIS とサービス管理機能」](#) を参照してください。SMF の概要については、『[Oracle Solaris 11.3 でのシステムサービスの管理](#)』の [第1章](#)、「[サービス管理機能の概要](#)」を参照してください。また、詳細については [svcadm\(1M\)](#) および [svcs\(1\)](#) のマニュアルページも参照してください。

NIS サービスはまた、`ypstart` および `ypstop` コマンドを使用して起動および停止することもできます。詳細については、[ypstart\(1M\)](#) および [ypstop\(1M\)](#) のマニュアルページを参照してください。

NIS のバインドに関する問題

NIS のバインドに関する問題の現象

NIS のバインドに関する一般的な問題には、次のようなものがあります。

- `ypbind` がサーバーを見つけられない、あるいはサーバーと通信できないというメッセージが表示される

- サーバーが応答していないというメッセージが表示される
- NIS が使用できないというメッセージが表示される
- クライアントのコマンドがバックグラウンドモードでゆっくりと処理されているか、通常よりも機能に時間がかかる
- クライアント上のコマンドがハングアップする。システム全体は正常で新しいコマンドを実行できる場合でも、コマンドがハングすることがあります
- クライアントのコマンドがあいまいなメッセージとともに、またはまったくメッセージなしでクラッシュする

単一の NIS クライアントに影響する問題

1 台か 2 台のクライアントだけで、NIS のバインドに関する問題を示す症状が発生している場合は、そのクライアントに問題があると考えられます。複数のクライアントが正しくバインドできない場合は、1 台以上の NIS サーバーに問題があると考えられます。[116 ページの「複数の NIS クライアントに影響する問題」](#)を参照してください。

ypbind が NIS クライアントで実行されていない

あるクライアントに問題がありますが、同じサブネット上のほかのクライアントは正常に動作しています。問題のあるクライアント上で、`ls -l` をたとえば `/usr` のような、多くのユーザーが所有するファイル (クライアント `/etc/passwd` ファイルにはないものも含む) を持つディレクトリで実行します。その結果の表示に、ローカルの `/etc/passwd` がないファイル所有者が名前ではなく、番号として一覧表示される場合は、そのクライアント上で NIS サービスが動作していないことを示します。

通常これらの症状は、クライアント `ypbind` プロセスが実行されていないことを示します。NIS クライアントサービスが実行されているかどうかを確認します。

```
client# svcs \*nis\*
STATE          STIME          FMRI
disabled       Sep_01         svc:/network/nis/domain:default
disabled       Sep_01         svc:/network/nis/client:default
```

これらのサービスが `disabled` 状態にある場合は、`root` としてログインするか、または同等の役割になり、NIS クライアントサービスを起動します。

```
client# svcadm enable network/nis/domain
client# svcadm enable network/nis/client
```

NIS ドメイン名が指定されていないか間違っている

あるクライアントに問題があり、ほかのクライアントは正常に動作していますが、ypbind はその問題のあるクライアント上で実行されています。クライアントのドメインの設定が間違っている可能性があります。

クライアント上で domainname コマンドを実行して、どのドメイン名が設定されているのかを調べます。

```
client7# domainname
example.com
```

この出力を、NIS マスターサーバー上の /var/yp 内の実際のドメイン名と比較します。実際の NIS ドメインは、/var/yp ディレクトリ内のサブディレクトリとして表示されます。

```
client7# ls
-l /var/yp
-rwxr-xr-x 1 root Makefile
drwxr-xr-x 2 root binding
drwx----- 2 root example.com
```

マシン上で domainname を実行することによって返されたドメイン名が、/var/yp 内のディレクトリとして一覧表示されたサーバードメイン名と同じでない場合は、マシンの /etc/defaultdomain ファイルで指定されたドメイン名が正しくありません。[107 ページの「マシンの NIS ドメイン名を設定する方法」](#)で示すように、NIS ドメイン名をリセットします。

注記 - NIS ドメイン名では大文字と小文字が区別されます。

NIS クライアントがサーバーにバインドされていない

ドメイン名が正しく設定されていて ypbind が実行中でもコマンドがまだハングする場合には、ypwhich コマンドを実行してクライアントがサーバーにバインドされていることを確認してください。ypbind を起動したばかりのときは、ypwhich を数回実行します。通常、1 回目ではドメインがバインドされていないことが通知され、2 回目は成功します。

NIS サーバーが使用できない

ドメイン名が正しく設定されていて ypbind が実行中のときに、クライアントがサーバーと通信できないというメッセージを受け取った場合には、いくつかの問題が考えられます。

- クライアントに、バインド先のサーバーのリストが含まれた `/var/yp/binding/domainname/ypservers` ファイルが存在しますか。ない場合には、`ypinit -c` を実行して、設定の順番にクライアントのバインド先のサーバーを指定します。
- クライアントに `/var/yp/binding/domainname/ypservers` ファイルがあり、1つ以上のサーバーが使用できない場合には、十分な数のサーバーがあるかどうかを調べます。ない場合には、`ypinit -c` を実行して、リストにサーバーを追加します。
- `/etc/inet/hosts` ファイル内に、選択された NIS サーバーのエントリがありますか。選択された NIS サーバーを表示するには、`svcprop -p config/ypservers nis/domain` コマンドを使用します。これらのホストがローカルの `/etc/inet/hosts` ファイル内に含まれていない場合は、Working With NIS Mapsの説明に従って `ypinit -c` または `ypinit -s` コマンドを実行することによって、93 ページの「NIS マップに関する作業」の NIS マップにサーバーを追加してマップを再構築します。
- ネームサービススイッチは NIS に加えて、マシンのローカルの `hosts` ファイルをチェックするように設定されていますか。スイッチの詳細は、第2章「ネームサービススイッチについて」を参照してください。
- ネームサービススイッチは、`services` と `rpc` で最初に `files` をチェックするように設定されていますか。スイッチの詳細は、第2章「ネームサービススイッチについて」を参照してください。

ypwhich の表示に一貫性がない

`ypwhich` を同じクライアントで数回使うと、NIS サーバーが変わるので結果の表示も変わります。これは正常です。NIS クライアントから NIS サーバーへのバインドは、ネットワークや NIS サーバーを使用中の場合は時間の経過に伴って変化します。ネットワークは、すべてのクライアントが受け入れ可能な応答時間を NIS サーバーから得られる点で安定した状態になります。クライアントのマシンが NIS サービスを得ている限りは、サービスの供給元は問題にはなりません。たとえば、NIS サーバーマシンがそれ自体の NIS サービスを、ネットワーク上の別の NIS サーバーから受けることもあります。

NIS サーバーのバインドが不可能な場合

ローカルなサーバーのバインドが不可能な場合には `ypset` コマンドを使用すると、別のネットワークまたはサブネットの別のサーバーが使用可能な場合には、そのサーバーへのバインドが一時的に可能になります。ただし、`-ypset` オプションを使用するためには、`ypbind` を `-ypset` または `-ypsetme` オプションのどちらかを指定して、実行する必要があります。詳細は、`ypbind(1M)` のマニュアルページを参照してください。

```
# /usr/lib/netsvc/yp/ypbind -ypset
```

別の方法については、[106 ページの「特定の NIS サーバーへのバインド」](#)を参照してください。



注意 - セキュリティー上の理由から、`-ypset` オプションや `-ypsetme` オプションの使用はお勧めできません。これらのオプションは、制御された環境でのデバッグの目的にのみ使用してください。`-ypset` オプションや `-ypsetme` オプションを使用すると、これらのデーモンの実行中にだれでもサーバーのバインドを変更でき、ほかのユーザーでトラブルが発生したり、機密データへの未承認のアクセスが許可されたりするため、重大なセキュリティ違反が発生する場合があります。これらのオプションを使用して `ypbind` デーモンを起動する必要がある場合は、問題を解決したあとに `ypbind` プロセスを強制終了し、これらのオプションなしでこのデーモンを再起動する必要があります。

`ypbind` デーモンを再起動するには、SMF を次のように使用します。

```
# svcadm enable -r svc:/network/nis/client:default
```

ypbind のクラッシュ

`ypbind` デーモンが、起動されるたびにほぼ即座にクラッシュする場合は、`svc:/network/nis/client:default` サービスログ内で問題を探します。次のように入力して、`rpcbind` デーモンが存在するかどうかをチェックします。

```
% ps -e |grep rpcbind
```

`rpcbind` が存在しないか、または安定しなかったり、異常な動作を行ったりする場合は、`svc:/network/rpc/bind:default` ログファイルをチェックします。詳細は、[rpcbind\(1M\)](#) および [rpcinfo\(1M\)](#) のマニュアルページを参照してください。

正常に機能しているマシンから、問題のあるクライアント上の `rpcbind` と通信ができる場合があります。正常に機能しているマシンから、次のように入力します。

```
% rpcinfo client
```

問題のあるマシン上の `rpcbind` が正常である場合、`rpcinfo` は次の出力を生成します。

program	version	netid	address	service	owner
...					
100007	3	udp6	:::191.161	ypbind	1
100007	3	tcp6	:::135.200	ypbind	1
100007	3	udp	0.0.0.0.240.221	ypbind	1
100007	2	udp	0.0.0.0.240.221	ypbind	1
100007	1	udp	0.0.0.0.240.221	ypbind	1
100007	3	tcp	0.0.0.0.250.107	ypbind	1

```

100007 2      tcp      0.0.0.0.250.107  ypbind  1
100007 1      tcp      0.0.0.0.250.107  ypbind  1
100007 3      ticlts   2\000\000\000    ypbind  1
100007 2      ticlts   2\000\000\000    ypbind  1
100007 3      ticotsord 9\000\000\000    ypbind  1
100007 2      ticotsord 9\000\000\000    ypbind  1
100007 3      ticots   @\000\000\000    ypbind  1
...

```

使用中のシステムには異なる複数のアドレスがあります。それらのアドレスが表示されない場合は、ypbind によってそのサービスが登録できていません。システムをリブートして再度 rpcinfo を実行します。ypbind プロセスがそこに存在し、NIS サービスを再起動しようとするたびに変更される場合は、rpcbind デーモンが実行されていてもシステムをリブートします。

複数の NIS クライアントに影響する問題

1 台か 2 台のクライアントだけで、NIS のバインドに関する問題を示す症状が発生している場合は、そのクライアントに問題があると考えられます。112 ページの「[単一の NIS クライアントに影響する問題](#)」を参照してください。複数のクライアントが正しくバインドできない場合は、1 台以上の NIS サーバーに問題があると考えられます。

rpc.yppasswdd が r で始まる制限のないシェルを制限付きと見なしている

1. 次のような特殊な文字列が含まれている /etc/default/yppasswdd を作成します。"check_restricted_shell_name=1"
2. 「check_restricted_shell_name=1」の文字列がコメントアウトされている場合、「r」のチェックは実行されません。

ネットワークまたは NIS サーバーに到達できない

ネットワークまたは NIS サーバーが過負荷状態であるために、ypserv デーモンが、クライアントの ypbind プロセスに返される応答をタイムアウト期間内に受信できない場合は、NIS がハングアップすることがあります。NIS はまた、ネットワークがダウンしている場合にもハングアップすることがあります。

こういった状態では、ネットワーク上のすべてのクライアントで同じまたは類似した問題が発生します。ほとんどの場合、この状態は一時的です。これらのメッセージは通常、NIS サーバーがリブートして ypserv を再起動するか、NIS サーバー上または

ネットワーク自体の負荷が減るか、またはネットワークが正常な動作を再開すると消えます。

NIS サーバーの誤動作

サーバーが起動され、実行されていることを確認します。サーバーが物理的に近くにない場合には、ping コマンドを使ってください。

NIS デーモンが実行されていない

サーバーが起動されていて実行中の場合には、クライアントマシンが正常に動作していることを調べて、ypwhich コマンドを実行します。ypwhich が応答しない場合は、そのコマンドを強制終了します。次に、NIS サーバー上で root としてログインし、次のように入力して、NIS プロセスが実行されているかどうかをチェックします。

```
# ptree |grep ypbind
100759 /usr/lib/netstvc/yp/ypbind -broadcast
527360 grep yp
```

ypserv (NIS サーバー) と ypbind (NIS クライアント) のどちらのデーモンも実行されていない場合は、次のように入力してそれらを再起動します。

```
# svcadm restart network/nis/client
```

NIS サーバー上で ypserv プロセスと ypbind プロセスの両方が実行されている場合は、ypwhich コマンドを実行します。このコマンドが応答しない場合は、ypserv デーモンがおそらくハングアップしているため、再起動してください。サーバー上で root としてログインしている間に、次のように入力して NIS サービスを再起動します。

```
# svcadm restart network/nis/server
```

NIS サーバーに別のバージョンの NIS マップが存在する

NIS はマップをサーバー間で伝播するので、ネットワーク上のさまざまな NIS サーバーに、同じマップの異なるバージョンが存在することがあります。このバージョンの不一致は、これらの違いが短時間しか続かなければ正常であり、許容可能です。

マップの不一致のもっとも一般的な原因は、マップの正常な伝播を妨げる何かが存在するためです。たとえば、NIS サーバーまたはルーターが、NIS サーバー間でダウンロードしている場合です。すべての NIS サーバーと、それらのサーバー間にあるルーターが実行されている場合、ypxfr は成功します。

サーバーとルーターが正常に機能している場合には、次のことをチェックします。

- `ypxfr` のログ出力をチェックします。118 ページの「[ypxfr の出力のロギング](#)」を参照してください。
- `svc:/network/nis/xfr:default` ログファイルにエラーが表示されていないかどうかをチェックします。
- 制御ファイルをチェックします。118 ページの「[crontab ファイルと ypxfr シェルスクリプトをチェックする](#)」を参照してください。
- マスターサーバー上の `ypservers` マップをチェックします。(119 ページの「[ypservers マップをチェックする](#)」を参照)。

ypxfr の出力のロギング

特定のスレーブサーバーでマップの更新に関する問題が発生した場合は、そのサーバーにログインし、`ypxfr` コマンドを対話的に実行します。このコマンドが失敗した場合は、失敗した理由が示されるため、問題を解決することができます。このコマンドは成功するが、ときどき失敗していたと思われる場合は、メッセージのロギングを有効にするためにログファイルを作成します。ログファイルを作成するには、スレーブ上で次のように入力します。

```
ypslave# cd /var/yp
ypslave# touch ypxfr.log
```

これによって、`ypxfr` からのすべての出力を保存する `ypxfr.log` ファイルが作成されます。

この出力は、`ypxfr` が対話形式で実行しているときに表示する出力と似ていますが、ログファイルの各行にはタイムスタンプが記録されます。(タイムスタンプが通常とは異なった順序になることがあります。これは問題ありません。タイムスタンプは、`ypxfr` が実行を開始した時間を示します。`ypxfr` の複数のコピーが同時に実行されたが、それらの動作時間が異なった場合は、各サマリーステータス行が、それぞれの起動順序とは異なる順序で実際にログファイルに書き込まれることがあります。)断続的に発生するあらゆる種類の障害がログに記録されます。

注記 - 問題を解決したら、ログファイルを削除してログを停止します。削除し忘れた場合は、そのファイルが無制限に拡張し続けます。

crontab ファイルと ypxfr シェルスクリプトをチェックする

`root` の `crontab` ファイルを調べ、それが起動する `ypxfr` シェルスクリプトを確認します。これらファイルにタイプミスがあると、伝播に関する問題が発生します。`/var/spool/cron/crontabs/root` ファイル内でシェルスクリプトを参照できな

い場合や、任意のシェルスクリプト内でマップを参照できない場合にも、エラーが発生します。

ypservers マップをチェックする

NIS スレーブサーバーが、ドメインに対するマスターサーバー上の `ypservers` マップにリストされていることも確認してください。リストされていない場合には、スレーブサーバーはサーバーとして正しく機能しますが、`yppush` はマップの変更をスレーブサーバーに伝播しません。

壊れたスレーブサーバー上のマップを更新するための回避方法

NIS スレーブサーバーの問題が明らかでない場合は、`scp` または `ssh` コマンドを使用して、一貫性のないマップの最新バージョンをいずれかの正常な NIS サーバーからコピーすることによって、問題のデバッグ中に回避方法を実行できます。次に、問題のあるマップを転送する方法を示します。

```
ypslave# scp ypmaster:/var/yp/mydomain/map.* /var/yp/mydomain
```

* の文字はコマンド行でエスケープされているため、`ypslave` 上でローカルにではなく、`ypmaster` 上で展開されます。

ypserv のクラッシュ

`ypserv` プロセスがほぼ即座にクラッシュし、起動を繰り返しても安定しない場合は、デバッグプロセスが [115 ページの「ypbind のクラッシュ」](#) で説明されている状況と実質的に同じです。まず、次のコマンドを実行して、何らかのエラーが報告されているかどうかを確認します。

```
# svcs -vx nis/server
```

`rpcbind` デーモンが存在するかどうかを、次のようにチェックします。

```
# ptree |grep rpcbind
```

デーモンが見つからない場合は、NIS サーバーをリブートします。それ以外の場合には、デーモンが実行されていれば、次のように入力して同様の出力を探します。

```
% rpcinfo -p ypserver
```

```
% program    vers    proto    port    service
100000        4       tcp      111     portmapper
100000        3       tcp      111     portmapper
100068        2       udp      32813   cmsd
...
100007        1       tcp      34900   ypbind
100004        2       udp      731     ypserv
```

```
100004      1      udp      731      ypserv
100004      1      tcp      732      ypserv
100004      2      tcp     32772    ypserv
```

使用中のマシンには、異なる複数のポート番号があることがあります。ypserv プロセスを表す 4 つのエントリは次のとおりです。

```
100004      2      udp      731      ypserv
100004      1      udp      731      ypserv
100004      1      tcp      732      ypserv
100004      2      tcp     32772    ypserv
```

これらのエントリが存在せず、ypserv がそのサービスを rpcbind に登録できない場合は、システムをリブートします。エントリがある場合には、rpcbind からサービスの登録を解除してから ypserv を再起動します。rpcbind からこのサービスの登録を解除するには、サーバー上で次のように入力します。

```
# rpcinfo -d number 1
# rpcinfo -d number 2
```

ここで、*number* は rpcinfo によって報告された ID 番号 (前の例では 100004) です。

用語集

アプリケーションレベルのネームサービス	ファイル、メール、印刷などのサービスを提供するアプリケーションに組み込まれているネームサービスのこと。アプリケーションレベルのネームサービスは、企業レベルのネームサービスの下に位置する。企業レベルのネームサービスが提供するコンテキストの中に、アプリケーションレベルのネームサービスのコンテキストを組み込むことができる。
暗号化	データのプライバシーを保護するための手段。
暗号化鍵	データ暗号化鍵の項を参照。
インターネットアドレス	TCP/IP を使用してシステムに割り当てられた 32 ビットアドレス。「ドット形式の 10 進表記」の項を参照。
インデックス付き名前	テーブル内のエントリを識別するために使用される命名形式。
エントリ	データベーステーブル内の 1 行のデータ (DIT 内の LDAP 要素など)。
鍵 (暗号化)	鍵の管理および配布システムの一部として、ほかの鍵を暗号化および暗号化解除するために使用される鍵。「データ暗号化鍵」の項も参照。
鍵サーバー	非公開鍵を格納する、Oracle Solaris オペレーティング環境のプロセス。
企業レベルのネットワーク	「企業レベルの」ネットワークは、ケーブル、赤外線ビーム、または無線ブロードキャスト経由で通信する単一のローカルエリアネットワーク (LAN) である場合や、ケーブルまたは直接電話接続でリンクされた 2 つ以上の LAN のクラスタである場合がある。企業レベルのネットワーク内では、DNS や X.500/LDAP などのグローバルネームサービスを使用せずに、どのマシンからでも任意のマシンにアクセスできる。
逆解決	DNS ソフトウェアを使用して、ワークステーションの IP アドレスをワークステーション名に変換するプロセス。
クライアント	(1) クライアントは、ネームサーバーにネームサービスをリクエストする主体 (マシンまたはユーザー)。 (2) ファイルシステムのクライアントサーバーモデルでは、クライアントとは、計算パワーや大きな記憶容量などの計算サーバーのリソースにリモートアクセスするマシン。

(3) クライアントサーバーモデルでは、クライアントは「サーバープロセス」からサービスにアクセスするアプリケーションのこと。このモデルでは、クライアントとサーバーを同一マシンまたは個別のマシンで実行できる。

クライアントサーバーモデル ネットワークサービスと、これらのサービスのモデルユーザープロセス (プログラム) を説明するための一般的な方法。たとえば、「ドメインネームシステム (DNS)」のネームサーバー/ネームリゾルバパラダイムなど。クライアントも参照。

グループ ID ユーザーのデフォルトのグループを識別する番号。

グローバルネームサービス グローバルネームサービスは、電話、衛星、またはその他の通信システムでリンクされているこれらの世界中の企業レベルのネットワークを識別 (名前付け) する。この世界規模でリンクされたネットワークの集合は、「インターネット」と呼ばれている。グローバルネームサービスは、ネットワーク名の指定のほかに、特定のネットワーク上の個々のマシンとユーザーも識別する。

広域ネットワーク (WAN) 異なる地理的な場所に存在する複数のローカルエリアネットワーク (LAN) またはシステムを電話、光ファイバ、衛星などのリンクで接続するネットワーク。

公開鍵 数学的に生成された 1 対の番号の公開コンポーネントであり、非公開鍵と組み合わせれば DES 鍵が生成される。この DES 鍵を使用すれば、情報のエンコードとデコードを行える。公開鍵は、すべてのユーザーとマシンが使用できる。どのユーザーやマシンにも、固有の公開鍵と非公開鍵が 1 対ある。

コンテキスト N2L サービスの場合、コンテキストは、一般に NIS ドメインがその下でマップされるもの。「baseDN」の項も参照。

サーバー (1) NIS、DNS、および LDAP では、ネットワークにネームサービスを提供するホストマシン。

(2) ファイルシステムの「クライアントサーバーモデル」では、サーバーとは計算リソース (計算サーバーとも呼ばれる) と大きな記憶容量を備えたマシン。クライアントマシンはリモートアクセスが可能であり、これらのリソースを使用できる。ウィンドウシステムのクライアントサーバーモデルでは、サーバーはウィンドウサービスをアプリケーション (クライアントプロセス) に提供するプロセスのこと。このモデルでは、クライアントとサーバーを同一マシンまたは個別のマシンで実行できる。

(3) ファイルの提供を実際に処理するデーモン。

サーバーリスト 「優先サーバーリスト」の項を参照。

サブネット ルーティングを単純化するために、1つの論理ネットワークをより小さな物理ネットワークに分割する実際的なスキーム。

資格	クライアントソフトウェアが各リクエストとともにネームサーバーに送信する認証情報。この情報によって、ユーザーまたはマシンの ID が検査される。
識別名	識別名は、X.500 ディレクトリ情報ベース (DIB) 内のエントリであり、ルートから指定されたエントリまでつながるパスに沿った、ツリー内の各エントリから選択された属性で構成される。
スキーマ	任意の特定の LDAP DIT 内にどのような種類のデータを格納できるかを定義する一連の規則。
スレーブサーバー	NIS データベースのコピーを保持するサーバーシステム。このシステムには、ディスクと動作環境の完全なコピーが存在する。
接尾辞	LDAP では、DIT の識別名 (DN)。
ソース	NIS ソースファイル
属性	各 LDAP エントリは、いくつかの名前付き属性で構成され、各属性は 1 つまたは複数の値を持つ。 また、N2L サービスマッピングおよび構成ファイルもそれぞれ、いくつかの名前付き属性で構成される。各属性は 1 つまたは複数の値を持つ。
ディレクトリ	(1) LDAP ディレクトリは、LDAP オブジェクトのコンテナ。(2) UNIX では、ファイルまたはサブディレクトリのコンテナのこと。
ディレクトリキャッシュ	ディレクトリオブジェクトに関連付けられたデータを格納するために使用されるローカルファイル。
ディレクトリ情報ツリー	DIT は、特定のネットワークの分散型ディレクトリ構造です。デフォルトでは、クライアントは、DIT が特定の構造を持っていると想定して情報にアクセスする。LDAP サーバーがサポートするドメインごとに、想定された構造を持つ想定されたサブツリーがある。
データ暗号化鍵	暗号化を実行するプログラムを対象としたデータを暗号化および暗号化解除するために使用される鍵。「鍵(暗号化)」の項も参照。
データ暗号化規格 (DES)	データを暗号化および復号化するための、アメリカ商務省標準局によって開発された一般的に使用されている高度なアルゴリズム。「SUN-DES-1」の項も参照。
ドット形式の 10 進表記	32 ビット整数の構文表現であり、ピリオド (ドット) で区切られた 4 つの 10 進表記の 8 ビット数で構成される。192.168.67.20 のように、インターネットでの IP アドレスを表現するために使用される。
ドメイン	(1) インターネットでは、ネーミング階層の一部であり、通常はローカルエリアネットワーク (LAN)、広域ネットワーク (WAN)、またはこのようなネットワークの一部に

対応する。構文上、インターネットドメイン名は小数点(ドット)によって区切られた一連の名前(ラベル)から構成される。たとえば、`sales.example.com` などがある。

(2) ISO の開放型システム間相互接続 (OSI) では、「ドメイン」は、MHS プライベート管理ドメイン (PRMD) やディレクトリ管理ドメイン (DMD) などのように、複雑な分散システムの管理パーティションとして使用されるのが普通。

ドメインネームサービス (DNS)	ドメイン名やマシン名を企業の外部のアドレス (インターネット上のアドレスなど) にマップするためネーミングポリシーおよびメカニズムを提供するサービス。すなわち DNS は、ドメイン名とマシン名をインターネットなどの企業外部のアドレスにマップする場合のネーミングポリシーとメカニズムを提供する。
ドメイン名	DNS 管理ファイルを共有するローカルネットワーク上のシステムのグループに割り当てられた名前。ネットワーク情報サービスのデータベースが正常に動作するためにはドメイン名が必要。ドメインの項も参照。
名前解決	ワークステーション名またはユーザー名をアドレスに変換するプロセス。
名前空間	(1) 名前空間は、ユーザー、ワークステーション、およびアプリケーションがネットワーク全体にわたって通信する必要のある情報を格納する。 (2) ネーミングシステムで使用される名前セット。
認証	サーバーがクライアントの識別情報を検証できるようにするための手段。
ネームサーバー	1 つまたは複数のネットワークネームサービスを実行するサーバー。
ネームサービス	マシン、ユーザー、プリンタ、ドメイン、ルーター、その他のネットワーク名とアドレスを処理するネットワークサービス。
ネームサービススイッチ	ネームサービスクライアントが自身のネットワーク情報を取得できるソースを定義する <code>svc:/system/name-service/switch</code> サービス。
ネットワークパスワード	「Secure RPC パスワード」の項を参照。
ネットワークマスク	ローカルサブネットアドレスを特定のインターネットプロトコルアドレスの残りから分離するためにソフトウェアによって使用される数値。
非公開鍵	数学的に生成された 1 対の番号の非公開コンポーネントであり、公開鍵と組み合わせれば DES 鍵が生成される。この DES 鍵を使用すれば、情報のエンコードとデコードを行える。送信側の非公開鍵は、その鍵の所有者だけが使用できる。どのユーザーやマシンにも、固有の公開鍵と非公開鍵が 1 対ある。
フィールド	NIS マップエントリは、いくつかのコンポーネントと区切り文字で構成される可能性がある。N2L サービスマッピングプロセスの一部として、エントリはまず、いくつかの名前付きフィールドに分解される。

マスターサーバー	特定のドメインのネットワーク情報サービスデータベースのマスターコピーを保持するサーバー。名前空間に対する変更は、必ずマスターサーバーのネームサービスデータベース上で行う。ドメイン中に複数のマスターサーバーを作成できない。
マッピング	NIS エントリと DIT エントリの間の変換を行うプロセス。この処理は、「マッピング」ファイルにより制御される。
メール交換レコード	DNS ドメイン名とそれらに対応するメールホストの一覧が含まれているファイル。
メールホスト	サイトの電子メールのルーターおよび受信側として機能するワークステーション。
優先サーバーリスト	client_info テーブルまたは client_info ファイルのこと。優先サーバーリストには、あるクライアントマシンまたはドメインから見た優先サーバーが指定される。
リモート手続き呼び出し (RPC)	分散コンピューティングのクライアントサーバーモデルを実装するための容易で、一般的なパラダイム。与えられた引数を使用することによって、要求がリモートシステムに送信され、指定された手順が実行される。そのあと、その結果が呼び出し側に返される。
レコード	エントリの項を参照。
ローカルエリアネットワーク (LAN)	データやソフトウェアの共有および交換の目的のためにまとまって接続されている、地理的に 1 か所に存在する複数のシステム。
baseDN	DIT の一部のベースとなっている DN。これが NIS ドメインエントリの baseDN である場合は、コンテキストとも呼ばれる。
databaseID	N2L サービスの場合、databaseID は、同じ形式の NIS エントリを含む (LDAP への同じマッピングを持つ) マップのグループの別名。これらのマップの鍵は異なっている可能性がある。
DBM	NIS マップを格納するために当初使用されるデータベース。
DES	データ暗号化規格 (DES) の項を参照。
DIT	「ディレクトリ情報ツリー」の項を参照。
DN	LDAP 内の識別名。ツリー構造を持つ LDAP ディレクトリのアドレススキーム。各 LDAP エントリに一意的な名前を付与する。
DNS	ドメインネームシステムの項を参照。
DNS ゾーン	ネットワークドメイン内の管理境界であり、多くの場合は 1 つまたは複数のサブドメインで構成される。
DNS ゾーンファイル	DNS ソフトウェアがドメイン内の全ワークステーションの名前と IP アドレスを格納する一連のファイル。

DNS 転送	NIS サーバーは、自身で応答できないリクエストを DNS サーバーに転送する。
GID	グループ ID の項を参照。
IP	インターネットプロトコル。インターネットプロトコル体系の「ネットワーク層」プロトコル。
IP アドレス	ネットワーク内の各ホストを識別する一意の番号。
LDAP	Lightweight Directory Access Protocol は、LDAP ネームサービスクライアントおよびサーバーが互いに通信するために使用する、標準の拡張可能なディレクトリアクセスプロトコル。
MIS	経営情報システム (またはサービス)。
N2L サーバー	NIS-to-LDAP サーバー。N2L サービスを使用して、N2L サーバーとして再構成された NIS マスターサーバー。再構成には、NIS デモンの置き換えと新しい構成ファイルの追加が含まれる。
NDBM	DBM の改良されたバージョン。
NIS	ネットワーク上のシステムおよびユーザーに関する重要な情報が取められている分散型ネットワーク情報サービス。NIS データベースは、「マスターサーバー」とすべての「スレーブサーバー」に格納されている。
NIS マップ	NIS によって使用されるファイルであり、ネットワーク上の全ユーザーのパスワードエントリやネットワーク上の全ホストマシンの名前など特定種類の情報を格納する。NIS サービスの一部であるプログラムはこれらのマップを参照する。NIS の項も参照。
RDN	相対識別名。DN の一部。
RFC 2307	標準の NIS マップから DIT エントリへの情報のマッピングを指定した RFC。デフォルトでは、N2L サービスは、更新されたバージョン RFC 2307bis で指定されたマッピングを実装している。
RPC	リモート手続き呼び出し (RPC) を参照。
SASL	Simple Authentication and Security Layer (簡易認証セキュリティ層)。アプリケーション層プロトコルにおける認証およびセキュリティ層の意味上の取り決め。
searchTriple	特定の属性を検索する DIT 内の場所についての説明。searchTriple は、「ベース DN」、「スコープ」、および「フィルタ」で構成される。これは、RFC 2255 で定義された LDAP URL 形式の一部である。
Secure RPC パスワード	Secure RPC プロトコルに必要なパスワード。非公開鍵の暗号化に使用される。このパスワードはユーザーのログインパスワードと同じでなければならない。

SSL	SSL は Secure Sockets Layer プロトコルである。LDAP セキュアなどのアプリケーションプロトコルを作成するためのトランスポート層のセキュリティーメカニズムの総称。
TCP	「 <i>Transport Control Protocol (TCP)</i> 」の項を参照。
TCP/IP	Transport Control Protocol/Interface Program の略語。このプロトコル群は、最初はインターネット用に開発された。「インターネット」プロトコル群とも呼ばれる。Oracle Solaris ネットワークは、デフォルトでは TCP/IP 上で動作する。
Transport Control Protocol (TCP)	信頼性の高い、コネクション型の全二重ストリームを提供する、インターネットプロトコル群内の主要なトランスポートプロトコル。配信には IP を使用する。「TCP/IP」の項を参照。
Transport Layer Security (TLS)	TLS は、LDAP クライアントとディレクトリサーバーの間の通信をセキュリティー保護して、プライバシーとデータの完全性の両方を提供する。TLS プロトコルは、Secure Sockets Layer (SSL) プロトコルのスーパーセットである。
X.500	開放型システム間相互接続 (OSI) 規格によって定義されたグローバルレベルのディレクトリサービス。LDAP の前身。
yp	Yellow Pages™。NIS コード内部で今も使用される NIS の古い名前。

索引

数字・記号

\$PWDIR/security/passwd.adjunct, 96

あ

暗号化

定義, 121

暗号化鍵

定義, 121

インストール

DNS クライアント, 41

DNS パッケージ, 38

インターネット

NIS および, 56

インターネットアクセス

ネームサービススイッチおよび, 25

インターネットアドレス

定義, 121

インデックス付き名前

定義, 121

エントリ

定義, 121

か

鍵(暗号化)

定義, 121

鍵サーバー

定義, 121

企業レベルのネットワーク

定義, 121

起動

NIS デーモン, 78

逆解決

定義, 121

クライアント

NIS, 58

NIS の設定, 86

定義, 121

クライアントサーバーモデル

定義, 122

グループ

ネットグループ (NIS), 92, 93

グループ ID

定義, 122

グローバルネームサービス

定義, 122

検証

/etc/named.conf ファイル, 42

公開鍵

定義, 122

構成

DNS サーバー, 38

DNS サーバーのオプション, 40

コマンド

DNS, 46

NIS, 59

コンテキスト

定義, 122

コンパイルフラグ

DNS, 48

さ

サーバー

NIS サーバーの準備, 71

NIS スレーブサーバー, 81

NIS スレーブの設定, 81

ypservers ファイル, 84

使用できない (NIS), 113

- 定義, 122
- サーバーリスト
 - NIS のバインド, 64
 - 定義, 122
- サービス管理機能 参照 SMF
- サービス管理機能 (SMF)
 - および Oracle Solaris ネームサービス, 21
- サービス検出 参照 DNS サービス検出
- 作成
 - rndc.conf ファイル, 39
- サブネット
 - 定義, 122
- 資格
 - 定義, 123
- 識別名
 - 定義, 123
- スキーマ
 - 定義, 123
- スレーブサーバー
 - 定義, 123
- セキュリティー
 - NIS, 71, 72
 - NIS, および, 89
 - NIS マップのルート, 89
- 設定
 - NIS Makefile, 74
 - NIS クライアント, 86
 - NIS スレーブサーバー, 81, 81
 - NIS のための準備, 67, 71
 - 複数の NIS ドメイン, 78
- 接尾辞
 - 定義, 123
- ソース
 - 定義, 123
- 属性
 - 定義, 123
- た**
- タスク
 - DNS, 38
- 停止
 - NIS デーモン, 78
- ディレクトリ
 - 定義, 123
- ディレクトリキャッシュ
 - 定義, 123
- ディレクトリ情報ツリー
 - 定義, 123
- データ暗号化鍵
 - 定義, 123
- データ暗号化規格 参照 DES
- デーモン
 - DNS, 46
 - NIS, 59
- ドット形式の 10 進表記
 - 定義, 123
- ドメイン
 - NIS, 56, 58, 70
 - 定義, 123
 - 複数の NIS, 78
- ドメインネームシステム (DNS) 参照 DNS
 - IPv6 用の拡張, 24
- ドメイン名
 - NIS スレーブサーバーおよび, 81
 - 設定, 70
 - 定義, 124
- な**
- 名前解決
 - 定義, 124
- 名前空間
 - 定義, 124
- 認証
 - 定義, 124
- ネーミング
 - DNS, 23
 - IPv6 拡張, 24
 - LDAP, 23
 - NIS, 22
 - Oracle Solaris ネームサービス, 20
 - 概要, 13
 - 比較, 24
 - ファイルベースの, 22
- ネームサーバー
 - 定義, 124
- ネームサービス
 - 定義, 124
- ネームサービススイッチ

auto_home テーブル, 27
auto_master テーブル, 27
DNS および, 25
mDNS および, 44
NIS, 56
NOTFOUND=continue 検索条件, 29
nsswitch.conf ファイルからの移行, 32
SMF プロパティ, 26
SUCCESS=return 検索条件, 29
timezone テーブルおよび, 26
TRYAGAIN=3 検索条件, 29
TRYAGAIN=forever 検索条件, 29
UNAVAIL=continue 検索条件, 29
アクション, 28
インターネットアクセス, 25
オプション, 28
概要, 25
検索条件, 29
構成
 検索条件, 31
 データベースのソース, 30
 デフォルトのソース, 32
ステータスメッセージ, 28, 29
ソース形式, 27
定義, 124
データベース, 26
パスワード情報, 33
パスワードデータおよび, 33
変更, 29
メッセージ, 28
ネットワークサービス
 DNS および, 36
ネットワークパスワード 参照 Secure RPC パスワード
ネットワークマスク
 定義, 124
ノード名
 設定, 70

は

パスワード
 NIS, 92
 rpc.yppasswdd デーモン, 92
パスワードデータ

NIS, 71, 72
NIS および, 89
NIS マップのルート, 89
 ネームサービススイッチ, 33
非公開鍵
 定義, 124
ファイル
 DNS, 46
ファイルベースのネーミング, 22
フィールド
 定義, 124
ブロードキャスト
 NIS のバインド, 65
ホスト コマンド
 説明, 47
ホスト (マシン)
 NIS ドメインの変更, 107
ホスト名
 設定, 70

ま

マシン
 NIS クライアント, 57
 NIS サーバー, 57
マスターサーバー
 定義, 125
マッピング
 定義, 125
マルチキャスト DNS 参照 mDNS
メール交換レコード
 定義, 125
メールホスト
 定義, 125

や

ユーザー
 NIS, 90
 NIS パスワード, 92
 passwd マップの更新, 91
 useradd, 90
 userdel (NIS), 91
 ネットグループ, 92, 93
ユーザーの承認

- DNS, 40

- 5
- レコード
 - 定義, 125

- A**
- AAAA レコード, 24
- Active Directory
 - AD ネームサービス, 49
 - nss_ad の構成, 50
 - クライアントの設定, 49
 - 取得
 - group 情報, 53
 - passwd 情報, 52
 - shadow 情報, 53
 - パスワードの更新, 52
- adjunct ファイル, 75
- aliases ファイル, 73
- audit_attr マップ
 - 説明, 61
- audit_user マップ
 - 説明, 61
- auto_direct.time マップ, 98
- auto_home.time マップ, 98
- auto_home テーブル
 - ネームサービススイッチおよび, 27
- auto_master テーブル
 - ネームサービススイッチおよび, 27

- B**
- baseDN
 - 定義, 125
- bootparams マップ
 - 説明, 61

- C**
- CHKPIPE, 99
- crontab ファイル
 - NIS の問題および, 118

- ypxfr および, 102

- D**
- daemons
 - NIS
 - 実行中でない, 117
- databaseID
 - 定義, 125
- dbm ファイル, 105, 106
- DES
 - 定義, 123, 125
- dig コマンド
 - 説明, 46, 47
- DIR ディレクトリ, 73
- DIT 参照 ディレクトリ情報ツリー
- DN
 - 定義, 125
- dns-sd コマンド
 - 説明, 46, 47
 - リソースの通知, 45
- DNS
 - NIS および, 55, 56, 107
 - SMF および, 36
 - 概要, 21, 35
 - 関連情報, 36
 - コマンド, 46
 - コンパイルフラグ, 48
 - タスク, 38
 - 定義, 125, 124
 - デーモン, 46
 - ネームサービススイッチおよび, 25
 - ファイル, 46
 - ユーザーの承認, 40
 - リソースの通知, 45
- DNS クライアント
 - インストール, 41
- DNS サーバー
 - オプションの構成, 40
 - 構成, 38
- DNS サービス検出
 - 概要, 22, 36
 - 構成, 44
- DNS ゾーン
 - 定義, 125

DNS ゾーンファイル

定義, 125

DNS 転送

定義, 126

DNS パッケージ

インストール, 38

dnssec-dsfromkey コマンド

説明, 46, 47

dnssec-keyfromlabel コマンド

説明, 46, 47

dnssec-keygen コマンド

説明, 46, 47

dnssec-signzone コマンド

説明, 46, 47

DOM 変数, 77, 78

domainname コマンド

NIS および, 113

E

/etc/inet/hosts ファイル, 14

NIS スレーブサーバーおよび, 82

/etc/mail/aliases ファイル, 73

/etc/mail ディレクトリ, 74

/etc/named.conf ファイル

DNS ユーザーの承認, 40

構成の検証, 42

説明, 46

/etc/rndc.conf ファイル

説明, 46

/etc files

naming and, 20

/etc ファイル, 60

ethers.byaddr マップ

説明, 62

ethers.byname マップ

説明, 62

exec_attr マップ

説明, 62

F

FMRI

mDNS, 44

NIS, 68

G

getaddrinfo()

ネームサービススイッチおよび, 25

gethostbyname()

ネームサービススイッチおよび, 25

getpwnam()

ネームサービススイッチおよび, 25

getpwuid()

ネームサービススイッチおよび, 25

getXbyY() インタフェース

ネームサービススイッチおよび, 25

group.bygid マップ

説明, 62

group.byname マップ

説明, 62

H

host.byaddr マップ

説明, 62

host.byname マップ

説明, 62

host コマンド

説明, 46

hosts.byaddr マップ, 61

hosts.byname マップ, 61

hosts データベース, 100

hosts ファイル

NIS スレーブサーバーおよび, 82

I

IP

定義, 126

IP アドレス

定義, 126

L

LAN

定義, 125

LDAP

定義, 126
Lightweight Directory Access Protocol 参照 LDAP

M

mail.aliases マップ
説明, 62
mail.byaddr マップ
説明, 62
make コマンド
Makefile の構文, 97
NIS マップ, 63
ypinit および, 77
説明, 60
マップの更新後, 101
makedbm コマンド
Makefile および, 75
make コマンドおよび, 61
ypinit および, 77
スレーブサーバーの追加, 84
説明, 60
デフォルト以外のマップおよび, 104
マップサーバーの変更, 95
Makefile の NOPUSH, 99
Makefile ファイル
NIS, 61
NIS セキュリティー, 90
NIS への変換および, 74
passwd マップおよび, 75
オートマウントマップおよび, 98
準備, 74
ソースディレクトリの変更, 71, 75
デフォルト以外のマップ
変更, 104
プライマリサーバーの設定, 77
マップ
サポートされるリスト, 96
マップのマスターサーバーの変更, 95
mapname.dir ファイル, 75
mapname.pag ファイル, 75
mDNS
エラーログ, 45
概要, 22, 35
構成, 44
MIS

定義, 126

N

named-checkconf コマンド
/etc/named.conf ファイルの検証, 42
DNS サーバーの構成, 38
説明, 46, 47
named-checkzone コマンド
説明, 46, 47
named-compilezone コマンド
説明, 46, 47
named.conf ファイル 参照 /etc/named.conf
ファイル
named デーモン
SMF および, 36
構成ファイル
説明, 46
コンパイルフラグの表示, 48
説明, 46, 47
ユーザーの承認と, 40
ndbm 形式, 74
NIS マップおよび, 60
netgroup.byhost マップ
概要, 92
説明, 62
netgroup.byuser マップ
概要, 92
説明, 62
netgroup マップ
エントリ, 93
概要, 92
netid.byname マップ
説明, 62
netmasks.byaddr マップ
説明, 62
networks.byaddr マップ
説明, 62
networks.byname マップ
説明, 62
nicknames ファイル, 64
NIS, 22
/var/yp/domainname ディレクトリおよび, 61
DNS および, 56, 107
Makefile, 61

- Makefile の準備, 74
- Makefile のフィルタリング, 97
- ndbm 形式, 60
 - 「not responding」メッセージ, 112
- passwd マップの更新, 91
- passwd マップの自動更新, 102
- rpc.yppasswdd デーモン, 92
- SMF および, 68
 - 「unavailable」メッセージ, 112
- useradd, 90
- userdel, 91
- ypbind デーモン, 65
- ypbind の「can't」メッセージ, 111
- ypbind の障害, 115
- ypinit, 76
- ypservers ファイル, 84
- ypwhich, 65
- ypwhich の一貫性のない表示, 114
- アーキテクチャー, 56
- インターネットおよび, 56
- 概要, 55
- クライアント, 57, 58
- クライアントに関する問題, 112
- クライアントの設定, 86
- 構成ファイルの変更, 96
- 構造, 56
- コマンド, 59
- コマンドのハンガアップ, 112
- コンポーネント, 58
- サーバー, 57, 57
- サーバーが過負荷および, 116
- サーバーが使用できない, 113
- サーバーのバインディングが不可能, 114
- サーバー, 別のバージョンのマップ, 117
- サーバーリストによるバインド, 65
- 自動起動, 79
- 手動のバインド, 106
- 準備, 67
- スレーブサーバー, 57
- スレーブサーバーの設定, 81, 81
- セキュリティ, 89
- 設定の準備, 71
- ソースファイル, 71, 72
- 定義, 126
- 停止, 108, 108
- デーモン, 59
- デーモンの起動, 78
- ドメイン, 56, 58
- ドメイン名, 70
- ネットグループ, 92, 93
- バインド, 64
- パスワードデータ, 71, 72
- 複数のドメイン, 78
- ブロードキャストによるバインド, 66
- マスターサーバー, 57
- 問題, 111
- ユーザー, 管理, 90
- ユーザーパスワード, 92
- ユーザーパスワードのロック, 90
- ルートエントリ, 89
- NIS クライアント
 - サーバーにバインドされない, 113
- NIS サーバー
 - 誤動作, 117
- NIS スレーブサーバー
 - 初期設定, 85
 - 追加, 84
- NIS デーモン
 - 実行中でない, 117
- NIS ドメイン
 - 変更, 107
- NIS ドメイン名
 - 指定されていない, 113
 - 間違っている, 113
- NIS ホスト
 - ドメインの変更, 107
- NIS マップ
 - /var/yp/domainname ディレクトリおよび, 61
 - Makefile DIR 変数, 98
 - Makefile DOM 変数, 98
 - Makefile PWDIR 変数, 98
 - Makefile および, 97
 - Makefile の CHKPIPE, 99
 - Makefile の NOPUSH, 99
 - Makefile の yppush, 99
 - Makefile のフィルタリング, 97
 - Makefile 変数の変更, 98
 - Makefile マクロの変更, 98
 - ndbm 形式, 60
 - 管理, 93

- キーボードからの作成, 105
- 検索, 64
- 更新, 63
- 構成ファイルの変更, 96
- サーバーの変更, 95
- 作業, 63
- 作成, 63
- 定義, 126
- デフォルト, 61
- デフォルト以外, 100
- 内容の表示, 64, 94
- ニックネーム, 64
- の一覧, 61
- ファイルからの作成, 105
- 「not responding」メッセージ (NIS), 112
- NOTFOUND=continue 検索条件
- ネームサービススイッチおよび, 29
- nscd デーモン
- 説明, 59
- nscfg コマンド
- 説明, 46, 47
- nslookup コマンド
- 説明, 47, 47
- nsswitch.conf
- SMF への移行, 32
- nsupdate コマンド
- 説明, 47, 47

O

- Oracle Solaris ネームサービス, 20
- およびサービス管理機能 (SMF), 21

P

- /PWDIR/shadow ファイル, 75
- /PWDR/security/passwd.adjunct, 75
- passwd.adjunct.byname マップ
- 説明, 62
- passwd.adjunct ファイル, 75, 96
- passwd.byname マップ
- 説明, 62
- passwd.byuid マップ
- 説明, 62

- passwd
- 自動更新された NIS マップ, 102
- passwd コマンド, 92
- passwd ファイル
- Solaris 1.x フォーマット, 90
- passwd マップ, 72
- ユーザー, 追加, 91
- prof_attr マップ
- 説明, 63
- protocols.byname マップ
- 説明, 63
- protocols.bynumber マップ
- 説明, 63
- publickey.byname マップ
- 説明, 62
- PWDIR, 72

R

- rndc-confgen コマンド
- DNS サーバーの構成, 38
- rndc.conf ファイルの作成, 39
- 説明, 47, 48
- rndc.conf ファイル
- 作成, 39
- rndc コマンド
- 構成ファイル
- 説明, 46
- 説明, 47, 48
- rpc.bynumber マップ
- 説明, 63
- rpc.yppasswdd デーモン
- NIS パスワードおよび, 92
- passwd コマンドによるマップの更新, 102
- 説明, 59
- rpc.yppupdated デーモン
- 説明, 59
- RPC
- 定義, 125, 126

S

- SASL
- 定義, 126

- searchTriple
 定義, 126
- Secure RPC パスワード
 定義, 126
- Secure Sockets Layer 参照 SSL
- services.byname マップ
 説明, 63
- services.byservice マップ
 説明, 63
- shadow ファイル, 75
 Solaris 1.x フォーマット, 90
- sites.byname マップ
 マップサーバーの変更, 95
- SMF, 78
 DNS および, 36
 NIS および, 68
- SSL
 定義, 127
- SUCCESS=return 検索条件
 ネームサービススイッチおよび, 29
- svc:/network/dns/client
 説明, 37
- svc:/network/dns/server
 説明, 37
- svcadm
 NIS での, 85
- T**
- TCP/IP
 定義, 127
- TCP 参照 Transport Control Protocol
- timezone テーブル, 26
- TLS 参照 Transport Layer Security
- Transport Control Protocol
 定義, 127
- Transport Layer Security
 定義, 127
- U**
- /usr/bin/dns-sd コマンド
 説明, 46, 47
- /usr/sbin/dig コマンド
 説明, 46, 47
- /usr/sbin/dnssec-dsfromkey コマンド
 説明, 46, 47
- /usr/sbin/dnssec-keyfromlabel コマンド
 説明, 46, 47
- /usr/sbin/dnssec-keygen コマンド
 説明, 46, 47
- /usr/sbin/dnssec-signzone コマンド
 説明, 46, 47
- /usr/sbin/host コマンド
 説明, 46, 47
- /usr/sbin/makedbm コマンド
 デフォルト以外のマップの変更, 104
- /usr/sbin/named-checkconf コマンド
 説明, 46, 47
- /usr/sbin/named-checkzone コマンド
 説明, 46, 47
- /usr/sbin/named-compilezone コマンド
 説明, 46, 47
- /usr/sbin/named デーモン
 説明, 46, 47
- /usr/sbin/nscfg コマンド
 説明, 46, 47
- /usr/sbin/nslookup コマンド
 説明, 47, 47
- /usr/sbin/nsupdate コマンド
 説明, 47, 47
- /usr/sbin/rndc-configgen コマンド
 説明, 47, 48
- /usr/sbin/rndc コマンド
 説明, 47, 48
- UNAVAIL=continue 検索条件
 ネームサービススイッチおよび, 29
 「unavailable」メッセージ (NIS), 112
- user_attr マップ
 説明, 63
- useradd, 90
 パスワードのロック, 90
- userdel, 91, 91
- usermod コマンド
 DNS ユーザーの承認, 40
- V**
- /var/spool/cron/crontabs/root ファイル

- NIS の問題および, 118
- /var/svc/log/network-dns-multicast:default.log ファイル, 45
- /var/yp/binding/domainname/ypservers ファイル, 114
- /var/yp/domainname ディレクトリ, 61
- /var/yp/Makefile, 77
 - マップ
 - サポートされるリスト, 96
- /var/yp/mymap.asc ファイル, 105
- /var/yp/nicknames ファイル, 64
- /var/yp ディレクトリ
 - NIS セキュリティー, 90

W

- WAN
 - 定義, 122

X

- X.500
 - 定義, 127

Y

- yp
 - 定義, 127
- ypbind デーモン, 78
 - 「can't」メッセージ, 111
 - 過負荷のサーバーおよび, 116
 - クライアントがバインドされない, 113
 - サーバーリストモード, 65
 - 障害, 115
 - スレーブサーバーの追加, 85
 - 説明, 59
 - ブロードキャストモード, 66, 86
- ypcat コマンド, 64
 - 説明, 60
- ypinit コマンド
 - Makefile ファイルおよび, 74
 - make コマンドおよび, 77
 - ypserv の起動, 79

- クライアントの設定, 86
- スレーブサーバーおよび, 81
- スレーブサーバーの初期化, 81
- スレーブサーバーの追加, 85
- 説明, 60
- デフォルトのマップ, 100
- マスターサーバーの設定, 76
- ypmatch コマンド
 - 説明, 60
- yppush コマンド
 - Makefile および, 99
 - NIS の問題, 119
 - 説明, 60
 - マップサーバーの変更, 96
- ypserv デーモン, 66, 78
 - 過負荷のサーバーおよび, 116
 - 障害, 119
 - 説明, 59
 - ブロードキャストモード, 66
- ypservers ファイル
 - NIS のトラブルシューティング, 114
 - 作成, 84
 - スレーブサーバーの追加, 84
- ypservers マップ
 - NIS の問題, 119
 - 説明, 63
- ypset コマンド
 - 説明, 60
- ypwhich コマンド
 - 説明, 60
 - バインドされたサーバーの識別, 65
 - 表示に一貫性がない, 114
 - マスターサーバーの識別, 64
- ypxfr コマンド
 - crontab ファイルおよび, 102
 - 新しいマップのスレーブサーバーへの転送, 105
 - シェルスクリプト, 118
 - 出力のをロギング, 118
 - 説明, 60
 - マップサーバーの変更, 96
- ypxfrd デーモン
 - 説明, 59