IRIX® Admin: Backup, Security, and Accounting （日本語版）

ドキュメント番号 007-2862-004JP

編集協力者

著作　Karen Johnson、Jeffrey B. Zurschmeide、John Raithel、Bill Tuthill

イラスト　Dany Galgani

製作　Heather Hermstad

技術協力　Rob Bradshaw、Chuck Bullis、Robert Clark、Dave Olson、Rebecca Underwood、Vesna Vrdoljak、Supriya Wickrematillake、Laura Wirth-Peters、Donna Yobs

このマニュアルの変更点

第 2 章「restore によるファイルシステムの回復」でファイルシステムの復元に関する情報が変更されています。ここでは、ルート・ファイルシステムの復元方法を説明しています。

改訂履歴

バージョン	説明
004	1999 年 2 月 IRIX 6.5.3 リリース情報の追加

目次

このマニュアルの変更点 iii

改訂履歴 v

図一覧 xvii

表一覧 xix

IRIX Admin マニュアル・セット xxi

このマニュアルについて xxiii

このマニュアルの対象読者 xxiii

このマニュアルの内容 xxiii

第I部 xxiii

第II部 xxiv

第III部 xxiv

このマニュアルの使い方 xxiv

表記上の決まり xxv

参考資料 xxv

書籍 xxvi

インターネット上のリソース xxvi

システム・セキュリティに関するWWW上のリソース xxvi

USENETニュース・グループ xxviii

有料および無料のプロダクト xxviii

PART I　バックアップ

1.　バックアップ体制の導入計画 3
バックアップ・メディアの種類 3
IRIX バックアップ・ツール 4
IRIX NetWorker. 6
バックアップ計画 7
バックアップの実施時期およびバックアップの対象 8
ルート・ファイルシステムのバックアップ 8
ユーザ・ファイルシステムのバックアップ 9
インクリメンタル・バックアップのスケジュール 10
ネットワークを通じてのファイルのバックアップ 10
cron によるバックアップの自動化 11
バックアップの保存 12
バックアップの保持期間 13
テープの再利用に関するガイドライン 13

2.　バックアップと回復の手順 15
バックアップ手順 16
システムのバックアップ・ツール 17
システム破壊後のデータの回復 18
デフォルトのバックアップ・デバイスの変更 22
データ圧縮したファイルのバックアップ 24
Backup ユーティリティと Restore ユーティリティ 24
Backup によるデータの保存. 25
Restore によるデータのリストア 26
dump と restore について 27
dump によるファイルシステムのバックアップ 28
インクリメンタル・バックアップの実行 28
restore によるファイルシステムの回復 29
restore による個々のファイルの回復 30

xfsdump と xfsrestore について 31
xfsdump と xfsrestore の機能 32
xfsdump のメディア形式 34
各種の xfsdump 形式 35
xfsdump によるデータの保存 40
xfsdump によるローカル・メディアの指定 41
xfsdump によるリモート磁気テープ・ドライブの指定 42
xfsdump によるファイルへのバックアップ 44
xfsdump による磁気テープの再利用 44
使用済み磁気テープの消去 45
インクリメンタル・ダンプとダンプの再開 46
xfsdump によるインクリメンタル・ダンプの実行 46
xfsdump によるダンプの再開 47
xfsdump アーカイブの検査. 48
xfsrestore について 51
xfsrestore による単純なリストアの実行 53
xfsrestore による個々のファイルのリストア 55
xfsrestore によるネットワークを通じたリストアの実行 56
xfsrestore の対話モードによるリストアの実行 56
xfsrestore による漸増リストアの実行 58
xfsrestore の中断 61
housekeeping ディレクトリと orphanage ディレクトリについて 62
xfsdump と xfsrestore によるファイルシステムのコピー 63
tar について 64
tar によるファイルのバックアップ 64
tar による変更日付別のファイルのバックアップ 64
tar によるインクリメンタル・バックアップの実行 65
tar パフォーマンスの向上 65
tar アーカイブの検査 66
tar アーカイブのリストア 66

cpioについて 67
cpioによるファイルのバックアップ 67
cpioによる変更日付別のファイルのバックアップ 68
cpioによるインクリメンタル・バックアップの実行 68
cpioアーカイブの検査 69
cpioアーカイブのリストア 69
ddについて 69

3. バックアップと回復のトラブルシューティング 71
読取り不能なバックアップのトラブルシューティング 71
他のシステムで作成したメディアの読取り 72
バックアップ中のエラーのトラブルシューティング 74
誤ったバックアップをリストアした後の正しいバックアップのリストア 75
不良メディアの検査 76
バックアップと回復に関するエラー・メッセージと対策 77

PART II セキュリティ

4. IRIXシステム・セキュリティ 81
システム・セキュリティについて 82
IRIXの標準セキュリティ機能 82
セキュリティの保護と注意 83
パスワード管理 87
パスワード作成のガイドライン 87
PROMパスワードについて 88
nvramによるPROMパスワードの消去 89
コマンド・モニタからのPROMパスワードの設定 89
第2（ダイヤルアップ）パスワードの設定 90
シャドウ・パスワードについて 93
シャドウ・パスワード・ファイルの使用 93

パスワード・エージングについて 94
passwd コマンドによるパスワード・エージングの管理 94
/etc/passwd の編集によるパスワード・エージングの管理 95
pwck によるパスワード・ファイルのチェック 97
システム・ログインとアカウントの管理 97
特殊アカウントについて 98
使用しないログインのロック 99
システム・ログイン・オプション 100
root によるログインの制限 101
ログイン試行の制限（MAXTRYS） 101
回線の使用禁止時間の設定（DISABLETIME） 102
ログイン試行の記録 103
パスワードの強制 103
アカウントの無効化（LOCKOUT） 103
最後のログイン時刻の表示 104
set-UID と set-GID のパーミッションについて 104
root が所有する set-UID ファイルの検査 105
root ファイルシステムでの set-UID ファイルの検査 105
root 以外のファイルシステムでの set-UID ファイルの検査 107
汎用のファイルとディレクトリのパーミッションについて 107
パスワードなしで出荷されるアカウント 108
セキュリティ関連のファイルとコマンド・リファレンス 109
セキュリティの強化機能 110
アクセス・コントロール・リスト（ACL: Access Control Lists） 111
ACL の長いテキスト形式 112
ACL の短いテキスト形式 114
ls -D と chacl の使用 115

最小限の特権機能 116
/etc/capability ファイル 117
本リリースの特権機能 120
ファイル特権機能 128
カスタム特権機能の作成 128
特権機能の問題を解決する attrinit の使用 129

5. ネットワークのセキュリティ 131

ローカル・エリア・ネットワークのアクセス 132
ネットワーク・アクセス制御ファイル 132
ローカル inetd サービス 133
X11 ネットワーク・アクセス 133
セキュリティと X サーバの初期設定 134
X0.hosts ファイルによるアクセスの制限 134
xhost コマンドによるアクセスの制限 135
xhost コマンドの対話モードでの使い方 135
X 権限 136
ネットワーク・セキュリティとファイアウォールについて 137
インターネットについて 137
ネットワークのセキュリティ上の問題 138
ファイアウォールについて 138
ファイアウォールの設計方針 140
ファイアウォールのモニタリング 140
World Wide Web のセキュリティ上の問題. 141
ファイアウォールのハードウェアの設定 141
ルータとファイアウォール 142
ファイアウォールとして機能するハードウェアの設定 143
二重ホーム・ホストのファイアウォール 143
隠蔽されたホストのゲートウェイ 143

セキュリティに関するIRIXの設定 146
二重ホーム・ホストでのネットワーク・ソフトウェアの設定 146
IRIXでのセキュリティの強化 146
IPパケットの転送禁止 147
inetdサービスの制限 148
ファイアウォールでのパスワードの保護 150
ファイアウォールでのrpcサービス・アクセスの制限 151
ファイアウォールでのNIS（YP）の無効化 151
ファイアウォールでのNFSアクセスの禁止 152
ファイアウォールでのログ・ファイルについて 152
ファイアウォールでのソフトウェアの整合性検査 153
ファイアウォールに関するユーザの教育 154
内部ネットワークのセキュリティの設定 154
ドメイン・ネーム・システム（DNS: Domain Name System）のセキュリティに関するガイドライン . 154
メールの設定のセキュリティに関するガイドライン 155
Sendmailの設定とメールのエリアス 155
メール・スプールの分割 156
プロクシ・サーバについて 156

PART III アカウンティング

6. システム監査トレールの管理 159
MACとDACについて 160
監査プロセスの起動 161
デフォルトの監査 162
監査のカスタマイズ 163
監査対象の活動 163
監査可能なイベント 165
satconfigについて 169
satconfigの使い方 170

sat_selectについて 170
sat_selectの使い方 171
監査環境の保存と検索 171
監査ファイルの格納 172
特定のユーザの監査 173
セキュリティ違反を確認するための監査 173
ユーザの活動の監査 174
ファイルの監査 174
Trusted IRIX/Bでのラベルの監査 175
監査データについて 175
セキュリティ違反について 176
部外者による使用と悪用 177
システムへの無許可侵入の試行 177
通常と異なる時間帯または場所からのシステム使用 178
ローカル・ネットワーク外のマシンとの接続 178
部内者によるシステムの使用と悪用 179
部内者によるファイル・パーミッションの違反 179
部内者によるルート特権の乱用 180
特定の部内者による活動 181
特定のファイルまたはリソースへのアクセス 181
システム管理の適否について 181
システムのデータ・ファイルの変更 181
システム・プログラムの属性の変更 182
監査追跡の取扱い 183
監査データのアーカイブ 183
監査データの削除 183
監査ファイルのオーバーフローについて 184
監査ファイルのオーバーフローからの回復 184

7. システム・アカウンティング 187
プロセス・アカウンティング・システムについて 188
プロセス・アカウンティング・システムの機能 188
プロセス・アカウンティングの有効化 189
プロセス・アカウンティングの無効化 190
アカウンティング・ファイルとアカウンティング・ディレクトリ 190
アカウンティング・ファイルのサイズの管理 191
/var/admディレクトリのファイル 191
/var/adm/acct/niteディレクトリのファイル 192
/var/adm/acct/sumディレクトリのファイル 193
/var/adm/acct/fiscalディレクトリのファイル 193
日別システム・アカウンティングについて 194
アカウンティング・システムの設定 195
runacctによる日別システム・アカウンティング 196
runacctのサマリ・ファイル 196
runacctの再入可能な状態 197
runacctの障害からの復旧 199
runacctの再起動 200
破壊されたアカウンティング・ファイルの修復 200
wtmpエラーの修復 201
tacctエラーの修復 201
アカウンティングのための休日の反映 202
runacctの日別レポート 203
日別使用状況レポート 204
日別コマンド・サマリと月計コマンド・サマリ 205

IRIX の拡張アカウンティング 206
拡張アカウンティングについて 207
拡張アカウンティングの使用 208
配列セッション 209
プロジェクト ID 210

索引 213

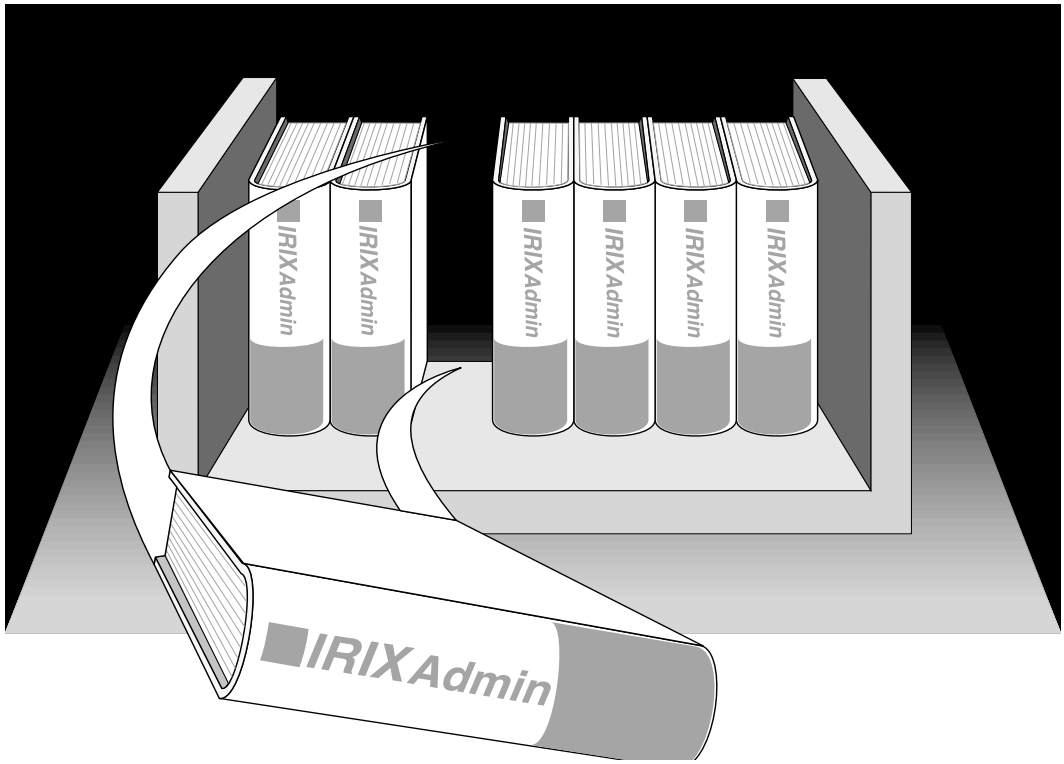
図一覧

図2-1 単一のメディア・オブジェクトに出力された単一のダンプ 35
図2-2 複数のメディア・オブジェクトに分割して出力された単一のダンプ 36
図2-3 単一のメディア・オブジェクトに出力された複数のダンプ 38
図2-4 複数のメディア・オブジェクトに出力された複数のダンプ 39
図5-1 単純なファイアウォールの環境 139
図5-2 隠蔽されたホスト 144
図5-3 隠蔽されたサブネット 145

表一覧

表 1-1 バックアップ・ユーティリティのまとめ 5
表 2-1 ファイルシステムとダンプ・ユーティリティ 32
表 2-2 ファイルシステムとリストア・ユーティリティ 32
表 2-3 tar のファイル比較キー・キャラクタ 66
表 4-1 パスワード・エージングの文字コード 96
表 4-2 IRIX のセキュリティ関連のファイル 109
表 4-3 IRIX セキュリティ・コマンド 110
表 6-1 デフォルトで監査されるイベント 162

IRIX Admin マニュアル・セット



このマニュアルは、IRIX Admin マニュアル・セットの中の 1 冊です。このマニュアルは、サーバ、マルチ・システム、およびファイル構造（ユーザのホーム・ディレクトリと作業用ディレクトリを除く）を管理するシステム管理者を対象としています。システムの保守を任されている方や、IRIX に関して一般のエンド・ユーザ向けのマニュアルよりさらに専門的な知識が必要な場合は、このマニュアル・セットを参照してください。IRIX Admin マニュアルは、オンラインの IRIS InSight で参照できます。

IRIX Admin マニュアル・セットは、次のマニュアルで構成されています。

- 『IRIX Admin: Software Installation and Licensing』— このマニュアルでは、IRIX 上で実行するソフトウェアのインストール方法とライセンス管理方法について説明します。IRIX は、Silicon Graphics 社の UNIX オペレーティング・システムです。このマニュアルでは、IRIX のインストール・ユーティリティのコマンド行インタフェースである Inst を使用してミニルート・インストールとライブ・インストールを行う手順について説明します。また、IRIX で実行する特定のアプリケーションへのアクセスを制限するライセンス管理製品とそのマニュアルも紹介します。
- 『IRIX Admin: System Configuration and Operation』— このマニュアルでは、標準的なシステム管理方法について説明します。また、システム管理に関する作業として、オペレーティング・システムの設定、ユーザ・アカウント、ユーザ・プロセス、ディスク・リソースの管理、PROM モニタを介したシステムの操作、およびシステム・パフォーマンスについても説明します。
- 『IRIX Admin: Disks and Filesystems』— このマニュアルでは、ディスク、ファイルシステム、および論理ボリュームの各概念について説明します。また、SCSI ディスク、XFS ファイルシステム、EFS ファイルシステム、XLV 論理ボリューム、および帯域保証 I/O についてのシステム管理手順についても説明します。
- 『IRIX Admin: Networking and Mail』— このマニュアルでは、メール送信、UUCP、SLIP、PPP などを含むネットワーク・システムとメール・システムの計画、設定、使用、管理について説明します。
- 『IRIX Admin: Backup, Security, and Accounting』— このマニュアルでは、ファイルのバックアップとリストア、システムとネットワークのセキュリティ、ユーザ別のシステムの利用記録について説明します。
- 『IRIX Admin: Peripheral Devices』— このマニュアルでは、端末、モデム、プリンタ、CD-ROM、テープ・ドライブなどの周辺デバイスに対するソフトウェアの設定と管理方法について説明します。

このマニュアルについて

ここでは、このマニュアルの概要と、このマニュアルで使用する表記法について説明します。また、参考資料の入手先も示します。

このマニュアルの対象読者

このマニュアルは、IRIX のバックアップ、セキュリティ管理、アカウント業務を担当するシステムおよびネットワーク管理者を対象にしています。パーソナル・ワークステーションを管理する場合は、『Personal System Administration Guide』を参照してください。

このマニュアルの内容

『IRIX Admin: Backup, Security, and Accounting』では、データのバックアップとリストア、ホストとネットワークのセキュリティ、IRIX コンピュータ・サイトでのホスト・リソースの監査とアカウンティングについて説明します。このマニュアルは、次の章で構成されています。

第 I 部

このマニュアルの第 I 部では、バックアップとリストアについて説明します。

- 第 1 章「バックアップ体制の導入計画」— バックアップ・メディアの種類、利用できるツール、バックアップを行うときに考慮する点について説明します。
- 第 2 章「バックアップと回復の手順」— 各バックアップ・ツールについて詳しく説明します。各ツールの使い方の例も示します。
- 第 3 章「バックアップと回復のトラブルシューティング」— バックアップに関するエラーをタイプ別に説明します。一般的なエラー・メッセージも示します。

第 II 部

このマニュアルの第 II 部では、システムとネットワークのセキュリティについて説明します。

- 第 4 章「IRIX システム・セキュリティ」— ローカル・システムのセキュリティを実装する方法について説明します。
- 第 5 章「ネットワークのセキュリティ」— ローカル・エリア・ネットワークのセキュリティを実装する方法と、ネットワークのファイアウォールについて説明します。

第 III 部

このマニュアルの第 III 部では、システム・アカウントの処理と監査について説明します。

- 第 6 章「システム監査トレールの管理」— IRIX システムで発生するすべての活動の監査方法について説明します。
- 第7章「システム・アカウンティング」— システムの利用状況の記録方法について説明します。

このマニュアルの使い方

バックアップに関しては、第 I 部を参照してください。バックアップの方針を決めるには、第 1 章「バックアップ体制の導入計画」を読んでください。各バックアップ・ツールの使い方を習得するには、第 2 章「バックアップと回復の手順」を読んでください。バックアップ作成で問題が生じた場合は、第 3 章「バックアップと回復のトラブルシューティング」を読んでください。

セキュリティに関しては、第 II 部を参照してください。IRIX ホスト・セキュリティの構築方法については、第 4 章「IRIX システム・セキュリティ」を読んでください。ネットワーク・セキュリティについては、第 5 章「ネットワークのセキュリティ」を読んでください。

システムを監査する場合は、第 III 部の第 6 章「システム監査トレールの管理」を参照してください。システムの利用を監視（アカウンティング）するには、第 III 部の第 7 章「システム・アカウンティング」を読んでください。

表記上の決まり

このマニュアルでは、次の表記法を用いています。

『』	ほかのマニュアルのタイトルを表します。
「」	本書のほかの章や節のタイトルを表します。
[ ]	メニュー名やボタン名などの UI（User Interface）を表します。
->	プルダウン・メニューの階層構造を表します。
< >	キーボードのジェネリック・キー（Ctrl、Shift、Alt など）を表します。キーの操作方法として、次に例を示します。
<Enter>	<Enter> キーを押します。
<Alt>-h	<Alt> キーを押しながら h キーを押します。
<Alt>-h c	<Alt> キーを押しながら h キーを押した後、すぐに c キーのみを押します。
<Shift>-<Ctrl>-n	<Shift> キーを押しながら <Ctrl> キーと n キーを同時に押します。
<Ctrl>-x <Ctrl>-c	<Ctrl> キーを押しながら x キーを押した後、すぐに <Ctrl> キーを押しながら c キーを押します。

ほかのマニュアルへのリンクや、アプリケーションなどの実行可能な語句は赤く表示されます。

本書のほかの章、節、または図などへのリンクは青く表示されます。

参考資料

次の書籍、ネットワーク・リソース、プロダクト・リソースは、システムまたはネットワークのセキュリティを確立する際の参考にしてください。

書籍

次の書籍は、システムおよびネットワークのセキュリティに関する追加情報を提供しています。

- William Cheswick and Steven Bellovin. *Firewalls and Internet Security, Repelling the Wily Hacker*. Addison-Wesley. ISBN 0-201-63466-X, second edition 1998.
- Douglas E. Comer and David L. Stevens. *Internetworking with TCP/IP: Client-Server Programming and Applications, BSD Socket Version*, Volume 3. Prentice-Hall, Inc. ISBN 0-13-260969-X, second edition 1996.
- David A. Curry. *UNIX System Security*. Addison-Wesley. ISBN 0-201-56327-4, 1992.
- Simson Garfinkle and Eugene Spafford. *Practical UNIX and Internet Security*. O'Reilly & Associates, Inc. ISBN 1-565921-48-8, second edition 1996.

インターネット上のリソース

セキュリティに関する各種のリソースは、インターネット上でも入手できます。ただし、内容が頻繁に変更されるため、ここではポインタ（URL）のみを示し、詳細な説明は省略します。

システム・セキュリティに関するリソースには、さまざまなニュースグループからの問い合わせの多い質問（FAQ）に対する回答、セキュリティの実務と理論に関するドキュメント、セキュリティに関する新しい話題の伝言板への掲載、セキュリティの問題をインタラクティブに論じるメーリング・リストなどがあります。次に、そのポインタ（URL）を紹介します。

システム・セキュリティに関するWWW上のリソース

セキュリティに関する情報を扱うWWW（World Wide Web）上のURL（universal resource locator）を紹介します。

- http://www.sgi.com/ — Silicon Graphicsプラットフォーム関連の情報とプロダクトについての情報を掲載しています。
- http://www.sgi.com/Support/security/security.html — Silicon Graphicsサポート・サービスが維持管理しているセキュリティ・ページです。
- ftp://sgi.sgigate.com/~ftp/Security — Silicon Graphics社製品のセキュリティに関連するパッチを無償でダウンロードできます。

- http://www.lib.ox.ac.uk/internet/news/faq/comp.security.unix.html — UNIX のセキュリティに関する一般的な FAQ リストを掲載しています。
- http://www.alw.nih.gov/Security/ — さまざまな FAQ を含む、セキュリティに関連した各種のリソースへのリンク・ページです。
- http://www.telstra.com.au/info/security.html — 一般的なネットワーク・セキュリティに関連した各種のリソースに接続します。セキュリティ関連のメーリング・リストも掲載しています。
- http://www.cert.org/ — Computer Emergency Response Team（CERT）Coordination Center は、Advanced Research Projects Agency によって設立された機関であり、インターネット・ユーザのセキュリティ対策に関する情報を取りまとめています。
- http://ciac.llnl.gov/ — U.S. Department of Energy's Computer Incident Advisory Capability（CIAC）ページで、諮問電子掲示板、メーリング・リスト、ドキュメントなどに接続します。
- http://www.faqs.org/faqs/firewalls-faq/ — Firewall FAQ。ファイアウォールについて問い合わせの多い質問事項を掲載しています。
- http://www.faqs.org/faqs/by-newsgroup/comp/comp.security.unix.html — UNIX のセキュリティに関する FAQ をまとめてあります。
- http://www-ns.rutgers.edu/www-security/index.html — World Wide Web（WWW）に関するセキュリティの問題を扱うホーム・ページです。
- http://www.socks.nec.com/ — プロキシについて調べるには、このページから始めてください。プロキシの概要、ダウンロード可能なプロキシ、その他のプロキシに関する情報が掲載されています。

これらの URL は変更されていたり、すでに無効になっている場合もあります。WWW 検索ツールを使用し、「security」、「network security」、「firewall」などのさまざまなキーワードを使って検索し、情報収集をしてください。

USENETニュース・グループ

セキュリティ関連の最新情報が提供されているニュース・グループを紹介します。

- comp.security.unix — UNIX 関連のセキュリティ問題を扱うニュース・グループです。
- comp.security.announce — セキュリティ関連のプロダクトとサービスに関する情報を提供するニュース・グループです。
- comp.sys.sgi.admin — Silicon Graphics 社製プロダクトのユーザを対象として、システム管理の問題を扱うニュース・グループです。
- comp.sys.sgi.announce — Silicon Graphics 社製プロダクトのユーザを対象として、新しいプロダクトとサービスを紹介するニュース・グループです。
- comp.security.firewalls — すべてのプラットフォームに関連するネットワーク・ファイアウォールの問題を扱うニュース・グループです。

有料および無料のプロダクト

Silicon Graphics 社では、IRIX 用に 2 つのセキュリティ・オプションを用意しています。1 つは Commercial Security Pak で、管理者とユーザの両方に高度な特権管理機能を提供します。もう 1 つは Trusted IRIX で、識別、認証、監査機能付きで B1 レベルのセキュリティを提供します。

IRIX 用の Gauntlet やその他のセキュリティ関連プロダクトの詳細については、日本シリコングラフィックス株式会社のサポート部門までお問い合わせください。Silicon Graphics 社は、暗号化およびプロキシ・サーバを通じてインターネット・アクセスの安全性を高める Netscape プロダクトも提供しています。

上記にリストした Web ページでは、ほかのプロダクトも紹介されていますが、Silicon Graphics 社が推奨するプロダクトではない点に注意してください。これらのプロダクトの設定やサポートについては、各ベンダーに問い合わせるか、ユーザ自身で行ってください。

第 I 部

バックアップ

第 I 部のバックアップには次の章が含まれます。

第 1 章
バックアップ体制の導入計画

第 2 章
バックアップと回復の手順

第 3 章
バックアップと回復のトラブルシューティング

バックアップ体制の導入計画

サイト管理者は、必ずファイルをバックアップします。ユーザが間違ってファイルを消してしまった場合や、ハードウェアの故障によりファイルを損失した場合は、管理者の作成したバックアップが必要です。

この章では、以下について説明します。

- 「バックアップ・メディアの種類」（3 ページ）
- 「IRIX バックアップ・ツール」（4 ページ）
- 「バックアップ計画」（7 ページ）

バックアップ業務についての詳細な知識があり、担当のサイトで必要なものが明確な場合は、第 2 章「バックアップと回復の手順」を参照し、利用計画を立てます。

バックアップ・メディアの種類

以下は Silicon Graphics 社のシステムで使用できる一般的なバックアップ・メディアです。

- 1/4 インチ・カートリッジ・テープ、4 トラック
- 8mm カートリッジ・テープ
- 4mm DAT
- DLT（デジタル・リニア・テープ）

個々のシステムに付随するバックアップ装置以外にも、ネットワークに接続し、さまざまな種類と容量のバックアップ装置にアクセスできます。ローカルにアクセスできる装置についてはユーザ・マニュアルを参照してください。また、ネットワークを通じてアクセスできる装置については、各ベンダ提供のマニュアルを参照してください。

メディアの種類によっては、この章で説明する容量制限や使用条件と関係なく使用できるものがあります。たとえば、8mm のカートリッジ・ドライブ（最大記憶容量 1.2GB）を使用して 350MB のファイルシステムをバックアップする場合は、1 本のテープで十分です。テープ容量の詳細については、『IRIX Admin: Peripheral Devices』を参照してください。

最近では、オートチェンジャ、ジュークボックスとも呼ばれるメディア・チェンジャの人気が高まっています。シーケンシャル・モードでは、複数のテープを 1 本の長いテープとして扱うことにより、メディア・チェンジャを標準の IRIX ユーティリティで使用することができます。また、これにより全体容量も向上します。ただし、メディア・チェンジャのすべての機能をフルに利用するには、Open Vault などの専用ソフトウェアが必要です。専用ソフトウェアを使うと、メディア・ライブラリにあるすべてのボリュームにランダムにアクセスできるようになります。アップデート情報については、Silicon Graphics のウェブ・サイト（http://www.sgi.com）の「Open Vault」を参照してください。

IRIX バックアップ・ツール

IRIX システムには、さまざまなバックアップ・ツールが用意されています。どのツールを選択するかは自由です。サイトのユーザが特定のバックアップ・ツールにすでに慣れている場合は、そのツールを一貫して使用できます。他社製のワークステーションがサイトで使用されている場合は、すべてのワークステーションで共通に利用できるバックアップ・ユーティリティを選択します。

IRIX は、データのバックアップを作成するために、以下のユーティリティを用意しています。

- System Manager, Backup & Restore
- `cpio` を使用する Backup(1) と Restore(1)
- dump(1M) と restore(1M)
- XFS ファイルシステム用の xfsdump(1M) と xfsrestore(1M)
- cpio(1)
- dd(1M)
- tar(1)

Silicon Graphics システム用の別売のプロダクトも利用できます。IRIX NetWorker は、データのバックアップと復元を行うための機能を完備したデータ管理ツールです。ユーザは IRIX NetWorker でハイエンド・サーバのデータをバックアップしたり、ネットワーク上のすべてのワークステーションとファイル・サーバのバックアップを集中管理できます。詳細については、後の 6 ページの「IRIX NetWorker」を参照してください。

バックアップ・ツールは、`Backup` と `dump` などのファイルシステム向けと、`tar` と `cpio` などのファイルまたはディレクトリ向けに用途が分かれます。各ツールの用途は一方だけに制限されませんが、その適した用途に用いるのが一般的です。また、`dd` コマンドを使用すると、格納されている画像データをそのまま、または変換して読取ることができます。通常、`dd` はバックアップを作成するためではなく、他のバックアップ・ユーティリティと互換しないフォーマットで書かれたデータを読取るために使用します。

表 1-1 は IRIX で利用できるバックアップ・ユーティリティのリストです。

表 1-1 バックアップ・ユーティリティのまとめ

ユーティリティ	概要	参照先
System Manager Backup & Restore	`cpio` ユーティリティを使用するためのグラフィカル・インタフェースです。使用中のシステムだけをバックアップするときに最適なツールです。	『Personal System Administration Guide』
`Backup` と `Restore`	`cpio` ユーティリティのフロント・エンドのコマンド行です。	Backup(1) と Restore(1) のマン・ページおよび 24 ページの「Backup ユーティリティと Restore ユーティリティ」
`dump` と `restore`	インクリメンタル・バックアップと対話方式のリストアをサポートしています。特殊な環境下でも利用可能な UNIX の標準バックアップ・ユーティリティです。ただし、XFS のバックアップは作成できません。	dump(1M) と restore(1M) のマン・ページおよび 27 ページの「dump と restore について」
`xfsdump` と `xfsrestore`	インクリメンタル・バックアップ、対話方式のリストア、中断後の復帰をサポートしています。XFS には、`dump` と `restore` ではなく、このユーティリティを使用します。	xfsdump(1M) と xfsrestore(1M) のマン・ページおよび 31 ページの「xfsdump と xfsrestore について」
`tar`	最も一般的な UNIX のバックアップ・ユーティリティです。移植性も高く、特殊な環境下でも広く利用されています。	tar(1) マン・ページと 64 ページの「tar について」

表 1-1 バックアップ・ユーティリティのまとめ（続き）

ユーティリティ	概要	参照先
`cpio`	柔軟性に富んだ UNIX の標準コマンドです。通常、コマンド行で他のコマンドと組合わせて使用します。	cpio(1) マン・ページと 67 ページの「cpio について」
`dd`	UNIX の標準コマンドです。入力ファイルを読込み、指定した変換後に出力ファイルに書出します。	dd(1M) マン・ページと 69 ページの「dd について」

IRIX NetWorker

IRIX NetWorker は、特殊な環境下のネットワークでバックアップとアーカイブ記憶領域を管理するための別売のプロダクトです。IRIX NetWorker は、ファイルの拡張属性であるセキュリティ情報、ユーザ・プロファイル、アクセス・コントロール・リストなどを含めて、ネットワークのすべてのデータを完全に保護する信頼性の高いツールです。

IRIX NetWorker を使用すると、Silicon Graphics サーバのデータをバックアップできるだけでなく、その強力な I/O 機能を使用してネットワークのすべてのワークステーションとファイル・サーバのバックアップを集中管理できます。NetWorker は、すべての主要な UNIX システムのほかに、PC、NetWare、Macintosh の各システムに対してネットワーク・バックアップをサポートしています。IRIX NetWorker では、以下の機能を使用できます。

- グラフィカル・ユーザ・インタフェースによって NetWorker を簡単に使用し、管理できます。また、管理者インタフェースによって、どのネットワーク・ノードからのデータ管理操作でも統一したフォーマットで表示できます。
- オプションとしてサポートされている広範なオートチェンジャ（ジュークボックス）を使用すると、無人のバックアップとテープの自動管理を行うことができます。オートチェンジャ（ジュークボックス）によるバー・コード処理を利用すると、人的な手間とメディアの管理時間は大幅に削減されます。
- セーブセット・クローン機能を使用すると、予備とセキュリティのためにバックアップ・データの複数のコピーを作成し、管理できます。
- バックアップと復元の並列実行によって、複数のデータ・ストリームをメディアに同時に読み書きできます。また、装置の並列サポートによって、複数の記憶装置にデータ・ストリームを同時に読み書きできます。データ圧縮によって、バックアップ時間とネットワーク・トラフィックが減少します。

- クライアント・サーバ・アーキテクチャによって、新しいシステムと、Archive や階層型記憶装置管理（HSM: Hierarchial Storage Management）などの高度なデータ管理アプリケーションを簡単に統合できます。
- オンスクリーンのインデックス・ブラウザとスケジューラによって、デスクトップ・ユーザは復元とバックアップを迅速かつ簡単に開始できます。結果として、管理者の時間も節約されます。

別売のバックアップ・プロダクトの詳細については、日本シリコングラフィックス株式会社のサポート部門までお問い合わせください。

バックアップ計画

各サイトではシステムのバックアップに関する方法を決定し、必ずそれに従います。これによって、障害が発生したときにリストアできるデータとリストアできないデータを正確に判断できます。

実際にバックアップを作成する方法は、ワークステーションの構成などの要因によって異なります。その方法にかかわらず、最新のフル・バックアップを少なくとも 2 つは作成します。また、ユーザにも独自のバックアップ、特に重要なファイルやよく変更されるファイルなどのバックアップを作成させます。ユーザのニーズは変わりやすく、データの価値はユーザ自身が一番よく知っているためです。

ワークステーションのユーザは、画面上の［ツールチェスト（Toolchest）］の［システム（System）］->［システム・マネージャ（System Manager）］で重要なファイルのバックアップを作成できます。［システム・マネージャ（System Manager）］の詳細については、『Personal System Administration Guide』を参照してください。ユーザが利用できるカートリッジ・テープなどのバックアップ・メディアは、中古でも新品でも、常に十分な数量を確保します。

1 つのボリュームで最大のファイルシステムのバックアップを作成できるメディアの場合、インクリメンタル・バックアップを使用する必要はありません。ただし、インクリメンタル・バックアップを使用する方が、バックアップの時間を短縮できます。常に複数のボリュームを使用しないとファイルシステムのバックアップを作成できない場合は、インクリメンタル・バックアップによりテープの使用本数を減らすことができます。

以下では、データのバックアップに伴う考慮事項について説明します。

バックアップの実施時期およびバックアップの対象

データのバックアップを作成する頻度は、システムの使用頻度とデータの重要度によって決まります。

交換不可能なデータや入力に手間のかかるデータは必ずバックアップを作成します。

ルート・ファイルシステムのバックアップ

ルート・ファイルシステムとユーザ・ファイルシステムが分かれているシステムの場合、ルート・ファイルシステムが変更されることは少ないので、`/usr` ファイルシステムほど頻繁にバックアップを作成する必要はありません。

ルート・ファイルシステムが変更されるのは、新規ソフトウェアを追加した場合、ハードウェアを再構成した場合、サイトのネットワーキングを変更した場合（システムがサーバであるか、またはネットワーク・インフォメーション・サービス（NIS: Network Information Service）のマスター・ワークステーションである場合)、ワークステーションの構成を一部変更した場合などです。状況に応じて、変更した個々のファイル（`/unix`、`/etc/passwd` など）だけをバックアップできる場合もあります。

個々のファイルをバックアップする作業は、必ずしも簡単ではありません。新規ユーザの追加のような小規模なシステム変更であっても、システム全体のファイルに影響を与えるためです。また、グラフィック表示された［システム・マネージャ（System Manager)］を使用している場合は、どのファイルを変更したのかを忘れてしまうことがあります。さらに、サイトに複数の管理者がいる場合は、他の管理者が行った変更を互いに把握できないことがあります。このような問題を回避するには、システム・マネージャやバックアップなどのファイルシステム向けのバックアップ・ユーティリティを定期的に利用します。

ルート・パーティションは、月に 1 回程度バックアップを作成するのが適切です。以下の場合は、定期的なバックアップのほかに、ルート・ファイルシステムのバックアップを作成します。

- ユーザをシステムに追加したとき（特にシステムが NIS のマスター・ワークステーションである場合）
- 新規ソフトウェアをインストールする直前
- インストールした新規ソフトウェアの正常な動作を確認できたとき

システムの使用頻度が高い場合、またはサイトに複数の管理者がいる場合は、ルート・ファイルシステムのバックアップを定期的に作成します。

ユーザ・ファイルシステムのバックアップ

システム・プログラム（`/usr/bin` など）とユーザ・アカウントの両方を含む `/usr` ファイルシステム[1] は、ルート・ファイルシステムよりも使用頻度が高いため、より頻繁にバックアップを作成します。

通常のマルチユーザ・システムでは、インクリメンタル・バックアップで 1 日に 1 回バックアップを作成します。

ユーザのメール・ボックスの内容などが書込まれている `/var` ファイルシステムについても同様です。

[1] `/usr` が別のファイルシステムでない場合には、ルート・ファイルシステムになります。

インクリメンタル・バックアップのスケジュール

インクリメンタル・バックアップを使用すると、ファイルシステム全体のバックアップを繰返し作成するのと同じ保護レベルを、より少数のテープで実現できます。また、インクリメンタル・バックアップはシステム上のファイル全体のバックアップを作成するよりも短時間でできます。

特定のファイルシステムのインクリメンタル・バックアップを作成するには、次の手順に従います。

1. 最初の日は、ファイルシステム全体のバックアップを作成します。これは月単位のバックアップです。
2. 2 日目から 7 日目までは、前日以降に変更されたファイルだけのバックアップを作成します。これは日単位のバックアップです。
3. 8 日目は、前週以降に変更された全ファイルのバックアップを作成します。これは週単位のバックアップです。
4. 手順 2 から手順 3 を 4 週間（約 1 か月）繰返します。
5. 4 週間（約 1 か月）後、手順 1 から手順 4 を再度繰返します。

日単位のテープは、1 か月ごと、または安全と思われる期間をおいて再利用します。週単位のテープは、数か月間保持します。月単位のテープは約 1 年間保持してから再利用します。

ネットワークを通じてのファイルのバックアップ

ネットワークに接続されたワークステーションが数多く存在するサイトを管理している場合は、中央のワークステーション上のデバイスにバックアップを保存できます。

ネットワークを通じてバックアップを作成する場合も、通常と同様の基本的なバックアップ・コマンドを使用しますが、次に示すように、デバイスの指定方法が多少異なります。

system_name`:/dev/tape`

必要に応じて、リモート・デバイス上のアカウントを指定します。

user`@`*system_name*`:/dev/tape`

ユーザは上記の方法で各自のワークステーションから中央のテープ・ドライブを利用できます。ただし、Silicon Graphics 以外のワークステーション上のリモート・テープ・ドライブにバック

アップを作成する場合は、テープ・ドライブのデバイス名が /dev/tape でない場合があります。常にテープ・デバイスのパス名を確認してから、バックアップ・コマンドを実行してください。

例：

```
tar cvf guest@alice:/dev/tape ./bus.schedule
```

または

```
echo "./bus.schedule" | cpio -ovcO guest@alice:/dev/tape
```

cron によるバックアップの自動化

cron ユーティリティを使用すると、設定した時刻に自動的にファイルシステムのバックアップを作成できます。この場合、バックアップ・メディアは前もってドライブに装着しておきます。また、完全に自動化するには、バックアップする全データを保存できるだけの容量を持つメディアを用意します。全データが収まらない場合は、手動でメディアを変更します。

次に示す cron コマンドの例では、Backup を使用し、毎朝 3 時に /usr/src 階層ディレクトリのバックアップを /dev/tape（テープ・ドライブ）に作成します。

```
0 3 * * * /usr/sbin/Backup -t /dev/tape /usr/src
```

/var/spool/cron/crontabs/root などの crontabs ファイルに上記の行を入力します。

このコマンドは安全対策としては有効ですが、バックアップの自動化に頼りすぎないでください。バックアップ作業が正常に実施されることを最初から最後まで自分で確認し、バックアップが完了したらメディアにラベルを貼るという方法が最も理想的です。cron を使用したジョブの自動については、『IRIX Admin: System Configuration and Operation』を参照してください。

バックアップの保存

バックアップ･テープは大切に保存します｡光ディスクなどの耐久性のあるメディアにバックアップを作成している場合でも、メディアを乱暴に取扱わないように注意します。バックアップを作成した直後にテープを書込み禁止状態にします（このテープの内容に上書きする場合は、書込み禁止をはずしてください)。

バックアップ・メディアを極度の高温／低温や湿気にさらしたり、強い磁気に近付けることは避けます。サイトに多数のワークステーションがある場合は、バックアップ保存専用の部屋を設置するなどします。

1/4 インチや 8mm のカートリッジなどの磁気テープは垂直に立てて保存します。磁気テープを横向きに寝かせて保存すると、テープ素材が変形して正しく読取れなくなる場合があります。

各バックアップ・メディアには分かりやすいラベルを貼り、できれば書込み禁止にします。ラベルを色分けして、バックアップしたシステム名、バックアップのレベル（フル・バックアップまたはインクリメンタル・バックアップ)、ファイルシステム名などの特性を識別できるようにします。

サイトでの火災などの被害を最小限に抑えるために、バックアップのメイン・コピーをワークステーションの設置場所とは別の建物に保存するのも一案です。ただし、バックアップは手元に置いていつでも復旧可能な状態にしておきます。

だれでも、必要なユーティリティを搭載したシステムで、バックアップ・テープを読取ることができます。機密データを含むバックアップについては、鍵のかかった安全な部屋に保存するなどの適切な保護処置を取ります。

バックアップの保持期間

バックアップは必要な期間だけ保持できますが、実際、大部分のサイトでは 1 年以内に、バックアップ・テープを新しいバックアップに再利用しています。通常、特定の目的やプロジェクトに使用されるデータは、開始時や終了時など、プロジェクトの節目ごとにバックアップを作成します。

サイト管理者は、ユーザと十分に協議して、ファイルシステムのバックアップの保持期間を決めます。

ただし、磁気テープには物理的な耐久限度があります。磁気テープは時間の経過と共に磁気がしだいに薄れるので、2 年ほど使用しているとデータを正しく記録できなくなる場合があります。

テープの劣化によるデータ損失を避けるには、毎年または 1 年半おきに磁気テープをコピーし直します。また、sum(1) ユーティリティなどのチェックサム・プログラムをできるだけ使用し、コピー過程でデータの劣化や変質が発生していないことを確認します。数年以上に渡ってデータを確実に保存するには、光ディスクの使用をお薦めします。

テープの再利用に関するガイドライン

テープは再利用できますが、長く使用していると品質がしだいに低下します。重要なデータほど、新しいテープを使用するなどの細心の予防策を取ります。

使用できなくなったテープは不良品として廃棄します。他の人が誤って使用しないように、テープ・ケースに不良品と書いて廃棄します。明らかに不良品とわかっているテープの再利用は避けてください。保存されるデータの価値と比べれば、新しいテープの費用はささいなものです。

バックアップと回復の手順

この章では、第 1 章「バックアップ体制の導入計画」で説明した各種のバックアップ・ツールと回復ツールの使い方の例を示します。

ここでは、ユーティリティのすべてのオプションについては説明しませんが、この章で示す例と第 1 章「バックアップ体制の導入計画」の説明に従えば、各ユーザの状況に応じた最適のツールを選択できます。

ツール別のオプションの詳細については、該当ツールのマン・ページを参照してください。たとえば、`tar` コマンドについては tar(1) マン・ページを参照します。

この章では、以下について説明します。

- 「バックアップ手順」（16 ページ）
- 「システム破壊後のデータの回復」（18 ページ）
- 「デフォルトのバックアップ・デバイスの変更」（22 ページ）
- 「データ圧縮したファイルのバックアップ」（24 ページ）
- 「Backup ユーティリティと Restore ユーティリティ」（24 ページ）
- 「dump と restore について」（27 ページ）
- 「xfsdump と xfsrestore について」（31 ページ）
- 「tar について」（64 ページ）
- 「cpio について」（67 ページ）
- 「dd について」（69 ページ）

バックアップ手順

どの種類のバックアップ・ユーティリティを使用する場合でも、バックアップを作成するには、次の手順に従います。

1. テープ・ドライブが汚れていないことを確認します。ドライブをクリーニングする方法と頻度については、テープ・ドライブに添付されているハードウェア・マニュアルを参照してください。

   テープ・ヘッドが汚れていると、読取りエラーや書込みエラーが発生します。また、新しいテープは古いテープよりも酸化しやすく、ヘッドを汚すので、新しいテープを大量に使用する場合は通常よりも頻繁にドライブをクリーニングします。

2. 十分な容量のバックアップ・メディアを用意します。du(1M) と df(1) の各ユーティリティを使用すると、ディレクトリとファイルシステムのサイズをそれぞれ確認できます。

   データの重要性を考慮して、できるだけ高品質のメディアを使用します。

3. ファイルシステム全体のバックアップを作成する場合は、EFS で fsck(1M) を実行し、破損したファイルシステムのバックアップが作成されないように注意します。`fsck` で検査する前にファイルシステムをアンマウントする必要があります。この点も考慮したバックアップ計画を立てます。

   `tar` などでファイルを数個だけバックアップする場合、この手順は不要です。

4. 磁気テープ・ドライブのデフォルト・デバイスは `/dev/tape` です。このデフォルト・デバイスを使用しないときは、バックアップ・コマンドでデバイスを 1 つ指定します。

5. バックアップ・メディアにラベルを貼ります。メディアを再利用する予定がある場合は、鉛筆を使用します。このラベルには、日時、マシン名、内容を示すタイトルを書きます。後でファイルをリストアできるように、バックアップを作成したユーティリティの名前と正確なコマンド行も書きます。同じサイトでバックアップを作成する管理者が複数いる場合は、管理者名も書きます。

6. バックアップが完了したら、その内容を確認します。一部のユーティリティには、バックアップの整合性を確認するための明示的なオプション（`xfsdump -C` など）があります。通常のプログラムでは、アーカイブの内容のリストしか作成できないので、そのリストでバックアップのエラーを検出します。

7. バックアップを作成した後で、メディアを書込み禁止にします。

8. 各テープの使用回数を記録します。たとえば、テープ・ラベルに「正」の字で通算の使用回数を示します。

バックアップ・メディアを安全に保管する方法については、12 ページの「バックアップの保存」を参照してください。

システムのバックアップ・ツール

グラフィカル・ユーザ・インタフェースを持つシステムでシステムのバックアップを作成するには、システムの［ツールチェスト（Toolchest）］の［システム（System）］->［バックアップ／リストア（Backup & Restore）］を選択します。バックアップ作業は、このウィンドウのメッセージに従って行います。詳しい手順については、『Personal System Administration Guide』を参照してください。

［バックアップ／リストア（Backup & Restore）］ウィンドウを使用すると、バックアップを最も簡単に作成して利用できます。システムのフル・バックアップを作成した場合、そのバックアップは［System Maintenance］->［Recover System］オプションからも利用できます。システムのフル・バックアップの作成時に、Backup コマンドはディスクのボリューム・ヘッダ内にあるファイル名のバックアップも作成し、その情報をファイルとしてテープに保存します。システムの回復時には、このファイルから損傷したボリューム・ヘッダがリストアされます。

IRIX コマンドでシステムのバックアップを作成する場合は、Backup(1) コマンドを使用します。`Backup` コマンドは、cpio(1) コマンドのフロント・エンド・インタフェースですが、磁気テープにディスク・ボリューム・ヘッダを書込みます。［Recover System］オプションは、このヘッダ情報に基づいてブート・ブロックを再構築できます。他のバックアップ・コマンドではヘッダ情報を磁気テープに書込めません。詳細については、24 ページの「Backup ユーティリティと Restore ユーティリティ」を参照してください。

システム破壊後のデータの回復

ルート・ファイルシステムが損傷してシステムを起動できない場合は、［System Maintenance］メニューの［Recovery System］オプションを使用してシステムを回復します。［System Maintenance］メニューは、オペレーティング・システムが起動する前に処理を中断すると表示されます。システムの回復には、以下のものが必要です。

- 使用中のシステムの IRIX が格納されている CD。
- システムのフル・バックアップ・テープ（ルート（/）から始まり、システムのすべてのディレクトリとファイルを含む）。このバックアップは、前の 17 ページの「システムのバックアップ・ツール」で説明した「バックアップ／リストア・マネージャ」を使用して作成されたものです。

`Backup` コマンドまたは「バックアップ／リストア（Backup & Restore）」ウィンドウで作成したシステムのフル・バックアップが存在せず、`root` または `usr` のファイルシステムが損傷してオペレーティング・システムを起動できない場合は、システム・ソフトウェアをインストールし直します。バックアップ・テープから必要な内容を読込み、新しくインストールしたソフトウェアに上書きします。この場合は、どのバックアップ・ツールで作成したバックアップ・テープでも使用できます。

ミニルートからもファイルシステムを回復できます。たとえば、ルート・ファイルシステムが破壊された場合、ミニルートを起動し、ルート・ファイルシステムをアンマウントしてシステムから切放します。ミニルート版の `restore`、`xfs_restore`、`Restore`、`cpio`、`tar` のいずれかのコマンドを使用し、ルート・ファイルシステムをリストアします。次に、各コマンドの使い方を詳しく説明します。

破壊されたシステムを、［System Maintenance］メニューの［Recover System］オプションで回復するには、次の手順に従います。

1. マシンを最初に起動するか、リセット・ボタンを押すと、次のメッセージが表示されます。

   ```
   Starting up the system...
   ```

   ［*Stop for Maintenance*］ボタンをクリックするか、**<Esc>** キーを押すと［System Maintenance］メニューが表示されます。

2. ［System Maintenance］メニューの［Recover System］アイコンをクリックするか、次のように入力します。

   4

   次の［System Recovery］メニュー、または対応するグラフィックが表示されます。

```
                    System Recovery...

                Press <Esc> to return to the menu.

1) Remote Tape  2) Remote Directory  3) Local CD-ROM  4) Local Tape

Enter 1-4 to select source type, <Esc> to quit,
or <Enter> to start:
```

3. メニューの項目番号を入力するか、使用するドライブ・アイコンをクリックして、IRIX の CD またはソフトウェア・ディストリビューション・ディレクトリを指定します。

メモ：IRIX 6.2 では、［System Recovery］ウィンドウの［Remote Tape］と［Local Tape］のオプションを指定できません。起動可能（ミニルート）なソフトウェア提供テープがサポートされていないためです。

- システムに CD-ROM ドライブが接続されているときは、**3** と入力するか、［*Local CD-ROM*］アイコンをクリックします。［*Accept*］をクリックすると回復作業が始まります。

  「指定のドライブに CD を挿入してください」という指示が表示されます。システムに付属の IRIX CD を挿入し、［*Continue*］をクリックします。

- CD-ROM ドライブがなくても、ネットワークの他のシステムに接続されている CD-ROM ドライブを利用できます。［System Recovery］メニューで **2** と入力するか、［*Remote Directory*］のアイコンをクリックします。

  リモート・ホスト名を指定するように指示が表示されるので、システム名、コロン（**:**）、CD-ROM ドライブのフルパス名、**/dist** を入力します。たとえば、mars というシステムの CD-ROM ドライブにアクセスする場合は、次のように入力します。

  mars:/CDROM/dist

  指示が出たウィンドウで［*Accept*］をクリックした後、［System Recovery］ウィンドウで［*Accept*］をクリックします。

  グラフィックスがないシステムでは、上記の指示の後に、次のメニューが表示されます。

```
1) Remote Tape 2)[Remote Directory] 3) Local CD-ROM 4) Local Tape
      *a) Remote directory /CDROM/dist from server mars.

Enter 1-4 to select source type, a to select source, <Esc> to quit,
or <Enter> to start:
```

<Enter> キーを押します。

- リモートのソフトウェア・ディストリビューション・ディレクトリを使用する場合は、**2** と入力するか、［*Remote Directory*］のアイコンをクリックします。

  リモート・ホスト名を指定するように指示が表示されるので、システム名、コロン（**:**）、ソフトウェア・ディストリビューション・ディレクトリのフルパス名を入力します。次に例を示します。

  mars:/dist/6.2

  指示が出たウィンドウで［*Accept*］をクリックした後、［System Recovery］ウィンドウで［*Accept*］をクリックします。

  グラフィックスがないシステムでは、上記の指示の後に、次のメニューが表示されます。

```
1) Remote Tape 2)[Remote Directory] 3) Local CD-ROM 4) Local Tape
      *a) Remote directory /dist/6.2 from server mars.

Enter 1-4 to select source type, a to select source, <Esc> to quit,
or <Enter> to start:
```

  <Enter> キーを押します。

4. システムが CD から回復とインストレーションに関する情報を読始めます。必要な情報をコピーするのに約 5 分かかります。CD またはリモート・ディレクトリからシステム・ディスクへのコピーが終了すると、次のメッセージが表示されます。

```
************************************************************
*                                                          *
*                 SYSTEM    RECOVERY                       *
*                                                          *
************************************************************
You may type sh to get a shell prompt at most questions
Checking for tape devices
```

   次に、テープ・ドライブの位置を問い合わせるメッセージが表示されます。このテープ・ドライブを使用して、「バックアップ／リスト（Backup & Restore）」ツールまたは Backup(1) コマンドでシステム破壊前に作成したシステム・バックアップ・テープを読取ります。

5. ローカル磁気テープ・デバイスがあるときは、次のメッセージが表示されます。

   ```
   Restore will be from tapename. OK? ([y]es, [n]o): [y]
   ```

   `tapename` はローカル・テープ・デバイスの名前です。正しいテープ・デバイスであれば **y** と、正しくなければ **n** と入力します。

6. リモート（ネットワーク）磁気テープ・デバイスがある場合、磁気テープ・デバイスが見つからない場合、または上記の質問に「いいえ（n）」と答えた場合は、次のメッセージが表示されます。

   ```
   Remote or local restore ([r]emote, [l]ocal): [l]
   ```

   - 「リモート（r）」と応答した場合は、ネットワークを通じて回復作業を行います。リモート・システムのホスト名、リモート・システムの磁気テープ・デバイス名、リモート・システムの IP アドレス、およびローカル・システムの IP アドレスを入力するように指示されます。IP アドレスは、192.0.2.1 のように、ピリオド（.）で区切られた 2 ～ 4 個の数字で構成されたものだけを使用できます。
   - 「ローカル（l）」と応答した場合は、ローカル・システムの磁気テープ・デバイスを選択します。磁気テープ・デバイス名を入力するように指示されます。

7. 次のメッセージが表示されます。最新のフル・バックアップ・テープをセットし、**<Enter>** キーを押します。

   ```
   Insert the first BackuSp tape in the drive, then
   press (<Enter>, [q]uit (from recovery), [r]estart):
   ```

8. プログラムがテープのファイルシステムを確認し、各ファイルシステムを `/root` の下にマウントする準備ができるまで、処理が一時停止されます。その後、次のメッセージが表示されます。

   ```
   Erase all old filesystems and make new ones (y, n, sh): [n]
   ```

   以下の 3 つから選択できます。

   - 既存のファイルシステムを消去しない場合は、**n** と入力します。読込むファイルシステムを確認するメッセージが表示された後、テープのファイルが抽出されます。ディスクにある既存のファイルの方が新しくても、磁気テープの内容に書換えられます。
   - 既存のファイルシステムを消去し、新規にファイルシステムを作成する場合は、**y** と入力します。確認のメッセージとファイルシステムの種類についてのメッセージが表示された後、既存のファイルシステムが消去され、バックアップ・テープのすべてのファイルがディスクにコピーされます。

- シェルを起動する場合は、**sh** と入力します。ミニルートの状態になり、システムの破壊状況の調査、またはバックアップ・テープの作成後に作成または変更されたファイルの保存を行うことができます。シェルを終了すると、ファイルシステムを再設定したり、バックアップ・テープを読込むことができます。

9. フル・バックアップ・テープの読込みが完了すると、次のメッセージが表示され、インクリメンタル・バックアップ・テープを読込むことができます。

   ```
   Do you have incremental backup tapes to restore ([y]es, [n]o
   (none)): [n]
   ```

   別のバックアップ・テープを挿入する場合は、そのテープを磁気テープ装置にセットして **y** と入力します。ほかにバックアップ・テープがない場合は、**n** と入力します。

10. 次のメッセージでは、回復が完了した後にシステムを再起動するか、回復処理を最初からやり直すか、または最初のバックアップ・テープを読直すかを指定できます。

    ```
    Reboot, start over, or first tape again? ([r]eboot, [s]tart,
    [f]irst) [r]
    ```

    再起動する場合は、**r** と入力します。再起動しない場合は、やり直し **(s)** または最初のテープ **(f)** と入力します。

デフォルトのバックアップ・デバイスの変更

使用しているワークステーションに新しいメディアを追加したい場合があります。デフォルトのバックアップ・デバイスを新しいハードウェアに変更するには、以下の手順に従うか、グラフィック形式のシステム・マネージャを使用します。システム・マネージャを使用する方が簡単です。システム・マネージャの詳細については、『Personal System Administration Guide』を参照してください。どちらの方法を使用しても、新しいシステム・ソフトウェアをインストールするか、MAKEDEV(1M) コマンドを使用すると、デフォルトのバックアップ・デバイスがリセットされます。メディアの追加方法の詳細については、『IRIX Admin: Peripheral Devices』を参照してください。

デフォルトのテープ・デバイスを変更するには、`/dev/nrtape` を変更先のデバイスとリンクし直します。次の手順に従います。

1. 次のコマンドを入力します。

```
ls -l /dev/tape
lrwxr-xr-x    1 root    sys       10 9月30日 11時23分 tape -> rmt/tps0d5
ls -l /dev/rmt/tps0d5
crw-rw-rw-    1 root    sys   0,1416 1月29日 18時21分 /dev/rmt/tps0d5
```

   rmt は、/hw/tape へのシンボリック・リンクであるため、/dev/tape は実際には /hw/tape/tps0d5 になります。

2. 次のコマンドを入力し、すべてのテープ・デバイスのデバイス番号を調べます。

```
ls -l /hw/tape
```

   以下のような内容が表示されます。

```
crw-rw-rw-    1 root    sys    0,1416 1月29日 18時14分 tps0d5
crw-rw-rw-    1 root    sys    0,1424 1月29日 18時14分 tps0d5c
crw-rw-rw-    1 root    sys    0,1417 1月29日 18時14分 tps0d5nr
crw-rw-rw-    1 root    sys    0,1425 1月29日 18時14分 tps0d5nrc
crw-rw-rw-    1 root    sys    0,1417 1月29日 18時14分 tps0d5nrns
crw-rw-rw-    1 root    sys    0,1425 1月29日 18時14分 tps0d5nrnsc
crw-rw-rw-    1 root    sys    0,1421 1月29日 18時14分 tps0d5nrnsv
crw-rw-rw-    1 root    sys    0,1429 1月29日 18時14分 tps0d5nrnsvc
crw-rw-rw-    1 root    sys    0,1419 1月29日 18時14分 tps0d5nrs
crw-rw-rw-    1 root    sys    0,1427 1月29日 18時14分 tps0d5nrsc
crw-rw-rw-    1 root    sys    0,1423 1月29日 18時14分 tps0d5nrsv
crw-rw-rw-    1 root    sys    0,1431 1月29日 18時14分 tps0d5nrsvc
crw-rw-rw-    1 root    sys    0,1421 1月29日 18時14分 tps0d5nrv
crw-rw-rw-    1 root    sys    0,1429 1月29日 18時14分 tps0d5nrvc
crw-rw-rw-    1 root    sys    0,1416 1月29日 18時14分 tps0d5ns
crw-rw-rw-    1 root    sys    0,1424 1月29日 18時14分 tps0d5nsc
crw-rw-rw-    1 root    sys    0,1420 1月29日 18時14分 tps0d5nsv
crw-rw-rw-    1 root    sys    0,1428 1月29日 18時14分 tps0d5nsvc
crw-rw-rw-    1 root    sys    0,1418 1月29日 18時14分 tps0d5s
crw-rw-rw-    1 root    sys    0,1426 1月29日 18時14分 tps0d5sc
crw-rw-rw-    1 root    sys    0,1399 1月29日 18時14分 tps0d5stat
crw-rw-rw-    1 root    sys    0,1422 1月29日 18時14分 tps0d5sv
crw-rw-rw-    1 root    sys    0,1430 1月29日 18時14分 tps0d5svc
crw-rw-rw-    1 root    sys    0,1420 1月29日 18時14分 tps0d5v
crw-rw-rw-    1 root    sys    0,1428 1月29日 18時14分 tps0d5vc
```

   このリストの一番上に表示されているデバイスが、現在の /dev/tape です。ただし tps0d5ns も同じデバイスです。接尾辞の「c」は compression（圧縮）を、「nr」は no-rewind（巻戻しなし）を、non-swapping（スワッピングなし）を、「s」は byte-swapping（バイト・スワッピング）を、「v」は variable block size（可変ブロック・サイズ）を表します。

3. /dev/tape リンクを削除し、同じ名前で新しいリンクを作成します。たとえば、可変ブロック・サイズをデフォルトに指定するには、次のコマンドを実行します。

```
rm /dev/tape
ln -s rmt/tps0d5v /dev/tape
rm /dev/nrtape
ln -s rmt/tps0d5nrv /dev/nrtape
```

大部分のプログラムは、/dev/tape または /dev/nrtape をデフォルトのテープ・デバイスとして使用しています。プログラムの動作が異常な場合は、正しいテープ・デバイスが使用されていることを確認します。

データ圧縮したファイルのバックアップ

IRIX テープ・インタフェースでは、DLT などのハードウェア圧縮を行うテープ・ドライブに対して、オプション文字の「c」を使用して圧縮デバイスを指定できます。たとえば、ハードウェア圧縮を使用してファイルを保存するには、/dev/rmt を設定するときに、/dev/tape と /dev/nrtape をリンクするデバイスの最後に「c」を付けます。

```
ln -s rmt/tps0d5vc /dev/tape
ln -s rmt/tps0d5nrvc /dev/nrtape
```

ソフトウェアを使用してデータを圧縮してからテープに保存することもできます。その方法には 2 種類があります。詳細については、compress(1) と pack(1) を参照してください。

Backup ユーティリティと Restore ユーティリティ

Backup ユーティリティと Restore ユーティリティは、cpio のフロント・エンド・インタフェースです。この 2 つのユーティリティを使用すると、リモート・ホスト名とテープ・デバイスを指定できます。Backup は、ボリューム・ヘッダ・ファイルのリストを作成します。Restore は、このリストを使用してファイルとディレクトリをリストアします。詳細については、Backup(1) マン・ページと Restore(1) マン・ページを参照してください。

［System Maintenance］メニューの［Recovery］オプションを使用する場合は、Backup コマンド、またはグラフィック表示されたシステム・マネージャのバックアップ機能を使用します。シ

ステム・マネージャの詳細については、『Personal System Administration Guide』を参照してください。

Backup によるデータの保存

バックアップを開始する前に、`df` コマンドを使用してアーカイブ全体に必要な容量を見積もります。このコマンドでは、`root` パーティションをバックアップするために必要な KB 数などを確認できます。

```
df -k /
```

`Backup` によって、ファイル、ディレクトリ、ファイルシステム全体、ローカルまたはリモート・デバイスのシステム全体のバックアップを作成できます。システム全体のバックアップを作成すると、破壊されたボリューム・ヘッダを回復したり、最新のバックアップ以降に変更されたファイルだけのバックアップも作成できます。`Backup` コマンドの構文は、次のとおりです。

`Backup` *[***`-h`** ホスト名 *][***`-t`** デバイス名 *][***`-i`***]* ディレクトリ名 | ファイル名

デフォルト・テープ・デバイスにディスク全体のバックアップを作成するには、次のコマンドを入力します。

```
Backup /
```

この `Backup` コマンドによってシステム全体のバックアップが作成されます。その作成日は `/etc/lastbackup` ファイルに保存されます。

メモ：［System Maintenance］メニューで `Backup` コマンドを使用してシステムをリストアするには、システムのフル・バックアップを作成します。システムのフル・バックアップの作成時に、`Backup` コマンドはディスクのボリューム･ヘッダ内にあるファイル名のバックアップも作成し、その情報をファイルとしてテープに保存します。システムの回復時には、このファイルから損傷したボリューム・ヘッダがリストアされます。

最後に作成したシステムのフル・バックアップと同様のバックアップを作成するには、次のように入力します。

```
Backup -i /
```

特定のディレクトリとサブディレクトリのバックアップを作成する場合は、そのトップレベルのディレクトリ名を入力します。たとえば、usr 階層のバックアップを作成するには、次のように入力します。

```
Backup /usr
```

リモート・テープ・ドライブを使用する場合は、**-h** *hostname* オプションを指定します。

```
Backup -h guest@alice.cbs.tv.com:/dev/tape /usr/people/ralph
```

このように指定すると、/usr/people/ralph ディレクトリのバックアップが、alice.cbs.tv.com というホストの /dev/tape テープ・デバイスに作成されます。リモート・テープ・ドライブを使用するには、リモート・システムでの少なくとも *guest* としてのログイン特権が必要です。

ファイルのバックアップを作成する場合は、そのファイル名を入力します。次に例を示します。

```
Backup people.tar.Z
```

上記の例のように、相対パス名を指定してバックアップを作成すると、カレント・ディレクトリからの相対パスでファイルやディレクトリが保存されます。相対パス名とは、スラッシュ（/）で始まらないパス名のことです。スラッシュで始まるパス名は絶対パス名です。たとえば、/usr/bin/vi は絶対パスです。先頭のスラッシュが、システムのルート・ディレクトリから始まるパス名であることを表しています。一方、work/special.project/chapter1 は相対パスです。パスの先頭にスラッシュがないので、パスはカレント・ディレクトリから始まることを表しています。

Restore によるデータのリストア

Restore コマンドは、tar を使用してバックアップからファイルを展開するシェル・スクリプトです。64 ページの「tar について」を参照してください。Restore では、グラフィック形式のシステム・マネージャで作成されたテープも読取れます。『Personal System Administration Guide』を参照してください。

Restore では、マルチボリュームのバックアップをリストアできます。次のコマンドを入力します。

```
Restore
```

テープをドライブに挿入するよう指示するメッセージが表示されます。

1 つのファイルをリストアするには、次のコマンドを入力します。

Restore *file1*

-h オプションを使用すると、別のホスト・ワークステーションのテープ・ドライブを指定できます。リモート・ドライブからデータを読出すには、guest としてのログイン特権が必要です。

Restore -h guest@alice.cbs.tv.com *file1*

相対パス名を使用してバックアップを作成した場合、ファイルはカレント・ディレクトリにリストアされます。相対パス名とは、スラッシュ（/）で始まらないパス名のことです。スラッシュで始まるパス名は絶対パス名です。たとえば、`/usr/bin/vi` は絶対パスです。先頭のスラッシュが、システムのルート・ディレクトリから始まるパス名であることを表しています。一方、`work/special.project/chapter1` は相対パスです。パスの先頭にスラッシュがないので、パスはカレント・ディレクトリから始まることを表しています。

ディスクに同じパス名のファイルが存在する場合は、それがテープのファイルより新しいものでも、リストア処理中に上書きされます。絶対パス名でファイルをリストアするときは、特に注意が必要です。作業中のカレント・ディレクトリとは関係なく、パス名で指定された箇所にファイルがリストアされるからです。

たとえば、カレント・ディレクトリを `/tmp` とした場合、`/etc/passwd` として作成されたファイルのバックアップをリストアすると、`/etc/passwd` というファイルが上書きされます。`passwd` として作成されたファイルのバックアップをリストアすると、`passwd` というファイルが `/tmp` にリストアされます。

dump と restore について

`dump` プログラムと `restore` プログラムは、多くの UNIX システムで利用されている標準的なファイルシステムのバックアップ・ユーティリティです。この 2 つのコマンドは EFS だけに使用します。XFS のダンプとリストアについては、31 ページの「xfsdump と xfsrestore について」を参照してください。`dump` プログラムは、ファイルシステム全体のインクリメンタル・バックアップを作成します。

`dump` のアーカイブからファイルを抽出するには `restore` を使用します。`restore` を使用すると、ファイルシステム全体または特定のファイルをリストアできます。対話モードがあるので、アーカイブの内容を参照し、特定のファイルを選択してリストアすることもできます。

dump によるファイルシステムのバックアップ

dump ユーティリティを使用すると、通常のファイルだけでなくデバイス・ファイルと特殊ファイル（リンクや名前付きパイプなど）のバックアップを作成できます。アーカイブからファイルを回復するには、restore コマンドを使用します。更新を記録するように **u** オプションを指定した場合は、dump を実行した最新の日付が /etc/dumpdates ファイルに記録されます。

たとえば、次のコマンドは /usr ファイルシステム内のすべてのファイルのバックアップを作成します。

```
dump 0 /dev/usr
```

メモ：上記の 0 はインクリメンタル・レベルです。レベル番号については次で説明します。

インクリメンタル・バックアップの実行

dump ユーティリティはインクリメンタル・バックアップ用に設計されており、通常のファイルとディレクトリだけでなく、特殊ファイル、リンク、パイプなどのバックアップも作成できます。

インクリメンタル・バックアップを作成するには、dump を使用するときにインクリメンタル番号を指定します。dump プログラムは、指定されたインクリメンタル番号のバックアップ以降に変更されたすべてのファイル（リンクや名前付きパイプなどの特殊ファイルを含む）を作成します。アーカイブからファイルを回復する場合は、restore コマンドを使用します。

dump プログラムは、インクリメンタル・バックアップの作成を目的として設計されています。インクリメントの度合はレベルと呼ばれ、各レベルには番号が割当てられます。

- レベル 0 のバックアップには、ファイルシステム内のすべてのファイルが保存されます。
- レベル 1 ～ 9 のバックアップには、前回の同レベル以下のバックアップ後に変更されたすべてのファイルが保存されます。

たとえば、次のコマンドは /usr ファイルシステム内のすべてのファイルのバックアップを作成します。

```
dump 0 /dev/usr
```

次のコマンドは、前回のレベル 0 のバックアップ以降に変更されたファイルのバックアップを作成します。

```
dump 1 /dev/usr
```

次のコマンドは、前回のレベル 1 のバックアップ以降に変更されたファイルのバックアップを作成します。

```
dump 2 /dev/usr
```

dump コマンドにレベル 1 を次に指定すると、dump はレベル 0 以降に変更されたファイルのユーティリティ・バックアップは作成しますが、レベル 2 以降に変更されたファイルのバックアップは作成しません。レベル番号を使用すると、必要に応じた柔軟なバックアップ計画を立案できます。

restore によるファイルシステムの回復

dump プログラムで作成したファイルとファイルシステムを回復するには、restore を使用します。restore の使い方には対話モードと非対話モードがあります。

比較的少数のファイルを dump アーカイブから回復する場合は、対話モードを使用します。restore の対話モードでは、テープの内容を参照し、特定のファイルを探して抽出できます。

バックアップ全体を回復する場合は、非対話モードを使用します。この場合は、バックアップ・メディアをドライブに挿入し、次のように入力します。

```
restore -x
```

破壊されたルート・ファイルシステムを完全に復元する必要がある場合は、18 ページの「システム破壊後のデータの回復」を参照してください。この節では、システムが起動しない場合に、破壊されたルート・ファイルシステムを復元する方法を説明しています。また、ルート・ファイルシステムのシステム・ファイルを安全に復元する方法も説明しています。

注意：アクティブなルート・ファイル・システム・ディスクにある IRIX システム・ファイルを復元すると、ファイルシステムが破壊されたり、システム障害が発生する可能性があります。

---

restore による個々のファイルの回復

dumpユーティリティを使用して作成されたアーカイブから個々のファイルを回復するには、次の手順に従います。

1. テープが書込み禁止になっていることを確認します。テープをテープ・ドライブに挿入し、次のように入力します。

```
# restore vi
Verify tape and initialize maps
Tape block size is 32
Dump  date: Wed Feb 13 10:18:59 1991
Dumped from: the epoch
Level 0 dump of an unlisted filesystem on ralph:/dev/rusr
Label: none
Extract directories from tape
Initialize symbol table.
restore >>
```

2. *restore*>> プロンプトで cd と ls を使用し、テープの内容を参照します。

```
restore > ls
2      *./          973     source        1502  net/
2      *../         149     d2/           1445  os/
10     .cshrc       155016  debug/        1437  proto3.5/
1463   .gamma       69899   dev/          1494  revE
1464   .gamtables   696     etc/          2122  stand/
160    .kshrc       137     bin/          3     tmp/
1540   .lastlogin   1311412 jake/         128   unix
819    .login       424     lib/          128   unix.debug
820    .profile     9       lost+found/   4     usr/
```

参照を続ける場合は、*restore*>> プロンプトの後に次のコマンドを入力します。

```
restore >> cd etc
restore >> pwd
/etc
```

3. 復元するファイルのリストを作成します。リストアの対象のリストにファイル名を追加するには次のコマンドを入力します。

```
restore >> add fstab
restore >> add fsck
```

この時点で ls と入力すると、表示されるファイルのリストで fsck と fstab にアスタリスク（*）が付き、リストアの対象であることが示されます。リストの対象のリストからファイルを削除するには、delete コマンドを使用します。

```
restore > delete fstab
```

4. 特定のファイルを復元するには、extract コマンドを使用します。

```
restore > extract
Extract requested files
You have not read any tapes yet.
Unless you know which volume your file(s) are on you should
start with the last volume and work towards the first.
Specify next volume #: 1
Mount tape volume 1
then enter tape name (default: /dev/tape) <Return>
extract file ./etc/fsck
Add links
Set directory mode, owner, and times.
set owner/mode for '.'? [yn] n
restore > q
```

小数のファイルだけを回復するには、restore の非対話オプションを使用します。たとえば、次のように入力します。

```
restore -x ./usr/people/ralph/bus.schedule ./etc/passwd
```

このコマンドによって、アーカイブから bus.schedule と passwd のファイルが復元されます。

xfsdump と xfsrestore について

ここでは、xfsdump と xfsrestore の各ユーティリティの機能と、各ユーティリティを使用して XFS のデータのバックアップおよび回復を行う方法について説明します。xfsdump(1M) と xfsrestore(1M) のマン・ページには、この 2 つのユーティリティに関するオンライン情報が掲載

されています。表 2-1 は、xfsdump と xfsrestore、および EFS 用の dump(1M) の使分けを示しています。

表 2-1 ファイルシステムとダンプ・ユーティリティ

ファイルシステム	使用するダンプ・ユーティリティ
EFS	dump
XFS	xfsdump

表 2-2 に、目的に応じた xfsrestore とこの EFS の対コマンドである restore(1M) の使い分けについてまとめます。

表 2-2 ファイルシステムとリストア・ユーティリティ

使用したダンプ・ユーティリティ	使用するリストア・ユーティリティ	ファイルシステム
dump	restore	EFS または XFS
xfsdump	xfsrestore	EFS または XFS

EFS と XFS のどちらのデータでも復元できますが、バックアップの作成時に使用したダンプ・ユーティリティに対応するリストア・ユーティリティを使用します。

xfsdump と xfsrestore の機能

xfsdumpとxfsrestoreの2つのユーティリティはXFSを完全にサポートしています。xfsdumpとxfsrestoreを使用すると、ローカルまたはリモートのドライブを通じてデータのバックアップやリストアを行うことができます。ファイルシステム、ディレクトリ、個々のファイルのバックアップを作成し、そのバックアップ方法とは無関係にそれぞれを復元できます。xfsdump を使うと、マウントされている使用中のファイルシステムのバックアップも作成できます。

xfsdump と xfsrestore を使用すると、意図的または偶発的に中断された回復処理を再開できます。つまり、ダンプ処理またはリストア処理をいつでも中断し、必要に応じて再開できます。xfsrestore は、xfsdump で作成されたデータを EFS に復元できます（xfsdump はマウントされた XFS のバックアップだけを作成します）。xfsdump と xfsrestore は、インクリメンタ

ル・ダンプをサポートし、1 つのメディア・オブジェクトに複数のダンプを挿入できます。両ユーティリティは、1 つのダンプを自動的に複数のドライブに分割し、複数のドライブからのダンプを 1 つにまとめて復元できます。これにより、ダンプ処理とリストア処理の効率が向上します。

`xfsdump` と `xfsrestore` は、64 ビット i ノード番号、ファイルの長さ、ホール、ユーザが選択できるエクステント・サイズなどの XFS 形式をサポートしています。また、複数のメディア・タイプ、IRIX がサポートするすべてのファイル・タイプ（通常のファイル、ディレクトリ、シンボリック・リンク、キーボードからの入力データなどのブロック型／文字型の特殊ファイル、FIFO、およびソケット）をサポートし、ファイル間の物理的リンクを保存します。`xfsdump` は、ダンプ中のファイルシステムの状態に影響を与えません。たとえば、アクセス時刻は変更されません。`xfsrestore` は、メディアのエラーを検出すると、それを迂回して速やかに処理を続行します。`xfsdump` は、ローカルでもリモートでも、同時に複数のマウント・ポイントをダンプできません。

`xfsdump` は、1 つのメディアの最後に達すると、別のメディアをセットすることを指示できます。したがって、メディア容量の見積もりは必要ありません。`xfsdump` は自動バックアップもサポートしています。`xfsdump` は、実行されたすべてのダンプ・リストをオンラインで保存します。このリストの内容はさまざまなフィルタを使って見られるので、特定のダンプ情報を簡単に見つけ出せます。`xfsrestore` は対話型の操作もサポートしているので、個々のファイルまたはディレクトリを選択して回復できます。複数のダンプがある場合は、作成時期の異なるバックアップからも選択できます。ダンプの内容は非対話モードでも参照できます。

メモ：XFS ファイルシステムでディスクの割当てを行う場合の詳細については、『IRIX Admin: Disks and Filesystems』の第 7 章「XFS ファイルシステム上でのディスク割当ての管理」を参照してください。

---

xfsdump のメディア形式

ここでは、いくつかの用語を紹介した後、xfsdump でデータをフォーマットしてメディアに出力する方法について説明します。そのデータは xfsrestore によって読取られます。

通常、xfsdump と xfsrestore は磁気テープに適用しますが、ほかの種類のメディアもサポートしています。したがって、次の説明では、メディア・オブジェクトという用語で各種のメディアを総称します。ダンプとは、xfsdump コマンドを 1 回使用し、データ・ファイルを指定のメディア・オブジェクトに出力した結果を表します。xfsdump を実行することをダンプ・セッションといいます。

ダンプ・セッションは、1 つのダンプ・ストリームをメディア・オブジェクトに出力します。ダンプ・ストリームは、1 つのファイルだけのことも、ファイルシステム全体のこともあります。ダンプ・ストリームは、以下のダンプ・オブジェクトで構成されます。

- 1 つまたは複数のデータ・セグメント
- オプションとしてのダンプ・リスト
- ストリーム終了子

データ・セグメントには、実際のデータが格納されます。ダンプ・リストには、ダンプ内のダンプ・オブジェクトのリストが格納されます。ストリーム終了子は、ダンプ・ストリームの終わりを示します。ダンプ・ストリームが複数のダンプ・オブジェクトで構成されている場合、各オブジェクトはメディア・ファイルに格納されます。標準出力など、出力デバイスの中にはメディア・ファイルの概念をサポートしないものもあります。この種のデバイスにとっては、ダンプ・ストリームは単なるデータを表します。

各種の xfsdump 形式

1 つの磁気テープに少量のデータを出力するような最も単純なダンプは、ダンプ・オブジェクトとしてデータ・セグメントとストリーム終了子だけを作成します。オプションであるダンプ・リストを追加すると、図 2-1 のようなダンプになります。ここで紹介するデータ形式の図には、オプションのダンプ・リストを常に含めます。

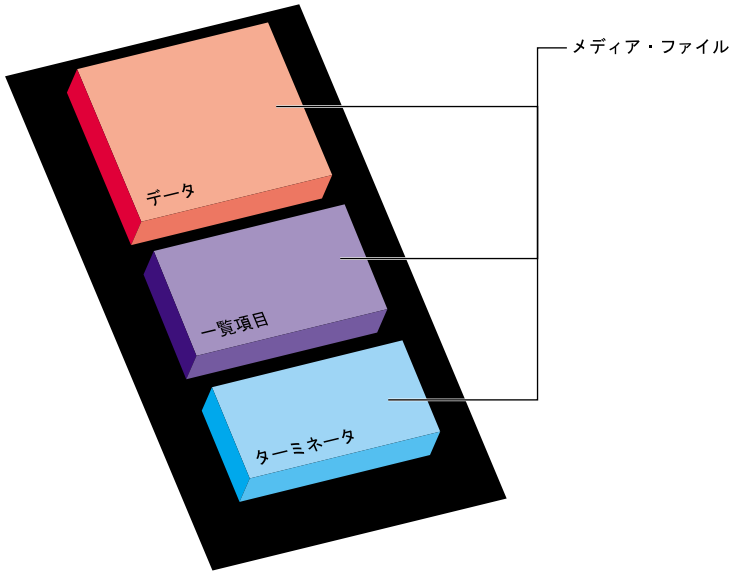


図 2-1 単一のメディア・オブジェクトに出力された単一のダンプ

単一のメディア・オブジェクトを超えるデータ・ストリームをダンプすることもできます。この場合、データ・ストリームを、データ・セグメントの境界をまたぎ、2 つのメディア・ファイルに分割できます。リストは複数セグメントに分割されません。さらに、複数のドライブを指定した場合、ダンプは自動的に複数のストリームに分割されます。`xfsdump` ユーティリティは、1 つのメディア・オブジェクトの終わりに達すると、別のメディア・オブジェクトをセットするように指示します。

図 2-2 は、2 つのデバイスの各メディア・オブジェクトに分割されたダンプ・セッションのデータ形式を示しています。

図 2-2　複数のメディア・オブジェクトに分割して出力された単一のダンプ

xfsdump ユーティリティは、複数のダンプを単一のメディア・オブジェクトに出力することもできます。たとえば、磁気テープにダンプする場合は、ストリーム終了子[1] を取除き、既存のダンプ・セッションの終わりの位置に、新しいダンプ・データとストリーム終了子を書込みます。

図 2-3 は、2 つのダンプを単一のメディア・オブジェクトに出力したときのメディア・ファイルの形式です。

図 2-4 は、複数のダンプが複数のメディア・オブジェクトを使用する例です。メディア・オブジェクトにすでにダンプされたものがある場合、xfsdump ユーティリティは、ストリーム終了子を取除き、新たにダンプ・データの書込みを始めます。

[1] ストリーム終了子を取除く方法が使用できないドライブの場合には、別の方法を使用して同じようにアーカイブすることができます。

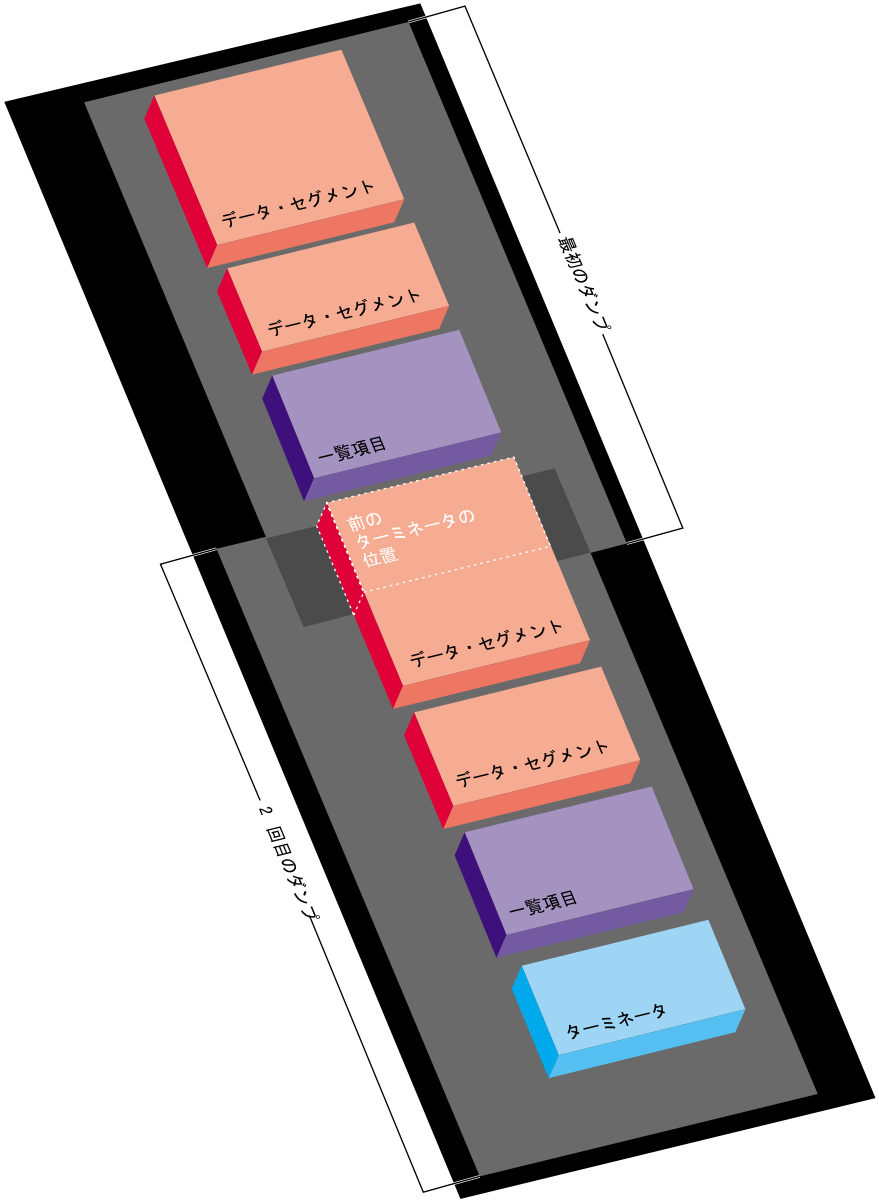


図 2-3　単一のメディア・オブジェクトに出力された複数のダンプ

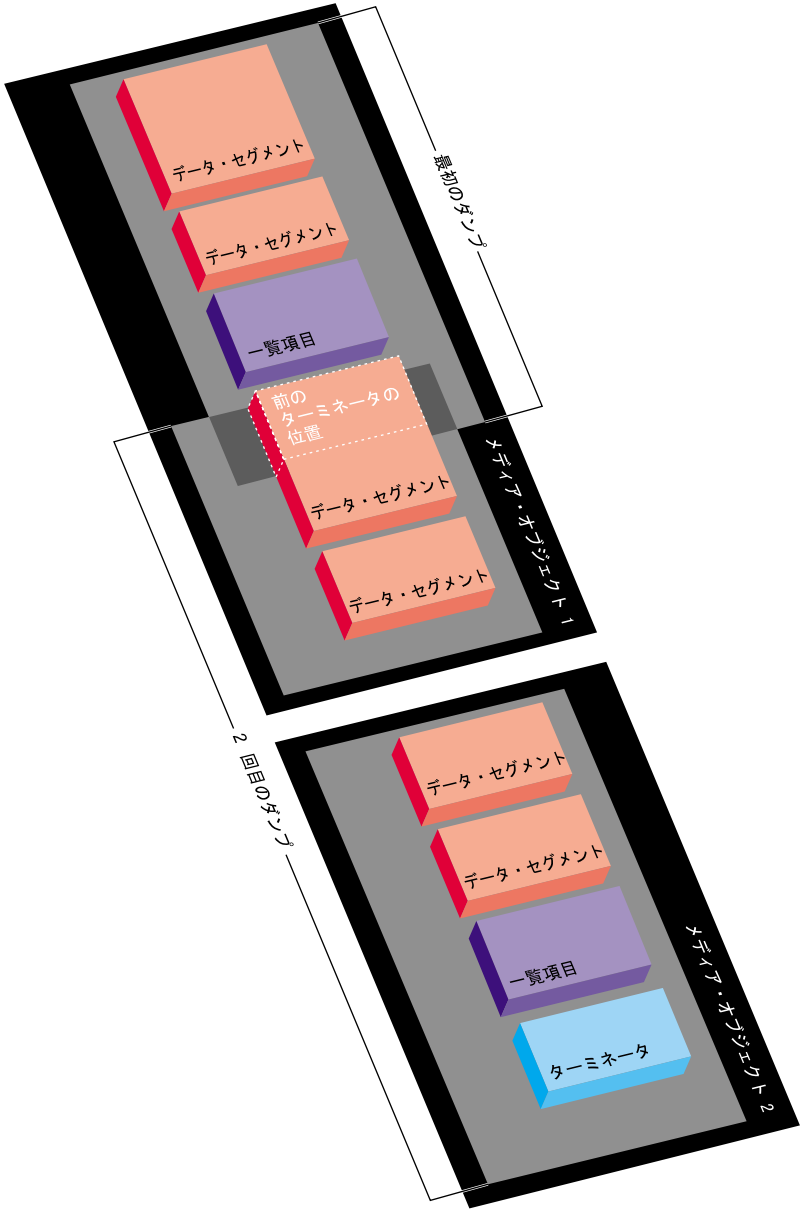


図 2-4　　複数のメディア・オブジェクトに出力された複数のダンプ

xfsdump によるデータの保存

ここでは、xfsdump コマンドを使ってローカルまたはリモート・デバイスにデータのバックアップを作成する方法を説明します。xfsdump の構文の要約は **-h** オプションを指定すると表示されます。

```
# xfsdump -h
xfsdump: version X.X
xfsdump: usage: xfsdump [ -b <blocksize> (with minimal rmt option) ]
                        [ -c <media change alert program> ]
                        [ -f <destination> ... ]
                        [ -h (help) ]
                        [ -l <level> ]
                        [ -m <force usage of minimal rmt> ]
                        [ -o <overwrite tape > ]
                        [ -p <seconds between progress reports> ]
                        [ -s <subtree> ... ]
                        [ -v <verbosity {silent, verbose, trace}> ]
                        [ -A (don't dump extended file attributes) ]
                        [ -B <base dump session id> ]
                        [ -E (pre-erase media) ]
                        [ -F (don't prompt) ]
                        [ -I (display dump inventory) ]
                        [ -J (inhibit inventory update) ]
                        [ -L <session label> ]
                        [ -M <media label> ... ]
                        [ -O <options file> ]
                        [ -R (resume) ]
                        [ -T (don't timeout dialogs) ]
                        [ -Y <I/O buffer ring length> ]
                        [ - (stdout) ]
                        [ <source (mntpnt|device)> ]
```

xfsdump を使用するには特権ユーザであることが条件です。詳細については、xfsdump(1M) マン・ページを参照してください。

xfsdumpによるローカル・メディアの指定

xfsdump を使用すると、さまざまなメディアにデータのバックアップを作成できます。たとえば、磁気テープやハードディスクにデータをダンプできます。メディア・オブジェクトの場所は、ローカル・システムに接続されたドライブでも、ネットワークを通じてアクセスできるドライブでもかまいません。

次に、レベル 0 のダンプをローカルの磁気テープ・デバイスに出力する例を示します。レベルが 0 のときはダンプ・レベルを指定する必要はありません。ダンプ・レベルの詳細については、28 ページの「インクリメンタル・バックアップの実行」を参照してください。

```
#xfsdump -f /dev/tape -L testers_11_21_94 -M test_1 /disk2
xfsdump: version 2.0 - type ^C for status and control
xfsdump: level 0 dump of cumulus:/disk2
xfsdump: dump date: Wed Oct 25 16:19:13 1995
xfsdump: session id: d2a6123b-b21d-1001-8938-08006906dc5c
xfsdump: session label: “testers_11_21_94”
xfsdump: ino map phase 1: skipping (no subtrees specified)
xfsdump: ino map phase 2: constructing initial dump list
xfsdump: ino map phase 3: skipping (no pruning necessary)
xfsdump: ino map phase 4: skipping (size estimated in phase 2)
xfsdump: ino map phase 5: skipping (only one dump stream)
xfsdump: ino map construction complete
xfsdump: preparing drive
xfsdump: creating dump session media file 0 (media 0, file 0)
xfsdump: dumping ino map
xfsdump: dumping directories
xfsdump: dumping non-directory files
xfsdump: ending media file
xfsdump: media file size 16777216 bytes
xfsdump: dumping session inventory
xfsdump: beginning inventory media file
xfsdump: media file 1 (media 0, file 1)
xfsdump: ending inventory media file
xfsdump: inventory media file size 4194304 bytes
xfsdump: writing stream terminator
xfsdump: beginning media stream terminator
xfsdump: media file 2 (media 0, file 2)
xfsdump: ending media stream terminator
xfsdump: media stream terminator size 2097152 bytes
xfsdump: I/O metrics: 3 by 2MB ring; 14/22 (64%) records streamed; 145889B/s
xfsdump: dump complete: 141 seconds elapsed
```

この例では、セッション・ラベル（**-L** オプション）とメディア・ラベル（**-M** オプション）が提供され、ファイルシステム全体がダンプされています。メッセージ・レベルが提供されていないので、デフォルトで `verbose` が指定され、メッセージが詳細に画面に表示されます。**-J** オプションが指定されていないので、このバックアップの記録によって、ダンプ・リストが更新されます。

次に、ファイルシステムのサブディレクトリのバックアップを作成する例を示します。この例では、メッセージ・レベルが `silent` に設定され、ダンプ情報は更新されません（**-J** オプション）。

```
# xfsdump -f /dev/tape -v silent -J -s people/fred /usr
```

バックアップを作成したサブディレクトリ(`/usr/people/fred`)はファイルシステムに相対的に指定されているので、ファイルシステム名（この例では `/usr`）は指定されていません。`/usr` が大きなファイルシステムである場合は、**-v silent** オプションを指定すると、長い処理時間中、画面には何も表示されません。

xfsdump によるリモート磁気テープ・ドライブの指定

リモート磁気テープ・ドライブにデータのバックアップを作成するには、標準リモート・システムの指定方法に従います。システムを指定し、コロン（:）で区切って特殊ファイルのパス名を記述します。システムは、名前サーバが機能しているときはホスト名で、機能していないときは IP アドレスで指定します。

メモ：リモート・バックアップでは、できるだけ可変長ブロック・サイズの磁気テープ・デバイスを使用します。それがサポートされていないときは、固定長ブロック・サイズのデバイスを使用します。intro(7) マン・ページを参照してください。

次に、リモート磁気テープ・デバイスにサブツリーのバックアップを作成する例を示します。

```
# xfsdump -f magnolia:/dev/tape -L mag_10-95 -s engr /disk2
xfsdump: version 2.0 - type ^C for status and control
xfsdump: level 0 dump of cumulus:/disk2
xfsdump: dump date: Wed Oct 25 16:27:39 1995
xfsdump: session id: d2a6124b-b21d-1001-8938-08006906dc5c
xfsdump: session label: “mag_10-95”
xfsdump: ino map phase 1: parsing subtree selections
xfsdump: ino map phase 2: constructing initial dump list
xfsdump: ino map phase 3: pruning unneeded subtrees
xfsdump: ino map phase 4: estimating dump size
xfsdump: ino map phase 5: skipping (only one dump stream)
xfsdump: ino map construction complete
xfsdump: preparing drive
xfsdump: positioned at media file 0: dump 0, stream 0
xfsdump: positioned at media file 1: dump 0, stream 0
xfsdump: positioned at media file 2: dump 0, stream 0
xfsdump: stream terminator found
xfsdump: creating dump session media file 0 (media 0, file 2)
xfsdump: dumping ino map
xfsdump: dumping directories
xfsdump: dumping non-directory files
xfsdump: ending media file
xfsdump: media file size 6291456 bytes
xfsdump: dumping session inventory
xfsdump: beginning inventory media file
xfsdump: media file 1 (media 0, file 3)
xfsdump: ending inventory media file
xfsdump: inventory media file size 4194304 bytes
xfsdump: writing stream terminator
xfsdump: beginning media stream terminator
xfsdump: media file 2 (media 0, file 4)
xfsdump: ending media stream terminator
xfsdump: media stream terminator size 2097152 bytes
xfsdump: I/O metrics: 3 by 2MB ring; 12/22 (55%) records streamed; 99864B/s
xfsdump: dump complete: 149 seconds elapsed
```

この例では、リモート・システム *magnolia* の可変長ブロック・サイズの磁気テープ・デバイスに `/disk2/engr` のバックアップが作成されます。新しいデータは、maglolia にマウントされた磁気テープにすでに書込まれているダンプの後に記録されます。

メモ：ローカル・システムの特権ユーザ・アカウントは、リモート・システムに対してパスワードを使わずに rsh を起動できることが必要です。詳細については、hosts.equiv(4) マン・ページを参照してください。

xfsdump によるファイルへのバックアップ

デバイスではなくファイルにバックアップを作成できます。次に、ファイル（Makefile）とディレクトリ（Source）のバックアップを、ローカル・システムの /usr/tmp 内のダンプ・ファイル（monday_backup）に作成する例を示します。

```
# xfsdump -f /usr/tmp/monday_backup -v silent -J -s \
people/fred/Makefile -s people/fred/Source /usr
```

リモート・システムのファイルにダンプすることもできます。ただし、そのファイルはリモート・システムの /dev というディレクトリにあることが条件です。次に、ローカル・システムのサブディレクトリ /usr/people/fred のバックアップを、リモート・システム *theduke* の通常のファイル /dev/fred_mon_12-2 に作成する例を示します。

```
# xfsdump -f theduke:/dev/fred_mon_12-2 -s people/fred /usr
```

リモート・システムのファイルが NFS マウント・ファイルシステムにある場合は、どのファイルへもダンプできます。いずれの場合も、リモート・システムへのファイルの書込みが許可されている必要があります。

xfsdump と xfsrestore の標準入出力機能を使用して、ファイルシステム間またはネットワークを介してデータをパイプ処理する方法については、63 ページの「xfsdump と xfsrestore によるファイルシステムのコピー」を参照してください。

xfsdump による磁気テープの再利用

新しい磁気テープをダンプ・セッションのメディア・オブジェクトとして使用する場合、xfsdump はメッセージを表示せずに磁気テープの先頭からダンプ・データを書始めます。磁気テープにダンプ・データがすでにある場合、xfsdump は最後のダンプ・ストリームの終わりからデータを書始めます。このときもメッセージは表示されません。

ただし、磁気テープにダンプ・セッション以外のデータがある場合、xfsdump は処理を続行する前に次のようなメッセージを表示します。

```
# xfsdump -f /dev/tape /test
xfsdump: version X.X - type ^C for status and control
xfsdump: dump date: Fri Dec 2 11:25:19 1994
xfsdump: level 0 dump
xfsdump: session id: d23cc072-b21d-1001-8f97-080069068eeb
xfsdump: preparing tape drive
xfsdump: this tape contains data that is not part of an XFS dump
xfsdump: do you want to overwrite this tape?
type y to overwrite, n to change tapes or abort (y/n):
```

ダンプ・セッションを続行する場合は、**y** と入力します。中止する場合は、**n** と入力します。**y** と入力すると、ダンプ・セッションが再開し、磁気テープは上書きされます。このメッセージに応答しないと、時間切れになります。つまり、`crontab` エントリによって起動された自動バックアップは失敗します。ただし、`xfsdump` コマンドに **-F** オプションを指定しておくと、了解を確認するメッセージは表示されず、磁気テープは強制的に上書きされます。

使用済み磁気テープの消去

磁気テープの既存のデータを消去する場合は、`mt erase` コマンドを使用します。磁気テープが書込み禁止になっていないことを確認します。

たとえば、デフォルトのローカル磁気テープ・ドライブで使用済み磁気テープのデータを消去するには、次のように入力します。

```
# mt -f /dev/tape erase
```

注意：ダンプ・セッションも含め、磁気テープのすべてのデータが消去されます。

この磁気テープは、了解確認のメッセージなしで、`xfsdump` の出力メディアとして使用できます。

インクリメンタル・ダンプとダンプの再開

インクリメンタル・ダンプとは、データのバックアップを部分的に作成しながら、ファイルやディレクトリなどのすべてのバックアップを最新の状態に維持する方法です。インクリメンタル・ダンプはレベル 0 から 9 までの番号で管理されます。レベル 0 のダンプは常にファイルシステム全体のバックアップを作成します。ダンプ・レベルが 0 以外の場合は、それより低い番号のダンプ以降に変更されたファイルだけのバックアップを作成します。

たとえば、レベル 2 でファイルシステムのバックアップを作成したと仮定します。後日、レベル 3 でダンプすると、レベル 2 でバックアップを作成した後に変更されたファイルだけがダンプされます。この場合、レベル 2 のバックアップを、レベル 3 のバックアップのベース・ダンプといいます。ベース・ダンプとは、最も低いレベル番号に対応する最新のバックアップのことです。

ダンプを再開するときも同様の処理が行われます。一時中断されたダンプ・セッションが再開されると、ダンプする予定だった残りのファイルとともに、中断中に変更されたファイルのバックアップも作成されます。中断されたダンプは、インクリメンタル・ダンプとしてリストアする必要があります。58 ページの「xfsrestore による漸増リストアの実行」を参照してください。

xfsdump によるインクリメンタル・ダンプの実行

次の例では、最初に、何も書込まれていない新しい磁気テープにレベル 0 でダンプを書込みます。

```
# xfsdump -f /dev/tape -l 0 -M Jun_94 -L week_1 -v silent /usr
```

1 週間後、同じ磁気テープにレベル 1 でファイルシステムのダンプを実行します。

```
# xfsdump -f /dev/tape -l 1 -L week_2 /usr
```

磁気テープの既存のダンプ・データは読飛ばされ、その後ろから新しいデータがレベル 1 のダンプとして書込まれます。継続して行うダンプではメディア・ラベルを指定する必要はありません。

1 週間後、レベル 2 のダンプを実行します。4 週目にレベル 3 のダンプを実行します。レベルは 9 までサポートされています。

```
# xfsdump -f /dev/tape -l 2 -L week_3 /usr
```

インクリメンタル・ダンプをリストアする手順については、58 ページの「xfsrestore による漸増リストアの実行」を参照してください。

xfsdump によるダンプの再開

ダンプ・セッションは中断し、後から再開できます。ダンプ・セッションを中断するには、**<CTRL>-C** キーを押します。メニューが表示され、セッションの中断、メッセージ・レベルの変更、またはセッションの再開のいずれかを選択できます。

次に、xfsdump がファイルシステムを約 37% ダンプした後で中断する例を示します。

```
# xfsdump -f /dev/tape -M march95 -L week_1 -v silent /disk2
======================== status and control dialog ========================
status at 16:49:16: 378/910 files dumped, 37.8% complete, 32 seconds elapsed
please select one of the following operations
1: interrupt this session
2: change verbosity
3: display metrics
4: other controls
5: continue (default) (timeout in 60 sec)
 -> 1
please confirm
1: interrupt this session
2: continue (default) (timeout in 60 sec)
 -> 1
interrupt request accepted
 -------------------------------- end dialog --------------------------------
xfsdump: initiating session interrupt
xfsdump: dump interrupted prior to ino 1053172 offset 0
```

このダンプ処理を後で再開する場合は、**-R** オプションを指定し、異なるセッション・ラベルを入力します。

```
# xfsdum -f /dev/tape -R -L week_1.contd -v silent /disk2p
```

ダンプ処理が再開されると、処理の中断前にバックアップが作成されなかったファイルと、中断中に変更されたファイルのバックアップが作成されます。

メモ：-R オプションを指定した場合は、ダンプ・リストを作成しながらダンプ処理が行われます。その場合、xfsdump の **-J** オプションは無効になります。

---

xfsdump アーカイブの検査

ここでは、xfsdump を使用して xfsdump のリストを参照する方法について説明します。

xfsdump のリストは、xfsdump によって作成される /var/xfsdump というディレクトリに保存されます。このダンプ・リストは、xfsdump -I と指定すれば、いつでも表示できます。xfsdump -I だけを指定した場合は、ダンプ・リスト全体が表示されます。xfsdump -I の指定では、ルート特権は必要ありません。

次に、ダンプ・リストの一部の出力例を示します。

```
# xfsdump -I | more
file system 0:
        fs id:          d23cb450-b21d-1001-8f97-080069068eeb
        session 0:
                mount point:    magnolia.abc.xyz.com:/test
                device:         magnolia.abc.xyz.com:/dev/rdsk/dks0d3s2
                time:           Mon Nov 28 11:44:04 1994
                session label:  ""
                session id:     d23cc0c3-b21d-1001-8f97-080069068eeb
                level:          0
                resumed:        NO
                subtree:        NO
                streams:        1
                stream 0:
                        pathname:       /dev/tape
                        start:          ino 4121 offset 0
                        end:            ino 0 offset 0
                        interrupted:    YES
                        media files:    2
                        media file 0:
                                mfile index:    0
---more---
```

ダンプ情報を階層的に表示するため、ダンプ・リストの記録は順番にインデントされています。

ダンプ情報の深度（1、2、または 3）を指定すると、ダンプ・リストの一部だけを表示できます。たとえば、depth=2 と指定すると多くの情報が省略されます。次の出力例を前のものと比べてください。

```
# xfsdump -I depth=2
file system 0:
        fs id:          d23cb450-b21d-1001-8f97-080069068eeb
        session 0:
                mount point:    magnolia.abc.xyz.com:/test
                     device:    magnolia.abc.xyz.com:/dev/rdsk/dks0d3s2
                time:           Mon Nov 28 11:44:04 1994
                session label:  ""
                session id:     d23cc0c3-b21d-1001-8f97-080069068eeb
                level:          0
                resumed:        NO
                subtree:        NO
                streams:        1
        session 1:
                mount point:    magnolia.abc.xyz.com:/test
                device:         magnolia.abc.xyz.com:/dev/rdsk/dks0d3s2
                .
                .
                .
```

ファイルシステムのマウント・ポイントを `mnt` オプションで指定すると、そのファイルシステムだけのリストを表示できます。次は、`depth` を `1` に設定し、ファイルシステムを 1 つだけ指定して表示した例です。

```
# xfsdump -I depth=1,mnt=magnolia.abc.xyz.com:/test
filesystem 0:
        fs id:          d23cb450-b21d-1001-8f97-080069068eeb
```

次の例のように、xfsrestore に **-t** オプションを指定すると、ダンプ・メディア自体の内容をリストできます。xfsrestore ユーティリティについては、次で詳しく説明します。たとえば、ローカルの磁気テープ・ドライブに現在セットされているダンプ・テープの内容をリスト表示するには、次のように入力します。

```
# xfsrestore -f /dev/tape -t -v silent | more
xfsrestore: dump session found
xfsrestore: session label: "week_1"
xfsrestore: session id: d23cbcb4-b21d-1001-8f97-080069068eeb
xfsrestore: no media label
xfsrestore: media id: d23cbcb5-b21d-1001-8f97-080069068eeb
do you want to select this dump? (y/n):y
selected
one
A/five
people/fred/TOC
people/fred/ch3.doc
people/fred/ch3TOC.doc
people/fred/questions
A/four
people/fred/script_0
people/fred/script_1
people/fred/script_2
people/fred/script_3
people/fred/sub1/TOC
people/fred/sub1/ch3.doc
people/fred/sub1/ch3TOC.doc
people/fred/sub1/questions
people/fred/sub1/script_0
people/fred/sub1/script_1
people/fred/sub1/script_2
people/fred/sub1/script_3
people/fred/sub1/xdump1.doc
people/fred/sub1/xdump1.doc.backup
people/fred/sub1/xfsdump.doc
people/fred/sub1/xfsdump.doc.auto
people/fred/sub1/sub2/TOC
---more---
```

xfsrestore について

ここでは、xfsrestore コマンドについて説明します。このコマンドは、xfsdump で作成したダンプ・データを参照したり展開するのに使用します。**-h** オプションを指定すると、xfsrestore の記述方法の要約を表示できます。

```
# xfsrestore -h
xfsrestore: version X.X
xfsrestore: usage: xfsrestore [ -a <alt. workspace dir> ... ]
                    [ -e (don't overwrite existing files) ]
                    [ -f <source> ... ]
                    [ -h (help) ]
                    [ -i (interactive) ]
                    [ -n <file> (restore only if newer than) ]
                    [ -o (restore owner/group even if not root) ]
                    [ -p <seconds between progress reports> ]
                    [ -r (cumulative restore) ]
                    [ -s <subtree> ... ]
                    [ -t (contents only) ]
                    [ -v <verbosity {silent, verbose, trace}> ]
                    [ -A (don't restore extended file attributes) ]
                    [ -C (check tape record checksums) ]
                    [ -D (restore DMAPI event settings) ]
                    [ -E (don't overwrite if changed) ]
                    [ -F (don't prompt) ]
                    [ -I (display dump inventory) ]
                    [ -J (inhibit inventory update) ]
                    [ -L <session label> ]
                    [ -N (timestamp messages) ]
                    [ -O <options file> ]
                    [ -P (pin down I/O buffers) ]
                    [ -Q (force interrupted session completion) ]
                    [ -R (resume) ]
                    [ -S <session id> ]
                    [ -T (don't timeout dialogs) ]
                    [ -U (unload media when change needed) ]
                    [ -V (show subsystem in messages) ]
                    [ -W (show verbosity in messages) ]
                    [ -X <excluded subtree> ... ]
                    [ -Y <I/O buffer ring length> ]
                    [ -Z (miniroot restrictions) ]
                    [ - (stdin) ]
                    [ <destination> ]
```

xfsrestore を使用すると、xfsdump でバックアップしたデータを復元できます。バックアップ方法に関係なしに、ファイル、サブディレクトリ、ファイルシステムを復元できます。たとえば、ファイルシステム全体のバックアップを 1 回のダンプで作成した場合でも、そのファイルシステムから個々のファイルやサブディレクトリを選択して復元できます。

xfsrestore は、対話モードでも非対話モードでも使用できます。対話モードでは、バックアップされたファイルシステムやファイル群を参照して、復元するファイルを選択できます。非対話モードでは、コマンド 1 行で、選択したファイル、サブディレクトリ、またはファイルシステム全体を復元できます。データのリストア先として、元のファイルシステムでも、EFS または XFS 内の任意の位置でも指定できます。

xfsrestore を連続して使用し、ベース・ダンプを基準にしたインクリメンタル・ダンプを復元できます。データは、ダンプされた順番で復元されます。

xfsrestore による単純なリストアの実行

単純な復元とは非漸増リストアのことです。インクリメンタル・ダンプの復元については、58 ページの「xfsrestore による漸増リストアの実行」を参照してください。次に、`xfsrestore` を非対話モードで使用する簡単な例を示します。

```
# xfsrestore -f /dev/tape /disk2
xfsrestore: version 2.0 - type ^C for status and control
xfsrestore: searching media for dump
xfsrestore: preparing drive
xfsrestore: examining media file 0

========================== dump selection dialog ===========================

the following dump has been found on drive 0

hostname: cumulus
mount point: /disk2
volume: /dev/rdsk/dks0d2s0
session time: Wed Oct 25 16:59:00 1995
level: 0
session label: “tape1”
media label: “media1”
file system id: d2a602fc-b21d-1001-8938-08006906dc5c
session id: d2a61284-b21d-1001-8938-08006906dc5c
media id: d2a61285-b21d-1001-8938-08006906dc5c

restore this dump?
1: skip
2: restore (default)
 -> 2
this dump selected for restoral

------------------------------- end dialog -------------------------------

xfsrestore: using online session inventory
xfsrestore: searching media for directory dump
xfsrestore: reading directories
xfsrestore: directory post-processing
xfsrestore: restoring non-directory files
xfsrestore: I/O metrics: 3 by 2MB ring; 9/13 (69%) records streamed; 204600B/s
xfsrestore: restore complete: 104 seconds elapsed
```

この例で、xfsrestore は、磁気テープの最初のダンプに達すると、それが復元するダンプかどうかを問い合わせてきます。**1**（スキップ）と入力すると、xfsrestore は磁気テープの次のダンプに進みます。次のダンプを見つけると、それが復元するダンプかどうかを問い合わせてきます。

xfsdump にセッション・ラベルを指定している場合は、特定のダンプを要求できます。次に例を示します。

```
# xfsrestore -f /dev/tape -L Wed_11_23 /usr
xfsrestore: versionX.X - type ^C for status and control
xfsrestore: preparing tape drive
xfsrestore: dump session found
xfsrestore: advancing tape to next media file
xfsrestore: dump session found
xfsrestore: restore of level 0 dump of magnolia.abc.xyz.com:/usr created Wed
Nov 23 11:17:54 1994
xfsrestore: beginning media file
xfsrestore: reading ino map
xfsrestore: initializing the map tree
xfsrestore: reading the directory hierarchy
xfsrestore: restoring non-directory files
xfsrestore: ending media file
xfsrestore: restoring directory attributes
xfsrestore: restore complete: 200 seconds elapsed
```

この方法では、コマンドを 1 つ入力してダンプを復元できます。見つかったダンプ・セッションが目的のものかどうかを確認するメッセージに対して **y**、または **n** と応答する必要はありません。さらに限定する場合は、**-S** オプションで特定のダンプ・セッションの固有なセッション ID を指定します。

```
# xfsrestore -f /dev/tape -S \
d23cbf47-b21d-1001-8f97-080069068eeb /usr2/tmp
xfsrestore: versionX.X - type ^C for status and control
xfsrestore: preparing tape drive
xfsrestore: dump session found
xfsrestore: advancing tape to next media file
xfsrestore: advancing tape to next media file
xfsrestore: dump session found
xfsrestore: restore of level 0 dump of magnolia.abc.xyz.com:/test resumed Mon
Nov 28 11:50:41 1994
xfsrestore: beginning media file
xfsrestore: media file 0 (media 0, file 2)
xfsrestore: reading ino map
xfsrestore: initializing the map tree
xfsrestore: reading the directory hierarchy
xfsrestore: restoring non-directory files
xfsrestore: ending media file
xfsrestore: restoring directory attributes
xfsrestore: restore complete: 229 seconds elapsed
```

セッション ID はダンプ・リストから探せます。48 ページの「xfsdump アーカイブの検査」を参照してください。セッション・ラベルは重複することがありますが、セッション ID は重複しません。

xfsrestore による個々のファイルのリストア

`xfsrestore` コマンド行で、復元する個々のファイルまたはサブディレクトリを指定できます。次に、ファイル `people/fred/notes` を復元し、ディレクトリ `/usr/tmp` に挿入する例を示します。ファイルのフルパス名は `/usr/tmp/people/fred/notes` となります。

```
# xfsrestore -f /dev/tape -L week_1 -s people/fred/notes /usr/tmp
```

バックアップが作成されたところにファイルを直接復元することもできます。ただし、**-e**、**-E**、または **-n** のオプションを指定しない場合は、同じ名前を持つ既存のファイルが上書きされます。

次の例では、サブディレクトリ people/fred を /usr に復元しています。/usr/people/fred 内のすべてのファイルとサブディレクトリは、ダンプ・テープのデータによって上書きされます。

```
# xfsrestore -f /dev/tape -L week_1 -s people/fred /usr
```

xfsrestore によるネットワークを通じたリストアの実行

標準ネットワーク・リファレンスを使用して、ネットワークのデバイスやファイルを指定できます。たとえば、*magnolia* という名前のホストにある磁気テープ・ドライブから復元する場合は、次のコマンドを使用できます。

```
# xfsrestore -f magnolia:/dev/tape -L 120694u2 /usr2
xfsrestore: versionX.X - type ^C for status and control
xfsrestore: preparing tape drive
xfsrestore: dump session found
xfsrestore: advancing tape to next media file
xfsrestore: dump session found
xfsrestore: restore of level 0 dump of magnolia.abc.xyz.com:/usr2 created Tue
Dec 6 10:55:17 1994
xfsrestore: beginning media file
xfsrestore: media file 0 (media 0, file 1)
xfsrestore: reading ino map
xfsrestore: initializing the map tree
xfsrestore: reading the directory hierarchy
xfsrestore: restoring non-directory files
xfsrestore: ending media file
xfsrestore: restoring directory attributes
xfsrestore: restore complete: 203 seconds elapsed
```

この例では、ダンプ・データは *magnolia* の磁気テープから読込まれ、リストア先はローカル・システムのディレクトリ /usr2 となります。xfsrestore の標準入力オプションの使用例については、63 ページの「xfsdump と xfsrestore によるファイルシステムのコピー」を参照してください。

xfsrestore の対話モードによるリストアの実行

xfsrestore に **-i** オプションを指定すると、対話モードでファイルを復元できます。対話モードでは、ls、pwd、cd の各オプションを使用して、ファイルシステムを参照できます。また、add と delete の各コマンドを使用して、復元するファイルやサブディレクトリのリストを作成できます。次に、extract コマンドを入力してファイルを復元します。quit コマンドを入力す

ると、ファイルはリストアされずに対話モードのリストア・セッションが終了します。これらのコマンドでは、ワイルドカードを使用できません。

メモ: 標準入力 (STDIN) からリストアする場合、`xfsrestore` の対話モードは使用できません。

次に、対話モードによる単純なリストアの画面出力の例を示します。

```
# xfsrestore -f /dev/tape -i -v silent .
xfsrestore: dump session found
xfsrestore: no session label
xfsrestore: session id:    d23cbeda-b21d-1001-8f97-080069068eeb
xfsrestore: no media label
xfsrestore: media id:      d23cbedb-b21d-1001-8f97-080069068eeb
do you want to select this dump? (y/n): y
selected

 --- interactive subtree selection dialog ---

the following commands are available:
        pwd
        ls [ { <name>, ".." } ]
        cd [ { <name>, ".." } ]
        add [ <name> ]
        delete [ <name> ]
        extract
        quit
        help
 -> ls
            4122 people/
            4130 two
            4126 A/
            4121 one
 -> add two
 -> cd people
 -> ls
            4124 fred/
 -> add fred
 -> ls
    *       4124 fred/
 -> extract

 ---------------- end dialog ----------------
```

上記の対話モードのリストア・セッションでは、サブディレクトリ people/fred とファイル two が現在の作業ディレクトリ（.）からの相対ディレクトリにリストアされます。ls の出力中のアスタリスク（*）は、選択されたものを表します。

xfsrestore による漸増リストアの実行

漸増リストアとは、インクリメンタル・ダンプを順次リストアしながらファイルシステムを再構築する方法です。中断したダンプを復元するときにも使用されます。ファイルシステムの漸増リストアを実行するには、レベル番号の最も低いベース・ダンプがあるメディア・オブジェクトから始めて、より高いダンプ・レベル番号のインクリメンタル・ダンプを順に回復します。漸増リストアを実行する場合は、xfsrestore に **-r** オプションを指定します。

次の例では、/dev/tape にあるダンプを、ベースダンプのレベル 0 から始めてより高いレベルの順に復元します。

```
# /usr/tmp/xfsrestore -f /dev/tape -r -v silent .

 ========================= dump selection dialog ===========================

the following dump has been found on drive 0

hostname: cumulus
mount point: /disk2
volume: /dev/rdsk/dks0d2s0
session time: Wed Oct 25 14:37:47 1995
level: 0
session label: “week_1”
media label: “Jun_94”
file system id: d2a602fc-b21d-1001-8938-08006906dc5c
session id: d2a60b26-b21d-1001-8938-08006906dc5c
media id: d2a60b27-b21d-1001-8938-08006906dc5c

restore this dump?
1: skip
2: restore (default)
 -> Enter
this dump selected for restoral

 ------------------------------- end dialog --------------------------------

#
```

次に、同じコマンドをもう一度入力します。次のダンプに処理が進むので、再びデフォルトを選択します。

```
# xfsrestore -f /dev/tape -r -v silent .

 ========================= dump selection dialog ==========================

the following dump has been found on drive 0

hostname: cumulus
mount point: /disk2
volume: /dev/rdsk/dks0d2s0
session time: Wed Oct 25 14:40:54 1995
level: 1
session label: “week_2”
media label: “Jun_94”
file system id: d2a602fc-b21d-1001-8938-08006906dc5c
session id: d2a60b2b-b21d-1001-8938-08006906dc5c
media id: d2a60b27-b21d-1001-8938-08006906dc5c

restore this dump?
1: skip
2: restore (default)
 -> Enter
this dump selected for restoral

 -------------------------------- end dialog --------------------------------
#
```

インクリメンタル・ダンプのすべての結果が回復されるまで、この処理を繰返します。最終的に、ファイルシステム全体が最新の状態で復元されます。この例では、`xfsrestore` コマンドを発行したディレクトリ (.) にリストアされます。

中断されたダンプもインクリメンタル・ダンプと同じ手順で復元します。インクリメンタル・リストアを実行するには、`xfsrestore` に **-r** オプションを指定します。必要に応じて **y** または **n** と入力し、復元するインクリメントを選択します。58 ページの「xfsrestore による漸増リストアの実行」を参照してください。

中断されたダンプを、中断されていない非インクリメンタル・ダンプとして復元すると、中断前にダンプされた部分だけが復元され、残りの部分はリストアされません。中断されたダンプかどうかは、オンライン・リストから判断できます。

次に、中断されたダンプ・セッションを示すダンプ・リストの例を示します。太字は重要な部分です。

```
# xfsdump -I depth=3,mobjlabel=AugTape,mnt=indy4.xyz.com:/usr
file system 0:
        fs id:          d23cb450-b21d-1001-8f97-080069068eeb
        session 0:
                mount point:    indy4.xyz.com.com:/usr
                device:         indy4.xyz.com.com:/dev/rdsk/dks0d3s2
                time:           Tue Dec  6 15:01:26 1994
                session label:  "180894usr"
                session id:     d23cc0c3-b21d-1001-8f97-080069068eeb
                level:          0
                resumed:        NO
                subtree:        NO
                streams:        1
                stream 0:
                        pathname:       /dev/tape
                        start:          ino 4121 offset 0
                        end:            ino 0 offset 0
                        interrupted:    YES
                        media files:    2
        session 1:
                mount point:    indy4.xyz.com.com:/usr
                device:         indy4.xyz.com.com:/dev/rdsk/dks0d3s2
                time:           Tue Dec  6 15:48:37 1994
                session label:  "Resumed180894usr"
                session id:     d23cc0cc-b21d-1001-8f97-080069068eeb
                level:          0
                resumed:        YES
                subtree:        NO
                streams:        1
                stream 0:
                        pathname:       /dev/tape
                        start:          ino 4121 offset 0
                        end:            ino 0 offset 0
                        interrupted:    NO
                        media files:    2
.
.
.
```

このダンプ・リストから、セッション 0 が中断されて、その後に再開され、セッション 1 で完了していることが分かります。

上記の中断されたダンプ・セッションをリストアするには、次のコマンドを入力します。

```
# xfsrestore -f /dev/tape -r -L 180894usr .
# xfsrestore -f /dev/tape -r -L Resumed180894usr .
```

これで、/usr のバックアップ全体がカレント・ディレクトリからの相対ディレクトリに復元されます。作業終了後は、リストア先のディレクトリから housekeeping ディレクトリを削除します。

xfsrestore の中断

xfsdump を中断するのと同様に xfsrestore セッションを中断できます。リストア・セッションを中断し、後で再開できます。通常、リストア・セッションを中断するには、中断文字として **<CTRL>-C** キーを押します。セッションを中断または継続するためのオプションがリスト表示されます。

```
# xfsrestore -f /dev/tape -v silent /disk2

 ========================= dump selection dialog ===========================

the following dump has been found on drive 0

hostname: cumulus
mount point: /disk2
volume: /dev/rdsk/dks0d2s0
session time: Wed Oct 25 17:20:16 1995
level: 0
session label: “week1”
media label: “newtape”
file system id: d2a602fc-b21d-1001-8938-08006906dc5c
session id: d2a6129e-b21d-1001-8938-08006906dc5c
media id: d2a6129f-b21d-1001-8938-08006906dc5c

restore this dump?
1: skip
2: restore (default)
 -> 2
this dump selected for restoral

 ------------------------------- end dialog --------------------------------
```

```
 ======================== status and control dialog ========================

status at 17:23:52: 131/910 files restored, 14.4% complete, 42 seconds elapsed

please select one of the following operations
1: interrupt this session
2: change verbosity
3: display metrics
4: other controls
5: continue (default) (timeout in 60 sec)
 -> 1

please confirm
1: interrupt this session
2: continue (default) (timeout in 60 sec)
 -> 1
interrupt request accepted

 -------------------------------- end dialog --------------------------------

xfsrestore: initiating session interrupt
```

xfsrestore セッションを再開するには、**-R** オプションを使用します。

```
# xfsrestore -f /dev/tape -R -v silent /disk2
```

データのリストア処理が中断した時点から続行します。

housekeeping ディレクトリと orphanage ディレクトリについて

xfsrestore ユーティリティは、リストア先に housekeeping と orphanage の 2 つのサブディレクトリを作成できます。

housekeeping ディレクトリは、漸増リストアを行うときに利用される一時作業用ディレクトリです。このディレクトリを通じて、1 つの xfsrestore から次の処理に情報が引渡されます。漸増リストの処理中は削除してはいけませんが、完了後は削除します。

orphanage ディレクトリは、ダンプのファイルシステム構造内で参照されないファイルやサブディレクトリが復元されたときに作成されます。たとえば、変更の激しいファイルシステムをダンプすると、新しいファイルがディレクトリの一部でないことがあり、ダンプされたディレクトリのどれもがそのファイルを参照しない場合があります。この場合は、警告メッセージが表示され、そのファイルは元の i ノード番号と世代番号から構成される名前（123479.14 など）が付けられて orphanage ディレクトリに格納されます。

xfsdump と xfsrestore によるファイルシステムのコピー

xfsdump と xfsrestore を 1 行のコマンドで使用して、ファイルシステム間で、またはネットワークを通じてデータをパイプできます。xfsdump の標準出力を xfsrestore の標準入力になるようにパイプすると、ファイルシステムの正確なコピーを作成できます。

たとえば、/usr/people/fred を /usr2 ディレクトリにコピーするには、次のコマンドを入力します。

```
# xfsdump -J -s people/fred - /usr | xfsrestore - /usr2
```

/usr/people/fred をネットワーク・ホスト *magnolia* の /usr/tmp ディレクトリにコピーするには、次のコマンドを入力します。

```
# xfsdump -J -s people/fred - /usr | rsh magnolia \
xfsrestore - /usr/tmp
```

magnolia に /usr/tmp/people/fred が作成されます。

メモ：ローカル・システムの特権ユーザ・アカウントは、パスワードを使わずにリモート・システムに対して rsh を起動できることが必要です。詳細については、hosts.equiv(4) マン・ページを参照してください。

---

tar について

tar ユーティリティでは、ファイルやディレクトリのバックアップを作成できます。tar を使用すると、磁気テープへのファイルのコピー、tar ファイルの作成、磁気テープのファイルとディスクのファイルの比較、標準入力の読込み、tar からほかのプロセスへの出力のパイプを行うことができます。このコマンドは、世界中の UNIX システムで広く使用されています。詳細については、tar(1) マン・ページを参照してください。

メモ：tar コマンドで 2GB 以上のファイルを処理する場合は **-K** オプションを指定します。**-K** オプションを指定しないと、tar は 2GB より大きなファイルに対して警告を表示するだけで、処理をスキップします。このオプションを指定すると、XFS 以外のシステムでは使用できない tar アーカイブが作成される場合があります。**-K** オプションは、**-O** オプションと組合わせて指定することはできません。後者は、tar アーカイブを 1 世代前の pre-POSIX 形式で作成します。

---

tar によるファイルのバックアップ

tar を使用して個々のファイルのバックアップを作成するには、次のコマンドを使用します。

tar c *file*

tar による変更日付別のファイルのバックアップ

tar コマンドには、変更日付別にファイルを保存する機能が組込まれていません。しかし、find コマンドを使用すると、ファイルが変更されなかった日数別に、ファイルをアーカイブできます。

find /usr -mtime 5 -local -type f -o -type *othertypes* **-print | tar cv -**

この find コマンドは、5 日間変更されなかった通常のローカル（非 NFS）ファイルを探し、その結果を出力として tar コマンドに送ります。

tar によるインクリメンタル・バックアップの実行

tar コマンドにはインクリメンタル・バックアップの機能は組込まれていませんが、同じ機能を他のシステム・コマンドで代行できます。

次の例では、前の節で説明したバックアップ計画に従って /usr ファイルシステムのインクリメンタル・バックアップを作成します。ただし、find コマンドを使用して、バックアップを作成するファイルを特定します。

1. バックアップを作成するファイルシステムに作業ディレクトリを移動します。たとえば、次のコマンドを入力します。

   ```
   cd /usr
   ```

2. ファイルシステム全体のバックアップを作成します。

   ```
   tar cv .
   ```

3. 前日のバックアップ以降に変更されたファイルのバックアップを毎日作成します。

   ```
   find /usr -mtime 1 -local -print | tar cvf -
   ```

4. 前週のバックアップ以降に変更されたファイルのバックアップを毎週作成します。

   ```
   find /usr -mtime 7 -local -type f -print | tar cvf -
   ```

5. 4 週間後、フル・バックアップを再度作成し、上記の手順を繰返します。

tar パフォーマンスの向上

通常、ブロックのサイズを大きくすると、テープの書込み、読取り効率が向上します。IRIX の tar コマンドは自動的にテープ・デバイスに照会し、最適なブロック・サイズを決定します。ブロック・サイズが大きすぎると移植性が低下する場合があります。詳細については、tar(1) の **b** オプションを参照してください。

tar アーカイブの検査

tar アーカイブを検査するには、キーワード **v** を指定し、アーカイブの内容を詳細に表示します。

```
tar tv
```

tar を使用して、アーカイブされたファイルと元のファイルを比較できます。

```
tar C
```

ファイルの状態に関するメッセージが表示されます。各メッセージは、アーカイブされたファイルと元のファイルの状態を表すキー・キャラクタ（文字または記号）で始まります。表 2-3 を参照してください。

表 2-3 tar のファイル比較キー・キャラクタ

キー	意味
=	ファイルの内容が同じ
!	ファイルの内容が異なる
?	ディスクのファイルを読取れない
>	ディスクのファイルが存在しない
L	テープの旧ファイルとリンク
S	シンボリック・リンク
B	ブロック型特殊ファイル
C	キャラクタ型特殊ファイル
P	名前付きパイプ

tar アーカイブのリストア

tar アーカイブから個々のファイルをリストアするには、コマンド行で各ファイル名を指定します。

```
tar xv file1 file2 directory/file3
```

cpio について

tar と同様に、cpio ではファイルやディレクトリをアーカイブできます。cpio を使用すると、磁気テープやディスクへのファイルのコピー、空のディレクトリのアーカイブ、バイト・スワップ、移植可能な ASCII アーカイブの作成、標準出力に対する読み書きを行うことができます。cpio ユーティリティを使用し、cp(1) コマンドでコピーできないファイルやディレクトリをコピーすることもできます。たとえば、cp は異なるファイルシステムにディレクトリをコピーできません。

システム・マネージャでは、cpio コマンドを使ってバックアップを作成することもできます。サーバを使用していて、グラフィカルなシステム・マネージャにアクセスできない場合は、代りに cpio を使います。cpio で作成したバックアップは、システム・メンテナンス・メニューおよびコマンド・モニタで読取ることができます。

メモ：XFS と cpio について：2GB より大きなファイルを処理する場合は cpio(1) コマンドに **-K** オプションを指定します。**-K** オプションを指定しないと、cpio は 2GB より大きなファイルに対して警告を表示するだけで、処理をスキップします。このオプションを指定すると、XFS 以外のシステムで使用できない cpio アーカイブが作成されることがあります。**-K** オプションを指定するときは **-o**（出力）オプションも必ず指定します。**-K** オプションは、**-c** オプション（ASCII ヘッダを付けて cpio アーカイブを作成する）または **-H** オプション（各種のヘッダ形式を指定する）と組合わせて指定することはできません。

---

cpio によるファイルのバックアップ

cpio を使用してファイルのバックアップを作成するには、次のコマンドを使用します。

cat *filelist* **| cpio -o > /dev/tape**

ヒント：他のシステム、特に IRIX 以外のシステムに移植できるようにするには、**-H odc** オプションを指定して小さいデバイス番号でテキストのヘッダ情報を作成し、必要に応じて **-B** オプションを指定してブロック・サイズを 5120 バイト／レコードに設定します。**-B** オプションを指定しないと、cpio はデバイスに照会して最適なブロック・サイズを決定します。

---

cpio による変更日付別のファイルのバックアップ

cpio コマンドには、変更日付別にファイルを保存する機能が組込まれていません。しかし、find コマンドを使用すると、ファイルが変更されなかった日数別に、ファイルをアーカイブできます。

```
find /usr -depth -mtime 5 -print | cpio -ocvO /dev/tape
```

find に引数 *-depth* を指定すると、ディレクトリ内のファイル名が出力された後でそのディレクトリ名が出力されます。これにより、cpio は読取り専用のディレクトリ内のファイルでも処理できます。**-O** オプションは出力ファイルを指定します。

cpio によるインクリメンタル・バックアップの実行

tar と cpio にはインクリメンタル・バックアップの機能は組込まれていませんが、同じ機能を他のシステム・コマンドで代行できます。

次の例では、前の節で説明したバックアップ計画に従って /usr ファイルシステムのインクリメンタル・バックアップを作成します。ただし、find コマンドを使用して、バックアップを作成するファイルを特定します。

1. バックアップを作成するファイルシステムに作業ディレクトリを移動します。このファイルシステム全体のバックアップを作成します。たとえば、次のコマンドを実行します。

   ```
   cd /usr
   find . -depth -print | cpio -ocLO /dev/tape
   ```

2. 前日のバックアップ以降に変更されたファイルのバックアップを毎日作成します。

   ```
   cd /usr
   find . -depth -mtime 1 -print | cpio -ocLO /dev/tape
   ```

3. 前週のバックアップ以降に変更されたファイルのバックアップを毎週作成します。

   ```
   cd /usr
   find . -depth -mtime 7 -print | cpio -ocLO /dev/tape
   ```

4. 4週間ごとに、ファイルシステム全体のバックアップを再度作成し、上記の手順を繰返します。

cpio アーカイブの検査

cpio アーカイブを検査するには、次のコマンドを使用し、アーカイブの内容を詳細に表示します。

```
cpio -itvI /dev/tape
```

-t オプションを指定すると、目次を出力できます。また、**-v** オプションは冗長モードを意味し、**-I** オプションは入力ファイルの指定です。

cpio プログラムには、ファイルを比較する機能は組込まれていません。cpio アーカイブのファイルを比較するには、そのアーカイブをディスクに出力します。次に、diff(1)、cmp(1)、または dircmp(1) などのプログラムを使用し、ファイルを比較します。または、ディスクに出力したファイルと元のファイルの合計検査を sum(1) を使用して比較します。

cpio アーカイブのリストア

cpio アーカイブから個々のファイルをリストアするには、コマンド行で各ファイル名を指定します。

cpio -id *file1 directory/file2* **< /dev/tape**

-i オプションを指定すると、cpio は磁気テープ・ドライブから入力データを読込みます。**-d** オプションを指定すると、出力先のディレクトリがない場合は、それを作成します。

dd について

dd プログラムは、指定の入力ファイル（デフォルトは *stdin*）を読み、指定された変換を行い、その結果を指定の出力ファイル（デフォルトは *stdout*）に出力します。このプログラムは、特にバックアップ・ツールではありませんが、以下のような便利な機能が備わっています。

- アーカイブ内の指定のブロックをスキップする
- 出力ブロックをスキップする
- 入出力ブロック・サイズを指定する
- 指定の数だけブロックをコピーする
- バイト・スワップなど各種のデータ変換を実行する

dd コマンドの使い方の詳細については、dd(1M) マン・ページを参照してください。

バックアップと回復のトラブルシューティング

バックアップの際に障害が発生した場合は、その障害の原因を特定します。障害の原因の大半は、使い古されたメディアや不良なメディアによるものです。障害の原因を特定しないでバックアップを続けると、バックアップの信頼性が低下し、バックアップの意味がなくなります。

この章では、以下について説明します。

- 「読取り不能なバックアップのトラブルシューティング」（71 ページ）
- 「他のシステムで作成したメディアの読取り」（72 ページ）
- 「バックアップ中のエラーのトラブルシューティング」（74 ページ）
- 「誤ったバックアップをリストアした後の正しいバックアップのリストア」（75 ページ）
- 「不良メディアの検査」（76 ページ）
- 「バックアップと回復に関するエラー・メッセージと対策」（77 ページ）

読取り不能なバックアップのトラブルシューティング

バックアップが読取り不能になる原因としては次のことが考えられます。

- 年数経過やメディアの不良によってバックアップ・テープのデータが破壊されている。
- テープ・ヘッドの位置が違っている、またはバックアップの作成時に違っていた。
- テープ・ヘッドが汚れている、またはバックアップの作成時に汚れていた。

`/var/adm/SYSLOG` を調べると、使用しているテープ・ドライブが上記の状態にあるかどうかを確認できます。

他のシステムで作成したメディアの読取り

他社のワークステーションで作成したテープは、tar や cpio などの標準ユーティリティで作成したものでも、使用しているワークステーションで読取れない場合があります。原因の 1 つとして、テープの形式に互換性がないことが考えられます。この場合は、バックアップを作成したテープ・ドライブと使用しているテープ・ドライブとの互換性を確認します。

ドライブの完全な互換性を確認できない場合は、dd で最低限のレベルでテープが読取れるかどうかを調べます。テープをドライブに挿入し、次のコマンドを入力します。

```
mt blksize
```

このオプションを指定して mt(1) コマンドを実行すると、テープの書込み時に使用されたブロック・サイズが表示されます。dd でテープを読取る場合は、これ以上のブロック・サイズを指定します。たとえば、書込み時に使用されたブロック・サイズが 1024 バイトである場合は、次のコマンドを入力します。

```
dd if=/dev/tape of=/usr/tmp/outfile bs=1024
```

dd がテープを読取ると、読取られたレコード数と書込まれたレコード数が表示されます。dd がテープを読取れない場合は、テープ・ドライブが汚れていないことと正常に動作することを検査します。ドライブの検査は、各自のシステムで作成したテープを用いて行います。

dd で読取れるテープで tar や cpio などの標準ユーティリティで作成されたものは、dd を使用してデータ・フォーマットを変換できます。

- swab — 2 バイトごとに値を入換えます。
- sync — ibs で指定された長さになるように各入力ブロックの埋込み処理を行います。
- block — 可変長 ASCII を固定長 ASCII に変換します。
- unblock — 固定長 ASCII を可変長 ASCII に変換します。
- noerror — エラーが発生しても処理を中止しないようにします。

dd プログラムはまったく異なるデータ・フォーマット間の変換もできます。

- ascii — EBCDIC を ASCII に変換します。
- ebcdic — ASCII を EBCDIC に変換します。
- ibm — IBM の ASCII を EBCDIC に変換します（IBM の ASCII は多少異なるため）。

また、大文字と小文字の変換もできます。

- lcase — 英字をすべて小文字にします。
- ucase — 英字をすべて大文字にします。

他社のシステムで作成されたデータは、dd を使用して変換し、その変換した出力を別のユーティリティにパイプ処理して読取ることができます。

他社のシステムでは、IRIX とは逆のバイト・オーダが使用されている場合があります。その場合は、次のコマンドを使用して隣接バイトを交換します。

dd if=/dev/tape conv=swab of=/usr/tmp.O/*tapefile*

次に、適切なアーカイブ・ユーティリティを使用し、/tmp/tapefile、または各自で指定した任意のファイルから必要な情報を取り出します。たとえば、他のシステムで tar ユーティリティを使用してテープを作成した場合は、次のコマンドを使用して情報を取出します。

tar xvf /usr/tmp.O/tapefile .

ディスクにファイルを書出さない場合、またはその必要がない場合は dd の出力をローカル・ドライブにパイプ処理できます。また、リモート磁気テープ・ドライブが使用可能ならば、それにパイプ処理することもできます。

バイトの交換を必要としないテープ・デバイスについては、次の tar コマンドでファイルを読取ることができます。

tar xvf /dev/rmt/tps0d4ns

テープ・デバイスが SCSI ユニット 4 で構成されていない場合は、/dev/rmt のデバイス名は、この例とは多少異なる名前（/dev/rmt/tps0d3ns など）になります。

tar アーカイブからファイルを取り出す前に、**t** オプションを使用してアーカイブの内容を確認することをお薦めします。テープ内のシステム・ファイルが絶対パス名で指定されていると、現在のシステムのシステム・ファイルが上書きされる場合があります。たとえば、テープからカーネル /unix をリストアすると、使用しているシステム・カーネルが破壊されます。次のコマンドで上記のアーカイブを事前に確認できます。

tar tvf /tmp/*tarfile*

既存ファイルを上書きせずに、この種のテープからデータを取出すには、次のコマンドを使用して相対パス名でリストアします。

tar Rx

または、これと同等の機能を持つ bru コマンドを使用します。

bru -j

バックアップ中のエラーのトラブルシューティング

バックアップの作成時にシステム・コンソールにエラーが表示される場合、以下の原因が考えられます。

- テープがドライブにロックされていません。この場合は、次のエラー・メッセージが表示されます。

  /dev/nrtape rewind 1 failed:Resource temporarily unavailable

  テープがドライブにロックされていることを確認します。テープをドライブにロックする方法については、ハードウェアの『Owner's Guide』を参照してください。

- ファイルのアクセス権に問題があります。特にファイル指向のバックアップ・プログラムを使用する場合は、バックアップするディレクトリ階層内のすべてのファイルに対するアクセス権があることを確認します。
- ドライブのクリーニングまたは修理が必要です。
- メディアが不良です。76 ページの「不良メディアの検査」を参照してください。

バックアップ作成時に発生した問題は早急な解決を要します。

誤ったバックアップをリストアした後の正しいバックアップのリストア

誤ったバックアップを復元した場合は、バックアップを使用してシステムを再構築します。この作業内容を完全に把握していないかぎり、誤ったバックアップに単に正しいバックアップを重ねて復元するだけでは不十分です。誤ったバックアップによって変更されたファイルは、正しいバックアップで復元できない場合があります。

最悪の場合は、システムを再インストールしてから、バックアップを使用してシステムを適正な状態に復元します。誤ったバックアップによって影響を受けたファイルシステムを復元するには、以下の手順に従います。

`backup` や `bru` などで作成したインクリメンタル・バックアップを誤って復元した場合は、次の手順に従います。

1. 現在のファイルシステムのフル・バックアップを作成します。ファイルシステムを正常にリストアできた場合、このバックアップは不要です。問題が発生した場合は、このバックアップを使用してシステムを現在の状態（不適切でも）に戻します。
2. 誤ってリストアしたバックアップの前に作成した最初のフル・バックアップからリストアを開始します。
3. 正しいバックアップに到達するまで、一連のインクリメンタル・バックアップを適用してシステムを復元します。

`tar` や `cpio` で作成したファイル指向のバックアップを誤って復元した場合は、次の手順に従います。

1. 影響を受けたファイルシステムまたはディレクトリ階層のフル・バックアップを作成します。このフル・バックアップは、万一の障害に対する予防としてだけでなく、バックアップ間のギャップを埋めるためにも必要です。
2. 誤ったバックアップを復元した直前の状態にシステムを戻します。

   インクリメンタル・バックアップを使用する場合は、上記の手順 2 と手順 3（誤ったインクリメンタル・バックアップバックアップからのリストア）に従います。

   `tar` や `cpio` などのユーティリティだけを使用してバックアップを作成している場合は、任意のバックアップを使用してシステムを適正な状態に戻します。

3. システムが適正な状態に近くなったら、正しいバックアップを復元します。システムが適正な状態に戻れば、以下の手順は省略します。これで作業は完了です。

   誤ったバックアップを復元した直前の状態にシステムを戻せない場合は、次の手順に従います。

4. 誤ったバックアップを復元した直前の正しい状態にシステムを戻せない場合は、できるだけ正しい状態に近づけます。

5. 現在の仮状態のバックアップを作成します。

6. 現在の仮状態、手順 1 で作成したバックアップ（誤ったバックアップで復元した状態）、および復元したい正しい状態のバックアップの 3 つを比較します。システムを適正な状態に戻すには、変更されたファイル、追加または削除されたファイル、変更されていないファイルをメモに書留めます。

   上記のメモを使用し、各テープから適正なバージョンのファイルを手作業で取出します。

不良メディアの検査

高品質のメディアであっても時間とともに劣化が進みます。劣化の兆候は次の形で表れます。

- データが正しくテープに収録されたように見えても、バックアップの検証でエラーが発生する場合。バックアップを作成した直後は必ず検証します。

  バックアップ直後の検証でエラーが検出されれば、別のテープを使用してデータをバックアップし、検証にパスすることを確認します。

- 同じデータを保存した 2 つのテープで、一方はデータを正常に取出せるが、他方はデータが破壊されている場合。

- テープにアクセスしようとすると、バックアップ・メディアのデバイス・ドライバ（SCSI テープ・ドライバなど）からシステム・コンソールにエラーが表示される場合。

- テープに情報が書込めない場合。

テープに情報を書込む際にエラーが発生する場合は、テープが書込み禁止になっていないことを確認します。また、ドライブに適合した長さや密度のテープを使用していることも確認します。

ドライブが汚れていないこととテープ・ヘッドの位置が適正であることを確認します。これまで正常であったテープが何本も突然異常を示す場合は、テープ・ヘッドの位置を確認します。

テープに明らかに問題がある場合は、不良品として廃棄します。他のユーザが誤って使用しないように、テープ・ケースに「不良品」と書いて廃棄します。

バックアップと回復に関するエラー・メッセージと対策

ここでは、バックアップまたは回復に関する問題を示すエラー・メッセージについて説明します。

```
unix: dks0d1s0: Process [tar] ran out of disk space
```
([tar] 実行中にディスクの空領域が不足)

この種のディスク領域不足を示すエラーは、データのバックアップ先のディスク・パーティションに十分な空領域がないときに起こります。

同様のエラーは、データの回復先のディスク・パーティションに十分な空領域がない場合にも起こります。圧縮したバックアップ・データを解凍すると、元の 2 倍の領域を必要とする場合があります。

この場合は、ディスク領域の追加、ディスク領域の再要求、または現在のディスク・パーティションの再分割を行います。『IRIX Admin: Disks and Filesystems』を参照してください。また、データ圧縮を採用するなど、バックアップ手順を再検討します。24 ページの「データ圧縮したファイルのバックアップ」を参照してください。

```
unix: ec0: no carrier: check Ethernet cable
```

```
unix: NFS write error 151 on host garfield
```

```
unix: NFS2 getattr failed for server some.host.name: Timed out
```

この種のネットワーク・エラーは、ネットワークのリソース（リモートの磁気テープ・ドライブやディスク・ドライブなど）を利用してバックアップまたは回復を行う場合だけに起こります。この場合は、ネットワークを正しく接続し直します。『IRIX Admin: Networking and Mail』を参照してください。その後、バックアップまたは回復に問題がないことを確認し、問題があれば処理をやり直します。

```
unix: Tape 3: Hardware error, Non-recoverable

unix: Tape 3: requires cleaning

unix: Tape 3: Unrecoverable media error

unix: NOTICE: SCSI tape #0, 6 had 1 successful retried commands

unix: NOTICE: SCSI tape #0,7 Incompatible media when reading

Could not access device /dev/rmt/tps0d6nr, Device busy
```

以上は、すべてテープ・アクセス・エラーの例です。バックアップの場合は読込みエラー、回復の場合は書込みエラーがシステムに発生したことを示します。エラー・メッセージが示すドライブにテープがセットされていること、バックアップの場合はテープが書込み禁止になっていないことを確認します。また、磁気テープ装置は、製造元の指示に従って定期的にクリーニングします。

ほかの原因である場合は、バックアップまたは回復用のユーティリティを使用し、そのテープに書込みまたは読込みができるかをテストします。テープでメディア・エラーが起きた箇所は限定できないので、書込みまたは読込みのテストはテープ全体に対して行います。［ツールチェスト（Toolchest)］の［システム（System）］->「信頼性試験を実行（Run Confidence Tests）」を選択し、［テープ・ドライブ・テスト（Tape Drive Test)］をダブル・クリックすることもできます。

テープが古くなるなど、品質に疑問が生じた場合は、そのテープのデータを新しいテープにコピーし、古いテープを廃棄します。初めて使用する磁気テープ装置の場合は、その装置とテープの種類が適合することを次の手順で確認します。mt(1) コマンドを実行してテープ・ドライブをリセットします。次に、hinv(1M) コマンドを実行し、システムがその磁気テープ装置を認識することを確認します。

［devece already in use］または ［device busy］のエラーが表示されたときは、別のプログラムがそのテープ・ドライブを使用している可能性があります。

第 II 部

セキュリティ

第 II 部のセキュリティには次の章が含まれます。

第 4 章
IRIX システム・セキュリティ

第 5 章
ネットワークのセキュリティ

IRIX システム・セキュリティ

この章では、ローカル・システムのセキュリティを取上げます。以下について説明します。

- 「システム・セキュリティについて」（82 ページ）
- 「IRIX の標準セキュリティ機能」（82 ページ）では、IRIX に組込まれている標準のセキュリティ機能を説明します。
- 「セキュリティの保護と注意」（83 ページ）では、セキュリティの一般的な盲点を指摘し、そのチェック方法について説明します。
- 「パスワード管理」（87 ページ）では、システム・ソフトウェアとユーザ・アカウントを正しく設定および管理し、セキュリティを強化する方法について説明します。
- 「システム・ログインとアカウントの管理」（97 ページ）では、ユーザのログインと特殊アカウントの適切な管理について説明します。
- 「set-UID と set-GID のパーミッションについて」（104 ページ）では、実行時にユーザ ID とグループ ID の設定を許可するファイルのパーミッションの性質と管理方法について説明します。
- 「汎用のファイルとディレクトリのパーミッションについて」（107 ページ）では、パーミッションの設定について説明し、すべてのユーザが読み書きできる IRIX のファイルとディレクトリをリストアップします。
- 「パスワードなしで出荷されるアカウント」（108 ページ）では、`/etc/passwd` 内のユーザ・アカウントで、出荷時にパスワードが設定されていないものをリストアップします。
- 「セキュリティ関連のファイルとコマンド・リファレンス」（109 ページ）では、IRIX のセキュリティ関連ファイルとコマンドをリストアップします。
- 「セキュリティの強化機能」（110 ページ）では、アクセス・コントロール・リスト（ACL: Access Control List）と最小限の特権機能について説明します。

システム・セキュリティについて

システムのセキュリティを確立した後では、安全領域をネットワークまで拡大できます。しかし、ローカル・セキュリティを確立しないかぎり、セキュリティの範囲を広げる意味はありません。

セキュリティが完全に確立されることはありません。セキュリティの問題は日々変化します。その問題を理解し、変化に対応し、多数のツールを駆使してシステムを継続的に監視する必要があります。この章で安全管理の方針を確立し、それを具体化するための方法について説明します。

その 1 つは孤立した IRIX システムを安全にすること、次にローカル・エリア・ネットワークを安全にすること、そして、インターネットのような外部のワイド・エリア・ネットワークに接続されたサイトを安全にすることです。この章では、孤立したシステムの安全対策について主に説明します。しかしその多くは、ネットワークまで安全領域を拡大する前に行わなければいけません。

第 5 章「ネットワークのセキュリティ」では、この章で取上げる問題を十分に理解した上で、ネットワークのセキュリティの重要性について説明します。システムを信頼性のあるローカル・エリア・ネットワークだけに接続し、インターネットに接続しない場合でも、第 5 章 132 ページの「ローカル・エリア・ネットワークのアクセス」の最初の部分は必ず参照してください。

メモ：システム・マネージャの GUI には、システムのセキュリティを管理するためのセキュリティとアクセス制御に関する機能があります。詳細については、『Personal System Administration Guide』を参照してください。この章では、GUI が提供する同じ機能の多くをコマンド行とファイル・インタフェースに基づいて説明します。

IRIX の標準セキュリティ機能

IRIX には、新しいソフトウェアを追加しなくても、一般的に許容できるセキュリティ・レベルを確保する機能が用意されています。IRIX システムの大きな長所は、ユーザがファイルとデータを簡単に共有できることです。しかし、この章で示すセキュリティに関するガイドラインに従うと、システムにアクセスしにくくなる場合もあります。システム管理者は、ユーザが希望する簡単なアクセスとセキュリティ対策という相反する事項のバランスを取る必要があります。

IRIX の標準セキュリティ機能は、以下のとおりです。

- ファイルの所有権には — 所有者、グループ、一般ユーザ — の 3 つのクラスがあります。どのクラスにファイルのパーミッションを与えるかは、ファイルの所有者が指定できます。
- パーミッションには — 読込み、書込み、実行 —（または execute ディレクトリの場合には探索）の 3 種類があります。どの種類のパーミッションをユーザに与えるかは、ファイルの所有者が指定できます。
- 暗号化されたパスワードで個別に保護されたユーザ・アカウント。
- ログイン試行とシステムの利用状況を監視するツール。次のツールがあります。
  - who(1) コマンドを使用して、システムにログインしているユーザを確認するツール
  - ps(1) コマンドを使用して、実行中のプロセスを確認するツール
  - プロセス・アカウンティング・コマンドを使用して、システムの利用状況のログを記録するツール
- crypt(1) コマンドを使用して、データを暗号化する機能。

セキュリティの保護と注意

コンピュータのセキュリティは、サイト管理者だけでなく、サイトのコンピュータを利用するすべてのユーザの責任です。システムのユーザは、各自が管理するアカウント・パスワードを使用するほかに、ファイルとディレクトリの適切なパーミッションを使用して、各自のデータを保護する必要があります。

サイト管理者と、必要に応じてシステムのユーザは、以下の事項に注意してください。

- コンピュータやディスク・ドライブは、物理的に近付ける者ならだれでも持去ることができます。
- システムのバックアップも持去られる可能性があります。バックアップは安全な場所に保管します。バックアップ・テープに保存されている情報には、物理的に近付ける者ならだれでもアクセスできます。
- ディレクトリとファイルのパーミッションは、所有者、グループ、一般ユーザ別に必要なものだけを設定します。特定のアカウントのセキュリティが損なわれた場合に、被害を最小限に抑制できます。

- アカウントとパスワードでシステムを保護する方法は、いくつかあります。
  - ログイン・ユーザのアカウントを限定すると、システムで行われた活動の責任の所在が判明します。
  - IRIX システムのファイルのパーミッションを利用すると、ユーザはデータを安全に管理できます。システムの他のユーザが機密情報に偶然触れる機会は減ります。
  - すべてのアカウントにパスワードを設定すると、パーミッションのない者がシステムにアクセスする機会は減少します。しかし、パスワードの定期的な変更や、効果的なパスワードの選択を怠ると、パーミッションのない者がシステムにアクセスする可能性が高まります。効果的なパスワードの選び方については、次の節で説明します。
- アクティブ・アカウントには例外なくパスワードが必要です。パスワードは定期的に変更します。簡単に推測できるパスワードは使用しません。パスワードを単純なテキスト形式でオンラインに保存することも避けます。パスワードを書留める場合は、書留めたメモを安全な場所に保管します。

  パスワードの選び方については、87 ページの「パスワード作成のガイドライン」を参照してください。
- 共用アカウントはセキュリティの盲点です。共用アカウントの一例としては、部門または作業グループのメンバ全員が共用するアカウントが挙げられます。サイトのすべてのワークステーションにある標準「guest」アカウントも、共用アカウントの一例です。共用アカウントを使用すると、サイトのユーザ全員が各ワークステーション固有のアカウントがなくても、他のワークステーションにアクセスできます。

  共用アカウントの落とし穴は、特定のワークステーションで共有アカウントによる活動が行われた場合、その責任の所在を明確に特定できないことです。また、サイトのワークステーションに侵入しようとする者は、*guest* のようなありふれたアカウント名を試さないはずがないと考えられます。

  共用アカウントは便利ですが、セキュリティに関する重大な問題を起こしかねません。パスワードを設定していない共用アカウントは特に危険です。
- `last` コマンドを使用し、定期的にシステムにログインしている人を監視してください。3 番目のフィールドに、リモートからログインしたユーザのホスト名または IP アドレスが表示されます。last(1) マン・ページを参照してください。
- 使用しなくなったアカウントは、ロックするか、バックアップを作成して削除します。使用していないアカウントもカレント・アカウント同様、簡単に悪用されるおそれがあります。

退職者が出た場合は、ダイヤルアップも含め、重要なパスワードは必ず変更します。退職者はサイトのワークステーションまたはサーバーにアクセスできないようにします。

- ダイヤルアップ・ポートを備えたシステムには、特別なダイヤルアップ・アカウントとパスワードを設定します。これは上で説明したように、共用アカウントがあるサイトにとって特に重要です。90 ページの「第 2（ダイヤルアップ）パスワードの設定」の `/etc/d_passwd` の説明を参照してください。

  この予防措置を追加した場合でも、ダイヤルアップ・アクセスの可能なワークステーションには機密データを格納しないでください。

- サイトで ftp(1C) などを通じてインターネットにアクセスできる場合は、予防措置として、特定のゲートウェイ・ワークステーションへのアクセスを分離します。外部ネットワークへの接続の詳細については、137 ページの「ネットワーク・セキュリティとファイアウォールについて」を参照してください。

- su(1M) コマンドは、必要な場合だけに使用します。`su` コマンドを使用すると、自分のユーザ ID を他人のユーザ ID に変更できます。ただし、正当な理由があって他のユーザが所有する情報にアクセスするために、`su` コマンドを使用することもあります。この場合は、注意が必要です。`su` でユーザ ID を切替えたユーザは、他人のパスワードを知り、そのアカウントにアクセスできるようになります。

メモ：`/var/adm/sulog` ファイルには、`su` コマンドが成功した場合と失敗した場合（- 記号が付きます）の両方がログに記録されます。デフォルトでは、`/etc/default/su` ファイル内ですでにログ機能が有効に設定されています。

- 各ユーザのホーム・アカウントと特にシェル・スタートアップ・ファイルの `.profile`、`.login`、`.cshrc` は、そのユーザだけが書込めることを確認します。これは、ユーザのログイン・ファイルに「トロイの木馬」プログラムが入ることを防止するためです。トロイの木馬プログラムとは、見かけは普通のファイルですが、正規のユーザが起動すると破壊的な動作をするファイルです。

- `/` (root)、`/bin`、`/usr/bin`、`/etc` などのシステム・ディレクトリとその中のファイルは所有者しか書込めないようにします。これもトロイの木馬の不正行為を防ぎます。

- コンソール、ワークステーション、または端末を無人にする場合は、システムをログ・オフします。*root* としてログインしている場合、これは特に大切です。xlock(1) のマン・ページも参照してください。ローカル X ディスプレイのロックについて説明しています。

- 機密のデータ・ファイルは暗号化します。crypt(1) コマンドはエディタ（`ed` と `vi`）の暗号化機能と共に、機密情報を保護します。
- 定評あるメーカから入手したソフトウェアだけを使用します。公共のプログラム、特にコンパイル済みのバイナリは要注意です。公共の掲示板システム（ベンダが運営または後援している BBS システムを除く）や公共のコンピュータ・ネットワークには、システム・セキュリティを侵害し、データ損失の原因となる悪質なウィルスが侵入しているおそれがあります。

  パブリック・ドメインのソース・コードはコンパイル済みのプログラムよりは安全ですが、コンパイル前にコードを徹底して検査することが前提です。実行するため ID を *root* に設定（set-UID root）してインストールするようなプログラムは、疑う余地があります。
- ネットワークのハードウェアは保護し、定期的に検査します。コンピュータ・システムに侵入する手口の 1 つは、ネットワーク・ケーブルに物理的なタップを適用してネットワーク・トラフィックを盗聴することです。タップは、バンパイヤ・タップなどの物理的な接続の場合も誘導タップの場合もあります。

  ネットワーク・ケーブルは安全な領域に敷設して、定期検査が容易にできるようにします。無許可タップを発見しやすくするために、ネットワークのハード・コピー・マップを作成し、管理します。この種の攻撃を防ぐ別の手段としては、光ファイバ（FDDI）ネットワーク・ハードウェアを使用します。これは、ケーブルに侵入があると正しく動作しないハードウェアです。ネットワーク・ソフトウェアを安全に構成する方法については、第 5 章「ネットワークのセキュリティ」を参照してください。

IRIX でのシステム・セキュリティでは、システム・ログイン・アカウントとパスワードが重要です。大部分のサイトでは、適切な管理、ユーザ教育、提供されている機能の活用によって適切なセキュリティを実現できます。セキュリティ違反のほとんどは、操作ミスと提供されているセキュリティ機能の不適切な使用によるものです。基本的なセキュリティ機能を使用しなかったり、ユーザの不注意が是正されないと、セキュリティ対策を強化しても安全性は向上しません。anonymous FTP で定期的に `sgigate.sgi.com` にログインし、`~ftp/security` ディレクトリにシステムの安全対策に関するパッチがないかどうかを調べます。

メモ: システムで NFS または NIS を使用している場合は、151 ページの「ファイアウォールでの NIS（YP）の無効化」と 152 ページの「ファイアウォールでの NFS アクセスの禁止」を参照してください。

パスワード管理

パスワード管理では、以下について説明します。

- 87 ページの「パスワード作成のガイドライン」では、他人が簡単に推測できない安全なパスワードの作成方法について説明します。
- 88 ページの「PROM パスワードについて」では PROM パスワードの使い方を説明します。
- 90 ページの「第 2（ダイヤルアップ）パスワードの設定」では、第 2 パスワードを特定の tty 回線に接続する方法を説明します。第 2 パスワードは、システム・パスワードまたはダイヤルアップ・パスワードとも呼ばれます。
- 93 ページの「シャドウ・パスワードについて」では、シャドウ・パスワード・ファイルを使用して、標準パスワード・ファイルに格納されている暗号化されたパスワードも含めて隠す方法を説明します。
- 94 ページの「パスワード・エージングについて」では、一定の期間ごとにユーザにパスワードを変更させる方法を説明します。
- 97 ページの「pwck によるパスワード・ファイルのチェック」では、パスワード・ファイルを検査するための便利なツールを紹介します。

パスワードの管理については、『IRIX Admin: System Configuration and Operation』にも説明があります。

パスワード作成のガイドライン

システムが最も安全なのは、アカウントとパスワードなしではだれもアクセスできず、いずれのパスワードも簡単に推測または入手できない場合です。侵入者が簡単に推測できるパスワードを使用したり、パスワードを書留めたメモをワークステーションのそばに放置するユーザが大勢います。

複数の管理用アカウントで共通のパスワードを使用するシステム管理者も少なくありません。これは歓迎できない方法です。複数のアカウントには、同一のパスワードを使用しないでください。

安全性の高いパスワードは、以下のものです。

- 長いパスワード（最初の 8 文字が認識されます）
- 複数の語を工夫して組合わせるか、配列したパスワード
- 複数の言語の語を固有な方法で組合わせたパスワード
- 数字や句読点などの異なる種類の文字で構成したパスワード
- 以上のすべての特徴を備えたパスワード

推測されやすいパスワードとは、以下のものです。

- 短いパスワード
- 辞書に載っている単語
- アカウント名と同じ、または逆順に綴ったパスワード
- ユーザの所属部門やプロジェクトの名前
- ユーザの名前またはイニシャル
- ユーザの免許証の番号、配偶者や友人の名前、住所、電話番号、年令などの明白な情報
- top secret、secret、private、password、friend、key、god、me などのありふれた語

PROM パスワードについて

システムには、コマンド（PROM）モニタ・プログラムにアクセスするユーザに対してパスワードを要求する機能があります。この機能を使用すると、システム管理業務に従事する者の監督を強化できます。

従来より、侵入者がシステム・ハードウェアにアクセスできた場合は、システム・セキュリティをほとんど保持できません。最も単純なケースでは、侵入者はシステムの電源を落としてから再起動し、コンソールから通常のオペレーティング・システム以外のプログラムを起動するようにシステムに命令します。または、単にマシンからハード・ディスクを持去り、それを別のマシンにインストールしてファイルを盗み読みすることもあります。ハードウェアの物理的な盗難防止にはシステム・ソフトウェアで対抗できません。しかし、PROM パスワードを使用すると、侵入者が自分のプログラムを起動したり、システムに損害を加えるのを最小限に抑制できます。

PROM パスワードを忘れてしまっても、*root* のパスワードを覚えていれば、マシンによっては `nvram` コマンドで PROM パスワードをリセットできます。PROM パスワードのリセットがうまくいかない場合は、CPU ボードから PROM またはジャンパを取外します。この手順の詳細については、『Owner's Guide』を参照してください。

PROM パスワードを忘れてしまったとき、新しいものを割当てるには、まず現在の PROM パスワードを、`nvram` コマンドで IRIX システムから消去し、`passwd` コマンドを PROM モニタから起動して新しいものを割当てます。

nvram による PROM パスワードの消去

PROM パスワードを nvram(1M) コマンドを使って消去するには、次の手順に従います。

1. *root* でログインします。
2. 次のコマンドを入力します。

   ```
   nvram passwd_key ""
   ```

   これで PROM パスワードが消去されます。

コマンド・モニタからの PROM パスワードの設定

コマンド・モニタから PROM パスワードを設定するには、次の手順に従います。

1. *root* でログインし、システムを停止します。
2. 次のメッセージが表示されたら、**<Esc>** キーを押して「System Maintenance」メニューを表示します。

   ```
   Starting up the system...
   To perform system maintenance instead, press <Esc>
   ```

3. ［System Maintenance］メニューからオプション 5 を選択して、コマンド・モニタに入ります。コマンド・モニタのプロンプトが表示されます。

   ```
   >>
   ```

4. passwd コマンドを入力して、**<Enter>** キーを押します。

   ```
   passwd
   ```

   次のプロンプトが表示されます。

   ```
   Enter new password:
   ```

5. マシンで使用するパスワードを入力し、**<Enter>** キーを押します。次のプロンプトが表示されます。

   ```
   Confirm new password:
   ```

6. 前に入力したのと同じパスワードを再度入力します。最初と同じパスワードを入力すると、コマンド・モニタのプロンプトが再び表示されます。これでパスワードの設定は完了です。コマンド・モニタにアクセスする場合は、このパスワードを必ず入力します。

87 ページの「パスワード作成のガイドライン」を参照し、適切なパスワードを選択します。

第 2（ダイヤルアップ）パスワードの設定

システムに追加の保護機能が必要な場合は、システム・パスワードを設定できます。特定のポート（tty）にログインするユーザは、アカウント・パスワードに加えてシステム・パスワードを入力するように要求されます。この機能は、システム・コンソール、あるいは clogin または xdm が使用される端末には設定できません。

通常、システム・パスワードはダイヤルアップ回線だけで使用されるので、ダイヤルアップ・パスワード（dialup password）とも呼ばれます。ダイヤルアップ・パスワードは、標準回線でも使用できますが、通常は必要ありません。

システム・パスワードを設定するには、以下の手順に従います。

1. *root* でログインします。
2. `/etc/dialups` ファイルを編集します。第 2 パスワードを必要とするポート（tty）のリストをファイルに入力します。以下に例を示します。

   ```
   /dev/ttyd1
   /dev/ttyd2
   /dev/ttyd3
   ```

   ポートとして可能な名前はリンクも含めてすべて入力します。ファイルに入力した後、エディタを終了します。
3. 希望する 1 つまたは複数のパスワードを決定します。システム・パスワードをシェル単位で割当てます。システムで使用できるすべてのシェルに同じパスワードを割当てることも、シェルごとに異なるパスワードを割当てることもできます。2 つ方法を組合わせて割当てることもできます。
4. 希望するパスワードを暗号化します。暗号化するには、`passwd` プログラムを使用します。crypt(1) コマンドは使用できません。

   パスワードを暗号化するには、`/etc/passwd` ファイルのバックアップ・コピーを作成し、アカウントのパスワードを変更して（たとえば、*dialup* などの新しいアカウントを作成して）、`/etc/d_passwd` に使用するパスワードを取得します。パスワードの暗号化が完了したら、`/etc/passwd` から作成したバックアップ・コピーを復元するか、入力した *dialup* アカウントを削除してパスワード・ファイルを元の状態に戻します。アカウントの暗号化されたパスワードは、`/etc/passwd` ファイルの 2 番目のフィールドに記述されます。2 番目のフィールドが **x** 表示になっているときは、`/etc/shadow` に記述されています。

   たとえば、`bin` アカウントのパスワードを「2themoon」に変更するには、次のように入力します。

   ```
   passwd bin
   New password:
   ```

   文字列「2themoon」を入力した後 **<Enter>** キーを押しても、文字列「2themoon」は画面上に表示されません。

   次のメッセージが表示されます。

   ```
   Re-enter password:
   ```

「2themoon」を再入力して、**<Enter>** キーを押します。この文字列も画面上には表示されません。

`/etc/passwd` ファイルの `bin` アカウントの入力を確認します。次のように表示されます。

```
bin:SaXub4uaL5NP:2:2:System Tools Owner:/bin
```

1 番目と 2 番目のコロンにはさまれた 2 番目のフィールドは、パスワード「2themoon」を暗号化したものです。実際は、これと違ったものになることもあります。同じパスワードでも、パスワードの暗号化はシステムに与える初期値に依存するためです。

5. `/etc/d_passwd` ファイルを編集します。ファイルに次の形式の行を入力します。

   ```
   shell:password:
   ```

   shell はパスワードを必要とするコマンド・インタープリタ（シェル）であり、`password` は暗号化されたパスワードです。`/etc/passwd` で使用されるすべての shell（7 番目と最後のフィールド）は、UUCP、PPP、SLIP などを含めて、このファイルに入力します。

   たとえば、前の手順 4 で「2themoon」を暗号化したパスワードを、`/etc/dialups` に指定したポート（tty）にログインする各 C シェル・ユーザに割当てるには、次のコマンドを使用します。

   ```
   /bin/csh:SaXub4uaL5NP2:
   ```

   暗号化したパスワードの最後にはコロンを入れ、`/etc/passwd` に使用されているのと同じシェル・プログラムのパス名を入力します。

   ファイルに入力し、エディタを終了します。

6. 次のコマンドを入力して、ファイルに適切なパーミッションがあるかどうかを確認します。

   ```
   chmod 640 /etc/d_passwd /etc/dialups
   ```

7. 手順 4 でシステム・アカウントに割当てたパスワードを取除きます。`/etc/passwd` ファイルを編集して、2 番目のフィールドの文字列を除去します。このフィールドを、この手順を開始したときと同じ状態に戻します。

これで、`/etc/dialups` に指定したポート（tty）にログインしたすべての C シェル・ユーザは、アカウント・パスワードのほかにシステム・パスワードとして「2themoon」の入力を要求されます。

`/bin/ksh`、`/usr/local/bin/bash`、`/usr/bin/tcsh` など、システムで使用されるほかのログイン・シェルに対しても、同様のエントリを設定します。

シャドウ・パスワードについて

シャドウ・パスワード・ファイルは標準パスワード・ファイルのコピーですが、権限のないユーザはアクセスできません。標準構成では、すべてのユーザが `/etc/passwd` を読取ることができます。`/etc/passwd` ファイルには、ユーザの暗号化されたパスワードが入っているので、だれでもパスワードをコピーして復号化し、悪用できます。シャドウ・パスワード・ファイルを使用すると、侵入者によるパスワードの復号化を防止できます。

シャドウ・パスワード・ファイルは `/etc/shadow` と呼ばれます。シャドウ・パスワードが設定されると、各 `/etc/passwd` エントリのパスワード・フィールドは文字「x」に置き換えられます。すべての標準パスワード・ツールは、シャドウ・パスワードを透過的に使用します。ユーザには、`/etc/passwd` ファイルの暗号化されたパスワードを見ることができない点を除いて、その違いに気付かれてはいけません。

システム動作の1つの相違点として、古いアプリケーションでは、getpwent(3C)とgetpwnam(3C)のライブラリ呼出しから *pw_passwd* の正しい値を得られません。この点は、ルートの権限がないスクリーン・セーバ・プログラムに影響します。

メモ：NIS でのシャドウ・パスワードの扱いは異なります。NIS でのシャドウ・パスワードの使い方については、shadow(4) マン・ページを参照してください。

シャドウ・パスワード・ファイルの使用

`/etc/shadow` を初期設定し、シャドウ・パスワードを呼出すには、`pwconv` コマンドを実行します。pwconv(1M) マン・ページを参照して下さい。このコマンドを実行すると、シャドウ・パスワードが有効になります。

パスワード・ファイルとシャドウ・パスワード・ファイルを同時に更新するには、`passmgmt` コマンドを使用します。詳しくは、passmgmt(1M) マン・ページを参照してください。グラフィカルなシステム・ユーザ・マネージャを使うと、シャドウ・パスワードが有効な場合に限り、同時に更新することができます。

パスワード・エージングについて

パスワード・エージング機構は、パスワードを定期的に変更するようにユーザに要求します。指定の期限前にパスワードを変更するのを防止したり、逆にパスワードをすぐに変更させることもできます。

メモ：NIS のエントリは、パスワード・エージングをサポートしていません。passwd(4) マン・ページを参照してください。

実際には、パスワード・エージングを利用するユーザは、アカウントのパスワードを最低 2 つ用意して交互に入替えています。古いパスワードが期限切れになるごとに新しいパスワードを考えるよりは、2 つの覚えやすいパスワードを交互に使用する方が簡単だからです。IRIX には、ユーザが 1 組のパスワードから選択しているかどうかを確認し、完全に別のパスワードを選択させる機能はありません。

passwd コマンドによるパスワード・エージングの管理

ユーザが各自のパスワードを変更するまでの最大日数を設定するには、passwd(1) コマンドを次の構文で使用します。

passwd -x *max name*

max はユーザ *name* のパスワードの有効期限です。たとえば、次のコマンドは、ユーザ *alice* にパスワードを 2 週間（14 日）ごとに変更させます。

```
passwd -x 14 alice
```

max を 0 に設定すると、ユーザは次のログインでパスワードを変更しなければいけません。それ以降はパスワード・エージングが無効になります。**-x** を -1 に設定すると、ユーザのパスワード・エージングはすぐにオフになります。

ユーザがパスワードを変更できるまでの最小日数を設定することもできます。これは、ユーザが自分のパスワードを変更した後に元のパスワードに再度変更するのを防止します。次に例を示します。

```
passwd -x 14 -n 7 ralph
```

この例では、ユーザ *ralph* がパスワードを変更できるのは最大で 14 日ごとですが、最小では 7 日に 1 回だけです。最小値を最大値よりも大きく設定すると、ユーザはパスワードを一切変更できなくなります。

ユーザにパスワードをすぐに変更させるには、**-f** オプションを使用します。たとえば、次のように入力します。

```
passwd -f trixie
```

/etc/passwd の編集によるパスワード・エージングの管理

パスワード・エージングを強制する別の方法としては、`/etc/passwd` ファイルを編集し、目的とするアカウント・エントリのパスワード・フィールドの後に必要な情報を挿入します。

パスワード・エージング情報は、`/etc/passwd` ファイルの暗号化されたパスワード・フィールドに追加されます。パスワード・エージング情報は、次の形式に従って 1 つのカンマ (,) と最大 4 バイトまでの文字で構成されます。

```
,Mmww
```

以下に、これらのフィールドの意味を示します。

`,`	パスワードとエージング情報を区切ります。
`M`	パスワードの最大有効期限です。
`m`	ユーザが既存のパスワードを変更できるまでの最小期間です。
`ww`	パスワードが最後に変更された週（1970 年の年頭から計算）で、2 文字 `ww` で表します。この情報は入力しません。この文字は、システムがパスワード・エージング情報に自動的に追加します。

すべての時間は、64 文字のアルファベットを使用して週（0 から 63）単位で指定します。次に、数値と文字コードの関係を説明します。パスワード・エージング情報の 4 つのフィールドには、

どの文字コードでも使用できます。表 4-1 にパスワード・エージング・コードとその意味を示します。

表 4-1　パスワード・エージングの文字コード

文字	週の数
.（ピリオド）	0（ゼロ）
/（スラッシュ）	1
0 から 9	2 から 11
A から Z	12 から 37
a から z	38 から 63

この文字コードには、2 つの特殊なケースが適用されます。

- `M` と `m` がゼロである場合、ユーザは次のログイン時にパスワードを変更します。そのログイン・アカウントには、以後、パスワード・エージングが適用されません。
- `m` が `M` より大きい場合、そのログイン・アカウントのパスワードを変更できるのは *root* だけです。

次に示す例では、パスワード・エージング情報は、ユーザ *ralph* に 2 週間（0）ごとに新しいパスワードを設定させ、そのパスワードを 1 週間（/）以内に変更することを拒否します。

```
ralph:RSOE2m.E,0/:100:1:Ralph P. Cramden:/usr/people/ralph:
```

変更後に *ralph* が最初に行うログインの後、システムは自動的に 2 文字、つまり、最後に変更した情報をパスワード・フィールドに追加します。

```
ralph:RSOE2m.E,0/W9:100:1:Ralph P. Cramden:/usr/people/ralph:
```

この例では、*ralph* は W9 週にパスワードを変更しました。*ralph* に次のログイン時のパスワード変更を一度だけ強制するには、パスワード・フィールドにコード **,..** を追加します。

```
ralph:RSOE2m.E,..:100:1:Ralph P. Cramden:/usr/people/ralph:
```

ralph がパスワードを変更すると、システムはパスワード・フィールドから自動的にエージング・コード（**,..**）を取除きます。*ralph* にパスワードを変更させないためには、コード **,./** を使用します。`/etc/passwd` ファイルを編集し、カンマ（,）、ピリオド（.）、スラッシュ（/）をパスワード・フィールドに追加します。

```
ralph:RSOE2m.E,./:100:1:Ralph P. Cramden:/usr/people/ralph:
```

これで、*ralph* アカウントのパスワードを変更できるのは *root* のみになります。*ralph* がパスワードを変更しようとすると、`permission denied` というメッセージが表示されます。

pwck によるパスワード・ファイルのチェック

時々、pwck(1M) ユーティリティを実行し、パスワード・ファイルを検査します。このプログラムは、ファイルを読取り、各エントリの整合性をチェックし、矛盾がある場合には記録します。パスワード・チェックでは、以下の項目が検査されます。

- 各エントリのフィールド数
- ログイン名
- ユーザ ID 番号
- グループ ID 番号
- ログイン・ディレクトリ
- 実行されたプログラム

デフォルトのパスワード・ファイルとして `/etc/passwd` がチェックされます。シャドウ・パスワード（93 ページの「シャドウ・パスワードについて」を参照）が有効になっている場合は、`/etc/shadow` ファイルがチェックされます。

同様に、grpck(1M) コマンドは `/etc/group` ファイルのすべてのエントリを検査します。デフォルトのグループ・ファイルとして `/etc/group` がチェックされます。デフォルトと異なるファイルをチェックする場合は、いずれのコマンドでもコマンド行で指定します。

システム・ログインとアカウントの管理

ここでは、特殊アカウントとログイン・アカウントの管理方法を説明します。特殊アカウントは、システムが特定のシステム機能を実行するためのもので、ログイン・アカウントは汎用のシステム・アクセスをユーザ・アカウントに許可するためのものです。

特殊アカウントについて

特殊アカウントは、デーモンが UUCP ジョブやプリント要求のスプーリングなどのシステム機能を実行するためのアカウントです。キー・ファイルは特殊アカウントが所有しているため、特殊アカウントのどれかにアクセスできた者やシステムでデーモンを起動できた者は、セキュリティを部分的に突破できます。セキュリティを部分的にしか破れないのは、各種システム・ファイルの所有権が複数の特殊アカウントに分散されているからです。

すべての特殊アカウントへのアクセスは、*root* アカウントの場合と同様に保護します。各アカウントには、パスワードを設定するか、99 ページの「使用しないログインのロック」で説明しているいずれかの方法でロックします。

次に、システムのすべての管理用アカウントと特殊アカウントをリスト表示し、各アカウントの用途について説明します。

root　このログインには一切の制約がなく、他のすべてのログイン、保護、パーミッションに優先します。オペレーティング・システム全体にアクセスできます。*root* ログインのパスワードは入念に保護します。

sys　このログインは、`/usr/src` に存在する sys 所有下のファイルに対し、正規ユーザ・ログインの権限を持っています。このログインは使用禁止にします。

bin　このログインは、システム全体に存在する bin 所有下のファイルに対し、正規ユーザ・ログインの権限を持っています。このログインは使用禁止にします。

adm　このログインは、`/var/adm` に存在する adm 所有下のファイルに対して、正規ユーザ・ログインの権限を持っています。su コマンドを使用して adm にログインできます。このログインは使用禁止にします。

uucp　このログインは、オブジェクト・ファイルとスプールされたデータ・ファイルを `/usr/lib/uucp` と `/etc/uucp` に所有しています。

nuucp　このログインは、リモート・ワークステーションがシステムにログインし、`/usr/lib/uucp/uucico` を経由してファイル転送を開始するのに使用されます。

daemon　このログインはシステム・デーモンであり、バックグラウンド処理を制御します。このログインは使用禁止にします。

lp　　このログインは、オブジェクト・ファイルとスプールされたデータ・ファイルを /var/spool/lp に所有しています。このシステムがプリント・サーバである場合を除き、ログインを禁止します。

使用しないログインのロック

使用しない不要なログインは、使用禁止（ロック）します。ただし、アカウントを削除してはなりません。将来、UID を再使用してしまう可能性があるためです。ユーザ ID 番号は、アカウントを使用したユーザに永久に割当てることになっています。UID 番号を再使用すると、その ID 番号を前に所有していた人のファイルに、新しいユーザがアクセスする可能性があります。そのファイルには、システムを損傷する「トロイの木馬」プログラムがある場合もあります。バックアップを作成した上でホーム・ディレクトリとファイルを削除することはできますが、/etc/passwd ファイルからはエントリを絶対に削除しないでください。

アカウントをロックするには 2 つの方法があります。1 つは、passwd コマンドに **-l** オプションを指定する方法です。たとえば、/etc/passwd 内のユーザ *jones* に対する現在のエントリを次のとおりとします。

```
jones:6.D/N3ZFGmq7U:3333:10:Jeremiah
Jones:/usr/people/jones:/bin/tcsh
```

/etc/passwd 内のユーザ *jones* に対するパスワード・フィールドを *LK* に変更し、このアカウントに対するすべてのログインを防ぐには、次のコマンドを入力します。

```
passwd -l jones
```

パスワード・フィールドのエントリは次のようになります。

```
jones:*LK*:3333:10:Jeremiah Jones:/usr/people/jones:/bin/tcsh
```

アカウントをロックする別の方法としては、パスワード・ファイルを直接編集します。パスワード・フィールドを変更し、パスワード暗号化プログラムによってパスワードの暗号化に使用されない文字列にします。passwd コマンドに **-l** オプションを指定すると、文字列 *LK* が使用されます。別の文字列を使用してアカウントをロックすることもできます。

たとえば、「LOCKED;」のような語句を使用すると、このアカウントを使用禁止にしたことが一目瞭然です。

```
ralph:LOCKED;:100:1:Ralph P. Cramden:/usr/people/ralph:
```

暗号化されたパスワードには、セミコロン（;）が使用されていないので、アカウントはロックされます。「LOCKED」という文字は、アカウントがロックされていることを注意するだけです。

パスワードを使用禁止にする一般的な方法としては、パスワード・フィールドにアスタリスク（*）を入れます。IRIX のデフォルトの `/etc/passwd` ファイルは、この方法に従って使用しないログインを使用禁止にします。ファイルのすべてのログインが、パスワードを与えられているか、または禁止されていることを確認します。

システム・ログイン・オプション

セキュリティ保持のために、以下のログイン・オプションを設定できます。

- *root* によるログインを、特定のデバイス（通常はシステム・コンソール）に制限する。
- ログインが終了するまでに、ログイン試行が失敗できる回数を指定する。
- ログイン・プロセスが使用禁止になった場合、それを再開できるまでの時間を指定する。
- ログインのログを保存するかどうかと、保存する場合の内容（すべてのログインまたは失敗したログインだけなど）を指定する。
- パスワードを持たないユーザに、ログイン直後にパスワードを作成させるかどうかを指定する。
- ログイン試行の失敗が指定の最大許容回数を超えたら、パスワードをロックすることによってアカウントを無効にする。ただし、*root* アカウントなどは除外される。
- ログインが成功した場合、ユーザが最後にログインした日付と時刻を表示するかどうかを指定する。

ログイン・オプションは、テキスト・ファイルの `/etc/default/login` に設定します。このファイルには 1 行に 1 つのオプションを指定します。以下に、各オプションについて説明します。詳細については、login(1) マン・ページを参照してください。

ログイン・オプションが大切である理由は、ログインの手順が権限のないアクセスを締出す有効な手段となるためです。たとえば、ログが記録される設定（デフォルト）になっている場合、`/var/adm/SYSLOG` に記録されたログイン失敗のパターンから、システムに侵入しようとした形跡があるかどうかを判断できます。

システムの安全性を確保するには、パスワードとアカウント名を推測する能率を低下させます。ここで説明するログイン・オプションは、失敗したログインが手間取るようにして、次々とパスワードを試すことができないようにしています。

ビジュアルなログイン手順 clogin(1) には、このような安全対策のオプションはありません。ログインのセキュリティ機能を利用するには、cloginをオフにし、標準のログイン手順であるgetty(1M)と login(1) を使用します。chkconfig で visuallogin と xdm の設定変数をオフにします。ビジュアル・ログインをオン／オフにする方法については、『IRIX Admin: System Configuration and Operation』と visuallogin(4) マン・ページを参照してください。`clogin` のユーザ・アイコンによるログインを禁止するように、`chkconfig` で *noiconlogin* 変数を設定することもできます。

root によるログインの制限

root によるログイン先を 1 つのデバイスに限定できます。*root* ユーザは、そのデバイスまたは `su` コマンドだけを使用できます（`su` コマンドを使用した場合は、その記録が `/var/adm/sulog` に残ります）。たとえば、`/etc/default/login` に次の行を追加すると、*root* によるログイン先はシステム・コンソールに限定されます。

```
CONSOLE=/dev/console
```

メモ：デバイス名として `/dev/syscon` または `/dev/systty` を指定してはいけません。これらのデバイスは `/dev/console` と同一ですが、ログイン・ソフトウェアによる取扱いが異なります。

ログイン試行の制限（MAXTRYS）

MAXTRYS は、不正なログイン試行を何回まで許すかを指定するものです。このパラメータを設定すると、権限のない者がシステムに侵入しようとした場合、その試行にかかる時間を引伸ばすことができます。このパラメータを設定すると、権限のない者はシステムに侵入しにくくなります。一般的な侵入の手口は、知っているアカウントのパスワードを推測することです。この手口は、侵入者が多くのアカウント名を知っており、次々と推測できる場合に確率が高くなります。同じ *tty* 回線で不正なログイン試行が何回か繰返された場合は、ログインの処理を遅延し、正しいパスワードを推測するのに手間取るようにします。

ログイン試行の最大回数を設定するには、/etc/default/login ファイルを編集します。ファイル内に次のような行を入れます。

```
MAXTRYS=4
```

この例では、最大試行回数を 4 回に設定します。このオプションを設定しない場合のデフォルトは 3 回です。

設定した最大試行回数を超えてログインが試みられると、login プログラムは特定の秒数（次の節の DISABLETIME 変数を参照）停止し、その回線で引続いてログインできないようにします。デフォルトの遅延時間（DISABLETIME）は 20 秒です。次の例では、3 回の再試行後にログインを禁止します。

```
login: guest
password:
Login incorrect
login: guest
password:
Login incorrect
login: guest
password:
Login incorrect
```

この時点から DISABLETIME で指定された時間が経過するまでは、ログイン・プロンプトの表示が凍結されます。

回線の使用禁止時間の設定（DISABLETIME）

このオプションは、*MAXTRYS* と組合わせて指定します。ログイン試行が何回か失敗した後に回線を使用禁止にする秒数を設定するには、/etc/default/login ファイルを編集し、次のような行を追加します。

```
DISABLETIME=30
```

この例では、回線は 30 秒間使用禁止になります。各自のシステムに応じて任意の値を選択できます。システムのデフォルトは 20 秒です。

ログイン試行の記録

成功したログインも失敗したログインも、デフォルトで `/var/adm/SYSLOG` ファイルに記録されます。`/etc/default/login` ファイルに次の行があると、成功したログインも含めてすべてのログイン試行を記録することがデフォルトとして設定されます。

```
SYSLOG=ALL
```

失敗したログインだけを記録する場合は、`login` ファイルで上の行を次の行に置き換えます。

```
SYSLOG=FAIL
```

失敗したログイン、特に同一のアカウント名によるログインが頻発する場合は、そのアカウントを通じてシステムに侵入しようとしている者がいると考えられます。この場合は、セキュリティ処理の監視が役立つことがあります。詳しくは、第 6 章「システム監査トレールの管理」を参照してください。

パスワードの強制

アカウントにパスワードを設定していないユーザにパスワードをすぐに設定させるには、`/etc/default/login` ファイルに次の行を追加します。

```
PASSREQ
```

または、パスワードが設定されていないユーザのログインを禁止するには、次の行を入力します。

```
MANDPASS=YES
```

アカウントの無効化（LOCKOUT）

LOCKOUT オプションでは、ユーザ・アカウントがロックされるまで試行できるログイン回数を指定します。指定回数を超えて失敗すると、そのアカウントは `passwd -l` *username* の形式でロックされます。アカウントのロックの詳細については、99 ページの「使用しないログインのロック」を参照してください。

LOCKOUT オプションを設定する場合は、*root* アカウントだけは LOCKOUTEXEMPT オプションで除外することをお薦めします。除外することにより、サービスが提供されなくなることを防ぐことができます。login(1) マン・ページを参照してください。

最後のログイン時刻の表示

アカウントの無断使用をユーザ自身が発見できると、システム・セキュリティの保全に役立ちます。デフォルトによって、最新のログイン日時と、ユーザがログインした端末回線名（*tty* 名）またはリモート・ホスト名が、ログイン時に表示されます。このログイン情報は、ユーザ・アカウント別に同名のファイルとして、`/var/adm/lastlog` ディレクトリに記録されます。

ホーム・ディレクトリに `.hushlogin` ファイルを挿入すると、最後のログイン情報が表示されなくなります。それは、できるかぎり避けてください。ユーザには、ログインするごとにログイン情報を参照して異常の有無を確認することを、定期的に通知します。

set-UID と set-GID のパーミッションについて

ユーザ ID の設定（set-UID）とグループ ID の設定（set-GID）のパーミッションは慎重に使用します。どちらかのパーミッションが設定された実行可能ファイルを実行するユーザには、実行可能ファイルの所有者としてのパーミッションがシステムから与えられます。この種のパーミッションを実行可能ファイルに追加するには、chmod(1) コマンドを使用します。

set-UID と set-GID の各プログラムには正規の用途がありますが、その危険性を考慮し、システムではできるかぎり実行しません。すべてのユーザが書込めるディレクトリ（`/tmp`、`/usr/tmp.O`、`/var/tmp`、`/usr/spool/uucppublic` など）に格納されているプログラムで、一般的なシステム・ファイル（`vi` や `rm` など）と同名のものには注意します。*root* アカウントの環境変数 PATH が（多くのユーザのデフォルト・パスのように）カレント・ディレクトリを含まない理由の 1 つは、booby-trap などの危険なプログラムを *root* が誤って実行するのを避けるためです。

ユーザが `-rwsrwxrwx` パーミッションが設定されたファイルに別のプログラムをコピーすると、システム・セキュリティを損なうおそれがあります。極端な例を挙げると、他のユーザが書込めるパーミッションが設定された `su` コマンドには、だれでもシェルをコピーし、`su` のパスワード不要バージョンを入手できます。

システムで set-UID パーミッションが設定されたファイルを識別するコマンド例について、以下に説明します。set-UID ビットと set-GID ビットの詳細については、chmod(1) と chmod(2) のマン・ページを参照してください。

root が所有する set-UID ファイルの検査

次のコマンド行を入力すると、*root* が専有するすべての set-UID ファイルがリスト表示されます。

```
find / -user root -perm -4000 -print
```

結果が画面に表示されます。マウントされているすべてのディレクトリも含めて、すべてのパスが／（ルート）からチェックされます。かなりの数のファイルがリスト表示されます。その中に異常なファイル名がないかを検査するのは、システム管理者の責任です。1 つの検査方法としては、インストール直後にこのプログラムの実行結果をファイルに出力し、その出力を後の出力と比較できます。検査の結果、異常なファイルが見つかったときは、すぐに調査します。

疑わしいファイルは次のように表示されます。

```
-r-sr-xr-x 1 root  bin   38836 8月 10日 16時16分 /usr/bin/at
-r-sr-xr-x 1 root  bin   19812 8月 10日 16時16分 /usr/bin/crontab
-r-sr-xr-x 1 root  bin   27748 8月 10日 16時16分 /usr/bin/shl
---s--x--x 1 root  sys   46040 8月 10日 15時18分 /usr/bin/ct
-r-sr-sr-x 1 root  bin   33208 8月 10日 15時55分 /usr/lib/lpadmin
-r-sr-sr-x 1 root  bin   38696 8月 10日 15時55分 /usr/lib/lpsched
---s--x--- 1 root  user  45376 8月 18日 15時11分 /usr/jbond/bin/sh
-r-sr-xr-x 1 root  sys   11416 8月 11日 01時26分 /bin/mkdir
-r-sr-xr-x 1 root  sys   11804 8月 11日 01時26分 /bin/rmdir
-r-sr-xr-x 1 root  bin   12524 8月 11日 01時27分 /bin/df
-rwsr-xr-x 1 root  sys   21780 8月 11日 01時27分 /bin/newgrp
-r-sr-sr-x 1 root  sys   23000 8月 11日 01時27分 /bin/passwd
-r-sr-xr-x 1 root  sys   23824 8月 11日 01時27分 /bin/su
```

この例では、ユーザ *jbond* が `/bin/sh` の個人用コピーを持っており、それが *root* に set-UID として入っています。つまり、グループ `user` のだれもが `/usr/jbond/bin/sh` を実行して特権ユーザになることができます。

root ファイルシステムでの set-UID ファイルの検査

次のコマンド行は、EFS で *root* が所有するファイルだけでなく、*root* ファイルシステムで set-UID が設定されたすべてのファイルをリスト表示します。

```
ncheck -s /dev/root | xargs ls -ld | cut -f2 | grep -v ~/dev/
ls -l ‘/etc/ncheck -s /dev/root | cut -f2 | grep -v dev‘
```

ncheck(1M) コマンドは、マウントまたはアンマウントされたファイルシステムで使用できます。ncheck を使用できるのは特権ユーザだけです。ncheck -s コマンドの通常の出力には、特殊ファイルが含まれます。ここでは、grep コマンドが出力からデバイス・ファイルを削除します。このフィルタリングが適用できるのは、*root* ファイルシステムだけです。変更された ncheck の出力は ls コマンドの引数として使用されます。ファイルシステムがマウントされていないと、ls コマンドは失敗します。次の出力例に異常はありません。

```
-r-sr-xr-x  1 root   bin    12524   8月 11日 01時27分 /bin/df
-rwxr-sr-x  1 root   sys    32272   8月 10日 15時53分 /bin/ipcs
-r-xr-sr-x  2 bin    mail   32852   8月 11日 01時28分 /bin/mail
-r-sr-xr-x  1 root   sys    11416   8月 11日 01時26分 /bin/mkdir
-rwsr-xr-x  1 root   sys    21780   8月 11日 01時27分 /bin/newgrp
-r-sr-sr-x  1 root   sys    23000   8月 11日 01時27分 /bin/passwd
-r-xr-sr-x  1 bin    sys    27964   8月 11日 01時28分 /bin/ps
-r-xr-sr-x  2 bin    mail   32852   8月 11日 01時28分 /bin/rmail
-r-sr-xr-x  1 root   sys    11804   8月 11日 01時26分 /bin/rmdir
-r-sr-xr-x  1 root   sys    23824   8月 11日 01時27分 /bin/su
-r-xr-sr-x  1 bin    sys    21212   8月 10日 16時08分 /etc/whodo
```

XFS に対しては find コマンドを使います。

```
find / -perm -4000 -print
```

root 以外のファイルシステムでの set-UID ファイルの検査

次の例では、ncheck コマンドを使用し、/home パーティション（この場合は /dev/dsk/dks0d2s7）で set-UID のパーミッションが設定されているファイルを検査しています。

```
/etc/ncheck -s /dev/dsk/dks0d2s7 | cut -f2
```

次の出力例で、/home は ncheck 出力に含まれていませんが、ファイルには i ノード番号とともに /home から始まる完全なパス名が表示されます。

```
/dev/dsk/dks0d2s7:
3971    /jbond/bin/sh
```

この ncheck 出力では、/home/jbond/bin/sh プログラムの検査が必要です。このプログラムは、システム・ディレクトリにありません。これはユーザのホーム・ディレクトリにあるコマンド・シェルです。通常、ユーザは set-UID ファイルを所有していません。

汎用のファイルとディレクトリのパーミッションについて

ファイルとディレクトリのパーミッションのビットを設定または変更する場合は慎重に行います。書込みを禁止するのが最も安全な設定です。書込み禁止にできない場合は、ファイルまたはディレクトリの所有者と、必要に応じて所有者とグループだけに書込み権を与えます。

出荷時の IRIX に含まれるファイルとディレクトリのうち、すべてのユーザが読み書きできるものを以下にリストアップします。サイトの必要に応じて、各ファイルのパーミッションをより制限することもできます。/tmp（IRIX のデフォルト）などのディレクトリにスティッキー・ビットを設定し、ファイルの削除と名前の変更を禁止する方法については、chmod(1) マン・ページを参照してください。

- /tmp
- /usr/demos/.xsession
- /usr/Insight/tmp
- /usr/Insight/tmp/ebtpriv
- /usr/Insight/tmp/ebtpub
- /usr/Insight/tmp/install.insight.log

- /usr/lib/emacs/maclib
- /usr/lib/showcase/fonts
- /usr/lib/showcase/images
- /usr/lib/showcase/models
- /usr/lib/showcase/templates
- /usr/tmp.O
- /var/spool/locks
- /var/spool/uucppublic
- /var/tmp

注意：従来公開されているディレクトリである /tmp、/usr/tmp.O、var/tmp（/usr/tmp にリンクされています）などのパーミッションを制限すると、多くのプログラム、アプリケーション、システム・ユーティリティの動作に支障が起こります。これらのディレクトリにテンポラリ・ファイルを書込めなくなるためです。

---

rfindd デーモンを起動したままにしないでください。ファイル、ディレクトリ、パーミッションのリストに外部からアクセスできるので問題が生じます。詳細については、rfindd(1M) マン・ページを参照してください。

パスワードなしで出荷されるアカウント

次に示すデフォルトの /etc/passwd ファイルのアカウントは、パスワードなしで出荷されます。少なくてもルート・アカウントにはすぐにパスワードを作成してください。

- root—Superuser
- lp—Print Spooler Owner
- nuucp—Remote UUCP User
- EZsetup—System Setup

- demos—Demonstration User
- OutOfBox—Out of Box Experience
- guest—Guest Account
- 4Dgifts—4Dgifts Account

注意：従来より公開されているアカウントである `lp` などにパスワードを設定すると、それに関連したアプリケーションの動作または操作に支障が生じる場合があります。

セキュリティ関連のファイルとコマンド・リファレンス

ここでは、セキュリティを確立して管理するための IRIX のファイルとコマンドをリストとして 2 つの表に示します。表 4-2 はセキュリティに関する IRIX ファイルのリストです。表 4-3 はセキュリティに関するコマンドのリストです。

表 4-2 IRIX のセキュリティ関連のファイル

ファイル	用途	リファレンス
`/etc/default/login`	ログイン動作の管理	login(1)
`/etc/default/su`	`su` コマンドのデフォルトの定義	su(1M)
`/etc/passwd`	パスワードとアカウント情報の格納	passwd(1)、passwd(4)
`/etc/shadow`	パスワード情報の隠蔽	shadow(4)、pwconv(1M)
`/var/adm/sulog`	`su` コマンドの利用状況の記録	su(1M)
`/var/adm/SYSLOG`	システム・メッセージの記録	syslogd(1M)

表 4-3 IRIX セキュリティ・コマンド

コマンド例	用途	リファレンス
arp -a	現在の ARP エントリの表示	arp(1M)、arp(7P)
crypt *password*	入出力のエンコード／デコード	crypt(1)
last	ユーザと端末の最後のログインの表示	last(1)
ncheck	i- 番号からのパス名の作成	ncheck(1M)
passwd	パスワードの変更	passwd(1)、passwd(4)
ps -elf	現在実行しているすべてのプロセスの詳細情報の表示	ps(1)
pwck	`/etc/passwd` ファイルの不整合性の表示	pwck(1M)、passwd(4)
sar	システムの活動報告	sar(1)、sadc(1M)
satd	システム監査トレールの確実な保存	satd(1M) と 172 ページの「監査ファイルの格納」
vi -x	暗号化ファイルの編集	vi(1)、crypt(1)
w	現在作業中のログイン・ユーザの表示	w(1)
who	ログイン・ユーザ、tty、ログイン時刻の表示	who(1)

セキュリティの強化機能

IRIX6.5 およびそれ以降のリリースでは、アクセス・コントロール・リスト（ACL: Access Control Lists）と最小限の特権機能というセキュリティ強化機能が Commercial Security Pak に含まれています。

アクセス・コントロール・リスト（ACL: Access Control Lists）

ACL は標準のファイル・パーミッションと同じ働きをしますが、標準のファイル・パーミッションよりもアクセス・コントロールの対象となるユーザをより細かく指定できます。ACL を使うと、ユーザごとにファイル・パーミッションを設定することが可能になります。

各システム・ファイルまたはディレクトリには、アクセス権を任意に決定するためのアクセス・コントロール・リストが含まれています。この ACL は、ファイルまたはディレクトリ用アクセス ACL と呼ばれます。また、ディレクトリには、ディレクトリ内に作成されたファイルやサブディレクトリへの最初のアクセス権を決定する ACL が含まれている場合もあります。この ACL はデフォルト ACL と呼ばれます。ディレクトリ内のファイルにアクセスしたいユーザは、上記の 2 つの ACL に指定されると同時に、確実にアクセスできるようにするには、IRIX ファイル・パーミッションでアクセスを許可されている必要があります。特定のファイルに対するアクセス ACL を作成していない場合は、デフォルト ACL が上記両方の ACL として機能します。

この章の以降の説明では、ディレクトリはファイルと同様に扱われます。このため、ファイルという言葉は、ディレクトリとファイルの両方を指します。

ACL は標準のファイル・パーミッション同様、ファイルまたはディレクトリの属性として保存されます。ファイルの ACL を表示するには、次のように ls(1) コマンドに **-D** オプションを指定して入力します。

```
ls -D /usr/people/ernie/testfile
```

上のコマンドを実行すると、次のような行が表示されます。

```
testfile [user::rwx ,user:332:r--,user:ernie:rw-]
```

この例では、一番上の行の最初のエントリの所有者のパーミッション全部が表示され、2 番目のエントリのユーザ ID 332 に読込み権を設定し、ユーザ・アカウント ernie に対しては、読込み／書込み権を設定しています。ACL の詳しい入力形式については、「ACL の長いテキスト形式」を参照してください。

ACL を設定または変更するには、chacl(1) コマンドを使用します。

chacl *acl_entry*[,*acl_entry*]...

ACL はカンマで区切られた複数の ACL エントリで構成されます。1 つの ACL エントリで、1 人のユーザまたは複数ユーザのグループに対して、関連ファイルへのアクセス権を指定します。ACL 内部のエントリの保存順序は、評価順序とは無関係です。オブジェクトから ACL を読むには、そのファイルに読込み権が必要です。ACL を作成または変更するには、プロセスがこのファイルを所有している必要があります。

ACL には長いテキストと短いテキストの 2 つの形式があります。長いテキスト形式は、すべてのエントリに対してすべての指定項目を入力するためのもので、最初に定義します。この後、長いテキスト形式に対応する短いテキスト形式を定義します。

ACL の長いテキスト形式

長いテキスト形式は、ACL の入力または出力に使用されるもので、次のように設定します。

acl_entry[,*acl_entry*]...

1 行に複数のエントリを入力してもかまいませんが、1 行につきエントリを 1 つにした方が読みやすくなります。

各エントリは 1 つの ACL ステートメントと、必要とされるコロンで区切られた 3 つのフィールド、およびオプションのコメントで構成されます。

```
entry tag type:entry qualifier:discretionary access
permissions#comment
```

コメントをエントリに含めることもできます。行をコメントで開始すると、行全体がコメントとして解釈されてしまいます。最初のフィールドには、必ず ACL エントリ・タグ・タイプを入力してください。

第 1 のフィールドには、次の ACL エントリ・タグ・タイプのキーワードを必ず指定するようにします。

user ファイルの所有者または指定のユーザ・アカウントにアクセスを許可します。

group ファイルを所有しているユーザ・グループまたは指定のユーザ・グループにアクセスを許可します。

other ユーザ、グループ、または実行時に定義された ACL エントリと一致しないプロセスにアクセスを許可します。

mask ファイルの所有者の *user* エントリまたは *other* エントリを除くすべての ACL によって許可される最大のアクセス。

第 2 のフィールドには ACL のエントリ修飾子（以後、単に修飾子と記述します）が含まれます。次の修飾子がデフォルトで定義されています。

uid ユーザ・アカウント名またはユーザ ID 番号

gid ユーザ・グループ名またはグループ ID 番号

empty *uid* または *gid* 情報は ACL エントリに適用されません。エントリは、ファイルの所有者にだけ適用されます。空の修飾子は、空の文字列または空白スペースで表されます。

第 3 のフィールドには、最初のフィールドに指定されたユーザまたはグループに適用される任意のアクセス権を指定します。このフィールドには、アクセス権の種類を表す次の文字のうちの 1 つをそのまま入力してください。

r 読込み権

w 書込み権

x 実行権

アクセス権を何も許可しない場合には、上の文字を、ダッシュ（-）に置換えます。

ユーザ・エントリに空の修飾子が付いている場合は、ファイルの所有者にだけアクセス権が与えられます。ユーザ・エントリに *uid* 修飾子が付いている場合は、*uid* の値に一致するユーザ名に対してだけアクセス権が与えられます。*uid* の値がユーザ名と一致しない場合、ACL エントリは *uid* の値に一致するユーザ ID に対してだけアクセス権を与えます。

グループ・エントリに空の修飾子が付いている場合は、デフォルトのファイル所有者のユーザ・グループに対してだけアクセス権が与えられます。グループ・エントリに *gid* 修飾子が付いている場合は、*gid* の値に一致するグループ名に対してだけアクセス権が与えられます。*gid* の値がグループ名と一致しない場合、ACL エントリは *gid* の値に一致するグループ ID に対してだけアクセス権を付与します。*umask* およびその他のエントリには空の修飾子が含まれます。ACL エントリでは、コメントをシャープ記号（#）で開始します。コメントは行の最初、必要とされるフィールドの後、またはカスタム定義のコロンで区切られたフィールドの後から始めることができます。行の最後がコメントの最後になります。

ACL エントリに *umask* エントリには含まれていないパーミッションが含まれている場合は、上述のように、そのエントリの出力形式に続いてシャープ記号（#）と「effective:」という文字列、およびその ACL エントリに対して有効なファイルへのアクセス権が表示される必要があります。

空白スペースは次のような入力が許されます。ただし、必要ではありません。

- 行の最初
- セパレータであるコロン（:）のすぐ後ろ
- コメントの開始文字である最初のシャープ記号（#）のすぐ前
- コメントの開始文字である最初のシャープ記号（#）の後ならどこでも

コメントは関連するオブジェクトの任意のアクセス・チェックには何の影響も与えません。

ファイル用 ACL の長いテキスト形式の例を下に示します。

```
user::rwx,user:332:r--,user:ernie:rw-
```

この例では、行の最初に指定されている所有者にすべてのアクセス権を設定し、2 番目に指定されているユーザ ID332 には読込み権を、ユーザ・アカウント enrie には読込み／書込み権を設定しています。

コメント付きの入力例を下に示します。

```
group:10:rw-# User Group 10 has read/write access
other::---# No one else has any permission
mask::rw-# The maximum permission except for the owner is read/write
```

ACLの短いテキスト形式

短いテキスト形式は chacl(1) コマンドが ACL の入力に使用するもので、次のように設定します。

```
acl_entry[,acl_entry]...
```

1 行に複数のエントリを入力してもかまいませんが、1 行につきエントリを 1 つにした方が読みやすくなります。

各コマンド行は 1 つの ACL で構成されます。ただし、最初のフィールドには、ACL のエントリ・タグ・タイプを表すキーワードを省略名ではない完全な形式または 1 文字の省略名の形式で入力します。

ユーザは *u*、グループは *g* という 1 文字の省略名で表します。その他（other）は *o*、マスク（mask）は *m* で表します。

ACL の短いテキスト形式の 2 番目のフィールドには例外はありません。任意のアクセス権を 3 番目のフィールドに絶対記号または相対記号の形式で入力します。

相対記号の前には、アクセスの追加を表すプラス記号（+）または削除を表すキャレット（^）を指定します。相対記号文字列は、少なくとも 1 つの文字を含む必要があります。

記号文字列には、次の文字をそれぞれ 1 つづつ指定できます。指定順序は無関係です。

- r
- w
- x

たとえば、短いテキスト形式は次のようになります。

```
u: :rwx # The file owner has complete access
u:332:+r # User Acct 332 has read access only
g:10:rw- # User Group 10 has read/write access
u:653:^w # User Acct 653 (who is in group 10) has read access only
o::--- # No one else has any permission
m::rw- # The maximum permission except for the owner is read/write
```

ls -D と chacl の使用

`ls -D` コマンドの出力を `chacl` への入力として使用することができます。この方法は、デフォルトでこのような ACL を使用していないディレクトリ内のファイルに複数の要素で構成されるカスタムの ACL をコピーするのに便利です。次の例について考えてみましょう。

`ls -dD` *testdir*

このコマンドを実行すると、次の出力が表示されます。

```
testdir [u::rwx,g::r-x,o::--x/u::rwx,g::r-x,o::---]
```

次のコマンドで新しいディレクトリを作成します。場所は問いません。

mkdir *newdir*

次のコマンドを使って ACL を編集、コピーします。すべて 1 行に入力してください。

chacl -b `ls -dD** *testdir* **| cut -d"[" -f2 | cut -d"/" -f1` `ls -dD** *testdir* **| cut -d"[" -f2 |  cut -d"/" -f2 | cut -d"]" -f1` *newdir*

testdir の ACL が newdir に複製されます。上のコマンド行では cut コマンドが使用されていることに注目してください。cut の詳しい使い方については、cut(1) マン・ページを参照してください。このコマンドを実行して newdir の内容を表示すると、testdir の ACL が複製されていることがわかります。

ls -dD *newdir*

```
newdir [u::rwx,g::r-x,o::--x/u::rwx,g::r-x,o::---]
```

ウィンドウ・マネージャのカット・アンド・ペースト機能を使って ACL のエントリを ls -D から chacl にコピーすることもできます。

最小限の特権機能

最小限の特権機能とは、以前は特権ユーザのために予約されていた処理を特定のアカウントが実行できるよう割当てる特権のことです。最小限の特権の原則を守るため、特権ユーザ・アカウントの機能を細分化してから、それぞれのアカウントに割当てます。対応する機能がシステム上の損傷を受けやすい実行ファイルとプログラムに配置されます。プログラムを実行するには、アカウント機能と実行機能との間に互換性があることが必要です。技術的な詳細については、capabilities(4) マン・ページを参照してください。

最小限の特権機能の基本的な目的は、特権ユーザやその他の特権付きアカウントがなくても、標準のログイン・アカウントで管理作業ができるようにすることです。この機能は、どのユーザ・アカウントにも許可することができ、対応する特権機能の必要条件は、このユーザ・アカウントの所有者にとって使用する正当な理由があるシステム･オブジェクトに対してのみ適用されます。また、最小特権の信頼できる原則として、処理の実行に必要な最小限の特権を使用するということが挙げられます。最小限の特権では、各種処理の特権を付与するユーザの数、および処理の実行に必要な 1 つのプログラムだけ、またはプログラム内の一部だけに特権を制限します。

通常、一般ユーザに特権機能を与えないでください。一般ユーザに特権機能を付与する場合は、そのユーザにとって本当に必要な特権機能だけを与えるという、最小限の特権の原則を守ってください。

特権機能を使うと、プロセスに適用される特権をより詳細に管理することができます。プロセスでは、特権付きシステム・コールを実行するための特定の機能は付与されますが、setuid *root* プログラムのようなシステムのプロテクション・スキームを無効にするような処理は許可されません。IRIX の機能方式は、POSIX 1003.le Draft 15 仕様の Draft 15 に準拠しています。

/etc/capability ファイル

`/etc/capability` は、ユーザ・アカウントに付与された機能のデータベースです。次に例を示します。

```
root:all+eip:all+eip
auditor:CAP_AUDIT_WRITE,CAP_AUDIT_CONTROL,CAP_KILL+eip
ernie:all=:CAP_FOWNER,CAP_SETFCAP+eip
casey:all=:all+eip # We trust Casey.
jeff:all+eip CAP_NETWORK_MGT-eip:all+eip
fred:all=:all=
```

各エントリは、次のように、コロンで区切られた 3 つのフィールドで構成されます。

```
username : default_capability : maximum_capability
```

- username は、ユーザのログイン名のことです。ログイン名は、必ず `/etc/passwd` ファイルに記述されているとおりに入力してください。
- デフォルトの機能セットは、ログイン時にユーザのシェル・プロセスに適用されます。ユーザは、ログイン時に追加の機能を要求することができます。このエントリにない機能をログイン時に要求すると、ログインは失敗に終わります。
- maximum capability のフィールドには、ユーザのプロセスが要求し、受取る可能性のあるすべての機能が記述されます。

default と maximum capability のフィールドは次のような形式で入力します。

```
capname,capname operator flags
```

`capname` には、「本リリースの特権機能」の機能リストに記載されているものを指定します。

operator には次のいずれかを指定します。

+ この機能を次のセットに追加します。

- この機能を次のセットに対して削除します。

= このプロセスの間、この機能を次のセットに対して取消します。

機能セットを表す *flags* には、次のうちの 1 つまたはそれ以上を指定します。

i 継承可能な機能セット：子プロセスに渡すことのできる機能

e 有効な機能セット：現在アクティブな機能

p 許可されている機能セット：プロセスに対する最大の機能セット

各フィールドには、複数の節が含まれます。それぞれの節は、スペースで区切られた複数の機能と operator/set 文で構成されます。# で始まる行は、その行の最後までがコメントとして解釈され、無視されます。節は読取り順に（左から右へと）解釈されます。このため、エントリの最後に指定された処理が重要になります。

`/etc/capability` ファイルの例を再度下に示します。

```
root:all+eip:all+eip
auditor:CAP_AUDIT_WRITE,CAP_AUDIT_CONTROL,CAP_KILL+eip:
ernie:all=:CAP_FOWNER,CAP_SETFCAP+eip
casey:all=:all+eip # We trust Casey.
jeff:all+eip CAP_NETWORK_MGT-eip:all+eip
fred:all=:all=
```

このファイルでは、次の点に注目してください。

- *root* アカウントには、デフォルトですべてのフラグがセットされ、すべての特権機能が追加されています。
- auditor（監査人）のアカウントには、システムの監査トレールを管理し、プロセスを中断するのに必要な特権機能だけが付与されています。
- ernie のアカウントには、デフォルトでは特権機能が与えれていませんが、必要なときは、他の人のファイルで作業する機能と、実行ファイルに対して機能要求を設定できるようになっています。
- casey のアカウントには、デフォルトでは特権機能与えられていませんが、必要なときにすべての機能を取得できるようになっています。また、その結果に対するコメントも追加されています。
- jeff のアカウントには、デフォルトですべての特権機能が与えれれていますが、続く節でネットワーク管理用特権機能が削除されています。ただし、必要なときはすべての機能を要求できるようになっています。
- fred のアカウントには、特権機能が何も与えられていなく、要求もできません。

実行中のすべてのプロセスには *effective*、*permitted*、*inheritable* という 3 つの機能セットが用意されています。

- effective セットは、プロセスに対するアクセス制御の判断に使用します。
- inheritable セットは、ユーザが実行ファイルを呼出す exec(2) プロセス実行中に新しい機能セットを計算するのに使用します。
- permitted セットは、プロセスに付与できる最大の機能です。

各実行ファイルにも、同じ 3 つの機能セットがあります。これらのセットは、ユーザがプログラムを呼出すときに作成される新しいプロセスに付与される機能セットを最終的に決定するものです。

- 新しい effective セットは、親プロセスの permitted セットと実行ファイルの effective セットの共通部分です。つまり、実行ファイルの effective セットに、呼出し側プロセスの permitted セットにあってプロセスの effective セットにはない機能が含まれている場合、その機能は子プロセスの effective セットには追加されません。

- 新しい inheritable セットは、呼出し側プロセスと実行ファイルの継承可能な特権機能の共通部分です。つまり、実行ファイルから継承され、親プロセスによって継承可能であると指定された機能だけが新しいプロセスに継承されます。
- 新しい permitted セットは、実行ファイルの permitted セットに、新しい inheritable セットと親プロセスの permitted セットの共通部分を合わせたものです。つまり、それぞれの機能が親プロセスと実行ファイルの両方によって継承可能な限り、ファイルと親プロセスの両方のすべての permitted 機能が許可されます。

親プロセスの effective 機能セットが新しいセットに影響を与えることはありません。また、実行ファイルの継承可能セットが新しいプロセスに付与できる機能の上限を設定します。

本リリースの特権機能

本リリースには、次の特権機能が提供されています。

ALL
: すべての機能を表示します。

CAP_ACCT_MGT
: acct(2) のようなアカウンティング設定システム・コールを発行できる特権です。

CAP_AUDIT_CONTROL
: satread(2) および satwrite(2) システム・コールのような、システム監査トレールを管理できる特権です。

CAP_AUDIT_WRITE
: satwrite(2) システム・コールのような、システム監査トレールへの書込みが可能な特権です。

CAP_CHOWN
: ファイルの所有権を変更する _POSIX-CHOWN_RESTRICTED 用に構成されているシステムのファイルで、プロセスによって所有されていないファイルの所有者を変更できる特権です。

CAP_CHROOT
: chroot(2) システム・コールを実行できる特権です。

CAP_DAC_EXECUTE
: パーミッションまたは ACL で禁止されているファイルを実行できる特権です。

CAP_DAC_READ_SEARCH

パーミッションまたは ACL で禁止されているファイルを読込んだり、ディレクトリを検索できる特権です。

CAP_DAC_WRITE

パーミッションまたは ACL で禁止されているファイルに書込んだり、ディレクトリを更新できる特権です。

CAP_DEVICE_MGT

次のような制限されているデバイスの管理コールや ioctl を発行できる特権です。

- XLV 論理ボリューム・インタフェース — 論理ボリュームと、その各種パラメータを定義します。
- `syssgi`(SGI_FS_INUMBERS) — XFS ファイルシステム上のすべての有効な内部ハンドル（inode 番号）を返します。
- `syssgi`(SGI_FS_BULKSTAT) — ファイルシステム全体に「in bulk」のファイルの状態（struct stat）を返します。
- `fcntl`(F_FSSETDM) — ファイルの DMA パラメータを設定します。
- DMI インタフェース — サードパーティ製の記憶管理製品が使用します。
- ioctl に TCSETA、TCSETAF または TCSETAW コントロール・パラメータを指定して使用し、CD_MODEM と表示されているポートに CLOCAL フラグを設定します。
- ioctle を使用してディスク上で特権処理を実行します。
- syssgi(2) を使ってハードウェアのパフォーマンス・モニタにアクセスします。
- ロード可能なデバイス・ドライバ、ストリーム・モジュール、およびファイルシステムをロード、アンロード、登録またはその登録を取消します (mload(4))。
- vhangup(2) を使ってデバイスへのアクセスを取消します。
- syssgi(2) を使ってメモリ・エラーの処理を制御します。
- ユーザ・レベルの割込みハンドラを確立します (uli(3))。
- ファイルシステムの属性を取得、設定します。

CAP_FOWNER

プロセスがファイルを所有しているかのようにファイルを操作する特権です。この特権機能を使うと、プロセスのユーザ ID がファイル所有者の ID と一致しなければならないという必要条件を無効にすることができます。ただし、CAP_FSETID が適用される場合は例外です。通常、この特権機能が有効の場合、プロセスは、ファイルの所有者がファイルに対して実行できるすべての機能を実行できます。

CAP_FSETID

所有者でなくてもファイルの setuid または setgid ビットを設定できる特権です。また、setuid または setgid ビットを設定してファイルの所有者を変更することもできる特権です。この特権機能を使うと、次の制限を無効にできます。

- ファイルの set-user-ID (S_ISUID) および set-group-ID (S_ISGID) ビットを設定するには、呼出し側プロセスのユーザ ID がファイルの所有者と一致していること。
- ファイルの set-group-ID ビットを設定するには、呼出し側プロセスのグループ ID または付属するグループ ID の 1 つがファイルのグループ ID に一致していること。
- `chown` が成功して復帰したときは、ファイル・モードの set-user-ID と set-group-ID ビットがクリアされること。

CAP_KILL

送信者が所有していない別のプロセスに対して kill(1) を送信する特権です。また、プロセスに対してプロセス同期コール（procblk）を使用することもできる特権です。

CAP_MEMORY_MGT

主としてメモリのロックなど、制限されたメモリ管理用コールを発行する特権です。この特権機能を使うと、プロセスはシステム・メモリの管理方針を操作できないという制限を無効にすることができます。この特権機能によって可能になる操作は次のとおりです。

- shmctl(2) インタフェースを介した共有メモリ・セグメントのロックまたはアンロック
- メモリ内のプロセスの他セグメントのロックまたはアンロック (mpin(2)、plock(2))
- `syssgi`(SGI_MINRSS) システム・コールの使用
- ページの物理アドレスの検索

CAP_MKNOD

CAP_DEVICE_MGT のエリアスです。

CAP_MOUNT_MGT

mount(2) および umount(2) コールを発行する特権です。

CAP_NETWORK_MGT

ネットワーク・インタフェース・アドレスやネットワーク・インタフェース・デバイスの管理を設定ための制限されているネットワーキング・コールを発行できる特権です。この特権機能は、システムのネットワーク設定を変更するときに必要になります。この特権機能によって可能になる機能は次のとおりです。

- ネットワーク・デバイス・インタフェースへのファームウェアのダウンロードと起動。
- メディア・アクセス制御（MAC: Media Access Control）アドレスの設定。たとえば、インタフェースのイーサネット・アドレスの設定など。
- ネットワーク・デバイスからのデバイス管理情報の検索。
- FDDI SMT 情報の設定、制御、および検査。
- ARP 方式の制御。
- ネットワーク・インタフェースの IP アドレス、パラメータ、およびフラグの制御。
- IP フィルタの設定。
- lockd(1M) 専用インタフェースの使用。
- NFS サービス・デーモン専用インタフェースの使用

CAP_NVRAM_MGT

CAP_SYSINFO_MGT のエリアスです。

CAP_PRIV_PORT

特権 TCP ポート上でソケットを開くことができる特権です。

CAP_PROC_MGT

制限されているプロセス管理コールを発行できる特権です。このこの特権機能は、他プロセスの属性変更時の制限を無効にし、特権プロセス処理を実行するときに必要になります。この特権機能によって可能になる処理は次のとおりです。

- setuid/setgid 実行権のトレース。
- システムまたはプロセス単位の制限よりも大きいリソース極限値の設定。
- カーネル・スレッド機能の使用。
- UID または GID 変更時に共有グループ内で実際の UID/GID を更新。この機能を使用しないと、有効な ID が更新されます。
- prctl(2)を使って共有グループ内のプロセスごとのスタックのサイズを設定。
- プロセスを、レジデントの prctl(2) にします。

CAP_QUOTA_MGT

制限されているディスク割当て管理コールを発行できる特権です。

CAP_SCHED_MGT

リアルタイム・スケジューラ・インタフェースなど、制限されているスケジューラ・コールを発行できる特権です。この特権機能は、システムのプロセス・スケジューラの操作に必要になります。この特権機能によって可能になる操作は次のとおりです。

- プロセスの優先順位を高い値に変更。
- 別のプロセスのプロセス優先順位を変更。
- プロセスに優先順位が下がらないことを設定。
- プロセスのリアルタイムな優先順位を設定。
- プロセスのタイム・スライスの値を設定。
- プロセスからプロセスへの結合を制御。
- プロセスの作業セットの優先順位を設定。
- フレーム速度スケジューリング機能を使用。
- プロセスのリソース極限値を変更。
- バッファ・キャッシュのフラッシュ・ルーチンの動作レートを制御。

CAP_SETFCAP

CAP_SETFPRIV のエリアスです。

CAP_SETFPRIV

ファイルの機能セットを変更できる特権です。

CAP_SETGID

プロセスの実際の、有効な保存された GID を変更できる特権です。また、プロセスのグループ ID を変更することもできる特権です。

CAP_SETPCAP

CAP_SETPPRIV のエリアスです。

CAP_SETPPRIV

プロセスの機能セットを変更できる特権です。

CAP_SETUID

プロセスの実際の、有効な保存された UID を変更できる特権です。

CAP_SHUTDOWN

システムをシャットダウンまたは再起動できる特権です。この特権機能では、次の処理を実行できる uadmin(2) システム・コールの使用が必要です。

- システムを停止。
- システムを再起動。
- 破損したファイルシステムの自動修復後、ルートの再マウントを強制的に実行。
- すべてのプロセスに正規の手順で終了するよう通知。
- システムの電源を切る（すべてのシステムでサポートされているわけではありません）。

CAP_STREAMS_MGT

制限されている STREAMS コールと操作が発行できる特権です。

CAP_SWAP_MGT

swap(2) コールを発行できる特権です。

CAP_SYSINFO_MGT

ホスト名、NVRAM の値などのシステム情報を設定できる特権です。この特権機能は、システムの識別情報を操作するのに必要です。この特権機能の対象となる情報は次のとおりです。

- FDDI インタフェースなどのアダプタ上の NVRAM の内容（通常はアドレスまたは名前）。
- ホスト ID、ノード名、およびドメイン名。
- VM 不良トレースの起動。
- UID 0：の取扱いの制御。

  典型的な特権ユーザ・モード。UID 0 にはすべての特権が与えられられているため特権機能は使用されません。

  変更された特権ユーザ。特権機能は使用されますが、*root* は特権機能を必要としません。*root* が特権機能を必要とする操作を行う場合は、レコードが維持されます。

  非特権ユーザモードです。UID 0 と *root* アカウントは特殊でなくなります。
- システム調整パラメータを変更。
- 内部のカーネル・デバッグのサポートの呼出し。
- 自動電源オン時間を設定。
- マシン ID を設定（シリアル番号）。

CAP_TIME_MGT

システム時間を設定できる特権です。この特権機能は、システム・クロックを変更するときに必要です。この特権機能には、次の機能が含まれます。

- 時間の微調整を設定。外部ソースとクロックを一致させるときに使用します。
- システム・クロックを調整。
- システム・クロックを設定。
- 高速クロックを有効にする。
- クロック割込みをどのプロセッサで処理するかを制御。

ファイル特権機能

ファイル特権機能は、XFS 形式のファイルシステム上の実行可能ファイルに対してだけ有効です。ファイルの機能必要条件は、システム管理者が chcap(1) を使って設定できます。次の形式で入力します。

chcap *CAP, CAP, CAP file*

たとえば、監査人アカウントに許可する特権機能を設定したいときは次のように入力します。

```
auditor:CAP_AUDIT_WRITE,CAP_AUDIT_CONTROL,CAP_KILL+eip
```

このコマンドを使用します。

chcap CAP_AUDIT_WRITE,CAP_AUDIT_CONTROL,CAP_KILL+eip *file*

ファイルまたはディレクトリの特権機能の必要条件をリスト表示するには、次のコマンドを使用します。

```
ls -P
```

オプション **-P** は、「Privilege」（特権）を表します。ファイルの特権機能を読取るには、ファイルの読込みを可能にする機能が必要です。

カスタム特権機能の作成

サイト固有の特権機能を作成することができます。まず、必要な機能タグ（一意なタグ）を、特権機能を与えるユーザの `/etc/capability` ファイルに追加し、次に chcap(1) コマンドを使って必要なファイルに特権機能を追加します。

特権機能の問題を解決する attrinit の使用

ファイルまたはディレクトリの特権機能の必要条件で問題があることに気が付いた場合には、attrinit(1M) コマンドを使ってこれらの必要条件を復元します。

`attrinit` コマンドには `/etc/irixcap` ファイルを次のように使用します。

ルートとしてログインし、ディレクトリをルート（/）ディレクトリに変更してから次のコマンドを入力します。

```
attrinit -script=/etc/irixcap
```

これで特権機能が復元されます。この処理は数分かかります。

ネットワークのセキュリティ

この章では、ネットワークの安全性を高めるためのさまざまな方法について説明します。通常は、信頼性のあるローカル・グループ内のネットワーク・アクセスに関する方針と、信頼性のないインターネットのような外部ネットワークとの相互アクセスに関する方針を別々に確立します。

メモ：システム・マネージャの GUI には、ローカル・ネットワークのセキュリティを管理するためのセキュリティとアクセス制御に関する機能があります。詳細については、『Personal System Administration Guide』を参照してください。この章では、GUI が提供する同じ機能の多くをコマンド行とファイル・インタフェースを通じて使用します。

この章では、以下について説明します。

- 「ローカル・エリア・ネットワークのアクセス」（132 ページ）では、ローカル・エリア・ネットワークの構築または接続に関連したセキュリティ上の問題について説明します。
- 「ネットワーク・セキュリティとファイアウォールについて」（137 ページ）では、ローカル・システムまたはローカル・サイトと、インターネットのような信頼性のない外部ネットワークの間に、ファイアウォール（防護壁）を構築する際の問題について説明します。
- 「ファイアウォールのハードウェアの設定」（141 ページ）では、セキュリティの観点からネットワーク・ハードウェアの設計構成について説明します。
- 「セキュリティに関する IRIX の設定」（146 ページ）では、IRIX ホストをファイアウォールとして使用するための設定方法について詳しく説明します。
- 「内部ネットワークのセキュリティの設定」（154 ページ）では、ファイアウォールと内部ネットワークでの Sendmail と DNS の設定に関する問題について説明します。

メモ：この章では、第 4 章「IRIX システム・セキュリティ」で説明したホスト・システムの安全対策がすでに確立されていることを前提として説明します。

ローカル・エリア・ネットワークのアクセス

ローカル・エリア・ネットワーク内では、管理が及ばない外部ネットワークでは不可能または不適切と判断されるレベルのアクセスでも許容されます。ここでは、ネットワーク・ホストとユーザ・パーミッション・ファイルを使用してローカル・ネットワーク内のアクセスを制御する方法について説明します。

ネットワーク・アクセス制御ファイル

ネットワーク内のホストに対するアクセスは、次の 3 つのファイルを使用して制御できます。

`/etc/hosts.equiv`	ローカル・エリア・ネットワークと同等に信頼できるホストのリスト
`.rhosts`	特定のユーザ・アカウントへのアクセスを許可されたホストのリスト
`/etc/passwd`	システム・アカウントとその暗号化されたパスワードのリスト

この 3 つのファイルを使用すると、リモート・ホストから rlogin(1C)、rcp(1C)、rsh(1C) または rdist(1) を要求されたときに、アクセスを許可するか拒否するかを制御できます。

アクセスの要求があると、`hosts.equiv` ファイルがチェックされます。このファイルにホストが登録されており、目的のユーザ・アカウントが `/etc/passwd` にあれば、それ以上のチェックが行われず、リモート・アクセスは許可されます。この場合、ローカルのユーザ ID を持つリモート・ユーザは、リモート・ホストからローカル・ユーザと同等のアクセスを行うことができます。デフォルトでは、すべての成功したリモート・アクセスの記録が auth.info メッセージとして `SYSLOG` ファイルに保存されます。syslogd(1M) マン・ページを参照してください。

ユーザは、ホーム・ディレクトリの `.rhosts` ファイルにホストと特定のアカウントを登録すれば、このアクセス許可範囲を拡張できます。*root* ログインは `/etc/hosts.equiv` ファイルのチェックをバイパスし、*root* ディレクトリの `/.rhosts` ファイルだけを使用してアクセス許可範囲をチェックします。*root* の `/.rhosts` ファイルにエントリがあれば、リモート・システムの *root* ユーザはこのシステムに対する *root* 特権を持ちますが、これは安全な方法ではありません。特権を持たない `guest` のようなアカウントを使って、ファイル転送を行う方が安全です。`.rhosts` ファイルに、システム名「localhost」が記述されていれば、`su` をパスワードなしで起動できます。詳細については、su(1M) マン・ページを参照してください。

.rhosts ファイルの所有者は、そのファイルが常駐しているホーム・ディレクトリのユーザまたは特権ユーザの *root* に限ります。ファイルの所有者が別のユーザである場合、または所有者以外のだれでもがファイルを変更できるようにファイルのパーミッションが設定されている場合は、セキュリティ上の理由から、ユーザの .rhosts ファイルの内容は自動的に無視されます。

信頼性のないネットワークに接続するときは .rhosts ファイルの使用を全面的に禁止できます。その場合は、/etc/inetd.conf で rshd を呼出すときに **-l** オプションを指定します。詳細については、rshd(1M) マン・ページを参照してください。信頼性のないネットワークに接続するときの安全な設定については、137 ページの「ネットワーク・セキュリティとファイアウォールについて」で説明します。/etc/hosts.equiv ファイルと .rhosts ファイルの詳細については、hosts.equiv(4) マン・ページを参照してください。

ローカル inetd サービス

inetd プロセスは、多くのネットワーク・サービスを管理します。どのサービスをサポートするかは、各ローカル・エリア・ネットワークによって異なります。どのサービスを提供し、どのサービスのアクセスを記録するかを指定するには、/etc/inetd.conf ファイルを編集します。148 ページの「inetd サービスの制限」を参照してください。ただし、これは、信頼性のないネットワークに接続するときに inetd サービスを制限する場合の説明です。信頼できるローカル・ネットワークの方針を確立する場合は、inetd の制限を緩和できます。

X11 ネットワーク・アクセス

X Window System を使用すると、ワークステーションはネットワークの他のホストでクライアント・プログラムを表示せずに実行できます。このアクセスは、ログイン・アカウントやパスワードなどの制御とは一切関係なしに、X プロトコルを通じて制御されます。

ローカル X サーバへのアクセスを主に制御するのは、特定のシステム・ファイルとユーザ・コマンドの xhost(1) です。特定のシステム制御ファイルとは、/var/X11/xdm/Xsession、/var/X11/xdm/Xsession.dt、/var/X11/xdm/xdm-config の 3 つです。3 つのファイルの設定と xhost コマンドの制御によって、ローカル X サーバへのリモート・アクセスが決まります。この 3 つのファイルと xhost コマンドの詳細については、以下に説明します。

メモ：新しいデフォルトのシステム設定では、X サーバによるリモート・アクセスが無効になります。従来のバージョンでは、リモート・アクセスがデフォルトで有効になりました。デフォルト設定を変更して X サーバからホストにアクセスできるようにする方法については、135 ページの「xhost コマンドによるアクセスの制限」を参照してください。

セキュリティと X サーバの初期設定

X サーバが起動すると、最初に `/etc/X*.hosts`[1] ファイルの有無を調べます。このファイルが存在しない（IRIX の出荷時のデフォルト設定）と、リモート・ホストからローカル X サーバへのアクセスは全面的に拒否され、その後に続く `Xsession` ファイルの評価も停止されます。次に、X サーバは `/var/X11/xdm/Xsession` ファイルと `/var/X11/xdm/Xsession.dt` ファイルを読込みます。どちらのファイルも `xhost` コマンドを実行しない場合（`xhost +` コマンドはデフォルトでコメントアウトされる）、リモート・ホストからローカル X サーバへのアクセスは全面的に拒否されます。

X サーバのデフォルト初期設定を変更する方法については、以下に説明します。

X0.hosts ファイルによるアクセスの制限

特定のリモート・ホストにアクセスを許可するには、そのリモート・ホスト名を `/etc/X*.hosts` ファイルに登録します。たとえば、`/etc/X0.hosts` ファイルに次の行があると、リモート・ホストの bronx ワークステーションだけが X サーバ 0: のローカル・サーバにアクセスできます。

```
bronx
```

この例では、他のすべてのホストは、サーバの初期設定時にローカル・サーバへのアクセスを拒否されます。ただし、`/var/X11/xdm/Xsession` ファイルまたは `/var/X11/xdm/Xsession.dt` ファイルで、サーバの起動時またはそれ以降にユーザによって相反する `xhost` コマンドが実行された場合を除きます。

[1] この設定ファイル名の中で、アスタリスク (*) はローカル・ホストの X サーバ数を表します。通常の X サーバ数は 0 なので、大部分のワークステーションではファイル名が `/etc/X0.hosts` になります。X サーバ 0 が起動すると `/etc/X0.hosts` がチェックされ、X サーバ 1 が起動すると `/etc/X1.hosts` がチェックされます（以下同様）。

メモ：X*.hosts ファイルを /etc/hosts や /etc/hosts.equiv などのネットワーク・ホスト・データベースにリンクしないでください。X サーバが起動すると、X*.hosts ファイルでアクセス権が認められているすべてのホストに接続しようとします。このファイルにサーバへのアクセスが許可されているホストが多数ある場合は、サーバを起動する前に、個々のホストとの接続が確立されるのを待つ必要があります。

xhostコマンドによるアクセスの制限

xhost コマンドは、X サーバの内部状態を変更します。xhost を使用すると、特定のホストまたはすべてのホストに対してサーバ・アクセスを許可または拒否できます。アクセス制御に関する xhost オプションを実行できるのは、サーバと同じワークステーションだけです。

たとえば、ほかのすべてのホストに X サーバにアクセスすることを許可するには、/var/X11/xdm/Xsession と /var/X11/xdm/Xsession.dt の xhost 行からコメントを削除します。デフォルトでは、次のようなコメント行になっています。

```
# Gives anyone on any host access to this display
# /usr/bin/X11/xhost +
```

すべてのリモート・ホストにアクセスを許可するには、次のように変更します。

```
# Gives anyone on any host access to this display
/usr/bin/X11/xhost +
```

サーバへのアクセスを特定のリモート・ホスト（たとえば brooklyn）だけに許可するには、次のように変更します。

```
# Gives anyone on any host access to this display
/usr/bin/X11/xhost +brooklyn
```

/var/X11/xdm/Xsession ファイルと /var/X11/xdm/Xsession.dt ファイルに挿入できる xhost コマンドの他の例を次に示します。

xhostコマンドの対話モードでの使い方

xhost コマンドは対話モードでも使用できます。ネットワークのすべてのホストに X プロトコルを使ったアクセスを全面的に拒否するには、次のコマンドを実行します。

```
# xhost -
```

ネットワークのすべてのホストにアクセスを全面的に許可するには、次のコマンドを実行します。

```
# xhost +
```

アクセスを許可または拒否するホストを限定するには、コマンド行でホスト名を指定します。たとえば、次のコマンドは brooklyn という名前のホストにアクセスを許可します。

```
# xhost +brooklyn
```

アクセスを許可するとき、正符号 (+) は省略可能です。

次のコマンドは brooklyn と bronx の両方にアクセスを拒否します。

```
# xhost -brooklyn -bronx
```

現在サーバへのアクセスが許可されているホストを確認するには、コマンド行から引数を指定しないで `xhost` を実行します。

```
# xhost
```

安全性が損なわれるおそれがある場合は、ユーザに `xhost +` の使用を禁止するか、システムからコマンド自体を削除できます。

X 権限

134 ページの「セキュリティと X サーバの初期設定」で説明した X サーバのデフォルト設定より優れたセキュリティを実現するには、X 権限を使用します。X 権限を使用するには、`/var/X11/xdm/xdm-config` の `DisplayManager*authorize` エントリを次のように変更します。

```
DisplayManager*authorize: on
```

`xdm` は、各ユーザの `$HOME/.Xauthority` ファイルに雛形を作成します。この雛形がなければ、どの X クライアントも X サーバに接続できないので、X サーバへのアクセスを有効に制御できます。X 権限の使用は、システムのデフォルト設定になっている場合もあります。

X のセキュリティと権限の詳細については、xsecurity(1)、xhost(1)、xauth(1)、xserver(1)、X(1) の各マン・ページを参照してください。

ネットワーク・セキュリティとファイアウォールについて

ホストとサイトのセキュリティ方針を確立した後で、そのサイトをインターネットのような外部ネットワークに接続する場合があります。ここでは、セキュリティの管理が及ばないインターネットのような信頼できないネットワークと接続を確立する方法について説明します。外部ネットワークに接続する場合は、内部の信頼できるネットワークと外部ネットワークの間のインタフェースについて特別な配慮が必要です。信頼できるネットワークへの信頼性のないトラフィックの侵入を防止するインタフェースをファイアウォールと呼びます。ここでは、ファイアウォールについて主に説明します。ここでの説明は、インターネットに接続する場合を例として取上げますが、ほかの信頼性のないネットワークと接続する場合にも適用できます。

インターネットについて

インターネットは、各種のコンピュータ・リソースを接続した巨大なネットワークです。その規模は、地球全体に及び、日々拡大しています。インターネットにアクセスする個人や団体は増加の一途を辿っており、電子メール、大量に蓄積された情報の利用、各分野の最新動向の把握など、インターネットのサービスとリソースはビジネスなどの目的に広く活用されています。

最近、インターネットが特に脚光を浴びるようになったのは、World Wide Web (WWW) の発展によります。WWW は、インターネット・リソースへのフレンドリなグラフィカル・インタフェースと、リソースを提示およびアクセスするための標準化された方法を提供します。WebFORCE のようなインターネット用の製品を使用してインターネットに提示された情報は、世界中からアクセスできます。

インターネットはデータを共有する手段を提供しますが、データを保護するのはユーザ自身の役割です。ここでは、インターネットを介したアクセスの重要な面として、ローカル・コンピュータとネットワークのセキュリティを確立および維持する必要性について説明します。特に、コンピュータ・サイトは、明らかな違法行為からだけでなく競争相手からもデータの機密と安全性を確保する必要と権利があります。

ネットワークのセキュリティ上の問題

インターネットに接続する場合は、重要なデータが簡単に読まれたり破壊されたりしないような設定を行います。どのデータを公開するのかを正確に判断し、不法侵入者からサイトを守る必要があります。インターネットには、不法侵入や破壊行為の例がたくさんあります。そのような事実があることを認識し、それを予防することがインターネットへの参加を楽しく、実りあるものにします。

この章では、コンピュータ・ネットワークでの悪質な行為や犯罪行為を具体的に説明しません。関連する情報は、インターネット自体に豊富に掲載されています。コンピュータ・セキュリティの担当者は、xxv ページの「参考資料」のインターネット上のリソースを参照してください。

通常、防御線を確立するのは、ユーザの信頼できるコンピュータ・リソース（内部ネットワーク）と、インターネットを通じて外部からアクセスできるコンピュータ・リソース（外部ネットワーク）の間です。この防御線の役割は、外部からの直接アクセスを防ぐことです。防御線は、単一のルータまたはコンピュータ・ホストという単純なものから、複数のルータとコンピュータ・ネットワーク全体で構成される複雑なものまであります。ここでは、単一のルータで構成される制限されたファイアウォールではなく、コンピュータ・ホストまたはネットワークで実現できる安全なファイアウォールの確立方法について説明します。防御線を確立した後で、内部の信頼できるユーザからインターネットにアクセスする場合と、外部のユーザから内部のリソースにアクセスする場合のアクセスの許可レベルを指定します。

ファイアウォールについて

ファイアウォールを構築することによって、外部の信頼できないホストと内部の信頼できるホストとの間に防御線ができ上がります。ファイアウォールは、インターネットなどの外部ネットワークとのやり取りを必要に応じて制限するためのコンピュータ・ハードウェアとソフトウェアの組合せです。簡単に言うと、アクセスを許すほど安全性に対する危険が高まり、アクセスを制限するほど安全性を監視および維持しやすくなります。つまり、利便性と安全性の兼合いになります。システムやネットワークの管理者は、ユーザに許可するアクセスの度合いと管理者の職務である安全性の確保の間でバランスを取ることになります。

ファイアウォールで問題となるのは、場合によって一般の従業員がセキュリティ保護のためアクセス制限されているパケットに合法的にアクセスする必要が生じることです。別の方法として、

ネットワーク接続されたコンピュータごとに高レベルなセキュリティを設定、管理しているサイトもあります。

単純なファイアウォールの例を図 5-1 に示します。この図では、単一のコンピュータ・ホストが 2 つのネットワーク・インタフェースで設定されています。これは、ホストが 2 つの異なるネットワークにそれぞれ存在する、いわゆる二重ホームのホストです。この章の説明どおりに設定したファイアウォールは、内部ネットワークとインターネットを隔てる唯一の制御された障壁となり、そこにセキュリティ対策を集中できます。この章で、ファイアウォール・ホストとは、ネットワーク・セキュリティのために設定された IRIX ホストを意味します。IRIX 版 Gauntlet は市販されている IRIX 用ファイアウォールの例です。詳細については、日本シリコングラフィックス株式会社のサポート部門にお問い合わせください。

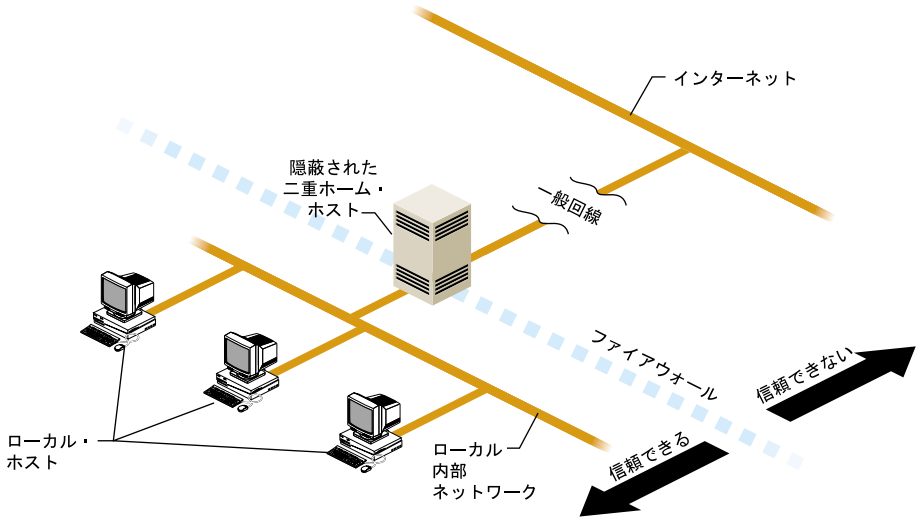


図 5-1　　単純なファイアウォールの環境

ファイアウォールは、内部ネットワークでの対話を一切制限しません。ローカル・ホストによるリソースの共有方法は、ファイアウォールに接続する前と同じです。接続後に異なるのは、ローカル・ホストが外部サイトと対話する方法です。外部サイトと対話できる度合いは、各自のサイトの方針によって決定されます。二重ホームのホストによるネットワークの設定例については、154 ページの「内部ネットワークのセキュリティの設定」で説明します。

ファイアウォールの設計方針

ネットワークのセキュリティを管理する秘訣はファイアウォールにあります。内部セキュリティに関する問題も重要ですが、その問題はインターネットに接続するしないに関係ありません。ホストのセキュリティについては、第 4 章「IRIX システム・セキュリティ」を参照してください。ローカル・ネットワークのセキュリティについては、132 ページの「ローカル・エリア・ネットワークのアクセス」を参照してください。

ファイアウォール自体についての方針

- ユーザ数を制限します。— できるかぎり 1 人のシステム管理者で管理できるユーザ数に制限します。ユーザを追加する場合は、150 ページの「ファイアウォールでのパスワードの保護」と 154 ページの「ファイアウォールに関するユーザの教育」を参照してください。
- サービス数を制限します。— サービス数を増やすほど、セキュリティの盲点も増えます。通常、サービスを提供するソフトウェアが複雑になるほど、危険性が増します。また、ソフトウェアが新しいほど、実用化されてからのテスト経験が浅いことになります。
- システムを監視します。— このマニュアルを参考にして、アクセスの時刻やアクセス試行の失敗など、ファイアウォール・ホストへのアクセスに関する情報をログ・ファイルに記録するように IRIX ソフトウェアのファイアウォールを設定できます。また、w(1) や ps(1) などの多数の標準 IRIX ツールを活用すると、現在のシステムの利用状況を確認できます。セキュリティ関連のコマンドのリストについては、表 4-3 を参照してください。
- ファイアウォールでは他のアプリケーションを実行しません。— アプリケーションが増えると、セキュリティ上の問題が持ち込まれるおそれがあり、ソフトウェア環境も複雑になってセキュリティを管理しにくくなります。

ファイアウォールのモニタリング

ファイアウォールを設けても、適切な方法でモニタリングしなければ意味がありません。ファイアウォールのセキュリティをチェックする次のようなツールがあります。

Cops　パーミッション・モード、パスワード、`/etc/password` と `/etc/group` の整合性、cron(tab) ファイル、*setuid* ファイル、重要なバイナリ・ファイルのチェックサム、ホーム・ディレクトリとスタートアップ・ファイル、anonymous

`ftp` のセットアップ、制限なしのアクセス方法、特権ユーザ（root）のセキュリティ、および CERT advisory の日付に対するキー・ファイルをチェックし、Kuang エキスパート・システムを使用します。

Cops は ftp://ftp.cert.org/pub/tools/cops にあります。

Tripwire　ファイルとディレクトリの整合性とチェックし、指定のファイルとディレクトリを以前に作成されたデータベースの情報と照合します。追加、削除を含む相違箇所が見つかると、フラグがセットされ、ログに記録されます。定期的に実行することで、致命的なシステムに変更が生じた場合にフラグを立て、破損を回復する手段をただちに講じることができます。

Tripwire は ftp://ftp.cert.org/pub/tools/tripwire にあります。

World Wide Web のセキュリティ上の問題

World Wide Web のソフトウェアにアクセスするときにも同様なセキュリティ上の問題があります。未知のソースまたは信頼できないソースからソフトウェアを入手するときには、同じような問題が発生する可能性があります。たとえば、クリックするだけで危険な実行可能ファイルがダウンロードされるおそれがあります。この種の問題に真剣に対処するサイトでは、World Wide Web にアクセスするホストを切離し、制限することを検討します。

World Wide Web に関連したセキュリティ上の問題の詳細については、xxv ページの「参考資料」を参照してください。

ファイアウォールのハードウェアの設定

ここでは、ファイアウォールのハードウェア部分を構成するネットワーク・ハードウェアの設定方法について説明します。ファイアウォール機能を備えた Silicon Graphics ソフトウェアの設定方法については、146 ページの「セキュリティに関する IRIX の設定」を参照してください。ルータだけのファイア・ウォールでは制限があるので、ここでは IRIX ホストを組込んだファイアウォールの設定方法について説明します。ファイアウォール・ホストには、特定のアプリケーションの許可または制限、ログ・ファイルの作成、ネットワーク・アクセスへの認証の追加を行えるという利点があります。

ルータとファイアウォール

通常、ファイアウォール・ホストはルータと組合わされます。ルータは、インターネット・サービス・プロバイダへの接続の一環として提供される場合と、各ユーザの設定に追加される場合とがあります。

正しく設定したルータは、IP パケットを選別することによって、ある程度のセキュリティを確保します。IRIX ホストは、IP パケット・フィルタとして使用できます。詳細については、ipfilterd(1M) マン・ページを参照してください。通常、ルータは IP パケットの高速フィルタ機能を提供する純然たるハードウェア装置です。多くのルータは、IP パケット・レベルのセキュリティを提供するように設定できますが、プロクシや認証などの機能はサポートしていません。

プロクシは代理サーバのことで、ネットワーク・リソースをアプリケーション・レベルで管理できるようにします。[2] 認証はユーザに身元を証明することを要求する方法です。プロクシや認証などの機能を追加するには、以下に説明する IRIX ホストの設定を参考にして、ネットワーク・ハードウェアを設定します。

ルータとファイアウォールを結合すると、IP パケットの選別とアプリケーション・レベルの管理が可能になります。ルータとファイアウォール・ホストを結合する場合は、ファイアウォール・ホストに向かうトラフィックだけを許可するようにルータを設定します。以下のトラフィックは除外します。

- ICMP[3] からの転送で、ルータからでないもの。
- 経路指定が不明確な IP パケット。
- 内部ネットワークから来たと主張する外部パケット（spoofing）。

インターネットに接続して利用できるパケットのフィルタ・オプションを選択するには、インターネット・サービス・プロバイダに相談してください。また、次の説明に従ってファイアウォールの設定にルータを追加し、ルータにフィルタ・オプションを組込むこともできます。詳細については、ルータのベンダから提供されるマニュアルを参照してください。また、xxv ページの「参

[2] たとえば、Netscape Proxy Server は World Wide Web の HTTP サーバなど一般のネットワーク・サービスに対してアプリケーション・プロクシを提供します。

[3] Internet Control Message Protocol

考資料」で紹介している Cheswick と Bellovin によって書かれた『Firewalls and Internet Security』書籍の「Packet Filtering Gateways」も参照してください。

ファイアウォールとして機能するハードウェアの設定

この章では、ファイアウォールとして機能する二重ホーム・ホストの基本的な設定と、ホストの隠蔽およびネットワークの隠蔽のためのファイアウォールの設定について説明します。

二重ホーム・ホストのファイアウォール

Silicon Graphics のホスト・ハードウェアは、ファイアウォールとしても機能します。その場合は、ハードウェアをネットワークの接続先を 2 つ持つ二重ホームのゲートウェイとして設定します。図 5-1 に、二重ホーム・ホストをファイアウォールとして使用するための概念を示します。

二重ホーム・ホストを構築するには、別のイーサネット制御ボードが必要になる場合があります。サイトによっては、2 つのイーサネットがすでに接続されている場合もあります。ネットワーク・ハードウェアの詳細については、システムのマニュアルを参照してください。

隠蔽されたホストのゲートウェイ

隠蔽されたホストの設定では、インターネットとファイアウォール・ホストの外部ネットワーク接続間のトラフィックをルータを使用して選別します。ルータは多種多様ですが、大部分は特定のアドレスまたは設定を除外することによって IP パケットを選別します。外部から内部ネットワークの対象外のホストに向かうトラフィックは除外されます。これは、インターネット・サービス・プロバイダがルータを提供する場合のインターネットとの一般的な接続方法です。図 5-2 に隠蔽されたホストの基本的な設定を示します。

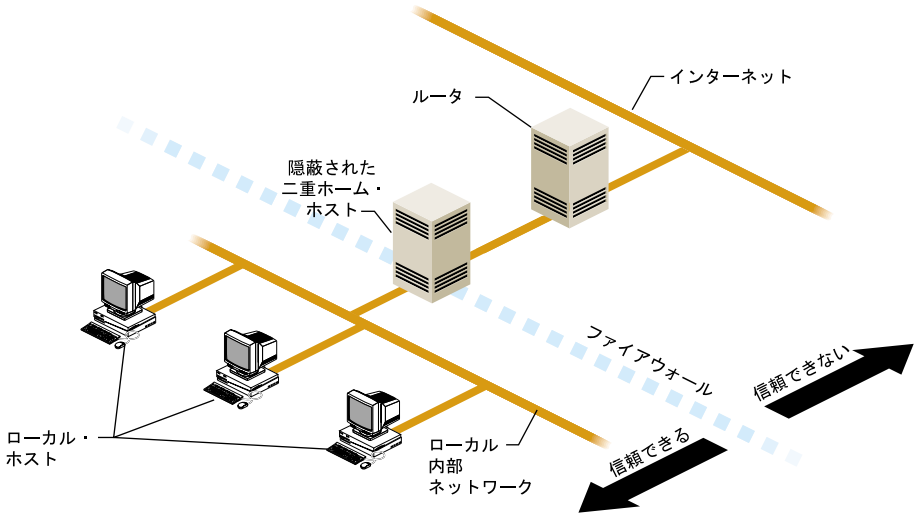


図 5-2　　隠蔽されたホスト

隠蔽されたホストの設定をネットワークにまで広げると、複雑さも増しますが柔軟性も増します。基本的な設定は同じですが、隠蔽されたネットワークは外部からのすべてのトラフィックを受付けます。インターネットも内部ネットワークも、隠蔽されたネットワークにアクセスしますが、内部ネットワークに関するトラフィックは依然としてファイアウォール・ホストを経由します。この方法は、サイトでインターネットに複数のサーバを使用可能にする一方で、内部ネットワークの安全性を確保するときに有効です。公開するデータと、そのために必要な CPU の負荷に応じて、パブリック・ホストの 1 つを WWW サーバとして、もう 1 つを FTP サーバとして利用することなどができます。

図 5-3 に隠蔽されたサブネットを示します。[4]

[4] 隠蔽されたサブネットのことを DMZ またはレッド・ゾーンと呼ぶ場合もあります。

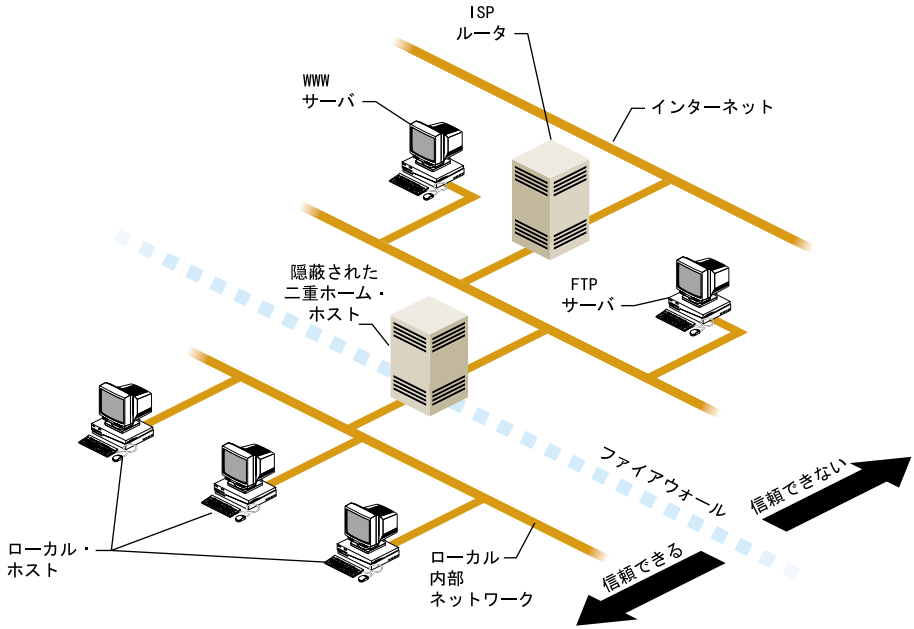


図 5-3　　隠蔽されたサブネット

図 5-3 に示した設定では、セキュリティ対策を依然として単一のファイアウォール・ホストに集中できます。ただし、ファイアウォール外のサーバは、ルータに保護されているだけなので、危険性が高いことに注意します。非公開データは内部ネットワークに保存し、パブリック・サーバで収集した重要なデータは内部ホストに転送します。ソフトウェア設定の詳細については、次に説明します。

セキュリティに関する IRIX の設定

ここでは、ファイアウォール・ホストに要求される基本的なネットワークのアドレス設定について説明します。ホストでセキュリティを強化するための IRIX ソフトウェアの設定方法についても詳しく説明します。

メモ：特に明記しないかぎり、ここで説明するソフトウェアのすべての変更は、ファイアウォール・ホストで行うものとします。

二重ホーム・ホストでのネットワーク・ソフトウェアの設定

ネットワーク・ソフトウェアでは、二重ホーム・ホストは 2 つのホストであるかのように設定されます。2 つのホストには異なるネットワーク・アドレスと、必要に応じて異なる名前も与えることができます。この 2 つ（以上）のネットワーク・インタフェースに対しては別々の IP アドレスを使用します。『IRIX Admin: Networking and Mail』を参照してください。

IRIX でのセキュリティの強化

ここでは、ネットワーク・セキュリティを強化するために IRIX オペレーティング・システム・ソフトウェアに適用できる各種の変更について説明します。変更の中には、ファイアウォールに必須なものもあれば、ユーザが希望するセキュリティ・レベルに応じて各自で選択できるものもあります。変更が必要な理由と可能な理由についても説明します。

ファイアウォール・ホスト・ソフトウェアの変更に関する次の説明は、図 5-3 に示した隠蔽されたサブネットの WWW サーバや FTP サーバなど、一般にアクセス可能なホストにも当てはまります。

メモ：ここで説明する変更を行うまでは、ハードウェアを外部ネットワークに接続しないでください。変更を完了した後で、ファイアウォール・システムを再起動し、すべての変更が有効に反映されたことを確認します。多くの変更は、システムを再起動するまで反映されません。

IP パケットの転送禁止

デフォルトでは、IRIX は複数のネットワーク・ハードウェア・インタフェースを備えたコンピュータで IP パケットを転送します。このデフォルトを無効にするには、カーネルの設定ファイルを編集し、`autoconfig` を実行し、システムを再起動します。

IP パケットの自動転送を禁止するには、以下の手順に従います。

1. 特権ユーザとして、ファイル `/var/sysgen/master.d/bsd` を編集し、*ipforwarding* の値を 0 に変更します。

   変更前の行は次のとおりです。

   ```
   int ipforwarding = 1;
   ```

   変更後の行は次のとおりです。

   ```
   int ipforwarding = 0;
   ```

2. 変更した `/var/sysgen/master.d/bsd` ファイルを保存し、エディタを終了します。

3. `autoconfig` に **`-f`** オプションを指定して実行します。

   ```
   # autoconfig -f
   ```

   `/unix.install` ファイルが作成されます。システムを再起動すると、このファイルが新しい `/unix` になります。

4. システムを再起動します。reboot(1M) マン・ページを参照してください。

5. システムが起動した後、`netstat` コマンドを使用して、IP パケットの転送が禁止されていることを確かめます。

   ```
   # netstat -s -p ip | grep forwarding
   ```

   次のメッセージが表示されます。

   ```
    0 packets forwarded (forwarding disabled)
   ```

   メッセージが表示されない場合は、表示されるまで手順1～5を繰返します。`/unix.install` ファイルを正しく作成できるだけのディスク容量がルート・ファイルシステムにあることを確認します。詳細については、autoconfig(1M) マン・ページを参照してください。

inetd サービスの制限

システムが起動すると、inetd プロセスがサポートする TCP/IP サービスのリストが /etc/inetd.conf ファイルから読込まれます。読込まれたサービスの中から、安全性に疑問があるものや使用しないものをコメントにして取除きます。

メモ：取除いたサービスは、ファイアウォールでのみ無効になります。ファイアウォールのシステム・ファイルでコメントにして取除いたサービスは、ファイアウォール・ホストでは利用できませんが、内部ネットワークでは依然として利用できます。

選択した inetd サービスを無効にするには、次の手順に従います。

1. /etc/inetd.conf ファイルを編集し、シャープ記号 (#) を次の各行の先頭に書込んでコメントとして取除きます。すでにコメントになっている場合もあります。

```
exec         stream  tcp     nowait  root    /usr/etc/rexecd         rexecd
bootp        dgram   udp     wait    root    /usr/etc/bootp          bootp
rstatd/1-3   dgram   rpc/udp wait    root    /usr/etc/rpc.rstatd     rstatd
walld/1      dgram   rpc/udp  wait    root    /usr/etc/rpc.rwalld     rwalld
rusersd/1    dgram   rpc/udp  wait    root    /usr/etc/rpc.rusersd    rusersd
rquotad/1    dgram   rpc/udp  wait    root    /usr/etc/rpc.rquotad    rquotad
bootparam/1 dgram    rpc/udp wait     root    /usr/etc/rpc.bootparamd
bootparam
ypupdated/1 stream   rpc/tcp wait     root    /usr/etc/rpc.ypupdated
ypupdated
rexd/1       stream  rpc/tcp wait     root    /usr/etc/rpc.rexd       rexd
```

   # 記号を追加した後の各行は次のようになります。

```
#exec         stream  tcp     nowait  root    /usr/etc/rexecd         rexecd
#bootp        dgram   udp     wait    root    /usr/etc/bootp          bootp
#rstatd/1-3   dgram   rpc/udp wait    root    /usr/etc/rpc.rstatd     rstatd
#walld/1      dgram   rpc/udp wait    root    /usr/etc/rpc.rwalld     rwalld
#rusersd/1    dgram   rpc/udp wait    root    /usr/etc/rpc.rusersd    rusersd
#rquotad/1    dgram   rpc/udp wait    root    /usr/etc/rpc.rquotad    rquotad
#bootparam/1 dgram   rpc/udp wait    root   /usr/etc/rpc.bootparamd bootparam
#ypupdated/1 stream  rpc/tcp wait   root   /usr/etc/rpc.ypupdated  ypupdated
#rexd/1       stream  rpc/tcp wait    root    /usr/etc/rpc.rexd       rexd
```

   無効にするサービスの詳細については、それぞれのマン・ページを参照してください。たとえば、リモート実行サーバに関する情報については rexecd(1M) マン・ページを、RPC ベースのリモート実行サーバに関する情報については rexd(1M) マン・ページを参照してください。

2. `/etc/inetd.conf` の次のサービスをコメントにするか、制限します。

```
ftp     stream  tcp     nowait  root    /usr/etc/ftpd   ftpd -la
telnet  stream  tcp     nowait  root    /usr/etc/telnetd        telnetd
shell   stream  tcp     nowait  root    /usr/etc/rshd           rshd -L
login   stream  tcp     nowait  root    /usr/etc/rlogind        rlogind
tftp    dgram   udp     wait    guest   /usr/etc/tftpd  tftpd -s \
/usr/local/boot /usr/etc/boot
```

すべてのエントリをコメントにした場合は、次のようになります。

```
#ftp     stream  tcp     nowait  root    /usr/etc/ftpd   ftpd -l
#telnet  stream  tcp     nowait  root    /usr/etc/telnetd        telnetd
#shell   stream  tcp     nowait  root    /usr/etc/rshd           rshd -L
#login   stream  tcp     nowait  root    /usr/etc/rlogind        rlogind
#tftp    dgram   udp     wait    guest   /usr/etc/tftpd  tftpd -s \
/usr/local/boot /usr/etc/boot
```

上に示したように、コメント文字ですべてのサービスを無効にするのが最も安全な方法です。ただし、その場合、ホストにアクセスできるのはローカル・コンソールからだけになります。サービスの中でも、`rshd` を有効にしておくのが最も危険です。また、`tftpd` がファイアウォールで必要になることはありません。`ftpd` については、『IRIX Admin: Networking and Mail』を参照してください。この 2 つのサービスを使用する場合は、次のように変更して、使用履歴を `/var/adm/SYSLOG` ファイルに残します。

```
ftp     stream  tcp   nowait  root   /usr/etc/ftpd   ftpd -ll
shell   stream  tcp   nowait  root   /usr/etc/rshd   rshd -Lal
tftp   dgram   udp   wait    guest  /usr/etc/tftpd  tftpd -s -l -h /dev/null
```

各デーモンを起動すると、ログ・オプションが追加されます。詳細については、変更するデーモンのマン・ページを参照してください。

`telnetd` と `rlogind` の各サービスは、この変更に含まれていません。リモート・ログインはワン・タイム・パスワードで制御されるためです。ワン・タイム・パスワードとは、アクセスを獲得するために一度だけ使用できるパスワードのことで、次からは同じパスワードを使用できません。ワン・タイム・パスワードを実現する方法はいろいろありますが、それをユーザ・サイトで使用するかどうかと使い方は、リモート・ログイン機能の必要性とリモート・ログインの許可レベルに応じて異なります。xxvi ページの「書籍」の『Firewalls and Internet Security』を参照してください。

3. `fingerd` サービスもセキュリティの盲点になる可能性があります。それがアカウント名のソースであるためです。**`-S`** オプションを指定すると、ログイン・ステータス、ホーム・ディレクトリ、シェルに関する情報がセキュリティ侵害に悪用されるのを阻止できます。

```
finger  stream  tcp     nowait  guest   /usr/etc/fingerd        fingerd -S
```

安全性を高めるには、fingerd に **-f** オプションを指定し、メッセージ・ファイルだけを返させることもできます。次の例では、メッセージが /etc/fingerd.message ファイルに格納されます。

```
finger  stream  tcp     nowait  guest   /usr/etc/fingerd        fingerd -f \
/etc/fingerd.message
```

/etc/fingerd.message の内容は次のようになります。

```
Thank you for your interest in XYZ company. Please contact us at
xyz.email.address or 1-800-XYZ-PHON for more information.
```

finger にアクセスすると、このメッセージが必ず返されます。

4. /etc/inetd.conf ファイルを変更した後、それを保存してエディタを終了します。システムを再起動すると変更が反映されます。すぐに変更を反映させる場合は、次のように入力します。

   ```
   # killall -HUP inetd
   ```

5. 変更したサービスをテストし、変更が反映されていることを確認します。

ファイアウォールでのパスワードの保護

ファイアウォール・システムでは、ログイン・アカウントを持つユーザの数をできるかぎり制限します。/etc/passwd のすべてのアカウントには、パスワードを設定します。passwd(1) マン・ページを参照してください。

/etc/hosts.equiv ファイルまたは $HOME/.rhosts ファイルの有無を確認します。この 2 つのファイルは、パスワードが保護されていないリモート・アクセスを許可するように設定できるので、ファイアウォール・ホストでの使用を禁止します。詳細については、hosts.equiv(4) マン・ページを参照してください。

ホストにアクセスするときのパスワード保護の詳細については、87 ページの「パスワード管理」を参照してください。

ファイアウォールでの rpc サービス・アクセスの制限

ファイアウォール・ホストの RPC サービスへのアクセスを制限するには、`portmap` コマンドに **`-a`** オプションを指定して実行します。この指定によって、RPC ベースのサービスにアクセスを許可するホストやネットワークを指定できます。`/etc/config/portmap.options` ファイルを編集し、システムの起動時に実行される `portmap` コマンドにオプションを追加します。

たとえば、`/etc/config/portmap.options` ファイルを編集し、次のようなエントリを追加したとします。

```
-a 192.0.2.0
-a 192.14.12.0
```

これによって、ファイアウォール・ホストの RPC サービスにアクセスできるのは、クラス C のネットワークである 192.0.2 と 192.13.12 のホストだけになります。

`-a` オプションの構文を使用すると、複数のネットワーク・マスク、ネットワーク・アドレス、ホスト・アドレスを指定できます。通常、アクセスを許可するホストまたはネットワークの数が少ないほど、セキュリティは向上します。詳細については、portmap(1M) マン・ページを参照してください。

ファイアウォールでの NIS（YP）の無効化

NIS（以前のイエロー・ページ）は、その性質上、セキュリティの要請に対応しないので、ファイアウォール・ホストから取除きます。

1. NIS ソフトウェアをファイアウォール・ホストから取除くには、`versions` コマンドを使用します。

   ```
   # versions remove nfs.sw.nis
   ```

2. データベースの中には、エントリにプラス記号 (+) を含めることによって NIS 情報を追加しているものがあります。エディタを使用し、`/etc/passwd`、`/etc/group`、`/etc/aliases` の各ファイルから + 記号で始まる行を削除します。

3. `/etc/netgroups` というファイルがある場合は、それを削除します。

注意：ファイアウォールでは NIS を実行しないでください。NIS を実行する場合は、サーバが安全であることを確認し、クライアントには必ず `ypbind` に **`-ypsetme`** オプションを指定して実行させます。これにより最小限のセキュリティを確保できます。

ファイアウォールでの NFS アクセスの禁止

ファイアウォールでファイルシステムをエクスポートしたり、外部システムをリモート・マウントすることは、セキュリティ上の問題を生じます。`NFS` は、`versions remove nfs.sw.nis` で削除し、ファイアウォールで使用できないようにします。何かの理由で NFS を削除できない場合は、次の方法を使うこともできますが、セキュリティは低下します。

- 次のコマンドで NFS の使用を全面的に禁止できます。

  ```
  # chkconfig nfs off
  ```

- `/etc/exports` ファイルを編集して、エクスポートするファイルシステムのパーミッションとアクセスを制限できます。たとえば、`rw=`*hostname* オプションを使用して、特定のホストに対する読み書きを制限できます。`access=`*client* オプションを使用して、マウント先を特定のホストに制限することもできます。詳細については、exports(4) マン・ページを参照してください。

- ファイアウォール・ホストに外部システムをマウントする場合は、`mount` コマンドに **`nosuid`** オプションを指定して、トロイの木馬の実行を防ぎます。詳細については、fstab(4) マン・ページを参照してください。

通常、システム・ディレクトリ以外のすべてのファイルシステムをマウントするには、**`nosuid`** オプションと **`nodev`** オプションを指定します。mount(1M) マン・ページを参照してください。

ファイアウォールでのログ・ファイルについて

ログ・ファイルは、ファイアウォール・ホストに対する特定またはすべてのログイン試行を記録し、ファイアウォール管理者に有用な情報を提供します。各デーモンに対してログを起動するオプションについては、デーモン別に説明しています。ログ・ファイルを有効に利用するには、定期的にチェックすることが必要です。

ログ・ファイルの情報は機密扱いにして、ファイアウォール・ホストには保存しないようにします。ファイアウォール・ホストからファイアウォール内の信頼できるホストに `syslog` メッセージを転送する方法については、syslogd(1M) マン・ページを参照してください。

ファイアウォールでのソフトウェアの整合性検査

ファイアウォール・ホストのどのソフトウェアも変更されないように注意します。ソフトウェアの照合総数を記録し、定期的に比較して違法な変更を検出します。この目的のためにも、ファイアウォール・ホストにインストールするソフトウェアの数は少ない方が得策です。`/dev` 以外のデバイス・ファイル、SUID や GUID のパーミッションが設定されたファイルなどには常に注意します。

`versions` コマンドを使用すると、インストール後に変更されたシステム・ファイルをリスト表示できます。次に例を示します。

```
# versions changed
構成ファイル

m   = 最初のインストールからの変更されている。
?   = 変更については不明
空白 = ファイルは元々インストールされています。

  /etc/init.d/netsite
m /etc/init.d/netsite.O
m /etc/init.d/netsite.N
m /etc/uucp/Devices
  /etc/uucp/Devices.N
m /var/X11/xdm/Xsession.dt
  /var/X11/xdm/Xsession.dt.O
  /var/X11/xdm/xdm-config
<etc>
```

`versions -m` コマンドを使用すると、インストール済みのファイルの中で、変更されたものだけをリスト表示できます。詳細については、versions(1M) マン・ページを参照してください。

ファイアウォールに関するユーザの教育

安全なファイアウォールを苦労して確立しても、ユーザの不注意な行為によって台無しになる場合があります。ファイアウォール・ホストでは、ユーザ・アカウントをできるかぎり使用できないようにします。ユーザ・アカウントを許可する場合は、アカウントの保有者に次のことを徹底します。

- `.rhosts` ファイルを使用しません。特権ユーザとして、`/etc/inetd.conf` で `rshd` を起動するときに **`-l`** オプションを指定すると、ファイアウォールでの `.rhosts` ファイルの使用を禁止できます。詳細については、rshd(1M) マン・ページを参照してください。
- パスワードには、長くて辞書にも載っていない ASCII 文字列を使用します。また、パスワードを頻繁に変更し、何かに書留めたりしないようにします。
- 「`xhost +`」コマンドは使用しません。特権ユーザとして、その実行ファイルを削除するか、特権ユーザだけが実行できるようにします。

安全であるはずの内部ネットワークでも、ユーザの不注意な行為によって安全でなくなる場合があります。たとえば、内部ホストのユーザがモデムを取付けて外部サイトと PPP セッションまたは SLIP セッションを確立すると、外部世界から内部ネットワークに 2 通りの方法でアクセスできることになります。1 つの方法ではファイアウォールを経由しますが、別の方法では保護されていない内部ホストに直接アクセスできます。

内部ネットワークのセキュリティの設定

ここでは、内部ネットワークの基本的な設定方法に関する説明を省き、特にファイアウォールのセキュリティに関連した DNS と Sendmail の設定について説明します。Sendmail と DNS の基本的な設定方法については、『IRIX Admin: Networking and Mail』を参照してください。

ドメイン・ネーム・システム（DNS: Domain Name System）のセキュリティに関するガイドライン

ドメイン・ネーム・システム（DNS: Domain Name System）は、インターネットで使用されるネーム・サービスです。各サイトは DNS を設定し、他のサイトから接続するときのアドレスを提供します。DNS には、外部からの接続を受付けるためのルータやファイアウォール・ホストなどのマシンのアドレスが含まれます。二重ホーム・ホストで構成された単純なファイアウォールの場合、その二重ホーム・ホストが DNS サーバとして機能し、ネットワーク接続のインターネッ

ト側のアドレスを提供します。隠蔽されたサブネットの場合は、サブネット内の任意の公開ホストが DNS サーバとして機能し、各ホストとルータのアドレスを提供します。

DNSMX (Mail eXchanger) のレコードを設定し、各サイトのメール処理を担当するホストの名前を宣言します。このホストは、ファイアウォール・ホストでも他のホストでもかまいません。

ファイアウォール・ホストでは、内部のホスト名やアドレスを公開しないでください。単一のファイアウォール・ホストに FTP や WWW などの複数のサービスを割当てている場合は、CNAME レコードを使ってホスト名にサービスのエリアスを与えてください。後で、サービスを分割して複数のホストに移動するのが簡単になります。

メールの設定のセキュリティに関するガイドライン

ここでは、電子メール・システムを通じてユーザ・サイトが侵害されないように予防する方法について説明します。インターネットの電子メールは SMTP (Simple Mail Transfer Protocol) に基づいています。通常、SMTP を実現するプログラムは `sendmail` と呼ばれます。`sendmail` は大きくて複雑なプログラムであり、セキュリティが侵害されやすい傾向があります。

Sendmail の設定とメールのエリアス

メール・システムの設定は DNS の設定に対応させます。つまり、DNS サーバによって MX (Mail eXchanger) ホストとして宣言されたマシンでは、ユーザのネットワーク向けのメールを受付け、受付けたメールを適切に処理するように、`sendmail` を設定します。通常は、メールをマスター・メール・コンピュータに転送するように設定します。マスター・メール・コンピュータは、ユーザの内部アドレスとその内部アドレスにメールを転送する方法を知っているマシンです。

現行の規定に関する注意 : 通常は、ユーザのネットワークのドメイン名を電子メールのアドレスにします。たとえば、XYZ 社のユーザ harry のドメイン名が XYZ.com である場合、電子メールのアドレスは harry@machine1.XYZ.com ではなく harry@XYZ.com にします。この設定を行うには、`/etc/sendmail.cf` を編集します。『IRIX Admin: Networking and Mail』を参照してください。

サイトの電子メール・アドレスを補強し、他のサイトが各ユーザのメールに応答しやすくするには、上記の規定に準拠してすべてのメール・アドレスを書換えるように、`sendmail` を設定することをお薦めします。

sendmail.cf の設定方法の詳細については、『IRIX Admin: Networking and Mail』を参照してください。

メール・スプールの分割

大量の電子メールがファイアウォール・ホストに殺到すると、ディスクが満杯になり、作業を継続できなくなる可能性があります。この可能性がある場合は、メール・スプールを専用のディスクまたはディスク・パーティションに置いて分割します。分割すると、大量の電子メールが殺到した場合でも、/usr などの重要なシステム・ディスク・パーティションが満杯になるのを防げます。

プロクシ・サーバについて

プロクシ・サーバは、特定のネットワーク・サービスのセキュリティを実現するアプリケーションです。基本的には、特定のアプリケーションのプロトコルを理解することによって、トラフィックを透過的に肩代わりし、プロトコル固有のセキュリティ、ログ処理、認証などを実現するアプリケーション・レベルのゲートウェイです。

ファイアウォールにプロクシ・サーバを設定すると、内部ユーザは Netscape Navigator を通じて World Wide Web にアクセスしたり、ftp を通じて内部ネットワークとインターネットのホスト間でファイルを転送したり、telnet を通じて外部ホストと対話セッションを持つことができます。

プロクシ・サーバとして代表的なのは、サーバ側のプロクシ・サーバと SOCKS プロクシ・サーバの 2 つです。IRIX ファイアウォールにオプションとして提供される Gauntlet で利用できるプロクシ・サーバは、サーバ側だけのアプリケーションを実現します。この場合、サポートされるアプリケーションごとに 1 つのプロクシ・サーバが存在します。SOCKS プロクシ・サーバの場合は、サーバで socksd プロセスを利用し、そのプロセスと通信するすべてのアプリケーションに対して SOCKS として機能することを要求します。つまり SOCKS ライブラリを実装してコンパイルされていることを要求します。たとえば、Netscape Navigator は SOCKS として機能するようにすでに設定されています。

ユーザ独自のプロクシ・サーバの作成方法については、xxv ページの「参考資料」を参照してください。IRIX 用 Gauntlet と Netscape Proxy Server の詳細については、日本シリコングラフィックス株式会社のサポート部門にお問い合わせください。

第 III 部

アカウンティング

第 III 部のセキュリティには次の章が含まれます。

第 6 章
システム監査トレールの管理

第 7 章
システム・アカウンティング

システム監査トレールの管理

システム監査トレール（追跡）機能を使用すると、管理者はシステムの全体的な使用状況の記録を調査できます。システムの使用状況に関する記録から、システム使用に関する一般的な傾向と管理者方針の違反を確認できます。たとえば、システム・リソースを使用しようとして失敗した場合は、それが監査追跡に記録されます。また、他のユーザの所有するファイルへのアクセス試行を繰返したり、ルートのパスワードを推測しようとした場合も、監査追跡に記録されます。サイトの管理者は、監査追跡を通じてシステムのすべての使用状況を監視できます。この章では、以下について説明します。

- 「MAC と DAC について」（160 ページ）
- 「監査プロセスの起動」（161 ページ）
- 「デフォルトの監査」（162 ページ）
- 「監査のカスタマイズ」（163 ページ）
- 「監査データについて」（175 ページ）
- 「セキュリティ違反について」（176 ページ）
- 「監査データのアーカイブ」（183 ページ）

MAC と DAC について

この章では、強制アクセス制御（MAC: Mandatory Access Control）イベントについて触れる場合があります。MAC イベントは、高位の MAC クリアランスによって保護されたファイルにアクセスしようとしたときなどに生成されます。ここで説明する監査システムは、すべての IRIX オペレーティング・システムで生成されるすべてのイベントを監査できます。ただし、MAC を利用できるのは、オプションの Trusted IRIX/B オペレーティング・システムだけです。標準の IRIX では MAC 監査イベントは生成されません。Trusted IRIX/B をインストールしている場合は、その特別なセキュリティ機能に関するマニュアルが別に付属しています。標準 IRIX のユーザは、MAC、ラベル、および `dbedit`、`chlabel`、`newlabel` の各コマンドについての説明を無視しても問題はありません。システムが Trusted IRIX/B を実行しているかどうかを知るには、`versions` コマンドを使って `trix_eoe` 製品のイメージがインストールされているかを確認します。

また、システムが Trusted IRIX/B を実行しているかどうかを知るには、`sysconf` コマンドを使って MAC が設定されているかどうかを確認するという方法もあります。

```
sysconf MAC
1
```

標準 IRIX と Trusted IRIX/B で `uname -a` を実行すると同じような応答が表示されます。

`IRIX64` *SystemName* `6.5 10301649 IP27`

任意アクセス制御（DAC: Discretionary Access Control）は、UNIX の標準ファイル・パーミッション・システムに対して監査サブシステムが使用する用語です。IRIX は、UNIX ベースのすべてのオペレーティング・システムに共通する標準パーミッション・システムを使用しています。

監査プロセスの起動

監査サブシステムは IRIX オペレーティング・システムのメディアに含まれていますが、デフォルトではインストールされません。監査プロセスを有効にするには、Inst ユーティリティを使用し、配布メディアから *eoe.sw.audit* サブシステム・パッケージをインストールします。Inst の詳細については、『IRIX Admin: Software Installation and Licensing』を参照してください。このパッケージをインストールした後で、システムを再起動し、`chkconfig` ユーティリティを使用して監査プロセスを有効にします。`chkconfig` の使用方法の詳細については、chkconfig(1M) マン・ページを参照してください。簡単に説明すると、設定可能なオプションのリストが表示され、各オプションのオンとオフを切替えられるようになっています。リストはアルファベット順になっています。

例として、`chkconfig` で表示される監査オプション・リストの一部を示します。

```
Flag                State
====                =====
audit               off
automount           on
windowsystem        on
xdm                 off
```

次のコマンドを入力すると、監査プロセスが起動します。

```
chkconfig audit on
```

システムはデフォルトで設定されている監査イベントに関する監査データの収集を開始します。次に、デフォルトの監査イベントのリストを示し、説明を加えます。

デフォルトの監査

デフォルトの監査環境は、IRIX のインストール時にすでに設定されています。デフォルトの監査環境を管理するのに特別な操作は不要です。IRIX には、デフォルトで `/etc/init.d/audit` ファイルが格納されています。このファイルには、監査追跡に必要なデフォルトの初期設定が入っています。デフォルト設定では、最小限のディスク領域を使用してシステムの使用状況に関する完全な記録が作成されます。表 6-1 に、デフォルトで監査されるすべてのイベント・タイプを示します。このリストでは、個々のイベント・タイプについては説明していません。イベント・タイプの詳細については、165 ページの「監査可能なイベント」を参照してください。

表 6-1 デフォルトで監査されるイベント

デフォルト監査イベント		
sat_access_denied	sat_domainname_set	sat_mount
sat_ae_custom	sat_exec	sat_open
sat_ae_dbedit	sat_exit	sat_proc_attr_write
sat_ae_identity	sat_fchdir	sat_proc_attr_write
sat_ae_mount	sat_fd_attr_write	sat_proc_attr_write2
sat_bsdipc_create	sat_file_attr_write	sat_proc_read
sat_bsdipc_create_pair	sat_file_crt_del	sat_proc_write
sat_bsdipc_expl_addr	sat_file_crt_del2	sat_svipc_change
sat_bsdipc_mac_change	sat_file_write	sat_svipc_create
sat_bsdipc_shutdown	sat_fork	sat_svipc_remove
sat_chdir	sat_hostid_set	sat_sysacct
sat_chroot	sat_hostname_set	sat_tty_setlabel
sat_clock_set		

監査のカスタマイズ

監査システムをインストールすると、使用する監査のレベルとタイプを選択できます。インストール時には、上に示したデフォルトの監査環境が作成されます。通常は、デフォルトの監査環境で十分です。ただし、システム監査トレールは、`sat_select` ユーティリティと `satconfig` ユーティリティを使用して、いつでも設定を変更できます。

`satconfig` ユーティリティはグラフィックス・システム用です。便利なグラフィカル・インタフェースを通じて監査可能なイベント・タイプごとにオン／オフに切替えられます。`sat_select` ユーティリティは、`satconfig` ユーティリティを使用しないサーバのユーザなどが使用します。各ユーティリティの詳細については、169 ページの「satconfig について」と 170 ページの「sat_select について」を参照してください。

監査対象の活動

システムでのすべての活動を監査することも、ファイルの削除やアクセスの拒否など、特定の活動を監査することもできます。監査追跡では、ユーザ ID（UID）番号でユーザを追跡します。すべての監査対象の活動は、その活動を行ったユーザの UID と関連付けられます。システム監査トレールでは、su コマンドによって有効な UID が変更されることがあっても、システム監査トレールの ID（SAT ID）は変更されません。ユーザがログインした後に行うすべての活動は、元のログイン UID で監査されます。

監査対象の活動の種類を選択する場合は、いくつかの監査オプションを追加できます。たとえば、ファイルの削除を監視するには、次の 2 つの場合に分けて監査記録を作成できます。

- 活動の失敗時（sat_access_denied, sat_access_failed）
- 活動の成功時（sat_file_crt_del, sat_file_crt_del2）

信頼できるコンピュータ・システムでは、各種の活動が実施されます。活動を大きく分類すると、ログイン試行、ファイル操作、プリンタやテープ・ドライブなどの装置の使用、管理活動に分かれます。この大きな分類を、さらに細かく分類して監査することもできます。

監査対象の活動を以下にリストアップします。活動ごとに簡単な定義と監査可能なイベント・タイプの例を示します。重要な活動については、常に監査が必要です。

- ログインとログアウト（sat_ae_identity）

  すべてのログイン試行は、成否を問わずに必ず監査します。ユーザがログアウトするときにも監査記録を作成します。

- su (sat_check_priv, sat_ae_identity)

  ユーザが `su` コマンドを実行する場合は、ルート・アカウントや別のユーザ・アカウントなど、特定の管理アカウントを特権ユーザとして使用するかどうかに関係なしに、必ず監査します。特に、失敗した試行については、権限のないアクセスを試行しているおそれがあるので、必ず監査します。

- chlabel と newlabel (file_attr_write, sat_proc_own_attr_write)

  ユーザが Trusted IRIX/B システムで MAC ラベルを変更するごとに、監査記録を作成します。このイベントは、標準 IRIX では生成されません。

- パスワードの変更（sat_ae_identity）

  ユーザがパスワードを変更するごとに、監査記録を作成します。

- 管理活動（sat_ae_mount、sat_clock_set、sat_hostid_set など）

  システム管理に関連するすべての活動は入念に監査します。たとえば、`/etc/fstab` ファイルの編集などです。

- DAC パーミッションの変更（sat_fd_attr_write, sat_file_attr_write）

  ユーザが `chmod` コマンドを実行してファイルの DAC パーミッションを変更する場合、または `chown` を実行してファイルの所有権を変更する場合に監査します。

- ファイルの作成（sat_file_crt_del, sat_file_crt_del2）

  新しいリンク、ファイル、またはディレクトリを作成するごとに監査します。

- ファイルの削除（sat_file_crt_del, sat_file_crt_del2）

  リンク、ファイル、またはディレクトリを削除するごとに監査します。

- プロセスに関する活動（sat_exec, sat_exit, sat_fork）

  新しいプロセスを作成、複製、終了または強制終了するときに監査します。

監査管理者（監査人）は、新しい sat_select コマンドを入力して監査対象のイベントを変更できます。cron ユーティリティを使用して sat_select を定期的に実行すると、時刻別に監査対象のイベント・タイプを変更できます。

各自のニーズに応じた監査を指定するには、sat_select ユーティリティまたは satconfig ユーティリティを使用します。

監査可能なイベント

次に、すべての監査可能なイベント・タイプを示します。

sat_access_denied
: ファイルへのアクセスまたはパスの一部の要素が、MAC または DAC のパーミッションの強制によって拒否されました。

sat_access_failed
: 指定されたパスが存在しないので、ファイルへのアクセスが拒否されました。

sat_chdir
: 現在の作業ディレクトリが chdir によって変更されました。

sat_chroot
: 現在のルート・ディレクトリが chroot によって変更されました。

sat_open
: ファイルが書込み権を使用して開かれました。

sat_open_ro
: ファイルが読取り専用として開かれました。

sat_read_symlink
: シンボリック・リンクの内容が readlink で読取られました。リンクがポイントするファイルにアクセスする方法はありません。

sat_file_crt_del
: ファイルが追加されたか、ディレクトリから削除されました。

sat_file_crt_del2
: sat_file_crt_del と同じですが、2 つのファイル（またはリンク）が削除されたことを知らせます。

sat_file_write
: ファイル内のデータが truncate によって変更されました。

sat_mount
: ファイルシステムがマウントまたはアンマウントされました。

sat_file_attr_read
: ファイルの属性が `stat` によって読取られました。

sat_file_attr_write
: ファイルの属性が `chmod` によって書込まれました。

sat_exec
: 新しいプロセスが `exec` によって起動されました。

sat_sysacct
: システム・アカウンティングがオンまたはオフになりました。

sat_fchdir
: ユーザが、現在の作業ディレクトリを、指定されたオープン記述子がポイントするディレクトリに変更しました。

sat_fd_read
: `read` によってファイル記述子から情報が読取られました。

sat_fd_read2
: sat_fd_read と同じですが、複数のファイル記述子があります。

sat_tty_setlabel
: ユーザが `ioctl` を通じてポートのラベルを設定しました。

sat_fd_write
: ユーザがファイル記述子の変更を確定しました。

sat_fd_attr_write
: ユーザが `fchmod` を使用して、指定されたファイル記述子がポイントするファイルの属性を変更しました。

sat_pipe
: ユーザが名前のないパイプを作成しました。

sat_dup
: ユーザがファイル記述子をコピーしました。

sat_close
: ユーザがファイル記述子を終了しました。

sat_proc_read
: ユーザが `ptrace` を使用してプロセスのアドレス領域から読取りました。

sat_proc_write
: ユーザが `ptrace` を使用してプロセスのアドレス領域の変更を確定しました。

sat_proc_attr_read
: ユーザがプロセスの属性を読取りました。

sat_proc_attr_write
: ユーザがプロセスの属性の変更を確定しました。

sat_fork
: ユーザがカレント・プロセスをコピーし、新しいプロセスを作成しました。

sat_exit　　ユーザがカレント・プロセスを終了しました。

sat_proc_own_attr_write
　　プロセスの属性が変更されました。

sat_clock_set
　　システム・クロックが設定されました。

sat_hostname_set
　　ホスト名が設定されました。

sat_domainname_set
　　ドメイン名が設定されました。

sat_hostid_set
　　ホスト ID が設定されました。

sat_check_priv
　　特権ユーザの権限を要する操作が実行されました。

sat_control　　`sat_select` コマンドが使用されました。

sat_svipc_access
　　ユーザが System V IPC データ構造にアクセスしました。

sat_svipc_create
　　ユーザが System V IPC データ構造を作成しました。

sat_svipc_remove
　　ユーザが System V IPC データ構造を削除しました。

sat_svipc_change
　　ユーザが System V IPC データ構造の属性を設定しました。

sat_bsdipc_create
　　ユーザがソケットを作成しました。

sat_bsdipc_create_pair
　　ユーザがソケット対を作成しました。

sat_bsdipc_shutdown
　　ユーザがソケットを停止しました。

sat_bsdipc_mac_change
: ユーザがソケットの MAC ラベルを変更しました。

sat_bsdipc_address
: `accept`、`bind`、または `connect` のシステム・コールを通じてネットワーク・アドレスが明示的に使用されました。

sat_bsdipc_resvport
: 予約ポートの結合が成功しました。

sat_bsdipc_deliver
: パケットがソケットに転送されました。

sat_bsdipc_cantfind
: ソケットを検出できないので、パケットが転送されませんでした。

sat_bsdipc_snoop_ok
: パケットが raw (snoop(1M)) ソケットに転送されました。

sat_bsdipc_snoop_fail
: MAC 方針により阻止されたので、パケットがロー・ソケットに転送されませんでした。

sat_bsdipc_rx_ok
: パケットがインタフェースに受信されました。

sat_bsdipc_rx_range
: MAC 違反によりパケットがインタフェースの許容ラベル範囲内に転送されませんでした。

sat_bsdipc_rx_missing
: パケットを受信したインタフェースの MAC ラベルがないか、破損しています。

sat_bsdipc_tx_ok
: パケットがインタフェースに送られました。

sat_bsdipc_tx_range
: MAC 違反によりパケットが送られませんでした。

sat_bsdipc_tx_toobig
: MAC ラベルが大きすぎて IP ヘッダに入らないので、パケットが送られませんでした。

sat_bsdipc_if_config
: インタフェース構造の属性が変更されました。

sat_bsdipc_if_invalid
: MAC 特権がないので、MAC ラベル変更の試みが許可されませんでした。

sat_bsdipc_if_setlabel
: インタフェース構造の MAC ラベルが変更されました。

すべての `sat_ae` イベントはアプリケーションの監査に使用されます。これは、特権プログラムがカーネルではなく監査記録を作成したことを意味します。

sat_ae_identity
: ログインまたはログアウトに関連するイベントが発生しました。

sat_ae_dbedit
: `dbedit` ユーティリティを使用してファイルが変更されました。このユーティリティを使用できるのは、オプションの Trusted IRIX/B を実行している場合だけです。

sat_ae_mount
: NFS ファイルシステムがマウントされました。

sat_ae_custom
: アプリケーション定義のイベントが発生しました。アプリケーション開発者は、このイベントが生成されるようにアプリケーションを設計できます。

satconfig について

`satconfig` は、システムで監査するイベントを厳密に設定するためのグラフィカル・ユーティリティです。どのユーザも `satconfig` を呼出せますが、監査環境を実際に変更できるのは特権ユーザだけです。

監査追跡を初めて使用する場合は、デフォルトの監査イベントが用意されています。デフォルト設定は `satconfig` で変更できます。しかし、`satconfig` ウィンドウにある「edit」という名前のプルダウン・メニューを使用すると、現在の監査環境をデフォルトの環境にいつでも戻せます。メニューには、元の SGI 監査のデフォルト設定、ローカルのデフォルト設定、すべてのイベント・タイプの選択、すべてのイベント・タイプの選択解除、現在のイベント設定などの項目が

あります。現在のイベント設定を選択すると、最後に保存した監査環境が復元されます。ローカルのデフォルト環境は、ユーザが必要なイベント・タイプを組合わせて構成できます。ローカルのデフォルト環境を作成するには、171 ページの「監査環境の保存と検索」で説明する手順に従ってください。

satconfig の使い方

`satconfig` を呼出すと、画面に新しいウィンドウが開きます。ウィンドウの中央には使用可能なイベント・タイプがリスト表示されます。各イベント・タイプ名の隣にはボタンがあり、監査のオン／オフを設定できます。ボタンをオンにしたイベント・タイプは監査されます。ボタンをオフにしたイベント・タイプは監査されません。イベント・タイプの監査のオン／オフを切替えるには、マウスの左ボタンを使用します。

`satconfig` 画面の下部には、［*Apply*］、［*Revert*］、［*Quit*］の 3 つのボタンがあります。監査環境を設定する場合は、マウスの左ボタンで［*Apply*］をクリックします。途中で設定を取消す場合は、［Revert］ボタンをクリックします。各イベント・タイプのボタンが元の設定に戻ります。［*Quit*］ボタンは、`satconfig` ウィンドウを閉じます。選択した内容を適用せずにウィンドウを閉じようとすると、変更を適用せずに破棄して終了してよいかどうかを `satconfig` は確認してきます。

sat_select について

`sat_select` ユーティリティは、監査イベント・タイプの設定を変更する文字ベースのプログラムです。`sat_select` ユーティリティを使用して、ローカルのデフォルト監査環境を変更したり、プリセットされたイベント・タイプの設定をファイルから読取ることもできます。このように、状況に応じた各種のプリセットされた監査環境をファイルに用意し、環境を簡単に切替えることができます。グラフィカル・システムには、`satconfig` を使用して監査イベント・タイプを設定することをお薦めします。非グラフィカル・システムまたは大規模なファイル指向の変更には、`sat_select` を使用します。

sat_selectの使い方

`sat_select` の使い方の詳細については、sat_select(1M) マン・ページを参照してください。通常使用される構文は、次の 2 つです。

`sat_select` *-on event*

`sat_select` *-off event*

`sat_sel`*ect -on event* は、システム監査トレールに、特定のイベントの記録を収集することを指示します。*event* の文字列に「all」を指定すると、すべてのイベント・タイプが収集されます。

`sat_select` *-off event* は、システム監査トレールに、特定のイベント・タイプの情報を収集しないことを指示します。*event* の文字列に “all” を指定すると、すべてのイベント・タイプが無視されます。

`sat_select` に引数を指定しないで実行すると、現在収集されている監査イベントがリスト表示されます。続いて `sat_select` プログラムを実行すると、結果が累積されます。**-h** オプションを指定すると、ヘルプが表示されます。

監査環境の保存と検索

監査環境を変更する場合は、`sat_select` コマンドを使用します。一時的に変更する場合は、簡単に元に戻せるように現在の監査環境を保存すると便利です。監査環境を保存するには、次のコマンドを使用します。

```
sat_select -out > /etc/config/sat_select.options
```

保存した状態に監査環境を戻すには、次のコマンドを使用します。

```
sat_select 'cat /etc/config/sat_select.options'
```

この例の単一引用符 (') は必要なので、省略しないでください。

複数の監査環境を別々のファイル名で保存できます。保存する環境のファイル名を上の例に挿入します。`/etc/config/sat_select.options` ファイルは、起動時に読取られるデフォルトの監査環境ファイルです。Trusted IRIX/B を実行している場合は、

/etc/config/sat_select.options ファイルに dblow というラベルを付けます。さらに、オペレーティング・システムの種類に関係なく、DAC のファイル・パーミッションはルートだけに与えます。

監査ファイルの格納

監査記録ファイルの記憶場所も設定できます。監査記録の保存先として、磁気テープなどの任意のメディアを指定できます。satd は、引数 *path* で指定されたディレクトリまたはファイルに入力データを保存します。

satd の **-f** オプションは、出力パスを指定します。出力パスはファイルでもディレクトリでもかまいません。出力パスが特定のファイル名の場合、satd はそのファイルに対して書込みを行います。出力パスがディレクトリの場合、satd は一意な名前のファイルを作成して、そのディレクトリに保存します。ファイルには作成時を示す名前が付けられます。たとえば、sat_9101231636 ファイルは、1991 年 1 月 23 日の午後 4 時 36 分に作成されたことを示します。出力パスが特定のファイル名である場合、satd はそのファイルに書込みます。satd コマンド行には、複数の出力パスを指定できます。各パス名の前に **-f** を付けます。または、各パス名をカンマ (,) で区切り、空白スペースは入れません。コマンド行に指定した出力パスの全体をパス・リストと呼びます。パス・リストを含むコマンド行の例を次に示します。

```
satd -f /sat1 -f /sat2 -f /sat3 -f /dev/null
satd -f /sat1,/sat2,/sat3,/dev/null
```

-f フラグの後に出力パスを指定しないと、監査追跡記録は保存されず、システムは停止します。また、コマンド行パラメータとして無効なパスを指定すると、警告が表示され、そのパスはパス・リストから削除されます。satd は、残りの有効なパスだけを使用して動作を続けます。指定したパスが存在しない場合、satd はその名前のファイルを作成します。

ファイルまたはディレクトリは、その常駐先のファイルシステムに使用可能な領域がなくなると、満杯になります。ディレクトリを出力パスとして指定すると、そのディレクトリの下に監査ファイルが作成されます。監査ファイルが内部で指定した最大サイズに達すると、その監査ファイルは閉じられ、新しい監査ファイルが同じディレクトリの下に作成されます。

1 つの出力パスが満杯になると、satd はその出力パスを満杯になっていないパスと交換します。この交換方法は、**-r** オプションで設定できます。satd は、kill コマンドなどで送られた SIGHUP シグナルを受取った場合にも出力パスを交換します。

出力パスがほとんど満杯になると、システム・コンソールに警告が表示され、監査追跡をテープに移すように管理者に知らせます。すべての出力パスが満杯になると、システム状態はシングル・ユーザ・モードに移行し、まもなくシステムが停止します。

システム状態の突然の移行によるデータ損失を防ぐために、satd は動作開始時に `/satd.reserve` というファイルを作成します。これは 250,000 バイトのファイルです。すべての出力パスが満杯になると、`satd` は `satd.reserve` ファイルを削除して 250,000 バイトを解放し、システムがシングル・ユーザ・モードに移行している間に監査記録を保存できるようにします。システムが停止しようとすると、`satd` は `/satd.reserve-n` という一連のファイルに監査記録を保存します。`/satd.reserve-n` の *n* は 0 から始まります。`satd` は、この操作中に `wall` 経由ですべてのユーザに警告を表示し、10 秒後にシステムが停止することを知らせます。

`/satd.emergency-0` ファイルがすでに存在する場合、`satd` は使用可能な最初のファイル名（通常は `/satd.emergency-1`）に移動します。この移動が起こるのを防ぐため、`/satd.emergency` ファイルの有無を確認する警告が起動時に表示されます。

監査デーモンの詳細については、audit(1M)、satd(1M)、audit_filters(5) の各マン・ページと `/etc/init.d/audit` のコメントを参照してください。

特定のユーザの監査

特定のユーザの監査記録を調べる場合もあります。たとえば、ユーザに関するシステム・セキュリティの違反歴や、プロジェクトから外れるユーザの活動記録を確認する場合があります。

セキュリティ違反を確認するための監査

ユーザのセキュリティ違反の試行を監査して確認するには、次のコマンド行を使用します。

`sat_reduce -P` *satfile* **`|`** **`sat_summarize -u`** *user_name*

このコマンド行は、違反の試行を示す監査記録だけを選択します。`sat_reduce` の **`-P`** フラグが違反の試行を選択します。`sat_summarize` コマンドの **`-u`** フラグは、ユーザが生成した記録の数を表示します。

セキュリティ違反のすべての試行が、ユーザの悪意に基づくとはかぎりません。違反の試行記録の大半は、通常の作業中に生成されます。監査管理者として調査する必要があるのは、通常の作業に関係ない、給与や雇用に関する情報などへのアクセスが繰返されている場合です。

ユーザの活動の監査

従業員がプロジェクトから外れる場合、監査管理者は、この従業員が使用していたすべてのファイルを確認し、ファイルとディレクトリを後任の所有者に確実に引継ぎます。

上のコマンド行はユーザの活動の概要を示すだけですが、次のコマンド行を使用すると、ユーザの活動を詳しく調べることができます。

sat_reduce -u *user_name satfile* **| sat_interpret | more**

sat_reduce コマンドは、ユーザが生成したすべての監査記録を選択します。sat_interpret コマンドは、選択された記録を読取り可能な形式に変換します。sat_interpret の出力はかなり大きくなります。出力がファイルに収まらない場合は、出力先として画面を指定し、more などの画面ページング・プログラムで表示します。

この 2 つのコマンド行を使用すると、システムでのユーザの活動を確認し、その活動の目的をほぼ特定できます。すべてのセキュリティ違反とファイルまたはリソースへのアクセスの記録を読取り可能な形式で作成することもできます。

ファイルの監査

特定のファイルに関するすべての監査記録を調べる場合もあります。たとえば、重要なファイルが変更された場合は、ファイルを変更したユーザを確認します。機密ファイルに対するすべてのアクセス記録が必要になる場合もあります。ファイルが開かれるたびに記録を取得するには、監査デーモンが sat_open イベントと sat_open_ro イベントを記録していることを確認します。この確認を行うには、sat_select コマンドを使用します。この 2 つのイベントの監査ログを検索するには、次のコマンド行を使用します。

sat_reduce -e sat_open -e sat_open_ro *satfile* **|**
sat_interpret | grep *filename*

Trusted IRIX/B でのラベルの監査

Trusted IRIX/B を使用している場合、システムはすべてのファイルとプロセスの強制アクセス制御 (MAC: Mandatory Access Control) ラベルをサポートします。ここでは、特定のセキュリティ・ラベルの監査追跡を調べる方法について説明します。

標準 IRIX を使用している場合、システムは MAC ラベルをサポートしていないので、ラベルに関するイベントの監査追跡は読取れません。

Trusted IRIX/B では、設定できるラベル数が多いので、サイトでのプロジェクトまたはプロジェクトの各部ごとに専用ラベルを使用できます。したがって、ラベルごとに監査して、その活動記録を作成することもできます。ラベルでの活動のログを作成するには、次のコマンドを使用します。

`sat_reduce -l` *label* `satfile`

このコマンドは、特定のラベルに関する監査記録だけを選択します。次のコマンドの構文を使用すると、複数のラベルを選択して記録を作成できます。

`sat_reduce -l` *label* **`-l`** *label2 satfile*

`sat_reduce` から得た出力は、必要に応じて `sat_interpret` または `sat_summarize` を使用して表示します。

監査データについて

活発なシステムの完全な監査追跡は、膨大すぎて 1 人で読みこなせない場合があります。監査追跡の中でトラブルを示すエントリはごくまれです。監査追跡の中から方針違反の例を直接見つけ出すのは、無駄が多すぎます。監査データを減らして解釈しやすくする方法があります。`sat_reduce`、`sat_interpret`、`sat_summarize` の各コマンドを使用すると、余分な情報を削除し、監査履歴を簡潔な形に変換できます。各コマンドの使い方の詳細については、それぞれのマン・ページを参照してください。

監査データを減らして変換すると、各記録は次のようになります。

```
Event type = sat_ae_identity
Outcome = Failure
```

```
Sequence number = 5
Time of event = Mon Mar 11 12:46:13.33 PST 1991
System call = syssgi,SGI_SATWRITE
Error status = 0 (No error)
SAT ID = anamaria
Identity event = LOGIN|-|/dev/ttyq4|anamaria|That user gave an invalid label.
```

`sat_summarize` コマンドは、監査追跡の記録のタイプとタイプ別の記録数を簡単なリストとして示します。記録をすばやく検査し、システム使用の傾向や一貫した問題を特定するのに便利なツールです。

監査記録内のファイル・パス名は、システムのシェルが通常使用するパス名とは異なります。監査記録はセキュリティを目的とした正確な記録であるため、監査ログには、通常隠されるようなパス名の多くの属性が明示されます。たとえば、ダブル・スラッシュ（//）は、ディレクトリ・レベルのクロスを表します。通常、シェルではシングル・スラッシュ (/) で表されます。スラッシュと感嘆符（/!）は、ファイルシステムのマウント・ポイントのクロスを表します。スラッシュとアットマーク記号（/@）は、シンボリック・リンクにパスが続くことを表します。Trusted IRIX/B を実行している場合は、スラッシュと不等記号（/>）が表示されることもありますが、これはクロスされているディレクトリ・ラベルがマルチレベル・ディレクトリであることを意味します。この表記法は `egrep` でサポートされているので、この形式で表記したパス名を正規表現の検索に使用することもできます。次に監査記録パス名の例を 2 つ示します。

```
/usr/!orange2/@/fri//usr//src//lib//libmls//libmls.a
```

```
/usr/!tmp/>L_e//sat//sat_9012280805
```

システムは、監査データをシステムのファイルに格納します。各ファイルはファイルの開始日時、マシン名、ホスト ID で始まり、終了日時で終わります。停電などでシステムが中断されると、そのときに使用していた監査ファイルは終了エントリを持ちません。また、ファイルが管理可能な指定のサイズに達すると、監査デーモンが自動的にそのファイルを閉じて、別のファイルを開きます。システムが起動した場合は、新しいファイルが必ず開始されます。各ファイルとその形式については、satd(1M) マン・ページを参照してください。

セキュリティ違反について

監査追跡のほとんどの記録は、ユーザの通常の活動結果を示します。システム・セキュリティ違反の記録を特定するような自動化ツールはありません。しかし、管理者は一般的な規則を適用し

てセキュリティ方針の悪用または違反を検出できます。ここに示す規則群は、すべてを網羅しているわけではなく、すべてに通用するわけでもありません。各管理者は、サイト固有の事情に応じて規則群をカスタマイズする必要があります。

部外者による使用と悪用

悪用の中でも特に注意が必要なのは部外者による侵入です。ただし、この種の悪用には独特の監査記録パターンがあるので、簡単に検出できます。監査システムで検出できる部外者による悪用のパターンを、以下に紹介します。ただし、この種のパターンは許可ユーザによる操作ミスや誤解から生じることもあります。

システムへの無許可侵入の試行

無許可侵入のすべての試行は、sat_ae_identity イベント・タイプの監査記録に入ります。この種の記録を収集して表示するには、sat_select、sat_reduce、sat_interpret を使用します。この種のイベントを変換して出力すると、侵入の試行を示すテキスト文字列が得られます。外部からの侵入者は、許可ユーザよりもログイン試行を失敗する回数が多いのが特徴です。

無許可侵入の試行を示す重要なテキスト文字列が 3 つあります。

- ログイン試行の失敗
- 無効ラベルの入力
- デバイスのラベル接続の設定不能

次に、失敗したログイン試行を示す変換済みの監査記録の例を示します。

```
Event type = sat_ae_identity
Outcome = Failure
Sequence number = 1
Time of event = Mon Mar 11 12:45:40.34 PST 1991
System call = syssgi,SGI_SATWRITE
Error status = 0 (No error)
SAT ID = anamaria
Identity event = LOGIN|-|/dev/ttyq4|guest|Unsuccessful login attempt.
```

通常と異なる時間帯または場所からのシステム使用

通常の作業時間外にシステムが使用されたり、端末の物理的なセキュリティを確立しているシステムが、通常の場所以外から使用された場合は、注意を要します。このようなシステム使用のほとんどは正当な理由によるものですが、やはり個々の使用には注意と調査が必要です。外部からのセキュリティ違反は非ピーク時に起こりがちです。セキュリティ違反が当の物理サイト内から起こることはほとんどありません。

通常と異なる時間帯の使用状況を観察するには、次のコマンドを順に実行します。

1. **sat_reduce -a** *start_time* **satfile > /usr/tmp/early+late**
2. **sat_reduce -A** *end_time* **satfile >> /usr/tmp/early+late**
3. **sat_reduce -U root -U sys -U daemon -U adm -U lp /usr/tmp/early+late > /usr/tmp/e+l_ordusers**
4. **sat_interpret /usr/tmp/e+l_ordusers | more**

各ユーザに端末を割当て、端末ごとに物理的なセキュリティ対策を確立しているサイトでは、通常と異なる場所からのログインを監視できます。たとえば、通常はグループ・コンピュータ・ラボで作業しているユーザが個人のオフィスからログインを試みた場合、このイベントは注意に値します。ログイン・イベントのリストを得るには、次のコマンドを入力します。

```
sat_reduce -e sat_ae_identity sat_file | sat_interpret | grep LOGIN
```

ユーザが通常と異なる端末で作業したり、複数の端末で同時にログインしたとしても、セキュリティ違反を意味するとはかぎりません。たとえば、ユーザが誤りを訂正するために、不要になったプロセスを終了させる目的で他の場所から明示的にログインする場合もあります。注意が必要なのは、ユーザによる正当なログインが重複した場合ではなく、侵入者による不当なログインが行われた場合です。

ローカル・ネットワーク外のマシンとの接続

信頼できるローカル・ネットワーク外のマシンと接続する場合は、監査記録を必ず作成します。外部ホストとの接続には注意が必要ですが、すべての接続がセキュリティ違反をつながるとはかぎりません。トロイの木馬プログラムの中には、後になってシステムを外部と接続するように設定されているものがあるので、注意が必要です。外部ネットワークとの接続を確認するには、次のコマンド・シーケンスを使用します。

1. **sat_reduce -e sat_bsdipc_addr** *satfile* **>** /usr/tmp/connect
2. **sat_interpret** /usr/tmp/connect **>** /usr/tmp/connect.int
3. **grep -n "Remote host"** /usr/tmp/connect.int

このコマンド・シーケンスは、各ネットワーキング・ソフトウェアの実装に応じて異なります。現在のネットワーキング状態を反映するには、必要に応じてコマンドを変更します。たとえば、使用しているソフトウェアが sat_bsdipc_addr と異なる監査イベント・タイプを生成する場合は、そのイベント・タイプを代わりに使用します。

部内者によるシステムの使用と悪用

侵入者による使用と悪用だけでなく、部内者による悪用が起こる場合もあります。その代表的なセキュリティ違反のイベント・タイプについて、以下に説明します。セキュリティに違反する許可ユーザは信頼できない、または悪意を抱いていると決めつけるのは非生産的です。ユーザがシステム・セキュリティに違反する原因のほとんどは、管理者による作業環境の整備不足にあります。ユーザの関心は各自の業務遂行にあり、必要なツールを得るためにコンピュータ・システムを手直しすることにはありません。したがって、明らかに必然性のないセキュリティ違反が繰返されないかぎり、セキュリティに違反したユーザを疑ってはいけません。

部内者によるファイル・パーミッションの違反

システムはファイルまたはリソースへのアクセスが拒否されるごとにそれを記録しますが、その監査記録がセキュリティ違反を意味することはほとんどありません。アプリケーションやユーティリティの多くは、標準操作の一環としてアクセス拒否を行います。アクセス拒否のイベントは必ず記録されますが、それが違反を意味することはまれです。たとえば、ライブラリ関数の **getutent** は、/etc/utmp を常に開いて読み書きしようとします。この操作に失敗すると、**getutent** は再試行を行い、読取り専用アクセスを要求します。/etc/utmp のパーミッションは、このファイルを開いて読み書きすることを、ルート以外のすべてのユーザに禁じています。特権のないユーザが getutent() を呼出すプログラムを実行すると、sat_access_denied の記録が作成されます。監査追跡には、この記録に続けて sat_open_ro の記録が作成され、アクセスが許可されたことを示します。この例でもわかるように、アクセス拒否は必ずしもセキュリティ違反を意味しません。

sat_access_failed イベントは、拒否イベントと混同される場合がありますが、まったく異なります。このイベントには、アクセス拒否ほどの不安もありません。ユーザが /bin/csh などの対話シェルにコマンドを入力すると、シェルはそのコマンドをユーザの検索パスの各ディレクトリで実行しようとします。そのコマンドが実際に入っているディレクトリが見つかるまで、何度も実行に失敗します。たとえば、ユーザが xterm を入力したとします。ユーザのパス変数は次のとおりとします。

```
/bin:/usr/bin:/usr/sbin:/usr/local/bin:/usr/bin/X11:~/bin
```

コマンドが見つかって実行されるまで、パスの各ディレクトリに対して sat_access_failed が作成されます。この例では、存在しないプログラム /bin/xterm、/usr/bin/xterm、/usr/sbin/xterm、/usr/local/bin/xterm のそれぞれに対して失敗したアクセスが記録され、実際に存在する /usr/bin/X11/xterm に対しては成功したアクセスの記録として sat_file_exec が作成されます。

部内者によるルート特権の乱用

変換済みの各監査記録には、キーワード **Outcome** で始まる行が含まれています。このキーワードに続くフィールドは、**Success**、**Failure**、または **Success due to privilege** のいずれかになります。最後のケースは、特権ユーザの権限が必要とされたため、ユーザによるシステム・コールに失敗していたことを示します。これは必ずしもセキュリティ違反またはルート特権の乱用を意味しません。通常のユーザが実行したプログラムにルート特権を使用するコードが含まれていると、必ず **Success due to privilege** という結果になります。その代表例は passwd プログラムです。通常のユーザが自分のアカウントのパスワードを変更するだけで、この種の記録が作成されます。

SAT ID または Effective ID のフィールドが「User ID」フィールドと異なる場合は注意が必要です。フィールドの相違は、ユーザが /bin/su を実行してルート特権を得た場合、またはセッションの権限レベルを昇格させた場合に起こります。通常は、セキュリティ違反を意味しません。/bin/su コマンドの実行が成功するには、ルート・パスワードが必要だからです。

ただし、特権ユーザの権限の使用は必ず監査追跡で調べます。たとえば、ユーザが自分のログイン・セッションをルートに昇格させた場合は、そのユーザがルート・パスワードを知る権限を持っていることを確認します。権限を持っていない場合は、本当に /bin/su コマンドを実行したかどうか、またはセッションの権限を昇格させる手段としてトロイの木馬の setuid シェル・コマンドを使用していないかどうかを調べます。

ユーザが /bin/su を実行して自分のログイン・セッションの権限を昇格させた場合、監査管理者はユーザが昇格させた権限を使用して行った活動を必ず調べます。

特定の部内者による活動

サイトの管理者が、特定のユーザを正式に調査する場合もあります。対象となるのは、試用期間中のユーザや不満を持って退社したユーザです。監査管理者は、sat_reduce コマンドを通して監査追跡に次のように指示するだけで、そのユーザの活動記録を調査できます。

1. **sat_reduce -u** *jeff* **< satfile >** /tmp/sat.jeff
2. **sat_interpret** /tmp/sat.jeff **| more**

ユーザを、この種の調査対象にすることは特別な場合にかぎります。この機能は、相手の個人を尊重し、注意して使用します。

特定のファイルまたはリソースへのアクセス

特定のファイルまたはリソースを調査する場合があります。たとえば、情報が漏洩し、その漏洩経路を調べる場合です。侵入者を罠にかけるために特別なファイルまたはリソースを作成する場合もあります。いずれの場合にも、監査管理者は該当のファイルまたはリソースを入念に調べます。

sat_reduce -n interesting_file **-e sat_open -e sat_open_ro** sat_filename **| sat_interpret**

システム管理の適否について

管理者またはルートによる活動が、通常と異なる監査記録を示す場合があります。管理者またはルートの権限は大きいので、セキュリティ違反に相当するような監査記録が作成されることも相応に多くなります。ただし、そのような記録のほとんどは、システムの通常の使用によるものか、単純な操作ミスによるものです。

システムのデータ・ファイルの変更

システムのデータ・ファイルの各変更は、監査管理者による調査対象となります。データ・ファイルは、システムのセキュリティ下にあるというだけでなく、実際にシステム・セキュリティを

定義します。したがって、無許可アクセスはセキュリティの全面的な崩壊につながるおそれがあります。

システムに対するユーザの追加または削除、ファイルとハードウェアのアクセス制御の管理、ネットワークの接続の維持管理など、多くの問題について、サイトごとに独自の方針があります。監査責任者は、サイトの方針を実施し、監査ツールを有効に利用してその実施を徹底する責任があります。

Trusted IRIX/B を実行している場合、システムのデータ・ファイルを変更できるのは専用の編集ツールである `dbedit` だけです。汎用のテキスト・エディタでは変更できません。特権ユーザだけが、`dbedit` ツールを使用し、システムのデータ・ファイルの内容を変更できます。他のエディタをシステムのデータ・ファイルに使用するのは、セキュリティ方針の違反であり、監査管理者の調査対象となります。変換済みの監査追跡に `sat_open` の記録があり、その Actual name フィールドに文字列 /secadm が含まれている場合があります。その場合は、PID と実行中のプログラムの名前を示す Process ID フィールドに、vi、ex、emacs などの汎用テキスト・エディタの名前が指定されていないことを確認します。このフィールドに指定できる名前は dbedit だけです。

システム・プログラムの属性の変更

管理者はシステム・プログラムのパーミッション、所有権、またはラベルを変更してはいけません。管理者がシステム・プログラムの属性を変更しようとした証拠が監査追跡に発見された場合は、その変更の理由を調べて確認します。この場合も、変更の理由は正当であり、管理者を疑う根拠にならない場合がほとんどです。ただし、念のためにシステムのセキュリティ方針を調べ、ユーザも管理者もセキュリティ方針に違反していないことを確認します。

次のコマンドは、この種の問題を示す記録を監査追跡から検索します。

```
sat_reduce -e sat_file_attr_write -e sat_fd_attr_write < satfile
```

変換済みの監査追跡で、Actual name フィールドを含む行を探します。`/bin`、`/sbin`、`/etc`、`/lib`、`/var`、`/usr/bin`、`/usr/lib`、`/usr/share`、`/usr/bsd`、*/usr/sbin*、または */usr/bin/X11* のリソースの属性が変更されていることを示す監査記録については、詳しい調査が必要です。

監査追跡の取扱い

監査追跡にアクセスするのは監査管理者だけにします。他のユーザによる監査追跡の読み書き、ファイルの削除、またはファイル属性の変更は禁止します。監査管理者のアカウント・パスワードを知っている当事者以外のユーザが生成したすべての記録を調べ、監査追跡の情報が保存されている `/var/adm/sat` などのディレクトリのファイルにアクセスした記録がないことを確認します。

監査データのアーカイブ

監査追跡は通常のシステム・ファイルに保存されるので、監査データのアーカイブは、バックアップ・テープと同様に簡単に作成できます。監査追跡のコピーを保存し、同時にディスク領域を節約するには、監査データをアーカイブします。システムへの侵入や損傷の証拠はすぐに見つかるとはかぎりません。セキュリティ違反を究明するには、監査追跡を長期に渡って調査できる方が有利です。`compress` ユーティリティを使用すると、古い監査ファイルのサイズを 80% までに縮小できます。

監査データの削除

監査追跡は通常のシステム・ファイルに保存されているので、アーカイブ作成後は、監査追跡ファイルを削除しても問題ありません。`df` コマンド（空きディスク容量を調べるコマンド）を入力し、監査追跡の保存先のファイルシステムが 90% 以上使用されていることがわかった場合は、古い監査ファイルを削除します。監査ファイルが `/var/adm/sat` に保存されている場合は、次のコマンドを入力します。

```
df -k /var/adm/sat
```

このコマンドで、次のような出力が表示されます。

```
Filesystem  Type kbytes  use   avail %use Mounted on
/dev/root   efs  245916 218694 27222  89%    /
```

この例では、ファイルシステムが 89% 使用されているので、監査管理者は監査追跡ファイルをアーカイブした後、ファイルを削除します。

監査ファイルのオーバーフローについて

監査ファイルが増えすぎないように注意します。監査ファイルが増えすぎて使用可能ディスク領域を使い果たすと、システムは新しい記録の作成を拒否して操作を停止するので、作業も監査記録も失われます。監査ファイルシステムには、常に 10% 以上の空き領域を残しておきます。

システムでは、監査デーモン satd(1M) が常に動作していなければなりません。ディスクの空き領域が 0% まで減少すると、デーモンは監査ファイルに書込めなくなります。監査ファイルに書込めなくなると、デーモンはエラー・メッセージを出力して終了し、システムは実行レベルをシングルユーザ・モードに移行します。この場合は、監査ファイルをアーカイブして削除し、ディスク領域を解放しないと、システムをマルチユーザ・モードに戻せません。システムで `satd` デーモンが強制終了または中断された場合にも、実行レベルはシングルユーザ・モードに移行します。この場合は、システムが再起動すれば、デーモンも再起動します。

監査ファイルのオーバーフローからの回復

ディスクに監査追跡用の領域を作成するには、まず、シングルユーザ・モードでシステムを起動します。シングルユーザ・モードでは監査記録は生成されません。シングルユーザ・モードで、監査ファイルをアーカイブし、ディスクから削除します。ファイルシステムの空き領域が 10% 以上になると、マルチユーザ・モードでシステムを起動できます。

監査ファイルの保存先を / (root) ファイルシステムまたは `/usr` ファイルシステムにしている監査システムでは、そのファイルシステムが満杯になったときに、システムをシングルユーザ・モードに移行しても、古い監査ファイルをアーカイブして削除できません。この場合、古い監査ファイルを削除するには、次の手順に従います。

1. マスターのメディアからシステムを起動し、`inst` ユーティリティを起動します。
2. 「inst」メイン・メニューで「admin」メニューを選択した後、「admin」メニューから `shell` オプションを選択します。シェル・プロンプトが表示されます。

   このシェルから、古い監査ファイルをアーカイブし、ファイルを削除します。inst（`miniroot` とも呼ばれる）シェルの実行中は、システムのルート・ディレクトリが、

   `/`

   ではなく、

   `/root/`

と表示されます。/usr ファイルシステムは、次のように表示されます。

```
/root/usr
```

これは、システムのファイルシステムが Inst ファイルシステムにマウントされているためです。

3. / (root) ファイルシステムと /usr ファイルシステムに空きディスク領域を作成すると、システムを通常どおりに起動できます。それでも問題が再発した場合は、監査ファイルの記憶場所の変更方法について、satd(1M) マン・ページを参照してください。

システム・アカウンティング

IRIX には、システムでの特定の活動を記録するユーティリティがあります。プロセス・アカウンティングとシステム・アカウンティングを行うユーティリティです。この章では、以下について説明します。

- 「プロセス・アカウンティング・システムについて」（188 ページ）では、システムの使用状況を記録するアカウンティング・サブシステムについて説明します。
- 「アカウンティング・ファイルとアカウンティング・ディレクトリ」（190 ページ）では、アカウンティング情報の保存先について説明します。
- 「日別システム・アカウンティングについて」（194 ページ）では、日別システム・アカウンティングを定義し、設定手順を示します。
- 「runacct による日別システム・アカウンティング」（196 ページ）では、`runacct` プログラムの操作方法について説明します。

最初の 4 節では、UNIX System V の標準のアカウンティング手順について説明します。IRIX では拡張アカウンティングも使用できます。詳しくは以下で説明します。

- 規模の大きいコンピュータ・サイトで有用なアカウンティング・サブシステムの詳細については、「IRIX の拡張アカウンティング」（206 ページ）を参照してください。

IRIX で利用できるほかのツールについては、日本シリコングラフィックス株式会社のサポート部門までお問い合わせください。たとえば、オプションの製品である IRIX 用 SHARE II を使用すると、ディスク領域、CPU 権の付与、メモリ（リアルまたは仮想）、プロセス数、プリンタのページ数、端末とモデムの接続時間、ネットワーク・パケットなどのシステム・リソースを管理できます。

プロセス・アカウンティング・システムについて

IRIX プロセス・アカウンティング・システムは、次の情報を提供します。

- ユーザが実行するプログラムの数
- ユーザ・プログラムのサイズと使用時間
- データのスループット（入出力）

以上の情報は、次の用途に利用できます。

- システム・リソースの使用状況と、リソースを不当に占有しているユーザがいないかどうかを調査できます。
- ユーザ別および時刻別に実行されたプロセスのリストを調べて、セキュリティ違反などの重大なシステム・イベントを追跡できます。
- 請求システムを設定し、ログイン・アカウントにシステム・リソースの使用料を請求できます。

以下では、プロセス・アカウンティングの機能、プロセス・アカウンティングのオン／オフを切替える方法、各種ログ・ファイルの参照方法について説明します。

プロセス・アカウンティング・システムの機能

IRIX プロセス・アカウンティング・システムには、次のような機能があります。

- IRIX カーネルは、システムで終了した各プロセスのレコードを `/var/adm/pacct` ファイルに書込みます。このファイルには、`/usr/include/sys/acct.h` で定義されている形式に従って、終了したプロセスごとに 1 つのレコードが記述されます。

  この機能は明示的にオンに設定します。189 ページの「プロセス・アカウンティングの有効化」を参照してください。

- プロセス・アカウンティングをオンにすると、*cron* プログラムは `/var/spool/cron/crontabs/adm` と `/var/spool/crontabs/root` に指定されているアカウンティング・コマンドを実行します。`adm` のコマンドは、月別アカウンティング（`monacct`）、`pacct` ファイルのサイズ確認（`ckpacct`）、プロセスと接続時間の日別アカウンティング（*runacct*）を行います。*root* の crontab ファイルは、*dodisk* プログラムを実行

し、現在のディスク使用状況を知らせます。プロセス・アカウンティングをオンにすると、各コマンドが自動的に実行されます。

- `login` プログラムと `init` プログラムは、`/etc/wtmp` に接続セッションのレコードを書込みます。`wtmp` ファイルがあると、この書込みはデフォルトで実行されます。
- 日付変更、再起動、停止の各レコードは、`acctwtmp` コマンドによって `/etc/utmp` から `/etc/wtmp` にコピーされます。
- `acctwtmp` ユーティリティは、プロセス・アカウンティングをオンにすると、`runacct`、`/usr/lib/acct/startacct`、`/usr/lib/shutacct` によって自動的に呼出されます。
- `acctdusg` と `diskusg` のディスク使用プログラムは、ディスクの使用状況についてログイン別にレポートを作成します。この 2 つのプログラムは `dodisk` スクリプトによって実行されます。詳細については、acct(1M)、acctsh(1M)、および diskusg(1M) マン・ページを参照してください。

XFS の場合は、eoe.sw.quotas サブシステムで設定されるディスク割当てをアカウンティング手段として使用すると、ディスクの使用状況を効果的に調査できます。詳細については、『IRIX Admin: Disks and Filesystems』を参照してください。

プロセス・アカウンティングの有効化

プロセス・アカウンティングをオンにするには、次の手順に従います。

1. システムに *root* でログインします。
2. `eoe.sw.acct` サブシステムがインストールされていることを確認します。インストールされていない場合は、このサブシステムをインストールします。
3. 次のコマンドを入力します。

   ```
   chkconfig acct on
   ```

4. 次のコマンドを入力します。

   ```
   /usr/lib/acct/startup
   ```

   このコマンドによって、カーネルは `/var/adm/pacct` ファイルへの情報の書込みを開始します。

プロセス・アカウンティングは、システムを起動するたびに開始されます。次のメッセージが表示されます。

```
System accounting started
```

プロセス・アカウンティング・ファイル、特に /var/adm/pacct のサイズは大きくなることがあります。サーバでプロセス・アカウンティングをオンにする場合は、ディスクの空き容量に常に注意します。191 ページの「アカウンティング・ファイルのサイズの管理」を参照してください。

プロセス・アカウンティングの無効化

プロセス・アカウンティングをオフにするには、次の手順に従います。

1. *root* でログインします。
2. 次のコマンドを入力します。

   chkconfig acct off

3. 次のコマンドを入力します。

   /usr/lib/acct/shutacct

   このコマンドによって、カーネルは /var/adm/pacct ファイルにアカウンティング・データを書込むのを停止します。これでプロセス・アカウンティングがオフになります。

アカウンティング・ファイルとアカウンティング・ディレクトリ

/usr/lib/acct ディレクトリには、アカウンティング・システムを実行するためのプログラムとシェル・スクリプトがあります。プロセス・アカウンティングは、タスクの実行に /var/adm を使用します。/var/adm には、プロセス・アカウンティングが使用するアクティブなデータを集めたファイルがあります。/var/adm の主なサブディレクトリを次に示します。

- /var/adm/acct/nite には、runacct が毎日繰返し使用するファイルがあります。
- /var/adm/acct/sum には、runacct が更新する累積サマリ・ファイルがあります。
- /var/adm/acct/fiscal には、monacct が作成する定期サマリ・ファイルがあります。

アカウンティング・ファイルのサイズの管理

プロセス・アカウンティング・ファイルとディスク・アカウンティング・ファイルは大きくなることがあります。活発なシステムでは、急速に増大します。

/var/adm/pacct ファイルのサイズを管理するには、cron コマンドで /usr/lib/acct/ckpacct を実行し、ファイルのサイズとファイルシステムの使用可能なディスク領域を調べます。

pacct ファイルのサイズが 1000 ブロック（デフォルト）を超えると、turnacct コマンドが引数 switch を指定して実行されます。この引数により、turnacct は pacct ファイルの新しいバックアップを作成し、既存のバックアップを削除します。さらに、新しい空の pacct ファイルを作成します。結果として、pacct ファイルの情報が占めるディスク領域が 2000 ブロックを超えることはありません。

ファイルシステムの空き領域が 500 ブロック未満になると、ckpacct は turnacct コマンドを引数 off で実行し、プロセス・アカウンティングを自動的にオフにします。ディスクの空き領域が 500 ブロック以上になると、cron が次回に ckpacct を実行したときに、アカウンティングは再びオンになります。

/var/adm ディレクトリのファイル

次に示すのは、/var/adm ディレクトリにあるファイルです。

diskdiag	ディスク・アカウンティング・プログラム実行中の診断出力。
dtmp	acctdusg プログラムの出力。
fee	chargefee プログラムの出力である ASCII tacct レコード。
pacct	アクティブなプロセス・アカウンティング・ファイル。
pacct?	turnacct によって切替えられた（switch された）プロセス・アカウンティング・ファイル。
Spact?.MMDD	runacct 実行中の MMDD 用プロセス・アカウンティング・ファイル。

/var/adm/acct/nite ディレクトリのファイル

次に示すのは、/var/adm/acct/nite ディレクトリにあるファイルです。

active	runacct が進行を記録し、警告とエラー・メッセージを表示するためのファイル。activeMMDD は、runacct がエラーを検出した後は、active と同じになります。
cms	prdaily が使用する ASCII 形式の合計コマンド・サマリ。
ctacct.MMDD	tawcct.h 形式の接続アカウンティング・レコード。
ctmp	acctcon1プログラムの出力である、ctmp.h形式の接続セッションのレコード。
daycms	prdaily が使用する ASCII 形式の日別コマンド・サマリ。
daytacct	tacct.h 形式の 1 日の合計アカウンティング・レコード。
disktacct	dodisk プロシージャによって作成される tacct.h 形式のディスク・アカウンティング・レコード。
fd2log	runacct 実行中の診断出力（cron の項を参照）。
lastdate	日付 +%*m*%*d* 形式で runacct が実行された最後の日。
lock lock1	runacct の連続使用を管理するためのファイル。
lineuse	prdaily が使用する tty 回線の使用状況レポート。
log	acctcon1 の診断出力。
logMMDD	runacct がエラーを検出した後は log と同じ。
reboots	wtmp の開始日と終了日、および再起動のリスト。
statefile	runacct の実行中に現在の状態を記録するファイル。
tmpwtmp	wtmpfix によって修復した wtmp ファイル。
wtmperror	wtmpfix エラー・メッセージ用ファイル。
wtmperrorMMDD	runacct がエラーを検出した後は wtmperror と同じ。
wtmp.MMDD	前日の wtmp ファイル。

/var/adm/acct/sum ディレクトリのファイル

次に示すのは、/var/adm/acct/sum ディレクトリにあるファイルです。

cms	内部サマリ形式の当会計期間の合計コマンド・サマリ・ファイル
cmsprev	最新の更新を行っていないコマンド・サマリ・ファイル。
daycms	内部サマリ形式の前日のコマンド・サマリ・ファイル。
loginlog	lastlogin が作成するファイル。
pact.MMDD	*MMDD* 用のすべての pacct ファイルを連結したファイル。再起動すると、remove プロシージャによって削除されます。
rprtMMDD	prdaily プログラムの出力を保存したファイル。
tacct	当会計期間の累積合計アカウンティング・ファイル。
tacctprev	tacct と同じですが、最新更新を行っていません。
tacctMMDD	*MMDD* 用の合計アカウンティング・ファイル。
wtmp.MMDD	*MMDD* 用の wtmp ファイルのコピーを保存したファイル。再起動すると、remove プロシージャによって削除されます。

/var/adm/acct/fiscal ディレクトリのファイル

次に示すのは、/var/adm/acct/fiscal ディレクトリにあるファイルです。

cms?	内部サマリ形式の *fiscal?* 用の合計コマンド・サマリ・ファイル。
fiscrpt?	*fiscal?* 用の prdaily と同じレポート。
tacct?	*fiscal?* 用の合計アカウンティング・ファイル。

日別システム・アカウンティングについて

IRIXがマルチユーザ･モードになると､/usr/lib/acct/startupが次のように実行されます。

- acctwtmp プログラムがブート・レコードを /etc/wtmp に追加します。このレコードは、システム名をログイン名として、wtmp レコードに記録されます。
- プロセス・アカウンティングが turnacct によって開始されます。これにより、/var/adm/pacct で acct が実行されます。
- remove が実行され、runacct によって sum ディレクトリに保存された pacct ファイルと wtmp ファイルが削除されます。

ckpacct プロシージャが cron によって 1 時間ごとに実行され、/var/adm/pacct ファイルのサイズを調べます。サイズが 1000 ブロック（デフォルト）を超えると、turnacct が引数 switch で実行されます。pacct ファイルを分割しておくと、これらのレコードの処理に失敗したときに runacct を簡単に再起動できます。

chargefee プログラムを使用すると､ファイルのリストアなどに対してユーザに請求できます。このプログラムによって /var/adm/fee に追加されたレコードは、runacct の次回の実行時にピックアップされて処理され､合計アカウンティング･レコードにマージされます。runacct は cron によって毎晩実行されます。このコマンドは、アクティブなアカウンティング・ファイルの /var/adm/pacct､/etc/wtmp､/var/adm/acct/nite/disktacct､/var/adm/fee を処理します。ログイン名ごとにコマンド・サマリと使用状況のサマリも作成します。

shutdown でシステムを停止すると、shutacct シェル・プロシージャが実行されます。このプロシージャは、システムの停止理由を示すレコードを /etc/wtmp に書込み、プロセス・アカウンティングをオフにします。

システム管理者は毎朝最初にシステムを起動した後に /usr/lib/acct/prdaily を実行し､前日のアカウンティング・レポートを確認します。

アカウンティング・システムの設定

システム・アカウンティング・オプションをインストールすると、インプリメントに必要なすべてのファイルとコマンド行が正しく設定されます。ただし、システム設定ファイルの各エントリが正しく設定されていることを確認することもできます。アカウンティング・システムの動作を自動化するには、次のことを確認します。

1. `/etc/init.d/acct` ファイルに次の行があることを確認します。

   ```
   /usr/lib/acct/startup
   /usr/lib/acct/shutacct
   ```

   1 行目はシステムの起動時にプロセス・アカウンティングを開始します。2 行目はシステムの終了時にプロセス・アカウンティングを停止します。

2. 大部分のアカウンティング・システムでは、`cron` が日別アカウンティングを自動的に実行できるように、`/var/spool/cron/crontabs/adm` に次のエントリが設定されていることを確認します。

   ```
   0 4 * * 1-6 if /etc/chkconfig acct; then /usr/lib/acct/runacct 2>
   /var/adm/acct/nite/fd2log; fi
   ```

   ```
   5 * * * 1-6 if /etc/chkconfig acct; then /usr/lib/acct/ckpacct; fi
   ```

   `cron` コマンドは、`crontabs` ファイルで 1 行で記述されます。次のコマンドも `crontabs` ファイルで 1 行で記述されます。このコマンドは `/var/spool/cron/crontabs/root` に設定します。

   ```
   0 2 * * 4 if /etc/chkconfig acct; then /usr/lib/acct/dodisk >
   /var/adm/acct/nite/disklog; fi
   ```

3. アカウンティング・データを月別にマージするには、`/var/spool/cron/crontabs/adm` に次のエントリを追加します。このエントリを追加すると、`monacct` はすべての日別レポートと日計アカウンティング・ファイルを削除し、1 つの月計レポートと 1 つの月計アカウンティング・ファイルを会計用ディレクトリに作成します。

   ```
   0 5 1 * * if /etc/chkconfig acct; then /usr/lib/acct/monacct; fi
   ```

   このコマンドは、ソース・ファイルでは 1 行で記述されます。また、`monacct` のデフォルト設定に従って、現在の日付がファイル名のサフィックスとして使用されます。このエントリは、`runacct` の実行に十分な時間がある場合に実行され、各月の第 1 日目に、1 か月分のデータをまとめた月別アカウンティング・ファイルを作成します。

4. 必要に応じて、adm にアカウントがあるかどうかを確認します。また、PATH シェル変数が /var/adm/.profile で次のように設定されていることを確認します。

   ```
   PATH=/usr/lib/acct:/bin:/usr/bin
   ```

5. 次の 2 つのコマンドを入力し、システム・アカウンティングを起動します。

   ```
   chkconfig acct on
   /usr/lib/acct/startup
   ```

システムを次回に起動したときに、アカウンティングが開始します。

runacct による日別システム・アカウンティング

runacct コマンドは、主要な日別アカウンティング・シェル・プロシージャです。通常、非ピーク時に cron によって起動されます。runacct は接続、使用料、ディスク、プロセス・アカウンティングに関する各ファイルを処理します。請求目的または prdaily に必要な日別サマリ・ファイルと累積サマリ・ファイルも作成します。

runacct のサマリ・ファイル

runacct によって作成される次のファイルは特に重要です。

nite/lineuse
: acctcon によって作成されるファイルであり、wtmp ファイルを読取り、システム上の端末回線ごとの使用統計をまとめます。このレポートは、特に不良回線を見つけるのに役立ちます。ログオフとログインの比率が約 3:1 を超えていると、回線が故障している可能性があります。

nite/daytacct
: tacct.h 形式による前日の合計アカウンティング・ファイルです。

sum/tacct
: 毎日の nite/daytacct の累計は請求目的に役立ちます。毎月または会計期間ごとに、monacct によって再起動されます。

sum/daycms
: acctcms によって作成されるファイルであり、日別コマンド・サマリが記述されています。このファイルの ASCII バージョンは nite/daycms です。

`sum/cms` 毎日のコマンド・サマリの累計です。`monacct` で再起動します。このファイルの ASCII バージョンは `nite/cms` です。

`sum/loginlog`
最後のログイン・シェル・プロシージャによって作成されるファイルです。最後に `login` 名が使用されたときの記録を保持します。

`sum/rprtMMDD`
`runacct` を実行するたびに、`prdaily` によって出力される日別レポートのコピーを保存します。

`runacct` は、エラーが発生した場合にファイルを損傷しないようにします。一連の保護メカニズムとして、エラーを認識し、最適な診断を提供し、最小限の操作で `runacct` を再起動できるように処理を終了させます。その経過を説明するメッセージを `active` ファイルに書込みます。特に明記しないかぎり、`runacct` が使用するファイルは `nite` ディレクトリにあるものとします。`runacct` 実行中のすべての診断結果は、`fd2log` に書込まれます。起動時に `lock` ファイルと `lockl` ファイルが存在していると、`runacct` はエラーを出力します。`lastdate` ファイルは、`runacct` が最後に起動した月日を記録し、1 日に 1 回しか起動できないようにします。`runacct` がエラーを検出すると、`/dev/console` にメッセージが書込まれ、*root* と *adm* にメールが送られ、ロックが解除され、診断ファイルが保存されて、実行が終了します。

runacct の再入可能な状態

`runacct` の再起動を可能にするには、処理を複数の再入可能な状態に分割します。最後に終了した状態は、ファイルに記録されます。状態が終了するたびに、`statefile` は更新されて次の状態を反映します。ある状態の処理が終了すると、`statefile` が読取られ、次の状態が処理されます。`runacct` は、CLEANUP 状態に達すると、ロックを解除して終了します。各状態は次のように実行されます。

SETUP `turnacct` コマンドが引数 switch (on/off) で実行されます。プロセス・アカウンティング・ファイルの `/var/adm/pacct?` は `/var/adm/Spacct?.MMDD` に移されます。`/etc/wtmp` ファイルは `/var/adm/acct/nite/wtmp.MMDD` に移され、現在時刻が最後に追加されます。

WTMPFIX　wtmpfixプログラムがniteディレクトリのwtmpファイルの正当性をチェックします。一部の日付が変更されていると、acctcon1 が失敗します。日付の変更記録があると、wtmpfix は wtmp ファイルのタイプ・スタンプを調整しようとします。

CONNECT1　接続セッション・レコードが ctmp.h 形式で ctmp に書込まれます。回線使用ファイルが作成され、さらにリブート・ファイルが作成されて wtmp ファイルにあるすべてのブート・レコードが表示されます。

ctmp が ctacct.MMDD に変換されます。これは接続アカウンティング・レコードです。アカウンティング・レコードの形式は tacct.h です。

acctprc1 プログラムと acctprc2 プログラムによって、プロセス・アカウンティング・ファイル /var/adm/Spacct?.MMDD が ptacct?.MMDD の合計アカウンティング・レコードとして変換されます。Spacct ファイルと ptacct ファイルは番号で対応しているので、runacct に障害が発生しても Spacct ファイルの不要な再処理は行われません。ただし、この状態で runacct を再起動する場合は、最後の ptacct ファイルは完全でないので削除します。

MERGE　プロセス・アカウンティング・レコードと接続アカウンティング・レコードをマージして、daytacct を作成します。

FEES　fee ファイルの ASCII tacct レコードを daytacct にマージします。

DISK　dodisk プロシージャが実行された翌日に、disktacct と daytacct をマージします。

MERGETACCT　daytacct と累積合計アカウンティング・ファイル sum/tacct をマージします。daytacct は sum/tacctMMDD に毎日保存されるので、sum/tacct が破壊または消失した場合でも再生できます。

CMS　今日のコマンド・サマリを累積コマンド・サマリ・ファイル sum/cms にマージします。ASCII 形式と内部形式のコマンド・サマリ・ファイルを作成します。

USEREXIT　すべてのインストール依存型（ローカル）アカウンティング・プログラムをここに置きます。

CLEANUP　テンポラリ・ファイルを削除し、prdaily を実行してその出力を sum/rprtMMDD に保存し、ロックを削除し、終了します。

runacct の障害からの復旧

runacct プロシージャでは、さまざまな理由から障害が起きる場合があります。多くは、システム・クラッシュ、/usr の容量不足、または *wtmp* ファイルの破壊が原因です。activeMMDD ファイルがある場合は、そのファイルでエラー・メッセージの有無を確認します。active ファイルとロック・ファイルがある場合は、fd2log に異常なメッセージが記録されていないかどうかを確認します。runacct によって生成されるエラー・メッセージとその対応策を、以下に示します。

ERROR: locks found, run aborted

/var/adm/acct/nite/lock ファイルと /var/adm/acct/nite/lock1 ファイルが見つかりました。この 2 つのファイルを削除しないと、*runacct* を再起動できません。

ERROR: acctg already run for date: check /var/adm/acct/nite/lastdate

lastdate の日付と今日の日付が同じです。lastdate を削除します。

ERROR: turnacct switch returned rc=?

turnacct と accton の整合性を調べます。accton プログラムは root の所有とし、*setuid* ビット・セットを持つ必要があります。

ERROR: Spacct?.MMDD already exists

ファイルのセットアップがすでに実行されています。ファイルの状態を調べ、手作業でセットアップを実行します。

ERROR: /var/adm/acct/nite/wtmp.MMDD already exists, run setup manually

wtmp.MMDD ファイルがすでに存在しています。手作業でセットアップを実行します。

ERROR: wtmpfix detected a corrupted wtmp file. Use fwtmp to correct the corrupted file.

wtmpfix が破損した wtmp ファイルを検出しました。fwtmp を使用して破損ファイルを修復します。

ERROR: connect acctg failed: check /var/adm/acct/nite/log

acctcon1プログラムが不良wtmpファイルを検出しました。fwtmpで不良ファイルを修復します。

ERROR: Invalid state, check /var/adm/acct/nite/active

statefileファイルが破壊された可能性があります。再起動する前にstatefileで異常を調べ、activeを読取ります。

runacct の再起動

runacct プログラムに引数を指定しないで実行すると、当日の最初の実行と見なされます。runacct を再起動してアカウンティングを再実行する月日を指定するには、引数 MMDD を指定します。処理のエントリ・ポイントは statefile の内容に基づいています。statefile をオーバーライドするには、必要な状態をコマンド行に含めます。たとえば、runacct を起動するには、次のコマンドを入力します。

```
nohup runacct 2 /var/adm/acct/nite/fd2log &
```

runacct を再起動するには、次のコマンドを入力します。

```
nohup runacct 0601 2 /var/adm/acct/nite/fd2log &
```

特定の状態で runacct を再起動するには、次のコマンドを入力します。

```
nohup runacct 0601 WTMPFIX 2 /var/adm/acct/nite/fd2log &
```

破壊されたアカウンティング・ファイルの修復

アカウンティング・システムにエラーが発生し、ファイルが破損または消失する場合があります。エラーの中には無視できるものもあります。破損または消失したファイルをバックアップからリストアするだけで済む場合もあります。しかし、アカウンティング・システムの完全性を維持するために、修復を要するファイルもあります。

wtmp エラーの修復

wtmp ファイルは、アカウンティング・システムの最もデリケートな部分です。IRIX システムがマルチユーザ・モードのときに日付が変更されると、日付変更レコードのセットが /etc/wtmp に書込まれます。wtmpfix は、日付変更があった場合に wtmp レコードのタイムスタンプを調整するプログラムです。ただし、日付変更と再起動が同時に実行されると、wtmpfix で検出されずに acctcon1 が失敗する場合があります。

wtmp ファイルを修復するには、次の手順に従います。

1. **cd /var/adm/acct/nite** と入力します。
2. **fwtmp < wtmp.MMDD > xwtmp** と入力します。
3. **ed xwtmp** と入力します。
4. 破壊されたレコードを削除するか、または、最初から日付が変更された日までのすべてのレコードを削除します。
5. **fwtmp -ic <wtmp> wtmp.MMDD** と入力します。

修復不能な *wtmp* ファイルは削除し、空の *wtmp* ファイルを作成します。

6. **rm /etc/wtmp** と入力します。
7. **touch /etc/wtmp** と入力します。

これで接続時間に対する請求は行われません。acctprc1 は、特定のプロセスを所有していたログインを確定できませんが、そのユーザ ID に対応するパスワード・ファイルの最初のログインに請求が行われます。

tacct エラーの修復

アカウンティング・システムを使用し、システム・リソースの使用に対してユーザに請求する場合は、sum/tacct の完全性が重要です。異常な tacct レコードが、負の数、重複しているユーザ ID、または 65,535 というユーザ ID の形で現れることがあります。このような場合は、prtacct で sum/tacctprev を調べます。問題がないようならば、最新の sum/tacct.MMDD を修正し、sum/tacct を再作成します。次に、簡単な修正手順を示します。

1. 次のコマンドを入力します。

   cd /var/adm/acct/sum

2. 次のコマンドを入力します。

   acctmerg -v < tacct.MMDD > xtacct

3. 次のコマンドを入力します。

   ed xtacct

4. 不良レコードを削除します。
5. 重複している UID レコードを別のファイルに書込みます。
6. 次のコマンドを入力します。

   acctmerg -i < xtacc t > tacct.MMDD

7. 次のコマンドを入力します。

   acctmerg tacctprev <tacct.MMDD> tacct

monacct プロシージャは、すべての tacct.MMDD ファイルを削除します。したがって、tacct.MMDD ファイルをマージすれば、sum/tacct を再作成できます。

アカウンティングのための休日の反映

/usr/lib/acct/holidays ファイルには、アカウンティング・システムのためのプライム／非プライム・テーブルがあります。このテーブルには、年間の休日を反映させます。テーブルの書式は、次の 3 つのエントリで構成されます。

- コメント行：行の最初の文字をアスタリスク（*）にします。ファイルの任意の場所に記述できます。
- 年度指定行：ファイルの第 1 データ行（非コメント行）を使用し、一度だけ記述します。この行は、3 つのフィールドで構成されます。先行空白スペースは無視されます。各フィールドには 4 桁の数字が入ります。たとえば、年度を 1992 年、プライム・タイムを午前 9:00、非プライム・タイムを午後 4:30 と指定するには、次のように入力します。

  ```
  1992 0900 1630
  ```

  ただし、時刻のフィールドでは 2400 が自動的に 0000 に変換されます。
- 会社休日行：年度指定行に続く行であり、次の形式で記述します。

```
day-of-year Month Day Description of Holiday
```

day-of-year フィールドは 1 ～ 366 までの数字であり、対応する休日を表します。先行空白スペースは無視されます。残りの 3 つのフィールドは実際には注釈であり、ほかのプログラムには現在使用されていません。

runacct の日別レポート

`runacct` は起動するたびに 5 つの基本的なレポートを作成します。接続アカウンティング、ユーザ別の日別使用状況、日計および月計のコマンド使用状況、ユーザの最終ログイン日です。次に、各レポートとその内部データの意味について説明します。

レポートの前半では、最初に from/to バナーで管理者に報告期間を示します。報告期間は、最後のアカウンティング・レポートの作成時点から現在のアカウンティング・レポートの作成時点までの期間です。バナーの後に、システムの再起動や停止、停電の復旧など、`acctwtmp` プログラムによって `/etc/wtmp` にダンプされたレコードのログが続きます。詳細については、acct(1M) マン・ページを参照してください。

レポートの後半は、回線使用の内訳です。TOTAL DURATION フィールドは、システムがマルチユーザ・モードになっていた時間、つまり端末回線からアクセス可能であった時間を示します。各フィールドを次に示します。

LINE　端末回線またはアクセス・ポート。

MINUTES　アカウンティング期間中の回線の総使用時間（分単位）。

PERCENT　アカウンティング期間の全時間に対して、回線の総使用時間（分単位）が占める割合。

SESS　`login` セッションのためにポートがアクセスされた回数。

ON　このフィールドはあまり重要ではありません。以前はユーザがログインするためにポートを使用した回数を示していました。現在では、新しいユーザのログインのために`login`を明示的に実行しなくなったので、このフィールドはSESSと同じです。

OFF　ユーザがログオフした回数。この回線で起きた割込みの回数も示します。通常、ポートに割込みが起こるのは、システムがマルチユーザ・モードになった後で最初に `getty` が起動したときです。このフィールドに注目するのは #OFF が

#ON を大きく上回ったときです。通常は、マルチプレクサ、モデム、またはケーブルが不良になったか、どこかに接続不良があることを示します。最も一般的な原因として、ケーブルが外れてマルチプレクサと接続されていない場合があります。

リアルタイムのアカウンティング中は、`/etc/wtmp` を監視します。このファイルから、接続アカウンティングが始まるからです。このファイルが急激に大きくなった場合は `acctcon1` を実行し、最もノイズの多い回線を探します。割込みが盛んに起こると、システム全体のパフォーマンスに影響します。

日別使用状況レポート

日別使用状況レポートは、システム・リソースの使用状況の内訳をユーザ別に報告します。各フィールドを次に示します。

UID　ユーザ ID

LOGIN NAME　ユーザのログイン名。同一のユーザ ID に対して複数のログイン名が存在する場合がありますが、このデータはリソースを使用したログイン名を示します。

CPU (MINS)　ユーザのプロセスが CPU を使用した時間。PRIME（プライム）時と NPRIME（非プライム）時の使用に分けて値が示されます。時間帯の分け方は、`/usr/lib/acct/holidays` ファイルに定義されています。納品時のプライム・タイムは、9 ～ 17 時に設定されています。

KCORE-MINS　プロセスの実行中に使用されるメモリの累積量を示す尺度。値は、1 分当たりに使用されたメモリのキロバイト・セグメント数で示されます。PRIME 時と NPRIME 時に分けて値が示されます。

CONNECT (MINS)
ユーザがシステムにログインしていた時間。この時間が長く、# OF PROCS の値が小さいと、ユーザが実際にはシステムを使用せずに長時間ログインしていたことを意味します。このフィールドも PRIME 時と NPRIME 時に分けて示されます。

DISK BLOCKS　ディスク・アカウンティング・プログラムが実行されると、その出力が合計アカウンティング・レコード（`tacct.h`）にマージされ、このフィールドに表示されます。このディスク・アカウンティングは、`acctdusg` プログラムによって実行されます。

OF PROCS　ユーザが呼出したプロセスの数。この値が大きいと、ユーザがシェルの制御を失っている可能性があります。

O SESS　ユーザがシステムにログインした回数。

DISK SAMPLES
前に示した DISK BLOCKS の平均値を得るためにディスク・アカウンティングが実行された回数。

FEE　合計アカウンティング・レコードのフィールドですが、あまり使用されません。このフィールドは `chargefee` シェル・プロシージャによってユーザに請求したウィジェットの累計値を示します。acctsh(1M) マン・ページを参照してください。`chargefee` プロシージャは、ファイルのリストアなどの特別なサービスが行われた場合にユーザに請求します。

日別コマンド・サマリと月計コマンド・サマリ

この 2 種類のレポートは実質的に同じです。ただし、日別コマンド・サマリは現在のアカウンティング期間だけを対象とし、月計コマンド・サマリは会計期間の開始日から当日までの期間を対象とします。つまり、月計レポートは `monacct` を最後に呼出した時点からの累積データを反映します。

システム管理者は、2 種類のレポートのデータから使用回数の多いコマンドを判断できます。各コマンドによるシステム・リソースの使い方に着目し、その使い方に応じてシステムを調整できます。各レポートは TOTAL KCOREMIN に従ってソートされます。TOTAL KCOREMIN は便宜的な尺度ですが、システムのメモリ・ドレインを簡単に計算できます。

COMMAND NAME
コマンド名。ただし、すべてのシェル・プロシージャは `sh` という名前で一括されます。プロセス・アカウンティング・システムが報告するのはオブジェクト・モジュールだけであるからです。管理者は、`a.out` や `core` などの正当でない名前でプログラムが呼出される回数を監視します。ユーザは個人用プログラムを使いたがりますが、そのことを他人に隠す傾向があります。正当でない名前のコマンドを実行したユーザや、特権ユーザの権限が乱用されていないかどうかを確認するには、`acctcom` も有効なツールです。

NUMBER CMDS
特定のコマンドの呼出し総数。

TOTAL KCOREMIN
: プロセスの実行中に使用されたメモリの累積総量を示す尺度。1 分当たりのキロバイト・セグメント数で示されます。

TOTAL CPU-MIN
: プログラムの累積総処理時間。

TOTAL REAL-MIN
: プログラムの累積総リアルタイム（ウォール・クロック）時間。この総時間は、プロセスをバックグラウンドで実行する時間ではなく、実際の待ち時間を示します。

MEAN SIZE-K
: NUMBER CMDS の呼出し数で割った TOTAL KCOREMIN の平均値。

MEAN CPU-MIN
: NUMBER CMDS と TOTAL CPU-MIN から算出された平均値。

HOG FACTOR
: 使用可能なCPU時間に対してプロセスの実行中に費やされた総CPU時間を相対的に示す尺度。システムの使用可能時間に対するシステムの使用時間の割合であり、次の式で求められます。

  total CPU time / elapsed time

CHARS TRNSFD このフィールドは、負の値になることもあります。**read** と **write** のシステム・コールで出入りした文字の総数です。

BLOCKS READ プロセスによる読取りと書込みの対象となった物理ブロックの総数。

IRIX の拡張アカウンティング

規模の大きなコンピュータ・サイトでは多くの外部ユーザを抱え、リソースの利用に対して別途料金を請求しなければならない場合があります。IRIX は利用情報を提供する機能を備えていますが、この方法は多くのサイトにとって不十分です。標準の IRIX アカウンティングは、いくつかの重要なメトリックスを欠き、大量のディスク領域を消費し、利用料金の請求方法にも柔軟性がありません。サードパーティのアカウンティング・ソフトウェアの中にはこのような問題を解決できるものもありますが、IRIX が提供するデータでは限界があります。配列クラスタやハイパー

キューブでは、1 人のユーザが利用したリソースが複数のシステムに分散されてしまうことから、問題はさらに複雑になります。

IRIX はこのような規模の大きなコンピュータ・サイトでのアカウンティングのニーズに応えるため、拡張アカウンティング、配列セッション、プロジェクト ID という 3 つの機能を提供します。

拡張アカウンティングについて

最初の IRIX リソース・アカンティング方式は、標準の System V アカウンティングをベースにしていました。プロセスが存在するときは常にカーネルはリソースの利用情報を含むレコードをファイルに書込みます。カーネルがファイルの入出力処理を行なうため、アカウンティングの処理は負荷の重いシステムにとって小さなボトルネックとなっていました。もう 1 つの問題は、System V アカウンティングで書込まれたデータの形式です。利用情報は 16 ビットの浮動小数点番号まで扱いにくい圧縮形式で格納されます。値は正確さを失い、最大値が最先端のシステム (2^{34} または 16GB) の許容範囲を超えてしまうことさえあります。当然、アカウンティング・レコードを拡張する余地はありません。しかし、十分な空き領域がないにもかかわらず、別のフィールドがレコード・サイズの増大を要求すれば、実質的にアカウンティング・データを使用しているすべての既存のソフトウェアを破壊してしまう恐れがあります。

IRIX リリース 6.1 およびそれ以降のリリースでは、System V のアカウンティング機能がそのまま提供され、さらに拡張アカウンティングが追加されます。

アカウンティング拡張機能の最も重要な違いは、その配送方式にあります。レコードはシステム監査トレール（SAT: System Audit Trail）機能を使って書込まれます。SAT では特殊なシステム・コールを使用し、カーネルからデーモンを使って監査レコードを収集します。また、SAT はシステム管理者が選択した宛先にレコードを書込みます（satd(1M) マン・ページを参照）。これにより、カーネルはファイルの入出力作業から解放され、システム管理者が柔軟な方法でアカウンティング・データを取扱えるようになります。

`sat_select` コマンドを使って監査サブシステムが監視するアカウンティング・イベントを選択することができます。詳しくは、sat_select(1M) マン・ページを参照してください。

監査ファイルを入替えたり、ファイルシステムが一杯になったときの処理には、`satd` プログラムを使います。サードパーティのソフトウェアは監査ファイル全体を読込むことも、既存の `sat_reduce` プログラムを使って必要なアカウンティング・レコードだけを取出すこともでき

ます。サイトがアカウンティング以外の情報も監査することを選択した場合は、アカウンティング以外のレコードも含まれることがあります。sat_reduce(1M) マン・ページを参照してください。各レコードの内容は `sat_interpret` プログラムを使って ASCII 形式でダンプされます。sat_interpret(1M) マン・ページを参照してください。

拡張アカウンティング・レコードに含まれるリソース・データは圧縮されずに 64 ビットの値で格納されます。将来的には大抵のメトリックスにとって十分な値と言えます。レコードには、将来の拡張用にスペアのフィールドと、ソフトウェアが将来の形式変更に正規に対応するためのバージョン・コードが含まれています。System V アカウンティングが報告するすべてのメトリックスに加え、新たに次のメトリックスが追加されています。

- スワップ数
- 読込まれた、書込まれたバイト数
- 読込み、書込み要求の数
- 待ち時間
  - ブロック I/O
  - 物理 I/O
  - 実行待ち行列上

拡張アカウンティングの使用

システム上で拡張アカウンティングを使用するには、次の手順に従います。

1. `systune` コマンドを使って *do_sessacct* または *do_extpacct* パラメータをゼロ（0）以外の値に設定し、カーネルにおけるアカウンティング・セッションを有効にします。systune(1M) マン・ページを参照してください。
2. IRIX 配布ディスクから `eoe.sw.audit` サブシステムをインストールします。
3. 次のコマンドを実行して監査機能を有効にします。

   ```
   chkconfig audit on
   ```

4. `satconfig` コマンドを使って *sat_proc_acct* または *sat_session_acct* 監査イベントを有効にします。satconfig(1M) マン・ページを参照してください。アカウンティングの目的だけに監査機能を使用している場合は、ディスク領域を節約するためそのほかのすべてのイベントをオフにすることもできます。

詳しくは、extacct(5) マン・ページを参照してください。また、拡張アカウンティングのためのカーネル・パラメータのリストが『IRIX Admin: System Configuration and Operation』の 付録 A「IRIX カーネルのチューニング・パラメータ」に記載されています。

配列セッション

ディスク領域の消費量とアカウンティング・レコードの処理時間を削減するため、IRIX は配列セッションによってアカウンティング情報を蓄積してから報告できるようになっています。アカウンティング処理は別に制御されます。サイトではこれらのアカウンティング方式のいずれか、または両方を使用できます。セッション・アカウンティング・レコードには、プロセス・アカウンティング・レコードと同様のデータが含まれます。ただし、プロセス・アカウンティングでは、カウンタと値が、セッションのメンバーであるすべての蓄積情報に反映されます。

配列セッションは、1 つの一意な識別子である配列セッション・ハンドル（ASH: Array Session Handle）によってすべて互いに関連付けられている複数のプロセスをまとめたものです。通常、子プロセスはその作成時に親プロセスの ASH を継承するため、親の配列セッションのメンバーになります。しかし、プロセスがその親の配列セッションを離れ、新しいセッションを開始するためのシステム・コールが用意されています。`login` および `rshd` のようなプログラムはこのシステム・コールを使い、ログインするだけで効果的に新しいセッションを開始します。`cron`、`su`、およびいくつかのバッチ・キューイング・システムはこのシステム・コールを使い、別のユーザのために実行された作業がそれ自体のセッションを持てるようにします。ASH 指定の最後のプロセスが終了すると、配列セッションは終了し、セッション・アカウンティングが書込まれます。

ASH は 64 ビットの値です。一意で、増分する値（プロセス ID と同様）がデフォルトで各新しい配列セッションにハンドルとして割当てられます。ただし、システム・コールを使い、必要に応じて配列セッションのハンドルを変更することも可能です。この方法を使うと複数のシステムにある配列セッションのハンドルを 1 つの配列内で同期させることができるため、分散されたアカウンティング作業を 1 つにまとめることができます。

詳しくは、array_sessions(5) マン・ページを参照してください。

システムが割当てるハンドルの範囲は設定可能です。これにより、自動的に割当てられるハンドルとプロセスで指定されるハンドルとの重複を避けることができます。システムは、ローカル・システム上の複数のセッションが同時に特定の ASH を使用しないようにします。

蓄積された各種プロセスのアカウンティング・データ全体に加え、セッション・アカウンティング・レコードには、サービス・プロバイダ情報に使用する 64 バイトのフィールドが含まれています。特に、バッチ・キューイング・システムでは、このフィールドを使用してキュー名、イニシエータなどのデータを記録します。デフォルトでは、新しい配列セッションに対するサービス・プロバイダの情報が親プロセスの配列セッションから継承されます。

標準の `init` プログラムでは、常にサービス・プロバイダ情報をすべてゼロ（0）に設定しています。また、標準のログイン・ユーティリティ（`login`、`su`、`rshd`）がサービス・プロバイダ情報を変更することは絶対にありません。一方、バッチ・キュー・システムでは、サービス・プロバイダ情報をゼロ以外の値に設定します。このように、サービス・プロバイダの情報がゼロに設定されているかどうかで、バッチ作業と対話型セッションを識別することができます。

プロジェクト ID

指定システムの利用に対し、部署ごとに別々に料金を精算する必要があるサイトが多数あります。これは通常、各システムのユーザ ID の利用料金の合計を該当部署に請求するだけで済みます。ただし、1 人のユーザが複数の部署に属しているようなサイトでは、料金全部を 1 つの部署に請求するのは不都合な場合があります。

このようなアカウンティング上の難問を解決するため、IRIX ではプロジェクト ID 機能を導入しています。プロジェクト ID は、以下の点を除いてはグループ ID と同じです。

- 現在のプロジェクト ID は、1 個のプロセスではなく、配列セッションに関連付けられます。
- プロジェクト ID がアクセス権に影響を与えることはありません。アカウンティングの目的だけに使用されます。

デフォルトのプロジェクト ID は各ユーザ ID に関連付けられます。別のプロジェクトに請求すべき料金が生じた場合は、常に `newproj` コマンドを使ってプロジェクト ID を変更します。詳しくは newproj(1) マン・ページを参照してください。このコマンドは、新しいシェルと配列セッションを起動します。古いシェルで実行されるバックグラウンド・プロセスは、元のプロジェク

ト ID の下で引続きアカウントされます。また、ユーザ ID もグループ ID も変更されないため、アクセス権は何の影響も受けません。許可されていないプロジェクト ID をユーザが指定するのを防止するため、`newproj` コマンドは各ユーザの有効なプロジェクト ID をリストしたファイルを参照します。プロジェクト ID を設定するシステム・コールを実行できるのは特権ユーザだけです。

ユーザ ID と許可されているプロジェクトを記述したファイル `/etc/project` のスタイルは `/etc/passwd` または `/etc/group` と同様です。詳しくは、project(4) マン・ページを参照してください。このファイルでは、`/etc/passwd` が変更されるのを避けるため、各ユーザのデフォルトのプロジェクトも指定します。プロジェクト ID は単純な番号であるため、`/etc/projid` という別のファイルでニーモニックの ASCII 名と数値によるプロジェクト ID を対応付けています。詳しくは、projid(4) マン・ページを参照してください。システム管理者は `systune` の *dfltprid* 変数を使って標準のデフォルト・プロジェクト ID を設定できます。

デフォルトでは、配列セッションはその生成元であるセッションのプロジェクト ID を継承します。新しい配列セッションを起動する標準のログイン・ユーティリティ（`login`、`su`、`rshd`）はプロジェクト ID を新しいユーザのデフォルトのプロジェクト ID に変更するよう更新されています。

プロジェクト ID ファイルを読込むためのライブラリ・ルーチンも提供されています。これは、パスワード・ファイルのデータを読込むのに使用するライブラリ・ルーチンに相当します。詳しくは、projid(3C) マン・ページを参照してください。

索引

A

acl 139
  ls オプション 139
attrinit コマンド 157

B

Backup 53
backup と restore
  xfsdump と xfsrestore の使用 59–91

C

Capabilities 144
Capabilities、デフォルト 148
capabilities、ファイルの 156
chacl コマンド 142
cpio
  概要 32
  機能 95
  システム・マネージャ 95
  バックアップの作成 96
  ファイルのリストア 94、97

D

dbedit ユーティリティ 210
dd
  概要 32
  機能 97
  変換オプション 100
/dev/tape 44
dump
  /etc/dumpdates 56
  vs. xfsdump 60
  インクリメンタル・バックアップ 56
  概要 32
  バックアップの作成 56

E

/etc/capability ファイル 145
/etc/dumpdates 56
/etc/hosts.equiv ファイル 160
/etc/inetd.conf ファイル 161
/etc/passwd ファイル 160

F

FTP サービス 176

H

housekeeping ディレクトリ　90

I

inetd サービス
　制限　176
inetd デーモン　161
IP パケットの転送　175
IP パケットの転送禁止　175
IRIX admin
　マニュアル　xxi–xxii

L

ls -d オプション　143

N

ncheck コマンド　133
NFS
　制限または無効化　180
NFS の無効化　180
NIS
　無効化　179
NIS の無効化　179

O

orphanage ディレクトリ　90

P

PROM パスワード
　消去　117
　設定　117
　の使い方　116
pwck コマンド　125

R

Restore
　概要　32
　データのリストア　54
restore
　vs. xfsrestore　60
　概要　32
　ファイルシステムのリストア　57
.rhosts ファイル　160
RPC サービス
　制限　179

S

SAT
　sat_select　193
　イベント・タイプ　193
　カスタマイズ　191
　記録の例　203
　データの解釈　203
sat_interpret ユーティリティ　203
sat_reduce ユーティリティ　203
sat_select　193
sat_select ユーティリティ　198

sat_summarize ユーティリティ　203
satconfig ユーティリティ　197
sendmail
　設定　183
Set-GID　133
Set-UID　133

T

tar
　概要　32
　機能　92
　バックアップの作成　92
　比較キー・キャラクタ　94
　ファイルのリストア　94、97

W

World Wide Web
　とセキュリティ　169

X

xfsdump
　STDOUT　91
　インクリメンタル・ダンプ　74
　ダンプ一覧表　76
　ダンプの再開　74
　使い方　68
　ネットワークでの使い方　91
　メディア形式　62
　メディアの再利用　72
　メディアの指定　69
xfsrestore
　STDIN　91
　セッション ID　81
　セッション・ラベル　81
　漸増リストア　86
　対話モードによるリストア　84
　単純なリストア　81
　中断されたダンプのリストア　87
　使い方　79
　ネットワークでの使い方　84、91
　ファイルのリストア　83
　リストアの中断　89
xhost コマンド　163
X サーバ・アクセス
　制御　161
X サーバへのアクセス
　チェック　164

あ

アカウンティング
　システム　215
　プロセス　215
アクセス・コントロール・リスト　139
アクセス制御の違反　207

い

違反
　アクセス制御セキュリティの　207
　異例なシステム使用によるセキュリティの　206
　外部との接続を通じての　206
　部外者によるセキュリティ　205
　部内者によるセキュリティ　207

無許可侵入によるセキュリティ　205
ルート特権セキュリティの　208
異例なシステムの使用　206
インクリメンタル・ダンプ、xfsdump　74
インターネット、定義　165
隠蔽されたネットワーク
ハードウェアの設定　172
隠蔽されたホスト
ハードウェアの設定　171

え

エラー・メッセージ、バックアップと回復　105

お

オートチェンジャ　32

か

回復
エラー・メッセージ　105
システム破壊後　46
「回復 システムの」　46
外部との接続　206
監査
sat_select　193
イベント・タイプ　193
ガイドライン　209
カスタマイズ　191
監査項目の一覧　193
監査追跡　211
起動　189
記録の例　203
結果の読取り　203
システム・データ・ファイルの変更　209
システム・プログラムの変更　210
設定ユーティリティ　191
データのアーカイブ　211
データの削除　211
デフォルトの環境　190
ファイル　202、209
復元　199
不適切な使用　209
保存　199
保存されたファイル　200
ユーザ　201、209
要注意ユーザ　209
ラベル　203
監査、satconfig ユーティリティ　197
監査、説明　187
監査データ
解釈　203
変換　203
監査データのアーカイブ　211
監査データの解釈　203
監査のカスタマイズ　191
管理、システム
マニュアル　xxi–xxii

さ

削除
監査データ　211

し

磁気テープ、xfsdump による再利用　72
システム・アカウンティング　215
システム・アクセス　124、125
システム管理
　マニュアル　xxi–xxii
システム・データ・ファイル
　変更　209
システム・データ・ファイルの変更　209
システムの回復　46
システムの操作
　一般　215
システムのバックアップ　45
システム・パスワード
　パスワード
　　システム　116
システム・プログラムの変更　210
［システム・メンテナンス（System Maintenance）］メニュー　46
ジュークボックス　32
侵害
　可能性　204

す

ストリーム終了子、xfsdump　62

せ

セキュリティ
　IRIX 標準　110
　LAN　160
　xhost コマンド　163
　セキュリティ・ガイドライン　111
　トロイの木馬　113
　ネットワーク　159
　ファイアウォールのための強化　174
　プロセス・アカウンティング　215
セキュリティ違反
　部内者　207
セキュリティ違反（監査）
　アクセス制御　207
　異例なシステム使用　206
　外部との接続　206
　可能性　204
　部外者　205
　無許可侵入　205
　ルート特権　208
セキュリティ違反の可能性　204
セキュリティに関するユーザの教育　182
絶対パス名、テープの読取り　102
漸増リストア、xfsrestore　86

そ

ソフトウェア
　整合性検査　181

た

対話モードによるリストア、xfsrestore　84
ダンプ一覧表 , xfsdump　62、76
ダンプ・ストリーム、xfsdump　62
ダンプ・セッション、xfsdump　62

ち

中断されたダンプのリストア、xfsrestore 87

て

データ・セグメント、xfsdump 62
データのリストア
  cpio 94、97
  Restore 54
  restore 57
  tar 94、97
テープ
  検査 104
  再利用 41
  保存 40
テープ、絶対パス名 102
テープ・デバイス、デフォルト 44
デフォルトのバックアップ・デバイス
  変更 50

と

トロイの木馬 113

な

内部ネットワークの DNS 設定 182
内部ネットワークの設定 182

に

二重ホームホスト
  ソフトウェア設定 174
二重ホーム・ホスト
  ハードウェアの設定 171

ね

ネットワーク
  アクセス制御 160
  隠蔽された 172
  セキュリティの問題 166
  バックアップ 38

は

ハードウェアの設定
  ファイアウォール 169–173
  ルータ 170
パスワード
  PROM 116
  エージング 122–125
  選び方 115
  管理 115
  強制 131
  ダイヤルアップ 118
  チェック 125
  変更 124
  保護 178
パスワード PROM 116
パスワードの変更 124
バックアップ
  bru による容量の見積もり 53
  cpio によるインクリメンタル 93、96
  dd 変換オプション 100
  dump によるインクリメンタル 56

tar によるインクリメンタル　93、96
誤ったバックアップのリストア　103
インクリメンタル　38
エラー　102
エラー・メッセージ　105
概要　31
計画　35
作成　44
自動化　39
ネットワークを通じての　38
バイトの入れ換え　101
頻度　36
保存　40
有効なプログラム　32–33
ユーザ・ファイルシステム　37
読取り不能　99
ルート・ファイルシステム　36
［バックアップとリストア（Backup & Restore）］ウィンドウ　45

ふ

ファイアウォール　165–184
設計方針　168
ソフトウェア設定　174–184
定義　166
ハードウェアの設定　169–173
モニタリングセキュリティ　168
ファイアウォールのモニタリング　168
ファイル Capabilities　156
ファイルの監査　202
部外者によるセキュリティ違反　205
部内者によるセキュリティ違反　207
プロクシ・サーバ　184
プロセス・アカウンティング　215

ほ

ホスト
隠蔽された　171
二重ホーム　171

む

無許可侵入　205

め

メール
スプールの分割　184
内部ネットワークの設定　183
メディア
オブジェクト、xfsdump　62
形式、xfsdump　62
保存　40
メディアチェンジャ　32

ゆ

ユーザ・アカウント
パスワードの強制　131
ユーザとセキュリティ　182
ユーザの監査　201

ら

ラベルの監査　203

り

リストアの中断、xfsrestore 89

る

ルータとファイアウォール 170
ルート特権の違反 208

ろ

ログイン
  root の制限 129
  オプション 128
  記録 131
  最大試行回数 129
  使用禁止時間 130
  特殊アカウント 126
  ロック 125、127
ログインのロック 125
ログ・ファイル 180