



Solaris のシステム管理 (基本編)

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 817-7238-10
2004 年 9 月

Copyright 2004 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

本製品およびそれに関連する文書は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社による事前の許可なく、本製品および関連する文書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびに他の国における登録商標です。フォント技術を含む第三者のソフトウェアは、著作権により保護されており、提供者からライセンスを受けているものです。

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

本製品に含まれる HG-MinchoL, HG-MinchoL-Sun, HG-PMinchoL-Sun, HG-GothicB, HG-GothicB-Sun, および HG-PGothicB-Sun は、株式会社リコーがリョービマジクス株式会社からライセンス供与されたタイプフェースマスタをもとに作成されたものです。HeiseiMin-W3H は、株式会社リコーが財団法人日本規格協会からライセンス供与されたタイプフェースマスタをもとに作成されたものです。フォントとして無断複製することは禁止されています。

Sun, Sun Microsystems, docs.sun.com, AnswerBook, AnswerBook2, JumpStart, Sun Ray, Sun Blade, PatchPro, SunOS, Solstice, Solstice AdminSuite, Solstice DiskSuite, Solaris Solve, Java, JavaStation, OpenWindows, NFS, iPlanet, および Netra は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。

サン・ロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャに基づくものです。

OPENLOOK, OpenBoot, JLE は、サン・マイクロシステムズ株式会社の登録商標です。

Wnn は、京都大学、株式会社アステック、オムロン株式会社で共同開発されたソフトウェアです。

Wnn6 は、オムロン株式会社、オムロンソフトウェア株式会社で共同開発されたソフトウェアです。© Copyright OMRON Co., Ltd. 1995-2000. All Rights Reserved. © Copyright OMRON SOFTWARE Co., Ltd. 1995-2002 All Rights Reserved.

「ATOK」は、株式会社ジャストシステムの登録商標です。

「ATOK Server/ATOK12」は、株式会社ジャストシステムの著作物であり、「ATOK Server/ATOK12」にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本製品に含まれる郵便番号辞書 (7 桁/5 桁) は郵政事業庁が公開したデータを元に制作された物です (一部データの加工を行なっています)。

本製品に含まれるフェイスマーク辞書は、株式会社ビレッジセンターの許諾のもと、同社が発行する『インターネット・パソコン通信フェイスマークガイド '98』に添付のものを使用しています。© 1997 ビレッジセンター

Unicode は、Unicode, Inc. の商標です。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザインタフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

DtComboBox ウィジェットと DtSpinBox ウィジェットのプログラムおよびドキュメントは、Interleaf, Inc. から提供されたものです。(© 1993 Interleaf, Inc.)

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社による事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典: *System Administration Guide: Basic Administration*

Part No: 817-6958-10

Revision A



040701@9061



目次

はじめに	13
1 Solaris 管理ツール (製品概要)	19
Solaris 管理ツールの新機能	19
Solaris 管理ツールのサポート状況のマトリックス	20
Solaris 9 管理ツールの機能説明	21
Solaris 8 管理ツールの機能説明	22
以前の Solaris 管理ツールの機能説明	24
Solaris 管理コマンドの有効性	24
Solaris 9 システム管理コマンド	25
Solaris 8 システム管理コマンド	26
以前の Solaris 管理コマンドの説明	26
Solaris 管理ツールの詳細情報	27
2 Solaris 管理コンソールの操作 (手順)	29
Solaris 管理コンソール (概要)	29
Solaris 管理コンソールについて	29
Solaris 管理コンソールツール	30
Solaris 管理コンソールを使用する理由	32
Solaris 管理コンソールの構成	33
Solaris 管理コンソールウィンドウの変更	33
Solaris 管理コンソールのマニュアル	34
役割によるアクセス制御の程度	34
スーパーユーザー (root) になるか役割を引き受ける	35
▼ スーパーユーザー (root) になるか役割を引き受ける方法	36

Solaris 管理ツールを RBAC と組み合わせて使用する (作業マップ)	38
コンソールに最初にログインした場合	39
プライマリ管理者の役割を作成する	39
▼ 最初の役割 (プライマリ管理者) を作成する方法	41
▼ プライマリ管理者の役割を引き受ける方法	41
Solaris 管理コンソールを起動する	42
▼ スーパーユーザーまたは役割としてコンソールを起動する方法	42
ネームサービス環境で Solaris 管理ツールを使用する (作業マップ)	44
RBAC セキュリティファイル	44
ネームサービス環境で Solaris 管理コンソールを使用するための前提条件	46
管理範囲	46
/etc/nsswitch.conf ファイル	46
▼ 特定環境用のツールボックスを作成する方法	47
▼ ツールボックスにツールを追加する方法	49
▼ ネームサービス環境で Solaris 管理コンソールを起動する方法	50
Solaris 管理コンソールにツールを追加する	50
▼ レガシーツールをツールボックスに追加する方法	50
▼ 別製品のツールをインストールする方法	51
Solaris 管理コンソールの障害追跡	52
▼ Solaris 管理コンソールの障害追跡の方法	52
3 ユーザーアカウントとグループの管理 (概要)	55
ユーザーとグループの管理における新機能	55
Solaris 管理コンソールツール群	56
Solaris のディレクトリサービス	56
プロジェクトでユーザーおよびリソースを管理する	56
ユーザーアカウントとグループとは	57
ユーザーアカウント管理のガイドライン	58
ネームサービス	58
ユーザー (ログイン) 名	58
ユーザー ID 番号	59
パスワード	62
パスワードの有効期限を設定する	63
ホームディレクトリ	63
ユーザーの作業環境	64
グループを管理するガイドライン	64
ユーザーアカウントとグループを管理するツール	65

Solaris ユーザー管理ツールで実行できる作業	66
Solaris 管理コンソールによるホームディレクトリの管理	70
ユーザーアカウントの変更	70
ユーザーアカウントの削除	71
カスタマイズしたユーザー初期設定ファイルの追加	71
パスワードの管理	71
ユーザーアカウントを無効にする	71
ユーザーアカウントとグループ情報の格納場所	72
passwd ファイルのフィールド	72
shadow ファイルのフィールド	74
group ファイルのフィールド	75
ユーザーの作業環境のカスタマイズ	77
サイト初期設定ファイルの使用方法	79
ローカルシステムへの参照を避ける	79
シェル機能	80
シェル環境	80
PATH 変数	83
ロケール変数	84
デフォルトのファイルアクセス権 (umask)	85
ユーザー初期設定ファイルとサイト初期設定ファイルの例	86
例 — サイト初期設定ファイル	87
4 ユーザーアカウントとグループの管理 (手順)	89
ユーザーアカウントの設定 (作業マップ)	89
ユーザー情報を収集する方法	90
▼ ユーザー初期設定ファイルをカスタマイズする方法	91
▼ Solaris 管理コンソールのグループツールを使ってグループを追加する方法	93
▼ Solaris 管理コンソールのユーザーツールを使ってユーザーを追加する方法	94
例 — Solaris 管理コンソールのグループツールを使ってユーザーを追加する	94
CLI ツールを使ってグループやユーザーを追加する方法	95
▼ ユーザーのホームディレクトリを共有する方法	96
▼ ユーザーのホームディレクトリをマウントする方法	97
ユーザーアカウントの管理 (作業マップ)	99
Solaris ユーザー登録	99
Solaris Solve へのアクセス	100
Solaris ユーザー登録の問題の障害追跡	101
▼ Solaris ユーザー登録をやり直す方法	101

▼ ユーザー登録を無効にする方法	102
5	サーバーとクライアントサポートの管理 (概要) 103
サーバーおよびクライアント管理の新機能	103
ディスクレスクライアントサポート	104
サーバーとクライアントタスクの操作手順	104
サーバー、クライアント、およびアプライアンスとは	105
クライアントサポートとは	105
システムタイプの概要	106
サーバー	107
スタンドアロンシステム	107
ディスクレスクライアント	108
アプライアンス	108
システムタイプ選択のガイドライン	108
ディスクレスクライアント管理の概要	109
OS サーバーおよびディスクレスクライアントのサポート情報	110
ディスクレスクライアント管理機能	110
OS サーバーに必要なディスク容量	113
6	ディスクレスクライアントの管理 (手順) 115
ディスクレスクライアントの管理 (作業マップ)	115
ディスクレスクライアントの管理	117
▼ ディスクレスクライアントの追加の準備	118
▼ ディスクレスクライアントサポートの OS サービスの追加方法	120
▼ ディスクレスクライアントの追加方法	121
▼ ディスクレスクライアントの起動方法	123
▼ ディスクレスクライアントサポートの削除方法	123
▼ ディスクレスクライアントの OS サービスを削除する方法	124
ディスクレスクライアント OS サービスにパッチを適用する	124
ディスクレスクライアントの OS パッチの表示	125
▼ ディスクレスクライアントの OS パッチの追加方法	125
ディスクレスクライアント問題の障害追跡	128
7	システムのシャットダウンとブート (概要) 131
システムのシャットダウンとブートに関する新機能	131
PXE ネットワークブート	132
シャットダウンとブートについての参照先	133

	シャットダウンとブートの用語	133
	システムのシャットダウンに関するガイドライン	134
	システムのブートに関するガイドライン	134
	ネットワークからのシステムのブート	135
	システムをシャットダウンする場合	135
	システムをブートする場合	136
8	実行レベルとブートファイル (手順)	139
	実行レベル	139
	システムの実行レベルを確認する方法	140
	/etc/inittab ファイル	141
	例 — デフォルトの inittab ファイル	142
	システムが実行レベル 3 になると実行される処理	143
	実行制御スクリプト	144
	実行制御スクリプトのまとめ	145
	実行制御スクリプトを使用してサービスを起動または停止する	149
	▼ 実行制御スクリプトを使用してサービスを起動または停止する方法	149
	実行制御スクリプトを追加する	150
	▼ 実行制御スクリプトを追加する方法	150
	実行制御スクリプトを無効にする	151
	▼ 実行制御スクリプトを無効にする方法	151
	x86: ブートファイル	152
9	システムのシャットダウン (手順)	155
	システムのシャットダウン	155
	システムシャットダウンコマンド	156
	システムのダウン時間をユーザーに通知する	157
	▼ システムにログインしているユーザーを知る方法	157
	▼ サーバーをシャットダウンする方法	157
	▼ スタンドアロンシステムをシャットダウンする方法	161
	すべてのデバイスの電源を落とす	162
	▼ すべてのデバイスの電源を落とす方法	163
10	SPARC: システムのブート (手順)	165
	SPARC: システムのブート (作業マップ)	165
	SPARC: ブート PROM の使用	167
	SPARC: システムの PROM リビジョンを確認する方法	167

- ▼ SPARC: システム上のデバイスを確認する方法 167
- ▼ SPARC: デフォルトのブートデバイスを変更する方法 169
- SPARC: システムをリセットする方法 171
- SPARC: システムのブート 171
 - ▼ SPARC: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法 172
 - ▼ SPARC: システムを実行レベル S (シングルユーザーレベル) でブートする方法 173
 - ▼ SPARC: システムを対話式でブートする方法 174
 - ▼ SPARC: システムをネットワークからブートする方法 175
 - ▼ SPARC: 復元を目的としてシステムを停止する方法 177
 - ▼ SPARC: 復元を目的としてシステムをブートする方法 177
 - ▼ SPARC: カーネルデバッガ (kadb) を使ってシステムをブートする方法 179
- SPARC: クラッシュダンプを強制してシステムをリブートする 180
 - ▼ SPARC: クラッシュダンプを強制してシステムをリブートする方法 180

- 11 x86: システムのブート (手順) 183**
 - x86: システムのブート (作業マップ) 183
 - x86: Solaris Device Configuration Assistant (デバイス構成用補助) のブート 185
 - ▼ x86: Solaris Device Configuration Assistant (デバイス構成用補助) をブートする方法 185
 - x86: システムのブート 185
 - ▼ x86: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法 186
 - ▼ x86: システムを実行レベル S (シングルユーザーレベル) でブートする方法 187
 - ▼ x86: システムを対話式でブートする方法 188
 - ▼ x86: システムをネットワークからブートする方法 190
 - ▼ x86: 復元を目的としてシステムを停止する方法 191
 - ▼ x86: 復元を目的としてシステムをブートする方法 191
 - ▼ x86: カーネルデバッガ (kadb) を使ってシステムをブートする方法 196
 - x86: クラッシュダンプを強制してシステムをリブートする 197
 - ▼ x86: クラッシュダンプを強制してシステムをリブートする方法 197

- 12 ブートプロセス (参照情報) 199**
 - SPARC: ブート PROM 199
 - SPARC: ブートプロセス 200
 - x86: PC BIOS 200
 - x86: ブートサブシステム 201
 - x86: Solaris リリースのブート 202

x86: デバイス識別段階で表示される画面	203
x86: ブート段階で表示されるメニュー	205
x86: ブートプロセス	206

13 ソフトウェアの管理 (概要) 209

Solaris 9 Update リリースの新しいソフトウェア管理機能	209
pkgadd と patchadd による署名付きパッケージおよびパッチのサポート	210
prodreg コマンドの拡張機能	210
Solaris 9 リリースの新しいソフトウェア管理機能	210
署名付きパッチ	211
Solaris Product Registry 3.0	211
パッチアナライザ	211
Solaris 管理コンソールのパッチマネージャ	211
ソフトウェア管理作業についての参照先	212
ソフトウェアパッケージの概要	212
署名付きパッケージおよびパッチ	213
ソフトウェアパッケージを管理するためのツール	218
ソフトウェアパッケージの追加または削除 (pkgadd)	219
ソフトウェアパッケージの追加にあたっての重要な注意点 (pkgadd)	220
パッケージの削除に関するガイドライン (pkgrm)	220
パッケージ追加時のユーザーの対話操作を省略する (pkgadd)	221
管理ファイルの使用	221
応答ファイルの使用 (pkgadd)	222

14 ソフトウェアの管理 (手順) 223

ソフトウェアパッケージを管理するためのコマンド	223
Solaris Web Start プログラムによるソフトウェアの追加	224
▼ Solaris Web Start プログラムを使ってソフトウェアをインストールする方法	225
Solaris Product Registry の GUI によるソフトウェアの管理 (作業マップ)	226
▼ インストールまたはアンインストールしたソフトウェアの情報を Solaris Product Registry の GUI を使って表示する方法	228
▼ Solaris Product Registry の GUI を使ってソフトウェアをインストールする方法	229
▼ Solaris Product Registry の GUI を使ってソフトウェアをアンインストールする方法	230
Solaris Product Registry のコマンド行インタフェースによるソフトウェアの管理 (作業マップ)	231

▼ インストールまたはアンインストールしたソフトウェアの情報を表示する方法 (prodreg)	232
▼ ソフトウェア属性を表示する方法 (prodreg)	235
▼ ソフトウェアコンポーネント間の依存関係を確認する方法 (prodreg)	237
▼ 損傷を受けたソフトウェア製品を特定する方法 (prodreg)	238
▼ ソフトウェアをアンインストールする方法 (prodreg)	241
▼ 損傷を受けたソフトウェアをアンインストールする方法 (prodreg)	245
▼ 損傷を受けたソフトウェアコンポーネントを再インストールする方法 (prodreg)	248
署名付きパッケージの追加と削除 (作業マップ)	250
▼ 信頼される証明書をパッケージキーストアにインポートする方法 (pkgadm addcert)	250
▼ 証明書の情報を表示する方法 (pkgadm listcert)	252
▼ 証明書を削除する方法 (pkgadm removecert)	253
▼ プロキシサーバーをセットアップする方法 (pkgadd)	253
▼ 署名付きパッケージを追加する方法 (pkgadd)	254
パッケージコマンドによるソフトウェアパッケージの管理 (作業マップ)	256
▼ ソフトウェアパッケージを追加する方法 (pkgadd)	256
ソフトウェアパッケージをスプールディレクトリに追加する	259
インストール済みのすべてのパッケージに関する情報を表示する方法 (pkginfo)	261
▼ インストール済みのソフトウェアパッケージの整合性を検査する方法 (pkgchk)	262
ソフトウェアパッケージを削除する	263
▼ ソフトウェアパッケージを削除する方法 (pkgrm)	264
Admintool によるソフトウェアパッケージの追加と削除 (作業マップ)	265
▼ Admintool を使ってソフトウェアパッケージを追加する方法	265
▼ Admintool を使ってソフトウェアパッケージを削除する方法	266
15 Solaris パッチの管理 (概要)	269
パッチについて	270
署名付きパッチについて	270
Solaris パッチへのアクセス	271
Solaris パッチの番号付け	272
Solaris パッチの管理用ツール	272
署名付きパッチの追加方法の選択	273

16 Solaris パッチの管理 (手順)	275
Solaris 環境でのパッチ管理 (作業マップ)	275
署名付きパッチまたは署名のないパッチの選択	276
patchadd コマンドによる署名付きパッチの追加 (作業マップ)	277
信頼される証明書をパッケージキーストアにインポートする方法 (pkgadm addcert)	277
▼ プロキシサーバーをセットアップする方法 (patchadd)	279
▼ 署名付き Solaris パッチのダウンロードと追加を手動で行う方法 (patchadd)	279
▼ 署名付き Solaris パッチのダウンロードと追加を自動で行う方法 (patchadd)	280
署名のない Solaris パッチの管理 (作業マップ)	282
署名のない Solaris パッチの情報を表示する	282
Solaris パッチの情報を表示する方法	282
署名のない Solaris パッチの追加	283
▼ 署名のない Solaris パッチのダウンロード方法	284
▼ 署名のない Solaris パッチの追加方法 (patchadd)	285
▼ 署名のない Solaris パッチの追加方法 (smpatch)	285
署名のない Solaris パッチの削除	286
▼ 署名のない Solaris パッチを削除する方法	286
索引	289

はじめに

『Solaris のシステム管理 (基本編)』は、Solaris™ システム管理に関する重要な情報を提供するマニュアルの一部です。本書は、SPARC® および x86 の両方のプラットフォームにおけるシステム管理について解説しています。

本書では、次のことを前提としています。

- SunOS 5.9 オペレーティングシステムがすでにインストールされていること
- 使用するネットワークソフトウェアの設定が完了していること

SunOS 5.9 オペレーティングシステムは、多くの機能と Solaris 共通デスクトップ環境 (CDE) を含む Solaris 製品の一部です。また、SunOS 5.9 は、AT&T System V リリース 4 オペレーティングシステムに準拠しています。

システム管理者にとって重要な Solaris 9 リリースの新機能については、各章のはじめにある新機能に関する節を参照してください。

注 - Solaris オペレーティングシステムは、SPARC と x86 という 2 種類のハードウェア (プラットフォーム) で動作します。また、Solaris オペレーティングシステムは、64 ビットのアドレス空間でも、32 ビットのアドレス空間でも動作します。このマニュアルで説明する情報は、章、節、注、箇条書き項目、図、表、例、コード例などで特に明記しないかぎり、両方のプラットフォーム、およびアドレス空間に適用されます。

注 - このマニュアル内で引用する第三者の Web サイトの可用性について Sun は責任を負いません。こうしたサイトやリソース上の、またはこれらを通じて利用可能な、コンテンツ、広告、製品、その他の素材について、Sun は推奨しているわけではなく、Sun はいかなる責任も負いません。こうしたサイトやリソース上の、またはこれらを経由して利用可能な、コンテンツ、製品、サービスを利用または信頼したことに伴って発生した (あるいは発生したと主張される) いかなる損害や損失についても、Sun は一切の責任を負いません。

対象読者

このマニュアルは、Solaris 9 リリースを実行するシステムの管理者を対象にしています。このマニュアルを読むには、UNIX[®] のシステム管理について 1 ～ 2 年の経験が必要です。UNIX システム管理のトレーニングコースに参加することも知識の習得に役立ちます。

Solaris システム管理マニュアルセットの構成

システム管理マニュアルセットに含まれる各マニュアルとその内容は、次のとおりです。

マニュアル名	内容
『Solaris のシステム管理 (基本編)』	ユーザーアカウントとグループ、サーバーとクライアントのサポート、システムのシャットダウンとブート、およびソフトウェアの管理 (パッケージとパッチ)
『Solaris のシステム管理 (上級編)』	印刷サービスの管理、端末とモデムの設定、システムリソースの管理 (ディスク割り当て、アカウントティング、および crontab ファイルの管理)、システムプロセスの管理、および Solaris ソフトウェアの障害追跡
『Solaris のシステム管理 (デバイスとファイルシステム)』	リムーバブルメディア、ディスクとデバイス、ファイルシステム、およびデータのバックアップと復元
『Solaris のシステム管理 (IP サービス)』	TCP/IP ネットワーク、IPv4 と IPv6、DHCP、IP セキュリティ、モバイル IP、および IP ネットワークのマルチパス化
『Solaris のシステム管理 (ネーミングとディレクトリサービス : DNS、NIS、LDAP 編)』	DNS、NIS、および LDAP のネーミングとディレクトリサービスの管理
『Solaris のシステム管理 (ネーミングとディレクトリサービス : FNS、NIS+ 編)』	FNS および NIS+ のネーミングとディレクトリサービスの管理
『Solaris のシステム管理 (資源管理とネットワークサービス)』	リソース管理、リモートファイルシステム、メール、SLP、PPP
『Solaris のシステム管理 (セキュリティサービス)』	監査、PAM、RBAC、および SEAM

この Solaris リリースに含まれているオープンソースソフトウェアのライセンス条件、特性、著作権の記述を表示する場合、デフォルトのパスは `/usr/share/src/freeware-name` または `/usr/sfw/share/src/freeware-name` です。Solaris オペレーティングシステムがデフォルトの場所以外にインストールされている場合は、所定のパスを修正して、インストールした場所にあるファイルにアクセスします。

Sun のオンラインマニュアル

`docs.sun.com` では、Sun が提供しているオンラインマニュアルを参照することができます。マニュアルのタイトルや特定の主題などをキーワードとして、検索を行うこともできます。URL は、`http://docs.sun.com` です。

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-1 表記上の規則

字体または記号	意味	例
<code>AaBbCc123</code>	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	<code>.login</code> ファイルを編集します。 <code>ls -a</code> を使用してすべてのファイルを表示します。 <code>system%</code>
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	<code>system% su</code> <code>password:</code>
<i>AaBbCc123</i>	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、 <code>rm filename</code> と入力します。
『 』	参照する書名を示します。	『コードマネージャ・ユーザーズガイド』を参照してください。

表 P-1 表記上の規則 (続き)

字体または記号	意味	例
「」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	sun% grep `^#define \ XV_VERSION_STRING`

コード例は次のように表示されます。

- C シェル

```
machine_name% command y|n [filename]
```

- C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

- Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

- Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

一般規則

このマニュアルでは、英語環境での画面イメージを使っています。このため、実際に日本語環境で表示される画面イメージとこのマニュアルで使っている画面イメージが異なる場合があります。本文中で画面イメージを説明する場合には、日本語のメニュー、ボタン名などの項目名と英語の項目名が適宜併記されています。

このマニュアルでは、「x86」という用語は、Intel 32ビット系列のマイクロプロセッサチップ、およびAMDが提供する互換マイクロプロセッサチップを意味します。

- このマニュアル中の手順を実行したり、例 (コマンド入力、コードなど) を使用する場合には、二重引用符 (""), 左一重引用符 ('), 右一重引用符 (') をそれぞれ間違えないように注意してください。
- このマニュアル中で「Return キー」と表記しているキーは、キーボードによっては「Enter キー」という名前になっていることがあります。
- /sbin、/usr/sbin、/usr/bin、/etc ディレクトリにあるコマンドについては、このマニュアルでは絶対パス名で表記していない場合があります。ただし、それ以外のあまり一般的でないディレクトリにあるコマンドについては、このマニュアル中の例では絶対パスで表記します。
- このマニュアル中の例は、SunOS ソフトウェアが標準的にインストールされていることを前提としています。つまり、バイナリ互換パッケージがインストールされていることや /usr/ucb が検索パスに設定されていることは、前提としていません。



注意 - /usr/ucb を検索パスに設定する場合は、パスの一番最後に設定してください。ps コマンドや df コマンドなどは、SunOS コマンドと /usr/ucb コマンドとで形式やオプションがそれぞれ異なります。

第 1 章

Solaris 管理ツール (製品概要)

この章では、Solaris 管理ツールの製品概要について説明します。

- 19 ページの「Solaris 管理ツールの新機能」
- 20 ページの「Solaris 管理ツールのサポート状況のマトリックス」
- 21 ページの「Solaris 9 管理ツールの機能説明」
- 22 ページの「Solaris 8 管理ツールの機能説明」
- 24 ページの「以前の Solaris 管理ツールの機能説明」
- 24 ページの「Solaris 管理コマンドの有効性」
- 27 ページの「Solaris 管理ツールの詳細情報」

Solaris 管理ツールの新機能

Solaris 9 リリースでは、次のツールが新たに追加または変更されました。

- ディスクレスクライアントサポート
- Solaris DHCP
- リソース管理
- Solaris 管理コンソールツール群
- Solaris ボリュームマネージャ (旧 Solstice™ DiskSuite)
- Solaris パッチマネージャ
- Product Registry

次の表に、各ツールの概要と詳しい説明の参照先を示します。

表 1-1 Solaris 9 で新たに追加または変更された Solaris 管理ツール

Solaris 管理ツール	説明	参照先
ディスクレスクライアントサポ ート	ディスクレスクライアントシ ステムを管理するためのコマ ンド行インタフェースを提供 する	第 6 章
リソース管理	利用可能なシステムリソース をアプリケーションでどのよ うに使用するかを管理でき る	『Solaris のシステム管理 (資源管理とネットワーク サービス)』
Solaris DHCP	ネットワークで DHCP を管理 する際のパフォーマンス、容 量、柔軟性が改善される	『Solaris のシステム管理 (IP サービス)』の「Solaris DHCP (概要)」
Solaris 管理コンソール ¹	各種の GUI ベースシステム 管理ツールの起動ポイントと して機能する	このマニュアルとコンソ ールのオンラインヘルプ
Solaris ポリウムマネージャ (旧 Solstice™ DiskSuite)	信頼性の高い記憶装置の管 理を行い、Solaris 管理コン ソールから起動される。コ マンド行インタフェースも 使用できる	『Solaris ポリウムマネ ージャの管理』
Solaris パッチマネージャ	このツールを使うと、署名 付きまたは署名なしのパ ッチをシステムに追加でき る	第 15 章
Solaris Product Registry	prodreg コマンドには、So laris Product Registry グラフィカルユーザーイ ンタフェースと同様に、 browse、info、unregis ter、uninstall の各サブ コマンドが含まれている	231 ページの「Solaris Product Registry のコ マンド行インタフェース によるソフトウェアの管 理 (作業マップ)」

¹ このツールを Sun Management Center (SunMC) と混同しないでください。Sun Management Center 製品については、<http://www.sun.com/solaris/sunmanagementcenter/docs> を参照してください。

Solaris 管理ツールのサポート状況のマトリックス

この節では、主としてユーザー、グループ、クライアント、ディスク、プリンタ、シリアルポートの管理に使われるツールについて説明します。

次の表に、Solaris 管理用の各種 GUI ツールと、それらの現在のサポート状況を示します。

表 1-2 Solaris 管理ツールのサポート状況を示すマトリックス

	Solaris 2.6 以前のリリース	Solaris 7	Solaris 8	Solaris 9
admintool	O	O	O	O
Solstice AdminSuite 2.3	O	O	X	X
Solstice AdminSuite 3.0	O (Solaris 2.6 リリースのみ)	O	O	X
Solaris Management Tools 1.0	O	O	O	X
Solaris Management Tools 2.0	X	X	O (Solaris 8 01/01、4/01、7/01、10/01、2/02 の各リリースのみ)	X
Solaris Management Tools 2.1	X	X	X	O

コンソールにテキストベースの端末を使用しているシステムで管理作業を行う場合は、Solaris 管理コンソールのコマンドを使用してください。詳細については、表 1-6 を参照してください。

Solaris 9 管理ツールの機能説明

次の表に、Solaris 9 リリースで使用できるツールを示します。

表 1-3 Solaris 9 管理ツールの機能説明

機能またはツール	admintool でのサポート	Solaris 管理コンソール 2.1 でのサポート
コンピュータとネットワークツール	なし	あり

表 1-3 Solaris 9 管理ツールの機能説明 (続き)

機能またはツール	admintool でのサポート	Solaris 管理コンソール 2.1 でのサポート
ディスクレスクライアントサポート	なし	あり。ディスクレスクライアント CLI を使用できる
ディスクツール	なし	あり
拡張ディスクツール (Solaris ボリュームマネージャ)	なし	あり
ジョブスケジューラ	なし	あり
ログビューア	なし	あり
メール別名サポート	なし	あり
マウントと共有ツール	なし	あり
ネームサービスサポート	なし	ユーザー、グループ、ネットワーク情報に対してのみ
パッチツール	なし	あり
パフォーマンスツール	なし	あり
プリンタサポート	あり	Solaris Print Manager は個別に使用できる
プロジェクトツール	なし	あり
RBAC サポート	なし	あり
RBAC ツール	なし	あり
シリアルポートツール	あり	あり
ソフトウェアパッケージツール	あり	なし
システム情報ツール	なし	あり
ユーザー/グループツール	あり	あり

Solaris 8 管理ツールの機能説明

次の表に、Solaris 8 リリースと Solaris 8 の各種 Update リリース (更新版) で使用できるツールを示します。

表 1-4 Solaris 8 管理ツールの機能説明

機能またはツール	admintool でのサ ポート	Solstice AdminSuite 3.0 で のサポート (Solaris 8、Solaris 8 6/00、 Solaris 8 10/00 の み)	Solaris Management Console 1.0 でのサ ポート	Solaris Management Console 2.0 でのサ ポート (Solaris 8 1/01、4/01、7/01、 10/01、2/02 のみ)
ディスクレスクライ アントサポート	なし	なし	なし	なし (ただし、 ディスクレスク ライアント CLI は個別に使用で きる)
ディスクツール	なし	なし	なし	あり
ジョブスケジューラ	なし	なし	なし	あり
ログビューア	なし	あり	なし	あり
メール別名サポート	なし	あり	なし	あり
マウントと共有ツ ール	なし	あり	なし	あり
ネームサービスサ ポート	なし	あり	なし	ユーザー、グ ループ、ネット ワーク情報に対 してのみ
プリンタサポート	あり	Solaris Print Manager は使用 できる	あり	なし (ただし、 Solaris Print Manager は使用 できる)
ソフトウェア パッケージツール	あり	なし	あり	なし
RBAC サポート	なし	あり (権利のみ サポート)	なし	あり
RBAC ツール	なし	RBAC CLI は個 別に使用できる	なし	あり
シリアルポートツ ール	あり	あり	あり	あり
ユーザー/グループ ツール	あり	あり	あり	あり

以前の Solaris 管理ツールの機能説明

次の表に、Solaris 8 より前のリリースで使用できるツールを示します。

表 1-5 以前の Solaris 管理ツールの機能説明

機能またはツール	admintool でのサポート	Solstice AdminSuite 2.3 でのサポート	Solstice AdminSuite 3.0 でのサポート (Solaris 2.6 のみ)
ディスクレスクライアントサポート	なし	あり	なし
ディスクツール	なし	あり	なし
ログビューア	なし	なし	あり
メール別名サポート	なし	あり	あり
マウントと共有ツール	なし	あり	あり
ネームサービスサポート	なし	あり	あり
プリンタサポート	あり	あり	Solaris Print Manager は使用できる
RBAC サポート	なし	なし	あり (権利のみサポート)
RBAC ツール	なし	なし	RBAC CLI は個別に使用できる
シリアルポートツール	あり	あり	あり
ユーザー/グループツール	あり	あり	あり

Solaris 管理コマンドの有効性

次の一連の表に、Solaris 管理ツールと同じ作業を実行するコマンドを示します。ディスクレスクライアントサポートについては、第 6 章を参照してください。

Solaris 9 システム管理コマンド

次の表に、Solaris 管理ツールと同じ機能を備えたコマンドを示します。これらのコマンドを使用する場合は、スーパーユーザーになるか、それと同等の役割を引き受ける必要があります。コマンドの中には、ローカルシステム専用のものもあります。また、ネームサービス環境で動作するものもあります。それぞれのマニュアルページで、-D オプションを参照してください。

表 1-6 Solaris 管理コマンドの説明

コマンド	説明	マニュアルページ
smc	Solaris 管理コンソールを起動する	smc (1M)
smcron	crontab ジョブを管理する	smcron (1M)
smdiskless	ディスクレスクライアントサポートを管理する	smdiskless (1M)
smexec	exec_attr データベースのエントリを管理する	smexec (1M)
smgroup	グループエントリを管理する	smgroup (1M)
smlog	WBEM ログファイルを管理/表示する	smlog (1M)
smmultiuser	複数のユーザーアカウントに対する一括操作を管理する	smmultiuser (1M)
smosservice	OS サービスとディスクレスクライアントサポートを追加する	smosservice (1M)
smprofile	prof_attr および exec_attr データベースのプロファイルを管理する	smprofile (1M)
smrole	役割アカウントの役割とユーザーを管理する	smrole (1M)
smserialport	シリアルポートを管理する	smserialport (1M)
smuser	ユーザーエントリを管理する	smuser (1M)

次の表は、コマンド行から RBAC を管理するためのコマンドです。これらのコマンドを使用する場合は、スーパーユーザーになるか、それと同等の役割を引き受ける必要があります。なお、これらのコマンドは、ネームサービス環境で RBAC 情報を管理する場合には使用できません。

表 1-7 RBAC コマンドの説明

コマンド	説明	参照ページ
auths	ユーザーに付与されている権限を表示する	auths (1)
profiles	ユーザーの実行プロファイルを表示する	profiles (1)
roleadd	システムに新しい役割を追加する	roleadd (1M)
roles	ユーザーに付与されている役割を表示する	roles (1)

次の表は、コマンド行からユーザー、グループ、RBAC 機能を管理するためのコマンドです。これらのコマンドを使用する場合は、スーパーユーザーになるか、それと同等の役割を引き受ける必要があります。なお、これらのコマンドは、ネームサービス環境でユーザーやグループの情報を管理する場合には使用できません。

表 1-8 Solaris ユーザー/グループコマンドの説明

コマンド	説明	参照ページ
useradd、usermod、 userdel	ユーザーを追加、変更、削除する	useradd (1M)、 usermod (1M)、 userdel (1M)
groupadd、groupmod、 groupdel	グループを追加、変更、削除する	groupadd (1M)、 groupmod (1M)、 groupdel (1M)

Solaris 8 システム管理コマンド

Solaris 8 リリースでは、表 1-7 および 表 1-8 に記載されているコマンドをすべて使用できます。

以前の Solaris 管理コマンドの説明

次の表は、Solstice AdminSuite™ 2.3 の GUI ツールと同等の機能を備えたコマンドです。これらのコマンドを使用する場合は、スーパーユーザーになるか、sysadmin グループのメンバーになる必要があります。

注 – Solstice AdminSuite 2.3 のコマンドのマニュアルページは、オンラインで使用できません。このため、これらのマニュアルページを見る場合は、Solstice AdminSuite 2.3 ソフトウェアにアクセスできなければなりません。

以前の Solaris リリースでは、表 1-8 に記載されているコマンドもすべて使用できません。

表 1-9 Solstice AdminSuite™ 2.3 のコマンドの説明

コマンド	説明	参照先
admhostadd, admhostmod, admhostdel, admhostls	AdminSuite ソフトウェアを使って設定したクライアント/サーバーシステムのサポートを追加、変更、削除、表示する	『Solstice AdminSuite 2.3 管理者ガイド』
admuseradd, admusermod, admuserdel, admuserls, admgroupadd, admgroupmod, admgroupdel, admgrouppls	ユーザーやグループを追加、変更、削除、表示する	『Solstice AdminSuite 2.3 管理者ガイド』

Solaris 管理ツールの詳細情報

次の表に、Solaris 管理ツールの詳細情報の参照先を示します。

表 1-10 Solaris 管理ツールの詳細情報

ツール	有効な Solaris のバージョン	参照先
Solaris 管理コンソール 2.1 のツール群	Solaris 9 リリース	このマニュアルとコンソールのオンラインヘルプ
Solaris Management Console 2.0 のツール群	Solaris 8 1/01、4/01、7/01、10/01、2/02 の各リリース	Solaris 管理コンソールのオンラインヘルプ
Solaris Management Console 1.0 のツール群	Solaris 2.6、Solaris 7、Solaris 8 の各リリース	『Solaris Easy Access Server 3.0 のインストール』
admintool	Solaris 9、Solaris 8、以前の Solaris リリース	admintool (1M)

表 1-10 Solaris 管理ツールの詳細情報 (続き)

ツール	有効な Solaris のバージョン	参照先
AdminSuite 2.3	Solaris 2.4、Solaris 2.5、 Solaris 2.5.1、Solaris 2.6、 Solaris 7 の各リリース	『Solstice AdminSuite 2.3 管理 者ガイド』
AminSuite 3.0	Solaris 8、Solaris 8 6/00、 Solaris 8 10/00 の各リリース	『Solaris Easy Access Server 3.0 のインストール』
	Solaris 9 リリース-フ ラッシュ (Flash) インストール 機能の使用を検討する	『Solaris 9 9/04 インストール ガイド』
ディスクレスクライアント CLI	Solaris 8 1/01、4/01、7/01、 10/01、2/02、Solaris 9 の各 リリース	第 6 章

第 2 章

Solaris™ 管理コンソールの操作 (手順)

この章では、システム管理作業を行うための Solaris 管理ツールの概要について説明します。この章は、Solaris 管理コンソールの起動、コンソールで使用する役割によるアクセス制御 (RBAC) の設定、ネームサービス環境での Solaris 管理ツールの操作から構成されています。

Solaris 管理コンソールによるシステム管理作業の実行手順については、以下を参照してください。

- 38 ページの「Solaris 管理ツールを RBAC と組み合わせて使用する (作業マップ)」
- 44 ページの「ネームサービス環境で Solaris 管理ツールを使用する (作業マップ)」

Solaris 管理コンソール問題の障害追跡については、52 ページの「Solaris 管理コンソールの障害追跡」を参照してください。

Solaris 管理コンソール (概要)

この節では、Solaris 管理コンソールについて説明します。

Solaris 管理コンソールについて

Solaris 管理コンソールは、「ツールボックス」というコレクションに格納されている GUI ベースの管理ツール用のコンテナです。このコンソールには、多数の基本管理ツールが含まれているデフォルトのツールボックスがあります。基本管理ツールには、ユーザー、プロジェクト、cron ジョブを管理するためのツール、ファイルシステムをマウント/共有するためのツール、ディスクやシリアルポートを管理するためのツールなどがあります。それぞれの Solaris 管理ツールについては、表 2-1 を参照してください。

いつでも既存のツールボックスにツールを追加したり、新しいツールボックスを作成することができます。

Solaris 管理コンソールには、主に次の3つの構成要素があります。

- Solaris 管理コンソールクライアント
「コンソール」と呼ばれるビジュアルインタフェースで、管理作業を行うための GUI ツールが含まれています。
- Solaris 管理コンソールサーバー
コンソールと同じマシン上またはリモートマシン上にあり、コンソールによる管理を可能にする「バックエンド」機能を提供します。
- Solaris 管理コンソールツールボックスエディタ
コンソールによく似たアプリケーションで、ツールボックスを追加または変更したり、ツールボックスにツールを追加したり、ツールボックスの適用範囲を広げたりする場合に使用します。たとえば、ネームサービスのドメインを管理するためのツールボックスを追加する場合に使用します。

コンソールを起動すると、デフォルトのツールボックスが表示されます。

Solaris 管理コンソールツール

次の表に、Solaris 管理コンソールのデフォルトのツールボックスに含まれているツールと、各ツールの説明の参照先を示します。

表 2-1 Solaris 管理コンソールツール群

カテゴリ	ツール	説明	参照先
システム状態	システム情報	日付、時間、時間帯などのシステム情報を監視および管理する	『Solaris のシステム管理 (上級編)』の「システム情報の表示と変更 (手順)」
	ログビューア	Solaris 管理コンソールツールのログとシステムログを監視および管理する	『Solaris のシステム管理 (上級編)』の「ソフトウェアの問題解決 (概要)」
	プロセス	システムプロセスを監視および管理する	『Solaris のシステム管理 (上級編)』の「プロセスとシステムのパフォーマンス」

表 2-1 Solaris 管理コンソールツール群 (続き)

カテゴリ	ツール	説明	参照先
	パフォーマンス	パフォーマンスを監視する	『Solaris のシステム管理 (上級編)』の「システムパフォーマンスの管理 (概要)」
システム構成	ユーザー	ユーザー、権利、役割、グループ、メーリングリストを管理する	57 ページの「ユーザーアカウントとグループとは」と『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」
	プロジェクト	/etc/project データベースのエントリを作成および管理する	『Solaris のシステム管理 (資源管理とネットワークサービス)』の「プロジェクトとタスク」
	コンピュータとネットワーク	コンピュータとネットワークの情報を作成および監視する	Solaris 管理コンソールのオンラインヘルプ
	バッチ	バッチを管理する	第 15 章
サービス	スケジュールされたジョブ	スケジュールされる cron ジョブを作成および管理する	『Solaris のシステム管理 (上級編)』の「システムタスクを自動的に実行する方法」
記憶装置	マウントと共有	ファイルシステムをマウントおよび共有する	『Solaris のシステム管理 (デバイスとファイルシステム)』の「ファイルシステムの管理 (概要)」
	ディスク	ディスクパーティションを作成および管理する	『Solaris のシステム管理 (デバイスとファイルシステム)』の「ディスクの管理 (概要)」
	拡張記憶装置	ボリューム、ホットスペアの集合、状態データベースの複製、ディスクセットを作成および管理する	『Solaris ボリュームマネージャの管理』
デバイスとハードウェア	シリアルポート	端末とモデムを設定する	『Solaris のシステム管理 (上級編)』の「端末とモデムの管理 (概要)」

コンテキストヘルプは、ツールの起動後に使用できるようになります。コンテキストヘルプよりも詳細なオンライン情報については、コンソールの「ヘルプ (Help)」メニューから利用できる拡張ヘルプトピックを参照してください。

Solaris 管理コンソールを使用する理由

このコンソールには、管理者にさまざまなメリットをもたらすツールセットが用意されています。コンソールは、次の処理を実行します。

- あらゆる経験レベルをサポートする

ダイアログボックス、ウィザード、コンテキストヘルプなどのグラフィカルインタフェースを使用すれば、経験の浅い管理者でも作業を完了することができます。また、経験を積んだ管理者であれば、何十または何百にも及ぶシステム上に分散している何百もの構成パラメータを管理する場合に、vi よりもこのコンソールを使用した方が便利で安全性が高いことに気がつきます。

- システムへのユーザーアクセスを管理する

デフォルトではどのユーザーもこのコンソールにアクセスできますが、初期設定を変更できるのはスーパーユーザーに限られます。『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」に説明されているように、特定のシステム変更を許可されているユーザー (通常は管理者) に割り当てられる「役割」という特殊なユーザーアカウントを作成できます。

RBAC の主なメリットは、ジョブの実行に必要な作業だけに役割を制限できることです。Solaris 管理ツールを使用する場合に、RBAC は必須ではありません。何も変更せずスーパーユーザーとしてすべてのツールを実行できます。

- コマンド行インタフェースを提供する

管理者は、必要に応じてコマンド行インタフェース (CLI) から Solaris 管理ツールを操作することができます。コマンドの中には、特に GUI ツールのユーザーを管理するコマンドなどの機能をまねて作られたものもあります。表 1-6 に、これらの新しいコマンドの名前と簡単な説明が記載されています。各コマンドのマニュアルページもあります。

専用のコマンドを持たない Solaris 管理ツール (マウントツールや共有ツールなど) では、標準の UNIX コマンドを使用します。

RBAC の機能、メリット、ユーザーサイトでのそれらのメリットの活かし方については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

RBAC を Solaris 管理ツールと組み合わせて使用する方法については、38 ページの「Solaris 管理ツールを RBAC と組み合わせて使用する (作業マップ)」を参照してください。

Solaris 管理コンソールの構成

次の図は、ユーザーツールが開いた状態でのコンソールです。

コンソールの主要部分は、次の3つの区画で構成されます。

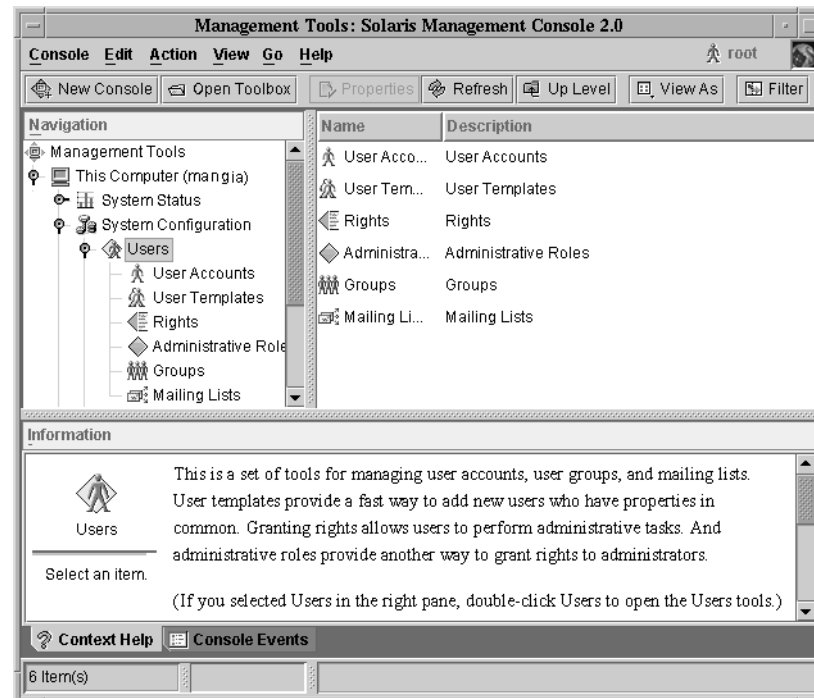


図 2-1 Solaris 管理コンソール - ユーザーツール

- ナビゲーション区画 (左側) - ツール (ツールセット)、フォルダ、他のツールボックスへのアクセス用。ナビゲーション区画内にあるアイコンは、ノードと呼ばれ、それらがフォルダやツールボックスである場合は拡張可能です。
- 表示区画 (右側) - ナビゲーション区画で選択したノードに関する情報の表示用。選択したフォルダの内容、従属ツール、選択したツールに関連付けられたデータを表示します。
- 情報区画 (下側) - コンテキストヘルプまたはコンソールイベントの表示用。

Solaris 管理コンソールウィンドウの変更

コンソールウィンドウのレイアウトは、非常に柔軟に設定することができます。次の機能を使用して、コンソールウィンドウのレイアウトを変更できます。

- 「表示 (View)」メニュー - 「表示 (View)」メニューの「表示 (Show)」オプションを使用すると、オプションのバーと区画を表示または非表示にすることができます。「表示 (View)」メニューの他のオプションは、表示区画内にあるノードの表示を制御します。
- 「コンソール (Console)」メニュー - 「設定の変更 (Preferences)」オプションを使用すると、次の設定ができます。初期ツールボックス、区画の向き、選択肢のクリックまたはダブルクリック、ツールバーのテキストまたはアイコン、フォント、デフォルトのツールロード、認証プロンプト、拡張ログイン。
- 「コンテキストヘルプ (Context Help)」と「コンソールイベント (Console Events)」のトグル - 情報区画の一番下にあるアイコンを使用すると、コンテキストヘルプとコンソールイベントの表示を切り替えることができます。

Solaris 管理コンソールのマニュアル

コンソールやそのツールの使用方法に関するマニュアルは、主にオンラインヘルプシステムから利用できます。オンラインヘルプは、コンテキストヘルプと拡張ヘルプトピックの2つの形式で利用できます。

- コンテキストヘルプは、コンソールツールの使用状況に対応しています。タブ、入力フィールド、ラジオボタンなどでカーソルをクリックすると、該当するヘルプが情報区画に表示されます。情報区画を閉じたり、開き直したりするには、ダイアログボックスやウィザードの疑問符ボタンをクリックします。
- 拡張ヘルプトピックは、「ヘルプ (Help)」メニューから使用可能であり、一部のコンテキストヘルプから相互参照リンクのクリックにより使用することもできます。これらのトピックは個別のビューアに表示され、コンテキストヘルプよりも詳細な情報が含まれています。具体的には、各ツールの概要、各ツールの機能説明、特定のツールで使用されるファイル、障害追跡などです。

各ツールの概要については、表 2-1 を参照してください。

役割によるアクセス制御の程度

32 ページの「Solaris 管理コンソールを使用する理由」に説明されているように、Solaris 管理ツールを使用する主な利点は、役割によるアクセス制御 (RBAC) を使用できることです。RBAC を使用すると、管理者はジョブの実行に必要なツールとコマンドだけを使用できます。

組織のセキュリティ要件に応じて、さまざまなレベルの RBAC を使用できます。

RBAC の方法	説明	参照先
RBAC を使用しない	すべての作業をスーパーユーザーとして実行できる。ユーザーとしてログインできる。Solaris 管理ツールを選択するときは、ユーザーとして root に入力し、root のパスワードを入力する	36 ページの「スーパーユーザー (root) になるか役割を引き受ける方法」
ルートを役割とする	匿名のルートログインを削除し、ユーザーが root としてログインできないようにする。この方法では、ユーザーが root の役割を持つ前にユーザー自身としてログインする必要がある この方法は他の役割を使用しているかどうかに関係なく適用できるので注意が必要	『Solaris のシステム管理 (セキュリティサービス)』の「root を役割にする」
単一の役割のみ	プライマリ管理者の役割を使用する。これは、root アクセス権だけを持つ場合とほぼ同じ	39 ページの「プライマリ管理者の役割を作成する」
推奨される役割	簡単に設定できる 3 つの役割 (プライマリ管理者、システム管理者、オペレータ) を使用する。これらの役割は、さまざまな責任レベルの管理者がいて、それぞれのジョブ機能が推奨される役割にほぼ合っている組織に適している	『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」
カスタムの役割	組織のセキュリティの必要性に応じて、独自の役割を追加できる	『Solaris のシステム管理 (セキュリティサービス)』の「RBAC の計画」

スーパーユーザー (root) になるか役割を引き受ける

ほとんどの管理作業 (ユーザー、ファイルシステム、プリンタの追加など) では、まず root (UID=0) としてログインするか、役割を引き受ける (RBAC を使用している場合) 必要があります。root アカウント (「スーパーユーザーアカウント」とも呼ばれる) は、システムを変更したり、緊急時にユーザーファイルの保護を無効にしたりする場合に使用します。

システムの変更がむやみに行われないように、スーパーユーザーアカウントと役割は、管理作業を実行するためだけに使用してください。スーパーユーザーアカウントに関連するセキュリティ問題は、小さな作業を行うときにでもユーザーがシステムへの完全なアクセス権を持ってしまうことにあります。

RBAC が実装されていない環境では、スーパーユーザーとしてシステムにログインすることも、su コマンドを使ってスーパーユーザーアカウントに変更することもできます。RBAC が実装されている場合は、コンソールを介して役割を引き受けることも、su を使用して役割を指定することもできます。

コンソールを使用して管理作業を行う場合は、次のいずれかを実行できます。

- ユーザーとしてコンソールにログインし、root ユーザー名とパスワードを入力する
- ユーザーとしてコンソールにログインし、役割を引き受ける

RBAC の大きなメリットは、特定の機能しか使用できないように役割を作成できることです。RBAC を使用している場合は、スーパーユーザーになるのではなく役割を引き受けることで、制限付きのアプリケーションを実行できます。

プライマリ管理者の役割の作成手順については、41 ページの「最初の役割 (プライマリ管理者) を作成する方法」を参照してください。役割を使用できるように RBAC を構成する方法については、『Solaris のシステム管理 (セキュリティサービス)』の「RBAC の構成 (作業マップ)」を参照してください。

▼ スーパーユーザー (root) になるか役割を引き受ける方法

スーパーユーザーになるか役割を引き受ける場合は、次のいずれかの方法を使用します。各方法では、スーパーユーザーのパスワードまたは役割のパスワードを知っている必要があります。

1. スーパーユーザーになる場合は、次のいずれかの方法を選択します。

- a. ユーザーとしてログインし、Solaris 管理コンソールを起動し、Solaris 管理ツールを選択して root としてログインする方法。

この方法では、コンソールから任意の管理作業を実行できます。

Solaris 管理コンソールの起動方法については、50 ページの「ネームサービス環境で Solaris 管理コンソールを起動する方法」を参照してください。

- b. システムコンソールでスーパーユーザーとしてログインする方法。

```
hostname console: root
Password: root-password
```

```
#
```

ポンド記号 (#) は、Bourne シェルにおける、スーパーユーザーアカウント用のプロンプトです。

この方法では、すべてのシステムコマンドとツールに完全にアクセスできません。

- c. ユーザーとしてログインし、コマンド行で **su** コマンドを実行してスーパーユーザーアカウントに変更する方法。

```
% su
Password: root-password
#
```

この方法では、すべてのシステムコマンドとツールに完全にアクセスできません。

- d. スーパーユーザーとしてリモートでログインする方法。この方法は、デフォルトでは使用できません。システムコンソールのスーパーユーザーとしてリモートログインするには、**/etc/default/login** ファイルを変更する必要があります。このファイルの変更方法については、『Solaris のシステム管理 (セキュリティサービス)』の「システムのセキュリティ(手順)」を参照してください。

この方法では、すべてのシステムコマンドとツールに完全にアクセスできません。

2. 役割を引き受ける場合は、次のいずれかの方法を選択します。

- a. ユーザーとしてログインし、コマンド行で **su** コマンドを実行して役割に変更する方法。

```
% su role
Password: role-password
$
```

この方法では、設定した役割がアクセスできるすべてのコマンドとツールを使用できます。

- b. ユーザーとしてログインし、**Solaris** 管理コンソールを起動し、**Solaris** 管理ツールを選択して、役割を引き受ける方法。

Solaris 管理コンソールの起動方法については、[42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」](#)を参照してください。

この方法では、引き受けた役割がアクセスできる Solaris 管理ツールを使用できます。

Solaris 管理ツールを RBAC と組み合わせて使用する (作業マップ)

この作業マップでは、スーパーユーザーアカウントを使用するのではなく、役割によるアクセス制御 (RBAC) のセキュリティ機能を使用して管理作業を実行する場合に行う作業について説明します。

注 - この節の内容は、コンソールを RBAC と組み合わせて使用方法について書かれています。最初にコンソールを使って RBAC を設定する方法について説明するため、RBAC の概要や作業にも触れています。

RBAC の詳細や、RBAC を他のアプリケーションと組み合わせて使用方法については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

作業	説明	参照先
1. コンソールを起動する	ユーザーアカウントをすでに設定してある場合は、まずユーザーとしてコンソールを起動し、次に root としてコンソールにログインする。ユーザーアカウントを設定していない場合は、まずスーパーユーザーになり、次にコンソールを起動する	42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」
2. 自分のユーザーアカウントを追加する	自分のユーザーアカウントが存在しない場合はそれを追加する	Solaris 管理コンソールのオンラインヘルプ
3. プライマリ管理者の役割を作成する	プライマリ管理者の役割を作成し、自分をこの役割に追加する	41 ページの「最初の役割 (プライマリ管理者) を作成する方法」
4. プライマリ管理者の役割を引き受ける	プライマリ管理者の役割を作成後、その役割を引き受ける	41 ページの「プライマリ管理者の役割を引き受ける方法」
5. (省略可能) root を役割にする	root を役割にし、他のユーザーが su コマンドを使用して root になれないようにルートの役割に自分を追加する	『Solaris のシステム管理 (セキュリティサービス)』の「root を役割にする」

作業	説明	参照先
6. (省略可能) 他の管理役割を作成する	他の管理役割を作成し、各役割に適切な権利を付与する。 次に、各役割に該当するユーザーを追加する	『Solaris のシステム管理 (セキュリティサービス)』の「管理役割ツールを使用して役割を作成する方法」

次の節では、Solaris 管理コンソールと RBAC のセキュリティ機能の使い方に関する概要とその手順について説明します。

コンソールに最初にログインした場合

管理者としてコンソールに最初にログインした場合は、まずユーザー (自分自身) としてコンソールを起動し、次にスーパーユーザーとしてログインします。この方法では、コンソールのすべてのツールに完全にアクセスできます。

ここで、RBAC を使用しているかどうかに応じて、一般的な手順を示します。

- RBAC を使用しない – RBAC を使用しない場合は、スーパーユーザーとして作業を続けます。他のすべての管理者も、ジョブを実行するのに root アクセス権が必要になります。
- RBAC を使用する – 次の手順を実行する必要があります。
 - ユーザーアカウントがない場合は、設定する
 - プライマリ管理者という役割を作成する
 - 作成中の役割にプライマリ管理者の権利を割り当てる
 - この役割にユーザーアカウントを割り当てる

プライマリ管理者の役割の作成手順については、[41 ページの「最初の役割 \(プライマリ管理者\) を作成する方法」](#)を参照してください。

役割を使用できるように RBAC を構成する方法については、『Solaris のシステム管理 (セキュリティサービス)』の「RBAC の構成 (作業マップ)」を参照してください。

プライマリ管理者の役割を作成する

管理役割は、特殊なユーザーアカウントの 1 つです。この役割を引き受けたユーザーは、定義済みの管理作業を実行することができます。

プライマリ管理者の役割は、スーパーユーザーと同様に、すべての管理機能の実行が許可されています。

スーパーユーザー、またはプライマリ管理者の役割を引き受けたユーザーは、他の管理者が実行できる作業を定義することができます。「管理役割を追加 (Add Administrative Role)」ウィザードを使用すると、役割を作成し、その役割に権利を付

与し、その役割を引き受けられるユーザーを指定できます。権利とは、特定のアプリケーションを使用するため、またはアプリケーション内にある特定の機能を実行するためのコマンド、つまり承認と他の権利 (その使用は管理者が付与または拒否できる) をまとめて名前を付けたものです。

プライマリ管理者の役割を作成するときは、次の情報の入力を求めるプロンプトが表示されます。

表 2-2 コンソールを使用して役割を追加するための項目の説明

項目	説明
役割名	管理者が特定の役割にログインするために使用する名前を選択する
役割の正式名称	この役割の名前をフルネームでわかりやすく入力する (省略可能)
説明	この役割の詳細な説明
役割 ID 番号	この役割に割り当てられている ID 番号を選択する。この番号は、UID の ID セットと同じ
役割のシェル	ユーザーが端末またはコンソールのウィンドウにログインするか、そのウィンドウで役割を引き受けるときに実行するシェルを選択する
役割のメーリングリストを作成	項目をチェックすると、役割と同じ名前でメーリングリストを作成する。メーリングリストを使用すると、その役割に割り当てられているすべてのユーザーに電子メールを送信できる
役割パスワードとパスワードを確認	役割のパスワードを設定および再入力する
有効な権利と許可された権利	「有効な権利 (Available Rights)」のリストから権利を選択し、「許可された権利 (Granted Rights)」のリストに追加することにより、この役割に権利を割り当てる
ホームディレクトリの選択	この役割の専有ファイルが格納されるホームディレクトリサーバーを選択する
ユーザーの役割への割り当て	特定のユーザーが特定の作業を行うための役割を持てるようにユーザーを役割に追加する

役割によるアクセス制御の詳細と、役割を使用して安全な環境を作成する方法については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

▼ 最初の役割 (プライマリ管理者) を作成する方法

この手順では、プライマリ管理者の役割を作成し、それをユーザーアカウントに割り当てる方法について説明します。ユーザーアカウントはすでに作成してあるものとします。

1. ユーザーとしてコンソールを起動します。

```
% /usr/sadm/bin/smc &
```

コンソールの起動方法については、42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」を参照してください。

ユーザーアカウントを作成する必要がある場合は、コンソールのオンラインヘルプを参照してください。

2. ナビゲーション区画で「このコンピュータ (Computer)」アイコンをクリックします。
3. 「System Configuration」→「ユーザー (Users)」→「管理役割 (Administrative Roles)」の順にクリックします。
4. 「アクション (Action)」→「管理役割を追加 (Add Administrative Role)」の順にクリックします。
「管理役割を追加 (Add Administrative Role)」ウィザードが開きます。
5. 次の手順に従って、「管理役割を追加 (Add Administrative Role)」ウィザードでプライマリ管理者の役割を作成します。
 - a. 役割名、役割の正式名、備考欄、役割 ID 番号、役割のシェル、役割のメーリングリストを作成するかどうかを指定する。「次へ (Next)」をクリックする。
 - b. 役割のパスワードを設定し、確認する。「次へ (Next)」をクリックする。
 - c. 「有効な権利 (Available Rights)」欄からプライマリ管理者の権利を選択後、「追加 (Add)」をクリックし、「許可された権利 (Granted Rights)」欄に追加する。「次へ (Next)」をクリックする。
 - d. 役割のホームディレクトリを選択する。「次へ (Next)」をクリックする。
 - e. この役割を引き受けることができるユーザーのリストに自分を割り当てる。「次へ (Next)」をクリックする。
必要に応じて、表 2-2 で役割の各項目の説明を参照してください。
6. 「完了 (Finish)」をクリックする。

▼ プライマリ管理者の役割を引き受ける方法

プライマリ管理者の役割を作成し終わったら、まずユーザー (自分自身) としてコンソールにログインし、次にプライマリ管理者の役割を引き受けます。

役割を引き受けるときは、その役割の権利を含むすべての属性を引き受けます。同時に、自分自身のユーザープロパティはすべて放棄します。

1. コンソールを起動します。

```
% /usr/sadm/bin/smc &
```

コンソールの起動方法については、42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」を参照してください。

2. ユーザー名とパスワードを使ってログインします。
引き受けることができる役割のリストが表示されます。
3. プライマリ管理者の役割にログインし、役割のパスワードを入力します。

Solaris 管理コンソールを起動する

次の手順では、コンソールを起動し、Solaris 管理ツールにアクセスする方法について説明します。

▼ スーパーユーザーまたは役割としてコンソールを起動する方法

自分自身のユーザーアカウントを使ってユーザーとしてコンソールを起動する場合は、Solaris 管理ツールへのアクセスが制限されます。十分なアクセス権を得るために、まずユーザー (自分自身) としてログインし、次に引き受けることができる役割のいずれかでログインできます。プライマリ管理者の役割を引き受けることができる場合は、スーパーユーザーの場合と同様に、すべての Solaris 管理ツールにアクセスできます。

1. CDE 環境などのウィンドウ環境で操作していることを確認します。
2. 次のいずれかの方法でコンソールを起動します。

- コマンド行から、次のように入力する

```
% /usr/sadm/bin/smc &
```

コンソールが初めて起動するときは 1 ~ 2 分かかります。

- CDE フロントパネルの「ツール (Tool)」メニューから起動する
 - CDE のアプリケーションマネージャまたはファイルマネージャの「Solaris 管理コンソール」アイコンをダブルクリックして起動する
- 「Solaris 管理コンソール」ウィンドウが表示されます。

注 - Solaris 管理コンソールのスタートアップメッセージを表示する場合は、自分のウィンドウ環境でコンソールを開いてください。Solaris 管理コンソールを起動する前に、Solaris 管理コンソールサーバーを手動で起動しないでください。このサーバーは、Solaris 管理コンソールを起動すると、自動的に起動します。コンソールに関する問題の障害追跡については、52 ページの「Solaris 管理コンソールの障害追跡」を参照してください。

3. ナビゲーション区画の「管理ツール (Management Tools)」アイコンの下にある「このコンピュータ (This Computer)」アイコンをダブルクリックします。カテゴリのリストが表示されます。
4. (省略可能) 適切なツールボックスを選択します。

デフォルト以外のツールボックスを使用する場合は、ナビゲーション区画から該当するツールボックスを選択します。あるいは、コンソールメニューから「ツールボックスを開く (Open Toolbox)」を選択し、任意のツールボックスをロードします。

各種ツールボックスの使用方法については、47 ページの「特定環境用のツールボックスを作成する方法」を参照してください。
5. カテゴリのアイコンをダブルクリックして、特定のツールにアクセスします。特定の作業の実行方法を確認する場合は、オンラインヘルプを使用します。
6. ツールのアイコンをダブルクリックします。

「ログイン (Log-In)」ポップアップウィンドウが表示されます。
7. スーパーユーザーまたは役割のどちらでこのツールを使用するかを決めます。
 - スーパーユーザーとしてログインし、スーパーユーザーとして作業する場合は、手順 8 に進む
 - ユーザー (自分自身) としてログインし、プライマリ管理者の役割を引き受ける場合は、手順 9 と 10 に進む
8. スーパーユーザーとしてログインしている場合は、**root** のパスワードを入力します。
9. ユーザー (自分自身) としてログインしている場合は、バックスペースキーを使って **root** のユーザー名を削除します。次に、適切なユーザー **ID** とユーザーパスワードを入力します。

ユーザーが引き受けることができる役割のリストが表示されます。
10. プライマリ管理者の役割、またはそれと同等の役割を選択し、役割のパスワードを入力します。

プライマリ管理者の役割の作成手順については、41 ページの「最初の役割 (プライマリ管理者) を作成する方法」を参照してください。

ツールのメインメニューが表示されます。

ネームサービス環境で Solaris 管理ツールを使用する (作業マップ)

デフォルトでは、Solaris 管理ツールはローカル環境で動作するように設定されます。たとえば、マウントと共有ツールを使用すると、特定のシステム上でディレクトリをマウントおよび共有できますが、NIS や NIS+ 環境ではできません。ただし、ネームサービス環境では、ユーザーツールやコンピュータとネットワークツールを使って情報を管理できます。

ネームサービス環境でコンソールのツールを使用する場合は、まずネームサービスのツールボックスを作成し、次にそのツールボックスにツールを追加する必要があります。

作業	説明	参照先
1. 前提条件を確認する	ネームサービス環境でコンソールを使用する前に、前提条件が満たされていることを確認する	46 ページの「ネームサービス環境で Solaris 管理コンソールを使用するための前提条件」
2. ネームサービス用のツールボックスを作成する	新規ツールボックスウィザードを使用して、ネームサービスツール用のツールボックスを作成する	47 ページの「特定環境用のツールボックスを作成する方法」
3. ネームサービスツールボックスにツールを追加する	作成したネームサービスツールボックスにユーザーツール、または他のネームサービスツールを追加する	49 ページの「ツールボックスにツールを追加する方法」
4. 直前に作成したツールボックスを選択する	直前に作成したツールボックスを選択して、ネームサービス情報を管理する	50 ページの「ネームサービス環境で Solaris 管理コンソールを起動する方法」

RBAC セキュリティファイル

Solaris 管理コンソールで使用する RBAC セキュリティファイルは、Solaris 9 へのアップグレードまたはそのインストール時に作成されます。Solaris 管理コンソールのパッケージをインストールしない場合、RBAC を使用するのに必要なデータがない状態で RBAC セキュリティファイルがインストールされます。Solaris 管理コンソールのパッケージについては、52 ページの「Solaris 管理コンソールの障害追跡」を参照してください。

Solaris 9 リリースの RBAC セキュリティファイルは、ネームサービス環境で Solaris 管理コンソールのツールを使用できるように、作成したネームサービスに組み込まれます。

ローカルサーバー上のセキュリティファイルは、標準アップグレードの一環として、`ypmake` コマンド、`nispopulate` コマンド、または同様の LDAP コマンドによってネームサービス環境に作成されます。サポートされているネームサービスは次のとおりです。

- NIS
- NIS+
- LDAP
- files

注 - NIS+ 環境では `projects` データベースはサポートされていません。

RBAC セキュリティファイルは、Solaris 9 リリースへのアップグレードまたはそのインストール時に作成されます。

次の表で、Solaris 9 でインストールされる定義済みのセキュリティファイルとその簡単な説明を示します。

表 2-3 RBAC セキュリティファイル

ローカルファイル名	テーブルまたはマップ名	説明
<code>/etc/user_attr</code>	<code>user_attr</code>	ユーザーと役割を承認と権利プロファイルに関連付ける
<code>/etc/security/auth_attr</code>	<code>auth_attr</code>	承認とその属性を定義し、関連付けられたヘルプファイルを識別する
<code>/etc/security/prof_attr</code>	<code>prof_attr</code>	権利プロファイルを定義し、権利プロファイルによって割り当てられた承認のリストを表示し、関連付けられたヘルプファイルを識別する
<code>/etc/security/exec_attr</code>	<code>exec_attr</code>	権利プロファイルに割り当てられている特権付きの操作を定義する

アップグレードに失敗した場合は、次のときに `smattrpop` コマンドで RBAC セキュリティファイルを作成してください。

- 権利プロファイルを作成または変更するとき
- `usr_attr` ファイルをカスタマイズして、ユーザーと役割を追加する必要があるとき

詳細については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

ネームサービス環境で Solaris 管理コンソールを使用するための前提条件

次の表に、ネームサービス環境で Solaris 管理コンソールを使用する前に行う必要のある操作を示します。

前提条件	参照先
Solaris 9 をインストールする	『Solaris 9 9/04 インストールガイド』
ネームサービス環境を設定する	『Solaris のシステム管理 (ネーミングとディレクトリサービス : DNS、NIS、LDAP 編)』
管理範囲を選択する	46 ページの「管理範囲」
ネームサービスデータにアクセスできるように /etc/nsswitch.conf ファイルが構成されていることを確認する	46 ページの「/etc/nsswitch.conf ファイル」

管理範囲

Solaris 管理コンソールでは、選択した管理ツールで使用するネームサービス環境を指すときに「管理範囲 (management scope)」という用語を使用します。ユーザーおよびコンピュータとネットワークツールの管理範囲は、LDAP、NIS、NIS+、files のの中から選択します。

コンソールセッション時に選択する管理範囲は、/etc/nsswitch.conf ファイルで識別される一次ネームサービスと一致している必要があります。

/etc/nsswitch.conf ファイル

各システムの /etc/nsswitch.conf ファイルは、そのシステムのネームサービスルックアップ (ここからデータが読み取られる) のポリシーを示します。

注 - コンソールからアクセスされるネームサービス (コンソールのツールボックスエディタで指定する) が /etc/nsswitch.conf ファイルの検索パスに含まれていることを確認する必要があります。指定のネームサービスがその検索パスにない場合、ツールが予測のつかない動作をしてエラーまたは警告が発生する可能性があります。

ネームサービス環境で Solaris 管理ツールを使用するときは、1 回の操作で多数のユーザーに影響を及ぼす可能性があります。たとえば、NIS ネームサービスのユーザーを削除すると、そのユーザーは NIS を使用しているすべてのシステムで削除されます。

ネットワーク内のさまざまなシステムで `/etc/nsswitch.conf` 構成が異なっていると、予期しない結果が生じる可能性があります。これを回避するために、Solaris 管理ツールで管理するすべてのシステムには、一貫したネームサービス構成を設定する必要があります。

▼ 特定環境用のツールボックスを作成する方法

Solaris オペレーティングシステムを管理するためのアプリケーションは「ツール」と呼ばれ、それらのツールは「ツールボックス」というコレクションに格納されます。ツールボックスは、ローカルサーバー (コンソールがある場所) にもリモートマシンにも配置できます。

ツールボックスエディタを使用すると、新しいツールボックスの追加、既存のツールボックスへのツールの追加、またはツールボックスの適用範囲の変更を行うことができます。たとえば、ドメインをローカルファイルからネームサービスに変更できます。

注 - ツールボックスエディタは、通常のコマンドラインとして起動できます。ただし、変更を加えたり、変更内容をデフォルトのコンソールツールボックス `/var/sadm/smc/toolboxes` に保存する場合は、`root` としてツールボックスエディタを起動する必要があります。

1. ツールボックスエディタを起動します。

```
# /usr/sadm/bin/smc edit &
```

2. 「ツールボックス (Toolbox)」メニューから「開く (Open)」を選択します。
3. 「ツールボックスを開く (Toolboxes)」ウィンドウで「このコンピュータ (This Computer)」アイコンを選択します。
4. 「開く (Open)」を選択します。
「このコンピュータ (This Computer)」ツールボックスがウィンドウに開きます。
5. ナビゲーション区画で「このコンピュータ (This Computer)」アイコンを再度選択します。
6. 「アクション (Action)」メニューから「フォルダを追加 (Add Folder)」を選択します。
7. 「フォルダ (Folder)」ウィザードを使用して、ネームサービス環境用の新しいツールボックスを追加します。

- a. 名前 (**Name**) と説明 (**Description**) – 「名前 (**Full Name**)」ウィンドウに名前を入力する。「次へ (**Next**)」をクリックする。
たとえば、NIS 環境の場合は「NIS tools」と入力する。
 - b. 「説明 (**Description**)」ウィンドウに説明を入力する。「次へ (**Next**)」をクリックする。
たとえば、「tools for NIS environment」と入力する。
 - c. アイコン (**Icons**) – デフォルト値を使用する。「次へ (**Next**)」をクリックする。
 - d. 管理範囲 (**Management Scope**) – 「上書き (**Override**)」を選択する。
 - e. 「管理範囲 (**Management Scope**)」プルダウンメニューから該当するネームサービスを選択する。
 - f. 必要に応じて、ネームサービスのマスター名を「サーバー (**Server**)」フィールドに追加する。
 - g. サーバーが管理しているドメインを「ドメイン (**Domain**)」フィールドに追加する。
 - h. 「完了 (**Finish**)」をクリックする。
左側のナビゲーション区画に新しいツールボックスが表示されます。
8. 新しいツールボックスのアイコンを選択します。
 9. 「ツールボックス (**Toolbox**)」メニューから「別名保存 (**Save As**)」を選択します。
 10. 「ローカルツールボックス (**Local Toolbox**)」ダイアログボックスの「フォルダ (**Folder**)」からフォルダを選択し、「ファイル名 (**Filename**)」にツールボックスのファイル名を入力します。接尾辞 **.tbx** を使用します。

```
/var/sadm/smc/toolboxes/this_computer/toolbox-name.tbx
```
 11. 「保存 (**Save**)」をクリックします。
コンソールウィンドウのナビゲーション区画に新しいツールボックスが表示されます。

次に進む手順

ネームサービスツールボックスを作成し終わったら、その中にネームサービスを入れることができます。詳細については、49 ページの「ツールボックスにツールを追加する方法」を参照してください。

▼ ツールボックスにツールを追加する方法

コンソールに添付されているデフォルトのツールのほかに、コンソールから起動できるツールがいくつか開発されています。これらのツールが使用可能になったら、既存のツールボックスに追加できます。

また、ローカル管理またはネットワーク管理用に新しくツールボックスを作成し、その新しいツールボックスにツールを追加することもできます。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 必要に応じて、ツールボックスエディタを起動します。

```
# /usr/sadm/bin/smc edit &
```

3. ツールボックスを選択します。
ネームサービス環境で作業する場合は、直前に作成したツールボックスをツールボックスエディタで選択します。
詳細については、47 ページの「特定環境用のツールボックスを作成する方法」を参照してください。
4. 「アクション (Action)」メニューから「ツールを追加 (Add Tool)」を選択します。
5. 「ツールを追加 (Add Tool)」ウィザードを使用して新しいツールを追加します。
 - a. 「サーバーの選択 (Server Selection)」 - ネームサービスのマスターを「サーバー (Server)」ウィンドウに追加する。「次へ (Next)」をクリックする。
 - b. 「ツールの選択 (Tools Selection)」 - 追加するツールを「ツール (Tools)」ウィンドウから選択する。「次へ (Next)」をクリックする。
このツールボックスがネームサービス用のツールボックスである場合は、ネームサービス環境で作業するツールを選択します。たとえば、ユーザーツールを選択します。
 - c. 「ツールクラス名 (Name) と説明 (Description)」 - デフォルト値を使用する。「次へ (Next)」をクリックする。
 - d. 「アイコン (Icons)」 - カスタムアイコンを作成していない場合は、デフォルト値を使用する。「次へ (Next)」をクリックする。
 - e. 「管理範囲 (Management Scope)」 - デフォルト値「親から継承 (Inherit from Parent)」を使用する。「次へ (Next)」をクリックする。
 - f. 「ツールの読み込み (Tool Loading)」 - デフォルト値「選択された時にツールを読み込む (Load tool when selected)」を使用する。「完了 (Finish)」をクリックする。
6. 「ツールボックス (Toolbox)」メニューから「保存 (Save)」を選択して、更新したツールボックスを保存します。

「ローカルツールボックス (Local Toolbox)」ウィンドウが表示されます。

▼ ネームサービス環境で Solaris 管理コンソールを起動する方法

ネームサービスのツールボックスを作成し、そのツールボックスにツールを追加し終わったら、Solaris 管理コンソールを起動し、作成したツールボックスを開いてネームサービス環境を管理できます。

1. 次の前提条件を満たしていることを確認します。
 - a. ログインしているシステムがネームサービス環境で動作するように設定されているかを確認する。
 - b. `/etc/nsswitch.conf` ファイルがユーザーのネームサービス環境に適合するように設定されているかを確認する。
2. Solaris 管理コンソールを起動します。

詳細については、42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」を参照してください。
3. ネームサービス用に作成したツールボックスを選択します。選択したツールボックスがナビゲーション区画に表示されます。

ネームサービス用のツールボックスの作成方法については、47 ページの「特定環境用のツールボックスを作成する方法」を参照してください。

Solaris 管理コンソールにツールを追加する

この節では、レガシーツールまたは別製品のツールをコンソールに追加する方法について説明します。これらのツールに認証を追加する場合は、『Solaris のシステム管理 (セキュリティサービス)』の「レガシーアプリケーションのセキュリティ保護」を参照してください。

▼ レガシーツールをツールボックスに追加する方法

レガシーツールとは、特に Solaris 管理ツールとして設計されていないアプリケーションのことです。コンソールのツールボックスには、X アプリケーション、コマンド行インタフェース、HTML の 3 種類のレガシーツールアプリケーションを追加できます。ツールボックスに追加した各ツールは、Solaris 管理コンソールから起動できます。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 必要に応じて、**Solaris** 管理コンソールのツールボックスエディタを起動します。

```
# /usr/sadm/bin/smc edit &
```
3. レガシーアプリケーションを追加したいツールボックスを開きます。
選択したツールボックスはツールボックスエディタで開きます。
4. レガシーアプリケーションを追加するツールボックス内のノードを選択します。
レガシーアプリケーションは、ツールボックスの最上位のノードにも別のフォルダにも追加できます。
5. 「アクション (Action)」 → 「従来のアプリケーションを追加 (Add Legacy Application)」の順にクリックします。
「従来のアプリケーションのウィザード (Legacy Application Wizard: General)」の最初のパネルが表示されます。
6. ウィザードの指示に従います。
7. エディタでツールボックスを保存します。

▼ 別製品のツールをインストールする方法

コンソールから起動できるツールパッケージを新たに追加する場合は、この手順に従います。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 新しいツールパッケージをインストールします。

```
# pkgadd ABCDtool
```
3. コンソールを再起動して、コンソールに新しいツールを認識させます。
 - a. コンソールサーバーを停止する。

```
# /etc/init.d/init.wbem stop
```
 - b. コンソールサーバーを起動する。

```
# /etc/init.d/init.wbem start
```
4. コンソールを起動して、新しいツールが表示されることを確認します。
詳細については、[42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」](#)を参照してください。

Solaris 管理コンソールの障害追跡

この障害追跡の手順を使用する前に、次のパッケージがインストールされていることを `pkginfo(1)` コマンドで確認してください。

SUNWmc	Solaris 管理コンソール 2.1 (サーバー構成要素)
SUNWmcc	Solaris 管理コンソール 2.1 (クライアント構成要素)
SUNWmccom	Solaris 管理コンソール 2.1 (共通の構成要素)
SUNWmcdev	Solaris 管理コンソール 2.1 (開発キット)
SUNWmcex	Solaris 管理コンソール 2.1 (例)
SUNWwbmc	Solaris 管理コンソール 2.1 (WBEM 構成要素)

これらのパッケージには、基本的な Solaris 管理コンソール起動ツールが用意されています。Solaris 管理コンソールとそのすべてのツールを使用するには、`SUNWCprog` クラスタをインストールする必要があります。

▼ Solaris 管理コンソールの障害追跡の方法

Solaris 管理コンソールを起動すると、クライアントとサーバーが自動的に起動します。

コンソールが表示可能で、ツールの実行に問題がある場合は、サーバーが実行していない可能性があります。あるいは、サーバーがエラー状態にあり、いったん停止してから再起動することによって解決できる可能性もあります。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 以下のコマンドでコンソールサーバーが稼動しているかどうかを調べます。

```
# /etc/init.d/init.wbem status
```

コンソールサーバーが稼動している場合は、次のようなメッセージが表示されません。

```
SMC server version 2.1.0 running on port 898.
```

3. コンソールサーバーが稼動していない場合は、以下のコマンドで起動します。

```
# /etc/init.d/init.wbem start
```

すぐに、次のようなメッセージが表示されます。

```
SMC server is ready.
```

4. サーバーが稼動しているのに問題が引き続き発生している場合は、いったんコンソールサーバーを停止してから、再起動します。

- a. コンソールサーバーを停止する。

```
# /etc/init.d/init.wbem stop
```

次のようなメッセージが表示されます。

```
Shutting down SMC server on port 898.
```

- b.** コンソールサーバーを起動する。

```
# /etc/init.d/init.wbem start
```


第 3 章

ユーザーアカウントとグループの管理 (概要)

この章では、ユーザーアカウントとグループを管理するためのガイドラインと管理計画の概要について説明します。また、ユーザーの作業環境のカスタマイズについても説明します。

この章の内容は次のとおりです。

- 55 ページの「ユーザーとグループの管理における新機能」
- 57 ページの「ユーザーアカウントとグループとは」
- 58 ページの「ユーザーアカウント管理のガイドライン」
- 64 ページの「グループを管理するガイドライン」
- 65 ページの「ユーザーアカウントとグループを管理するツール」
- 72 ページの「ユーザーアカウントとグループ情報の格納場所」
- 77 ページの「ユーザーの作業環境のカスタマイズ」

ユーザーアカウントおよびグループの管理手順については、[第 4 章](#)を参照してください。

ユーザーとグループの管理における新機能

この節では、ユーザーとグループを管理するための Solaris 9 リリースの新機能について説明します。

Solaris 管理コンソールツール群

Solaris 管理コンソールでは、Solaris 管理ツール群を利用でき、このツール群を使用してすべてのユーザーおよびグループ機能を管理できます。Solaris 管理コンソールの詳細については、第 2 章を参照してください。特定のユーザーおよびグループ管理作業の実行方法については、66 ページの「Solaris ユーザー管理ツールで実行できる作業」を参照してください。

Solaris のディレクトリサービス

Sun ONE Directory Server および他の LDAP ディレクトリサーバーを使用すると、LDAP (Lightweight Directory Access Protocol) ディレクトリサービスのユーザーおよびグループ情報を管理できます。ユーザーおよびグループ情報は、NIS、NIS+、またはファイル環境でも管理できます。

LDAP の設定の詳細については、『Solaris のシステム管理 (ネーミングとディレクトリサービス : DNS、NIS、LDAP 編)』を参照してください。

Sun ONE Directory Server を使ってユーザーおよびグループを管理する方法については、<http://docs.sun.com> の『iPlanet Directory Server 5.1 管理者ガイド』を参照してください。

プロジェクトでユーザーおよびリソースを管理する

Solaris 9 リリースでは、ユーザーおよびグループを「プロジェクト (project)」(システム使用率またはリソースアロケーションチャージバックの基礎として使用される、作業負荷の構成要素を示す識別子) のメンバーにすることができます。プロジェクトは、Solaris リソース管理機能の一部で、システムリソースの管理に使用されます。

Solaris 9 リリースを実行するシステムにログインするには、ユーザーはプロジェクトのメンバーになる必要があります。デフォルトでは、ユーザーは Solaris 9 リリースのインストール時に `group.staff` プロジェクトのメンバーになり、他のプロジェクト情報は設定されていません。

ユーザーのプロジェクト情報は、`/etc/project` ファイルに格納され、このファイルは、ローカルシステム (ファイル)、NIS ネームサービス、または LDAP ディレクトリサービスに保存できます。Solaris 管理コンソールを使用すると、プロジェクト情報を管理できます。

`/etc/project` ファイルは、ユーザーがログインするために必要ですが、プロジェクトを使用しない場合は管理する必要はありません。

プロジェクトの使用法および設定方法の詳細については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「プロジェクトとタスク」を参照してください。

ユーザーアカウントとグループとは

基本的なシステム管理作業の1つに、サイトの各ユーザーにユーザーアカウントを設定することがあります。通常、ユーザーアカウントには、ユーザーがスーパーユーザーのパスワードを知らなくても、システムにログインして、システムを使用するのに必要な情報が含まれています。ユーザーアカウント情報は、次の要素で構成されています。

構成要素	説明
ユーザー名	ユーザーがシステムにログインするのに使用する名前。ログイン名とも呼ばれます。
パスワード	ユーザーがシステムにアクセスするために、ユーザー名とともに入力しなければならない文字の組み合わせ。
ユーザーのホームディレクトリ	通常、ログイン時にユーザーのディレクトリになるディレクトリ。通常、ユーザーのホームディレクトリにはそのユーザーの大部分のファイルが含まれます。
ユーザー初期設定ファイル	ユーザーがシステムにログインするときに、そのユーザーの動作環境の設定を制御するシェルスクリプト。

また、ユーザーアカウントを設定するとき、ユーザーをあらかじめ定義されたユーザーグループに追加できます。グループは一般に、ファイルまたはディレクトリへのグループアクセス権を設定して、グループ内のユーザーだけがファイルとディレクトリにアクセスできるようにするために使用されます。

たとえば、ごく少数のユーザーだけにアクセスさせたい機密ファイルを入れるディレクトリを作成できます。topsecret プロジェクトに携わるユーザーを含む topsecret という名前のグループを設定します。そして、topsecret ファイルの読み取り権を topsecret グループに対して設定します。こうすれば、topsecret グループ内のユーザーだけが、ファイルを読み取ることができます。

また、「役割」という特別な種類のユーザーアカウントは、指定したユーザーに特別な特権を与えるときに使用します。詳細については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

ユーザーまたはグループは、1つまたは複数のプロジェクトのメンバーになることができます。プロジェクトは、システムリソースのチャージバックに使用される識別子です。プロジェクトの使用方法については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「プロジェクトとタスク」を参照してください。

ユーザーアカウント管理のガイドライン

次の節では、ユーザーアカウントを作成するガイドラインと計画方法について説明します。

ネームサービス

大規模なサイトでユーザーアカウントを管理する場合、LDAP、NIS、NIS+などのネームサービスまたはディレクトリサービスを使用できます。ネームサービスまたはディレクトリサービスを使うと、ユーザーアカウント情報を各システムの /etc 内のファイルに格納するのではなく、一元管理できます。ユーザーアカウントにネームサービスまたはディレクトリサービスを使用すれば、サイト全体のユーザーアカウント情報をシステムごとにコピーしなくても、同じユーザーアカウントのままシステム間を移動できます。ネームサービスまたはディレクトリサービスを使用すると、ユーザーアカウント情報を集中化および一元化して管理できます。

ユーザー (ログイン) 名

ユーザーは、ユーザー名 (ログイン名とも呼ばれる) を使って、自分のシステムと、適切なアクセス権を持つリモートシステムにアクセスできます。作成するユーザーアカウントそれぞれに、ユーザー名を選択しなければなりません。

ユーザー名または役割名を作成するときは、次のガイドラインに従ってください。

- 複数のドメインにまたがることもあるユーザーの組織内で、固有であること。
- 2～8文字の英数字を使用すること。最初の文字は英字でなければならず、少なくとも1文字は小文字にする必要があります。

ユーザー名には、ピリオド (.)、アンダースコア (_)、ハイフン (-) を使用できますが、これらの文字により障害が発生するソフトウェアもあるため、使用はお勧めできません。

ユーザー名を探しやすいように、ユーザー名の標準的な割り当て方法を使用することを検討してください。また、ユーザー名はユーザーが覚えやすいものにしてください。単純な規則の例としては、ユーザーのファーストネームの頭文字とラストネームの最初の7文字を使用します。たとえば、Ziggy Ignatz は zignatz になります。他のユーザー名と重複する場合は、ユーザーのファーストネームの頭文字、ミドルネームの頭文字、ラストネームの最初の6文字を使用します。たとえば、Ziggy Top Ignatz は ztignatz になります。さらに重複する場合、次の方法を検討してください。

- ファーストネームの頭文字、ミドルネームの頭文字、ラストネームの最初の5文字を使用する

- 固有の名前になるまで 1、2、3 などの数字を使用する

注 - それぞれの新しいユーザー名は、システムまたは NIS や NIS+ のドメインに登録されているメール別名(エイリアス)とは異なるものでなければなりません。そうしないと、メールは実際のユーザーではなく別名に送られることがあります。

ユーザー ID 番号

ユーザー名に関連するものとして、ユーザー識別 (UID) 番号があります。ユーザーがログインしようとするシステムは、UID 番号によってユーザー名を識別したり、ファイルとディレクトリの所有者を識別したりします。多数の異なるシステム上で、ある個人用に複数のユーザーアカウントを作成する場合は、常に同じユーザー名とユーザー ID を使用してください。そうすれば、そのユーザーは、所有権の問題を起こすことなく、システム間で簡単にファイルを移動できます。

UID 番号は、2147483647 以下の整数でなければなりません。UID 番号は、通常のユーザーアカウントと特殊なシステムアカウントに必要です。次の表にユーザーアカウントとシステムアカウントに予約されている UID 番号を示します。

表 3-1 予約済みの UID 番号

ユーザー ID 番号	ユーザー/ログインアカウント	説明
0 - 99	root、daemon、bin、sys など	システムアカウント
100 - 2147483647	通常のユーザー	汎用アカウント
60001 と 65534	nobody および nobody4	匿名ユーザー
60002	noaccess	アクセス権のないユーザー

0 から 99 までの UID 番号は予約されていますが、これらの番号でユーザーを追加することはできます。ただし、0 から 99 までの番号を通常のユーザーアカウントには使用しないでください。システム上の定義により、root には常に UID 0、daemon には UID 1、擬似ユーザー bin には UID 2 が設定されます。また、UID が passwd ファイルの先頭にくるように、uucp ログインや、who、tty、ttytype などの擬似的なユーザーログインには低い UID を与えるようにしてください。

ユーザー (ログイン) 名と同様に、固有の UID を割り当てる方法を決めてください。企業によっては、従業員に固有の番号を割り当て、管理者がその従業員番号にある番号を加えて固有の UID 番号を作成している場合もあります。

セキュリティ上のリスクを最小限に抑えるために、削除したアカウントの UID を再利用することは避けてください。どうしても UID を再利用する必要がある場合、はじめから作りなおして、新しいユーザーが前のユーザーの属性に影響されないようにしてください。たとえば、前のユーザーがプリンタの拒否リストに含まれていたためプリンタにアクセスできなかった場合、その属性を新しいユーザーに適用することが正しいとは限りません。

大きな数値のユーザー ID とグループ ID の使用

UID と GID には、符号付き整数の最大値 (つまり、2147483647) までの数値を割り当てることができます。

ただし、60000 より大きな数値の UID と GID は機能的に完全でなく、多くの Solaris の機能と互換性がありません。したがって、60000 を超える UID と GID を使用することは避けてください。

次の表では、Solaris の以前のリリースとの相互運用性に関する問題について説明します。

表 3-2 60000 より大きな数値の UID または GID の相互運用性に関する問題

カテゴリ	製品またはコマンド	問題または注意
NFS™ 互換性	SunOS™ 4.0 NFS ソフトウェアおよびその互換リリース	NFS サーバーとクライアントのコードは、大きな UID と GID を 16 ビットに切り捨てる。この状況では、SunOS 4.0 およびその互換リリースが動作しているシステムが、大きな UID と GID を使用している環境で使用されると、セキュリティの問題が発生する可能性がある。この問題を回避するには、SunOS 4.0 およびその互換リリースが動作しているシステムにパッチをあてる必要がある
ネームサービスの互換性	NIS ネームサービスおよびファイルベースのネームサービス	60000 より大きな数値の UID を持つユーザーは、Solaris 2.5 およびその互換リリースが動作しているシステムでは、ログインしたり、su コマンドを使用したりできるが、そのユーザーの UID と GID は 60001 (nobody) に設定される
	NIS+ ネームサービス	60000 より大きな数値の UID を持つユーザーは、Solaris 2.5 およびその互換リリースと NIS+ ネームサービスが動作しているシステムではアクセスが拒否される

表 3-3 大きな UID または GID の制限の要約

UID または GID の値	制限
60003 以上	<ul style="list-style-type: none"> ■ この分類のユーザーが、Solaris 2.5 およびそれ以前の互換リリースと NIS ネームサービスまたはファイルネームサービスが動作しているシステムにログインすると、nobody の UID および GID を取得する
65535 以上	<ul style="list-style-type: none"> ■ Solaris 2.5 およびその互換リリースが NFS バージョン 2 ソフトウェアと一緒に動作しているシステムでは、この分類の UID は 16 ビットに切り捨てられるため、セキュリティの問題が発生する可能性がある ■ この分類のユーザーがデフォルトのアーカイブフォーマットで cpio コマンドを使用してファイルをコピーすると、ファイルごとにエラーメッセージが表示される。そして、UID と GID はアーカイブにおいて nobody に設定される。 ■ SPARC システム: この分類のユーザーが SunOS 4.0 およびその互換バージョンで動作可能なアプリケーションを実行すると、一部のシステムコールから EOVERFLOW が戻されて、そのユーザーの UID と GID は nobody にマップされる ■ x86 システム: この分類のユーザーが SVR3 互換のアプリケーションを実行すると、一部のシステムコールから EOVERFLOW が返される場合がある ■ x86 システム: この分類のユーザーが、マウントされた System V ファイルシステムでファイルまたはディレクトリを作成しようとした場合、System V ファイルシステムは EOVERFLOW エラーを返す
100000 以上	<ul style="list-style-type: none"> ■ ps -l コマンドは最大 5 桁の UID を表示する。したがって、99999 より大きな UID または GID を含むときは、出力される列が揃わない
262144 以上	<ul style="list-style-type: none"> ■ この分類のユーザーが -H odc フォーマットで cpio コマンドまたは pax -x cpio コマンドを使用してファイルをコピーすると、ファイルごとにエラーメッセージが戻される。そして、UID と GID はアーカイブにおいて nobody に設定される。
1000000 以上	<ul style="list-style-type: none"> ■ この分類のユーザーが ar コマンドを使用すると、そのユーザーの UID と GID はアーカイブにおいて nobody に設定される
2097152 以上	<ul style="list-style-type: none"> ■ この分類のユーザーが tar コマンド、cpio -H ustar コマンド、または pax -x tar コマンドを使用すると、そのユーザーの UID と GID は nobody に設定される

パスワード

ユーザーを追加するときにそのユーザーのパスワードを指定できます。または、ユーザーが最初にログインするときにパスワードを指定するよう強制できます。ユーザーのパスワードは、次の構文に準拠している必要があります。

- パスワード長は、`/etc/default/passwd` ファイル内の `PASSLENGTH` 変数に指定された値に一致している必要がある。`PASSLENGTH` にはデフォルトで 6 が設定されている
- パスワードの最初の 6 文字には、2 文字以上の英字と 1 文字以上の数字または特殊文字を含める必要がある

ユーザー名は公表されますが、パスワードを知っているのは各ユーザーだけでなければなりません。各ユーザーアカウントには、6～8 文字の英数字と特殊文字を組み合わせたパスワードを割り当ててください。

コンピュータシステムのセキュリティを強化するには、ユーザーにパスワードを定期的に変更するよう指示してください。高いレベルのセキュリティを確保するには、ユーザーに 6 週間ごとにパスワードを変更するよう要求してください。低いレベルのセキュリティなら、3 ヶ月に 1 度で十分です。システム管理用のログイン (`root` や `sys` など) は、毎月変更するか、`root` のパスワードを知っている人が退職したり交替したりするたびに交換してください。

コンピュータセキュリティが破られる原因の多くは、正当なユーザーのパスワードが解読される場合です。ユーザーについて何か知っている人が簡単に推測できるような固有名詞、名前、ログイン名、パスワードを使わないよう各ユーザーに対して指示してください。

良いパスワードの例としては以下のようなものが考えられます。

- 英語の単語を組み合わせたフレーズ (たとえば、`beammeup`)
- フレーズ内の各単語の頭文字だけを集めた、意味のない文字列。たとえば、`SomeWhere Over The RainBow` から取った `swotrb`
- 文字を数字や記号に代えた単語。たとえば、`snoopy` から文字を代えた `sn00py`

次のものは、パスワードに不適當です。

- 自分の名前そのもの、逆読み、飛ばし読みのもの
- 家族やペットの名前
- 免許証番号
- 電話番号
- 社会保健番号
- 従業員番号
- 趣味や興味に関連した名前
- 季節に関係のある名前 (たとえば 12 月に `Santa` を使うなど)
- 辞書にある単語

パスワードの有効期限を設定する

NIS+ または /etc 内のファイルを使用してユーザーアカウント情報を格納する場合は、ユーザーのパスワードにパスワード有効期限を設定できます。Solaris 9 12/02 リリース以降は、LDAP ディレクトリサービスでもパスワードの有効期限がサポートされています。

パスワード有効期限の設定によって、ユーザーに定期的なパスワード変更を強制したり、あるパスワードを保持するのに必要な最低日数以前にパスワードを変更するのを防止したりできます。不正ユーザーが、古くて使用されていないアカウントを使用して、発覚せずにシステムのアクセス権を得るような場合を防止するために、アカウントが無効になる日付を設定することができます。パスワードの有効期限属性を設定するには、passwd コマンドまたは Solaris 管理コンソールのユーザーツールを使用します。

ホームディレクトリ

ホームディレクトリは、ユーザーが独自のファイルを格納するのに割り当てられるファイルシステムの一部です。ホームディレクトリに割り当てる大きさは、ユーザーが作成するファイルの種類、サイズ、および数によって異なります。

ホームディレクトリは、ユーザーのローカルシステムまたはリモートファイルサーバーのどちらにでも配置できます。どちらの場合も、慣例により、ホームディレクトリは /export/home/username として作成します。大規模なサイトでは、ホームディレクトリをサーバーに格納してください。ホームディレクトリのバックアップおよび復元を簡単にするには、/export/homen ディレクトリごとに別々のファイルシステムを使用してください。たとえば、/export/home1 と /export/home2 を使用します。

ホームディレクトリが配置される場所に関係なく、ユーザーは通常 /home/username という名前のマウントポイントを介してホームディレクトリにアクセスします。Autofs を使用してホームディレクトリがマウントされていると、どのシステムでも /home マウントポイントの下にディレクトリを作成することは許可されません。Autofs が使用されていると、システムはマウントされている /home を特別なものと認識します。ホームディレクトリを自動マウントする方法の詳細については、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「autofs 管理作業の概要」を参照してください。

ネットワーク上の任意の位置からホームディレクトリを使用するには、/export/home/username ではなく、常に \$HOME という環境変数の値によって参照するようにしてください。前者はマシンに固有の指定です。さらに、ユーザーのホームディレクトリで作成されるシンボリックリンクはすべて相対パス (たとえば ../.././x/y/x) を使用する必要があります。こうすることによって、そのリンクはどのシステムにホームディレクトリがマウントされても有効になります。

ユーザーの作業環境

ファイルを作成して格納するホームディレクトリのほかに、ユーザーには仕事をするために必要なツールとリソースにアクセスできる環境が必要です。ユーザーがシステムにログインするとき、C、Korn、Bourne シェルなどユーザーの起動シェルで定義される初期設定ファイルによって、ユーザーの作業環境が決定されます。

ユーザーの作業環境を管理するのに便利な方法として、カスタマイズしたユーザー初期設定ファイル (.login、.cshrc、.profile など) をユーザーのホームディレクトリに置くという方法があります。ユーザー初期設定ファイルをユーザー用にカスタマイズする方法については、77 ページの「ユーザーの作業環境のカスタマイズ」を参照してください。ユーザー初期設定ファイルをカスタマイズした後、新しいユーザーアカウントを作成するときにそれらをユーザーのホームディレクトリに追加できます。

1 回だけ行う作業としてお勧めするのは、「スケルトン」ディレクトリと呼ばれる別々のディレクトリをサーバーに設定することです。ユーザーのホームディレクトリが格納されるサーバーと同じサーバーを使用できます。スケルトンディレクトリによって、タイプの異なるユーザーに合わせてカスタマイズしたユーザー初期設定ファイルを格納できます。

注 - システム初期設定ファイル (/etc/profile、/etc/.login など) を使用してユーザーの作業環境を管理しないでください。これらのファイルはローカルシステムに存在するため、一元管理されません。たとえば、Autofs を使用してネットワーク上の任意のシステムからユーザーのホームディレクトリをマウントした場合、ユーザーがシステム間を移動しても環境が変わらないよう保証するには、各システムでシステム初期設定ファイルを修正しなければなりません。

また、役割によるアクセス制御でユーザーアカウントをカスタマイズする方法もあります。詳細については、『Solaris のシステム管理 (セキュリティサービス)』の「役割によるアクセス制御 (概要)」を参照してください。

グループを管理するガイドライン

「グループ」とは、ファイルやその他のシステム資源を共有できるユーザーの集合のことです。たとえば、同じプロジェクトで作業するユーザーはグループを形成することになります。グループは、従来の UNIX グループのことです。

各グループには、名前、グループ識別 (GID) 番号、およびそのグループに属しているユーザー名のリストが必要です。システムは GID によって内部的にグループを識別します。ユーザーは次の 2 つの種類のグループに所属できます。

- 一次グループ - オペレーティングシステムが、ユーザーによって作成されたファイルに割り当てるグループです。各ユーザーは、1つの一次グループに所属していません。
- 二次グループ - ユーザーが所属する1つまたは複数のグループです。ユーザーは、最大15個の二次グループに所属できます。

ユーザーの二次グループは、場合によっては重要でないことがあります。たとえば、ファイルの所有権は、一次グループだけが関係し、二次グループは関係しません。ただし、アプリケーションによってはユーザーの二次グループが関係することがあります。たとえば、ユーザーは、`Admintool` ソフトウェアを使用するとき `sysadmin` グループ (グループ 14) のメンバーでなければなりません、グループ 14 がそのユーザーの現在の一次グループであるかどうかは問題にはなりません。

`groups` コマンドを使って、ユーザーが所属しているグループを表示できます。ユーザーは一度に1つの一次グループにしか所属できません。ただし、`newgrp` コマンドを使用して、ユーザーがメンバーとなっている他のグループに一時的に一次グループを変更することはできます。

ユーザーアカウントを追加するとき、ユーザーに一次グループを割り当てるか、デフォルトの `staff` (グループ10) を使用しなければなりません。一次グループは、すでに存在しているものでなければなりません。存在しない場合は、GID 番号でグループを指定します。ユーザー名は、一次グループに追加されません。ユーザー名が追加されると、リストが長くなりすぎるからです。ユーザーを新しい二次グループに割り当てる前に、そのグループを作成し、それに GID 番号を割り当てなければなりません。

グループは、システムにとってローカルになるか、またはネームサービスを介して管理することができます。グループ管理を単純にするために、NIS のようなネームサービス、または LDAP のようなディレクトリサービスを使用してグループメンバーを集中管理してください。

ユーザーアカウントとグループを管理するツール

次の表に、ユーザーとグループを管理するための推奨ツールを示します。これらのツールはすべて、Solaris 管理コンソールツール群に含まれています。Solaris 管理コンソールの起動および使用方法については、第2章を参照してください。

表 3-4 ユーザーとグループを管理するためのツール

Solaris 管理ツール	用途	使用情報
ユーザー	ユーザーを管理する	Solaris 管理コンソールオンラインヘルプ
ユーザーテンプレート	学生、技術者、教師のように特定のユーザーの属性セットを作成する	Solaris 管理コンソールオンラインヘルプ
権限	RBAC 権限を管理する	Solaris 管理コンソールオンラインヘルプ
管理役割	RBAC 管理の役割を管理する	Solaris 管理コンソールオンラインヘルプ
グループ	グループ情報を管理する	Solaris 管理コンソールオンラインヘルプ
プロジェクト	プロジェクト情報を管理する	Solaris 管理コンソールオンラインヘルプ
メーリングリスト	メーリングリストを管理する	Solaris 管理コンソールオンラインヘルプ

Solaris 管理コンソールを使わずにユーザーアカウントおよびグループを管理する場合に使用する Solaris 管理コマンドについては、表 1-6 を参照してください。これらのコマンドは、認証およびネームサービスサポートを含め、Solaris 管理ツールと同じ機能を提供します。

Solaris ユーザー管理ツールで実行できる作業

Solaris ユーザー管理ツールを使用すると、ローカルシステムまたはネームサービス環境のユーザーアカウントを管理できます。

次の表で、ユーザーツールのユーザーアカウント機能を使って実行可能な作業について説明します。

表 3-5 ユーザーアカウント管理作業

作業	説明	内容説明
ユーザーの追加	ユーザーをローカルシステムまたはネームサービスに追加できる	57 ページの「ユーザーアカウントとグループとは」および 58 ページの「ユーザーアカウント管理のガイドライン」
ユーザーテンプレートの作成	ユーザー、契約者、技術者など、同じグループのユーザーを作成するために、定義済みのユーザー属性のテンプレートを作成できる	同上
ユーザーテンプレートを使用したユーザーの追加	テンプレートを使って、定義済みのユーザー属性を使用してユーザーを追加できる	同上
ユーザーテンプレートの複製	定義済みのユーザー属性を少しだけ変更して使用したい場合は、ユーザーテンプレートを複製する。次に、必要な属性のみを変更する	同上
ユーザープロパティの設定	ユーザーを追加する前にユーザープロパティを使用し、ユーザーの追加時にユーザーテンプレートを使用するかどうか、ユーザー削除時に、デフォルトでホームディレクトリやメールボックスを削除するかどうかなどを設定できる	同上
複数ユーザーの追加	ユーザー名を入力したテキストファイルを指定、または自動的に一連のユーザー名を生成することにより、ローカルシステムまたはネームサービスに複数のユーザーを追加できる	同上
ユーザープロパティの表示および変更	ログインシェル、パスワード、またはパスワードオプションのようなユーザープロパティを表示または変更できる	同上
ユーザーへの権限割り当て	特定の管理作業の実行を許可する権限をユーザーに割り当てることができる	同上

表 3-5 ユーザーアカウント管理作業 (続き)

作業	説明	内容説明
ユーザーの削除	ユーザーをローカルシステムまたはネームサービスから削除することができる。またオプションでユーザーのホームディレクトリまたはメールを削除するかどうかを指定できる。ユーザーは、グループまたは役割からも削除される	同上

表 3-6 ユーザー権限の管理作業

作業	説明	内容説明
権限を付与する	管理者だけが実行できた特定のコマンドまたはアプリケーションの実行権限をユーザーに付与することができる	『Solaris のシステム管理 (セキュリティサービス)』の「RBAC の権利プロファイル」
既存の権限のプロパティの表示および変更	既存の権限を表示または変更できる	同上
承認の追加	承認、つまり役割またはユーザーに個別に付与できる権限を追加できる	『Solaris のシステム管理 (セキュリティサービス)』の「RBAC の承認」
承認の表示および変更	既存の承認を表示または変更できる	同上

表 3-7 ユーザーの役割の管理作業

作業	説明	内容説明
管理の役割の追加	RBAC の承認役割を追加できる	『Solaris のシステム管理 (セキュリティサービス)』の「RBAC の役割」
管理の役割への権限の割り当て	作業を実行できるように、役割に特定の権限を割り当てることができる	同上
管理の役割の変更	役割に権限を追加したり削除したりすることができる	同上

表 3-8 グループ管理作業

作業	説明	内容説明
グループの追加	ユーザーを追加する前にグループ名を使用できるように、ローカルシステムまたはネームサービスにグループを追加する	64 ページの「グループを管理するガイドライン」
グループへのユーザーの追加	グループが所有するファイルにユーザーがアクセスする場合、ユーザーをグループに追加する	同上
グループからのユーザーの削除	ユーザーがグループファイルにアクセスする必要がなくなった場合は、グループからユーザーを削除できる	同上

表 3-9 プロジェクト管理作業

作業	説明	内容説明
プロジェクトの作成および複製	新しいプロジェクトを作成できるか、または新しいプロジェクトで必要な属性が酷似している既存のプロジェクトがある場合は、そのプロジェクトを複製できる	Solaris 管理コンソールのオンラインヘルプ
プロジェクト属性の変更および表示	既存のプロジェクト属性を表示または変更できる	Solaris 管理コンソールのオンラインヘルプ
プロジェクトの削除	不要になった場合は、プロジェクトを削除できる	Solaris 管理コンソールのオンラインヘルプ

表 3-10 メーリングリスト管理作業

作業	説明	内容説明
メーリングリストの作成	メーリングリスト (電子メールのメッセージの宛先のリスト) を作成できる	Solaris 管理コンソールのオンラインヘルプ
メーリングリスト名の変更	メーリングリストの作成後、その内容を変更できる	Solaris 管理コンソールのオンラインヘルプ
メーリングリストの削除	不要になった場合は、メーリングリストを削除できる	Solaris 管理コンソールのオンラインヘルプ

Solaris 管理コンソールによるホームディレクトリの管理

Solaris 管理コンソールツールを使用してユーザーのホームディレクトリを管理するときは、次のことに注意してください。

- ユーザーツールの「ユーザーを追加 (Add User)」ウィザードを使用してユーザーアカウントを追加し、ユーザーのホームディレクトリを `/export/home/username` として指定すると、ホームディレクトリが自動マウントされるように自動的に設定されて、次のエントリが `passwd` ファイルに追加されます。

```
/home/username
```

- ユーザーツールを使用して、ホームディレクトリを自動マウントしないユーザーアカウントを設定するには、この機能を無効にするユーザーアカウントのテンプレートを設定するほかに方法はありません。その後、このテンプレートを使ってユーザーを追加します。「ユーザーを追加 (Add User)」ウィザードでこの機能を無効にすることはできません。
- `-x autohome=N` オプションを指定して `smuser add` コマンドを使用すると、ユーザーのホームディレクトリを自動マウントしないでユーザーを追加できます。ただし、`smuser delete` コマンドには、ユーザーを追加した後でホームディレクトリを削除するオプションはありません。その場合は、ユーザーツールを使用して、ユーザーとユーザーのホームディレクトリを削除する必要があります。

ユーザーアカウントの変更

既存のものと重複するユーザー名や UID 番号を定義しないかぎり、ユーザーアカウントのログイン名や UID 番号を変更する必要はありません。2つのユーザーアカウントが、同じユーザー名または UID 番号を持つ場合、次の手順に従ってください。

- 2つのユーザーアカウントが同じ UID 番号を持つ場合、ユーザーツールを使用して、どちらか一方のアカウントを削除し、もう一度、異なる UID 番号で追加します。ユーザーツールを使用して、既存のユーザーアカウントの UID 番号を変更することはできません。
- 2つのユーザーアカウントが同じユーザー名を持つ場合、ユーザーツールを使用して、どちらか一方のアカウントを修正し、ユーザー名を変更します。

ユーザーツールを使用してユーザー名を変更する場合でも、ユーザーのホームディレクトリが存在すれば、ホームディレクトリの所有権は変更されます。

ユーザーアカウントの中で変更できる情報に、ユーザーのグループメンバーシップがあります。ユーザーツールの「アクション (Action)」メニューの「プロパティ (Properties)」を選択すると、ユーザーの二次グループを追加したり、削除したりできます。また、グループツールを使ってグループのメンバーリストを直接修正したりすることもできます。

ユーザーアカウントの次の部分も変更できます。

- 説明 (コメント)
- ログインシェル
- パスワードおよびパスワードオプション
- ホームディレクトリおよびホームディレクトリアクセス
- 権限および役割

ユーザーアカウントの削除

ユーザーツールでユーザーアカウントを削除すると、`passwd` ファイル、`group` ファイル内のエントリが自動的に削除されます。さらに、ユーザーのホームディレクトリおよびメールディレクトリにあるファイルを削除します。

カスタマイズしたユーザー初期設定ファイルの追加

ユーザーツールを使って、カスタマイズしたユーザー初期設定ファイルを作成することはできませんが、指定された「スケルトン」ディレクトリ内のユーザー初期設定ファイルでユーザーのホームディレクトリを生成することができます。このためには、ユーザーテンプレートツールを使ってユーザーテンプレートを作成し、コピーするユーザー初期設定ファイルを保存するスケルトンディレクトリを指定します。

`/etc/skel` ディレクトリにあるユーザー初期設定テンプレートをカスタマイズし、ユーザーのホームディレクトリへコピーできます。

パスワードの管理

ユーザーツールを使って、次のようなパスワード管理ができます。

- ユーザーアカウントに通常のパスワードを指定する
- ユーザーが最初のログイン時にパスワードを作成できるようにする
- ユーザーアカウントを無効にするかロックする
- 有効期限とパスワード有効期限情報を指定する

注 - パスワード有効期限は、NIS ネームサービスではサポートされません。

ユーザーアカウントを無効にする

一時的にまたは永続的に、ログインアカウントを無効にしなければならないことがあります。ユーザーアカウントを無効にしたりロックしたりすると、無効なパスワード *LK* がユーザーアカウントに割り当てられ、それ以後ログインできなくなります。

もっとも簡単にユーザーアカウントを無効にする方法は、ユーザーツールを使用してアカウントのパスワードをロックすることです。

「ユーザープロパティ (User Properties)」画面の「有効期限 (Expiration Date)」フィールドに有効期限を入力して、ユーザーアカウントの有効期間に制限を設けることもできます。

パスワード有効期限を設定するかパスワードを変更することによって、ユーザーアカウントを無効にすることもできます。

ユーザーアカウントとグループ情報の格納場所

ユーザーアカウントとグループ情報は、サイトの方針に応じて、ネームサービスまたはローカルシステムの /etc ファイルのどちらかに格納できます。情報は、NIS+ ネームサービスではテーブルに格納され、NIS ネームサービスではマップに格納され、LDAP ディレクトリサービスではインデックス付きのデータベースファイルに格納されます。

注 - 混乱を避けるために、ユーザーアカウントとグループ情報の位置は、「データベース」、「テーブル」、「マップ」という3種類の呼び方ではなく、単に「ファイル」と呼びます。

ほとんどのユーザーアカウント情報は、passwd ファイルに格納されます。ただし、パスワード暗号とパスワード有効期限は、NIS か NIS+ を使用するときには passwd ファイルに、/etc ファイルを使用するときは /etc/shadow ファイルに格納されます。NIS を使用するとき、パスワード有効期限は使用できません。

グループ情報は group ファイルに格納されます。

passwd ファイルのフィールド

passwd ファイルの各フィールドはコロンで区切られ、次のような情報が入っています。

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

たとえば、次のようになります。

```
kryten:x:101:100:Kryten Series 4000 Mechanoid:/export/home/kryten:/bin/csh
```


次の表では、passwd ファイルの各フィールドについて説明します。

表 3-11 passwd ファイルのフィールド

フィールド名	説明
<i>username</i>	ユーザー (またはログイン) 名。ユーザー名は固有で、1～8 文字の英字 (A-Z、a-z) と数字 (0-9) を使用する。最初の文字は英字で、少なくとも 1 文字は小文字を使用する
<i>password</i>	暗号化されたパスワードの代わりに x (パスワードフィールドはもう使用されない)。暗号化されたパスワードは shadow ファイルに格納される
<i>uid</i>	ユーザーをシステムに識別させるユーザー識別番号 (UID)。一般ユーザーの UID 番号は 100 から 60000 までの範囲とする。UID 番号はすべて固有でなければならない
<i>gid</i>	ユーザーの一次グループのグループ識別番号 (GID)。各 GID は 0 から 60002 までの範囲の整数でなければならない。60001 と 60002 は nobody と noaccess に割り当てられる。65534 は nobody4 に割り当てられる
<i>comment</i>	通常はユーザーのフルネーム。このフィールドはコメントとしての情報専用。このフィールドは、もともとは、Bell 研究所の UNIX システムから GECOS (General Electric Computer Operating System) を実行するメインフレームにバッチジョブを依頼する場合、必要なログイン情報を保持するために使われていたので、GECOS フィールドと呼ばれることもある
<i>home-directory</i>	ユーザーのホームディレクトリのパス名
<i>login-shell</i>	ユーザーのデフォルトログインシェル。これは /bin/sh、/bin/csh、/bin/ksh のどれかになる。表 3-18 のシェル機能の説明を参照

デフォルトの passwd ファイル

Solaris のデフォルトの passwd ファイルには、標準のデーモン用のエントリが入っています。デーモンとは、通常ブート時に起動され、システム全体で有効な操作 (印刷、ネットワークの管理、ポートの監視など) を実行するプロセスです。

```
root:x:0:1:Super-User:/:/sbin/sh
daemon:x:1:1:/:
bin:x:2:2:/:usr/bin:
sys:x:3:3:/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/usr/spool/lp:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
smmsp:x:25:25:SendMail Message Submission Program:/
listen:x:37:4:Network Admin:/usr/net/nls:
nobody:x:60001:60001:Nobody:/:
```

```
noaccess:x:60002:60002:No Access User:/:
nobody4:x:65534:65534:SunOS 4.x Nobody:/:
```

表 3-12 デフォルトの passwd ファイルのエントリ

ユーザー名	ユーザー ID	説明
root	0	スーパーユーザーのアカウント
daemon	1	ルーチンシステムタスクに関連するシステム包括デーモン
bin	2	ルーチンシステムタスクを実行するシステムバイナリの実行に関連する管理デーモン
sys	3	システムのログの記録や一時ディレクトリのファイルの更新に関連する管理デーモン
adm	4	システムのログの記録に関連する管理デーモン
lp	71	ラインプリンタのデーモン
uucp	5	uucp 関数に関連するデーモン
nuucp	6	uucp 関数に関連するデーモン
smmsp	25	Sendmail メッセージ送信プログラムデーモン
listen	37	ネットワーク監視デーモン
nobody	60001	特別なアクセス権を必要としない、あるいは持つべきではないユーザーまたはソフトウェアプロセスに割り当てられる
noaccess	60002	あるアプリケーションを経由するが実際にログインせずに、システムにアクセスする必要があるユーザーまたはプロセスに割り当てられる
nobody4	65534	SunOS 4.0 または 4.1 の nobody ユーザーアカウント

shadow ファイルのフィールド

shadow ファイルの各フィールドはコロンで区切られ、次のような情報が入っています。

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

たとえば、次のようになります。

```
rimmer:86Kg/MNT/dGu.:8882:0::5:20:8978
```

次の表では、shadow ファイルの各フィールドについて説明します。

表 3-13 shadow ファイルのフィールド

フィールド名	説明
<i>username</i>	ユーザー (またはログイン) 名
<i>password</i>	次のエントリのいずれかになる。13 文字の暗号化されたユーザーパスワード。アクセス不可能なアカウントを示す *LK*。アカウントのパスワードがないことを示す NP
<i>lastchg</i>	1970 年 1 月 1 日から最後にパスワードを変更した日付までの日数
<i>min</i>	パスワードの変更から次の変更までに必要な最短日数
<i>max</i>	ユーザーが新しいパスワードの指定をもとめられるまで、パスワードを変更しないで使い続けることができる最長日数
<i>inactive</i>	アカウントを使用 (ログイン) しなくてもよい最長日数
<i>expire</i>	ユーザーアカウントの有効期限が切れる日付。この日付が過ぎると、ユーザーはシステムにログインできない

group ファイルのフィールド

group ファイルの各フィールドはコロンで区切られ、次のような情報が入っています。

```
group-name:group-password:gid:user-list
```

たとえば、次のようになります。

```
bin::2:root,bin,daemon
```

次の表では、group ファイルの各フィールドについて説明します。

表 3-14 group ファイルのフィールド

フィールド名	説明
<i>group-name</i>	グループに付けられた名前。たとえば、大学の化学部のメンバーであれば chem などと指定する。グループ名に許される最大文字数は 8 文字
<i>group-password</i>	通常は空のままか、アスタリスクを指定する。group-password フィールドは初期バージョンの UNIX のなごり。グループにパスワードがある場合、newgrp コマンドはユーザーにグループパスワードを入力するよう求める。ただし、グループパスワードを設定するためのユーティリティはない

表 3-14 group ファイルのフィールド (続き)

フィールド名	説明
<i>gid</i>	グループの GID 番号。ローカルシステムで固有にする必要があり、組織全体を通じても固有であることが望ましい。GID 番号は 0 から 60002 までの範囲の整数で指定する。ただし、100 未満の番号はシステムのデフォルトグループアカウント用に予約されている。したがって、ユーザー定義グループの範囲は 100 から 60000。60001 と 60002 はそれぞれ <i>nobody</i> と <i>noaccess</i> に割り当てられている
<i>user-list</i>	コンマで区切られたユーザー名のリスト。ユーザーの二次グループメンバーシップを表す。各ユーザーは最大 15 個の二次グループに所属できる

デフォルトの group ファイル

Solaris のデフォルトの group ファイルには、システム全体に有効な操作 (印刷、ネットワーク管理、電子メールなど) をサポートする次のようなシステムグループが記述されています。これらのグループの多くは、passwd ファイルのエントリに対応しています。

```

root::0:root
other::1:
bin::2:root,bin,daemon
sys::3:root,bin,sys,adm
adm::4:root,adm,daemon
uucp::5:root,uucp
mail::6:root
tty::7:root,adm
lp::8:root,lp,adm
nuucp::9:root,nuucp
staff::10:
daemon::12:root,daemon
smmsp::25:smmsp
sysadmin::14:root
nobody::60001:
noaccess::60002:
nogroup::65534:

```

表 3-15 デフォルトの group ファイルのエントリ

グループ名	グループ ID	説明
root	0	スーパーユーザーのグループ
other	1	オプションのグループ
bin	2	システムバイナリの実行に関連する管理グループ

表 3-15 デフォルトの group ファイルのエントリ (続き)

グループ名	グループ ID	説明
sys	3	システムのログの記録や一時ディレクトリに関連する管理グループ
adm	4	システムのログの記録に関連する管理グループ
uucp	5	uucp 関数に関連するグループ
mail	6	電子メールのグループ
tty	7	tty デバイスに関連するグループ
lp	8	ラインプリンタのグループ
nuucp	9	uucp 関数に関連するグループ
staff	10	一般的な管理グループ
daemon	12	ルーチンシステムタスクに関連するグループ
sysadmin	14	Admintool と Solstice AdminSuite ツールに関連する管理グループ
smmsp	25	Sendmail メッセージ送信プログラムデーモン
nobody	60001	特別なアクセス権を必要としない、あるいは持つべきではないユーザーまたはソフトウェアプロセスに割り当てられるグループ
noaccess	60002	あるアプリケーションを経由するが実際にログインをせずに、システムにアクセスする必要があるユーザーまたはプロセスに割り当てられるグループ
nogroup	65534	既知のグループのメンバーでないユーザーに割り当てられるグループ

ユーザーの作業環境のカスタマイズ

ユーザーのホームディレクトリの設定には、ユーザーのログインシェルにユーザー初期設定ファイルを提供することも含まれます。ユーザー初期設定ファイルは、ユーザーがシステムにログインしたあとにユーザーのために作業環境を設定するシェルスクリプトです。基本的にシェルスクリプトで行える処理はどれもユーザー初期設定ファイルで実行できます。主に、ユーザーの検索パス、環境変数、ウィンドウ機能の環境など、ユーザーの作業環境を定義します。次の表に示すように、各ログインシェルには、1つまたは複数の、固有のユーザー初期設定ファイルがあります。

表 3-16 Bourne、C、Korn シェルのユーザー初期設定ファイル

シェル	初期設定ファイル	目的
Bourne	\$HOME/.profile	ログイン時のユーザー環境の定義
C	\$HOME/.cshrc	すべての C シェルに対するユーザー環境の定義で、ログインシェルのあとに起動される
	\$HOME/.login	ログイン時のユーザー環境の定義
Korn	\$HOME/.profile	ログイン時のユーザー環境の定義
	\$HOME/\$ENV	ログイン時のユーザー環境の定義で、Korn シェルの ENV 環境変数によって指定される

Solaris 環境には、次の表に示すように、各システムの /etc/skel ディレクトリに、各シェル用のデフォルトのユーザー初期設定ファイルが提供されています。

表 3-17 デフォルトのユーザー初期設定ファイル

シェル	デフォルトファイル
C	/etc/skel/local.login
	/etc/skel/local.cshrc
Bourne または Korn	/etc/skel/local.profile

これらのファイルを変更して、すべてのユーザーに共通な作業環境を提供する標準のファイルセットを作成できます。または、異なるタイプのユーザーに作業環境を提供することもできます。異なるタイプのユーザーにユーザー初期設定ファイルを作成する手順については、91 ページの「ユーザー初期設定ファイルをカスタマイズする方法」を参照してください。

ユーザーツールで新しいユーザーアカウントを作成して、ホームディレクトリを作成するオプションを選択すると、選択したログインシェルに合わせて次のファイルが作成されます。

シェル	作成されるファイル
C	/etc/skel/local.cshrc ファイルと /etc/skel/local.login ファイルがユーザーのホームディレクトリにコピーされ、それぞれ、.cshrc と .login という名前に変更される
Bourne と Korn	/etc/skel/local.profile ファイルがユーザーのホームディレクトリにコピーされ、.profile という名前に変更される

useradd コマンドで新しいユーザーアカウント追加するために、`-k` オプションと `-m` オプションで `/etc/skel` ディレクトリを指定した場合、3つの `/etc/skel/local*` ファイルと `/etc/skel/.profile` ファイルがすべてユーザーのホームディレクトリにコピーされます。この時点で、これらのファイルの名前をユーザーのログインシェルに合わせて変更する必要があります。

サイト初期設定ファイルの使用方法

ユーザー初期設定ファイルは管理者とユーザーの両方によってカスタマイズできます。この重要な機能は、サイト初期設定ファイルと呼ばれる、グローバルに配布されるユーザー初期設定ファイルによって実現します。サイト初期設定ファイルを使用して、ユーザーの作業環境に新しい機能を絶えず導入でき、しかもユーザーはユーザー初期設定ファイルをカスタマイズすることもできます。

ユーザー初期設定ファイルでサイト初期設定ファイルを参照するとき、サイト初期設定ファイルに対して行なったすべての更新は、ユーザーがシステムにログインするときかユーザーが新しいシェルを起動するとき自動的に反映されます。サイト初期設定ファイルは、ユーザーを追加したときにはなかったサイト全体の変更をユーザーの作業環境に配布するよう設計されています。

ユーザー初期設定ファイルでできるカスタマイズは、サイト初期設定ファイルでも行えます。これらのファイルは通常はサーバー、またはサーバーのグループにあり、ユーザー初期設定ファイルの最初の行に現れます。また、各サイト初期設定ファイルは、それを参照するユーザー初期設定ファイルと同じ型のシェルスクリプトでなければなりません。

C シェルのユーザー初期設定ファイルでサイト初期設定ファイルを参照するには、ユーザー初期設定ファイルの最初に次のような行を挿入してください。

```
source /net/machine-name/export/site-files/site-init-file
```

Bourne または Korn シェルのユーザー初期設定ファイルでサイト初期設定ファイルを参照するには、ユーザー初期設定ファイルの最初に次のような行を入れてください。

```
./net/machine-name/export/site-files/site-init-file
```

ローカルシステムへの参照を避ける

ユーザー初期設定ファイルに、ローカルシステムへの固有の参照は追加しないでください。初期設定ファイルの設定は、ユーザーがどのシステムにログインしても有効になる必要があります。たとえば、次のようになります。

- ユーザーのホームディレクトリをネットワーク上の任意の位置で利用できるようにするには、常に環境変数の値 `$HOME` を使用してホームディレクトリを参照してください。たとえば、`/export/home/username/bin` ではなく `$HOME/bin` を使用してください。`$HOME` は、ユーザーが別のシステムにログインする場合でも有効

で、その場合ホームディレクトリは自動マウントされます。

- ローカルディスクのファイルにアクセスするには、`/net/system-name/directory-name` などのグローバルパス名を使用してください。システムが `AutoFS` を実行していれば、`/net/system-name` で参照されるディレクトリはすべてユーザーがログインする任意のシステムに自動的にマウントできます。

シェル機能

次の表に各シェルの基本的な機能を示します。ユーザー初期設定ファイルを作成するのにどのシェルがどんな機能を提供するか参考にしてください。

表 3-18 Bourne、C、Korn シェルの基本機能

機能	Bourne	C	Korn
UNIX で標準シェルとして知られる	O	X	X
Bourne シェルと互換性がある構文	-	X	O
ジョブ制御	O	O	O
履歴リスト	X	O	O
コマンド行の編集	X	O	O
別名 (エイリアス)	X	O	O
ログインディレクトリの 1 文字省略形	X	O	O
ファイルの上書き保護 (noclobber)	X	O	O
CTRL + D 無視 (ignoreeof)	X	O	O
拡張 cd	X	O	O
.profile とは別の初期設定ファイル	X	O	O
ログアウトファイル	X	O	X

シェル環境

シェルは、`login` プログラム、システム初期設定ファイル、ユーザー初期設定ファイルによって定義される変数を含む環境を管理します。また、一部の変数はデフォルトで定義されます。シェルには次の 2 種類の変数があります。

- 環境変数 - シェルによって生成されるすべてのプロセスにエクスポートされる変数。環境変数の設定値は `env` コマンドで表示できます。PATH などを含む環境変数の一部が、シェルそのものの動作に影響を与えます。

- シェル (ローカル) 変数 – 現在使用中のシェルだけに関係する変数。C シェルの場合は、シェル変数は環境変数と特別に対応しています。これらのシェル変数は `user`、`term`、`home`、`path` です。シェル変数は、対応する環境変数の値によって初期設定されます。

C シェルでは、小文字を使って `set` コマンドでシェル変数を設定し、大文字を使って `setenv` コマンドで環境変数を設定します。シェル変数を設定すると、対応する環境変数が設定され、その逆もあります。たとえば、`path` シェル変数を新しいパスで更新すると、シェルは `PATH` 環境変数も新しいパスで更新します。

Bourne、Korn の両シェルでは、任意の大文字の変数名を使って、シェル変数と環境変数の両方を設定できます。その後に行うコマンドに対して変数を有効にするためには、`export` コマンドの実行も必要です。

すべてのシェルで、シェル変数と環境変数は一般的に大文字の名前で参照します。

ユーザー初期設定ファイルで、ユーザーのシェル環境を、あらかじめ定義された変数の値を変更するか、変数を追加することによってカスタマイズできます。次の表に、ユーザー初期設定ファイルで環境変数を設定する方法を示します。

表 3-19 ユーザー初期設定ファイルでの環境変数の設定方法

環境変数を設定したいシェルタイプ	ユーザー初期設定ファイルに追加する行
C シェル	<code>setenv VARIABLE value</code> 例: <code>setenv MAIL /var/mail/ripley</code>
Bourne または Korn シェル	<code>VARIABLE=value; export VARIABLE</code> 例: <code>MAIL=/var/mail/ripley;export MAIL</code>

次の表では、ユーザー初期設定ファイルでカスタマイズできる環境変数とシェル変数について説明します。各シェルで使用される変数の詳細については、`sh(1)`、`ksh(1)`、`cs(1)` の各マニュアルページを参照してください。

表 3-20 シェル変数と環境変数の説明

変数	説明
CDPATH (C シェルでは <code>cdpath</code>)	cd コマンドで使用する変数を設定する。cd コマンドの対象ディレクトリを相対パス名で指定すると、cd コマンドは対象ディレクトリをまず現在のディレクトリ (.) 内で検索する。対象ディレクトリが見つからなかった場合は、CDPATH 変数のリストの順で検索され、見つかると、ディレクトリの変更が行われる。CDPATH で対象ディレクトリが見つからなかった場合は、現在の作業ディレクトリは変更されない。たとえば、CDPATH 変数を <code>/home/jean</code> に設定し、その下に <code>bin</code> と <code>rje</code> の2つのディレクトリがある場合、 <code>/home/jean/bin</code> ディレクトリの中で <code>cd rje</code> と入力すると、絶対パスを指定しなくても、ディレクトリを <code>/home/jean/rje</code> に変更することになる
history	C シェルの履歴を設定する
HOME (C シェルでは <code>home</code>)	ユーザーのホームディレクトリへのパスを設定する
LANG	ロケールを設定する
LOGNAME	現在ログインしているユーザーの名前を設定する。LOGNAME のデフォルト値は、 <code>passwd</code> ファイルに指定されているユーザー名にログインプログラムによって自動的に設定される。この変数は参照用のみ使用し、設定を変更してはならない
LPDEST	ユーザーのデフォルトプリンタを設定する
MAIL	ユーザーのメールボックスへのパスを設定する
MANPATH	アクセスできるマニュアルページの階層を設定する
PATH (C シェルでは <code>path</code>)	ユーザーがコマンドを入力したときに実行するプログラムについて、シェルが検索するディレクトリを順番に指定する。ディレクトリが検索パス上にない場合は、ユーザーはコマンドの絶対パス名を入力しなければならない デフォルトの PATH は、ログインプロセスで <code>.profile</code> (Bourne または Korn シェル) または <code>.cshrc</code> (C シェル) の指定どおりに自動的に定義され、設定される 検索パスの順序が重要となる。同じコマンドが異なる場所にそれぞれ存在するときは、最初に見つかったコマンドが使用される。たとえば、PATH が Bourne および Korn シェル構文で <code>PATH=/bin:/usr/bin:/usr/sbin:\$HOME/bin</code> のように定義されていて、 <code>sample</code> というファイルが <code>/usr/bin</code> と <code>/home/jean/bin</code> の両方にあるものとする。ユーザーが <code>sample</code> コマンドを、その絶対パスを指定しないで入力した場合は、 <code>/usr/bin</code> で見つかったバージョンが使用される
prompt	C シェルのシェルプロンプトを設定する
PS1	Bourne または Korn シェルのシェルプロンプトを設定する

表 3-20 シェル変数と環境変数の説明 (続き)

変数	説明
SHELL (C シェルでは shell)	make、vi、その他のツールが使うデフォルトシェルを設定する
TERMINFO	terminfo ファイルに追加した、サポートされていない端末のパス名を指定する。/etc/profile または /etc/.login で TERMINFO 変数を使用する TERMINFO 環境変数を設定すると、システムはまずユーザーが定義した TERMINFO パスを調べる。ユーザーが定義した TERMINFO ディレクトリ内に端末の定義が見つからなかった場合は、システムはデフォルトディレクトリ /usr/share/lib/terminfo で定義を探す。どちらにも見つからなかった場合、端末は dumb として定義される
TERM (C シェルでは term)	端末を設定する。この変数は /etc/profile または /etc/.login で再設定する必要がある。ユーザーがエディタを起動すると、システムはこの環境変数の定義と同じ名前のファイルを探す。システムは、TERMINFO が参照するディレクトリ内を探して端末の特性を知る
TZ	時間帯を設定する。これは、たとえば ls -l コマンドで日付を表示する場合に使われる。TZ をユーザーの環境に設定しないと、システムの設定が使用される。設定する場合、グリニッジ標準時が使用される

PATH 変数

ユーザーが絶対パス名でコマンドを入力すると、シェルはそのパス名を使ってコマンドを探します。ユーザーがコマンド名しか指定しないと、シェルは PATH 変数で指定されているディレクトリの順でコマンドを探します。コマンドがいずれかのディレクトリで見つければ、シェルはコマンドを実行します。

デフォルトのパスがシステムで設定されますが、大部分のユーザーはそれを変更して他のコマンドディレクトリを追加します。環境の設定や、正しいバージョンのコマンドまたはツールへのアクセスに関連して発生するユーザーの問題の多くは、パス定義の誤りが原因です。

パスの設定のガイドライン

次に、効率的な PATH 変数を設定するためのガイドラインをいくつか示します。

- セキュリティが特に問題とならないときは、現在の作業ディレクトリ (.) をパスの最初に指定します。しかし、現在の作業ディレクトリをパスに入れると、セキュリティ上の問題となることがあり、特にスーパーユーザーにとって問題となります。
- 検索パスはできるだけ短くしておきます。シェルはパスで各ディレクトリを探します。コマンドが見つからないと、検索に時間がかかり、システムのパフォーマンスが低下します。

- 検索パスは左から右に読まれるため、通常使用するコマンドをパスの初めの方に指定するようにしてください。
- パスでディレクトリが重複していないか確認してください。
- 可能であれば、大きなディレクトリの検索は避けてください。大きなディレクトリはパスの終わりに指定します。
- NFSTM サーバーが応答しないときに「ハング」の可能性を少なくしたり、不要なネットワークトラフィックを削減するよう、NFS がマウントするディレクトリより前にローカルディレクトリを指定します。

例 — ユーザーのデフォルトパスの設定

次の例は、ユーザーのデフォルトパスがホームディレクトリと他の NFS マウントディレクトリを含むように設定する方法を示します。現在の作業ディレクトリはパスの最初に指定されます。C シェルユーザー初期設定ファイルでは、次の行を追加してください。

```
set path=(. /usr/bin $HOME/bin /net/glrr/files1/bin)
```

Bourne または Korn シェルユーザー初期設定ファイルでは、次の行を追加してください。

```
PATH=./usr/bin:/$HOME/bin:/net/glrr/files1/bin
export PATH
```

ロケール変数

LANG および LC 環境変数は、時間帯と照合順序、および日付、時間、通貨、番号の書式など、ロケール固有の変換と表記をシェルに指定します。さらに、ユーザー初期設定ファイルで stty コマンドを使って、端末のセッションが複数バイト文字をサポートするかどうかを設定できます。

LANG は、ロケールのすべての変換と表記を設定します。特に必要な場合、これとは別に LC_COLLATE、LC_CTYPE、LC_MESSAGES、LC_NUMERIC、LC_MONETARY、LC_TIME の各 LC 変数によってその他の設定を行えます。

次の表は、LANG と LC 環境変数の値の一部を示します。

表 3-21 LANG と LC 変数の値

値	ロケール
de_DE.ISO8859-1	German
en_US.UTF-8	American English (UTF-8)
es_ES.ISO8859-1	Spanish

表 3-21 LANG と LC 変数の値 (続き)

値	ロケール
fr_FR.ISO8859-1	French
it_IT.ISO8859-1	Italian
ja_JP.eucJP	Japanese (EUC)
ko_KR.EUC	Korean (EUC)
sv_SE.ISO8859-1	Swedish
zh_CN.EUC	Simplified Chinese (EUC)
zh_TW.EUC	Traditional Chinese (EUC)

サポートされるロケールの詳細については、『国際化対応言語環境の利用ガイド』を参照してください。

例 — LANG 変数によるロケールの設定

次の例は、LANG 環境変数を使ってロケールを設定する方法を示しています。C シェルユーザー初期設定ファイルでは、次の行を追加してください。

```
setenv LANG de_DE.ISO8859-1
```

Bourne または Korn シェルユーザー初期設定ファイルでは、次の行を追加してください。

```
LANG=de_DE.ISO8859-1; export LANG
```

デフォルトのファイルアクセス権 (umask)

ファイルまたはディレクトリを作成したときに設定されるデフォルトのファイルアクセス権は、ユーザーマスクによって制御されます。ユーザーマスクは、初期設定ファイルで umask コマンドによって設定されます。現在のユーザーマスクの値は、umask と入力して Return キーを押すと表示できます。

ユーザーマスクは、次の 8 進値で構成されます。

- 最初の桁でそのユーザーのアクセス権を設定する
- 2 桁目でグループのアクセス権を設定する
- 3 桁目で「その他」(ワールドとも呼ばれる)のアクセス権を設定する

最初の桁がゼロの場合、その桁は表示されません。たとえば、umask を 022 に設定すると、22 が表示されます。

設定する umask の値は、与えたいアクセス権の値を 666 (ファイルの場合) または 777 (ディレクトリの場合) から引きます。引いた残りが umask に使用する値です。たとえば、ファイルのデフォルトモードを 644 (rw-r--r--) に変更したいとします。このとき 666 と 644 の差 022 が umask コマンドの引数として使用する値です。

また、次の表から umask 値を決めることもできます。この表は、umask の各 8 進値から作成される、ファイルとディレクトリのアクセス権を示します。

表 3-22 umask 値のアクセス権

umask 8 進値	ファイルアクセス権	ディレクトリアクセス権
0	rw-	rwX
1	rw-	rw-
2	r--	r-X
3	r--	r--
4	-w-	-wX
5	-w-	-w-
6	--x	--x
7	--- (なし)	--- (なし)

次の例は、デフォルトのファイルアクセス権を rw-rw-rw- に設定します。

```
umask 000
```

ユーザー初期設定ファイルとサイト初期設定ファイルの例

ここでは、ユーザー自身の初期設定ファイルをカスタマイズする場合に使用する、ユーザー初期設定ファイルとサイト初期設定ファイルの例を示します。例の中のシステム名やパス名は、実際のサイトに合わせて置き換えてください。

例 — .profile ファイル

```

1 PATH=$PATH:$HOME/bin:/usr/local/bin:/usr/ccs/bin:.
2 MAIL=/var/mail/$LOGNAME
3 NNTPSERVER=server1
4 MANPATH=/usr/share/man:/usr/local/man
5 PRINTER=printer1
6 umask 022
7 export PATH MAIL NNTPSERVER MANPATH PRINTER

```

1. ユーザーのシェル検索パスを設定する

2. ユーザーのメールファイルへの検索パスを設定する
3. ユーザーの Usenet ニュースサーバーを設定する
4. マニュアルページへのユーザーの検索パスを設定する
5. ユーザーのデフォルトプリンタを設定する
6. ユーザーのデフォルトのファイル作成アクセス権を設定する
7. 指定された環境変数をエクスポートする

例 — .cshrc ファイル

```

1 set path=($PATH $HOME/bin /usr/local/bin /usr/ccs/bin)
2 setenv MAIL /var/mail/$LOGNAME
3 setenv NNTPSERVER server1
4 setenv PRINTER printer1
5 alias h history
6 umask 022
7 source /net/server2/site-init-files/site.login

```

1. ユーザーのシェル検索パスを設定する
2. ユーザーのメールファイルへの検索パスを設定する
3. ユーザーの Usenet ニュースサーバーを設定する
4. ユーザーのデフォルトプリンタを設定する
5. history コマンドの別名を作成する。これにより、h と入力するだけで history コマンドを実行できる
6. ユーザーのデフォルトのファイル作成アクセス権を設定する
7. 指定されたサイト初期設定ファイルを読み込む

例 — サイト初期設定ファイル

次のサイト初期設定ファイルの例では、ユーザーは特定のバージョンのアプリケーションを選択できます。

```

# @(#)site.login
main:
echo "Application Environment Selection"
echo ""
echo "1. Application, Version 1"
echo "2. Application, Version 2"
echo ""
echo -n "Type 1 or 2 and press Return to set your
application environment: "

set choice = $<

if ( $choice !~ [1-2] ) then
goto main
endif

```

```
switch ($choice)

case "1":
setenv APPHOME /opt/app-v.1
breaksw

case "2":
setenv APPHOME /opt/app-v.2
endsw
```

次のようにして、このサイト初期設定ファイルをユーザーの `.cshrc` ファイル (C シェルユーザーのみ使用可能) で参照させることができます。

```
source /net/server2/site-init-files/site.login
```

この行では、サイト初期設定ファイルは `site.login` という名前で、`server2` という名前のサーバー上にあります。また、自動マウントがユーザーのシステムで実行されていることを前提としています。

第 4 章

ユーザーアカウントとグループの管理 (手順)

この章では、Solaris 管理コンソールを使ってユーザーアカウントとグループを設定し、管理する方法について説明します。

Solaris 管理コンソールを使ったユーザーアカウントとグループの設定および管理に関連する手順については、89 ページの「ユーザーアカウントの設定 (作業マップ)」および 99 ページの「ユーザーアカウントの管理 (作業マップ)」を参照してください。

ユーザーアカウントとグループ管理の概要については、第 3 章を参照してください。

ユーザーアカウントの設定 (作業マップ)

作業	説明	参照先
(省略可能) ユーザー情報の収集	標準の書式を使ってユーザー情報を収集すると、情報を整理しやすくなる	90 ページの「ユーザー情報を収集する方法」
(省略可能) ユーザー初期設定ファイルのカスタマイズ	新規ユーザーに一貫した環境を提供できるようにユーザー初期設定ファイル (.cshrc、.profile、.login) を設定する	91 ページの「ユーザー初期設定ファイルのカスタマイズする方法」
(省略可能) グループの追加	次のツールを使ってグループを追加できる	

作業	説明	参照先
	Solaris 管理コンソールのグループツール	93 ページの「Solaris 管理コンソールのグループツールを使ってグループを追加する方法」
	Solaris コマンド行インタフェースツール	95 ページの「CLI ツールを使ってグループやユーザーを追加する方法」
ユーザーの追加	次のツールを使ってユーザーを追加できる	
	Solaris 管理コンソールのユーザーツール	94 ページの「Solaris 管理コンソールのユーザーツールを使ってユーザーを追加する方法」
	Solaris コマンド行インタフェースツール	95 ページの「CLI ツールを使ってグループやユーザーを追加する方法」
(省略可能) ユーザーテンプレートの設定	類似したユーザープロパティをすべて手動で追加する必要がないようにユーザーテンプレートを作成できる	Solaris 管理コンソールオンラインヘルプ
(省略可能) ユーザーへの権限または役割の追加	特定のコマンドまたは操作を実行できるようにユーザーに権限または役割を追加する	Solaris 管理コンソールオンラインヘルプ
ユーザーのホームディレクトリの共有	ユーザーのホームディレクトリを共有することによって、ユーザーのシステムからそのディレクトリをリモートでマウントできる	96 ページの「ユーザーのホームディレクトリを共有する方法」
ユーザーのホームディレクトリのマウント	ユーザーのホームディレクトリをユーザーのシステムにマウントする	97 ページの「ユーザーのホームディレクトリをマウントする方法」

ユーザー情報を収集する方法

ユーザーアカウントを追加する前に、ユーザーに関する情報を次のような書式で記録しておくことができます。

項目	説明
ユーザー名	

役割名	
プロファイルまたは承認	
ユーザー名	
UID	
一次グループ	
二次グループ	
コメント	
デフォルトシェル	
パスワードの状態と有効期限	
ホームディレクトリのサーバー名	
ホームディレクトリのパス名	
マウント方法	
ホームディレクトリのアクセス権	
メールサーバー	
所属部署	
部署管理者	
管理者	
従業員名	
役職	
資格	
従業員番号	
開始日	
メール別名への追加	
デスクトップシステム名	

▼ ユーザー初期設定ファイルをカスタマイズする方法

1. ユーザーのホームディレクトリを作成し、共有するシステムでスーパーユーザーになるか、同等の役割を引き受けます。
2. 各タイプのユーザー用にスケルトンディレクトリを作成します。

▼ Solaris 管理コンソールのグループツールを使ってグループを追加する方法

システムにグループを追加するには、次の手順を実行します。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Solaris** 管理コンソールを起動します。

```
# /usr/sadm/bin/smc &
```

Solaris 管理コンソールの起動方法については、42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」または 50 ページの「ネームサービス環境で Solaris 管理コンソールを起動する方法」を参照してください。

3. ナビゲーション区画の「管理ツール (**Management Tools**)」アイコンの下にある「このコンピュータ (**This Computer**)」アイコンをダブルクリックします。カテゴリのリストが表示されます。
4. (省略可能) ネームサービス環境に適したツールボックスを選択します。
5. 「**System Configuration**」アイコンをダブルクリックします。
6. 「ユーザー (**Users**)」アイコンをクリックします。
7. スーパーユーザーのパスワードまたは役割のパスワードを入力します。
8. 「グループ (**Groups**)」アイコンをクリックします。コンテキストヘルプを使用して、グループをシステムに追加します。

例 — Solaris 管理コンソールのグループツールを使ってグループを追加する

次の例では、グループ `mechanoids` (グループ ID 101) をシステム `starbug` に追加します。この例では、起動ツールが起動済みで、ユーザーツールが開いていることを前提とします。

グループを追加するときに既存のユーザーをグループに追加できます。また、グループだけを追加し、その後でユーザーを追加するときに既存のユーザーをそのグループに追加することもできます。

- 「アクション (Action)」メニューから「グループを追加 (Add Group)」を選択します。
- 「グループ ID (Group Identification)」の下にある「グループ (Group Name)」プロンプトで、グループ名 `mechanoids` を指定します。
- 「グループ ID 番号 (Group ID number)」プロンプトで、グループ番号 101 を指定します。
- 「了解 (OK)」をクリックします。

▼ Solaris 管理コンソールのユーザーツールを使ってユーザーを追加する方法

ユーザーをシステムに追加するには、次の手順を実行します。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Solaris** 管理コンソールを起動します。

```
# /usr/sadm/bin/smc &
```

Solaris 管理コンソールの起動方法については、42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」または 50 ページの「ネームサービス環境で Solaris 管理コンソールを起動する方法」を参照してください。

3. ナビゲーション区画の「管理ツール (**Management Tools**)」アイコンの下にある「このコンピュータ (**This Computer**)」アイコンをダブルクリックします。カテゴリのリストが表示されます。
4. (省略可能) ネームサービス環境に適したツールボックスを選択します。
5. 「**System Configuration**」アイコンをダブルクリックします。
6. 「ユーザー (**Users**)」アイコンをクリックします。
7. スーパーユーザーのパスワードまたは役割のパスワードを入力します。
8. 「ユーザーアカウント (**User Accounts**)」アイコンをクリックします。コンテキストヘルプを使用して、ユーザーをシステムに追加します。

例 — Solaris 管理コンソールのグループツールを使ってユーザーを追加する

次の例では、ユーザー kryten (ユーザー ID 1001) をシステム starbug に追加します。この例では、起動ツールが起動済みで、ユーザーツールが開いていることを前提とします。

次の手順に進むときは「次へ (Next)」をクリックしてください。

- 「アクション (Action)」メニューから「ユーザーを追加 (Add User)」→「ウィザードを使用 (With Wizard)」を選択します。
- 手順 1 - 「ユーザー名 (User Name)」プロンプトで、ユーザー名またはログイン名 kryten を指定します。
- (省略可能) 「正式名 (Full Name)」プロンプトで、ユーザーのフルネーム kryten series 3000 を指定します。
- (省略可能) 「備考欄 (Description)」プロンプトで、このユーザーの詳細な説明を入力します。

- 手順 2 – 「ユーザー ID 番号 (User ID Number)」プロンプトで、ユーザー ID 1001 を入力します。
- 手順 3 – 「ユーザーは最初のログイン時にこのパスワードを使用する必要があります (User Must Use This Password At First Login)」オプションを選択します。
「パスワード (Password)」プロンプトでユーザーのパスワードを入力し、「パスワードを確認 (Confirm Password)」プロンプトでそのパスワードを確認します。
- 手順 4 – ユーザーの一次グループ `mechanoids` を選択します。
- 手順 5 – 「サーバー (Server)」と「パス (Path)」の各プロンプトでデフォルトを受け入れることにより、ユーザーのホームディレクトリを作成します。
- 手順 6 – メールサーバーを指定します。
- 手順 7 – 入力した情報を確認し、必要があれば前の画面に戻って情報を訂正します。訂正する必要がない場合は、「完了 (Finish)」をクリックします。

CLI ツールを使ってグループやユーザーを追加する方法

この節では、CLI ツールを使ってユーザーやグループを追加する例を示します。

例 — `groupadd` および `useradd` コマンドを使ってグループやユーザーを追加する

次の例では、`groupadd` および `useradd` の各コマンドを使って、グループ `scutters` やユーザー `scutter1` をローカルシステムのファイルに追加します。なお、これらのコマンドは、ネームサービス環境でユーザーを管理する場合には使用できません。

```
# groupadd -g 102 scutters
# useradd -u 1003 -g 102 -d /export/home/scutter1 -s /bin/csh -c "Scutter 1"
-m -k /etc/skel scutter1
64 blocks
```

詳細については、`groupadd(1M)` および `useradd(1M)` のマニュアルページを参照してください。

例 — `smgroup` および `smuser` コマンドを使ってグループやユーザーを追加する

次の例では、`smgroup` および `smuser` の各コマンドを使って、グループ `gelfs` やユーザー `camille` をホスト `starbug` の NIS ドメイン `solar.com` に追加します。

```
# /usr/sadm/bin/smgroup add -D nis:/starbug/solar.com -- -g 103 -n gelfs
# /usr/sadm/bin/smuser add -D nis:/starbug/solar.com -- -u 1004 -n camille
-c "Camille G." -d /export/home/camille -s /bin/csh -g gelfs
```

詳細については、smgroup(1M) および smuser(1M) のマニュアルページを参照してください。

▼ ユーザーのホームディレクトリを共有する方法

1. ユーザーのホームディレクトリを含むシステムでスーパーユーザーになるか、同等の役割を引き受けます。

2. 次のように入力して、**mountd** デーモンが動作していることを確認します。

```
# ps -ef | grep mountd
```

```
root 176 1 0 May 02 ? 0:19 /usr/lib/nfs/mountd
```

mountd デーモンが動作している場合には、`/usr/lib/nfs/mountd` と表示されます。

3. **mountd** デーモンが動作していない場合は、**mountd** デーモンを起動します。

```
# /etc/init.d/nfs.server start
```

4. システム上で共有されているファイルシステムをリスト表示します。

```
# share
```

5. ユーザーのホームディレクトリを含むファイルシステムがすでに共有されているかどうかによって、次のいずれかの手順を選択します。

- a. ユーザーのホームディレクトリがすでに共有されている場合、次の確認手順へ進みます。

- b. ユーザーのホームディレクトリが共有されていない場合、**手順 6** に進みます。

6. **/etc/dfs/dfstab** ファイルを編集して、次の行を追加します。

```
share -F nfs /file-system
```

/file-system は、共有するユーザーのホームディレクトリを含むファイルシステムです。規約上、このファイルシステムは `/export/home` になります。

7. **/etc/dfs/dfstab** ファイルで指定されたファイルシステムを共有します。

```
# shareall -F nfs
```

このコマンドは、`/etc/dfs/dfstab` ファイルにある `share` コマンドをすべて実行するので、システムをリブートする必要はありません。

8. ユーザーのホームディレクトリが共有されていることを確認します。

```
# share
```


次に進む手順

ユーザーのホームディレクトリがユーザーのシステム上にない場合、それが配置されているシステムから、ユーザーのホームディレクトリをマウントしなければなりません。詳細な手順については、97 ページの「ユーザーのホームディレクトリをマウントする方法」を参照してください。

例 — ユーザーのホームディレクトリを共有する

```
# ps -ef | grep mountd
# /etc/init.d/nfs.server start
# share
# vi /etc/dfs/dfstab

(share -F nfs /export/home 行を追加)
# shareall -F nfs
# share
-                /usr/dist                ro    ""
-                /export/home/user-name    rw    ""
```

▼ ユーザーのホームディレクトリをマウントする方法

ホームディレクトリを自動マウントする場合は、『Solaris のシステム管理 (資源管理とネットワークサービス)』の「autofs 管理作業の概要」を参照してください。

1. ユーザーのホームディレクトリが共有されていることを確認します。
詳細については、96 ページの「ユーザーのホームディレクトリを共有する方法」を参照してください。
2. ユーザーのシステムにスーパーユーザーとしてログインします。
3. **/etc/vfstab** ファイルを編集して、次のようなユーザーのホームディレクトリ用のエントリを作成します。

```
system-name:/export/home/user-name - /export/home/user-name nfs - yes rw
```

<i>system-name</i>	ホームディレクトリが配置されているシステムの名前
<i>/export/home/user-name</i>	共有されるユーザーのホームディレクトリの名前。慣例上、 <i>/export/home/user-name</i> にユーザーのホームディレクトリが含まれる。ただし、別のファイルシステムでも構わない
-	エントリに必要な可変部分

`/export/home/user-name` ユーザーのホームディレクトリがマウントされるディレクトリの名前

エントリを `/etc/vfstab` ファイルに追加する方法については、『Solaris のシステム管理 (デバイスとファイルシステム)』の「ファイルシステムのマウントとマウント解除 (手順)」を参照してください。

4. ユーザーのホームディレクトリのマウント先を作成します。

```
# mkdir -p /export/home/user-name
```

5. ユーザーのホームディレクトリをマウントします。

```
# mountall
```

現在の `vfstab` ファイルにあるすべてのエントリ (このファイルの「mount at boot」フィールドが `yes` に設定されている) がマウントされます。

6. ホームディレクトリがマウントされているかどうかを確認します。

```
# mount | grep user-name
```

例 — ユーザーのホームディレクトリをマウントする

```
# vi /etc/vfstab
```

```
( venus:/export/home/ripley - /export/home/ripley  
nfs - yes rw 行を追加)
```

```
# mkdir -p /export/home/ripley
```

```
# mountall
```

```
# mount
```

```
/ on /dev/dsk/c0t0d0s0 read/write/setuid/intr/largefiles/xattr/onerror= ...
```

```
/usr on /dev/dsk/c0t0d0s6 read/write/setuid/intr/largefiles/xattr/onerror= ...
```

```
/proc on /proc read/write/setuid/dev=3dc0000 on Fri Apr 2 13:26:30 2004
```

```
/etc/mnttab on mnttab read/write/setuid/dev=3e80000 on Fri Apr 2 13:26:30 2004
```

```
/dev/fd on fd read/write/setuid/dev=3ec0000 on Fri Apr 2 13:26:33 2004
```

```
/var/run on swap read/write/setuid/xattr/dev=1 on Fri Apr 2 13:26:34 2004
```

```
/tmp on swap read/write/setuid/xattr/dev=2 on Fri Apr 2 13:26:38 2004
```

```
/export/home on /dev/dsk/c0t0d0s7 read/write/setuid/intr/largefiles/xattr...
```

```
/export/home/ripley on venus:/export/home/ripley remote/read/write/setuid/xattr/dev=...
```


きに、自動的に起動されます。このユーザー登録は、今すぐに登録することも、後で登録することも、あるいは登録しないことも可能です。ユーザー登録によって、Solaris のバージョン、ユーザー情報、プラットフォーム、ハードウェア、およびロケールの情報が Sun に提供されます。

Solaris Solve へのアクセス

Solaris ユーザー登録が完了すると、Solaris SolveSM にアクセスできるようになります。Solaris Solve は、Solaris の貴重な製品情報やソリューションが 1 箇所にまとめて提供されている会員制の Web サイトです。Solaris Solve を使用すると、Solaris の最新リリースに関する最新情報をすばやく簡単に入手できます。また、Sun のその他の契約やサービスについても知るすることができます。

Solaris ユーザー登録を完了して、Solaris Solve にアクセスする基本的な手順は次のとおりです。

1. Solaris ユーザー登録電子プロフィールに入力します。
2. 上記プロフィールを電子メールで送信するか、印刷したものを FAX か郵便で送ります。
3. Solaris Solve サイトにアクセスするためのログイン ID とパスワードを作成します。

Solaris Solve サイトにすぐにアクセスしない場合でも、Solaris ユーザーの登録時に自分の Solaris Solve ログイン ID とパスワードを作成しておくことをお勧めします。Solaris Solve ログイン ID とパスワードは、6～8 文字の英数字で作成します。空白とコロンは使用できません。

4. Solaris Solve サイトにアクセスします。

注 - システム管理者またはスーパーユーザーとしてログインすると、Solaris ユーザー登録は起動されません。

登録するように選択した場合、完成したフォームのコピーが \$HOME/.solregis/uprops に格納されます。登録しないように選択した場合でも、次のいずれかの方法でユーザー登録を起動して、後から登録することができます。

- コマンド行プロンプトで /usr/dt/bin/solregis と入力する
- アプリケーションマネージャのデスクトップツールフォルダにある、「Solaris ユーザー登録」アイコンをクリックする (CDE デスクトップの場合)

詳細については、solregis(1) のマニュアルページを参照してください。

Solaris ユーザー登録の問題の障害追跡

この節では、Solaris ユーザー登録に関する問題の障害追跡の方法について説明します。

次の表に、登録時に発生する可能性のある問題とその対処方法について説明します。

表 4-1 登録時の問題と対処方法

問題	対処方法
登録フォームが初期化されない。Web ページウィンドウに、システム管理者に登録設定を妨げている問題を解決してもらうように促すメッセージが表示される	登録ファイルが失われていないかチェックする
フォームを電子メールで送信できない。ダイアログボックスに、システム管理者に問題を解決してもらうように促すメッセージが表示される	電子メールが正しく設定されているかどうかをチェックする。完成した登録フォームを電子メールで送信するには CDE が必要なため、CDE がユーザーのシステム上にあるかを確認する。別の方法としてフォームを印刷して FAX か郵便で送信することもできる
フォームを印刷できない。ダイアログボックスに、システム管理者に問題を解決してもらうように促すメッセージが表示される	プリンタが正しく構成されているかどうかをチェックする。別の方法として電子メールでフォームを送信することもできる
フォームを保存できない。ダイアログボックスに、登録は成功したが、将来登録を更新するときに登録情報を呼び出せないことが示される。	ユーザーのホームディレクトリをチェックする。必要な処置は、システムの構成によって異なる
Solaris Solve のログイン ID とパスワードを忘れてしまった	SolarisSolve@sun.com に問題を説明するメールを英語で送るか、101 ページの「Solaris ユーザー登録をやり直す方法」を参照
登録プロセスをやり直したい	101 ページの「Solaris ユーザー登録をやり直す方法」

▼ Solaris ユーザー登録をやり直す方法

Solaris ユーザー登録を最初からやり直すには、次の手順に従ってください。

1. `$HOME/.solregis` ディレクトリに移動します。

```
% cd $HOME/.solregis
```

2. `uprops` ファイルを削除します。

```
% rm uprops
```

3. 登録プロセスを再起動します。

```
% /usr/dt/bin/solregis &
```

▼ ユーザー登録を無効にする方法

Solaris のインストール前またはインストール後に、ユーザー登録を無効にすることができます。Solaris ユーザー登録を無効にする前に、組織のシステム管理者をユーザー登録しておくことをお勧めします。

1. **Solaris** リリースをインストールする前にユーザー登録を無効にするには、次のいずれかを選択します。

- SUNWsregu パッケージを選択解除する (対話式インストール)
- SUNWsregu パッケージをインストールしないように、カスタム JumpStart プロファイルを変更する
- **finish** スクリプトを作成・実行して、1つまたは複数のシステム上の `/etc/default` ディレクトリに、下記の行を持つ `solregis` という名前のファイルを作成する。

```
DISABLE=1
```

詳細については、『Solaris 9 9/04 インストールガイド』または `solregis(1)` のマニュアルページを参照してください。

2. **Solaris** リリースをインストールした後にユーザー登録を無効にするには、次のいずれかを選択します。

- SUNWsregu パッケージを削除する
- `/etc/default` ディレクトリに上記の `solregis` ファイルを追加する

第 5 章

サーバーとクライアントサポートの管理 (概要)

この章では、ネットワーク環境でのサーバーとクライアントの管理について説明し、Solaris 環境でサポートされる各システム構成(「システムタイプ」と呼びます)に関する情報を紹介します。また、目的に合った最適なシステムを選択するためのガイドラインも示します。

この章の内容は次のとおりです。

- 103 ページの「サーバーおよびクライアント管理の新機能」
- 104 ページの「サーバーとクライアントタスクの操作手順」
- 105 ページの「サーバー、クライアント、およびアプライアンスとは」
- 105 ページの「クライアントサポートとは」
- 106 ページの「システムタイプの概要」
- 109 ページの「ディスクレスクライアント管理の概要」

ディスクレスクライアントサポートの管理手順については、[第 6 章](#)を参照してください。

サーバーおよびクライアント管理の新機能

この節では、Solaris 9 リリースのサーバーとクライアントの新しい管理機能について説明します。

ディスクレスクライアントサポート

この Solaris リリースでは、smossservice および smdiskless コマンドを使ってディスクレスクライアントを管理できます。ディスクレスクライアントとは、ディスクが搭載されておらず、すべてのサービスがサーバーに依存しているシステムのことです。

これらのコマンドは、Solaris 管理コンソールツール群の一部です。しかし、Solaris 管理コンソールでは、ディスクレスクライアントを管理できません。ディスクレスクライアントの管理に使用できるのは、smossservice コマンドおよび smdiskless コマンドに限られます。

ディスクレスクライアントの管理方法の詳細については、109 ページの「ディスクレスクライアント管理の概要」および第 6 章を参照してください。

サーバーとクライアントタスクの操作手順

サーバーとクライアントサポートの設定手順については、次の表を参照してください。

サーバー/クライアントサービス	参照先
インストールまたは JumpStart クライアント	『Solaris 9 9/04 インストールガイド』
Solaris 9 でのディスクレスクライアントシステム	109 ページの「ディスクレスクライアント管理の概要」および第 6 章
Solaris の以前のリリースのディスクレスクライアントシステム	『Solstice AdminSuite 2.3 管理者ガイド』

サーバー、クライアント、およびアプライアンスとは

通常、ネットワーク上のシステムは、次のいずれかに該当します。

システムタイプ	説明
サーバー	ネットワーク上のほかのシステムにサービスを提供するシステム。ファイルサーバー、ブートサーバー、Web サーバー、データベースサーバー、ライセンスサーバー、印刷サーバー、インストールサーバー、さらに、特定のアプリケーション用のサーバーなどもあります。この章では、サーバーとは、ネットワーク上の他のシステムにブートサービスとファイルシステムを提供するシステムのことを意味します。
クライアント	サーバーから提供されるリモートサービスを利用するシステム。クライアントによってはまったくディスクを持たず、サーバーから提供されるファイルシステムに依存するものもあります。ディスクレスシステム、およびアプライアンスシステムは、このタイプのクライアントの一例です。 またサーバーが提供するリモートサービス (インストールソフトウェアなど) を利用しながらも、サーバーに依存しなくても機能するクライアントもあります。この種類のクライアントの例として、ルート (/)、/usr、/export/home ファイルシステムとスワップ空間をそのハードディスクに含むスタンドアロンシステムがあります。
アプライアンス	Sun Ray アプライアンスのようなネットワークアプライアンスを使うと、アプリケーションや Solaris 環境にアクセスできます。アプライアンスを使うと、サーバー管理を一元化できるので、クライアント管理またはアップグレードを行う必要がなくなります。Sun Ray アプライアンスでは、「ホットデスク」も提供されます。この機能を使うと、サーバーグループのどのアプライアンスからでもただちに、作業中のコンピュータセッション (正確に言うと作業を中断したところ) にアクセスできます。詳細については、 http://www.sun.com/products/sunray を参照してください。

クライアントサポートとは

クライアントに対するサポートとは、クライアントの機能を促進するソフトウェアおよびサービスを提供することです。サポートには、次の項目が含まれます。

- システムをネットワークに認識させる (ホスト名と Ethernet アドレス情報)。
- システムをリモートからインストールおよびブートできるインストールサービスを提供する。
- ディスク容量に限りがある、またはディスク容量がないシステムにオペレーティングシステム (OS) サービスおよびアプリケーションサービスを提供する。

システムタイプの概要

システムタイプは、ルート (/) と /usr ファイルシステム (スワップ領域を含む) にアクセスする方法によって決まる場合があります。たとえば、スタンドアロンとサーバーシステムでは、これらのファイルシステムをローカルディスクからマウントしていますが、その他のクライアントでは、これらのファイルシステムをリモートからマウントし、サーバーから提供されるサービスに依存しています。次の表にそれぞれのシステムタイプの特徴を示します。

表 5-1 一般的なシステムタイプの特徴

システムタイプ	ローカルファイルシステム	ローカルスワップ領域	リモートファイルシステム	ネットワーク利用度	相対パフォーマンス
サーバー	ルート (/) /usr /home /opt /export/home /export/root	あり	なし	高	高
スタンドアロンシステム	ルート (/) /usr /export/home	あり	なし	低	高
ディスクレスクライアント	なし	なし	ルート (/) スワップ /usr /home	高	低
アプライアンス	なし	なし	なし	高	高

サーバー

サーバーシステム上には、次のファイルシステムがあります。

- ルート (/) と /usr ファイルシステム、およびスワップ領域
- /export と /export/home ファイルシステム。クライアントシステムをサポートし、ユーザーにホームディレクトリを提供する
- アプリケーションソフトウェアを格納する /opt ディレクトリまたはファイルシステム

サーバー上には、他のシステムをサポートするために次のソフトウェアも格納できません。

- サーバーとは異なるプラットフォームで、別のリリースまたはクライアントを稼働させるディスクレスシステム用のオペレーティングシステム (OS) サービス
- ネットワークに接続されたシステムがリモートインストールを実行するのに必要な Solaris CD のイメージソフトウェアとブート用ソフトウェア
- ネットワークに接続されたシステムがカスタム JumpStart™ インストールを行うのに必要な JumpStart ディレクトリ

スタンドアロンシステム

「ネットワークに接続されたスタンドアロンシステム」は、ネットワーク上の他のシステムと情報を共有できますが、ネットワークから切り離されても機能できます。

スタンドアロンシステムは、ルート (/)、/usr、/export/home の各ファイルシステムとスワップ空間を含むハードディスクを自ら持つため、独立して動作できます。つまり、スタンドアロンシステムは、オペレーティングシステムのソフトウェア、実行可能ファイル、仮想メモリー空間、ユーザーが作成したファイルにローカルにアクセスできます。

注 - スタンドアロンシステムに必要なファイルシステムを保持するには、十分なディスク領域が必要です。

「ネットワークに接続されないスタンドアロンシステム」は、ネットワークに接続されていない点を除き、ネットワークに接続されたスタンドアロンシステムと同じです。

ディスククライアント

「ディスククライアント」とは、ディスクが搭載されておらず、必要なすべてのソフトウェアおよび記憶装置をサーバーに依存しているシステムのことです。ディスククライアントには、サーバーからリモートで、ルート (/)、/usr、および /home ファイルシステムがマウントされます。

ディスククライアントでは、ネットワークを介してオペレーティングシステムソフトウェアおよび仮想メモリー領域に継続的にアクセスする必要があるため、かなりのネットワークトラフィックが発生します。ディスククライアントは、ネットワークから切り離されたり、そのサーバーが正しく機能しない場合は機能できません。

ディスククライアントの概要については、109 ページの「ディスククライアント管理の概要」を参照してください。

アプライアンス

アプライアンス (たとえば、Sun Ray アプライアンス) は、管理を必要としない X ディスプレイデバイスです。このデバイスには、CPU、ファン、ディスクがなく、メモリーもわずかしが搭載されていません。アプライアンスは Sun ディスプレイモニターに接続されていますが、アプライアンスユーザーのデスクトップセッションは、サーバーで実行され、その結果がユーザーのモニタに表示されます。ユーザーの X 環境は自動的に設定されます。この環境には次の特徴があります。

- ほかのファイルシステムおよびソフトウェアアプリケーションへのアクセスは、サーバーに依存
- ソフトウェアの一元管理およびリソース共有機能を提供
- 永続的なデータがない、FRU (現場交換可能ユニット) として使用

システムタイプ選択のガイドライン

次の特徴に基づいてそれぞれのシステムタイプを比較することにより、使用中の環境にどのシステムタイプが適切かを判断することができます。

- 一元管理
 - システムを FRU (現場交換可能ユニット) として扱えるか。これは、時間がかかるバックアップや復旧操作を必要とせず、またシステムデータを失わずに、障害が発生したシステムを直ちに新しいシステムと交換できることを意味する
 - システムをバックアップする必要があるか。数多くのデスクトップシステムのバックアップを実行するには、時間とリソースの点で多大の費用コストがかかる場合がある
 - システムのデータは、中央サーバーから変更できるか
 - クライアントシステムのハードウェアを操作せず、短時間で簡単に、システムを中央サーバーからインストールできるか

- 性能
 - この構成は、デスクトップで使用しても性能が低下しないか
 - ネットワークにシステムを追加すると、既存のネットワーク上のシステムの性能に影響を与えるか
- ディスク使用率
 - この構成を効果的に導入するには、どれくらいのディスク容量が必要か

次の表では、各システムタイプの順位をカテゴリ別に表示しています。1は、もっとも効果があることを意味します。4は、もっとも効果が低いことを意味します。

表 5-2 システムタイプの比較

システムタイプ	一元管理	性能	ディスク使用率
スタンドアロンシステム	4	1	4
ディスクレスクライアント	1	4	1
アプライアンス	1	1	1

ディスクレスクライアント管理の概要

次の節および第 6 章では、Solaris 9 リリースでのディスクレスクライアントサポートの管理方法について説明します。

「ディスクレスクライアント」とは、オペレーティングシステム、ソフトウェア、および記憶装置を「OS サーバー」に依存しているシステムのことです。ディスクレスクライアントは、そのルート (/)、/usr、およびその他のファイルシステムを OS サーバーからマウントします。ディスクレスクライアントは独自の CPU と物理メモリーを持っており、データをローカルで処理することができます。しかしディスクレスクライアントは、ネットワークから切り離されたり、その OS サーバーが正しく機能しない場合は機能できません。ディスクレスクライアントは、ネットワークを経由して継続的に機能する必要があるため、多大なネットワークトラフィックを発生させます。

以前の Solaris リリースでは、Solstice グラフィカル管理ツールでディスクレスクライアントが管理されました。Solaris 9 リリースでは、ディスクレスクライアントの smossservice コマンドおよび smdiskless コマンドを使って、OS サービスおよびディスクレスクライアントサポートを管理できるようになりました。

OS サーバーおよびディスクレスクライアントのサポート情報

次の表に `smosservice` コマンドおよび `smdiskless` コマンドをサポートする Solaris リリースおよびアーキテクチャタイプを示します。

アーキテクチャタイプ	Solaris 2.6	Solaris 7	Solaris 8 1/01、4/01、7/01、10/01、2/02	Solaris 9
SPARC サーバー	O	O	O	O
x86 サーバー	O	O	O	O
SPARC クライアント	O	O	O	O
x86 クライアント	X	X	X	O

次の表に `smosservice` コマンドおよび `smdiskless` コマンドがサポートする OS サーバーとクライアントの組み合わせを示します。

	Solaris 2.6 リリースサポート	Solaris 7 リリースサポート	Solaris 8 1/01、4/01、7/01、10/01、2/02 サポート	Solaris 9 サポート
OS サーバー / クライアント OS リリース	Solaris 2.6-Solaris 2.6	Solaris 7-Solaris 2.6、または 7	Solaris 8 1/01、4/01、7/01、10/01、2/02-Solaris 2.6、7、または 8 1/01、4/01、7/01、10/01、2/02	Solaris 9-Solaris 2.6、7、8 1/01、4/01、7/01、10/01、2/02

ディスクレスクライアント管理機能

`smosservice` コマンドおよび `smdiskless` コマンドを使うと、ネットワークにディスクレスクライアントサポートを追加したり、維持したりすることができます。ネームサービスを使うと、システム情報を一元管理できるので、ホスト名などの重要なシステム情報をネットワーク上のすべてのシステムに複製する必要がありません。

`smosservice` コマンドおよび `smdiskless` コマンドを使うと次の作業が実行できます。

- ディスクレスクライアントサポートの追加および変更

- OS サービスの追加および削除
- LDAP、NIS、NIS+、またはファイル環境でのディスクレスクライアント情報の管理

ディスクレスクライアントコマンドは、ディスクレスクライアントの起動の設定にのみ使用できます。このコマンドは、リモートインストールまたはプロファイルサービスなど、ほかのサービスの設定では使用できません。リモートインストールを設定するには、`sysidcfg` ファイルにディスクレスクライアント仕様を定義する必要があります。詳細については、『Solaris 9 9/04 インストールガイド』を参照してください。

ディスクレスクライアントコマンドの使用

次の表のコマンドを使って独自のシェルスクリプトを記述すると、簡単にディスクレスクライアント環境を設定および管理できます。

表 5-3 ディスクレスクライアントコマンド

コマンド	サブコマンド	作業
<code>/usr/sadm/bin/smosservice</code>	<code>add</code>	OS サービスを追加する
	<code>delete</code>	OS サービスを削除する
	<code>list</code>	OS サービスをリスト表示する
	<code>patch</code>	OS サービスのパッチを管理する
<code>/usr/sadm/bin/smdiskless</code>	<code>add</code>	ディスクレスクライアントを OS サーバーに追加する
	<code>delete</code>	ディスクレスクライアントを OS サーバーから削除する
	<code>list</code>	OS サーバー上のディスクレスクライアントをリスト表示する
	<code>modify</code>	ディスクレスクライアントの属性を変更する

次に示す 2 種類の方法で、これらのコマンドに関するヘルプを参照することができます。

- コマンド、サブコマンド、および必要なオプションを入力するときに、`-h` オプションを使用する。たとえば、`smdiskless add` の使用方法を表示するには、次のように入力する。

```
% /usr/sadm/bin/smdiskless add -p my-password -u my-user-name -- -h
```

- `smdiskless (1M)` または `smosservice (1M)` のマニュアルページを参照する。

ディスクレスクライアント管理に必要な RBAC 権限

`smosservice` コマンド および `smdiskless` コマンドはスーパーユーザーとして使用できます。役割によるアクセス制御 (RBAC) を使用している場合、割り当てられた RBAC 権限に応じて、すべてのディスクレスクライアントコマンドまたはそのサブセットのいずれかを使用できます。次の表にディスクレスクライアントコマンドを使用するのに必要な RBAC 権限を示します。

表 5-4 ディスクレスクライアントの管理に必要な権限

RBAC 権限	コマンド	作業
基本的な Solaris ユーザー、 ネットワーク管理	<code>smosservice list</code>	OS サービスをリスト表示する
	<code>smosservice patch</code>	OS サービスパッチをリスト表示する
	<code>smdiskless list</code>	ディスクレスクライアントをリスト表示する
ネットワーク管理	<code>smdiskless add</code>	ディスクレスクライアントを追加する
システム管理者	すべてのコマンド	すべての作業

OS サービスの追加

Solaris OS サーバーとは、ディスクレスクライアントシステムをサポートするオペレーティングシステム (OS) サービスを提供するサーバーです。OS サーバーをサポートするか、`smosservice` コマンドを使用してスタンドアロンシステムを OS サーバーに変換することができます。

サポートする各プラットフォームグループおよび Solaris リリース用に、特定の OS サービスを OS サーバーに追加する必要があります。たとえば、Solaris 8 リリース上で実行中の SPARC Sun4m システムをサポートする場合には、Sun4m/Solaris 8 OS サービスを OS サーバーに追加する必要があります。また、それぞれのシステムが属するプラットフォームグループが異なるため、Solaris 8 のリリースを実行する SPARC Sun4c システムまたは x86 システムをサポートする OS サービスも追加する必要があります。

OS サービスを追加するには、適切な Solaris CD またはディスクイメージへのアクセス権が必要です。

OS サーバーにパッチが適用された場合に OS サービスを追加する

OS サーバーに OS サービスを追加しようとする、サーバー上の OS と追加するサービスの OS のバージョンが一致しないというエラーメッセージが表示される場合があります。このメッセージは、OS サーバーにインストールされている OS のパッケージにパッチが適用されていても、追加しようとしている OS サービスのパッケージにはパッチが適用されていない (パッチがパッケージに組み込まれているため) 場合に表示されます。

たとえば、Solaris 7 リリースを実行するサーバーがあるとして。また、このサーバーには、パッチが適用された Solaris 2.6 SPARC sun4m OS サービスなど、ほかの OS サービスも追加されているとして。CD-ROM から Solaris 2.6 SPARC sun4c OS サービスをこのサーバーに追加しようとする、次のようなエラーメッセージが表示される場合があります。

```
Error: inconsistent revision, installed package appears to have been
patched resulting in it being different than the package on your media.
You will need to backout all patches that patch this package before
retrying the add OS service option.
```

OS サーバーに必要なディスク容量

ディスクレスクライアント環境を設定する前に、各ディスクレスクライアントディレクトリに必要なディスク容量があるかどうか確認する必要があります。

以前のバージョンの Solaris リリースでは、インストール中にディスクレスクライアントサポートについてのプロンプトが表示されました。Solaris 9 リリースでは、手動でインストール中に /export ファイルシステムを割り当てるか、またはインストール後に作成してください。具体的なディスク容量要件については、次の表を参照してください。

表 5-5 OS サーバーに必要なディスク容量

ディレクトリ	必要な容量 (M バイト)
/export/Solaris_version	10
/export/exec	800
/export/share	5
/export/swap/diskless_client	32 (デフォルトのサイズ)
/export/dump/diskless_client	32 (デフォルトのサイズ)
/export/root/templates/Solaris_version	30
/export/root/clone/Solaris_version/ machine_class	30 - 60 (マシンクラスによって異なる)

表 5-5 OS サーバーに必要なディスク容量 (続き)

ディレクトリ	必要な容量 (M バイト)
/export/root/diskless_client (上記のクローン)	30 - 60 (マシンクラスによって異なる)
/tftpboot/inetboot.machine_class.Solaris_ version	machine_class.Solaris_version ごとに 200 K バイト

第 6 章

ディスクレスクライアントの管理 (手順)

ここでは、Solaris 環境でディスクレスクライアントを管理する方法について説明します。

ディスクレスクライアントの管理手順については、115 ページの「ディスクレスクライアントの管理 (作業マップ)」を参照してください。

ディスクレスクライアント管理の概要については、第 5 章を参照してください。

Solstice AdminSuite™ ソフトウェアを使ってクライアントを管理する方法については、『Solstice AdminSuite 2.3 管理者ガイド』を参照してください。

ディスクレスクライアントの管理 (作業マップ)

次の表にディスクレスクライアントの管理に必要な手順を示します。

作業	説明	参照先
1. (省略可能) 既存のディスクレスクライアントサポートを削除する	Solstice AdminSuite 製品と一緒に追加されたディスクレスクライアントが存在する場合、Solaris リリースをインストールする前に <code>admhostdel</code> コマンドおよび <code>admhostmod</code> コマンドを使って、ディスクレスクライアントサポートおよび OS サービスを削除する	『Solstice AdminSuite 2.3 管理者ガイド』

作業	説明	参照先
2. (省略可能) Solaris 管理コンソールログインを有効にしてディスクレスクライアントのエラーメッセージを表示する	コンソールのメインウィンドウからログビューアを選択し、ディスクレスクライアントのエラーメッセージを表示する	42 ページの「Solaris 管理コンソールを起動する」
3. ディスクレスクライアントを追加する準備をする	サポートされているリリースを確認し、各ディスクレスクライアントの「プラットフォーム」、「メディアパス」、および「クラスタ」(またはソフトウェアグループ)を識別する	118 ページの「ディスクレスクライアントの追加の準備」
4. 必要な OS サービスを OS サーバーに追加する	smossservice コマンドを使用してサポートしたいディスクレスクライアントの OS サービスを追加する。サポートする各ディスクレスクライアントプラットフォームのプラットフォームおよびメディアパスを識別する必要がある	120 ページの「ディスクレスクライアントサポートの OS サービスの追加方法」
5. ディスクレスクライアントを追加する	smdiskless コマンドを使用してすべての必須情報を指定し、ディスクレスクライアントサポートを追加する	121 ページの「ディスクレスクライアントの追加方法」
6. ディスクレスクライアントを起動する	ディスクレスクライアントを起動して、ディスクレスクライアントサポートが正常に追加されたことを確認する	123 ページの「ディスクレスクライアントの起動方法」
7. (省略可能) ディスクレスクライアントサポートを削除する	必要がなくなった場合には、ディスクレスクライアントのサポートを削除する	123 ページの「ディスクレスクライアントサポートの削除方法」
8. (省略可能) ディスクレスクライアントの OS サービスを削除する	必要がなくなった場合には、ディスクレスクライアントの OS サービスを削除する	124 ページの「ディスクレスクライアントの OS サービスを削除する方法」
9. (省略可能) OS サービスにパッチを適用する	ディスクレスクライアント OS サービスのパッチを追加、削除、リスト表示、または同期化する	125 ページの「ディスクレスクライアントの OS パッチの追加方法」

ディスクレスクライアントの管理

ここでは、ディスクレスクライアントの管理に必要な手順について説明します。

ディスクレスクライアントを管理する場合には、次の点に注意してください。

- Solaris インストールプログラムでは、ディスクレスクライアントサポートのセットアップを促すメッセージが表示されません。ディスクレスクライアントをサポートするには、手動で `/export` パーティションを作成してください。インストール中またはインストール後に `/export` パーティションを作成してください。
- `/export` パーティションには、サポートするクライアントの数に応じて、最低 800 – 1000 MB の容量が必要です。詳細については、113 ページの「OS サーバーに必要なディスク容量」を参照してください。
- `smosservice` コマンドまたは `smdiskless` コマンドで指定したネームサービスは、`/etc/nsswitch.conf` ファイルで指定されたプライマリネームサービスと一致していなければなりません。`smdiskless` コマンドまたは `smosservice` コマンドにネームサービスを指定しない場合、デフォルトのネームサービスは `files` になります。

各ディスクレスクライアントについてプラットフォーム、メディアパス、およびクラスタを決定したら、OS サービスを追加する準備ができたこととなります。追加する各 OS サービスについて、次のディレクトリが作成され移植されます。

- `/export/Solaris_version/Solaris_version_instruction_set.all`
(`/export/exec/Solaris_version/Solaris_version_instruction_set.all` へのシンボリックリンク)
- `/export/Solaris_version`
- `/export/Solaris_version/var`
- `/export/Solaris_version/opt`
- `/export/share`
- `/export/root/templates/Solaris_version`
- `/export/root/clone`
- `/export/root/clone/Solaris_version`
- `/export/root/clone/Solaris_version/machine_class`

追加する各ディスクレスクライアントについて、次のデフォルトのディレクトリが OS サーバー上に作成されます。

- `/export/root/diskless_client`
- `/export/swap/diskless_client`
- `/tftpboot/diskless_client_ipaddress_in_hex /export/dump/diskless_client (-x dump` オプションを指定した場合)

注 -x オプションを使うと、ルート、 /swap、および /dump ディレクトリのデフォルト位置を変更することができます。ただし、 /export ファイルシステムの下にはこれらのディレクトリを作成しないでください。

▼ ディスクレスクライアントの追加の準備

サポートしているリリースが、OS サービスに指定されたシステム上で実行されていることを確認します。さらに、OS サーバリリースとディスクレスクライアントリリースの組み合わせがサポートされていることも確認します。

smossservice add コマンドを使用して OS サービスを追加する場合は、サポートしたい各ディスクレスクライアントプラットフォームの「プラットフォーム」、「メディアパス」、および「クラスタ」(またはソフトウェアグループ)を指定する必要があります。

1. サポートされる **Solaris** リリースの組み合わせが、対象の **OS** サーバおよびディスクレスクライアントで実行されているかどうか確認します。
詳細については、110 ページの「OS サーバおよびディスクレスクライアントのサポート情報」を参照してください。

2. 次の形式で使用されるディスクレスクライアントプラットフォームを識別します。
instruction_set.machine_class.Solaris_version
たとえば、次のようになります。

```
sparc.sun4u.Solaris_9
```

以下に、利用できるプラットフォームのオプションを示します。

<i>instruction_set</i>	<i>machine_class</i>	<i>Solaris_version</i>
sparc	sun4d*, sun4c*, sun4m*, および sun4u	Solaris_9, Solaris_8, Solaris_2.7, Solaris_2.6
i386	i86pc	Solaris_9, Solaris_8, Solaris_2.7, Solaris_2.6

* sun4c アーキテクチャは、Solaris 8 および Solaris 9 リリースではサポートされていません。sun4d アーキテクチャは、Solaris 9 リリースではサポートされていません。

3. メディアパスを特定します。これは、ディスクレスクライアントにインストールしたいオペレーティングシステムを含むディスクイメージのフルパスです。
Solaris オペレーティングシステムは、複数の CD で配布されます。ただし、smossservice コマンドでは、複数の CD から OS を読み込むことはできません。ユーザーは、次の操作を行うために Solaris Software CD (および付属する

Languages CD) にあるスクリプトを実行する必要があります。

- サーバー上にインストールイメージを作成します。インストールサーバーのセットアップについての情報は、『Solaris 9 9/04 インストールガイド』を参照してください。
- 次のいずれかのスクリプトを使用して、必要な OS サービスを CD イメージから読み込みます。
 - Solaris 9 Software 1 of 2 CD 用 –
/cdrom/cdrom0/s0/Solaris_9/Tools/setup_install_server
 - Solaris 9 Software 2 of 2 CD 用 –
/cdrom/cdrom0/Solaris_9/Tools/add_to_install_server
 - Solaris 9 Languages CD 用 –
/cdrom/cdrom0/Tools/add_to_install_server

たとえば、ローカルに接続された CD-ROM デバイス上の Solaris 9 Software 1 of 2 CD (SPARC 版) から setup_install_server スクリプトを使用する場合は、構文が次のようになります。

```
# mkdir /export/install/sparc_9
# cd /cd_mount_point/Solaris_9/Tools
# ./setup_install_server /export/install/sparc_9
```

- ディスクに Solaris CD イメージをインストールした後、ディスクイメージのパスを指定してください。たとえば、次のようになります。

```
/net/export/install/sparc_9
```

4. OS サービスを追加するときに SUNWCxall クラスタを特定します。

同じシステム (SPARC または x86) 上で同じオペレーティングシステムを実行するディスクレスクライアントには、「同じクラスタ」を使用しなければなりません。たとえば、次のディスクレスクライアントについて考えてみましょう。

- sparc.sun4m.Solaris_9
- sparc.sun4u.Solaris_9

sun4u および sun4m システムでは SUNWCxall クラスタを要求するため、これらのディスクレスクライアントをセットアップするには、各ディスクレスクライアントに SUNWCxall クラスタを指定する必要があります。また同じシステムで同じオペレーティングリリース (ここでは Solaris_9) を実行するディスクレスクライアントでは同じクラスタを使用する必要があります。

注 – sun4u システム、または 8 ビットの高速カラーメモリーフレームバッファ (cgsix) が搭載されたシステムを使用している場合には、クラスタに必ず SUNWCxall を指定してください。

▼ ディスクレスクライアントサポートの OS サービスの追加方法

ディスクレスクライアントの OS サービスをサーバーに追加するには、次の手順を実行します。

1. サーバーでスーパーユーザーになるか、同等の役割を引き受けます。
詳細については、36 ページの「スーパーユーザー (root) になるか役割を引き受ける方法」を参照してください。
2. **Solaris** 管理コンソールサーバーが実行中であり、システムでディスクレスクライアントツールが使用できることを確認します。

```
# /usr/sadm/bin/smosservice list -H starbug:898 --
ツール com.sun.admin.osservmgr.cli.OsServerMgrCli を ... から読み込み中
ユーザー root として starbug にログインしました。
starbug:898 から com.sun.admin.osservmgr.cli.OsServerMgrCli がダウンロードされました。
Platform
-----
```

3. OS サービスを追加します。

```
# /usr/sadm/bin/smosservice add -H hostname:898 -- -o hostname
-x mediapath=path -x platform=instruction-set.machine-class.Solaris-version
-x cluster=cluster-name -x locale=locale-name
```

<code>add</code>	指定された OS サービスを追加する
<code>-H hostname:898</code>	接続するホスト名とポートを指定する。 ポートを指定しない場合は、デフォルト ポート 898 に接続される
<code>--</code>	これ以降のサブコマンド引数が始まること を示す
<code>-x mediapath=path</code>	Solaris イメージのフルパスを指定する
<code>-x platform=instruction-set.machine-class.Solaris- version</code>	追加する命令アーキテクチャ、マシ ンクラス、および Solaris バージ ョンを指定する
<code>-x cluster=cluster-name</code>	インストールする Solaris クラ スタを指定する
<code>-x locale=locale-name</code>	インストールするロケールを指 定する

注 - サーバーのスピードおよび選択した OS サービスの構成により、インストールプロセスには 45 分ほどかかることがあります。

詳細については、smossservice(1M)のマニュアルページを参照してください。

4. (省略可能) ほかの OS サービスを追加します。
5. OS サービスを追加し終わったら、OS サービスがインストールされているかどうか確認します。

```
# /usr/sadm/bin/smossservice list -H hostname:898 --
```

例 — ディスクレスクライアントサポートの OS サービスを追加する

この例では、サーバー starbug に Solaris 8 OS サービスを追加する方法を示します。サーバー starbug では Solaris 9 リリースを実行しています。

```
# /usr/sadm/bin/smossservice add -H starbug:898 -- -o starbug
-x mediapath=/net/install/export/sparc_8 -x platform=sparc.sun4u.Solaris_8
-x cluster=SUNWCXall -x locale=en_US
Authenticating as user: root
```

ヘルプを参照するには /? を入力してください。

Enter キーを押すと、[] で囲まれたデフォルトが選択されます。

文字列の値を入力してください: password :: xxx

ツール com.sun.admin.osservmgrp.cli.OsServerMgrCli を starbug:898 から読み込み中
ユーザー root として starbug にログインしました。

starbug:898 から com.sun.admin.osservmgrp.cli.OsServerMgrCli がダウンロードされました。

▼ ディスクレスクライアントの追加方法

OS サービスを追加した後に、ディスクレスクライアントを追加するには、次の手順を実行します。

1. スーパーユーザーになるか、同等の役割を引き受けます。

詳細については、36 ページの「スーパーユーザー (root) になるか役割を引き受ける方法」を参照してください。

2. ディスクレスクライアントを追加します。

```
# /usr/sadm/bin/smdiskless add -- -i ip-address -e ethernet-address
-n client-name -x os=instruction-set.machine-class.Solaris-version
-x root=/export/root/client-name -x swap=/export/swap/client-name
-x swappsize=size -x tz=timezone -x locale=locale-name
```

add

指定したディスクレスクライアントを追加する

--	これ以降、サブコマンド引数が始まることを示す
-i <i>ip-address</i>	ディスクレスクライアントの IP アドレスを指定する
-e <i>ethernet-address</i>	ディスクレスクライアントの Ethernet アドレスを識別する
-n <i>client-name</i>	ディスクレスクライアントの名前を指定する
-x <i>os=instruction-set.machine-class.Solaris-version</i>	ディスクレスクライアントの命令アーキテクチャ、マシンクラス、OS、および Solaris バージョンを指定する
-x <i>root=root=/export/root/client-name</i>	ディスクレスクライアントのルートディレクトリを指定する
-x <i>swap=root=/export/root/client-name</i>	ディスクレスクライアントのスワップファイルを指定する
-x <i>swapsize=size</i>	スワップファイルのサイズをメガバイト (MB) で指定する。デフォルトは 24 (MB)
-x <i>tz=timezone</i>	ディスクレスクライアントの時間帯を指定する
-x <i>locale=locale-name</i>	ディスクレスクライアントをインストールするロケールを指定する

詳細については、`smdiskless (1M)` のマニュアルページを参照してください。

3. (省略可能) `smdiskless add` コマンドを続けて使用して、各ディスクレスクライアントを追加します。
4. ディスクレスクライアントがインストールされたことを確認します。

```
# /usr/sadm/bin/smosservice list -H hostname:898 --
```

例 — ディスクレスクライアントを追加する

この例では、サーバー `starbug` から Solaris 8 クライアント、`holoship` の追加方法を示します。

```
# /usr/sadm/bin/smdiskless add -- -i 172.20.27.103 -e 8:0:20:92:4e:f3
-n holoship -x os=sparc.sun4u.Solaris_8 -x root=/export/root/holoship
-x swap=/export/swap/holoship -x swapsize=128 -x tz=US/Mountain
-x locale=en_US
```

この例では、サーバー `starbug` から Solaris 7 クライアント、`inquisitor` の追加方法を示します。

```
# /usr/sadm/bin/smdiskless add -- -i 172.20.27.102 -e 8:0:20:1f:31:be
-n inquisitor -x os=sparc.sun4u.Solaris_2.7 -x root=/export/root/inquisitor
-x swap=/export/swap/inquisitor -x swappsize=64 -x tz=US/Mountain
```

▼ ディスクレスクライアントの起動方法

1. OS サーバーの次の必要条件を確認します。

- ディスクレスクライアントおよび OS サービスの追加に使用するネームサービスがサーバーの `/etc/nsswitch.conf` ファイルのプライマリ名に一致していることを確認します。

一致しない場合は、ディスクレスクライアントが起動しません。

- `rpc.bootparamd` デーモンが実行中かどうか確認します。実行していない場合には、それを起動してください。

2. ディスクレスクライアントを起動します。

```
ok boot net
```

▼ ディスクレスクライアントサポートの削除方法

1. スーパーユーザーになるか、同等の役割を引き受けます。

詳細については、[36 ページの「スーパーユーザー \(root\) になるか役割を引き受ける方法」](#)を参照してください。

2. ディスクレスクライアントサポートを削除します。

```
# /usr/sadm/bin/smdiskless delete -- -o hostname:898 -n client-name
```

3. ディスクレスクライアントサポートが削除されたことを確認します。

```
# /usr/sadm/bin/smosservice list -H hostname:898 --
```

例 — ディスクレスクライアントサポートを削除する

この例では、OS サーバー `starbug` から ディスクレスクライアント `holoship` を削除する方法を示します。

```
# /usr/sadm/bin/smdiskless delete -- -o starbug -n holoship
Authenticating as user: root
```

ヘルプを参照するには `/?` を入力してください。

Enter キーを押すと、`[]` で囲まれたデフォルトが選択されます。

文字列の値を入力してください: `password ::`

Starting SMC server version 2.0.0.

endpoint created: :898

SMC server is ready.
ツール com.sun.admin.osservicemgr.cli.OsServerMgrCli を starbug から読み込み中
ユーザー root として starbug にログインしました。
starbug から com.sun.admin.osservicemgr.cli.OsServerMgrCli がダウンロードされました。

▼ ディスクレスクライアントの OS サービスを削除する方法

1. スーパーユーザーになるか、同等の役割を引き受けます。
詳細については、36 ページの「スーパーユーザー (root) になるか役割を引き受ける方法」を参照してください。

2. ディスクレスクライアントの OS サービスを削除します。

```
# /usr/sadm/bin/smosservice delete -H hostname:898 --  
-x rmlatform=instruction-set.machine-class.Solaris-version
```

3. OS サービスが削除されたことを確認します。

```
# /usr/sadm/bin/smosservice list -H hostname:898 --
```

例 — ディスクレスクライアントの OS サービスを削除する

次の例では、サーバー starbug からディスクレスクライアント OS サービス (sparc.all.Solaris_9) を削除する方法を示します。

```
# /usr/sadm/bin/smosservice delete -H starbug:898 --  
-x rmlatform=sparc.all.Solaris_9  
Authenticating as user: root  
ヘルプを参照するには /? を入力してください。  
Enter キーを押すと、[ ] で囲まれたデフォルトが選択されます。  
文字列の値を入力してください: password :  
ツール com.sun.admin.osservicemgr.cli.OsServerMgrCli を starbug:898 から読み込み中  
ユーザー root として starbug にログインしました。  
starbug:898 から com.sun.admin.osservicemgr.cli.OsServerMgrCli がダウンロードされました。
```

ディスクレスクライアント OS サービスにパッチを適用する

smosservice patch コマンドを使うと次の操作を実行できます。

- OS サーバー上に /export/diskless/Patches パッチスプールディレクトリを設定する

- パッチスプールディレクトリにパッチを追加する。追加するパッチにより、スプールの既存のパッチが不要になると、不要なパッチは `/export/diskless/Patches/Archive` に移動される
- パッチスプールディレクトリからパッチを削除する
- パッチスプールディレクトリ内のパッチをリスト表示する
- スプールされたパッチをクライアントに同期させる。クライアントが更新されたパッチを認識できるように、同期させた各クライアントをリブートする必要がある

注 – 推奨 OS パッチを定期的にインストールして、OS サーバーを最新の状態に保ってください。

パッチのダウンロード方法については、284 ページの「署名のない Solaris パッチのダウンロード方法」を参照してください。

ディスクレスクライアントの OS パッチの表示

ディスクレスクライアントのパッチは、パッチのタイプに応じて、異なるディレクトリにロギングされます。

- カーネルパッチは、ディスクレスクライアントの `/var/sadm/patch` ディレクトリにロギングされます。カーネルパッチを表示するには、ディスクレスクライアントで次のコマンドを入力します。

```
% patchadd -p
```

- `/usr` パッチは、OS サーバーの `/export/Solaris_version/var/patch` ディレクトリにロギングされます。パッチ ID ごとにディレクトリが作成されます。`/usr` パッチを表示するには、OS サーバーで次のコマンドを入力します。

```
% patchadd -S Solaris_8 -p
```

```
Patch: 111879-01 Obsoletes: Requires: Incompatibles: Packages: SUNWwsr
```

OS およびアーキテクチャによりスプールされたすべてのパッチを表示するには、`-p` オプションを指定した `smosservice` コマンドを使用します。

▼ ディスクレスクライアントの OS パッチの追加方法

1. サーバーでスーパーユーザーになるか、同等の役割を引き受けます。
詳細については、36 ページの「スーパーユーザー (root) になるか役割を引き受ける方法」を参照してください。
2. ディスクレスクライアントシステムにログインし、シャットダウンします。

```
# init 0
```

3. パッチをスプールディレクトリに追加します。

```
# /usr/sadm/bin/smosservice patch -- -a /var/patches/patch-ID-revision
Authenticating as user: root
```

ヘルプを参照するには /? を入力してください。
Enter キーを押すと、[] で囲まれたデフォルトが選択されます。
文字列の値を入力してください: password ::
ツール com.sun.admin.osservmgrp.cli.OsServerMgrCli を starbug から読み込み中
ユーザー root として starbug にログインしました。
starbug から com.sun.admin.osservmgrp.cli.OsServerMgrCli がダウンロードされました。
追加するパッチが別のパッチに依存する場合は、次のメッセージが表示され、
パッチを追加できません。

```
The patch patch-ID-revision could not be added
because it is dependent on other patches which have not yet been spooled.
You must add all required patches to the spool first.
```

4. パッチがスプールされたことを確認します。

```
# /usr/sadm/bin/smosservice patch -- -P
```

5. スプールされたパッチをディスククライアントにプッシュします。

```
# /usr/sadm/bin/smosservice patch -- -m -U
Authenticating as user: root
```

ヘルプを参照するには /? を入力してください。
Enter キーを押すと、[] で囲まれたデフォルトが選択されます。
文字列の値を入力してください: password ::
ツール com.sun.admin.osservmgrp.cli.OsServerMgrCli を starbug から読み込み中
ユーザー root として starbug にログインしました。
starbug から com.sun.admin.osservmgrp.cli.OsServerMgrCli がダウンロードされました。

注-パッチをディスククライアントにプッシュおよび同期させるには、1
パッチあたり最大 90 分ほどかかることがあります。

6. パッチがディスククライアントに適用されたことを確認します。

```
# /usr/sadm/bin/smosservice patch -- -P
Authenticating as user: root
```

ヘルプを参照するには /? を入力してください。
Enter キーを押すと、[] で囲まれたデフォルトが選択されます。
文字列の値を入力してください: password ::
ツール com.sun.admin.osservmgrp.cli.OsServerMgrCli を starbug から読み込み中
ユーザー root として starbug にログインしました。
starbug から com.sun.admin.osservmgrp.cli.OsServerMgrCli がダウンロードされました。
Patches In Spool Area
Os Rel Arch Patch Id Synopsis

8 sparc 111879-01 SunOS 5.8: Solaris Product Registry patch SUNWwsr

```

Patches Applied To OS Services
Os Service                               Patch
-----
Solaris_8

Patches Applied To Clone Areas
Clone Area                               Patch
-----
Solaris_8/sun4u

```

例 — ディスクレスクライアントの OS パッチを追加する

この例では、Solaris 8 パッチ (111879-01) をサーバーのディスクレスクライアントの OS サービスに追加する方法を示します。

```

# /usr/sadm/bin/smosservice patch -- -a /var/patches/111879-01
Authenticating as user: root
.
.
.
# /usr/sadm/bin/smosservice patch -- -P
Patches In Spool Area
Os Rel Arch  Patch Id  Synopsis
-----
8          sparc  111879-01  SunOS 5.8: Solaris Product Registry patch SUNWwsr
.
.
.
# /usr/sadm/bin/smosservice patch -- -m -U
Authenticating as user: root
.
.
.
# /usr/sadm/bin/smosservice patch -- -P
Authenticating as user: root
.
.
.
Patches In Spool Area
Os Rel Arch  Patch Id  Synopsis
-----
8          sparc  111879-01  SunOS 5.8: Solaris Product Registry patch SUNWwsr

Patches Applied To OS Services
Os Service                               Patch
-----
Solaris_8

Patches Applied To Clone Areas
Clone Area                               Patch
-----
Solaris_8/sun4u

```

ディスクレスクライアント問題の障害追跡

ここでは、ディスクレスクライアントの一般的な問題と解決策をリスト表示します。

問題

- OS サーバーがクライアント RARP 要求に応答しない
- OS サーバーがクライアントの `bootparam` 要求に応答しない
- OS サーバーがディスクレスクライアントのルートファイルシステムをマウントできない

解決法

ファイル環境

- OS サーバーの `/etc/nsswitch.conf` ファイルの `hosts`、`ethers`、および `bootparams` の最初のソースとして `files` が設定されていることを確認する。
- クライアントの IP アドレスが `/etc/inet/hosts` ファイルに設定されていることを確認する。
- クライアントの Ethernet アドレスが `/etc/ethers` ファイルに表示されていることを確認する。
- `/etc/bootparams` ファイルにクライアントのルートおよびスワップ領域への次のパスが含まれていることを確認する。

```
client root=os-server:/export/root/client swap=os-server:  
/export/swap/client
```

スワップのサイズは、ディスクレスクライアントの追加時に `-x swapsize` オプションを指定したかどうかによって異なる。ディスクレスクライアントの追加時に `-x dump` オプションを指定した場合は、次の行が表示される。

```
dump=os-server:/export/dump/client dumpsize=24
```

ダンプサイズも、ディスクレスクライアントの追加時に `-x dumpsize` オプションを指定したかどうかによって異なる。

- OS サーバーの IP アドレスが `/export/root/client/etc/inet/hosts` ファイルに設定されているかどうかを確認する。

ネームサービス環境

- OS サーバーとクライアントの Ethernet アドレスおよび IP アドレスが正しくマップされていることを確認する。
- `/etc/bootparams` ファイルに、次のようにクライアントのルートおよびスワップ領域へのパスが含まれていることを確認する。


```
client root=os-server:/export/  
root/client swap=os-server:/export/  
swap/client swapsize=24
```

スワップのサイズは、ディスクレスクライアントの追加時に `-x swapsize` オプションを指定したかどうかによって異なる。ディスクレスクライアントの追加時に `-x dump` オプションを指定した場合は、次の行が表示される。

```
dump=os-server:/export/dump/client dumpsize=24
```

ダンプサイズも、ディスクレスクライアントの追加時に `-x dumpsize` オプションを指定したかどうかによって異なる。

問題

ディスクレスクライアントパニック

解決法

- OS サーバーの Ethernet アドレスが IP アドレスに正しくマップされていることを確認する。システムをあるネットワークから別のネットワークに物理的に移動した場合、システムの新しい IP アドレスを再マップするのを忘れていた可能性がある。
- クライアントの RARP、TFTP、または `bootparam` 要求に応答する「同じサブネット」の別のサーバーのデータベースに、クライアントのホスト名、IP アドレス、および Ethernet アドレスが存在しないことを確認する。インストールサーバーから OS をインストールするために、しばしばテストシステムがセットアップされる。このような場合、インストールサーバーはクライアントの RARP または `bootparam` 要求に対して、正しくない IP アドレスを返す。この不正なアドレスにより、間違ったアーキテクチャのブートプログラムをダウンロードしたり、クライアントのルートファイルシステムのマウントが失敗したりしている可能性がある。
- ディスクレスクライアントの TFTP 要求にインストールサーバー (または以前の OS サーバー) が応答しないことを確認する。このサーバーは不正なブートプログラムを転送している。ブートプログラムのアーキテクチャが異なる場合は、クライアントがただちにパニックになる。ブートプログラムが非 OS サーバーから読み込まれた場合、クライアントはそのルートパーティションを非 OS サーバー上に確保し、`/usr` パーティションを OS サーバー上に確保することがある。この状況では、ルートおよび `/usr` パーティションのアーキテクチャまたはバージョンが競合する場合、クライアントがパニックに陥る。
- インストールサーバーと OS サーバーの両方を使用している場合は、`/etc/dfs/dfstab` ファイルに次のエントリがあることを確認する。

```
share -F nfs -o -ro /export/exec/Solaris_version_instruction_set.all/usr
```

ここで、`version=2.6`、`2.7`、`8`、または `9`、また `instruction_set=sparc` または `i386`

- ディスクレスクライアントのルート、`/swap` パーティション、および `/dump` (指定されている場合) パーティションに共有エントリがあることを確認する。

```
share -F nfs -o rw=client,root=client /export/root/client  
share -F nfs -o rw=client,root=client /export/swap/client
```

```
share -F nfs -o rw=client,root=client /export/dump/client
```

- OS サーバーで次のコマンドを入力し、共有されているファイルを確認する。

```
% share
```

OS サーバーでは、ディスクレスクライアントを追加したときに指定した、
/export/root/client および/export/swap/client_name (デフォルト)、または
ルート、/swap パーティション、および /dump パーティションを共有する必要がある。

次のエントリが /etc/dfs/dfstab ファイルにあるかを確認する。

```
share -F nfs -o ro /export/exec/Solaris_version_instruction_set.all/usr  
share -F nfs -o rw=client,root=client /export/root/client  
share -F nfs -o rw=client,root=client /export/swap/client
```

問題

OS サーバーがディスクレスクライアントの RARP 要求に応答しない

解決法

クライアントの目的の OS サーバーから、次のクライアントの Ethernet アドレスを使って snoop コマンドを root として実行する。

```
# snoop xx:xx:xx:xx:xx:xx
```

問題

ブートプログラムをダウンロードしたが、プロセスの初期にパニックが発生した

解決法

snoop コマンドを使用して、対象の OS サーバーがクライアントの TFTP および NFS 要求に応答するかどうかを確認する。

問題

- ディスクレスクライアントのハングアップ
- ディスクレスクライアントの RARP 要求に対するサーバーの応答が不正である

解決法

OS サーバーで次のデーモンを再起動する。

```
# /usr/sbin/rpc.bootparamd  
# /usr/sbin/in.rarpd -a
```

第 7 章

システムのシャットダウンとブート (概要)

この章では、システムのシャットダウンとブートについて概要を説明します。Solaris オペレーティングシステムは、電子メールとネットワークリソースをいつでも利用できるように、停止することなく動作するように設計されています。しかし、システム構成の変更、定期保守、停電などの理由で、システムをシャットダウンまたはリポートしなければならない場合があります。

この章の内容は次のとおりです。

- 131 ページの「システムのシャットダウンとブートに関する新機能」
- 133 ページの「シャットダウンとブートについての参照先」
- 133 ページの「シャットダウンとブートの用語」
- 134 ページの「システムのシャットダウンに関するガイドライン」
- 134 ページの「システムのブートに関するガイドライン」
- 135 ページの「ネットワークからのシステムのブート」
- 135 ページの「システムをシャットダウンする場合」
- 136 ページの「システムをブートする場合」

システムのシャットダウンとブートに関する新機能

この節では、Solaris 9 におけるシステムのシャットダウンとブートに関する新機能について説明します。

PXE ネットワークブート

PXE (Preboot Execution Environment) ネットワーク起動プロトコルをサポートしている x86 システムでは、Solaris ブートフロッピーディスクを使用しなくても、ネットワークから直接 Solaris x86 版をブートできます。PXE ネットワークブートは、Intel の PXE 仕様を実装しているデバイスでのみ動作します。

システム BIOS またはネットワークアダプタ BIOS のどちらか一方、またはその両方の BIOS 設定プログラムを使うことによって、クライアントシステム上で PXE ネットワークブートを使用できるようにします。いくつかのシステムでは、他のデバイスからのブートよりも先にネットワークブートが実行されるように、ブートデバイスの優先順位を調整する必要があります。各設定プログラムに関しては、製造業者のマニュアルを参照するか、またはブート中に表示される設定プログラムの指示を参照してください。

PXE 対応ネットワークアダプタの中には、ブート時にしばらく表示されるプロンプトに対して特定のキーを押すと、PXE ブートを実行する機能を持つものがあります。この機能は、PXE の設定を変更する必要がないので、通常はディスクドライブからブートを実行するシステムのインストールブートにおいて PXE を使用する場合に適しています。アダプタにこの機能がない場合は、システムのインストール後に行われるリブート時に、BIOS の設定で PXE を使用しないように設定してください。システムがディスクドライブからブートするようになります。

一部の初期バージョンの PXE ファームウェアでは、Solaris システムをブートすることができません。このようなファームウェアを使用する場合、システムがブートサーバーから PXE ネットワークブートストラッププログラムを読み込むことはできませんが、ブートストラップはパケットを転送しません。この問題が発生した場合は、アダプタの PXE ファームウェアをアップグレードしてください。ファームウェアのアップグレードに関する情報は、アダプタの製造業者の Web サイトから入手してください。詳細については、e1x1 (7D) および iprb (7D) のマニュアルページを参照してください。

ブートフロッピーディスクを使用するか使用しないかに関わらず、x86 システムのブートについては、190 ページの「x86: システムをネットワークからブートする方法」を参照してください。

シャットダウンとブートについての参照先

システムをシャットダウンおよびブートする手順については、次を参照してください。

シャットダウンとブート作業	参照先
SPARC システムまたは x86 システムのシャットダウン	第 9 章
SPARC システムのブート	第 10 章
x86 システムのブート	第 11 章
電源管理ソフトウェアによる SPARC システムの管理	power.conf (4) および pmconfig (1M) の各マニュアルページ

シャットダウンとブートの用語

この節では、シャットダウンとブートに関する用語について説明します。

- **実行レベルと init 状態** – 「実行レベル」とは、システムの状態を表す文字または数字のことで、どのシステムサービスを使用できるのかを示します。システムは常に定義済み実行レベルの 1 つで動作します。実行レベルを変更するために init プロセスが使用されるため、実行レベルは「init 状態」と呼ばれることもあります。システム管理者は、init (1M) コマンドを使用して、実行レベルを変更します。このマニュアルでは、init 状態を実行レベルと呼びます。

実行レベルの詳細については、139 ページの「**実行レベル**」を参照してください。

- **ブートタイプ** – 「ブートタイプ」とはシステムのブート方法を表します。次のようなブートタイプがあります。
 - **対話式ブート** – システムのブート方法に関する情報 (カーネルやデバイスのパス名など) を入力するプロンプトが表示される
 - **再構成用ブート** – システムが再構成され、新しく追加したハードウェアや新しい擬似デバイスがサポートされる
 - **回復ブート** – システムがハング状態になったとき、無効なエントリがあるためシステムが正常にブートできないとき、またはユーザーがログインできないときに使用する

システムのシャットダウンに関するガイドライン

システムをシャットダウンするときは、次の点に注意してください。

- システムのシャットダウンには、`init` および `shutdown` コマンドを使用します。これらのコマンドは、すべてのシステムプロセスとサービスを正常に終了させてからシャットダウンします。
- サーバーをシャットダウンする場合は、`shutdown` コマンドを使用してください。`shutdown` コマンドは、シャットダウンを実行する前に、サーバーにログインしているユーザーやサーバーのリソースをマウントしているシステムにシャットダウンを通知します。システムのシャットダウンについては、ユーザーが予定を立てられるようあらかじめ電子メールで知らせておくようにします。
- `shutdown` または `init` コマンドを使用してシステムをシャットダウンするには、スーパーユーザー権限が必要です。
- `shutdown` および `init` コマンドはどちらも実行レベルを引数に指定します。もっともよく使用される実行レベルは次の3つです。
 - 実行レベル 3 - すべてのシステムリソースを使用でき、ユーザーもログインできる状態。デフォルトでは、システムをブートすると実行レベル 3 になります。通常の運用で使用されます。NFS リソースを共有できるマルチユーザーレベルとも呼ばれます。
 - 実行レベル 6 - オペレーティングシステムを停止して、`/etc/inittab` ファイルの `initdefault` エントリに定義されている状態でリブートします。
 - 実行レベル 0 - オペレーティングシステムがシャットダウンされ、安全に電源切断できる状態。システムの設置場所を変更したり、ハードウェアを追加または削除する場合は、システムを実行レベル 0 にする必要があります。

実行レベルの詳細については、第 8 章を参照してください。

システムのブートに関するガイドライン

システムをブートするときは、次の点に注意してください。

- シャットダウン後にシステムをブートするには、SPARC システムの場合は、PROM レベルで `boot` コマンドを使用します。x86 システムの場合は、一次ブートサブシステムメニューで `boot` コマンドを使用します。
- 電源を切断した後に再投入すればシステムをリブートできます。ただし、この方法ではシステムサービスやプロセスが突然終了してしまうので、適切なシャットダウンとは言えません。緊急時のリブート以外には使用しないようにします。

- SPARC システムと x86 システムとでは、ブート時に使用するハードウェアが異なります。これらのハードウェアの違いについては、第 12 章を参照してください。

ネットワークからのシステムのブート

次のような場合に、システムをネットワークからブートする必要があります。

- システムを最初にインストールする場合
- システムをローカルディスクからブートできない場合
- システムがディスクレスクライアントである場合

さらに、次の 2 つのネットワーク構成ブート方法も利用できます。

- RARP (Reverse Address Resolution Protocol and ONC+ RPC Bootparams Protocol)
- DHCP (Dynamic Host Configuration Protocol)

デフォルトのネットワークブート方法は RARP に設定されています。

ネットワーク経由でシステムをブートする方法については、次の表を参考にしてください。

ネットワークブート作業	参照先
SPARC システムまたは SPARC ディスクレスクライアントのブート	第 10 章
x86 システムまたは x86 ディスクレスクライアントのブート	第 11 章
インストール時の DHCP クライアントのブート	『Solaris 9 9/04 インストールガイド』
DHCP マネージャを備えた DHCP クライアントの構成	『Solaris のシステム管理 (IP サービス)』

システムをシャットダウンする場合

次の表に、システム管理作業とそれに伴って必要となるシャットダウンの種類を示します。

表 7-1 システムのシャットダウン

システムシャットダウンの理由	適切な実行レベル	参照先
停電のためシステムの電源を切断する	実行レベル 0。安全に電源を切れる状態	第 9 章
/etc/system ファイル内のカーネルパラメータを変更する	実行レベル 6 (システムのリブート)	第 9 章
ファイルシステムを保守する (システムデータのバックアップや復元など)	実行レベル S (シングルユーザーレベル)	第 9 章
/etc/system などのシステム構成ファイルを修復する	136 ページの「システムをブートする場合」を参照	なし
システムにハードウェアを追加する (または、システムからハードウェアを削除する)	再構成用ブート (ハードウェアを追加または削除したら電源を切断する)	『Solaris のシステム管理 (デバイスとファイルシステム)』の「デバイスの管理 (手順)」
ブート失敗の原因となっていた重要なシステムファイルを修復する	136 ページの「システムをブートする場合」を参照	なし
カーネルデバッガ (kadb) をブートして、システムの障害を調査する	実行レベル 0 (可能な場合)	第 9 章
ハング状態から回復させ、クラッシュダンプを強制する	136 ページの「システムをブートする場合」を参照	なし

サーバーまたはスタンドアロンシステムのシャットダウンの例については、第 9 章を参照してください。

システムをブートする場合

次の表に、システム管理作業とそれに伴って必要となるブートタイプを示します。

表 7-2 システムのブート

システムリブートの理由	適切なブートタイプ	SPARC のブート手順の参照先	x86 のブート手順の参照先
停電のためシステムの電源を切断する	システムの電源を再投入する	第 9 章	第 9 章

表 7-2 システムのブート (続き)

システムリブートの理由	適切なブートタイプ	SPARC のブート手順の参照先	x86 のブート手順の参照先
/etc/system ファイル内のカーネルパラメータを変更する	システムを実行レベル 3 でリブートする (NFS リソースを共有できるマルチユーザーレベル)	172 ページの「SPARC: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法」	186 ページの「x86: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法」
ファイルシステムを保守する (システムデータのバックアップや復元など)	実行レベル S で Control + D を押し、システムを実行レベル 3 に戻す	173 ページの「SPARC: システムを実行レベル S (シングルユーザーレベル) でブートする方法」	187 ページの「x86: システムを実行レベル S (シングルユーザーレベル) でブートする方法」
/etc/system などのシステム構成ファイルを修復する	対話式ブート	174 ページの「SPARC: システムを対話式でブートする方法」	188 ページの「x86: システムを対話式でブートする方法」
システムにハードウェアを追加する (または、システムからハードウェアを削除する)	再構成ブート (ハードウェアを追加または削除したら電源を投入する)	『Solaris のシステム管理 (デバイスとファイルシステム)』の「システムディスクまたは二次ディスクの追加 (作業マップ)」	『Solaris のシステム管理 (デバイスとファイルシステム)』の「システムディスクまたは二次ディスクの追加 (作業マップ)」
カーネルデバッグ (kadb) をブートして、システムの障害を調査する	kabd をブートする	179 ページの「SPARC: カーネルデバッグ (kadb) を使ってシステムをブートする方法」	196 ページの「x86: カーネルデバッグ (kadb) を使ってシステムをブートする方法」
ブート失敗の原因となっていた重要なシステムファイルを修復する	回復ブート	177 ページの「SPARC: 復元を目的としてシステムをブートする方法」	191 ページの「x86: 復元を目的としてシステムをブートする方法」
ハング状態から回復させ、クラッシュダンプを強制する	回復ブート	180 ページの「SPARC: クラッシュダンプを強制してシステムをリブートする方法」の例を参照	197 ページの「x86: クラッシュダンプを強制してシステムをリブートする方法」の例を参照

システムブートの例については、第 10 章または第 11 章を参照してください。

第 8 章

実行レベルとブートファイル (手順)

この章では、実行レベルとブートファイルに関する概要と手順について説明します。

この章で説明する手順は次のとおりです。

- 149 ページの「実行制御スクリプトを使用してサービスを起動または停止する方法」
- 150 ページの「実行制御スクリプトを追加する方法」
- 151 ページの「実行制御スクリプトを無効にする方法」

この章の内容は次のとおりです。

- 139 ページの「実行レベル」
- 141 ページの「/etc/inittab ファイル」
- 144 ページの「実行制御スクリプト」
- 152 ページの「x86: ブートファイル」

実行レベル

システムの「実行レベル」(「init 状態」とも呼ばれる)は、ユーザーが使用できるサービスとリソースを定義します。システムが一度に持つことのできる実行レベルは 1 つだけです。

Solaris 環境には 8 つの実行レベルがあります (次の表を参照)。デフォルトの実行レベル 3 は、/etc/inittab ファイルに指定されています。

表 8-1 Solaris 実行レベル

実行レベル	init 状態	種類	目的
0	電源切断状態	電源切断	オペレーティングシステムをシャットダウンしてシステムの電源を安全に落とせるようにする。
s または S	シングルユーザー状態	シングルユーザー	一部のファイルシステムがマウントされ使用可能な状態で、シングルユーザーとして動作する。
1	システム管理状態	シングルユーザー	すべての使用可能なファイルシステムにアクセスする。ユーザーロゲインは使用不可である。
2	マルチユーザー状態	マルチユーザー	通常の運用に使用する。複数のユーザーがシステムとすべてのファイルシステムにアクセスできる。NFS サーバーデーモンを除く、すべてのデーモンが動作する。
3	NFS リソースを共有したマルチユーザーレベル	マルチユーザー	NFS リソースを共有する通常の運用に使用する。Solaris 環境におけるデフォルトの実行レベル。
4	マルチユーザー状態 (予備)		現在は使用できない。
5	電源切断状態	電源切断	オペレーティングシステムをシャットダウンしてシステムの電源を安全に落とせるようにする。可能であれば、この機能をサポートしているシステムでは電源を自動的に切断する。
6	リポート状態	リポート	システムをシャットダウンして実行レベル 0 にした後、NFS リソースを共有するマルチユーザーレベル (または、 <code>inittab</code> ファイルに指定されたデフォルトのレベル) でリポートする。

システムの実行レベルを確認する方法

`who -r` コマンドを使用すると、実行レベルに関する情報が表示されます。

```
$ who -r
```

`who -r` コマンドを使用して、システムの現在の実行レベルを調べます (ただし、実行レベル 0 を除く)。

例 — システムの実行レベルを確認する

次の例では、システムの現在の実行レベルと以前の実行レベルに関する情報を表示します。

```
$ who -r
.      run-level 31  Dec 13 10:102  33  04 s5
$
```

1. 現在の実行レベル
2. 実行レベルが最後に変更された日時
3. 現在の実行レベル
4. 最後にリブートしてからシステムがこの実行レベルになった回数
5. 以前の実行レベル

/etc/inittab ファイル

init または shutdown コマンドを使用してシステムをブートしたり実行レベルを変更したりすると、init デーモンは、/etc/inittab ファイルから情報を読み取ってプロセスを起動します。/etc/inittab ファイルには、init プロセスにとって重要な3つの情報が定義されています。

- システムのデフォルトの実行レベル
- 起動、監視するプロセス、および停止時に再起動するプロセス
- システムが新しい実行レベルに移行したとき行う処理

/etc/inittab ファイル内の各エントリは、次のフィールドからなります。

id:*rstate*:*action*:*process*

次の表に、inittab エントリの各フィールドを要約します。

表 8-2 inittab ファイルのフィールドの説明

フィールド	説明
<i>id</i>	エントリに固有の (一意の) 識別子。
<i>rstate</i>	このエントリが適用される実行レベルのリスト。
<i>action</i>	プロセスフィールドに指定されたプロセスの実行方法。指定できる値は、initdefault、sysinit、boot、bootwait、wait、respawn。 initdefault はデフォルトの実行レベル。他の action キーワードについては、inittab(4) のマニュアルページを参照。

表 8-2 inittab ファイルのフィールドの説明 (続き)

フィールド	説明
<i>process</i>	実行するコマンドまたはスクリプトを定義する。

例 — デフォルトの inittab ファイル

次の例では、Solaris リリースでインストールされるデフォルトの inittab ファイルを注釈付きで示します。

```

1 ap::sysinit:/sbin/autopush -f /etc/iu.ap
2 ap::sysinit:/sbin/soconfig -f /etc/sock2path
3 fs::sysinit:/sbin/rcS sysinit >/dev/msglog 2<>/dev/msglog </dev/console
4 is:3:initdefault:
5 p3:s1234:powerfail:/usr/sbin/shutdown -y -i5 -g0 >/dev/msglog 2<>/dev/...
6 sS:s:wait:/sbin/rcS >/dev/msglog 2<>/dev/msglog </dev/console
7 s0:0:wait:/sbin/rc0 >/dev/msglog 2<>/dev/msglog </dev/console
8 s1:1:respawn:/sbin/rc1 >/dev/msglog 2<>/dev/msglog </dev/console
9 s2:23:wait:/sbin/rc2 >/dev/msglog 2<>/dev/msglog </dev/console
10 s3:3:wait:/sbin/rc3 >/dev/msglog 2<>/dev/msglog </dev/console
11 s5:5:wait:/sbin/rc5 >/dev/msglog 2<>/dev/msglog </dev/console
12 s6:6:wait:/sbin/rc6 >/dev/msglog 2<>/dev/msglog </dev/console
13 fw:0:wait:/sbin/uadmin 2 0 >/dev/msglog 2<>/dev/msglog </dev/console
14 of:5:wait:/sbin/uadmin 2 6 >/dev/msglog 2<>/dev/msglog </dev/console
15 rb:6:wait:/sbin/uadmin 2 1 >/dev/msglog 2<>/dev/msglog </dev/console
16 sc:234:respawn:/usr/lib/saf/sac -t 300
17 co:234:respawn:/usr/lib/saf/ttymon -g -h -p "`uname -n` console login: "
    -T terminal-type -d /dev/console -l console
-m ldterm,ttcompat

```

1. STREAMS モジュールを初期化します。
2. ソケット転送プロバイダを構成します。
3. ファイルシステムを初期化します。
4. デフォルトの実行レベルを定義します。
5. 電源障害の場合のシャットダウンを指定します。
6. シングルユーザーレベルを定義します。
7. 実行レベル 0 を定義します。
8. 実行レベル 1 を定義します。
9. 実行レベル 2 を定義します。
10. 実行レベル 3 を定義します。
11. 実行レベル 5 を定義します。
12. 実行レベル 6 を定義します。
13. 未使用レベル *firmware* を定義します。
14. 未使用レベル *off* を定義します。
15. 未使用レベル *reboot* を定義します。
16. サービスアクセスコントローラを初期化します。
17. コンソールを初期化し、端末のタイプを識別します。

システムが実行レベル 3 になると実行される処理

1. `init` プロセスが起動されます。`init` プロセスは、`/etc/default/init` ファイルを読み取って環境変数を設定します。デフォルトでは、`TIMEZONE` 変数だけが設定されます。
2. `init` は `inittab` ファイルを読み取り、次の処理を行います。
 - a. デフォルトの実行レベル 3 を定義する `initdefault` エントリを識別します。
 - b. `action` フィールドが `sysinit` になっているすべてのプロセスエントリを実行して、ユーザーがログインする前に特別な初期設定処理がすべて行われるようにします。
 - c. `rstate` フィールドが 3 になっている (デフォルトの実行レベル 3 と一致する) プロセスエントリを実行します。

`init` プロセスが `inittab` ファイルを使用する方法の詳細については、`init` (1M) のマニュアルページを参照してください。

次の表に、実行レベル 3 の `action` フィールドに使用するキーワードの説明を示します。

表 8-3 実行レベル 3 の `action` キーワードの説明

キーワード	説明
<code>powerfail</code>	<code>init</code> プロセスが電源切断シグナルを受信したときにだけプロセスを起動する。
<code>respawn</code>	プロセスを起動し、そのプロセスが終了したら再起動する。
<code>wait</code>	プロセスを起動し、そのプロセスが終了するまで待ってから、この実行レベルの次のエントリに進む。

次の表に、実行レベル 3 で実行されるプロセス (またはコマンド) の説明を示します。

表 8-4 実行レベル 3 のコマンドの説明

コマンドまたはスクリプト名	説明
<code>/usr/sbin/shutdown</code>	システムをシャットダウンする。 <code>init</code> プロセスは、システムが電源切断シグナルを受信した場合にのみ <code>shutdown</code> コマンドを実行する
<code>/sbin/rcS</code>	ルート (<code>/</code>)、 <code>/usr</code> 、 <code>/tmp</code> 、 <code>/var</code> 、 <code>/var/adm</code> 、および <code>/var/run</code> ファイルシステムをチェックし、マウントする
<code>/sbin/rc2</code>	標準のシステムプロセスを起動して、システムを実行レベル 2 (マルチユーザーレベル) に移行する

表 8-4 実行レベル 3 のコマンドの説明 (続き)

コマンドまたはスクリプト名	説明
/sbin/rc3	実行レベル 3 で使用される NFS リソース共有を開始する
/usr/lib/saf/sac -t 30	ポートモニターを起動する。このプロセスは失敗すると再起動される
/usr/lib/saf/ttymon -g -h -p " `uname -n` console login: " -T <i>terminal_type</i> -d /dev/console -l console	コンソールでのログイン要求を監視する ttymon プロセスを起動する。このプロセスは失敗すると再起動される
	SPARC システムの <i>terminal_type</i> は sun である
	x86 システムの <i>terminal_type</i> は AT386 である

実行制御スクリプト

注 - Solaris 環境でシステムサービスを起動および停止する方法は、将来のリリースで変更される可能性があります。

Solaris ソフトウェア環境では、一連の詳細な実行制御 (rc) スクリプトを使用して実行レベルの移行を制御しています。各実行レベルには次の rc スクリプトが対応しています。このスクリプトは、/sbin ディレクトリにあります。

- rc0
- rc1
- rc2
- rc3
- rc5
- rc6
- rcS

/sbin ディレクトリ内の各 rc スクリプトには /etc/rcn.d という名前のディレクトリが対応しており、その中にはその実行レベルのさまざまな処理を実行するスクリプトがあります。たとえば、/etc/rc2.d には、実行レベル 2 のプロセスを起動および停止するためのファイル (スクリプト) があります。

```
# ls /etc/rc2.d
K03samba*      S05RMTMPFILES*  S72inetsvc*    S88utmpd*
K03sshd*      S10lu*          S72slpd*       S89PRESERVE*
K05appserv*   S20syssetup*    S73cachefs.daemon* S89bdconfig@
K05volmgt*    S21perf*        S73nfs.client* S90wbem*
K06mipagent*  S30sysid.net*   S74autofs*     S91zuluinit*
```


K07dmi*	S401lc2*	S74syslog*	S93cacheos.finish*
K07snmpdx*	S42ncakmod*	S74xntpd*	S94ncalogd*
K15imq*	S47pppd*	S75cron*	S95IIim*
K16apache*	S69inet*	S75flashprom*	S95svm.sync*
K21dhcp*	S70sckm*	S75savecore*	S98efcode*
K27boot.server*	S70uucp*	S76nsd*	S99audit*
K28kdc*	S71ldap.client*	S77sf880dr*	S99dtlogin*
K28kdc.master*	S71rpc*	S80lp*	S99emul64*
K28nfs.server*	S71sysid.sys*	S80spc*	S99rcapd*
README	S72autoinstall*	S85power*	
S01MOUNTFSYS*	S72directory*	S88sendmail*	

/etc/rcn.d 内のスクリプトは常に、スクリプト名を ASCII 文字列としてソートした順に実行されます。スクリプト名の形式は次のとおりです。

[KS] [0-9] [0-9] *

名前が **K** で始まるスクリプトを実行すると、システムサービスが停止 (kill) します。名前が **S** で始まるスクリプトを実行すると、システムサービスが起動します。

実行制御スクリプトは、/etc/init.d ディレクトリにもあります。これらのファイルは、/etc/rcn.d ディレクトリ内の対応する実行制御スクリプトにリンクされています。

各実行制御スクリプトの処理については、次の節で要約します。

実行制御スクリプトのまとめ

以降の節に、実行レベルを変更したときにシステムサービスを起動および停止するための実行制御スクリプトをまとめてあります。

/sbin/rc0 スクリプト

/sbin/rc0 スクリプトは、/etc/rc0.d スクリプトを実行して次の作業を行います。

- システムサービスとデーモンの終了
- 実行中の全プロセスの停止
- 全ファイルシステムのマウント解除

/sbin/rc1 スクリプト

/sbin/rc1 スクリプトは、/etc/rc1.d スクリプトを実行して次の作業を行います。

- システムサービスとデーモンの終了
- 実行中のすべてのユーザープロセスの停止

- すべてのリモートファイルシステムのマウント解除
- 以前の実行レベルが S であった場合、すべてのローカルファイルシステムのマウント

/sbin/rc2 スクリプト

/sbin/rc2 スクリプトは、/etc/rc2.d スクリプトを実行して、機能別にグループ化された次の作業を行います。

ローカルシステム関連の作業

- 以前の実行レベルが S であった場合、すべてのローカルファイルシステムのマウント
- quota オプションを指定してマウントされたファイルシステムが 1 つでも存在する場合は、ディスク割り当てを有効にする
- エディタの一時ファイルを /usr/preserve ディレクトリに保存する
- /tmp ディレクトリ内のすべてのファイルおよびサブディレクトリを削除する
- 設定されている場合、システムアクティビティデータの収集、システムアカウント、システム監査を起動する
- システムロギングデーモン (syslogd) を起動し、デフォルトのダンプデバイスを設定し、/var/adm/messages ファイルをローテーションする
- /etc/dispadmin.conf ファイルが存在する場合は、デフォルトのスケジューリングクラスを設定する
- ローカルプリンタが設定され、印刷待ち行列がクリーンアップされている場合は、LP 印刷サービス (lpsched) を起動する
- 必要に応じて、電源管理を設定する
- utmpd デーモンを起動する
- cron および vold デーモンを起動する
- シリアルデバイスストリームを設定する
- WBEM サービスを設定する
- 必要に応じてボリュームを同期化し、mdmonitor デーモンを起動してボリュームの物理コンポーネントを監視する
- 必要に応じて、CDE デスクトップログインプロセス dtlogin を起動する

ネットワークサービスまたはセキュリティ関連の作業

- 必要に応じて、ネットワークインタフェースを構成し、ifconfig ネットマスクを設定し、ネットワークルーティングを設定する
- ネットワークサービス (inetd および rpcbind) デーモンを起動する
- 論理リンクコントローラ (llc2) が設定されている場合は起動する
- システムがネームサービス用に設定されているかどうか、およびシステムがクライアントまたはサーバーのどちらであるかによって、ネームサービスのドメイン名を設定し、各種ネームサービスデーモンを起動する

- 必要に応じて、keyserv、statd、lockd、xntpd の各デーモンを起動する
- すべての NFS エントリをマウントする
- 必要に応じて、Solaris NCA (Network Cache and Accelerator) および NCA ログインを設定する
- Solaris PPP サーバーまたはクライアントデーモン (pppoed または pppd) が設定されている場合は起動する
- LDAP キャッシュマネージャ (ldap_cachemgr) が設定されている場合は起動する
- ディレクトリサーバー (slapd) デーモンが設定されている場合は起動する
- DNS (in.named) デーモンが設定されている場合は起動する
- サービスロケーションプロトコル (slpd) デーモンが設定されている場合は起動する
- /etc/rctladm.conf および /etc/pooladm.conf ファイルが存在する場合は、システムリソース制御とシステムプールを設定する
- 必要に応じて、cachefsd、automount、sendmail の各デーモンを起動する
- http_server プロセスを起動する

インストール関連の作業

- システム起動時またはシステム停止時の Live Upgrade ソフトウェアのブート環境を設定する
- /etc/.UNCONFIGURE ファイルがあるかどうかをチェックして、システムを設定し直す必要があるかどうか調べる
- /.PREINSTALL または /AUTOINSTALL のどちらかが存在する場合、インストールメディアまたはブートサーバーからシステムをリブートする

ハードウェア関連の作業

- 必要に応じて、Sun Fire 15000 のキー管理デーモン (sckmd) を起動する
- 必要に応じて、Sun Fire 880 の動的再構成デーモン (sf880drd) を起動する
- フラッシュ PROM の更新スクリプトを実行する
- グラフィックフレームバッファまたはグラフィックアクセラレータを設定する
- 必要に応じて、FCode インタプリタデーモン (efdaemon) を実行する

実行レベルを変更したときは、次のサービスを移行します。

- Apache (tomcat)
- ブートサーバー (in.rarpd、rpc.bootparamd、または rpld)
- DHCP (in.dhcpd)
- Kerberos KDC (krb5kdc) および Kerberos 管理 (kadmind)
- Mobile IP (mipagent)
- NFS サーバー (nfsd、mountd、nfslogd)
- Samba (smbd および nmbd)
- ssh (secure shell) (sshd)
- Solstice Enterprise Agents (dmispd および snmpXdmid)

注 - 実行レベル 2 で起動されるシステムサービスとアプリケーションの多くは、システム上にインストールされているソフトウェアによって決まります。

/sbin/rc3 スクリプト

/sbin/rc3 スクリプトは、/etc/rc3.d スクリプトを実行して次の作業を行います。

- Apache サーバーデーモン (tomcat) が設定されている場合は起動する
- 必要に応じて、DHCP デーモン (in.dhcpd) を起動する
- Kerberos KDC (krb5kdc) および Kerberos 管理 (kadmind) デーモンが設定されている場合は起動する
- Mobile IP デーモン (mipagent) が設定されている場合は起動する
- Samba デーモン (smbd および nmbd) が設定されている場合は起動する
- 必要に応じて、ssh (secure shell) デーモン (sshd) を起動する
- Solstice Enterprise Agents (dmispd および snmpXdmid) を起動する
- /etc/dfs/sharetab ファイルをクリーンアップする
- 必要に応じて、NFS サーバーデーモン nfsd、mountd、および nfslogd を起動する
- ブートサーバーの場合は、rarpd、rpc.bootparamd、rpld デーモンを起動する

/sbin/rc5 および /sbin/rc6 スクリプト

/sbin/rc5 および /sbin/rc6 スクリプトは、/etc/rc0.d/K* スクリプトを実行して、次の作業を行います。

- すべてのアクティブなプロセスを停止する
- ファイルシステムのマウント解除

/sbin/rcS スクリプト

/sbin/rcS スクリプトは、/etc/rcS.d スクリプトを実行して、システムを実行レベル S に移行します。これらのスクリプトによって次の作業が行われます。

- 最小限のネットワークの確立
- ルート (/)、/usr、/tmp、/var、/var/adm、および /var/run ファイルシステムのチェックとマウント
- システム名の設定
- 擬似ファイルシステム (/proc と /dev/fd) のマウント

- 再構成用ブートのデバイスエントリの再構築
- シングルユーザーレベルでマウントされる他のファイルシステムのチェックとマウント

実行制御スクリプトを使用してサービスを起動または停止する

注 - Solaris 環境でシステムサービスを起動および停止する方法は、将来のリリースで変更される可能性があります。

実行レベルごとに対応するスクリプトを持つことの利点は、`/etc/init.d` ディレクトリ内の個々のスクリプトを実行することによって、システムの実行レベルを変更しないで (現在の実行レベルの) システムサービスを停止できる点です。

▼ 実行制御スクリプトを使用してサービスを起動または停止する方法

1. スーパーユーザーになります。
2. システムサービスを停止します。

```
# /etc/init.d/filename stop
```
3. システムサービスを再開します。

```
# /etc/init.d/filename start
```
4. サービスが停止または起動していることを確認します。

```
# pgrep -f service
```

例 — 実行制御スクリプトを使用してサービスを起動または停止する

NFS サーバーデーモンを停止するには、次のように入力します。

```
# /etc/init.d/nfs.server stop
# pgrep -f nfs
#
```

そして、NFS サーバーデーモンを再開するには、次のように入力します。

```

# /etc/init.d/nfs.server start
# pgrep -f nfs
341
343
347
345
# pgrep -f nfs -d, | xargs ps -fp
UID    PID  PPID  C   STIME TTY      TIME CMD
daemon 341   1    0   Aug 21 ?        0:00 /usr/lib/nfs/statd
root   343   1    0   Aug 21 ?        0:00 /usr/lib/nfs/lockd
root   347   1    0   Aug 21 ?        0:41 /usr/lib/nfs/nfsd
root   345   1    0   Aug 21 ?        0:02 /usr/lib/nfs/mountd

```

実行制御スクリプトを追加する

注 - Solaris 環境でシステムサービスを起動および停止する方法は、将来のリリースで変更される可能性があります。

サービスを起動または停止するための実行制御スクリプトを追加するには、そのスクリプトを `/etc/init.d` ディレクトリにコピーします。次に、サービスを起動または停止する `rcn.d` ディレクトリ内にリンクを作成します。

実行制御スクリプトの命名法の詳細については、`/etc/rcn.d` ディレクトリ内の `README` ファイルを参照してください。次に、実行制御スクリプトを追加する方法について説明します。

▼ 実行制御スクリプトを追加する方法

1. スーパーユーザーになります。
2. スクリプトを `/etc/init.d` ディレクトリにコピーします。

```

# cp filename /etc/init.d
# chmod 0744 /etc/init.d/filename
# chown root:sys /etc/init.d/filename

```

3. 適切な `rcn.d` ディレクトリへのリンクを作成します。

```

# cd /etc/init.d
# ln filename /etc/rc2.d/Snnfilename
# ln filename /etc/rcn.d/Knnfilename

```

4. スクリプトのリンクが指定のディレクトリ内にあることを確認します。

```

# ls /etc/init.d/ /etc/rc2.d/ /etc/rcn.d/

```

例 — 実行制御スクリプトを追加する

次の例は、xyz サービスの実行制御スクリプトを追加する方法を示しています。

```
# cp xyz /etc/init.d
# chmod 0744 /etc/init.d/xyz
# chown root:sys /etc/init.d/xyz
# cd /etc/init.d
# ln xyz /etc/rc2.d/S100xyz
# ln xyz /etc/rc0.d/K100xyz
# ls /etc/init.d /etc/rc2.d /etc/rc0.d
```

実行制御スクリプトを無効にする

実行制御スクリプトを無効にするには、ファイル名の先頭に下線 () を付けてファイル名を変更します。下線またはドット (.) で始まるファイルは実行されません。接尾辞を追加してファイルをコピーすると、両方のファイルが実行されます。

▼ 実行制御スクリプトを無効にする方法

1. スーパーユーザーになります。
2. スクリプト名の先頭に下線 () を付けて、スクリプト名を変更します。

```
# cd /etc/rcn.d
# mv filename _filename
```

3. スクリプト名が変更されたことを確認します。

```
# ls _*
# _filename
```

例 — 実行制御スクリプトを無効にする

次の例は、S100datainit スクリプトの名前を変更する方法を示しています。

```
# cd /etc/rc2.d
# mv S100datainit _S100datainit
# ls _*
# _S100datainit
```

x86: ブートファイル

これまでに説明した実行制御スクリプトおよびブートファイル以外にも、Solaris x86 システムのブートに使用する追加ブートファイルがあります。

表 8-5 x86: ブートファイル

ファイル	説明
/etc/bootrc	Solaris リリースのブートに使用するメニューとオプションを含む
/boot	システムをブートするために必要なファイルとディレクトリを含む
/boot/mdboot	第 1 レベルのブートストラッププログラム (strap.com) をディスクからメモリーにロードする DOS 実行可能ファイル
/boot/mdbootbp	第 1 レベルのブートストラッププログラム (strap.com) をフロッピーディスクからメモリーにロードする DOS 実行可能ファイル
/boot/rc.d	インストールスクリプトを含むディレクトリ。内容の変更はできない
/boot/solaris	ブートサブシステムの項目を含むディレクトリ
/boot/solaris/boot.bin	Solaris カーネルまたはスタンドアロンの kadb をロードする。この実行可能ファイルは、ブートファームウェアサービスも提供する
/boot/solaris/boot.rc	Solaris x86 版を識別し、DOS エミュレーションモードで Device Configuration Assistant (デバイス構成用補助) を実行する
/boot/solaris/bootconf.exe	Device Configuration Assistant (デバイス構成用補助) の DOS 実行可能ファイル
/boot/solaris/bootconf.txt	Device Configuration Assistant (デバイス構成用補助) の国際化されたメッセージを含むテキストファイル (bootconf.exe)
/boot/solaris/bootenv.rc	ブート環境の設定用の eeprom 変数を格納する
/boot/solaris/devicedb	master ファイル (リアルモードドライバでサポートされるすべての有効なデバイスのデータベース) を含むディレクトリ
/boot/solaris/drivers	リアルモードドライバを含むディレクトリ

表 8-5 x86: ブートファイル (続き)

ファイル	説明
/boot/solaris/itup2.exe	インストール時更新 (ITU) プロセスで実行される DOS 実行可能ファイル
/boot/solaris/machines	以前は使用されていたが、現在はもう使用されていないディレクトリ
/boot/solaris/nbp	ネットワークブート関連ファイル
/boot/solaris/strap.rc	どのロードモジュールをロードするか、メモリー内のどこにロードするかの指示を含むファイル
/boot/strap.com	第 2 レベルのブートストラッププログラムをメモリーにロードする DOS 実行可能ファイル

第 9 章

システムのシャットダウン (手順)

この章では、システムのシャットダウン手順について説明します。この章で説明する手順は次のとおりです。

- 157 ページの「システムにログインしているユーザーを知る方法」
- 157 ページの「サーバーをシャットダウンする方法」
- 161 ページの「スタンドアロンシステムをシャットダウンする方法」
- 163 ページの「すべてのデバイスの電源を落とす方法」

この章の内容は次のとおりです。

- 156 ページの「システムシャットダウンコマンド」
- 157 ページの「システムのダウン時間をユーザーに通知する」
- 162 ページの「すべてのデバイスの電源を落とす」

システムの実行レベルについての概要は、第 8 章を参照してください。

システムのシャットダウン

Solaris ソフトウェアは、電子メールやネットワークソフトウェアをいつでも利用できるように、停止することなく動作するように設計されています。しかし、システム管理作業を行う場合や緊急事態が発生した場合は、システムをシャットダウンして安全に電源を切断できる状態にする必要があります。次のような場合には、システムを一部のシステムサービスしか利用できない中間の実行レベルまで移行する必要があります。

- ハードウェアを追加または削除する
- 予定された停電に備える
- ファイルシステムの保守を行う (バックアップなど)

システムをシャットダウンする必要があるシステム管理作業の詳細については、第 7 章を参照してください。

システムの電源管理機能を使用する方法については、『Solaris 共通デスクトップ環境 ユーザーズ・ガイド』を参照してください。

システムシャットダウンコマンド

システムをシャットダウンするもっとも基本的な方法は、`init` コマンドおよび `shutdown` コマンドを使用する方法です。どちらのコマンドもシステムを「クリーンな状態でシャットダウン」するため、すべてのファイルシステムに対する変更がディスクに書き出され、すべてのシステムサービス、プロセス、オペレーティングシステムが正常に終了します。

システムのアポルトキーシーケンスを使用したり、電源をオフにしてからオンにする方法では、システムサービスが突然終了してしまうので、クリーンなシャットダウン方法とはいえません。しかし、緊急時には、これらの方法を使用しなければならない場合もあります。システムの復元方法については、[第 10 章](#)または[第 11 章](#)を参照してください。

次の表に、いくつかのシャットダウンコマンドとその用途を要約します。

表 9-1 シャットダウンコマンド

コマンド	説明	用途
<code>shutdown</code>	<code>init</code> を呼び出してシステムをシャットダウンする実行可能なシェルスクリプト。デフォルトでは、システムは実行レベル S に移行する	実行レベル 3 で動作しているサーバーで使用する。サーバーにログインしているユーザーに、サーバーが間もなくシャットダウンされることが通知される。シャットダウンされるサーバーのリソースをマウントしているシステムにも通知される
<code>init</code>	すべてのアクティブなプロセスを終了し、ディスクを同期させてから実行レベルを変更する実行可能ファイル	他のユーザーが影響を受けないスタンダアロンシステムで使用する。ユーザーはまもなく行われるシャットダウンについて通知されないので、シャットダウンにかかる時間が短くて済む
<code>reboot</code>	ディスクを同期させ、ブート命令を <code>uadmin</code> システムコールに渡す実行可能ファイル。実際にプロセッサを停止するのは、 <code>uadmin</code> システムコールである	推奨されない。代わりに、 <code>init</code> コマンドを使用する
<code>halt</code>	ディスクを同期させ、プロセッサを停止する実行可能ファイル	<code>/etc/rc0</code> スクリプトを実行しないので推奨されない。このスクリプトは、すべてのプロセスを停止し、ディスクを同期させ、すべてのファイルシステムをマウント解除する

システムのダウン時間をユーザーに通知する

shutdown コマンドは起動時に、ログインしているすべてのユーザーと、システムリソースをマウントしているすべてのシステムに、警告と最終メッセージという形でシャットダウンを通知します。

サーバーをシャットダウンする必要がある場合に、init コマンドではなく shutdown コマンドを使用することを推奨するのはこのためです。どちらを使用するにしても、ユーザーには予定されているシャットダウンについてあらかじめ電子メールで知らせるようにしてください。

システム上のどのユーザーに通知する必要があるかを確認するには、who(1) コマンドを使用します。このコマンドは、システムの現在の実行レベルを調べる場合にも役立ちます。140 ページの「システムの実行レベルを確認する方法」を参照してください。

▼ システムにログインしているユーザーを知る方法

1. シャットダウンするシステムにログインします。
2. ログイン中のユーザーを表示します。

```
$ who
```

例 — システムにログインしているユーザーを知る

次の例は、システムにログインしているユーザーを表示する方法を示しています。

```
$ who
holly 1      console      May  7 07:30
kryten     pts/0 2      May  7 07:35      (starbug) 4
lister     pts/1      May  7 07:40 3 (bluemidget)
```

1. ログインしているユーザーのユーザー名
2. ログインしているユーザーの端末回線
3. ユーザーがログインした日時
4. (省略可能) リモートシステムからログインしているユーザーのホスト名

▼ サーバーをシャットダウンする方法

サーバーをシャットダウンする必要がある場合は、次の手順に従います。

1. スーパーユーザーになります。
2. システムにユーザーがログインしているか調べます。

```
# who
```

ログインしているすべてのユーザーがリスト表示されます。システムがシャットダウンされることを、メールかブロードキャストメッセージで知らせることをお勧めします。

3. システムをシャットダウンします。

```
# shutdown -iinit-level -ggrace-period -y
```

-iinit-level	システムをデフォルトのS以外のinitレベルにする。0、1、2、5、6のいずれかを指定できる
-ggrace-period	システムがシャットダウンするまでの時間(秒)を示す。デフォルト値は60秒。
-y	ユーザーの介入なしにシャットダウンを継続する。このオプションを指定しないと、シャットダウンを継続するかどうか60秒後にたずねられる

詳細については、shutdown(1M)のマニュアルページを参照してください。

4. シャットダウンを継続するかどうかたずねられたら、yを入力します。

```
Do you want to continue? (y or n): y
```

-y オプションを指定した場合、このプロンプトは表示されません。

5. プロンプトが表示されたら、スーパーユーザーのパスワードを入力します。

```
Type Ctrl-d to proceed with normal startup,  
(or give root password for system maintenance): xxx
```

6. システム管理作業を終了したら、Control + Dを押してデフォルトの実行レベルに戻ります。

7. 次の表を使用して、システムがshutdownコマンドで指定した実行レベルに移行したことを確認します。

指定した実行レベル	SPARC システムのプロンプト	x86 システムのプロンプト
S (シングルユーザーレベル)	#	#
0 (電源切断レベル)	ok または >	type any key to continue
実行レベル3 (リモートリソースを共有できるマルチユーザーレベル)	hostname console login:	hostname console login:

SPARC: 例 — サーバーを実行レベル S にする

次の例では、shutdown コマンドを使用して、3 分後に、SPARC システムを実行レベル S (シングルユーザーレベル) にしています。

```
# who
root      console      Dec 13 14:30
# shutdown -g180 -y

Shutdown started.      Thu Dec 13 14:30:32 MST 2001

Broadcast Message from root (console) on earth Thu Dec 13 14:30:33...
The system earth will be shut down in 3 minutes
.
.
Broadcast Message from root (console) on earth Thu Dec 13 14:30:33...
The system earth will be shut down in 30 seconds
.
.
INIT: New run level: S
The system is coming down for administration.  Please wait.
Unmounting remote filesystems: /vol nfs done.
Shutting down Solaris Management Console server on port 898.
Print services stopped.
Dec 13 14:34:00 earth syslogd: going down on signal 15
Killing user processes: done.

INIT: SINGLE USER MODE

Type control-d to proceed with normal startup,
(or give root password for system maintenance): xxxx
Entering System Maintenance Mode ...
#
```

SPARC: 例 — サーバーを実行レベル 0 にする

次の例では、shutdown コマンドを使用して、SPARC システムを 5 分後に実行レベル 0 にしています。確認用プロンプトが表示されないように -y オプションを指定しています。

```
# who
root      console      Dec 12 08:08
rimmer    pts/0          Dec 11 14:48      (starbug)
pmorph    pts/1          Dec 13 12:31      (bluemidget)
# shutdown -i0 -g300 -y
Shutdown started.      Thu Dec 13 14:51:39 MST 2001

Broadcast Message from root (console) on earth Thu Dec 13 14:51:39...
The system earth will be shut down in 5 minutes
.
.
```

```

.
Changing to init state 0 - please wait
#
INIT: New run level: 0
The system is coming down. Please wait.
System services are now being stopped.
.
.
.
The system is down.
syncing file systems... done
Program terminated
Type help for more information
ok

```

システムを実行レベル 0 にしてすべてのデバイスの電源を落とす場合は、163 ページの「すべてのデバイスの電源を落とす方法」を参照してください。

SPARC: 例 — サーバーをリブートして実行レベル 3 にする

次の例では、`shutdown` コマンドを使用して SPARC システムをリブートし、2 分後に実行レベル 3 にしています。確認用プロンプトが表示されないように `-y` オプションを指定しています。

```

# who
root      console      Dec 12 08:08
rimmer    pts/0          Dec 11 14:48   (starbug)
pmorph    pts/1          Dec 13 12:31   (bluemidget)
# shutdown -i6 -g120 -y
Shutdown started.   Thu Dec 13 15:56:30

Broadcast Message from root (console) on earth Thu Dec 13 15:56:30...
The system earth will be shut down in 2 minutes
.
.
.
Changing to init state 6 - please wait
#
INIT: New run level: 6
The system is coming down. Please wait.
.
.
.
The system is down.
syncing file systems... done
rebooting...
.
.
.
earth console login:

```


次に進む手順

システムをシャットダウンした理由が何であれ、最終的には、すべてのファイルリソースが使用でき、ユーザーがログインできる実行レベル 3 に戻すことになるでしょう。システムをマルチユーザーレベルに戻す手順については、[第 10 章](#)または[第 11 章](#)を参照してください。

▼ スタンドアロンシステムをシャットダウンする方法

スタンドアロンシステムをシャットダウンする必要がある場合は、次の手順を実行してください。

1. スーパーユーザーになります。
2. システムをシャットダウンします。

```
# init run-level
```

run-level は新しい実行レベルを指定します。

詳細については、`init(1M)` のマニュアルページを参照してください。

3. 次の表を使用して、システムが `init` コマンドで指定した実行レベルに移行したことを確認します。

指定した実行レベル	SPARC システムのプロンプト	x86 システムのプロンプト
S (シングルユーザーレベル)	#	#
2 (マルチユーザーレベル)	#	#
0 (電源切断レベル)	ok または >	type any key to continue
3 (NFS リソースを共有できる マルチユーザーレベル)	hostname console login:	hostname console login:

x86: 例 — スタンドアロンシステムを実行レベル 0 にする

次の例では、`init` コマンドを使用して、スタンドアロンの x86 システムを安全に電源を落とせるレベルにします。

```
# init 0
#
INIT: New run level: 0
The system is coming down. Please wait.
.
```

```
.  
.  
The system is down.  
syncing file systems... [11] [10] [3] done  
Type any key to continue
```

システムを実行レベル 0 にしてすべてのデバイスの電源を落とす場合は、[163 ページ](#)の「すべてのデバイスの電源を落とす方法」を参照してください。

SPARC: 例 — スタンドアロンシステムを実行レベル S にする

次の例では、`init` コマンドを使用して、スタンドアロンの SPARC システムを実行レベル S (シングルユーザーレベル) にしています。

```
# init s  
#  
INIT: New run level: S  
The system is coming down for administration. Please wait.  
Unmounting remote filesystems: /vol nfs done.  
Print services stopped.  
syslogd: going down on signal 15  
Killing user processes: done.  
INIT: SINGLE USER MODE  
  
Type Ctrl-d to proceed with normal startup,  
(or give root password for system maintenance): xxx  
Entering System Maintenance Mode  
#
```

次に進む手順

システムをシャットダウンした理由が何であれ、最終的には、すべてのファイルリソースが使用でき、ユーザーがログインできる実行レベル 3 に戻すことになるでしょう。システムをマルチユーザーレベルに戻す手順については、[第 10 章](#)または[第 11 章](#)を参照してください。

すべてのデバイスの電源を落とす

次のような場合は、すべてのシステムデバイスの電源を落とす必要があります。

- ハードウェアを置換または追加する
- システムの設置場所を変更する

- 予定された停電や自然災害 (接近中の雷雨など) に備える

電源を落とすシステムデバイスとは、CPU、モニター、外部デバイス (ディスク、テープ、プリンタ) などを意味します。

すべてのシステムデバイスの電源を落とす前に、前の節の説明に従って、システムをクリーンにシャットダウンする必要があります。

▼ すべてのデバイスの電源を落とす方法

1. 次のどちらかを選択して、システムをシャットダウンします。
 - a. サーバーをシャットダウンする場合は、[157 ページの「サーバーをシャットダウンする方法」](#)を参照
 - b. スタンドアロンシステムをシャットダウンする場合は、[161 ページの「スタンドアロンシステムをシャットダウンする方法」](#)を参照
2. システムをシャットダウンしたら、すべてのデバイスの電源を落とします。必要な場合は、電源ケーブルをコンセントから引き抜きます。
3. 電源が回復したら、次の手順に従ってシステムとデバイスの電源を投入します。
 - a. 電源ケーブルをコンセントに差し込みます。
 - b. モニターの電源を入れます。
 - c. ディスクドライブ、テープドライブ、プリンタの電源を入れます。
 - d. CPU の電源を入れます。
システムが実行レベル 3 になります。

第 10 章

SPARC: システムのブート (手順)

この章では、OpenBoot™ PROM モニターを使用する手順と、SPARC システムを異なる実行レベルでブートする手順について説明します。

SPARC システムのブート手順については、165 ページの「[SPARC: システムのブート \(作業マップ\)](#)」を参照してください。

ブートプロセスの概要については、[第 7 章](#)を参照してください。ブートに関する問題の障害追跡を行うには、『Solaris のシステム管理 (上級編)』の「[リブートが失敗した場合の対処](#)」を参照してください。

x86 システムのブート手順については、[第 11 章](#)を参照してください。

SPARC: システムのブート (作業マップ)

作業	説明	参照先
ブート PROM の使用	ブート PROM はシステムのブートに使用する。必要に応じて、次のいずれかの作業を行う PROM のリビジョン番号を確認する システムのブートに使用できるデバイスを確認する	 167 ページの「SPARC: システムの PROM リビジョンを確認する方法」 167 ページの「SPARC: システム上のデバイスを確認する方法」

作業	説明	参照先
システムのブート	<p>デフォルトのブートデバイスを変更する(新しいディスクを追加した場合、あるいはシステムのブート方法を変更する必要がある場合)</p> <p>次のいずれかのブート方法を選択する</p> <p>実行レベル3でブートする - システムをシャットダウンするか、なんらかのシステムハードウェアの保守作業を行った後で使用する</p> <p>実行レベルSでブートする - なんらかのシステム保守作業(ファイルシステムのバックアップなど)を行った後で使用する このレベルでは、一部のファイルシステムだけがマウントされ、ユーザーはシステムにログインできない</p> <p>対話式でブートする - テストのために、システムファイルまたはカーネルを一時的に変更した後で使用する</p> <p>ネットワークからブートする - システムをネットワークからブートするときに使用する。この方法は、ディスクレスクライアントをブートする場合に使用する</p> <p>復元を目的としてブートする - 損傷したファイルまたはファイルシステムによってシステムのブートが行えないときにシステムをブートするために使用する。復元を目的としてブートするには、必要に応じて次のどちらかまたは両方の作業を行う</p> <p>まず、システムを停止して復元を試みる</p> <p>システムをブートして、システムの正常なブートを妨げている重要なシステムファイルを修復する</p> <p>kadbd をブートする - システムに関する問題の障害追跡を行うときに使用する</p>	<p>169 ページの「SPARC: デフォルトのブートデバイスを変更する方法」</p> <p>172 ページの「SPARC: システムを実行レベル3(マルチユーザーレベル)でブートする方法」</p> <p>173 ページの「SPARC: システムを実行レベルS(シングルユーザーレベル)でブートする方法」</p> <p>174 ページの「SPARC: システムを対話式でブートする方法」</p> <p>175 ページの「SPARC: システムをネットワークからブートする方法」</p> <p>177 ページの「SPARC: 復元を目的としてシステムを停止する方法」</p> <p>177 ページの「SPARC: 復元を目的としてシステムをブートする方法」</p> <p>177 ページの「SPARC: 復元を目的としてシステムを停止する方法」</p>

作業	説明	参照先
	クラッシュダンプを強制実行し、システムをリブートする - 障害追跡の目的でクラッシュダンプを強制実行するとき使用する	180 ページの「SPARC: クラッシュダンプを強制してシステムをリブートする方法」

SPARC: ブート PROM の使用

システム管理者は、通常 PROM レベルを使ってシステムをブートします。ただし、システムのブート方法の変更が必要な場合があります。たとえば、システムをマルチユーザーレベルにする前に、どのデバイスからブートするかを設定し直したり、ハードウェア診断プログラムを実行したりする場合があります。

次の作業を行う場合は、デフォルトのブートデバイスを変更する必要があります。

- 新しいドライブを永久または一時的にシステムに追加する
- ネットワークブート方法を変更する
- スタンドアロンシステムを一時的にネットワークからブートする

すべての PROM コマンドについては、`monitor(1M)` または `eeprom(1M)` のマニュアルページを参照してください。

SPARC: システムの PROM リビジョンを確認する方法

システムの PROM リビジョンレベルを `banner` コマンドで表示します。

```
ok banner
Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIIi 333MHz), No Keyboard
OpenBoot 3.15, 128 MB memory installed, Serial #number.
Ethernet address number, Host ID: number.
```

ハードウェア構成情報が表示され、その中に PROM のリビジョン番号があります。この例では、PROM のリビジョン番号は 3.15 です。

▼ SPARC: システム上のデバイスを確認する方法

システム上のデバイスを確認して、ブートに適したデバイスを見つけ出すことが必要な場合があります。

`probe` コマンドを使用してシステムに接続されているデバイスを安全に見つけるためには、次のことを行なっておく必要があります。

- PROM の `auto-boot?` パラメータの値を `false` に変更する
- `reset-all` コマンドを発行して、システムのレジスタをクリアする

次のように `sifting probe` コマンドを使用すると、システム上で利用可能な `probe` コマンドを表示できます。

```
ok sifting probe
```

システムのレジスタをクリアしないで `probe` コマンドを実行すると、次のメッセージが表示されます。

```
ok probe-scsi
```

```
This command may hang the system if a Stop-A or halt command
has been executed. Please type reset-all to reset the system
before executing this command.
```

```
Do you wish to continue? (y/n) n
```

1. PROM の `auto-boot?` パラメータの値を `false` に変更します。

```
ok setenv auto-boot? false
```

2. システムのレジスタをクリアします。

```
ok reset-all
```

3. システム上のデバイスを確認します。

```
ok probe-device
```

4. (省略可能) 電源障害後や `reset` コマンドの使用後にシステムをリポートするようにするには、`auto-boot?` パラメータを `true` に戻します。

```
ok setenv auto-boot? true
auto-boot? = true
```

5. システムをマルチユーザーモードでブートします。

```
ok reset
```

SPARC: 例 — システム上のデバイスを確認する

次の例は、Ultra10 システムに接続されているデバイスの確認方法を示しています。

```
ok setenv auto-boot? false
auto-boot? = false
ok reset-all
Resetting ...
```

```
Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIIi 333MHz), No Keyboard
OpenBoot 3.15, 128 MB memory installed, Serial #10933339.
Ethernet address 8:0:20:a6:d4:5b, Host ID: 80a6d45b.
```

```
ok probe-ide
```

```
Device 0 ( Primary Master )
```



```

                ATA Model: ST34321A

Device 1 ( Primary Slave )
        Not Present

Device 2 ( Secondary Master )
        Removable ATAPI Model: CRD-8322B

Device 3 ( Secondary Slave )
        Not Present

ok setenv auto-boot? true
auto-boot? =                true

```

devalias コマンドを使うと、システムに接続されている「可能性のある」デバイスの別名と関連パスを確認できます。

```

ok devalias
screen                /pci@1f,0/pci@1,1/SUNW,m64B@2
net                   /pci@1f,0/pci@1,1/network@1,1
cdrom                 /pci@1f,0/pci@1,1/ide@3/cdrom@2,0:f
disk                  /pci@1f,0/pci@1,1/ide@3/disk@0,0
disk3                 /pci@1f,0/pci@1,1/ide@3/disk@3,0
disk2                 /pci@1f,0/pci@1,1/ide@3/disk@2,0
disk1                 /pci@1f,0/pci@1,1/ide@3/disk@1,0
disk0                 /pci@1f,0/pci@1,1/ide@3/disk@0,0
ide                   /pci@1f,0/pci@1,1/ide@3
floppy                /pci@1f,0/pci@1,1/ebus@1/fdthree
ttyb                  /pci@1f,0/pci@1,1/ebus@1/se:b
ttya                  /pci@1f,0/pci@1,1/ebus@1/se:a
keyboard!             /pci@1f,0/pci@1,1/ebus@1/su@14,3083f8:forcemode
keyboard              /pci@1f,0/pci@1,1/ebus@1/su@14,3083f8
mouse                 /pci@1f,0/pci@1,1/ebus@1/su@14,3062f8
name                  aliases

```

▼ SPARC: デフォルトのブートデバイスを変更する方法

デフォルトのブートデバイスをほかのデバイスに変更するには、まずシステム上のデバイスを確認しておく必要があります。システム上のデバイスを確認する方法については、167 ページの「SPARC: システム上のデバイスを確認する方法」を参照してください。

1. スーパーユーザーになります。
2. 実行レベル **0** に変更します。

```
# init 0
ok PROM プロンプトが表示されます。
詳細については、init(1M) のマニュアルページを参照してください。
```

3. **boot-device** パラメータの値を変更します。

```
ok setenv boot-device device [n]
```

<code>boot-device</code>	ブートするデバイスを設定するパラメータを示す
<code>device[n]</code>	<code>boot-device</code> の値 (disk または net) を設定する。n はディスク番号

ディスク番号を確認する必要がある場合は、`probe` コマンドのいずれかを使用します。

4. デフォルトのブートデバイスが変更されていることを確認します。

```
ok printenv boot-device
```

5. 新しい **boot-device** 値を保存します。

```
ok reset
```

新しい `boot-device` 値が PROM に書き込まれます。

SPARC: 例 — デフォルトのブートデバイスを変更する

この例では、デフォルトのブートデバイスをディスクに設定しています。

```
# init 0
#
INIT: New run level: 0
.
.
.
The system is down.
syncing file systems... done
Program terminated
ok setenv boot-device disk
boot-device =          disk
ok printenv boot-device
boot-device          disk          disk
ok reset
Sun Ultra 5/10 UPA/PCI (UltraSPARC-III 333MHz), No Keyboard
OpenBoot 3.15, 128 MB memory installed, Serial #number.
Ethernet address number, Host ID: number.

Boot device: disk File and args:
SunOS Release 5.9 Version 64-bit
.
.
.
pluto console login:
```

この例では、デフォルトのブートデバイスをネットワークに設定しています。

```
# init 0
#
INIT: New run level: 0
.
.
.
The system is down.
syncing file systems... done
Program terminated
ok setenv boot-device net
boot-device =          net
ok printenv boot-device
boot-device           net                disk
ok reset
Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 333MHz), No Keyboard
OpenBoot 3.15, 128 MB memory installed, Serial #number.
Ethernet address number, Host ID: number.
```

```
Boot device: net  File and args:
```

```
.
.
.
pluto console login:
```

SPARC: システムをリセットする方法

ok プロンプトで `reset` コマンドを実行します。

```
ok reset
```

セルフテストプログラムが実行され、ハードウェアで診断テストを行なった後、システムがリブートされます。

SPARC: システムのブート

システムの電源を切ってから入れ直すと、マルチユーザーのブートシーケンスが開始されます。次に示す手順は、ok PROM プロンプトからさまざまな実行レベルでブートする方法です。これらの手順では、特に指示がない限り、システムがクリーンな状態でシャットダウンしていることを前提とします。

`who -r` コマンドを使って、システムが指定した実行レベルになっていることを確認します。実行レベルについての説明は、[第 8 章](#)を参照してください。

▼ SPARC: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法

次の手順に従って、現時点で実行レベル 0 になっているシステムを実行レベル 3 でブートします。

1. システムを実行レベル 3 でブートします。

```
ok boot
```

自動ブート処理では、一連のスタートアップメッセージ表示して、システムを実行レベル 3 にします。

詳細については、boot (1M) のマニュアルページを参照してください。

2. システムが実行レベル 3 になっていることを確認します。

ブートプロセスが正常に終了すると、ログイン画面かログインプロンプトが表示されます。

```
hostname console login:
```

SPARC: 例 — システムを実行レベル 3 (マルチユーザーレベル) でブートする

次の例は、システムを実行レベル 3 でブートしたときに表示されるメッセージを示しています。

```
ok boot
Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 333MHz)
OpenBoot 3.15, 128 MB memory installed, Serial #number.
Ethernet address number, Host ID: number.

Rebooting with command: boot
Boot device: disk:a File and args:
SunOS Release 5.9 Version Generic 64-bit
Copyright (c) 1983-2002 by Sun Microsystems, Inc.
configuring IPv4 interfaces: hme0.
Hostname: starbug
The system is coming up. Please wait.
checking ufs filesystems
/dev/rdisk/c0t0d0s7: is clean.
/dev/rdisk/c0t0d0s4: is clean.
NIS domainname is Solar.COM
starting rpc services: rpcbind keyserver ypbind done.
Setting netmask of hme0 to 255.255.255.0
Setting default IPv4 interface for multicast: add net 224.0/4:
gateway starbug
syslog service starting.
Print services started.
volume management starting.
The system is ready.
```

starbug console login:

▼ SPARC: システムを実行レベル S (シングルユーザーレベル) でブートする方法

次の手順に従って、現時点で実行レベル 0 になっているシステムを実行レベル S でブートします。

1. システムを実行レベル S でブートします。

```
ok boot -s
```

2. 次のメッセージが表示されたら、スーパーユーザーのパスワードを入力します。

```
INIT: SINGLE USER MODE
Type Ctrl-d to proceed with normal startup,

(or give root password for system maintenance): xxx
```

3. システムが実行レベル S になっていることを確認します。

```
# who -r
.          run-level S  Jun 10 15:27    3      0
```

4. システム保守作業の後に、システムをマルチユーザー状態にするには、**Control + D**を押します。

SPARC: 例 — システムを実行レベル S (シングルユーザーレベル) でブートする

次の例は、システムを実行レベル S でブートしたときに表示されるメッセージを示しています。

```
ok boot -s
.
.
.

Sun Microsystems Inc.  SunOS 5.9  Generic May 2002
Copyright 1983-2003 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.
configuring IPv4 interfaces: hme0.
Hostname: starbug

INIT: SINGLE USER MODE

Type control-d to proceed with normal startup,
(or give root password for system maintenance): xxx
```

```

Sun Microsystems Inc.   SunOS 5.9  Generic May 2002
# who -r
.      run-level S   Sep 19 08:49      S      0  ?
(保守作業を行う)
# Control + D を押す

```

▼ SPARC: システムを対話式でブートする方法

次の手順に従ってシステムをブートし、代替カーネルまたは `/etc/system` ファイルを指定する必要があります。

1. システムを対話式でブートします。

```
ok boot -a
```

2. 次の表に示すように、システムプロンプトに答えてください。

システムプロンプト	操作
Enter filename [kernel/[sparcv9]/unix]:	ブートに使用するカーネルの名前を入力する。あるいは、そのまま Return キーを押してデフォルトのカーネルを使用する
Enter default directory for modules [/platform/'uname - i'/kernel /platform/'uname - m/kernel /kernel /usr/kernel]:	modules ディレクトリの代替パスを入力する。あるいは、そのまま Return キーを押してデフォルトのカーネルモジュールディレクトリを使用する
Name of system file [etc/system]:	代替システムファイルの名前を入力して Return キーを押す。/etc/system ファイルが破損している場合、/dev/null を入力する。あるいは、そのまま Return キーを押してデフォルトの etc/system ファイルを使用する
root filesystem type [ufs]:	そのまま Return キーを押してデフォルトのルート (/) ファイルシステムを使用する。ローカルディスクからのブートの場合は UFS、ネットワークブートの場合は NFS を入力する
Enter physical name of root device [physical_device_name]:	代替デバイス名を入力して、Return キーを押す。あるいは、そのまま Return キーを押してルートデバイスのデフォルトの物理名を使用する

3. 前の表の質問に応答するためのプロンプトが表示されない場合は、`boot -a` コマンドを正しく入力していることを確認してください。

SPARC: 例 — システムを対話式でブートする

次の例では、利用できるデフォルトの選択例 ([] で囲まれた部分) を示します。

```
ok boot -a
.
.
.
Rebooting with command: boot -a
Boot device: /pci@1f,0/pci@1,1/ide@3/disk@0,0:a File and args: -a
Enter filename [kernel/sparcv9/unix]: <Return キーを押す>
Enter default directory for modules [/platform/SUNW,Ultra-5_10/kernel
/platform/sun4u/kernel /kernel /usr/kernel]: <Return キーを押す>
Name of system file [etc/system]: <Return キーを押す>
SunOS Release 5.9 Version Generic 64-bit
Copyright (c) 1983-2002 by Sun Microsystems, Inc.
root filesystem type [ufs]: <Return キーを押す>
Enter physical name of root device
[/pci@1f,0/pci@1,1/ide@3/disk@0,0:a]: <Return キーを押す>
configuring IPv4 interfaces: hme0.
Hostname: starbug
The system is coming up. Please wait.
checking ufs filesystems
.
.
.
The system is ready.
starbug console login:
```

▼ SPARC: システムをネットワークからブートする方法

ブートサーバーが利用できれば、どのようなシステムもネットワークからブートできます。たとえば、スタンドアロンのシステムがローカルディスクからブートできない場合は、このシステムを一時的にネットワークからブートできます。デフォルトのブートデバイスを変更または再設定する方法については、[169 ページの「SPARC: デフォルトのブートデバイスを変更する方法」](#)を参照してください。

sun4u システムでは次の 2 つのネットワーク構成ブート方法を利用できます。

- RARP (Reverse Address Resolution Protocol and ONC+ RPC Bootparams Protocol)
- DHCP (Dynamic Host Configuration Protocol)

デフォルトのネットワークブート方法は RARP に設定されています。ネットワークで利用できるブートサーバーによって、RARP または DHCP を選択できます。

注 - Sun Ultra システムで DHCP ネットワークブート方法を使用するには、PROM のバージョンが 3.25.nm 以上でなければなりません。PROM のバージョンの確認方法については、167 ページの「SPARC: システムの PROM リビジョンを確認する方法」を参照してください。

RARP と DHCP の両方のネットワークブート方法を利用できる場合、どちらのサービスを使用するかを boot コマンドに一時的に指定できます。あるいは、NVRAM 別名を設定すれば、システムをリブートしても有効に PROM レベルでネットワークブート方法を永続的に保存することができます。次の nvalias コマンドの例では、Sun Ultra 10 システムにおいてデフォルトで DHCP でブートするように、ネットワークデバイスの別名を設定します。

```
ok nvalias net      /pci@1f,4000/network@1,1:dhcp
```

この別名を設定している場合、boot net と入力するだけで、システムは DHCP ネットワークブート方法を使用してブートします。



注意 - nvunalias コマンドと nvalias コマンドの構文を十分理解するまで、nvunalias コマンドで NVRAMRC ファイルを変更しないでください。これらのコマンドの使用方法については、『OpenBoot 3.x コマンド・リファレンスマニュアル』を参照してください。

1. 必要な場合、システムをシャットダウンします。
2. ネットワークからブートする方法を決定し、次のどちらかを選択します。

どちらの方法でブートする場合でも、RARP または DHCP のブートサーバーがすでにネットワークに設定されていなければなりません。

 - a. DHCP を使用してシステムをネットワークからブートするには、次のように入力します。

```
ok boot net[:dhcp]
```

上記 nvalias の例のように、デフォルトで DHCP でブートするように PROM 設定を変更してある場合は、boot net と指定するだけで、システムは DHCP でブートします。

- b. RARP を使用してシステムをネットワークからブートするには、次のように入力します。

```
ok boot net[:rarp]
```

RARP はデフォルトのネットワークブート方法です。このため、デフォルトで DHCP でブートするように PROM 値を変更してある場合にだけ、boot net:rarp と指定しなければなりません。

▼ SPARC: 復元を目的としてシステムを停止する方法

1. システムのアポートキーシーケンスを入力します。
モニターに、ok PROM プロンプトが表示されます。

```
ok
```

アポートキーシーケンスは、キーボードのタイプによって異なります。たとえば、Stop+A または L1+A を押します。端末では、Break キーを押します。

2. ファイルシステムを同期させます。

```
ok sync
```

3. **syncing file systems...** というメッセージが表示されたら、システムのアポートキーシーケンスをもう一度入力します。

4. 該当する **boot** コマンドを入力して、ブートプロセスを起動します。
詳細については、boot (1M) のマニュアルページを参照してください。

5. システムが指定した実行レベルになっていることを確認します。

```
# who -r
.          run-level 3  May  2 07:39      3      0  S
```

SPARC: 例 — 復元を目的としてシステムを停止する

```
<Stop + A を押す>
ok sync
syncing file systems...
<Stop + A を押す>
ok boot
```

▼ SPARC: 復元を目的としてシステムをブートする方法

次の手順は、/etc/passwd などの重要なファイルに無効なエントリがあり、正常にブートできない場合に使用します。

次の手順で使用されている変数 *devicename* は、修復するファイルシステムのデバイス名に置き換えてください。システムのデバイス名を調べる場合は、『Solaris のシステム管理 (デバイスとファイルシステム)』の「デバイスへのアクセス (概要)」を参照してください。

1. システムのアポートキーシーケンスを使用して、システムを停止します。
システムのアポートキーシーケンスは、ルートのパスワードがわからない場合やシステムにログインできない場合に使用します。詳細については、177 ページの「SPARC: 復元を目的としてシステムを停止する方法」を参照してください。

2. **Solaris 9 Installation CD** または **DVD** からブートしているか、あるいはネットワークからブートしているかによって、次の表のそれぞれの手順を使い分けてください。

ブート方法	手順
Solaris 9 installation CD または DVD	1. Solaris 9 Installation CD または DVD をドライブに挿入する 2. Solaris 9 Installation CD または DVD から、シングルユーザーモードでブートする ok boot cdrom -s
ネットワーク (インストールサーバーまたはリモート CD/DVD ドライブが使用できる場合)	次のコマンドを使う ok boot net -s

3. ファイル内に無効なエントリがあるファイルシステムをマウントします。

```
# mount /dev/dsk/device-name /a
```

4. 新しくマウントしたファイルシステムに移動します。

```
# cd /a/file-system
```

5. 端末タイプを設定します。

```
# TERM=sun
# export TERM
```

6. エディタを使って、ファイルから無効なエントリを削除します。

```
# vi filename
```

7. ルート (/) ディレクトリに変更します。

```
# cd /
```

8. /a ディレクトリのマウントを解除します。

```
# umount /a
```

9. システムをリブートします。

```
# init 6
```

10. システムが実行レベル 3 になっていることを確認します。

ブートプロセスが正常に終了すると、ログイン画面かログインプロンプトが表示されます。

```
hostname console login:
```

SPARC: 例 — 復元を目的としてシステムをブートする (パスワードファイルが損傷した場合)

次の例は、ローカルの CD-ROM からブートした後、重要なシステムファイル (この場合は /etc/passwd) を修復する方法を示しています。

```
ok boot cdrom -s
# mount /dev/dsk/c0t3d0s0 /a
# cd /a/etc
# TERM=vt100
# export TERM
# vi passwd
(無効なエントリを削除する)
# cd /
# umount /a
# init 6
```

SPARC: 例 — ルートのパスワードを忘れた場合にシステムをブートする

次の例は、ルートのパスワードを忘れた場合に、ネットワークからブートしてシステムを復元する方法を示しています。この例では、ネットワークのブートサーバーが利用可能になっていることを前提とします。システムをリブートした後で必ず新しいルートパスワードを適用してください。

```
ok boot net -s
# mount /dev/dsk/c0t3d0s0 /a
# cd /a/etc
# TERM=vt100
# export TERM
# vi shadow
(ルートの暗号化パスワードを削除する)
# cd /
# umount /a
# init 6
```

▼ SPARC: カーネルデバッグ (kadb) を使ってシステムをブートする方法

1. システムを停止する必要がある場合は、システムのアボートキーシーケンスを入力します。

アボートキーシーケンスは、キーボードのタイプによって異なります。たとえば、Stop+A または L1+A を押します。端末では、Break キーを押します。

ok PROM プロンプトが表示されます。

2. ファイルシステムを同期させ、クラッシュダンプを書き出します。

```
> n
ok sync
```

3. **syncing file systems...** というメッセージが表示されたら、システムのアポートキーシーケンスをもう一度入力します。

4. カーネルデバッグを使ってシステムをブートします。

```
ok boot kadb
```

5. ブートメッセージで、システムがカーネルデバッグ (**kadb**) を使用してブートしているかどうかをチェックします。

```
Rebooting with command: kadb
Boot device: /iommu/sbus/espdma@4,800000/esp@4,8800000/sd@3,0
.
.
.
```

SPARC: 例 — カーネルデバッグ (kadb) を使ってシステムをブートする

```
<Stop + A を押す>
ok sync
syncing file systems...
<Stop + A を押す>
ok boot kadb
```

SPARC: クラッシュダンプを強制してシステムをリブートする

場合によっては、障害追跡の目的でクラッシュダンプを強制実行し、システムをリブートする必要があります。デフォルトでは、`savecore` 機能を使用できません。

システムのクラッシュダンプについては、『Solaris のシステム管理 (上級編)』の「システムクラッシュ情報の管理 (手順)」を参照してください。

▼ SPARC: クラッシュダンプを強制してシステムをリブートする方法

`savecore` 機能を使用できる場合は、次の手順に従ってクラッシュダンプを強制実行し、システムをリブートします。

1. システムのアポートキーシーケンスを入力します。

アポルトキーシーケンスは、キーボードのタイプによって異なります。たとえば、Stop + A または L1 + A を押します。端末では、Break キーを押します。ok PROM プロンプトが表示されます。

2. ファイルシステムを同期させ、クラッシュダンプを書き出します。

```
> n  
ok sync
```

クラッシュダンプがディスクに書き出されると、システムはそのままリブートします。

3. システムが実行レベル 3 になっていることを確認します。

ブートプロセスが正常に終了すると、ログイン画面かログインプロンプトが表示されます。

```
hostname console login:
```

SPARC: 例 — クラッシュダンプを強制してシステムをリブートする

```
<Stop + A を押す>  
ok sync
```


第 11 章

x86: システムのブート (手順)

この章では、x86 システムをブートする手順を説明します。

x86 システムのブート手順については、183 ページの「x86: システムのブート (作業マップ)」を参照してください。

ブートプロセスの概要については、第 7 章を参照してください。

SPARC システムのブート手順については、第 10 章を参照してください。

x86: システムのブート (作業マップ)

作業	説明	参照先
Solaris Device Configuration Assistant (デバイス構成用補助) のブート	システムのハードウェア構成を変更した後で使用する。このユーティリティを使用すると、異なるブートデバイスから Solaris システムをブートする、新しいハードウェアを構成する、間違って構成したハードウェアを構成し直すなど、デバイス関連やブート関連の作業を行うことができる	185 ページの「x86: Solaris Device Configuration Assistant (デバイス構成用補助) をブートする方法」
システムのブート	次のいずれかのブート方法を選択する	

作業	説明	参照先
	<p>実行レベル 3 でブートする - システムをシャットダウンするか、なんらかのシステムハードウェアの保守作業を行なった後で使用する</p> <p>実行レベル S でブートする - なんらかのシステム保守作業 (ファイルシステムのバックアップなど) を行なった後で使用する</p> <p>対話式でブートする - テストのために、システムファイルまたはカーネルを一時的に変更した後で使用する</p> <p>ネットワークからブートする - システムをネットワークからブートするときに使用する。この方法は、ディスクレスクライアントをブートする場合に使用する</p> <p>復元を目的としてブートする - 損傷したファイルによってシステムのブートが行えないときにシステムをブートするために使用する。復元を目的としてブートするには、必要に応じて次のどちらかまたは両方の作業を行う</p> <p>まず、システムを停止して復元を試みる</p> <p>システムをブートして、システムの正常なブートを妨げている重要なシステムファイルを修復する</p> <p>kadb をブートする - システムに関する問題の障害追跡を行うときに使用する</p> <p>クラッシュダンプを強制実行し、システムをリブートする - 障害追跡の目的でクラッシュダンプを強制実行するときに使用する</p>	<p>186 ページの「x86: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法」</p> <p>187 ページの「x86: システムを実行レベル S (シングルユーザーレベル) でブートする方法」</p> <p>188 ページの「x86: システムを対話式でブートする方法」</p> <p>190 ページの「x86: システムをネットワークからブートする方法」</p> <p>191 ページの「x86: 復元を目的としてシステムを停止する方法」</p> <p>191 ページの「x86: 復元を目的としてシステムをブートする方法」</p> <p>196 ページの「x86: カーネルデバッガ (kadb) を使ってシステムをブートする方法」</p> <p>197 ページの「x86: クラッシュダンプを強制してシステムをリブートする」</p>

x86: Solaris Device Configuration Assistant (デバイス構成用補助) のブート

Solaris Device Configuration Assistant (デバイス構成用補助) (Solaris x86 版) は、さまざまなハードウェア構成作業やブート作業を実行するためのプログラムです。Solaris Device Configuration Assistant (デバイス構成用補助) にアクセスする方法は2つあります。

- Solaris ブートフロッピーディスク
- Solaris 9 Installation CD または DVD

この章で説明する手順では、Solaris Device Configuration Assistant (デバイス構成用補助) ブートフロッピーディスクを挿入して、Device Configuration Assistant (デバイス構成用補助) をブートするように要求されることがあります。システムの BIOS が CD または DVD からのブートをサポートしている場合は、Solaris 9 Installation CD または DVD を挿入して、Device Configuration Assistant (デバイス構成用補助) をブートすることも可能です。

▼ x86: Solaris Device Configuration Assistant (デバイス構成用補助) をブートする方法

1. **Solaris Device Configuration** ブートフロッピーディスク、**Solaris 9 Installation CD** または **DVD** を適切なドライブに挿入します。

2. **Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。

あるいは、リセットボタンを使用することもできます。システムが停止している場合は、電源スイッチを押してシステムを起動します。

Device Configuration Assistant (デバイス構成用補助) の最初のメニューが、数分後に表示されます。

x86: システムのブート

次に、リセットボタンを使ってシステムを再起動する手順を示します。システムにリセットボタンがない場合は、電源スイッチを使ってシステムを再起動します。システムの状態によっては Control + Alt + Del キーを使って、システムの動作に割り込むことができます。

▼ x86: システムを実行レベル 3 (マルチユーザーレベル) でブートする方法

次の手順に従って、システム (現時点では実行レベル 0 になっている) を実行レベル 3 でブートします。

1. **Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。
あるいは、リセットボタンを使用することもできます。システムが停止している場合は、電源スイッチを押してシステムを起動します。
「Current Boot Parameters」メニューが、数分後に表示されます。
2. システムを実行レベル 3 にするには、**b** を入力します。**Enter** キーを押します。
5 秒以内に選択しない場合、システムは自動的に実行レベル 3 でブートします。
3. システムが実行レベル 3 になっていることを確認します。
ブートプロセスが正常に終了すると、ログイン画面かログインプロンプトが表示されます。

```
hostname console login:
```

x86: 例 — システムを実行レベル 3 (マルチユーザーレベル) でブートする

```
Type any key to continue
.
.
.
          <<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:
Type    b [file-name] [boot-flags] <ENTER>    to boot with options
or      i <ENTER>                               to enter boot interpreter
or      <ENTER>                                to boot with defaults

          <<< timeout in 5 seconds >>>

Select (b)oot or (i)nterpreter: b
.
.
.
venus console login:
```

▼ x86: システムを実行レベル S (シングルユーザーレベル) でブートする方法

次の手順に従って、システム (現時点では実行レベル 0 になっている) システムを実行レベル S でブートします。

1. **Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。
あるいは、リセットボタンを使用することもできます。システムが停止している場合は、電源スイッチを押してシステムを起動します。

「Current Boot Parameters」メニューが、数分後に表示されます。

2. システムを実行レベル S にするには、**b -s** を入力します。**Enter** キーを押します。
5 秒以内に選択しない場合、システムは自動的に実行レベル 3 でブートします。
3. プロンプトが表示されたら、スーパーユーザーのパスワードを入力します。
4. **who -r** を使用して、システムが実行レベル S になっていることを確認します。

```
# who -r
.          run-level S  Jul 19 14:37      S      0 3
```

5. 実行レベルを S に移行して行う必要がある保守作業を行います。
6. システムを実行レベル 3 に戻すには、**Control + D** を押します。

x86: 例 — システムを実行レベル S (シングルユーザーレベル) でブートする

```
Type any key to continue
.
.
.

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:
Type      b [file-name] [boot-flags] <ENTER>      to boot with options
or        i <ENTER>                                to enter boot interpreter
or        <ENTER>                                  to boot with defaults

<<< timeout in 5 seconds >>>

Select (b)oot or (i)nterpreter: b -s
.
.
.
```

```

INIT: SINGLE USER MODE

Type Ctrl-d to proceed with normal startup,
(or give root password for system maintenance): xxx
Entering System Maintenance Mode
.
.
.
# who -r
.      run-level S  Jul 19 14:37      S      0  3
(システム保守作業を行う)
# <Control + D を押す>

```

▼ x86: システムを対話式でブートする方法

次の手順に従ってシステムをブートし、代替カーネルまたは /etc/system ファイルを指定する必要があります。

- 1. Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。

あるいは、リセットボタンを使用することもできます。システムが停止している場合は、電源スイッチを押してシステムを起動します。

「Primary Boot Subsystem」メニューが、数分後に表示されます。
- 2. Solaris** パーティションにアクティブを示すマークが付いていない場合、その **Solaris** パーティションをリストから選択して、**Enter** キーを押します。

5 秒以内に選択しない場合、アクティブなブートパーティションが自動的に選択されます。

「Current Boot Parameters」メニューが、数分後に表示されます。
- 3. b -a** と入力して、システムを対話式でブートします。**Enter** キーを押します。

5 秒以内に選択しない場合、システムは自動的に実行レベル 3 でブートします。
- 4.** 次の表に示すように、システムプロンプトに答えてください。

システムプロンプト	操作
Enter default directory for modules: [/platform/i86pc/kernel /kernel /usr/kernel]:	modules ディレクトリの代替パスを指定し、 Enter キーを押す。あるいは、そのまま Enter キーを押してデフォルトの modules ディレクトリを使用する

システムプロンプト	操作
Name of system file [etc/system]:	代替システムファイルの名前を入力して Enter キーを押す。あるいは、そのまま Enter キーを押してデフォルトの /etc/system ファイルを使う。 /etc/system ファイルが破損している場合は、ファイルとして /dev/null を入力する
root filesystem type [ufs]:	そのまま Enter キーを押してデフォルトの ルート (/) ファイルシステムを使用する。 ローカルディスクからのブートの場合は UFS、ネットワークブートの場合は NFS と入力する
Enter physical name of root device [physical_device_name]:	代替デバイス名を入力して、Enter キーを押す。あるいは、そのまま Enter キーを押して ルートデバイス bootpath のデフォルトの 物理名を使用する

x86: 例 — システムを対話式でブートする

次の例では、デフォルトの選択肢 ([] で囲まれた部分) を使用しています。

```
Type any key to continue
.
.
.

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:
Type      b [file-name] [boot-flags] <ENTER>      to boot with options
or        i <ENTER>                                to enter boot interpreter
or        <ENTER>                                  to boot with defaults

<<< timeout in 5 seconds >>>>
Select (b)oot or (i)nterpreter: b -a
Enter default directory for modules [/platform/i86pc/kernel /kernel
/usr/kernel]: <Enter キーを押す>
Name of system file [etc/system]:<Enter キーを押す>
SunOS Release 5.9 Version Generic 32-bit
Copyright (c) 1983-2002 by Sun Microsystems, Inc.
root filesystem type [ufs]: <Enter キーを押す>
Enter physical name of root device
[/pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a]: <Enter キーを押す>
configuring IPv4 interfaces: dnet0.
Hostname: venus
(fsck メッセージが表示される)
The system is coming up. Please wait
(メッセージが表示される)
venus console login:
```

▼ x86: システムをネットワークからブートする方法

ブートサーバーが利用できれば、どのようなシステムもネットワークからブートできます。たとえば、スタンドアロンのシステムがローカルディスクからブートできない場合は、このシステムを一時的にネットワークからブートできます。

システムが PXE ネットワークブートに対応している場合は、Device Configuration Assistant (デバイス構成用補助) ブートフロッピーディスクや Solaris 9 Installation CD または DVD を使用しなくても、ネットワークから直接システムをブートできます。

Device Configuration Assistant (デバイス構成用補助) の「Boot Tasks」メニューにある「Set Network Configuration Strategy」メニューを使用して、適切なブート方法を選択できます。

1. ネットワークからブートする方法 (**RARP/bootparams** または **DHCP**) を決定します。
2つのネットワーク構成方法、RARP (Reverse Address Resolution Protocol) または DHCP (Dynamic Host Configuration Protocol) があります。デフォルトのネットワークブート方法は RARP に設定されています。ネットワークで利用できるブートサーバーによって、RARP または DHCP を選択できます。
PXE ネットワークブートは、DHCP でのみ使用できます。
2. **Device Configuration Assistant** (デバイス構成用補助) ブートフロッピーディスク、**Solaris 9 Installation CD** または **DVD** を適切なドライブに挿入します。
あるいは、システムまたはネットワークアダプタの BIOS 構成プログラムを使用して、PXE ネットワークブートを使用可能にします。
3. **Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。
あるいは、リセットボタンを使用することもできます。システムが停止している場合は、電源スイッチを押してシステムを起動します。
「Solaris Device Configuration Assistant」画面が表示されます。
4. 「**F2_Continue**」(**F2** キー) を押して、デバイスを走査します。
デバイスの識別が行われ、識別されたデバイスを表示する画面が表示されます。
5. 「**F2_Continue**」(**F2** キー) を押して、ドライバを読み込みます。
ブート可能なドライバが読み込まれます。
「Boot Solaris」メニューが表示されます。
6. 「**F4_Boot Tasks**」(**F4** キー) を押します。
7. 「**Set Network Configuration Strategy**」を選択して、「**F2_Continue**」(**F2** キー) を押します。
8. 「**RARP**」または「**DHCP**」を選択して、「**F2_Continue**」(**F2** キー) を押します。
新しいネットワークブート方法を確認する画面が表示されます。

ここで選択したネットワークブート方法は、以降、このフロッピーディスクでブートしたときのデフォルトのネットワークブート方法として保存されます。

9. 「**F3_Back**」を押して、「**Boot Solaris**」メニューに戻ります。
10. 「**NET**」をブートデバイスとして選択します。そして、「**F2_Continue**」を押して、ネットワークデバイスをブートします。
「Solaris boot option」画面が表示されます。

▼ x86: 復元を目的としてシステムを停止する方法

1. 可能であれば、次のどちらかのコマンドを使用してシステムを停止します。
 - システムが動作している場合、スーパーユーザーになって `init 0` と入力してシステムを停止します。Type any key to continue プロンプトが表示されたら、任意のキーを押してシステムをリブートします。
 - システムが動作している場合、スーパーユーザーになって `init 6` と入力してシステムをリブートします。
2. マウスまたはキーボードからの入力にシステムが応答しない場合、リセットキーがあればそのキーを押してシステムをリブートします。あるいは、電源スイッチを使用してシステムをリブートします。

▼ x86: 復元を目的としてシステムをブートする方法

重要なシステムリソースを修復するには、次の手順に従ってシステムをブートします。この例は、Solaris 9 Installation CD またはネットワークからブートし、ルート (/) ファイルシステムをディスクにマウントし、`/etc/passwd` ファイルを修復する方法を示しています。

次の手順で使用されている変数 `devicename` は、修復するファイルシステムのデバイス名に置き換えてください。システムのデバイス名を調べる場合は、『Solaris のシステム管理 (デバイスとファイルシステム)』の「デバイスへのアクセス (概要)」を参照してください。

1. まず、システムのアボートキーシーケンスを使用して、システムを停止します。
システムのアボートキーシーケンスは、ルートのパスワードがわからない場合やシステムにログインできない場合に使用します。詳細については、[191 ページの「x86: 復元を目的としてシステムを停止する方法」](#)を参照してください。
2. **Solaris 9 Installation CD** または **DVD**、あるいはネットワークから、シングルユーザーモードでブートします。
 - a. **Device Configuration Assistant** (デバイス構成用補助) ブートフロッピーディスク、**Solaris 9 Installation CD** または **DVD** を適切なドライブに挿入します。

- b. **Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。
あるいは、リセットボタンを使用することもできます。システムが停止している場合は、電源スイッチを押してシステムを起動します。
「Solaris Device Configuration Assistant」画面が表示されます。
- c. 「**F2_Continue**」(**F2** キー) を押します。
デバイスの識別が行われ、識別されたデバイスを表示する画面が表示されます。
- d. 「**F2_Continue**」(**F2** キー) を押します。
ブート可能なドライバが読み込まれます。
「Boot Solaris」メニューが表示されます。
- e. **CD-ROM** ドライブまたはネットワークデバイスを選択します。次に「**F2_Continue**」(**F2** キー) を押します。
「Current Boot Parameters」メニューが表示されます。
- f. プロンプトで **b -s** と入力して、**Enter** キーを押します。
数分後に、シングルユーザーモードの # プロンプトが表示されます。
3. 無効な **passwd** ファイルがあるルート (**/**) ファイルシステムをマウントします。
`# mount /dev/dsk/devicename /a`
4. 新しくマウントした **etc** ディレクトリに移動します。
`# cd /a/etc`
5. エディタを使用して **passwd** ファイルに必要な変更を加えます。
`# vi filename`
6. ルート (**/**) ディレクトリに変更します。
`# cd /`
7. **/a** ディレクトリのマウントを解除します。
`# umount /a`
8. システムをリブートします。
`# init 6`
9. システムが実行レベル **3** になっていることを確認します。
ブートプロセスが正常に終了すると、ログイン画面かログインプロンプトが表示されます。
`hostname console login:`

x86: 例 — 復元を目的としてシステムをブートする

次の例は、ローカルの CD-ROM からブートした後で /etc/passwd ファイルを修復する方法を示しています。

Type any key to continue

SunOS Secondary Boot version 3.00

Solaris Intel Platform Edition Booting System

Running Configuration Assistant...

Autobooting from Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a

If the system hardware has changed, or to boot from a different device, interrupt the autoboot process by pressing ESC.

Press ESCape to interrupt autoboot in 5 seconds.

.
.
.

Boot Solaris

Select one of the identified devices to boot the Solaris kernel and choose Continue.

To perform optional features, such as modifying the autoboot and property settings, choose Boot Tasks.

An asterisk (*) indicates the current default boot device.

> To make a selection use the arrow keys, and press Enter to mark it [X].

```
[ ] NET : DEC 21142/21143 Fast Ethernet
on Board PCI at Dev 3
[ ] DISK: (*) Target 0, QUANTUM FIREBALL1280A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 1:ST5660A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 0:Maxtor 9 0680D4
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[X] CD : Target 1:TOSHIBA CD-ROM XM-5602B 1546
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
```

F2_Continue F3_Back F4_Boot Tasks F6_Help

.
.
.

<<< Current Boot Parameters >>>

Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a

Boot args: kernel/unix -r

Select the type of installation you want to perform:

```
1 Solaris Interactive
2 Custom JumpStart
3 Solaris Web Start
```

Enter the number of your choice followed by <ENTER> the key.

If you enter anything else, or if you wait for 30 seconds, an interactive installation will be started.

```
Select type of installation: b -s
```

```
.
.
.
# mount /dev/dsk/c0t0d0s0 /a
.
.
.
# cd /a/etc
# vi passwd
(無効なエントリを削除する)
# cd /
# umount /a
# init 6
```

x86: 例 — ルートのパスワードを忘れた場合にシステムをブートする

次の例は、ルートのパスワードを忘れた場合に、ネットワークからブートしてシステムを復元する方法を示しています。この例では、ブートサーバーがすでに有効になっているものとします。システムをリブートした後で必ず新しいルートパスワードを適用してください。

Type any key to continue

SunOS Secondary Boot version 3.00

Solaris Intel Platform Edition Booting System

Running Configuration Assistant...

Autobooting from Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a

If the system hardware has changed, or to boot from a different device, interrupt the autoboot process by pressing ESC.

Press ESCape to interrupt autoboot in 5 seconds.

```
.
.
.
Boot Solaris
```

Select one of the identified devices to boot the Solaris kernel and

choose Continue.

To perform optional features, such as modifying the autoboot and property settings, choose Boot Tasks.

An asterisk (*) indicates the current default boot device.

> To make a selection use the arrow keys, and press Enter to mark it [X].

```
[X] NET : DEC 21142/21143 Fast Ethernet
on Board PCI at Dev 3
[ ] DISK: (*) Target 0, QUANTUM FIREBALL1280A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 1:ST5660A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 0:Maxtor 9 0680D4
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] CD : Target 1:TOSHIBA CD-ROM XM-5602B 1546
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
```

```
F2_Continue F3_Back F4_Boot Tasks F6_Help
```

```
.
.
.
```

```
<<< Current Boot Parameters >>>
```

```
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
```

```
Boot args: kernel/unix -r
```

Select the type of installation you want to perform:

- 1 Solaris Interactive
- 2 Custom JumpStart
- 3 Solaris Web Start

Enter the number of your choice followed by <ENTER> the key.

If you enter anything else, or if you wait for 30 seconds, an interactive installation will be started.

Select type of installation: **b -s**

```
.
.
.
```

```
# mount /dev/dsk/c0t0d0s0 /a
```

```
.
.
.
```

```
# cd /a/etc
```

```
# vi shadow
```

(ルートの暗号化パスワード文字列を削除する)

```
# cd /
```

```
# umount /a
```

```
# init 6
```

▼ x86: カーネルデバッグ (kadb) を使ってシステムをブートする方法

1. **Type any key to continue** プロンプトが表示されている場合は、任意のキーを押してシステムをリブートします。
あるいは、リセットボタンを使用することもできます。
システムが停止している場合は、電源スイッチを押してシステムを起動します。
2. **b kadb** と入力して、カーネルデバッグをブートします。**Enter** キーを押します。
5秒以内に選択しない場合、システムは自動的に実行レベル3でブートします。
3. システムが実行レベル3になっていることを確認します。
ブートプロセスが正常に終了すると、ログイン画面かログインプロンプトが表示されます。

`hostname console login:`
4. **F1 + A** を押して、カーネルデバッグにアクセスできることを確認します。
カーネルデバッグがブートすると、`kadb[0]`: プロンプトが表示されます。

x86: 例 — カーネルデバッグ (kadb) を使ってシステムをブートする

```
Type any key to continue
.
.
.
      <<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:
Type      b [file-name] [boot-flags] <ENTER>      to boot with options
or        i <ENTER>                                to enter boot interpreter
or        <ENTER>                                  to boot with defaults

      <<< timeout in 5 seconds >>>

Select (b)oot or (i)nterpreter: b kadb
.
.
.
naboo console login: (ログイン名とパスワードを入力する)
(F1 + A を押して、カーネルデバッグにアクセスできることを確認する)
```

x86: クラッシュダンプを強制してシステムをリブートする

場合によっては、障害追跡の目的でクラッシュダンプを強制実行し、システムをリブートする必要があります。デフォルトでは、`savecore` 機能を使用できます。

システムのクラッシュダンプについては、『Solaris のシステム管理 (上級編)』の「システムクラッシュ情報の管理 (手順)」を参照してください。

▼ x86: クラッシュダンプを強制してシステムをリブートする方法

`kadb[0]`: プロンプトを表示して、クラッシュダンプを強制するには、カーネルデバッグオプション `kadb` を指定してシステムをブートする必要があります。

注 - カーネルデバッグ (`kadb`) に入るにはテキストモードでなければなりません。したがって、まずウィンドウシステムを終了してください。

1. **F1 + A** を押します。

```
kadb[0]:
```

```
kadb[0]: プロンプトが表示されます。
```

2. 次のコマンドを `kadb[0]`: プロンプトで入力します。

```
<F1 + A を押す>
```

```
kadb[0]: vfs_syncall/W ffffffff
```

```
kadb[0]: 0>eip
```

```
kadb[0]: :c
```

```
kadb[0]: :c
```

```
kadb[0]: :c
```

最初の `:c` を入力すると、システムはパニックを起こします。そこでもう一度 `:c` を入力します。システムは再度パニックを起こすため、3 度目の `:c` を入力し、クラッシュダンプを強制して、システムをリブートします。

クラッシュダンプがディスクに書き込まれた後、システムはリブートします。

3. コンソールログインプロンプトでログインして、システムがリブートされていることを確認します。

第 12 章

ブートプロセス (参照情報)

この章では、SPARC システム と x86 システムをブートするためのファームウェアについて説明します。また、各プラットフォームのブートプロセスの概要も説明します。

この章の内容は次のとおりです。

- 199 ページの「SPARC: ブート PROM」
- 200 ページの「SPARC: ブートプロセス」
- 200 ページの「x86: PC BIOS」
- 201 ページの「x86: ブートサブシステム」
- 206 ページの「x86: ブートプロセス」

システムのブート手順については、第 10 章または第 11 章を参照してください。

SPARC: ブート PROM

SPARC システムごとに、「モニター」と呼ばれるプログラムを格納している PROM (プログラム可能な読み取り専用メモリー) チップがあります。モニターは、Solaris のカーネルが使用される前に、システムの動作を制御します。システムをオンにすると、モニターはシステムのハードウェアやメモリーを検査するセルフテスト手順を実行します。エラーが検出されなければ、システムは、自動ブートプロセスを開始します。

注 - Solaris システムが動作する前に、以前のシステムの一部には PROM アップグレードが必要なものもあります。詳細については、ご購入先にお問い合わせください。

SPARC: ブートプロセス

次の表では、SPARC システムのブートプロセスについて説明します。

表 12-1 SPARC: ブートプロセスの説明

ブート段階	説明
ブート PROM	1. PROM は、システム識別情報を表示し、セルフテスト診断を実行してシステムのハードウェアとメモリーを検査する 2. PROM は一次ブートプログラム bootblk を読み込む。このプログラムの目的は、デフォルトのブートデバイスから二次ブートプログラム (ufs ファイルシステムにある) を読み込むことにある
ブートプログラム	3. bootblk プログラムは二次ブートプログラム ufsboot を検索して実行し、それをメモリーに読み込む 4. ufsboot プログラムが読み込まれた後、ufsboot プログラムはカーネルを読み込む
カーネル初期設定	5. カーネルが自身を初期設定し、ファイルの読み込みに ufsboot を使ってモジュールのロードを開始する。カーネルはルート (/) ファイルシステムをマウントするのに十分なモジュールをロードすると、ufsboot プログラムの対応づけを解除し、自身のリソースを使って動作を続ける 6. カーネルがユーザープロセスを作成し、/etc/inittab ファイルを読み取って他のプロセスを起動する /sbin/init プロセスを起動する
init	7. /sbin/init プロセスが実行制御 (rc) スクリプトを起動し、このスクリプトは一連の他のスクリプトを実行する。それらのスクリプト (/sbin/rc*) はファイルシステムを検査、マウントし、さまざまなプロセスを実行して、システム保守作業を実行する

x86: PC BIOS

カーネルが起動される前は、システムは、PC 上のファームウェアインタフェースである、読み取り専用メモリー (ROM) の Basic Input/Output System (BIOS) によって制御されます。

ハードウェアアダプタにオンボード BIOS を搭載することができ、それによってデバイスの物理的特性を表示したり、デバイスにアクセスしたりすることができます。

起動シーケンスの間、PC の BIOS はアダプタの BIOS があるかどうか調べ、あれば、それぞれのアダプタ BIOS をロードして、実行します。アダプタの BIOS はそれぞれセルフテスト診断を実行して、デバイス情報を表示します。

x86: ブートサブシステム

Solaris のブート時に、ブートシステムについて次の選択を行うことができます。

- 一次ブートサブシステム (「Partition Boot」メニュー) — この最初のメニューは、複数のオペレーティングシステムがディスク上にある場合に表示されます。このメニューで、インストールされたどのオペレーティングシステムからブートするかを指定できます。デフォルトでは、アクティブなオペレーティングシステムからブートします。

Solaris オペレーティングシステム以外のオペレーティングシステムからブートする場合には、次の 2 つのメニューは表示されません。

- 自動ブートプロセスへの割り込み — 自動ブートプロセスに割り込むと、Solaris Device Configuration Assistant (デバイス構成用補助) にアクセスできます。

Solaris Device Configuration Assistant (デバイス構成用補助) を使用すると、異なるブートデバイスから Solaris システムをブートする、新しいハードウェアを構成する、間違っ構成したハードウェアを構成し直すなど、デバイス関連やブート関連の作業を実行できます。

- 「Current Boot Parameters」メニュー — このメニューには 2 つの形式があります。1 つは、通常の Solaris ブート用で、もう 1 つは Solaris インストールブート用です。
 - 通常の「Current Boot Parameters」メニューでは、オプションを付けて Solaris システムをブートしたり、ブートインタプリタを実行したりできます。
 - インストール用の「Current Boot Parameters」メニューでは、インストールの種類を選択したり、ブートをカスタマイズしたりできます。

次の表に、主な x86 ブートインタフェースの目的を示します。以降の節では、各ブートインタフェースを詳細に説明し、例を示します。

表 12-2 x86: ブートサブシステム

ブートサブシステム	目的
一次ブートサブシステム	このメニューは、現在ブート中のディスクに複数のオペレーティングシステム (Solaris オペレーティングシステムを含む) が含まれている場合に表示される
二次ブートサブシステム	Solaris リリースをブートするたびに表示される。自動ブートプロセスに割り込んで、Solaris Device Configuration Assistant を実行することを選択していない限り、Solaris リリースが自動的にブートされる

表 12-2 x86: ブートサブシステム (続き)

ブートサブシステム	目的
Solaris Device Configuration Assistant ブートフロッピーディスク	<p>「Solaris Device Configuration Assistant」メニューを表示するには次の2つの方法がある</p> <ol style="list-style-type: none"> 1. Solaris Device Configuration Assistant ブートフロッピーディスクまたは Solaris 9 Installation CD (CD-ROM ドライブからブートできるシステムのみ) を使用してシステムをブートする 2. インストールしたディスクから Solaris ソフトウェアをブートするときに自動ブートプロセスに割り込む
「Current Boot Parameters」メニュー	このメニューは、Solaris をディスク、CD-ROM、またはネットワークからブートするときに使用する。このメニューにはブートオプションが表示される

注 - Solaris Device Configuration Assistant ブートフロッピーディスクを作成する必要がある場合は http://www.sun.com/bigadmin/hcl/drivers/dca_diskettes/ を参照してください。

ブートプロセス中は、ブートサブシステムメニューは異なるデバイスとブートオプションを表示します。システムは何回かのタイムアウトの後で応答を受け付けなくなった場合、デフォルトの設定値を使って自動的にブートを継続します。ブートサブシステムメニューが表示されるたびに、ブートプロセスを停止することができます。また、自動的に継続させることもできます。

次の節では、各ブートサブシステムの画面の例を示します。

x86: Solaris リリースのブート

デバイスを識別する段階では、Solaris Device Configuration Assistant (デバイス構成用補助) は次の処理を実行します。

- システムにインストールされているデバイスを走査する
- 識別されたデバイスを表示する
- キーボードの選択やデバイスとそのリソースの編集などのオプションの作業を可能にする

ブートの段階では、Solaris Device Configuration Assistant (デバイス構成用補助) は次の処理を実行します。

- ブートするデバイスのリストを表示する。アスタリスク (*) が付いたデバイスがデフォルトのブートデバイスである

- 自動ブート設定やプロパティ設定の編集、ネットワーク構成方法の選択などのオプションの作業を可能にする

次の節では、デバイスを識別する段階で表示されるメニューの例を示します。デバイス出力は、各システム構成によって異なります。

x86: デバイス識別段階で表示される画面

Solaris Device Configuration Assistant (デバイス構成用補助) がシステム上のデバイスを識別するときに、いくつかの画面が表示されます。

x86: Device Configuration Assistant 画面

この画面は、Solaris Device Configuration Assistant (デバイス構成用補助) をブートするたびに表示されます。Solaris Device Configuration Assistant (デバイス構成用補助) はシステムがブートするたびに実行されますが、自動ブートプロセスではこのメニューは省略されます。

```
Solaris Device Configuration Assistant
```

```
The Solaris(TM) (Intel Platform Edition) Device Configuration Assistant scans to identify system hardware, lists identified devices, and can boot the Solaris software from a specified device. This program must be used to install the Solaris operating environment, add a driver, or change the hardware on the system.
```

```
> To perform a full scan to identify all system hardware, choose Continue.
```

```
> To diagnose possible full scan failures, choose Specific Scan.
```

```
> To add new or updated device drivers, choose Add Driver.
```

```
About navigation...
```

- The mouse cannot be used.
- If the keyboard does not have function keys or they do not respond, press ESC. The legend at the bottom of the screen will change to show the ESC keys to use for navigation.
- The F2 key performs the default action.

```
F2_Continue
```

```
F3_Specific Scan
```

```
F4_Add Driver
```

```
F6_Help
```

x86: Bus Enumeration 画面

この画面は、Solaris Device Configuration Assistant (デバイス構成用補助) が自動検出できるデバイスのハードウェア構成データを集めているときに一時的に表示されません。

```
Bus Enumeration
```

```
Determining bus types and gathering hardware configuration data ...
```

Please wait ...

x86: Scanning Devices 画面

この画面は、特別なドライバだけで検出できるデバイスを Solaris Device Configuration Assistant (デバイス構成用補助) が手動で走査しているときに表示されます。

Scanning Devices

The system is being scanned to identify system hardware.

If the scanning stalls, press the system's reset button. When the system reboots, choose Specific Scan or Help.

Scanning: Floppy disk controller

```
#####
|           |           |           |           |
0           20          40          60          80          100
```

Please wait ...

x86: Identified Devices 画面

この画面は、システムで識別されたデバイスを表示します。この画面からは、「Boot Solaris」メニューに移動でき、また、キーボード構成の設定、デバイスの表示や編集、シリアルコンソールの設定、構成の保存や削除など、オプションのデバイス作業も実行できます。

Identified Devices

The following devices have been identified on this system. To identify devices not on this list or to modify device characteristics, such as keyboard configuration, choose Device Tasks. Platform types may be included in this list.

```
ISA: Floppy disk controller
    ISA: Motherboard
    ISA: PnP bios: 16550-compatible serial controller
    ISA: PnP bios: 16550-compatible serial controller
    ISA: PnP bios: Mouse controller
    ISA: PnP bios: Parallel port
    ISA: System keyboard (US-English)
    PCI: Bus Mastering IDE controller
    PCI: Universal Serial Bus
    PCI: VGA compatible display adapter
```

F2_Continue F3_Back F4_Device Tasks F6_Help

x86: ブート段階で表示されるメニュー

この段階では、システムをブートする方法を指定できます。

x86: Boot Solaris メニュー

「Boot Solaris」メニューからは、Solaris リリースをブートするデバイスを選択できます。また、autoboot やプロパティ設定の表示や編集など、オプションの作業も実行できます。ブートデバイスを選択し、「Continue」を選択すると、Solaris カーネルがブートを開始します。

```
Boot Solaris
Select one of the identified devices to boot the Solaris kernel and
choose Continue.

To perform optional features, such as modifying the autoboot and property
settings, choose Boot Tasks.

An asterisk (*) indicates the current default boot device.

> To make a selection use the arrow keys, and press Enter to mark it [X].

[X] DISK: (*) Target 0:QUANTUM FIREBALL1280A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 1:ST5660A
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] DISK: Target 0:Maxtor 9 0680D4
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1
[ ] CD : Target 1:TOSHIBA CD-ROM XM-5602B 1546
on Bus Mastering IDE controller on Board PCI at Dev 7, Func 1

F2_Continue F3_Back F4_Boot Tasks F6_Help
```

x86: 「Current Boot Parameters」メニュー

このメニューは、ローカルディスクから Solaris リリースをブートするたびに表示されます。デフォルトの Solaris カーネルからブートしたい場合は、5 秒間のタイムアウトが経過するのを待ちます。別のオプションでブートする場合は、タイムアウト期間が経過する前に、適切なオプションを選択します。

```
<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:
Type      b [file-name] [boot-flags] <ENTER>      to boot with options
or        i <ENTER>                             to enter boot interpreter
or        <ENTER>                               to boot with defaults

<<< timeout in 5 seconds >>>
```

Select (b)oot or (i)nterpreter:

x86: ブートプロセス

次の表では、x86 システムのブートプロセスについて説明します。

表 12-3 x86: ブートプロセスの説明

ブート段階	説明
BIOS	<p>1. システムの電源をオンにすると、BIOS がセルフテスト診断を実行してシステムのハードウェアとメモリーを検査する。エラーが検出されなければ、システムは自動的にブートを開始する。エラーが検出されると、復元オプションを示すエラーメッセージが表示される</p> <p>追加のハードウェアデバイスの BIOS がここで実行される</p> <p>2. BIOS ブートプログラムが、ブートデバイスの最初の物理セクターを読もうとする。ブートデバイス上のこの最初のディスクセクターにはマスターブートレコード mboot が格納されており、このレコードが読み込まれて実行される。mboot ファイルが見つからなかった場合は、エラーメッセージが表示される</p>
ブートプログラム	<p>3. マスターブートレコード mboot には、アクティブなパーティションと、Solaris ブートプログラム pboot の位置を見つけるのに必要なディスク情報が格納されており、mboot は、pboot を読み込んで実行する</p> <p>4. Solaris ブートプログラム pboot が一次ブートプログラム bootblk を読み込む。このプログラムは、ufs ファイルシステムにある二次ブートプログラムを読み込む</p> <p>5. ブート可能なパーティションが複数ある場合、bootblk は fdisk テーブルを読んでデフォルトのブートパーティションを探し、利用可能なパーティションのメニューを作成して表示する。ブートするパーティションを選択する際に、30 秒経過するとタイムアウトになる。これは、ブート可能なパーティションが複数ある場合のみ発生する</p> <p>6. bootblk はルート (/) ファイルシステムで二次ブートプログラム boot.bin または ufsboot を探し、実行する。Solaris Device Configuration Assistant (デバイス構成用補助) を起動するときは、5 秒以内に自動ブートに割り込みをかける</p> <p>7. 二次ブートプログラム boot.bin または ufsboot が /etc/bootrc スクリプトを実行するコマンドインタプリタを起動する。ここでシステムのブート方法を選択するメニューが表示される。デフォルトでは、カーネルが読み込まれ、実行される。ブートオプションを指定したり、ブートインタプリタを起動したりするときは、5 秒以内に行う</p>

表 12-3 x86: ブートプロセスの説明 (続き)

ブート段階	説明
カーネル初期設定	<p>8. カーネルは二次ブートプログラム <code>boot.bin</code> または <code>ufsboot</code> を使用してファイルを読み込み、カーネル自身を初期化して、モジュールの読み込みを開始する。カーネルはルート (<code>/</code>) ファイルシステムをマウントするのに十分なモジュールを読み込むと、二次ブートプログラムの対応づけを解除し、自身のリソースを使って動作を続ける</p> <p>9. カーネルがユーザープロセスを作成し、<code>/etc/inittab</code> ファイルを読み取って他のプロセスを起動する <code>/sbin/init</code> プロセスを起動する</p>
init	<p>10. <code>/sbin/init</code> プロセスは実行制御 (<code>rc</code>) スクリプトを開始し、このスクリプトが一連の他のスクリプトを実行する。それらのスクリプト (<code>/sbin/rc*</code>) はファイルシステムを検査、マウントし、さまざまなプロセスを実行して、システム保守作業を実行する</p>

第 13 章

ソフトウェアの管理 (概要)

ソフトウェアの管理には、スタンドアロンシステム、サーバー、およびそのクライアントへのソフトウェアの追加やソフトウェアの削除が含まれます。この章では、ソフトウェアのインストールや管理に使用できる各種ツールについて説明します。

この章では、新しいシステムでの Solaris ソフトウェアのインストールについては説明しません。また、新バージョンの Solaris ソフトウェアのインストールやアップグレードについても説明しません。Solaris ソフトウェアのインストールやアップグレードについては、『Solaris 9 9/04 インストールガイド』を参照してください。

この章の内容は次のとおりです。

- 209 ページの「Solaris 9 Update リリースの新しいソフトウェア管理機能」
- 210 ページの「Solaris 9 リリースの新しいソフトウェア管理機能」
- 212 ページの「ソフトウェア管理作業についての参照先」
- 212 ページの「ソフトウェアパッケージの概要」
- 218 ページの「ソフトウェアパッケージを管理するためのツール」
- 219 ページの「ソフトウェアパッケージの追加または削除 (pkgadd)」
- 220 ページの「ソフトウェアパッケージの追加にあたっての重要な注意点 (pkgadd)」
- 220 ページの「パッケージの削除に関するガイドライン (pkgrm)」
- 221 ページの「パッケージ追加時のユーザーの対話操作を省略する (pkgadd)」

ソフトウェアを管理するための手順については、第 14 章を参照してください。

Solaris 9 Update リリースの新しいソフトウェア管理機能

この節では、この Solaris リリースの新しいソフトウェア管理機能について説明します。

pkgadd と patchadd による署名付きパッケージ およびパッチのサポート

Solaris 9 12/03 – この Solaris リリースでは、拡張された pkgadd コマンドと patchadd コマンドを使うことで、デジタル署名を含む Solaris パッケージと Solaris パッチを安全にダウンロードできるようになりました。

以前の Solaris リリースでは、Solaris パッチ管理ツールをダウンロードし、smpatch コマンドと PatchPro を併用することで、署名付きパッチの管理を行えました。smpatch コマンドを使って署名付きパッチを管理する手順については、『*Signed Patches Administration Guide for PatchPro 2.2*』(英語版)の「*Managing Signed Patches by Using Solaris Patch Management Tools (Tasks)*」を参照してください。

署名付きパッケージの概要については、212 ページの「ソフトウェアパッケージの概要」を参照してください。

patchadd コマンドを使って署名付きパッチを追加する手順については、277 ページの「patchadd コマンドによる署名付きパッチの追加 (作業マップ)」を参照してください。

pkgadd コマンドを使って署名付きパッケージを追加する手順については、250 ページの「署名付きパッケージの追加と削除 (作業マップ)」を参照してください。

prodreg コマンドの拡張機能

Solaris 9 4/03 – prodreg コマンドのオプションを使って、コマンド行から Solaris Product Registry にアクセスし、管理できます。

prodreg コマンドを使ってソフトウェアパッケージを管理する方法については、231 ページの「Solaris Product Registry のコマンド行インタフェースによるソフトウェアの管理 (作業マップ)」を参照してください。

Solaris 9 リリースの新しいソフトウェア 管理機能

この節では、Solaris 9 リリースの新しいソフトウェア管理機能について説明します。

署名付きパッチ

Solaris 2.6、7、8、9 リリースのすべてのパッチには、デジタル署名が付いています。有効なデジタル署名は、署名が適用された以降にパッチの変更が行われていないことを保証します。

署名付きパッチには、パッチがシステムに適用される前に確認できるデジタル署名が含まれているため、パッチのダウンロードやパッチの適用を安全に行うことができます。

署名付きパッチは、Java™ アーカイブ形式のファイル (*abc.jar*) 内に格納され、SunSolve OnlineSM から入手できます。

`smpatch` コマンドを使って署名付きパッチを追加する方法については、『*Signed Patches Administration Guide for PatchPro 2.2*』(英語版)の「*Managing Signed Patches by Using Solaris Patch Management Tools (Tasks)*」を参照してください。

Solaris Product Registry 3.0

Solaris Product Registry 3.0 は、ソフトウェアパッケージのインストールとアンインストールを行うための GUI ツールです。

この製品を使ってソフトウェアパッケージを管理する方法については、[226 ページ](#)の「*Solaris Product Registry の GUI によるソフトウェアの管理 (作業マップ)*」を参照してください。

パッチアナライザ

Solaris™ Web Start プログラムを使用して Solaris 9 Update リリースにアップグレードするときに、パッチアナライザを使用すると、システムの分析が行われて、Solaris Update リリースへのアップグレードによってどのパッチ (存在する場合は) が削除またはダウングレードされるかが確認されます。Solaris 9 リリースにアップグレードするときは、パッチアナライザを使用する必要はありません。

Solaris 9 Update リリースへのアップグレード時にこのツールを使用する方法については、『*Solaris 9 9/04 インストールガイド*』の「*Solaris Update リリースへのアップグレード(作業)*」を参照してください。

Solaris 管理コンソールのパッチマネージャ

Solaris 管理コンソールには、パッチを管理するための新しいパッチツールが用意されています。このパッチツールを使用するだけで、Solaris 9 リリースを実行しているシステムにパッチを追加できます。

Solaris 管理コンソールの起動方法については、42 ページの「スーパーユーザーまたは役割としてコンソールを起動する方法」を参照してください。

ソフトウェア管理作業についての参照先

ソフトウェアを管理する手順については、次の表を参照してください。

ソフトウェア管理作業	参照先
Solaris ソフトウェアのインストール	『Solaris 9 9/04 インストールガイド』
インストール後の Solaris ソフトウェアパッケージの追加または削除	第 14 章
インストール後の Solaris パッチの追加または削除	第 16 章
ソフトウェアパッケージ問題の障害追跡	『Solaris のシステム管理 (上級編)』の「ソフトウェアパッケージで発生する問題の解決 (手順)」

ソフトウェアパッケージの概要

ソフトウェア管理には、ソフトウェア製品のインストールと削除が含まれます。Sun および Sun 以外のベンダーは、「パッケージ」という形式でソフトウェア製品を提供しています。

「パッケージング」という用語は一般に、ソフトウェア製品が使用されるシステムに、その製品を配布してインストールする方法を指します。パッケージは、定義済みフォーマットによるファイルとディレクトリの集まりです。このフォーマットは、アプリケーションバイナリインタフェース (ABI) に準拠します。ABI は、System V インタフェース定義を補足するものです。Solaris オペレーティング環境には、このフォーマットを解釈し、パッケージをインストールまたは削除したり、パッケージインストールを検査したりする方法を提供する 1 組のユーティリティがあります。

パッチは、既存のソフトウェアの正常な実行の妨げとなっているファイルとディレクトリを置換または更新するためのファイルとディレクトリの集まりです。パッチの詳細については、第 15 章を参照してください。

署名付きパッケージおよびパッチ

パッケージにはデジタル署名を含めることができます。有効なデジタル署名付きのパッケージは、署名が適用された以降はパッケージの変更が行われていないことを保証します。署名付きパッケージを使用すると、パッケージをシステムに追加する前にデジタル署名を検証できるため、パッケージを安全にダウンロードまたは追加できます。

同じことが、署名付きパッチにも言えます。有効なデジタル署名付きのパッチは、署名が適用された以降はパッチの変更が行われていないことを保証します。署名付きパッチを使用すると、パッチをシステムに追加する前にデジタル署名を検証できるため、パッチを安全にダウンロードまたは追加できます。

署名付きパッチをシステムに追加する方法については、277 ページの「[patchadd コマンドによる署名付きパッチの追加 \(作業マップ\)](#)」を参照してください。

署名付きパッケージの作成方法については、『*Application Packaging Developer's Guide* (英語版)』を参照してください。

署名付きパッケージは、デジタル署名が含まれる点以外は、署名なしパッケージとまったく同一です。署名付きパッケージのインストール、照会、または削除は、既存の Solaris パッケージングツールを使って行えます。また、署名付きパッケージと署名なしパッケージは、バイナリレベルでも互換性があります。

デジタル署名されたパッケージまたはパッチをシステムに追加するには、信頼される証明書を使ってパッケージキーストアを設定しておく必要があります。これらの証明書は、パッケージやパッチ上のデジタル署名の有効性を確認する際に使用されます。

次の表は、署名付きパッケージおよびパッチに関する一般的な用語の一覧です。

用語	定義
キーストア	証明書と鍵を格納するためのリポジトリ。それらの情報が必要になると、このリポジトリが検索される。 <ul style="list-style-type: none">■ Java キーストア - Solaris リリースでデフォルトでインストールされる証明書のリポジトリ。 Java キーストアは通常、<code>/usr/j2se/jre/lib/security</code> ディレクトリに格納される。■ パッケージキーストア - 署名付きパッケージおよびパッチをシステムに追加する際にインポートする証明書用のリポジトリ。 パッケージキーストアは、デフォルトで <code>/var/sadm/security</code> ディレクトリに格納される。

用語	定義
信頼される証明書	<p>別のエンティティに属する公開鍵を備えた証明書。「信頼される証明書」という呼び名は、証明書に含まれている公開鍵が、その証明書のサブジェクトまたは所有者によって示された本人のものであることを、キーストアの所有者が信頼することに由来している。この信頼を表明するために、証明書の発行者はその証明書に署名する。</p> <p>信頼される証明書は、署名を検証する際やセキュリティ保護されたサーバー (SSL サーバー) への接続を確立する際に使用される。</p>
ユーザー鍵	<p>暗号鍵に関する機密情報を保持する。この情報は、不正なアクセスを防ぐために、セキュリティが施された形式で格納される。ユーザー鍵は、ユーザーの非公開鍵と対応する公開鍵証明書から構成される。</p>

署名付きパッケージまたはパッチをシステムに追加する手順は、基本的に次の3つから構成される。

1. pkgadm コマンドを使ってシステムのパッケージキーストアに証明書を追加する
2. (省略可能) pkgadm コマンドを使って証明書を一覧表示する
3. pkgadd コマンドを使ってパッケージを追加するか、patchadd コマンドを使ってパッチを追加する

署名付きパッケージをシステムに追加する手順については、250 ページの「署名付きパッケージの追加と削除 (作業マップ)」を参照してください。署名付きパッチをシステムに追加する手順については、277 ページの「patchadd コマンドによる署名付きパッチの追加 (作業マップ)」を参照してください。

Sun の証明書による署名付きパッケージおよびパッチの検証

「ストリーム形式」の SVR4 署名付きのパッケージまたはパッチには、PEM でコード化された PKCS7 署名が埋め込まれています。この署名には少なくとも、パッケージまたはパッチの暗号化されたダイジェストと署名者の X.509 公開鍵証明書とが含まれています。また、パッケージまたはパッチには「証明書チェーン」を含めることもできます。証明書チェーンを使うと、署名者の証明書からローカルに格納されている信頼される証明書へ、信頼関係のチェーンを形成することができます。

PEM でコード化された PKCS7 署名を使って、次のことを検証します。

- そのパッケージに署名したエンティティがそのパッケージの作成者である
- そのエンティティが実際にそのパッケージに署名した
- そのパッケージがエンティティによる署名後に変更されていない
- そのパッケージに署名したエンティティが信頼されたエンティティである

次の表は、署名付きパッケージおよびパッチに関する暗号化関連の用語の一覧です。

用語	定義
ASN.1	ASN.1 (Abstract Syntax Notation 1) は、一連の抽象オブジェクトを表現するための方式の 1 つである。たとえば、ASN.1 では、公開鍵証明書、公開鍵証明書を構成するすべてのオブジェクト、オブジェクトの収集順序などを定義している。ただし、ASN.1 では、オブジェクトを保存用または転送用にシリアライズする方法は定義していない。
base64	base64 は、任意のバイナリデータを ASCII テキストとしてコード化する方式の 1 つである。
DER	DER (Distinguished Encoding Rules) は、ASN.1 オブジェクトのバイナリ表現の 1 つである。DER では、コンピューティング環境で ASN.1 オブジェクトを保存用または転送用にシリアライズする方法を定義している。
PEM	PEM (Privacy Enhanced Message) は、base64 エンコーディングといくつかの任意のヘッダーを使って、ファイルを (DER またはその他のバイナリ形式で) コード化する方式である。もともとは、MIME タイプの電子メールメッセージをコード化するために使用された。また、PEM は、証明書と非公開鍵をファイルシステム上または電子メールメッセージ内のファイルにコード化する際にも広く使用される。
PKCS7	PKCS7 (Public Key Cryptography Standard #7) は、デジタル署名やデジタルエンベロープなどの暗号化データに対する汎用的な構文を定めた規格である。
X.509	ITU-T (International Telecommunication Union-Telcom) 勧告 X.509 は、広く採用されている X.509 公開鍵証明書の構文を定めた仕様である。 この勧告では、認証サービスを提供するためのフレームワークが定義されている。X.509 で定義された 2 つの認証レベルを次に示す。 <ul style="list-style-type: none"> ■ 単純な認証 – パスワードを使ってユーザーの身元を検証する方式 ■ 強い認証 – 暗号化技術を使って作成された資格情報に基づく方式。単純な認証でも不正なアクセスに対する多少の備えにはなるが、安全なサービスの提供基盤として使えるのは、強度の高い認証のみである。

Sun Microsystems から発行された認証済みデジタル署名を使って、ダウンロードされたパッケージやパッチが改ざんされていないことを検証できます。こうした証明書は、システムのキーストアにインポートされます。

Sun のすべての証明書は、先頃 GTE CyberTrust を買収した Baltimore Technologies によって発行されています。

Sun の証明書をインポートするときに特別なパスワードを使用することで、キーストアへのアクセスを保護できます。

pkgadm listcert コマンドを使えば、ローカルのパッケージキーストア内に格納されている証明書の情報を表示できます。たとえば、次のようになります。

```
# pkgadm listcert -P pass:store-pass
キーストア別名: GTE CyberTrust Root
共通名: GTE CyberTrust Root
証明書の種類: Trusted Certificate
発行者の共通名: GTE CyberTrust Root
有効な日付: <Feb 23 23:01:00 1996 GMT> - <Feb 23 23:59:00 2006 GMT>
MD5 フィンガープリント: C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58
SHA1 フィンガープリント: 90:DE:DE:9E:4C:4E:9F:6F:D8:86:17:57:9D:D3:91:BC:65:A6...
```

次の表は、pkgadm listcert コマンドの出力を一覧にまとめたものです。

フィールド	説明
キーストア別名	印刷、署名、削除のいずれかの目的で証明書を取得する場合、この名前を証明書の参照名として使用する必要がある。
共通名	証明書の共通の名前。信頼される証明書の場合、この名前はキーストア別名と同じになる。
証明書の種類	次の2つの種類がある。 <ul style="list-style-type: none"> ■ Trusted Certificate (信頼される証明書) - ほかの証明書を検証する際の信頼基盤として使用可能な証明書。信頼される証明書には非公開鍵は関連付けられない。 ■ Signing Certificate (署名証明書) - パッケージまたはパッチへの署名用として使用可能な証明書。署名証明書には非公開鍵が関連付けられる。
発行者の共通名	この証明書の署名、発行をしたエンティティの名前。信頼される証明書発行局 (CA, Certificate Authority) が発行した証明書の場合、発行者の共通名は、共通名と同じになる。
有効な日付	証明書の有効な日付。
MD5 フィンガープリント	証明書の MD5 ダイジェスト。このダイジェストを使えば、その証明書が証明元からの転送中に改変されなかったかどうかを検証できる。
SHA1 フィンガープリント	MD5 フィンガープリントに似ているが、異なるアルゴリズムを使って計算される。

証明書を認証するには、各証明書の MD5 ハッシュおよび SHA1 ハッシュ(「フィンガープリント」とも呼ばれる)を、発行者によって公開されている既知の正しいフィンガープリントと比較します。

SunSolve Online の信頼される証明書

SunSolve Online は、以下の証明書を使って署名付きパッチ上のデジタル署名を検証します。なお、その際、PatchPro を含む以前の Solaris パッチ管理ツール (smpatch コマンド) を使います。

- ルート CA 証明書と呼ばれるトップレベルの証明書
- Class B CA 証明書と呼ばれるサブ CA (Sun Microsystems Inc.) による証明書
- Sun Enterprise™ Services によって発行される、「パッチ管理証明書」と呼ばれる証明書

「証明書発行局」は、パッチ上のデジタル署名の復号時に使用される公開鍵と、こうした公開鍵の所有者の関係を認証します。

Sun ルート CA 証明書、Sun Class B CA 証明書、およびパッチ署名証明書は、PatchPro を含む Solaris パッチ管理ツールに含まれています。この 3 つの証明書により、Sun のルート CA 証明書から Class B CA 証明書、Class B CA 証明書からパッチ管理証明書にさかのぼってパッチの信頼性を検証する認証チェーンが構成されます。最終的に、Sun のルート CA は、GTE CyberTrust CA によって認証されます。

Sun の信頼される証明書のインポート

署名付きパッケージおよびパッチを追加するための Sun の信頼される証明書を取得するには、次の方法を使います。

- **Java** キーストア – Sun のルート CA 証明書をインポートします。この証明書は、この Solaris リリースをインストールする際に Java キーストア内にデフォルトで格納されます。
- **Sun の PKI (Public Key Infrastructure)** サイト – システム上に Java キーストアが存在していない場合は、以下のサイトから証明書をインポートできます。
<https://ra.sun.com:11005/>
- **PatchPro** のキーストア – smpatch コマンドを使って署名付きパッチを追加するための PatchPro がすでにインストールされている場合、Java キーストアから Sun のルート CA 証明書をインポートできます。

パッケージキーストアのセットアップ

以前の Solaris リリースでは、パッチ管理ツールをダウンロードし、keytool コマンドを使って証明書をインポートすることで、PatchPro が使用する Java キーストアを作成することができました。

証明書の格納された Java キーストアがシステム上にすでに存在している場合、Sun Microsystems のルート CA 証明書を Java キーストアから `keytool` コマンドを使ってエクスポートできます。続いて、その証明書を、`pkgadm` コマンドを使ってパッケージキーストア内にインポートします。

ルート CA 証明書のパッケージキーストアへのインポートが完了すると、`pkgadd` コマンドや `patchadd` コマンドを使って署名付きパッケージおよびパッチをシステムに追加できるようになります。

注 – なお、Sun Microsystems のルート証明書が必要となるのは、Sun が署名したパッチやパッケージを追加する場合のみです。

証明書をパッケージキーストアにインポートする手順については、250 ページの「信頼される証明書をパッケージキーストアにインポートする方法 (`pkgadm addcert`)」を参照してください。

`pkgadd` コマンドを使って署名付きパッケージを追加するための完全な手順については、250 ページの「署名付きパッケージの追加と削除 (作業マップ)」を参照してください。

ソフトウェアパッケージを管理するためのツール

Solaris リリースをシステムにインストールした後でソフトウェアパッケージをシステムに追加したり、システムから削除するためのツールは、次のとおりです。

表 13-1 ソフトウェアパッケージツール

ソフトウェアパッケージ情報の追加、削除、表示に使用するツール	追加機能
Solaris Web Start プログラム	インストーラを起動して、Solaris 9 メディアパックに含まれている製品を追加する。 個々のソフトウェアパッケージを追加することはできない

表 13-1 ソフトウェアパッケージツール (続き)

ソフトウェアパッケージ情報の追加、削除、表示に使用するツール	追加機能
Solaris Product Registry (GUI)	インストーラを起動して、ソフトウェア製品情報を追加、削除、または表示する。Solaris Web Start プログラムや Solaris pkgadd コマンドを使って初めからインストールされているソフトウェア製品に関する情報を削除または表示する場合は、Solaris Product Registry を使用する
Solaris Product Registry prodreg ビューア (コマンド行インタフェース)	prodreg コマンドでは、Solaris Web Start プログラムや Solaris pkgadd コマンドを使ってインストールされているソフトウェア製品に関する情報を削除または表示できる
パッケージコマンド (pkgadd、pkgrm、pkginfo)	これらのコマンドをスクリプトに組み込む、オプションのファイルを設定してユーザーの対話操作を省略したり特別なチェック作業を追加したりする、ソフトウェアパッケージをスプールディレクトリにコピーする

ソフトウェアパッケージの追加または削除 (pkgadd)

表 13-1 に記載されているソフトウェア管理ツールはすべて、インストール済みソフトウェアに関する情報の追加、削除、または照会に使用できます。Admintool、Solaris Product Registry の prodreg ビューア、および Web Start プログラムはいずれも、Solaris Product Registry に格納されているインストールデータにアクセスします。pkgadd コマンドや pkgrm コマンドといったパッケージツールでも、インストールデータにアクセスし、管理できます。

パッケージを追加する際、pkgadd コマンドは、ファイルを解凍して、インストール用メディアからローカルシステムのディスクにコピーします。パッケージを削除する際、pkgrm コマンドは、そのパッケージに関連するファイルが他のパッケージと共有されている場合を除いて、それらをすべて削除します。

パッケージファイルはパッケージ形式で配布され、配布されたままの状態では使用できません。pkgadd コマンドは、ソフトウェアパッケージの制御ファイルを解釈してから、製品ファイルを解凍して、システムのローカルディスクにインストールします。

pkgadd と pkgrm の各コマンドは、標準の場所にそのログ出力を記録しませんが、インストールまたは削除される製品を常時監視しています。pkgadd と pkgrm の各コマンドは、インストールまたは削除されたパッケージに関する情報をソフトウェア製品用データベースに格納します。

このデータベースを更新することによって、pkgadd と pkgrm の各コマンドは、システムにインストールされているすべてのソフトウェア製品の記録を保持します。

ソフトウェアパッケージの追加にあたっての重要な注意点 (pkgadd)

使用中のシステムでパッケージのインストールまたは削除を行う場合は、次の点に注意する必要があります。

- パッケージの命名規則 – Sun パッケージは、SUNWaccr、SUNWadmap、SUNWcsu などのように、必ず接頭辞 SUNW で始まります。Sun 以外のパッケージは、通常、その会社を表す接頭辞で始まります。
- インストール済みのソフトウェア – Web Start プログラム、Solaris Product Registry の prodreg ビューア (GUI または CLI)、Admintool、または pkginfo コマンドを使って、システムにインストールされているソフトウェアを特定できません。
- サーバーとクライアントによるソフトウェアの共有の状態 – クライアントのソフトウェアは、一部がサーバーに、一部がクライアントに置かれる場合があります。このような場合、クライアントにソフトウェアを追加するには、サーバーとクライアントの両方にパッケージを追加する必要があります。

パッケージの削除に関するガイドライン (pkgrm)

パッケージを削除するときは、rm コマンドを代わりに使いたい場合でも、これらのツールのいずれかを使用する必要があります。たとえば、rm コマンドを使用すると、バイナリ実行可能ファイルを削除することができますが、これは pkgrm コマンドを使用してバイナリ実行可能ファイルを含むソフトウェアパッケージを削除する場合は異なります。rm コマンドを使用してパッケージのファイルを削除すると、ソフトウェア製品用データベースが破壊されます。1つのファイルだけを削除したい場合は、removef コマンドを使用します。このコマンドは、該当ファイルがパッケージから削除されるように、ソフトウェア製品データベースを正しく更新します。詳細については、removef (1M) のマニュアルページを参照してください。

複数のバージョンのパッケージ (たとえば、複数バージョンの文書処理アプリケーション) をインストールしておきたい場合は、`pkgadd` コマンドを使用して、すでにインストール済みのパッケージとは異なるディレクトリに新しいバージョンをインストールしてください。パッケージがインストールされているディレクトリは、ベースディレクトリと呼ばれます。ベースディレクトリは、管理ファイルと呼ばれる特殊ファイルに `basedir` キーワードを設定することによって操作できます。管理ファイルの使用法とベースディレクトリの設定方法については、221 ページの「パッケージ追加時のユーザーの対話操作を省略する (`pkgadd`)」と、`admin(4)` のマニュアルページを参照してください。

注 - Solaris ソフトウェアをインストールするときにアップグレードオプションを使用すると、Solaris インストール用ソフトウェアは、ソフトウェア製品用データベースを調べて、すでにシステムにインストールされている製品があるかどうかを確認します。

パッケージ追加時のユーザーの対話操作を省略する (`pkgadd`)

管理ファイルの使用

`pkgadd -a` コマンドを使用すると、特殊な管理ファイルを調べて、インストールの進め方についての情報を入手できます。通常、`pkgadd` コマンドはいくつかのチェックを行い、指定されたパッケージを実際に追加する前に、プロンプトを表示してユーザーに確認します。ただし、管理ファイルを作成すれば、このようなチェックを省略して、ユーザーの確認なしでパッケージをインストールするように `pkgadd` コマンドに指示できます。

デフォルトでは、`pkgadd` コマンドは現在の作業用ディレクトリに管理ファイルがないか調べます。現在の作業用ディレクトリの中に管理ファイルを見つけることができなかつた場合、`pkgadd` コマンドは `/var/sadm/install/admin` ディレクトリに指定の管理ファイルがないか調べます。また、`pkgadd` コマンドには管理ファイルへの絶対パスも使用できます。



注意 - 管理ファイルは注意して使用してください。管理ファイルを使用して、`pkgadd` が通常実行するチェックとプロンプトを省略する場合は、事前にパッケージのファイルがインストールされている場所や、パッケージのインストール用スクリプトがどのように実行されるのかを知っておく必要があります。

次の例は、pkgadd コマンドがパッケージのインストール前にユーザーに確認のプロンプトを表示しないようにするための管理ファイルを示しています。

```
mail=
instance=overwrite
partial=nocheck
runlevel=nocheck
idepend=nocheck
rdepend=nocheck
space=nocheck
setuid=nocheck
conflict=nocheck
action=nocheck
networktimeout=60
networkretries=3
authentication=quit
keystore=/var/sadm/security
proxy=
basedir=default
```

パッケージを追加するときにユーザーの対話操作を省略する以外にも、いろいろな目的で管理ファイルを使用できます。たとえば、管理ファイルを使用すれば、エラーが発生した場合に(ユーザーの対話操作なしに)パッケージのインストールを終了したり、pkgrm コマンドでパッケージを削除する場合に対話操作を省略できます。

また、特別なインストールディレクトリをパッケージに割り当てることもできます。この方法は、1つのシステム上で複数のバージョンのパッケージを管理する場合に役に立ちます。これを行うには、パッケージがインストールされる場所を示す代替ベースディレクトリを管理ファイル内に設定します(basedir キーワードを使用)。詳細については、admin(4) のマニュアルページを参照してください。

応答ファイルの使用 (pkgadd)

応答ファイルには、「対話型パッケージ」で尋ねられる特定の質問に対するユーザーの応答が格納されます。対話型パッケージには、パッケージをインストールする前にいくつかの質問(たとえば、パッケージのオプションをインストールするかどうかなど)をユーザーに尋ねる request スクリプトが格納されています。

インストールするパッケージが対話型パッケージであることがインストール前にわかっていて、さらにそのパッケージをこの先インストールするときにユーザーの対話操作を省略できるように応答を格納しておきたい場合は、pkgask コマンドを使用してユーザーの応答を保存できます。このコマンドの詳細については、pkgask(1M) のマニュアルページを参照してください。

request スクリプトが尋ねる質問への応答を格納した後は、pkgadd -r コマンドを使用して、ユーザーの対話操作なしにパッケージをインストールすることができます。

第 14 章

ソフトウェアの管理 (手順)

この章では、ソフトウェアパッケージを追加、検証、削除する方法について説明します。

ソフトウェアの管理作業に関連する手順については、次の節を参照してください。

- 225 ページの「Solaris Web Start プログラムを使ってソフトウェアをインストールする方法」
- 226 ページの「Solaris Product Registry の GUI によるソフトウェアの管理 (作業マップ)」
- 231 ページの「Solaris Product Registry のコマンド行インタフェースによるソフトウェアの管理 (作業マップ)」
- 250 ページの「署名付きパッケージの追加と削除 (作業マップ)」
- 256 ページの「パッケージコマンドによるソフトウェアパッケージの管理 (作業マップ)」
- 265 ページの「Admintool によるソフトウェアパッケージの追加と削除 (作業マップ)」

ソフトウェアパッケージを管理するためのコマンド

次の表に、Solaris のインストール後に、ソフトウェアパッケージの追加、削除、検査に使用するコマンドを示します。

表 14-1 ソフトウェアパッケージを管理するためのツールまたはコマンド

ツールまたはコマンド	マニュアルページ	説明
admintool	admintool (1M)	グラフィカルツールを使用してソフトウェアパッケージをインストールまたは削除する
installer	installer (1M)	インストーラを使用してソフトウェアパッケージをインストールまたは削除する
pkgadd	pkgadd (1M)	署名付きまたは署名なしのソフトウェアパッケージをインストールする
pkgadm	pkgadm (1M)	署名付きパッケージおよびパッチの管理に使用する鍵と証明書を保守する
pkgchk	pkgchk (1M)	ソフトウェアパッケージのインストールを検査する
pkginfo	pkginfo (1)	ソフトウェアパッケージ情報を表示する
pkgparam	pkgparam (1)	ソフトウェアパッケージのパラメータ値を表示する
pkgrm	pkgrm (1M)	ソフトウェアパッケージを削除する
prodreg	prodreg (1M)	Solaris Product Registry 内のソフトウェアをブラウズ、登録解除、およびアンインストールする
pkgtrans	pkgtrans (1)	インストール可能なパッケージをある形式から別の形式に変換する。-g オプションを指定して pkgtrans コマンドを実行すると、署名が生成され、該当するデータストリーム内に格納される

Solaris Web Start プログラムによるソフトウェアの追加

この節では、Solaris オペレーティングシステムがすでにインストールされているシステムに、Solaris Web Start プログラムを使ってソフトウェアを追加する方法を説明します。Solaris Web Start プログラムでは、Solaris オペレーティングシステムを最初にインストールしたときに、インストール対象から外したソフトウェアグループのコンポーネントだけをインストールします。インストールまたはアップグレード後に別のソフトウェアグループのアップグレードを行うことはできません。4つのソフトウェアグループについては、『Solaris 9 9/04 インストールガイド』を参照してください。

▼ Solaris Web Start プログラムを使ってソフトウェアをインストールする方法

注 – この手順では、システムがボリューム管理 (vol1d) を実行しているものとします。システムがボリューム管理を実行していない場合、ボリューム管理を使用しないでリムーバブルメディアにアクセスする方法については、『Solaris のシステム管理 (デバイスとファイルシステム)』の「リムーバブルメディアへのアクセス (手順)」を参照してください。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. インストールを **CD**、**DVD**、またはネットワークのいずれから行うかを決めます。次のいずれかの手順に従います。
 - CD からインストールする場合は、CD-ROM ドライブに CD を挿入する。
Solaris 9 LANGUAGES CD を挿入すると、Solaris Web Start プログラムが自動的に起動されます。手順 6 へ進んでください。
 - DVD からインストールする場合は、DVD-ROM ドライブに DVD を挿入する。
 - ネットワークからインストールする場合は、インストールするソフトウェアのネットイメージを見つける。
3. **Solaris Web Start** インストーラが見つかるまで、ディレクトリを変更します。
Solaris Web Start インストーラは、CD や DVD 上のさまざまなディレクトリにあります。
 - Solaris 9 Software 1 of 2 CD
 - Solaris 9 Software 2 of 2 CD
 - Solaris 9 Documentation CD
 - Solaris 9 Languages CD。この CD を挿入すると、Solaris Web Start プログラムが自動的に起動されるCD や DVD の構成については、『Solaris 9 9/04 インストールガイド』の「Solaris 9 メディアの構成 (リファレンス)」を参照してください。
4. 指示に従って、ソフトウェアをインストールします。
 - ファイルマネージャから、「インストール (Installer)」をダブルクリックする。
 - コマンド行から、次のように入力する。

```
% ./installer [options]
```

```
-nodisplay
```

GUI を使用しないでインストーラを実行する。

-noconsole

対話式のテキストコンソールデバイスを使用しないで実行する。ソフトウェアをインストールするための UNIX スクリプトに `installer` コマンドを組み込んだときは、このオプションを `-nodisplay` オプションと合わせて使用する。

指示に従って、ソフトウェアをインストールします。

5. 「インストール (**Installer**)」をダブルクリックします。
「インストール (**Installer**)」ウィンドウが表示され、続いて「Solaris Web Start」ダイアログボックスが表示されます。
6. 画面の指示に従ってソフトウェアをインストールします。
7. ソフトウェアの追加が終わったら、「終了 (**Exit**)」をクリックします。
Solaris Web Start プログラムが終了します。

Solaris Product Registry の GUI によるソフトウェアの管理 (作業マップ)

次の作業マップに、Solaris Product Registry を使って行うことができるソフトウェアの管理作業を示します。

作業	説明	参照先
インストールまたはアンインストールしたソフトウェアを Solaris Product Registry を使って表示する	インストールまたはアンインストールしたソフトウェアを Solaris Product Registry を使って表示できる	228 ページの「インストールまたはアンインストールしたソフトウェアの情報を Solaris Product Registry の GUI を使って表示する方法」
Solaris Product Registry を使ってソフトウェアをインストールする	Solaris Product Registry を使ってソフトウェアを検索し、Solaris Web Start プログラムを起動できる。このプログラムの指示に従って、ソフトウェアのインストールを行う	229 ページの「Solaris Product Registry の GUI を使ってソフトウェアをインストールする方法」

作業	説明	参照先
Solaris Product Registry を使ってソフトウェアをアンインストールする	Solaris Product Registry を使ってソフトウェアをアンインストールできる	230 ページの「Solaris Product Registry の GUI を使ってソフトウェアをアンインストールする方法」

Solaris Product Registry は、インストールされているソフトウェアをユーザーが管理するためのツールです。ソフトウェアのインストールが終わると、Solaris Product Registry は、Solaris Web Start プログラム 3.0 または Solaris pkgadd コマンドを使用して、インストールされたすべてのソフトウェアの一覧を表示します。

Solaris Product Registry は、GUI またはコマンド行インタフェース (CLI) で使用できます。Solaris Product Registry CLI の使用方法については、231 ページの「Solaris Product Registry のコマンド行インタフェースによるソフトウェアの管理 (作業マップ)」を参照してください。

Solaris Product Registry の GUI インタフェースでは、次の作業を実行できます。

- インストールおよび登録されているソフトウェアとソフトウェア属性の一覧表示
- System Software Localizations ディレクトリにインストールされている地域対応版のすべての Solaris システム製品の表示
- インストーラの選択と起動
- 追加のソフトウェア製品のインストール
- ソフトウェアや個々のソフトウェアパッケージのアンインストール

Solaris Product Registry の GUI のメインウィンドウは、次の 3 つの情報領域から構成されています。

- インストール、登録、削除されたソフトウェア
- 選択されているソフトウェアの標準属性
- カスタマイズされた属性と登録されているソフトウェアの内部属性

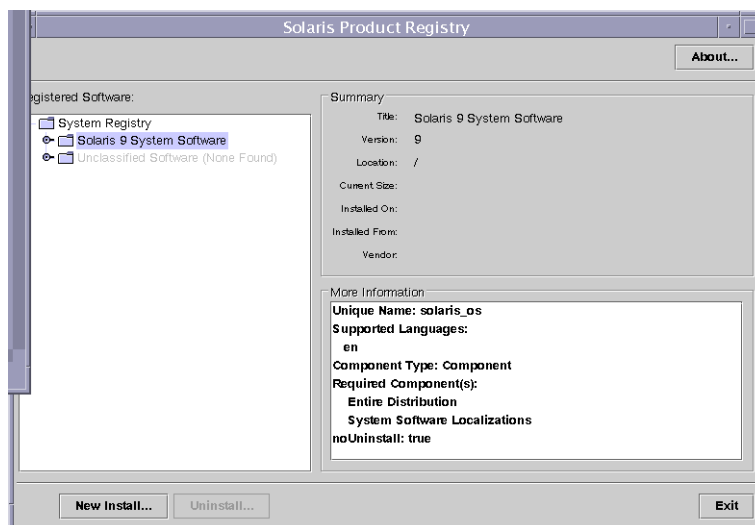


図 14-1 「Solaris Product Registry」 ウィンドウ

▼ インストールまたはアンインストールしたソフトウェアの情報を Solaris Product Registry の GUI を使って表示する方法

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Solaris Product Registry** ツールを起動します。

```
# prodreg &
```

「Solaris Product Registry」メインウィンドウが表示されます。

3. 「登録されているソフトウェア (**Registered Software**)」ボックスの「システムレジストリ (**System registry**)」ディレクトリの左側にあるアイコンをクリックします。

右側を指していたアイコンが下側が変わります。Solaris Product Registry の項目は、左側にテキストファイルアイコンがあるものを除き、どの項目も展開または縮小できます。

「登録されているソフトウェア (Registered Software)」ボックスにインストールされているソフトウェアには、必ず次のものが含まれます。

- Solaris のインストール時に選択した構成ソフトウェアグループ。ソフトウェアグループは、「コアシステムサポート (Core System Support)」、「エンドユーザーシステムサポート (End User System Support)」、「開発者システムサポート (Developer System Support)」、「全体ディストリビューション (Entire

Distribution)」、「全体ディストリビューションと OEM サポート (Entire Distribution plus OEM Support)」のいずれかです。

- 追加のシステムソフトウェア (選択したソフトウェアグループに含まれていない Solaris 製品)。
 - 分類していないソフトウェア (pkgadd コマンドを使ってインストールした Solaris 製品以外のパッケージまたはソフトウェアグループに含まれていないパッケージ)。
4. 表示するソフトウェアアプリケーションが見つかるまでディレクトリを選択します。
ディレクトリを開くと、リストが展開されます。
 5. 属性を表示する場合は、ディレクトリまたはファイルを選択します。
属性情報が「一覧 (Summary)」ボックスと「詳細情報 (more information)」ボックスに表示されます。
 - Solaris Web Start プログラムでインストールしたソフトウェア製品の場合、Solaris Product Registry には以下の値が必ず表示されます。「タイトル (Title)」、「バージョン (Version)」、「インストール場所 (Location)」、および「インストール日時 (Installed on)」。製品またはソフトウェアグループの下で展開されているリスト内の項目の場合、その製品のバージョン情報が表示されます。
 - pkgrm コマンドを使って製品のすべてまたは一部を削除した場合は、ソフトウェア製品の名前の横にそれを警告するアイコンが表示されます。

▼ Solaris Product Registry の GUI を使ってソフトウェアをインストールする方法

Solaris Product Registry を使用すると、ソフトウェアを検索し、ソフトウェアのインストール手順をガイドしてくれる Solaris Web Start プログラムを起動できます。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Solaris Product Registry** ツールを起動します。

```
# prodreg
```

「Solaris Product Registry」ウィンドウが表示されます。
3. インストールを **CD**、**DVD**、またはネットワークのいずれから行うかを決めます。次のいずれかの手順に従います。
 - CD からインストールする場合は、CD-ROM ドライブに CD を挿入する。
 - DVD からインストールする場合は、DVD-ROM ドライブに DVD を挿入する。
 - ネットワークからインストールする場合は、インストールするソフトウェアのネットイメージを見つける。

4. インストールおよび登録されているソフトウェアの一覧を表示するには、その左側にあるアイコンをクリックします。
5. 「**Solaris Product Registry**」ウィンドウの下にある「新規インストール (**New Install**)」ボタンをクリックします。
「インストーラの選択 (**Select Installer**)」ダイアログボックスが表示され、最初は /cdrom ディレクトリまたは現在使用中のディレクトリが選択されています。
6. **Solaris Web Start** プログラムのインストーラが見つかるまで、ディレクトリを選択します。
Solaris Web Start インストーラは、CD や DVD 上のさまざまなディレクトリにあります。CD や DVD の構成については、『Solaris 9 9/04 インストールガイド』の「Solaris 9 メディアの構成 (リファレンス)」を参照してください。
 - Solaris 9 Software 1 of 2 と 2 of 2 CD
 - Solaris 9 Software 2 of 2 CD
 - Solaris 9 Documentation CD
 - Solaris 9 Languages CD。この CD を挿入すると、Solaris Web Start プログラムが自動的に起動される
7. 起動するインストーラを見つけたら、「ファイル (**Files**)」ボックスでそのインストーラの名前を選択します。
8. 「了解 (**OK**)」をクリックします。
選択したインストーラが起動されます。
9. インストーラによって表示される手順に従って、ソフトウェアをインストールします。

▼ Solaris Product Registry の GUI を使ってソフトウェアをアンインストールする方法

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Solaris Product Registry** ツールを起動します。

```
# prodreg
```

「Solaris Product Registry」ウィンドウが表示されます。
3. インストールおよび登録されているソフトウェアの一覧を表示するには、その左側にあるアイコンをクリックします。
4. アンインストールするソフトウェアの名前が見つかるまでディレクトリを選択します。
5. ソフトウェア属性を読んで、このソフトウェアがアンインストールするソフトウェアであることを確認します。

6. 「Solaris Product Registry」ウィンドウの下にある「アンインストール (Uninstall)」ボタンをクリックします。
 選択したソフトウェアがアンインストールされます。

Solaris Product Registry のコマンド行インタフェースによるソフトウェアの管理 (作業マップ)

次の作業マップに、Solaris Product Registry のコマンド行インタフェースを使って行うことができるソフトウェアの管理作業を示します。

作業	説明	参照先
インストールまたはアンインストールしたソフトウェアを prodreg を使って表示する	browse サブコマンドを使ってソフトウェア情報を表示できる	232 ページの「インストールまたはアンインストールしたソフトウェアの情報を表示する方法 (prodreg)」
prodreg を使ってソフトウェア属性を表示する	info サブコマンドを使って特定のソフトウェア属性を表示できる	235 ページの「ソフトウェア属性を表示する方法 (prodreg)」
prodreg コマンドを使ってソフトウェアコンポーネント間の依存関係を確認する	info サブコマンドを使って特定のソフトウェアコンポーネントに依存するコンポーネントを表示できる	237 ページの「ソフトウェアコンポーネント間の依存関係を確認する方法 (prodreg)」
損傷を受けたソフトウェア製品を prodreg を使って特定する	適切なアンインストーラを使用しないでインストール済みのソフトウェアファイルやパッケージを削除すると、システム上のソフトウェアが損傷を受ける可能性がある	238 ページの「損傷を受けたソフトウェア製品を特定する方法 (prodreg)」
prodreg を使ってソフトウェアをアンインストールする	uninstall サブコマンドを使ってソフトウェアをシステムから削除できる	241 ページの「ソフトウェアをアンインストールする方法 (prodreg)」
損傷を受けたソフトウェアを prodreg を使ってアンインストールする	ソフトウェアコンポーネントのアンインストーラをシステムから削除した場合は、損傷を受けたコンポーネントのアンインストールが失敗することがある	245 ページの「損傷を受けたソフトウェアをアンインストールする方法 (prodreg)」

作業	説明	参照先
損傷を受けたソフトウェアコンポーネントを <code>prodreg</code> を使って再インストールする	損傷を受けたソフトウェアコンポーネントに依存するソフトウェアがある場合は、損傷を受けたコンポーネントと依存ソフトウェアをアンインストールするのではなく、損傷を受けたコンポーネントを再インストールできる	248 ページの「 損傷を受けたソフトウェアコンポーネントを再インストールする方法 (prodreg) 」

`prodreg` コマンドは、Solaris Product Registry のコマンド行インタフェース (CLI) です。`prodreg` コマンドは、システム上のソフトウェアを管理するための複数のサブコマンドをサポートしています。

端末ウィンドウで `prodreg` コマンドを実行して、次の作業を行うことができます。

- インストールおよび登録されているソフトウェアとソフトウェア属性の一覧表示
- System Software Localizations ディレクトリにインストールされている地域対応版のすべての Solaris システム製品の表示
- 損傷を受けたソフトウェアの特定
- Solaris Product Registry のソフトウェアエントリの削除
- ソフトウェアや個々のソフトウェアパッケージのアンインストール

コマンド行インタフェースから Solaris Product Registry を管理する方法については、`prodreg (1M)` のマニュアルページを参照してください。

▼ インストールまたはアンインストールしたソフトウェアの情報を表示する方法 (`prodreg`)

Solaris Product Registry 内のソフトウェアに関する情報を確認するには、端末ウィンドウで `prodreg` コマンドのサブコマンド `browse` を実行します。

1. 端末ウィンドウを開きます。
2. **Solaris Product Registry** をブラウズします。

```
% prodreg browse
  BROWSE # +/-/. UUID                               # NAME
  =====
  1      -   root                                     1 System
                                                Registry
  2      +   a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1 Solaris 9 4/03
                                                System
                                                Software
  3      +   8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified
                                                Software
```


prodreg のサブコマンド browse では、登録済みソフトウェアに関する次の情報を表示できます。

BROWSE #	prodreg browse コマンドを実行すると、Solaris Product Registry により、登録済みの各ソフトウェアコンポーネントのブラウザ番号が生成されます。この番号を prodreg browse コマンドや prodreg info コマンドの引数に指定することにより、登録済みの特定のコンポーネントの下位コンポーネントを表示できます。 注 - ブラウズ番号 (BROWSE#) は、システムのリポートや再インストールによって変更される可能性があります。ブラウザ番号をスクリプト内に記述したり、その他のログインセッションで再利用したりすることはできません。
+/-/.	このフィールドは、ソフトウェアコンポーネントが Solaris Product Registry に登録された子コンポーネントを持っているかどうかを示します。このフィールドには、次の文字が表示されます。 <ul style="list-style-type: none">■ +- ソフトウェアコンポーネントは、現在表示されていないが子コンポーネントを持っていることを示します。■ -- ソフトウェアコンポーネントは、現在表示されている子コンポーネントを持っていることを示します。■ .- ソフトウェアコンポーネントが子コンポーネントを持っていないことを示します。
UUID	このフィールドには、Solaris Product Registry に格納されているソフトウェア固有の識別子が表示されます。
#	このフィールドには、システム上のソフトウェアコンポーネントのインスタンス番号が表示されます。システム上に複数のソフトウェアコンポーネントインスタンスがある場合、Solaris Product Registry はそれぞれのインスタンスに別々のインスタンス番号を割り当てます。
NAME	このフィールドには、ソフトウェアの地域対応化された名前が表示されます。この出力例では、Solaris オペレーティングシステムの名前は「Solaris 9 4/03」リリースになっています。

3. Solaris Product Registry 内の任意のソフトウェアコンポーネント 1 個の情報をブラウザします。

```
% prodreg browse -m "name"
```

```
-m name
```

name で名前を指定されたソフトウェアコンポーネントの情報を表示する

システム上に *name* ソフトウェアのインスタンスが複数存在する場合は、次のコマンドを実行して Solaris Product Registry をブラウズします。

```
% prodreg browse -u name-UUID -i instance
```

-u *name-UUID* 一意の識別子 *name-UUID* で指定された *name* ソフトウェアコンポーネントの情報を表示する

-i *instance* インスタンス番号 *instance* で指定された *name* ソフトウェアコンポーネントの情報を表示する

4. ブラウズするそれぞれのソフトウェアコンポーネントに対して、手順 3 を繰り返します。

例 — コンポーネント名によるソフトウェア情報の表示 (prodreg)

次の例では、コンポーネント名を指定してソフトウェア情報を表示します。

```
% prodreg browse
BROWSE # +/-/.  UUID                                     #  NAME
===== =====
1      -      root                                     1  System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b  1  Solaris 9 4/03
System
Software
3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b  1  Unclassified
Software
```

```
% prodreg browse -m "Solaris 9 4/03 System Software"
```

例 — コンポーネントのブラウズ番号によるソフトウェア情報の表示 (prodreg)

次の例では、`prodreg browse` コマンドの `-n` オプションでコンポーネントのブラウズ番号を指定して、ソフトウェアの情報を表示します。

```
% prodreg browse
BROWSE # +/-/.  UUID                                     #  NAME
===== =====
1      -      root                                     1  System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b  1  Solaris 9 4/03
System
Software
3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b  1  Unclassified
```

```
% prodreg browse -n 2
```

例 — コンポーネントの UUID によるソフトウェア情報の表示 (prodreg)

次の例では、`prodreg browse` コマンドの `-u` オプションでコンポーネントの UUID を指定して、ソフトウェアの情報を表示します。

```
% prodreg browse
BROWSE # +/-/.  UUID                                     #  NAME
=====  =====  =====
1      -      root                                     1  System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b  1  Solaris 9 4/03
System
Software
3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b  1  Unclassified
Software

% prodreg browse -u a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b
```

▼ ソフトウェア属性を表示する方法 (prodreg)

`prodreg info` コマンドを使って、特定のソフトウェア属性を表示できます。`prodreg info` コマンドは、登録済みソフトウェアに関するさまざまな情報を表示します。次に例を挙げます。

- ソフトウェアコンポーネントの名前
- ソフトウェアコンポーネントの説明
- ソフトウェアの必須コンポーネント
- このソフトウェアを必要とするその他のコンポーネント
- ソフトウェアのベースディレクトリ
- ソフトウェアコンポーネントのパス

1. 端末ウィンドウを開きます。
2. **Solaris Product Registry** をブラウズします。

```
% prodreg browse
BROWSE # +/-/.  UUID                                     #  NAME
=====  =====  =====
1      -      root                                     1  System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b  1  Solaris 9 4/03
System
Software
```

```
3          +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified
                                                    Software
```

3. **Solaris Product Registry** 内の任意のソフトウェアコンポーネントの属性を表示します。

`prodreg info` コマンドで出力された任意のソフトウェアコンポーネントの属性を表示できます。次のコマンドを使用します。

```
% prodreg info -m "name"
```

```
-m name
```

name で名前を指定されたソフトウェアコンポーネントの属性を表示する

4. 表示するそれぞれのソフトウェアコンポーネントに対して、[手順 3](#) を繰り返します。

例 — コンポーネント名によるソフトウェア属性の表示 (prodreg)

次の例では、コンポーネント名を指定してソフトウェア属性を表示します。

```
% prodreg browse
  BROWSE # +/-/.  UUID                                     #  NAME
  =====
  1      -      root                                     1  System
                                           Registry
  2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1  Solaris 9 4/03
                                           System
                                           Software
  3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1  Unclassified
                                           Software

% prodreg info -m "Solaris 9 4/03 System Software"
```

例 — コンポーネントのブラウザ番号によるソフトウェア属性の表示 (prodreg)

次の例では、`prodreg info` コマンドの `-n` オプションでコンポーネントのブラウザ番号を指定して、ソフトウェアの属性を表示します。

```
% prodreg browse
  BROWSE # +/-/.  UUID                                     #  NAME
  =====
  1      -      root                                     1  System
                                           Registry
  2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1  Solaris 9 4/03
                                           System
                                           Software
```

```
3          +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1  Unclassified
                                                Software
```

```
% prodreg info -n 2
```

例 — コンポーネントの UUID によるソフトウェア属性の表示 (prodreg)

次の例では、`prodreg info` コマンドの `-u` オプションでコンポーネントの UUID を指定して、ソフトウェア属性を表示します。

```
% prodreg browse
BROWSE # +/-/.  UUID                                     #  NAME
===== =====  =====
1      -      root                                     1  System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1  Solaris 9 4/03
System
Software
3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1  Unclassified
Software
```

```
% prodreg info -u a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b
```

▼ ソフトウェアコンポーネント間の依存関係を確認する方法 (prodreg)

特定のソフトウェアコンポーネントに依存するコンポーネントを表示するには、`prodreg info` コマンドを使用します。特定のコンポーネントをアンインストールする前に、ソフトウェア製品間の依存関係を確認できます。

1. 端末ウィンドウを開きます。
2. **Solaris Product Registry** をブラウズします。

```
% prodreg browse
BROWSE # +/-/.  UUID                                     #  NAME
===== =====  =====
1      -      root                                     1  System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1  Solaris 9 4/03
System
Software
3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1  Unclassified
Software
```

Solaris Product Registry に確認するソフトウェアコンポーネントが表示されるまで `prodreg browse` コマンドを繰り返し実行します。 `prodreg browse` コマンド

を使って Solaris Product Registry をブラウズする方法については、232 ページの「インストールまたはアンインストールしたソフトウェアの情報を表示する方法 (prodreg)」を参照してください。

3. 特定のソフトウェアコンポーネントの依存関係を表示します。

```
% prodreg info -m "name" -a "Dependent Components"
```

-m *name* *name* で名前を指定されたソフトウェアコンポーネントの属性を表示する

-a "Dependent Components" Dependent Components 属性の値を表示することにより、*name* ソフトウェアに依存するコンポーネントを表示する

このコマンドは、*name* ソフトウェアに依存するソフトウェアコンポーネントの一覧を出力します。

例 — その他のソフトウェア製品に依存するコンポーネントの表示 (prodreg)

次の例では、ExampleSoft というソフトウェア製品に依存するコンポーネントを一覧表示します。

```
% prodreg -m "ExampleSoft" -a "Dependent Components"
```

Dependent Components:

Name	UUID	#
-----	-----	-
ExampleSoftA	7f49ecvb-1ii2-11b2-a3f1-0800119u7e8e	1

▼ 損傷を受けたソフトウェア製品を特定する方法 (prodreg)

適切なアンインストーラを使用しないでインストール済みのソフトウェアファイルやパッケージを削除すると、システム上のソフトウェアが損傷を受ける可能性があります。損傷を受けたソフトウェアは、正しく機能しない場合があります。prodreg info コマンドを使って、ソフトウェア製品が損傷を受けていないかどうかを確認できます。

1. 確認するソフトウェアの Solaris Product Registry 情報を表示します。

```
% prodreg browse -m name
```

BROWSE #	+/-/.	UUID	#	NAME
=====	=====	=====	=	=====
1	-	root	1	System Registry
2	+	a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b	1	Solaris 9 8/03 System

3	+	8f64eabf-1dd2-11b2-a3f1-0800209a5b6b	1	Software Unclassified Software
4	-	<i>name-UUID</i>	1	<i>name</i>
233	.	<i>component-a-pkg</i>	1	<i>component-a</i>
234	.	<i>component-b-pkg</i>	1	

-m name *name* で名前を指定されたソフトウェアコンポーネントの情報を表示する

name-UUID *name* ソフトウェアコンポーネントのUUID

component-a-pkg *name* ソフトウェアに依存する *component-a* コンポーネントのパッケージ名

component-a *name* ソフトウェアに依存するコンポーネントの名前

component-b-pkg *name* ソフトウェアに依存する *component-b* コンポーネントのパッケージ名

前出の出力例では、Name 列に *component-b-pkg* エントリの名前が表示されていません。Solaris Product Registry にソフトウェアコンポーネント名が表示されない場合、そのコンポーネントは損傷を受けている可能性があります。

- ソフトウェアコンポーネントが損傷を受けていないかどうかを検証します。

```
% prodreg info -u name-UUID -i 1 -d
isDamaged=TRUE
```

-u name-UUID *name* ソフトウェアコンポーネントの情報を表示する

-i 1 *name* ソフトウェアコンポーネントの 1 番目のインスタンスの情報を表示する

-d *name* ソフトウェアコンポーネントの *isDamaged* 属性の値を表示する

isDamaged=TRUE と出力された場合、*name* ソフトウェアコンポーネントは損傷を受けています。

- name-UUID* ソフトウェアコンポーネントを構成するパッケージを特定します。

```
% prodreg info -u name-UUID -i 1 -a PKGS
pkgs:
component-a-pkg component-b-pkg
```

- これらのパッケージがシステムにインストールされていることを確認します。

```
% pkginfo component-a-pkg
application component-a-pkg component-a
```

```
% pkginfo component-b-pkg
ERROR: information on "component-b-pkg" was not found
```

pkginfo component-b-pkg コマンドのエラーメッセージは、component-b-pkg パッケージがシステムから削除されたことを示しています。name ソフトウェアコンポーネントは、component-b-pkg パッケージがないと正しく機能しない可能性があります。

例 — 損傷を受けたソフトウェアコンポーネントの特定 (prodreg)

次の例では、ExampleSoft ソフトウェアが損傷を受けていないかどうかを確認します。

```
% prodreg browse -m Examplesoft
BROWSE # +/-/.  UUID                               #  NAME
=====  =====  =====
1         -      root                               1  System
                                           Registry
2         +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1  Solaris 9 8/03
                                           System
                                           Software
3         +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1  Unclassified
                                           Software
4         -      95842091-725a-8501-ef29-0472985982be 1  ExampleSoft
233      .      90209809-9785-b89e-c821-0472985982be 1  Example Doc
234      .      EXSOzzt                             1
235      .      EXSOblob                             1  Example Data
```

Solaris Product Registry の NAME 列で、ExampleSoft の子コンポーネント EXSOzzt のエントリがありません。ExampleSoft ソフトウェアは損傷を受けている可能性があります。prodreg info コマンドに -u、-i、および -d オプションを指定して、ExampleSoft ソフトウェアが損傷を受けているかどうかを確認します。

```
% prodreg info -u 95842091-725a-8501-ef29-0472985982be -i 1 -d
isDamaged=TRUE
```

前出のコマンドの出力結果から、ExampleSoft ソフトウェアが損傷を受けていることがわかります。prodreg info コマンドの -a PKGS オプションを実行して、ExampleSoft ソフトウェアパッケージを特定します。

```
% prodreg info
-u 95842091-725a-8501-ef29-0472985982be
-i 1 -a PKGS
pkgs:
EXSOzzt EXSOblob
```

pkginfo コマンドを実行して、EXSOzzt および EXSOblob パッケージがシステムにインストールされていることを確認します。


```
% pkginfo EXSOzzt
ERROR: information for "EXSOzzt" was not found
```

```
% pkginfo EXSOblob
application EXSOblob      Example Data
```

pkginfo コマンドの出力結果から、EXSOzzt パッケージがシステムにインストールされていないことがわかります。

▼ ソフトウェアをアンインストールする方法 (prodreg)

prodreg uninstall コマンドを使って、システムからソフトウェアを削除できます。prodreg uninstall コマンドを使ってソフトウェアをアンインストールすると、指定のソフトウェアとそのソフトウェアに関連付けられている子コンポーネントがすべて削除されます。アンインストールするソフトウェアに依存するソフトウェアがないことを確認してから、ソフトウェアを削除してください。ソフトウェアの依存関係を確認する方法については、237 ページの「ソフトウェアコンポーネント間の依存関係を確認する方法 (prodreg)」を参照してください。

ソフトウェアコンポーネントのアンインストールが完了したら、prodreg unregister -r コマンドを使って、ソフトウェアそのものと、その子コンポーネント全部を Solaris Product Registry から削除できます。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. アンインストールするソフトウェアの情報を表示します。

```
# prodreg browse -u name-UUID
BROWSE # +/-/. UUID # NAME
===== =====
1 - root 1 System Registry
2 + a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1 Solaris 9 8/03 System Software
3 + 8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified Software
1423 - name-UUID 1 name
1436 . component-a-UUID 1 component-a
1437 - component-b-UUID 1 component-b
1462 . component-c-UUID 1 component-c

-u name-UUID
```

一意の識別子 *name-UUID* で指定されたソフトウェアコンポーネントの情報を表示する

<i>name</i>	一意の識別子 <i>name-UUID</i> でアンインストールするソフトウェアコンポーネントの名前
<i>. component-a-UUID</i>	<i>name</i> ソフトウェアによって必要とされているソフトウェアコンポーネント <i>component-a</i> の一意の識別子
<i>component-a</i>	<i>name</i> ソフトウェアによって必要とされているコンポーネントの名前
<i>- component-b-UUID</i>	<i>name</i> ソフトウェアによって必要とされているソフトウェアコンポーネント <i>component-b</i> の一意の識別子。ハイフン (-) は、 <i>component-b</i> がその他のソフトウェアコンポーネントを必要としていることを表す
<i>component-b</i>	<i>name</i> ソフトウェアによって必要とされているソフトウェアコンポーネントの名前
<i>. component-c-UUID</i>	<i>component-b</i> ソフトウェアによって必要とされているソフトウェアコンポーネント <i>component-c</i> の一意の識別子
<i>component-c</i>	<i>component-b</i> ソフトウェアによって必要とされているソフトウェアコンポーネントの名前

3. ソフトウェアをアンインストールします。

```
# prodreg uninstall -u name-UUID
```

4. アンインストールするソフトウェアの依存関係を確認します。

```
# prodreg info -u name-UUID
```

```
Title: name
```

```
.
```

```
.
```

```
.
```

```
Child Components:
```

Name	UUID	#
-----	-----	-
<i>component-a</i>	<i>component-a-UUID</i>	1
<i>component-b</i>	<i>component-b-UUID</i>	1

```
Required Components:
```

Name	UUID	#
-----	-----	-
<i>component-a</i>	<i>component-a-UUID</i>	1
<i>component-b</i>	<i>component-b-UUID</i>	1

prodreg info コマンドによって出力された次の情報を確認します。

- 「Child Components」 - *name* ソフトウェアコンポーネントに関連付けられたソフトウェアコンポーネントを一覧表示します。 *name* ソフトウェアの登録を解除すると、その子コンポーネントの登録も解除されます。 前出の `prodreg info` コマンドでは、すべての子コンポーネントが一覧表示されます。これらの子コンポーネントの登録を解除してもよいかどうか確認してください。
- 「Required Components」 - *name* ソフトウェアによって必要とされているソフトウェアコンポーネントを一覧表示します。ソフトウェアコンポーネントは、子コンポーネント以外のコンポーネントを必要とする場合があります。コンポーネントとともにアンインストールされ、登録を解除されるのは、子コンポーネントだけです。
- 「Dependent Components」 - *name* ソフトウェアがないと実行できないコンポーネントを一覧表示します。 *name* ソフトウェアの登録を解除すると、その依存コンポーネントの登録も解除されます。前出の `prodreg info` コマンドでは、すべての依存コンポーネントが一覧表示されます。これらの依存コンポーネントの登録を解除してもよいかどうか確認してください。

前出の出力例では、 *name* ソフトウェアに依存コンポーネントはありません。

5. *name* ソフトウェアの子コンポーネントの依存関係を確認します。

```
# prodreg info -u component-a-UUID -i 1 -a "Dependent Components"
Dependent Components:
Name                               UUID                               #
-----
name                                name-UUID                          1

# prodreg info -u component-b-UUID -i 1 -a "Dependent Components"
Dependent Components:
Name                               UUID                               #
-----
name                                name-UUID                          1

# prodreg info -u component-c-UUID -i 1 -a "Dependent Components"
Dependent Components:
Name                               UUID                               #
-----
component-b                         component-b-UUID                    1
```

前出の出力例では、 *name* ソフトウェアの子コンポーネントに依存するソフトウェアはありません。

6. ソフトウェアコンポーネントと子コンポーネントの登録を解除します。

```
# prodreg unregister -r -u name-UUID -i 1

-r                                一意の識別子 name-UUID を指定して、
                                ソフトウェアとその子コンポーネント全
                                部の登録を再帰的に解除する

-u name-UUID                       登録を解除するソフトウェアを表す一意
                                の識別子を指定する
```

例 — ソフトウェアコンポーネントのアンインストール (prodreg)

次の例では、ExampleSoft ソフトウェアとその子コンポーネントすべてをアンインストールします。

```
# prodreg browse -m "ExampleSoft"
BROWSE # +/-/. UUID # NAME
===== =====
1 - root 1 System
Registry
2 + a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1 Solaris 9 8/03
System
Software
3 + 8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified
Software
1423 - 95842091-725a-8501-ef29-0472985982be 1 ExampleSoft
1436 . 90209809-9785-b89e-c821-0472985982be 1 Example Doc
1437 - EXSOztt 1 Example Data
1462 . EXSOblob 1 Example Data

# prodreg uninstall -u 95842091-725a-8501-ef29-0472985982be -i 1

# prodreg info -u 95842091-725a-8501-ef29-0472985982be
Title: ExampleSoft Software
.
.
Child Components:
Name UUID #
-----
Example Doc 90209809-9785-b89e-c821-0472985982be 1
Example Data EXSOztt 1

Required Components:
Name UUID #
-----
Example Doc 90209809-9785-b89e-c821-0472985982be 1
Example Data EXSOztt 1

# prodreg info -u 90209809-9785-b89e-c821-0472985982be -i 1
-a "Dependent Components"
Dependent Components:
Name UUID #
-----
ExampleSoft 95842091-725a-8501-ef29-0472985982be 1

# prodreg info -u EXSOztt -i 1 -a "Dependent Components"
```

```

Dependent Components:
Name                               UUID                               #
-----
ExampleSoft                        95842091-725a-8501-ef29-0472985982be 1

# prodreg info -u EXSOblob -i 1 -a "Dependent Components"
Dependent Components:
Name                               UUID                               #
-----
Example Data                        EXSOztt                            1

# prodreg unregister -r -u 95842091-725a-8501-ef29-0472985982be -i 1

```

▼ 損傷を受けたソフトウェアをアンインストールする方法 (prodreg)

損傷を受けたソフトウェアコンポーネントを `prodreg uninstall` コマンドでアンインストールしようとする、失敗します。これは、ソフトウェアコンポーネントのアンインストールプログラムがシステムから削除されている場合に発生する現象です。

システム上にソフトウェアコンポーネントのアンインストールプログラムがない場合は、次の手順に従ってアンインストールを行なってください。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. アンインストールするソフトウェアの情報を表示します。

```

# prodreg browse -m "name"
BROWSE # +/-/.  UUID                               #  NAME
=====  =====  =====  =  =====
1         -      root                               1  System
Registry
2         +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1  Solaris 9 8/03
System
Software
3         +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1  Unclassified
Software
4         -      UUID                               1  name
1436      .      component-a-UUID                   1  component-a
1437      .      component-b-UUID                   1

```

`-m "name"`

アンインストールする `name` ソフトウェアコンポーネントの情報を表示する

`UUID`

アンインストールするソフトウェアコンポーネントの `UUID`

<code>. component-a-UUID</code>	<code>component-a</code> ソフトウェアコンポーネントの UUID
<code>component-a</code>	<code>name</code> ソフトウェアの子ソフトウェアコンポーネントの名前
<code>. component-b-UUID</code>	<code>name</code> ソフトウェアの子ソフトウェアコンポーネントの UUID

前出の出力例では、`component-b-UUID` エントリにコンポーネント名が関連付けられていません。このように名前前の値がない場合、このコンポーネントは損傷を受けている可能性があります。

3. ソフトウェアをアンインストールします。

```
# prodreg uninstall -u UUID -i 1
The install program requested could not be found
```

<code>-u UUID</code>	アンインストールするソフトウェアコンポーネントの UUID を指定する
<code>-i 1</code>	アンインストールするソフトウェアのインスタンスを指定する

前出のエラーメッセージは、システム上にアンインストールプログラムが存在しないことを示しています。

4. ソフトウェアコンポーネントのアンインストールプログラムを特定します。

```
# prodreg info -m "name" -a uninstallprogram
uninstallprogram: /usr/bin/java -mx64m -classpath
uninstaller-location uninstall_name
```

<code>-m "name"</code>	<code>name</code> ソフトウェアコンポーネントの情報を表示する
<code>-a uninstallprogram</code>	<code>name</code> ソフトウェアコンポーネントに関連付けられたアンインストールプログラムの情報を表示する
<code>uninstaller-location</code>	<code>name</code> ソフトウェアコンポーネントのアンインストールプログラムの登録場所

5. 登録場所にアンインストーラが存在するかどうか確認します。

```
# ls uninstaller-location
uninstaller-location:
No such file or directory
```

`ls (1)` コマンドの出力結果から、登録場所にアンインストールプログラムが存在しないことがわかります。

6. システムからソフトウェアを削除します。

次のいずれかの方法で削除できます。

- システムのバックアップを使用できる場合は、次の手順に従います。

- a. バックアップからアンインストールプログラムをロードします。
 - b. 端末ウィンドウなどのシェルコマンド行インタフェースからアンインストールプログラムを実行します。
- バックアップのアンインストールプログラムにアクセスできない場合は、次の手順に従います。
 - a. ソフトウェアコンポーネントの登録を解除します。


```
# prodreg unregister -u UUID -i 1
```
 - b. 削除するソフトウェアによって必要とされている登録済みコンポーネントをすべて削除します。


```
# pkgrm component-a-UUID
```

例 — 損傷を受けたソフトウェアのアンインストール (prodreg)

次の例では、損傷を受けた ExampleSoft ソフトウェアをアンインストールします。この例では、システムバックアップ上のアンインストールプログラムにはアクセスできないものとします。

```
# prodreg browse -m Examplesoft
BROWSE # +/-/. UUID # NAME
=====
1 - root 1 System Registry
2 + a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1 Solaris 9 8/03 System Software
3 + 8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified Software
4 - 95842091-725a-8501-ef29-0472985982be 1 ExampleSoft
233 . 90209809-9785-b89e-c821-0472985982be 1 Example Doc
234 . EXSOztt 1
235 . EXSOblob 1 Example Data

# prodreg uninstall -u 95842091-725a-8501-ef29-0472985982be -i 1
The install program requested could not be found

# prodreg info -m "ExampleSoft" -a uninstallprogram
uninstallprogram: /usr/bin/java -mx64m -classpath
/var/sadm/prod/org.example.ExampleSoft/987573587 uninstall_ExampleSoft

# ls /var/sadm/prod/org.example.ExampleSoft/987573587
/var/sadm/prod/org.example.ExampleSoft/987573587:
No such file or directory

# prodreg unregister -u 95842091-725a-8501-ef29-0472985982be -i 1

# pkgrm EXSOblob
```

▼ 損傷を受けたソフトウェアコンポーネントを再インストールする方法 (prodreg)

損傷を受けたソフトウェアコンポーネントに依存するソフトウェアがある場合は、損傷を受けたコンポーネントを再インストールできます。損傷を受けたコンポーネントと依存ソフトウェアをアンインストールする必要はありません。prodreg unregister コマンドの -f オプションを実行して、損傷を受けたコンポーネントの登録を強制的に解除し、インストールし直します。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 再インストールするソフトウェアの情報を表示します。

```
# prodreg browse -m "name"
BROWSE # +/-/. UUID # NAME
===== =====
1 - root 1 System
Registry
2 + a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1 Solaris 9 8/03
System
Software
3 + 8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified
Software
4 . UUID 1 name
```

-m "name" 再インストールする *name* ソフトウェアコンポーネントの情報を表示する

UUID 再インストールするソフトウェアコンポーネントの UUID

3. 再インストールするソフトウェアに依存するソフトウェアを特定します。

```
# prodreg info -m "name" -a "Dependent Components"
Dependent Components:
Name UUID #
-----
component-a component-a-UUID 1
```

-m "name" 再インストールするソフトウェアコンポーネントの名前を *name* で指定する

-a "Dependent Components" *name* ソフトウェアに依存するコンポーネントを表示する

component-a *name* ソフトウェアに依存するソフトウェアコンポーネントの名前

component-a-UUID component-a ソフトウェアコンポーネントの UUID

component-a ソフトウェアコンポーネントは、再インストールするソフトウェアに依存しています。component-a の登録を解除しないで *name* ソフトウェアを再インス

トールする場合は、*name* ソフトウェアの登録を強制的に解除したあと、*name* ソフトウェアを再インストールします。

4. 再インストールするソフトウェアコンポーネントだけの登録を解除します。

```
# prodreg unregister -f -u UUID
```

5. ソフトウェアコンポーネントを再インストールします。

```
# /usr/bin/java -cp /usr/installers/installer
```

installer

name ソフトウェアのインストールプログラム
の名前を指定する

例 — 損傷を受けたソフトウェアコンポーネントの再インストール (prodreg)

次の例では、損傷を受けたソフトウェアコンポーネント *ComponentSoft* を、依存コンポーネントである *ExampleSoft* の登録解除やアンインストールを行うことなく再インストールします。

```
# prodreg browse -m "ComponentSoft"
BROWSE # +/-/.  UUID                                     # NAME
===== =====
1      -      root                                     1 System
Registry
2      +      a01ee8dd-1dd1-11b2-a3f2-0800209a5b6b 1 Solaris 9 8/03
System
Software
3      +      8f64eabf-1dd2-11b2-a3f1-0800209a5b6b 1 Unclassified
Software
4      .      86758449-554a-6531-fe90-4352678362fe 1 ComponentSoft

# prodreg info -m "ComponentSoft" -a "Dependent Components"
Dependent Components:
Name                                     UUID                                     #
-----
ExampleSoft                             95842091-725a-8501-ef29-0472985982be 1

# prodreg unregister -f -u 86758449-554a-6531-fe90-4352678362fe -i 1

# /usr/bin/java -cp /usr/installers/org.example.componentsoft
```

署名付きパッケージの追加と削除 (作業マップ)

次の作業マップは、署名付きパッケージの追加および削除に関する作業を示したものです。

作業	説明	参照先
証明書をインポートする	pkgadm addcert コマンドを使って信頼される証明書をインポートする	250 ページの「信頼される証明書をパッケージキーストアにインポートする方法 (pkgadm addcert)」
(省略可能) 1 つ以上の証明書の詳細情報を表示する	pkgadm listcert コマンドを使って証明書の詳細情報を表示する	252 ページの「証明書の情報を表示する方法 (pkgadm listcert)」
(省略可能) 証明書を削除する	pkgadm removecert コマンドを使って証明書を削除する	253 ページの「証明書を削除する方法 (pkgadm removecert)」
(省略可能) プロキシサーバーをセットアップする	プロキシを備えたファイアウォールの背後にシステムが存在している場合、プロキシサーバーを指定する	253 ページの「プロキシサーバーをセットアップする方法 (pkgadd)」
署名付きパッケージを追加する	ルート証明書がインポートされると、pkgadd コマンドを使って署名付きパッケージを追加できる	254 ページの「署名付きパッケージを追加する方法 (pkgadd)」
(省略可能) 署名付きパッケージを削除する	署名付きパッケージの削除方法は、署名なしパッケージの削除方法と同一である	264 ページの「ソフトウェアパッケージを削除する方法 (pkgrm)」

▼ 信頼される証明書をパッケージキーストアにインポートする方法 (pkgadm addcert)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Java** キーストア内にルート **CA** 証明書が存在することを確認します。

```
# keytool -storepass storepass -list -keystore certfile
```

<code>keytool</code>	非公開鍵とそれらに対応する公開鍵を認証するための X.509 証明書チェーンの Java キーストア (データベース) を管理する。また、信頼されるエンティティが発行した証明書も管理する。 <code>keytool</code> コマンドの詳細については、『 <i>keytool-Key and Certificate Management Tool</i> 』を参照。
<code>-storepass storepass</code>	Java キーストアの完全性を維持するためのパスワードを指定する。
<code>-list</code>	デフォルトで、証明書の MD5 フィンガープリントを出力する。
<code>-keystore certfile</code>	Java キーストア永続ファイルの名前と場所を指定する。

3. Java キーストア内のルート CA 証明書を一時ファイルへエクスポートします。

```
# keytool -export -storepass storepass -alias gtecybertrustca -keystore
gtecybertrustca -keystore /usr/j2se/jre/lib/security/cacerts -file filename
```

<code>-export</code>	信頼される証明書をエクスポートする。
<code>-storepass storepass</code>	Java キーストアの完全性を維持するためのパスワードを指定する。
<code>-alias gtecybertrustca</code>	信頼される証明書の別名を指定する。
<code>-keystore certfile</code>	キーストアファイルの名前と場所を指定する。
<code>-file filename</code>	エクスポートされた証明書を格納するファイルを指定する。

4. 信頼される証明書をパッケージキーストアにインポートします。

```
# pkgadm addcert -t -f format certfile
```

<code>-t</code>	証明書が信頼される CA 証明書であることを示す。このオプションを指定すると、コマンド出力に証明書の詳細情報が含まれ、ユーザーはその情報の確認を求められる。
<code>-f format</code>	証明書または非公開鍵の形式を指定する。インポートする証明書は、PEM (pem)、バイナリ DER (der) のいずれかの形式を使ってコード化されている必要がある。
<code>certfile</code>	証明書を含むファイルを指定する。

詳細については、`pkgadm` のマニュアルページを参照。

5. 一時ファイルを削除します。

例 — 信頼される証明書をインポートする

次の例は、信頼される証明書をインポートする方法を示しています。この例では、keytool コマンドを使って、Java キーストアからパッケージキーストアへ Sun のルート CA 証明書をインポートしています。

```
# keytool -export -storepass changeit -alias gtecybertrustca -keystore
gtecybertrustca -keystore /usr/j2se/jre/lib/security/cacerts -file
/tmp/root.crt
Certificate stored in file </tmp/root.crt>
# pkgadm addcert -t -f der /tmp/root.crt
キーストアのパスワードを入力してください: storepass
キーストア別名: GTE CyberTrust Root
共通名: GTE CyberTrust Root
証明書の種類: Trusted Certificate
発行者の共通名: GTE CyberTrust Root
有効な日付: <Feb 23 23:01:00 1996 GMT>-<Feb 23 23:59:00 2006 GMT>
MD5 フィンガープリント: C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58
SHA1 フィンガープリント: 90:DE:DE:9E:4C:4E:9F:6F:D8:86:17:57:9D:D3:91:BC...
証明書 <GTE CyberTrust Root> を信頼します
キーストアの保護パスワードを入力してください。
保護パスワードがない場合は ENTER を押してください (推奨されません): xxx
For Verification: キーストアの保護パスワードを入力してください。
保護パスワードがない場合は ENTER を押してください (推奨されません): xxx
</tmp/root.crt> からの証明書が信頼されるようになりました
# rm /tmp/root.crt
```

▼ 証明書の情報を表示する方法 (pkgadm listcert)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. パッケージキーストアの内容を表示します。

```
# pkgadm listcert
```

例 — 証明書の情報を表示する (pkgadm listcert)

次の例は、ローカルに格納されている証明書の詳細情報を表示する方法を示しています。

```
# pkgadm listcert -P pass:storepass
キーストア別名: GTE CyberTrust Root
共通名: GTE CyberTrust Root
証明書の種類: Trusted Certificate
発行者の共通名: GTE CyberTrust Root
有効な日付: <Feb 23 23:01:00 1996 GMT> - <Feb 23 23:59:00 2006 GMT>
MD5 フィンガープリント: C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58
SHA1 フィンガープリント: 90:DE:DE:9E:4C:4E:9F:6F:D8:86:17:57:9D:D3:91:BC...
```

▼ 証明書を削除する方法 (pkgadm removecert)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 信頼される証明書をパッケージキーストアから削除します。

```
# pkgadm removecert -n "certfile"
-n "certfile" オプションには、ユーザーの証明書と鍵のペアの別名、信頼される証明書の別名のいずれかを指定します。
```

注 - 証明書の別名を表示するには、`pkgadm listcert` コマンドを使用します。

例 — 証明書を削除する (pkgadm removecert)

次の例は、証明書を削除する方法を示しています。

```
# pkgadm listcert
キーストアのパスワードを入力してください: storepass
キーストア別名: GTE CyberTrust Root
共通名: GTE CyberTrust Root
証明書の種類: Trusted Certificate
発行者の共通名: GTE CyberTrust Root
有効な日付:<Feb 23 23:01:00 1996 GMT>-<Feb 23 23:59:00 2006 GMT>
MD5 フィンガープリント: C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58
SHA1 フィンガープリント: 90:DE:DE:9E:4C:4E:9F:6F:D8:86:17:57:9D:D3:91:BC...
# pkgadm removecert -n "GTE CyberTrust Root"
キーストアのパスワードを入力してください: storepass
別名 <GTE CyberTrust Root> を持つ証明書を削除するのに成功しました
```

▼ プロキシサーバーをセットアップする方法 (pkgadd)

プロキシを備えたファイアウォールの背後にシステムが存在している場合、プロキシサーバーをセットアップしてからでないと、`pkgadd` コマンドを使用して HTTP サーバーからパッケージを追加できません。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 次のいずれかの方法を使ってプロキシサーバーを指定します。
 - a. 環境変数 `http_proxy`、`HTTPPROXY`、`HTTPPROXYPORT` のいずれかを使ってプロキシサーバーを指定します。
たとえば、次のようになります。

```
# setenv http_proxy http://mycache.domain:8080
または、次のいずれかを指定します。
```

```
# setenv HTTPPROXY mycache.domain
# setenv HTTPPROXYPORT 8080
```

- b. **pkgadd** コマンド行でプロキシサーバーを指定します。
たとえば、次のようになります。

```
# pkgadd -x mycache.domain:8080 -d http://myserver.com/pkg SUNWpkg
```

- c. プロキシサーバーの情報を含む管理ファイルを作成します。
たとえば、次のようになります。

```
# cat /tmp/admin
mail=
instance=unique
partial=ask
runlevel=ask
idepend=ask
rdepend=ask
space=ask
setuid=ask
conflict=ask
action=ask
networktimeout=60
networkretries=3
authentication=quit
keystore=/var/sadm/security
basedir=default
proxy=mycache.domain:8080
```

続いて、**pkgadd -a** コマンドでこの管理ファイルを指定します。たとえば、次のようになります。

```
# pkgadd -a /tmp/admin -d http://myserver.com/pkg SUNWpkg
```

▼ 署名付きパッケージを追加する方法 (pkgadd)

次の手順では、Sun のルート CA 証明書がすでにインポートされているものと仮定しています。詳細については、[250 ページの「信頼される証明書をパッケージキーストアにインポートする方法 \(pkgadm addcert\)」](#)を参照してください。

プロキシサーバーをセットアップする方法については、[253 ページの「プロキシサーバーをセットアップする方法 \(pkgadd\)」](#)を参照してください。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 署名付きパッケージを追加します。

```
# pkgadd -d /pathname/package-name
```

`-d device-name` オプションには、パッケージのインストール元デバイスを指定します。デバイスとして指定できるのは、ディレクトリ、テープ、フロッピーディスク、リムーバブルディスクのいずれかです。また、`pkgtrans` コマンドによって

作成されたデータストリームを、デバイスとして指定することもできます。

例 — 署名付きパッケージを追加する (pkgadd)

次の例は、ダウンロード済みの署名付きパッケージを追加する方法を示しています。

```
# # pkgadd -d /tmp/signed_pppd
次のパッケージを利用できます:
  1  SUNWpppd      Solaris PPP Device Drivers
                        (sparc) 11.10.0,REV=2003.05.08.12.24
```

パッケージ (複数可) を選択してください
(すべてのパッケージを処理するには 'all' を入力してください)。

(default: all) [?,??,q]: all

キーストアのパスワードを入力してください:

署名者 <User Cert 0> の署名を検証しています

.
.
.

次の例は、デバイス名として HTTP URL を指定して署名付きパッケージをインストールする方法を示しています。指定した URL は、ストリーム形式のパッケージを指している必要があります。

```
# pkgadd -d http://install/signed-video.pkg

## ダウンロード中...
.....25%.....50%.....75%.....100%
## ダウンロードが完了しました
.
.
.
```

パッケージコマンドによるソフトウェアパッケージの管理 (作業マップ)

次の作業マップは、署名付きパッケージと署名なしパッケージの両方に対して、パッケージコマンドで行えるソフトウェア管理作業を示したものです。

作業	説明	参照先
ソフトウェアパッケージをローカルシステムに追加する	pkgadd コマンドを使ってソフトウェアパッケージをローカルシステムに追加できる	256 ページの「ソフトウェアパッケージを追加する方法 (pkgadd)」
ソフトウェアパッケージをスプールディレクトリに追加する	ソフトウェアを実際にインストールしないで、ソフトウェアパッケージをスプールディレクトリに追加できる	259 ページの「ソフトウェアパッケージをスプールディレクトリに追加する」
インストールされたすべてのソフトウェアパッケージに関する情報を表示する	pkginfo コマンドを使用して、インストール済みのパッケージに関する情報を表示できます。	261 ページの「インストール済みのすべてのパッケージに関する情報を表示する方法 (pkginfo)」
インストールされたソフトウェアパッケージの整合性を検査する	pkgchk コマンドを使って、インストールされたソフトウェアパッケージの整合性を検査できる	262 ページの「インストール済みのソフトウェアパッケージの整合性を検査する方法 (pkgchk)」
ソフトウェアパッケージを削除する	pkgrm コマンドを使って不要になったソフトウェアパッケージを削除できる	264 ページの「ソフトウェアパッケージを削除する方法 (pkgrm)」

▼ ソフトウェアパッケージを追加する方法 (pkgadd)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. インストール済みのパッケージの中で、追加しようとしているものと同じ名前のものがあれば、すべて削除します。

この手順によって、追加および削除されたソフトウェアの正しい記録が取れます。同じアプリケーションの複数のバージョンをシステムで管理する場合があります。ソフトウェアの複数のコピーを管理する方法については、220 ページの「パッケージの削除に関するガイドライン (pkgrm)」を、その作業手順については、264 ページの「ソフトウェアパッケージを削除する方法 (pkgrm)」を参照してください。
3. ソフトウェアパッケージをシステムに追加します。

```
# pkgadd -a admin-file -d device-name pkgid ...
```


<code>-a admin-file</code>	(省略可能) インストール時に <code>pkgadd</code> コマンドが参照する管理ファイルを示す。管理ファイルの使用方法については、221 ページの「管理ファイルの使用」を参照。
<code>-d device-name</code>	ソフトウェアパッケージへの絶対パスを指定する。 <code>device-name</code> には、デバイス、ディレクトリ、またはスプールディレクトリへのパスを指定できる。パッケージのあるパスを指定しないと、 <code>pkgadd</code> コマンドはデフォルトのスプールディレクトリ (<code>/var/spool/pkg</code>) をチェックする。パッケージがそこにある場合、パッケージのインストールは失敗する。
<code>pkgid</code>	(省略可能) インストールする 1 つまたは複数のパッケージの名前を空白で区切って指定する。この引数を省略すると、使用可能なすべてのパッケージがインストールされる。

パッケージのインストール中に `pkgadd` コマンドによって問題が検出されると、その問題に関連するメッセージに続いて、次のプロンプトが表示されます。

インストールを続けますか?

このプロンプトには、`yes`、`no`、または `quit` のいずれかで応答します。複数のパッケージを指定した場合は、`no` と入力して、インストール中のパッケージのインストールを中止してください。他のパッケージのインストールが続行されます。インストールを中止する場合は、`quit` と入力してください。

4. 指定したパッケージが正常にインストールされていることを確認します。

```
# pkgchk -v pkgid
```

エラーがない場合は、インストールされたファイルのリストが返されます。エラーがある場合は、`pkgchk` コマンドによってエラーが報告されます。

例 — マウント済みの CD からソフトウェアパッケージを追加する

次の例は、マウント済みの Solaris 9 CD から `SUNWp15u` パッケージをインストールする方法を示しています。また、パッケージファイルが正しくインストールされているかどうかを確認する方法も示します。

```
# pkgadd -d /cdrom/cdrom0/s0/Solaris_9/Product SUNWp15u
```

```
.
.
.
```

<SUNWp15u> のインストールに成功しました。

```
# pkgchk -v SUNWp15u
```

```
/usr
/usr/bin
/usr/bin/perl
/usr/perl5
/usr/perl5/5.00503
```

・
・
・

例 — リモートパッケージサーバーからソフトウェア パッケージをインストールする

インストールするパッケージがリモートシステムから利用できる場合は、そのパッケージが (パッケージ形式で) 格納されているディレクトリを手動でマウントして、ローカルシステムにパッケージをインストールすることができます。

次の例は、リモートシステムからソフトウェアパッケージをインストールする方法を示しています。この例では、`package-server` というリモートシステムの `/latest-packages` ディレクトリにソフトウェアパッケージが格納されているものとします。まず `mount` コマンドによってパッケージが `/mnt` にローカルにマウントされ、次に `pkgadd` コマンドによって `SUNWp15u` パッケージがインストールされます。

```
# mount -F nfs -o ro package-server:/latest-packages /mnt
# pkgadd -d /mnt SUNWp15u
```

```
・  
・  
・
```

<SUNWp15u> のインストールに成功しました。

オートマウンタがサイトで実行されている場合は、リモートパッケージサーバーを手動でマウントする必要はありません。代わりに、オートマウンタパス (この場合は `/net/package-server/latest-packages`) を、`-d` オプションの引数として使用します。

```
# pkgadd -d /net/package-server/latest-packages SUNWp15u
```

```
・  
・  
・
```

<SUNWp15u> のインストールに成功しました。

次の例は、前の例に似ていますが、`-a` オプションを使用して `noask-pkgadd` という管理ファイルを指定している点が異なります。この管理ファイルについては、[221 ページの「パッケージ追加時のユーザーの対話操作を省略する \(pkgadd\)」](#) に説明されています。この例では、`noask-pkgadd` 管理ファイルがデフォルトの格納場所である `/var/sadm/install/admin` にあるものとします。

```
# pkgadd -a noask-pkgadd -d /net/package-server/latest-packages SUNWp15u
```

```
・  
・  
・
```

<SUNWp15u> のインストールに成功しました。

例 — ソフトウェアパッケージを特定の HTTP URL からインストールする

次の例は、特定の HTTP URL をデバイス名として使ってパッケージをインストールする方法を示しています。指定した URL は、ストリーム形式のパッケージを指している必要があります。

```
# pkgadd -d http://install/xf86-4.3.0-video.pkg

## ダウンロード中...
.....25%.....50%.....75%.....100%
## ダウンロードが完了しました
```

次のパッケージを利用できます：

```
1  SUNWxf86r      XFree86 Driver Porting Kit (Root)
   (i386) 4.3.0,REV=0.2003.02.28
2  SUNWxf86u      XFree86 Driver Porting Kit (User)
   (i386) 4.3.0,REV=0.2003.02.28
```

·
·
·

ソフトウェアパッケージをスプールディレクトリに追加する

使用頻度の高いインストール済みパッケージを、スプールディレクトリにコピーしておく便利です。パッケージをデフォルトのスプールディレクトリである `/var/spool/pkg` にコピーする場合、`pkgadd` コマンドを使用するときに、パッケージのソース位置 (`-d device-name` 引数) を指定する必要はありません。`pkgadd` コマンドは、デフォルトで `/var/spool/pkg` ディレクトリを調べて、コマンド行に指定されたパッケージをすべて見つけます。パッケージをスプールディレクトリにコピーすることと、パッケージをシステム上にインストールすることとは異なることに注意してください。

▼ ソフトウェアパッケージをスプールディレクトリに追加する方法 (pkgadd)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. スプール済みのパッケージの中で、追加しようとしているものと同じ名前のものがあれば、すべて削除します。
スプール済みのパッケージを削除する方法については、[264 ページの「例 — スプールされたソフトウェアパッケージを削除する」](#)を参照してください。
3. ソフトウェアパッケージをスプールディレクトリに追加します。

```
# pkgadd -d device-name -s spooldir pkgid ...
```

-d device-name	ソフトウェアパッケージへの絶対パスを指定する。 device-name には、デバイス、ディレクトリ、またはス プールディレクトリへのパスを指定できる。
-s spooldir	パッケージがスプールされるスプールディレクトリの名 前を指定する。spooldir を指定する必要がある。
pkgid	(省略可能) スプールディレクトリに追加する 1 つまたは 複数のパッケージの名前を空白で区切って指定する。こ の引数を省略すると、使用可能なすべてのパッケージが コピーされる。

4. 指定したパッケージがスプールディレクトリに正常にコピーされていることを確認
します。

```
$ pkginfo -d spooldir | grep pkgid
```

pkgid が正常にコピーされると、pkginfo コマンドによって、pkgid に関する 1 行
の情報が返されます。正常にコピーされなかった場合は、pkginfo コマンドに
よってシステムプロンプトが返されます。

例 — マウント済みの CD からスプールディレクトリを設定する

次の例は、SUNWman パッケージをマウント済みの SPARC Solaris 9 CD からデフォル
トのスプールディレクトリ (/var/spool/pkg) に転送する方法を示しています。

```
# pkgadd -d /cdrom/cdrom0/s0/Solaris_9/Product -s /var/spool/pkg SUNWman  
パッケージインスタンス <SUNWman> を転送中
```

例 — リモートソフトウェアパッケージサーバーからスプール ディレクトリを設定する

コピーするパッケージがリモートシステムから利用できる場合は、そのパッケージが
(パッケージ形式で) 格納されているディレクトリを手動でマウントして、ローカルス
プールディレクトリにパッケージをコピーすることができます。

次の例は、このためのコマンドを示しています。この例では、package-server と
いうリモートシステムの /latest-packages ディレクトリにソフトウェアパッケー
ジが格納されているものとします。まず mount コマンドによってパッケージが /mnt
にローカルにマウントされ、次に pkgadd コマンドによって SUNWp15p パッケージが
/mnt からデフォルトのスプールディレクトリ (/var/spool/pkg) にコピーされま
す。

```
# mount -F nfs -o ro package-server:/latest-packages /mnt  
# pkgadd -d /mnt -s /var/spool/pkg SUNWp15p  
パッケージインスタンス <SUNWp15p> を転送中
```

オートマウンタが実行されている場合は、リモートパッケージサーバーを手動でマウントする必要はありません。代わりに、オートマウンタパス (この場合は、`/net/package-server/latest-packages`) を、`-d` オプションの引数として使用してください。

```
# pkgadd -d /net/package-server/latest-packages -s /var/spool/pkg SUNWp15p
パッケージインスタンス <SUNWp15p> を転送中
```

例 — デフォルトのプールディレクトリからソフトウェアパッケージをインストールする

次の例は、デフォルトのプールディレクトリから `SUNWp15p` パッケージをインストールする方法を示しています。オプションを何も指定しないと、`pkgadd` コマンドは `/var/spool/pkg` ディレクトリを検索して名前付きのパッケージを見つけます。

```
# pkgadd SUNWp15p
.
.
.
<SUNWp15p> のインストールに成功しました。
```

インストール済みのすべてのパッケージに関する情報を表示する方法 (`pkginfo`)

`pkginfo` コマンドを使用して、インストール済みのパッケージに関する情報を表示します。

```
$ pkginfo
```

例 — インストール済みのすべてのパッケージを表示する

次は、システムがスタンドアロンまたはサーバーのどちらであるかにかかわらず、`pkginfo` コマンドが、ローカルシステムにインストールされているすべてのパッケージを表示する例です。基本カテゴリ、パッケージ名、およびパッケージの説明が表示されています。

```
$ pkginfo
system      SUNWaccr      System Accounting, (Root)
system      SUNWaccu      System Accounting, (Usr)
system      SUNWadmap     System administration applications
system      SUNWadmc      System administration core libraries
.
.
.
```

例 — ソフトウェアパッケージに関する詳細情報を表示する

```
$ pkginfo -l SUNWcar
PKGINST: SUNWcar
NAME: Core Architecture, (Root)
CATEGORY: system
ARCH: sparc.sun4u
VERSION: 11.9.0,REV=2002.04.06.15.27
BASEDIR: /
VENDOR: Sun Microsystems, Inc.
DESC: core software for a specific hardware platform group
PSTAMP: leo20031111173915
INSTDATE: Apr 02 2004 11:29
HOTLINE: Please contact your local service provider
STATUS: completely installed
FILES:      114 installed pathnames
          36 shared pathnames
          40 directories
          57 executables
          21725 blocks used (approx)
```

▼ インストール済みのソフトウェアパッケージの整合性を検査する方法 (pkgchk)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. インストール済みのパッケージの状態を調べます。

```
# pkgchk -a| -c -v pkgid ...
# pkgchk -d spooldir pkgid ...
```

-a	ファイルの属性と内容 (デフォルト) ではなく、ファイルの属性 (つまりアクセス権) だけを検査することを指定する。
-c	ファイルの内容と属性 (デフォルト) ではなく、ファイルの内容だけを検査することを指定する。
-v	冗長モードを指定する。このモードでは、処理されるたびにファイル名が表示される。
-d spooldir	スプールディレクトリへの絶対パスを指定する。
pkgid	(省略可能) 1 つまたは複数のパッケージの名前を空白で区切って指定する。pkgid を指定しない場合は、システムにインストールされているすべてのソフトウェアパッケージが検査される。

例 — インストール済みのソフトウェアパッケージの内容を検査する

次の例は、パッケージの内容を検査する方法を示しています。

```
# pkgchk -c SUNWbash
```

エラーがない場合は、システムプロンプトが返されます。エラーがある場合は、pkgchk コマンドによってエラーが報告されます。

例 — インストール済みのソフトウェアパッケージのファイル属性を検査する

次の例は、パッケージのファイル属性を検査する方法を示しています。

```
# pkgchk -a SUNWbash
```

エラーがない場合は、システムプロンプトが返されます。エラーがある場合は、pkgchk コマンドによってエラーが報告されます。

例 — スプールディレクトリにインストールされているソフトウェアパッケージを検査する

次の例は、スプールディレクトリ (/export/install/packages) にコピーされたソフトウェアパッケージを検査する方法を示しています。

```
# pkgchk -d /export/install/packages
## checking spooled package <SUNWadmap>
## checking spooled package <SUNWadmfw>
## checking spooled package <SUNWadmc>
## checking spooled package <SUNWsadml>
```

注 – スプールされたパッケージの検査には限界があります。これは、パッケージがインストールされないとすべての情報を検査できないからです。

ソフトウェアパッケージを削除する

ソフトウェアパッケージを削除 (アンインストール) するときは、このパッケージの追加 (インストール) 時に使用した関連ツールを使用します。たとえば、Web Start インストーラを使ってインストールしたソフトウェアは、Web Start アンインストーラでアンインストールします。



注意 - ソフトウェアパッケージを削除するのに `rm` コマンドを使用しないでください。

▼ ソフトウェアパッケージを削除する方法 (`pkgrm`)

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. インストール済みのパッケージを削除します。

```
# pkgrm pkgid ...
```

`pkgid` は、削除する 1 つまたは複数のパッケージの名前 (空白で区切られている) を識別します。この引数を省略すると、使用可能なパッケージがすべて削除されます。

例 — ソフトウェアパッケージを削除する

次の例は、パッケージの削除方法を示しています。

```
# pkgrm SUNWctu
```

次のパッケージは現在インストールされています:

```
SUNWctu          Netra ct usr/platform links (64-bit)
                  (sparc.sun4u) 11.9.0,REV=2001.07.24.15.53
```

このパッケージを削除しますか **y**

```
## インストール済みパッケージインスタンス <SUNWctu> を削除中です。
```

```
## パッケージの依存関係を検査しています。
```

```
## パッケージ情報を処理中です。
```

```
## クラス <none> 中のパス名を削除中です。
```

```
.
.
.
```

例 — スプールされたソフトウェアパッケージを削除する

次の例は、スプールされたパッケージの削除方法を示しています。

```
# pkgrm -s /export/pkg SUNWaudh
```

次のパッケージは現在スプールされています:

```
SUNWaudh          Audio Header Files
                  (sparc) 11.10.0,REV=2003.08.08.00.03
```

このパッケージを削除しますか **y**

スプールされたパッケージインスタンス <SUNWaudh> を削除中です

Admintool によるソフトウェアパッケージの追加と削除 (作業マップ)

次の作業マップに、Admintool を使って行うことができるソフトウェアの管理作業を示します。

作業	説明	参照先
Admintool を使ってソフトウェアパッケージを追加する	ソフトウェアパッケージを表示または追加できる	265 ページの「Admintool を使ってソフトウェアパッケージを追加する方法」
Admintool を使ってソフトウェアパッケージを削除する	ソフトウェアパッケージを表示または削除できる	266 ページの「Admintool を使ってソフトウェアパッケージを削除する方法」

Solaris オペレーティングシステムには Admintool が組み込まれています。このツールは、ソフトウェアパッケージの追加と削除など、いくつかの管理作業を実行するためのグラフィカルユーザーインターフェースです。特に、Admintool を使用すると次の作業を行うことができます。

- ソフトウェアパッケージをローカルシステムに追加する
- ソフトウェアパッケージをローカルシステムから削除する
- ローカルシステムにすでにインストールされているソフトウェアを表示する
- インストールされるソフトウェアパッケージをカスタマイズする
- ソフトウェアパッケージの代替インストール用ディレクトリを指定する

▼ Admintool を使ってソフトウェアパッケージを追加する方法

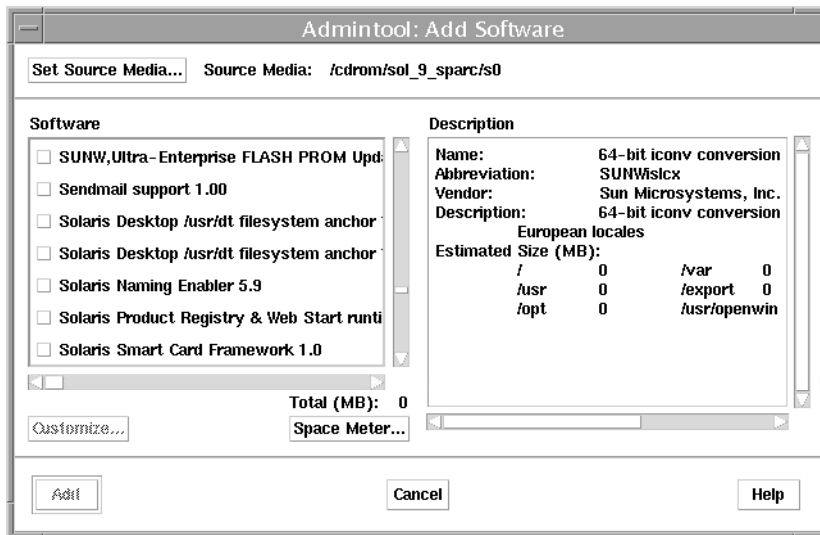
1. スーパーユーザーになります。

Admintool を使ってソフトウェアパッケージを追加または削除する場合、`sysadmin` グループ (グループ 14) のメンバー以外のユーザーは、スーパーユーザーになるか、同等の役割を引き受ける必要があります。

2. **Solaris 9 Software CD** または **DVD** をドライブに挿入します。
ボリュームマネージャが自動的に CD をマウントします。
3. **Admintool** を起動します。

```
# admintool &  
「ユーザー (Users)」ウィンドウが表示されます。
```

4. 「ブラウズ (Browse)」メニューから「ソフトウェア (Software)」を選択します。
「ソフトウェア (Software)」ウィンドウが表示されます。
5. 「編集 (Edit)」メニューから「追加 (Add)」を選択します。
「ソースメディアの設定 (Set Source Media)」ウィンドウが表示されます。必要であれば、インストールメディアへのパスを指定して、「了解 (OK)」をクリックします。デフォルトのパスは、マウント済みの Solaris CD です。
「ソフトウェアを追加 (Add Software)」ウィンドウが表示されます。



6. ローカルシステムにインストールするソフトウェアを選択します。
ウィンドウの「ソフトウェア (Software)」部分で、インストールするソフトウェアに対応するチェックボックスをクリックします。
7. 「追加 (Add)」をクリックします。
パッケージがインストールされるごとに、コマンドツールウィンドウが表示され、インストールの出力が示されます。
「ソフトウェア (Software)」ウィンドウがリフレッシュされて、追加されたばかりのパッケージが表示されます。

▼ Admintool を使ってソフトウェアパッケージを削除する方法

1. スーパーユーザーになります。

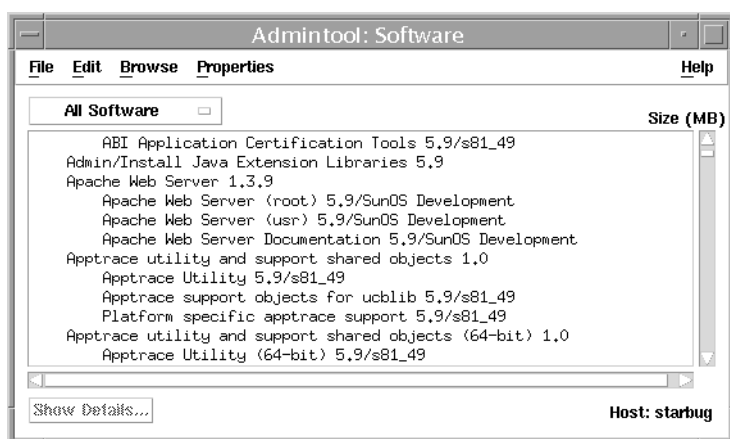
Admintool を使ってソフトウェアパッケージを追加または削除する場合、`sysadmin` グループ (グループ 14) のメンバー以外のユーザーは、スーパーユーザーになるか、同等の役割を引き受ける必要があります。

2. **Admintool** を起動します。

```
# admintool &
```

「ユーザー (Users)」ウィンドウが表示されます。

3. 「ブラウズ (**Browse**)」メニューから「ソフトウェア (**Software**)」を選択します。
「ソフトウェア (Software)」ウィンドウが表示されます。



4. ローカルシステムから削除するソフトウェアを選択します。
5. 「編集 (**Edit**)」メニューから「削除 (**Delete**)」を選択します。
本当にソフトウェアを削除するのかどうかを確認するための警告ポップアップウィンドウが表示されます。
6. 「削除 (**Delete**)」をクリックして、ソフトウェアの削除を確定します。
パッケージが削除されるごとに、コマンドツールウィンドウが表示され、もう一度ソフトウェアを削除するのかどうか確認されます。y、n、またはqを入力します。ソフトウェアを削除することを選択した場合、削除プロセスからの出力が表示されます。

第 15 章

Solaris パッチの管理 (概要)

パッチの管理には、Solaris を実行しているシステムでの Solaris パッチの表示または追加が含まれます。また、不要なパッチや障害の発生したパッチの削除が含まれる場合もあります。パッチの削除は、パッチの「バックアウト」とも呼ばれます。

この章の内容は次のとおりです。

- 270 ページの「パッチについて」
- 270 ページの「署名付きパッチについて」
- 271 ページの「Solaris パッチへのアクセス」
- 272 ページの「Solaris パッチの管理用ツール」

システムにパッチを追加する手順については、275 ページの「Solaris 環境でのパッチ管理 (作業マップ)」を参照してください。

パッチをディスクレスクライアントシステムに追加する方法については、124 ページの「ディスクレスクライアント OS サービスにパッチを適用する」を参照してください。

注 - smpatch コマンドと PatchPro を使用してパッチを管理する方法については、本書では説明していません。smpatch コマンドと PatchPro を使用方法については、『*Signed Patches Administration Guide for PatchPro 2.2*』(英語版)を参照してください。

パッチについて

パッチは、既存のソフトウェアの正常な実行の妨げとなっているファイルとディレクトリを置換または更新するためのファイルとディレクトリの集まりです。既存のソフトウェアと同様に、パッチはアプリケーションバイナリインタフェースに準拠している指定の「パッケージ」形式を元に作成されます。パッケージの詳細については、[第 13 章](#)を参照してください。

システムのパッチの管理は、`patchadd` コマンドを使用して行うことができます。署名なしパッチをシステムに追加する手順については、[282 ページ](#)の「署名のない Solaris パッチの管理 (作業マップ)」を参照してください。

署名付きパッチについて

「署名付き」パッチとは、デジタル署名の付いたパッチのことです。有効なデジタル署名付きのパッチは、署名が適用された以降はパッチの変更が行われていないことを保証します。署名付きパッチには、パッチをシステムに追加する前に確認できるデジタル署名が含まれています。このため、パッチのダウンロードやパッチの追加をより安全に行うことができます。

Solaris 2.6、7、8、および 9 リリースのパッチには、デジタル署名が付属しています。デジタル署名のないパッチ（「署名なしパッチ」）もありますが、最終的にはすべてのパッチが「署名付きパッチ」になる予定です。有効なデジタル署名は、署名が適用された以降にパッチの変更が行われていないことを保証します。

署名付きパッチは Java アーカイブ形式 (JAR) ファイルに格納され、SunSolve OnlineSM から入手できます。

以前の Solaris リリースでは、`smpatch` コマンドと PatchPro を併用することで、署名付きパッチをシステムに追加できました。`smpatch` コマンドの使用手順については、『*Signed Patches Administration Guide for PatchPro 2.2*』（英語版）の「Managing Signed Patches by Using Solaris Patch Management Tools (Tasks)」を参照してください。

この Solaris リリースでは、`patchadd` コマンドを使って署名付きパッチをシステムに追加できます。`patchadd` コマンドの使用手順については、[277 ページ](#)の「`patchadd` コマンドによる署名付きパッチの追加 (作業マップ)」を参照してください。

署名付きパッチの概要に関する追加情報については、[213 ページ](#)の「署名付きパッケージおよびパッチ」を参照してください。

Solaris パッチへのアクセス

Sun のすべてのユーザーは、SunSolve OnlineSM の Web サイトからパッチにアクセスできます。次の表に、Solaris パッチにアクセスするためのさまざまな方法を示します。

表 15-1 パッチにアクセスする方法

ユーザーの種類	説明
SunSpectrum 契約ユーザー	パッチ情報を格納した SunSolve データベースにアクセスできる。これらのパッチ情報は、SunSolve Online サイトから入手できる。匿名 ftp サービスを利用して入手することもできる。 これらのパッチは、夜間に更新される。
SunSpectrum 契約ユーザー以外	セキュリティパッチの一般セットなどの推奨されるパッチにアクセスできる。これらのパッチ情報は、SunSolve Online から入手できる。

Solaris パッチには、Web サイトから、または匿名 ftp サービスを利用してアクセスできます。

Web サイトからパッチにアクセスする場合は、システムが次の要件を満たしている必要があります。

- インターネットに接続されている
- NetscapeTM などの Web ブラウザを実行できる

匿名 ftp を使用してパッチにアクセスする場合は、システムが次の要件を満たしている必要があります。

- インターネットに接続されている
- ftp プログラムを実行できる

次の URL を使って、SunSolve OnlineSM の Web サイトからパッチにアクセスします。

`http://sunsolve.Sun.COM/pub-cgi/show.pl?target=patches/patch-access`

推奨されるいくつかのパッチからなるパッチクラスタをインストールすることも、自由に使用できる個々のパッチをインストールすることもできます。パッチレポートも入手できます。

Solaris パッチの番号付け

パッチは、固有の英数字文字列によって識別されます。これは、パッチのベース番号、ハイフン(-)、パッチの改訂バージョン番号の順で構成されています。たとえば、パッチ 108528-10 は SunOS 5.8 カーネル更新用パッチの「パッチ ID」です。

Solaris パッチの管理用ツール

次の表に、Solaris パッチの管理機能の概要を示します。

機能	patchadd/patchrm コマンド	Solaris 2.6、7、および 8 のパッチ管理ツール	Solaris 9 のパッチ管理ツール	PatchPro Interactive または PatchPro Expert
入手方法	Solaris リリースに付属 (SUNWswmt)	http://www.sun.com/PatchPro からダウンロードする必要あり	http://www.sun.com/PatchPro からダウンロードする必要あり	http://www.sun.com/PatchPro からツールを実行
利用できる Solaris リリース	Solaris 2.6、7、8、および 9	Solaris 2.6、7、および 8	Solaris 9	Solaris 2.6、7、8、および 9
署名付きパッチの追加	可。署名付きパッチはダウンロード時に自動的に検証される	可。署名付きパッチはダウンロード時に自動的に検証される	可。署名付きパッチはダウンロード時に自動的に検証される	不可
署名なしパッチの追加	可	不可	可。ただし、最初にパッチを解凍する必要あり	可。ただし、最初にパッチを解凍する必要あり
GUI	なし	なし	あり	なし
システムごとに必要なパッチを分析し、署名付きまたは署名なしパッチをダウンロードする機能	なし	あり。署名付きパッチ、署名なしパッチの両方をダウンロード	あり。署名付きパッチ、署名なしパッチの両方をダウンロード	あり。署名なしパッチのみダウンロード
ローカル / リモートのシステムパッチサポート	ローカル	ローカル	ローカルおよびリモート	なし
RBAC サポート	あり	なし	あり	なし

パッチのインストール方法とバックアウト方法の詳細については、patchadd(1M) および patchrm(1M) のマニュアルページを参照してください。各パッチには、パッチ情報が記載されている README ファイルも含まれています。

署名付きパッチの追加方法の選択

パッチ管理ツールのインストールが完了したら、署名付きパッチをダウンロードし、システムに追加できます。これには、複数の方法が存在します。次の表を参考にして、現在の要件にもっとも適した方法を選択してください。

コマンドまたはツール	説明	参照先
patchadd	Solaris 9 12/03 リリース以降 – パッケージキーストアをセットアップした後、このコマンドを使って署名付きパッチをシステムに追加する	patchadd (1M)
smpatch update	Solaris 2.6、Solaris 7、Solaris 8、および Solaris 9 12/03 以前 – 推奨されるパッチを特定し、自動的にシステムにダウンロードして適用する。プロパティを対話形式により設定するパッチには適用されないことに注意	smpatch (1M)
smpatch analyze	現在のシステムに必要なパッチを特定し、パッチ ID を一覧表示する。これらのパッチをシステムにダウンロードし、追加するには、smpatch download および smpatch add コマンドを使用する	smpatch (1M)
smpatch download および smpatch add	システムに 1 つまたは複数のパッチをダウンロードして適用する。これらのコマンドで、必須パッチのダウンロードおよび適用も可能	『Signed Patches Administration Guide for PatchPro 2.2』(英語版)の「Downloading and Applying Signed Patches to a Solaris System (Task Map)」
ftp および smpatch add	ftp コマンドで、システムにパッチ (1 つまたは複数) を転送する。次に、smpatch add コマンドで、システムにパッチを追加する	『Signed Patches Administration Guide for PatchPro 2.2』(英語版)の「Downloading and Applying Signed Patches to a Solaris System (Task Map)」
Solaris 管理コンソールのパッチツール	Solaris 9 システムのみ – GUI 機能を使って署名付きパッチを管理したい場合にのみ、このツールを使用する	Solaris 管理コンソールのオンラインヘルプ

第 16 章

Solaris パッチの管理 (手順)

この章では、Solaris が動作するシステム上でパッチを管理する手順について説明します。

この章では、次の内容について説明します。

- 275 ページの「Solaris 環境でのパッチ管理 (作業マップ)」
- 277 ページの「patchadd コマンドによる署名付きパッチの追加 (作業マップ)」
- 282 ページの「署名のない Solaris パッチの管理 (作業マップ)」

Solaris 環境での基本的なパッチの管理方法については、第 15 章を参照してください。

注 - smpatch コマンドと PatchPro を使用してパッチを管理する方法については、本書では説明していません。smpatch コマンドと PatchPro を使用する方法については、『*Signed Patches Administration Guide for PatchPro 2.2*』(英語版)を参照してください。

Solaris 環境でのパッチ管理 (作業マップ)

Solaris 環境でのすべてのパッチ管理作業を次のマップに一覧表示します。各作業は、署名付きパッチや署名のないパッチの管理をはじめとする一連の追加作業を示しています。

作業	説明	参照先
署名付きパッチと署名のないパッチのどちらを追加するかを判断する	現在の環境にとって署名付きパッチと署名のないパッチのどちらが適しているかを判断する。	276 ページの「署名付きパッチまたは署名のないパッチの選択」
システムに署名付きパッチまたは署名のないパッチを追加する	署名付きパッチを追加するには、次のいずれかのコマンドを使用する。	
	Solaris 9 12/03 以降のリリースでは patchadd コマンドを使用する。	277 ページの「patchadd コマンドによる署名付きパッチの追加 (作業マップ)」
	Solaris 2.6、7、8、または 9 リリースでは smpatch コマンドを使用する。	『Signed Patches Administration Guide for PatchPro 2.2』(英語版)の「Downloading and Applying Signed Patches to a Solaris System (Task Map)」
	システムに署名のないパッチを追加する。	282 ページの「署名のない Solaris パッチの管理 (作業マップ)」

署名付きパッチまたは署名のないパッチの選択

署名付きパッチと署名のないパッチのどちらを追加するかは、パッチのダウンロード時にセキュリティ保護を適用する必要があるかどうかによって決定します。パッチのダウンロード時にセキュリティ保護を適用する必要がある場合は、署名付きパッチを追加します。

patchadd コマンドによる署名付きパッチの追加 (作業マップ)

作業	説明	参照先
1. パッケージキーストアをセットアップする	Sun のルート CA 証明書をパッケージキーストアにインポートする	277 ページの「信頼される証明書をパッケージキーストアにインポートする方法 (pkgadm addcert)」
(省略可能) プロキシサーバーをセットアップする	プロキシを備えたファイアウォールの背後にシステムが存在している場合、プロキシサーバーを指定する	279 ページの「プロキシサーバーをセットアップする方法 (patchadd)」
2. 署名付きパッチをダウンロードし、追加する	署名付きパッチをダウンロードし、それを patchadd コマンドを使ってシステムに追加する際に、次のいずれかの方法を選択する	
	署名付き Solaris パッチのダウンロードと追加を手動で行う	279 ページの「署名付き Solaris パッチのダウンロードと追加を手動で行う方法 (patchadd)」
	署名付き Solaris パッチのダウンロードと追加を自動で行う	280 ページの「署名付き Solaris パッチのダウンロードと追加を自動で行う方法 (patchadd)」

信頼される証明書をパッケージキーストアにインポートする方法 (pkgadm addcert)

patchadd コマンドを使って署名付きパッチをシステムに追加するには、署名付きパッチ上の署名を検証できるように、Sun のルート CA 証明書を事前に追加しておく必要があります。この証明書は、Java キーストアからパッケージキーストアへインポートできます。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. **Java** キーストア内のルート CA 証明書を一時ファイルへエクスポートします。たとえば、次のようになります。

```
# keytool -export -storepass changeit -alias gtecybertrustca -keystore
gtecybertrustca -keystore /usr/j2se/jre/lib/security/cacerts -file
/tmp/root.crt
Certificate stored in file </tmp/root.crt>
```

-export	信頼される証明書をエクスポートする。
-storepass <i>storepass</i>	Java キーストアの完全性を維持するためのパスワードを指定する。
-alias <i>gtecybertrustca</i>	信頼される証明書の別名を指定する。
-keystore <i>certfile</i>	キーストアファイルの名前と場所を指定する。
-file <i>filename</i>	エクスポートされた証明書を格納するファイルを指定する。

3. 一時ファイル内のルート **CA** 証明書をパッケージキーストアにインポートします。たとえば、次のようになります。

```
# pkgadm addcert -t -f der /tmp/root.crt
キーストアのパスワードを入力してください: storepass
キーストア別名: GTE CyberTrust Root
共通名: GTE CyberTrust Root
証明書の種類: Trusted Certificate
発行者の共通名: GTE CyberTrust Root
有効な日付: <Feb 23 23:01:00 1996 GMT>-<Feb 23 23:59:00 ...
MD5 フィンガープリント: C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58
SHA1 フィンガープリント: 90:DE:DE:9E:4C:4E:9F:6F:D8:86:17:57:9D:D3:91...
```

```
本当にこの証明書を信頼しますか? yes
証明書 <GTE CyberTrust Root> を信頼します
キーストアの保護パスワードを入力してください。
保護パスワードがない場合は ENTER を押してください (推奨されません):
For Verification: キーストアの保護パスワードを入力してください。
保護パスワードがない場合は ENTER を押してください (推奨されません):
</tmp/root.crt> からの証明書が信頼されるようになりました
```

-t	証明書が信頼される CA 証明書であることを示す。このオプションを指定すると、コマンド出力に証明書の詳細情報が含まれ、ユーザーはその情報の確認を求められる。
-f <i>format</i>	証明書または非公開鍵の形式を指定する。インポートする証明書は、PEM (<i>pem</i>)、バイナリ DER (<i>der</i>) のいずれかの形式を使ってコード化されている必要がある。
<i>certfile</i>	証明書を含むファイルを指定する。

4. 証明書の情報を表示します。たとえば、次のようになります。

```
# pkgadm listcert -P pass:storepass
キーストア別名: GTE CyberTrust Root
共通名: GTE CyberTrust Root
証明書の種類: Trusted Certificate
発行者の共通名: GTE CyberTrust Root
```

```
有効な日付: <Feb 23 23:01:00 1996 GMT>-<Feb 23 23:59:00 2006 GMT>  
MD5 フィンガープリント: C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58  
SHA1 フィンガープリント: 90:DE:DE:9E:4C:4E:9F:6F:D8:86:17:57:9D:D3:91:  
BC:65:A6:89:64
```

5. 一時ファイルを削除します。
たとえば、次のようになります。

```
# rm /tmp/root.crt
```

▼ プロキシサーバーをセットアップする方法 (patchadd)

プロキシを備えたファイアウォールの背後にシステムが存在している場合、プロキシサーバーをセットアップしてからでないと、patchadd コマンドを使用して HTTP サーバーからパッケージを追加できません。

1. スーパーユーザーになるか、同等の役割を引き受けます。
2. 次のいずれかの方法を使ってプロキシサーバーを指定します。
 - a. 環境変数 `http_proxy`、`HTTPPROXY`、`HTTPPROXYPORT` のいずれかを使ってプロキシサーバーを指定します。
たとえば、次のようになります。

```
# setenv http_proxy http://mycache.domain:8080
```

または、次のいずれかを指定します。

```
# setenv HTTPPROXY mycache.domain
```

```
# setenv HTTPPROXYPORT 8080
```

- b. `patchadd` コマンド行でプロキシサーバーを指定します。

たとえば、次のようになります。

```
# patchadd -x mycache.domain:8080 -M http://www.sun.com/solaris/patches/latest 101223-02  
102323-02
```

▼ 署名付き Solaris パッチのダウンロードと追加を手動で行う方法 (patchadd)

次の手順は、署名付き Solaris パッチのダウンロードと追加を別個に手動で行いたい場合に使用できます。

この手順では、パッケージキーストアのセットアップが完了しているものと仮定しています。

1. (省略可能) パッチを適用するシステムにログインします。

あるいは、パッチをダウンロードし、ftp コマンドを使用してそのパッチをターゲットのシステムにコピーすることもできます。

2. **Web** ブラウザを開き、**SunSolve Online** のサイトに移動します。

<http://sunsolve.Sun.COM/pub-cgi/show.pl?target=patches/patch-access>

3. 特定のパッチだけをダウンロードするか、パッチクラスタをダウンロードするかを決めます。次のいずれかの手順に従います。

- a. 「パッチ検索 (**Find Patch**)」 検索フィールドにパッチ番号 (*patch-ID*) を入力し、「パッチ検索 (**Find Patch**)」 をクリックします。

patch-ID を入力すると、最新バージョンのパッチがダウンロードされます。

自由に使用できるパッチの場合、そのパッチの README が表示されます。このパッチが自由に使用できない場合は、ACCESS DENIED メッセージが表示されます。

SPARC システムと x86 システムではパッチ番号が異なります。これについては、表示されるパッチの README に記載されています。使用しているシステムアーキテクチャに適合したパッチをインストールしてください。

- b. パッチを適用するシステムで実行している **Solaris** に基づいて、推奨されるパッチクラスタをクリックします。

4. 「この署名付きパッチ (*n bytes*) をダウンロードする **HTTPS FTP (Download Signed Patch HTTPS or FTP)**」 をクリックします。

署名付きパッチ (またはパッチクラスタ) が正常にダウンロードされたら、Web ブラウザを閉じます。

5. 必要に応じて、ダウンロードされたパッチパッケージが含まれているディレクトリに移動します。

6. スーパーユーザーになるか、同等の役割を引き受けます。

7. 署名付きパッチを追加します。
たとえば、次のようになります。

```
# patchadd /tmp/114861-01.jar
```

▼ 署名付き Solaris パッチのダウンロードと追加を自動で行う方法 (patchadd)

次の手順は、署名付き Solaris パッチのダウンロードと追加を、自動的に一度で行いたい場合に使用できます。

この手順では、パッケージキーストアのセットアップが完了しているものと仮定しています。

1. スーパーユーザーになるか、同等の役割を引き受けます。

2. ダウンロードするパッチの **HTTP URL** を特定します。

a. **Web** ブラウザを開き、**SunSolve Online** のサイトに移動します。

```
http://sunsolve.Sun.COM/pub-cgi/show.pl?target=patches/patch-access
```

b. ダウンロードするパッチ番号を入力し、「パッチ検索 (**Find Patch**)」をクリックします。

たとえば、次のようになります。

```
114861-01
```

c. パッチページの最上部の **HTTPS** リンクの上にマウスを置きます。

パッチの **URL** が、画面の下部にあるブラウザステータス行に表示されます。

3. 署名付きのパッチを **SunSolve Online** のサイトからダウンロードし、システムに追加します。

たとえば、次のようになります。

```
# patchadd "http://sunsolve.central.sun.com/cgi/patchDownload.pl?target=
114684&method=hs"
.
.
.
Downloading patch from ...
+ dwnld_file http://sunsolve.central.sun.com/cgi/patchDownload.pl?target=
114684&method=hs /tmp/patchadd-dwnld /var/sadm/security console patchadd
.....20%.....40%.....60%.....80%.....100%
## Downloading...
## Download Complete
.
.
.
Enter keystore password: xxx
.
.
.
```

署名のない Solaris パッチの管理 (作業マップ)

作業	説明	参照先
1. (省略可能) 署名のないパッチの情報を表示する	システムにインストール済みの署名のないパッチに関する情報を表示する	282 ページの「Solaris パッチの情報を表示する方法」
2. 署名のないパッチをダウンロードする	署名のないパッチをシステムにダウンロードする	284 ページの「署名のない Solaris パッチのダウンロード方法」
3. 署名のないパッチを追加する	システムに署名のないパッチを追加する。	285 ページの「署名のない Solaris パッチの追加方法 (patchadd)」
4. (省略可能) 署名のないパッチを削除する	必要に応じて、署名のないパッチをシステムから削除する	286 ページの「署名のない Solaris パッチを削除する方法」

署名のない Solaris パッチの情報を表示する

パッチをインストールするにあたっては、以前にインストールされたパッチの詳細を知る必要がある場合もあります。次の表に、すでにシステムにインストール済みのパッチについての情報を表示するためのコマンドを示します。

表 16-1 Solaris パッチの管理用コマンド

コマンド	説明
<code>patchadd -p</code> , <code>showrev -p</code>	システムに適用されたすべてのパッチを表示する
<code>pkgparam pkgid PATCHLIST</code>	<i>pkgid</i> (パッケージの名前) によって識別されるパッケージに適用されたすべてのパッチを表示する。たとえば、 <code>SUNWadmap</code>
<code>patchadd -s Solaris-OS -p</code>	OS サーバーにインストールされているすべての <code>/usr</code> パッチを表示する

Solaris パッチの情報を表示する方法

`patchadd -p` コマンドを使用して、システムにインストールされているパッチに関する情報を表示します。

```
$ patchadd -p
```

次のコマンドを使用して、特定のパッチがシステムにインストールされているかどうかを確認します。たとえば、次のようになります。

```
$ patchadd -p | grep 111879
```

署名のない Solaris パッチの追加

次のツールを使用して、署名のないパッチをサーバーまたはスタンドアロンシステムに追加できます。

- patchadd
- smpatch
- Solaris 管理コンソールのパッチマネージャ

ディスクレスクライアントシステムにパッチを追加する必要がある場合は、[124 ページの「ディスクレスクライアント OS サービスにパッチを適用する」](#)を参照してください。

パッチを追加すると、パッチツールが pkgadd コマンドを呼び出し、パッチディレクトリのパッチパッケージをローカルシステムのディスクにインストールします。パッチツールが実際に行う処理は、以下のとおりです。

- 管理ホストとターゲットホストの Solaris バージョン番号を確認する。
- インストール中のパッチによって置換されるパッチ、このパッチに必要な他のパッチ、およびこのパッチと互換性を持たないパッチに関する情報を使用して、パッチパッケージの pkginfo ファイルを更新する。

パッチのインストール時に、patchadd コマンドはインストールプロセスの記録を現在の Solaris バージョンの /var/sadm/patch/patch-ID/log ファイルに保存します。

この patchadd コマンドは、次の条件ではパッチをインストールしません。

- パッケージがホストに完全にインストールされていない。
- パッチパッケージのアーキテクチャが、システムのアーキテクチャと異なる。
- パッチパッケージのバージョンが、インストールされているパッケージのバージョンと一致しない。
- 同じベースコードと上位のバージョン番号を持つパッチがすでにインストールされている。
- パッチが、すでにインストールされている他のパッチと互換性を持たない。インストール済みの各パッチは、この情報を pkginfo ファイルに格納している。
- インストールしようとしているパッチには他のパッチが必要だが、そのパッチがインストールされていない。

smpatch コマンドを使用してパッチを追加または削除するときに、コマンド行に認証情報を指定しなかった場合は、認証情報の入力を要求されます。

次の構文を使用して、smpatch コマンドに認証情報を指定できます。

```
# smpatch add -p mypassword -u root -- -i patch-ID-revision
```

add や remove などの smpatch サブコマンドでは、区切り文字「--」の後に認証のオプションと引数を指定します。

smpatch コマンドに認証情報を要求するプロンプトを表示させることもできます。

```
# /usr/sadm/bin/smpatch add -i patch-ID-revision
ヘルプを参照するには /? を入力してください。Enter キーを押すと、[ ] で囲まれたデフォルトが選択されます。
文字列の値を入力してください: password ::
ツール com.sun.admin.patchmgr.cli.PatchMgrCli を holoship から読み込み中
ユーザー root として holoship にログインしました。
holoship から com.sun.admin.patchmgr.cli.PatchMgrCli がダウンロードされました。
```

▼ 署名のない Solaris パッチのダウンロード方法

1. (省略可能) パッチを適用するシステムにログインします。
あるいは、パッチをダウンロードし、ftp コマンドを使用してそのパッチをターゲットのシステムにコピーすることもできます。
2. **Web** ブラウザを開き、**SunSolve Online** のサイトに移動します。

```
http://sunsolve.Sun.COM/pub-cgi/show.pl?target=patches/patch-access
```
3. 特定のパッチだけをダウンロードするか、パッチクラスタをダウンロードするかを決めます。次のいずれかの手順に従います。
 - a. 「パッチ検索 (**Find Patch**)」検索フィールドにパッチ番号 (*patch-ID*) を入力し、「パッチ検索 (**Find Patch**)」をクリックします。
patch-ID を入力すると、最新バージョンのパッチがダウンロードされます。
自由に使用できるパッチの場合、そのパッチの README が表示されます。このパッチが自由に使用できない場合は、ACCESS DENIED メッセージが表示されます。
SPARC システムと x86 システムではパッチ番号が異なります。これについては、表示されるパッチの README に記載されています。使用しているシステムアーキテクチャに適合したパッチをインストールしてください。
 - b. パッチを適用するシステムで実行している **Solaris** に基づいて、推奨されるパッチクラスタをクリックします。
4. 「このパッチ (*n bytes*) をダウンロードする **HTTP FTP (Download Patch HTTP or FTP)**」をクリックします。
パッチ (またはパッチクラスタ) が正常にダウンロードされたら、Web ブラウザを閉じます。
5. 必要に応じて、ダウンロードされたパッチパッケージが含まれているディレクトリに移動します。
6. パッチパッケージを解凍します。

```
% unzip patch-ID-revision
```

▼ 署名のない Solaris パッチの追加方法 (patchadd)

この手順は、パッチがダウンロードおよび解凍済みであることを前提としています。

1. スーパーユーザーになります。
2. パッチ (または複数のパッチ) を追加します。
3. パッチが正常に削除されたことを確認します。

```
# patchadd patch-ID-revision
```

```
# patchadd -p | grep patch-ID-revision
```

例 — 署名のない Solaris パッチの追加

次の例では、patchadd コマンドを使用して、システムに Solaris 8 パッチ (パッチ番号 111879-01) を追加します。システムへのパッチのダウンロードおよび解凍は完了しているものとします。

```
# patchadd /export/Sol8patch/111879-01
```

```
Checking installed patches...
Verifying sufficient filesystem capacity (dry run method)...
Installing patch packages...
```

```
Patch number 111879-01 has been successfully installed.
See /var/sadm/patch/111879-01/log for details
```

```
Patch packages installed:
```

```
SUNWwsr
```

```
# patchadd -p | grep 111879-01
```

```
Patch: 111879-01 Obsoletes: Requires: Incompatibles: Packages: SUNWwsr
```

▼ 署名のない Solaris パッチの追加方法 (smpatch)

Solaris 9 リリースが稼働するシステムで、次の手順を実行します。

この手順は、パッチを /var/sadm/spool ディレクトリにダウンロードして解凍済みであることを前提としています。

1. スーパーユーザーになります。
2. パッチ (または複数のパッチ) を追加します。

```
# /usr/sadm/bin/smpatch add patch-ID-revision
```

たとえば、次のようになります。

```
# /usr/sadm/bin/smpatch add -i 115028-01
Authenticating as user: root
```

ヘルプを参照するには /? を入力してください。Enter キーを押すと、[] で囲まれたデフォルトが選択されます。
文字列の値を入力してください: password ::
ツール com.sun.admin.patchmgr.cli.PatchMgrCli を holoship から読み込み中
ユーザー root として holoship にログインしました。
holoship から com.sun.admin.patchmgr.cli.PatchMgrCli がダウンロードされました。

パッチ 115028-01 もしくはパッチ 115028-01 が必要とするパッチは、インストール後に、システムを再起動する必要があります。インストール後に、再設定のために、システムを直ちに再起動してください。

```
On machine holoship ...  
Installing patch 115028-01
```

3. システムのリポートを要求するメッセージがあるかどうかを **smpatch** メッセージで確認します。

リポートが要求されている場合は、システムをリポートします。

```
# init 6
```

4. パッチが正常に削除されたことを確認します。

```
# patchadd -p | grep patch-ID-revision
```

署名のない Solaris パッチの削除

パッチをバックアウトすると、次の場合を除いて、そのパッチによって変更されたすべてのファイルをパッチツールが復元します。

- パッチが `patchadd -d` オプションを使用してインストールされた場合 (-d オプションは、更新中または置換中のファイルのコピーを保管しないように `patchadd` に指示する)
- パッチが最新のパッチによって置換されている場合
- パッチが他のパッチによって必要とされている場合

パッチツールは、`pkgadd` コマンドを呼び出して最初のパッチインストールで保管されたパッケージを復元します。

パッチを削除している間、`patchrm` コマンドは、`/tmp/backoutlog.process_id` にバックアウトプロセスのログを格納します。パッチが正常にバックアウトされた場合、このログファイルは削除されます。

▼ 署名のない Solaris パッチを削除する方法

署名のない Solaris パッチを削除する必要がある場合には、`smpatch` コマンド、`patchrm` コマンド、または Solaris 管理コンソールのパッチマネージャを使用できます。

1. スーパーユーザーになります。

2. パッチを削除します。

```
# patchrm patch-ID-revision
```

または

```
# /usr/sadm/bin/smpatch remove patch-ID-revision
```

3. パッチが削除されたことを確認します。

```
# patchadd -p | grep patch-ID-revision
```

例 — 署名のない Solaris パッチの削除

次の例では、`patchrm` コマンドを使用して Solaris 8 パッチ (パッチ番号 111879-01) を削除します。

```
# patchrm 111879-01
```

```
Checking installed patches...
```

```
Backing out patch 111879-01...
```

```
Patch 111874-02 has been backed out.
```

```
# showrev -p | grep 111879-01
```

```
#
```

次の例では、`smpatch remove` コマンドを使用して Solaris 9 パッチを削除します。

```
# /usr/sadm/bin/smpatch remove -i 115028-01
```

```
Authenticating as user: root
```

```
.
```

```
.
```

```
.
```


索引

A

- Admintool
 - パッケージの削除 (手順), 267
 - パッケージの追加 (手順), 265, 267
 - パッケージの追加と削除
 - 概要, 265
- ASN.1 (Abstract Syntax Notation 1), 214

B

- banner コマンド (PROM), 167
- base64, 214
- basedir キーワード (管理ファイル), 221, 222
- bin グループ, 59
- Bourne シェル
 - 「ユーザー初期設定ファイル」も参照
 - 環境変数, 81, 85
 - 基本機能, 80
 - シェル (ローカル) 変数, 81, 83
- Break キー, 177, 179

C

- CD-ROM デバイス
 - マウント済み CD からのソフトウェアの追加例, 257
- CDPATH 環境変数, 82
- .cshrc ファイル
 - カスタマイズ, 64, 80, 86
 - 説明, 78

C シェル

- 環境変数, 81, 85
- 基本機能, 80
- シェル (ローカル) 変数, 81, 83
- ユーザー初期設定ファイル, 78, 86, 92
 - 「ユーザー初期設定ファイル」を参照
- サイト初期設定ファイルの参照用, 79
- 作成, 80

D

- daemon グループ, 59
- DER (Distinguished Encoding Rules), 214
- dfstab ファイル, ユーザーのホームディレクトリの共有, 96

E

- env コマンド, 80
- /etc/dfs/dfstab ファイル, ユーザーのホームディレクトリの共有, 96
- /etc/init.d ディレクトリ, 150
- /etc/inittab ファイル, 141, 143
 - エントリの説明, 141
 - デフォルトの例, 142
- /etc/passwd ファイル, 72
 - 修復
 - SPARC, 179
 - 修復 (例)
 - x86, 193
 - 説明, 72

/etc/passwd ファイル (続き)
フィールド, 72
ユーザー ID 番号の割り当て, 59
ユーザーアカウントの削除, 71
/etc/passwd ファイルの修復
SPARC, 179
x86, 193
/etc/shadow ファイル, 説明, 72
/etc/skel ディレクトリ, 78
/etc/vfstab ファイル, 97
/etc ファイル
ユーザーアカウント情報, 58, 72
/export/home ファイルシステム, 63

G

GECOS フィールド (passwdファイル), 73

GID

大きな値, 60
定義, 64
割り当て, 65

GIDs, 59

groups コマンド, 65

group ファイル

説明, 72
フィールド, 75
ユーザーアカウントの削除, 71

H

halt コマンド, 156

history 環境変数, 82

HOME 環境変数, 82

/home ファイルシステム, ユーザーのホーム
ディレクトリ, 63

I

ID 番号

グループ, 59, 64, 65
ユーザー, 59, 60, 70

init コマンド

スタンドアロンシステムのシャットダウン, 161
説明, 156

init 状態, 「実行レベル」を参照

J

Java キーストア, 217

K

keytool コマンド, 217

概要, 250

Korn シェル

環境変数, 81, 85

基本機能, 80

シェル (ローカル) 変数, 81, 83

ユーザー初期設定ファイル, 78, 79, 80, 86, 92
「ユーザー初期設定ファイル」を参照

L

L1+A キー, 177, 179

LANG 環境変数, 82, 84, 85

LC 環境変数, 84, 85

LK パスワード, 71, 75

local.cshrc ファイル, 78

local.login ファイル, 78

local.profile ファイル, 78

locale 環境変数, 82

.login ファイル

カスタマイズ, 64, 80, 86

説明, 78

LOGNAME 環境変数, 82

LPDEST 環境変数, 82

M

MAIL 環境変数, 81, 82

MANPATH 環境変数, 82

N

newgrp コマンド, 65

NIS

ユーザーアカウント, 58, 72

NIS+

- グループ, 65
- ユーザーアカウント, 58, 72
- noaccess ユーザー/グループ, 59, 76
- noask_pkgadd 管理ファイル, 222, 258
- nobody ユーザー/グループ, 59, 76
- NP パスワード, 75

O

- OS サーバー, 説明, 112

P

- passwd ファイル, 72
 - 修復
 - SPARC, 179
 - 修復 (例)
 - x86, 193
 - フィールド, 72, 73
 - ユーザー ID 番号の割り当て, 59
 - ユーザーアカウントの削除, 71
- patchadd コマンド
 - 署名付きパッチ, 210
 - 署名付きパッチの追加 (手順), 280
- PatchPro, キーストア, 217
- patchrm コマンド, 286
- PATH 環境変数
 - 設定, 83, 84
 - 説明, 82, 83
- path シェル変数, 81
- PC BIOS (ブート), 200
- PEM (Privacy Enhanced Message), 214
- PKCS7 (Public Key Cryptography Standard #7), 214
- pkgadd コマンド
 - a オプション (管理ファイル), 221, 222, 256, 258
 - d オプション (デバイス名), 256, 257, 258, 259, 260, 261
 - s オプション (スプールディレクトリ), 259, 260
 - 概要, 218, 223
 - 使用にあたっての前提条件, 220
 - 署名付きパッケージ, 210
 - 署名付きパッケージの追加, 254
 - pkgadd コマンド (続き)
 - スプールディレクトリと, 259
 - スプールディレクトリと (例), 261
 - 代替ベースディレクトリと, 222
 - パッケージの追加 (手順), 256
 - HTTP URL の使用, 259
 - ユーザーの対話操作を省略, 221, 222
- pkgadm addcert コマンド, 「pkgadm コマンド」を参照
- pkgadm listcert コマンド, 「pkgadm コマンド」を参照
- pkgadm removecert コマンド, 「pkgadm コマンド」を参照
- pkgadm コマンド
 - pkgadm addcert コマンド
 - 概要, 250
 - 信頼される証明書のインポート, 250
 - pkgadm listcert コマンド
 - 概要, 250
 - 出力, 214
 - 証明書の情報の表示, 252
 - pkgadm removecert コマンド
 - 概要, 250
 - 証明書の削除, 253
 - 概要, 223
- pkgchk コマンド
 - 概要, 223
 - 使用 (例), 263
- pkginfo コマンド
 - インストール済みのすべてのパッケージの表示 (例), 261
 - 概要, 220, 223
 - 使用方法, 261
- pkgparam コマンド, 概要, 223
- pkgrm コマンド
 - rm コマンドと, 220
 - rm コマンドとの対比, 264
 - 概要, 218, 223, 264
 - 使用にあたっての前提条件, 220
 - 注意, 220
 - 注意事項, 264
 - パッケージの削除 (手順), 264
- pkgtrans コマンド, 概要, 223
 - /pkg ディレクトリ, 261
- PKI (Public Key Infrastructure) サイト, 217
- prodreg コマンド, 210
 - インストールした製品情報の表示 (手順), 232

prodreg コマンド (続き)
概要, 223, 232
ソフトウェア製品間の依存関係の確認 (手順)
 , 237
ソフトウェア属性の一覧 (手順), 235
ソフトウェアのアンインストール (手順)
 , 241
損傷を受けたソフトウェアのアンインストール (手順), 245
損傷を受けたソフトウェアの再インストール (手順), 248
損傷を受けたソフトウェアの特定 (手順)
 , 238

Product Registry

インストールした製品情報の表示 (手順)
 , 228, 232
ソフトウェア製品間の依存関係の確認 (手順)
 , 237
ソフトウェア属性の一覧 (手順), 235
ソフトウェアのアンインストール (手順)
 , 230, 241
ソフトウェアのインストール (手順), 229
ソフトウェアの削除, 227
ソフトウェアの追加, 227
損傷を受けたソフトウェアのアンインストール (手順), 245
損傷を受けたソフトウェアの再インストール (手順), 248
損傷を受けたソフトウェアの特定 (手順)
 , 238
目的, 227

.profile ファイル

カスタマイズ, 64, 80, 86
説明, 78

PROM

PROM リビジョン番号の確認, 167
ブート設定の変更, 169
モニター, 199
リビジョンレベルの確認, 167

prompt シェル変数, 82

PROM からのブートの設定, 169

PROM のリビジョンレベル, 167

PS1 環境変数, 82

R

reboot コマンド, 156

removef コマンド, 220

reset コマンド, 171

S

/sbin/rc0 スクリプト, 145

/sbin/rc1 スクリプト, 145

/sbin/rc2 script, 146

/sbin/rc3 スクリプト, 148

/sbin/rc5 スクリプト, 148

/sbin/rc6 スクリプト, 148

/sbin/rcS スクリプト, 148

setenv コマンド, 81

set コマンド, 81

shadow ファイル

説明, 72

フィールド, 74, 75

SHELL 環境変数, 83

shutdown コマンド

サーバーのシャットダウン, 134

サーバーのシャットダウン (手順), 158

説明, 156

ユーザーへの通知, 157

/skel ディレクトリ, 78

smpatch remove コマンド, 286, 287

smpatch コマンド, 210

Solaris Product Registry

インストールした製品情報の表示 (手順)

, 228

ソフトウェア製品間の依存関係の確認 (手順)

, 237

ソフトウェア属性の一覧 (手順), 235

ソフトウェアのアンインストール (手順)

, 230, 241

ソフトウェアのインストール (手順), 229

ソフトウェアの削除, 227

ソフトウェアの追加, 227

損傷を受けたソフトウェアのアンインストール

(手順), 245

損傷を受けたソフトウェアの再インストール

(手順), 248

損傷を受けたソフトウェアの特定 (手順)

, 238

目的, 227

Solaris Web Start, ソフトウェアの追加 (手順)

, 225

Solaris Device Configuration Assistant (デバイス構成用補助), 概要, 185
Solaris 管理コンソール
RBACと組み合わせて使用, 38
起動(手順), 42
使用する理由, 32
説明, 29
ツールの説明, 30
Solaris ユーザー登録, 「ユーザー登録」を参照
staff グループ, 65
Stop+A キー, 177, 179
stty コマンド, 84
SunSolve, 信頼される証明書, 217
SUNw接頭辞, 220
Sun ソフトウェアパッケージ
インストール, 258
追加(例), 257
sync コマンド, 179, 181
sync コマンドによるファイルシステムの同期化, 179, 181

T

TERMINFO 環境変数, 83
TERM 環境変数, 83
ttytype タイプの疑似ユーザーログイン, 59
tty(疑似), 59
TZ 環境変数, 83

U

UID, 70
大きな値, 60
定義, 59
割り当て, 60
umask コマンド, 85
UNIX グループ, 64
uucp グループ, 59

V

/var/sadm/install/admin ディレクトリ, 221
/var/sadm/patch, 283
/var/spool/pkg ディレクトリ, 259, 261

W

who コマンド, 140, 157

X

X.509, 214

あ

アクセス権, 85
アプライアンス, 定義, 108
暗号化, 72

い

一次グループ, 65

か

鍵、ユーザー, 「ユーザー鍵」を参照
確認

pkginfo コマンドによるソフトウェア
パッケージのインストール, 259
PROM リビジョンレベル, 167
システムにログインしているユーザー, 157
システムの実行レベル(手順), 140
ソフトウェアのインストール(例), 263
ソフトウェアパッケージのインストール
pkginfo コマンド, 259

環境変数

LOGNAME, 82
LPDEST, 82
PATH, 82, 84
SHELL, 83
TZ, 83
説明, 80, 85

き

キーストア, 213
疑似 tty, 59
疑似ユーザーログイン, 59

起動, ディスクレスクライアント (手順), 123
共有

ユーザーのホームディレクトリ, 97
ユーザーのホームディレクトリ (手順), 96

く

クリーンなシャットダウン, 156

グループ

ID 番号, 59, 64, 65

UNIX, 64

アクセス権の設定, 85

一次, 65

一次グループの変更, 65

管理のガイドライン, 64, 65

情報の格納, 72, 75

説明, 57, 64

デフォルト, 65

名前

説明, 64

名前の説明, 64

二次, 65

ネームサービス, 65

ユーザーが所属するグループの表示, 65

グループ ID 番号, 59, 64, 65

け

検査, インストール済みのパッケージ (例), 263

さ

サーバー

OS サーバー, 112

説明, 107

サーバーとクライアントのサポート, 説明, 105

サービスの起動と停止, 149

最小値, ユーザーログイン名の長さ, 58

最小文字数, ユーザーパスワードの長さ, 62

再設定, SPARC システム, 171

最大数, ユーザーが所属できる二次グループ, 65

最大値

ユーザー ID 番号, 59

ユーザーログイン名の長さ, 58

最大文字数, ユーザーパスワードの長さ, 62

サイト初期設定ファイル, 79

削除

Solaris Product Registry によるソフトウェア
の削除, 227

管理ファイルによるパッケージの削除, 222

署名のないパッチ

概要, 286

手順, 286

ソフトウェアパッケージ

ガイドライン, 220

ソフトウェアパッケージ (手順), 264

ディスクレスクライアント OS サービス (手
順), 124

ディスクレスクライアント OS サービス (例
, 124

ユーザーのホームディレクトリ, 71

ユーザーのメールボックス, 71

し

シェル

環境, 80, 83

環境変数, 80, 81, 85

基本機能, 80

ユーザー初期設定ファイル, 77, 79, 80, 86, 92

ローカル変数, 81, 83

シェル変数, 81, 83

シェル変数のエクスポート, 81

時間帯の環境変数, 83

システムアカウント, 59

システムシャットダウンコマンド, 156

システム初期設定ファイル, 64

システムタイプ

アプライアンス, 108

概要, 106

サーバー, 107

スタンドアロンシステム, 107

選択のガイドライン, 108

ディスクレスクライアント, 108

実行制御スクリプト, 144

サービスの起動と停止, 149

追加 (手順), 150

無効 (手順), 151

実行レベル

0 (電源切断レベル), 140

1 (シングルユーザーレベル), 140

2 (マルチユーザーレベル), 140

- 実行レベル (続き)
 - 3 (NFS を使用できるマルチユーザー)
 - ブート, 172
 - 3 (NFS を使用できるマルチユーザー状態)
 - ブート, 186
 - 3 (NFS を使用できるマルチユーザーレベル)
 - , 140
 - システムが移行すると実行される処
 - 理, 143
 - 実行されるプロセス, 143
 - 6 (リブートレベル), 140
 - s または S (シングルユーザー状態)
 - ブート, 173
 - s または S (シングルユーザーレベル), 140
 - ブート, 187
 - 確認 (手順), 140
 - 定義, 139
 - デフォルトの実行レベル, 139
 - 自動マウント, ユーザーのホームディレクト
 - リ, 64
 - シャットダウン
 - shutdown と init コマンドによるクリーン
 - なシャットダウン, 156
 - サーバー (手順), 157
 - システム、ガイドライン, 134
 - スタンドアロンシステム (手順), 161
 - シャットダウン時間のユーザーへの通知, 157
 - 障害追跡, ディスクレスクリクライアントの問
 - 題, 128
 - 証明書
 - 削除, 253
 - 表示, 252
 - 証明書、信頼される
 - インポート, 250
 - 概要, 214
 - 取得, 217
 - 定義, 213
 - 証明書発行局, 217
 - 初期設定ファイル, システム, 64
 - 署名付きパッチ
 - 「パッチ」も参照
 - 最適な追加方法, 273
 - ダウンロード (手順), 280
 - 追加 patchadd (手順), 280
 - シングルユーザーレベル, 「実行レベル s または
 - S」を参照
- す
 - スーパーユーザー (ルート)、なる, 36
 - スーパーユーザー (ルート) になる, 36
 - スーパーユーザー (ルート) のパスワード、忘れ
 - る
 - SPARC, 179
 - x86, 194
 - スケルトンディレクトリ (/etc/skel), 64, 71
 - スタンドアロンシステム, 説明, 107
 - スプールディレクトリ
 - ソフトウェアパッケージのインストール (手
 - 順), 259
 - ソフトウェアパッケージのインストール (例)
 - , 261, 263
 - すべてのデバイスの電源を落とす、手順, 163
 - せ
 - セキュリティ, ユーザー ID 番号の再利用, 60
 - そ
 - その他 (アクセス権の設定), 85
 - ソフトウェア管理
 - ツール, 218
 - パッケージ, 212
 - パッケージの命名規則, 220
 - ソフトウェアパッケージ
 - インストール, 261
 - スプールディレクトリからのインストール
 - (例), 260
 - た
 - ダウンロード
 - 署名のないパッチ
 - 手順, 284
 - つ
 - 追加
 - Solaris Product Registry によるソフトウェア
 - の追加, 227
 - 管理ファイルによるパッケージの追加, 221

追加 (続き)

- サーバーとクライアントのサポート
 - 説明, 105
- 実行制御スクリプト (手順), 150
- 署名のないパッチ
 - 概要, 283
- 署名のないパッチで patchadd コマンドを使用
 - 手順, 285
- 署名のないパッチで smpatch コマンドを使用
 - 手順, 285
- スプールディレクトリからのパッケージの追加 (例), 261
- スプールディレクトリへのパッケージの追加 (例), 263
- ディスクレスクライアント OS サービス (手順), 120
- ディスクレスクライアントの OS サービスの追加準備 (手順), 118
- パッケージ (前提条件), 220
- パッケージ、例, 258
- 複数バージョンのパッケージ, 221
- ベースディレクトリによるパッケージの追加, 222
- マウント済みの CD からパッケージを (例), 257
- ユーザー初期設定ファイル, 71
- リモートパッケージサーバーからパッケージを (例), 258

て

停止

- 復元のためにシステムを
 - SPARC, 177
- 復元のためにシステムを (方法)
 - x86, 191

ディスクレスクライアント

- OS サービスの削除 (手順), 124
- OS サービスの削除 (例), 124
- OS サービスの追加準備 (手順), 118
- OS サービスの追加 (手順), 120
- 起動 (手順), 123
- 説明, 108

ディスクレスクライアント管理コマンド

- smosservice
 - OS サービスの追加, 112

ディレクトリ

- PATH 環境変数, 82, 83, 84
- アクセス制御, 57, 85
- スケルトン, 64, 71
- ベースディレクトリ (basedir), 221, 222
- ホーム, 63
- ユーザーアカウント用の所有権の変更, 70
- デバイスの電源を落とす場合, 162

な

名前

- SUNW 接頭辞, 220
- グループ
 - 説明, 64
- ソフトウェアパッケージの命名規則, 220
- ユーザーログイン
 - 説明, 57, 58
 - 変更, 70

に

- 二次グループ, 65

ね

- ネームサービス
 - グループ, 65
 - ユーザーアカウント, 58, 72

は

- パスワード (ユーザー)
 - *LK* パスワード, 71, 75
 - NP パスワード, 75
- 暗号化, 72
- 設定, 62, 71
- 説明, 57, 62, 63
- 選択, 62
- 注意点, 62

- パスワード (ユーザー) (続き)
 - 変更
 - 頻度, 62, 75
 - ユーザーツール, 71
 - ユーザーによる変更, 62
 - 有効期限, 63, 71, 72, 75
 - ユーザーアカウントの無効化/ロック, 71, 75
 - ユーザーツール, 71
 - パッケージ
 - 概要, 212
 - 署名付き
 - 「パッケージ、署名付き」を参照
 - 追加
 - 「pkgadd コマンド」も参照
 - 定義, 212
 - パッケージキーストア, セットアップ, 217
 - パッケージ、署名付き
 - 概要, 213
 - 証明書の削除, 253
 - 証明書の情報の表示, 252
 - 信頼される証明書のインポート, 250
 - 追加, 254
 - パッチ
 - patchadd コマンドを使用した追加 (例), 285
 - patchadd コマンドを使用して追加する方法 (手順), 285
 - smpatch コマンドを使用した追加 (例), 285
 - smpatch コマンドを使用して追加する方法 (手順), 285
 - Sun Service カスタマが利用できるもの, 271
 - WWW (World Wide Web) からアクセス, 271
 - 一般的な利用可能性, 271
 - インストール用 README, 272
 - 管理, 275
 - 削除, 287
 - 削除 (手順), 286
 - 情報の表示, 282
 - 情報の表示 (手順), 282
 - 署名付き, 211
 - 追加, 213
 - 定義, 270
 - 署名のないパッチのダウンロード, 284
 - 追加するためのツール, 283
 - ツールとコマンド (概要), 272
 - 定義, 270
 - 入手先, 271
 - 番号付けの方式, 272
 - パッチ、署名付き, 「パッチ」を参照
- ## ひ
- 表示
 - インストール済みのソフトウェアの情報, 261
 - 環境変数, 80
 - パッケージ情報 (例), 261
 - パッケージに関する詳細情報 (例), 262
 - ユーザーマスク, 85
- ## ふ
- ファイル
 - アクセス制御, 57, 85
 - 新しくインストールしたパッケージの属性の確認, 263
 - ユーザーアカウント用の所有権の変更, 70
 - ファイルとディレクトリへのアクセス制御, 57, 85
 - ブート
 - PC BIOS, 200
 - Solaris Device Configuration Assistant (デバイス構成用補助) (方法)
 - x86, 185
 - カーネルデバッグを使って (手順)
 - SPARC, 179
 - x86, 196
 - クラッシュダンプを強制してリポート (手順)
 - SPARC, 180
 - x86, 197
 - システム、ガイドライン, 134
 - 実行レベル 3
 - SPARC, 172
 - 実行レベル 3 (手順)
 - x86, 186
 - 実行レベル S
 - SPARC, 173
 - 実行レベル S (手順)
 - x86, 187
 - 対話式 (手順)
 - SPARC, 174
 - x86, 188
 - ネットワークから
 - SPARC, 175

ブート、ネットワークから (続き)

x86, 190

復元のために (手順)

SPARC, 177

x86, 191

ブートタイプ、説明, 133

ブートプロセス

x86, 206

説明 (SPARC), 200

複数バージョンのソフトウェアパッケージ

ジ, 221, 222

プライマリ管理者の役割

作成 (概要), 40

作成 (手順), 41

引き受ける (手順), 41

へ

ベースディレクトリ (basedir), 221, 222

別名, ユーザーログイン名との比較, 59

変更

デフォルトのブートデバイス

SPARC, 169

ユーザー ID 番号, 70

ユーザーアカウント用のディレクトリ所有権
の変更, 70

ユーザーアカウント用のファイル所有権の変
更, 70

ユーザーパスワード

頻度, 62, 75

ユーザーツール, 71

ユーザーによる変更, 62

ユーザーログイン名, 70

変数

環境, 80, 85

シェル (ローカル), 81, 83

ま

マウント

ユーザーのホームディレクトリ

自動マウント, 64

リモート, 97

ユーザーのホームディレクトリ (手順), 97

マルチユーザーレベル, 「実行レベル 3」を参照

む

無効

実行制御スクリプト (手順), 151

ユーザーアカウント

パスワード, 71, 75

ユーザーツール, 71

め

メール別名, ユーザーログイン名との比較, 59

も

モニター (PROM), 199

ゆ

ユーザー ID 番号, 59, 60, 70

ユーザーアカウント, 57

ID 番号, 59, 60, 70

ガイドライン, 58, 64

情報の格納, 58, 72

設定

記録シート, 90

説明, 57

ネームサービス, 58, 72

無効化/ロック

パスワード, 71, 75

ユーザーツール, 71

ログイン名, 57, 58, 70

ユーザー鍵, 213

ユーザー初期設定ファイル

Bourne シェル, 77

カスタマイズ, 77, 86

概要, 64, 77, 78

カスタマイズしたファイルを追加, 71

環境変数, 81, 85

サイト初期設定ファイル, 79

シェル変数, 81, 83

ユーザーマスクの設定, 85

ローカルシステムへの参照を避ける, 79

カスタマイズ (手順), 91

シェル, 77, 79, 80, 86

説明, 57, 64, 77

デフォルト, 78

- ユーザー初期設定ファイル (続き)
 - 例, 86
- ユーザー初期設定ファイルのカスタマイズ (手順), 91
- ユーザーツール
 - アカウントの無効化, 71
 - パスワードの管理, 71
- ユーザー登録
 - solregis コマンド, 99
 - 説明, 100
 - 無効にする, 102
 - 問題, 101
- ユーザーの初期設定ファイル, 説明, 64
- ユーザーのホームディレクトリ
 - カスタマイズした初期設定ファイル, 64, 71
 - 共有, 97
 - 共有 (手順), 96
 - 削除, 71
 - 所有権の変更, 70
 - 説明, 57, 63
 - マウント
 - 自動マウント, 64
 - リモート, 97
 - マウント (手順), 97
 - ローカル以外からの \$HOME への参照, 79
 - ローカル以外の位置からの参照 (\$HOME), 63
- ユーザーパスワードの有効期限, 63, 71, 72
- ユーザーマスク, 85
- ユーザーログイン (疑似), 59
- ユーザーログイン名
 - 説明, 57, 58
 - 変更, 70

り

- リモートパッケージサーバー
 - スプールディレクトリへのパッケージの追加 (例), 260
 - ソフトウェアのインストール, 258
 - ソフトウェアのインストール (例), 258
- リモートマウント, 97

る

- ルート (スーパーユーザー)、なる, 36

- ルートのパスワードの復元 (手順)
 - SPARC, 179
 - x86, 194
- ルートのパスワード、忘れる
 - SPARC, 179
 - x86, 194
- ルートのパスワードを忘れる
 - SPARC, 179
 - x86, 194

ろ

- ログイン名 (ユーザー)
 - 説明, 58
 - 変更, 70

わ

- ワールド (アクセス権), 85

