

# HP OpenVMS

---

## システム・セキュリティ・ガイド

2005 年 4 月

ソフトウェア・バージョン:   OpenVMS Alpha Version 7.3-2  
                                  OpenVMS VAX Version 7.3

本書は、OpenVMS オペレーティング・システムを通じて利用可能な、セキュリティ関連の機能について説明します。具体的なセキュリティ・ニーズに照らして、各機能の目的と適切な応用方法を説明します。

---

© 2005 Hewlett-Packard Development Company, L.P.

本書の著作権は Hewlett-Packard Development Company, L.P. が保有しており、本書中の解説および図、表は Hewlett-Packard Development Company, L.P. の文書による許可なしに、その全体または一部を、いかなる場合にも再版あるいは複製することを禁じます。

日本ヒューレット・パカードは、弊社または弊社の指定する会社から納入された機器以外の機器で対象ソフトウェアを使用した場合、その性能あるいは信頼性について一切責任を負いかねます。

本書に記載されている事項は、予告なく変更されることがありますので、あらかじめご承知おきください。万一、本書の記述に誤りがあった場合でも、弊社は一切その責任を負いかねます。

本書で解説するソフトウェア(対象ソフトウェア)は、所定のライセンス契約が締結された場合に限り、その使用あるいは複製が許可されます。

Microsoft®, Windows NT® は米国 Microsoft 社の登録商標です。Intel®, Pentium®, Intel Inside® は米国 Intel 社の登録商標です。UNIX®, The Open Group™ は、The Open Group の米国ならびに他の国における商標です。

このドキュメントに記載されているその他の会社名および製品名は、各社の商標または登録商標です。

原典: HP OpenVMS Guide to System Security  
© 2004 Hewlett-Packard Development Company, L.P.

## まえがき

**Part 1 セキュリティの概要**
**1 システム・セキュリティ**

1.1	コンピュータ・セキュリティ問題のタイプ .....	1-1
1.2	セキュリティ要件のレベル .....	1-3
1.3	安全なシステム環境の構築 .....	1-4
1.4	CDSA (Common Data Security Architecture) .....	1-5
1.5	SSL (Secure Sockets Layer) .....	1-6
1.6	Kerberos .....	1-7

**2 OpenVMS のセキュリティ・モデル**

2.1	安全なオペレーティング・システムの構造 .....	2-1
2.1.1	リファレンス・モニタの概念 .....	2-1
2.1.2	リファレンス・モニタによるセキュリティ規則の実施 .....	2-2
2.2	リファレンス・モニタの実装 .....	2-3
2.2.1	サブジェクト .....	2-3
2.2.2	オブジェクト .....	2-4
2.2.3	登録データベース .....	2-5
2.2.4	監査証跡 .....	2-6
2.2.5	リファレンス・モニタ .....	2-7
2.2.6	アクセス・マトリックスで表した登録データベース .....	2-7
2.3	要約: システム・セキュリティ設計 .....	2-10

**Part 2 一般ユーザのためのセキュリティ**
**3 システムの安全な使用**

3.1	アカウントのパスワードの選択 .....	3-1
3.1.1	初期パスワードの取得 .....	3-2
3.1.2	パスワードに関するシステム制限の順守 .....	3-2
3.2	使用するパスワードのタイプ .....	3-2
3.2.1	システム・パスワードの入力 .....	3-3
3.2.2	第2パスワードの入力 .....	3-4

3.3	アカウントのタイプごとのパスワード要件 .....	3-5
3.4	ログインのタイプとログイン・クラス .....	3-5
3.4.1	会話型ログイン(ローカル, ダイアルアップ, および遠隔ログイン) .....	3-6
3.4.2	外部認証を使用したログイン .....	3-6
3.4.3	情報メッセージの解釈 .....	3-6
3.4.4	システムがユーザに代わってログインする場合(ネットワーク・ログインとバッチ・ログイン) .....	3-8
3.5	ログインの失敗(ユーザがログインできない場合) .....	3-9
3.5.1	システム・パスワードを必要とするターミナルの使用 .....	3-9
3.5.2	ログイン・クラスの制限の順守 .....	3-10
3.5.3	特定の日に使用が限定されたアカウントの使用 .....	3-10
3.5.4	ダイアルアップ・ログインで正しいパスワードを入力しなかった場合 .....	3-10
3.5.5	侵入回避手順が有効になる条件 .....	3-11
3.6	パスワードの変更 .....	3-11
3.6.1	ユーザ自身によるパスワードの選択 .....	3-12
3.6.2	生成パスワードの使用 .....	3-12
3.6.3	第2パスワードの変更 .....	3-14
3.6.4	ログイン時に行うパスワード変更 .....	3-14
3.7	パスワードとアカウントの有効期限 .....	3-14
3.7.1	期限切れパスワードの変更 .....	3-15
3.7.2	期限切れアカウントの更新 .....	3-16
3.8	パスワードの保護に関するガイドライン .....	3-16
3.9	ネットワーク・セキュリティに関する考慮事項 .....	3-18
3.9.1	アクセス制御文字列内の情報の保護 .....	3-18
3.9.2	代理ログイン・アカウントの使用によるパスワードの保護 .....	3-19
3.10	アカウントおよびファイルへのアクセスの監査 .....	3-21
3.10.1	最終ログイン時間の確認 .....	3-21
3.10.2	重要ファイルへのアクセス制御エントリ(ACE)の追加 .....	3-21
3.10.3	セキュリティ管理者への監査の有効化の依頼 .....	3-22
3.10.3.1	ファイル・アクセスの監査 .....	3-22
3.10.3.2	監査対象イベントの追加 .....	3-24
3.11	システム・セキュリティを損なわないログアウト .....	3-24
3.11.1	ターミナル画面の消去 .....	3-25
3.11.2	ハードコピー出力の破棄 .....	3-25
3.11.3	切断されたプロセスの削除 .....	3-26

3.11.4	ダイアルアップ回線への接続の切断 .....	3-26
3.11.5	ターミナルの電源遮断 .....	3-26
3.12	システム・セキュリティへの貢献のためのチェックリスト .....	3-27
<b>4</b>	<b>データの保護</b>	
4.1	ユーザのセキュリティ・プロファイルの内容 .....	4-1
4.1.1	スレッド別セキュリティ .....	4-2
4.1.2	ペルソナ・セキュリティ・ブロック (PSB) データ構造体 .....	4-2
4.1.3	以前のセキュリティ・モデル .....	4-3
4.1.4	スレッド別セキュリティのモデル .....	4-3
4.1.5	ユーザ識別コード (UIC) .....	4-4
4.1.5.1	UIC の形式 .....	4-4
4.1.5.2	UIC 作成に関するガイドライン .....	4-5
4.1.5.3	プロセスによる UIC の取得 .....	4-6
4.1.6	ライト識別子 .....	4-6
4.1.6.1	識別子のタイプ .....	4-6
4.1.6.2	プロセス・ライト・リストとシステム・ライト・リスト .....	4-7
4.1.6.3	プロセスのライト識別子の表示 .....	4-8
4.1.6.4	監査証跡に現れるライト識別子 .....	4-8
4.1.7	特権 .....	4-10
4.2	オブジェクトのセキュリティ・プロファイル .....	4-11
4.2.1	保護オブジェクトの定義 .....	4-11
4.2.2	オブジェクトのプロファイルの内容 .....	4-11
4.2.2.1	所有者 .....	4-12
4.2.2.2	保護コード .....	4-12
4.2.2.3	アクセス制御リスト (ACL) .....	4-13
4.2.3	セキュリティ・プロファイルの表示 .....	4-14
4.2.4	セキュリティ・プロファイルの変更 .....	4-15
4.2.5	オブジェクトのクラスの指定 .....	4-15
4.2.6	プロファイルの変更に必要なアクセス権 .....	4-17
4.3	システムによる保護オブジェクトへのユーザのアクセス可否の判定 ..	4-17
4.4	ACL によるアクセスの制御 .....	4-23
4.4.1	識別子用アクセス制御エントリ (ACE) の使用 .....	4-23
4.4.2	特定ユーザへのアクセスの許可 .....	4-24
4.4.3	オブジェクトへのユーザのアクセスの禁止 .....	4-25
4.4.4	デバイスへのアクセスの制限 .....	4-25

4.4.5	環境へのアクセスの制限 .....	4-26
4.4.6	リスト内の ACE の順序 .....	4-26
4.4.7	ファイルの継承方式の設定 .....	4-27
4.4.8	ACL の表示 .....	4-29
4.4.9	既存の ACL への ACE の追加 .....	4-30
4.4.10	ACL の削除 .....	4-31
4.4.11	ACL からの ACE の削除 .....	4-31
4.4.12	ACL の部分的な置き換え .....	4-31
4.4.13	ファイルのデフォルト ACL の復元 .....	4-32
4.4.14	ACL のコピー .....	4-32
4.5	保護コードによるアクセスの制御 .....	4-33
4.5.1	保護コードの形式 .....	4-33
4.5.2	保護コード内のアクセスのタイプ .....	4-34
4.5.3	保護コードの処理 .....	4-35
4.5.4	保護コードの変更 .....	4-36
4.5.5	機密オブジェクトに対する保護の強化 .....	4-37
4.5.6	ディレクトリ構造に対するデフォルトの保護コードの提供 .....	4-37
4.5.7	ファイルのデフォルト・セキュリティ・プロファイルの復元 ....	4-38
4.6	特権と制御アクセス .....	4-39
4.6.1	保護メカニズムに対する特権の影響 .....	4-39
4.6.2	制御アクセスによるオブジェクトのプロファイルの変更 .....	4-39
4.6.3	アクセスに関するオブジェクト固有の考慮事項 .....	4-40
4.7	保護オブジェクトの監査 .....	4-40
4.7.1	システムが監査するイベントの種類 .....	4-41
4.7.2	オブジェクトのクラスに対する監査の有効化 .....	4-41
4.7.3	セキュリティ 監査用 ACE の追加 .....	4-41

## 5 オブジェクト・クラスの詳細

5.1	ケーパビリティ .....	5-1
5.1.1	命名規則 .....	5-1
5.1.2	アクセスのタイプ .....	5-1
5.1.3	テンプレート・プロファイル .....	5-2
5.1.4	実行される監査の種類 .....	5-2
5.1.5	オブジェクトの永続性 .....	5-2
5.2	コモン・イベント・フラグ・クラスタ .....	5-2
5.2.1	命名規則 .....	5-3
5.2.2	アクセスのタイプ .....	5-3

5.2.3	テンプレート・プロファイル .....	5-3
5.2.4	必要な特権 .....	5-3
5.2.5	実行される監査の種類 .....	5-4
5.2.6	オブジェクトの永続性 .....	5-4
5.3	デバイス .....	5-4
5.3.1	命名規則 .....	5-4
5.3.2	アクセスのタイプ .....	5-5
5.3.3	入出力操作に必要なアクセス権 .....	5-5
5.3.4	テンプレート・プロファイル .....	5-7
5.3.5	新しいデバイスに対するプロファイルの設定 .....	5-7
5.3.6	必要な特権 .....	5-9
5.3.7	実行される監査の種類 .....	5-9
5.3.8	オブジェクトの永続性 .....	5-10
5.4	ファイル .....	5-10
5.4.1	命名規則 .....	5-10
5.4.2	アクセスのタイプ .....	5-10
5.4.3	必要なアクセス権 .....	5-11
5.4.4	作成時の要件 .....	5-12
5.4.5	プロファイルの割り当て .....	5-13
5.4.5.1	所有権の割り当て規則 .....	5-13
5.4.5.2	保護コードと ACL の割り当て規則 .....	5-13
5.4.5.3	COPY コマンドと RENAME コマンドの使用 .....	5-15
5.4.6	実行される監査の種類 .....	5-15
5.4.7	ディスク領域再割り当て時の情報の保護 .....	5-16
5.4.7.1	ディスク・ブロックの上書き .....	5-16
5.4.7.2	ハイウォータ・マークの設定 .....	5-17
5.4.7.3	ファイル内のデータのアクセス制御 .....	5-17
5.4.8	ファイル・セキュリティの最適化に関する推奨事項 .....	5-18
5.5	グローバル・セクション .....	5-19
5.5.1	命名規則 .....	5-19
5.5.2	アクセスのタイプ .....	5-19
5.5.3	テンプレート・プロファイル .....	5-20
5.5.4	必要な特権 .....	5-20
5.5.5	実行される監査の種類 .....	5-20
5.5.6	オブジェクトの永続性 .....	5-21
5.6	論理名テーブル .....	5-21

5.6.1	命名規則 .....	5-21
5.6.2	アクセスのタイプ .....	5-21
5.6.3	テンプレート・プロファイル .....	5-22
5.6.4	必要な特権 .....	5-22
5.6.5	実行される監査の種類 .....	5-22
5.6.6	オブジェクトの永続性 .....	5-23
5.7	キュー .....	5-23
5.7.1	命名規則 .....	5-23
5.7.2	アクセスのタイプ .....	5-23
5.7.3	テンプレート・プロファイル .....	5-24
5.7.4	必要な特権 .....	5-24
5.7.5	実行される監査の種類 .....	5-24
5.7.6	オブジェクトの永続性 .....	5-24
5.8	資源ドメイン .....	5-25
5.8.1	命名規則 .....	5-25
5.8.2	アクセスのタイプ .....	5-25
5.8.3	テンプレート・プロファイル .....	5-25
5.8.4	必要な特権 .....	5-26
5.8.5	実行される監査の種類 .....	5-26
5.8.6	オブジェクトの永続性 .....	5-26
5.9	セキュリティ・クラス .....	5-26
5.9.1	命名規則 .....	5-26
5.9.2	アクセスのタイプ .....	5-27
5.9.3	テンプレート・プロファイル .....	5-27
5.9.4	実行される監査の種類 .....	5-27
5.9.5	オブジェクトの永続性 .....	5-28
5.10	ボリューム .....	5-28
5.10.1	命名規則 .....	5-28
5.10.2	アクセスのタイプ .....	5-28
5.10.3	テンプレート・プロファイル .....	5-29
5.10.4	必要な特権 .....	5-29
5.10.5	実行される監査の種類 .....	5-29
5.10.6	オブジェクトの永続性 .....	5-29

## Part 3 システム管理者のためのセキュリティ



## 6 システムとそのデータの管理

6.1	セキュリティ管理者の役割 .....	6-1
6.2	サイトのセキュリティ・ポリシー .....	6-2
6.3	安全なシステムを設定するためのツール .....	6-5
6.4	セキュリティ管理者のアカウント要件 .....	6-5
6.5	新規ユーザのトレーニング .....	6-6
6.6	ユーザのセッションのログ取得 .....	6-7
6.7	安全なシステムを維持するための継続的な作業 .....	6-9

## 7 システム・アクセスの管理

7.1	システムにアクセス可能な時間と条件の定義 .....	7-1
7.1.1	作業時間の制限 .....	7-2
7.1.2	操作モードの制限 .....	7-3
7.1.3	アカウントの有効期間の制限 .....	7-3
7.1.4	アカウントの無効化 .....	7-4
7.1.5	ディスク・ボリュームの制限 .....	7-4
7.1.6	外部認証用アカウントのマーク付け .....	7-4
7.2	ユーザへの適切なアカウントの割り当て .....	7-4
7.2.1	システム・アカウントのタイプ .....	7-5
7.2.1.1	会話型アカウントの例 .....	7-6
7.2.1.2	限定アクセス・アカウントの例 .....	7-7
7.2.2	特権アカウント .....	7-8
7.2.3	会話型アカウント .....	7-9
7.2.4	キャプティブ・アカウント .....	7-9
7.2.4.1	キャプティブ・アカウントの設定 .....	7-10
7.2.4.2	キャプティブ・コマンド・プロシージャのガイドライン .....	7-11
7.2.5	制限付きアカウント .....	7-13
7.2.6	自動ログイン・アカウント .....	7-14
7.2.7	ゲスト・アカウント .....	7-15
7.2.8	代理アカウント .....	7-16
7.2.9	外部認証アカウント .....	7-16
7.3	パスワードを使用したシステム・アクセスの制御 .....	7-17
7.3.1	パスワードのタイプ .....	7-17
7.3.1.1	第1パスワード .....	7-17
7.3.1.2	システム・パスワード .....	7-18

7.3.1.3	第2パスワード .....	7-20
7.3.1.4	コンソール・パスワード .....	7-21
7.3.1.5	認証カード .....	7-22
7.3.2	最低限のパスワード基準の適用 .....	7-23
7.3.2.1	パスワードの有効期限 .....	7-23
7.3.2.2	期限切れのパスワードの強制変更 .....	7-24
7.3.2.3	パスワードに必要な最低限の長さ .....	7-25
7.3.2.4	生成パスワード .....	7-26
7.3.2.5	サイト・パスワードのアルゴリズム .....	7-26
7.3.3	新しいパスワードの検査 .....	7-27
7.3.3.1	システム辞書 .....	7-27
7.3.3.2	履歴リスト .....	7-27
7.3.3.3	サイト固有のフィルタ .....	7-28
7.3.4	パスワード保護のチェックリスト .....	7-29
7.4	外部認証の有効化 .....	7-31
7.4.1	外部認証のオーバーライド .....	7-32
7.4.2	レイヤード・プロダクトおよびアプリケーションに対する影響 ..	7-33
7.4.3	新しいパスワードの設定 .....	7-34
7.4.4	パスワードとユーザ名における大文字小文字の区別 .....	7-34
7.4.5	ユーザ名マッピングおよびパスワード検証 .....	7-35
7.4.6	パスワード同期 .....	7-36
7.4.7	SYS\$SINGLE_SIGNON 論理名ビットの指定 .....	7-36
7.4.8	ACME (Authentication and Credentials Management Extensions) サブシステム .....	7-39
7.4.8.1	ACME サブシステムの概要 .....	7-39
7.4.8.1.1	ACME エージェントの動作環境 .....	7-40
7.4.8.1.2	ACME エージェントの順序 .....	7-40
7.4.8.2	認証ポリシー .....	7-41
7.4.8.2.1	OpenVMS ポリシー .....	7-41
7.4.8.2.2	Advanced Server for OpenVMS のポリシー .....	7-42
7.4.8.3	ACME サブシステムの制御 .....	7-42
7.4.8.3.1	SET および SHOW SERVER ACME コマンド .....	7-43
7.4.8.3.2	新しいSYSUAF フラグ .....	7-43
7.4.8.3.3	新しいシステム・パラメータ SECURITY_POLICY の ビット・マスク値 .....	7-44
7.5	ログイン・プロセスの制御 .....	7-44
7.5.1	ログイン時の情報表示 .....	7-45

7.5.1.1	通知メッセージ .....	7-45
7.5.1.2	ウェルカム・メッセージ .....	7-45
7.5.1.3	最終ログイン・メッセージ .....	7-45
7.5.1.4	新着メールの通知 .....	7-46
7.5.2	切断されたプロセスの限定 .....	7-46
7.5.3	自動ログインの実現 .....	7-47
7.5.4	安全ターミナル・サーバの使用 .....	7-47
7.5.5	侵入者の検出 .....	7-48
7.5.6	侵入データベースについて .....	7-49
7.5.6.1	侵入検出の仕組み .....	7-51
7.5.6.2	除外期間の設定 .....	7-52
7.5.6.3	ログイン試行を制御するシステム・パラメータ .....	7-52
7.5.7	セキュリティ・サーバ・プロセス .....	7-53

## 8 システムのデータと資源へのアクセスの制御

8.1	ユーザ・グループの設計 .....	8-1
8.1.1	UIC グループの設計の例 .....	8-1
8.1.2	UIC グループの設計に関する制約 .....	8-3
8.2	ACL での個別ユーザの指定 .....	8-3
8.3	権限の共有の定義 .....	8-4
8.4	ユーザごとの識別子の条件指定 .....	8-4
8.5	ACL の設計 .....	8-5
8.6	ライト・データベースへの登録 .....	8-6
8.6.1	データベースの表示 .....	8-7
8.6.2	識別子の追加 .....	8-7
8.6.3	ライト・データベースの復元 .....	8-8
8.6.4	ユーザへの識別子の割り当て .....	8-8
8.6.5	保持者レコードの削除 .....	8-8
8.6.6	識別子の削除 .....	8-9
8.6.7	識別子のカスタマイズ .....	8-9
8.6.7.1	Dynamic 属性 .....	8-10
8.6.7.2	Holder Hidden 属性 .....	8-11
8.6.7.3	Name Hidden 属性 .....	8-11
8.6.7.4	No Access 属性 .....	8-12
8.6.7.5	Resource 属性 .....	8-12
8.6.7.6	Subsystem 属性 .....	8-13

8.6.8	システムまたはプロセス・ライト・リストの変更 .....	8-14
8.7	ユーザへの特権の付与 .....	8-14
8.7.1	特権のカテゴリ .....	8-15
8.7.2	推奨される特権の割り当て .....	8-16
8.7.3	ユーザ特権の制限 .....	8-17
8.7.4	特権イメージのインストール .....	8-18
8.7.5	コマンド出力の制限 .....	8-19
8.8	デフォルトの保護と所有権の設定 .....	8-19
8.8.1	ファイル・アクセスの制御 .....	8-19
8.8.1.1	保護のデフォルトの調整 .....	8-23
8.8.1.2	資源識別子により所有されるディレクトリのデフォルトの設 定 .....	8-25
8.8.1.2.1	資源識別子の設定 .....	8-25
8.8.1.2.2	資源識別子のディレクトリの設定 .....	8-26
8.8.1.2.3	ACL の設定 .....	8-26
8.8.2	ファイル以外のオブジェクトのデフォルトの設定 .....	8-27
8.8.2.1	クラスのデフォルトの表示 .....	8-28
8.8.2.2	クラス・テンプレートの変更 .....	8-29
8.9	システムのデータと資源の追加保護 .....	8-30
8.9.1	ソフトウェアの新規インストール時に必要な安全対策 .....	8-30
8.9.1.1	潜在的に有害なプログラム .....	8-30
8.9.1.2	特権を与えてのプログラムのインストール .....	8-32
8.9.2	システム・ファイルの保護 .....	8-32
8.9.3	DCL コマンドの使用の制限 .....	8-34
8.9.4	ファイルの暗号化 .....	8-35
8.9.5	ディスクの保護 .....	8-35
8.9.5.1	除去テクニック .....	8-35
8.9.5.2	ハイウォータ・マーク処理による防止 .....	8-36
8.9.5.3	防止テクニックの要約 .....	8-37
8.9.6	バックアップ・メディアの保護 .....	8-37
8.9.6.1	ディスクのバックアップ .....	8-37
8.9.6.2	バックアップ・セーブ・セットの保護 .....	8-38
8.9.6.3	バックアップ・セーブ・セットからのファイルの取り出し ..	8-39
8.9.7	ターミナルの保護 .....	8-39
8.9.7.1	ターミナルの使用制限 .....	8-39
8.9.7.2	アプリケーション・ターミナルなどのデバイスの制限 .....	8-40
8.9.7.3	モデム用のターミナル回線の設定 .....	8-40

## 9 セキュリティ監査の実施

9.1	監査プロセスの概要 .....	9-1
9.2	セキュリティ関連イベントの報告 .....	9-2
9.2.1	監査情報の生成方法 .....	9-2
9.2.1.1	活動の監査のカテゴリ .....	9-3
9.2.1.2	セキュリティ監査 ACE の関連付け .....	9-6
9.2.1.3	ユーザ登録レコードの変更 .....	9-8
9.2.2	オペレーティング・システムが報告できるシステム活動の種類 ..	9-8
9.2.2.1	一部の特権監査の抑制 .....	9-9
9.2.2.2	一部のプロセス制御監査の抑制 .....	9-10
9.2.3	イベント情報のソース .....	9-11
9.3	監査計画の策定 .....	9-12
9.3.1	監査要件の評価 .....	9-12
9.3.2	イベント・メッセージの出力先の選択 .....	9-15
9.3.3	性能への影響の考慮 .....	9-15
9.4	イベント・メッセージの取得方法 .....	9-16
9.4.1	監査ログ・ファイルの使用 .....	9-16
9.4.1.1	ファイルの保守 .....	9-17
9.4.1.2	システム・ディスクからのファイルの移動 .....	9-18
9.4.2	ターミナルのアラーム受信の有効化 .....	9-19
9.4.3	イベント・メッセージの第 2 の出力先 .....	9-20
9.4.3.1	遠隔ログ・ファイルの使用 .....	9-20
9.4.3.2	リスナ・メールボックスの使用 .....	9-21
9.5	ログ・ファイルの分析 .....	9-22
9.5.1	推奨される手順 .....	9-22
9.5.2	監査分析ユーティリティの起動 .....	9-24
9.5.3	レポートの指定 .....	9-24
9.5.4	会話形式での監査分析ユーティリティの使用 .....	9-27
9.5.5	レポートの調査 .....	9-28
9.6	監査サブシステムの管理 .....	9-29
9.6.1	監査サーバにより実行されるタスク .....	9-30
9.6.2	監査サーバのスタートアップの無効化と再有効化 .....	9-31
9.6.3	スタートアップにおける、オペレーティング・システムが監査を 開始するポイントの変更 .....	9-32
9.6.4	プロセス中断のきっかけとなる未処理メッセージの数の指定 ....	9-33

9.6.4.1	メッセージのフロー制御 .....	9-33
9.6.4.2	プロセスの一時中断の防止 .....	9-34
9.6.5	メモリ不足への対応 .....	9-34
9.6.6	メッセージの正確なタイムスタンプ設定の維持 .....	9-35
9.6.7	ディスクへのメッセージ転送の調整 .....	9-35
9.6.8	監査ログ・ファイル用のディスク領域の割り当て .....	9-36
9.6.9	監査機能におけるエラー処理 .....	9-37
9.6.9.1	ディスク監視の無効化 .....	9-37
9.6.9.2	遠隔ログ・ファイルへのリンクの喪失 .....	9-37
 <b>10 システムのセキュリティ侵害</b>		
10.1	システム攻撃の形態 .....	10-1
10.2	問題の兆候 .....	10-2
10.2.1	ユーザからの報告 .....	10-2
10.2.2	システムの監視 .....	10-3
10.3	システムの定期的な監視 .....	10-4
10.3.1	システムの会計記録 .....	10-4
10.3.2	セキュリティ監査の実施 .....	10-4
10.4	セキュリティ侵害への対処 .....	10-7
10.4.1	失敗に終わった侵入行為 .....	10-7
10.4.1.1	侵入行為の検出 .....	10-7
10.4.1.2	実行者の特定 .....	10-8
10.4.1.3	侵入行為の防止 .....	10-8
10.4.2	成功した侵入 .....	10-9
10.4.2.1	成功した侵害実行者の特定 .....	10-9
10.4.2.2	システムのセキュリティ保護 .....	10-10
10.4.2.3	侵入を許したあとの復旧 .....	10-11
 <b>11 クラスタのセキュリティ保護</b>		
11.1	クラスタの概要 .....	11-1
11.2	共通環境の構築 .....	11-2
11.2.1	必須の共通システム・ファイル .....	11-2
11.2.2	推奨される共通システム・ファイル .....	11-3
11.2.3	複数のバージョンが存在するファイルの同期 .....	11-4
11.3	登録データの同期 .....	11-6
11.4	監査ログ・ファイルの管理 .....	11-8
11.5	オブジェクトの保護 .....	11-8

11.6	プロファイルおよび監査情報の格納 .....	11-9
11.7	クラスタ全体での侵入検出 .....	11-9
11.8	システム管理ユーティリティの使用 .....	11-10
11.9	クラスタ所属の管理 .....	11-10
11.10	クラスタ・ノード間での DECnet の使用 .....	11-11
<b>12</b>	<b>ネットワーク環境におけるセキュリティ</b>	
12.1	ネットワーク・セキュリティの管理 .....	12-1
12.1.1	セキュリティ確保のための要件 .....	12-2
12.1.2	ネットワークにおける監査 .....	12-3
12.2	アクセス制御の階層 .....	12-3
12.2.1	明示的アクセス制御の使用 .....	12-4
12.2.2	代理ログインの使用 .....	12-4
12.2.3	デフォルト・アプリケーション・アカウントの使用 .....	12-5
12.3	代理アクセス制御 .....	12-5
12.3.1	代理アクセスに関する特別なセキュリティ対策 .....	12-6
12.3.2	代理データベースの設定 .....	12-7
12.3.2.1	着信代理アクセスの有効化および無効化 .....	12-8
12.3.2.2	代理アクセスの削除 .....	12-9
12.3.2.3	代理アカウントの作成手順 .....	12-10
12.3.3	代理アカウントの例 .....	12-10
12.4	DECnet アプリケーション (オブジェクト) アカウントの使用 .....	12-12
12.4.1	ネットワーク・オブジェクトのまとめ .....	12-12
12.4.2	手作業でのネットワーク・オブジェクトの設定 .....	12-14
12.4.3	システムへのデフォルトの DECnet アクセスの削除 .....	12-17
12.4.4	遠隔オブジェクト接続の特権要件の設定 .....	12-18
12.5	ルーティング初期化パスワードの指定 .....	12-18
12.5.1	動的非同期接続の確立 .....	12-19
12.6	ネットワークにおけるファイルの共用 .....	12-24
12.6.1	メール・ユーティリティの使用 .....	12-24
12.6.2	ローカル・ユーザおよび遠隔ユーザのアカウントの設定 .....	12-25
12.6.3	複数アカウントに対する遠隔ユーザの許可 .....	12-26
<b>13</b>	<b>保護サブシステムの使用</b>	
13.1	保護サブシステムの利点 .....	13-1
13.2	保護サブシステムの適用範囲 .....	13-2

13.3	保護サブシステムの仕組み .....	13-2
13.4	設計に関する検討事項 .....	13-3
13.5	システム管理の要件 .....	13-4
13.6	サブシステムの構築 .....	13-5
13.7	トラステッド・ボリュームにおける保護サブシステムの有効化 .....	13-6
13.8	ユーザへのアクセス権の付与 .....	13-7
13.9	保護サブシステムの例 .....	13-8
13.9.1	最上位ディレクトリの保護 .....	13-9
13.9.2	サブシステムのディレクトリの保護 .....	13-10
13.9.3	イメージおよびデータ・ファイルの保護 .....	13-11
13.9.4	プリンタの保護 .....	13-12
13.9.5	サブシステム構築のためのコマンド・プロシージャ .....	13-12

## A 特権の割り当て

A.1	ACNT 特権 (Devour) .....	A-2
A.2	ALLSPOOL 特権 (Devour) .....	A-2
A.3	ALTPRI 特権 (System) .....	A-2
A.4	AUDIT 特権 (System) .....	A-3
A.5	BUGCHK 特権 (Devour) .....	A-3
A.6	BYPASS 特権 (All) .....	A-3
A.7	CMEXEC 特権 (All) .....	A-5
A.8	CMKRNL 特権 (All) .....	A-6
A.9	DIAGNOSE 特権 (Objects) .....	A-7
A.10	DOWNGRADE 特権 (All) .....	A-7
A.11	EXQUOTA 特権 (Devour) .....	A-8
A.12	GROUP 特権 (Group) .....	A-8
A.13	GRPNAM 特権 (Devour) .....	A-8
A.14	GRPPRV 特権 (Group) .....	A-9
A.15	IMPERSONATE 特権 (All) (旧名称 DETACH) .....	A-10
A.16	IMPORT 特権 (Objects) .....	A-11
A.17	LOG_IO 特権 (All) .....	A-11
A.18	MOUNT 特権 (Normal) .....	A-12
A.19	NETMBX 特権 (Normal) .....	A-12
A.20	OPER 特権 (System) .....	A-12
A.21	PFNMAP 特権 (All) .....	A-16
A.22	PHY_IO 特権 (All) .....	A-16
A.23	PRMCEB 特権 (Devour) .....	A-18



A.24	PRMGBL 特権 (Devour) .....	A-18
A.25	PRMMBX 特権 (Devour) .....	A-18
A.26	PSWAPM 特権 (System) .....	A-19
A.27	READALL 特権 (Objects) .....	A-19
A.28	SECURITY 特権 (System) .....	A-20
A.29	SETPRV 特権 (All) .....	A-20
A.30	SHARE 特権 (All) .....	A-21
A.31	SHMEM 特権 (Devour) .....	A-21
A.32	SYSGBL 特権 (Files) .....	A-21
A.33	SYSLCK 特権 (System) .....	A-21
A.34	SYSNAM 特権 (All) .....	A-22
A.35	SYSPRV 特権 (All) .....	A-22
A.36	TMPMBX 特権 (Normal) .....	A-24
A.37	UPGRADE 特権 (All) .....	A-24
A.38	VOLPRO 特権 (Objects) .....	A-24
A.39	WORLD 特権 (System) .....	A-25
 <b>B OpenVMS システム・ファイルの保護</b>		
B.1	標準の所有権と保護 .....	B-1
B.2	OpenVMS システム・ファイルの一覧 .....	B-3
B.2.1	最上位ディレクトリのファイル .....	B-3
B.2.2	SYS\$KEYMAP のファイル .....	B-3
B.2.3	SYS\$LDR のファイル .....	B-4
B.2.4	SYS\$STARTUP および SYS\$ERR のファイル .....	B-6
B.2.5	SYSEXE のファイル .....	B-7
B.2.6	SYSHLP のファイル .....	B-9
B.2.7	SYSLIB のファイル .....	B-12
B.2.8	SYSMGR のファイル .....	B-14
B.2.9	SYSMGR のファイル .....	B-14
B.2.10	SYSTEST のファイル .....	B-15
B.2.11	SYSUPD のファイル .....	B-16
B.2.12	VUE\$LIBRARY のファイル .....	B-16
 <b>C C2 環境における OpenVMS システムの運用</b>		
C.1	C2 システムについて .....	C-1
C.1.1	C2 環境の定義 .....	C-1
C.1.1.1	ドキュメント .....	C-2

C.2	C2 システム向けのトラステッド・コンピューティング・ベース (TCB) .....	C-2
C.2.1	TCB に含まれるハードウェア .....	C-3
C.2.2	TCB に含まれるソフトウェア .....	C-3
C.2.3	オブジェクトの保護 .....	C-5
C.2.4	TCB の保護 .....	C-6
C.2.4.1	ファイルの保護 .....	C-6
C.2.4.2	信頼できるユーザに付与する特権 .....	C-6
C.2.4.3	信頼の低いユーザに付与する特権 .....	C-7
C.2.4.4	物理的セキュリティ .....	C-7
C.2.5	C2 システムの設定 .....	C-8
C.2.5.1	ユーザを確実に特定できることの保証 .....	C-9
C.2.5.2	監査証跡の管理 .....	C-10
C.2.5.3	オブジェクトの再利用 .....	C-11
C.2.5.4	クラスタの設定 .....	C-12
C.2.5.5	システムのスタートアップと運用 .....	C-13
C.2.5.6	アクセス権変更後の指定サブジェクトの即時再認証の実行 ..	C-14
C.3	C2 システム構築のためのチェックリスト .....	C-15

## D アラーム・メッセージ

### 用語一覧

### 索引

### 例

3-1	ローカル・ログイン・メッセージ .....	3-6
4-1	プロセスの許可された特権とデフォルト特権 .....	4-10
6-1	セキュリティ管理者のアカウントの例 .....	6-5
7-1	一般的な会話型ユーザ・アカウントの作成 .....	7-6
7-2	限定アクセス・アカウントの作成 .....	7-7
7-3	特権アカウント用のキャプティブ・プロシージャの例 .....	7-12
7-4	非特権アカウント用のキャプティブ・コマンド・プロシージャの例 ..	7-12
7-5	侵入データベースの表示 .....	7-50
9-1	アラーム・メッセージのサンプル .....	9-2
9-2	オブジェクト・アクセス・イベントにより作成される監査 .....	9-5
9-3	セキュリティ要件が中程度であるサイトのイベントの監査 .....	9-14
9-4	簡略監査レポート .....	9-26

9-5	完全な監査レポートの1つのレコード .....	9-27
9-6	監査ログ・ファイルのイベントの要約 .....	9-27
9-7	監査レポートにある疑わしい活動の特定 .....	9-28
9-8	疑わしいレコードの調査 .....	9-29
9-9	監査サーバのデフォルトの特性 .....	9-30
12-1	代理アカウントの例 .....	12-11
12-2	MAIL\$SERVER アカウントのUAF レコード .....	12-16
12-3	動的非同期接続のコマンドの例 .....	12-23
12-4	保護ファイルのネットワークにおける共用 .....	12-26
13-1	メンバ・リスト管理用の識別子とアプリケーション・アクセス権の設定 .....	13-5
13-2	SUPPLIERS_SUBSYSTEM.DIR の保護 .....	13-9
13-3	SYS\$SYSDEVICE:[SUPPLIERS_SUBSYSTEM] の保護 .....	13-10
13-4	サブシステムの ORDERS.EXE イメージおよび PAYMENTS.EXE イメージへのアクセス .....	13-11
13-5	キューの保護 .....	13-12
13-6	サブシステム・コマンド・プロシージャ .....	13-12
B-1	SYS\$KEYMAP のファイル .....	B-4
B-2	SYS\$LDR のファイル .....	B-4
B-3	SYS\$STARTUP および SYS\$ERR のファイル .....	B-6
B-4	SYSEXEC のファイル .....	B-7
B-5	SYSHLP のファイル .....	B-9
B-6	SYSLIB のファイル .....	B-12
B-7	SYSMGR のファイル .....	B-14
B-8	SYMSMSG のファイル .....	B-14
B-9	SYSTEST のファイル .....	B-15
B-10	SYSUPD のファイル .....	B-16
B-11	VUE\$LIBRARY のファイル .....	B-16



2-1	リファレンス・モニタ .....	2-2
2-2	登録アクセス・マトリックス .....	2-8
2-3	交点にラベルを付けた登録アクセス・マトリックス .....	2-9
4-1	以前のスレッド別セキュリティのモデル .....	4-3
4-2	スレッド別セキュリティ・プロファイルのモデル .....	4-4
4-3	アクセス要求評価のフローチャート .....	4-19
4-4	アクセス要求評価のフローチャート(続き) .....	4-20

4-5	アクセス要求評価のフローチャート(続き) .....	4-21
4-6	アクセス要求評価のフローチャート(続き) .....	4-22
4-7	アクセス要求評価のフローチャート(続き) .....	4-23
8-1	ファイル作成のフローチャート .....	8-21
8-2	ファイル作成のフローチャート .....	8-22
8-3	ファイル作成のフローチャート .....	8-23
8-4	セキュリティ・クラス・オブジェクト .....	8-28
12-1	ネットワークにおける参照モニタ .....	12-2
12-2	典型的な動的非同期接続 .....	12-23
13-1	通常のアクセス制御と保護サブシステムとの違い .....	13-3
13-2	Taylor 社のサブシステムのディレクトリ構造 .....	13-9

## 表

1-1	セキュリティ要件の基準となるイベント許容度 .....	1-3
2-1	セキュリティ制御によって保護されるオブジェクト .....	2-5
2-2	セキュリティ監査機能の概要 .....	2-7
3-1	安全なパスワードと安全でないパスワード .....	3-1
3-2	パスワードのタイプ .....	3-3
3-3	ログイン失敗の原因 .....	3-9
4-1	主なライト識別子のタイプ .....	4-6
4-2	保護オブジェクトのクラス .....	4-16
6-1	サイトのセキュリティ・ポリシーの例 .....	6-3
7-1	ログインの時間と条件を制御する AUTHORIZE 修飾子 .....	7-1
7-2	キャプティブ・アカウントにより許可されないログイン修飾子 .....	7-9
7-3	キャプティブ・アカウントの定義に必要な修飾子 .....	7-10
7-4	パスワード履歴リストのデフォルト .....	7-28
7-5	SYS\$SINGLE_SIGNON 論理名ビット .....	7-36
7-6	侵入の例 .....	7-51
7-7	ログイン試行を制御するためのパラメータ .....	7-52
8-1	部署と職務による従業員のグループ分け .....	8-2
8-2	OpenVMS の特権 .....	8-15
8-3	システム・ユーザの最低限の特権 .....	8-17
8-4	ファイル保護に使用する DCL コマンド .....	8-33
9-1	デフォルトで監査されるイベント・クラス .....	9-4
9-2	セキュリティ監査用のアクセス制御エントリ (ACE) .....	9-7
9-3	システムが報告できるセキュリティ・イベントの種類 .....	9-8
9-4	サイトのセキュリティ要件に応じて監視すべきイベント .....	9-13

9-5	監査ログ・ファイルの特徴 .....	9-17
9-6	監査分析ユーティリティの修飾子 .....	9-25
9-7	監査イベント・メッセージのフロー制御 .....	9-33
10-1	ACL ベースの監査が有効なシステム・ファイル .....	10-6
11-1	クラスタ内で一本化する必要の必須システム・ファイル .....	11-3
11-2	共通化が推奨されるシステム・ファイル .....	11-4
11-3	複数バージョンの必須クラスタ・ファイルの使用 .....	11-5
11-4	同期をとる必要のある SYSUAF.DAT のフィールド .....	11-7
11-5	クラスタにおけるオブジェクトの動作 .....	11-8
12-1	ネットワーク代理アクセスの管理に使用する AUTHORIZE コマンド	12-8
12-2	ネットワーク・オブジェクトのデフォルト設定 .....	12-15
B-1	標準の OpenVMS システム・ファイル保護の例外 .....	B-1
C-1	C2 の評価済みシステムに含まれていないソフトウェア .....	C-3
C-2	信頼の低いユーザに付与する特権 .....	C-7



## 対象読者

本書は、許可のないユーザによる改ざん、参照、サービスの盗用からオペレーティング・システムを保護する責任のあるユーザおよびシステム管理者を対象としています。本書では、システム・セキュリティの責任者のことを「セキュリティ管理者」と呼びます。

## 本書の構成

本書では次の内容を取り上げます。

- Part 1:  
セキュリティ管理者向けに、セキュリティ問題の概略、概念設計機能、OpenVMS システムに固有のセキュリティ機能を説明します。
  - 第 1 章では、セキュリティ要件のレベルと、セキュリティ障害の 3 つの原因について説明します。
  - 第 2 章では、セキュリティ設計におけるリファレンス・モニタの概念と、オペレーティング・システムのセキュリティ機能の概要について説明します。
- Part 2:  
一般ユーザ向けのセキュリティ措置と機能について説明します。
  - 第 3 章では、ログインとログアウトの手続きと、パスワードの責任ある利用についての、一般ユーザ向けの情報を取り上げます。
  - 第 4 章と第 5 章では、オブジェクト保護機能について詳しく説明します。
- Part 3:  
セキュリティ管理者向けのセキュリティ措置と機能について説明します。
  - 第 6 章では、セキュリティ管理者の一般的な作業について説明します。
  - 第 7 章では、システム・アクセスを制御する手法について説明します。
  - 第 8 章では、システムのデータと資源へのアクセスを制御する手法について説明します。
  - 第 9 章では、セキュリティ監査機能について説明します。
  - 第 10 章では、システムが攻撃されていることを検出する方法と、システムを保護し、防御する方法について説明します。
  - 第 11 章では、共通のシステム・ファイルのセット・アップや認証データの同期化など、クラスタ環境に固有のセキュリティ関連措置について説明します。

- 第 12 章では、ネットワークを利用しているシステムのセキュリティに関する考慮事項を取り上げます。
- 第 13 章では、保護サブシステムのセット・アップと管理の方法について説明します。
- 付録 A では、オペレーティング・システムで利用可能なすべてのユーザ特権の概略と、ユーザ特権を必要とするユーザについて説明します。
- 付録 B では、クリティカルなシステム・ファイルのために HP が提供している、保護コードと所有権を紹介します。
- 付録 C では、大分類 (Division) C、中分類 (Class) 2 (C2) のセキュリティ環境で OpenVMS システムを運用する方法について説明します。
- 付録 D では、セキュリティ・アラーム・メッセージの例を取り上げます。
- 用語一覧では、本書で取り上げているセキュリティ関連用語の定義を示します。

## 関連資料

『*OpenVMS システム・セキュリティ・ガイド*』では、読者は『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』にある、次のセキュリティ関連ユーティリティに関する内容を理解していることを前提としています。

- アクセス制御リスト・エディタ (ACL エディタ)
- 会計情報ユーティリティ
- 監査分析ユーティリティ
- 登録ユーティリティ
- バックアップ・ユーティリティ
- システム管理 (SYSMAN) ユーティリティ

次のマニュアルのセキュリティ情報も参考になります。

- 『*HP Open Source Security for OpenVMS, Volume 1: CDSA*』
- 『*HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS*』
- 『*HP Open Source Security for OpenVMS, Volume 3: Kerberos*』
- 『*OpenVMS DCL デイクシヨナリ*』
- 『*OpenVMS システム管理者マニュアル*』
- 『*OpenVMS Cluster システム*』

HP OpenVMS 製品およびサービスの詳細については、弊社の Web サイトにアクセスしてください。アドレスは次のとおりです。

<http://www.hp.com/go/openvms/>

## 本書で使用する表記法

本書では、次の表記法を使用しています。



表記法	意味
Ctrl/x	Ctrl/x という表記は、Ctrl キーを押しながら別のキーまたはポインティング・デバイス・ボタンを押すことを示します。
PF1 x	PF1 x という表記は、PF1 に定義されたキーを押してから、別のキー (x) またはポインティング・デバイス・ボタンを押すことを示します。
Return	例の中で、キー名が太字で書かれている場合には、そのキーを押すことを示します。
...	例の中の水平方向の反復記号は、次のいずれかを示します。 <ul style="list-style-type: none"> <li>• 文中のオプションの引数が省略されている。</li> <li>• 直前の 1 つまたは複数の項目を繰り返すことができる。</li> <li>• パラメータや値などの情報をさらに入力できる。</li> </ul>
・	垂直方向の反復記号は、コードの例やコマンド形式の中の項目が省略されていることを示します。このように項目が省略されるのは、その項目が説明している内容にとって重要ではないからです。
( )	コマンドの形式の説明において、括弧は、複数のオプションを選択した場合に、選択したオプションを括弧で囲まなければならないことを示しています。
[ ]	コマンドの形式の説明において、大括弧で囲まれた要素は省略可能な選択肢です。項目をすべて選択しても、いずれか 1 つを選択しても、あるいは 1 つも選択しなくても構いません。コマンド行には、大括弧は入力しないでください。ただし、OpenVMS ファイル指定のディレクトリ名の構文や、割り当て文の部分文字列指定の構文の中では、大括弧も含めて入力しなければなりません。
	コマンド形式の説明では、縦棒は大括弧や中括弧内の選択肢を区切っています。大括弧内の選択肢は省略可能ですが、中括弧内の選択肢は少なくとも 1 つ選択する必要があります。コマンド行には、縦棒は入力しないでください。
{ }	コマンドの形式の説明において、中括弧で囲まれた選択肢は必須なので、いずれか 1 つを選択しなければなりません。コマンド行には、中括弧は入力しないでください。
太字体	太字体のテキストは、新しい用語、引数、属性、条件を示しています。
<i>italic type</i>	イタリック体は、重要な情報を示します。また、システム・メッセージ (たとえば内部エラー <i>number</i> )、コマンド・ライン (たとえば /PRODUCER= <i>name</i> )、コマンド・パラメータ (たとえば <i>device-name</i> ) などの変数を示す場合にも使用されます。
UPPERCASE TYPE	英大文字は、コマンド、ルーチン名、ファイル名、システム特権の短縮形を示します。
Example	この字体は、コード例、コマンド例、および対話型の画面表示を示します。テキスト内では、この字体は URL、UNIX® のコマンドとパス名、PC ベースのコマンドとフォルダ、および C プログラミング言語の要素も示します。
—	コマンド形式の記述の最後、コマンド・ライン、コード・ラインにおいて、ハイフンは、要求に対する引数がその後の行に続くことを示します。
数字	特に明記しない限り、本文中の数字はすべて 10 進数です。10 進数以外 (2 進数、8 進数、16 進数) は、その旨を明記してあります。



# Part 1

---

## セキュリティの概要

このパートの各章では、以下のトピックについて説明します。

- セキュリティ障害の原因 (1.1 節)
- セキュリティ要件のレベル (1.2 節)
- セキュリティ設計におけるリファレンス・モニタの概念 (2.1 節)
- オペレーティング・システムのセキュリティ機能 (2.2 節)



## システム・セキュリティ

オペレーティング・システムのセキュリティ対策が効果的であれば、不正アクセス、コンピュータ時間の盗用、および各種の機密情報（マーケティング計画、製法、自社開発ソフトウェアなど）の盗難を防ぐのに役立ちます。これらの対策によって、装置、ソフトウェア、およびファイルを改ざんによる損害から守ることもできます。

この章では、オペレーティング・システムに用意されているセキュリティ対策の概要をセキュリティ管理者に示します。Part 3 では、サイトのセキュリティ・ポリシー、セキュリティ管理者の作業、およびサイトのコンピュータ・システムや資源を保護する方法について、さらに詳しい情報を示します。

### 1.1 コンピュータ・セキュリティ問題のタイプ

どのようなシステムにも、権限を持つユーザと権限のないユーザの、2 種類のユーザが存在します。コンピュータ・システムの使用を許可された人は、サイトのセキュリティ管理者が設定した登録基準に従ってシステムとシステム内の資源を利用する権限を持っています。使用基準には、使用できる時間帯、ログインのタイプ、使用できる資源の種類（プリンタやターミナルなど）、その他が含まれます。権限のないユーザは、システムを使用する権限をまったく持たないか、指定された時間帯しか使用できないか、または特定のシステム資源を使用する権限を持っていません。

通常、コンピュータ・システムのセキュリティ侵害は、以下の 4 種類の行為の結果として生まれます。

- ユーザの無責任行為は、ユーザが故意または過失によって著しい損害を与えることを指します。たとえば、一定のファイルの利用を許可されたユーザが重要なファイルのコピーを作成して売のような場合が、これに該当します。

このようなセキュリティ障害の原因からサイトを保護するためにオペレーティング・システムにできることは、ほとんどありません。この問題は、多くの場合、アプリケーション設計の不備や、ユーザとセキュリティ管理者が行う制御の方法に一貫性がないことに原因があります。環境面のセキュリティを十分に実施していないために、いつの間にかこのタイプのセキュリティ問題を助長している場合もあります。

たとえ最高のセキュリティ・システムであっても、実装方法に一貫性がなければ失敗します。このことに加えて、適切なセキュリティ手順に従うようにユーザを仕向けなければ、システムはユーザの無責任行為によって起きるセキュリ

ティ障害に対して脆弱になります。第 3 章では、システム・セキュリティを維持する上でユーザができることについて説明します。

- ユーザの詮索行為は、ユーザがシステムの十分に保護されていない部分を不正に利用することを指します。ユーザの中には、システムを相手にゲームをする感覚で、禁じられたシステム領域へのアクセスを獲得しようとする人がいます。たとえ本人に悪意がなくても、サービスの盗用は犯罪です。もっと重大な悪意を持ったユーザが、詮索行為によって極秘情報を探したり、横領を企てたり、データを破壊したりすることもあります。ユーザの詮索行為は、常に真剣に取り扱う必要があります。

システムには、ユーザの詮索行為に対抗するためにさまざまなセキュリティ機能が用意されています。セキュリティ管理者は、セキュリティ条件に基づいて機能を一時的または永続的に実装します。保護コードとアクセス制御リストによるデータや資源の保護については、第 4 章を参照してください。

- ユーザの侵入行為は、ユーザがセキュリティ制御を突破してコンピュータ・システムへのアクセスを得ることを指します。システムには侵入行為をきわめて難しくするセキュリティ機能がありますが、どんなオペレーティング・システムでも侵入行為を完全に阻止することは不可能です。

システムへの侵入に成功するユーザは、スキルと悪意の両方を持っています。このため、侵入行為は最も深刻で大きな危険を伴うセキュリティ侵害です。ただし、OpenVMS のセキュリティ機能を正しく実装すれば、侵入行為は並外れたスキルと忍耐を必要とする最も起こりにくいセキュリティ侵害になります。

- ソーシャル・エンジニアリングは、技術的な手段ではなく、ユーザ、オペレータ、または管理者をだますことで侵入者がシステムへのアクセスを得ることを指します。侵入者は、たとえば電話を使って権限を持つユーザになります。侵入者は、電話番号やパスワードなど、システムへのアクセスを得るための情報を引き出そうとしたり、何も知らないオペレータに対してシステムのセキュリティを脅かす行為を実行するように要求したりします。

近年になってオペレーティング・システムの技術的なセキュリティ機能が強化されるのに伴って、ソーシャル・エンジニアリングを要因とするセキュリティ事件の割合が増えています。権限のない人に不用意にアクセスを許可しないようにするには、オペレータのトレーニング、管理手順、およびユーザの意識向上がいずれも重要な要因になります。

これらの問題を回避する方法について、以下の章で説明します。

- 第 7 章 では、侵入検出システムとそのパラメータの設定方法について説明します。
- 第 8 章 では、システムのファイルと資源の保護を強化する方法について説明します。

- 第9章では、システムの動作を監視して、悪意のある行為に関する通知を受ける方法について説明します。
- 第10章では、システムへの侵入行為に対する対処方法を示します。
- 第3章および第6章では、サイトのトレーニング・プログラムに取り入れるべきトピックを示します。

## 1.2 セキュリティ要件のレベル

各サイトには、それぞれ固有のセキュリティ要件があります。中には、ほとんど悪影響を及ぼさない形態の不正アクセスが許容されるために、ごく限られた対策しか必要としないサイトもあります。その対極にあるのは、戦略的軍事防衛センターのように、ほんの些細な詮索行為でも見逃すことができないサイトです。銀行など、商用サイトの多くは、これらの中間になります。

セキュリティ条件を決定するにはさまざまな考慮事項がありますが、表 1-1 の質問はその出発点になります。これらの質問に対する回答が、セキュリティ条件のレベルを決定するのに役立ちます。サイトのセキュリティ要件のより具体的な例については、6.2 節も参照してください。

表 1-1: セキュリティ要件の基準となるイベント許容度

質問：以下のイベントを許容できますか。	許容度の回答に基づくセキュリティ要件のレベル		
	低	中	高
ユーザがシステム上で実行されるイメージを知っている	Y	Y	N
ユーザが他のユーザのファイル名を知っている	Y	Y	N
ユーザがグループ内の別のユーザのファイルにアクセスする	Y	Y	N
部外者がダイアルアップしたばかりのシステムの名前を知っている	Y	Y	N
ユーザが他のユーザのファイルをコピーする	Y	N	N
ユーザが他のユーザの電子メールを読む	Y	N	N
ユーザが他のユーザのファイルにデータを書き込む	Y	N	N
ユーザが他のユーザのファイルを削除する	Y	N	N

表 1-1: セキュリティ要件の基準となるイベント許容度 (続き)

ユーザが、各種の古いファイルが格納されているかもしれないディスクのセクションを読み込める	Y	N	N
ユーザが (ゲームをする場合を含めて) 関係ない作業や許可されていない作業を実行するためにマシン時間と資源を消費する	Y	N	N

表に挙げた大部分のイベントを許容できる場合は、セキュリティ要件がかなり低いと言えます。許容できるものとできないものが混在する場合は、セキュリティ要件が中～高の範囲にあります。表に挙げた大部分のイベントを許容できないサイトのセキュリティ要件は、一般に非常に高いレベルにあります。

サイトのセキュリティ条件を調べるときは、サイトの運用や回復手順における弱点をセキュリティ問題と混同しないでください。システムのセキュリティ要件を評価する前に、運用ポリシーが有効で一貫していることを確認してください。

### 1.3 安全なシステム環境の構築

オペレーティング・システムの外部に存在するセキュリティ問題の原因として、従業員の不注意と施設の防備の脆弱性の 2 つがあります。従業員に不注意や悪意があったり、施設が無防備だったりすれば、このガイドで説明するどのセキュリティ手段を使ってもセキュリティ侵害からシステムを保護することはできません。

システム侵入のほとんどは、このような環境上の弱点を突いて行われます。アクセス保護コードを突破したり、ファイルの保護を変更したりするより、小さなリールから物理的にテープを取り外す方がはるかに簡単です。

サイトのセキュリティを見直すときは、オペレーティング・システムの保護だけでなく、環境上の問題にも重点を置くことを強くおすすめします。

本書では、オペレーティング・システムのセキュリティ手段について説明します。実装する手段を決定するときは、サイトのセキュリティ条件を現実的に把握することが重要です。サイトに十分なセキュリティを導入することは大切ですが、実際のニーズを超えたセキュリティを導入するのはコストと時間の無駄です。

システムに適用するセキュリティ手段を決定するときは、以下の点に留意してください。

- システムのセキュリティを高めるほど、システムの使い勝手は悪くなります。
- セキュリティを高めると、データのアクセス速度、マシン操作の速度、およびシステム性能がいずれも低下するため、コストが増加します。
- セキュリティ手段の数を増やすほど、必要な要員の工数も増えます。



オペレーティング・システムには、システムやデータへのアクセスを制御する基本的なメカニズムが備わっています。また、権限を持つユーザにのみアクセスが限定されていることを確認するための監視ツールも用意されています。しかし、コンピュータ犯罪の多くは、権限を持つユーザがオペレーティング・システムのセキュリティ制御を侵害せずに起こしています。

したがって、業務のセキュリティは、これらのセキュリティ機能をどのように適用し、従業員とサイトをどのように管理するかにかかっています。あらかじめアプリケーションに適切な監視制御機能を組み込み、不正使用の可能性を最小限に抑えるという目標に合わせてアプリケーションを設計することで、オペレーティング・システムとサイトのセキュリティ機能を実装し、弱点の少ない環境を構築することができます。組織のセキュリティ計画の例については、第 6 章を参照してください。

米国政府によるオペレーティング・システムのセキュリティ認定レベル C2 に適合するシステムが必要な場合は、付録 C を参照してください。

安全な OpenVMS システムのために高いレベルのコンピュータ・セキュリティが必要な場合は、SEVMS をおすすめします。SEVMS は OpenVMS のセキュリティ強化バージョンで、強制アクセス制御によってシステム全体にセキュリティ・ポリシーを適用できます。

SEVMS は、米国国防省が認定する B1 レベルのセキュリティに対応するオペレーティング・システムです。

## 1.4 CDSA (Common Data Security Architecture)

CDSA (Common Data Security Architecture) は、業界標準のマルチプラットフォーム・セキュリティ・インフラストラクチャです。

バージョン 7.3-1 以降の OpenVMS Alpha では、CDSA がオペレーティング・システムの一部として提供されています。CDSA は、OpenVMS Alpha バージョン 7.2-2 以降と互換性があります。

CDSA は、アプリケーションからオペレーティング・システムのセキュリティ・サービスへのアクセスを実現する、安定した標準ベースのプログラミング・インタフェースです。CDSA を使用することにより、クロスプラットフォームのセキュリティ対応アプリケーションを作成できます。セキュリティ・サービス (暗号やその他の公開鍵操作など) を、一連のプラグイン・モジュールに対する動的に拡張可能なインタフェースを介して使用します。これらのモジュールは、業務の条件や技術の進歩に応じて追加または変更できます。

CDSA は、さまざまなアプリケーションとセキュリティ・サービスに対応する柔軟な統合ソリューションを提供するセキュリティ・ミドルウェアです。CDSA によって、アプリケーションにセキュリティをどう組み込むかという問題から解放さ

れ、アプリケーションそのものに集中することができます。ユーザは、土台となるセキュリティを気にせずに済みます。

CDSA は、元々インテル・アーキテクチャ研究所で開発され、2000 年 5 月にオープンソース・コミュニティにリリースされました。HP の CDSA は、Intel V2.0 Release 3 リファレンス・プラットフォームを基に、The Open Group の Technical Standard C914 (2000 年 5 月) で定義された CDSA V2.0 (with corrigenda) を実装したものです。

CDSA の詳細については、『*HP Open Source Security for OpenVMS, Volume 1: Common Data Security Architecture (CDSA)*』を参照してください。

## 1.5 SSL (Secure Sockets Layer)

SSL (Secure Sockets Layer) は、機密情報をインターネット経由で安全に転送するための公開標準セキュリティ・プロトコルです。SSL では、暗号化によるプライバシー、サーバ認証、メッセージの完全性の 3 つが提供されます。クライアント認証は、オプション機能として用意されています。

OpenVMS Alpha バージョン 7.3-1 以降では、SSL がオペレーティング・システムの一部として提供されています。HP の SSL は、OpenVMS Alpha バージョン 7.2-2 以降および OpenVMS VAX バージョン 7.3 以降と互換性があります。

OpenVMS アプリケーションとの TCP/IP 接続による通信リンクの保護は、SSL を使用することで実現できます。OpenSSL の API は、非公開の認証済みで信頼性の高いアプリケーション間通信リンクを確立します。

SSL プロトコルは、他の複数のプロトコル上で協調して動作します。SSL はアプリケーション・レベルで動作し、その下位のメカニズムとして TCP/IP (Transmission Control Protocol/Internet Protocol) がインターネット経由のデータの転送と経路制御を担います。HTTP (HyperText Transport Protocol)、LDAP (Lightweight Directory Access Protocol)、IMAP (Internet Messaging Access Protocol) などのアプリケーション・プロトコルは TCP/IP 上で動作します。これらは、TCP/IP を使用して一般的なアプリケーション・タスク (Web ページの表示や電子メール・サーバの実行など) に対応します。

SSL は、インターネットなどの TCP/IP ネットワーク経由の通信に関する以下の 3 つの基本的なセキュリティ問題に対応します。

- SSL サーバ認証により、ユーザがサーバの身元を確認できます。SSL 対応クライアントは、公開鍵暗号の標準的な手法を使用して、サーバの証明書と公開 ID が有効かどうか、およびクライアントが持っている信頼される認証局 (CA) のリストに記載された CA がそれらを発行しているかどうかを確認します。たとえば、PC ユーザが Web 上でのショッピングのためにクレジッ

ト・カード番号を送信しようとして、送信先サーバの身元確認をするときなどにサーバ認証が使用されます。

- SSL クライアント認証により、サーバがユーザの身元を確認できます。SSL 対応サーバ・ソフトウェアは、サーバ認証と同じ手法を使用して、クライアントの証明書と公開 ID が有効かどうか、およびサーバが持っている信頼される認証局 (CA) のリストに記載された CA がそれらを発行しているかどうかを確認します。たとえば、銀行が機密の金融情報を顧客に送信しようとして、送信先の身元確認をするときなどにクライアント認証が使用されます。
- 暗号化された SSL 接続では、クライアントとサーバの間で送信されるすべての情報を送信側のソフトウェアで暗号化し、それを受信側のソフトウェアで復号化する必要があります、これによって高度な機密保護機能が提供されます。非公開のトランザクションでは、機密保護がどちら側の当事者にとっても重要です。また、暗号化された SSL 接続を介して送信されるすべてのデータは、送信中にデータが改変されたかどうかを自動的に検出するメカニズムによって保護されます。

SSL の詳細については、『*HP Open Source Security for OpenVMS, Volume 2: HP SSL for OpenVMS*』または以下の HP SSL に関する Web サイトを参照してください。

<http://h71000.www7.hp.com/openvms/products/ssl/>

## 1.6 Kerberos

Kerberos は、秘密鍵暗号を使用してクライアント/サーバ型アプリケーションに強力な認証機能を提供するネットワーク認証プロトコルです。1980 年代中頃に Athena プロジェクトの一部として MIT (Massachusetts Institute of Technology) で開発されました。Athena プロジェクトの使命は、コンピュータの多様な利用方法を研究し、コンピュータを MIT のカリキュラムに組み入れる方法に関する長期的な戦略意思決定を行うために必要な知識ベースを構築することでした。

バージョン 7.3-1 以降の OpenVMS Alpha では、Kerberos がオペレーティング・システムの一部として提供されています。Kerberos は、OpenVMS Alpha バージョン 7.2-2 以降および OpenVMS VAX バージョン 7.3 以降と互換性があります。

Kerberos V4 までは、この技術を一般に入手できませんでした。それまでのバージョンは、Athena プロジェクトの内部だけで使用されていました。最新バージョンである Kerberos V5 が、商用に対応する初めてのリリースです。

Kerberos プロトコルには強力な暗号技術が使用されており、安全でないネットワーク接続を介してクライアントがサーバの（およびサーバがクライアントの）身元を証明できるようになっています。Kerberos を使用して互いの身元を証明した後、

クライアントとサーバはすべての通信内容を暗号化して、プライバシーとデータの一貫性を保証することができます。

Kerberos の詳細については、『*HP Open Source Security for OpenVMS, Volume 3: Kerberos*』または以下の OpenVMS 用 Kerberos の Web サイトを参照してください。

**<http://h71000.www7.hp.com/openvms/products/kerberos/>**

---

## OpenVMS のセキュリティ・モデル

この章では、オペレーティング・システムに組み込まれたセキュリティの機能やメカニズムを設計および実装する際の指針となった概念を示します。ここでの目的は、システム・セキュリティの全体像を考える際の枠組みを提供することです。後の各章では、セキュリティ機能とその使用方法について詳しく説明します。

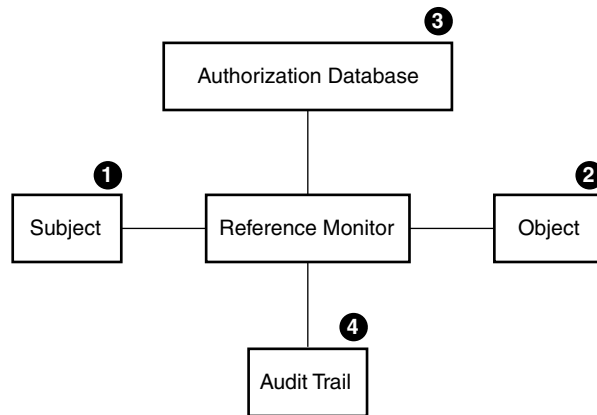
### 2.1 安全なオペレーティング・システムの構造

1960 年代後半、マルチユーザのコンピュータ・システムでどのようにセキュリティを実現するかという問題に対する研究開発が数多く行われました。開発作業の多くは、システムのセキュリティを損なうおそれのある要素をすべて洗い出し、それらの不具合を 1 つ 1 つ修正していくことに充てられました。やがて研究者は、このプロセスが無駄であり、有効なシステム・セキュリティは安全なコンピュータ・システムの構造に関する基本モデルからしか生まれないことに気づきました。リファレンス・モニタの概念は、このようなモデルとして提案され、広く受け入れられました。

#### 2.1.1 リファレンス・モニタの概念

リファレンス・モニタの概念によれば、図 2-1 のように、コンピュータ・システムはサブジェクト、オブジェクト、登録データベース、監査証跡、およびリファレンス・モニタによって表現されます。リファレンス・モニタは、サブジェクトを認証し、サブジェクトによるオブジェクトへのあらゆるアクセスに関してセキュリティ・ポリシーを実装および実施する管理センターです。

図 2-1: リファレンス・モニタ



VM-0994A-AI

次の表は、図 2-1 に示した各要素の説明です。

項番	要素	説明
1	サブジェクト	人間のために情報にアクセスする能動的なエンティティ（ユーザ・プロセスなど）。
2	オブジェクト	保護すべき情報の受動的な格納場所（ファイルなど）。
3	登録データベース	サブジェクトとオブジェクトのセキュリティ属性の格納場所。リファレンス・モニタは、これらの属性に基づいて、許可されたアクセスの種類を（もしあれば）特定します。
4	監査証跡	すべてのセキュリティ関連イベント（成功または失敗したアクセス操作など）のレコード。

### 2.1.2 リファレンス・モニタによるセキュリティ規則の実施

リファレンス・モニタは、サブジェクトの作成を許可し、サブジェクトによるオブジェクトへのアクセスを認め、必要に応じて監査証跡にイベントを記録することによって、セキュリティ・ポリシーを実施します。理想のシステムでは、リファレンス・モニタが以下の 3 つの要件を満たす必要があります。

- サブジェクトがオブジェクトへのアクセスを得ようとする操作をすべて仲介する。
- 許可されない参照や変更から完全に保護された、不正操作に強いデータベースと監査証跡を提供する。
- セキュリティ要件を効果的に適用できるように、ソフトウェアを小規模で単純な合理化されたものにする。

上記は、侵入行為があっても安全が確保できるシステムに関して提案される要件です。このようなシステムでは、オペレーティング・システムのセキュリティ関連サブセット (セキュリティ・カーネル) によってリファレンス・モニタが実装されます。

## 2.2 リファレンス・モニタの実装

OpenVMS オペレーティング・システムでは、リファレンス・モニタがセキュリティ関連サブセット (セキュリティ・カーネル) として実装されませんが、リファレンス・モニタの概念で要求されている基本構造は、ユーザやシステム管理者に対するインタフェースに反映されています。これまでの経験から、詮索行為やほとんどの侵入行為に対抗できるシステムを構築するには、このような構造を組み込むことが最善の方法であることがわかっています。

以下の各セクションでは、OpenVMS オペレーティング・システムにおけるリファレンス・モニタ・モデルの実装について説明します。

### 2.2.1 サブジェクト

サブジェクトは、情報にアクセスする (場合によっては情報へのアクセスを禁止される) ユーザまたはユーザ・エージェント (ユーザ・プロセス) です。サブジェクトは会話形式のプロセス、ネットワーク・プロセス、またはバッチ・ジョブであり、次の特徴を持っています。

- 以下の時点でセキュリティ制御を通過する必要があります。

プロセスの作成時  
情報へのアクセス時

- 以下の識別が必要です。

ユーザ名  
パスワード  
ユーザ識別コード  
ライト識別子

ユーザがオペレーティング・システムを会話形式で使用するためのログインした時点、またはバッチ・ジョブやネットワーク・ジョブが開始した時点で、オペレーティング・システムはそのユーザの識別情報を含むプロセスを作成します。作成されたプロセスは、第 4 章で説明するように、ユーザのエージェントとして情報にアクセスします。

作成中のプロセスや情報にアクセスしているプロセスは、セキュリティ侵害に対して脆弱です。システムは、登録データと内部のメカニズム (ハードウェア制御など) を使用して、プロセスによる情報へのアクセスを処理します。プロセスの作成にはさまざまな分野でセキュリティの脆弱性があるため、オペレーティング・

システムのセキュリティ機能の多くはプロセス(またはサブジェクト)作成の分野に集中しています。

ユーザは、システムにログインしようとするときに、ユーザ名(作成されるプロセスに与えられる名前)とパスワードを入力します。パスワードは、ユーザとオペレーティング・システムだけが知っている認証情報としての役割を果たします。

短いパスワードや自明のパスワードではこの要件を満たせない可能性があるため、システムにはさまざまなパスワード保護メカニズムが組み込まれており、それらをユーザが呼び出したり、セキュリティ管理者が要求したりできるようになっています(第7章を参照)。オペレーティング・システムには、侵入者がパスワードを推測するために行う操作の回数を制限する機能もあります。

ユーザ・パスワードのファイルは、セキュリティ・データベースの一部であり、許可されない参照や変更から保護する必要があります。この要件を満たすために、システムでは一般のアクセスから保護されたファイルにパスワードが保存されます。このファイルをシステム・ユーザ登録ファイル(SYSUAF.DAT)と呼びます。また、追加的な予防措置として、パスワードが盗まれても簡単には使用できないように、エンコードされた形式でパスワードが保存されます。

オペレーティング・システムは、ユーザのプロセスを作成すると、ユーザ登録レコードに基づいてユーザ識別コード(UIC)をそのプロセスに割り当てます。UICは、プロセスを作成したユーザの名前(ユーザのパスワードによって認証されたもの)に対応します。また、UICはユーザが自分の部署、プロジェクト、または職務に対応するグループのメンバであることも示します。プロセスの作成やプロセス所有者の他のグループとの関係に関する追加情報をプロセスに関連付けることもできます。これらの追加情報は、登録データベースを適用するときに一定の役割を果たします。UICについては、第4章と第8章で説明します。

### 2.2.2 オブジェクト

リファレンス・モニタの概念では、オブジェクトは情報の受動的な格納場所です。表2-1に示すように、OpenVMSにはファイルやデバイスなどのさまざまなオブジェクトがあり、保護の対象になっています。オペレーティング・システムは、不正なアクセスからオブジェクトを保護し、(第4章および第5章で説明するように)それらを制御された方法で共用するための各種のメカニズムを提供します。これらのメカニズムは、システム資源へのアクセスを制御するときにも使用されます。



表 2-1: セキュリティ制御によって保護されるオブジェクト

クラス名	定義
ケーパビリティ	システムによってアクセスが制御される資源。現時点で定義されているケーパビリティは、ベクタ・プロセッサだけです。
コモン・イベント・フラグ・クラスタ	連携するプロセス同士がイベント通知を相互に提示できるようにするために、32 個のイベント・フラグをセットにしたもの。
デバイス	プロセッサに接続された周辺機器のクラスの 1 つで、データを受信、保存、または伝送する機能をもつもの。
ファイル	Files-11 オン・ディスク構造レベル 2 または 5 のファイルおよびディレクトリ。
グループ・グローバル・セクション	同じグループ内のすべてのプロセスが使用できる共用可能なメモリ・セクション。
論理名テーブル	システムまたは特定のグループに関する論理名とその等価名を格納した共用可能なテーブル。
キュー	バッチ、ターミナル、サーバ、またはプリント・ジョブ・キューで処理される一連のジョブ。
資源ドメイン	ロック・マネージャの資源へのアクセスを制御するネームスペース。
セキュリティ・クラス	セキュリティ・クラスのすべてのメンバに関する要素と管理ルーチンを格納するデータ構造。
システム・グローバル・セクション	システム内のすべてのプロセスが使用できる共用可能なメモリ・セクション。
ボリューム	ディスクやテープなどの、ODS-2 形式または ODS-5 形式の大容量ストレージ媒体。ボリュームは、ファイルを格納するもので、デバイスにマウントすることができます。

### 2.2.3 登録データベース

リファレンス・モニタの概念では、各サブジェクトがオブジェクトへのアクセスを得るための認証は、抽象的な登録データベースに基づいて行われます。このデータベースは、サブジェクトによるオブジェクトへのアクセスを常に統御する動的なセキュリティ属性を集めたものです。OpenVMS システムでは、このデータベースは保護するオブジェクトとの関連に基づいて分散して保存されます。たとえば、ファイルやディレクトリの登録データはそのファイルまたはディレクトリのヘッダに保存されます。次の表は、登録データベースに保存される情報についてまとめたものです。

ファイル	内容	解釈に使用されるデータ
#SYSUAF.DAT	ユーザ名	ログイン
	パスワード	ログイン
	UIC	アクセス制御のチェック
#NETPROXY.DAT	ユーザ名	ログイン
#NET\$PROXY.DAT	ユーザ名	ログイン
#RIGHTSLIST.DAT	ライト識別子	アクセス制御のチェック
#VMS\$OBJECTS.DAT	UIC	アクセス制御のチェック
	保護コード	アクセス制御のチェック
	アクセス制御リスト	アクセス制御のチェック
#VMS\$AUDIT_ #SERVER.DAT	監査可能イベント	イベントの報告

2.2.2 項からわかるように、OpenVMS システムの各オブジェクトには、共用時の柔軟性にいくつかのレベルがあります。保護オブジェクトは、保護コードに従っています。このコードは、システム・ユーザ、オブジェクトの所有者であるユーザ、所有者が属する UIC グループの他のメンバ、およびその他すべてのユーザのために実行されるプロセスに対して、アクセスを許可（または拒否）するかどうかを指定します。

保護コードに加えて、アクセス制御リスト (ACL) の制御に基づいてオブジェクトを共用することもできます。ACL は、特にユーザ・グループやそのサブセットに対して、UIC に基づく保護よりも細かいアクセス制御を提供します。ACL には、オブジェクトに対する特定のタイプのアクセスを許可または拒否するユーザまたはユーザ・グループが記述されます。ACL では、UIC の識別情報だけでなく、プロセスに関連付けることができる他のグループ分類や識別子に基づいて共用を指定できます。たとえば、ダイアルアップ回線でターミナルに接続されたプロセスによってファイルが読み込まれないように指定することができます。2.2.6 項では、アクセス・マトリックスを使用して ACL の概念を説明します。4.4 節では ACL と識別子に関する一般的な説明を示し、第 8 章ではセキュリティ管理者が識別子を作成してシステム資源の ACL を構築する方法について説明します。

## 2.2.4 監査証跡

すべてのセキュリティ関連イベントは、監査ログ・ファイルに記録されるか、オペレータ・ターミナルに送信されるか、またはその両方が行われます。ターミナルをセキュリティ・オペレータ・ターミナルに指定すると、すべての監査可能イベントがそのターミナルに表示されます。監査ログ・ファイルには、セキュリティ・イベントが永続的に記録されます。システム管理者は、ログ・ファイルを調べることで処理のパターンを見つけることができます。このパターンを監査証跡と呼びます。

オペレーティング・システムは、表 2-2 に示すセキュリティ・イベントのクラスをデフォルトで監査します。他のイベント（ボリュームのマウントやシステム時刻の変更など）を監査対象として選択することもできます。

表 2-2: セキュリティ監査機能の概要

出力先	デフォルトで監査されるイベント
ログ・ファイルまたはターミナルのディスプレイ	登録データベースの変更  侵入行為  ログインの失敗  DCL の SET AUDIT コマンドの使用  監査用 ACE またはアラーム用 ACE によって起動するイベント

ユーザとセキュリティ管理者は、監査ログにさまざまなイベントを記録できます。すべてのイベントを調べるのは時間がかかり過ぎるので、セキュリティ状況の把握に役立つ情報を豊富に含むイベントだけを監査するのが最も効率的です。セキュリティ監査機能の詳細については、第 9 章を参照してください。

### 2.2.5 リファレンス・モニタ

OpenVMS オペレーティング・システムでは、エグゼクティブがリファレンス・モニタの役割を実行します。カーネル・モードとエグゼクティブ・モードで実行されるすべてのシステム・プログラムが、コマンド行インタプリタや特権で実行される特定のユーザ・モード・イメージとともに、リファレンス・モニタの実装に一役買っています。エグゼクティブを構成するコードの量は膨大ですが、それらのコードがシステム・セキュリティの適用を回避する手段にならないようにするための努力が払われています。

特権の中には、リファレンス・モニタを変更または無効化する権限をユーザに与えるものがあります。たとえば、CMKRNL 特権を持つプロセスは、自身のコードをシステム・カーネル内で実行することにより、リファレンス・モニタの内部データや保護オブジェクトの内部表現にアクセスできます。当然ながら、このような重要な特権は厳しく制限する必要があります。

同じように、SYSPRV や SECURITY などの特権は、リファレンス・モニタや登録データベースの維持に役立つプロセスのユーザのみに付与します。

### 2.2.6 アクセス・マトリックスで表した登録データベース

リファレンス・モニタのモデルには、登録データベースが規定されています。登録データベースには、すべてのサブジェクトとすべてのオブジェクトに関するシ

システム内のすべてのアクセス登録情報が記述されます。このデータベースは、多くの場合、一方の軸にサブジェクトを並べ、もう一方の軸にオブジェクトを並べたアクセス・マトリックスで表現されます (図 2-2 を参照)。マトリックス内の交点は、それぞれ、あるサブジェクトがあるオブジェクトに関して許可されているアクセスを表します。

図 2-2: 登録アクセス・マトリックス

Objects:	V	W	X	Y	Z
Subjects:					
A		*			*
B		*	*	*	
C		*	*	*	
D	*	*	*	*	
E	*				

VM-0995A-AI

このアクセス・マトリックスでは、該当するサブジェクトが該当するオブジェクトへのアクセスを許可されている場合にアスタリスク (\*) が付いています。説明を簡単にするため、この例ではアクセスのタイプ (読み込みや書き込みなど) は省略しています。たとえば、サブジェクト B, C, および D は、すべてオブジェクト W, X, および Y へのアクセスを許可されています。さらに、サブジェクト A はオブジェクト W および Z へのアクセスを、サブジェクト D はオブジェクト V へのアクセスを、サブジェクト E はオブジェクト V へのアクセスを、それぞれ許可されています。

アクセス・マトリックスを行単位で分割すると、ケーパビリティ・ベースのモデルになります。ケーパビリティ・ベースのモデルでは、アクセスできるオブジェクトのリストがサブジェクトごとに保持されます。たとえば、このアクセス・マトリックスをケーパビリティに基づいて表現すると、次のようになります。

A: W, Z B: W, X, Y C: W, X, Y D: V, W, X, Y E: V

一方、アクセス・マトリックスを列単位で分割して、アクセスが許可されているサブジェクトをオブジェクトごとに列挙することもできます。この場合は、権限ベースのモデルになり、OpenVMS では ACL によって実装されています (第 4 章を参照)。ACL での表現は、次のようになります。

V: D, E W: A, B, C, D X: B, C, D Y: B, C, D Z: A

オペレーティング・システムで使用される ACL と識別子による制御は、ケーパビリティ・ベースのシステムと権限ベースのシステムの両方の性質を兼ね備えています。OpenVMS システムでは、サブジェクトとオブジェクトの両方が識別子を保持します。サブジェクトは、一致する識別子をオブジェクトが持っている場合と、

要求したアクセスがオブジェクトのアクセス・ステートメントによって許可される場合に、そのオブジェクトにアクセスできます。

ケーパビリティ・ベースのシステムと権限ベースのシステムの両方の性質を兼ね備えた結果、複雑なアクセス・マトリックスを簡潔かつ手頃な方法で表現できる、きわめて強力で柔軟性に富んだシステムになっています。上記のアクセス・マトリックスの例について、図 2-3 のように一部の交点にラベルを付けた場合を考えてみましょう。

図 2-3: 交点にラベルを付けた登録アクセス・マトリックス

Objects:	V	W	X	Y	Z
Subjects:					
A		*			*
B		Q	Q	Q	
C		Q	Q	Q	
D	P	Q	Q	Q	
E	P				

VM-0996A-AI

同じラベルを付けた複数の交点は、1 つのエンティティとしてグループ化して扱うことができます。たとえば、図 2-3 で Q というラベルの付いた交点は、サブジェクト B、C、および D がオブジェクト W、X、および Y に関して許可されているアクセスを表します。Q という交点は、すべて 1 つの関心領域として考えることができます。識別子の概念は、このような関心領域のグループ化の利点を実用化するために提供されています。

図 2-3 では、P と Q という 2 つのアクセス・グループを表す識別子をそれぞれ定義できます。マトリックスにはラベルの付いていない交点が 2 つ残っていることに注意してください。識別子は個々のサブジェクトを表すこともできるので、従来の ACL の機能も使用できます。

OpenVMS オペレーティング・システムでは、アクセス・マトリックスの 2 つの次元を表すために以下の 2 つの構造を使用します。

- ライト・リスト (RIGHTSLIST.DAT) は、アクセス・マトリックスの行を表し、ケーパビリティ・ベースのモデルに対応します。図 2-3 のマトリックスの場合は、次のライト・リストが必要になります。

B: Q C: Q D: P, Q E: P

- 保護オブジェクトの ACL は、アクセス・マトリックスの列を表します。上記の例では、次の ACL が必要になります。

V: P W: A, Q X: Q Y: Q Z: A

アクセス・マトリックスを表すのに必要なシステム構造は、従来のケーパビリティ・ベースのモデルや権限ベースのモデルより簡単で、全体としてはより少ない字数で表すことができます。この例ではわずかな違いしかありませんが、アクセス・マトリックスの複雑さは規模の 2 乗に比例して増大します。

## 2.3 要約：システム・セキュリティ設計

システム全体のセキュリティ計画を設計するときは、以下の質問に回答してください。

- ユーザはサブジェクトとどのように関連付けられていますか。認証メカニズムにはどの程度の信頼性がありますか。
- このシステムまたはアプリケーションでは、どのオブジェクトに機密情報が含まれていますか。それらのオブジェクトに対するアクセスは制御されていますか。
- 登録データベースにはサイトのセキュリティ・ポリシーが反映されていますか。機密オブジェクトへのアクセスは誰に許可されていますか。制限が十分に行われていますか。
- 監査証跡に記録される情報は十分ですか。または、多過ぎませんか。監査証跡は誰が監視しますか。監査証跡をどの程度の頻度でチェックしますか。
- リファレンス・モニタの構成要素として機能するのはどのプログラムですか。セキュリティ・ポリシーと登録データベースを変更できるのはどのユーザですか。それは、望ましい構成ですか。

これらの考慮事項は、土台となるリファレンス・モニタの設計と同じように、ファイルやデータベースのレコードへのアクセスを許可するシステム上のタイムシェアリング・システム、広範囲のネットワーク、または個々のアプリケーションに等しく適用されます。オペレーティング・システムは、ユーザとセキュリティ管理者がシステム・セキュリティを実現するために適用すべき一般的なメカニズムを提供します。セキュリティ・ポリシーの設計と実装の詳細については、第 6 章を参照してください。

# Part 2

---

## 一般ユーザのためのセキュリティ

このパートの各章では、以下のトピックについて説明します。

- パスワードの使用 (第 3 章)
- ログインとログアウトのプロセス (第 3 章)
- サブジェクトとオブジェクトのセキュリティ・プロファイル (第 4 章)
- オブジェクト保護メカニズム (第 4 章)
- オブジェクト・クラスの特長 (第 5 章)





## システムの安全な使用

この章では、システムを安全に使用方法について基本的な情報を示します。サイトの個別のセキュリティ・ポリシーを守りながら、これらの知識を一貫して正しく利用すれば、安全なシステムと権限のないユーザから攻撃を受けやすいシステムとの間に一線を引くことができます。

### 3.1 アカウントのパスワードの選択

安全なパスワードを選択するには、以下のガイドラインに従います。

- パスワードを数字と文字で構成します。たとえば、同じ6文字のパスワードでも、文字のみで構成されたパスワードよりも、文字と数字の両方で構成されたパスワードの方がはるかに安全です。
- パスワードを6～10文字で構成します。パスワードの長さを十分に確保すると、より安全です。パスワードの長さは、最大32文字までです。
- 辞書や自国語にある単語をパスワードに使用しないでください。
- 自分のコンピュータ・サイトや自分自身に関連する単語（製品名や自家用車の車種名など）の使用は避けます。
- 毎回新しいパスワードを選択します。前に使用したパスワードは再利用しないでください。

セキュリティ管理者が追加の制限事項（たとえば、10文字未満のパスワードは許可しないなど）を設定する場合があります。

表 3-1 に、安全なパスワードと危険なパスワードの例を示します。

表 3-1: 安全なパスワードと安全でないパスワード

安全なパスワード	安全でないパスワード
意味のない文字の並び (aladaskgam , eojfuvcue , joxyois など)	個人との強い関連性がある単語 (自分の名前、好きな人の名前、ペットの名前、住んでいる町の名前、自家用車の名前など)
英数字や記号が混在する文字列 (492_weid , \$924spa , zu_\$rags など)	仕事に関連する用語 (自分の会社、特別プロジェクト、作業グループの名前など)

### 3.1.1 初期パスワードの取得

通常、ユーザはシステムに自分のアカウントが作成されたことを知られるとき、ユーザ・パスワードが必要かどうかとも知らされます。ユーザ・パスワードが有効になっている場合、最初のログイン時に指定のパスワードを入力するように指示されます。このパスワードは、自分のアカウントの使用法に関する他の情報とともに、システム・ユーザ登録ファイル (SYSUAF.DAT) に格納されています。

簡単に推測できるパスワードを持つことは、お勧めできません。アカウントの作成担当者と相談して、推測しにくいパスワードを指定してください。与えられるパスワードの決定にまったく関与できない場合は、自分の名前がそのままパスワードになっている場合もあります。そのような場合は、ログイン後、直ちにパスワードを変更してください。自分の名前をパスワードに使用するのには、よくある方法であり、セキュリティの観点からは望ましくありません。

アカウントが作成されたら、直ちにそのアカウントにログインして、パスワードを変更してください。アカウントの作成から最初のログインまでの間隔が空くと、他のユーザがそのアカウントへのログインに成功し、システムに損害を与える機会を得る可能性があります。同様に、パスワードの変更を怠ったり、変更できなかったりすると、システムが脆弱な状態のままになります。どのような損害が生じるかは、他にどのようなセキュリティ対策を講じているかに大きく依存します。

アカウントの作成時には、パスワードの最小限の長さと、パスワードをユーザが選択できるのか、それともシステムに生成させるのかも知らされるはずです。

### 3.1.2 パスワードに関するシステム制限の順守

システムは、次のようにしてパスワードが許容可能かどうかを調べます。

- 新しいパスワードをシステム辞書と自動的に比較します。これにより、パスワードが自国語の単語ではないことを確認します。
- 以前使用したパスワードの履歴リストを保持し、新しいパスワードをこのリストと比較して、古いパスワードが再利用されていないことを確認します。
- パスワードの最小限の長さを強制します。この値は、システム管理者が各ユーザの UAF レコードに指定します。

## 3.2 使用するパスワードのタイプ

OpenVMS システムで認識されるパスワードには、複数のタイプがあります。一般に、ログインするときはユーザ・パスワードを入力する必要があります。場合によっては、ユーザ・パスワードでログインする前にシステム・パスワードを入力して、特定のターミナルへのアクセス権を獲得する必要があります。セキュリティ要件の高いシステムでは、第 1 パスワードと第 2 パスワードの入力が必要な場合もあります。

外部認証機能が有効になっているシステムで外部認証されたユーザの場合は、OpenVMS のパスワード・プロンプトで LAN マネージャのパスワードを入力します。詳細については、7.4 節を参照してください。パスワードのタイプを、表 3-2 に示します。

表 3-2: パスワードのタイプ

パスワード	説明
ユーザ・パスワード	ほとんどのアカウントに対して要求されます。ユーザ名を入力すると、パスワードの入力を求められます。第 1 パスワードと第 2 パスワードの両方を要求するアカウントの場合は、2 つのパスワードを入力する必要があります。
システム・パスワード	特定のターミナルへのアクセスを制御するためのパスワードで、セキュリティ管理者の判断により要求されます。システム・パスワードは、通常、ダイアルアップ回線やバブリック・ターミナル・ラインなど、不正使用の対象となる恐れがあるターミナルへのアクセスを制御するために使用します。
第 1 パスワード	第 1 パスワードと第 2 パスワードの両方を要求するアカウントに対して入力する 2 つのユーザ・パスワードのうち、最初に入力するパスワード。
第 2 パスワード	<p>第 1 パスワードと第 2 パスワードの両方を要求するアカウントに対して入力する 2 つのユーザ・パスワードのうち、2 番目に入力するパスワード。第 2 パスワードによって、ユーザ・アカウントに対するセキュリティ・レベルが向上します。</p> <p>通常、一般のユーザは第 2 パスワードを知りません。管理者やその他の責任者が立ち会って、第 2 パスワードを入力する必要があります。アプリケーションによっては、アカウントが使用されている間、監督者がずっと立ち会う場合もあります。このように、第 2 パスワードを設定することで、ログインの制御とログイン後の処置がしやすくなります。</p> <p>第 2 パスワードは、手間がかかり、不便場合があります。第 2 パスワードの使用が妥当と考えられるは、セキュリティ要件が最高であるサイトに限られます。二重パスワードの使用が妥当なアカウントの例としては、データベースの緊急修復を可能にするために通常のアクセス制御を迂回するアカウントなどが考えられます。</p>

### 3.2.1 システム・パスワードの入力

ユーザは、自分に割り当てられた 1 つまたは複数のターミナルにログインするときに、システム・パスワードを指定する必要があるかどうかについて、セキュリティ管理者から通知されます。最新のシステム・パスワード、システム・パスワードの

変更頻度，および変更された場合の新しいシステム・パスワードの入手方法については，セキュリティ管理者に問い合わせてください。

システム・パスワードを入力するには，以下の手順を実行します。

1. ターミナルから認識文字（通常はベル）による応答があるまで Return キーを押します。

```
Return
<bell>
```

2. システム・パスワードを入力し，Return キーを押します。

```
Return
```

上記の例で示したように，プロンプトや入力した文字のエコーバックは表示されません。正しいシステム・パスワードを入力しなかった場合は，システムから何の応答もありません。このため，そのターミナルでシステム・パスワードが要求されることを知らないと，最初はシステムが機能不全に陥っているように見えます。システムからの応答がない場合は，入力したパスワードが正しくなかったと判断して，パスワードを入力し直してください。

3. 正しいシステム・パスワードを入力すると，システム通知メッセージが（設定されていれば）表示され，続いて Username: プロンプトが表示されます。

たとえば，次のように表示されます。

```
MAPLE - A member of the Forest Cluster
        Unauthorized Access Is Prohibited
```

```
Username:
```

### 3.2.2 第 2 パスワードの入力

セキュリティ管理者は，アカウントの作成時に，そのアカウントに第 2 パスワードを使用する必要があるかどうかを決定します。第 1 パスワードと第 2 パスワードを要求するアカウントでは，ログイン時に 2 つのパスワードを入力する必要があります。どちらのパスワードにも，パスワードの最小限の長さ（セキュリティ管理者が各ユーザの UAF レコードに指定した値）が適用されます。

第 1 パスワードと第 2 パスワードを要求するログインの例を以下に示します。

```
WILLOW - A member of the Forest Cluster
        Welcome to OpenVMS on node WILLOW
```

```
Username: RWOODS
Password: Return
Password: Return
```

```
Last interactive login on Friday, 11-DEC-2001 10:22
$
```

単独のパスワードによるログインと同じように、ログイン操作全体に対して一定の制限時間が設定されています。第2パスワードを時間内に入力しないと、ログイン時間が時間切れとなります。

### 3.3 アカウントのタイプごとのパスワード要件

OpenVMS システムには、5 種類のユーザ・アカウントが用意されています。

- ユーザまたはセキュリティ管理者が定期的に変更するパスワードによって保護されるアカウント。このタイプのアカウントが最も一般的です。
- パスワードがプログラミングされた認証カードによって保護されるアカウント。サードパーティ製品の多くは、このタイプの認証メカニズムをサポートします。
- 常にパスワードを要求するが、ユーザによるパスワードの変更を認めないアカウント。パスワードをロックする (UAF レコードに LOCKPWD フラグを設定する) ことにより、セキュリティ管理者はパスワードに対するすべての変更を制御します。
- 制限付きアカウントでは、ユーザによるシステムの使用が制限され、必要に応じてパスワードが要求されます。
- オープン・アカウントでは、パスワードが要求されません。パスワードはヌルです。オープン・アカウントにログインするときには、パスワードの入力は求められず、パスワードを入力する必要がありません。直ちにコマンドの入力を開始できます。オープン・アカウントを使用するとシステムへのアクセスが誰にでも許可されるため、オープン・アカウントはセキュリティ要件が最小限のサイトでのみ使用し、通常は制限付きアカウントとして設定してください。

### 3.4 ログインのタイプとログイン・クラス

ログインには、会話型と非会話型があります。会話型ログインの場合は、OpenVMSでのユーザ名とパスワードを入力します。非会話型ログインの場合は、システムがユーザの識別と認証の処理を行うため、ユーザ名とパスワードの入力は求められません。ここで使用されている会話型という用語の意味は、DCL のレキシカル関数 F\$MODE() で定義されている会話モード・プロセスの場合とは異なります。F\$MODE 関数の詳細については、『*OpenVMS DCL* デイクショナリ』を参照してください。

会話型ログインと非会話型ログインの他にも、OpenVMS オペレーティング・システムではログインのさまざまなクラスが認識されます。ユーザが属するログイン・クラスは、ユーザがシステムにログインする方法によって決まります。システム管理者は、ユーザのログイン・クラス、ログインした曜日および時刻に基づいて、当該ユーザによるシステムへのアクセスを制御します。

### 3.4.1 会話型ログイン (ローカル、ダイアルアップ、および遠隔ログイン)

会話型ログインには、以下のログイン・クラスがあります。

- ローカル

中央プロセッサに直接接続されているターミナル、または中央プロセッサと直接通信するターミナル・サーバからログインします。

- ダイアルアップ

モデムと電話回線を使用してコンピュータ・システムとの接続を確立するターミナルにログインします。システムが使用するターミナルによっては、いくつかの追加手順を実行する必要があります。必要な操作については、サイトのセキュリティ管理者に問い合わせてください。

- 遠隔

DCL の SET HOST コマンドを入力して、ネットワーク経由で特定のノードにログインします。たとえば、HUBBUB という遠隔ノードにアクセスするには、次のコマンドを入力します。

```
$ SET HOST HUBBUB
```

HUBBUB ノード上のアカウントへのアクセスが許可されている場合は、ローカル・ノードからそのアカウントにログインできます。この場合、HUBBUB ノード上の機能にアクセスできますが、物理的にはローカル・ノードに接続された状態のままです。

### 3.4.2 外部認証を使用したログイン

外部認証されたユーザの場合は、OpenVMS のログイン・プロンプトで LAN マネージャのユーザ ID とパスワードを入力してログインします。LAN マネージャのユーザ ID は、OpenVMS のユーザ名と同じ場合と異なる場合があります。

システムの外部認証機能を有効にした状態でのログインの詳細については、7.4 節を参照してください。

### 3.4.3 情報メッセージの解釈

コンピュータに直接接続されているターミナルからログインすると、OpenVMS システムが情報メッセージを表示します。例 3-1 は、これらのメッセージの大部分を示しています。

#### 例 3-1: ローカル・ログイン・メッセージ

---

```
WILLOW - A member of the Forest Cluster          [1]
      Unlawful Access is Prohibited

Username: RWOODS
Password:
```

### 例 3-1: ローカル・ログイン・メッセージ (続き)

---

You have the following disconnected process: [2]

Terminal	Process name	Image name
VTA52:	RWOODS	(none)

Connect to above listed process [YES]: **NO**

Welcome to OpenVMS on node WILLOW [3]

Last interactive login on Wednesday, 1-DEC-2001 10:20 [4]

Last non-interactive login on Monday, 30-NOV-2001 17:39 [5]

2 failures since last successful login [6]

You have 1 new mail message. [7]

\$

上記の例は、次のことを示します。

1. アナウンスメント・メッセージの中に、ノード（および、該当する場合はクラスタ）が表示されます。また、権限のないユーザに対して不法なアクセスの禁止を警告するメッセージも表示されます。システム管理者またはセキュリティ管理者は、このメッセージの形式と内容を調整できます。
2. 切断ジョブ・メッセージは、ユーザの最後のログイン後に、当該ユーザのプロセスが切断されたにもかかわらず、そのプロセスがまだ存在することを知らせるメッセージです。この場合は、その古いプロセスに再接続して、プロセスから切断される前の状態に戻すことができます。  
切断ジョブ・メッセージが表示されるのは、以下の条件が満たされた場合だけです。
  - 割り込みの発生したターミナルが仮想ターミナルとして設定されている。
  - ターミナルが切断可能なターミナルとして設定されている。
  - 最近のセッションで、そのターミナル経由の CPU への接続が、ログアウトする前に切断された。一般に、切断されたジョブに再接続することでシステム・セキュリティに特別な問題が発生することはないので、セキュリティ管理者は再接続をユーザに許可するはずです。ただし、セキュリティ管理者は、ターミナルの設定を変更し、システム上での仮想ターミナルの使用を禁止することによって、この機能の使用を禁止できます。
3. ウェルカム・メッセージには、実行中の OpenVMS オペレーティング・システムのバージョン番号とログインしているノードの名前が表示されます。システム管理者は、別のメッセージを使用したり、メッセージが表示されないようにしたりできます。
4. 最後の正常な会話型ログイン・メッセージには、ローカル、ダイアルアップ、または遠隔ログインの最後のログイン完了時間が表示されます。これらのタイプのいずれかを親として持つサブプロセスからのログインは、カウントされません。
5. 最後の正常な非会話型ログイン・メッセージには、非会話型（バッチまたはネットワーク）ログインの最後の完了時間が表示されます。
6. ログイン失敗回数メッセージは、失敗したログイン操作の回数を示します。カウントされるのは、誤ったパスワードに起因するログインの失敗のみです。ユーザの注意を促すため、メッセージの表示後にベルが鳴ります。
7. ユーザに新しいメール・メッセージが届いている場合は、新規メール・メッセージが表示されます。

セキュリティ管理者は、ノード名とオペレーティング・システムの識別情報が含まれる通知メッセージとウェルカム・メッセージの表示を抑制できます。ログイン手順はシステムによって異なるため、これらの情報が表示されなければ、ログインが難しくなります。

最後の正常ログイン・メッセージとログイン失敗回数メッセージは省略可能です。セキュリティ管理者は、これらの表示をまとめて有効化または無効化できます。中～高レベルのセキュリティを必要とするサイトでは、これらのメッセージを表示することで、進入の試みがあったかどうかを知ることができるようにします。また、これらのメッセージによってシステムがログインが監視されていることがわかるので、不正ユーザに対する抑止効果も得られます。

ユーザがログインするたびに、最後の正常ログインの値とログイン失敗回数の値が再設定されます。会話形式でアカウントにアクセスし、ログイン時に誤ったパスワードを指定しなかった場合、最後の正常な非会話型ログイン・メッセージとログイン失敗メッセージは表示されません。

#### 3.4.4 システムがユーザに代わってログインする場合 (ネットワーク・ログインとバッチ・ログイン)

非会話型ログインには、ネットワーク・ログインとバッチ・ログインがあります。

ユーザが遠隔ノード上でネットワーク・タスク(ディレクトリ内容の表示、別のノード上のディレクトリに格納されているファイルのコピーなど)を開始すると、システムがネットワーク・ログインを実行します。この場合、現在のシステムと遠隔システムの両方が同じネットワーク内のノードでなければなりません。ファイルを指定するときは、ターゲット・ノードとアクセス制御文字列を指定します。アクセス制御文字列としては、遠隔ノードにおけるユーザ名とパスワードを指定します。

たとえば、PARIS という遠隔ノードにアカウントのある Greg というユーザが次のコマンドを入力すると、ネットワーク・ログインが実行されます。

```
$ DIRECTORY PARIS"GREG 8G4FR93A"::WORK2:[PUBLIC]*.*;*
```

このコマンドにより、WORK2 というディスク上のパブリック・ディレクトリ内にあるすべてのファイルが一覧表示されます。また、パスワードが 8G4FR93A であることもわかります。同じことをさらに安全に実行するには、PARIS ノードで代理アカウントを使用します。代理ログインの例については、3.9.2 項を参照してください。

ユーザが発行したバッチ・ジョブが実行されると、システムがバッチ・ログインを実行します。ジョブの構築に対する許可は、ジョブを発行した時点で決定されます。システムがジョブ実行の準備を行うときに、ジョブ・コントローラがユーザのアカウントにログインする非会話型プロセスを生成します。ジョブがログインするときは、パスワードは必要ありません。



### 3.5 ログインの失敗 (ユーザがログインできない場合)

ログインはさまざまな理由で失敗します。たとえば、いずれかのパスワードが変更された場合や、アカウントの有効期限が切れた場合に失敗します。ネットワーク経由で、またはモデム経由のログインを試みたときにその権限がない場合にも失敗します。表 3-3 に、ログインに失敗する一般的な原因を示します。

表 3-3: ログイン失敗の原因

失敗の症状	原因
ターミナルから応答がない。	ターミナルに欠陥がある、システム・パスワードを必要とするターミナルである、ターミナルに電源が投入されていない、または配線の不備やモデムの誤設定・誤動作によって通信に問題が発生している。
どのターミナルからも応答がない。	システムがダウンしているか、過負荷になっている。
システム・パスワードを入力したが、ターミナルから応答がない。	システム・パスワードが変更されている。
以下のシステム・メッセージが表示される。	
“ User authorization failure ”	ユーザ名またはパスワードの入力に誤りがあった。アカウントまたはパスワードの有効期限が満了した。
“ Not authorized to log in from this source ”	現在のログイン・クラス（ローカル、ダイヤルアップ、遠隔、会話型、バッチ、ネットワーク）が禁止されている。
“ Not authorized to log in at this time ”	この時間帯または曜日のログインが許可されていない。
“ User authorization failure ”（かつ、既知のユーザ障害が発生していない）	該当するユーザ名を使用してターミナルからシステムへの侵入と考えられる行為があったため、当該ユーザ名によるそのターミナルからのすべてのログインがシステムによって一時的に無効にされた。

以下の各節では、ログイン失敗の原因についてさらに詳しく説明します。

#### 3.5.1 システム・パスワードを必要とするターミナルの使用

使用しようとしているターミナルがシステム・パスワードを要求する場合、そのことを知らないユーザはログインできません。ユーザがシステム・パスワードを入力するまで、ログインはすべて失敗します。

システム・パスワードを知っている場合は、3.2.1 項で説明した手順を実行します。それでもログインに失敗する場合は、システム・パスワードが変更されている可能性があります。システム・パスワードを必要としない別のターミナルにログインするか、新しいシステム・パスワードを教えてください。

システム・パスワードを知らず、そのことが問題であると考えられる場合は、別のターミナルでログインを試みます。

### 3.5.2 ログイン・クラスの制限の順守

UAF レコードで禁止されているクラスのログインを実行すると、ログインに失敗します。たとえば、セキュリティ管理者はユーザによるネットワーク経由のログインを制限できます。この場合、ネットワーク・ログインを実行すると、そこからのログインが許可されていないことを示すメッセージが表示されます。

ネットワーク・ジョブに割り当てられた作業時間を超過すると、ネットワーク・ジョブが終了しません。この制限は、新しいネットワーク接続のみに適用され、既存の接続には適用されません。

セキュリティ管理者は、ローカル、遠隔、ダイアルアップ、バッチ、ネットワークの各クラスを取捨選択することにより、ユーザのログインを制限できます。各クラスの詳細については、3.4.1 項と 3.4.4 項を参照してください。

### 3.5.3 特定の日に時に使用が限定されたアカウントの使用

作業時間に関する制限のためにログインできないこともあります。システム管理者またはセキュリティ管理者は、時間帯や曜日に基づいてシステムへのアクセスを制御できます。このような制限は、ログイン・クラスに対して適用されます。セキュリティ管理者は、同じ作業時間制限をすべてのログイン・クラスに適用したり、ログイン・クラスごとに異なる制限を設定したりできます。該当するログイン・クラスで禁止されている時刻にログインしようとする、ログインに失敗します。その時刻のログインが許可されていないことを示すメッセージが表示されます。

作業時間制限がバッチ・ジョブに適用されると、許可されている作業時間の範囲外で実行するようにスケジューリングされたジョブは、発行しても実行されません。また、システムは実行されなかったジョブを自動的に次の作業時間中に再発行することもあります。同様に、何らかのジョブを開始し、そのジョブを許可されている時間帯を超えて実行しようとしても、割り当てられた作業時間が終了すると、ジョブ・コントローラが未完了のジョブを強制終了します。このようなジョブの終了方法は、すべてのジョブに適用されます。

### 3.5.4 ダイアルアップ・ログインで正しいパスワードを入力しなかった場合

セキュリティ管理者は、ダイアルアップ・ログインで接続が自動的に切断されるまでの間にユーザがパスワードを入力できる回数を制限できます。

ログインに失敗しても、指定された操作回数に到達していない場合は、Return キーを押してもう一度ログイン操作を実行してください。ログインに成功するか、制限回数に到達するまでは、この操作を繰り返すことができます。接続が切断された場合は、アクセス回線に再ダイヤルして、始めからやり直します。

通常、ダイヤルアップ・ログインの失敗回数を制限するのは、権限のないユーザが試行錯誤の繰り返しによってパスワードを知る試みを阻止するためです。ダイヤルアップ回線では、もともと権限がなくても匿名での操作が可能です。もちろん、ダイヤルアップするたびにログインの再試行回数を制限しても、この種の侵入行為がなくなるわけではありません。回数制限の目的は、侵入を試みる人がログイン操作を新たに行うために、何度もダイヤルする必要が生じるようにすることです。

### 3.5.5 侵入回避手順が有効になる条件

誰かが同じターミナルから同じユーザ名でログインしようとして何度か失敗すると、システムは侵入者がそのユーザ名を使用してシステムに不正にアクセスしようとしていると判断します。

セキュリティ管理者は、自分の判断でシステムのすべてのユーザを対象に侵入回避手順を有効にできます。セキュリティ管理者は、パスワードを入力できる回数と期間を制御します。侵入回避手順が有効になると、指定された時間内は（たとえ正しいパスワードを入力しても）ターミナルにログインできません。ログインを再試行できるようになるまでの時間については、セキュリティ管理者に問い合わせてください。または、別のターミナルを使用してログインを実行することもできます。

侵入回避手順によってログインが禁止されていると考えられるが、自分自身はログインに失敗した覚えがない場合は、直ちにセキュリティ管理者に連絡します。セキュリティ管理者と共に、もう一度ログインを試み、最後のログイン以降のログイン失敗回数を示すメッセージをチェックして、侵入行為についての推測が正しいかどうかを確認します。ログイン・メッセージを通常は表示しないシステムの場合は、セキュリティ管理者が登録ユーティリティ (AUTHORIZE) を使用して UAF レコード内のデータを調べることができます。すばやく対応することで、セキュリティ管理者は誰かが別のターミナルからログインしようとしていることを突き止められます。

## 3.6 パスワードの変更

定期的にパスワードを変更すると、システムのセキュリティが向上します。パスワードを変更するには、DCL の SET PASSWORD コマンドを使用します。

システム管理者は、ユーザにパスワードを自由に選択させることも、パスワード変更時の自動パスワード・ジェネレータの使用を義務付けることもできます。ユーザが自分でパスワードを選択する場合は、パスワードの長さや許容条件に関する制限を守る必要があります（3.1.2 項を参照）。たとえば、選択したパスワードが短すぎると、次のメッセージが表示されます。

```
%SET-E-INVPWDLEN, invalid password length - password not changed
```

3.1 節に、安全なパスワードの指定に関するガイドラインと具体例を示します。

一定期間内にパスワードを変更できる回数に限度はありません。

### 3.6.1 ユーザ自身によるパスワードの選択

システム管理者が自動パスワード・ジェネレータの使用を義務付けていない場合は、SET PASSWORD コマンドを実行すると新しいパスワードを入力するよう求められます。続いて、次に示すように、確認のために新しいパスワードを再入力するよう求められます。

```
$ SET PASSWORD
```

```
Return
```

```
New password:
```

```
Verification:
```

同じパスワードを2回入力しないと、パスワードは変更されません。同じパスワードを2回入力したときは、画面に何も表示されません。コマンドによってパスワードが変更され、DCL プロンプトに戻ります。

セキュリティ管理者がパスワード・ジェネレータの使用を義務付けていない場合でも、システムのセキュリティを向上させるため、パスワード・ジェネレータを使用することを強くお勧めします。生成パスワードの使用方法については、3.6.2 項で説明します。

### 3.6.2 生成パスワードの使用

システムのセキュリティ管理者がシステム側でパスワードを自動生成する必要があると判断している場合は、DCL の SET PASSWORD コマンドを入力すると、パスワードの選択リストが表示されます。パスワードの自動生成が要求されないシステムでは、SET PASSWORD コマンドに /GENERATE 修飾子を指定すると、パスワードの選択リストが表示されます。生成される文字の並びは、簡単に覚えられるように普通の言葉の単語によく似ていますが、外部の人間が推測するのが困難なくらいには複雑です。システムによって生成されるパスワードは長さが一定でないため、推測するのはいっそう困難です。

---

#### 注意

---

パスワード・ジェネレータは、基本的な音節規則を使用して単語を生成しますが、実際の言葉に関する知識を持っているわけではありません。このため、偶然に不快な言葉が生成されてしまう場合があります。

---

次に示す OpenVMS VAX の例では、文字の無作為の並びで構成されたパスワードのリストが自動生成されています。この例のユーザに関しては、パスワードの最小限の長さが UAF レコードで 8 文字に設定されています。

```
$ SET PASSWORD
Old password:  [1]

cigtawdpau    cig-tawd-pau    [2]
adehecun      a-de-he-cun
ceebatorai    cee-ba-to-rai
arhoajabad    ar-hoa-ja-bad
Choose a password from this list, or press Return to get a new list [3]

New password:  [4]

Verification:  [5]

$ [6]
```

上記の例は、次のことを示します。

1. ユーザが古いパスワードを正しく入力して、Return キーを押します。
2. 長さが 8 ~ 10 文字のパスワード候補が 5 つ表示されます。各パスワード候補の右側には、同じ単語を音節で区切って表現したものが表示されます。通常、発音しやすいパスワードほど覚えやすく、パスワードに適しています。
3. 新しいリストを要求できることを示すメッセージが表示されます。新しいリストを要求するには、新しいパスワードを入力するプロンプトで Return キーを押します。
4. ここでは、ユーザが最初に表示された 5 つのパスワード候補の中から 1 つを入力して、Return キーを押します。
5. 入力されたパスワードが自動パスワード・ジェネレータによって作成されたパスワードであることが認識され、確認のプロンプトが表示されます。ユーザがもう一度新しいパスワードを入力し、Return キーを押します。
6. システムによってパスワードが変更され、DCL プロンプトに戻ります。

自動パスワード生成のデメリットの 1 つは、選択したパスワードを忘れてしまう可能性があるということです。ただし、表示されたどのパスワード候補も気に入らない場合や、どのパスワードも簡単に覚えられないと思われる場合は、別のリストを要求できます。

自動パスワード生成のさらに深刻な欠点は、このコマンドを実行したときにパスワードの候補が表示されてしまうことです。アカウントを保護するためには、誰にも知られないようにパスワードを変更しなければなりません。ビデオ・ターミナルで変更する場合は、コマンドが完了した後、パスワード候補が表示された画面を消去してください。DECwindows 環境で変更する場合は、「コマンド」メニューの「保存行消去」を使用して、画面リコール・バッファからパスワードを消

去してください。印刷ターミナルを使用する場合は、ハードコピー出力をすべて適切に廃棄してください。

このようにしてもパスワードを保護できなかったことが後で判明した場合は、直ちにパスワードを変更してください。サイトのポリシーに従い、あるいは、アカウントが危険な状態にあった時間の長さから判断して、自分のアカウントを介したセキュリティ侵害が生じた可能性をシステム管理者に知らせます。

### 3.6.3 第 2 パスワードの変更

第 2 パスワードを変更するには、DCL の SET PASSWORD/SECONDARY コマンドを使用します。第 1 パスワードの変更手順と同様に、現在の第 2 パスワードと新しい第 2 パスワードを指定するよう求められます。第 2 パスワードを削除するには、新しいパスワードの入力とその確認入力を求められたときに、それぞれ Return キーを押します。

第 1 パスワードと第 2 パスワードは別々に変更できますが、パスワードの有効期限は同じなので、同じ変更頻度の条件が適用されます。パスワードの有効期限については、3.7 節を参照してください。

### 3.6.4 ログイン時に行うパスワード変更

現在のパスワードの有効期限が満了していなくても、ユーザ名の後に /NEW\_PASSWORD 修飾子を付けることにより、ログイン時にパスワードを変更できます。

```
WILLOW - A member of the Forest Cluster

Username: RWOODS/NEW_PASSWORD
Password:
    Welcome to OpenVMS on node WILLOW
    Last interactive login on Tuesday, 7-NOV-2001 10:20
    Last non-interactive login on Monday, 6-NOV-2001 14:20

Your password has expired; you must set a new password to log in
New password:
Verification:

ユーザ名の後に /NEW_PASSWORD 修飾子を入力すると、ログインの直後に新しいパスワードを設定するよう要求されます。
```

## 3.7 パスワードとアカウントの有効期限

システム管理者は、パスワードまたはアカウントそのものの有効期限が特定の日時に自動的に期限切れとなるように、アカウントを設定できます。パスワードに有効期限を設定すると、ユーザが定期的にパスワードを変更しなければならないため、システムのセキュリティが向上します。アカウントの有効期限は、アカウントを必要な期間だけ使用可能にしたいときに便利です。

### 3.7.1 期限切れパスワードの変更

パスワードの有効期限が近づくと、予告の警告メッセージが表示されます。警告メッセージは、期限の 5 日前からログインするたびに表示されます。このメッセージは新着メールを知らせるメッセージの直後に表示され、注意を促すためにターミナルでベルが鳴ります。このメッセージは、次のようにパスワードの有効期限が迫っていることを示します。

```
WARNING -- Your password expires on Thursday 19-DEC-2001 15:00
```

期限切れになる前にパスワードを変更しなかった場合は、ログイン時に次のメッセージが表示されます。

```
Your password has expired; you must set a new password to log in  
New
```

```
password:
```

システムにより、新しいパスワードの入力を求められるか、自動パスワード生成が有効になっている場合には、新しいパスワードをリストから選択するように求められます (3.6.2 項を参照)。ここで Ctrl/Y を押すと、ログインを強制終了できます。その場合、次にログインしようとしたときに、パスワードの変更を再度求められます。

#### 第 2 パスワードを使用している場合

アカウントで第 2 パスワードを使用している場合 (3.2 節を参照) は、第 2 パスワードが第 1 パスワードと同時に有効期限を迎えることがあります。この場合、両方のパスワードを変更するよう求められます。第 1 パスワードを変更した後、第 2 パスワードを変更する前に Ctrl/Y を押すと、ログインに失敗します。この場合、パスワードの変更はシステムに記録されません。

#### パスワードを変更しなかった場合

システム管理者がログイン時の期限切れパスワードの変更をユーザに強制していない場合は、パスワードの期限切れ後にユーザがログインすると、最終警告メッセージが表示されます。

```
WARNING -- Your password has expired; update immediately with  
SET
```

```
PASSWORD!
```

この時点で、パスワードを変更しなかったり、パスワードを変更する前にシステムに障害が発生したりすると、二度とログインできなくなります。再度アクセスできるようにする方法については、システム管理者に問い合わせてください。

### 3.7.2 期限切れアカウントの更新

特定の目的で限られた期間だけアカウントが必要な場合は、アカウントの作成者がアカウントの有効期限を指定できます。たとえば、大学の学生用アカウントは、通常、学期ごとに1回ずつ登録します。

期限切れアカウントは、自動的にログインが拒否されます。アカウントの有効期限前に警告メッセージは表示されないため、あらかじめアカウントの有効期間を知っておくことが重要です。アカウントの有効期限は UAF レコードに格納されています。このレコードの取得や表示は、SYSPRV 特権またはそれと同等の権限を持つユーザ（通常はシステム管理者またはセキュリティ管理者）が登録ユーティリティ (AUTHORIZE) を使用した場合にのみ可能です。

アカウントの有効期限が切れると、次にログインしようとしたときに認証失敗メッセージが表示されます。有効期限の延長が必要な場合は、各サイトで定義されている手順に従ってください。

## 3.8 パスワードの保護に関するガイドライン

既知のパスワードを使用したシステムへの不正アクセスは、多くの場合パスワードの所有者が自分のパスワードを他人に漏らしたことに起因します。自分のパスワードを誰にも教えないことが、何よりも重要です。

以下の規則を守ることで、パスワードを適切に保護できるようになります。

- 簡単には推測できない長いパスワードを選択します。辞書に載っているような自国語の言葉は使用しないようにします。パスワードに数字を入れることを検討します。システムにパスワードを自動生成させる方法もあります。
- パスワードは絶対に書き留めないでください。
- 自分のパスワードは絶対に他人に教えないでください。パスワードを他のユーザに知られた場合は、直ちに変更します。
- 電子メール・メッセージの本文も含めて、どのようなファイルにもパスワードを記録しないようにします。他人がパスワードを知らせてきた場合は、直ちにその情報を削除します。

パスワードに特定の文字列が付随する場合、ファイル内でパスワードが簡単に検索できてしまいます。たとえば、アクセス制御文字列においては、ユーザ名とパスワードの後には、必ず二重引用符と2つのコロン (":") が付きます。システムに侵入しようとする人は、十分に保護されていないファイルからこの文字列を検索することにより、パスワードを入手できます。また、次のようにテキスト・ファイル内で “password” という言葉を使用していると、パスワードが簡単に漏れてしまう可能性があります。

```
My password is GOBBLEDYGOOK.
```



- カードを使用してバッチ・ジョブを発行する場合、パスワード・カードから別のユーザにパスワードが漏れることがあるため、パスワード・カードは放置しないようにします。
- 異なるシステムのアカウントに同じパスワードを使用しないようにします。  
同じユーザのアカウントが存在するすべてのシステムで、権限のないユーザが同じパスワードを試す可能性があります。最初にパスワードが漏れたアカウントに重要な情報がほとんどなくても、別のアカウントに重要な情報や特権があれば、最終的にはきわめて大きなセキュリティ侵害が発生する可能性があります。
- すでに電源が投入されているターミナルにログインする場合は、あらかじめ Break キーを押して安全ターミナル・サーバ機能を（使用可能であれば）起動します。安全サーバでは、OpenVMS のログイン・プログラムがユーザのログインを受け付ける唯一のプログラムになるため、パスワード・グラバ・プログラムにパスワードが漏れる可能性がなくなります。この処置は、特に公共のターミナル・ルームで作業する場合に必要です。

パスワード・グラバ・プログラムは、何も表示されていない画面、システムがクラッシュ後に初期化された直後のように見える画面、実際にはログアウトしていないのにログアウトしたように見える画面などを表示する特殊なプログラムです。ユーザがログインしようとする時、このプログラムは通常のログイン手順をたどるため、ユーザはいつものようにユーザ名とパスワードを入力しているものと考えます。しかし、このプログラムは、これらの重要な情報を受け取ると、不正に侵入しようとする人にその情報を渡した後、ログインに失敗したことを示す画面を表示します。ユーザはパスワードを誤って入力したと思い、自分のパスワードが他人に漏れたことに気づきません。

- パスワードは、共用する場合を除き、3 ～ 6 か月ごとに変更するようにします。パスワードの共用はできる限り避けてください。パスワードを共用する場合は、パスワードを毎月変更するようにします。
- 何らかの理由でパスワードが外部に漏れたと考えられる場合は、直ちにパスワードを変更します。また、その事実をセキュリティ管理者に報告します。
- ログインした後は、ターミナルを無人の状態にしないようにします。

システムに障害が発生し、その後復旧したと考えられる場合でも、実際には何者かがパスワード盗用プログラムをロードしている可能性があります。ターミナルに正常なログアウト・メッセージが表示されているように見える場合でも、通常のログアウト・プロセスによるものではない可能性があります。

- 最終ログイン・メッセージを定期的を確認してください。パスワード盗用プログラムが表示するログインの失敗は、ユーザにはログインの失敗のように見えても、実際にはログインの失敗回数に数えられません。ログイン失敗数がログインの失敗後に表示されなかったり、実際の失敗回数より 1 回分少ない場合は、

注意してください。ログインに時にこれらの現象を含む何らかの異常を経験した場合は、直ちにパスワードを変更して、セキュリティ管理者に知らせます。

## 3.9 ネットワーク・セキュリティに関する考慮事項

この節では、ファイル指定におけるアクセス制御文字列の使用方法と、代理ログインによってネットワーク・アクセスのセキュリティを向上させる方法について説明します。

### 3.9.1 アクセス制御文字列内の情報の保護

ネットワーク・アクセス制御文字列は、DECnet for OpenVMS ネットワークで使用する DCL コマンドのファイル指定の部分に入れることができます。アクセス制御文字列によって、ローカル・ノードのユーザが遠隔ノード上のファイルにアクセスできるようになります。

アクセス制御文字列は、次のように遠隔アカウントのユーザ名とユーザのパスワードを二重引用符で囲んだものです。

```
NODE"username password"::disk:[directory]file.typ
```

アクセス制御文字列は、遠隔アカウントに不正侵入するのに必要な情報をすべて含んでいるため、深刻なセキュリティの脅威になります。アクセス制御文字列の情報を保護するには、以下のようにします。

- ハードコピーおよびビデオ・ターミナルを通じて情報が漏れないようにします。ハードコピー・ターミナルを使用している場合は、ハードコピー出力を適切に廃棄します。ビデオ・ターミナルを使用している場合は、ネットワーク・ジョブを完了した後に画面を消去し、DCL の RECALL/ERASE コマンドを使用してリコール・バッファを空にします。こうすると、他のユーザがコマンド行を表示するための Ctrl/B キー・シーケンスまたは DCL の RECALL/ALL コマンドを使用しても、パスワードは表示されません。DECwindows ユーザは、「コマンド」メニューの「保存行消去」を使用して画面を消去できます。消去しないと、他の DECwindows ユーザがスクロール・バーを使って、以前に入力されたテキストを見る可能性があります。
- アクセス制御文字列を含むネットワーク・コマンドを、探索の対象になりやすいコマンド・プロシージャに指定しないようにします。
- アクセス制御文字列をコマンド・プロシージャに指定しなければならない場合は、第 4 章で説明する方法を使用してそれらのファイルに最適なファイル保護を設定します。
- 評価済みの構成でのアクセス制御文字列の使用は認められていません。評価済みの構成でシステムが運用されているかどうかは、システム管理者に問い合わせてください。

アクセス制御文字列の使用を避けるには、3.9.2 項で説明する代理ログイン・アカウントの使用をお勧めします。

### 3.9.2 代理ログイン・アカウントの使用によるパスワードの保護

代理ログインを使用すると、ユーザ名とパスワードを指定したアクセス制御文字列を使用せずに、ネットワーク経由でファイルにアクセスできます。したがって、代理ログインには、次のようなセキュリティ上の利点があります。

- 要求の発信元ターミナルにパスワードがエコーバックされません。
- システム間でのパスワードの受け渡しがないため、パスワードが暗号化されていない形式で傍受される恐れがありません。
- 遠隔アクセスの手順を実行するコマンド・ファイルにパスワードを指定する必要がありません。

ユーザが代理ログインを開始するには、遠隔ノードのシステム管理者またはセキュリティ管理者があらかじめ代理アカウントを作成しなければなりません。代理アカウントは、通常のアカунツと同じように登録ユーティリティ (AUTHORIZE) を使用して作成します。通常、代理アカウントは非特権アカウントです。セキュリティ管理者は、1 つのデフォルト代理アカウントと最大 15 のデフォルト以外の代理アカウントへのアクセスをユーザに許可できます。代理ログインを使用すると、システム管理者には設定の手間がかかりますが、より安全なネットワーク・アクセスが可能になり、ユーザがアクセス制御文字列を入力せずに済みます。

次の例は、通常のネットワーク・ログイン要求と代理ログイン要求の違いを示します。ここでは、以下のような条件を想定します。

- KMAHOGANY というユーザが次の 2 つのユーザ・アカウントを持っています。
  - BIRCH というノード上のアカウント (パスワードは "XYZ123ABC")
  - WALNUT というノード上のアカウント (パスワードは "A25D3255")
- KMAHOGANY は、BIRCH ノードに既にログインしています。
- KMAHOGANY は、WALNUT ノード上のアカウントにあるデフォルトのデバイスとディレクトリに格納されている BIONEWS.MEM というファイルをコピーしようとしています。

これらの条件を表した図を、次に示します。

KMAHOGANY は、次のようにアクセス制御文字列を使用して BIONEWS.MEM ファイルをコピーできます。

```
$ COPY WALNUT"KMAHOGANY A25D3255":BIONEWS.MEM BIONEWS.MEM
```

"A25D3255" というパスワードはエコー表示されるので、画面を見れば誰でもパスワードがわかります。一方、KMAHOGANY が BIRCH ノードから WALNUT ノードのアカウントに代理アクセスを行う場合、BIONEWS.MEM ファイルをコピーするためのコマンドは次のようになります。

```
$ COPY WALNUT::BIONEWS.MEM BIONEWS.MEM
```

KMAHOGANY は、アクセス制御文字列にパスワードを指定する必要があります。代わりに、システムが BIRCH ノードのアカウントから WALNUT ノードのアカウントへの代理ログインを実行します。このとき、パスワードの交換は行われません。

#### 汎用アクセス代理アカウントの使用

セキュリティ管理者は、フォーリン・ノードのユーザ・グループに汎用アクセス代理アカウントの共用を許可することもできます。たとえば、WALNUT ノードのセキュリティ管理者が以下の条件で汎用アクセス・アカウントを作成するとします。

- ユーザ名は GENACCESS。
- アクセスはネットワーク・ログインのみに限定。
- パスワードはアカウントの所有者のみが知っている。遠隔ユーザがパスワードを知る必要はないので、アカウントを保護しやすい。
- デフォルトのデバイスとディレクトリは STAFFDEV:[BIOSTAFF]。

セキュリティ管理者が BIRCH::KMAHOGANY に GENACCESS アカウントへの代理アクセスを許可すると、KMAHOGANY は次のコマンドを入力することによって BIONEWS.MEM ファイルをコピーできます。

```
$ COPY WALNUT::[KMAHOGANY]BIONEWS.MEM BIONEWS.MEM
```

BIONEWS.MEM ファイルは GENACCESS アccountのデフォルトのデバイスとディレクトリ (STAFFDEV:[BIOSTAFF]) がないため、KMAHOGANY は [KMAHOGANY] ディレクトリを指定しなければなりません。また、BIONEWS.MEM ファイルの保護は GENACCESS アccountに対してアクセスを許可するものでなければなりません。そうでない場合は、コマンドの実行に失敗します。

#### 代理アカウントの名前を指定する必要がある場合

特定のノード上にアクセスできる代理アカウントが複数あり、デフォルトの代理アカウントを使用したくない場合は、代理アカウントの名前を指定します。たとえば、KMAHOGANY が GENACCESS アccount (デフォルト) の代わりに PROXY2 という代理アカウントを使用する場合は、次のコマンドを入力します。

```
$ COPY WALNUT"PROXY2":[KMAHOGANY]BIONEWS.MEM BIONEWS.MEM
```

このコマンドにより、PROXY2 アカウントを使用して WALNUT ノード上の [KMAHOGANY] ディレクトリにある BIONEWS.MEM ファイルをコピーします。

### 3.10 アカウントおよびファイルへのアクセスの監査

侵入行為がないかどうかシステムを監視するのはセキュリティ管理者の仕事ですが、ユーザはセキュリティ管理者と協力して自分のアカウントやファイルへのアクセスを監査できます。

この節では、ユーザの最終ログイン時間を監視して、侵入行為があったかどうかを調べる方法について説明します。また、セキュリティ管理者と協力して特定の種類の監査を有効にする方法についても説明します。

#### 3.10.1 最終ログイン時間の確認

OpenVMS オペレーティング・システムでは、ユーザが自分のアカウントに最後にログインした時間に関する情報が UAF レコードに保管されます。ログイン時にこの情報を表示するかどうかは、セキュリティ管理者が決定します。中～高レベルのセキュリティ要件のサイトでは、多くの場合この情報を表示して、通常とは異なる説明のつかない正常ログインや、説明のつかないログインの失敗がないかどうかをユーザにチェックさせます。

ユーザが実際にはログインしなかった時間帯における会話型または非会話型のログインに関する情報が表示された場合は、直ちにそのことをセキュリティ管理者に知らせ、パスワードを変更します。セキュリティ管理者は会計情報ファイルと監査ログを使用してさらに詳しく調査します。

ログイン失敗メッセージが表示されたが、その失敗に覚えがない場合は、何者かがそのアカウントにアクセスしようとして失敗した可能性があります。このような場合は、パスワードをチェックして、3.8 節に示したパスワードのセキュリティに関する推奨事項を守っているどうかを確認します。守っていない場合は、直ちにパスワードを変更します。

ログイン失敗メッセージが表示されるはずなのに表示されない場合や、表示された失敗回数が実際より少ない場合は、パスワードを変更し、このようなログインの失敗に関する問題の兆候をセキュリティ管理者に報告します。

#### 3.10.2 重要ファイルへのアクセス制御エントリ (ACE) の追加

重要なファイルに不正にアクセスされた可能性がある場合は、セキュリティ管理者と協力してそれらのファイルへのアクセスを監査する戦略を立てることをお勧めします。

現状を調べて、標準の保護コードや汎用 ACL (第 4 章を参照) を使用してファイルを保護するために可能なあらゆる対策を講じたことを確認できた場合は、セキュリティ監査が必要であるという結論に達することもあります。

セキュリティ監査を指定するには、自分が所有するファイルや制御アクセス権を持つファイルに、特別なアクセス制御エントリ (ACE) を追加します。ただし、監査ログ・ファイルはシステム全体を対象としたメカニズムなので、サイトのセキュリティ管理者がファイル監査の使用を管理することをお勧めします。ユーザは自分が制御権を持つファイルに監査用 ACE を追加できますが、ファイルの監査機能はセキュリティ管理者がシステム・レベルで有効にする必要があります。

たとえば、RWOODS というユーザとセキュリティ管理者が CONFIDREVIEW.MEM という極秘ファイルへのアクセスを検出する必要があることで同意した場合、RWOODS は次のようにして CONFIDREVIEW.MEM ファイルの既存の ACL にエントリを追加できます。

```
$ SET SECURITY/ACL=(AUDIT=SECURITY,ACCESS=READ+WRITE-  
_ $ +DELETE+CONTROL+FAILURE+SUCCESS) CONFIDREVIEW.MEM
```

RWOODS がセキュリティ監査エントリを追加したら、セキュリティ管理者はアクセス行為を記録できるようにファイル・アクセスの監査機能を有効にします。ファイル・アクセスの監査の詳細については、3.10.3.1 項を参照してください。

1 つのファイルにアクセス違反があれば、多くの場合は他のファイルにもアクセス上の問題が発生しています。したがって、セキュリティ管理者はセキュリティ監査用 ACE を持つすべての重要ファイルに対するアクセスを監視する必要があります。重要なファイルに望ましくないアクセスがあった場合、セキュリティ管理者は直ちに処置を講じなければなりません。

### 3.10.3 セキュリティ管理者への監査の有効化の依頼

セキュリティ管理者は、セキュリティ関連イベントが発生するたびに、システム・セキュリティ監査ログ・ファイルに監査メッセージを送信するか、セキュリティ・オペレータ・ターミナルとして有効になっているターミナルにアラームを送信するように、オペレーティング・システムに指示できます。たとえば、書き込みアクセスが禁止されている 1 つ以上のファイルをセキュリティ管理者が指定したとします。その場合、それらのファイルへのアクセスが発生したことを示す監査メッセージを送信させることができます。

#### 3.10.3.1 ファイル・アクセスの監査

アカウントに対する侵入行為があったと考えられる場合、セキュリティ管理者はすべてのファイル・アクセスに対する監査機能を一時的に有効にできます。また、監査を有効にして、ファイルに対する読み込みアクセスを監視することにより、ファイルを閲覧したユーザを見つけることもできます。

たとえば、セキュリティ監査用 ACE ( 3.10.2 項を参照) を持つ CONFIDREVIEW.MEM というファイルを監査するとします。ABADGUY というユーザが CONFIDREVIEW.MEM ファイルにアクセスするときに削除アクセス権を持っていると、次のような監査レコードがシステム・セキュリティ監査ログ・ファイルに書き込まれます。

```
%%%%%%%%%% OPCOM 7-DEC-2001 07:21:11.10 %%%%%%%%%%%
Message

from user AUDIT$SERVER on BOSTON
Security audit (SECURITY)

on BOSTON, system id: 19424
Auditable event:      Attempted

file access
Event time:           7-DEC-2001 07:21:10.84
PID:

                                23E00231
Username:              ABADGUY
Image

name:                  BOSTON$DUA0: [SYS0.SYSCOMMON.] [SYSEXE]DELETE.EXE
Object

name:                  _BOSTON$DUA1: [RWOODS]CONFIDREVIEW.MEM;1
Object

type:                  file
Access requested:      DELETE
Status:

                                %SYSTEM-S-NORMAL, normal successful completion
Privileges

used:                  SYSPRV
```

この監査メッセージには、不正にアクセスしたユーザの名前、アクセス方法 ([SYSEXE]DELETE.EXE プログラムを使用した正常な削除操作)、アクセス時刻 (午前 7 時 21 分)、およびファイルへのアクセスに使われた特権 (SYSPRV) が示されています。セキュリティ管理者は、これらの情報に基づいて処置を講じます。

いずれかのファイルがアクセスされ、そのファイルの ACL の監査エントリ ( 3.10.2 項を参照) に指定されている条件が満たされるたびに、セキュリティ監査メッセージが監査ログ・ファイルに書き込まれます。CONFIDREVIEW.MEM ファイルへのアクセスがあると、セキュリティ監査機能によって保護されているシステム上の他のファイルにアクセスがあった場合と同じように、セキュリティ監査ログ・ファイルに監査レコードを書き込む指示が出されます。

監査機能を導入した後は、セキュリティ管理者とともに、新たな侵入行為が発生していないかどうかを定期的にチェックします。

### 3.10.3.2 監査対象イベントの追加

セキュリティ管理者は、ファイルの監査以外にも、発生時に特別な注意を払う必要があるイベントのタイプを選択できます。監査やアラームを起動するイベントには、たとえば次のようなものがあります。

セキュリティ監査やアラームを発生させるイベント	
ログイン、ログアウト、ログインの失敗、侵入行為、ボリュームのマウントおよびディスマウント	システム・パスワードの変更、ユーザ・パスワードの変更、システム時間の変更、システム登録ファイルの変更、ネットワーク代理ファイルの変更、ライト・データベースの変更、SYSGEN パラメータの変更
論理リンクの接続と終了	SET AUDIT コマンドの実行、NCP コマンドの実行
特定の保護オブジェクトの作成と削除	イメージのインストール
特定の保護オブジェクトに対する特定のタイプのアクセスおよびアクセス解除	保護オブジェクトの ACL によって要求されたアクセス・イベント
特権または識別子の使用 (成功または失敗)	プロセス制御システム・サービス (\$CREPRC や \$DELPRC など) の使用

## 3.11 システム・セキュリティを損なわないログアウト

セッションからログアウトすると、システム資源が節約され、ファイルも保護されます。ターミナルをオンライン状態のままにしておくことは、部内者による侵入行為の最大の原因になります。ターミナルをオンライン状態にしたままでオフィスを開放することは、自分のパスワードや特権を人に与え、自分のファイルやグループの他のメンバのファイルが無防備にしておくことと同じです。無防備な状態のアカウントからアクセス可能なすべてのファイルを、誰でも簡単に転送できてしまいます。悪意を持つ内部の人間が、ユーザのファイルやそのユーザが書き込みアクセス権を持つ他のファイルを削除したり、ファイル名を変更したりもできます。ユーザに特殊な特権 (特に Files カテゴリや All カテゴリの特権) があれば、悪意を持つユーザは大きな損害を与えることができます。

たとえ短時間でもオフィスを離れるときは、ログアウトするようにします。遠隔ログインを実行した場合は、ログインしたすべてのノードからログアウトする必要があります。以下の各節では、特定のタイプのターミナルやセッションからログアウトする場合のセキュリティに関する考慮事項について説明します。



### 3.11.1 ターミナル画面の消去

ターミナルからログアウトするときは、ユーザ名、ノード名、およびオペレーティング・システムの情報が他人に明らかにならないように、必ず画面を消去することをお勧めします。遠隔ログインの後でログアウトする場合は、戻り先のノード（ローカル・ノード）の名前も表示されます。遠隔ノードの複数のアカウントに（ネットワーク経由で）アクセスした場合は、最後の一連のログアウト・コマンドから、（最も遠隔のノードを除き）すべてのノード名と各ノードでアクセス可能なユーザ名が明らかになります。プロンプトやログアウト・メッセージからオペレーティング・システムを見分ける能力があれば、これらの表示内容からオペレーティング・システムもわかってしまいます。

サイトによっては、次のようにして画面上にログアウト・メッセージ以外の情報を残さないようにすることが重要になります。

- VT200 シリーズまたはそれ以降のターミナルを使用している場合は、Set-Up キーを押して、表示されたメニューから DECwindows の「コマンド」メニューの「画面消去」メニュー項目に相当する項目を選択することにより、画面を消去できます。
- VT100 シリーズ・ターミナルを使用している場合は、Set-Up キーを押し、リセットに割り当てられたキー（0 キー）を押してから、Return キーを押します。一時パラメータを残す場合は、Set-Up キーを押し、80/132 カラム切り替えに割り当てられたキー（9 キー）を 2 回押します。

画面が消去されると、画面上端の DCL プロンプトの横にカーソルが表示されます。このプロンプトで DCL の LOGOUT コマンドを入力します。ログアウト後に表示される情報は、次のように LOGOUT コマンドとログアウト完了メッセージだけになります。

```
$ LOGOUT
RDOGWOOD      logged out at 14-AUG-2001 19:39:01.43
```

### 3.11.2 ハードコピー出力の破棄

ハードコピー・ターミナルからログアウトした後は、機密情報の漏れる恐れがあるハードコピー出力をすべて取り除いて、ファイルに保存するか、破棄します。セキュリティ管理者が適切な手順を指示するはずですが、多くのサイトでは、シュレッダや鍵の付いた紙くず入れが使われます。出力を保存する場合も、取り扱いには十分に注意します。

ログアウトする前にシステム障害が発生した場合も、ハードコピー出力を破棄する必要があります。また、システムの初期化中に席を離れる場合は、ターミナルの電源を切っておきます。

### 3.11.3 切断されたプロセスの削除

切断されたプロセスは、一定の時間が経過した後、自動的に削除されます。しかし、次のように切断されたプロセスから直接ログアウトすることで、システム資源を節約できます。

1. DCL の SHOW USERS コマンドを入力して、切断されたジョブが他にあるかどうかを確認します。
2. DCL の CONNECT/LOGOUT コマンドを入力して、現在のプロセスからログアウトします。存在する最後のプロセスに到達するまで、該当する仮想ターミナル(先頭に "VTA" がついたターミナル) にそれぞれ接続します。
3. DCL の LOGOUT コマンドを入力します。

### 3.11.4 ダイアルアップ回線への接続の切断

ログアウトするときにダイアルアップ回線への接続を切断するようにセキュリティ管理者から指示される場合があります。回線をその後すぐに使用する予定がない場合は、LOGOUT コマンドに /HANGUP 修飾子を指定します。/HANGUP 修飾子を指定すると、ログアウトした後、ダイアルアップ回線への接続が自動的に切断されます。

---

#### 注意

---

/HANGUP 修飾子が機能するかどうかは、システム管理者によるモデム回線の設定方法および回線とコンピュータとの接続方法に依存します。回線がターミナル・サーバに接続されている場合、この修飾子は機能しません。

---

ダイアルアップ回線への接続を切断しておけば、接続されたままのアクセス回線が誰かに使用されるのを防げます。切断した回線にアクセスするには、アクセス番号を知っている必要があり、自分でダイアルし直さなければなりません。回線の接続を切断しておくことは、使用するダイアルアップ回線が公共の場合や、自分の使用後に他の人がターミナルを使用する可能性がある場合に、特に重要です。

また、必要となるダイアルアップ回線の数が減るため、資源も節約できます。

### 3.11.5 ターミナルの電源遮断

中～高レベルのセキュリティ要件のサイトでは、ログアウト後にターミナルの電源を切るようにセキュリティ管理者から指示される場合があります。この操作によって、ターミナルの属性が再設定され、メモリ・バッファが消去されます。トロイの木馬プログラムの中には、ハードウェア・フレーム・バッファを使用するものや、最新のターミナルに組み込まれているアンサーバック機能を使用するものがあります。

VAX システムでは、C2 環境で作業するユーザは自分のターミナルの電源を切らなければなりません。C2 は、米国政府によるオペレーティング・システムのセキュリティ認定レベルの 1 つです。C2 の要件については、付録 C で説明します。

### 3.12 システム・セキュリティへの貢献のためのチェックリスト

セキュリティ機能は、セキュリティ管理者がすべてのユーザについて要件としてインプリメントするものですが、この章ではユーザがシステム・セキュリティに貢献する方法について説明しました。次のリストは、ユーザがセキュリティのために自主的に行う作業をまとめたものです。

- 3.1 節のガイドラインに従って、安全なパスワードを選択します。
- 自分のパスワードを保護し、頻繁に変更します。
- ログインのたびに最終ログイン・メッセージをチェックし、説明のつかないメッセージについてはセキュリティ管理者に報告します (3.4.3 項)。
- 可能であれば、代理ログインを使用します (3.4 節)。
- ターミナルや職場を離れるときは、ログアウトして戸締りをします (3.11 節)。
- ダイヤルアップ回線に対する最後の LOGOUT コマンドに /HANGUP 識別子を指定します (3.11.4 項)。
- ターミナルのハードコピー出力を適切に破棄します (3.11.2 項)。
- ビデオ・ターミナルの画面を消去するか、ターミナルの電源を切って画面に表示された情報を消します (3.6.2 項および 3.11.1 項)。
- バックアップ媒体を施錠して保管します。媒体を手に入れば、誰でもテープやディスクに保存された情報にアクセスできます。
- セキュリティ管理者に依頼して、不適切なアクセスがあったと思われる保護オブジェクト(ファイルなど)に対するセキュリティ監査機能を有効にします (3.10.3.1 項)。



---

## データの保護

この章では、第 2 章で紹介したセキュリティ設計について、さらに詳しく説明します。OpenVMS オペレーティング・システムがユーザ・プロセスおよびアプリケーションによる保護オブジェクトへのアクセスをどのように制御するかを説明します。

簡単に言うと、Open VMS オペレーティング・システムは、共用可能な情報を含むすべてのオブジェクトへのアクセスを制御します。これらのオブジェクトを保護オブジェクトと呼びます。デバイス、ボリューム、論理名テーブル、ファイル、COMMON・イベント・フラグ・クラスタ、グループ・グローバル・セクション、システム・グローバル・セクション、資源ドメイン、キュー、ケーパビリティ、およびセキュリティ・クラスがこのカテゴリに入ります。アクセスするプロセスは、アクセス資格情報をライト識別子という形で持っています。一方、保護オブジェクトはすべて、当該オブジェクトに指定の方法でアクセスする権限を持つユーザを指定する一連のアクセス要件が設定されています。

この章では、次の内容を説明します。

- システムがオブジェクトに対するアクセス権を定義するためにプロセスに割り当てる識別情報のタイプ (4.1 節)。
- オブジェクトに設定できるアクセス制御 (4.2 節)。
- OpenVMS オペレーティング・システムがアクセス要求を処理する方法 (4.3 節)。
- オブジェクトへのアクセスの制御方法 (4.4 節, 4.5 節, 4.6 節, および 4.7 節)。

第 5 章では、保護オブジェクトのクラスごとの特徴を説明します。

### 4.1 ユーザのセキュリティ・プロファイルの内容

ユーザ・プロセスまたはアプリケーションのプロファイルには、以下の要素が含まれています。

- ユーザを識別するためのユーザ識別コード (UIC)
- プロセスが保持するライト識別子
- (あれば) 特権

### 4.1.1 スレッド別セキュリティ

OpenVMS Alpha バージョン 7.2 には、スレッド・レベルのセキュリティが実装されています。この機能はスレッド別セキュリティと呼ばれ、これによって、マルチスレッド・プロセスの実行スレッドごとに、プロセス内の他のスレッドのセキュリティ・プロファイルに影響を与えることなく、独立したセキュリティ・プロファイルを使用できます。

セキュリティ・プロファイルの情報は、以前はプロセス・レベルの各種データ構造体やデータ・セルに分散していましたが、現在は PSB (ペルソナ・セキュリティ・ブロック) と呼ばれる単一のデータ構造体に格納されており、それが個々の実行スレッドにバインドされています。これに合わせて、OpenVMS 内の関連する参照も参照先が変更されています。システム内のあらゆるプロセスに、プロセスのナチュラル・ペルソナとなる PSB が少なくとも 1 つあります。ナチュラル・ペルソナは、プロセスの作成時に作成されます。

スレッド・マネージャ (たとえば、HP POSIX Threads Library に組み込まれているスレッド・マネージャ) とセキュリティ・サブシステムのやり取りにより、スレッドの実行がスケジューリングされている間にプロファイルが自動的に切り替えられます。

### 4.1.2 ペルソナ・セキュリティ・ブロック (PSB) データ構造体

ユーザのセキュリティ・プロファイル (特権、権限、および識別情報) は、プロセス・レベルからユーザ・スレッド・レベルに移行しています。これまで複数のデータ構造体 (アクセス・ライト・ブロック (ARB)、プロセス制御ブロック (PCB)、プロセス・ヘッダ・ディスクリプタ (PHD)、ジョブ情報ブロック (JIB)、制御 (CTL) リージョン・フィールド) に格納されていたセキュリティ情報は、ペルソナ・セキュリティ・ブロック (PSB) という新しいデータ構造体に移され、これに合わせて参照先がすべて変更されています。これらのデータ構造体に含まれるフィールドの一部は、現在の OpenVMS では使用されていません。該当するフィールドは、使用廃止されたものと見なされています。『*OpenVMS* リリース・ノート』の「廃止されたデータ・セルとセキュリティ情報の新しい場所」という表を参照してください。

それぞれのプロセスに、そのプロセスに割り当てられているすべてのペルソナ・ブロックのアドレスを格納したペルソナ配列があります。

新しいペルソナ・ブロック (PSB) には、以下の情報が格納されています。

- UIC
- ペルソナ、およびシステム・ライト・チェーン
- 永続的な、許可された、有効な特権
- アカウント名

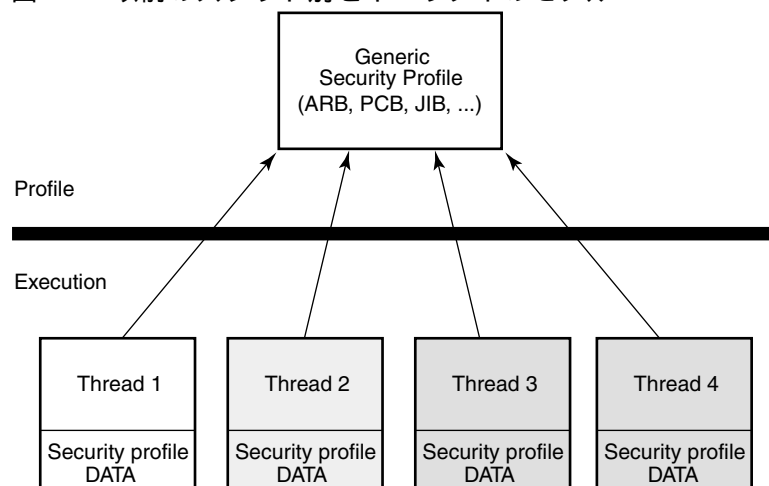
- ユーザ名
- 監査フラグとカウンタ

カーネル・スレッド・ブロック (KTB) は、現在アクティブなスレッドのペルソナ・ブロックを指します。

### 4.1.3 以前のセキュリティ・モデル

OpenVMS の以前のバージョンでは、ユーザのセキュリティ・プロファイルを構成する情報がプロセス・レベルでバインドされ、プロセス内のすべての実行スレッドで共用されていました。この関係を図 4-1 に示します。

図 4-1: 以前のスレッド別セキュリティのモデル



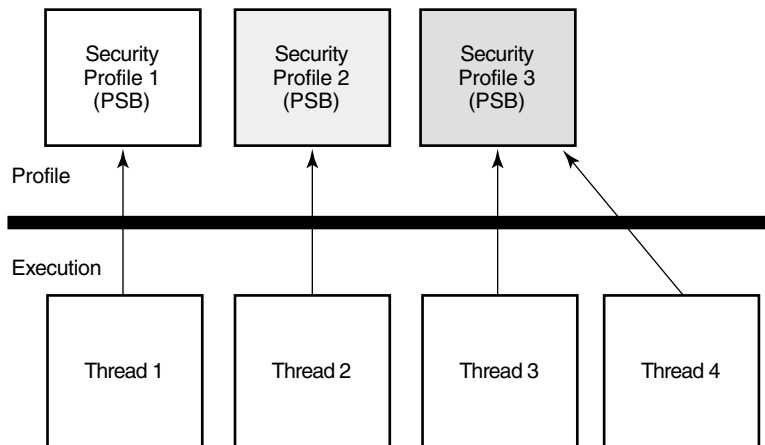
VM-0997A-AI

スレッド間でどのようにプロファイルの管理を行うかによって、あるスレッドがセキュリティ・プロファイルに加えた変更が他のスレッドから見える可能性があります。

### 4.1.4 スレッド別セキュリティのモデル

OpenVMS バージョン 7.2 では、各実行スレッドが他のスレッドとセキュリティ・プロファイルを共用することもできますが、そのスレッド専用のセキュリティ・プロファイルを持つこともできます。これらの関係を図 4-2 に示します。

図 4-2: スレッド別セキュリティ・プロファイルのモデル



VM-0998A-AI

以前のモデルと同じように、共用のプロファイルに加えた変更はプロファイルを共用するすべてのスレッドから見える可能性があります。一方、スレッド専用のセキュリティ・プロファイルに加えた変更は、特定のスレッドにのみ適用されます。

#### 4.1.5 ユーザ識別コード (UIC)

サブジェクトのセキュリティ・プロファイルの最初の要素は、ユーザ識別コード (UIC) です。UIC から、ユーザが属するシステム・グループと、そのグループ内でユーザを一意に識別するための情報が得られます。

##### 4.1.5.1 UIC の形式

UIC を指定するときは必ず大括弧で囲みますが、形式にはいくつかの種類があります。有効な形式を以下に示します。

- 英数字形式の UIC は、メンバ名と (必要に応じて) グループ名で構成されます。

[メンバ]

または

[グループ,メンバ]

グループ名とメンバ名には、それぞれ最大 31 文字の英数字 (そのうち少なくとも 1 文字は英字) で構成することができます。これらの名前には、大文字と小文字の A ~ Z, ドル記号 (\$), アンダースコア (\_), および数字の 0 ~ 9 で構成できます。

- 数値形式の UIC は、グループ番号とメンバ番号で構成されます。

[グループ,メンバ]



グループ番号には 1 ~ 37776 の 8 進数を指定し、メンバ番号には 0 ~ 177776 の 8 進数を指定します。グループ番号とメンバ番号を指定するときは、先頭のゼロを省略できます。グループ 1 とグループ 300 ~ 377 は、HP によって予約されています。

次の表に、適切な形式で指定した UIC の例を示します。

UIC のタイプ	例	意味
英数字形式	[USER,FRED]	グループ名が USER で、メンバ名が FRED。
	[EXEC,JONES]	グループ名が EXEC で、メンバ名が JONES。
	[JONES]	グループ名が EXEC で、メンバ名が JONES。
数値形式	[200,10]	グループ番号が 200 で、メンバ番号が 10。
	[3777,3777]	グループ番号が 3777 で、メンバ番号が 3777。

JONES というメンバ名を持てるユーザは 1 人だけなので、JONES は EXEC グループに属する必要があります。

#### 4.1.5.2 UIC 作成に関するガイドライン

UIC を恣意的に割り当てることはできません。UIC を作成するセキュリティ管理者は、以下のガイドラインを守る必要があります。

- メンバ名は、システムの各メンバに固有のもでなければなりません。
- 1 人のメンバが複数の UIC グループに属することはできません。

以下のガイドラインは、システムが UIC をグループ番号とメンバ番号を表す 32 ビットの値に変換するので必要となります。32 ビットのうち、上位 16 ビットがグループ番号、下位 16 ビットがメンバ番号になります。オペレーティング・システムは、[J\_JONES] のような英数字形式の UIC を変換するとき、英数字形式の UIC のメンバの部分の数値形式の UIC のグループとメンバの部分の両方に等しいものと見なします。この結果得られる 32 ビットの数値形式の UIC は、ライト・データベース（識別子、識別子の属性、および識別子の保持者を格納するファイル）に保存されます。たとえば、JONES というメンバに対応する数値形式の UIC は 1 つしか存在し得ないので、同じシステム上で [GROUP1,JONES] という UIC と [GROUP2,JONES] という UIC を作成することはできません。通常、英数字形式の UIC のメンバ名は、それに対応するログイン・ユーザ名と同じです。

4.1.5.3 プロセスによる UIC の取得

ユーザがシステムにログインすると、システム・ユーザ登録ファイル (SYSUAF.DAT) のユーザ登録 (UAF) レコードからユーザの UIC がコピーされ、ユーザのプロセスに割り当てられます。これは、そのプロセスが存続する間の識別情報になります。

デフォルトでは、独立プロセス (DCL の SUBMIT コマンドまたは RUN コマンドで作成されたもの) とサブプロセス (DCL の SPAWN コマンドで作成されたもの) の UIC は、プロセスの作成者と同じ UIC になります。IMPERSONATE 特権を持つユーザは、(RUN コマンドに /UIC 修飾子を指定することにより) 異なる UIC を持つ独立プロセスを作成できます。

4.1.6 ライト識別子

サブジェクトのセキュリティ・プロファイルの 2 番目の要素は、一連のライト識別子です。

ライト識別子は、個々のユーザまたはユーザ・グループを表します。セキュリティ管理者は、登録ユーティリティ (AUTHORIZE) を使用して識別子の作成と削除、およびこれらの識別子のユーザへの割り当てを行えます。ユーザは必要な期間のみ特定の識別子を保持するので、ライト識別子によるユーザ・グループの識別は一時的な方法です。

4.1.6.1 識別子のタイプ

OpenVMS オペレーティング・システムでは、複数のライト識別子のタイプをサポートしています。アクセス制御に最も一般的な使用される識別子を表 4-1 に示します。

表 4-1: 主なライト識別子のタイプ

タイプ	説明	形式	例
環境識別子	ユーザが最初にシステムにログインしたときの情報に基づいてユーザを分類します。	システムによって自動的に生成される英数字文字列です。詳細については、3.4 節を参照してください。	BATCH , NETWORK , INTERACTIVE , LOCAL , DIALUP , REMOTE

表 4-1: 主なライト識別子のタイプ (続き)

タイプ	説明	形式	例
汎用識別子	セキュリティ管理者が定義します。	1 ~ 31 文字の英数字文字列 (そのうち少なくとも 1 文字は英字) です。有効な文字は、数字の 0 ~ 9、大文字と小文字の A ~ Z、ドル記号 (\$)、およびアンダースコア (_) です。	SALES , PERSONNEL , DATA_ENTRY , RESERVE_DESK
UIC 識別子	システムのユーザを一意に識別し、そのユーザが属するグループを定義するユーザ識別コード (UIC) に基づきます。	英数字形式の UIC を使用します。大括弧は付けなくても構いません。有効な文字は、汎用識別子と同じです。	[GROUP1,JONES] , [JONES] , GROUP1 , JONES
機能識別子	アプリケーションが定義します。	汎用識別子と同じです。詳細については、 『 <i>HP OpenVMS Programming Concepts Manual</i> 』を参照してください。	DBM\$MOD_SCHEMA

表 4-1 に挙げた識別子の他に、SYS\$SYSTEM の STARTUP.COM システム・スタートアップ・プロシージャが SYS\$NODE\_ノード名という形式のシステム・ノード識別子を作成します。

#### 4.1.6.2 プロセス・ライト・リストとシステム・ライト・リスト

ユーザのプロセスには、そのプロセスに割り当てられたすべての識別子を含んだライト・リストが関連付けられています。また、システムのすべてのユーザが共用するシステム・ライト・リストもあります。システム管理者またはシステム・ソフトウェアが、システムに現在ログインしているすべてのユーザに割り当てられる識別子をシステム・ライト・リストに割り当てます。

#### 4.1.6.3 プロセスのライト識別子の表示

現在のプロセスに割り当てられている識別子は、次のように SHOW PROCESS コマンドを使用して表示できます。

```
$ SHOW PROCESS/ALL
25-JUN-2001 15:23:18.08   User: GREG                Process ID: 34200094
                           Node: ACCOUNTS            Process name: "_TWA2:"

Terminal:                TWA2:
User Identifier:         [DOC,GREG]                [1]

Base priority:           4
Default file spec:      WORK1:[GREG.FISCAL_91]

Devices allocated:      ACCOUNTS$TWA2:

Process Quotas::
Process rights:
  INTERACTIVE            [2]

  LOCAL                  [3]

  SALES                  [4]

  MINDCRIME              resource                [5]

System rights:
  SYS$NODE_ACCOUNTS     [6]
```

上記の SHOW PROCESS コマンドの出力には、次の 3 種類の識別子が表示されています。

1. Greg というユーザが DOC グループのメンバであることを示す UIC 識別子
2. Greg が会話型ユーザであることを示す環境識別子
3. Greg がローカルでログインしていることを示す環境識別子
4. Greg が SALES グループのメンバでもあることを示す汎用識別子
5. Greg が Resource 属性を含む MINDCRIME 識別子を持っていて、この識別子にディスク容量を割り当てることができることを示す汎用識別子
6. Greg が ACCOUNTS ノードから作業していることを示す環境識別子

#### 4.1.6.4 監査証跡に現れるライト識別子

プロセスのライト識別子は、監査レコードにも現れます。セキュリティ管理者がオブジェクトへのアクセスを監査するようにオペレーティング・システムを設定すると、オブジェクトにアクセスしたユーザとそのアクセスの日時を記録したレコードが生成されます。単独の監査レコードから十分な情報が得られることはま

れですが、長期にわたって蓄積されたレコードを追跡すると、何らかの活動のパターンが浮かび上がることがあります。

次の監査レコードは、Greg がファイルの削除を試みたが、MINDCRIME 識別子を持っているために失敗したことを示しています。93\_FORECAST.DAT ファイルには、MINDCRIME 識別子を有するプロセスによるアクセスを禁止する ACE が設定されています。これを示すのが、"Event information"、"Matching ACE"、および "Status" の各行です。

Message from user AUDIT\$SERVER on FNORD  
Security alarm

(SECURITY) and security audit (SECURITY) on ACCOUNTS,

system id: 19662

Auditable event:

Object deletion

Event information: file deletion

request (IO\$\_DELETE)

Event time: 24-APR-2001 13:17:24.59

PID:

34200094

Process name: \_TWA2:

Username:

GREG

Process owner: [DOC,GREG]

Terminal

name: TWA2:

Image name: DSA2264:[SYS51.SYSCOMMON.][SYSEXEC]DELETE.EXE

Object

class name: FILE

Object owner: [SYSTEM]

Object

protection: SYSTEM:RWEDC, OWNER:RWEDC, GROUP:RE, WORLD:RE

File

name: \_DSA2200:[GREG]93\_FORECAST.DAT;1

File ID:

(17481,6299,1)

Access requested: DELETE

Matching

ACE: (IDENTIFIER=MINDCRIME,ACCESS=NONE)

Sequence

key: 00008A41

```
Status:                %SYSTEM-F-NOPRIV,  
  
no privilege for  
                        attempted operation
```

#### 4.1.7 特権

サブジェクトのセキュリティ・プロファイルの3番目の(省略可能な)要素は、一連の特権です。

特権を持つことにより、通常は拒否されるシステム機能を使用または実行できるようになります。セキュリティ管理者は、特別な事情で既存の保護登録情報を変更せずにユーザに必要なタスクを実行させたいときに、そのユーザに特権を与えることができます。

特権のタイプによって、実行できるタスクは異なります。たとえば、ネットワーク経由のメール送受信を可能にする NETMBX と TMPMBX のように、通常のネットワーク操作を実行するための特権もあります。しかし、SYSNAM のように、システムの動作を左右する能力を与える特権もあります。SYSNAM 特権を有するユーザは、システム論理名テーブルを変更できます。

ユーザの特権は、ユーザの UAF レコードに 64 ビットの特権マスクとして記録されます。ユーザがシステムにログインすると、ユーザの特権ベクタがサブジェクト(プロセス)のセキュリティ・プロファイルに保存されます。

ユーザに許可された特権を、DCL の SET PROCESS/PRIVILEGES コマンドを使用して有効または無効にすることにより、ユーザが実行するイメージに適用可能な特権を制御できます。例 4-1 は、Puterman というユーザに多くの特権が与えられており、必要に応じてそれらを使用できるけれども、Puterman のプロセスがデフォルトでは NETMBX と TMPMBX の2つの特権のみが有効な状態で実行されることを示しています。

##### 例 4-1: プロセスの許可された特権とデフォルト特権

---

```
$ SHOW PROCESS/PRIVILEGE  
8-OCT-2001 16:58:58.77  
User: PUTERMAN Process ID: 27E00496  
Node:  
  
FNORD      Process name: "Hobbit"Authorized privileges:  
  
ACNT  
  
ALLSPOOL  ALTPRI  AUDIT   BUGCHK   BYPASS  CMEXEC   CMKRNL  
DIAGNOSE  
  
DOWNGRADE EXQUOTA GROUP   GRPNAM   GRPPRV  IMPERSONATE IMPORT  
LOG_IO
```

#### 例 4-1: プロセスの許可された特権とデフォルト特権 (続き)

---

MOUNT	NETMBX	OPER	PFNMAP	PHY_IO	PRMCEB	PRMGBL
PRMMBX						
PSWAPM	READALL	SECURITY	SETPRV	SHARE	SHMEM	SYSGBL
SYSLCK						
SYSNAM	SYSPRV	TMPMBX	UPGRADE	VOLPRO	WORLD	

Process

privileges:

NETMBX	may create network device
TMPMBX	
	may create temporary mailbox

---

Puterman は、必要に応じて許可された特定の特権を有効にできます。たとえば、スプールされたデバイスを割り当てるには ALLSPOOL 特権が必要であり、論理入出力操作を実行するには LOG\_IO 特権が必要です。

## 4.2 オブジェクトのセキュリティ・プロファイル

### 4.2.1 保護オブジェクトの定義

OpenVMS オペレーティング・システムで保護が必要となるオブジェクトは、いずれも情報の格納や受け取りに使用される受動的な格納場所です。これらのオブジェクトを保護するのは、オブジェクトにアクセスが可能になったユーザはそのオブジェクト内の情報にもアクセスできるためです。保護オブジェクトには、次のようなものがあります。

- メモリやストレージ・デバイス上のファイル
- ハードウェア・デバイスや仮想デバイス
- コモン・イベント・フラグ・クラスタや論理名テーブルなどのデータ構造

OpenVMS で保護されるオブジェクトのクラスの一覧は、4.2.5 項にあります。各クラスの詳細については、第 5 章を参照してください。

### 4.2.2 オブジェクトのプロファイルの内容

オブジェクトのセキュリティ要素が、オブジェクトのセキュリティ・プロファイルを構成します。オブジェクトのセキュリティ・プロファイルには、以下の情報が格納されています。

- オブジェクトの所有者。システムが保護コードを解釈するときに、この要素が使用されます。
- システム、所有者、グループ、ワールドの各カテゴリに基づいてオブジェクトへのアクセスを定義する保護コード。この保護コードによって、さまざまなカテゴリのユーザが制御されます。
- 個々のユーザまたはユーザ・グループごとにオブジェクトへのアクセスを制御するアクセス制御リスト (ACL)。

ファイル以外の新しいオブジェクトは、システムによって提供されるテンプレート・プロファイルのセキュリティ要素を継承します。サイトのセキュリティ管理者は、これらのセキュリティ要素を変更できます。ファイルにはさらに複雑な継承メカニズムがあり、新しいオブジェクトのセキュリティ要素をさらに細かく制御できます。いずれの場合も、オブジェクトの作成時にセキュリティ要素を割り当てることにより、オペレーティング・システムのデフォルト設定の使用を避けることができます。

この節では、保護コードと ACL の概要を示します。4.4 節と 4.5 節では、これらの保護メカニズムについてさらに詳しく説明します。個々のオブジェクト・クラスの詳細については、第 5 章を参照してください。

#### 4.2.2.1 所有者

オブジェクトのセキュリティ・プロファイルの最初の要素は、オブジェクト所有者の UIC です。

ほとんどの場合、オブジェクトを作成したユーザがそのオブジェクトの所有者になります。オブジェクトの所有者は、所有するオブジェクトのセキュリティ・プロファイルを変更できます。システムはユーザの UIC をオブジェクトに自動的に割り当て、それに基づいてアクセスを制御します。

所有権に関する規則には、いくつかの例外があります。資源識別子によって所有されるファイルには、UIC がありません。資源識別子が所有するディレクトリにユーザがファイルを作成すると、そのファイルは (ファイルを作成したユーザではなく) 資源識別子によって所有されます (5.4.5 項を参照)。各オブジェクト・クラスの所有権規則については、第 5 章を参照してください。

ファイルを除くオブジェクトの所有者は、4.2.4 項で説明するように、SET SECURITY/OWNER コマンドを使用して所有権を他のユーザに再割り当てできます。ファイルの所有者を変更するには、通常、特権が必要です (5.4.2 項を参照)。

#### 4.2.2.2 保護コード

オブジェクトのセキュリティ・プロファイルの 2 番目の要素は、オブジェクトの保護コードです。



システムは個々の新しいオブジェクトに保護コードを自動的に割り当てます。オブジェクトに関連付けられた保護コードは、ユーザの UIC と所有者の UIC の関係に基づいて、ユーザに許可するアクセスのタイプを決定します。ファイルと擬似ターミナル (FT) デバイスを除き、保護オブジェクトに割り当てられるコードは、そのクラスのテンプレート・プロファイルに基づきます。ファイルの保護コードは、5.4 節で説明するように、別のソースに基づきます。

通常、オブジェクトが (a) 所有者のみ、(b) システム上のすべてのユーザ、または (c) 特定の UIC ベースのユーザ・グループによってアクセスされる場合は、保護コードに基づいてオブジェクトが保護されます。UIC グループ外の特定のユーザ・グループにアクセス権を付与したいけれども、システム上のすべてのユーザには付与したくない場合には、ACL を追加する必要があります (4.2.2.3 項を参照)。

#### 保護コードの解釈

保護コードでは、(a) 所有者、(b) 所有者と同じグループ UIC を共用するユーザ (グループ・カテゴリ)、(c) システム上のすべてのユーザ (ワールド・カテゴリ)、(d) システム特権またはシステム権限を持つユーザ (システム・カテゴリ) の、計 4 種類のユーザに対してアクセス権が定義されます。保護コードには、必ずシステム・カテゴリ (S)、所有者 (O)、グループ (G)、ワールド (W) の順でアクセス権が記述されます。構文は次のとおりです。

[ユーザ・カテゴリ: 許可されるアクセス (,ユーザ・カテゴリ: 許可されるアクセス,...)]

オペレーティング・システムは、保護オブジェクトの使用に対する要求を処理するときに、ユーザの UIC とオブジェクトの所有者の UIC を比較します。ユーザの UIC がオブジェクトの所有者の UIC と同じ場合は、所有者保護フィールドのアクセス権がユーザに与えられます。UIC が一致しない場合は、他のユーザ・カテゴリとの比較が行われます。グループ・フィールドを比較して、同じグループのメンバかどうかを判定します。また、UIC のグループ番号を調べて、ユーザがシステム・カテゴリに属するかどうかを判定します。ワールド・カテゴリはすべてのユーザに適用されます。

たとえば、[14,1] という UIC を持つ Jones というユーザが、UIC [14,5] によって所有されているファイルを読み込むとします。Jones は同じグループ (14) に属するので、このファイルに対するアクセス権が与えられる可能性があります。最終的な決定は、保護コードに指定されているアクセス権によります。

保護コードの解釈方法と作成方法の詳細については、4.5 節を参照してください。

#### 4.2.2.3 アクセス制御リスト (ACL)

オブジェクトのセキュリティ・プロファイルの 3 番目の (省略可能な) 要素は、オブジェクトのアクセス制御リストです。

アクセス制御リスト (ACL) は、ユーザまたはユーザ・グループが特定の保護オブジェクト (ファイル、ディレクトリ、デバイスなど) に対して持つアクセス権を定義するエントリの集合です。

ACL は、オブジェクトの作成と同時に作成される場合 (デフォルト) と、セキュリティ管理者が作成する場合と、(ユーザが制御アクセス権を持つオブジェクトに関して) ユーザが作成する場合があります (4.6.2 項を参照)。

セキュリティ管理者はデフォルトの ACL を設定できるので、ユーザによっては自分のオブジェクトに ACL があることに気づかず、ACL をまったく変更しない場合があります。自分のファイルに ACL があるかどうかは、DCL の DIRECTORY/SECURITY コマンドまたは SHOW SECURITY コマンドを使用して確認できます。ユーザが自分の ACL の作成や管理に積極的に関わる場合もあります。

ACL を必ずしも使用する必要はありません。ACL を使用することで、アクセスを許可する対象となるユーザとアクセスの種類を細かく定義できるので、どのようなシステムでもオブジェクトのセキュリティを強化できますが、そのためにはユーザが ACL の作成と管理に時間をかけなければなりません。

ACL の作成および表示には DCL の SET SECURITY コマンドと SHOW SECURITY コマンドを使用しますが、より広範な作業については、アクセス制御リスト・エディタ (ACL エディタ) を使用します。

4.4 節では、ACL とその使用方法についてさらに説明を加えます。

### 4.2.3 セキュリティ・プロファイルの表示

保護オブジェクトのセキュリティ・プロファイルを表示するには、DCL の SHOW SECURITY コマンドを使用します。たとえば、次のコマンドは 93\_FORECAST.TXT というファイルのセキュリティ情報を要求します。

```
$ SHOW SECURITY 93_FORECAST.TXT
```

```
WORK_DISK$:[GREG]93_FORECAST.TXT;1 object of class FILE
  Owner: [ACCOUNTING,GREG]
  Protection: (System: RWED, Owner: RWED, Group: RE, World)
  Access Control List: <empty>
```

表示結果から、93\_FORECAST.TXT が Greg というユーザによって所有されていることがわかります。また、このファイルの保護コードも表示されています。保護コードにより、システム・ユーザと所有者に対して読み込み、書き込み、実行、削除の各アクセス権が与えられています。また、グループ・ユーザに対しては読み込みと実行のアクセス権が与えられ、ワールド・ユーザに対してはアクセス権が与えられていません。詳細については、4.5 節を参照してください。このファイルには、ACL はまだ設定されていません。

#### 4.2.4 セキュリティ・プロファイルの変更

SET SECURITY コマンドを使用して、保護オブジェクトの所有者、保護コード、ACL に対して新しい値を指定したり、オブジェクト間でプロファイルをコピーしたりできます。

たとえば、4.2.3 項に示した SHOW SECURITY の表示結果から、93\_FORECAST.TXT が Greg というユーザによって所有されていることがわかります。このユーザは、所有者としてこのファイルの保護コードを変更できます。変更前の保護コードでは、ワールド・ユーザに対してアクセス権が与えられていません。ここでは、Greg が次のように保護コードを変更して、ワールド・ユーザに読み込みと書き込みの各アクセスを許可します。

```
$ SET SECURITY/PROTECTION=(W:RW) 93_FORECAST.TXT
```

このファイルの新しい保護コードを確認するには、次のように SHOW SECURITY コマンドを使用します。

```
$ SHOW SECURITY 93_FORECAST.TXT
```

```
93_FORECAST.TXT object of class FILE
```

```
Owner: [GREG]
Protection: (System: RWED, Owner: RWED, Group: RE, World: RW)
Access Control List: <empty>
```

プロファイル内の他の要素の変更方法については、4.2.5 項で説明します。保護コードと ACL の詳細については、4.4 節と 4.5 節で説明します。SET SECURITY コマンドと SHOW SECURITY コマンドの詳細については、『*OpenVMS DCL* デictionaryナリ』を参照してください。

#### 4.2.5 オブジェクトのクラスの指定

特定の動作を行い、共通の属性のセットを持つオブジェクトのグループは、クラスに分けられます。ファイル、キュー、およびボリュームがその代表的な例です。表 4-2 に示すように、OpenVMS オペレーティング・システムでは 11 の保護オブジェクトのクラスがサポートされています。

オブジェクトのプロファイルを変更するときは、SET SECURITY コマンドにオブジェクトのクラスを指定する必要があります。指定しなかった場合、オブジェクトがファイルであると見なされます。

たとえば、次のコマンド・シーケンスはオブジェクトのプロファイルを変更しますが、/CLASS 修飾子によって LNM\$GROUP オブジェクトを論理名テーブルとして識別しています。

```
$ SET SECURITY /CLASS=LOGICAL_NAME_TABLE-
_$ /OWNER=ACCOUNTING /PROTECTION=(S:RWCD, O:RWCD, G:R, W:R)-
_$ /ACL=((IDENTIFIER=CHEKOV,ACCESS=CONTROL),-
```

```
_$(IDENTIFIER=WU,ACCESS=READ+WRITE)) LNM$GROUP
```

この SET SECURITY コマンドによって、Accounting グループを論理名テーブルの所有者に設定しています。また、保護コードを変更して、所有者とシステム・ユーザに対して読み込み、書き込み、作成、および削除のアクセス権を許可し、グループ・ユーザとワールド・ユーザに許可するアクセス権を読み込みに限定しています。最後に、ACL を作成して、Chekov というユーザに制御アクセスを許可し、Wu というユーザに読み込みと書き込みのアクセスを許可しています。

変更結果を表示するには、次のように SHOW SECURITY コマンドを使用します。

```
$ SHOW SECURITY LNM$GROUP /CLASS=LOGICAL_NAME_TABLE
```

```
LNM$GROUP object of class LOGICAL_NAME_TABLE
```

```
Owner: [ACCOUNTING]
Protection: (System: RWCD, Owner: RWCD, Group: R, World: R)
Access Control List:
  (IDENTIFIER=[USER,CHEKOV],ACCESS=CONTROL)
  (IDENTIFIER=[USER,WU],ACCESS=READ+WRITE)
```

表 4-2: 保護オブジェクトのクラス

クラス名	定義
ケーパビリティ	システムによってアクセスが制御される資源。現時点で定義されているケーパビリティは、ベクタ・プロセッサだけです。
コモン・イベント・フラグ・クラスタ	プロセス同士が連携してイベント通知を相互に通知できるようにするために、32 個のイベント・フラグをセットにしたもの。
デバイス	プロセッサに接続された周辺機器のクラスの 1 つで、データの受信、格納、または伝送が可能なもの。
ファイル	Files-11 オン・ディスク構造レベル 2 または 5 のファイルおよびディレクトリ。
グループ・グローバル・セクション	同じグループ内のすべてのプロセスが使用できる共用可能なメモリ・セクション。
論理名テーブル	システムまたは特定のグループに関する論理名とその等価名を格納した共用可能なテーブル。
キュー	バッチ、ターミナル、サーバ、またはプリント・ジョブ・キューで処理される一連のジョブ。
資源ドメイン	ロック・マネージャの資源へのアクセスを制御するネームスペース。
セキュリティ・クラス	セキュリティ・クラスのすべてのメンバに関する要素と管理ルーチンを格納するデータ構造。

表 4-2: 保護オブジェクトのクラス (続き)

クラス名	定義
システム・グローバル・セクション	システム内のすべてのプロセスが使用できる共用可能なメモリ・セクション。
ボリューム	ディスクやテープなどの、ODS-2 形式または ODS-5 形式の大容量ストレージ媒体。ボリュームは、ファイルを格納するもので、デバイスにマウントすることができます。

個々のクラスの詳細については、第 5 章を参照してください。

#### 4.2.6 プロファイルの変更に必要なアクセス権

セキュリティ・プロファイルを変更するには、オブジェクトに対する制御アクセス権が必要です。制御アクセス権は、ACL では明示的に与えられるのに対し、保護コードでは所有者カテゴリまたはシステム・カテゴリに属するユーザに対して暗黙のうちに与えられます。制御アクセス権の獲得方法の詳細については、4.6.2 項を参照してください。

### 4.3 システムによる保護オブジェクトへのユーザのアクセス可否の判定

ユーザが保護オブジェクトにアクセスしようとする時、OpenVMS オペレーティング・システムは保護チェック (\$CHKPRO) システム・サービスを呼び出して、ユーザ・プロセスのセキュリティ・プロファイルとオブジェクトのセキュリティ・プロファイルを比較します。保護チェックでは、\$CHKPRO がユーザのセキュリティ・プロファイルと保護オブジェクトのプロファイルを以下の順序で比較します。

1. アクセス制御リスト (ACL) を評価します。

オブジェクトに ACL がある場合、システムはユーザのライト識別子のいずれかと一致するエントリがないかどうか ACL を検索します。一致するアクセス制御エントリ (ACE) が見つかった場合、システムはアクセスを許可または拒否し、ACL のチェックをそこで終了します。

一致する ACE でアクセスが拒否されている場合でも、ユーザは保護コードのシステム・フィールドと所有者フィールド、または特権によってアクセス権を取得できます。ACL に一致する ACE がない場合、システムは保護コードのすべてのフィールドをチェックします。

2. 保護コードを評価します。

ACL でアクセスが許可されず、オブジェクトの所有者の UIC がゼロでない場合、<sup>1</sup>オペレーティング・システムは保護コードを評価します。OpenVMS オペレーティング・システムは、ユーザ識別コード (UIC) とオブジェクトの保護コードの関係に基づいてアクセスを許可または拒否します。

ACL によってアクセスが拒否されている場合、システムは保護コード内の 2 つのフィールド (システム・フィールドと所有者フィールド) を調べて、ユーザにアクセスが許可されているかどうかを判定します。ユーザは、システム・カテゴリまたは所有者カテゴリに属するか、特権を与えられることにより、アクセス権を獲得できます。(同じグループ UIC の) GRPPRV または SYSPRV を持っているユーザは、保護コードのシステム・カテゴリに指定されているアクセス権を与えられます。

3. 特別な特権を確認します。

ACL または保護コードによってアクセスが許可されない場合は、特権が評価されます。

特定のシステム特権を持つユーザには、ACL または保護コードによる保護に関係なくアクセス権が与えられる場合があります。バイパス特権 (BYPASS)、グループ特権 (GRPPRV)、全読み込み特権 (READALL)、およびシステム特権 (SYSPRV) は、その特権保持者がオブジェクトに対して持っているアクセス権を強化します。特権によるアクセス権への影響の詳細については、4.6.1 項を参照してください。

4. アクセス権に対する変更を評価します。

一部のオブジェクト・クラスでは、代替特権に基づいてアクセスが許可されることがあります。たとえば、キュー・オブジェクトでは、オペレータ特権 (OPER) を持つユーザに対してすべてのキューへのフル・アクセスが許可されます。また、論理名テーブル・オブジェクトでは、システム名特権 (SYSNAM) を持つユーザに対してシステム・テーブルへのアクセスが許可されます。

図 4-3 は、OpenVMS オペレーティング・システムがアクセス要求を評価する手順と、アクセスを制御する要素 (ACL、保護コード、特権、およびアクセス権の変更) の相互関係を示した図です。

---

<sup>1</sup> オブジェクトの所有者 UIC がゼロの場合は、保護コードがチェックされません。ACL に識別子用 ACE がない場合に限り、ユーザはオブジェクトに対する制御アクセス権以外のすべてを許可されます。識別子用 ACE がある場合は、ACL または特権によって明示的にアクセス権を与える必要があります。

図 4-3: アクセス要求評価のフローチャート

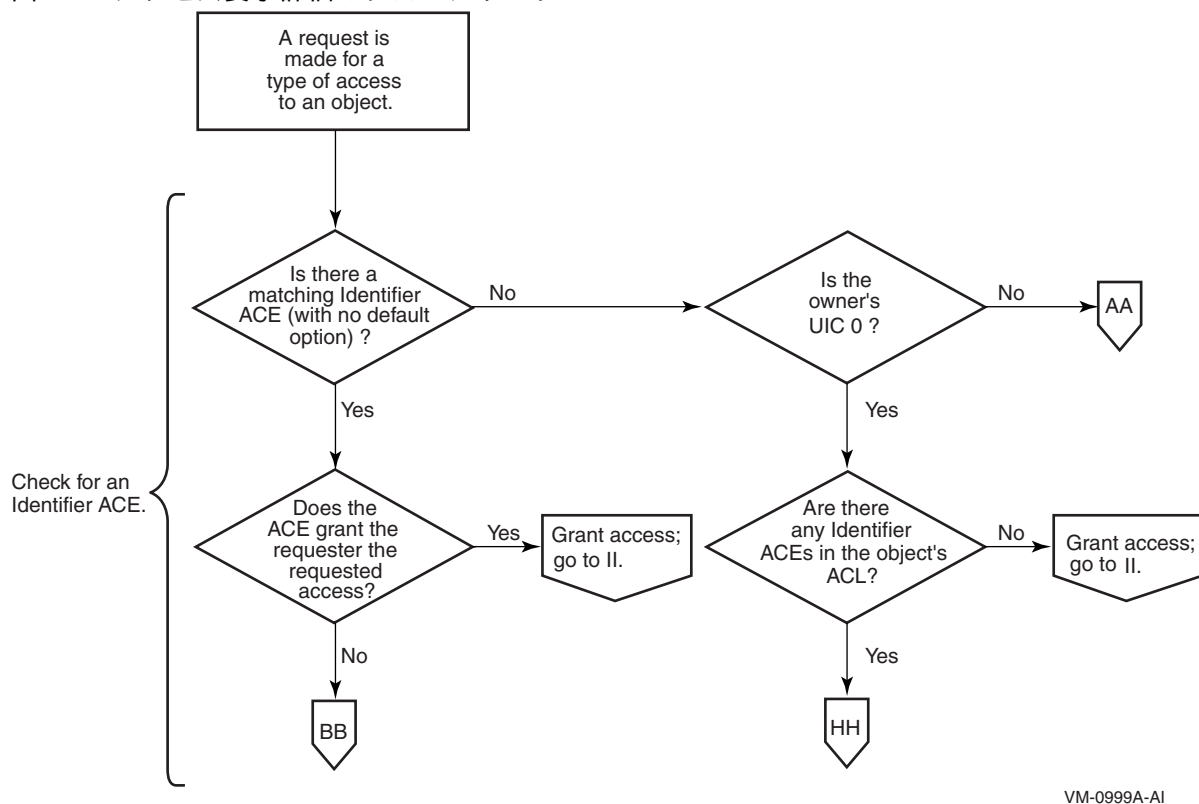
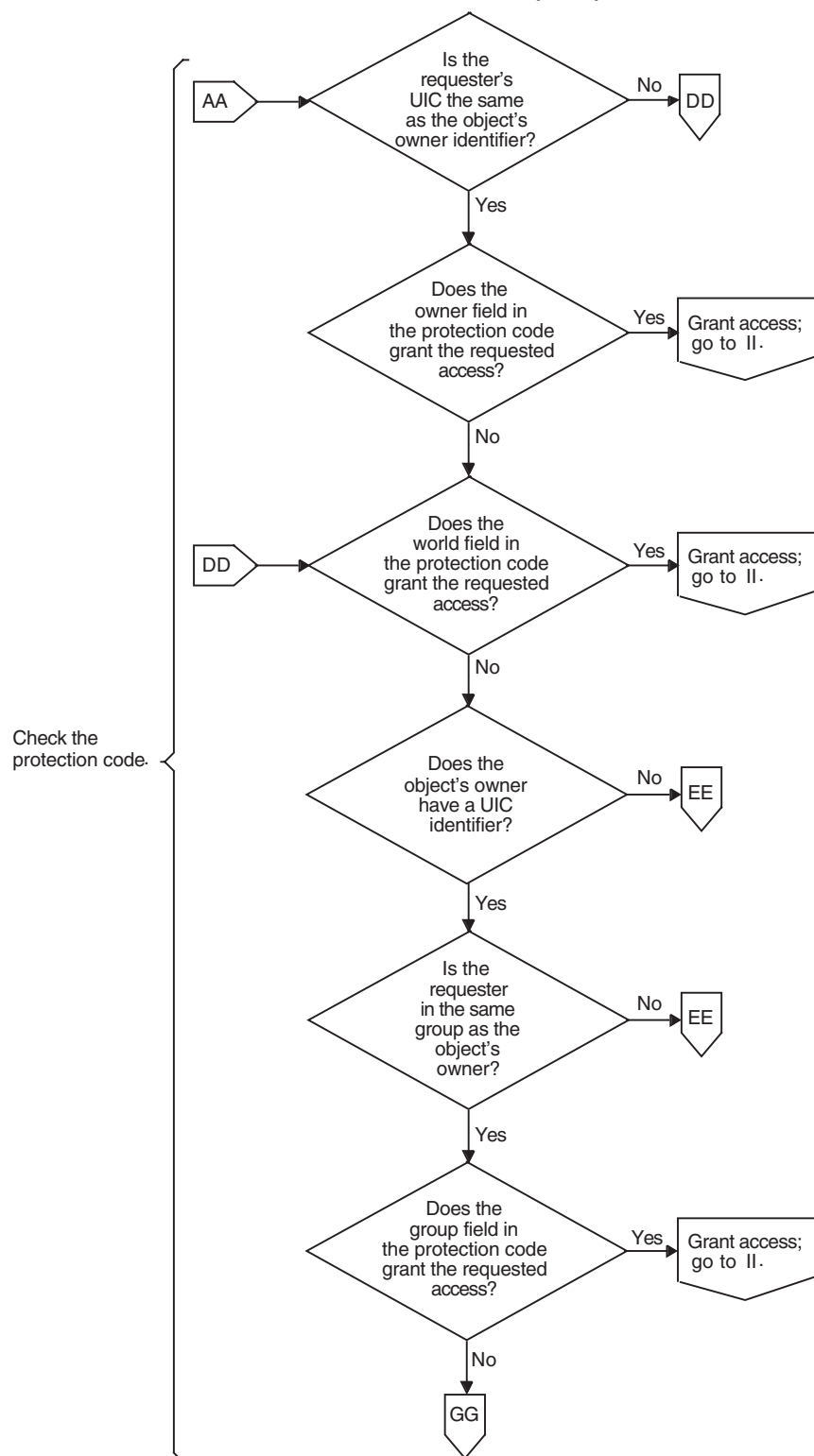


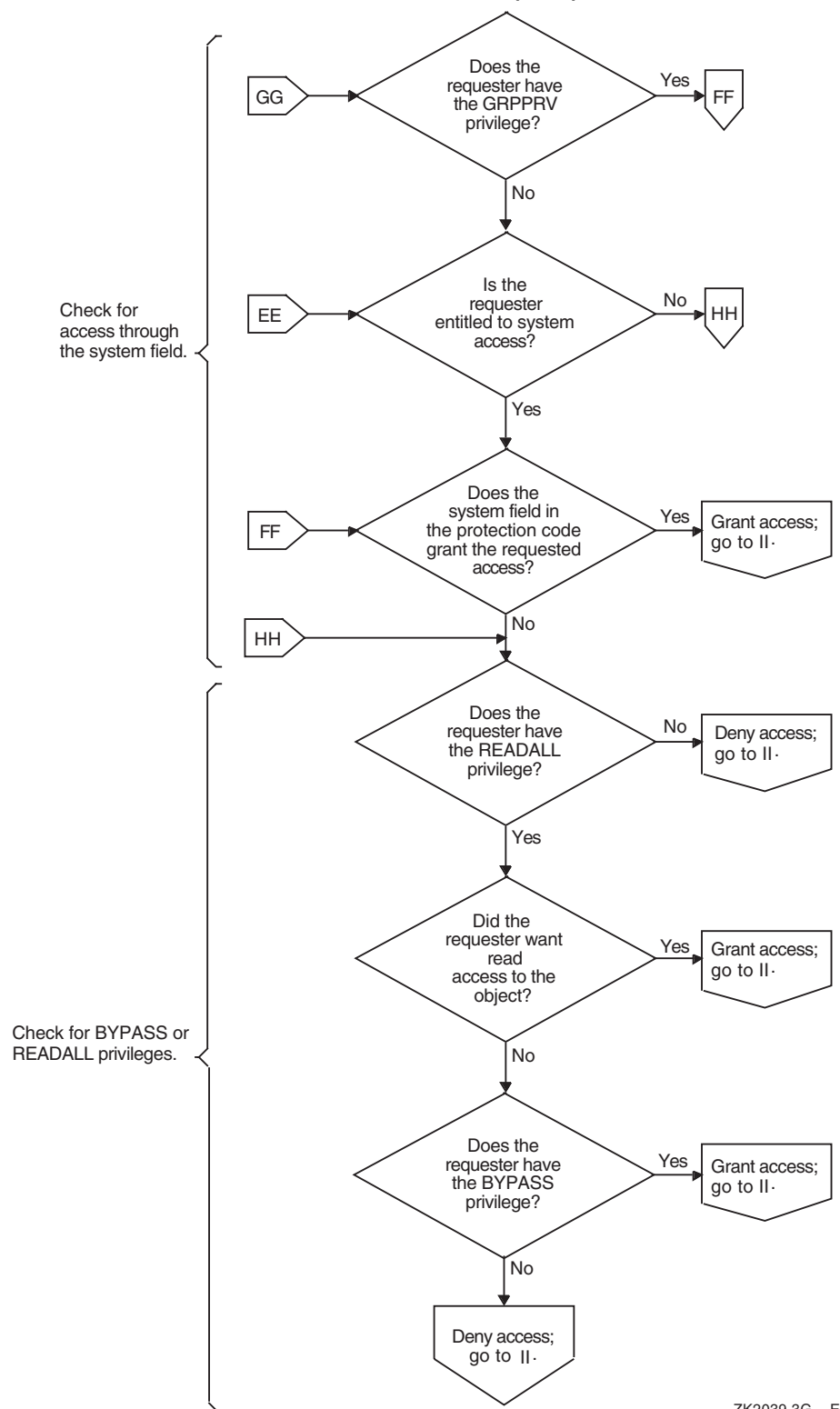
図 4-4: アクセス要求評価のフローチャート (続き)



ZK2039.2GE

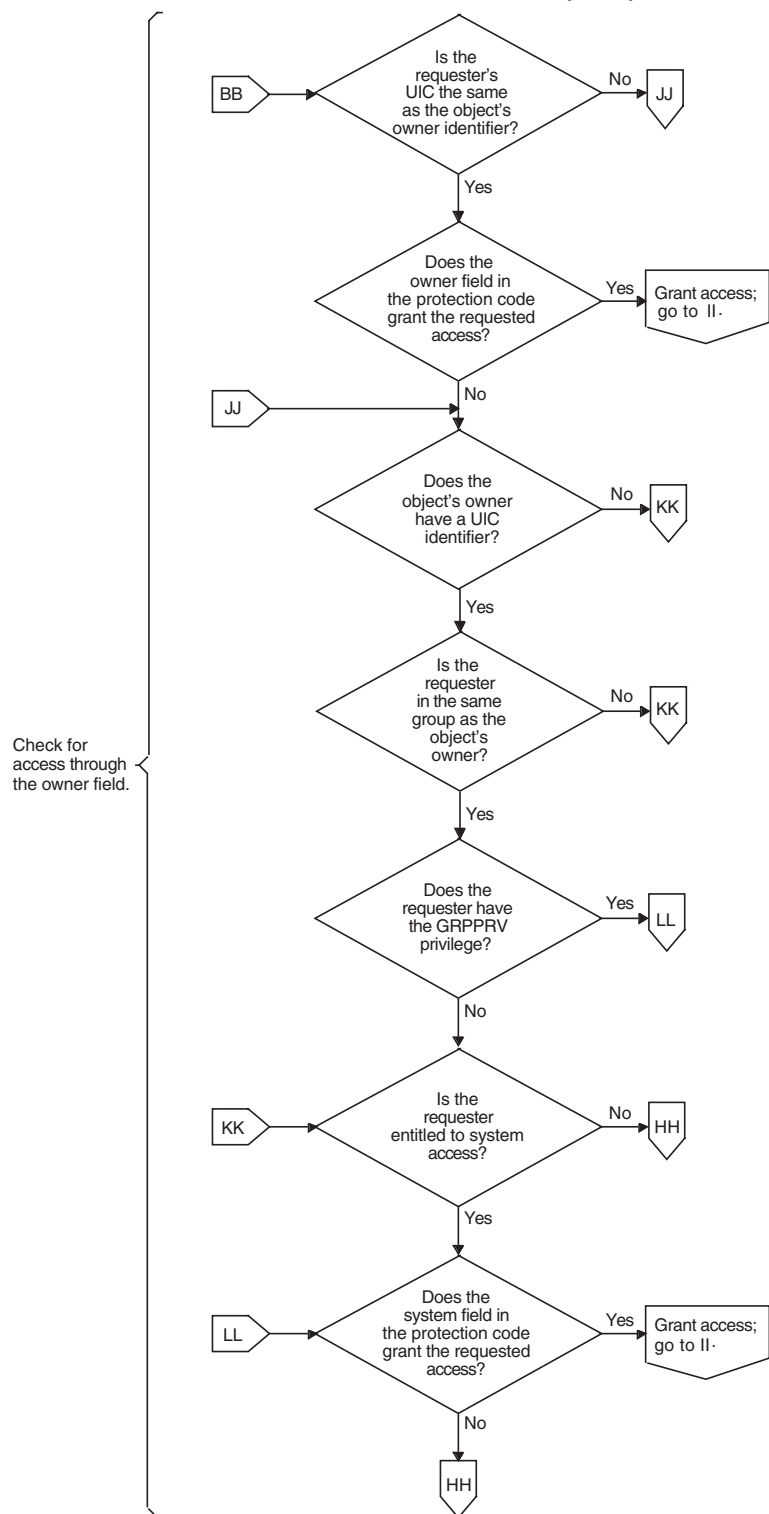


図 4-5: アクセス要求評価のフローチャート (続き)



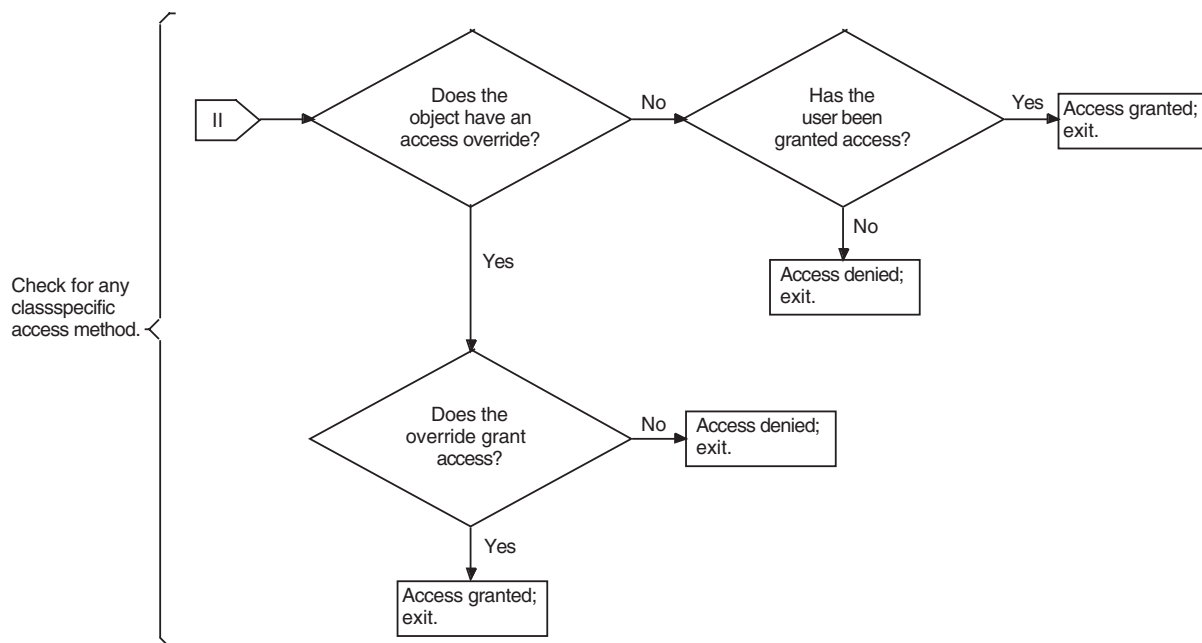
ZK2039.3G E

図 4-6: アクセス要求評価のフローチャート (続き)



ZK2039.4GE

図 4-7: アクセス要求評価のフローチャート (続き)



ZK2039.5GE

## 4.4 ACL によるアクセスの制御

4.2.2.3 項では、オブジェクトのセキュリティ・プロファイルを構成する要素の1つとしてアクセス制御リスト (ACL) を紹介しました。この節では、この保護メカニズムをさらに詳しく説明し、ACL を使ってオブジェクトを効果的に保護する方法の例を示します。

ほとんどの場合、オペレーティング・システムがオブジェクトに自動的に割り当てる保護コードで十分なため、多くのユーザは ACL について気にする必要がありません。しかし、特定のユーザに自分のファイルへのアクセスを許可する必要があることがあります。たとえば、同じプロジェクトで作業をしている場合などです。ACL は、重要なシステム・ファイル、デバイス、ボリューム、その他の保護オブジェクトを保護するための効果的なメカニズムなので、システム管理者やセキュリティ管理者は、一般ユーザよりも頻繁に ACL を使用します。

### 4.4.1 識別子用アクセス制御エントリ (ACE) の使用

アクセス制御リスト (ACL) 内のエントリは、アクセス制御エントリ (ACE) と呼ばれます。ACL は多数のエントリを持つことができ、個々のエントリはオブジェクトの何らかの属性を定義します。ACE には数多くの種類があります。詳細については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照してください。ここでは、オブジェクトへのアクセスを制御する識別子用 ACE について説明します。

識別子用 ACE には、1 つまたは複数のライト識別子と、その識別子を保持するユーザが行使を許可されているアクセスのタイプを示すリストが含まれています。システムは、オブジェクトに対するユーザの権限を評価するとき、アクセスするユーザが保持する 1 つまたは複数のライト識別子と一致する識別子用 ACE が見つかるまでオブジェクトの ACL を検索し、<sup>2</sup>見つかったエントリに基づいてアクセスを許可または拒否します。

ACE に応じて許可 (または拒否) するアクセスのタイプは、保護の対象となるオブジェクトによって異なります。たとえば、ファイルに対しては読み込み、書き込み、実行、および削除を実行でき、デバイスに対しては読み込みと書き込みの他に物理的操作と論理的操作を実行できます。したがって、ファイルは読み込み、書き込み、実行、および削除の各アクセスをサポートし、デバイスは読み込み、書き込み、物理、および論理の各アクセスをサポートします。他のオブジェクト・クラスがサポートするアクセスのタイプについては、第 5 章を参照してください。

識別子用 ACE を含む ACL を作成するには、DCL の SET SECURITY コマンドを次の形式で使します。

SET SECURITY/ACL=(IDENTIFIER=識別子,ACCESS=アクセスのタイプ)

たとえば、Fred というユーザが PROJECT-DATA.TXT というファイルを読めるようにするには、次のコマンドを入力します。

```
$ SET SECURITY/ACL=(IDENTIFIER=FRED,ACCESS=READ) PROJECT-DATA.TXT
```

"FRED" はユーザ識別コード (UIC) のメンバ名です。したがって、Fred に PROJECT-DATA.TXT ファイルへの読み込みアクセスを許可するエントリの UIC 識別子として機能します。

#### 4.4.2 特定ユーザへのアクセスの許可

個々のユーザまたはユーザ・グループの権限は識別子によって定義される (4.1.6.1 項を参照) ので、それらを識別子用 ACE に指定することによって、それらを保持するユーザに許可 (または拒否) するアクセスを定義します。システム上の個々のユーザまたはユーザ・グループは、UIC 識別子によって簡単に識別できます。それぞれが異なる機能グループ (つまり、さまざまな UIC グループ) に属するユーザのグループの全員が、ある保護オブジェクトへのアクセス権を必要とする場合、セキュリティ管理者は汎用識別子を作成し、アクセス権を必要としているすべてのユーザにその識別子を付与します。

たとえば、次のコマンドは UIC 識別子 [PAT] によって識別される Pat というユーザに、DISK1 上の ROBERTS ディレクトリ内のファイルに対する読み込み、書き込み、および実行アクセスを許可します。この ACL では、アクセス・ステートメ

---

<sup>2</sup> 識別子用 ACE に Default 属性があると、その ACE はアクセス評価時に無視されます。4.4.7 項を参照してください。

ントから削除アクセスと制御アクセスが除外されているため、Pat によるこれらのアクセスは拒否されます。

```
$ SET SECURITY/ACL=(IDENTIFIER=[PAT],ACCESS=READ+WRITE+EXECUTE)-  
_ $ DISK1:[ROBERTS]JULY-SALES.TXT
```

セキュリティ管理者は、登録ユーティリティを使用して汎用識別子を作成し、これを使用するすべてのユーザに与えます。たとえば、セキュリティ管理者が PAYROLL という識別子を作成し、それを給与計算ファイルにアクセスする必要がある従業員に割り当てたとします。この識別子の保持者が実際に給与計算ファイルにアクセスするには、管理者がそのファイルに識別子用 ACE を追加する必要があります。たとえば、次のコマンドは PAYROLL ファイルの ACL を作成し、PAYROLL 識別子の保持者にこのファイルへの読み込みアクセス権を与えます。

```
$ SET SECURITY/ACL=(IDENTIFIER=PAYROLL,ACCESS=READ) PAYROLL.DAT
```

ACL 内の ACE の順序は、オペレーティング・システムによる処理の規則に関わるので重要です。ACE の順序については、4.4.6 項を参照してください。

#### 4.4.3 オブジェクトへのユーザのアクセスの禁止

識別子用 ACE は、オブジェクトへのアクセスを許可する場合だけでなく、オブジェクトへの特定のユーザのアクセスを拒否する場合にも頻繁に使用されます。サイトによっては、モデムやネットワークを介してログインするユーザを制限するために ACL を使用場合があります。また、高価な装置や機密ファイルの入ったボリュームに ACE を設定して、それらへのアクセスを制限する場合があります。

特定の識別子の保持者に対してすべてのアクセスを拒否するには、アクセス・タイプ名として NONE キーワードを指定します。たとえば、次のコマンドは環境識別子 DIALUP の保持者に対して、PROJECT-ACCOUNTS ディレクトリ内のファイルへのあらゆるアクセスを拒否します。

```
$ SET SECURITY/ACL=(IDENTIFIER=DIALUP,ACCESS=NONE)-  
_ $ /CLASS=FILE PROJECT-ACCOUNTS.DIR
```

NONE キーワードを使ってアクセスを拒否する場合は、他にもいくつか考慮すべきことがあります。4.4.6 項で説明するように、OpenVMS オペレーティング・システムでは最初に一致する ACE に基づいてアクセスが許可または拒否されるため、ACL 内に ACE を正しく配置する必要があります。または、保護コードのグループ・カテゴリまたはワールド・カテゴリによって許可されるアクセスをすべて除外する方法もあります(具体的には 4.3 節と 4.5.5 項を参照)。セキュリティ管理者は、一致する ACE を変更できる特権を無効にすることもできます。

#### 4.4.4 デバイスへのアクセスの制限

セキュリティ管理者は、たとえば 4.4.2 項に示したように給与計算ファイルなどの共通ファイルへのアクセスを許可する一方で、小切手印刷用の高品質プリンタを使

用できるユーザの数を限定したい場合があります。限定しなければ、PAYROLL 識別子を保持するすべてのユーザが TTA8 プリンタに常時セットされている小切手フォームにアクセスできることになります。

この例では、小切手用プリンタがログインに使用されたり、キューの出力先に指定されることはないので、セキュリティ管理者はプリンタに ACL を追加して、McGrey というユーザにのみ読み込みアクセスと書き込みアクセスを許可するように設定できます。同時に、セキュリティ管理者は他の識別子の保持者によるプリンタへのアクセスを禁止する必要があります。次のコマンド・シーケンスを使用して、このような ACL を作成できます。

```
$ SET SECURITY/ACL=((IDENTIFIER=MCGREY,ACCESS=READ+WRITE)-  
_$(IDENTIFIER=*,ACCESS=NONE))/CLASS=DEVICE TTA8
```

McGrey には読み込みアクセスと書き込みアクセスが許可されますが、他のユーザは、4.4.3 項で説明したように、NONE キーワードによってアクセスを拒否されます。ただし、セキュリティ管理者がプリンタの保護コードを変更するまでは、TTA8 プリンタの ACL は意図したとおりに機能しない場合もあります。詳細については、4.5.5 項を参照してください。

#### 4.4.5 環境へのアクセスの制限

識別子用 ACE を使用し、特定の種類の識別子を組み合わせることによって条件付きのアクセスを提供できます。たとえば、代表的な例としては、BATCH や INTERACTIVE のような環境識別子とともに UIC 識別子を使用する方法があります（環境識別子の一覧については、4.1.6.1 項を参照してください）。この場合、ユーザはバッチ・モードまたは会話形式で実行されている場合にのみ保護オブジェクトにアクセスすることができ、ダイアルアップ回線経由ではアクセスできません。たとえば、次のコマンドは Fred というユーザにプリント・キューへの登録アクセスと管理アクセスを許可しますが、許可するのは Fred がバッチ・ジョブを実行している場合のみです。

```
$ SET SECURITY/ACL=(IDENTIFIER=[FRED]+BATCH,ACCESS=SUBMIT+MANAGE)-  
_$/CLASS=QUEUE SYSTEM6$LP40
```

#### 4.4.6 リスト内の ACE の順序

ACL には、1 つまたは複数のエントリを含めることができます。ACL に複数の ACE がある場合、最初に一致する ACE に基づいてアクセス権が決定されるため、エントリの順序が重要な意味を持ちます。オペレーティング・システムは ACL を先頭から順に検索し、最初に一致した ACE に指定されているアクセス権をユーザに付与します。それ以降のエントリはすべて無視されます。評価のプロセスについては、4.3 節を参照してください。

ACL を作成するときは、以下の原則に従います。

- 重要なユーザにアクセス権を付与する ACE をリストの先頭に置く。

- アクセス権の付与対象となるグループの規模が小さい ACE ほどリストの上位に置く。
- 選択的にアクセスを拒否する場合を除き、付与するアクセス権の数が多い ACE ほどリストの上位に置く。

PROJECT-ACCOUNTS.DIR ディレクトリ・ファイルに対する次の ACL を例に、ACL 内のエントリの順序の決め方を示します。この ACL では、重要なユーザ (Jones と Fred) にアクセス権を付与する ACE をリストの先頭に置き、その後一般の ACE を置いています。アクセスを拒否する ACE はリストの末尾に置いています。

```
$ SET SECURITY/ACL=( -
_$ (IDENTIFIER=[ACCOUNTING,JONES],ACCESS=READ+WRITE+EXECUTE),-
_$ (IDENTIFIER=[FRED]+BATCH,ACCESS=READ+WRITE+EXECUTE),-
_$ (IDENTIFIER=PAYROLL,ACCESS=READ),-
_$ (IDENTIFIER=DIALUP,ACCESS=NONE)) PROJECT-ACCOUNTS.DIR
```

プロジェクト・アカウントのディレクトリに設定されたこの ACL では、読み込み、書き込み、実行の各アクセスを Jones に対して常に許可し、Fred に対してはバッチ・ジョブの実行時にのみ許可しています。また、PAYROLL 識別子を保持するユーザには読み込みアクセス権を与えています。モデムからログインしたユーザについては、上位に置いた ACE によってアクセス権を付与しない限り、アクセスを拒否します。たとえば、Jones、Fred、または PAYROLL 識別子の保持者がダイアルアップした場合、これらのユーザに対する ACE を DIALUP の ACE の前に置いているため、アクセスが許可されます。

次の例は、STAFFING.DAT というデータ・ファイルの ACL です。この例では、最も多くのファイル・アクセス権を与えるエントリを ACL の先頭に置いています。

```
$ SET SECURITY/ACL=( -
_$ (IDENTIFIER=SECURITY,OPTIONS=PROTECTED,-
_$ ACCESS=READ+WRITE+EXECUTE+DELETE+CONTROL),-
_$ (IDENTIFIER=PERSONNEL,ACCESS=READ+WRITE+EXECUTE+DELETE),-
_$ (IDENTIFIER=SECRETARIES,ACCESS=READ+WRITE)-
_$ (IDENTIFIER=[PUB,*],ACCESS=READ),-
_$ (IDENTIFIER=NETWORK,ACCESS=NONE),-
_$ (IDENTIFIER=[SALES,JONES],ACCESS=NONE)) STAFFING.DAT
```

この ACL では、SECURITY 識別子を保持するユーザが最初の ACE によって最大限のアクセス権を取得し、PERSONNEL 識別子を保持するユーザがそれに次ぐアクセス権を取得します。Jones は、汎用識別子のいずれかを保持していない限り、ファイルへのあらゆるアクセスを禁止されます。これは、ACL の作成者のミスである可能性があります。Jones がファイルへのアクセス権を確実に獲得できないようにしたい場合は、このエントリを ACL の末尾から先頭に移動します。

#### 4.4.7 ファイルの継承方式の設定

ディレクトリ内またはディレクトリ構造内にあるファイルへのアクセスを制御するための計画を立て、各ファイル用の適切な ACL を作成したら、新しいファイルに

この ACL を自動的に割り当てるようにオペレーティング・システムに指示できます。このためには、**Default** 属性を持つ識別子用 ACE を作成し、対象となるファイルが登録されるディレクトリ・ファイルにその ACE を追加します。Default 属性を設定するには、OPTIONS キーワードを使用します。

たとえば、[MALCOLM] ディレクトリ内のすべての新しいファイルに対して、PERSONNEL 識別子を持つユーザに読み込みアクセスと書き込みアクセスを許可する ACL エントリを割り当てるには、MALCOLM.DIR ファイルに次の ACE を追加します。

```
$ SET SECURITY/ACL=(IDENTIFIER=PERSONNEL,OPTIONS=DEFAULT,-
_$ ACCESS=READ+WRITE) [000000]MALCOLM.DIR
```

この ACE を追加すると、[MALCOLM] ディレクトリ内に作成されたファイルには次の ACL が割り当てられます。

```
$ SHOW SECURITY APRIL_INTERVIEWS.TXT
```

```
WORK_DISK$: [MALCOLM]APRIL_INTERVIEWS.TXT;1 object of class FILE
```

```
Owner: [SALES,MALCOLM]
Protection: ...
Access Control List:
    (IDENTIFIER=PERSONNEL,ACCESS=READ+WRITE)
```

```
:
```

Default 属性は、新しいファイルの ACL には含まれず、ディレクトリ・ファイルの ACL にのみ含まれます。ただし、MALCOLM ディレクトリ内に作成されるサブディレクトリの ACL には、このエントリ (IDENTIFIER=PERSONNEL,OPTIONS=DEFAULT,ACCESS=READ+WRITE) が自動的に追加されます。このようにして、この ACE はディレクトリ木構造の全体に適用されます。

この ACE は、MALCOLM.DIR 内の既存のファイルに遡って適用されません。既存のファイルに ACE を追加するには、4.5.7 項で説明するように /DEFAULT 修飾子を使用するか、次のコマンドを使用します。

```
$ SET SECURITY/ACL=(IDENTIFIER=PERSONNEL,ACCESS=READ+WRITE)-
_$ [MALCOLM]*.*;*
```

Default 属性を持つ ACE は、その ACE の適用方法を制御するだけで、アクセス制御に対して影響を与えません。ファイルとディレクトリの両方へのアクセスを制御するには、次のように、ディレクトリの ACL に 2 つの ACE を追加します。

```
$ SET SECURITY/ACL=-
_$ ((IDENTIFIER=PERSONNEL,ACCESS=READ+WRITE),-
_$ (IDENTIFIER=PERSONNEL,OPTIONS=DEFAULT,ACCESS=READ+WRITE))-
_$ [000000]MALCOLM.DIR
```



#### 4.4.8 ACL の表示

DCL の SHOW SECURITY コマンドを使用して、オブジェクトの ACL を表示できます。ファイル以外のオブジェクトが対象の場合には、オブジェクト名とともにクラス名も指定する必要があります。たとえば、次のコマンドは PPA0 という名前のデバイスのセキュリティ属性を表示します。このデバイスはオペレーティング・システムによって所有されており、保護コードによってシステム・カテゴリと所有者カテゴリのユーザにフル・アクセス（読み込み、書き込み、物理、および論理アクセス）が許可されていますが、グループ・ユーザとワールド・ユーザにはアクセスが許可されていません。また ACL によって Svensen というユーザに制御アクセス権が付与されています。

```
$ SHOW SECURITY /CLASS=DEVICE PPA0:
```

```
_ACCOUNTS$PPA0: object of class DEVICE
```

```
Owner: [SYSTEM]
Protection: (System: RWPL, Owner: RWPL, Group, World)
Access Control List:
    (IDENTIFIER=[ADMIN,SVENSEN],ACCESS=CONTROL)
```

ACL を表示する方法は、他にもいろいろあります。アクセス制御リスト・エディタ (ACL エディタ) は、ACL に関するさまざまな作業を実行するときに便利なツールです。『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の ACL エディタに関する記述を参照してください。一方、以下の DCL コマンドでも ACL を表示できます。

---

```
SHOW SECURITY
DIRECTORY/ACL
DIRECTORY/SECURITY
DIRECTORY/FULL
SHOW LOGICAL/FULL/STRUCTURE
SHOW DEVICE/FULL
SHOW QUEUE/FULL
```

---

アプリケーションが ACE に **Hidden** 属性を追加して、ACE を変更できるのはその ACE を追加したアプリケーションだけであることを示す場合があります。隠された ACE は、ユーザが SECURITY 特権を持っていない限り、DCL コマンドでは表示されません。ACL エディタでは Hidden 属性を持つ ACE も表示されますが、ACL 内での相対的な位置を示すことが目的であり、権限のないユーザはそれらの ACE を編集できません。

ACL 内には、アクセス制御とは関係のない別の種類の ACE が設定されている場合があります。たとえば、セキュリティ管理者が LN03\$PRINT キューにセキュリティ監査用 ACE を設定した場合は、(AUDIT=SECURITY,ACCESS=アクセス・タイプ)という形式の ACE がリストの先頭に表示されます。このような ACE はセキュリティ監査システムの構成要素であり、アクセス制御には影響しないため、無視して構いません。

#### 4.4.9 既存の ACL への ACE の追加

4.4.2 項～ 4.4.5 項では、DCL の SET SECURITY コマンドを使って空の ACL にエントリを追加する方法を説明しています。ACL を広範に変更するには ACL エディタを使用しますが、多くの場合、SET SECURITY コマンドを使用する方が適切です。この節以降では、SET SECURITY を使用して ACL を変更する方法について説明します。

```
$ SET SECURITY/CLASS=QUEUE/ACL=(IDENTIFIER=WRITERS,-  
_ $ ACCESS=READ+WRITE) LN03$PRINT
```

ACL にエントリを追加するには、SET SECURITY コマンドに /ACL 修飾子を加え、新しい ACE を指定します。たとえば、文書作成者 (WRITERS) に LN03\$PRINT プリント・キューへのアクセス権を付与するには、次のコマンドを使用します。

次の SHOW SECURITY の表示結果からわかるように、新しい ACE はデフォルトで ACL の先頭に置かれます。

```
$ SHOW SECURITY /CLASS=QUEUE LN03$PRINT  
  
_LN03$PRINT: object of class QUEUE  
  
Owner: [SYSTEM]  
Protection: (System: RWPL, Owner: RWPL, Group, World)  
Access Control List:  
    (IDENTIFIER=WRITERS,ACCESS=READ+WRITE)  
    (IDENTIFIER=[PUB,*],ACCESS=READ)  
    (IDENTIFIER=NETWORK,ACCESS=NONE)
```

SET SECURITY のデフォルトの動作では新しい ACE が ACL の先頭に置かれるため、ACE を別の位置に入れたい場合は /AFTER 修飾子を使用する必要があります。たとえば、キューの ACL 内で TRADERS の ACE を WRITERS の ACE の後に置くには、次のコマンドを使用します。

```
$ SET SECURITY/CLASS=QUEUE/ACL=(IDENTIFIER=TRADERS,ACCESS=WRITE)-  
_ $ /AFTER=(IDENTIFIER=WRITERS,ACCESS=READ+WRITE) LN03$PRINT
```

表示結果から、/AFTER 修飾子の効果を確認できます。新しい ACE がリストの 2 番目の位置に置かれています。

```
$ SHOW SECURITY /CLASS=QUEUE LN03$PRINT  
  
_LN03$PRINT: object of class QUEUE
```

```

Owner: [SYSTEM]
Protection: (System: RWPL, Owner: RWPL, Group, World)
Access Control List:
    (IDENTIFIER=WRITERS, ACCESS=READ+WRITE)
    (IDENTIFIER=TRADERS, ACCESS=WRITE)
    (IDENTIFIER=[PUB, *], ACCESS=READ)
    (IDENTIFIER=NETWORK, ACCESS=NONE)

```

#### 4.4.10 ACL の削除

SET SECURITY コマンドに /DELETE 修飾子を加えると、ACL を削除できます。この修飾子の使い方次第で、ACL の全体または一部を削除できます。たとえば、次のコマンドはディスクの ACL を削除します。

```
$ SET SECURITY/CLASS=DEVICE/ACL/DELETE DUA0
```

**Protected** 属性が割り当てられている ACE は、不注意による削除から保護されます。保護されている ACE を削除するには、その ACE を明示的に削除するか、SET SECURITY/ACL コマンドに /DELETE=ALL 修飾子を指定する必要があります。

#### 4.4.11 ACL からの ACE の削除

ACL を部分的に削除するには、/ACL 修飾子を使って不要な ACE を指定した上で、/DELETE 修飾子を使用します。たとえば、次のコマンドは TRADERS 識別子と NETWORK 識別子の保持者に DBA0 ボリュームへの書き込みアクセス権を付与する ACE を削除します。

```

$ SET SECURITY/CLASS=VOLUME/ACL=-
_$ (IDENTIFIER=TRADERS, ACCESS=WRITE), -
_$ (IDENTIFIER=NETWORK, ACCESS=WRITE)/DELETE DBA0:

```

#### 4.4.12 ACL の部分的な置き換え

ACL 内の一定範囲の ACE を別の ACE にまとめて置き換えるには、次のように /REPLACE 修飾子を使って新しい ACE を指定し、/ACL 修飾子を使って削除する ACE を指定します。

```

$ SET SECURITY/CLASS=VOLUME/ACL=(IDENTIFIER=TRADERS, ACCESS=WRITE) -
_$ /REPLACE=((IDENTIFIER=RESEARCH, ACCESS=WRITE) -
_$ (IDENTIFIER=STATE_DEPARTMENT, ACCESS=READ+WRITE), -
_$ (IDENTIFIER=ENERGY_DEPARTMENT, ACCESS=READ+WRITE) -
_$ DBA0:

```

まず、/ACL で指定している TRADERS の ACE が削除されます。次に、/REPLACE 修飾子で指定している ACE (RESEARCH, STATE\_DEPARTMENT, ENERGY\_DEPARTMENT) が、削除された ACE の位置に挿入されます。

#### 4.4.13 ファイルのデフォルト ACL の復元

ファイルのデフォルト ACL を復元するには、SET SECURITY コマンドで /DEFAULT 修飾子を使用します。この修飾子を指定すると、ファイルの完全なセキュリティ・プロファイルが再生成されます。詳細については、4.5.7 項を参照してください。

#### 4.4.14 ACL のコピー

あるオブジェクトのセキュリティ・プロファイルを別のオブジェクトにコピーするには、SET SECURITY コマンドで /LIKE 修飾子を使用します。たとえば、論理名テーブルのような永続的でないオブジェクトに複雑な ACL を設定した場合でも、ファイルなどの永続的なオブジェクトにコピーすることによってその ACL を保存できます。管理者がコピー操作のテンプレートとして使用できるファイルを作成する場合もあります。これにより、管理者はオブジェクト間で ACL を簡単に転送することができます。たとえば、次のコマンドは ACL\_TEMPLATE.TXT ファイルから LNM\$GROUP 論理名テーブルに ACL をコピーします。

```
$ SET SECURITY/LIKE=NAME=ACL_TEMPLATE.TXT-  
_$_ /COPY_ATTRIBUTE=ACL/CLASS=LOGICAL_NAME_TABLE LNM$GROUP
```

/LIKE 修飾子に /COPY\_ATTRIBUTE 修飾子を追加すると、完全なプロファイルではなく 1 ~ 2 個の要素をコピーできます。たとえば、KITE\_FLYING ディレクトリに次のような ACL があるとします。

```
$ SHOW SECURITY [000000]KITE_FLYING.DIR;1 -
```

```
WORK_DISK$:[000000]KITE_FLYING.DIR;1 object of class FILE
```

```
Owner: [PROJECTX]  
Protection: (System: RWED, Owner: RWED, Group:, World)  
Access Control List:  
    IDENTIFIER=PROJECTX, ACCESS=READ+WRITE+EXECUTE  
    IDENTIFIER=PROJECTX, OPTIONS=DEFAULT, ACCESS=READ+WRITE+EXECUTE
```

次のコマンドは、上記の ACL を KITE\_FLYING ディレクトリから KITE\_DESIGNS ディレクトリにコピーします。

```
$ SET SECURITY/LIKE=KITE_FLYING.DIR;1 -  
_$_ /COPY_ATTRIBUTE=ACL KITE_DESIGNS.DIR;1
```

```
$ SHOW SECURITY [000000]KITE_DESIGNS.DIR;1 -
```

```
WORK_DISK$:[000000]KITE_DESIGNS.DIR;1 object of class FILE
```

```
Owner: [ENGINEERING]  
Protection: (System: RWED, Owner: RWED, Group:R, World:R)  
Access Control List:  
    IDENTIFIER=PROJECTX, ACCESS=READ+WRITE+EXECUTE  
    IDENTIFIER=PROJECTX, OPTIONS=DEFAULT, ACCESS=READ+WRITE+EXECUTE
```

SET SECURITY/LIKE コマンドによって、必ずしもコピー元オブジェクトの ACL 全体が複製されるとは限りません。たとえば、Nopropagate 属性を持つ ACE はコピー元の ACL からコピーされません。また、保護されている ACE も上書きされません。コピー先オブジェクトの保護されている ACE は維持され、コピーされた ACL に追加されます。たとえば、アプリケーションの多くはファイル・データの正しい表示方法を示すために特殊なタイプの保護されている ACE を使用しますが、これらの ACE はそのまま維持する必要があります。

ACE に設定できる属性の種類については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の ACL エディタに関する記述を参照してください。ACE のタイプについては、『*HP OpenVMS Programming Concepts Manual*』を参照してください。

## 4.5 保護コードによるアクセスの制御

保護コードは、特定のユーザまたはユーザのグループに対して許可（または拒否）するアクセスのタイプを制御します。アクセス・タイプは、保護オブジェクトに対する操作を実行するのに必要な権限を示します。OpenVMS オペレーティング・システムでは、1 つの操作を完了するのに複数のアクセス権が必要となる場合があります（4.7.2 項を参照）。保護コード内にユーザが属するカテゴリが見つかり、要求されたアクセスが（ACL で拒否されておらず）そのカテゴリで許可されていれば、ただちにユーザにオブジェクトへのアクセス権が付与されます。

### 4.5.1 保護コードの形式

保護コードの形式は、次のとおりです。

[ユーザ・カテゴリ: 許可されるアクセスのリスト (, ユーザ・カテゴリ: 許可されるアクセスのリスト,...)]

ユーザ・カテゴリ

ユーザ・カテゴリには、システム (S)、所有者 (O)、グループ (G)、およびワールド (W) があります。各カテゴリは、対応する英単語の先頭 1 文字で表すことができます。各カテゴリは次のように定義されます。

- システム: このカテゴリのメンバは、次のいずれかです。
  - 小さなグループ番号（通常は 8 進数の 1 ~ 10）を持つユーザ。一般的に、これらのグループ番号はシステム管理者、セキュリティ管理者、およびシステム・プログラマに割り当てられます。システム・グループ番号の正確な範囲は、セキュリティ管理者が MAXSYSGROUP システム・パラメータを設定することによって決まります。有効な範囲は 8 進数の 37776 までです。
  - SYSPRV 特権を持つユーザ。
  - オブジェクトの所有者と同じ UIC グループの GRPPRV 特権を持つユーザ。

- ボリュームの所有者と同じ UIC を持つユーザ (ディスク・ボリューム上のファイルに対するアクセス要求の場合)。
- 所有者: オブジェクトを現在所有しているユーザと同じ UIC を持つユーザ。一般的に、オブジェクトの作成者によるアクセスからオブジェクトを保護するための明示的な措置を取らない限り、オブジェクトに対する所有者アクセス権はその作成者に付与されます。
- グループ: オブジェクトの所有者と同じ UIC グループに属するすべてのユーザ。
- ワールド: 上記 3 つのカテゴリに属するユーザを含むすべてのユーザ。

複数のユーザ・カテゴリを指定する場合は、各カテゴリをコンマで区切り、コード全体を括弧で囲みます。ユーザ・カテゴリとアクセス・タイプは、任意の順序で指定できます。ただし、表示されるときは常にシステム、所有者、グループ、ワールドの順になります。

特定のユーザ・グループによるアクセスをすべて拒否するには、アクセス・タイプを指定せずにユーザ・カテゴリだけを指定します。ユーザ・カテゴリの後のコロンを省略することで、特定のユーザ・カテゴリによるアクセスを拒否できます。

特定のカテゴリ全体を省略すると、そのカテゴリに対するアクセス権は未指定となります。この場合、そのユーザ・カテゴリに現在許可されているアクセス権がそのまま適用されます。作成されたオブジェクトに (たとえば COPY/PROTECTION コマンドによって) 保護コードが適用される場合、省略されたカテゴリにはデフォルト値が割り当てられます。

#### アクセス・リスト

アクセス・タイプはオブジェクトによって決まります。アクセス・タイプについては第 5 章で説明します。ファイルに関するアクセス・タイプには、読み込み (R)、書き込み (W)、実行 (E)、および削除 (D) があります。アクセス・タイプはユーザ・カテゴリごとに割り当て、アクセス・タイプとユーザ・カテゴリはコロン (:) で区切ります。たとえば、SET SECURITY/PROTECTION=(S:RWE,O:RWE,G:RE,W) のようにします。

### 4.5.2 保護コード内のアクセスのタイプ

個々のユーザ・カテゴリに対して、異なるタイプのアクセスを許可または拒否できます。正確なタイプは、保護対象オブジェクトによって異なります。各オブジェクト・クラスには、そのクラスに対応するアクセス・タイプが定義されています。これらのアクセス・タイプは、ユーザがそのデータに対して実行する操作の典型を示しています。たとえば、ファイル・オブジェクトは読み込み、書き込み、実行、および削除の各アクセスをサポートするのに対して、デバイス (ターミナル、プリンタ、ディスクなど) は読み込み、書き込み、物理入出力、および論理入出力

の各アクセスをサポートします。各オブジェクト・クラスがサポートするアクセス・タイプについては、第 5 章を参照してください。

すべての保護オブジェクトは、制御アクセスをサポートします。制御アクセスにより、ユーザはオブジェクトのセキュリティ要素 (ACL、保護コード、UIC) と、場合によってはその他の属性を参照および変更できます。制御アクセス権は、ACL では明示的に記述されますが、UIC ベースの保護コードには現れません。保護コードのシステム・カテゴリまたは所有者カテゴリに属するユーザはすべて制御アクセス権を持っています。グループ・カテゴリとワールド・カテゴリのユーザは、保護コードによる制御アクセス権を獲得することはありませんが、ACL で獲得することは可能です。詳細については、4.6.2 項を参照してください。

読み込み、書き込み、実行、削除、および制御の各アクセス・タイプによって指定される機能は、適用される状況によって異なります。たとえば、実行アクセスで許可される操作は、ファイル・アクセスとディレクトリ・アクセスのどちらに対して許可されるかによって異なります。各アクセス・タイプが保護オブジェクトの各タイプに対して許可する機能については、第 5 章で説明します。

### 4.5.3 保護コードの処理

システムによる保護コードの評価は、所有者フィールド、ワールド・フィールド、グループ・フィールド、システム・フィールドの順に行われます。ユーザがあるカテゴリのメンバとしての条件を満たし、そのカテゴリによって必要なアクセス権が付与されると、保護コードの処理はそこで終了します (図 4-3 を参照)。

次の保護コードは、システム・カテゴリと所有者カテゴリのユーザが読み込み (R)、書き込み (W)、実行 (E)、および削除 (D) のアクセス権を持ち、グループ・カテゴリとワールド・カテゴリのユーザが読み込みと実行のアクセス権のみを持つことを指定します。

```
$ SET SECURITY/PROTECTION=(SYSTEM:RWED, OWNER:RWED,  
GROUP:RE, WORLD:RE) -  
_$ TAXES_91.DAT
```

特定のユーザ・カテゴリに対してアクセスを拒否する場合は、それより範囲の広いすべてのカテゴリに対してアクセスを拒否する必要があります。4.5.1 項で説明したように、ユーザ・プロセスおよびアプリケーションはすべてワールド・カテゴリのアクセスを認められます。グループ・カテゴリのアクセスは、ワールド・カテゴリよりも制限されていますが、所有者カテゴリおよびシステム・カテゴリほど制限が厳しくありません。

たとえば、次の保護コードは所有者カテゴリに対して削除アクセスを拒否しているように見えます。

```
$ SHOW SECURITY TAXES_91.DAT
```

```
WORK_DISK$:[GREG]TAXES_91.DAT;1 object of class FILE
```

```
Owner: [FINANCE,GREG]
Protection: (System: RWED,
Owner: RW, Group:RW, World:RWED)
Access Control List:
```

```
...
```

しかし、このファイルの所有者は依然としてこのファイルを削除できます。所有者カテゴリでは削除アクセスを許可していませんが、アクセスが許可されているかどうかのチェックは他のカテゴリについても行われます。所有者はワールド・カテゴリ（すべてのユーザに適用されるカテゴリ）にも属しており、ワールド・カテゴリでは削除アクセスが許可されているので、所有者にも削除アクセスが許可されます。

#### 4.5.4 保護コードの変更

SET SECURITY コマンドを使用して、既存オブジェクトの UIC ベースの保護を変更できます。次のコマンドは、SURVEY.DIR ファイルの保護コードを変更して、システム・カテゴリと所有者カテゴリのユーザに読み込み、書き込み、実行、および削除の各アクセス権を与え、グループ・カテゴリとワールド・カテゴリのメンバに読み込みと実行のアクセス権を与えます。

```
$ SET SECURITY/PROTECTION=(SYSTEM:RWED,OWNER:RWED, -
_$ GROUP:RE,WORLD:RE) SURVEY.DIR
```

保護コードからカテゴリを省略すると、現在のアクセス権がそのまま適用されます。たとえば、RECORDS\_91.DAT というファイルの保護コードを考えます。

```
$ SHOW SECURITY RECORDS_91.DAT
```

```
WORK_DISK$:[GREG]RECORDS_91.DAT object of class FILE
```

```
Owner: [VMS,GREG]
Protection: (System: RWED, Owner:
RWED, Group: RWED, World: RE)
```

現在の RECORDS\_91 ファイルでは、システム、所有者、グループの各カテゴリのユーザに対して読み込み、書き込み、実行、および削除の各アクセスが許可され、ワールド・カテゴリのユーザに対して読み込みアクセスと実行アクセスが許可されています。次の DCL コマンドは、RECORDS\_91.DAT の保護コードを再設定して、グループ・カテゴリに対して書き込みアクセスと削除アクセスを拒否し、ワールド・カテゴリに対してすべてのアクセスを拒否します。

```
$ SET SECURITY/PROTECTION=(G:RE,W) RECORDS_91.DAT
```



次のコマンドは、変更した保護コードを確認します。システム・カテゴリと所有者カテゴリのユーザには依然として読み込み、書き込み、実行、および削除の各アクセス権が与えられているのに対して、グループ・ユーザには読み込みアクセス権と実行アクセス権のみが与えられ、ワールド・ユーザにはアクセス権がまったく与えられていません。

```
$ SHOW SECURITY RECORDS_91.DAT
```

```
WORK_DISK$:[GREG]RECORDS_91.DAT object of class FILE
```

```
Owner: [VMS,GREG]
```

```
Protection: (System: RWED, Owner:
```

```
RWED, Group: RE, World:)
```

#### 4.5.5 機密オブジェクトに対する保護の強化

4.4.4 項では、重要なプリンタに ACL を設定して、プリンタへのアクセス権を 1 人のユーザに限定する方法を説明しています。しかし、この ACL を有効にするには、セキュリティ管理者が次のコマンドを使用して、プリンタの保護コードによって許可されるすべてのアクセスを排除する必要があります。

```
$ SET SECURITY/PROTECTION=(S,O,G,W)/CLASS=DEVICE TTA8:
```

次に、セキュリティ管理者は ACL を使用してアクセス権を明示的に割り当てます。

たとえば、キューへのアクセスを制限するには、ワールド・カテゴリのキュー登録アクセス権を削除します。次に、ACL を設定して、(ワールド・カテゴリの)どのユーザにキューへのジョブ登録を許可するかを指定します。次のコマンドは、PROJECTX 識別子の保持者にのみ LN03\$PRINT キューへのジョブ登録を許可します。

```
$ SET SECURITY/CLASS=QUEUE/PROTECTION=(W) -  
_$/ACL=(IDENTIFIER=PROJECTX,ACCESS=SUBMIT) -  
_$/LN03$PRINT
```

重要なファイルは多くの場合、特別な保護を必要とします。ディレクトリの内容を参照できないようにするには、ユーザによる読み込みアクセスを拒否します。ファイルの保護を強化するには、4.5.6 項に示すように、ディレクトリ・ファイルにデフォルトの保護用 ACE を追加します。

#### 4.5.6 ディレクトリ構造に対するデフォルトの保護コードの提供

特定のディレクトリ内の新しいファイルに対してデフォルトの保護を指定するには、ディレクトリ・ファイルの ACL にデフォルトの保護用 ACE を含めます。デフォルトの保護用 ACE は、以降そのディレクトリおよびその下のサブディレクトリに作成されるファイルに適用されます。ただし、ファイルに個別の保護が指定された場合は除きます。この ACE の形式は、次のとおりです。

(DEFAULT\_PROTECTION[,オプション],保護コード)

たとえば、次の ACE は、システム・カテゴリと所有者カテゴリのユーザに、このディレクトリ内で新たに作成されるファイルに対する読み込み、書き込み、実行、および削除の各アクセス権を与え、グループ・カテゴリとワールド・カテゴリのユーザにアクセス権をまったく与えないことを指定します。

```
$ SET SECURITY/ACL=(DEFAULT_PROTECTION,S:RWED,O:RWED,G,W) ARCHIVE.DIR
```

デフォルトの保護は新たに作成されるファイルにのみ適用される点に注意してください。現在のディレクトリとそのサブディレクトリに既に存在するファイルには適用されません。ディレクトリ・ファイルにデフォルトの保護用 ACE を追加し、既存のファイルにも同じ保護を適用したい場合は、次のコマンドを使用して明示的に保護を変更する必要があります。

```
$ SET DEFAULT [ARCHIVE]
```

```
$ SET SECURITY/PROTECTION=(S:RWED,O:RWED,G,W) [...] *.*;*
```

#### 4.5.7 ファイルのデフォルト・セキュリティ・プロファイルの復元

SET SECURITY コマンドに /DEFAULT 修飾子を加えると、ファイルのセキュリティ・プロファイルが再生成されます。/DEFAULT 修飾子は、ファイルの保護コード、ACL、および所有者要素を、ファイルの親ディレクトリに指定されているデフォルト（つまり、ディレクトリのデフォルト ACL、(もしあれば) デフォルトの保護用 ACE、および所有者 UIC）に再設定します。

セキュリティ・プロファイルは次の規則に従って再生成されます。

- 保護コードは、(もしあれば) ディレクトリのデフォルトの保護用 ACE から継承されます。ない場合は、プロセスのデフォルトから継承されます。
- ACL については、親ディレクトリの Default 属性を持つ ACE から継承されます。
- 所有者については、親ディレクトリの所有者が設定されます。ファイルの所有者を変更するには、通常、特権が必要です（5.4.2 項を参照）。

サブディレクトリ・ファイルの場合は、SET SECURITY によって親ディレクトリの所有者、保護、および ACL の各要素が割り当てられます。

SET SECURITY は、設定元のオブジェクトの ACE に Nopropagate 属性がある場合は、その ACE をコピーしません。また、設定先のオブジェクトの ACE に Protected 属性がある場合は、その ACE を変更しません。ファイルのすべてのバージョンに新しい要素を適用するには、オブジェクト名として ;\* を指定します。

継承規則の詳細については、5.4.5 項を参照してください。

## 4.6 特権と制御アクセス

オブジェクトは ACL と保護コードによって入念に保護できますが、ユーザは依然として特権および制御アクセスを利用することでアクセス権を獲得することができます。

### 4.6.1 保護メカニズムに対する特権の影響

セキュリティ管理者は、ユーザ・アカウントを作成または変更するときに、ユーザに特権を割り当てることができます。システム特権の READALL と BYPASS は、オブジェクトの ACL およびセキュリティ・プロファイルの他の要素によって指定されるアクセス権に関係なく、ユーザ・アクセスに影響を与えます。SYSPRV 特権と GRPPRV 特権は、保護コードのシステム・カテゴリで制御します。これらの特権の意味を次に示します。

BYPASS	BYPASS 特権を有するユーザは、オブジェクトの保護に関係なく、オブジェクトに対するあらゆるタイプのアクセス権を獲得します。
GRPPRV	GRPPRV 特権を有し、UIC グループがオブジェクトの所有者のグループと一致するユーザは、システム・カテゴリのユーザと同じアクセス権を獲得します。したがって、GRPPRV 特権を有するユーザは該当グループのすべてのオブジェクトを管理できます。
READALL	READALL 特権を有するユーザは、オブジェクトに対する読み込みアクセス権を獲得します。これは、ACL および保護コードによってそのアクセスが拒否される場合も含みます。さらに、ユーザは保護コードによって付与される他のアクセス権もすべて獲得します。
SYSPRV	SYSPRV 特権を有するユーザは、システム・カテゴリのユーザに付与されるアクセス権を獲得します。

オブジェクトに対して ACL や保護コードを定義するときは、強力な特権を有するユーザにはシステム全体のオブジェクトに対する特別なアクセス権を付与されることを忘れないでください。たとえば、BYPASS 特権を有するユーザによる特定のファイルへのアクセスを防ぐ方法はありません。GRPPRV 特権を有するユーザは、自分の UIC グループに属する他のメンバのためにさまざまなシステム管理機能を実行する権限を持っています。オブジェクトの保護は、これらの特権の付与に関するセキュリティ管理者の判断に左右されます。

### 4.6.2 制御アクセスによるオブジェクトのプロファイルの変更

オブジェクトに対する制御アクセス権を持つユーザは、オブジェクトの保護コードと ACL を変更することにより、オブジェクトへのアクセス権を獲得できます。ファイル以外のオブジェクト・クラスについては、制御アクセス権を持つユーザがオブジェクトの所有者を変更することもできます。ファイルの所有者の変更には、通常、特権が必要です（5.4.2 項を参照）。

制御アクセス権は、次のいずれかの方法で獲得できます。

- オブジェクトの ACL によって制御アクセス権が付与される識別子を持っている。
- オブジェクトの所有者と同じ UIC を持っている。
- システム・ユーザ・カテゴリのメンバとしての条件を満たし、ゼロ以外の UIC を持つユーザがオブジェクトの所有者になっている。たとえば、(同じグループ UIC に属して) GRPPRV 特権を持っているか SYSPRV 特権を持っている。システム・ユーザの詳細については、4.5 節を参照してください。
- BYPASS 特権を持っている。

オブジェクト・クラスによっては、制御アクセスが他の手段で許可される場合があります。これに該当する特別な状況については、4.6.3 項および第 5 章の各クラスの説明を参照してください。

### 4.6.3 アクセスに関するオブジェクト固有の考慮事項

オブジェクトによっては、(4.6.1 項に示した以外の) 特別な特権や包括的なタイプのアクセス権によってアクセスが許可される場合があります。特に、キューがこれに該当します。オペレータ (OPER) 特権を持つユーザには、キューに対するすべてのタイプのアクセスが許可されます。管理アクセス権を持つユーザは、キューに対する他のタイプのアクセス権 (読み込み、登録、および削除) を暗黙に保持しています。第 5 章に、各オブジェクト・クラスのアクセス・タイプ、意味、および特別な特権を示します。

## 4.7 保護オブジェクトの監査

プロセスがオブジェクトを使用するか、オブジェクトのセキュリティ・プロファイルを変更する (4.2.4 項を参照) たびに、オペレータ・ターミナルにアラームを送信するか、監査ログ・ファイルにメッセージが記録するようにできます。セキュリティ管理者は、ログ・ファイルを参照することにより、システムの動作を調べ、保護オブジェクトをいつ、誰が、どのように使用したかを確認できます。

監査システムによってどのような情報が報告されるかは、セキュリティ管理者がサイトの要件をどのように定義するかによります。オブジェクトの使用状況を監査する場合、システム管理者は適切なカテゴリのイベントに対する監査機能を有効にできます。

OpenVMS オペレーティング・システムでは、セキュリティ関連イベントにフィルタを適用して、オブジェクトが特定の 방법으로アクセスされたときにだけシステム管理者にメッセージを送信するようにできます。多くのサイトでは、すべてのファイル・アクセスではなく、特権を利用したファイルの使用やファイルへのアクセスの失敗が関心の対象になります。このようなサイトでは、プロセスがファイル

へのアクセスに成功した場合ではなく、失敗した場合の監査メッセージを要求できます。システムは、プロセスがオブジェクトにアクセスする権利をそもそもどのように（保護コード、ACE、または特権を介して）行使したか（または行使できなかったか）を報告します。

#### 4.7.1 システムが監査するイベントの種類

オブジェクト・クラスはそれぞれに固有の監査プロファイルがあるため（第5章を参照）、オブジェクトのクラスによっては他のクラスよりも詳細な情報を得ることができます。どのオブジェクトについても、ユーザまたはアプリケーションがオブジェクトにアクセスするか、セキュリティ要素を変更したときに、必ず監査メッセージが送信されるようにできます。場合によっては、プロセスがオブジェクトを作成したとき、オブジェクトの使用を止めた（アクセスを解除した）とき、またはオブジェクトを削除したときに通知を送信することもできます。

#### 4.7.2 オブジェクトのクラスに対する監査の有効化

オブジェクト・アクセス・イベントを監査するときは、1つの操作の間に、オブジェクトに対するユーザの権利がオペレーティング・システムによって複数回チェックされる場合があることに注意してください。たとえば、ファイル操作ではディレクトリ・アクセスに関するチェックとファイル・アクセスに関するチェックの両方が行われます。ユーザがファイルを削除する前には、そのファイルに対する削除アクセス権のチェックと、ディレクトリに対する書き込みアクセス権のチェックが行われます。

このため、セキュリティ管理者は、すべてのタイプのオブジェクト・アクセス・イベントについて監査機能を有効にするのが最善策です。たとえば、ユーザがファイルにアクセスしようとして失敗した場合をすべて追跡するには、セキュリティ管理者が SET AUDIT コマンドに /ENABLE=ACCESS=FAILURE=ALL 修飾子を指定します。

アクセス解除の監査をサポートするオブジェクト・クラス（ファイル・クラスなど）では、プロセスがオブジェクトへのアクセス権を獲得すると、そのプロセスがすでに許可されているアクセス・モードに適合しない操作を行わない限り、システムはそのオブジェクトに対する以降のアクセス操作を監査しません。この状況が発生すると、システムは監査対象となる追加的な保護チェックを実行します。このアクセス・ウィンドウは、オブジェクトのアクセスが解除される（たとえば、ファイルが閉じられる）まで継続します。

#### 4.7.3 セキュリティ監査用 ACE の追加

セキュリティ管理者とオブジェクトに対する制御アクセス権を持つユーザは、アラーム用 ACE または監査用 ACE をオブジェクトに追加することにより、オブジェクトのクラス全体を監査するのではなく、特定のオブジェクトに監査対象を絞るこ

とができます (3.10.2 項を参照)。ユーザは監査用 ACE を自分が所有するファイルまたは制御アクセス権を持つファイルに追加できますが、事前にセキュリティ管理者に相談することをお勧めします。オブジェクト・クラスの場合と同じように、監査メッセージを生成させるには、セキュリティ管理者が ACL の監査カテゴリを有効にする必要があります。

## オブジェクト・クラスの詳細

この章では、保護オブジェクトの各クラス（ファイル、ボリューム、デバイスなど）の特徴を説明します。クラスごとに、次のトピックに関する情報を示します。

トピック	説明
命名規則	クラス内のオブジェクトに対する命名規則の概要。
アクセスのタイプ	そのクラスでサポートされるアクセス・タイプ。太字の部分はアクセス・タイプの短縮形を示します（たとえば、読み込みアクセスは R）。
テンプレート・プロファイル	クラスの新しいオブジェクトに適用されるデフォルトのプロファイル。サイトのセキュリティ管理者は、デフォルトのプロファイルを変更できます。現在のテンプレートの設定値を表示するには、SHOW SECURITY コマンドを使用します。
必要な特権	オブジェクトに対する特定の操作に必要な特権（もしあれば）。
実行される監査の種類	監査イベント・メッセージを起動するイベント（そのイベント・クラスが有効になっている場合）。
オブジェクトの永続性	セキュリティ・プロファイルの格納。セキュリティ要素がシステムのスタートアップをまたいで保存されるかどうか、また、保存される場合はどこに保存されるかを説明します。

クラスに当てはまらないトピックは省略されます。

### 5.1 ケーパビリティ

ケーパビリティは、各サイトが標準のアクセス制御メカニズムを使ってアクセスを制御するリソースです。ベクタ命令を実行できる能力は、ケーパビリティ・オブジェクトです。このようなオブジェクトは、ベクタ・プロセッサを持つサイトにのみ存在します。

#### 5.1.1 命名規則

ケーパビリティ・オブジェクトに対する有効な名前は、VECTOR のみです。

#### 5.1.2 アクセスのタイプ

ケーパビリティ・クラスは、次のアクセス・タイプをサポートします。

使用 (U)	ベクタ・プロセッサを使用する権限がプロセスに与えられます。
制御 (C)	オブジェクトの保護と所有権の要素を変更する権限がユーザに与えられます。

### 5.1.3 テンプレート・プロファイル

ケーパビリティ・クラスは、次のテンプレート・プロファイルを提供します。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[SYSTEM]	S:U,O:U,G:U,W:U

VECTOR テンプレートに対する変更は、システムの次回起動時に有効になります。システムの起動後に VECTOR オブジェクトの要素を変更する場合は、そのオブジェクトを直接変更する必要があります。次に例を示します。

```
$ SET SECURITY/CLASS=CAPABILITY/PROTECTION=(S:U,O:U,G:U,W) VECTOR
```

### 5.1.4 実行される監査の種類

OpenVMS オペレーティング・システムでは、次のタイプのイベントを監査できます。

監査対象イベント	監査実行のタイミング
アクセス (A)	イメージの起動後、プロセスが最初にベクタ命令を使用したとき。

### 5.1.5 オブジェクトの永続性

ケーパビリティ・オブジェクトのセキュリティ・プロファイルは、システムが起動されるたびに再設定する必要があります。

## 5.2 コモン・イベント・フラグ・クラスタ

コモン・イベント・フラグ・クラスタは、プロセス同士が連携してイベント通知を相互に提供できるようにするために、32 個のイベント・フラグをセットにしたものです。

クラスタ内のイベント・フラグをセットまたはクリアすることにより、イベントの発生を示すことができます。32 個のイベント・フラグで構成されるクラスタ内にすべてのイベント・フラグが格納され、各プロセスは 4 つのクラスタ (0 ~ 3 番) にアクセスできます。このうち 2 つのクラスタは 1 つのプロセスがローカルに使用します。イベント・フラグ・クラスタ 2 と 3 はコモン・イベント・フラグ・クラスタと呼ばれ、プロセス間の同期化のために使用されます。1 つのサブジェクトに、最大 2



つのコモン・イベント・フラグ・クラスタを関連付けることができます。クラスタ内の各コモン・イベント・フラグは、イベント・フラグ番号によって参照されます。

### 5.2.1 命名規則

このオブジェクトの名前は、コモン・イベント・フラグ・クラスタ関連付けシステム・サービス (\$ASCEFC) に対する引数として指定した任意の文字列です。コモン・イベント・フラグ・クラスタの名前は、UIC グループ番号で修飾されます。

### 5.2.2 アクセスのタイプ

コモン・イベント・フラグ・クラスタ・クラスは、次のアクセス・タイプをサポートします。

関連付け (A)	プロセスがイベント・フラグにアクセスできるように、名前付きのクラスタとの対応関係を設定する権限がプロセスに与えられます。
削除 (D)	パーマネント・イベント・フラグ・クラスタを、コモン・イベント・フラグ・クラスタ削除 (\$DLCEFC) システム・サービスによる削除の対象に指定する権限がプロセスに与えられます。実際の削除処理は、クラスタからすべてのプロセスの関連付けが解除されたときに実行されます。
制御 (C)	コモン・イベント・フラグ・クラスタの保護要素を変更する権限がユーザに与えられます。

### 5.2.3 テンプレート・プロファイル

コモン・イベント・フラグ・クラスタは、1つのテンプレート・ファイルを提供します。このテンプレートは [0,0] という所有者 UIC を割り当てますが、この値は一時的なものです。オブジェクトが作成されると、作成したプロセスの UIC の対応するフィールド内の値が 0 の代わりに設定されます。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[0,0]	S:AD,O:AD,G:A,W

コモン・イベント・フラグ・クラスタを作成するプロセスが \$ASCEFC の *prot* 引数に 1 を指定すると、プロセスの UIC が所有者になるようにテンプレートが変更され、保護コードによってグループ・アクセスが拒否されます。

### 5.2.4 必要な特権

パーマネント・コモン・イベント・フラグ・クラスタの作成には、PRMCCEB 特権が必要です。この特権は、パーマネント・クラスタに対する削除アクセスも許可します。

### 5.2.5 実行される監査の種類

OpenVMS オペレーティング・システムでは、次のタイプのイベントを監査できます。

監査対象イベント	監査実行のタイミング
作成 (C)	特定のクラスタに最初に関連付けられるプロセスが \$ASCEFC を呼び出したとき。
アクセス (A)	\$ASCEFC を 2 番目以降に呼び出すプロセスがクラスタに関連付けられたとき。
アクセス解除 (D)	プロセスが \$DACEFC を呼び出すか、別のクラスタに関連付けられるか、またはイメージがランダウンされたとき。
削除 (D)	プロセスが \$DLCEFC を呼び出したとき。

### 5.2.6 オブジェクトの永続性

コモン・イベント・フラグ・クラスタとそのセキュリティ・プロファイルは、システムが起動されるたびに再設定する必要があります。

## 5.3 デバイス

デバイスは、プロセッサに物理的に接続されるか論理的に認識される周辺機器で、データを受信、保存、または伝送する機能を持つものです。デバイスには、ディスクやターミナルのように物理的なものと、メールボックスや擬似ターミナルのように仮想的なものがあります。仮想デバイスは、完全にソフトウェアで実装されています。仮想ターミナルは、ローカル・デバイスと見なされ、ネットワーク上またはローカル・システム上に作成されます。

### 5.3.1 命名規則

デバイスを参照するときは、物理名、論理名、または汎用名を使用できます。また、クラスタ環境内のシステムでは、デバイスによってはクラスタのすべてのメンバからアクセスできます。名前の形式は次のとおりです。

- ほとんどの物理デバイス名は、次の 3 つの部分で構成されます。
  - ハードウェア・デバイスのタイプを表すデバイス・コード (dd)
  - デバイスが接続されるハードウェア・コントローラを識別するコントローラ指示子 (c)
  - 特定のコントローラに接続されたデバイスを一意に識別するユニット番号 (u)

コントローラ番号とユニット番号を含むデバイス名フィールド全体の最大長は、15 文字です。

- 論理デバイス名を使用すると、わかりにくい物理デバイス名を簡潔な意味のある名前で表すことができます。デバイスを参照するときは、物理デバイス名の代わりに論理デバイス名を使用できます。
- 汎用デバイス名はデバイス・コードだけで構成され、コントローラ番号とユニット番号が省略されます。
- クラスタ・デバイス名は、デバイスの接続先ノードの名前と物理デバイス名をドル記号 (\$) で区切って表現します。

デバイス名の詳細については、『*OpenVMS システム管理者マニュアル*』と『*OpenVMS ユーザーズ・マニュアル*』を参照してください。

### 5.3.2 アクセスのタイプ

デバイスは共用と非共用のデバイスがあり、共用デバイスは同時に複数のユーザが使用でき、非共用デバイスは 1 人のユーザが使用します。

共用デバイスは、次のアクセス・タイプをサポートします。

読み込み (R)	デバイスからデータを読み込む権限がユーザに与えられます。
書き込み (W)	デバイスにデータを書き込む権限がユーザに与えられます。
物理 (P)	デバイスに対して物理入出力操作を実行する権限がユーザに与えられます。
論理 (L)	デバイスに対して論理入出力操作を実行する権限がユーザに与えられます。
制御 (C)	デバイスの保護要素と所有者を変更する権限がユーザに与えられます。

非共用デバイスは、読み込みアクセス、書き込みアクセス、および制御アクセスのみをサポートします。その他のタイプの操作に必要なアクセス権は、オペレーティング・システムのセキュリティ・ポリシーではなく、デバイス・ドライバによって決まります。

### 5.3.3 入出力操作に必要なアクセス権

デバイスに対する入出力操作に必要なアクセス権はかなり複雑になることがあります。一般的な操作に必要なアクセス権について、以下に説明します。

- \$ASSIGN によるチャンネルの割り当て  
スプールされていない非共用可能デバイスにチャンネルを割り当てるには、読み込みアクセス権、書き込みアクセス権、制御アクセス権、またはこれらの組み合わせが必要です。共用可能なデバイスにチャンネルを割り当てるには、必要なアクセス権はありません。
- \$ALLOC によるデバイスの割り当て  
\$ALLOC によってデバイスを割り当てるには、読み込みアクセス権、書き込みアクセス権、または制御アクセス権が必要です。

- スプールされたデバイスに対する \$QIO

OpenVMS のマウントされたボリュームと同じようにアクセスが処理されます。次の「ファイル指向デバイスに対する \$QIO」を参照してください。

- ファイル指向デバイス (ディスクおよびテープ) に対する \$QIO

ファイル指向デバイスでは、論理入出力機能と物理入出力機能に共通の要素があります。論理入出力機能を実行するには、物理または論理アクセス権とともに、ブロックを読み込む (READLBLK) 場合は読み込みアクセス権、ブロックを書き込む (WRITEBLK) 場合は書き込みアクセス権がそれぞれ必要です。物理入出力機能を実行するには、物理アクセス権とともに、ブロックを読み込む (READPBLK) 場合は読み込みアクセス権、ブロックを書き込む (WRITEPBLK) 場合は書き込みアクセス権がそれぞれ必要です。論理入出力と物理入出力には、それぞれ LOG\_IO 特権と PHY\_IO 特権も必要です。

さらに、必要なアクセス権はボリュームのマウント方法によっても異なります。

- OpenVMS でサポートされるボリューム

ボリュームに対する仮想入出力には、ファイル・クラスまたはボリューム・クラスと同じアクセス権が必要です (5.4 節および 5.10 節を参照)。

- フォーリン (/FOREIGN) としてマウントされたボリューム

仮想読み込み・書き込み機能は、論理入出力に変換されます。その他のすべての機能は、オペレーティング・システムでは処理されず、デバイス・ドライバに送られて処理されます。物理入出力機能には、PHY\_IO 特権も必要です。

- マウントされたボリュームを持たないデバイス

マウントされたボリュームを持たないデバイスにアクセスするには、特権が必要です。

- 非ファイル指向デバイスに対する \$QIO

非ファイル指向デバイスでは、仮想読み込み・書き込み入出力要求が処理前に論理入出力に変換されます。その他の種類のアクセス要求は、OpenVMS では処理されず、デバイス・ドライバに送られて処理されます。

一般的に、非ファイル指向デバイスに必要なアクセス権は、そのデバイスが共用可能かどうかによって異なります。

- 共用可能デバイス

共用可能デバイス (メールボックスなど) では、READVBLK/WRITEVBLK 以外の仮想入出力機能がシステム入出力ドライバ・プログラムによって処理されます。論理入出力機能には、特権またはデバイスに対する論理アクセス権が必要です。物理入出力機能には、特権またはデバイスに対する物理アクセス権が必要です。

– 非共用可能デバイス

非共用可能デバイス（ターミナルやプリンタなど）では，仮想および論理入出力機能を実行するための読み込みまたは書き込みアクセス権だけがチェックされます。物理入出力機能には，特権が必要です。

### 5.3.4 テンプレート・プロファイル

デバイス・クラスは，以下のテンプレート・プロファイルを提供します。

テンプレート名	デバイス・タイプ	所有者 UIC	保護コード
BUS	DC\$_BUS	[SYSTEM]	S:RWPL,O:RWPL,G,W
CARDREADER	DC\$_CARD	[SYSTEM]	S:RWPL,O:RWPL,G,W
COMMUNICATION	DC\$_SCOM	[SYSTEM]	S:RWPL,O:RWPL,G,W
DEFAULT		[SYSTEM]	S:RWPL,O:RWPL,G:RWPL,W:RWPL
DISK	DC\$_DISK	[SYSTEM]	S:RWPL,O:RWPL,G:R,W
MAILBOX	DC\$_MAILBOX	[SYSTEM]	S:RWPL,O:RWPL,G:RWPL,W:RWPL
PRINTER	DC\$_LP	[SYSTEM]	S:RWPL,O:RWPL,G,W
REALTIME	DC\$_REALTIME	[SYSTEM]	S:RWPL,O:RWPL,G:RWPL,W:RWPL
TAPE	DC\$_TAPE	[SYSTEM]	S:RWPL,O:RWPL,G:R,W
TERMINAL	DC\$_TERM	[SYSTEM]	S:RWPL,O:RWPL,G,W
WORKSTATION	DC\$_WORKSTATION	[SYSTEM]	S:RWPL,O:RWPL,G:RWPL,W:RWPL

### 5.3.5 新しいデバイスに対するプロファイルの設定

通常，デバイスのセキュリティ・プロファイルはそのデバイスのタイプに対応するテンプレート・プロファイルから生成されます。ただし，多くの場合テンプレートは変更されます。各種のデバイスにオペレーティング・システムによってプロファイルがどのように割り当てられるかについて，次に説明します。

- システム構成時に作成されるデバイス

システム構成時に CONNECT コマンドと LOAD コマンドによって導入されるデバイス（たとえば，擬似デバイスやワークステーション）は，それぞれのデバイス・タイプに対応するテンプレートからプロファイルを取得します。

- ディスクとテープ

ディスクとテープは，DISK テンプレートと TAPE テンプレートからそれぞれプロファイルを取得します。デバイスがクラス内で可視状態になると，デバイスのプロファイルは（変更も含めて）システムの再起動にまたがって維持されます。デバイスがセキュリティ・プロファイルを取得した後で DISK または

TAPE テンプレート・プロファイルを変更しても、そのデバイスには適用されません。このような場合は、DCL の SET SECURITY コマンドを使用して個々のオブジェクト・プロファイルを再設定する必要があります (4.2.4 項を参照)。

- テンプレート・デバイスからクローン化されたデバイス

テンプレート・デバイスからクローン化されたデバイス (たとえば、イーサネット・デバイス) は、クローンの元になったテンプレート・デバイスのセキュリティ・プロファイルを継承します。テンプレート・デバイスは、自動構成の処理中にロードされます。このとき、テンプレート・デバイスのプロファイルは、当該デバイスに対応するプロファイル・テンプレートから取得されます。

- メールボックス

メールボックス・デバイスは、MAILBOX テンプレート・プロファイルに変更が加えられたものを継承します。システムはテンプレートに変更を加えて、メールボックス作成プロセスの UIC を所有者とし、メールボックス作成 (\$CREMBX) システム・サービスの *promsk* 引数の値を (値が 0 でなければ) 保護コードとして設定します。

OpenVMS オペレーティング・システムの旧バージョンとの互換性を維持するため、MAILBOX テンプレートには (すべてのアクセスを許可する) 保護コード 0 が設定されています。アプリケーションによっては、テンプレートの値よりも制限の強いデフォルト値が必要になる場合があります。メールボックスへのアクセスを制限する場合は、アクセス制限の強化により、原因を把握しにくいアプリケーション障害が発生する恐れがあることに注意してください。

- ターミナル

ターミナル・デバイスは、TERMINAL テンプレート・プロファイルに変更を加えたものを継承します。

---

注意

---

OpenVMS バージョン 7.2-1 以前では、すべての擬似ターミナル (FT) デバイスの保護コードがドライバによって (S:RWLP,O:RWLP,G,W) に設定されていました。OpenVMS バージョン 7.3 以降では、FTA0 デバイスにのみこの強制保護が設定されます。このため、システム管理者は FTA0 デバイスの保護をブート・プロセスの中で後から変更することができます。変更された保護は、(ACL など、SECURITY クラスの DEVICE TERMINAL テンプレート・プロファイルに存在する他の設定値とともに) FTA0 から、以降に新たに作成される FT デバイスに継承されます。

システム管理者は、FTA0 を手動で変更するか、SYSTARTUP\_VMS.COM コマンド・プロシージャを変更できます。次に例を示します。

```
$ SET SECURITY/CLASS=DEVICE -  
_$_ /PROTECTION=(S:RWLP,O:RWLP,G:RW,W:R)
```

FTA0:

FTA0 のデバイス保護を変更しない場合は、バージョン 7.3 以前の OpenVMS と動作が変わりません。この動作では、FT 擬似ターミナル・デバイスを除くすべてのターミナルがデバイスの保護とその他のセキュリティ特性を TERMINAL テンプレート・プロファイルから継承します。すべての FTA 擬似ターミナル・デバイスは、デフォルトでは (S:RWLP,O:RWLP,G:W) に設定されているデバイスの保護を FTA0 から継承します。ACL などのその他の設定値は、TERMINAL テンプレート・プロファイルから継承されます。これにより、既存のアプリケーションとの互換性が確保されます。

---

ユーザがターミナルにログインすると、所有者がターミナルにログインしたプロセスの UIC になるようにプロファイルが変更されます。ターミナルの元のセキュリティ・プロファイルは、ユーザがログアウトした時点で復元されます。

### 5.3.6 必要な特権

スプールされたデバイスに対するすべての論理・物理入出力には、特権が必要です。

LOG\_IO 特権を有するユーザ・プロセスは、入出力要求キュー登録 (\$QIO) システム・サービスを実行して、論理レベルの入出力操作を実行できます。LOG\_IO 特権は、パーマネント・ターミナル要素の設定など、特定のデバイス制御機能にも必要です。

PHY\_IO 特権を有するユーザ・プロセスは、入出力要求キュー登録 (\$QIO) システム・サービスを実行して、物理レベルの入出力操作を実行できます。PHY\_IO 特権を取得すると、LOG\_IO 特権も付与されます。

パーマネント・メールボックスを作成したり、メールボックスを削除の対象に指定したりするには、PRMMBX 特権が必要です。

### 5.3.7 実行される監査の種類

以下のイベントのタイプを監査できます。ただし、セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	非共用可能デバイスについては、プロセスが \$ASSIGN を呼び出したとき。共用可能デバイスについては、プロセスが \$QIO を呼び出したとき。
作成 (C)	プロセスがメールボックスのような仮想デバイスを作成したとき。
削除 (D)	プロセスがメールボックスのような仮想デバイスを削除したとき。

### 5.3.8 オブジェクトの永続性

クラスタ全体のディスクとテープのプロファイルはオブジェクト・データベース (VMS\$OBJECTS.DAT) に保存されますが、他のオブジェクトのプロファイルはシステムが起動されるたびに再設定する必要があります。

## 5.4 ファイル

ファイルは、固定サイズ (512 バイト) のデータ・ブロックで構成される名前付き配列であり、属性のセットが関連付けられています。OpenVMS システムでは、ファイル・クラスにデータ・ファイルとディレクトリ・ファイルの 2 つが含まれます。Files-11 オン・ディスク構造レベル 2 または 5 (ODS-2 または ODS-5) のボリューム上に保存された個々のディスク・ファイルには、完全なセキュリティ保護が提供されます。テープ・ファイルは、ボリューム上の保護コードによって一括保護されますが、個別には保護されません。

ファイル・オブジェクトと他の保護オブジェクトには重要な違いが 1 つあります。ファイルは他のどのオブジェクト・クラスよりも柔軟性があるため、テンプレートからプロファイルを取得しません。5.4.5 項では、オペレーティング・システムがプロファイルを割り当てるときに適用する規則について説明します。

### 5.4.1 命名規則

ファイル名は、1 ~ 255 文字の文字列で指定します。詳細については、『*OpenVMS ユーザーズ・マニュアル*』を参照してください。

### 5.4.2 アクセスのタイプ

ファイル・クラスは、次のアクセス・タイプをサポートします。



---

読み込み (R)	ディスク・ファイルの読み込み，印刷，コピーを行う権限がユーザに与えられます。ディレクトリ・ファイルについては，ファイルの読み込みと一覧表示，およびワイルドカード文字を含むファイル名を使ってファイル検索を行う権限がユーザに与えられます。読み込みアクセスには実行アクセスも含まれます。
書き込み (W)	ファイルへの書き込みとファイル内容の変更を行う権限がユーザに与えられます (ファイルを削除する権限は与えられません)。書き込みアクセス権により，ファイル内容を記述するファイル要素を変更することが許可されます。また，書き込みアクセス権により，既存ファイルの主要名を新しく作成することが許可されます。ディレクトリ・ファイルについては，書き込みアクセス権によって，ファイルのカatalog内にあるエントリを作成または削除する権限がユーザに与えられます。
実行 (E)	実行可能なプログラム・イメージが格納されたファイルまたは DCL コマンド・プロシージャを実行する権限がユーザに与えられます。ディレクトリ・ファイルについては，実行アクセス権によって，名前を指定してファイルを検索する権限がユーザに与えられます。
削除 (D)	ファイルを削除する権限がユーザに与えられます。ファイルを削除するには，対象ファイルに対する削除アクセス権とそのファイルが置かれているディレクトリに対する書き込みアクセス権が必要です。ファイルの主要名を削除または変更する場合も，削除アクセス権が必要です。
制御 (C)	保護コードと ACL を変更する権限がユーザに与えられます。所有者を変更するには，次のいずれかの条件を満たす必要があります。 <ul style="list-style-type: none"> <li>古い所有者識別子と新しい所有者識別子の両方を保持していること。</li> <li>オブジェクトを所有する識別子に Resource 属性が割り当てられており，オブジェクトの ACL によってオブジェクトに対する制御アクセスも許可されていること。</li> <li>システム・ユーザとして登録され，SYSPRV 特権または BYPASS 特権を持っているか，該当するファイルまたはディレクトリが格納されているボリュームの所有者と同じ UIC を持っていること。</li> <li>GRPPRV 特権を持っており，オブジェクトの所有者と同じグループの UIC も持っていること。</li> </ul>

---

### 5.4.3 必要なアクセス権

ファイル・アクセスには，次の条件が適用されます。

- 一般的な規則

ファイルにアクセスするには、そのファイルおよびそのファイルが置かれているボリュームへのアクセスが許可されている必要があります。名前を指定してファイルにアクセスするには、対象ファイルが格納されているディレクトリに対する読み込みまたは実行アクセス権が必要です。ディレクトリまたはファイルへのアクセス権を評価する前に、ボリュームへのアクセス権を評価する必要があります。ディレクトリ・ファイルの保護によって、ディレクトリ内のファイルへのアクセスが制限される場合があります。このため、ユーザ・グループに対してファイルへのアクセスが許可されていても、そのファイルが置かれているディレクトリに対して適切なアクセス権がなければ、名前を指定したファイル・アクセスができないことがあります。

---

#### 注意

---

ユーザは、ファイル識別子によってファイルにアクセスできます。ファイル識別子によってファイルにアクセスすると、ディレクトリ・ファイルの保護が無視されます。このため、ファイルに対するアクセスの制御を、ディレクトリ・ファイルの保護にのみ頼るべきではありません。

---

- **書き込みアクセスの場合**

ファイルへの書き込みを行うには、読み込みアクセス権と書き込みアクセス権の両方が必要です。

- **ファイルの所有権の変更**

ファイルの所有権を変更するには、制御アクセス権を持ち、Resource 属性が割り当てられた古い識別子と新しい識別子の両方を保持する必要があります。ユーザ自身の UIC 識別子には、常に Resource 属性が割り当てられています。

#### 5.4.4 作成時の要件

ファイルを作成するときは、ユーザが次の条件を満たしているかどうかチェックされます。

- 十分なディスク領域があること。これには、使用可能なディスク・ブロックと必要なディスク・クォータ(クォータ機能が有効な場合)の両方が含まれます。
- 古いバージョンのファイルに対する読み込みアクセス権と書き込みアクセス権を持っていること。ファイルにゼロ以外のバージョン制限があり、新しいバージョンがこの値を超える場合は、最も古いバージョンのファイルに対する削除アクセス権も必要です。
- ファイルが作成されるディレクトリに対する書き込みアクセス権を持っていること。

- ファイルが格納されるボリュームに対する読み込み，書き込み，および作成の各アクセス権を持っていること。

#### 5.4.5 プロファイルの割り当て

新しいファイルの所有者，保護コード，および ACL のソースはさまざまです。新しいファイルの所有権の割り当ては，保護および ACL とは独立に行われます。

##### 5.4.5.1 所有権の割り当て規則

次のいずれかの条件に該当する場合，ユーザはファイルの所有者として識別子を割り当てることができます。

- 識別子がユーザのプロセス UIC と一致する。
- Resource 属性を含んだ識別子を持っている。
- GRPPRV 特権を持っており，識別子のグループ番号が自分の UIC グループと一致する。
- SYSPRV 特権を持っている。

ファイルは，ユーザが割り当ててることを許可された以下のソースのうち，最初に適用可能なソースから所有者識別子を受け取ります。

- CREATE コマンドまたは COPY コマンドを使ったファイル作成時の，/OWNER\_UIC 修飾子による明示的な所有者の割り当て
- 前のバージョン
- 親ディレクトリ
- プロセスの UIC

資源識別子によってファイルやディレクトリがどのように所有されるかについては，8.8.1.2 項を参照してください。

##### 5.4.5.2 保護コードと ACL の割り当て規則

新しいファイルの保護コードと ACL のソースは，所有権の場合とほぼ同じであり，適用順序も同じです。ファイルの保護コードと ACL は，次のいずれかのソースから割り当てられます。

###### 1. 作成時の明示的な要素の割り当て

ファイルは，CREATE コマンドまたは COPY コマンドを使用して作成します。ディレクトリの場合は，CREATE/DIRECTORY コマンドを使用します。

ファイルの作成時に保護コードを割り当てするには，COPY コマンドまたは CREATE コマンドに /PROTECTION 修飾子を追加します。ファイルを作成したら，SECURITY/ACL コマンドを使用して ACL を追加することができます。

たとえば、次のコマンドはデバイス USE1 からデフォルトのディスク・ディレクトリにファイルをコピーします。新しく作成される PAYSORT.DAT ファイルの保護が保護コードによって定義され、システム UIC を持つユーザがこのファイルの読み込みと書き込みを実行できるようになります。所有者はすべてのタイプのアクセス権を持ち、所有者と同じグループの他のユーザはファイルの読み込みと書き込みを実行できます。それ以外のすべてのユーザは、この保護コードではアクセス権を付与されません。

```
$ COPY USE1:[PAYDATA]PAYROLL.DAT PAYSORT.DAT -  
_ $ /PROTECTION=(SYSTEM:RW, OWNER:RWED, GROUP:RW, WORLD)
```

2. このファイルの古いバージョンのプロファイル (存在する場合)

ユーザがファイルの新しいバージョンを作成すると、(当然ながら、明示的に割り当てた場合を除いて) 古いバージョンの保護コードと ACL が新しいバージョンに設定されます。

3. 親ディレクトリのデフォルトの保護用 ACE とデフォルト ACL

明示的な割り当てとファイルの古いバージョンのいずれもなければ、オペレーティング・システムがファイルの作成先のディレクトリを調べます。

データ・ファイルの場合は、システムがデフォルトの保護用 ACE を探し、その ACE で指定されている保護コードを割り当てます。具体例については、4.5.6 項を参照してください。ディレクトリの ACL 内に Default 属性を持つ ACE がある場合は、その ACE もファイルに継承されます。具体例については、4.4.7 項を参照してください。

ディレクトリ・ファイルの場合は、システムが親ディレクトリの保護コードから削除アクセス権を除いたものを割り当てます。ディレクトリが最上位ディレクトリだった場合は、マスタ・ファイル・ディレクトリ (MFD) の保護が適用されます。新しく作成されるサブディレクトリは、親ディレクトリの ACL と Default 属性を持つ ACE を継承します。Nopropagate 属性を持つ ACE のみが除外されます。

4. コマンドを発行したプロセスの UIC および保護のデフォルト値

ディレクトリの ACL にデフォルトの保護用 ACE がない場合は、プロセスのデフォルトの保護が適用されます。この値は、RMS\_FILEPROT システム・パラメータによって設定され、ユーザのログイン時にプロセスに割り当てられます。ただし、ログイン時に割り当てられた値は、DCL の SET PROTECTION/DEFAULT コマンドによって変更できます。たとえば、このコマンドをログイン・コマンド・プロシージャに入れて、デフォルトの保護を設定することができます。プロセスのデフォルトの保護を表示するには、DCL の SHOW PROTECTION コマンドを使用します。

5. 上記のいずれか (ファイルを作成したユーザへの対応を含む)

資源識別子によって所有されているディレクトリにファイルを作成し、自分が Resource 属性が割り当てられた識別子を持っている場合、新しいファイルは他のファイルと同じ方法で保護コードと ACL を継承します。

オペレーティング・システムは、新しいファイルへのアクセス権を作成者に与えるために、ファイルの ACL を変更する場合があります。ディレクトリの ACL に作成者 ACE がある場合、作成者がファイルに対して持つアクセス権はその ACE によって決まります。作成者 ACE にアクセス権が指定されていない場合、追加の ACE は作成されません。このような ACE がない場合は、オペレーティング・システムによってファイルの ACL に ACE が追加され、制御アクセス権とファイルの保護コードの所有者フィールドに指定されているアクセス権が作成者に与えられます。

#### 5.4.5.3 COPY コマンドと RENAME コマンドの使用

COPY コマンドの出力ファイルは、新規作成ファイルとして扱われるため、新しいセキュリティ・プロファイルが割り当てられます。入力ファイルのセキュリティ・プロファイルは関係しません。

これに対し、ファイルの名前を変更した場合、当該ファイルはデフォルトでは既存のセキュリティ・プロファイルを維持します。ファイルを新しく作成したときのように新しいセキュリティ・プロファイルを割り当てるには、DCL の RENAME/INHERIT\_SECURITY コマンドを使用します。これにより、ファイルに新しいセキュリティ・プロファイルが割り当てられます。

セキュリティ・プロファイルの割り当て方法については、5.4.5.1 項と 5.4.5.2 項で説明します。

#### 5.4.6 実行される監査の種類

以下のイベントのタイプを監査できます。ただし、セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	プロセスがファイルのオープン、読み込み、書き込み、実行、またはファイル属性の照会を行うとき。
作成 (C)	プロセスがファイルを作成するとき。
アクセス解除 (D)	プロセスがファイルをクローズするとき。
削除 (D)	プロセスがファイルを削除するとき。

### 5.4.7 ディスク領域再割り当て時の情報の保護

通常のファイル保護メカニズムでは、ファイルにアクセスするユーザを制御することはできますが、ファイルを削除した後に残る古いデータをどう保護するかという問題には対処できません。

ファイルを削除すると、ファイルのヘッダがディレクトリから消去されますが、ファイルの内容は他のデータによって上書きされるまでディスク上に残っています。このようにデータがディスク上に存在するため、削除またはパージされたファイルの情報をディスク・スキベンジングから保護する必要があります。

OpenVMS オペレーティング・システムでは、ディスク・スキベンジングの問題を次の 2 つの手法を組み合わせで解決します。

- 割り当て前のディスク・ブロックの上書き
- 割り当てられるブロックに対するハイウォーター・マークの設定

#### 5.4.7.1 ディスク・ブロックの上書き

セキュリティ管理者やユーザは、ボリューム上の個々のファイルやボリューム全体に対して除去パターンを適用することができます。除去パターンは、ファイルを削除またはパージしたときにファイル全体に書き込まれる、反復するビット列です。

セキュリティ管理者は、ボリュームの初期化時に次のように /ERASE 修飾子を指定することにより、ボリューム上のすべてのブロックを除去パターンで初期化することができます。

INITIALIZE/ERASE デバイス名[:] ボリューム・ラベル

ボリュームがマウントされていれば、セキュリティ管理者は次のように /ERASE\_ON\_DELETE 修飾子を指定することにより、ファイルが削除されたときにそのファイルが占有していた領域に対して自動的に除去パターンを適用できます。

SET VOLUME/ERASE\_ON\_DELETE デバイス名[:]

この方法は既存のファイルに影響を与えないことに注意してください。

また、ファイルごとに除去パターンを指定する方法もあります。この場合は、DCL の SET FILE, DELETE, PURGE の各コマンドを入力するときに /ERASE 修飾子を使用するように、セキュリティ管理者がユーザに指示します。

セキュリティ管理者は、\$ERAPAT システム・サービスを使って除去ルーチンを作成することもできます。除去ルーチンは、ディスク・ブロックを除去するときに使用する除去パターンとパスの回数をシステムに対して指定します。

#### 5.4.7.2 ハイウォーター・マークの設定

OpenVMS オペレーティング・システムは、ファイルにディスク・ブロックを割り当てるときに自動的にハイウォーター・マークを設定します。ハイウォーター・マークは、ディスク上の割り当て領域内のどの位置までファイルが書き込まれたかを示します。ファイルの先頭からハイウォーター・マークまでのすべてのブロックは、ファイルへの割り当て後に書き込まれたことが保証されます。ユーザは、ハイウォーター・マークを超える読み込みが許可されないため、自分が実際に書き込んでいない古いデータを読み込むことができません。

より堅実でありながらもコストのかかる方法として、割り当ての前にすべてのディスク・ブロックを除去する方法があります。この割り当て時除去の方法は、ファイルが公開されていて、任意の方法での共用アクセスまたは非順次アクセスが許可される場合に使用します。割り当て時にブロックを除去すると、新しく割り当てられて除去された領域の最後にそのファイルのハイウォーター・マークが設定されます。

デフォルトでは、ボリュームの初期化時にハイウォーター・マーク処理が有効になります。セキュリティ管理者は、DCL の SET VOLUME/NOHIGHWATER\_MARKING コマンドを使用して、特定のボリュームに関してハイウォーター・マーク処理を無効にできます。

#### 5.4.7.3 ファイル内のデータのアクセス制御

ファイル・システムによってファイルにディスク・ブロックが割り当てられると、ユーザはそれらのディスク・ブロックをいつでも読み込んだり書き込んだりできます。ハイウォーター・マークはファイルの物理的な終わりを示し、ユーザはそれを越えて読み込むことができません。しかし、アプリケーションは論理的なファイル終端マークの位置を変更し、ファイルの論理的な終端から物理的な終端までの間の領域にデータを残すことができます。その場合、論理的なファイル終端マークに関係なく、ファイル・データの任意のブロックを読み込めます。

割り当てられたディスク・ブロックの管理は、アプリケーションにほとんど任されています。たとえば、OpenVMS RMS サービスは、論理的なファイル終端の位置を現在のレコードの先頭に再設定することによって順次ファイルを短縮します。しかし、ファイル終端の位置からファイルの物理的な終端までの領域の割り当ては解放されず、ファイル終端の位置からファイルの物理的な終端までのレコードが除去パターンによって上書きされることもありません。

そのため、ファイルに書き込まれたブロックはファイル終端マークに関係なく読み込むことができます。ファイルの論理的な終端と物理的な終端の間にあるデータを除去するには、削除したいデータをアプリケーション・プログラムが上書きする必要があります。OpenVMS システムでは、通常、DCL の COPY コマンドを使ってファイルの新しいバージョンを作成することにより、これを実現します。

#### 5.4.8 ファイル・セキュリティの最適化に関する推奨事項

ファイルとディレクトリを保護するため、次の予防措置を講じてください。

- ファイルを定期的にバージします。不要なファイルを削除します。これにより、ディレクトリの数が最小限に抑えられ、ファイルに対する保護と所有権の定期的なチェック作業が簡略化されます。
- DCL の DIRECTORY/SECURITY コマンドを定期的に行って、ファイルの所有権、保護コード、および ACL を監視します。十分な特権の獲得に成功したユーザがファイルの保護や所有権を変更して、ファイルへの即座および将来のアクセスを許可する可能性があります。これらのチェックを頻繁に行うことで、ファイルの保護や所有権に対する説明の付かない変更を検出して報告することができます。
- 自分のメール・ファイルの保護には特に注意します。通常、メール・ファイルにアクセスできるのは自分自身とシステム（メールの配送とバックアップのため）のみです。
- ファイルに ACL を設定するときは、指定した識別子をどのユーザが保持しているかを正確に知っておく必要があります。これについては、通常、サイトのセキュリティ管理者に問い合わせる必要があります。
- サイトのセキュリティ管理者のアドバイスに従って、ディスク・スキベンジングを防止します。自分のファイルの一部または全部について、SET FILE, DELETE, PURGE の各コマンドを実行するときに /ERASE 修飾子を使用するよう要請される場合もあります。
- コマンド・プロシージャや実行可能ファイルを格納しているファイルおよびディレクトリは必ず保護します。これらのディレクトリやファイルに対する書き込みアクセスの許可は慎重に制御します。これは、強力な特権を持っている場合や機密ファイルへのアクセス権を持っている場合、特に重要です。

---

#### 注意

---

他のユーザから受け取ったコマンド・プロシージャやプログラムは、内容を確認するまで実行しないようにします。コマンド・プロシージャやプログラムが特別な特権や機密ファイルに対するアクセス権を行使するものかどうかを確認します。そのソフトウェアを特権のないアカウントでテストします。別の目的を装って提供され、実際にはユーザの防御を突破してシステム・セキュリティを損ねることを目的とするプログラムやコマンド・プロシージャには、トロイの木馬プログラムと呼ばれるものがあります。

---



## 5.5 グローバル・セクション

OpenVMS のメモリ管理サービスでは、グローバル・セクションと呼ばれる共用のメモリ・ページを使ったプロセス間通信が可能です。グローバル・セクションを利用することで、複数のプロセスが同じページをそれぞれのプロセスの仮想アドレス空間にマップして、コードやデータのページを共用することができます。

グローバル・セクションは、ディスク・ファイルへのアクセスを提供したり（ファイルによるバックアップのあるグローバル・セクションと呼ばれる）、動的に作成されたストレージへのアクセスを提供したり（ページ・ファイルによるバックアップのあるグローバル・セクションと呼ばれる）、特定の物理メモリへのアクセスを提供したり（ページ・フレーム番号 (PFN) グローバル・セクションと呼ばれる）でできます。グローバル・セクション・オブジェクトは、一時的なオブジェクトにも永続的なオブジェクトにもなります。

OpenVMS オペレーティング・システムでは、次の 2 種類のグローバル・セクション・オブジェクトがサポートされます。

- グループ・グローバル・セクションは、同じグループ内のすべてのプロセスが使用できる共用可能なメモリ・セクションです。
- システム・グローバル・セクションは、システム内のすべてのプロセスが使用できる共用可能なメモリ・セクションです。

### 5.5.1 命名規則

オブジェクトの名前は、1 ~ 44 文字の文字列で指定します。グループ・グローバル・セクションの名前は、ユーザの UIC グループ番号で修飾します。

### 5.5.2 アクセスのタイプ

グローバル・セクション・クラスは、次のアクセス・タイプをサポートします。

読み込み (R)	読み込みアクセスのためにセクションをマップする権限がユーザに与えられます。
書き込み (W)	書き込みアクセスのためにセクションをマップする権限がユーザに与えられます。
実行 (E)	読み込みアクセスのためにセクションをマップする権限がユーザに与えられます。エグゼクティブ・モードまたはカーネル・モードで実行されているソフトウェアのみが、このアクセス権を要求できます。
制御 (C)	PFN グローバル・セクションとページ・ファイルによるバックアップのあるグローバル・セクションの保護要素を変更する権限がユーザに与えられます。

### 5.5.3 テンプレート・プロファイル

ファイルによるバックアップのあるグローバル・セクションは、対応するディスク・ファイルと同じセキュリティ・プロファイルが適用されます。バックアップ・ファイルのプロファイルが変更されると、グローバル・セクションのプロファイルも自動的に変更されます。ファイルによるバックアップのあるグローバル・セクションの保護要素を変更するには、代わりにバックアップ・ファイルを変更する必要があります。

グローバル・セクション・クラスは、次のテンプレート・プロファイルを提供します。このテンプレートは [0,0] という所有者 UIC を割り当てますが、この値は一時的なものです。オブジェクトが作成されると、作成したプロセスの UIC の対応するフィールド内の値が 0 の代わりに設定されます。

タイプ	テンプレート名	所有者 UIC	保護コード
システム	DEFAULT	[0,0]	S:RWE,O:RWE,G:RWE,W:RWE
グループ	DEFAULT	[0,0]	S:RWE,O:RWE,G:RWE,W:RWE

これらのテンプレートは、\$CRMPSC の *prot* 引数に指定された値に従って変更されます。ファイルによるバックアップのあるセクションでは、*prot* 引数は無視されます。

OpenVMS オペレーティング・システムの旧バージョンとの互換性を維持するため、DEFAULT テンプレートにはワールド・カテゴリ(すべてのユーザ)によるアクセスを許可する保護コードが設定されています。アプリケーションによっては、このテンプレートよりも厳しい設定のデフォルト値が必要になる場合があります。グローバル・セクションへのアクセスを制限する場合は、アクセス制限の強化により、原因を把握しにくいアプリケーション障害が発生する恐れがあることに注意してください。

### 5.5.4 必要な特権

システム・グローバル・セクションを作成または削除するには、SYSGBL 特権が必要です。ページ・フレーム・セクションを作成または削除するには、PFNMAP 特権が必要です。パーマネント・グローバル・セクションを作成または削除するには、PRMGBL 特権が必要です。

### 5.5.5 実行される監査の種類

以下のイベントのタイプを監査できます。ただし、セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
作成 (C)	セクション作成およびマップ・システム・サービス (\$CRMPSC) によって、ページ・ファイルによるバックアップのあるグローバル・セクションまたは PFN グローバル・セクションが作成されたとき。
アクセス (A)	\$CRMPSC またはグローバル・セクションのマップ・システム・サービス (\$MGBLSC) によって、ページ・ファイルによるバックアップのある既存グローバル・セクションまたは既存の PFN グローバル・セクションがアクセスされたとき。ファイルによるバックアップのあるグローバル・セクションに対するアクセスは、ファイル・アクセスとして監査されます。
アクセス解除 (D)	プロセスの仮想アドレス空間が再設定または削除されて、イメージまたはプロセスがランダウンされたとき。
削除 (D)	PRMGBL 特権、PFNMAP 特権、または SYSGBL 特権 (システム・グローバル・セクションの場合) を持つプロセスがパーマネント・グローバル・セクションを削除すると、そのイベントは特権の使用として監査されます。

### 5.5.6 オブジェクトの永続性

グローバル・セクションとそのセキュリティ・プロファイルは、システムが起動するたびに再設定する必要があります。

## 5.6 論理名テーブル

論理名の割り当ては、論理名テーブルで管理されます。論理名テーブルには 1 つのプロセスのみがアクセスできます。ただし、親テーブルが共用可能な場合は共用できます。共用可能な名前テーブルはすべて、システム・ディレクトリ・テーブル (LNM\$SYSTEM\_DIRECTORY) にリストされています。オペレーティング・システムによって保護されるのは、共用可能な論理名テーブルです。

### 5.6.1 命名規則

論理名テーブルの名前は、1 ~ 32 文字の文字列で指定します。

### 5.6.2 アクセスのタイプ

論理名テーブル・クラスは、次のアクセス・タイプをサポートします。

読み込み (R)	テーブル内の論理名を検索 (変換) する権限がユーザに与えられます。
書き込み (W)	テーブル内の論理名を作成または削除する権限がユーザに与えられます。

作成 (C)	子孫の論理名テーブルを作成する権限がユーザに与えられます。これには、子孫の論理名テーブルを作成するときに親の論理名テーブルに割り当てられた動的メモリのサブセットを使用する権限も含まれます。
削除 (D)	テーブルを削除する権限がユーザに与えられます。
制御 (C)	テーブルの保護要素と所有者を変更する権限がユーザに与えられます。

### 5.6.3 テンプレート・プロファイル

論理名テーブル・クラスは、次のテンプレート・プロファイルを提供します。このテンプレートは [0,0] という所有者 UIC を割り当てますが、この値は一時的なものです。オブジェクトが作成されると、作成したプロセスの UIC の対応するフィールド内の値が 0 の代わりに設定されます。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[0,0]	S:RW,O:RW,G:R,W:R
GROUP	[0,*]	S:RWCD,O:R,G:R,W
JOB	[0,0]	S:RWCD,O:RWCD,G,W

### 5.6.4 必要な特権

論理名テーブルへの読み込み・書き込みアクセスは、グループ論理名テーブルの場合は GRPNAM 特権によって、システム論理名テーブルの場合は SYSNAM 特権によってそれぞれ許可されます。

システム・ディレクトリから共用テーブルを削除するには、SYSNAM 特権が必要です。グループ・ディレクトリから論理名テーブルを削除するには、GRPNAM 特権が必要です。親論理名テーブルを削除すると、その子孫にあたる論理名テーブルはすべて削除されます。

内部モードの論理名や論理名テーブルを作成または削除するには、SYSNAM 特権（または内部モードにすること）が必要です。

### 5.6.5 実行される監査の種類

以下のイベントを監査できます。ただし、セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	名前を変換するとき、名前または子孫のテーブルを作成するとき、あるいは名前や子孫のテーブルを削除するとき。
作成 (C)	テーブルを作成する権限を得るために親テーブルにアクセスしたとき、またはテーブルそのものを作成したとき。

### 5.6.6 オブジェクトの永続性

論理名テーブルとそのセキュリティ・プロファイルは、システムがリブートされるたびに再設定する必要があります。

## 5.7 キュー

キューは、処理を待つジョブの集合です。一般に、キューには汎用キューと実行キューの2種類があります。汎用キューでは処理が実行されません。実行キューには、実行キューが使用可能になったときに実行キュー上で実行されるジョブが保持されます。実行キューには、バッチ・キュー、プリンタ・キュー、サーバ・キュー、およびターミナル・キューがあります。

### 5.7.1 命名規則

キュー名は、1 ~ 31 文字の文字列で指定します。この文字列には、任意の英数字、ドル記号 (\$)、およびアンダスコア (\_) を含めることができます。

### 5.7.2 アクセスのタイプ

キュー・クラスは、次のアクセス・タイプをサポートします。

読み込み (R)	キューまたはキュー内のジョブのセキュリティ要素を参照する権限がユーザに与えられます。
登録 (S)	キュー内にジョブを置く権限がユーザに与えられます。
削除 (D)	キュー内のジョブを削除したり、ジョブの要素を変更したりする権限がユーザに与えられます。
管理 (M)	キュー内の任意のジョブに影響を与える権限がユーザに与えられます。キューの起動、停止、削除、およびキューの状態やセキュリティに関係しない要素の変更が可能です。
制御 (C)	キューの保護要素と所有者を変更する権限がユーザに与えられます。

#### 注意

プロセスは、保護コードによって読み込みおよび書き込みアクセス権を与えられると、そのプロセスが操作できるのはキューに登録されたその

プロセスのジョブのみです。一方，ACL によって読み込みおよび書き込みのアクセス権が与えられた場合，プロセスはキューに登録されているすべてのジョブを操作できます。

### 5.7.3 テンプレート・プロファイル

キュー・クラスは，次のテンプレート・プロファイルを提供します。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[SYSTEM]	S:M,O:D,G:R,W:S

### 5.7.4 必要な特権

キュー・マネージャを停止または開始するには，SYSNAM 特権と OPER 特権が必要です。OPER 特権は，キューの作成，キューの削除，およびシンビオントの定義の変更に必要です。

### 5.7.5 実行される監査の種類

以下のイベントを監査できます。ただし，セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	ジョブがキューに登録されたとき，およびジョブとキューのいずれかが変更されたとき。
作成 (C)	キューが初期化されたとき。
削除 (D)	プロセスがキューからジョブを削除したとき，またはキュー自体が削除されたとき。キュー削除の監査を有効にするには，キューに対する管理 (M) アクセスの監査を有効にします。

ファイルとキューの両方のアクセス監査を有効にすると，1 回のキュー操作で複数の監査メッセージが生成される場合があります。これは，1 回の操作中にオペレーティング・システムがアクセス・チェックを複数回実行するためです。たとえば，システムはプリント・キューでジョブを実行する前に，ファイルに対するユーザの読み込みアクセス権の有無を確認し，そのファイルを印刷する前に読み込みアクセス権の有無を再度確認します。

### 5.7.6 オブジェクトの永続性

キューはパーマネント・オブジェクトです。キューは，セキュリティ・プロファイルとともにシステム・キュー・データベースに保存されます。

## 5.8 資源ドメイン

共用資源にアクセスするプロセスは、ロック・マネージャのサービスを使ってアクセス権を調整できます。プロセスはこれらのサービスを使用して、資源(ファイルやデータ構造など)と名前の関連付け、その資源に対するアクセス権のアービトレーション、およびロック値ブロックによる限定的な情報の交換を行います。ロックをかけることができる資源を登録するネームスペースを、資源ドメインと呼びます。

プロセスがロックの獲得と解除、および資源ドメイン内の資源に対応する値ブロックの読み書きを行うには、資源ドメインのメンバになる必要があります。プロセスは、システム・ドメインとグループ・ドメインに暗黙で参加していますが、他のドメインには \$SET\_RESOURCE\_DOMAIN システム・サービスの呼び出しによって明示的に参加します。ドメイン内のすべてのロックおよび値ブロックへのアクセスは、ドメイン自体へのアクセス権によって制御されます。

### 5.8.1 命名規則

\$SET\_RESOURCE\_DOMAIN に指定する資源ドメインは、ロングワードのバイナリ値として表現されます。しかし、資源ドメイン・オブジェクトの名前は大括弧 [] または角括弧 <> で囲んだ(8進数として解釈される)資源番号を含む文字列です。また、資源ドメイン・オブジェクトの名前を大括弧または角括弧で囲んだ識別子として表現することもできます。この識別子は UIC 値に変換され、UIC のグループ・フィールドが資源ドメイン番号として使用されます。

### 5.8.2 アクセスのタイプ

資源ドメイン・クラスは、次のアクセス・タイプをサポートします。

読み込み (R)	ドメイン内のロック値ブロックを読み込む権限 (\$GETLKI システム・サービスを使用してロック値ブロックを取得する権限を含む) がユーザに与えられます。
書き込み (W)	ドメイン内のロック値ブロックに書き込む権限がユーザに与えられます。
ロック (L)	\$ENQ によるロックの適用、\$DEQ によるロックの解除、および \$GETLKI によるロック・データベースに関する情報の取得を行う権限がユーザに与えられます。
制御 (C)	資源ドメインの保護要素を変更する権限がユーザに与えられます。

### 5.8.3 テンプレート・プロファイル

資源ドメイン・クラスは、次のテンプレート・プロファイルを提供します。このテンプレートは [n,\*] (n は資源ドメインの番号) という所有者 UIC を割り当てます。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[n,*]	S:RWL,O:RWL,G:RWL,W

#### 5.8.4 必要な特権

SYSLCK 特権は、システム資源ドメイン (ドメイン 0) に対するロック・アクセスを許可します。

#### 5.8.5 実行される監査の種類

以下のイベントを監査できます。ただし、セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	プロセスが \$SET_RESOURCE_DOMAIN または \$ENQ を呼び出してドメインに参加したとき。
作成 (C)	プロセスが初めて資源ドメインに参加したとき。
アクセス解除 (D)	プロセスが \$SET_RESOURCE_DOMAIN を呼び出したとき、またはイメージがプロセスがランダウンされたとき。

#### 5.8.6 オブジェクトの永続性

資源ドメインとそのセキュリティ・プロセスは、どちらも SYS\$SYSTEM:VMS\$OBJECTS.DAT に保存されます。

### 5.9 セキュリティ・クラス

セキュリティ・クラスは、保護オブジェクトのすべてのクラスの親であり、各種オブジェクト・クラスに関連付けられているテンプレート・プロファイルを保護します。セキュリティ・クラスの各オブジェクトは、次の情報を持っています。

- オブジェクト名
- クラスの新しいオブジェクトに対するセキュリティ・プロファイル
- 1 つまたは複数のテンプレート・プロファイル
- アクセス名のセット
- 監査制御

第 8 章では、セキュリティ・クラスのオブジェクトの管理方法について説明します。

#### 5.9.1 命名規則

セキュリティ・クラスには、次のメンバが存在します。



CAPABILITY	COMMON_EVENT_CLUSTER
DEVICE	FILE
GROUP_GLOBAL_SECTION	LOGICAL_NAME_TABLE
QUEUE	RESOURCE_DOMAIN
SECURITY_CLASS	SYSTEM_GLOBAL_SECTION
VOLUME	

## 5.9.2 アクセスのタイプ

セキュリティ・クラス・オブジェクトは、次のアクセス・タイプをサポートします。

読み込み (R)	テンプレート・プロファイルを読み込む権限がユーザに与えられます。テンプレート・プロファイルには、新しいオブジェクトに割り当てられるセキュリティ要素が入っています。
書き込み (W)	テンプレート・プロファイルの値を変更する権限がユーザに与えられます。
制御 (C)	セキュリティ・クラス・オブジェクトのセキュリティ・プロファイルを変更する権限がユーザに与えられます。制御アクセスには、読み込みアクセスと書き込みアクセスも含まれます。

## 5.9.3 テンプレート・プロファイル

セキュリティ・クラス・オブジェクトは、次のテンプレート・プロファイルを提供します。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[SYSTEM]	S:RW,O:RW,G:R,W:R

## 5.9.4 実行される監査の種類

以下のイベントを監査できます。ただし、セキュリティ管理者が該当するイベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	プロセスが DCL の SET SECURITY または SHOW SECURITY コマンドに /CLASS=SECURITY_CLASS 修飾子を付けて実行したとき、またはプロセスが \$SET_SECURITY または \$GET_SECURITY システム・サービスに SECURITY_CLASS という名前を指定して呼び出したとき。

### 5.9.5 オブジェクトの永続性

セキュリティ・クラス・オブジェクトおよびそのすべてのメンバのセキュリティ・プロファイルは、セキュリティ・オブジェクト・データベースに保存されます。

## 5.10 ボリューム

ボリューム・オブジェクトは、1 つまたは複数の ODS-2 または ODS-5 形式のディスク・ボリュームのことです。ボリュームがバインドされたボリューム・セットの構成要素である場合、オブジェクトは複数のボリュームで構成されます。ボリューム上のディレクトリやファイルに対するアクセス権を持っていたとしても、ボリューム自体に対するアクセス権を持っていないければそれらのディレクトリやファイルにはアクセスできません。

テープおよびフォーリン・ボリュームへのアクセスについては、『*OpenVMS システム管理者マニュアル*』および『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』のマウント・ユーティリティに関する記述を参照してください。

### 5.10.1 命名規則

ボリューム名は、ボリューム・ラベル、ボリュームがマウントされたデバイスの名前、ユーザが指定した論理名のいずれかです。ボリューム・ラベル名の長さは、0 ~ 12 文字です。

### 5.10.2 アクセスのタイプ

ボリューム・クラスは、次のアクセス・タイプをサポートします。

読み込み (R)	ボリューム上のファイルの名前参照、印刷、およびコピーを行う権限がユーザに与えられます。
書き込み (W)	ボリューム上の既存のファイルに対する変更または書き込みを行う権限がユーザに与えられます。サブジェクトが特定のファイルに対する操作を実行できるかどうかは、そのファイルの保護によって決まります。書き込みアクセス権が意味を持つには、読み込みアクセス権も必要です。
作成 (C)	ディスク・ボリューム上にファイルを作成する権限と作成したファイルを変更する権限がユーザに与えられます。作成アクセス権には、読み込みアクセス権と書き込みアクセス権も必要です。
削除 (D)	ユーザがディレクトリとファイルに対する適切なアクセス権を持っていることを前提として、ディスク・ボリューム上のファイルを削除する権限がユーザに与えられます。削除アクセス権には、読み込みアクセス権が必要です。
制御 (C)	ボリュームの保護と所有権の要素を変更する権限がユーザに与えられます。

### 5.10.3 テンプレート・プロファイル

このクラスは、次のテンプレート・プロファイルを提供し、初期化時に値を割り当てます。このテンプレートは [0,0] という所有者 UIC を割り当てますが、この値は一時的なものです。オブジェクトが作成されると、作成したプロセスの UIC の対応するフィールド内の値が 0 の代わりに設定されます。

テンプレート名	所有者 UIC	保護コード
DEFAULT	[0,0]	S:RWCD,O:RWCD,G:RWCD,W:RWCD

### 5.10.4 必要な特権

VOLPRO 特権を持つユーザには、ボリュームに対する制御アクセスが常に許可されます。ファイル構造ボリュームをフォーリン・ボリュームとしてマウントするには、VOLPRO 特権または制御アクセス権が必要です。

### 5.10.5 実行される監査の種類

すべてのボリューム・アクセスを監査できます。ただし、セキュリティ管理者が Access イベント・クラスの監査機能を有効にする必要があります。

監査対象イベント	監査実行のタイミング
アクセス (A)	ファイル・システム操作の実行時。

### 5.10.6 オブジェクトの永続性

ボリューム・オブジェクトのセキュリティ・プロファイルは、ディスクのマスタ・ファイル・ディレクトリ (MFD) に [000000]SECURITY.SYS として保存されます。



# Part 3

---

## システム管理者のためのセキュリティ

このパートの各章では、以下のトピックについて説明します。

- セキュリティ管理者の役割 (第 6 章)
- システムのセキュリティ保護 (第 7 章)
- データとリソースのセキュリティ保護 (第 8 章)
- セキュリティ監査の実施 (第 9 章)
- セキュリティ侵害への対応 (第 10 章)
- 安全なクラスタの作成 (第 11 章)
- ネットワーク・システムの考慮事項 (第 12 章)
- 保護サブシステムの設定および管理 (第 13 章)

本書のこのパートには、次のトピックに関する情報もあります。

- ユーザ特権とユーザ特権を必要とするユーザ (付録 A)
- 重要なシステム・ファイルに対するデフォルトの UIC ベースの保護 (付録 B)
- C2 セキュリティ環境における操作のガイドライン (付録 C)
- セキュリティ・アラーム・メッセージの例 (付録 D)



---

## システムとそのデータの管理

この章では、システム管理者が OpenVMS オペレーティング・システムのセキュリティ機能を実装する方法を説明します。平均的なセキュリティが必要とされる商用システムのセキュリティの要件に基づいて、セキュリティ管理の概要について説明します。次のトピックについて説明します。

- セキュリティ管理者としての役割
- サイトのセキュリティ・ポリシー
- セキュリティ管理者用のツール
- セキュリティ管理者のアカウント要件
- ユーザのトレーニングに関する推奨事項
- 新規ユーザの処理のログ取得
- 毎週行う定期的な作業に含めるべき作業

セキュリティ対策を確立する前に、この章の全体と、その後の 3 つの章を読んでおくことをお勧めします。これらの章を読んでおけば、サイトに適したセキュリティ対策をよりの確に判断できるようになります。また、セキュリティ対策を実装するための道具も手に入れることになります。

### 6.1 セキュリティ管理者の役割

セキュリティ管理者の役割は、組織のセキュリティ・ポリシーを実装し、維持することです。組織の中には、セキュリティ・ポリシーの策定にセキュリティ管理者を参加させるところもあれば、確立されたポリシーの実装と維持をセキュリティ管理者に任せるものもあります。企業のセキュリティ・ポリシーの例については、6.2 節を参照してください。

セキュリティ管理者（または担当者）の仕事は、セキュリティ・ポリシーが実装され、維持されていることの確認です。発生しうるセキュリティ違反や脆弱性が生じていないか、定期的にシステムを監視する仕事は必須です。問題を発見したら、必ず問題を是正する必要があります。

多くの場合、組織はコンピュータ管理者の義務を分割します。セキュリティ管理者はシステムを監視し、問題を報告します。システム管理者はポリシーを実装し、システムを管理します。このような管理構造において、セキュリティ管理者はシステム管理者と連携します。システム管理者は、ユーザ・アカウントを設定

し、アカウントの必要性の証明に必要なペーパーワークを処理するために、アカウント担当者を採用する場合があります。この担当者は必ず、本質的には共同システム管理者の役割を担う、信頼性の高い人物でなければなりません。仕事を分担することになるため、システム管理者とセキュリティ管理者が定期的に連絡を取ることが非常に重要です。セキュリティ管理者は、問題が是正されるように、ユーザや、必要に応じてシステム管理者やアカウント担当者にセキュリティ問題を報告する必要があります。

多くの OpenVMS 使用サイトに共通する仕事の分担のもう 1 つの形態に、セキュリティ管理者とシステム管理者の役割の統合があります。1 人の人物がセキュリティ・ポリシーを実装し、その要件を満たすようにシステムを維持します。

信頼性の高いシステム管理は、編成にバリエーションはあっても、必ずユーザのトレーニング、アカウントとパスワードの設定、重要なシステム・ファイルとリソースの保護、セキュリティ関連イベントの監査と分析が含まれます。システムの使用形態を把握し、「平常時の」システムの活動を把握することが、信頼性の高い管理にとって非常に重要です。

## 6.2 サイトのセキュリティ・ポリシー

組織の幹部は、通常、従業員向けの簡潔なセキュリティ・ポリシーを策定して、組織が従業員に期待する行為を強調します。たとえば、そのようなポリシーでは、従業員による企業データの持ち出しやパスワードの共有を禁じます。

部局やコンピュータ・サイトの管理者は、それに基づいて詳細なセキュリティ・ポリシーを策定します。セキュリティ・ポリシーとは、パスワードとシステム・アカウントの使用、コンピュータ・システム、通信デバイス、およびコンピュータ・ターミナルへの物理的なアクセス、および監査対象となるセキュリティ関連イベントの種類に関する、明文化された一連のガイドラインです。これらのセキュリティ・ガイドラインには、特定のオペレーティング・システム環境に適用される、より詳細な規定が補足される場合があります。

最終的にセキュリティ・ポリシーの複雑度は、部局のセキュリティ要件が高、中、低のいずれであるかに依存します。第 1 章に、組織がそのニーズを把握するのに役立つ、一連の質問があります。

たとえば、多くのサイトのセキュリティ・ポリシーでは、どの従業員が個々のシステムにアクセスできるか、および例外的な作業や開発を行う担当者に使用可能なアクセスの種類を定義しています。場合によっては、ポリシーには、システム・アクセスを決定する一連の複雑なルールがある場合があります。表 6-1 に、ある部局によって策定されたポリシーを示します。



表 6-1: サイトのセキュリティ・ポリシーの例

セキュリティ領域	サイトの要件
パスワード	<p>パスワード変更のスケジュール。</p> <p>最短のパスワードの長さとは有効期限を制御するプロセス。</p> <p>システム・パスワード変更のスケジュールです。</p>
アカウント	<p>必要理由，要求者，要求者の上司，システム管理者，またはアカウント設定の担当者の署名など，コンピュータ・システムのアカウントを付与する手順。アカウントは共有できません。</p> <p>従業員の異動や退職などに伴う組織の変化により，アカウントを無効にする手順。</p> <p>通常は 6 ～ 12 ヶ月ごとにアカウントを再認定するための予定表。</p> <p>定期的には使用されていないアカウントを無効にする指示。</p> <p>アクセスの期間。</p> <p>アカウントの有効期限のための予定表。</p> <p>厳密に割り当てを制御する権限を要求する手順。</p> <p>通常システム処理を行う特権ユーザが，非特権アカウントを使用する要件。</p> <p>アクティブでないアカウントを確認するスケジュール。</p> <p>承認されたセキュリティ・ツールのリスト。</p>
監査対象セキュリティ・イベント	<p>一部またはすべてのログイン元からのログイン。</p> <p>登録ファイル・レコードへの変更。</p> <p>特権アクションおよびシステム管理アクションのその他の使用。</p> <p>インストール・ユーティリティを介した既知のファイル・リストの変更。</p> <p>ネットワーク制御プログラム (NCP) を使用した，ネットワーク構成データベースの変更。</p>
コンピュータ・ルームへの物理的なアクセス	<p>アクセスの理由を含む，認証された担当者の明文化されたリスト。通常，1 名がこのリストを最新の状態に維持する責任があります。</p> <p>訪問者の安全な領域へのログの保管。</p> <p>ドアの施錠管理と，鍵，キー・カード，およびそれらの組み合わせを割り当てるための文書化された手順。これらのアクセス制御は，定期的に，および従業員の異動または退職時に変更されます。</p>

表 6-1: サイトのセキュリティ・ポリシーの例 (続き)

セキュリティ領域	サイトの要件
コンピュータ・ルームの外部にあるターミナルやパーソナル・コンピュータへの物理的なアクセス	<p>一定期間使用されていないターミナルをログアウトするプログラムの使用。</p> <p>(コンピュータ担当者以外の) 組織向けのセキュリティ意識向上プログラム。次のテーマを扱います。</p> <ul style="list-style-type: none"> <li>認定済みソフトウェアのリストの維持。</li> <li>コンピュータ・システム、ネットワーク・パスワードなどのシステム・アカウント情報に関連するハードコピー情報の卓上からの排除。</li> <li>ディスクおよびファイル・キャビネットの施錠。</li> <li>ワークステーション内またはワークステーション周辺でのディスクレットのアクセス防止。</li> <li>キーを人目に付く場所に置くことの禁止。</li> </ul>
ダイヤルアップ番号	<p>認定ユーザのリスト。</p> <p>番号を定期的に変更するためのスケジュール、および番号の変更をユーザに通知するための手順。</p> <p>ダイヤルアップ番号の公開を最小限にするポリシー。</p> <p>定期的なパスワード変更、およびアクセス権を持つ従業員が退職した際のパスワード変更に関するポリシー。</p> <p>モデムまたはターミナル・サーバにおけるパスワード、またはホスト・ダイヤルアップ・ポートにおけるシステム・パスワードの保護。</p> <p>以下の内容に関して使用可能なドキュメント。</p> <ul style="list-style-type: none"> <li>ダイヤル・バック・システム</li> <li>ネットワークに関する詳細</li> <li>インストールされているターミナル装置</li> <li>ターミナル切り替えシステム</li> <li>ネットワークに接続されているすべてのターミナル・デバイスに関する詳細</li> <li>すべてのダイヤルアップ装置に関する詳細</li> </ul>
通信	<p>TCP/IP, LAT, またはイーサネット接続を介してパスワードを使用している場合の、特権アカウントへのアクセス拒否。</p> <p>特権アカウントへのネットワーク・ログイン用の認証カードの使用。</p>

## 6.3 安全なシステムを設定するためのツール

以降の章では、セキュリティ・ポリシーに従って安全なシステムを設定する方法について説明します。登録ユーティリティ (AUTHORIZE) が、システム・セキュリティを実装するための主要なツールです。AUTHORIZE は、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』で詳しく説明しています。システム・パラメータ・ファイルの変更に使用する AUTOGEN コマンド・プロセスは、『*OpenVMS システム管理者マニュアル*』および『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』で説明しています。また、多くの DCL コマンドも重要なセキュリティ・ツールです。DCL コマンドは、『*OpenVMS DCL デクショナリ*』で説明しています。

## 6.4 セキュリティ管理者のアカウント要件

セキュリティ管理者には、そのタスクを実行する特権を備えたアカウントが必要です。

セキュリティ違反と、発生しうる脆弱性を確認する管理者には、少なくとも次の 3 つの特権が必要です。

- セキュリティ監査を有効にし、セキュリティ・オペレータ・ターミナルを設定するために必要な SECURITY および AUDIT 特権
- ファイルおよび資源の保護を確認するために必要な READALL 特権

多くの場合、セキュリティ管理者は、セキュリティ管理者とシステム管理者の両方を務めます。このユーザには、特権の完全なセットが必要です。『*OpenVMS システム管理者マニュアル*』で、システム管理カウントに必要な特性を説明しています。

例 6-1 に、セキュリティ管理者のカウントに適した AUTHORIZE 修飾子をいくつか示します。指定のない値のデフォルト値はすべて、SYSUAF.DAT のデフォルト・レコードに基づく値になります。

例 6-1: セキュリティ管理者のアカウントの例

---

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD RIRONWOOD/PASSWORD=VALTERSY/UIC=[001,100] -
_UAF> /DEVICE=SYS$SYSDEVICE/DIRECTORY=[RIRONWOOD] -
_UAF> /OWNER="Russ Ironwood"/ACCOUNT=SECURITY/FLAGS=GENPWD -      [1]
_UAF> /PWLIFETIME=30-/PWDMINIMUM=8 -                                [2]
_UAF> /PRIVILEGES=(AUDIT,SECURITY,READALL)                          [3]
identifier for value:[000001,000100] added to RIGHTSLIST.DAT
UAF>
```

---

次の点に注意してください。

1. パスワードの変更には、自動パスワード・ジェネレータを使用する必要があること。
2. パスワードの有効期間を短くすること。  
多くの重要な特権とアクセス権が与えられるため、アカウントの保護には、対策 1 および 2 は重要です。
3. SECURITY, AUDIT, および READALL 特権によって、システムの監視が可能になりますが、変更はできません。システム管理者の作業を行う場合には、SYSPRV 特権を持つアカウントが必要です。SYSPRV 特権によって、システム保護フィールドにより保護されたオブジェクトにアクセスし、所有者 UIC および保護を変更することが可能になります。オブジェクトの保護を変更して、そのオブジェクトへのアクセス権を取得することができます。

## 6.5 新規ユーザのトレーニング

新規ユーザにシステム・セキュリティについて教えることは、重要なセキュリティ保護手段になります。ユーザにセキュリティの手法と目標を認識させることが重要です。ユーザがシステムと侵入発生の仕組みに対する理解を深めれば、ユーザは侵入に対する防御力が向上します。

ユーザのトレーニングには、次の内容が必要です。

- ユーザのアカウントの場所について。具体的には、システムの種類、システムの場所、ネットワーク上にある場合にはその適切なノード名、システムがクラスタの一部である場合には使用可能な他のノードについて。
- ログインに使用できるターミナル、およびそれらの場所について。
- アカウントが、ローカル、ダイアルアップ、リモート、会話型、ネットワーク、またはバッチなどの操作に関して制限されているかどうかについて。制限されている場合、許可されている使用と制限の両方について説明します。
- ダイアルアップしてアカウントにアクセスできるかどうかについて。アクセスできる場合は、アクセス用の電話番号を教え、手順を説明します。接続が失敗するまでの、再試行の許容回数と、再試行の間隔として許容される最大秒数を指定します。
- ユーザが使用する可能性のあるターミナルに、システム・パスワードが実装されているかどうかについて。実装されている場合は、ターミナルの場所、システム・パスワードの変更頻度、およびユーザが新しいシステム・パスワードを知る方法を説明します。
- アカウントの有効期間、期限の日時、ユーザが延長を申請する場合の申請先について。

- ユーザ名，ユーザが保持する識別子，ユーザに関連付けられているグループ番号およびメンバ番号について。
- 必要なパスワード情報について。特に，最初のパスワード，パスワードがロックされているかどうか，ロックされていない場合は，パスワードを変更しなければならない頻度，パスワードの最低の長さ，アカウントに第2パスワードがあるかどうか，第2パスワードを知っているユーザ，ユーザが自由にパスワードを選択できるか，またはパスワードを自動的に生成する必要があるかについて。ユーザにとって望ましい習慣のチェックリストについては，3.12 節「システム・セキュリティへの貢献のためのチェックリスト」を参照してください。
- デフォルトのデバイスとディレクトリについて。
- デフォルトの保護について。
- ディスク使用量に制限があるかどうかについて。制限がある場合は，その値について。
- 使用に関する制限があるかどうかについて。たとえば，使用が推奨または強制される特定の曜日や時間帯があるかどうかについて。該当する場合は，主曜日と副曜日について説明します。
- 共有されているファイルやディレクトリが存在するかどうかについて。存在する場合は，詳細を説明します。
- ユーザに影響する ACL があるかどうか，ユーザが知っておかなければならない識別子について。
- ユーザが保持する特権，およびその意味について。
- コマンド言語インタプリタについて。
- アカウントが，オープン，キャプティブ，制限付き，会話型のどの種類であるかについて。
- ユーザ用の代理ログインが存在する場合，代理ログインを許可するノードについて。
- ユーザが使用する必要がある可能性のあるキューの名前について。
- 資料の施錠保管など，サイトの物理的なセキュリティを確保するためにユーザが取るべき対策について。

## 6.6 ユーザのセッションのログ取得

ユーザがシステムを学習している間，機密データへのアクセスやシステム運用の制御など，ユーザが特に機密性の高い機能を実行する場合，セキュリティ管理者はターミナル・セッションを監視することができます。場合によっては，ユーザは自分の操作を記録するために，自分自身のセッションのログ取得を選択できます。その場合，最初にログインした後で，ユーザは SET HOST 0/LOG コマンドを会話形

式で実行できます。この節では、制限付きアカウントを設定することでユーザのセッションのログを取得する、1つの方法を説明します。数多くのサードパーティ製品では、より効率的にセッションを監視できる他の方法を提供しています。セキュリティ管理者は、選択した方式に関係なく、その方式が許容可能であるかどうかを法務部に確認を取るべきです。

特別な制限付きアカウントと適切なコマンド・プロシーダを使用することで、選択したユーザのターミナル・セッションのログの取得を実施できます。これらのユーザは、まず制限付きアカウントにログインしてから、自分自身のアカウントにログインする必要があります。制限付きアカウントにより、セッションのログ取得が保証されます。

次の例に、制限付きアカウント(この例ではUSER\_LOG という名前)の設定方法に関するガイドラインと、適切なコマンド・プロシーダのサンプルを示します。

1. 次のように、制限付きアカウント USER\_LOG を設定します。

```
UAF> ADD USER_LOG /FLAGS=(RESTRICTED,DISMAIL,DISNEWMAIL)-
_UAF> /LGICMD=SYS$SYSROOT:[USER_LOG]SESSIONLOG-
_UAF> /DEV=SYS$SYSROOT: /DIR=[USER_LOG]-
_UAF> /NONNETWORK /NOBATCH /UIC=[200,256]
```

2. SESSIONLOG.COM コマンド・プロシーダにより、ターミナル・セッションのログ取得が有効になります。

```
$ ! SESSIONLOG.COM - log in to specified account with terminal session
$ ! logging enabled.
$ !
$ WRITE SYS$OUTPUT "Please log in to the account of your choice."
$ WRITE SYS$OUTPUT "Your terminal session will be recorded."
$ WRITE SYS$OUTPUT " "
$ !
$ ! Acquire the intended user name and save it in a temporary file. Use
$ ! it to name the log file, and pass it as the first line of input to
$ ! LOGIN.
$ !
$ READ/PROMPT= "Username: " SYS$COMMAND USERNAME
$ PID = F$GETJPI (0, "PID")
$ OPEN/WRITE OUTPUT USERNAME 'PID' .TMP
$ WRITE OUTPUT USERNAME
$ CLOSE OUTPUT
$ DEFINE/USER SYS$INPUT USERNAME 'PID' .TMP
$ SET HOST 0 /LOG= 'USERNAME' .LOG
$ DELETE USERNAME 'PID' .TMP;0
$ LOGOUT
```

3. セッション監査の対象にする各アカウントを設定します。次のコマンドは、ユーザ Smith のアカウントを設定します。

```
UAF> MODIFY SMITH /FLAGS=RESTRICTED /NOLOCAL /NODIALUP -
_UAF> /LGICMD=SYS$SYSROOT:[USER_LOG]CHECKLOG
```

制限付きログイン・コマンド・プロシージャにより、ログインが SET HOST コマンドを使用する USER\_LOG アカウントから行われることが保証されるため、セッションのログが取得されます。

4. また、各ユーザ・アカウントのバッチおよびネットワーク・アクセスを無効にして、USER\_LOG アカウントからのローカル・ログインのみを許可することもできます。次に例を示します。

```
UAF> MODIFY SMITH/FLAGS=RESTRICTED/NOLOCAL/NODIALUP/NOBATCH -  
/NONETWORK/LGICMD=SYS$SYSROOT:[USER_LOG]CHECKLOG
```

5. 次の CHECKLOG.COM コマンド・プロシージャは、ユーザが USER\_LOG アカウントにログインしていることを確認します。このプロシージャが正しく動作するためには、12.3.2 項の手順に従って、DECnet 代理アカウントを有効にしておく必要があります。

```
$ ! CHECKLOG.COM - ensure that the account is being logged in to  
$ ! the USER_LOG account.  
$ !  
$ IF F$MODE ( ) .NES. "INTERACTIVE" THEN EXIT  
$ !  
$ ! Verify that the connection originated from the local node and  
$ ! from the USER_LOG account.  
$ !  
$ IF F$LOGICAL ( "SYS$NODE" ) .EQS. F$LOGICAL ( "SYS$REM_NODE" ) -  
  .AND. F$LOGICAL ( "SYS$REM_ID" ) .EQS. "USER_LOG" -  
  THEN GOTO OK $ WRITE SYS$OUTPUT "You may log in to this account only with ", -  
    "the USER_LOG account."  
$ LOGOUT  
  
$ !  
$ ! When the login has been verified, enable Ctrl/Y to  
$ ! release the account, invoke the user's LOGIN.COM, and turn  
$ ! control over to the user.  
$ !  
$ OK:  
$ SET CONTROL_Y  
$ IF F$SEARCH ( "LOGIN.COM" ) .EQS. "" THEN EXIT  
$ @LOGIN
```

## 6.7 安全なシステムを維持するための継続的な作業

安全なシステムを維持するには、継続的に監視を行う必要があります。セキュリティ管理者の役割を担うユーザにとって、次の継続的な作業が重要となります。

- MONITOR IO レポートを使用して、さまざまな時点におけるシステムの通常の入出力の量を把握します。異常な変化に注意します。
- システムにインストールされるイメージに関する最新の情報を常に得られるようにしておきます。インストール・ユーティリティ (INSTALL) を使用して、予期しない追加がないかを調べます。既知ファイル・リストを監視する場合は、現在のリストと、有効なハードコピーのリストを比較します。
- AUTHORIZE の SHOW コマンドを定期的に使用して、不正なユーザ名がないことを確認します。

- AUTHORIZE の SHOW/PROXY コマンドを定期的に使用して、セキュリティ管理者が承認したすべての代理アクセスをすばやく確認します。予期しない追加に注意します。アクセスの必要がなくなったすべての遠隔ユーザを削除します。遠隔ノードのシステム管理者と、定期的に連絡を取ります。
- 会計情報ユーティリティ (ACCOUNTING) を定期的に適用して、通常の処理時間の基準を算出します。説明のつかない変化に注意します。
- 既知のユーザ名、未知のユーザ名、および適切なシステムの使用時間に関して、ACCOUNTING により生成されるアカウントのレポートを定期的にチェックします。
- 例外的な時間に正常（および異常）な処理が発生していることに気づけるように、システムの作業負荷を十分に把握しておきます。
- 予期していない事態にすぐ気づけるように、DCL の SHOW DEVICE コマンドを使用して、普段からデバイス割り当てを監視します。
- バッチ・キューで実行され、繰り返し発生するバッチ・ジョブの種類と、これらのバッチ・ジョブが実行される可能性が最も高い日時を把握しておきます。
- DIRECTORY/SECURITY コマンドを使用して、重要なファイルの保護と所有権を監視します。保護と所有権に関して、説明のつかない変化に注意します。
- ライト・リストを常に把握しておきます。追加された識別子や、現在の識別子の新しい保持者に気づけるように、最新のリストを維持します。
- 使用されていない識別子を削除します。ライト・リストを最新の状態に維持します。
- UAF レコードの設定に使用するテンプレートを定期的に確認します。必要な変更はすべて加えます。
- 第 9 章で説明しているセキュリティ監査機能を使用します。
- 監査分析ユーティリティ (ANALYZE/AUDIT) を定期的に適用して、異常な監査処理を検出します。
- 新規ユーザに初期パスワードの変更を許可する場合は、ユーザが変更したいと思うようなパスワードを割り当てるか、パスワード・ジェネレータを使用します。始めに割り当てたパスワードを使用してログインできるかどうかをあらためて確認します。必要に応じて、要求した変更が行われなかった理由をユーザに確認します。
- 保護されていないユーザ・ファイルを検索して、パスワードがネットワーク・アクセス制御文字列に埋め込まれていないかを調べます。パスワードの後は、3 文字の区切り文字 ("::") が付きます。また *password* という名詞を検索し、その近くにパスワードが出現していないかどうかを確認します。



- ユーザが適切にログアウトしていることを確認します。通常の業務時間の終わりに、物理的なチェックを行います。
- ユーザが適切なデフォルトの保護を実施していることを確認します。
- 磁気テープ、ディスク、およびプログラム・リストの目録を常に最新の状態に保ちます。普段からその目録をチェックして、物理的なセキュリティが低下している兆候がないかを確認します。
- オフィスとすべての重要なリストの施錠保管を徹底します。



## システム・アクセスの管理

この章では、ユーザのアカウントとパスワードを割り当てることで、ユーザにシステムへのアクセス権を指定する方法を説明します。アカウントを保護する必要があり、平均的なセキュリティが必要とされる商用システムのセキュリティの要件に基づいて説明します。また、平均よりも高度なセキュリティの要件についても説明します。システム・データおよびリソースへのアクセスの制御の詳細については、第 8 章を参照してください。ユーザ・アクションの監査の詳細については、第 6 章および第 9 章を参照してください。

登録ユーティリティ (AUTHORIZE) が、アカウントとパスワードを確立するための主要なツールです。このユーティリティの説明については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の A-L を参照してください。

### 7.1 システムにアクセス可能な時間と条件の定義

ユーザに与えられるシステム・アクセスのレベルは、サイトの要件、組織内におけるユーザの役割、およびユーザのアカウントの管理に依存します。セキュリティの要件が低く、システム・リソースが多いサイトでは、1 日のどの時間帯にアクセスしてもよいようにする一方で、セキュリティ要件が中程度のサイトでは、ログインを日中の時間帯のみに限定したり、ダイヤルアップまたはネットワーク接続を一部のユーザのみに許可するということが考えられます。

登録ユーティリティを使用して、ユーザがシステムにアクセスできる日時と方法を制御できます。表 7-1 に、適用可能な修飾子を示します。

表 7-1: ログインの時間と条件を制御する **AUTHORIZE** 修飾子

カテゴリ	修飾子	説明
時間帯	/ACCESS	デフォルトでは、ユーザは毎日いつでもアクセス権できます。アクセスの時間を指定することで、その他の時間におけるアクセスを禁止できます。キーワード PRIMARY を使用して主曜日の時間帯を指定し、キーワード SECONDARY を使用して副曜日の時間帯を指定します。
	/DIALUP	ダイヤルアップ・ログインに対して許可するアクセスの時間帯を指定します。

表 7-1: ログインの時間と条件を制御する **AUTHORIZE** 修飾子 (続き)

カテゴリ	修飾子	説明
曜日	/LOCAL	ローカル・ターミナルからの会話型ログインのアクセスの時間帯を指定します。
	/PRIMEDAYS	1 週間のうちログインが可能な主曜日と副曜日を定義します。
	/BATCH	バッチ・ジョブに対して許可するアクセスの時間帯を指定します。
操作モード	/INTERACTIVE	会話型ログインのアクセスの時間帯を指定します。
	/NETWORK	ネットワーク・バッチ・ジョブに対して許可するアクセスの時間帯を指定します。
	/REMOTE	(DCL の SET HOST コマンドを使用して) ネットワーク遠隔ターミナルからの会話型ログインに対して許可するアクセスの時間帯を指定します。
リソースの割り当て	/DEVICE	ログイン時のユーザのデフォルト・デバイスの名前を指定します。
	/DIRECTORY	ログイン時のユーザのデフォルト・ディレクトリの名前を指定します。
アカウントの有効性	/EXPIRATION	アカウントが期限切れになる日付と時刻を指定します。
	/FLAGS=DISUSER	ユーザがログインできないように、アカウントを無効にします。
外部認証	/FLAGS=EXTAUTH	ユーザが外部認証されるように指定します。

### 7.1.1 作業時間の制限

AUTHORIZE 修飾子を使用して、システムの使用を、1 週間の特定の曜日や 1 日の特定の時間帯に制限できます。作業時間の制限は、システムの負荷バランスの改善に便利です。アカウントへのアクセスの制限は、通常の業務時間以外におけるシステムの不正使用を防ぐ効果的な手段でもあります。

/PRIMEDAYS 修飾子を使用して 1 週間の主曜日と副曜日を定義するか、主曜日が月曜日から金曜日で副曜日が土曜日と日曜日であるデフォルトに従います。たとえば、火曜日から土曜日まで勤務するユーザのデフォルトを変更するには、/PRIMEDAYS 修飾子を次のように指定します。

```
/PRIMEDAYS=(NOMONDAY,TUESDAY,WEDNESDAY,THURSDAY,FRIDAY,SATURDAY,NOSUNDAY)
```

ただし、主曜日に祝日がある場合など、サイトの通常の曜日割り当てに当てはまらない変更が運用上必要になることがあります。通常の曜日割り当てをオーバーラ

イドするには、DCL の SET DAY コマンドを使用し、該当日に対して適用する曜日タイプの解釈を指定します。これには、OPER 特権が必要です。この変更は、すでにログインしているすべてのユーザのほか、以降その日にログインするすべてのユーザに適用されることに注意してください。曜日タイプを変更すると、現在ログインしているユーザのうち、その曜日タイプに関しては権限のないユーザは、次の 1 時間でシステムからログアウトされます。ジョブ・コントローラは、1 時間ごとに時間の制限を適用します。

時間帯ごとに制限する必要があるログイン・アクセスのタイプを決定します。ログイン・アクセスの修飾子には、/LOCAL、/REMOTE、/DIALUP、/INTERACTIVE、/BATCH、および /NETWORK があります。ただし、サイトで全タイプのログインに対して主時間と副時間のセットを 1 つだけ適用する場合、すべてのアクセスのモードに適用される /ACCESS 修飾子を指定できます。

次の例では、ユーザのアカウントに /BATCH 修飾子を適用して、通常の業務時間中にそのユーザがバッチ・ジョブを実行できないようにする方法を示します。

```
/NOBATCH=(PRIMARY, 9-17)
```

このように指定すると、該当するユーザは主曜日は午後 6:00 から午前 8:59 までの時間帯にのみバッチ・ジョブを実行できますが、副曜日は一日中実行できます。

### 7.1.2 操作モードの制限

次の理由で、一部のユーザのネットワーク・アクセスを禁止することが考えられます。

- ・ ユーザは、アクセスをローカル・ノード経由に限定しなければならないデータを持っている。
- ・ 接続の匿名性が高いため、侵入の試みがネットワーク経由で発生する可能性が高くなります。この問題は、ダイアルアップ接続にも該当します。

特定のユーザのネットワーク・アクセスを禁止するには、次の例のように、AUTHORIZE 修飾子の /NONETWORK を使用します。

```
UAF> ADD JSMITH /NONETWORK, ...
```

すべての AUTHORIZE アクセス・モードの修飾子 (/LOCAL、/REMOTE、/DIALUP、/INTERACTIVE、/BATCH、または /NETWORK) をこの方法で無効にして、システムへのアクセスを制限することができます。

### 7.1.3 アカウントの有効期間の制限

ユーザがアクセスを必要とすると想定される時間の最大の長さに合うように、アカウントの有効期限を設定することをお勧めします。有効期限が切れると、システムにより、そのアカウントへのアクセスが自動的に禁止されます。ただし、UAF レコードとユーザのファイルは、セキュリティ管理者が手作業で削除する必要があります。

また /EXPIRATION 修飾子を使用すると、定期的にアカウントを確認して必要なアカウントのみを再認証することを求められるようになります。

アカウントの有効期限を設定するには、ユーザの UAF レコードで AUTHORIZE 修飾子の /EXPIRATION を使用します。たとえば次の修飾子は、ユーザのアカウントが 2001 年 12 月 30 日に満了することを指定します。

```
/EXPIRATION=30-DEC-2001
```

#### 7.1.4 アカウントの無効化

特定のアカウントの使用を厳しく制限したい場合があります。たとえば、SYSTEST アカウントや FIELD アカウントなど、定期的にもみ使用される特定のアカウントを無効にして、アカウントの悪用を防止できます。アカウントを無効にするには、/FLAGS=DISUSER 修飾子を使用します。必要に応じて一時的にアカウントを有効にするには、/FLAGS=NODISUSER 修飾子を使用します。

#### 7.1.5 ディスク・ボリュームの制限

UAF レコードでユーザのデフォルトのデバイスとディレクトリを指定するには、AUTHORIZE 修飾子の /DEVICE と /DIRECTORY を使用します。『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の A-L で説明されているように、システム管理ユーティリティ (SYSMAN) のディスク制限機能により、そのディスク（および他すべてのディスク）上でユーザが使用できるブロックの数を制限できます。

他のディスクに設定されているボリューム保護により、ユーザがアクセス可能なディスク量が制御されます。AUTHORIZE 修飾子の /PRIVILEGES を使用して拡張または制限できるユーザの特権も、アクセス可能な量に影響を与えます（8.7 節を参照）。

#### 7.1.6 外部認証用アカウントのマーク付け

UAF レコードでユーザのアカウントに対して、AUTHORIZE 修飾子の /FLAGS=EXTAUTH を使用してマークを付けることで、そのユーザの外部認証を許可することができます。

詳細については 7.4 節を参照してください。

### 7.2 ユーザへの適切なアカウントの割り当て

ユーザが有するシステム・アクセスのタイプは、大部分が、システム・リソースに対するユーザの必要性和、サイトのセキュリティ要件に依存します。この節では、OpenVMS システムで使用可能なユーザ・アカウントのタイプと、あるタイプのアカウントが別のアカウントよりも適当である理由について説明します。ユーザ・アカウントの追加手順の順を追った説明については、『*OpenVMS システム管理者マニュアル*』を参照してください。

## 7.2.1 システム・アカウントのタイプ

アカウントには、次の2つの主要なタイプがあります。

- 会話型アカウントは、システム・ソフトウェアにアクセスできます。通常、このようなアカウントは独立したアカウントと見なされます。
- 限定アクセス・アカウントでは、システムへのログインが制限され、場合によっては、ユーザ・ソフトウェアへのアクセスが制限されます。限定アクセス・アカウントにより、システムおよびプロセスのログイン・コマンド・プロセスージャだけでなく、これらのプロセスージャから呼び出されるすべてのプロセスージャも実行できるようになります。

限定アクセス・アカウントには、キャプティブ・アクティブと制限付きアカウントの2つのタイプがあります。ゲスト、代理、および自動ログインのアカウントは、キャプティブおよび制限付きアカウントの例です。

現在、DECwindows ソフトウェアは、従来の意味でのキャプティブまたは制限付きログインをサポートしていません。ただし、ユーザがログインし、DECterm ウィンドウを作成すると、キャプティブまたは制限付きアカウントの従来の環境が適用されます。

7.2.2 項で説明されているように、会話型アカウントおよび限定アクセス・アカウントは、どちらも特権アカウントにすることが可能で、また外部で認証することもできます。

次の表に、ユーザが実行するタスクに基づいて作成すべきアカウントの種類を示します。

ユーザのタスク	作成するアカウントのタイプ
プログラム開発やテキスト編集などの一般的な作業	会話型
限定的な操作のみを必要とする日常的なコンピュータ作業	キャプティブ
無人の時間帯におけるバッチ処理	キャプティブ
機密情報を扱うアプリケーション・プログラムの実行	キャプティブ
MAIL などのネットワーク・アプリケーションの使用	制限付き
(制限付きで) 遠隔システムからのシステム上のリソースへのアクセス	キャプティブまたは制限付き
ネットワーク代理アカウントの使用	制限付き
スマート・カードなどの認証システムの使用	制限付き
レイヤード・プロダクトのインストールの一部として作成されたアカウントの使用	制限付き

ユーザのタスク	作成するアカウントのタイプ
特権操作の実行	会話型，制限付き，またはキャプティブ
パスワードなしでの遠隔システムからのリソースへのアクセス	キャプティブ
アプリケーション・ターミナルへの自動ログイン	キャプティブまたは制限付き
外部ユーザの ID とパスワードを使用した OpenVMS ログイン・プロンプトでのログイン	外部認証

多数のユーザに適用できる 1 つまたは複数のテンプレートを作成できます。ただし，単にテンプレートを適用するというレベルまで，アカウント作成のプロセスを過度に単純化しないでください。テンプレートだけに頼ると，個別のユーザに適用すべき特別な考慮事項を見逃し，セキュリティ管理者のみが行使できる重要な制御ができなくなる恐れがあります。

テンプレートを定期的に調べて，テンプレートが有効であり，必要な運用手順を反映していることを確認します。テンプレートは，すぐに古くなってしまいます。

#### 7.2.1.1 会話型アカウントの例

例 7-1 は，セキュリティの考慮が必要で，平均的なユーザのアクセスが制限されている商用サイトのアカウントに一般的に見られる，中程度に制限された会話型ユーザ・アカウントの作成方法を示します。

##### 例 7-1: 一般的な会話型ユーザ・アカウントの作成

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD RDOGWOOD /PASSWORD=TRALAYAM/UIC=[231,010] -           [1]
_UAF> /DEVICE=BOTANYDEV/DIRECTORY=[RDOGWOOD] -
_UAF> /OWNER= " Robert Dogwood " /ACCOUNT=BOTNYDPT -
_UAF> /FLAGS=(GENPWD) /PWDMINIMUM=6 -                             [2]
_UAF> /EXPIRATION=15-JUNE-2003/PWDLIFETIME=90 -                   [3]
_UAF> /PRIMEDAYS=(MON,TUES,WED,THURS,FRI,SAT,NOSUN) -            [4]
_UAF> /NOACCESS=(PRIMARY,23-6,SECONDARY)/NODIALUP                [5]
identifier for value:[000231,000010] added to RIGHTSList.DAT
UAF>
```

次の点に注意してください。

1. 必要なパスワードは 1 つのみです。
2. パスワードは 6 文字以上にする必要があります。
3. ユーザのパスワードは 90 日間有効で，これは例 6-1 にある管理者のパスワードの有効期間よりもはるかに長くなっています。



4. ユーザには、平日と土曜日のアクセスが許可されています。
5. この 6 日の間に、ユーザは 1 日 15 時間のアクセス権を有します。

### 7.2.1.2 限定アクセス・アカウントの例

例 7-2 に、ユーザが高度な制限が適用されるアプリケーション運用アカウントの作成方法を示します。このアカウントは、州立大学の成績の一覧を作成し、各学生の家庭への郵便物を作成するという 2 つの機能を実行する目的があります。

例では、指定のない値のデフォルト値はすべて、SYSUAF.DAT のデフォルト・レコードに基づく値になります。

#### 例 7-2: 限定アクセス・アカウントの作成

---

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD REPGRADES /DEVICE=ADMINDEV/DIRECTORY=[REPGRADES] -
_UAF> /FLAGS=(CAPTIVE,DISWELCOME,DISNEWMAIL,DISMAIL,DEFCLI) -      [1]
_UAF> /PASSWORD=GROBWACH/UIC=[777,031] -                             [2]
_UAF> /LOCAL=(PRIMARY,8-17)/PRIMEDAYS=(MON,TUES,WED,THU, -          [3]
_UAF> FRI,NOSAT,NOSUN) -
_UAF> /NONETWORK/NOREMOTE/NODIALUP -                                  [4]
_UAF> /LGICMD=GRADES /CLITABLES=GRADES_TABLES -                     [5]
UAF>
```

:

```
user record successfully added
identifier for value:[000777,000031] added to RIGHTSLIST.DAT
```

次の点に注意してください。

1. アカウント・ユーザには、システムによって通常表示されるウェルカム・メッセージが表示されません。このアカウントは、メールを受信できません。また、ログイン・コマンド・プロシージャとデフォルト・コマンド・インタプリタ (DCL) の制御下での実行に制限されます。
  2. ログインを開始するユーザは、パスワード GROBWACH を指定する必要があります。通常は、セキュリティ管理者のみがパスワードを変更します。
  3. ローカル・ログインからのジョブの実行は、月曜日から金曜日の午前 8 時から午後 5:59 までの時間帯に制限されます。バッチ・ログインとローカル・ログインのみが許可され、バッチ・モードには時間の制限がないことに注意してください。
  4. ジョブは、ダイヤルアップ回線経由での実行、および遠隔ジョブとしての実行ができません。また、このアカウントはネットワーク・アクセスも拒否します。
  5. プロセスは、特別なログイン・コマンド・プロシージャ (GRADES.COM) の制御下で実行されます。このログイン・コマンド・プロシージャは、おそらくオペレータに機能のメニューを提示します。
  6. プロセスが実行できるコマンドは、CLI テーブルである GRADES\_TABLES に定義されているコマンドに制限されます。
-

## 7.2.2 特権アカウント

特権により、ユーザがシステム上で実行を許可される機能が決まります。TMPMBX および NETMBX 以上の特権があるアカウントは、特権アカウントと見なされません。特権アカウントには、会話型、制限付き、またはキャプティブのアカウントが可能です。

特権アカウントの不正使用は深刻な損失を招く可能性があるため、次のように、最も強力な権限を持つアカウントには、特別な制御を適用することを検討します。

- アカウントへのアクセスを限定します。たとえば、遠隔地からの部外者の侵入行為を防ぐために、/NODIALUP または /NONETWORK 修飾子を使用して、ダイヤルアップまたはネットワーク・アクセスを禁止します。
- セキュリティ・アラームを適用して、BYPASS, SYSPRV, READALL, および GRPPRV の、ファイル保護に関連する特権の使用を検出します。セキュリティ・アラームの決定と監視の詳細については、第 9 章を参照してください。

SYSTEM アカウント以外のすべてのアカウントに対しては、次の制限も追加します。

- /PRIMEDAYS および /NOACCESS 修飾子を使用して、ログインが可能な時間帯や曜日を制限します。使用方法が適切かどうかを監視できる日時を選択します。
- 使用されていない時は、AUTHORIZE 修飾子の /FLAGS=DISUSER を使用してアカウントを無効にします。
- 追加の検証を行うには、キャプティブ・ログイン・コマンド・プロシーダを使用します。キャプティブ・ログイン・コマンド・プロシーダは、7.2.4 項で説明されています。

当然ですが、SYSTEM アカウントに制限を加える必要があります。最も安全な手法は、バッチ・アクセスを除くすべてに関して SYSTEM アカウントを無効にして、実行者をたどれるように個別の特権ユーザ・アカウントを使用してシステム管理作業を行うようにする方法です。

### 特殊用途の特権キャプティブ・アカウント

キャプティブ・アカウントの安全性はそのコマンド・プロシーダの完全性に依存しているため、信頼のないユーザに対して特権キャプティブ・アカウントを設定することはお勧めできません。ただし特権が必要な状況では、汎用の特権アカウントではなく、キャプティブ特権アカウントを介して特定の機密性の高い機能を実行する方が安全です。たとえば、バックアップ操作を実行するユーザには READALL 特権が必要です。バックアップを実行するアカウントをキャプティブ・アカウントにすることで、手順が確実にシステムのバックアップ・ポリシーに従って実行されることを保証できます。

キャプティブ・アカウントの設定のガイドラインについては、7.2.4 項を参照してください。

### 7.2.3 会話型アカウント

会話型アカウントは、セキュリティ要件が低から中である環境でよく使用します。会話型アカウントは、プログラム開発やテキスト編集などの一般的な作業に適しています。『*OpenVMS システム管理者マニュアル*』では、このタイプのアカウントの設定手順が説明されています。7.2.1.1 項に例があります。

### 7.2.4 キャプティブ・アカウント

キャプティブ・アカウントは、ユーザの操作を制限します。また、適切に管理すれば、DCL コマンド・レベルへのユーザ・アクセスを拒否します。アカウントを設定して、完全に特定のプログラムまたはキャプティブ・ログイン・コマンド・プロシージャの制御下にユーザの実行を制限することができます。

キャプティブ・アカウントの主要な機能は、そのログイン・コマンド・プロシージャです。このタイプのアカウントにより、システム・ログイン・コマンド・プロシージャ (SYLOGIN.COM) およびプロセス・ログイン・コマンド・プロシージャ (SYSUAF.DAT の/LGICMD 修飾子で指定) のほか、これらのコマンド・プロシージャから呼び出されるコマンド・プロシージャの実行も保証されます。ログイン時にユーザは、表 7-2 に示した修飾子を指定してキャプティブ・コマンド・プロシージャを変更することはできません。

キャプティブ・アカウントにログインすると、ユーザは Ctrl/Y シーケンス、SPAWN コマンド、または INQUIRE コマンドを使用して DCL コマンド・レベルに抜け出すことはできません。UAF レコードの DISCTLY フラグが有効になっているため、Ctrl/Y は使用できません。未処理のエラーまたは割り込みの試みがあると、システム・エラー・メッセージが生成され、セッションがログアウトします。SPAWN コマンドに /TRUSTED 修飾子が指定されている場合を除き、SPAWN コマンドはキャプティブ・アカウント内では無効です。SPAWN は、MAIL および DEC Text Processing Utility (DECTPU) でも (組み込みプロシージャとして) 無効です。ユーザ指定のレキシカル関数の実行を防止するために、INQUIRE コマンドも無効になります。

表 7-2: キャプティブ・アカウントにより許可されないログイン修飾子

修飾子	説明
/CLI	代替のコマンド言語インタプリタの名前を指定します。
/COMMAND	デフォルトのログイン・コマンド・プロシージャをオーバーライドします。
/NOCOMMAND	デフォルトのログイン・コマンド・プロシージャの実行を無効にします。

表 7-2: キャプティブ・アカウントにより許可されないログイン修飾子 (続き)

修飾子	説明
/DISK	代替のデフォルト・ディスクを要求します。
/TABLES	代替の CLI テーブルの名前を指定します。

#### 7.2.4.1 キャプティブ・アカウントの設定

アカウントの作成時に次の修飾子を含めることで、AUTHORIZE を使用してキャプティブ・アカウントを定義します。

/FLAGS=(CAPTIVE)

キャプティブ・アカウントには、表 7-3 に説明する修飾子も必要です。

表 7-3: キャプティブ・アカウントの定義に必要な修飾子

修飾子	アクション
/LGICMD	キャプティブ・アカウントのログイン・コマンド・プロシージャを指定し、デフォルトのログイン・コマンド・プロシージャ (ユーザのデフォルト・ディレクトリにある LOGIN.COM) をオーバーライドします。
/UIC	固有の UIC グループを割り当てます。UIC グループの一意性を確認するには、AUTHORIZE の SHOW コマンドを次の形式で使用します。  SHOW [groupuic,*]  アカウントを独立したグループに維持することにより、キャプティブ・アカウントのユーザは、全ユーザからアクセス可能なファイルと、キャプティブ・アカウント自身が所有するファイルにのみアクセスできるようになります。また、そのアカウントはシステム・グループ (システム・パラメータ MAXSYSGROUP により変更された場合を除き、グループ値が 10 <sub>8</sub> 以下のグループ) に属さないことが保証されます。
/NOPASS-WORD または /FLAGS=LOCK-PWD	パスワードを設定します。キャプティブ・アカウントでは、パスワードを要求しないか、セキュリティ管理者のみがパスワードを変更できるようにパスワードをロックします。  一般的に、オープン・キャプティブ・アカウント (パスワードがないアカウント) よりも、ロックされたパスワードの方が望ましいです。ロックされたパスワードを割り当てる場合は、キャプティブ・アカウントの全ユーザにそのパスワードを知らせます。
/PRCLM	サブプロセスの上限を 0 に設定することで、アカウントからサブプロセスをスポンするのを防止します。システム・パラメータ PQL_MPRCLM (サブプロセスの下限値) が 0 に設定されていることを確認します。

必須設定以外に、アカウントには次の追加の特性を指定できます。

- キャプティブ・アカウントに対して、ウェルカム・メッセージと電子メールを無効にできます。これには、DISWELCOME、DISMAIL、およびDISNEWMAIL ログイン・フラグを設定します。
- 会話型アカウントの使用を、ローカル・ターミナルからのみに限定することができます。アカウントを作成する際に、修飾子 /NODIALUP、/NOREMOTE、/NOBATCH、および /NONETWORK を追加します。
- アプリケーションには、特別な要件が存在する場合があります。操作モードを制限するため、/NODIALUP などの追加の AUTHORIZE 修飾子をアカウントに適用しなければならない場合があります。プロセスを実行できる時間帯と曜日を制限することも検討します。
- /CLITABLES 修飾子を使用して DCL テーブルの特別なセットを定義したり、DCL コマンド・プロシーダを使用することで DCL をエミュレートすることができます。DCL コマンド・プロシーダを用いて DCL をエミュレートするよりも、DCL テーブルを定義の方が効率的です。DCL テーブルの定義の詳細については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の A-L にあるコマンド定義ユーティリティ (CDU) の説明を参照してください。/CLITABLES 修飾子により定義される DCL テーブルは、ネットワーク・ジョブ (TASK オブジェクトを使用するネットワーク・ジョブなど) で使用されない点に注意してください。
- 複数の特権を指定できますが、キャプティブ・アカウントに TMPMBX 以外の特権を指定しなければならない場合はまれです。
- キャプティブ・アカウントのディスク使用量は、必要な量に制限できます。

#### 7.2.4.2 キャプティブ・コマンド・プロシーダのガイドライン

サイトのキャプティブ・コマンド・プロシーダを記述する際には、必ず次のガイドラインに従ってください。

- コマンド・プロシーダで、DCL の READ/PROMPT コマンドを使用します。たとえば、日付を入力するようユーザに求めるには、コマンド・プロシーダに次のコマンドを入力します。

```
READ/PROMPT="Enter date: " SYS$COMMAND DATE
```

- キャプティブ・コマンド・プロシーダでは、INQUIRE コマンドの使用は避けます。このコマンドは、あらかじめ行っておく必要のある ON 宣言により処理されないと、プロセスが削除されるエラーを引き起こします。
- ユーザの入力を求めた場合、入力された内容を絶対にそのまま実行しないでください。まず、想定している内容とユーザの入力を比較し、アポストロフィ (')、アットマーク記号 (@)、ドル記号 (\$)、引用符 (")、アンパサンド (&)、ハイフン (-) などの不正な文字がないか調べます。

- 「"x」(x にはユーザが入力した文字が含まれる)という形式のコンストラクションの使用は避けます。ユーザが入力した記号を、制限付きコマンド・プロシージャを使用して評価することを許可しないでください。レキシカル関数を使用すると、コマンド・プロシージャが損なわれる可能性があります。
- キャプティブ・コマンド・プロシージャでは、@TT: という文字を含む行を実行することは避けます。
- ファイルの変更を検出するために、キャプティブ・コマンド・プロシージャとそのホーム・ディレクトリに、Audit ACE を適用します。Audit ACE の詳細については、9.2.1.2 項を参照してください。
- キャプティブ・アカウント・ユーザに、ファイルの作成およびファイルに対するその他の操作を実行する許可を与えた場合は、ログイン・コマンド・プロシージャとそのディレクトリに対する書き込みアクセス権が与えられないことを確認します。実行アクセス権は必要です。

コマンド・プロシージャの機能にテキストの準備が必要である場合は、ユーザにテキスト・エディタへのアクセス権を与えなければならない場合があります。ただし、注意が必要です。TECO や DECTPU などのエディタでは、ユーザがファイルを操作したり、エディタから出て DCL インタフェースに移れるため、危険性があります。このような環境を設計する際には、大部分のテキスト・エディタは(アカウントのアクセス権の範囲内で)ファイルを読み書きできることに留意してください。ユーザに必要なツールを提供しても、ユーザがキャプティブ環境から抜け出すことは許可しないエディタを提供するようにします。

例 7-3 と 例 7-4 に、特権アカウントと非特権アカウント用のコマンド・プロシージャの例を示します。

#### 例 7-3: 特権アカウント用のキャプティブ・プロシージャの例

---

```
$ if f$mode() .nes."INTERACTIVE" then $logout
$ term = f$logical("SYS$COMMAND")
$ if f$locate("_T", term) .eq.0 then $goto allow
$ if f$locate("_OP",term) .ne.0 then $logout
$allow:
$ set control=(y,t)
```

---

#### 例 7-4: 非特権アカウント用のキャプティブ・コマンド・プロシージャ の例

---

```
$ deassign sys$input
$ previous_sysinput == f$logical("SYS$INPUT")
$ on error then goto next_command
$ on control_y then goto next_command
$ set control=(y,t)
$
$next_command:
```

---

#### 例 7-4: 非特権アカウント用のキャプティブ・コマンド・プロシージャの例 (続き)

---

```
$ on error then goto next_command
$ on control_y then goto next_command
$
$ if previous_sysinput .nes. f$logical("SYS$INPUT") then deassign sys$input
$ read/end=next_command/prompt="$ " sys$command command
$ command == f$edit(command,"UPCASE,TRIM,COMPRESS")
$ if f$length(command) .eq.0 then goto next_command
$
$ delete = "delete"$ delete/symbol/local/all
$ if f$locate("@",command) .ne. f$length(command) then goto illegal_command
$ if f$locate("=",command) .ne. f$length(command) then goto illegal_command
$ if f$locate("F$",command) .ne. f$length(command) then goto illegal_command
$ verb = f$element(0," ",command)
$
$ if verb .eqs."LOGOUT" then goto do_logout
$ if verb .eqs."HELP" then goto do_help
$
$ write sys$output "%CAPTIVE-W-IVVERB, unrecognized command \",verb,\""
$ goto next_command
$
$illegal_command:
$ write sys$output "%CAPTIVE-W-ILLEGAL, bad characters in command line"
$ goto next_command
$
$do_logout:
$ logout
$ goto next_command
$
$do_help:
$ define sys$input sys$command
$ help
$ goto next_command
```

---

### 7.2.5 制限付きアカウント

限定アクセス・アカウントの中には、キャプティブ・アカウントよりも制限の緩い環境を必要とするものがあります。たとえばネットワーク・オブジェクトを実行するアカウントは、DCL に一時的にアクセスできる必要があります。そのようなアカウントは、キャプティブ・アカウントではなく、制限付きアカウントとして設定する必要があります。制限付きアカウントは、ログイン・シーケンスが完了すると、通常のアカウントと区別できなくなります。制限付きアカウントの目的は、SYLOGIN、LOGIN、およびそれらの子プロセスが完全に実行できる、信頼できるログインを確保することです。

制限付きアカウントを作成する際に、登録ユーティリティを使用し、次の修飾子を追加することで制限付きアカウントを定義します。

```
/FLAGS=(RESTRICTED)
```

このフラグにより、アカウントが制限付きアカウントとして設定されます。制限付きアカウントは、7.2.4 項に示したキャプティブ・アカウントと機能と同じ機能を提供します。ただし制限付きアカウントでは、システムとプロセスのログイン・コマンド・プロシージャを実行することで、DCL コマンド・レベルへのユーザ・アクセスを許可する点が異なります。

コマンド・プロシージャの開始後、ユーザに Ctrl/Y キー・シーケンスの入力を許可することが適切である場合があります。次に例を示します。

- 制限付きログイン・コマンド・プロシージャの実行時に、特定の時点でユーザに Ctrl/Y 機能を提供したい場合があります。例 7-4 に示すように、プロシージャの中で Ctrl/Y 機能を評価したい場所に、ON CONTROL\_Y コマンドを追加します。
- 最終的に制御をユーザに渡す制限付きコマンド・プロシージャを使用したい場合があります。たとえば、追加のセキュリティ検証を実行する SYLOGIN.COM コマンド・プロシージャが考えられます。その有効性を保証するには、プロシージャの実行が可能である必要があります。ただし、SYLOGIN.COM が必要な処理を終えたら、制御をユーザに渡すようにできます。このためには、そのアカウントを制限付きアカウントとしてマーク付けし、制御をユーザに解放する準備ができた時点で、DCL の SET CONTROL=Y コマンドを入力します。

## 7.2.6 自動ログイン・アカウント

特定のターミナルを使用する個人を、特定のアプリケーション・プログラムに強制的にログインさせるには、そのアプリケーション用の独立したキャプティブ・アカウントを作成します。続いて、システム管理ユーティリティ (SYSMAN) を使用して、対象となるユーザに対して新しいアカウントへの自動ログインを設定します。

自動ログイン用に設定したターミナルは、指定のアカウント用にのみ使用できます。これは、コンピュータを使い慣れていない人々に使用されるアプリケーション・ターミナルに最適です。

自動ログイン機能では、ユーザ名の入力を求めるプロンプトが表示されません。その他すべてのログイン機能 (システム・パスワード、第 1 および第 2 パスワード、およびメッセージ) は、有効にしてあれば通常どおり機能します。

パスワードの利用は任意です。ターミナルの設置場所にいるすべてのユーザにアカウントを開放する場合は、パスワードを無効にします。パスワードが不要である場合、ユーザがログイン時に入力するデータはありません。オペレーティング・システムは Break キーまたは Return キーが押されると自動的にターミナルのログインを行い、そのアカウントがキャプティブ・ログイン・コマンド・プロシージャの制御のもとにある場合、直ちにアプリケーションに入ります。



自動ログイン・ファイル (ALF) には、ターミナルと、アプリケーション・アカウントへのアクセス権限が付けられたユーザのリストがあります。しかし自動ログイン・アカウントは、ALF ファイルにリストアップされているターミナル以外のターミナルや他の場所からアクセスできる可能性があるため、特にパスワードが設定されていない場合、保護が必要になります。次の対策を講じます。

- 必要に応じて、AUTHORIZE 修飾子の /NODIALUP、/NONETWORK、および /NOREMOTE を使用して、ネットワーク・アクセスおよびダイアルアップ・アクセスを制限します。
- アカウントの UAF レコードの AUTOLOGIN フラグを設定します。このフラグにより、アカウントは、自動ログイン、バッチ、およびネットワーク代理にのみ使用可能になります。

### 7.2.7 ゲスト・アカウント

ゲスト・アカウントとは、共通のアカウントを通じて、システム上のリソースへの複数の遠隔ユーザ・アクセスを許可する、キャプティブ・アカウントまたは制限付きアカウントの形式です。たとえば、ネットワーク経由でユーザがシステムにアクセスして、問題を報告したり、会社の連絡事項を読む場合に使用します。

ゲスト・アカウントを設定することはお勧めできません。ゲスト・アカウントは、権限がいくら制限されていても、悪意のあるユーザに、システムのセキュリティを危うくする機会を与えてしまいます。ゲスト・アカウントが必要な場合でも、ほとんどは(限定アクセス・アカウントでもある)特別な代理ログイン・アカウントで処理できます。

ゲスト・アカウントがどうしても必要な場合は、アカウントの安全を確保するために、次の手順に従います。

- ゲスト・アカウントには、わかりにくいパスワードを使用し、パスワードを頻繁に変更します。GUEST/GUEST や USER/USER などの、簡単に推測できるようなアカウント名とパスワードの組み合わせは絶対に使用しないでください。
- アカウントの使用を許可する対象となるユーザのリストを維持します。パスワードを定期的に変更すると、このリストを最新の状態に保つ手助けになります。
- 独立した UIC グループにゲスト・アカウントを設定します。ゲスト・アカウントがシステム・グループに属していないことを確認します。
- AUTHORIZE の MODIFY コマンドを次のように使用して、デフォルトのログイン・コマンド・プロシージャを SYS\$MANAGER ディレクトリに配置します。

```
MODIFY guest-account/LGICMD=SYS$MANAGER:filename.COM
```

- ゲスト・アカウントを制限付きアカウントまたはキャプティブ・アカウントにします。これには、それぞれ AUTHORIZE 修飾子の /FLAGS=RESTRICTED または /FLAGS=CAPTIVE を設定します。
- ゲスト・アカウントが制限付きアカウントとして設定する場合、AUTHORIZE 修飾子の /PRCLM=0 を使用して、そのアカウントが作成できるサブプロセスの数を 0 に制限します。システム・パラメータ PQL\_MPRCLM も 0 に設定されていることを確認します。
- ゲスト・アカウントには TMPMBX 特権のみを割り当てます。
- エラー状態を処理するため、デフォルトのログイン・コマンド・プロシージャには次のコマンドを追加します。

```
SET ON
SET NOCONTROLY
ON ERROR THEN LOGOUT/BRIEF
```

- システムで LOGOUT がグローバル・シンボルとして定義されていて、それがコマンド・プロシージャを指している場合（これを確認するには、DCL の SHOW SYMBOL LOGOUT コマンドを入力します）、アカウントのデフォルトのログイン・コマンド・プロシージャに次の DCL コマンドを追加します。

```
DELETE/SYMBOL LOGOUT/GLOBAL
```

このコマンドは、ユーザがログアウト時に Ctrl/Y を押すことで制限付きアカウントを破る操作を不可能にします。

- 部外者がゲスト・アカウントからバッチ・ジョブを実行してシステム・リソースを不正に使用するのを防ぐには、アカウントを作成する際に AUTHORIZE 修飾子の /NOBATCH を追加します。
- ゲスト・アカウント UIC のディスク使用量を、必要な量に制限します。
- DCL の INQUIRE コマンドが、どのコマンド・プロシージャにも現れないようにします。

## 7.2.8 代理アカウント

一般的に代理ログイン・アカウントは、制限付きアカウントとして設定する必要があります。代理ログイン・アカウントは、遠隔ユーザがパスワードを指定しなくてもローカル・アカウントにアクセスできるようにします。代理ログイン・アカウントについては、12.3.3 項に説明があります。推奨事項の多くは、制限付きアカウントの場合と同じです。

## 7.2.9 外部認証アカウント

外部認証アカウントは、ユーザの SYSUAF レコードに EXTAUTH フラグでマーク付けされているアカウントです。これにより、該当するユーザは外部ユーザ ID と

パスワードを使用して、OpenVMS ログイン・プロンプトからログインできます。  
外部認証の詳細については、7.4 節を参照してください。

## 7.3 パスワードを使用したシステム・アクセスの制御

平均的なセキュリティ保護を必要とするサイトでは、必ずパスワードを使用する必要があります。より高度なセキュリティを必要とするサイトでは、多くの場合、生成パスワード方式（7.3.2.4 項を参照）だけでなく、システム・パスワードも利用します。

この節では、パスワード管理について説明します。

### 7.3.1 パスワードのタイプ

自動ログイン・アカウントを除き、すべてのユーザにはログイン用に少なくとも 1 つのパスワードが必要です。セキュリティ要件が中または高であるサイトでは、複数のパスワードを利用する場合があります（表 3-2 表 3-2 を参照）。

外部認証ユーザは、OpenVMS パスワード・プロンプトで外部パスワードを入力します。詳細については 7.4 節を参照してください。

この節では、DCL および AUTHORIZE コマンドを使用してパスワードを割り当てる方法を説明します。

#### 7.3.1.1 第 1 パスワード

AUTHORIZE を使用して新規ユーザのアカウントを開設する際には、そのユーザにユーザ名と初期パスワードを与える必要があります。一時初期パスワードを割り当てる際には、3.8 節「パスワードの保護に関するガイドライン」で推奨している、すべてのガイドラインに従ってください。パスワードを割り当てる際には、わかりやすいパターンを避けます。自動パスワード・ジェネレータを使用することも可能です。

AUTHORIZE を使用したアカウントの開設時に自動パスワード・ジェネレータを使用するには、ADD または COPY コマンドのいずれかに /GENERATE\_PASSWORD 修飾子を追加します。システムによって、自動的に生成されたパスワードの選択肢のリストが提示されます。これらのパスワードから 1 つを選択し、アカウントの設定作業を続けます。

---

#### 注意

---

/GENERATE\_PASSWORD 修飾子と /PWDMINIMUM 修飾子の併用には、いくつかの制限事項があります。生成パスワードの長さは、必ず 12 文字です（7.3.2.3 項を参照）。/PWDMINIMUM の値と、生成パスワードの間に矛盾がある場合、オペレーティング・システムによって短い方の値が使用されます。

---

AUTHORIZE を使用して指定するパスワードは、デフォルトでは期限切れとして定義されます。これにより、ユーザは最初にログインした時に、初期パスワードを強制的に変更しなければなりません。詳細については、7.3.2 項を参照してください。ユーザが正しく対処できるように、ユーザ教育に必ず最初のログインに関する情報を含めるようにします。AUTHORIZE を使用して定義するパスワードを事前に期限切れにしたい場合は、パスワードを入力する際に修飾子 /NOPWDEXPIRED を追加します。これは、ユーザが自分でパスワードを設定することが許可されないアカウントの場合に必要です。

事前に有効期限が切れているパスワードは、UAF レコードのリスト内では一目でわかります。パスワードの最終変更日のエントリに、次の注釈が付いています。

(pre-expired)

### 7.3.1.2 システム・パスワード

3.2.1 項「システム・パスワードの入力」では、特定のターミナルへのアクセスを制御するシステム・パスワードを説明しています。システム・パスワードは、次のような、不正使用のターゲットになる可能性があるターミナルへのアクセスを制御するために使用します。

- アクセスのためにダイヤルアップ回線または公衆データ・ネットワークを使用するすべてのターミナル
- 大学のコンピュータ室にあるターミナルなど、誰でもアクセスが可能で、セキュリティ保護が厳しくない回線に接続されたターミナル
- あまり頻繁に検査されないターミナル
- 予備デバイスとしてのみの使用が想定されているターミナル
- セキュリティ操作に確保しておきたいターミナル

システム・パスワードの実装には、次の手順を実行します。

1. 登録ユーティリティを起動し、次のコマンドを入力して、システム・パスワード用に SYSUAF データベースにレコードを作成します。

```
UAF> MODIFY/SYSTEM_PASSWORD=password
```

---

#### 注意

---

SYSUAF データベースにレコードを作成する必要があるのは、システムにシステム・パスワードを初めて設定する時のみです。ただし、レコードが存在しない場合、SET PASSWORD/SYSTEM コマンドを実行すると次のエラーが返されます。

```
%SET-F-UAFERR, error accessing authorization file  
-RMS-E-RNF, record not found
```

---

2. どのターミナルにシステム・パスワードが必要かを判断します。続いて、各ターミナルに対して、DCL の SET TERMINAL/SYSPWD/PERMANENT コマンドを入力します。適切なターミナルを選択したことを確認したら、ターミナルの設定作業がシステム・スタートアップ時に自動的に行われるように、上記のコマンドを SYS\$MANAGER:SYSTARTUP\_VMS.COM に組み込みます。ターミナルに対して、DCL の SET TERMINAL/NOSYSPWD/PERMANENT コマンドを呼び出すことで、そのターミナルに対する制限をいつでも取り除くことができます。
3. システム・パスワードを決めて、DCL の SET PASSWORD/SYSTEM コマンドを使用してそのパスワードを実装します。これには SECURITY 特権が必要です。コマンドを実行すると、ユーザ・パスワードと同じように、パスワードの入力を求められ、確認のために再度入力を求められます。自動パスワード生成を要求するには、/GENERATE 修飾子を追加します。

(DCL の SET HOST コマンドにより実現される) ログインの遠隔クラス用のシステム・パスワードの使用を有効にするには、AUTOGEN を使用して、デフォルトのターミナル属性パラメータの適切なビットを設定します。これは、パラメータ TTY\_DEFCHAR2 の第 19 ビット (16 進値で 80000) です。この値を設定した場合には、この機能を使用したくないターミナルごとに、DCL の SET TERMINAL/NOSYSPWD/PERMANENT コマンドを実行してシステム・パスワードを無効にする必要があります。先述のように、テストが済んだ SET TERMINAL コマンドを、SYS\$MANAGER:SYSTARTUP\_VMS.COM に組み込むことを検討してください。続いて、先に示した手順に従って、システム・パスワードを設定します。

システム・パスワードを選択する際には、3.8 節「パスワードの保護に関するガイドライン」の推奨事項に従ってください。長さは 6 文字以上で、意味のある単語ではない、アルファベットと数字で構成される文字列を選択します。システム・パスワードには有効期限はありませんが、頻繁にパスワードを変更するようにします。パスワードを知っている人物がグループを離れたら、必ずシステム・パスワードを直ちに変更します。システム・パスワードは、パスワードを知っておかなければならないユーザとのみ共有します。

システム・パスワードは、独立した UAF レコードに格納され、表示できません。DCL の SET PASSWORD/SYSTEM コマンド (システム・パスワードを設定および変更する通常的手段) では、パスワードを変更する前に、それまでのシステム・パスワードを入力する必要があります。古いパスワードを指定せずにシステム・パスワードを変更するには、次のコマンドのように、AUTHORIZE の MODIFY/SYSTEM\_PASSWORD コマンドを使用します。

```
UAF> MODIFY/SYSTEM_PASSWORD=ABRACADABRA
```

システム・パスワードの主な機能は、万人がアクセスできるポートに最前線となる防御を施し、侵入を試みる者がシステムの身元を知ってしまうことを防ぐことで

す。ただし、権限を持つユーザが、一部のターミナルでシステム・パスワードが要求されることを知らない場合、システム・パスワードを要求すると混乱が生じる場合があります。ターミナルやシステムの障害の誤報告を防ぐために、ユーザに割り当てられているターミナルの中で、どのターミナルがシステム・パスワードを要求するかをユーザに知らせます。

ダイヤルアップ回線または公衆アクセス回線経由のアクセス制御にシステム・パスワードを適用していない場合は、システム・パスワードを知っているユーザがごくわずかである場合があります。パスワードを知っている担当者に連絡が取れない場合や、担当者がパスワードを忘れてしまった場合は、運用に支障をきたします。この問題を解決するには、AUTHORIZE を起動し、MODIFY/SYSTEM\_PASSWORD コマンドを入力します。これには、SYSPRV 特権が必要です。

### 7.3.1.3 第 2 パスワード

セキュリティの保護レベルが高いサイトでは、ユーザ・アカウントに第 2 パスワードを要求できます。一般的にユーザは第 2 パスワードを知らず、第 2 パスワードを入力する監督者などの重要人物が同席する必要があります。業務によっては、アカウントが使用されている間、監督者が同席し続ける場合もあります。監督者は第 2 パスワードを空文字列に変更することで削除できるため、第 2 パスワードの有効性は、それを提供する監督者の信頼性に完全に依存しています。

パスワードを二重に使用すると手間がかかりますが、次のようなメリットがあります。

- 二重パスワードを広範に使用すると、監督者などの重要人物が各ユーザをチェックできるため、二重パスワードは、ログイン時の各ユーザの身元確認に役立ちます。
- 限定的に使用すると、二重パスワードにより、2 人が同席している場合にのみログイン可能なアカウントとなります。
- また二重パスワードは、ユーザが DECnet ソフトウェア経由でアカウントにアクセスする場合の、アクセス制御文字列の使用を防止できます。

セキュリティ要件が中程度であるサイトでは、パスワードが変更され、パスワード・ジェネレータの使用が強制された後に説明のつかない侵入が発生した場合に、二重パスワードを道具として使用できます。問題のアカウントを選択し、それらをこの制限の一時的な対象にします。第 2 パスワードによる個人確認を実施すると問題が生じないようであれば、人選に問題があることが判明します。権限を持つユーザが、そのアカウントを不正使用している 1 人または複数のユーザに、そのアカウントのパスワードを漏洩している可能性が高いと考えられます。

二重パスワードを実装するには、AUTHORIZE 修飾子の /PASSWORD を使用します。たとえば、新規アカウントに二重パスワードを適用するには、AUTHORIZE を起動し、次の形式の ADD コマンドを使用します。

ADD newusername /PASSWORD=(primarypwd, secondarypwd)

既存のアカウントに第 2 パスワードを適用するには、次の形式の MODIFY コマンドを使用します。

MODIFY username /PASSWORD=( "", secondarypwd)

このコマンドは、アカウントに対してすでに設定されている第 1 パスワードには影響しませんが、以降のすべてのログインにおいて、必ず第 2 パスワードを入力するよう求めるようになります。第 2 パスワードのパスワード有効期間と長さの下限値には、第 1 パスワードと同じ値が適用されます。このアカウントに対して /FLAGS=GENPWD 修飾子が指定されている場合、第 2 パスワードの変更は、自動パスワード・ジェネレータの制御下でのみ可能です。ユーザ名パラメータにワイルドカードを使用して、1 つのコマンドで複数のユーザに第 2 パスワードを適用することはできません。

---

#### 注意

---

DCL の SET HOST コマンドを使用して、遠隔アクセスを必要とするアカウントに第 2 パスワードを指定できますが、ネットワーク・ファイル・アクセスを必要とするアカウントには、アクセス制御文字列を使用して第 2 パスワードを指定することはできません。第 2 パスワードを持つアカウントをネットワーク・アクセス（遠隔ファイル・アクセスなど）に使用する場合、アカウントのアクセス元となるすべての遠隔ノードに対する代理アクセスを設定する必要があります。

---

#### 7.3.1.4 コンソール・パスワード

コンソール・ターミナルは、CPU の処理を制御するため、結果的にシステムの処理を制御します。セキュリティ要件が高いサイトでは、パスワード・セキュリティ機能が使用できる場合、その使用を検討すべきです。一部の VAXstation 3100 以降のモデルは、この機能を提供します。

コンソール・パスワードを有効にすると、オペレータはコンソール・モードで特権コマンドを使用する前に、コンソール・パスワードを入力する必要があります。特権コマンドには、次の 2 つのタイプが含まれます。

- SET, EXAMINE, DEPOSIT, FIND, SHOW など、メモリとレジスタを調査または変更するコマンド。
- BOOT や START など、コンソール・モニタから別のプログラムに CPU の制御を移すコマンド。パラメータなしで BOOT コマンドを実行する必要があるデフォルト・ブートの呼び出しは、特権コマンドではないため、パスワードなしで実行できます。

コンソール・パスワードの機能を有効にするには、次の手順を実行します。

1. 次のように特権コマンドを入力します。

```
> >> SET PSWD
```

2. コマンドを入力すると、コンソールはパスワードの入力を求めます。

```
1 > >>
```

新しいパスワードを入力し、Return キーを押します。パスワードの入力時に、コンソールはパスワードを表示しないことに注意してください。

パスワードは、長さ 16 文字、(0 ~ 9 と A ~ F の) 16 進数文字から成る文字列である必要があります。

3. パスワード文字列の長さが正しければ、コンソールは確認のため新しいパスワードを再入力するよう求めます。

```
2 > >>
```

新しいパスワードを再入力し、Return キーを押します。ここでもパスワードは表示されない点に注意してください。

4. 次のコマンドを使用して、パスワード・セキュリティ機能を有効にします。

```
> >> SET PSE 1
```

ワークステーションを特権モードにして、すべてのコンソール・コマンドをアクセス可能にするには、LOGIN コマンドを使用します。SHOW PSE コマンドを使用して、パスワード機能の現在のステータスを表示できます。1 が表示された場合、パスワード機能は有効で、0 が表示された場合は無効です。この機能を無効にするには、引数に 0 を指定して SET PSE コマンドを使用します。

パスワードは不揮発性のメモリに保存されるため、パスワードを忘れてしまった場合は、カスタマ・サポート・センターに連絡する必要があります。

### 7.3.1.5 認証カード

パスワードとアカウント情報を配布する方法の代わりに、認証カードやスマート・トークンと呼ばれる携帯型のデバイスをシステム・ユーザに持たせるサイトもあります。

認証デバイスには、ユーザのパスワードがプログラムで組み込まれています。ハードウェア設計の複雑さに応じて、これらのデバイスは追加のログイン情報(アカウント名や会計参照番号など)をサポートすることもできます。サード・パーティ・ベンダからは、さまざまな認証デバイスが発売されています。こうしたデバイスは、ログイン・プログラム (LOGINOUT.EXE) と通信するソフトウェア・モジュールによりサポートされています。認証カードをサポートする LOGINOUT ルーチンの詳細については、『*HP OpenVMS Utility Routines Manual*』を参照してください。



### 7.3.2 最低限のパスワード基準の適用

セキュリティ管理者は、AUTHORIZE を使用して、個々のユーザに最低限のパスワード基準を課すことができます。具体的には、AUTHORIZE によって指定される修飾子とログイン・フラグにより、パスワードの有効期限、期限切れになった時にユーザにパスワード変更を強制するかどうか、およびパスワードの最低の長さを制御します。

#### 7.3.2.1 パスワードの有効期限

AUTHORIZE 修飾子の /PWDLIFETIME を使用すると、パスワードの最大有効期間を設定できます。ユーザは有効期間が切れる前にパスワードを変更する必要があります。変更しないとアカウントへのアクセス権を失います。デフォルトでは、/PWDLIFETIME の値は 90 日です。この修飾子に対して別のデルタ時間値を指定することで、ユーザ・パスワードの変更頻度の要件を変更できます。たとえば、30 日ごとにパスワードを変更するようユーザに要求するには、この修飾子を /PWDLIFETIME=30-0 と指定します。

/PWDLIFETIME 修飾子は、第 1 と第 2 の両方のユーザ・パスワードに適用されますが、システム・パスワードには適用されません。ユーザのそれぞれの第 1 と第 2 のパスワードの最大有効期間は同じです。ただし、それぞれのパスワードは異なる時期に変更することができます。ユーザがパスワードの変更を完了すると、そのパスワードの時計が直ちにリセットされます。新しいパスワードの値は、/PWDLIFETIME により指定される期間だけ変更せずにおけます。

/NOPWDLIFETIME 修飾子は、第 1 と第 2 のパスワードが期限切れにならないことを指定します。

---

#### 注意

---

/NOPWDLIFETIME を指定すると、初期パスワードの再設定を強制するデフォルトの動作が無効になります。ただし、初期パスワードを再設定させたいけれども、パスワードが期限切れにならないようにしたい場合は、/PWDLIFETIME="9999-" と指定します。

---

AUTHORIZE には、第 1 と第 2 のパスワードの有効期限に関する 2 つのログイン・フラグもあります。2 つのフラグ PWD\_EXPIRED と PWD2\_EXPIRED は、/FLAGS 修飾子を使用して指定します。第 1 のフラグ PWD\_EXPIRED は、第 1 パスワードの有効期限が切れてから、ユーザにそのパスワードを変更する最後のチャンスが 1 回与えられ、なおかつユーザがパスワードを変更しなかった場合に設定されます。第 2 のフラグ PWD2\_EXPIRED は、第 2 パスワードの有効期限が切れてから、ユーザに第 2 パスワードを変更する最後のチャンスが 1 回与えられ、なおかつユーザがパスワードを変更しなかった場合に設定されます。PWD\_EXPIRED

と PWD2\_EXPIRED のいずれかが設定されていると、ユーザは最後のログイン時にパスワードを変更する最後のチャンスを行使しなかったため、アカウントへのログインはできません。

ユーザがパスワードの変更に成功すると、システムは直ちにフラグを適切にリセットします。第 1 パスワードが変更されると、PWD\_EXPIRED フラグは直ちに NOPWD\_EXPIRED になります。同様に、第 2 パスワードが変更されると、PWD2\_EXPIRED フラグは直ちに NOPWD2\_EXPIRED になります。セキュリティ管理者は、AUTHORIZE を起動してフラグをリセットし、パスワードを再設定するチャンスをユーザに再度与えることができます。

パスワードの有効期間を設定すると、ユーザは定期的にパスワードを変更しなければなりません。ユーザに応じて、異なる有効期間を設定できます。一般的に、重要なファイルにアクセスできるユーザには、パスワードの有効期間を最も短くする必要があります。

システム・パスワードの有効期間に制限はありません。システム・パスワードを定期的に変更するのは、セキュリティ管理者の責任です。

---

#### 注意

---

SYS\$PASSWORD\_HISTORY\_LIFETIME は、UAF パラメータの PWDLIFETIME よりも大きくする必要があります。  
SYS\$PASSWORD\_HISTORY\_LIFETIME の値を PWDLIFETIME よりも小さい値に設定すると、SYSUAF でのパスワードの有効期限が切れる前に、履歴ファイルが原因で期限切れになります。これでは、パスワード履歴ファイルの目的が達成できません。PWDLIFETIME パラメータの詳細については、7.3.2.2 項を参照してください。

---

#### 7.3.2.2 期限切れのパスワードの強制変更

デフォルトでは、ユーザはログイン時に、期限切れのパスワードを変更するように求められます。パスワードの有効期限が切れているユーザには、ログイン時に新しいパスワードの入力を求められます。このパスワード機能は、パスワードの有効期限が /PWDLIFETIME 修飾子で指定されている場合に限り有効です。

パスワードの強制変更を無効にするには、ADD または MODIFY コマンドに対して、次の修飾子を指定します。

```
/FLAGS=DISFORCE_PWD_CHANGE
```

パスワード強制変更機能を無効している場合、次のようにログイン・フラグをクリアすることで、その機能を再度有効にすることができます。

```
/FLAGS=NODISFORCE_PWD_CHANGE
```

ログイン時に、期限切れのパスワードを変更するように求められたユーザは、Ctrl/Y を押すことでログインをキャンセルできます。

---

注意

---

第 2 パスワードが有効で、第 1 と第 2 の両方のパスワードが期限切れになっている場合は、ユーザは両方のパスワードの変更を求められます。ユーザが第 1 パスワードを変更してから、第 2 パスワードを変更する前に Ctrl/Y を押した場合、ユーザはログアウトされ、パスワード変更は記録されません。

---

### 7.3.2.3 パスワードに必要な最低限の長さ

AUTHORIZE の修飾子 /PWDMINIMUM を使用して、第 1 と第 2 の両方の、すべてのパスワード選択における文字数の下限を指定できます。下限値の指定にかかわらず、ユーザは最大 32 文字までのパスワードを指定できます。

ユーザのパスワードの最低の長さは、デフォルトの 6 文字か、/PWDMINIMUM 修飾子により指定する (10 以下の) 別の値になります。

Alpha システムでは、パスワード・ジェネレータは指定の長さのパスワードを作成しますが、上限は 10 文字です。

VAX システムでは、パスワード・ジェネレータは  $n \sim n+2$  の長さの範囲でパスワードを作成します (最低の長さ  $n$  は 1 ~ 10 の範囲の値です)。そのため、生成されるパスワード (/GENERATE\_PASSWORD または SET PASSWORD/GENERATE) の長さが、/PWDMINIMUM 修飾子により設定される値と矛盾する可能性があります。

$n$  と、/PWDMINIMUM により設定される値との間に矛盾がある場合、オペレーティング・システムは小さい方の値を使用しますが、10 を超える値は使用しません。たとえば、/PWDMINIMUM 修飾子を使用して長さを 25 に設定した場合、オペレーティング・システムは 10 ~ 12 文字のパスワードを生成します。システムは、値の違いを通知しません。

AUTHORIZE 修飾子の /GENERATE\_PASSWORD により生成されるパスワードの長さは、ソース UAF レコード (DEFAULT レコードまたはコピーされた UAF レコード) の Pwdminimum フィールドに由来します。Pwdminimum フィールドは、/PWDMINIMUM により設定される値を使用して更新されるため、SET PASSWORD/GENERATE を使用して作成されるパスワードは、新しい値を使用します。

システム・パスワードには、最低限の長さはありません。ユーザ・パスワードに適用されるガイドラインは、同じようにシステム・パスワードにも適用されます。1 ~ 32 文字の長さのシステム・パスワードを選択します。

#### 7.3.2.4 生成パスワード

AUTHORIZE で /FLAGS=GENPWD 修飾子を使用すると、ユーザがパスワードを変更する際には、自動パスワード・ジェネレータの使用が強制されます。すべてのアカウントがこの修飾子を使用して作成されるサイトもあれば、セキュリティ管理者が選択できるサイトもあります。

侵入により危険にさらされることがあってはならない重要なデータにユーザがアクセスする予定がある場合、該当するユーザにはパスワード・ジェネレータを使用するよう要求します。

ポリシーとしてパスワード・ジェネレータを自発的に使用するよう要求していても、ユーザが協力しない場合は、該当するユーザ・アカウントに /FLAGS=GENPWD 修飾子を追加して、ユーザにパスワード・ジェネレータの使用を強制することができます。また、AUTHORIZE 修飾子の /FLAGS=LOCKPWD をユーザ・アカウントに追加して、ユーザがパスワードを変更できないようにすることも可能です。パスワードを変更する権限が与えられるのは、セキュリティ管理者のみになります。

#### 7.3.2.5 サイト・パスワードのアルゴリズム

オペレーティング・システムは、暗号化によりパスワードを漏洩から保護します。OpenVMS のアルゴリズムでは、パスワードを平文文字列から暗号化テキストに変換し、システム・ユーザ登録ファイル (SYSUAF.DAT) に保存します。パスワード確認の際には、平文のパスワードではなく、必ず暗号化されたパスワードに基づいて確認が行われます。システム・パスワードの暗号化には、常にオペレーティング・システムにとって既知のアルゴリズムが使用されます。

AUTHORIZE で /ALGORITHM 修飾子を使用して、ユーザのパスワードの暗号化にオペレーティング・システムが使用するアルゴリズムを定義できます。選択肢としては、現在の OpenVMS のアルゴリズムと、サイト固有のアルゴリズムがあります。各アカウントの第 1 パスワードと第 2 パスワードには、別々に暗号化アルゴリズムを指定できます。構文は次のとおりです。

```
/ALGORITHM=keyword=type [=value]
```

あるユーザに OpenVMS のパスワード暗号化アルゴリズムを割り当てるには、次のようなコマンドを入力します。

```
UAF> MODIFY HOBBIT/ALGORITHM=PRIMARY=VMS
```

サイト固有のアルゴリズムを選択する場合は、次のように、アルゴリズムを識別する値を指定する必要があります。

```
UAF> MODIFY HOBBIT/ALGORITHM=CURRENT=CUSTOMER=128
```

『HP OpenVMS Programming Concepts Manual』には、カスタム・アルゴリズムの使用に関する指示があります。セキュリティ管理者は、指定したアルゴリズム番号を認識してパスワードを適切に暗号化するコードを含んだ、サイト固有の

システム・サービスを作成する必要があります。この番号は、AUTHORIZE の MODIFY/ALGORITHM コマンドで使用する番号に対応する必要があります。

ユーザにサイト固有のアルゴリズムが割り当てられている場合、AUTHORIZE では SHOW コマンドにより表示される画面にこの情報が表示されます。

### 7.3.3 新しいパスワードの検査

通常、システムは新しいパスワードを、SYS\$LIBRARY に保存されているシステム辞書と照合し、パスワードが母国語の単語ではないことを確認します。またシステムは、ユーザのパスワードの履歴リストを維持し、新しいパスワードをこのリストと照合して、古いパスワードが再利用されていないことを確認します。サイトにとって特に重要な単語が使用されていないかパスワードをチェックするイメージを開発してインストールすることで、パスワードをさらに詳しく検査できます。

#### 7.3.3.1 システム辞書

DCL の SET PASSWORD コマンドは、ユーザが候補として入力したパスワードを (必要に応じて) 小文字に変換し、システム辞書内のエントリと照合して、パスワードが母国語の単語でないことを確認します。パスワード候補が辞書内に見つかった場合、有効なユーザ・パスワードとしては拒否されるため、ユーザは別のパスワードを入力する必要があります。

システム・パスワード辞書を変更して、サイトに関係のある単語を追加できます。システム辞書に単語を追加するには、次の手順を行います。次の手順では、許容できないと考えられるパスワードのファイルを保持することもできます。

1. 辞書に追加するパスワードを含むファイルを作成します。各パスワードは、次のように独立した 1 行に配置し、小文字にする必要があります。

```
$ CREATE LOCAL_PASSWORD_DICTIONARY.DATA
somefamous
localheroes
Ctrl/Z
```

2. SYSPRV を有効にして、ローカルで作成した追加を結合します。

```
$ SET PROCESS/PRIVILEGE=SYSPRV
$ CONVERT/MERGE/PAD LOCAL_PASSWORD_DICTIONARY.DATA -
_$ SYS$LIBRARY:VMS$PASSWORD_DICTIONARY.DATA
```

辞書検索を無効にするには、AUTHORIZE で、/FLAGS 修飾子に対して DISPWDDIC オプションを指定します。

#### 7.3.3.2 履歴リスト

オペレーティング・システムは、ユーザが過去 365 日間に使用したパスワードのリストを維持し、パスワード候補をこのリストと照合して、パスワードが再利用されていないことを確認します。

ユーザが新しいパスワードの作成に成功すると、システムは古いパスワードを履歴リストに入力して、ファイルを更新します。パスワード履歴リストは多数の単語を保持できますが、デフォルトでは上限が 60 になっています。この数を上回ると、ユーザは生成パスワードを使用する必要があります。パスワードは、パスワード履歴リストに 365 日間（または SYS\$PASSWORD\_HISTORY\_LIFETIME により設定されるデフォルトの期間）保持されます。ユーザ・アカウントを削除すると、システムによって、そのアカウントに属するすべてのパスワード記録が削除されます。

DCL の DEFINE コマンドを使用して、パスワード履歴リストの容量と有効期間のデフォルト設定を、表 7-4 に示す任意の値に変更できます。

表 7-4: パスワード履歴リストのデフォルト

システム論理名	デフォルト	最小	最大	単位
SYS\$PASSWORD_HISTORY_LIFETIME	365	1	28000	日数
SYS\$PASSWORD_HISTORY_LIMIT	60	1	2000	絶対数量

たとえば、履歴リストの容量を最大 60 個のパスワードから最大 100 個に変更するには、SYS\$MANAGER にあるコマンド・プロシージャ SYLOGICALS.COM に、次の行を追加します。

```
$ DEFINE/SYSTEM/EXEC SYS$PASSWORD_HISTORY_LIMIT 100
```

パスワード履歴リストの有効期間と、リストに記録できるパスワードの数には、相関関係があります。たとえば、パスワード履歴の有効期間を 4 年に延ばし、パスワードが 2 週間ごとに期限切れになっている場合、パスワード履歴の上限数量を少なくとも 104 (1 年間に 26 パスワードで、4 年間であるため) に増やす必要があります。パスワード履歴の有効期間と上限は動的に変更できますが、これらはクラスタを構成するすべてのノードで同じである必要があります。

第 2 パスワードを使用するサイトでは、第 2 パスワードの保存用に、アカウントに対するパスワードの上限を 2 倍にしなければならない場合があります。

パスワード履歴リストは SYS\$SYSTEM にあります。論理名 VMS\$PASSWORD\_HISTORY を使用することで、そのリストをシステム・ディスクから移動できます。この論理名を /SYSTEM/EXEC として定義し、SYS\$MANAGER:SYLOGICALS.COM に入れます。

履歴検索を無効にするには、AUTHORIZE で /FLAGS 修飾子に対して DISPWDHIS オプションを指定します。

### 7.3.3.3 サイト固有のフィルタ

システム辞書と履歴リストと照合してパスワードを検査する以外にも、サイト固有のパスワード・フィルタを作成して、パスワードが適切で、サイトから簡単に連想される単語ではないことを確認できます。フィルタを使用して、パスワード

の長さ、特殊文字または文字の組み合わせの使用、製品名や個人名の名前の使用をチェックできます。

サイト固有の単語のリストを作成するには、ソース・コードを記述し、共有イメージを作成し、そのイメージをインストールし、最後にシステム・パラメータを設定してポリシーを有効にします。手順については、『*HP OpenVMS Programming Concepts Manual*』を参照してください。

サイト固有のパスワード・フィルタをインストールし、有効にするには、SYSPRV 特権と CMKRNL 特権の両方が必要です。INSTALL および SYSPRV ファイル・アクセス監査が有効になっていて、システム・パラメータに対する必要な変更がオペレータ・コンソールに記録される場合、パスワード・フィルタ・イメージをインストールすると、複数のセキュリティ・アラームが生成されます。

共有イメージには、ユーザがパスワードを変更するとパスワード設定ユーティリティ (SET PASSWORD) によって呼びだされる、2 つのグローバル・ルーチンが含まれています。

---

#### 注意

---

この 2 つのグローバル・ルーチンの使用によって、平文のパスワード候補とそれに対応するクォドワードのハッシュ値の両方が得られます。悪意ある特権ユーザがこの機能を悪用すると、システムのセキュリティが危険にさらされるため、セキュリティ管理者は全員この機能を認識している必要があります。

パスワード・フィルタ・イメージとその親ディレクトリに、セキュリティ・アラーム用 ACE を適用することをお勧めします。手順については、『*OpenVMS Programming Concepts Manual*』を参照してください。

---

### 7.3.4 パスワード保護のチェックリスト

3.8 節「パスワードの保護に関するガイドライン」のすべての推奨事項に加えて、パスワードを保護するために次のガイドラインに従ってください。

- SYSTEM などの標準アカウントのパスワードが安全であり、定期的に変更されていることを確認します。アカウント (FIELD や SYSTEST など) が使用されていない場合は、AUTHORIZE 修飾子の /FLAGS=DISUSER を使用してアカウントを無効にできます。
- 外部または社内のサービス組織には、システムへのサービス提供に使用するアカウントのパスワードを決定させないでください。このようなサービス・グ

グループは、すべてのシステムで同じパスワードを使用する傾向があり、また通常そのアカウントは特権アカウントになっています。

- ほとんど使用されないアカウントに対しては、AUTHORIZE 修飾子の /FLAGS=DISUSER を設定し、必要な時にのみアカウントを有効にします。使用後直ちにパスワードを変更し、次にサービス・グループがパスワードを必要とする際に、サービス・グループに新しいパスワードを知らせます。
- 使用されなくなったアカウントは削除します。
- システムに対する攻撃の手掛かりとして使用される可能性があるため、ユーザ名のリストは、人目につく場所や盗難の可能性がある場所に放置しないでください。リスト・ファイルが必要な場合は、ACL を使用して、アクセスを特定の個人に限定します。
- 登録ファイルを適切に保護します。システム・ユーザ登録ファイル (SYSUAF.DAT)、ネットワーク代理登録ファイル (NETPROXY.DAT)、およびライト・リスト (RIGHTSLIST.DAT) は、システム・アカウント ([SYSTEM]) によって所有されていることに注意します。このグループには、他のユーザ・アカウントを作成しないでください。通常、これらの登録ファイルについては、デフォルトの UIC ベースのファイル保護で十分です。システム・アカウントは NET\$PROXY.DAT ファイルも所有しています。
- すべてのユーザが固有の UIC を持っていることを確認します。

次の対策を行うと、パスワードを知られる可能性が低くなり、パスワードが破られた場合や迂回された場合の損害をある程度まで抑えられます。

- 同じアカウントに複数のユーザがアクセスできるようにすることは避けます。
- システムに接続されているダイアルアップ回線の電話番号を保護し、ダイアルアップ回線に対するシステム・パスワードの設定 (SET TERMINAL/SYSPASSWORD) を検討します。
- ゲスト・アカウントや、顧客からの直接の問い合わせのためのアカウントなど、外部ユーザが使用できるアカウントがシステムにある場合は、これらのアカウントを、キャプティブ・コマンド・プロシージャに限定されるキャプティブ (限定アクセス) アカウントにします。キャプティブ・アカウントの設定の詳細については、7.2.4 項を参照してください。
- パスワードを必要としないアカウントは、すべてキャプティブ・アカウントにします。
- ユーザに対する特権の拡大は注意深く行います。
- 5.4.8 項「ファイル・セキュリティの最適化に関する推奨事項」で推奨しているすべてのテクニックを使用して、ファイルを保護します。



- オペレーティング・システムのコンポーネントが含まれるファイルが、適切に保護されていることを確認します（8.9.2 項を参照）。
- 代理ログイン・アカウントを設定してほかのノードからのファイル・アクセスのみを許可する場合は、AUTHORIZE 修飾子の /NOINTERACTIVE と /NOBATCH を使用します。そのアカウントに対しては、会話型ログインとバッチ・ログインが無効になります。

## 7.4 外部認証の有効化

外部認証により、ユーザは外部ユーザ ID およびパスワードを使用して、OpenVMS ログイン・プロンプトでログイン（サイン・オン）できます。外部認証機能としては、PATHWORKS と Advanced Server for OpenVMS の認証モジュールがサポートされており、OpenVMS ユーザに対して NT 互換の認証を行うことができます。

認証に成功すると、外部ユーザ ID は適切な OpenVMS ユーザ名にマッピングされ、適切なユーザ・プロファイルが取得されます。

デフォルトでは、外部認証はシステム・レベルとユーザ・レベルの両方で無効になっています。しかし、次の段落で説明するように、システム管理者が SYSTARTUP\_VMS.COM で論理名を定義し、SYSUAF でユーザ・アカウントをマーク付けしている場合、PATHWORKS または Advanced Server for OpenVMS を起動すると、外部認証が自動的に有効になります。Advanced Server を有効にして外部認証プロセスに参加するために、Advanced Server が動作するクラスタ・メンバでは追加設定は必要ありません。

ユーザがログインする前に、システム管理者は次の作業を実行して、外部認証を有効にする必要があります。

- SYSTARTUP\_VMS.COM での論理名の定義
- システム・ユーザ登録ファイル (SYSUAF) のユーザ・アカウントのマーク付け

これらの作業について、次の節で説明します。

### 外部認証論理名の定義

システム・レベルでは、SYS\$SINGLE\_SIGNON システム単位エグゼクティブ・モード論理名を定義することにより、外部認証を有効にします。

#### 注意

SYS\$SINGLE\_SIGNON 論理名は、SYSTARTUP\_VMS.COM で定義されていない場合は、PWRK\$ACME\_STARTUP.COM (PATHWORKS および Advanced Server for OpenVMS のスタートアップ・プロシージャ) により、自動的に 1 (有効) に定義されます。外部認証を無効にする場合や、SYS\$SINGLE\_SIGNON 論理名を別の値に設定する場合は、

PATHWORKS または Advanced Server for OpenVMS を起動する前に、SYSTARTUP\_VMS.COM で SYS\$SINGLE\_SIGNON を定義します。

スタンドアロンの Advanced Server 外部認証イメージのみをインストールし、Advanced Server 全体をインストールしていない場合は、論理名 PWRK\$ACME\_SERVER を定義する必要があります。Advanced Server をインストールする時、Advanced Server のファイルとプリント・サーバ・ソフトウェア全体をインストールするのではなく、外部認証イメージのみをインストールすることを選べます。詳細については、PATHWORKS (Advanced Server) または Advanced Server for OpenVMS の『インストールおよび構成ガイド』を参照してください。SYS\$SINGLE\_SIGNON 論理名ビットの詳細については、表 7-5 を参照してください。

---

次に例を示します。

```
$ DEFINE/SYSTEM/EXECUTIVE SYS$SINGLE_SIGNON 3
```

#### **SYSUAF** におけるユーザ・アカウントのマーク付け

ユーザ・レベルでは、SYSUAF レコードの EXTAUTH フラグにより、外部認証が有効になります。EXTAUTH フラグが設定されていると、当該ユーザが外部認証されることを示します。たとえば登録ユーティリティでは、次のようなコマンドを入力します。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD username /FLAGS=([NO]EXTAUTH)
UAF> MODIFY username /FLAGS=([NO]EXTAUTH)
```

登録ユーティリティの EXTAUTH フラグの詳細については、『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル』の A-L にある SSL for OpenVMS を参照してください。SYS\$GETUAI および SYS\$SETUAI システム・サービスの UAI\$\_FLAGS アイテム・コードにおける UAI\$V\_EXTAUTH ビットの詳細については、『HP OpenVMS System Services Reference Manual』の GETUTC-Z を参照してください。

### **7.4.1 外部認証のオーバーライド**

ユーザは、ログイン・プロンプトで OpenVMS ユーザ名の後に /LOCAL\_PASSWORD 修飾子を入力することで、外部認証の代わりにローカル認証を実行することを OpenVMS に通知できます。/LOCAL\_PASSWORD 修飾子を使用する場合、ユーザは OpenVMS のユーザ名とパスワードを指定する必要があります。

/LOCAL\_PASSWORD 修飾子を使用すると、システム管理者により確立されたセキュリティ・ポリシーを実質的にオーバーライドすることになるため、次の条件下でのみ使用が許可されます。

- ログイン先のアカウントが、認証された特権として SYSPRV を持っている場合。
- SYS\$SINGLE\_SIGNON 論理名で第 1 ビットが設定されている場合、(通常は外部認証される) 非特権ユーザはローカル認証を要求できます。

LOGINOUT に対する /LOCAL\_PASSWORD 修飾子の詳細については、『*HP OpenVMS Utility Routines Manual*』を参照してください。

#### 7.4.2 レイヤード・プロダクトおよびアプリケーションに対する影響

従来の SYSUAF ベースのユーザ名とパスワードに基づく認証メカニズムを使用する一部のレイヤード・プロダクトおよびアプリケーション (\$HASH\_PASSWORD または \$GETUAI/\$SETUAI を呼び出して OpenVMS パスワードを変更、フェッチまたは検証するソフトウェアなど) には、次のいずれかのケースで問題が発生する場合があります。

- ユーザの外部ユーザ ID と OpenVMS ユーザ名が異なる環境で外部認証が使用された場合
- ユーザの SYSUAF パスワードが外部ユーザ・パスワードと異なる場合

このようなケースでは、レイヤード・プロダクトまたはアプリケーションにおいてユーザ認証が失敗するという結果になります。

外部認証を受けるユーザに関しては、通常のシステム登録データベース (SYSUAF.DAT) を使用して、OpenVMS プロセス・プロファイル (UIC、特権、クォータなど) を構築し、個々のログイン制限を適用します。ただし、外部認証されるユーザと通常の OpenVMS ユーザとの間には、重要な相違点が 2 つあります。外部認証されるユーザには、次の点が該当します。

- SYSUAF に保存されているパスワードは、ユーザ認証に使用されるパスワードではありません。
- SYSUAF に格納されていて、OpenVMS プロセスの識別に使用されるユーザ名は、ログイン時のユーザ認証に使用される外部ユーザ ID とは、必ずしも同じではありません。

OpenVMS は、このような問題を最小限に抑えるために、ユーザの SYSUAF と外部ユーザ・パスワードを同期させようとします。ユーザの外部パスワードの最新コピーは SYSUAF に保持されますが、外部パスワードに OpenVMS で無効な文字が含まれている場合や、SYSUAF のパスワード同期がシステム管理者によって無効

にされているような場合は、SYSUAF に保持されません。パスワード同期は、デフォルトでは有効になっています。

外部同期を有効にする場合は、次の操作を実行して、従来の SYSUAF ベースの認証を使用するレイヤード・プロダクトまたはアプリケーションとの非互換性を最小限にすることをお勧めします。

- パスワード同期は無効にしないでください。
- 外部ユーザ・パスワードは、OpenVMS の有効なパスワード文字セット (A ~ Z, 0 ~ 9, アンダースコア (\_), およびドル記号 (\$)) の文字に限定します。
- 外部認証サービスと OpenVMS の両方で、ユーザに同じユーザ名を割り当てます。
- 複数のユーザに、同じユーザ名またはユーザ ID を割り当てないでください。

\$GETUAI および \$SETUAI システム・サービスは、外部パスワードをサポートしていません。これらのサービスは、SYSUAF に格納されているパスワードのみを対象に動作し、更新は外部認証サービスに送信されません。これらのサービス呼び出してパスワードや更新をチェックするソフトウェアを使用するサイトでは、外部認証を有効にしないでください。HP は、将来のリリースで、外部パスワードをサポートする新しいプログラミング・インタフェースを提供する予定です。

### 7.4.3 新しいパスワードの設定

外部認証を受けるユーザの場合、DCL の SET PASSWORD コマンドは、外部認証機能にパスワード変更要求を送信し、そのユーザの OpenVMS システム上でパスワードを変更します。

システム管理者は、外部認証機能により提供されるユーティリティを使用することで、外部認証を受けるユーザのパスワードを設定できます。NT 互換の認証の場合、PATHWORKS および Advanced Server for OpenVMS の ADMINISTRATOR SET PASSWORD コマンドを使用できます。この方法を使用すると、新しいパスワードは直ちに外部認証機能に通知されます。

### 7.4.4 パスワードとユーザ名における大文字小文字の区別

OpenVMS のユーザ名プロンプトに対して、ユーザ名を引用符で囲んで指定することで、大文字小文字の区別があるユーザ名を入力できます。ユーザ名を引用符で囲まない場合、LOGINOUT はユーザ名を大文字に変換します。

OpenVMS システムの本来の動作に戻すには、SYS\$SINGLE\_SIGNON 論理名の強制大文字変換構成ビット (第 3 ビット) を設定します。詳細については、表 7-5 を参照してください。

OpenVMS と LAN Manager のユーザ名は、大文字小文字が区別されません。そのため、OpenVMS のユーザ名または LAN Manager のユーザ ID を入力する場合、引用符は必要ありません。

LAN Manager のユーザ ID とパスワードに有効な文字は、標準の IBM 拡張 (8 ビット) ASCII 文字セットに属します。LOGINOUT と SET PASSWORD は、これらの文字列の大文字と小文字を維持したまま LAN Manager に渡しますが、外部認証サービスはこの文字セットに従って両方の文字列を大文字に変換します。

LAN Manager のパスワードには、OpenVMS のパスワードでは無効な文字を使用できません。LAN Manager のパスワードに、OpenVMS のパスワードでは無効な文字が含まれている場合、パスワード同期は行われず、メッセージが発行されます。

OpenVMS のパスワードは、7 ビット ASCII 文字の A ~ Z, 0 ~ 9, \_ , および \$ に限られます。

#### 7.4.5 ユーザ名マッピングおよびパスワード検証

外部認証を受けるには、ユーザは OpenVMS ログイン・プロンプトに対して自分の外部ユーザ ID とパスワードを入力します。OpenVMS はユーザ名マッピングの実行時に、まず SYSUAF の中で一致する名前を探し、見つかった場合はその名前を使用します。見つからなかった場合は、外部認証データベースに一致するユーザ ID を照会します。認証に成功すると、適切なユーザ・プロファイルを取得するために、LAN Manager のユーザ ID は適切な OpenVMS ユーザ名にマッピングされ、ログイン・シーケンスが完了します。

外部認証は、(DECwindows を含む) 会話型ログインと、(代理ログインが使用されるか、ユーザ ID/パスワードが渡される) ネットワーク・ログインにおいてサポートされます。

システムの外部認証が有効になっている場合、DECnet プロキシまたは自動ログイン (ALF) データベースに指定されている対象ユーザ名が、SYSUAF に存在する必要があります。DECnet プロキシを使用したい外部認証ユーザは、SYSUAF ファイルと LAN Manager データベースにおけるユーザ名が同じである必要があります。

DECnet プロキシを使用する場合は、OpenVMS および LAN Manager のドメイン全体で一意的なユーザ名を維持することが重要です。SYSUAF ファイルと LAN Manager データベースの同じユーザ名が、それぞれ異なるユーザを示す場合、代理としてのこのユーザ名の使用は、あいまいになります。LOGINOUT がログインのためにその名前を OpenVMS ユーザ名として扱う一方で、LAN Manager にある同じ名前が別の OpenVMS ユーザ名にマッピングされる可能性があります。このようになるのは、名前マッピング規則により、OpenVMS は LAN Manager の前に SYSUAF で一致する名前を検索するためです。

外部認証を受けるユーザは単独のパスワードを持つと見なされ、通常の OpenVMS パスワード・ポリシー（パスワードの有効期限、パスワード履歴、パスワード長の下限と上限）には従いませんが、その代わりに、外部認証機能の任意の定義済みポリシーに従います。無効なアカウント、モード時刻の制限、クォータなど、その他すべての OpenVMS アカウントの制限は有効なままになります。

外部認証を受けるユーザは、その SYSUAF レコードで EXTAUTH フラグが設定されていることにより識別されます。アカウントの EXTAUTH フラグが設定されていない OpenVMS ユーザは、外部認証の影響を受けません。

#### 7.4.6 パスワード同期

パスワードは外部認証機能データベースを使用して検証されますが、OpenVMS は、外部パスワード・フィールドと SYSUAF パスワード・フィールドの同期を維持しようとします。

パスワード同期は、デフォルトでは有効になっています。

同期が行われるのは、外部認証ログインが正常に完了した時点です。外部パスワードが、SYSUAF ファイルに保存されているパスワードと異なる場合、LOGINOUT は外部パスワードを使用して SYSUAF パスワード・フィールドを更新します。OpenVMS と外部認証機能で利用できる有効な文字セットの違いにより、同期が不可能になる場合があります。

必要に応じて、パスワード同期を選択的に無効にできます。パスワード同期の有効/無効を制御する SYS\$SINGLE\_SIGNON 論理名ビットの詳細については、表 7-5 を参照してください。

#### 7.4.7 SYS\$SINGLE\_SIGNON 論理名ビットの指定

SYS\$SINGLE\_SIGNON システム単位エグゼクティブ・モード論理名は、外部認証の操作全般を制御します。この論理名は 16 進文字列に変換され、ビット・ベクタとして扱われ、各ビットが個別のコンポーネントを制御します。

表 7-5 に、右から左に向かって順に番号が付けられた SYS\$SINGLE\_SIGNON 論理名の各ビットの定義を(最下位ビットから順に)示します。

表 7-5: SYS\$SINGLE\_SIGNON 論理名ビット

ビット番号	ステータス	説明
0	ON	外部認証を有効にします。SYSUAF ファイルで、外部認証を受けるよう指定されているユーザは、ログインに外部認証機能を使用します。

表 7-5: SYS\$SINGLE\_SIGNON 論理名ビット (続き)

ビット番号	ステータス	説明
	OFF	外部認証を無効にします。ローカル認証が有効である (つまり第 1 ビットが ON である) 場合、システムはユーザの通常の SYSUAF ユーザ名およびパスワードを使用してローカル認証を試みます。ローカル認証が無効になっている場合、外部認証を受けるユーザには、ログインが許可されません。
1	ON	ローカル認証を有効にします。第 0 ビットが OFF である場合、システムは、ローカル認証を使用してユーザを自動的にログインさせます。システムは、ユーザの SYSUAF レコードの EXTAUTH フラグを無視します。第 0 ビットが ON であるにもかかわらず外部認証サーバが動作していない場合、ユーザは /LOCAL_PASSWORD 修飾子を使用してローカル認証を要求できます。
	OFF	ローカル認証を無効にします。ユーザは /LOCAL_PASSWORD 修飾子を使用して、ローカル認証を強制することができます。第 1 ビットが OFF である場合、この修飾子を使用するには SYSPRV 特権が必要です。
2	ON	HP により予約されています。
	OFF	HP により予約されています。
3	ON	ログイン時にターミナル入力を強制的に大文字にします。これは、ログイン・デバイスに対する RMS ROP\$V_CVT オプションと同等です。このビットを設定すると本来の OpenVMS の動作に戻りますが、ユーザ名とパスワードの大文字と小文字を区別して入力することはできなくなります。
	OFF	ログイン時のターミナル入力の大文字への強制変換を無効にします。
4	ON	ローカル・パスワード同期を無効にします。システムは、外部認証機能から SYSUAF へのパスワード同期を行いません。
	OFF	ローカル・パスワード同期を有効にします。ログインの成功時に、ログインに使用された外部パスワードの OpenVMS ハッシュ値を計算し、SYSUAF ファイルにそのハッシュ値を格納することにより、システムは SYSUAF パスワードと外部パスワード (それらが異なる場合) の同期を試みます。
31	ON	ユーザがログインする際または SET PASSWORD コマンドを使用する際に表示される、OPCOM デバッグ・メッセージを有効にします。これらのメッセージは、外部認証の設定に関して問題が生じたときの診断に役立ちます。
	OFF	OPCOM デバッグ・メッセージを無効にします。

SYS\$SINGLE\_SIGNON が定義されていないか、無効な 16 進文字列になる場合、すべてのビットは OFF であると見なされます。

次の定義例は、外部認証 (第 0 ビット) を有効にします。他のコンポーネントはすべてデフォルト値を取ります。

```
$ DEFINE/SYSTEM/EXECUTIVE SYS$SINGLE_SIGNON 1
```

次の定義例は、外部認証を有効にして (第 0 ビット)、ユーザ名プロンプトでのターミナル入力を強制的に大文字にし (第 3 ビット)、パスワード同期を無効にします (第 4 ビット)。

```
$ DEFINE/SYSTEM/EXECUTIVE SYS$SINGLE_SIGNON 19 !19 HEX
```

### HP DECnet-Plus の要件

SYSUAF アカウント・レコードで EXTAUTH ビットが設定されているユーザは、外部認証パスワードがすべて大文字である場合を除き、DECnet-Plus が稼働するシステムでは明示的なアクセス制御文字列を使用できません。

たとえば、次のコマンドを入力したとします。

```
$ DIRECTORY nodename "username password" ::
```

ここで *nodename* は DECnet-Plus が稼働するシステムで、*username* は EXTAUTH アカウントです。DECnet-Plus は、*password* の文字列を、外部認証エージェント (PATHWORKS または NT ドメイン・コントローラ) に渡す前に、大文字に変換します。

解決方法は、次の 2 つがあります。

- DECnet-Plus を使用しており、明示的なアクセス制御文字列を使用する必要がある場合、NT のパスワードを大文字で定義します。
- 機能を実行するために明示的なアクセス制御文字列を使用しなくてもよいように、DECnet-Plus ノードに代理アカウントを設定します。

### DECnet-Plus および NET\_CALLOUTS パラメータ

外部認証を有効にして DECnet-Plus for OpenVMS を実行するには、システム・パラメータ NET\_CALLOUTS を 255 に設定します。これにより、ユーザ確認と代理検索が、DECnet ではなく LOGINOUT で行われます。

### POP サーバでの接続の失敗

POP (Post Office Protocol) サーバは、OpenVMS システムでの接続の認証に外部認証を使用しません。このことが原因で、次のいずれかの条件が存在する場合に、接続の試みが失敗することがあります。

- 外部ユーザ ID が OpenVMS ユーザ名と異なる。



- OpenVMS パスワードが、外部ユーザ・パスワードと同期していない。

#### 7.4.8 ACME (Authentication and Credentials Management Extensions) サブシステム

この節では、OpenVMS システムでユーザを認証する必要があるアプリケーションに対して外部認証機能を提供する、SYS\$ACM システム・サービスを有効にする方法を説明します。

ACME (Authentication and Credentials Management Extensions) サブシステムは、認証および persona ベースの資格情報サービスを提供します。アプリケーションは、これらのサービスを使用してユーザと対話し、次の 1 つ以上の機能を実行できます。

- ユーザ認証
- パスワードの変更
- ペルソナの作成および変更

ACME は、標準的な OpenVMS 認証および外部認証ポリシーをサポートします。したがって、アプリケーションでは、システムの LOGINOUT および SET PASSWORD コンポーネントと同じ仕組みを利用します。

##### 7.4.8.1 ACME サブシステムの概要

ACME サブシステムは、次のコンポーネントで構成されています。

- SYS\$ACM システム・サービス

SYS\$ACM は、コンテキスト駆動型のシステム・サービスです。このサービスは、アプリケーションに変更を加えることなく、アプリケーションがさまざまな認証ダイアログに透過的に対応できるように設計されています。アプリケーションは SYS\$ACM を呼び出して、authenticate-principal や change-password などの機能を実行します。認証に成功すると、サービスは persona の形式で、ユーザの完全なセキュリティ・プロファイルを返すことができます。SYS\$ACM の詳細については、『*HP OpenVMS System Services Reference Manual*』および『*HP OpenVMS Programming Concepts Manual*』を参照してください。

- ACME\_SERVER プロセス

ACME\_SERVER プロセスは、1 つまたは複数の認証ポリシーをサポートするマルチスレッド・サーバです。各認証ポリシーは、標準インタフェースを介して ACME\_SERVER プロセスに「プラグ・イン」される、ACME エージェントの共有イメージを設定することによってインストールします。サーバは、定義済みのシーケンスに従って、各 ACME エージェントを順に呼び出すことによって、認証シーケンスを処理します。また ACME エージェン

トには、認証シーケンスにおけるエージェントの対話方法に関して特定の規則を課す役目もあります。

- ACME エージェント

各 ACME エージェントは、標準の OpenVMS 認証ポリシーの一部を補足または置き換える、単独の認証ポリシーを定義します。OpenVMS は、現在次の 2 つの ACME エージェントをサポートしています。

- VMS – 標準の OpenVMS 認証ポリシーを提供する OpenVMS ACME エージェントです。
- MSV1\_0 – Microsoft® NT LAN 分散認証プロトコルを使用して外部認証を提供する、Advanced Server for OpenVMS ACME エージェントです。このエージェントは、Advanced Server for OpenVMS レイヤード・プロダクトのインストールに付属します。

- DCL の SET および SHOW SERVER ACME コマンド

SET および SHOW SERVER ACME コマンドを使用して、ACME サブシステムの設定と管理を行うことができます。

#### 7.4.8.1.1 ACME エージェントの動作環境

ACME サブシステムは、相互に対話をして認証要求を処理できる、複数の ACME エージェントをサポートしています。このような対話は、制御された方法で行われる必要があります。

ユーザ認証ダイアログの処理が進行中の時、1 つの ACME エージェントが制御側エージェントになり、その他のエージェントはセカンダリ・エージェントとしてバックグラウンドで動作します。

制御側エージェントがユーザ名とパスワードのプロンプトを指示し、最終的にユーザを確認する役割があります。セカンダリ・エージェントは、それぞれが他のエージェントとどのように対話するように設定されているかに応じて、メッセージの表示、追加パスワードの要求、資格情報の発行、認証要求の拒否を行うことができます。

#### 7.4.8.1.2 ACME エージェントの順序

特定の認証要求の制御側エージェントになる ACME エージェントは、次の 2 つのいずれかの方法で決定されます。

- SYS\$ACM の呼び出しが、特定の ACME エージェント・ドメインの呼び出しをターゲットとしている。ドメインとは、主名/ユーザ名のマッピングと、(ACME エージェントにより定義された) 対応するユーザ資格情報のセットです。
- エージェントが、ユーザの主名を、ドメイン内の有効なユーザ名に正しくマッピングする最初のエージェントである。

このため、ACME エージェントの設定順序は重要です。2 つ以上の ACME エージェント・ドメインに同じ主名が存在し、SYS\$ACM 呼び出しで ACME エージェント・ドメインが指定されていない場合、マッピングに最初に成功したエージェントが、認証要求を制御します。これは、主名により、実際に 2 人の異なるユーザが識別された場合には望ましくないことがあります。デフォルトでは、VMS ACME エージェントが先に設定されます。

#### 7.4.8.2 認証ポリシー

認証ポリシーは、ユーザ識別属性、認証属性、および登録属性の特定の組み合わせにより定義されます。ポリシー属性には、次の要素が含まれます。

- 識別構文  
単純ユーザ名と、ドメイン/領域/主名の組み合わせが含まれます。
- 認証トークン・メカニズム
- トークン再利用フィルタ  
パスワード辞書、パスワード履歴、パスワードで利用できる文字セット、パスワード長の上限と下限、強制変更スケジュール、および有効期限が含まれます。
- 侵入検出
- 大文字小文字の区別
- アクセス制限  
時間帯、曜日、およびアクセスのタイプが含まれます。
- ユーザ・アカウント制御  
アカウント・ロック（無効化）およびアカウントの有効期限が含まれます。
- 資格情報  
ユーザとグループの識別子と特権が含まれます。

現在、次の 2 つの認証ポリシーがサポートされています。

- 標準 OpenVMS ポリシー
- Advanced Server for OpenVMS 分散認証ポリシーを使った外部認証

##### 7.4.8.2.1 OpenVMS ポリシー

OpenVMS ポリシーは、機能が豊富で、大文字小文字が区別されない、パスワード・ベースの認証ポリシーです。シングル・パスワードまたは二重パスワードのアカウント、パスワードの有効期限、パスワードのロック、パスワード長の下限、システム生成パスワード、侵入検出および侵入回避、パスワード辞書フィルタおよび履歴フィルタ、モード・アクセス制限、アカウントの有効期限、およびアカウントのロックが含まれます。

ユーザの資格情報は、ユーザのグループとメンバの識別コード (UIC)、特権、およびライト識別子で構成されます。この情報は、システム登録 (SYSUAF.DAT) データベースとライト識別子 (RIGHTSLIST.DAT) データベースに保存されます。

システム登録データベースには、ユーザがいつどのようにシステムにアクセスできるかに関する情報も含まれます。これらのモード制限により、時間帯、曜日、アクセスのタイプ (ダイアルアップ、遠隔、バッチなど) に基づいてアクセスが制限されます。

OpenVMS の資格情報は persona に保存されます。persona は、保護された、カーネル・ベースのデータ構造です。

#### 7.4.8.2.2 Advanced Server for OpenVMS のポリシー

Advanced Server for OpenVMS MSV1\_0 の認証ポリシーは、Microsoft LAN Manager ドメイン・プロトコルに基づく分散認証ポリシーです。この認証ポリシーは、パスワードとチャレンジ・レスポンス (NTLM) のメカニズムをサポートします。このポリシーは、大文字小文字の区別があるパスワード、パスワードの有効期限、パスワード変更までの時間の下限、およびアカウントのロックをサポートしています。

ユーザの資格情報は、ユーザのシステム識別子 (第 1 と第 2 の SID)、および特権で構成されます。

Advanced Server for OpenVMS の資格情報は NT persona 拡張に保存されます。この拡張は、Advanced Server データベースにより Microsoft ユーザ名にマッピングされている OpenVMS ユーザ名の OpenVMS 視覚情報を含んでいる標準の persona に添付されています。

#### 7.4.8.3 ACME サブシステムの制御

ACME サブシステムの操作の制御は、次の要素により管理されます。

- DCL の SET および SHOW SERVER ACME コマンド  
ACME\_SERVER プロセスおよびエージェントを起動、停止、および設定します。
- SYSUAF ユーザ・フラグ  
標準認証と外部認証、およびパスワード同期が可能なアカウントを選択します。SYSUAF ユーザ・フラグには、EXTAUTH、VMSAUTH、および DISPWDSYNCH があります。
- SECURITY\_POLICY ビット・システム・パラメータ  
システム全体を対象として、特定の ACME サブシステム機能を制御します。

#### 7.4.8.3.1 SET および SHOW SERVER ACME コマンド

これらのコマンドは、ACME サブシステムを起動、停止、および設定します。

ACME\_SERVER プロセスは、システムのブート時に、VMS ACME エージェントが設定された状態で自動的に起動します。

サーバを手動で起動または停止するには、次のコマンドを使用します。

```
$ SET SERVER ACME/START
$ SET SERVER ACME/EXIT [/ABORT]
```

VMS ACME エージェントを設定するには、次のコマンドを使用します。

```
$ SET SERVER ACME/CONFIGURE=(NAME=VMS)
```

MSV1\_0 ACME エージェントを設定するには、  
SYS\$STARTUP:NTA\$STARTUP\_NT\_ACME コマンド・プロシ  
ジャを実行するか、次のコマンドを使用します。

```
$ SET SERVER ACME/CONFIGURE=(NAME=MSV1_0,CRED=NT, FAC=PWRK)
```

---

#### 注意

MSV1\_0 ACME エージェントを使用するには、Advanced Server 製品がインストール済みで、実行中である必要があります。

---

ACME エージェントの設定が完了した後、次のコマンドを使用して ACME エージェントを有効にします。

```
$ SET SERVER ACME/ENABLE[=NAME=agent]
```

エラー情報は、ACME サブシステムのログ・ファイルである  
SYS\$MANAGER:ACME\$SERVER.LOG に書き込まれます。

ACME サブシステムの状態を表示するには、次のコマンドを使用します。

```
$ SHOW SERVER ACME [/FULL] [/AGENT=agent]
```

問題を診断するには、次のようにトレースを有効にします。

```
$ SET SERVER ACME/TRACE=n
```

これらのコマンドの詳細については、『*OpenVMS DCL* デictionary』を参照してください。

#### 7.4.8.3.2 新しい SYSUAF フラグ

次の新しいフラグは、VAX および Alpha システムの SYS\$SETUAI、  
SYS\$GETUAI、および登録ユーティリティにより操作できます。これらのフラグを  
認識するのは、Alpha 上の ACME サブシステムのみです。

フラグ	説明
VMSAUTH	EXTAUTH フラグによって外部認証が要求される場合に、アカウントが標準 (SYSUAF) 認証を使用できます。アプリケーションは、外部認証を通常なら使用するユーザ・アカウントに対して標準 VMS 認証を要求するために SYS\$ACM を呼び出す時に、解釈の VMS ドメインを指定します。
DISPWDSYNCH	このアカウントの外部パスワードを同期しません。システム単位のパスワード同期制御に関しては、SECURITY_POLICY システム・パラメータの GUARD PASSWORD 制御ビットを参照してください。

#### 7.4.8.3.3 新しいシステム・パラメータ **SECURITY\_POLICY** のビット・マスク値

次の新しいセキュリティ・ポリシー・ビットは、Alpha 上のシステム単位の ACME サブシステム操作を制御します。

- Guard Passwords

システム全体で ACME エージェント間のパスワード同期を無効にするには、このビットを設定します。これは、LOGINOUT に対する SYS\$SINGLE\_SIGNON 論理名ビット・マスク値 4 と、機能の上では同等です。

16 進値は 200 です。

- Allow NoAuthorization

期限が切れているか、またはモードの制限でアカウントの使用が禁止されている SYSUAF レコードに主名がマッピングされているユーザに対して、特権アプリケーションの認証が正常に行われるようにするには、このビットを設定します。無効であるかパスワードの有効期限が切れている SYSUAF レコード (従来の OpenVMS 認証の場合) は、この方法では認証を回避できません。SECURITY 特権のあるアプリケーションが登録チェックに優先するには、SYS\$ACM の ACME\$M\_NOAUTHORIZE 機能修飾子を指定します。

16 進値は 400 です。

- Ignore ExtAuth and VMSAuth SYSUAF flags

外部認証を使用して SYSUAF ファイルのすべてのレコードのマッピングを許可するには、このビットを設定します。

16 進値は 800 です。

## 7.5 ログイン・プロセスの制御

この節では、権限のないユーザからシステムを保護するために設計された、オペレーティング・システムの多くの機能を説明します。

## 7.5.1 ログイン時の情報表示

この節では、通知、ウェルカム・メッセージ、最終ログイン情報、および新着メール・メッセージなど、ログイン時にデフォルトで表示されるさまざまな情報の表示を制御する方法を説明します。ログイン制限の影響を理解できるように、オペレーティング・システムがシステム・ユーザ登録ファイル (SYSUAF.DAT) のログインのフィールドをどのように処理するかも説明します。さらに、この節では安全ターミナル・サーバの使用法と、侵入検出の設定方法も説明します。

### 7.5.1.1 通知メッセージ

システムで通知メッセージを表示するには、サイト固有のスタートアップ・コマンド・プロシージャ SYS\$MANAGER:SYSTARTUP\_VMS.COM の中で、システム論理名 SYS\$ANNOUNCE を定義します。この方法は『*OpenVMS システム管理者マニュアル*』で説明しています。通知メッセージはログイン時に表示されます。

ここ定義した内容は、システムの全ユーザに影響します。このメッセージはオペレーティング・システムの身元に対する手掛かりになる可能性があるため、このメッセージを表示しないことも選択できます。

### 7.5.1.2 ウェルカム・メッセージ

通知メッセージと同様に、ウェルカム・メッセージはシステム論理名 SYS\$WELCOME によって制御されます。SYS\$WELCOME を定義しない場合は、全ユーザに対して標準のウェルカム・メッセージが表示されます。このウェルカム・メッセージは、オペレーティング・システムとバージョン番号を表示し、SYS\$NODE が定義されている場合はノードも公にします。

SYS\$WELCOME に別のメッセージを定義するには、メッセージを含むテキスト・ファイルを作成します。このファイルの内容を表示するには、SYSTARTUP\_VMS.COM の中で次の行を使用します。

```
$ DEFINE/SYSTEM SYS$WELCOME "@SYS$MANAGER:WELCOME.TXT"
```

ウェルカム・メッセージを無効にするには、SYS\$MANAGER:SYSTARTUP\_VMS.COM に次の DCL コマンドを入れます。このコマンドは、標準のウェルカム・メッセージの代わりに空行を出力します。

```
$ DEFINE/SYSTEM SYS$WELCOME " "
```

各ユーザへのメッセージを個別に無効にするには、各 UAF レコードで AUTHORIZE 修飾子の /FLAGS=DISWELCOME を使用します。

### 7.5.1.3 最終ログイン・メッセージ

デフォルトでは、最終ログインと、失敗したログイン試行の回数に関する 3 つのメッセージがシステムにより表示されます (3.4.3 項「情報メッセージの解釈」を参

照)。これら 3 つのメッセージの表示は、個別に無効にすることができます。ユーザごとに AUTHORIZE 修飾子の /FLAGS=DISREPORT を入力します。

#### 7.5.1.4 新着メールの通知

デフォルトでは、ログイン時に新着メール・メッセージの数がシステムによってユーザに通知されます。AUTHORIZE 修飾子の /FLAGS=DISNEWMAIL を指定することで、ユーザがこの通知を受け取らないようにできます。

新着メールの通知は、セキュリティ上の問題ではなく、主にユーザの便宜を考慮したものです。制限付きアカウントのユーザがメール・ユーティリティ (MAIL) を起動できない場合は、メール・アクセスを禁止すると同時に新着メールのメッセージを無効にしたい場合があるためです。次の AUTHORIZE 修飾子は、両方を実現します。

/FLAGS=(DISMAIL,DISNEWMAIL)

### 7.5.2 切断されたプロセスの限定

仮想ターミナルを使用すると、ユーザは一度に複数の切断されたプロセスを維持することができます。仮想ターミナルは、安全ターミナル・サーバの機能でも必要です (7.5.4 項を参照)。仮想ターミナルの使用を制限したい場合があります。たとえば、非ページング・プールの量が問題である場合、この機能をシステム全体では有効にしたくない場合があります。

仮想ターミナルは、次のようにターミナル、ユーザ、またはシステムのレベルで無効にできます。

- 特定のターミナルを仮想ターミナルとしては使用できないようにするには、DCL の SET TERMINAL/PERMANENT/NODISCONNECT コマンドを使用します。
- 特定のユーザが切断されたプロセスにアタッチできないようにするには、該当するユーザに対して AUTHORIZE 修飾子の /FLAGS=DISRECONNECT を設定します。複数のユーザにより使用されるアプリケーション・アカウントは、ユーザがお互いのプロセスに接続するのを防止するために、DISRECONNECT フラグの対象候補として妥当です。
- システム単位で仮想ターミナルを無効にするには、システム・パラメータ TTY\_DEFCHAR2 から DISCONNECT 属性を削除します。

また、システム・パラメータ TTY\_TIMEOUT を使用すると、再接続に許可される時間を、デフォルトの 15 分よりも短く設定することもできます。このタイムアウト値よりも長く切断されたままの状態であるプロセスは、自動的にシステムによってログアウトされます。接続時間を制限すると、メッセージを受け取るユーザの数が最小になる傾向がありますが、接続機能の利便性にも影響を与えます。



仮想ターミナルの設定と、仮想ターミナルへの再接続の詳細については、  
『OpenVMS システム管理者マニュアル』を参照してください。

### 7.5.3 自動ログインの実現

特定のターミナルにアカウントを割り当てて、自動ログイン機能を有効にすることができます (7.2.6 項を参照)。この機能により、ユーザはユーザ名を指定せずにログインができるようになります。オペレーティング・システムは、ユーザ名をターミナル(またはターミナル・サーバのポート)に対応させ、その割り当て情報を SYS\$SYSTEM:SYSALF.DAT ファイル (自動ログイン・ファイル、別名 **ALF** ファイル) に保存します。このファイルを維持管理するには、次のシステム管理ユーティリティ (SYSMAN) コマンドを使用します。

作業	コマンド	例
ターミナル/ユーザ名の対応付けの追加	ALF ADD	ALF ADD TTA5 RENOLDS
ターミナル・サーバ/ユーザ名の対応付けの追加	ALF ADD/PORT	"M34C3/LC-1-2" RENOLDS
ALF ファイルのレコードの表示	ALF SHOW	ALF SHOW TTA5 ALF SHOW /USERNAME=PONTRE
ターミナル/ユーザ名の対応付けの削除	ALF REMOVE	ALF REMOVE TTA3 ALF REMOVE /USERNAME=DOUGLAS

ALF ファイルは、自動ログインが有効になっているターミナルごとに 1 つのレコードで構成されています。各レコードは、ターミナルのデバイス名またはターミナル・サーバ・ポート名と、それ続くアカウントのユーザ名の、2 つのフィールドで構成されています。デバイス名はファイル内で一意である必要があります。ただし、同じユーザ名が任意の数のレコード中に存在することができます。つまり、1 つのアカウントが自動的にログインできるターミナルの数に制限はありません。

ALF ファイルは、ページが不要な索引編成ファイルですが、変更があった場合にはバックアップが必要です。

### 7.5.4 安全ターミナル・サーバの使用

3.8 節「パスワードの保護に関するガイドライン」では、電源が入ったままのターミナルにログインした無警戒のユーザからパスワードを盗むことを目的としたプログラムの種類として、パスワード・グラバを説明しています。オペレーティング・システムは、ターミナルでのログイン開始前に現在実行中のプロセスをすべて停止する、安全ターミナル・サーバを提供しています。

ターミナルごとに安全ターミナル・サーバを個別に起動するには、次の DCL コマンドを使用します。

SET TERMINAL/PERMANENT/SECURE/DISCONNECT term-id

続いて、ユーザがログインを開始するには、Break キーの後に Return キーを押す必要があります。ログインは通常どおり進行します。

安全ターミナル・サーバをすべてのターミナルに適用する場合は、サイト固有のスタートアップ・コマンド・プロシージャに SET TERMINAL コマンドを入れることで、サイト全体でログイン・プロシージャに一貫性を持たせることができます。ただし、通信回線としてターミナルを使用する場合がある一部のアプリケーションは、独自の目的のために Break キーを使用する必要があるため、安全ターミナル・サーバと組み合わせて使用できなくなります。

安全ターミナル・サーバ機能は、通信速度の自動判別処理とも組み合わせることができません。ただし、通信速度の自動判別が必要なのはモデム・ターミナル(交換ターミナルおよびダイアルアップ回線使用ターミナル)のみであるため、このようなターミナルでのモデム処理は、安全ターミナル・サーバと同等の機能を果たします。安全な運用のためには、次のようにターミナル属性を設定します。

- (直結の) ローカル・ターミナルには、次の SET TERMINAL 修飾子を使用します。

/NOMODEM/SECURE/DISCONNECT/NOAUTOBAUD/PERMANENT

- 交換ターミナル(データ交換およびダイアルアップ)には、次の SET TERMINAL 修飾子を使用します。

/MODEM/AUTOBAUD/NOSECURE/DISCONNECT/PERMANENT

電話回線またはこれと同等の回線を経由してターミナル・ポートにアクセスできる場合、経路(直結モデム、データ交換、ターミナル・サーバ、または公衆データ・ネットワーク)に関係なく、/DIALUP 修飾子を指定します。

パスワード・グラバから保護するには、常に /DISCONNECT 修飾子を指定します。切断されたジョブでシステムが飽和するのを防ぐには、切断されたプロセスが削除されるまでの期間を指定するシステム・パラメータ TTY\_TIMEOUT を、低いタイムアウト値に設定します。

安全ターミナル・サーバを個別のターミナルに適用することにした場合は、公共の場所および遠隔の安全でない場所にある直結したターミナルを含めます。ローカル・ログインやダイアルアップ・ログインに使用されることのないターミナルは、このセキュリティ問題の影響を受けません。ログイン時に厳しく監視されるターミナルも、この対策が必要でない場合があります。

### 7.5.5 侵入者の検出

場合によっては、期限切れのパスワードの入力や、入力ミスのために、正しくログインできないことがあります。しかし、ログインの失敗がすべて無害であるとは限

りません。権限のない人物が期限切れのアカウントや未知のユーザ名でログインしようとしたり、有効なアカウントのパスワードを推測しようとしているために、ログイン失敗が起きる場合があります。

オペレーティング・システムは、ログインの失敗に対して敏感に反応します。1回の失敗の後、オペレーティング・システムは、ログインが行われているターミナル、ターミナル・サーバ接続、またはネットワーク接続の監視を開始します。まずオペレーティング・システムは、失敗したログインを侵入データベースに記録します。失敗が継続すると、オペレーティング・システムは失敗を記録するだけでなく、抑制対策を講じます。ログインを試みる人物はより詳細に監視され、一定期間内でのログイン再試行が一定の回数に制限されます。ログインを試みている人物が再試行や時間の制限を超えると、その人物は、たとえ有効なユーザ名とパスワードを使用しても、ある期間ログインできなくなります。しばらくすると制限が緩和され、ログインが再び許可されます。

### 7.5.6 侵入データベースについて

DCL の SHOW INTRUSION コマンドを使用すると、侵入データベースの内容が表示されます。例 7-5 に表示例を示します。データベースは、ログインの失敗に関する、次のタイプの情報を取得します。

フィールド	説明
Intrusion class	失敗の大まかな発生源 <ul style="list-style-type: none"><li>Network : 有効なユーザ名を使用した、遠隔ノードからのログインの失敗</li><li>Terminal : 1 つのターミナルからのログインの失敗</li><li>Term_User : 有効なユーザ名を使用した、1 つのターミナルからのログインの失敗</li><li>Username : 独立プロセスを作成しようとしたことによる失敗</li></ul>
Type	ログイン失敗の重大度 <ul style="list-style-type: none"><li>Suspect</li><li>Intruder</li></ul> 回数の限度 (LGI_BRK_LIM) と監視期間 (LGI_BRK_TMO) のシステム・パラメータによって、疑いのある行動が侵入行為と見なされる基準が定義されます。
Count	特定の発生源に関連するログイン失敗の回数

フィールド	説明
Expiration	疑いのある者のレコードが削除される日時，または侵入者に再度ログインのチャンスが認められた日時。侵入者のレコードが期限切れになると，侵入者は容疑者になり，失敗回数は LGI_BRK_LIM にリセットされます。有効期限は，古い有効期限に LGI_BRK_TMO を加えた値にリセットされます。
Source	ログイン失敗の発生源 <ul style="list-style-type: none"> <li>• Network クラスの場合はノードとユーザ名</li> <li>• Terminal クラスの場合はターミナル</li> <li>• Term_User クラスの場合はターミナルとユーザ名</li> <li>• Username クラスの場合はユーザ名</li> </ul>

システムは侵入者を検出すると，必ずセキュリティ・オペレータ・ターミナルまたはログ・ファイルに監査メッセージを送信して，セキュリティ管理者に警告します。DCL の SHOW INTRUSION コマンドを使用して，侵入の発生源とタイプを表示できます。例として例 7-5 に，ネットワーク経由でログインする，MAPLE という名前のユーザに関する問題を示します。このユーザはログインを 8 回試行しています。ユーザは監視期間内でのログインに失敗したため，オペレーティング・システムは OMNI::BOSTON.BIRCH::MAPLE からのすべてのログインを一時停止しました。表 7-6 では，システムがログインの一時停止をどのように決定するかをより詳細に説明しています。

多くの容疑者が表示されている点に注意してください。ユーザは，パスワードを忘れたり，パスワードを誤って入力することがあるためです。データベースからエントリを削除するには，DCL の DELETE/INTRUSION\_RECORD コマンドを使用します。

#### 例 7-5: 侵入データベースの表示

```
$ SHOW INTRUSION
```

Intrusion	Type	Count	Expiration	Source
NETWORK	SUSPECT	1	2-Jan-2002 13:20:30.89	PCD025::

Intrusion	Type	Count	Expiration	Source
NETWORK	SUSPECT	5	2-Jan-2002 13:36:39.42	DENIM::SYSTEM
NETWORK	SUSPECT	2	2-Jan-2002 13:25:17.30	N1KDO::SYSTEM

Intrusion	Type	Count	Expiration	Source
NETWORK	SUSPECT	2	2-Jan-2002 13:07:57.95	OMNI::LOWELL.ASH::TESTER
NETWORK	INTRUDER	8	2-Jan-2002 11:06:50.51	OMNI::BOSTON.BIRCH::MAPLE

Intrusion	Type	Count	Expiration	Source
NETWORK	SUSPECT	2	2-Jan-2002 13:20:10.09	JETTE::TIPH
NETWORK	SUSPECT	1	2-Jan-2002 13:21:40.75	FTSR::TFREDERICK

### 7.5.6.1 侵入検出の仕組み

ログイン失敗が1回発生すると、ユーザは容疑者になり、ある一定期間それ以降の失敗が監視されます。オペレーティング・システムは、容疑者によるログイン失敗を一定期間に渡って一定回数だけ許容し、それを超えるとログイン失敗の発生源を侵入者であると宣言します。つまり、容疑者は監視期間中に許されるログイン試行回数を超えると侵入者になります。

システム・パラメータ LGI\_BRK\_LIM によって設定される試行回数は、人がログインを試行できる回数を定義します。標準の制限では5回です。この試行回数パラメータは、システム・パラメータ LGI\_BRK\_TMO によって制御される時間の係数と連携します。ログインに失敗するたびに、容疑者の監視期間は LGI\_BRK\_TMO の値だけ増加します。そのため、失敗のたびに、容疑者が監視される期間が長くなります。

表 7-6 に、ユーザ George が5回ログインに失敗し、回避措置が取られる状況を示します。失敗するたびに、監視期間は5分延長されます。5回目の失敗で、オペレーティング・システムは George を侵入者と見なし、このユーザのログインを拒否します。この例では、パラメータ LGI\_BRK\_LIM と LGI\_BRK\_TMO が両方とも5に設定されているものと想定しています。

表 7-6: 侵入の例

ログイン失敗の時刻	失敗の回数	監視期間の延長
6:00	0	George がログインに失敗し、システムは George のターミナルからのログインの監視を開始します。システムはこれから5分間監視します。
6:00:30	1	30秒後、監視期間が4.5分残っている時点で、George が再び失敗します。監視期間は5分延長されます。したがって、システムはこれから9.5分間 George のログイン失敗を監視します。
6:01	2	30秒後、監視期間が9分残っている時点で、システムは監視期間を5分延長します。
6:02	3	1分後、George の監視期間は13分で、システムは監視期間を5分延長します。
6:02:30	4	30秒後、George の監視期間は17.5分で、システムは監視期間を5分延長します。したがって、システムはこれから22.5分間 George のログイン失敗を監視します。
6:04:30	5	2分後、George は6回目の試行を行います。監視期間で許可される時間であっても、George はチャンスを使い果たします。George は侵入者と見なされ、システムにアクセスできなくなります。

### 7.5.6.2 除外期間の設定

システム設定に応じて、侵入者を、一時的または永久に除外することができます。

- 一時的な除外の制御は、LGI\_HID\_TIM と、1 ~ 1.5 の範囲の乱数の積によって行います。一時的な除外期間が終わると、サブジェクトは容疑者に再度分類されます。容疑者の監視期間は、LGI\_BRK\_TMO の値により設定されます。新しい監視期間中は、LGI\_BRK\_LIM の値が許容失敗回数として設定され、サブジェクトが侵入者に再度分類される前に、もう一度ログインのチャンスが認められます。
- LGI\_BRK\_DISUSER が設定されている場合は、永久除外になります。これは、この設定により、オペレーティング・システムが侵入を検出すると、ユーザ登録レコードの DISUSER フラグが有効になるためです。

LGI\_BRK\_DISUSER パラメータを有効にすると、セキュリティ管理者が手動で操作するまでそのユーザ名は無効になるため、重大な結果を招くことがあります。LGI\_BRK\_DISUSER を有効にすると、悪意のあるユーザが、セキュリティ管理者のアカウントを含むすべての既知のアカウントを、短時間ですべて使用不能にすることができます。復旧するには、SYSTEM アカウントが常にログインを許可されているシステム・コンソールにログインする必要があります。

### 7.5.6.3 ログイン試行を制御するシステム・パラメータ

ログインと侵入検出を制御するシステム・パラメータを表 7-7 に示します。

表 7-7: ログイン試行を制御するためのパラメータ

制御対象	設定するパラメータ	説明
ログイン期間	LGI_PWD_TMO	時間内に次の操作が可能です。 <ul style="list-style-type: none"><li>• (使用されている場合) 正しいシステム・パスワードの入力。</li><li>• 個人用アカウントのパスワードの入力。</li><li>• SET PASSWORD コマンドを使用する場合の、旧パスワードの入力、新しいパスワードの入力、およびパスワードの検証。</li></ul>
電話回線またはネットワーク接続経由で、あるユーザがログインを試行できる回数	LGI_RETRY_LIM	再試行時間 (LGI_RETRY_TMO) の範囲内で、電話接続またはネットワーク・リンクを失わずに、ログイン・シーケンスを再試行することを許可します。監視期間中、侵入の上限 (LGI_BRK_LIM) を超えない限り、ユーザは再接続とログインの再試行を行うことができます。

表 7-7: ログイン試行を制御するためのパラメータ (続き)

制御対象	設定するパラメータ	説明
電話回線またはネットワーク接続経由でのログイン試行の間隔	LGI_RETRY_TMO	ログイン失敗後、次のログイン試行が許可される間隔の秒数を指定します。ログイン失敗後、LGI_RETRY_TMO の秒数の間ユーザの応答がない場合、LOGINOUT がセッションを切断します。
ログイン再試行可能回数	LGI_BRK_LIM	回避措置を呼び出す結果となる、監視期間中のログイン失敗の回数を指定します。失敗回数は、各ユーザ名、ターミナル、およびノードの個別のログイン試行ごとに適用されます。
失敗の監視期間の長さ	LGI_BRK_TMO	ログイン失敗のたびに容疑者の有効期限に加算される時間を示します。有効期限が切れた後、以前の失敗は破棄され、サブジェクトはクリーンな状態になります。
侵入データベースのソース名における、ユーザ名とターミナル名の関連付け	LGI_BRK_TERM	ターミナル・クラス・ログインの失敗を、ターミナル、ユーザ (デフォルト)、または全ターミナルのユーザのいずれによりカウントするかを制御します。TT_ACCPORNAM フィールドの内容に基づいて、LAT が発信元ポートまで遡って追跡されます。
ログイン拒否の期間	LGI_HID_TIM	ログイン拒否の期間を指定します。このパラメータの値に (1 ~ 1.5 の) 乱数を乗じた数によって、失敗回数が LGI_BRK_LIM を超えた場合の回避措置の実際の長さが決まります。
侵入者のアカウント	LGI_BRK_DISUSER	ユーザの登録レコードにある DISUSER フラグを有効にして、そのアカウントを永久にロック・アウトします。

## 7.5.7 セキュリティ・サーバ・プロセス

通常のオペレーティング・システムの起動処理の一部として作成されるセキュリティ・サーバ・プロセスは、次のタスクを実行します。

- システムの侵入データベースの作成と管理
- ネットワーク代理データベース・ファイル (NET\$PROXY.DAT) の維持管理

システムは侵入データベースを使用して、ログイン試行の失敗を追跡します。この情報はプロセス・ログイン中に走査され、システムが抑制対策を講じて、侵入者の疑いがあるユーザによるシステムへのアクセスを禁止すべきかどうかを判定されます。例 7-5 に示すように、DCL の SHOW INTRUSION コマンドを発行すること

で、このデータベースの内容を表示できます。DCL の DELETE/INTRUSION コマンドを発行すると、データベースから情報を削除できます。

特定の遠隔ユーザがパスワードを使用せずにローカル・アカウントにアクセスできるかどうかを判定するため、ネットワーク接続処理中に、ネットワーク代理データベース・ファイル (NET\$PROXY.DAT) が使用されます。このデータベースの情報の管理には、登録ユーティリティを使用します。



## システムのデータと資源へのアクセスの制御

この章では、ユーザ・グループを設計する方法と、作業の実行に必要な識別情報 (UIC, 識別子, 特権) をユーザに与える方法について説明します。システムのデータと資源を適切に保護すると同時に、ユーザが効率的に作業できるよう、適切な保護コードと ACL をオブジェクトに割り当てる方法も示します。この章では、読者が第 4 章と第 5 章の内容を習得していることを前提としています。

### 8.1 ユーザ・グループの設計

ユーザ・グループを設計する際には、セキュリティ管理者が作成するグループは、データと資源の保護に影響を与え、GROUP, GRPNAM, および GRPPRV 特権を受けるユーザに影響する点に留意してください。ユーザの職務に応じてグループ分けする方法が考えられます。会計、エンジニアリング、マーケティング、人事など共通の職務を行うユーザのグループを調べます。

また、組織における将来の計画を見越し、これらの考えを戦略に組み込みます。グループの設計の微調整はいつでもできますが、ユーザの職務に応じた合理的なグループ分けを把握することが最も重要です。

UIC グループへのユーザの配置を決定するには、次の 2 つのガイドラインに従います。

- 共有: 普段からデータおよびプロセスの制御をお互いに共有するユーザは、同じグループに配置します。
- 保護: お互いのデータへのアクセス、またはお互いのプロセスの制御が禁止されているユーザ同士は、別々のグループに割り当てます。

ただし、UIC グループの設計には制約があります。セキュリティ管理者が所有するファイルへのアクセス権を、UIC グループの少数のメンバにのみ与えることも、ワールド・アクセス権を付与することなく、セキュリティ管理者のファイルへのアクセス権を複数の UIC グループのメンバに付与することもできます。これらの制約については、8.1.2 項で説明しています。

#### 8.1.1 UIC グループの設計の例

架空の Rainbow Paint Company は、経営執行、会計、マーケティング、発送、管理の、5 つの部署がある流通企業です。表 8-1 に、さまざまな部署における、

コンピュータ資源を必要とする従業員を示します。この表には、従業員が担当する職務の一覧も示します。

表 8-1: 部署と職務による従業員のグループ分け

部署	従業員名	職務
経営執行	Samuel Gibson	社長
	Olivia Westwood	コンピュータ運用の責任者
会計	Carlo Ruiz	給与計算
	Rich Smith	経理
	Rod Jacobs	事務員
	Ruth Ross	事務員
マーケティング	Jason Chang	市場予測
	Alana Mack	売り上げ報告
発送	Scott Giles	在庫管理
管理	Jane Simon	通信管理/給与小切手印刷

この会社が複数の部署に編成されていることは、同じ部署の人員は、多くの同じ職務を遂行することを意味します。たとえば、この会社で経理作業を行う全従業員を、会計課にグループ分けする利点は、従業員の相互連絡と、共有しなければならないデータへのアクセスが簡単に行える点です。

Rainbow Paint のコンピュータ資源のシステム管理者である Olivia Westwood は、既存の組織構造に基づいて UIC グループを設定します。たとえば、会計課の従業員 (Ruiz, Smith, Jacobs, Ross) を、UIC グループ ACCOUNTING のメンバにすることができます。このように UIC グループを設定することで、ユーザ Ruiz は、ユーザ Smith などのデータに簡単にアクセスできるようになり、他のメンバについても同様です。

部署を効果的に組織化することにより、選ばれた従業員だけが会社内のすべてのデータと従業員にアクセスできるようになります。たとえば、会計課の職務の1つに、給与計算があります。給与計算の情報は機密情報であるため、発送とマーケティングの従業員は、その情報にアクセスすることは禁じられています。

Rainbow Paint のコンピュータ資源のシステム管理者として、Westwood は、ACCOUNTING, EXECUTIVE, MARKETING, SHIPPING, および ADMINISTRATION の UIC グループを設定します。これらのグループは、会社の各々の部署に対応します。同じ UIC グループのメンバには、次の例のように、ファイルへの共通のアクセス権を付与することができます。

```
$ SET SECURITY/PROTECTION=G:RWE GROUP_STATS.DAT
```

このコマンドを使用して、ファイル GROUP\_STATS.DAT の所有者は、UIC グループの各メンバに、ファイルに対する読み込みアクセス権、書き込みアクセス権、および実行アクセス権を許可します。

### 8.1.2 UIC グループの設計に関する制約

場合によって UIC ベースの保護が、オブジェクト保護のニーズに対する最適のソリューションではないことがあります。複数の UIC グループのユーザが、システム上の共通ファイルなどの資源へのアクセス権を必要とする場合、UIC ベースの保護で利用できる方法は、そのオブジェクトに対するワールド・アクセス権を付与する方法か(全ユーザがそのオブジェクトにアクセスできる)、各ユーザの特権を拡張する方法のみです。どちらの選択肢も望ましくありません。

また、UIC グループのユーザに複数のタイプのファイル・アクセス権を許可したり、同じグループ内の一部のユーザに対して、オブジェクトへのアクセス権を付与しない必要がある場合もあります。このような場合も、UIC ベースの保護は、これらのニーズを満たす適切なソリューションではありません。

以降の節で説明するアクセス制御リスト (ACL) は、システム上のファイルなどのオブジェクトを保護する別の手段を提供します。

4.5 節「保護コードによるアクセスの制御」で説明しているように、サイトのセキュリティ管理者は、UIC のカテゴリの詳細を十分認識することが非常に重要になります。特定の UIC グループにユーザを入れることで、そのユーザにシステム特権を付与できます。また、システム特権を持つユーザは、システム上のすべての保護オブジェクトに対する制御アクセス権を持っています。SYSPRV 特権は、デフォルトでは 10 以下のすべての UIC グループに付与されますが、システム UIC カテゴリの実際の範囲は、MAXSYSGROUP システム・パラメータの値によって決まります。システム・ファイルを所有するグループに、GRPPRV 特権を持つユーザを入れると、セキュリティ上問題になる場合があります。

## 8.2 ACL での個別ユーザの指定

データと資源の保護の問題を解決するために、UIC グループを再構築するのではなく、アクセス制御リスト (ACL) を使用して目的を達成できる場合があります。ACL の詳細については、4.4 節「ACL によるアクセスの制御」で説明しています。UIC は ACE において識別子として機能するため、簡単に ACL を作成して、さまざまな UIC グループの特定のユーザに、オブジェクトへのアクセスを許可することができます。

たとえば、Rainbow Paint Company の特定のユーザに、PAYROLL.DAT ファイルへのアクセスを許可する、次のような ACL を作成する場合を考えます。

```
(IDENTIFIER=OWESTWOOD, ACCESS=READ+WRITE+EXECUTE+DELETE)
(IDENTIFIER=CRUIZ, ACCESS=READ+WRITE+EXECUTE+DELETE)
```

```
(IDENTIFIER=RSMITH, ACCESS=READ+WRITE+EXECUTE+DELETE)
(IDENTIFIER=JSIMON, ACCESS=READ)
(IDENTIFIER=SGIBSON, ACCESS=READ)
```

## 8.3 権限の共有の定義

多数のユーザが同じアクセス権を必要とする場合があります。しかし、UIC 識別子のみで構成される ACL は、長くなりすぎる場合があります。ACL を短縮するために、システム定義の環境識別子を含めたり、汎用識別子を作成することができます (表 4-1 表 4-1 を参照)。

汎用識別子を作成する際には、システムで必要な識別子の名前を考え、その識別子の保持者のセットを作成します。続いて、識別子をライト・データベースに追加し、識別子を該当するユーザに割り当てます。

たとえば、Rainbow Paint Company で PAYROLL 識別子をライト・データベースに追加することにしたとします。その識別子の保持者は、PAYROLL.DAT に対する読み込みアクセス権、書き込みアクセス権、実行アクセス権、および削除アクセス権を必要とする全ユーザで、OWESTWOOD, CRUIZ, および RSMITH です。

識別子とその保持者を定義したら、セキュリティ管理者は次の ACL を使用して、同じタイプのアクセス権を PAYROLL.DAT に指定します。

```
(IDENTIFIER=PAYROLL, ACCESS=READ+WRITE+EXECUTE+DELETE)
(IDENTIFIER=JSIMON, ACCESS=READ)
(IDENTIFIER=SGIBSON, ACCESS=READ)
```

## 8.4 ユーザごとの識別子の条件指定

ACL と識別子の設計の最終ステップは、それぞれの識別子がいつどのように使用されるかを考慮することです。ユーザは多くの場合、データベースの更新やシステム操作の実行など、さまざまな目的のために識別子を保持します。このため、識別子の使用を限定したい場合があります。

識別子の使用を限定する方法はいくつかあります。1 つは、環境識別子を使用する方法で、もう 1 つは 8.6.7 項で説明しているように、識別子に特別な属性を追加する方法です。

環境識別子は、ユーザがシステムに最初に入ったときの方法に応じた、さまざまなタイプのユーザを規定します。ローカル、ダイアルアップ、遠隔、会話型、ネットワーク、およびバッチのいずれかとなるこれらの識別子は、ユーザがシステムを使用する形態に応じて、大規模なユーザ・グループを定義することができます。一般的にこれらのタイプの識別子は、ほかの識別子と組み合わせて使用します。

たとえば次の ACE は、ローカル・ターミナルからログインした場合にのみ、オブジェクトに対する読み込みアクセス権、書き込みアクセス権、実行アクセス権、および削除アクセス権をユーザ Martin に許可します。

(IDENTIFIER=MARTIN+LOCAL, ACCESS=READ+WRITE+EXECUTE+DELETE)

ACL で環境識別子を使用して、特定のログインのクラス全体に対して、アクセスを拒否することができます。たとえば、次の ACE はすべてのダイアルアップ・ユーザに対して、アクセスを拒否します。

(IDENTIFIER=DIALUP, ACCESS=NONE)

DECwindows 環境のユーザにこれらの環境識別子を割り当てる際には、DECwindows プロセスは事実上任意のタイプのプロセスになりうる点に留意してください。たとえば、ユーザはバッチ・ジョブの中で DECwindows Mail を実行できます。プロセスが DECwindows ワークステーションを介してユーザと会話型の通信を行っている場合であっても、そのプロセスはバッチ・ジョブに分類されます。

## 8.5 ACL の設計

ACL を設計する際には、次のような考慮事項があります。

- 汎用識別子を用いた短い ACL を使用すると、いくつかのメリットがあります。オペレーティング・システムは、ACL が短ければ、処理が高速になります。また、従業員の異動があっても職務が同じままであれば、システム全体で各 ACL を変更しなくても済みます。その代わりに、識別子の保持者を変更します。従業員がプロジェクトを離れる場合は、RIGHTSLIST.DAT にあるその従業員のレコードを編集して、以後その従業員が識別子を保持しないようにします。また従業員が退職する場合は、ユーザ登録ファイル (UAF) から従業員のレコードをまるごと削除できます。同じ仕事を担当する新しい従業員が雇用された場合は、その識別子を保持する権限を新しいユーザに付与します。これで、新しいユーザは、前のユーザと同じ ACL ベースのアクセス権を持つことになります。
- 設計の全般では、システム上のファイルなどのオブジェクトのタイプと、各オブジェクトの保護の要件を考慮する必要があります。グループと識別子を正しく指定すれば、ACL の設計と標準的な保護の定義が簡単にできます。ユーザの共通のアクセス要件を明らかにすることにかけた時間だけ、識別子と ACL の設計が簡単になります。また、自分のファイルに ACL を適用するユーザの作業も簡単になります。
- ACL を多用し過ぎないようにしてください。ACL は、ファイルがオープンされていると、システムの動的なページング・メモリを消費します。また、処理時間も余分に必要になります。ACL の適用が最適であるのは、保護が実際に必要な場面です。ACL が長すぎる (たとえば 200 エントリを超える) 場合は、ユーザを個別のカテゴリにグループ分けし、汎用識別子を作成することを検討します。
- 同時に、作成する識別子の数は適度に制限します。特に、1 人のユーザにあまりに多くの識別子を付与しないでください。1 人のユーザに 10 ~ 20 個以上の識別子があると、ACL の処理に過剰な時間が費やされます。あまりに多くの識別子を保持しているユーザが見つかった場合は、グループの構造を再検討する

とよいでしょう。または、そのユーザが例外的なケースである場合は、その個人を必要な ACL に直接入れることを検討します。

識別子の定義の詳細については、8.6 節と、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の AUTHORIZE の説明を参照してください。ACL の作成と保守の詳細については、第 4 章を参照してください。作業が多い場合は、アクセス制御リスト・エディタ (ACL エディタ) を使用するとよいでしょう。ACL エディタについては、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』で説明しています。

## 8.6 ライト・データベースへの登録

システムに必要な識別子の名前を考え、識別子の保持者のセットを作成したら、AUTHORIZE を使用して識別子をライト・データベースに追加し、対象ユーザに識別子を割り当てます。これらの関連付けは、ライト・データベース (RIGHTSLIST.DAT) に保持されます。ライト・データベースは、ユーザと識別子の追加/削除を行うことで保守します。

ライト・データベースは、初めにシステムのインストール時に作成され、[SYSEXEC] ディレクトリにあります。作成時にライト・データベースには、環境識別子の名前が含まれています。登録ファイルにユーザを追加すると、登録したユーザごとに 1 つの識別子が追加されます。UIC 識別子と呼ばれるこの識別子は、ユーザの UIC およびユーザ名と関連付けられています。

ライト・データベースにも、各 UIC グループ名に相当する識別子があります。新しい UIC グループの最初のメンバとして新規ユーザを追加し、そのユーザにアカウント・グループ名を指定すると、次の例のように、アカウント・グループ名に対応する識別子がライト・データベースに追加されます。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD ROB/PASSWORD=SP0152/UIC=[014,006] -
_UAF> /DIRECTORY=WORK:[ROB]/ACCOUNT=MGMT

UAF-I-ADDMSG, user record successfully added UAF-I-RDBADDMSGU,
identifier ROB value: [000014,000006]      added to RIGHTSLIST.DAT
UAF-I-RDBADDMSGU, identifier MGMT value: [000014,177777]
added to RIGHTSLIST.DAT
```

ROB のアカウントを追加する際にアカウント名 MGMT を指定していますが、その名前の UIC グループが存在しないため、ライト・データベースに MGMT 識別子が追加されます。

各サイトは、実際の使用状況と要件に従って、それぞれのライト・データベースを合わせていきます。

AUTHORIZE を使用してシステム・ユーザ登録ファイル (SYSUAF.DAT) のユーザ名の追加、削除または変更を行うと、ライト・リストが SYSUAF.DAT に対応するように、AUTHORIZE によって対応する変更が RIGHTSList.DAT に加えられます。

ライト・データベースの作成と保守は自動的に行われるため、AUTHORIZE の CREATE/RIGHTS コマンドは、ほとんど使用する必要がありません。ただし、ライト・データベースが破損したり削除された場合は、このコマンドを使用して新しいライト・データベースを作成できます。詳細については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照してください。

### 8.6.1 データベースの表示

定期的にライト・データベースを表示して、ライト・データベースが正確で、情報が最新であることを確認する必要があります。このためには、2 つの AUTHORIZE の SHOW/IDENTIFIER コマンドと SHOW/RIGHTS コマンドを使用します。ある識別子のすべての保持者を表示するには、次の例のように SHOW/IDENTIFIER コマンドを使用します。

```
UAF> SHOW/IDENTIFIER/FULL NETWORK
```

システム上の全識別子の全保持者を表示するには、次のようにアスタリスク (\*) ワイルドカードを使用します。

```
UAF> SHOW/IDENTIFIER/FULL *
```

特定のユーザが保持する識別子を表示するには、次のように SHOW/RIGHTS コマンドを使用します。

```
UAF> SHOW/RIGHTS/USER=ROBIN
```

全ユーザが保持する全識別子を表示するには、次のようにアスタリスクのワイルドカードを使用します。

```
UAF> SHOW/RIGHTS/USER=*
```

```
UAF> SHOW/RIGHTS/USER=[*,*]
```

最初のコマンドにより、ユーザがアルファベット順で表示されます。2 番目のコマンドでは、UIC 順にユーザが表示されます。

### 8.6.2 識別子の追加

ライト・リストに識別子を追加するには、次のように AUTHORIZE の ADD/IDENTIFIER コマンドを使用します。

```
UAF> ADD/IDENTIFIER PAYROLL
identifier PAYROLL value %X80080011 added to RIGHTSList.DAT
```

8.6.7 項 で説明している属性を使用してユーザに識別子を付与するには、識別子を追加する際にその属性を指定する必要があります。たとえば、識別子の追加または変更をユーザに許可するには、Dynamic 属性を指定します。

```
UAF> ADD/IDENTIFIER PROJECT_TEAM1 /ATTRIBUTES=DYNAMIC
```

### 8.6.3 ライト・データベースの復元

誤ってライト・リストを削除してしまい、バックアップ・コピーからも復元できない場合は、次のように CREATE/RIGHTS コマンドを入力し、その後に ADD/IDENTIFIER コマンドを入力して、RIGHTSLIST.DAT を再度作成します。

```
UAF> CREATE/RIGHTS
{message}
UAF> ADD/IDENTIFIER/USER=* or ADD/IDENTIFIER/USER=[*,*]
{messages}
```

ADD/IDENTIFIER コマンドは、ライト・リストに、SYSUAF.DAT の各ユーザ名に対応する UIC 識別子を作成します。この作業を完了するには、ADD/IDENTIFIER コマンドを使用して、失われたすべての汎用識別子を追加します。続いて、8.6.4 項で説明している方法で、GRANT/IDENTIFIER コマンドを使用して識別子の保持者を再定義します。

### 8.6.4 ユーザへの識別子の割り当て

識別子を追加した後、次の例のように AUTHORIZE の GRANT/IDENTIFIER コマンドを使用して、既存の識別子の保持者としてユーザを関連付けます。

```
UAF> GRANT/IDENTIFIER PAYROLL MARTIN
UAF-I-GRANTMSG, identifier PAYROLL granted to MARTIN
UAF> GRANT/IDENTIFIER PAYROLL IPPOLITO
UAF-I-GRANTMSG, identifier PAYROLL granted to IPPOLITO
```

ユーザ Martin に、PAYROLL 識別子に加えて EXECUTIVE 識別子を付与するには、GRANT/IDENTIFIER コマンドを再度使用する必要があります。GRANT/IDENTIFIER コマンドでは、一度に 1 つのみ保持者の関連付けを行うことができます。

上記のすべての例で、AUTHORIZE を使用して、ユーザ(具体的には Martin と Ippolito)に対応する UIC 識別子に PAYROLL 識別子を関連付けています。識別子は両方ともライト・データベースに存在する必要があります。

### 8.6.5 保持者レコードの削除

ユーザが退職した場合は、そのユーザの UAF レコードを削除します。そのユーザが代理アカウントにアクセスできるすべてのサイトの管理者に通知して、遠隔ノードの NETPROXY.DAT ファイルにある代理アクセス情報を削除してもらいます。AUTHORIZE を実行してユーザの UAF レコードを削除すると、AUTHORIZE によって、ライト・データベースにある識別子の保持者としてのユーザの関連付けも削除されます。ただし、退職したユーザが特定の識別子の唯一の保持者である場合は、今後の混乱を避けるために、その識別子を削除します。



## 8.6.6 識別子の削除

ライト・データベースから識別子を削除する前には、次の操作を行います。

1. システムの ACL から、対象識別子の出現をすべて削除します。たとえば次のコマンドは、複数のファイルの ACL から、古い識別子 87SUMMER を削除します。

```
$ SET SECURITY/ACL=(IDENTIFIER=87SUMMER)-  
_$/DELETE/LOG *.*;*
```

ACE が含まれないファイルに関するエラーが表示されますが、ACE が含まれる全ファイルからは ACE が削除されます。

2. 識別子 87SUMMER をライト・データベースから削除するには、AUTHORIZE の REMOVE/IDENTIFIER コマンドを使用します。たとえば、識別子 87TERM3 を削除するには、次の AUTHORIZE コマンドを使用します。

```
UAF> REMOVE/IDENTIFIER 87TERM3  
{message}
```

ACE に 16 進形式の識別子がある場合、それは汎用識別子がライト・データベースから削除されていることを示します。同様に、識別子が数値形式の UIC として表示されている場合、元の識別子は UIC で、削除されていることを示します。数値形式の UIC または 16 進形式の識別子を持つ ACE は削除します。

従業員の退職後には、UIC を再利用しないことをお勧めします。新しい従業員が、数値形式の古い UIC を依然として参照している ACL エントリを通じて、前任の従業員のアクセス権の一部またはすべてを手に入れる場合があるためです。

識別子の名前を変更するには、次の形式で AUTHORIZE の RENAME/IDENTIFIER コマンドを使用します。

```
RENAME/IDENTIFIER old-identifier new-identifier
```

識別子の名前を変更しても、以前の識別子を通じて利用できた資源のセットは維持されます。名前を変更した識別子が含まれる ACL は、自動的に新しい識別子名を表示します。

## 8.6.7 識別子のカスタマイズ

ライト・リストに識別子を追加する時、あるいは、ユーザに識別子を付与する時、その識別子に属性と呼ばれる特別な特性を持たせることができます。使用可能な属性は数多くありますが、大半のサイトでは次の属性をよく使用します。

Dynamic 属性	識別子の保持者に対して、DCL の SET RIGHTS_LIST コマンドを使用した、プロセス・ライト・リストからの識別子の削除および復元を許可します。
Resource 属性	識別子の保持者に対して、識別子へのディスク領域の割り当てを許可します。この属性はファイル・オブジェクトを対象に使用します。
Subsystem 属性	識別子の保持者に対して、アプリケーション・イメージへの Subsystem ACE の割り当てによる、保護サブシステムの作成と保守を許可します。
No Access 属性	識別子のすべてのアクセス権を空かつ無効にします。この属性は、資源識別子の修飾子として使用するか、アクセス制御とは無関係の目的に使用します。

多くの場合、セキュリティ要件が高いサイトでは、以上の他に、ユーザによるライト・データベースの検索を防止する次の 2 つの属性を使用します。

Holder Hidden 属性	識別子を所有している本人でない限り、その識別子を割り当てられているユーザのリストを取得することを禁止します。
Name Hidden 属性	識別子の保持者に、識別子の変換 (バイナリから ASCII、またはその逆) を許可しますが、権限のないユーザが識別子を変換することを禁止します。

RIGHTSLIST.DAT への読み取りアクセス権は、Holder Hidden 属性および Name Hidden 属性をオーバーライドします。デフォルトでは、ライト・リストはワールド・ユーザにアクセス権を付与しません。ライト・リストの保護は、"S:RWED,O;RWED,G:R,W:" となっています。

以降の節では各属性について解説し、どのような場合にサイトの識別子の一部に属性を追加する必要があるかを説明します。

#### 8.6.7.1 Dynamic 属性

ユーザに識別子を付与すると、そのユーザによって作成されるプロセスは、プロセスが存続し続ける間、識別子を保持します。しかし、Dynamic 属性を指定した識別子を付与すると、その識別子を保持するユーザは、DCL の SET RIGHTS\_LIST コマンドを使用して、必要に応じてプロセス・ライト・リストを対象に識別子またはその属性の追加/削除を行うことができます。

識別子の変更をユーザに許可するには、次の例のように AUTHORIZE を使用して、ライト・データベースに識別子を追加する際に Dynamic 属性を指定します。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD/IDENTIFIER MGMT101 /ATTRIBUTES=DYNAMIC
```

識別子の特定の保持者に識別子の変更を許可するには、次のように、識別子を付与する際に Dynamic 属性を追加します。

```
UAF> GRANT/IDENTIFIER MGMT101/ATTRIBUTES=DYNAMIC SCHWARTZ
```

以降、ユーザ Schwartz は、次のコマンドを使用してプロセス・ライト・リストから MGMT101 識別子を削除できます。

```
$ SET RIGHTS_LIST/DISABLE MGMT101
```

また、Dynamic 属性と Resource 属性を持つ識別子を保持するユーザは、SET RIGHTS\_LIST コマンドを使用してその識別子の Resource 属性のみを削除できます。

識別子を削除することで、設定されているセキュリティ・ポリシーをユーザが回避できる場合があるため、Dynamic 属性を持つ識別子をユーザに付与する際には注意してください。Dynamic 属性を持つ識別子を保持するユーザにアクセス権を付与しないために、ACL で識別子が使用されている場合、ユーザは、プロセス・ライト・リストから識別子を削除することで、別の ACL エントリを通じてオブジェクトへのアクセス権を取得できる場合があります。

#### 8.6.7.2 Holder Hidden 属性

セキュリティ要件が高いサイトでは、特定の識別子の保持者を隠して、侵入のターゲットとして望ましいアカウントを悪意のあるユーザが判定できないようにすることが可能です。

AUTHORIZE の MODIFY/IDENTIFIER コマンドを使用して、ユーザが保持する識別子に属性を適用します。次に例を示します。

```
UAF> MODIFY/IDENTIFIER /ATTRIBUTES=HOLDER_HIDDEN SECRET_PROJECT
```

これで、詮索行為を行う人物は、秘密プロジェクトに関与している人物がわからなくなります。

#### 8.6.7.3 Name Hidden 属性

セキュリティ要件が高いサイトでは、識別子の名前を隠すことができます。たとえば、強制アクセス制御を実装するサイトでは、セキュリティ・カテゴリに関連付けられた識別子の名前を隠すことができます。これにより、識別子を保持する本人でなければ、識別子の名前を表示できなくなります。識別子に Name Hidden 属性が指定されている場合、要求側のプロセスがその識別子を保持している場合を除き、オペレーティング・システムは、バイナリ値から ASCII または ASCII からバイナリ値への識別子の変換を拒否します。

この属性を識別子に割り当てるには、次のように AUTHORIZE の MODIFY/IDENTIFIER コマンドを使用します。

```
UAF> MODIFY/IDENTIFIER SECRET_NEWS /ATTRIBUTES=NAME_HIDDEN
```

#### 8.6.7.4 No Access 属性

No Access 属性は、識別子の保持をプロセスに許可しますが、オブジェクトへのアクセス権の判定において、その識別子を考慮しないようにします。

たとえば、Resource 属性と No Access 属性を持つユーザは、識別子にディスク領域を割り当てることができますが、その識別子が所有するオブジェクトにはアクセスできません。また、システム管理者はデータを管理し、そのデータに関連する作業を実行できますが、ファイルの読み書きはできません。

セキュリティ管理者は、識別子によるファイル領域の所有、および識別子へのファイル領域の割り当てを許可する一方で、ファイル・アクセスを禁止することができます。ライト・データベースに識別子を追加する際に、Resource 属性とともに No Access 属性を指定するには、次の例のように AUTHORIZE を使用します。

```
UAF> ADD/IDENTIFIER/ATTRIBUTES=(RESOURCE,NOACCESS)-  
_UAF> MGMT101
```

Resource 属性を持つ識別子を保持するユーザの権限を制限するには、次のように、対象の全ユーザに対して、Resource 属性のほかに No Access 属性も付属する識別子を付与します。

```
UAF> GRANT/IDENTIFIER/ATTRIBUTES=(RESOURCE,NOACCESS)-  
_UAF> MGMT101 SCHWARTZ
```

#### 8.6.7.5 Resource 属性

一般的にディスク領域の消費は、ファイル所有者のディスク制限からディスク領域を差し引くことにより、各ファイルの作成者に割り当てられます。システム管理者とセキュリティ管理者は、個別ユーザではなく、(部署やプロジェクトなどの)ユーザの論理グループに従って、ディスク領域の使用状況を追跡することが望ましい場合があります。このようなグループの指定には、汎用識別子を使用します。したがって、汎用識別子がディレクトリを所有する場合、ディレクトリに作成されたファイルにより使用されるディスク領域は、ファイルの作成者の UIC ではなく、識別子に割り当てられる場合があります。

ファイル領域を識別子が所有するようにして、識別子への割り当てを可能にするには、次の例のように、ライト・データベースに識別子を追加する際に、AUTHORIZE を使用して Resource 属性を指定します。

```
UAF> ADD/IDENTIFIER MGMT101 /ATTRIBUTES=RESOURCE
```

識別子の特定の保持者に対して、識別子へのディスク領域の割り当てを許可するには、次の手順を実行します。

1. Resource 属性を持つ識別子を、すべての対象ユーザに付与します。

```
UAF> GRANT/IDENTIFIER MGMT101/ATTRIBUTES=RESOURCE SCHWARTZ
```

2. ディレクトリに変更を加え、資源識別子への読み込みアクセス権と書き込みアクセス権を許可します。

```
$ SET SECURITY/ACL=(-  
_$(IDENTIFIER=MGMT101,ACCESS=READ+WRITE) -  
_$(IDENTIFIER=MGMT101,OPTIONS=DEFAULT,ACCESS=READ+WRITE))-  
_ $ INVENTORY.DIR
```

3. デフォルトで親ディレクトリ内のすべてのファイルが識別子に所有されるよう、親ディレクトリの所有権を変更します。

```
$ SET SECURITY/OWNER=MGMT01 INVENTORY.DIR
```

資源識別子 MGMT101 は、セキュリティ管理者がディレクトリ INVENTORY.DIR に作成するすべてのファイルを所有することになるため、セキュリティ管理者は ACE を使用して、与えられるファイル・アクセス権のタイプを指定します。ファイルの作成者に付与されるアクセス権を設定するには、作成者 ACE (CREATOR,ACCESS=READ+WRITE+EXECUTE+DELETE) を含めます。また、システムに ACE を割り当てさせることもできます。システムの ACE はファイルの作成者に対して、制御アクセス権と、保護コードの所有者フィールドで指定されているアクセス権を付与します。保護コードを設定するには、INVENTORY.DIR の ACE に、(DEFAULT\_PROTECTION, ACCESS=O:RW) のように、デフォルトの保護用 ACE を含めます。詳細については、8.8.1.2 項を参照してください。

識別子を保持する全ユーザが、その識別子と関連付けられた Resource 属性も保持するとは限りません。ある識別子によって所有されているディレクトリにファイルを作成しても、その識別子の Resource 属性を持っていなければ、そのファイルはユーザの UIC により所有され、必要なディスク領域はユーザのディスク制限から差し引かれます。

#### 8.6.7.6 Subsystem 属性

Subsystem 属性を持つサブシステム識別子をユーザに付与することで、保護サブシステムを管理する権限をユーザに付与できます。これによりユーザは、サブシステムによって管理されるオブジェクトにイメージがアクセスできるようにすることが可能になります。保護サブシステムの説明については、第 13 章を参照してください。

次の例では、識別子 MAIL\_SUBSYSTEM を使用して、ユーザ Schwartz にサブシステムを作成する権限を付与しています。また Schwartz には、アクセス制御を設定するためのアプリケーション・イメージへの制御アクセス権も付与しています。

```
$ SET DEFAULT SYS$SYSTEM  
$ RUN AUTHORIZE  
UAF> ADD/IDENTIFIER MAIL_SUBSYSTEM /ATTRIBUTES=SUBSYSTEM  
UAF> GRANT/IDENTIFIER MAIL_SUBSYSTEM -  
_UAF> /ATTRIBUTES=SUBSYSTEM SCHWARTZ  
UAF> Exit  
$ SET SECURITY/ACL=(IDENTIFIER=MAIL_SUBSYSTEM,ACCESS=CONTROL)-  
_ $ MEMBER_LIST.EXE
```

### 8.6.8 システムまたはプロセス・ライト・リストの変更

特権セキュリティ管理者は、SET RIGHTS\_LIST コマンドを使用して、システム上の任意のプロセスのライト・リストを変更したり、システム・ライト・リストの識別子を変更できます。システム・ライト・リストに識別子を追加すると、その識別子を全ユーザに付与することになります。また、SET RIGHTS\_LIST コマンドを使用して、既存の識別子に属性を追加することもできます。

システム・ライト・リストの使用法の 1 つに、サイト固有の環境条件の有効化があります。たとえば、午前 8 時に実行するスケジュールが組まれているバッチ・ジョブは、次の識別子を追加することができます。

```
$ SET RIGHTS_LIST/SYSTEM/ENABLE DAY_SHIFT
```

午後 5 時に実行するスケジュールが組まれている別のバッチ・ジョブは、次のように、識別子 DAY\_SHIFT を削除することができます。

```
$ SET RIGHTS_LIST/SYSTEM/DISABLE DAY_SHIFT
```

結果的に、識別子 DAY\_SHIFT を持つ保護オブジェクトへのアクセスが、午前 8 時から午後 5 時まで有効になります。

次の例のコマンドは、プロセス DEDNAM のライト・リストに SALES 識別子を追加することで、プロセス・ライト・リストを変更しています。Resource 属性を指定することで、SALES 識別子の保持者に、その識別子へのディスク領域の割り当てを可能にしています。

```
$ SET RIGHTS_LIST/ENABLE/ATTRIBUTES=RESOURCE/PROCESS=DEDNAM SALES
```

## 8.7 ユーザへの特権の付与

一部のシステム処理は、特定の特権を有するユーザに制限されます。このような制限により、オペレーティング・システムの性能の一貫性が守られる結果、ユーザに提供されるサービスの一貫性も保たれます。各ユーザへの特権の付与は、(a) 特権に対する正当なニーズがユーザにあるかどうか、(b) システムを妨害することなく特権を使用するスキルと経験をユーザが持っているかどうか、という 2 つの要素に基づいて判断します。

ユーザの特権は、そのユーザの UAF レコードに、2 つの特権ベクタとして記録されます。一方のベクタには許可された特権が格納され、もう一方のベクタにはデフォルトの特権が格納されます。デフォルトの特権は、ログイン時にユーザ・プロセスが獲得する、許可された特権のサブセットです。

ユーザがシステムにログインすると、ユーザの特権ベクタが、ユーザのプロセスのヘッダに保存されます。このようにして、そのユーザの特権は、ユーザに対して作成されるプロセスに渡されます。ユーザは、DCL の SET PROCESS/PRIVILEGES コマンドを使用して、ユーザに許可される特権を有効/無効にすることができます。

オペレーティング・システムは、特権の使用状況を監視および監査します。特定の特権に対する監査を有効にし、監査ログ・ファイルを調べることで、DCL コマンドまたはシステム・サービスの実行にどの特権が使用されたかを確認できます。詳細については、第 9 章を参照してください。

### 8.7.1 特権のカテゴリ

特権を有するユーザがシステムにもたらす可能性のある損害に従って、特権は次の 7 つのカテゴリに分けられています。

- None : 特権なし
- Normal : システムを有効に使用するための最低限の特権
- Group : 同一グループのメンバに影響が及ぶ可能性
- Devour : システム全体のクリティカルではない資源を消費する可能性
- System : 通常のシステム操作に影響が及ぶ可能性
- Objects : オブジェクトのセキュリティを危険にさらす可能性
- All : システムを制御する可能性

表 8-2 に、特権の分類と、各特権に関連付けられている能力の簡単な定義を示します。

表 8-2: OpenVMS の特権

カテゴリ	特権	許可される処理
None	None	特権を必要とする処理を拒否
Normal	NETMBX TMPMBX	ネットワーク接続の作成，一時メールボックスの作成
Group	GROUP GRPPRV	同一グループのプロセスの制御，グループのオブジェクトのシステム保護フィールドを通じたアクセスの取得
Devour	ACNT ALLSPOOL BUGCHK EXQUOTA GRPNAM PRMCEB PRMGBL PRMMBX SHMEM	アカウントの無効化，スプールされたデバイスの割り当て，バグチェック・エラー・ログのエントリの作成，ディスク制限の超過，名前テーブルへのグループ論理名の挿入，パーマメント・コモン・イベント・フラグ・クラスタの作成/削除，パーマメント・グローバル・セクションの作成，パーマメント・メールボックスの作成，共有メモリでの構造体の作成/削除

表 8-2: OpenVMS の特権 (続き)

カテゴリ	特権	許可される処理
System	ALTPRI AUDIT OPER PSWAPM WORLD SECURITY SYSLCK	割り当てよりも高いベース優先順位の 設定, 監査レコードの作成, オペレー タ機能の実行, プロセス・スワップ・ モードの変更, 任意のプロセスの制 御, セキュリティ関連機能の実行, シ ステム全体の資源のロック
Objects	DIAGNOSE IMPORT MOUNT READALL SYSGBL VOLPRO	デバイスの診断, ラベルのないテー プ・ボリュームのマウント, マウン ト・ボリューム QIO の実行, 全シス テム・オブジェクトに対する読み込み アクセス権の所有, システム全体の グローバル・セクションの作成, ボ リューム保護のオーバーライド
All	BYPASS CMEXEC CMKRNL IMPERSONATE DOWNGRADE LOG_IO PFNMAP PHY_IO SETPRV SHARE SYSNAM SYSPRV UPGRADE	保護の無視, エグゼクティブ・モード への変更, カーネル・モードへの変 更, 任意の UIC の独立プロセスの作 成, より低い秘密オブジェクトへの書 き込みまたはオブジェクトの分類の低 下, 論理入出力要求の発行, 特定の物 理ページへのマッピング, 物理入出力 要求の発行, 任意の特権の有効化, ほ かのユーザに割り当てられたデバイス へのアクセス, 名前テーブルへのシ ステム論理名の挿入, システム保護 フィールドを通じたオブジェクトへの アクセス, より高い統一性オブジェク トへの書き込みまたはオブジェクトの 統一性レベルの上昇

## 8.7.2 推奨される特権の割り当て

付録 A に, すべてのユーザ特権と, ユーザ特権を付与すべき条件に関する推奨事項を示します。ユーザ特権を割り当てる際には, 慎重に行います。

表 8-3 の要約ガイドラインは, システム・ユーザの一般的なクラスに対する最低限の特権の要件です。



表 8-3: システム・ユーザの最低限の特権

ユーザのタイプ	最低限の特権
一般	TMPMBX , NETMBX
オペレータ	OPER
グループ管理者	GROUP , GRPPRV
システム管理者	SYSPRV , OPER , SYSNAM , CMKRNL <sup>a</sup>
セキュリティ管理者	SECURITY , AUDIT , READALL

<sup>a</sup>多くの場合、汎用のシステム管理者は、BYPASS を除くすべての特権で構成される、許可された特権のセットが必要です。

### 8.7.3 ユーザ特権の制限

特権を付与すると、セキュリティ管理者が特権を削除するまで、ユーザに特権が認められます。このような全面的な許可を避けるには、必要に応じて特権を付与するようにします。たとえば一部のユーザが、強力な特権のいずれかを必要とするプログラムを実行しなければならない場合があります。その場合には、インストール・ユーティリティ (INSTALL) を使用して、必要な特権を与えてプログラムをインストールします。8.7.4 項で、特権イメージのインストールを詳細に説明しています。

全面的な特権の付与に代わる方法としては、緊急または専用の特権アカウントを設定する方法があります。ユーザは、特定の機能を実行するためにのみ、このような特権アカウントにログインします。この方法には、次の 2 つの選択肢があります。

- アカウントの存在を知っていて、その使用方法を知らされている、限定されたユーザのグループを作成する。
- ユーザ用に 2 つのアカウントを作成し、一方のアカウントには特権を付与し、もう一方のアカウントには特権を付与しない。この場合、対象ユーザは両方のアカウントで、同じ UIC と同じデフォルト・ディレクトリを持ちます。これは、(実際に存在するユーザが 1 人のみであるため) UIC の共有が推奨される唯一のケースです。この二重アカウントの手法を採用する場合は、どのアカウントが特権アカウントであるかがわかるような、明白なユーザ名は避けます。

どちらの方法でも、長いパスワード、短いパスワード有効期間、時間帯の制限、操作モードの制限 (ダイアルアップ、ネットワーク、遠隔、またはバッチ・ログインは不可) など、特権アカウントに特別な制限を設定できます。また、アカウントの有効期間を短くすれば、特権の要件を頻繁に検討するように求められます。

もう 1 つの方法として、第 13 章で説明している保護サブシステムを使用して、システム特権の必要性をなくすという方法もあります。

#### 8.7.4 特権イメージのインストール

必要な特権を有する既知イメージとしてイメージをインストールしない限り、ユーザは、そのユーザが所有していない特権を必要とするイメージは実行できません。既知イメージのインストール方法については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照してください。特権を有する既知イメージを実行すると、イメージを実行している間、イメージを実行しているユーザ・プロセスに特権が付与されます。したがって、(HP が提供する通常の設定以外の)強力な特権を有するイメージは、イメージの機能により特権が必要であり、イメージが安全に機能することが確認された後にのみインストールします。また、イメージへのアクセス権を、一部のユーザに制限することも検討してください。

特権を使用してインストールされたイメージは、強力な特権がすべて有効になった状態で起動されます。安全性を最大限にするため、強力な特権を使用して実行するよう設計されたイメージは、\$SETPRV システム・サービスを使用して、すべての強力な特権を起動直後に無効にし、必要な場合にのみ有効化するべきです。

特権を有するイメージのインストールの例を次に示します。System Dump Analyzer (SDA) ユーティリティは、実行中のシステムを分析するために CMKRNL 特権が必要です。

1. 次のように CMKRNL 特権を与えて、SDA.EXE をインストールします。  

```
$ INSTALL SDA.EXE /PRIVILEGED=CMKRNL
```
2. 次のように、SDA.EXE に ACL を適用し、UIC ベースの保護を設定して、ワールド・カテゴリのユーザにはすべてのアクセス権を拒否します。  

```
$ SET SECURITY/ACL=(IDENTIFIER=SDA,ACCESS=EXECUTE) -  
_$ SYS$SYSTEM:SDA.EXE  
$ SET SECURITY/PROTECTION=(WORLD) SYS$SYSTEM:SDA.EXE
```
3. SDA 識別子を保持するユーザが、プログラムを実行する予定のユーザであることを確認するには、AUTHORIZE コマンドを使用します。必要に応じて、このユーザ・リストを調整します。

---

#### 注意

---

オンラインでのデバッグとトレースバックを防止するため、特権を与えてインストールするイメージはすべて、/NOTRACEBACK 修飾子を使用してリンクする必要があります。

HP により、オペレーティング・システムに付属するすべてのシステム・プログラム (SDA など) は、オンラインでのデバッグやトレースバックを防止するため、/NOTRACEBACK 修飾子を使用してリンクされています。

---

### 8.7.5 コマンド出力の制限

一部の DCL コマンドは、ユーザが保持する特権に応じて動作が異なります。

たとえば、ユーザが GROUP 特権または WORLD 特権を保持している場合を除き、SHOW PROCESS コマンドは、プロセス情報の表示をユーザのプロセスに限定します。GROUP 特権を持つユーザは、自身が所属する UIC グループの他のプロセスを表示できます。また、WORLD 特権を持つユーザは、システム上の任意のプロセスを表示できます。

## 8.8 デフォルトの保護と所有権の設定

ユーザのグループと識別子を設計したら、どの保護オブジェクトに対するアクセス許可をユーザが必要とし、どの保護オブジェクトの制限を解除できるかを検討する必要があります。第 5 章に示す新しいオブジェクトのデフォルトの保護を十分に把握し、必要な場合は、以降の節で説明する手順でデフォルトを変更します。

オブジェクトの保護と所有権のデフォルトを設定する手順は、オブジェクトがファイルであるか、別のクラスの保護オブジェクトであるかに応じて異なります。

### 8.8.1 ファイル・アクセスの制御

5.4.5 項「プロファイルの割り当て」で説明しているように、ユーザに影響を与える保護のデフォルトを指定できる領域は、4 つ存在します。影響が大きい順に、次のとおりです。

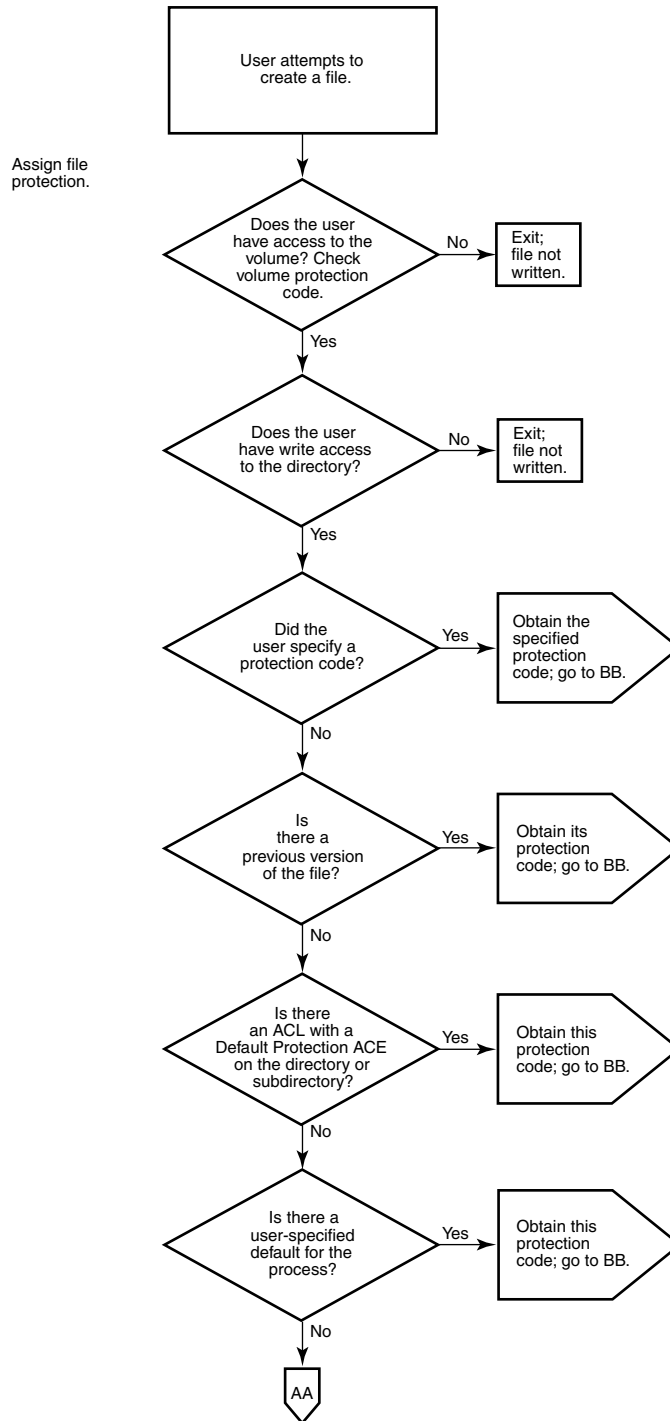
- システム・パラメータ RMS\_FILEPROT は、ファイル保護に関するシステム全体でのデフォルトを設定します。RMS\_FILEPROT の値は、AUTOGEN を使用して変更できます。ただし、この値は、次のデフォルト設定によりオーバーライドされる場合があります。
- DCL の SET PROTECTION/DEFAULT コマンドを使用して、ターミナル・セッション中にユーザが作成または修正するファイルに適用されるファイル保護を指定できます。通常このコマンドはユーザのログイン・コマンド・プロシージャに含まれていますが、ユーザがこのコマンドをセッション中の任意の時点で入力し、SET PROTECTION/DEFAULT コマンドによって事前に設定された値をオーバーライドすることもできます。SET PROTECTION/DEFAULT コマンドは、該当ユーザに対するシステム全体の保護設定を無効にします。
- 特定のディレクトリに対するデフォルトの保護設定は、ディレクトリに適用される ACL で指定できます。ディレクトリに対するデフォルトの保護用 ACE が存在する場合、ディレクトリに追加されるすべての新しいファイル(サブディレクトリおよびそこに格納されるファイルを含む)は、この保護コードの対象になります。このコードは、システム全体のデフォルト設定と、(存在する場合は)ユーザ指定のデフォルト設定をオーバーライドします。

- 作成されるファイルが、ファイルを作成するプロセスのユーザ識別コード (UIC) により所有されない特別なケース (たとえば、ディレクトリが資源識別子により所有されている場合) では、その新しいファイルのデフォルトの保護を、ディレクトリの ACL 内の作成者 ACE によって変更できます。作成者 ACE の説明については、5.4.5 項「プロファイルの割り当て」を参照してください。

また、DCL の SET VOLUME/PROTECTION コマンドによってボリュームに課せられる保護も考慮します。この保護コードが指定されている場合は、ディレクトリおよびファイルに対する保護コードに関係なく、ボリュームのあらゆる部分への該当ユーザによるアクセスが禁止されます。SET VOLUME コマンドを使用してボリューム保護を指定していない場合、該当ボリュームには全ユーザがアクセスできます。

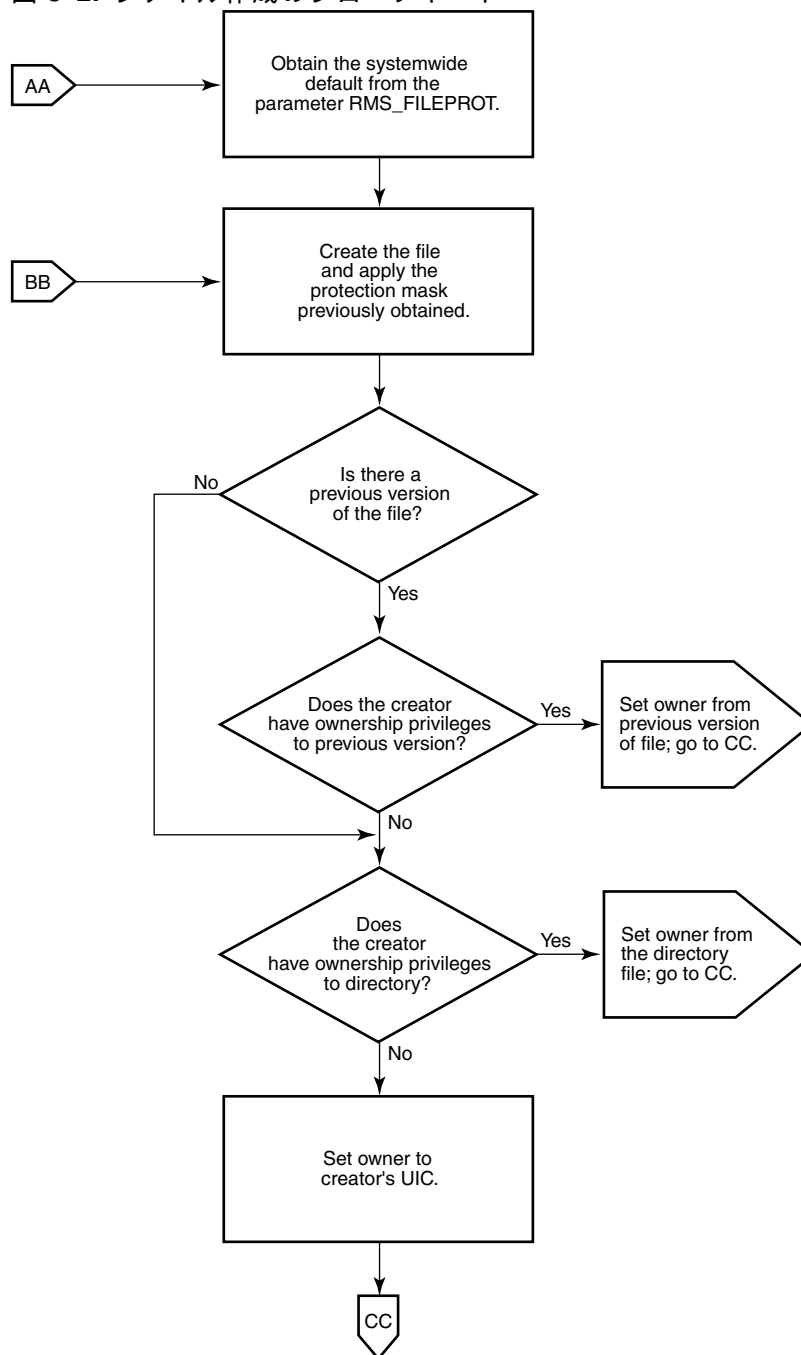
ファイル所有権の割り当ては、保護チェックの結果に影響します。この組み合わせによる保護構造の運用上の効果を、図 8-1、図 8-2、および図 8-3 に示します。

図 8-1: ファイル作成のフローチャート



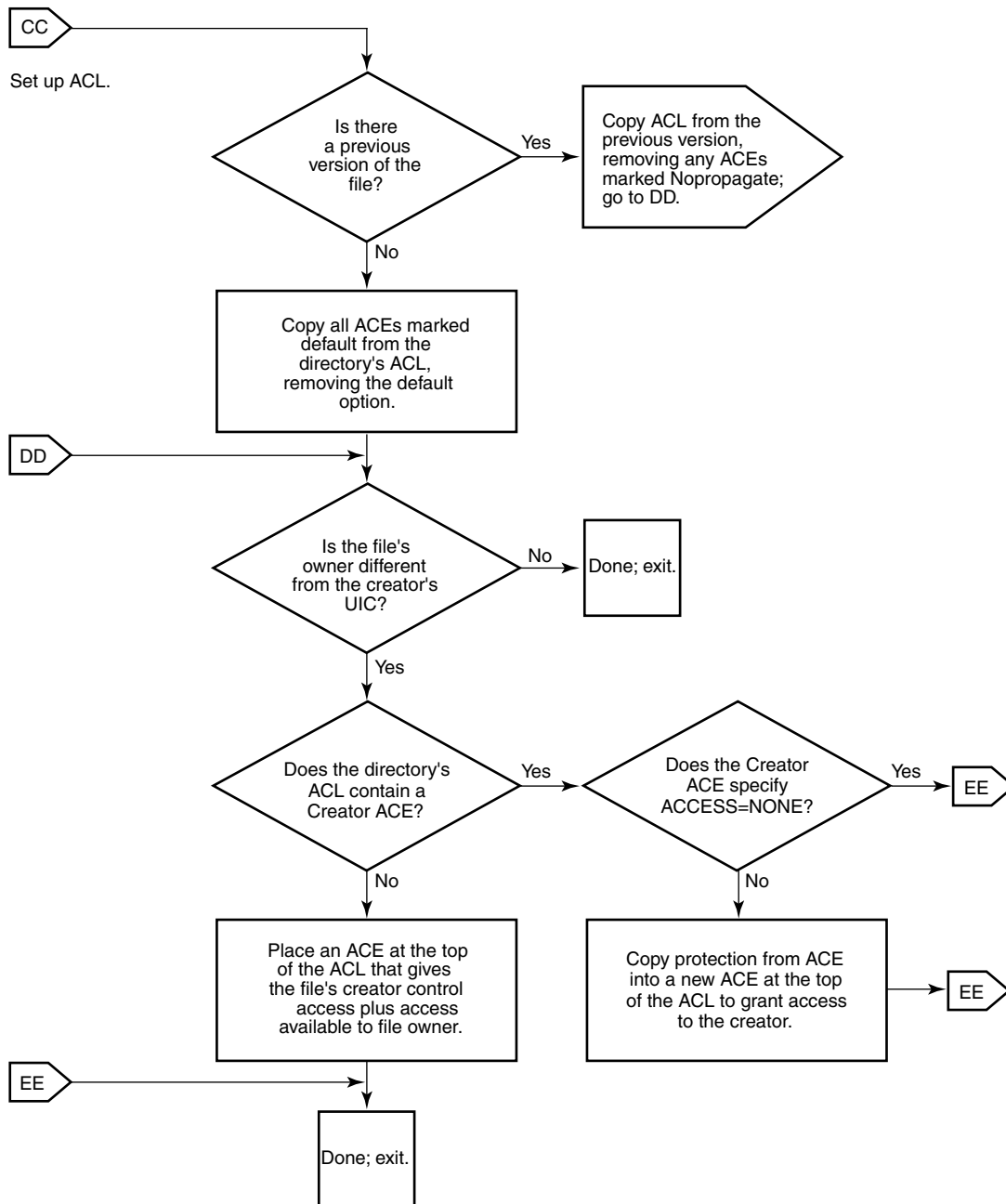
VM-1000A-AI

図 8-2: ファイル作成のフローチャート



VM-1000B-AI

図 8-3: ファイル作成のフローチャート



VM-1000C-AI

### 8.8.1.1 保護のデフォルトの調整

デフォルトの動作を制御するために調整を行うことができます。システム・パラメータ RMS\_FILE PROT により指定されるシステム全体のデフォルトの保護コードにより、ユーザのデフォルトの保護は次のように設定されます。

(S:RWED,O:RWED,G:RE,W)

ボリューム保護が、オペレータにより次のように設定されたとします。

(S:RWED,O:RWED,G:R,W)

ディレクトリ [PROJECT] に対するファイル保護が、次のように設定されています。

(S:RWED,O:RW,G:R,W)

サブディレクトリ [PROJECT.DIARY] に作成された全ファイルに高度な保護が必要である場合は、セキュリティ管理者、またはディレクトリへの制御アクセス権を持つ任意のユーザは、次のように、デフォルトの保護用 ACE で構成される ACL を持つこの特定のディレクトリに対して、特別なデフォルトの保護コードを定義できます。

(DEFAULT\_PROTECTION,S:RWED,O:RWED,G,W)

次の DCL コマンドによって、必要なデフォルトの保護を実現できます。

```
$ SET SECURITY/ACL=(DEFAULT_PROTECTION,S:RWED,O:RWED)-  
_$ [PROJECT]DIARY.DIR
```

この ACE がディレクトリ・ファイルに適用されると、そのディレクトリで作成または変更されるファイルは、デフォルトの保護コードの対象になります。これらの保護コードはデフォルトにすぎないため、ディレクトリ内のファイルへの制御アクセス権を持つユーザは、次の DCL コマンドを使用することで、ファイルのデフォルト値の置き換えとして、固有の保護コードを含めることができます。

- SET SECURITY/PROTECTION
- COPY/PROTECTION
- APPEND/PROTECTION
- CREATE/PROTECTION

デフォルトの保護コードを置き換えると、新しいコードがデフォルトとなり、それ以降のバージョンのファイルに反映されます。

一部のユーザに特別なログイン・コマンド・プロシーダを用意する場合、対象グループのユーザに対して、システム・パラメータ RMS\_FILEPROT により指定されるシステム全体のデフォルトのプロセス保護を追加することができます。デフォルトのプロセス保護を指定するには、次のように、ログイン・コマンド・プロシーダに SET PROTECTION/DEFAULT コマンドを追加します。

```
SET PROTECTION=(S:RWED,O:RWED,G,W)/DEFAULT
```

ユーザのディレクトリに作成されたファイルは、明示的にオーバーライドする場合を除き、このデフォルトの保護コードが適用されます。



### 8.8.1.2 資源識別子により所有されるディレクトリのデフォルトの設定

より柔軟性の高いデータ管理と、より正確なディスク使用量の会計管理を実現するために、資源識別子に所有されるディレクトリを設定し、ACL を使用して、ディレクトリとディレクトリ内で作成されるファイルへのアクセス権を制御することができます。

ACL は、プロジェクト識別子を保持するすべてのプロジェクトのメンバに対して、ファイル・アクセスを制限できます。このようなアクセス制限を実現するには、識別子用 ACE を追加して、ファイルへのグループのアクセス権を定義します。追加される 2 つ目の識別子用 ACE は、最初の識別子用 ACE の複製ですが、Default 属性を保持しています。この Default 属性によって、ディレクトリ内で作成される全ファイルに、その ACE がコピーされることが保証されます。ディレクトリのデフォルトの保護コードによっては、3 つ目の ACE であるデフォルトの保護用 ACE が必要になる場合があります。デフォルトの保護用 ACE は、ディレクトリのファイルに対する保護コードを設定します。4.3 節「システムによる保護オブジェクトへのユーザのアクセス可否の判定」の説明にあるように、ACL によってファイルへのアクセスが禁止されている場合であっても、保護コードを通してアクセス権を得ることが可能です。

ACL は、ファイルへのグループのアクセスを制限するだけでなく、共通ディレクトリ内にユーザが作成したファイルに対するユーザのアクセス権のタイプを制御できます。ファイルは資源識別子のディレクトリに作成されるため、資源識別子はそのファイルを所有します。ユーザが作成したファイルにユーザ自身がアクセスするために、オペレーティング・システムは通常、ファイルの作成者に対して制御アクセス権を付与するだけでなく、保護コードの所有者フィールドで指定されているアクセス権も付与します。ただし、ディレクトリの ACL に作成者 ACE を追加することで、この動作を変更できます。作成者 ACE は、ユーザがプロジェクトのディレクトリに作成したファイルに対してユーザが保持するアクセス権のタイプを定義します。

#### 8.8.1.2.1 資源識別子の設定

セキュリティ管理者が、次のコマンド・シーケンスを使用してプロジェクト識別子 PROJECTX を設定し、それをプロジェクトのメンバに付与したとします。プロジェクト識別子を、資源識別子を持っているライト・データベースに追加しているほか、資源識別子を持つユーザにも付与しています。プロジェクト識別子は、ディスク領域を所有できるように、Resource 属性を持つ必要があります。

```
$ RUN SYS$SYSTEM:AUTHORIZE
UAF> ADD/IDENTIFIER PROJECTX /ATTRIBUTES=RESOURCE
UAF> GRANT/IDENTIFIER PROJECTX user1 /ATTRIBUTES=RESOURCE
UAF> GRANT/IDENTIFIER PROJECTX user2 /ATTRIBUTES=RESOURCE
:
```

#### 8.8.1.2.2 資源識別子のディレクトリの設定

プロジェクトや部署に固有の識別子がディレクトリの所有者である場合、そのディレクトリに作成されるファイルによって使用される領域は、ファイルを作成した個人ではなく、適切な部署やプロジェクトに割り当てることができます。ユーザが複数のプロジェクトに関わっている場合は、ユーザの個人用アカウントではなく、該当するプロジェクトにディスク領域の要件を割り当てることができます。

資源識別子が保有するディレクトリを設定するには、まずプロジェクト識別子に許可されるディスク制限を作成します。たとえば、次のコマンドはシステム管理ユーティリティ (SYSMAN) を起動し、超過値を 200 ブロックとして、識別子 PROJECTX に 2000 ブロックのディスク制限を割り当てています。

```
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> DISKQUOTA ADD PROJECTX /PERMQUOTA=2000 /OVERDRAFT=200
```

ディスク制限を設定したら、プロジェクト・ディレクトリを作成します。たとえば、次の DCL コマンドでは、プロジェクト・ディレクトリ [PROJECTX] を作成し、その所有者として識別子 PROJECTX を設定しています。

```
$ CREATE/DIRECTORY [PROJECTX] /OWNER=[PROJECTX]
```

#### 8.8.1.2.3 ACL の設定

ディレクトリ [PROJECTX] を設定するには、ACL を使用して、プロジェクトのメンバにファイル・アクセス権を付与します。次の例に、複数の ACE を使用してアクセス権を定義する方法を示します。

```
$ SET SECURITY [PROJECTX] /ACL= (-
_$ (DEFAULT_PROTECTION,S:RWED,O:RWED,G,W),- [1]
_$ (IDENTIFIER=PROJECTX,ACCESS=READ+WRITE+EXECUTE),- [2]
_$ (IDENTIFIER=PROJECTX,OPTIONS=DEFAULT,ACCESS=READ+WRITE+EXECUTE),-
_$ (CREATOR,ACCESS=READ+WRITE+EXECUTE+DELETE)) [3]
```

1. デフォルトの保護用 ACE は、ディレクトリ内に作成されるファイルに対して保護コードを設定します。この ACE は、グループ・ユーザおよびワールド・ユーザにはアクセス権を付与しません。
2. 最初の識別子用 ACE は、PROJECTX 識別子の保持者に、ディレクトリの読み込みアクセス権、書き込みアクセス権、および実行アクセス権を付与します。
3. 2 番目の識別子用 ACE は、ディレクトリに作成されるすべてのファイルが、最初の識別子用 ACE を保持することを保証します。
4. 作成者 ACE は、PROJECTX ディレクトリにファイルを作成するユーザに、そのファイルの読み込みアクセス権、書き込みアクセス権、実行アクセス権、および削除アクセス権を付与することを指定します。

したがって、プロジェクトのメンバ Ross が [PROJECTX] ディレクトリにファイル SEPTEMBER-REPORTS.TXT を作成すると、ファイルには次のセキュリティ・プロファイルが与えられます。

```
$ SHOW SECURITY/CLASS=FILE [PROJECTX] SEPTEMBER-REPORTS.TXT
```

```
SEPTEMBER-REPORTS.TXT object of class FILE
Owner: [PROJECTX]
Protection: (System: RWED, Owner: RWED, Group, World)
Access Control List:
  (IDENTIFIER=CRANDALL, ACCESS=READ+WRITE+EXECUTE+DELETE)
  (IDENTIFIER=PROJECTX, ACCESS=READ+WRITE+EXECUTE)
```

プロジェクトのメンバは、ほかのユーザにより作成されたファイルを削除（または制御）することは許可されていませんが、作成者 ACE によって、ユーザ自身が作成したファイルの削除アクセス権がユーザに与えられます。

作成者 ACE がいない場合、プロジェクトの各メンバは、自分がディレクトリに作成したファイルへの完全なアクセス権を持っています。たとえば Ross には、プロジェクトのディレクトリに作成されたファイルへの、次のアクセス権が与えられます。

```
$ SHOW SECURITY/CLASS=FILE [PROJECTX] SEPTEMBER-REPORTS.TXT
```

```
SEPTEMBER-REPORTS.TXT object of class FILE
Owner: [ROSS]
Protection: (System: RWED, Owner: RWED, Group, World)
Access Control List:
  (IDENTIFIER=ROSS, OPTIONS=NOPROPAGATE,
    ACCESS=READ+WRITE+EXECUTE+DELETE+CONTROL)
  (IDENTIFIER=PROJECTX, ACCESS=READ+WRITE+EXECUTE)
```

この動作を無効にするには、ACL に対して、ACCESS=NONE を指定する作成者 ACE を追加します。

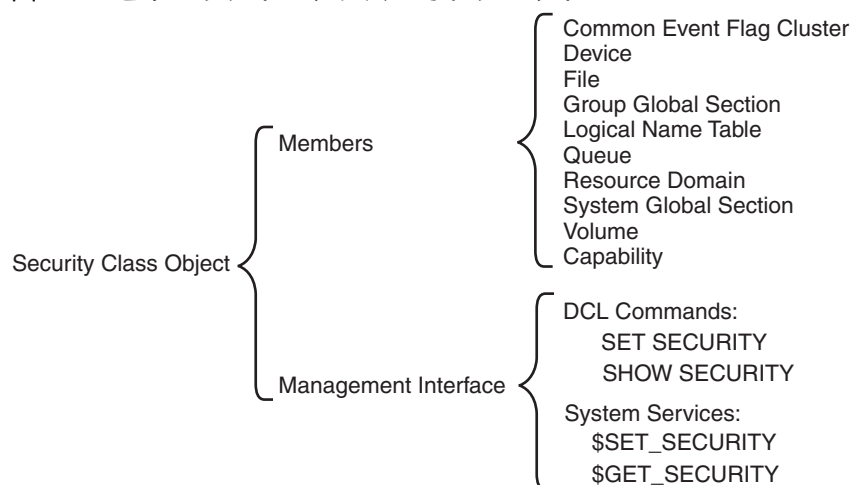
## 8.8.2 ファイル以外のオブジェクトのデフォルトの設定

ファイルと擬似ターミナル (FT) デバイスを除き、保護オブジェクトの全クラスに、新しいオブジェクトに対するセキュリティ要素を提供する 1 つ以上のテンプレート・プロファイルが用意されています。したがって、1 つのメカニズムで、オブジェクトに関するデフォルトの保護コード、ACL、および所有権の要素を設定することができます。あるシステム・スタートアップから次のシステム・スタートアップに移っても、これらの値が使用できるように、オペレーティング・システムは常にこれらの値を保存します。SHOW SECURITY コマンドを使用して、自分のサイトの現在のデフォルト値を表示できます。オペレーティング・システムのデフォルト値の一覧については、第 5 章を参照してください。

オペレーティング・システムは、セキュリティ・クラス・オブジェクトにより保存されたデータから、新しいオブジェクトのセキュリティ・プロファイルを作成します。これらのオブジェクトはすべて論理構造であり、有効なアクセス・タイ

ブ、テンプレート、および有効になっている監査のタイプなどのクラス要素を追跡するために使用されます。図 8-4 に示すように、保護オブジェクトの各クラスは、セキュリティ・クラスのメンバを持っています。独自の規則が適用されるメンバであるファイルを除き、すべてのメンバには、セキュリティ・プロファイル・テンプレートがあります。

図 8-4: セキュリティ・クラス・オブジェクト



VM-1001A-AI

### 8.8.2.1 クラスのデフォルトの表示

クラス・テンプレートを表示するには、SHOW SECURITY/CLASS=SECURITY\_CLASS コマンドを使用します。たとえば次のコマンドは、論理名テーブルに使用できるテンプレートを表示します。論理名テーブル・オブジェクトには、次の 3 つのテンプレートがあります。

```

$ SHOW SECURITY/CLASS=SECURITY_CLASS LOGICAL_NAME_TABLE
:
:
Template: GROUP
Owner: [TTSY,SYSTEM]
Protection: (System: RWCD, Owner: R, Group: R, World:R)
Access Control List: empty

Template: JOB
Owner: [TTSY,SYSTEM]
Protection: (System: RWCD, Owner: RWCD, Group, World)
Access Control List: empty

Template: DEFAULT
Owner: [TTSY,SYSTEM]
Protection: (System: RW, Owner: RW, Group, World)
Access Control List: empty

```

セキュリティ・クラスのオブジェクトはすべて、他のオブジェクトと同じ方法で保護されます。このため、SHOW SECURITY によるセキュリティ・クラス・オブジェクトの表示は、そのオブジェクト自身のセキュリティ・プロファイルから始まります。次の表示は、セキュリティ・クラスにおける論理名テーブル・オブジェクトのプロファイルを示しています。オブジェクトはシステムに所有され、その保護コードにより、すべてのユーザ・カテゴリに読み込みアクセス権が許可されていますが、書き込みアクセス権が許可されているのはシステムと所有者のカテゴリのみです。

```
$ SHOW SECURITY/CLASS=SECURITY_CLASS LOGICAL_NAME_TABLE

LOGICAL_NAME_TABLE object of class SECURITY_CLASS
  Owner: [SYSTEM]
  Protection: (System: RW, Owner: RW, Group: R, World: R)
  Access Control List: [empty]
```

### 8.8.2.2 クラス・テンプレートの変更

セキュリティ管理者と、セキュリティ・クラス・オブジェクトに対する制御アクセス権を持つユーザは、次のコマンドを使用して、指定されたテンプレートの要素を変更することができます。

```
SET SECURITY/CLASS=SECURITY_CLASS/PROFILE=TEMPLATE=template-name
```

次のコマンドは、デバイス・クラスの MAILBOX テンプレートを変更します。テンプレートの値を、S:RWPL,O:RWPL,G:RWPL,W:RWPL の保護から、グループ・アクセスとワールド・アクセスを許可しない保護に変更します。

```
$ SET SECURITY/CLASS=SECURITY_CLASS/TEMPLATE=MAILBOX -
_$ /PROTECTION=(S:RWPL,ORWPL,G,W) DEVICE
```

オペレーティング・システムは、この値をすべての新しいメールボックスに適用します。既存の各メールボックスの保護を変更するには、既存の各メールボックスに対して、明示的な SET SECURITY コマンドを入力します。次に例を示します。

```
$ SET SECURITY/CLASS=DEVICE -
_$ /PROTECTION=(S:RWPL,ORWPL,G,W) mailbox_name
```

オペレーティング・システムは、セキュリティ・テンプレートに指定されているデフォルトのオブジェクト保護を保存するため、システムをリブートすると、リブート後に作成される全オブジェクトが、新しいデフォルトの保護で作成されるようになります。

---

#### 注意

---

OpenVMS バージョン 7.2-1 およびそれ以前のバージョンでは、すべての擬似ターミナル (FT) デバイスの保護コードは、ドライバにより (S:RWLP,O:RWLP,G,W) に設定されていました。OpenVMS バージョン 7.3 以降では、この強制的な保護に設定されるのは、デバイス FTA0 のみです。これによりシステム管理者は、ブート・プロセスの後半で FTA0

デバイスの保護を変更できます。この新しい保護は、FTA0 から、以降作成されるすべての新しい FT デバイスによって継承されます（また、ACL などの、SECURITY クラスの DEVICE TERMINAL テンプレート・プロファイルに由来するその他の設定も継承されます）。

システム管理者は、FTA0 を手動で変更するか、SYSTARTUP\_VMS.COM コマンド・プロシージャを変更することができます。次に例を示します。

```
$ SET SECURITY/CLASS=DEVICE -  
_ $ /PROTECTION=(S:RWLP,O:RWLP,G:RW,W:R) FTA0:
```

FTA0 のデバイス保護に変更を加えなければ、動作は OpenVMS バージョン 7.3 よりも前のバージョンと変わりません。つまり、FT 擬似ターミナル・デバイスを除き、ターミナルはすべて、TERMINAL テンプレート・プロファイルからデバイス保護などのセキュリティ特性を継承します。FTA 擬似ターミナル・デバイスはすべて、FTA0 から保護を継承し、その保護はデフォルトでは (S:RWLP,O:RWLP,G,W) に設定されています。ACL などその他の設定は、TERMINAL テンプレート・プロファイルから継承されます。これにより、既存のアプリケーションとの互換性が保証されます。

---

DCL の SHOW SECURITY コマンドを使用すると、サイトの値を持つ使用可能なテンプレートがすべて表示されます。第 5 章に、デフォルトのシステム値の一覧があります。

## 8.9 システムのデータと資源の追加保護

この節では、ユーザが使用できるデータと資源を制限する、追加の方法を説明します。

### 8.9.1 ソフトウェアの新規インストール時に必要な安全対策

新しいソフトウェアをインストールする際には、セキュリティに関するいくつかの点について対策を講じる必要があります。通常のセキュリティ上の安全対策を何らかの方法で損なったり、弱体化させるソフトウェアのインストールを認めないようにする必要があります。また、インストールするソフトウェアに特権を与えるべきかどうかも考慮する必要があります。この節では、新しいソフトウェアをインストールするときのセキュリティの側面を説明します。

#### 8.9.1.1 潜在的に有害なプログラム

新しいソフトウェアには、システムに対する潜在的な危険のあるプログラムが含まれている可能性があります。トロイの木馬プログラムと呼ばれるこれらのプログラムは、損害を与えることを目的として作成されており、多くの場合、次の動作を行う機能が含まれています。

- プログラムを実行する人物の特権を、プログラムの作成者に渡す
- システムへの不正アクセスを許可する
- システム・ファイルの保護を変更する
- システムにパッチを適用する（オペレーティング・システムに特別なソフトウェアを追加する）
- 簡単に推測できるパスワードを検索するジョブを作成する

このタイプの侵入からシステムを守るには、必ず信頼できる販売元からソフトウェアを購入します。新しいユーザのトレーニングの際には、出所が定かでないソフトウェアの使用を避けることの重要性を強調します。

プログラムとディレクトリに対するもう1つの危険は、ウイルスと呼ばれるプログラムです。トロイの木馬のソフトウェアは、悪意のないユーザがトロイの木馬とは知らずにそのソフトウェアを使用することを利用するのに対して、ウイルスはユーザの協力を必要としません。ウイルスはファイル保護の欠陥を利用するプログラムで、システムに侵入し、コマンド・プロシージャと実行可能プログラムに変更を加えます。コマンド・プロシージャに変更を加えることで、ユーザのアクセス権と特権を利用して増殖できるようになります。

ウイルスは、パーソナル・コンピュータの環境と比較すると、OpenVMS 環境ではあまり大きな問題にはなりません。OpenVMS の保護機能と、環境の規模の大きさと多様性により、ウイルス攻撃が困難になっているためです。しかし、ソフトウェアとデータの共有が可能な環境の中で、ウイルス攻撃から安全な環境は存在しません。

このタイプのセキュリティ侵害の主なターゲットは、ユーザのログイン・コマンド・プロシージャです。一般的にログイン・コマンド・プロシージャには、定期的に行われ、簡単に変更が可能な DCL コマンドが含まれています。

ACL もターゲットになります。ユーザがアクセス特権を共有する設計になっているファイル保護では、このタイプのプログラムが、多数のユーザのプログラムを介して実行され、その過程で新しい特権が獲得される可能性があります。

このタイプのセキュリティ侵害からシステムを保護するには、ファイル保護を適切に設計することが非常に重要です。ターゲットになりやすいオブジェクトは、ユーザが変更できないようにします。たとえば、ログイン・コマンド・プロシージャが許可するのは、最大でもほかのユーザへの読み取りアクセス権までとなるようにファイル保護を設定します。また、ログイン・コマンド・プロシージャが含まれているディレクトリに対する書き込みアクセス権は、システム・カテゴリと所有者カテゴリのユーザにのみ許可するようにします。

損害の多くは、このようなプログラムが特権を持つターゲットのアカウントに到達すると発生するため、特権を持つユーザは、特にルート・ディレクトリ、実行可能ファイル、およびコマンド・プロシージャを慎重に保護する必要があります。トロ

イの木馬の攻撃を抑止するには、ユーザは、コマンド・プロシージャやイメージのソースを調べずに、特権アカウントでコマンド・プロシージャやイメージを実行すべきではありません。アプリケーション・イメージは、バイナリ・イメージが対応するソースを確実に反映するように、ソースからリビルドする必要があります。

#### 8.9.1.2 特権を与えてのプログラムのインストール

一部のソフトウェアは、実行に特権が必要です。ソフトウェアを実行する必要があると想定される全ユーザに対して特権を拡張したり、必要な特権を与えてプログラムをインストールすることができます。特権付きのソフトウェアをインストールすると、ユーザ個人が必要な特権を所有しているかどうかに関係なく、ユーザにソフトウェアの実行を許可することになります。結果として、プロセスがソフトウェアを実行する間、プロセスの特権が拡張されます。この方法にはいくつかのメリットがありますが、セキュリティに関わる危険性もあります。8.7 節では、これらの選択肢についてさらに詳しく説明しています。

#### 8.9.2 システム・ファイルの保護

最も開放的なシステムであっても、システム・ソフトウェアの保護は必要です。通常、HP は適切な UIC 保護を設定した状態で、システム・プログラムおよびデータベースを出荷しています。しかし、何らかの理由でデフォルトの保護が不十分である場合、必要な SYSPRV 特権を持っていれば、第 4 章で概要を説明したテクニックを使用して、デフォルトの保護を変更できます。また、追加の保護が必要であると判断されるファイルには、ACL も追加できます。

OpenVMS のインストール中に次の DCL コマンドを使用することで、システム管理者のアカウントから、システム・ファイルの完全なリストを取得できます。

```
$ DIRECTORY/SECURITY/OUTPUT=SYSTEM_FILES.LIS SYS$SYSROOT:[*...]
```

このようなリストを作成し、参照用に保存しておくことをお勧めします。定期的にこれらの値を現在のシステム・ファイルの保護と比較して、改ざんがないことを確認します。DCL の DIRECTORY/SECURITY/OUTPUT コマンドと DIFFERENCES を使用すると、このようなチェックが簡単に行えます。

Alpha システムでは、読み取り専用の CD 配布メディアから、システム・ファイルとその保護のリストを入手できます。OpenVMS ソフトウェアでは、インストールが正常に行われれば、この保護コードのセットが得られるはずです。

VAX システムでのシステム・ファイルとその保護のリストについては、付録 B を参照してください。OpenVMS ソフトウェアでは、インストールが正常に行われれば、この保護コードのセットが得られるはずです。

表 8-4 に、ファイル保護の設定と表示に使用する DCL コマンドの要約を示します。これらのコマンドについては、『*OpenVMS DCL* デictionary』で説明しています。



表 8-4: ファイル保護に使用する DCL コマンド

コマンド	機能
DIRECTORY/ACL	ファイルの ACL を表示します
DIRECTORY/OWNER	ファイル所有者の UIC を表示します
DIRECTORY/PROTECTION	ファイルの保護コードを表示します
DIRECTORY/SECURITY	DIRECTORY/ACL, DIRECTORY/OWNER, および DIRECTORY/PROTECTION によって生成されるファイル情報を結合して表示します
EDIT/ACL	アクセス制御リスト・エディタ (ACL エディタ) を起動します
SET PROTECTION/DEFAULT	以降作成される全ファイルに適用されるデフォルトの保護を設定します
SET SECURITY	任意のオブジェクト (所有者, 保護コード, および ACL) のセキュリティ・プロファイルを変更します
SHOW SECURITY	保護オブジェクトの所有権, UIC 保護コード, および ACL を表示します

OpenVMS のインストール手順では, 当初特権を与えずに MAIL.EXE をインストールします (MAIL.EXE は, その機能の実行に特権が必要ないためです)。OpenVMS オペレーティング・システムの以前のバージョンには, MAIL.EXE の再インストール時にシステム管理者が割り当てることのある一部の特権を MAIL.EXE がチェック, 無視, 付与, またはオーバーライドできるようにするメカニズムが含まれていました。これらの規制メカニズムは, 予期しない状態や望ましくない状態を生み出すことがあったため, 削除されました。

#### 注意

特定の特権を与えて MAIL.EXE を再インストールする場合は, セキュリティ侵害の可能性を含め, 発生しうる影響を注意深く検討する必要があります。たとえば, MAIL.EXE は Mail ユーティリティを起動するすべてのユーザにその特権を付与するため, 該当ユーザは, SPAWN コマンドを指定して Mail 内からサブプロセスを作成すると, これらの特権を継承することになります。

すでに述べたように, HP はシステム・プログラムにデフォルトの保護を提供しています。しかし, 特別な必要がある場合は, 要件を満たすために ACL の能力を検討します。たとえば, ACL を使用して, コンパイラなどのシステム・プログラムの

使用を制限できます。このような措置が必要になる要因として、パフォーマンスからライセンスの問題に至るまで、さまざまな要因が考えられます。

一部またはすべてのユーザがメディアを初期化できると不適切なケースがあるかどうかを考慮する必要もあります。そのようなケースがある場合は、システム・プログラム SYS\$SYSTEM:INIT.EXE に対して ACL を適用できます。UIC ベースの保護コードにおいて、ワールド・カテゴリにアクセス権を付与しないようにします。その後、ファイルを対象に、特定のユーザにアクセス権を付与する ACL を作成します。

同様に、会社の特定の部署がソフトウェア製品に対してライセンス料を支払った場合は、その部署に限定してソフトウェアを使用可能にして、他の部署には使用できないようにすることができます。ワールド・カテゴリに、標準の UIC ベースの保護コードによってアクセス権が付与されていないことを確認し、ファイルの ACL に部署の識別子を介してアクセスを許可するエントリを作成します。

ACL 保護は、一部のユーザや保護サブシステムにアクセス権を限定するなど、アプリケーション・データベースの保護にも必要な場合があります。

### 8.9.3 DCL コマンドの使用の制限

ユーザによる DCL コマンドの使用を制御するには、いくつかの方法があります。たとえば、以下の方法があります。

- SYS\$SYSROOT:[SYSEXEC] ディレクトリおよび SYS\$SYSROOT:[SYSLIB] ディレクトリのシステム・プログラム・ファイルに、ACL を適用します。
- AUTHORIZE の DISIMAGE フラグを設定することで、MCR コマンドまたは RUN コマンドの使用を禁止します。これによりユーザは、システム・イメージやユーザ記述イメージの実行、または外部コマンドとして定義されているイメージの実行を禁止されます。

DISIMAGE フラグは DCL コマンド言語インタプリタ (CLI) により適用されるため、DISIMAGE フラグの設定対象のアカウントは、DCL CLI にのみアクセスできることを確認する必要があります。DISIMAGE フラグは、AUTHORIZE の DEFCLI フラグと組み合わせて使用するか、制限付きアカウント内で使用します。アカウントに RESTRICTED フラグを設定すると、DEFCLI フラグが暗黙に設定されます。

- DCL コマンドの定義を削除または変更し、DCL テーブルをリビルドします。コマンド定義の作成方法については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』で説明しています。変更したテーブルを指定するには、ユーザの UAF レコードで /CLITABLES 修飾子を使用します。また、ユーザが、指定したコマンド言語インタプリタ (CLI) とテーブルにのみログインできるように、/FLAGS=DEFCLI も指定します。元の DCL テーブルを不正アクセスから守るには、SYS\$SYSROOT:[SYSEXEC] ディレクトリおよび SYS\$SYSROOT:[SYSLIB] ディレクトリのシステム・プログラム・ファイ

ルに、ACL を適用します。特に、SYS\$LIBRARY:DCLTABLES.EXE および SYS\$SYSTEM:CDU.EXE を保護します。

#### 8.9.4 ファイルの暗号化

ファイルの暗号化とは、データの内容を隠すために特定のアルゴリズムをデータに適用する処理を指します。復号は、逆の処理を行い、暗号化されている情報を変換して元の内容に戻します。自社開発ソフトウェアをメディアにコピーし、別のサイトに移動する必要がある場合には、ファイルを暗号化するとよいでしょう。メディア上のソフトウェアは、正しい復号コードがなければ利用できません。

ソフトウェアとハードウェアの両方で、さまざまなファイル暗号化システムが使用できます。お客様の国で利用できる製品の詳細については、HP のサポート・チャネルにお問い合わせください。

#### 8.9.5 ディスクの保護

ディスク・スカベンジングとは、ページまたは削除操作に続いて行われるファイル・ヘッダの削除の後で、データの磁気的な痕跡を読み取る処理です。ユーザがシステムからファイルを削除すると、ファイル・ヘッダのみが削除されます。データが上書きされるまで、そのデータはディスク・スカベンジングのターゲットになる可能性があります。セキュリティの要件が中または高であるサイトは、この行為を考慮する必要があります。

全体のセキュリティ機能確立した後、UIC ベースのボリューム保護を使用することで、重要な情報が格納されているディスクへのアクセスを制限します。ディスク・スカベンジングはしばしば権限を持つユーザによって実行されるため、次の各節で説明する除去パターンとハイウォータ・マーク処理の実装を検討します。

##### 8.9.5.1 除去テクニック

ディスク除去の実装には、いくつかの方法があります。

- DELETE コマンドまたは PURGE コマンドに /ERASE 修飾子を追加すると、ユーザがファイルを削除またはページする際に、システムによってそのファイル位置の全体がゼロの除去パターンで上書きされます。この修飾子を自発的に使用するようユーザを促すことも、システム・ログイン・コマンド・プロシージャ (通常は SYS\$MANAGER:SYLOGIN.COM) に次のコマンドの定義を追加して自動的に修飾子が追加されるようにもできます。

```
DEL*ETE ::= "DELETE/ERASE" PUR*GE ::= "PURGE/ERASE"
```

ただし、どのユーザも DELETE コマンドまたは PURGE コマンドに /NOERASE 修飾子を追加することで、これらの定義を迂回できます。

- 削除時除去を確実に行うには、DCL の SET VOLUME/ERASE\_ON\_DELETE コマンドを使用して、ボリューム全体を対象にこの機能を有効にします。ファ

イルが削除されると、このコマンドにより、ボリューム上のどのファイルでもゼロの除去パターンによって上書きされます。

- ボリュームの初期化時に、ボリュームを完全に除去し、ボリュームで削除時除去を有効にするには、DCL の INITIALIZE/ERASE コマンドを使用します。

デフォルトでは、削除時除去が有効である場合、オペレーティング・システムは、領域に対する単一の上書き操作時に適用されるデフォルトのゼロによるデータ・セキュリティ除去 (DSE) パターンを書き込みます。デフォルトのパターンであるゼロや (複数回の除去ではなく) 1 回の除去では要件に適さないと思われる場合は、\$ERAPAT (Get Security Erase Pattern) システム・サービスを使用して、カスタマイズされた除去パターンを書き込むことができます。詳細については、『*HP OpenVMS System Services Reference Manual*』にある \$ERAPAT の説明を参照してください。

セキュリティ要件のレベルが高いサイトでは、固定パターンよりもランダム・パターンの方が適しています。わずかに残る磁気の痕跡を検出して使用する技術が、すでに利用できるようになっているためです。そのため、ディスクが取り外され、このような特別な分析装置を使用して読み取られる危険性が十分あると結論付けられる場合は、除去パターンを複数回再書き込まなければならない場合があります。データ・セキュリティ除去パターンをニーズに合うようカスタマイズする方法については、SYS\$EXAMPLES:DOD\_ERAPAT.MAR ファイルに記載されている情報から学習できます。

除去パターンは、セキュリティ要件が最も厳しいディスクにのみ使用します。除去は時間を要し、システムのパフォーマンスに影響するためです。

#### 8.9.5.2 ハイウォーター・マーク処理による防止

ハイウォーター・マーク処理とは、各ファイルが書き込まれた位置の上限を追跡し、ユーザがその地点を超えてデータを読み取ろうとするのを禁止するテクニックを指します。

オペレーティング・システムは、さまざまなテキスト・エディタ、コンパイラ、およびリンカから出力されるファイル (つまり、プロセスが書き込む大部分のファイル) のセットをはじめとするすべての順編成、排他的アクセス・ファイルに対し、真のハイウォーター・マーク処理を実装します。ファイル・ヘッダにあるハイウォーター・マークは、ファイルの論理的な終端マークが更新された時点 (通常はファイルが閉じられる時点) で更新されます。

共有ファイル (索引編成と順編成の両方) では、オペレーティング・システムは割り当て時除去の原則を使用して、真のハイウォーター・マーク処理に近い結果を実現します。ファイルが作成または拡張されようとする時点で、システムはどれだけのディスク領域 (ファイルの範囲) が必要であるかを判断し、書き込み用に割り当てられる領域 (範囲) に、ゼロのセキュリティ除去パターンを適用します。その後、

ファイルは、そのファイルのために除去された領域に書き込まれます。したがって、ユーザが(範囲全体を含め)該当ファイルへのアクセス権を取得し、ファイルが書き込まれた領域を超える領域を読み込もうとしても、読み込めるのはデータ・セキュリティ除去パターンのみです。

デフォルトでは、オペレーティング・システムはすべてのボリュームに対してハイウォーター・マーク処理を有効にします。ハイウォーター・マーク処理は、ディスク・スキャベンジングの試みに対する抑止力になります。ただし、ハイウォーター・マーク処理は余分に入出力が必要となるため、システム・パフォーマンスに影響を与えます。

DCL の SET VOLUME/NOHIGHWATER\_MARKING コマンドを指定することで、ボリュームごとにハイウォーター・マーク処理と割り当て時除去を無効にすることができます。

### 8.9.5.3 防止テクニックの要約

セキュリティ管理者は、次の制御を適用することによって、ディスク・スキャベンジングを阻止することができます。

- 厳重な物理的セキュリティを適用します。最も重要な情報が格納されているディスクに関しては特に厳重にします。
- UIC ベースの保護を利用して、厳重なボリューム保護を適用します。
- ユーザの自発行為またはボリュームに対する強制適用によって、重要なファイルのパージまたは削除時の /ERASE 修飾子の使用を促します。
- 最も重要なディスクでは、デフォルトでハイウォーター・マーク処理を行うようにします。

### 8.9.6 バックアップ・メディアの保護

ファイル、ディレクトリ、およびディスクのコピーを作成することで、データを喪失や破損から守ることができます。問題が発生した場合は、バックアップ・コピーを復元して、作業を継続することができます。メディアの安全な保管と、メディアへのアクセスの管理は、このプロセスの重要な要素です。バックアップ・メディアは、対象サイト以外の場所に保管するのが理想的です。

#### 8.9.6.1 ディスクのバックアップ

効果的なバックアップ・スケジュールを立てることが、データの保護にとって非常に重要です。バックアップを定期的に行うことで、ファイルの誤削除や破損による喪失を防ぐことができます。

バックアップの実施とバックアップ・スケジュールの設定の詳細については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照し

てください。バックアップ・ユーティリティ (BACKUP) ユーティリティは、セキュリティ・ポリシーを実装しないことに注意してください。セキュリティ管理者がバックアップ・ユーティリティに明示的に指示する必要があります。バックアップ・ユーティリティは、オペレータのセキュリティ・プロファイルを使用して実行されます。多くの場合、そのセキュリティ・プロファイルは特権付きです。

#### 8.9.6.2 バックアップ・セーブ・セットの保護

バックアップ・セーブ・セットへのアクセスの制限は、システム・セキュリティの重要な要素です。ファイル・システムは、バックアップ・セーブ・セットがディスクまたは磁気テープのどちらに保存されていても、バックアップ・セーブ・セットを単一のファイルとして扱います。したがって、セーブ・セットにアクセスできるユーザであれば誰でもセーブ・セットの任意のファイルを読み込むことができます。BACKUP は、個別ファイルの保護をチェックしません。

システム・セキュリティを維持するには、セーブ・セットを適切に保護することが非常に重要になります。出力セーブ・セット修飾子の /BY\_OWNER および /PROTECTION を使用して、ディスク上のセーブ・セットと、磁気テープ・ボリュームに、制限付きの保護を割り当てます。保護が十分であれば、非特権ユーザがセーブ・セット・ボリュームをマウントしたり、セーブ・セットからファイルを読み込むことを防止できます。施錠されたキャビネットにバックアップ・メディアを保管することで、オフ・ラインで保管されるセーブ・セットについても、物理的なセキュリティ対策を講じる必要があります。

セーブ・セットを Files--11 ディスクや順編成ディスクに書き込み、/PROTECTION 修飾子は指定しない場合、BACKUP は、セーブ・セットにデフォルトのプロセス保護を適用します。/PROTECTION を指定した場合、未指定の保護カテゴリのデフォルトは、すべてデフォルトのプロセス保護になります。

保護情報は、磁気テープのボリューム・ヘッダ・レコードに書き込まれ、そのテープに保存されるすべてのセーブ・セットに適用されます。そのため、出力セーブ・セット修飾子 /BY\_OWNER および /PROTECTION が磁気テープ・セーブ・セットで有効であるのは、出力セーブ・セット修飾子 /REWIND を指定した場合のみです。この修飾子によりテープは、先頭への巻き戻し、ボリューム・ヘッダ・レコードへの保護データの書き込み、テープの初期化を行うことができます。/PROTECTION を指定した場合は、未指定の保護カテゴリのデフォルトは、すべてデフォルトのプロセス保護になります。/PROTECTION および /BY\_OWNER 修飾子と合わせて /REWIND を指定しなかった場合、磁気テープは既存の保護を保持します。ただし、/REWIND のみを指定すると、保護がまったくない磁気テープになってしまいます。

次の例に、ディレクトリをテープにバックアップする方法を示します。

```
$ BACKUP
 _FROM:  [PAYROLL]
```

```
_TO: MFA2:KNOX.BCK/LABEL=BANK01 - _$ /REWIND/BY_OWNER_UIC=[030,003] -  
_$ /TAPE_EXPIRATION=15-JAN-2001 - _$ /PROTECTION=(S:RWE,O:RWED,G:RE,W)
```

1. ディレクトリ [PAYROLL] の内容が、磁気テープ・ドライブ MFA2 のファイル KNOX.BCK にコピーされます。出力セーブ・セット修飾子 /LABEL は、そのテープのラベル BANK01 を指定しています。
2. 出力セーブ・セット修飾子 /BY\_OWNER は、セーブ・セットに [030,003] という所有者 UIC を割り当てています。
3. 出力セーブ・セット修飾子 /TAPE\_EXPIRATION は、テープに 2001 年 1 月 15 日の期限を割り当てています。
4. 出力セーブ・セット修飾子 /PROTECTION は、ボリュームの所有者に、読み込みアクセス権、書き込みアクセス権、実行アクセス権、および削除アクセス権を割り当てています。システム・ユーザには読み込みアクセス権、書き込みアクセス権、実行アクセス権が割り当てられ、グループ・ユーザには読み込みアクセス権と実行アクセス権が割り当てられ、ワールド・ユーザにアクセス権は割り当てられていません。

#### 8.9.6.3 バックアップ・セーブ・セットからのファイルの取り出し

セーブ・セットにアクセスできるユーザであれば誰でもセーブ・セットの任意のファイルを読み込むことができます。バックアップ・メディアのコピーをユーザには絶対に渡さないでください。悪意のあるユーザがテープやディスクからファイルを復元し、システムのセキュリティを危険にさらす可能性があるためです。

非特権ユーザが特定のファイルを復元する必要がある場合も、セーブ・セットが格納されているボリュームを貸さないでください。ボリューム上の全ファイルへのアクセス権を渡してしまう可能性があるためです。特定のファイルを復元する最も安全な方法は、次の例のように、個別にファイルを復元する方法です。

```
$ BACKUP MTA0:JULY.BCK/SELECT=[JONES.TEXTPROC]LASTMONTH.DAT -  
_$ [*...]/BY_OWNER=ORIGINAL
```

選択されたファイルは、元のディレクトリ、所有権、および保護とともに復元されます。この方法では、ファイル・システムによって、ユーザにファイル・アクセスが許可されているかどうか判定されます。

### 8.9.7 ターミナルの保護

以降の節では、ターミナルの使用制限に利用できる制御を説明します。

#### 8.9.7.1 ターミナルの使用制限

オペレーティング・システムは、デバイス・オブジェクト・クラス・テンプレート TERMINAL を使用して、SYSTEM アカウントのみがアクセスできるようにターミ

ナルを設定します。ユーザがログインすると、オペレーティング・システムは、所有権をシステム UIC から現在のプロセスの UIC に移します。

特定のターミナルへのログインを制限するには、次の方法があります。

- システム・パスワードを割り当てます。
- ターミナルを /NOTYPE\_AHEAD に設定し、ログインを不可能にします。

システム・パスワードを適用することで、対象ターミナルの使用がシステム・パスワードを知っているユーザに限定されます。

#### 8.9.7.2 アプリケーション・ターミナルなどのデバイスの制限

ターミナルを、アプリケーション・ターミナルとして一部のユーザがアクセスできるようにするには、対象デバイスのセキュリティ特性の一部または全部を変更します。(適切な保護コードを持つ) 特定のターミナルを対象に、コマンド・プロシージャ SYS\$MANAGER:SYSTARTUP\_VMS.COM に、DCL の SET SECURITY/CLASS=DEVICE コマンドを含めることができます。この DCL コマンドは、ファイル構造ではないすべてのデバイスへのアクセスを制限できます。また、デバイスに ACL を適用して、ユーザ・アクセスを制限することもできます。

#### 8.9.7.3 モデム用のターミナル回線の設定

モデム用のターミナル回線を設定する場合、会話型の使用が想定されているモデム装置が接続されている回線では、/COMMSYNC 修飾子を、DCL の SET TERMINAL コマンド (または TTDRIVER インタフェースの TT\$M\_COMMSYNC 属性) に対して絶対に設定しないでください。

この修飾子は、モデム電話回線に障害が発生した場合にターミナル回線からユーザ・プロセスを切断する、モデム・ターミナル属性を無効にします。/COMMSYNCH 修飾子が有効になっていると、ターミナル回線に対する次の着信が、前のユーザのプロセスに接続される可能性があります。/COMMSYNC 修飾子は、モデム信号をフロー制御として使用することにより、非同期プリンタなどのデバイスのターミナル・ポートへの接続を可能にすることを目的としています。



## セキュリティ監査の実施

この章では、OpenVMS の監査システムを使用および管理する方法を説明します。システムで発生するイベントを記録し、後でその監査ログを分析することにより、システムのセキュリティに関わる活動を監視する方法について説明します。

### 9.1 監査プロセスの概要

監査とは、システムで発生するセキュリティに関わる活動を記録し、後でこの監査ログを分析することを指します。監査により、システム上でのユーザの活動を監視し、必要に応じて、システムのセキュリティを危険にさらす試みに至るまでの一連のイベントを再現できます。したがって、システムとそのデータを保護する手法というよりも、むしろシステムの使用状況を分析および記録する手法です。

システムまたはシステム内の保護オブジェクトへのユーザ・アクセスに関係することはすべて、セキュリティに関わる活動と見なされます。このような活動は、イベントと呼ばれます。一般的なイベントには、次のものが含まれます。

- ログイン、ログアウト、またはログインの失敗
- 登録データベースにおける変更
- ファイル、デバイス、グローバル・セクションなどの保護オブジェクトへのアクセス
- 保護オブジェクトの特権またはセキュリティ属性における変更

オペレーティング・システムは、成功と失敗の両方のイベントを記録できます。場合によっては、失敗の方が大きな意味を持つことがあります。たとえば、プログラマがアクセス権を持っているファイルを表示したことを記録するよりも、同じプログラマが保護ファイルを表示しようとして阻止されたことを記録する方が重要です。

イベント・メッセージ自体は、監査ログ・ファイルと、セキュリティ・クラス・メッセージの受信が可能になっているオペレータ・ターミナルの、2つの場所に書き込むことができます。例 9-1 に示すように、メッセージには次のデータが含まれています。

1. メッセージの日付と時刻
2. イベントのタイプ
3. イベントが発生した日付と時刻

#### 4. イベントを引き起こしたユーザのプロセス識別番号 (PID)

監査メッセージのそれ以外の情報は、そのイベントのタイプに固有の情報です。メッセージのさまざまな例については、付録 D を参照してください。

#### 例 9-1: アラーム・メッセージのサンプル

---

```
%%%%%%%% OPCOM 25-JUL-2001 16:07:09.20 %%%%%%%%%
Message from user AUDIT$SERVER on GILMORE
Security alarm (SECURITY) on GILMORE, system id: 20300
Auditable event:          Process suspended ($SUSPND)
Event time:              25-JUL-2001 16:07:08.77
PID:                    30C00119
Process name:           Hobbit
Username:               HUBERT
Process owner:          [LEGAL,HUBERT]
Terminal name:          RTA1:
Image name:             $99$DUA0:[SYS0.SYSCOMMON.][SYSEXE]SET.EXE
Status:                 %SYSTEM-S-NORMAL, normal successful completion
Target PID:             30C00126
Target process name:     SMISERVER
Target username:         SYSTEM
Target process owner:    [SYSTEM]
```

---

## 9.2 セキュリティ関連イベントの報告

デフォルトで決まっている報告内容 (表 9-1 を参照) 以外にセキュリティ管理者が受け取るセキュリティ・イベント情報の種類は、発生しうるイベントの長いリストからセキュリティ管理者が選択した情報の種類によって決まります。この節では、セキュリティ・イベント情報の報告を有効にする方法を説明します。具体的には、次のトピックについて説明します。

- イベント・メッセージの生成方法
- システムが報告できるイベントのタイプ
- イベント情報のソース

### 9.2.1 監査情報の生成方法

システムのインストールまたはアップグレード時には、OpenVMS オペレーティング・システムによって少数のイベントが自動的に監査されます。表 9-1 に示すこれらのイベントのカテゴリは、システムのセキュリティの大幅な変更を意味します。サイトの要件に応じて、他の形式の報告を有効にすることができます。

セキュリティに関わる活動に関してオペレーティング・システムに報告させる方法は 3 つあります。

- 監査のためのイベントのカテゴリを有効にする方法。たとえば、すべてのログイン失敗や、システム・パラメータに対するすべての変更を報告できます。
- 保護オブジェクトにアクセス制御エントリ (ACE) を関連付ける方法。たとえば、特定のファイルがユーザによって変更されるたびにメッセージが生成されるようにできます。
- ユーザの登録レコードを変更して、当該アカウントから実行されるすべての操作をシステムが監査するようにする方法。

#### 9.2.1.1 活動の監査のカテゴリ

セキュリティ関連イベントは、イベント・クラスと呼ばれるいくつかのカテゴリに分けられています。オペレーティング・システムは、いくつかのイベント・クラスをデフォルトで監査します (表 9-1 を参照)。サイトのセキュリティ要件により追加の監視が必要である場合は、DCL の SET AUDIT コマンドを使用して、追加のイベント・クラスに関するセキュリティ監査を有効にします。

各種イベント・クラスの監査を有効にするには、次のコマンド形式を使用します。

```
SET AUDIT /ENABLE=event-class[...] {/ALARM | /AUDIT}
```

イベントを有効にするには、コマンドには次の 2 つの修飾子が必要です。

- /ENABLE 修飾子は、監査対象のイベント・クラスを定義します。イベント・クラスの一覧については、表 9-3 を参照してください。
- /AUDIT 修飾子または /ALARM 修飾子は、イベント・メッセージの出力先を定義します。

/AUDIT 修飾子は監査ログ・ファイルにメッセージを出力するように指示します。また /ALARM 修飾子は、セキュリティ・イベント・メッセージの受信が可能になっているオペレータ・ターミナルに、メッセージを出力するように指示します。重大なイベントは、監査とアラームの両方として報告する必要がありますが、重要度の低いイベントは、後で調査できるようにログ・ファイルに書き込むことができます。表 9-1 に示すデフォルトのイベント・クラスは、アラームと監査の両方として監査が実施されます。

新しいイベントを有効にすると、オペレーティング・システムは直ちにクラスタの全ノードで新しいイベントの監査を開始します。監査は、/DISABLE 修飾子を使用して明示的にクラスを無効にするまで継続されます。現在の監査設定は SYS\$MANAGER:VMS\$AUDIT\_SERVER.DAT に記録されるため、システムのブートをまたいで設定が維持されます。

SET AUDIT コマンドの詳細については、『*OpenVMS DCL* デictionary』を参照してください。

表 9-1: デフォルトで監査されるイベント・クラス

クラス	説明
ACL	セキュリティ監査 ACE を保持する任意のオブジェクトへのアクセス
Audit	SET AUDIT コマンドのすべての使用。このカテゴリは無効にできません。
Authorization	次の登録データベースに対するすべての変更。 <ul style="list-style-type: none"> <li>システム・ユーザ登録ファイル (SYSUAF.DAT)</li> <li>ネットワーク代理登録ファイル (NETPROXY.DAT または NET\$PROXY.DAT)</li> <li>ライト・データベース (RIGHTSLIST.DAT)</li> </ul>
Break-in	すべての侵入行為 (パッチ, 独立, ダイアルアップ, ローカル, ネットワーク, 遠隔)。
Logfailure	すべてのログイン失敗 (パッチ, ダイアルアップ, ローカル, 遠隔, ネットワーク, サブプロセス, 独立, サーバ)。

サイトで現在監査対象になっているイベント・クラスを表示するには、DCL の SHOW AUDIT コマンドを入力します。例 9-3 に、セキュリティ要件が中程度であるサイトの監査設定を示します。

#### イベント・クラスの有効化の例

セキュリティに関わる活動の全クラスの監査を有効にすることは可能ですが (/ENABLE=ALL)、このような方法では大量の監査メッセージが発生し、あまりに多くの情報が生成されるため、有効な分析ができなくなります。そのため、9.3.1 項で説明しているように必要性を評価し、システムの活動を選択的に監査することをお勧めします。

イベント・クラスの監査は、さまざまなレベルの精度で行えます。次のような方法があります。

- クラスの有効化

たとえば、すべてのログイン失敗の監査を有効にするには、次のコマンドを入力して、logfailure クラスを有効にします。

```
$ SET AUDIT/AUDIT/ENABLE=LOGFAILURE=ALL
```

このコマンドの結果、監査サーバはすべてのログイン失敗をセキュリティ監査ログ・ファイルに報告します。

- クラスのサブセットの有効化

一部のイベントについて、有効にする報告の種類をより細かく選択できます。たとえば、すべてのログイン・イベントを有効にするよりも、ネットワーク・ログイン・イベントと遠隔ログイン・イベントを有効にする方が合理的です。

ネットワーク・ログインと遠隔ログインのみの監査を有効にするには、次のコマンドを入力します。

```
$ SET AUDIT/AUDIT/ENABLE=LOGIN=(NETWORK,REMOTE)
```

- 成功イベント，失敗イベント，または特権イベントの有効化

失敗イベントのレポート，または特定の特権のもとで行われた活動の報告のみを有効にすることで，システムの通常の使用を報告するイベント・メッセージを簡単に除外できます。

特に，保護オブジェクトに対するアクセス・イベントを監査する場合は，ログインやインストール・ユーティリティの使用などのイベント・クラスの場合よりも，情報の要件をより細かく定義する必要があります。ファイルや，その他の一部の保護オブジェクトは頻繁にアクセスされるため，関連するアクセス・イベント・クラスをすべて有効にすると，膨大な数のイベント・メッセージが生成され，実際に調査を必要とする異常イベントが見失われる可能性があります。このため，失敗イベントや特権アクセス・イベントなどの例外的な状況についてのみアクセス監査を有効にすることをお勧めします。

ファイル・アクセス失敗イベントの監査を有効にするには，次のコマンドを入力します。

```
$ SET AUDIT/AUDIT/ENABLE=ACCESS=FAILURE/CLASS=FILE
```

このコマンドは，読み込みアクセスまたは書き込みアクセスの失敗だけでなく，すべてのファイル・アクセスの失敗の監査を有効にすることに注意してください。単純な書き込み操作と思われる操作も，複数のタイプのアクセスが関与するなど（たとえばファイルへの書き込みの前に，ディレクトリ内のファイルへのアクセス権だけでなく，ボリュームへのアクセス権と，ディレクトリへの読み込みアクセス権が必要であるなど），アクセス操作はかなり複雑な場合があるため，このような設定をお勧めします。

例 9-2 に，ファイル・アクセスの失敗によるイベント・メッセージを示します。ユーザ Robinson がファイル FOO.BAR を削除しようとしたましたが，そのファイルの ACE により阻止されました。メッセージから，Robinson が識別子 MINDCRIME を保持しており，FOO.BAR の識別子用 ACE が，当該識別子を保持する人物にアクセスを許可していないことがわかります。さらに，システムがこのファイルを所有しているため，Robinson は保護コードによりこのファイルの削除アクセス権も取得できません。

#### 例 9-2: オブジェクト・アクセス・イベントにより作成される監査

---

```
Message from user AUDIT$SERVER on BILBO
Security alarm (SECURITY) and security audit (SECURITY) on BILBO,
                                system id: 19662
Auditable event:                Object deletion
Event information:              file deletion request (IO$_DELETE)
Event time:                    24-APR-2001 13:17:24.59
```

## 例 9-2: オブジェクト・アクセス・イベントにより作成される監査 (続き)

---

```
PID: 47400085
Process name: Hobbit
Username: ROBINSON
Process owner: [ACCOUNTING,ROBINSON]
Terminal name: OPA0:
Image name: DSA2264:[SYS51.SYSCOMMON.][SYSEXEC]DELETE.EXE
Object class name: FILE
Object owner: [SYSTEM]
Object protection: SYSTEM:RWED, OWNER:RWED, GROUP:RE, WORLD:RE
File name: _DSA2200:[ROBINSON]FOO.BAR;1
File ID: (17481,6299,1)
Access requested: DELETE
Matching ACE: (IDENTIFIER=MINDCRIME,ACCESS=NONE)
Sequence key: 00008A41
Status: %SYSTEM-F-NOPRIV, no privilege for attempted operation
```

---

### 9.2.1.2 セキュリティ監査 ACE の関連付け

9.2.1.1 項の説明にあるように、保護オブジェクトへのアクセスに関するイベントは頻繁に発生するため、その監査については慎重に検討する必要があります。イベント・メッセージが多すぎると、その量に圧倒され、実際に調査を必要とする異常イベントが見失われる可能性があります。

保護オブジェクトより細かく監査する方法としては、オブジェクトのアクセス制御リスト (ACL) に監査 ACE を追加し、ACL イベント・クラスを有効にするという方法があります。このアプローチでは、クラスに属する全オブジェクトではなく、セキュリティ監査 ACE を持つオブジェクトへのアクセスのみがイベント・メッセージを生成します。

イベントの報告先に応じて、2 種類の監査 ACE を使用できます。アラーム ACE は、イベント・メッセージをオペレータ・ターミナルに出力し、監査 ACE は、イベント・メッセージを監査ログ・ファイルに出力します。表 9-2 に、監査 ACE の概要を示します。また『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』では、監査 ACE の詳細を説明しています。監査 ACE の対象になっているシステム・ファイルの一覧については、表 10-1 を参照してください。

表 9-2: セキュリティ監査用のアクセス制御エントリ (ACE)

ACE のタイプ	説明
アラーム ACE	指定の方法でオブジェクトがアクセスされるたびに、オペレータ・ターミナルにイベント・メッセージを書き込みます。構文は次のとおりです。  (ALARM=SECURITY[,OPTIONS=options],ACCESS=access-type[+access-type...])
監査 ACE	指定の方法でオブジェクトがアクセスされるたびに、セキュリティ監査ログ・ファイルにイベント・メッセージを書き込みます。構文は次のとおりです。  (AUDIT=SECURITY[,OPTIONS=options],ACCESS=access-type[+access-type...])

保護したいオブジェクトに ACE を関連付けるには、DCL の SET SECURITY/ACL コマンドを使用するか、アクセス制御リスト・エディタ (ACL エディタ) を使用します。監査 ACE の ACCESS 文には、必ず SUCCESS キーワードと FAILURE キーワードの一方 (または両方) を追加します。

自動ログイン・ファイルの SYSALF.DAT、オペレータ・ログ・ファイルの OPERATOR.LOG、システム会計ファイルの ACCOUNTING.DAT など、自動では監査されない重要なシステム・ファイルに対しては、監査 ACE を定義することをお勧めします。ただし、大部分が有用でない大量のメッセージが生成されるため、アクセスのすべての状況の監視は行わないでください。たとえば、OPERATOR.LOG への正常な書き込み操作を追跡しても、重要な情報は得られませんが、書き込みの試みの失敗を追跡すれば、重要な情報が得られる可能性があります。

監査対象のオブジェクトとしてはファイルが最も一般的ですが、任意のオブジェクトに監査 ACE を追加できます。機密文書を扱うプリント・キューや、パスワード・グラバの試みを検出するためにターミナルに、監査 ACE を追加できます。3.8 節「パスワードの保護に関するガイドライン」を参照してください。

#### 監査 ACE の追加例

ACCOUNTING.DAT ファイルのアラーム ACE を設定するには、次のコマンドを入力します。

```
$ SET
SECURITY/ACL=(ALARM=SECURITY,ACCESS=DELETE+CONTROL+SUCCESS+FAILURE)-
_$ SYS$MANAGER:ACCOUNTING.DAT
```

ACL イベント・クラスはデフォルトで有効になっていますが、サイトで無効になっている場合は、次のコマンドを入力して、監査 ACE の使用を再度有効にする必要があります。

```
$ SET AUDIT/ALARM/AUDIT/ENABLE=ACL
```

### 9.2.1.3 ユーザ登録レコードの変更

時として、不審な行動をするユーザが見つかることがあります。たとえば、複数のターミナルからログインしていたり、例外的な時間帯や曜日にログインしていたりします。ユーザの行動を監視するには、ユーザ登録レコードの監査属性を変更します。AUTHORIZE ユーティリティを実行し、Audit フラグを設定します。

AUDIT フラグを設定すると、極めて多数の監査メッセージが生成されることに注意してください。次のコマンド・シーケンスは、ユーザ Robin のアカウントを変更します。

```
$ RUN SYS$SYSTEM:AUTHORIZE
UAF> MODIFY ROBIN/FLAGS=AUDIT
%UAF-I-MDFYMSG, user record(s) updated
```

Audit フラグが設定してあると、オペレーティング・システムはそのユーザのプロセスを監査します。監査ログ・ファイルには、オペレーティング・システムによる監査が可能な、ユーザの任意の行動が含まれます (9.2.2 項を参照)。監査分析ユーティリティを使用して、ユーザの行動を確認できます。たとえば、ユーザ Robin の行動に関する報告を入手するには、次のコマンドを入力します。

```
$ ANALYZE/AUDIT/SELECT=(FLAGS=MANDATORY, USERNAME=ROBIN) -
_$ SECURITY.AUDIT$JOURNAL
```

監査分析ユーティリティの詳細な説明については、9.5 節を参照してください。

## 9.2.2 オペレーティング・システムが報告できるシステム活動の種類

DCL の SET AUDIT コマンドを使用すれば、表 9-3 に示すイベント・クラスの 1 つまたは複数に対する監査を有効にできます。多くのイベント・クラスには、イベント・クラスのサブセットの定義を可能にするキーワードがあります。<sup>1</sup>

表 9-3: システムが報告できるセキュリティ・イベントの種類

イベント・クラス	説明
Access	クラス内の全オブジェクトに対するアクセス要求。特定のクラスの全保護オブジェクトに対する (特権と非特権の両方の)、選択したタイプのアクセスを監査できます。
ACL	あるオブジェクトの ACL にあるセキュリティ監査 ACE またはアラーム ACE により要求されるイベント。
Authorization	SYSUAF.DAT, NETPROXY.DAT, NET\$PROXY.DAT, または RIGHTSLLIST.DAT の任意部分の変更。
Breakin	侵入行為。

<sup>1</sup> VAX 固有



表 9-3: システムが報告できるセキュリティ・イベントの種類 (続き)

イベント・クラス	説明
Connection	SYSMAN, DECnet Phase IV, HP DECwindows Motif for OpenVMS, またはプロセス間通信 (IPC) 呼び出しを介した論理リンクの接続または切断。
Create	保護オブジェクトの作成。
Deaccess	保護オブジェクトからのアクセス解除。
Delete	保護オブジェクトの削除。
Identifier	特権としての識別子の使用。
Install	インストール・ユーティリティによる既知ファイル・リストへの変更。
Logfailure	ログインの失敗。
Login	ログインの成功。
Logout	ログアウト。
Mount	ボリュームのマウントおよびディスマウント。
NCP	ネットワーク制御プログラム (NCP) を使用したネットワーク構成データベースへの変更。
Privilege	特権の使用の成功または失敗。
Process	1 つまたは複数のプロセス制御システム・サービスの使用。
SYSGEN	System Generation ユーティリティ (SYSGEN) または AUTOGEN を使用した, システム・パラメータの変更。
Time	システム時刻の変更。

#### 9.2.2.1 一部の特権監査の抑制

サイトで特権イベント・クラスを有効にしても, オペレーティング・システムは当該クラスのあらゆるイベントを報告するわけではありません。オペレーティング・システムは次のタイプの監査を抑制します。

- インストールされているイメージが有する特権の使用の成功  
たとえば, イメージ SHOW.EXE は, WORLD 特権を有するようにインストールされます。非特権ユーザが SHOW SYSTEM コマンドを入力すると, SHOW.EXE は WORLD 特権を使用して, ワイルドカードの \$GETJPI システム・サービス呼び出しを実行します。WORLD 特権のこのような使用は監査されません。ただし, 同じ非特権ユーザが SHOW PROCESS コマンドを使用して, アクセスを許可されていないプロセスのプロセス属性を表示しようとする, その操作は失敗します。SHOW.EXE が WORLD 特権を有するようにインストールされていても, この WORLD 特権の欠如は監査されます。
- インストールされているイメージが有する特権よりも下位の特権の使用の成功

イメージが、使用される特権よりも上位の特権を有するようにインストールされている場合、要求が成功したときには、下位の特権は監査されません。たとえば、CMKRNL 特権を有するようにインストールされたイメージが、\$CMEXEC システム・サービス呼び出しの実行に成功した場合、CMEXEC 特権は監査されません。特権には次のような関係が存在します。

上位の特権	下位の特権
PRMMBX	TMPMBX
CMKRNL	CMEXEC
SYSNAM	GRPNAM
WORLD	GROUP
SYSPRV	GRPPRV
BYPASS	SYSPRV , GRPPRV , READALL , DOWNGRADE , UPGRADE

- SETPRV を有するようにインストールされたイメージによる、SETPRV 特権の使用  
オペレーティング・システムは、SETPRV の使用を監査しませんが、SETPRV を使用して有効にされた特権の使用はすべて監査します。イメージをインストールする時は実際に必要な特権を与えるようにして、SETPRV 特権を与えてイメージをインストールすることは避けることをお勧めします。
- 保護サブシステムにおいて、サブシステム識別子を使用した正常アクセス

#### 9.2.2.2 一部のプロセス制御監査の抑制

サイトでプロセス・イベント・クラスを有効にしても、オペレーティング・システムは当該クラスのあらゆるイベントを報告するわけではありません。オペレーティング・システムは次のタイプの監査を抑制します。

- DCL の RUN/TRUSTED コマンド、または PRC\$M\_TCB フラグを設定した Create Process システム・サービス (\$CREPRC) を使用して作成されたサーバ・プロセス  
クライアントに関する情報を監査する必要があるサーバ・アプリケーションは、監査対象の呼び出しの間だけプロセスの非監査設定をオーバーライドする監査フラグ NSA\$M\_SERVER または CHP\$M\_SERVER を設定できます。
- 要求側と同じ UIC を持つ、プロセスのジョブ・ツリー内のプロセス制御イベント

自分のプロセスを対象に識別子の付与または削除を行っても、プロセス制御監査は発生しません。ただし、\$CREPRC と \$DELP RC の使用に関連するイベントは、常に監査されます。

### 9.2.3 イベント情報のソース

アプリケーションとシステム・プログラムは、次のシステム・サービスを呼び出すことにより、セキュリティ・イベント情報を提供できます。

- \$AUDIT\_EVENT
- \$CHECK\_PRIVILEGE
- \$CHKPRO および \$CHECK\_ACCESS

#### イベント監査 (\$AUDIT\_EVENT) システム・サービス

オペレーティング・システムは、システムでセキュリティに関わるイベントが発生するたびに、\$AUDIT\_EVENT システム・サービスを呼び出します。システム・サービスは、SET AUDIT 設定を参照して、当該イベントの監査が有効になっているかどうかを判定します。イベントのアラームまたは監査が有効である場合、\$AUDIT\_EVENT は監査レコードを作成します。この監査レコードは、関係したプロセス(サブジェクト)を示し、呼び出し元により提供されたイベント情報を列挙します。

#### 特権チェック (\$CHECK\_PRIVILEGE) システム・サービス

オペレーティング・システムは、ユーザが特権機能を実行しようとするたびに \$CHECK\_PRIVILEGE システム・サービスを呼び出します。OpenVMS 特権の現在のセットは、付録 A に示します。このシステム・サービスは特権チェックを実行し、SET AUDIT 設定を参照して、特権の監査が有効になっているかどうかを判定します。特権の監査が有効であれば、\$CHECK\_PRIVILEGE は監査レコードを作成します。監査レコードは、関係したプロセス(サブジェクト)と特権を示し、特権チェックの結果を提供し、その呼び出し元より提供された補足イベント情報を列挙します。通常、特権監査レコードには、特権チェックに対応する DCL コマンド行またはシステム・サービス名が含まれています。

#### 保護チェック (\$CHKPRO) およびアクセス・チェック (\$CHECK\_ACCESS) システム・サービス

オペレーティング・システムは、プロセス(サブジェクト)が保護オブジェクトにアクセスしようとするたびに \$CHKPRO システム・サービスを呼び出します。システム・サービスは、4.3 節「システムによる保護オブジェクトへのユーザのアクセス可否の判定」で説明している規則に従って、アクセス・アービトレーションを実行します。このシステム・サービスは、対応するオブジェクト・クラスの SET AUDIT 設定を参照することで、対応するオブジェクト・アクセス・イベントの監

査を有効にしたかどうかも判定します。アラームまたは監査が必要である場合、\$CHKPRO は監査レコードを作成します。この監査レコードは、関係したプロセス（サブジェクト）とオブジェクトを示し、チェックの最終的な結果と、呼び出し元による補足イベント情報を含みます。

特権サーバ・プロセスは、\$CHECK\_ACCESS システム・サービスを使用して、サービス対象の保護オブジェクトへのアクセス権をクライアントに与えるべきかどうかを判定します。\$CHECK\_ACCESS システム・サービスは、サーバに適した呼び出しインタフェースを提供し、\$CHKPRO サービスの上位の層に配置されます。そのため、\$CHKPRO 同じ方法でオブジェクト・アクセス監査を実行します。

## 9.3 監査計画の策定

システム管理者またはサイトのセキュリティ管理者が監査対象にするセキュリティ・イベントを把握するには、サイトで必要なセキュリティのレベルを決める必要があります。

### 9.3.1 監査要件の評価

監査要件の評価は、次の 2 つのステップからなるプロセスです。

1. サイトの全般的なセキュリティ要件が、高、中、低のいずれであるかを判定します。表 1-1 に、セキュリティ上のニーズを判定するためのガイダンスを示します。
2. サイトのニーズが判明したら、有効にすべきイベント・クラスの推奨リストについて、表 9-4 を参照します。

サイトの要件の全般的な方向性を決めたら、セキュリティ報告の現実的な量を検討する必要があります。表 9-4 の推奨事項と、次に示すサイトの要素とのバランスを考慮します。

- サイトのデータの機密保護の重要性
- ログ・ファイルの分析に費やせる時間
- 使用可能なディスク容量
- セキュリティの脅威に関する知識（発生源、または発生源の可能性が高い場所）
- システムのチューニング要件（性能への影響の詳細については、9.3.3 項を参照）

表 9-4: サイトのセキュリティ要件に応じて監視すべきイベント

	低	中	高
目標	影響の大きいローカル・イベントの監視	システム定義に対する変更の追跡	データベースへの変更の監視，プロセス制御システム・サービスの使用の追跡，DECnet Phase IV を介したネットワーク接続の監視 (VAX のみ)
アラームとして有効にすべきクラス	ACL，登録，侵入(全タイプ)，ログイン失敗(全タイプ)	「低」カテゴリと同じ内容に加え，SECURITY 特権の使用	「中」カテゴリと同じ内容に加え，INSTALL，時刻，SYSGEN，特権使用の失敗
監査として有効にすべきクラス	ACL，登録，侵入(全タイプ)，ログイン失敗(全タイプ)	「低」カテゴリのすべてに加え，INSTALL，時刻，SYSGEN，特権，ログイン(全タイプ)，ログアウト(全タイプ)，(BYPASS，SYSPRV，および READALL 特権を介した) ファイルのアクセス，ファイル，デバイス，およびボリュームへのアクセス失敗	「中」カテゴリのすべてに加え，識別子，プロセス，保護オブジェクトへのアクセス失敗，NCP，接続 (VAX のみ)

表 9-4 において，セキュリティ要件が低いサイトに推奨されているイベント・クラスは，オペレーティング・システムのデフォルト設定になっています。これらのクラスがシステムで現在のデフォルトになっていない場合は，次のコマンドを使用して有効にします。

```
$ SET AUDIT/ALARM/AUDIT -
_$ /ENABLE=(ACL,AUTHORIZATION,BREAKIN:ALL,LOGFAILURE:ALL)
```

セキュリティ要件が中程度であるサイトでは，システムを再定義するようなイベントを監査する必要があります。システム・ファイル，システム時刻，またはシステム・パラメータに対する変更が監視対象です。また，イメージのインストールと，特権の使用も監視します。例 9-3 に，セキュリティ要件が中程度であるサイトの監査設定を示します。

### 例 9-3: セキュリティ要件が中程度であるサイトの イベントの監査

```
System security alarms currently enabled for:
  Authorization
  Breakin:          dialup,local,remote,network,detached

System security audits currently enabled for:
  ACL
  Authorization
  INSTALL
  時刻
  SYSGEN
  Breakin:          dialup,local,remote,network,detached
  Login:            batch,dialup,local,remote,network,subprocess,detached,server
  Logfailure:       batch,dialup,local,remote,network,subprocess,detached,server
  Logout:           batch,dialup,local,remote,network,subprocess,detached,server

Privilege use:
  ACNT      ALLSPOOL  ALTPRI    AUDIT      BUG        BYPASS     CMEXEC     CMKRNL
  DIAGNOSE  DOWNGRADE  EXQUOTA   GROUP      GRPNAM     GRPPRV     IMPORT     IMPERSONATE
  LOG_IO    MOUNT        NETMBX    OPER       PFNMAP     PHY_IO     PRMCEB     PRMGBL
  PRMMBX    PSWAPM       READALL   SECURITY    SETPRV     SHARE      SHMEM      SYSGBL
  SYSLOCK   SYSNAM       SYSPRV    TMPMBX     UPGRADE    VOLPRO     WORLD

Privilege failure:
  ACNT      ALLSPOOL  ALTPRI    AUDIT      BUGCHK     BYPASS     CMEXEC     CMKRNL
  DIAGNOSE  DOWNGRADE  EXQUOTA   GROUP      GRPNAM     GRPPRV     IMPORT     IMPERSONATE
  LOG_IO    MOUNT        NETMBX    OPER       PFNMAP     PHY_IO     PRMCEB     PRMGBL
  PRMMBX    PSWAPM       READALL   SECURITY    SETPRV     SHARE      SHMEM      SYSGBL
  SYSLOCK   SYSNAM       SYSPRV    TMPMBX     UPGRADE    VOLPRO     WORLD

FILE access:
  SYSPRV:      read,write,execute,delete,control
  BYPASS:      read,write,execute,delete,control
  READALL:     read,write,execute,delete,control
```

中程度のレベルの監査の設定を有効にするには、デフォルトのイベントがすでに有効であるという前提で、次のコマンドのセットを入力します。

```
$ SET
AUDIT/ALARM/AUDIT/ENABLE=PRIVILEGE=(SUCCESS:SECURITY,FAILURE:SECURITY)
$ SET AUDIT/AUDIT/ENABLE=(INSTALL,SYSGEN,TIME,PRIVILEGE=(SUCCESS,FAILURE))
$ SET AUDIT/AUDIT/ENABLE=ACCESS=(BYPASS,SYSPRV,READALL)/CLASS=FILE
$ SET AUDIT/AUDIT/ENABLE=ACCESS=FAILURE/CLASS=(FILE,DEVICE,VOLUME)
```

セキュリティ要件が高いサイトでは、ネットワークの活動を含むように監査の範囲を拡張します。ネットワーク・データベースに対する変更、ネットワーク接続 (VAX のみ)、特権としての識別子の使用、および特権ファイル・アクセスを監視する必要があります。SYSPRV、BYPASS、または READALL 特権を介したすべてのファイル・アクセスを監視し、また GRPPRV 特権を介したファイル・アクセスの成功と失敗の両方を監視します。高レベルの監査の設定を有効にするには、中レベルがすでに有効であるという前提で、次のコマンドのセットを入力します。

```
$ SET AUDIT/ALARM/ENABLE=(INSTALL,SYSGEN,TIME,PRIVILEGE=(FAILURE:ALL))
$ SET AUDIT/AUDIT/ENABLE=(CONNECTION,IDENTIFIER,NCP,PROCESS:ALL)
$ SET AUDIT/AUDIT/ENABLE=ACCESS=FAILURE/CLASS=*
```

すべての監査を有効にするには、次のコマンドを入力します。

```
$ SET AUDIT/AUDIT/ENABLE=ALL/CLASS=*
```

すべての監査を無効にするには、次のコマンドを入力します。

```
$ SET AUDIT/AUDIT/DISABLE=ALL/CLASS=*
```

有効にすべき他の推奨イベント・クラスについては、10.3.2 項を参照してください。

### 9.3.2 イベント・メッセージの出力先の選択

オペレーティング・システムは、セキュリティ・イベントを、アラームと監査のどちらの形式でも報告できます(9.2.1.1 項を参照)。どちらの形式を選択するかは、イベントの性質によります。侵入行為や、システム・ユーザ登録ファイル(SYSUAF.DAT)への変更など、リアルタイムのイベントや直ちに対処しなければならないイベントは、アラームと監査の両方として有効にすべきクラスです。これらよりも重要性の低いイベントについては、監査のみ有効にするということができます。ハードコピー・オペレータ・ターミナルを使用する場合を除き、アラーム・レコードはすぐに他のシステム・メッセージに置き換わってしまいます。システム・セキュリティ監査ログに書き込まれる監査イベント・レコードは、まとめて調査できるように保存されます。

イベント・メッセージの調査には、メリットがあります。多くの場合、単独の監査メッセージから得られる情報はわずかですが、大量の監査レコードがあれば、セキュリティ違反を示す可能性のある活動パターンが明らかになります。たとえば、オブジェクトのアクセスを監査することで、セキュリティ管理者は時刻のパターン、アクセスされているオブジェクトのタイプ、およびその他のシステム情報を把握し、これらを総合することで、システム活動の全体像を描くことができます。9.5 節では、監査ログ・ファイルからレポートを作成する方法を説明します。

### 9.3.3 性能への影響の考慮

オペレーティング・システムにより実行されるデフォルトの監査は、主に登録データベースへの変更を追跡します。システム・ユーザ登録ファイル(SYSUAF.DAT)に対する変更やイメージのインストールなどのシステム・イベントは、それほど頻繁には発生しないため、システムの資源を大量に消費することはありません。

システムの使われ方を把握せずに、また監査情報の価値を評価することなく、サイトで追加イベント・クラスを有効にして、特にアクセス・イベントや特権イベントなどの追加のイベント・クラスを監査すると、大量のシステム資源が消費される可能性があります。この点では、監査報告システムの実装は、システムのチューニングに似ています。つまり、不要な詳細データが含まれていない、適切な報告のレベルを実現するには、少し時間がかかります。このため、一度にすべてではなく、段階的に監査を有効にして、適切なバランスに到達するまで徐々にイベント・クラスの追加/削除を行うことをお勧めします。次のガイドラインに従います。

- 9.3.1 項 で説明している方法で、監査の要件を評価します。

- オブジェクト・アクセス・イベントは選択して監査します。オブジェクト・アクセス・イベントは常時発生するため、システムの性能に最大の影響を与えます。通常は、ファイル・アクセスの成功ではなくファイル・アクセスの失敗を監査し、またファイル・クラス全体の監査を有効にするのではなく、重要なファイルに監査 ACE を適用します。
- 実行しているレイヤード・プロダクトを調査して、どの特権が使用される可能性があるかを把握します。また、バックアップ操作時の READALL 特権の使用など、サイト固有のプロシージャを把握しておきます。特権イベントは頻繁に発生するため、システム性能に大きな影響を与えます。
- 一度に有効にするイベント・クラスは少数にして、十分なイベント情報が得られるようになるまで、必要に応じて追加/削除します。有効にするクラスが多いほど、オーバーヘッドが大きくなり、システム上の有用な作業に使用できる資源が少なくなります。

特に次の 2 つのコマンドは、大量の監査メッセージを生成します。

- DCL の PIPE コマンドは、1 つの PIPE コマンドを実行するために多数のサブプロセスを作成することがあります。このことは、サブプロセスの活動に関連するイベント（プロセスの作成、プロセスの削除、ログイン、ログイン失敗、ログアウトなど）の監査が増大する可能性があることを意味します。
- UAF の MODIFY USER/FLAG=AUDIT コマンドは、非常に多くの監査メッセージを生成します。通常はこのフラグを設定する必要はありません。つまり、特定の AUDIT が有効である場合、ユーザ・フラグを設定する必要はありません。

## 9.4 イベント・メッセージの取得方法

オペレーティング・システムは、監査ログ・ファイルおよびオペレータ・ターミナルにイベント・メッセージを送信できます。サイトで追加のコピーが必要な場合、オペレーティング・システムは遠隔のログ・ファイルやアプリケーション・リスナ・メールボックスにメッセージのコピーを送信できます。

### 9.4.1 監査ログ・ファイルの使用

オペレーティング・システムは、最新バージョンのセキュリティ監査ログ・ファイルに、すべてのセキュリティ・イベント・メッセージを書き込みます。このログ・ファイルは、デフォルトではシステム・スタートアップ時に SYS\$COMMON:[SYSMGR] ディレクトリに作成され、SECURITY.AUDIT\$JOURNAL という名前が付けられています。表 9-5 に、このファイルの最も顕著な特徴を示します。



表 9-5: 監査ログ・ファイルの特徴

特徴	メリット
バイナリ	バイナリ・ファイルが必要とするディスク領域は最少です。
クラスタ・ワイド	クラスタ・ワイド・ファイルを監査分析ユーティリティで処理すると、クラスタ内のセキュリティ関連イベントについて単独のレポートが生成されます。
順次レコード形式	順次レコード形式は、ユーザが記述するプログラムで分析するのが簡単です。セキュリティ監査ログ・ファイルのメッセージ形式の説明については、『 <i>OpenVMS システム管理ユーティリティ・リファレンス・マニュアル</i> 』を参照してください。

通常、すべてのクラスタ・イベントが単独の監査ログ・ファイルに書き込まれます。クラスタで1つのセキュリティ監査ログ・ファイルを使用することで、システム上の全セキュリティ関連イベントの記録が一本化されます。このため、ノード固有の複数の監査ログよりも、1つのクラスタ・ワイド・ログ・ファイルの方が優れています。ノード固有の監査ログでは、クラスタ全体のイベントの相互関係が失われ、セキュリティ・イベントの分析が不完全になってしまうためです。必要であればノード固有の監査ログを作成できますが（9.4.1.1 項を参照）、お勧めしません。

セキュリティ監査ログ・ファイルの有用性は、どのような手順を採用するかによって決まります。

- イベントを早期に認識でき、ファイルが大きくなりすぎないように、ログ・ファイルを維持管理する（9.4.1.1 項を参照）。
- 定期的にログ・ファイルを確認し、不審な活動を調査する（9.5 節を参照）。

#### 9.4.1.1 ファイルの保守

セキュリティ監査ログ・ファイルは、何らかの対応を取るまで増大し続けるため、このファイルの保守計画を考えておく必要があります。

一般的には、サイトがログ・ファイルの名前を毎日変更し、新しいログ・ファイルを作成します。新しい、クラスタ・ワイドのセキュリティ監査ログ・ファイルを開くには、次のコマンドを使用します。

```
$ SET AUDIT/SERVER=NEW_LOG
```

ノード固有の新しいログを作成するには、SET AUDIT/SERVER=NEW\_LOG コマンドの前に SET AUDIT/DESTINATION=*filespec* コマンドを実行します。filespec は、ノード固有のファイルに解決される論理名 (SYS\$SPECIFIC:[SYSMGR]SECURITY など) が含まれるファイル指定です。

新しいログを開いたら、古いバージョンの名前を、データの開始日または終了日を含む名前に変更します。

システム・ディスクの領域を節約するために、ファイルを別のディスクにコピーして、システム・ディスクからそのログを削除できます。セキュリティ要件が高い環境では一般的な、専用の監査ディスクが装備されているサイトであっても、今後のメッセージ用に領域を空けるため、古いバージョンを移動させる場合があります。

ファイルをアーカイブしたら、古いログを対象に監査分析ユーティリティを実行します(9.5.2 項を参照)。このファイルをアーカイブすることにより、監査メッセージについてクラスタ・ワイドの履歴を維持管理します。万一システム上にセキュリティ上の危険が見つかった場合は、アーカイブされたログ・ファイルを分析して、指定した期間におけるユーザの疑わしい処理の形跡を調べることができます。

#### 9.4.1.2 システム・ディスクからのファイルの移動

SYS\$COMMON:[SYSMGR] ディレクトリからファイルを移動するには、コマンド・プロシージャ SYSECURITY.COM を編集します。このプロシージャは、システムのリブートのたびに監査サーバが起動する前に実行されます。

ファイルを移動するには、次の手順を実行します。

1. 監査サーバ・プロセスの起動後ではなく起動前に指定の監査ディスクをマウントするよう、オペレーティング・システムに指示するため、SYSECURITY.COM に 1 行を追加することで、スタートアップ・シーケンスを変更します。次に例を示します。

```
$ IF .NOT. F$GETDVI("$1$DUA2", "MNT") -  
_$_ THEN MOUNT/SYSTEM $1$DUA2 AUDIT AUDIT$ /NOREBUILD
```

この例のコマンドは、\$1\$DUA2 上にある AUDIT というラベルのボリュームをマウントし、システム全体でそのボリュームを使用できるようにします。また、MOUNT により、論理名 AUDIT\$ も割り当てられます。

2. 必要であれば、監査サーバ・データベースを監査ディスクに移動します。データベースのサイズは小さく、比較的变化が少ないため、この手順はあまり重要ではありません。

データベースを移動するには、SYSECURITY.COM に 2 行目を追加し、システム論理名 VMS\$AUDIT\_SERVER を定義します。2 行目は、監査ディスクをマウントする行の次の行にします。コマンドを使用してシステム論理名を定義し、論理名 AUDIT\$ を持つディスク上の VMS\$AUDIT\_SERVER データ・ファイルに、そのシステム論理名を割り当てます。次に例を示します。

```
$ DEFINE/SYSTEM/EXEC VMS$AUDIT_SERVER AUDIT$: [AUDIT]VMS$AUDIT_SERVER.DAT
```

このコマンドは、手順 1 でマウントした \$1\$DUA2 上のボリュームに、監査サーバ・データベースをリダイレクトします。

3. DCL レベルから、SYSECURITY.COM でマウントされたボリュームに、セキュリティ監査ログ・ファイルをリダイレクトします(手順 1 を参照)。SET

AUDIT コマンドを使用して、セキュリティ監査ログ・ファイルの新しい位置を監査サーバ・データベースに反映し、クラスタ内の各ノード上の監査サーバ・プロセスに対して、そのファイルの使用を開始するよう通知します。次に例を示します。

```
$ SET AUDIT/JOURNAL=SECURITY -  
_ $ /DESTINATION=AUDIT$: [AUDIT] SECURITY
```

システムの再起動のたびにこのコマンドを繰り返す必要はありません。

セキュリティ監査ログ・ファイルの指定で論理名を使用する場合、その論理名は、SYSECURITY.COM の /SYSTEM 論理名として定義する必要があります。

## 9.4.2 ターミナルのアラーム受信の有効化

オペレーティング・システムは、セキュリティ・クラス・メッセージが有効になっているターミナルに対して、アラーム・メッセージを送信します。ほとんどの場合、これらのセキュリティ・アラームはデフォルトでシステム・コンソールに表示されます。メッセージはすぐにスクロールして画面から消えてしまうため、セキュリティ・クラス・メッセージ用に独立したターミナルを有効にして、システム・コンソールへのメッセージ配信を無効にすることをお勧めします。安全な場所ハードコピー出力を行うターミナルを用意するか、専任スタッフにセキュリティ・オペレータ・ターミナルを監視させます。セキュリティ・オペレータとして有効にできるターミナルの数に制限はありません。

セキュリティ・クラス・アラームを受信するようターミナルを設定するには、使用するターミナルから次の DCL コマンドを入力します。

```
$ REPLY/ENABLE=SECURITY
```

特定のターミナルを長期間使用する場合は、サイト固有のスタートアップ・コマンド・プロシージャを変更して、そのターミナルを自動的に有効にすることができます。たとえば、スタートアップ・コマンド・プロシージャに次のコマンド行を含めると、システム・コンソールへのセキュリティ・アラームの配信が無効になり、ターミナル TTA3 でアラームが有効になります。

```
$ DEFINE/USER SYS$COMMAND OPA0:  
$ REPLY/DISABLE=SECURITY  
$ DEFINE/USER SYS$COMMAND TTA3:  
$ REPLY/ENABLE=SECURITY
```

登録イベント・クラスと SYSGEN イベント・クラスは、非常に長いアラーム・メッセージを生成することがあるため、メッセージが切り詰められることがあります。このため、これらのクラスではアラームと監査の両方を有効にすることをお勧めします。アラーム・メッセージが切り詰められると、そのテキストはアラーム・メッセージが不完全であることを示します。対象クラスの監査メッセージ出力を有効にしている限り、ANALYZE/AUDIT を使用して完全なメッセージを表示することができます。

### 9.4.3 イベント・メッセージの第2の出力先

オペレータ・ターミナルと監査ログ・ファイルは、セキュリティ・イベント・メッセージの第1の出力先です。サイトでは、(アーカイブ・ファイルと呼ばれる)遠隔ログ・ファイルまたはリスナ・メールボックスに、監査メッセージのコピーを送信することを選択できます。

#### 9.4.3.1 遠隔ログ・ファイルの使用

このオペレーティング・システムでは、管理能力が限定されているワークステーションやユーザが、別のノードに監査ログ・ファイルを複製することができます。セキュリティ・アーカイブ・ファイルと呼ばれるこの第2ログは、遠隔ノードに置かれ、それを分析する能力を持つセキュリティ管理者が利用できるようになります。場合によっては、アーカイブ・ファイルが、ローカルの監査ログ・ファイルが何らかの方法で改ざんされた場合の保険にもなります。ノード単位で監査メッセージをアーカイブ・ファイルに出力できます。有効にすると、監査サーバは各監査メッセージのコピーを、セキュリティ監査ログ・ファイルだけでなくセキュリティ・アーカイブ・ファイルにも書き込みます。

---

#### 注意

---

クラスタ内の各ノードは、それぞれ専用のアーカイブ・ファイルが必要です。アーカイブ・ファイルは、クラスタ内の複数のノードでは共用できません。

---

セキュリティ監査メッセージを遠隔セキュリティ・アーカイブ・ファイルに書き込むには、次の手順を実行します。

1. アーカイブ・ファイルが存在するノードにログインし、監査サーバのアカウントを作成します。そのアカウントに、AUDIT\_ARCHIVEのようなユーザ名を割り当てます。アカウントを非特権にし、ネットワーク・アクセスのみを残します。アカウントが、セキュリティ・アーカイブ・ファイルが格納されているデバイスとディレクトリにアクセスできることを確認します。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> ADD AUDIT_ARCHIVE /ACCESS=NETWORK /DEVICE=WORK2-
_UAF> /DIRECTORY=[AUDIT_ARCHIVE]
```

2. AUDIT\$SERVER に対する代理アカウントを遠隔ノードに追加します。これにより、監査サーバ・プロセスは、遠隔ノード上のアカウントにデータを書き込めるようになります。たとえば次のコマンドは、ノード SMLNOD 上の監査サーバ・プロセスに、ノード BIGNOD 上の AUDIT\_ARCHIVE アカウントへの代理アクセスを許可します。

```
UAF> ADD/PROXY SMLNOD::AUDIT$SERVER AUDIT_ARCHIVE/DEFAULT
UAF> EXIT
```

代理アカウントの設定の詳細については、12.3.2 項を参照してください。

3. 遠隔ノードからログアウトします。ローカル・ノード上で次のコマンドを入力して、ノードへのログ・ファイルのアーカイブ作成を有効にします。

```
$ SET AUDIT/ARCHIVE=ALL/DESTINATION=BIGNOD::WORK2:-
_$ [AUDIT_ARCHIVE]SMLNOD_MAY_93.AUDIT$JOURNAL
```

ディレクトリの指定は完全な形式で行う必要があります。論理名を含める場合は、ローカルの監査サーバ・プロセスが論理名を変換できることを確認します。

新しいアーカイブ・ファイルを作成するには、現在のファイルの名前を変更します。システムの次回起動時に、システムによって新しいファイルが作成されます。

ネットワークが停止すると、セキュリティ・アーカイブ・ファイル用のメッセージは失われます。セキュリティ・オペレータ・ターミナルは、接続が失われたという通知と、失われたメッセージの数を受信します。ネットワークが回復すれば、監査サーバは元のアーカイブ・ファイルへの接続を再度確立し、イベント・メッセージの書き込みを継続します。

セキュリティ・アーカイブ・ファイルの分析は、多くの点でセキュリティ監査ログ・ファイルの分析と同じです。ファイルが開いている状態でも、遠隔セキュリティ・アーカイブ・ファイルはいつでも分析できます。詳細については 9.5 節を参照してください。

#### 9.4.3.2 リスナ・メールボックスの使用

セキュリティ監査機能の追加機能として、リスナ・デバイスを作成してセキュリティ監査メッセージのバイナリ・コピーを受信することができます。リスナ・デバイスとは、メールボックス作成 [\$CREMBX] システム・サービスを使用して作成する、パーマネント・メールボックスまたは一時的メールボックスです。アプリケーションを設定して監査情報を受信および処理し、システム上でイベントが発生した時点でイベントに対応するようにできます。システムごとに 1 つのリスナ・デバイスを持つことができ、そのリスナ・デバイスはローカル・ノード上で発生するイベントのみを受け取れます。

リスナ・デバイスを有効にしてセキュリティ監査メッセージを受信するには、次の形式で SET AUDIT/LISTENER コマンドを実行します。

```
SET AUDIT/LISTENER=device-name
```

*device-name* パラメータとして、メールボックスの作成時に指定した論理名を使用するか、"MBAn" の形式でメールボックスの等価名を使用します (*n* はメールボックスのユニット番号を表します)。デバイスを一時的メールボックスとして作成する場合

合は、メールボックス・デバイス名を返すには、デバイスおよびボリューム情報取得(\$GETDVI) システム・サービスを使用する必要があります。

監査リスナ・デバイスを無効にするには、次のコマンドを入力します。

```
$ SET AUDIT/NOLISTENER
```

VAX システムの場合、DECtalk デバイス上のリスナ・メールボックスに送信される監査イベント・メッセージを処理するプログラムの例については、SYS\$EXAMPLES ディレクトリにある、AUDSRV\_LISTENER.B32 ファイル (VAX BLISS プログラム) および AUDSRV\_LISTENER.MAR ファイル (VAX MACRO プログラム) を参照してください。

## 9.5 ログ・ファイルの分析

セキュリティ監査ログ・ファイルでセキュリティ監査メッセージを収集しても、そのファイルを確認して疑わしい活動を調べなければ意味がありません。監査分析ユーティリティ (ANALYZE/AUDIT) を使用して、セキュリティ監査ログ・ファイルのデータを調べます。

システム上の通常の活動を把握し、例外的な活動を簡単に見つけられるように、ANALYZE/AUDIT はログ・ファイルからレポートを作成します。ANALYZE/AUDIT はイベントを要約し、クラスタ上で活動が行われている場所を示します。このユーティリティは、監査レポートから情報のサブセットを選択し、分析用により詳細な情報を提供できるため、例外的な処理の分析にも役立ちます。1 つの監査ログ・ファイルを分析してもあまり意味がない可能性はありますが、監査レコードが長期に渡れば、セキュリティ違反を示す活動パターンを明らかにすることがあります。

### 9.5.1 推奨される手順

この節では、システムの監査ログ・ファイルを分析する方法を説明します。ANALYZE/AUDIT の使用方法是サイトのセキュリティ要件に依存しますが、ユーティリティの使用範囲に関係なく、従うべき共通の手順がいくつか存在します。潜在的なセキュリティ問題を認識できるようになる前には、システムの通常の運用を把握する必要があります。定期的に監査レポートを作成および確認する手順を策定できるのは、その後です。監査ログ・ファイルの定期的な分析の中でセキュリティ問題の疑いが見つかった場合は、選択したセキュリティ・イベントの詳細な調査を実行する必要があります。

#### 手順 1：通常の状態の把握

セキュリティ管理者は、監査ログ・ファイルを分析するためには、次の質問に答えられる必要があります。

- システムの大部分のユーザが作業を行う一般的な時間帯。
- 通常時に高度な特権を使用して処理を行う特定のユーザが存在するかどうか。

- どのイメージが他のアプリケーションの一部としてシステム・セキュリティ・イベントを作成するか。
- 特定の時間帯に実行される定期的なバッチ・ジョブまたはネットワーク・ジョブが存在するかどうか。

以上の質問に対する答えを知っておけば、セキュリティ問題を誤って疑う原因となるアラームの誤報をなくすることができます。

## 手順 2：監査レポートの定期的な分析

最も一般的に作成するレポートのタイプは、簡潔な毎日のイベントのリストです。コマンド・プロシージャを作成し、毎晩、夜半前にバッチ・ジョブとして実行して、その日のセキュリティ・イベント・メッセージのレポートを生成するようにできます。同じプロシージャで、監査ログの新しいバージョンを作成するようにもできます（9.4.1.1 項を参照）。

次の例は、このレポートを生成するための ANALYZE/AUDIT コマンド行を示します。

```
$ ANALYZE/AUDIT/SINCE=TODAY/OUTPUT=31DEC2000.AUDIT - [1]
_$ SYS$MANAGER:SECURITY.AUDIT$JOURNAL
$ MAIL/SUBJECT="Security Events" 31DEC2000.AUDIT SYSTEM [2]
```

1. このプロシージャ例の最初のコマンドでは、31DEC2000.AUDIT という名前の監査レポートを生成しています。これには、当日生成された全セキュリティ・イベント・メッセージの情報が 1 行ずつ含まれています。
2. 2 番目のコマンドでは、調査のために、ファイルをセキュリティ管理者にメールで送信しています。

システム上で監査しているセキュリティ・イベントの数によっては、監査ログ・ファイルに書き込まれるすべての監査レコードを確認する作業は現実的ではない場合があります。このような場合、登録データベースの変更や侵入行為に関連するすべての監査レコードや、通常の業務時間外に発生したすべてのイベントなど、ログ・ファイルから特定のセットのレコードを選択することができます。

(DCL の PIPE コマンドにより作成される) PIPE サブプロセスが監査を生成できることを念頭において、サブプロセス関連の監査を分析します。PIPE コマンドは、1 つの PIPE コマンドを実行するために多数のサブプロセスを作成することがあります。このことは、サブプロセスの活動（プロセスの作成、プロセスの削除、ログイン、ログイン失敗、ログアウトなど）に関連する監査イベントが増大する可能性があることを意味します。

監査レポートの確認は、できる限りすみやかに行うことが重要です。レポートの検査が早いほど、システムに対するセキュリティ侵害の可能性の検出と、問題の程度の判断が早くできるようになります。前日の監査レポートの検査を朝の定期的な作

業の一部にしたり、レポートを確認し、疑わしいイベントが発生した場合に Mail ユーティリティ (MAIL) を使用して通知するプログラムを作成することもできます。

### 手順 3：疑わしい活動の調査

レポートの確認時に、通常の業務時間外のログインの試みなど、疑わしい、または不適切と思われるセキュリティ・イベントが見つかった場合は、監査分析ユーティリティを使用して、セキュリティ監査ログ・ファイルを詳細に調べます。完全なレポートを見ることで、監査ログ・ファイルに記録されているどのセキュリティ・イベントをより徹底的な調査するべきかを判断できます。

次のコマンドを使用して、選択したセキュリティ監査レコードの完全なレポートを生成できます。

```
$ ANALYZE/AUDIT/FULL/SINCE=TODAY/OUTPUT=31DEC2000.AUDIT -  
_ $ /EVENT_TYPE=(BREAKIN,RIGHTSDB,SYSUAF)  
$ MAIL/SUBJECT="Security Events" 31DEC2000.AUDIT SYSTEM
```

2000 年 12 月 31 日の監査レポートには、すべての侵入行為と、システム・ユーザ登録ファイル (SYSUAF.DAT) およびライト・データベース (RIGHTSLIST.DAT) へのすべての変更に関する情報が含まれています。

## 9.5.2 監査分析ユーティリティの起動

監査分析ユーティリティは、バイナリ・ログ・ファイルから、意味のあるレポートを作成するために使用するツールです。この節と以降の節では、このユーティリティの使用方法を説明しますが、ユーティリティのコマンドと修飾子の完全な説明については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照してください。

監査分析ユーティリティを起動するには、次の DCL コマンドを使用します。

```
ANALYZE/AUDIT file-name
```

*file-name* パラメータの部分には、監査レポートの元となるファイルの名前を使用します。セキュリティ監査ログ・ファイルのデフォルトの名前は、SECURITY.AUDIT\$JOURNAL です。ディレクトリ SYS\$MANAGER を指定する必要があります。

## 9.5.3 レポートの指定

監査分析ユーティリティを使用して、1 つの監査ログからセキュリティ・イベント・メッセージの一部または全部を抽出し、さまざまなレベルの詳細を含んだレポートを作成できます。

監査レポートには、サイトで有効になっているイベント・クラスのセットに含まれているイベントが反映されます (9.2 節を参照)。イベントの一部のみが抽出されるように、レポートを調整することができます。時間、イベント・クラス、また



はイベント・メッセージ内のデータのフィールドに基づいて選択基準を設定できます。『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の /SELECT 修飾子の解説を参照してください。レポートの内容を決定する修飾子の要約を、表 9-6 に示します。

表 9-6: 監査分析ユーティリティの修飾子

タイプ	修飾子	説明
内容	/BEFORE	指定時間の前に記録されたイベント・メッセージを抽出します。
	/SINCE	指定時間の後に記録されたイベント・メッセージを抽出します。
	/EVENT_TYPE	特定のイベント・クラスのイベント・メッセージを抽出します (表 9-3 を参照)。
	/SELECT	メッセージ内のデータに基づいてイベント・メッセージを抽出します。たとえば、/SELECT=USERNAME=JSNOOP と指定すると、ユーザ JSNOOP により作成されたセキュリティ・イベント・メッセージのみを列挙します。
	/IGNORE	メッセージ内のデータに基づいて、レポートからイベント・メッセージを除外します。
形式	/BRIEF	監査ログ・ファイル内のレコードごとに、イベントのタイプ、イベントが発生した日時、イベントの発生源であるターミナルなどの情報で構成される 1 行の情報を含むレポートを作成します (例 9-4 を参照)。これがデフォルトです。
	/FULL	処理される監査ログ・ファイル内のレコードごとに、可能なすべてのデータを提供します (例 9-5 を参照)。付録 D に、各イベント・クラスのアラーム・メッセージの例を示します。
	/SUMMARY	分析対象ログ・ファイル内のイベント・クラスごとの監査メッセージの合計数を列挙します (例 9-6 を参照)。また、各ノード上の 1 時間ごとのイベントの集計も出力できます。

表 9-6: 監査分析ユーティリティの修飾子 (続き)

タイプ	修飾子	説明
	/BINARY	独自に用意するデータ削減ツールを使用してさらに詳細な分析を行うためのレコードを抽出できるよう、バイナリ・ファイルを作成します。監査メッセージのレコードの形式については、『 <i>OpenVMS</i> システム管理ユーティリティ・リファレンス・マニュアル』を参照してください。
出力先	/OUTPUT	レポートの出力先を指定します。デフォルトでは SYS\$OUTPUT に出力されます。

ANALYZE/AUDIT は、さまざまな形式で監査レポートを作成します (表 9-6 を参照)。デフォルトでは、このユーティリティはログ・ファイルのレコードごとに 1 行の要約を作成します。簡潔な 1 行のレポートは、ログ・ファイルの定期的な分析には最も便利です。より詳細な完全レポートは、疑わしいレコードの分析に必要な詳細情報を提供します。ログ・ファイルの一部をアーカイブしたい場合は、バイナリ出力により監査ログ・ファイルのサブセットを保存することができます。

要約レポートによって、セキュリティ問題の可能性を素早く特定することができます。要約レポートは、セキュリティ・イベントのクラスごとに、分析対象のセキュリティ監査ログ・ファイルから抽出された監査メッセージの合計数を列挙することができます。また要約レポートは、イベント・メッセージを生成したシステム、イベントの発生時刻、および確認されたイベントの合計数に基づいて、監査活動の一覧表を表示することもできます。

例 9-4 に、システム・セキュリティ監査ログ・ファイルに記録された、全セキュリティ監査イベントの簡略レポートを示します。レポートを生成する ANALYZE/AUDIT コマンドでは、使用する監査ログ・ファイルの名前に置き換えます。

#### 例 9-4: 簡略監査レポート

```
$ ANALYZE/AUDIT/BRIEF SYS$MANAGER:SECURITY.AUDIT$JOURNAL
      Date / Time      Type      Subtype      Node      Username      ID      Term
-----
1-NOV-2000 16:00:03.37 ACCESS  FILE_ACCESS  HERE      SYSTEM      5B600AE4
1-NOV-2000 16:00:59.66 LOGIN   SUBPROCESS   GONE      ROBINSON    3BA011D4
1-NOV-2000 16:02:37.31 LOGIN   SUBPROCESS   GONE      MILANT      000000D5
1-NOV-2000 16:06:36.40 LOGFAIL LOCAL        SUPER      MBILLS     000000E5 _TTA1:
:
:
```

例 9-5 に、完全な形式の監査レポートから 1 行のレコードを抜き出して示します。レポートを生成する ANALYZE/AUDIT コマンドでは、使用する監査ログ・ファイルの名前に置き換えます。

#### 例 9-5: 完全な監査レポートの 1 つのレコード

---

```
$ ANALYZE/AUDIT/FULL SYS$MANAGER:SECURITY.AUDIT$JOURNAL
Security audit (SECURITY) on FNORD, system id: 19728
Auditable event:      Object access
Event time:           6-AUG-2000 11:54:16.21
PID:                  3D200117
Process name:         Hobbit
Username:             PATTERSON
Process owner:        [ACCOUNTING,PATTERSON]
Terminal name:        RTA1:
Object class name:    LOGICAL_NAME_TABLE
Object name:          LNM$SYSTEM_DIRECTORY
Access requested:     WRITE
Status:               %SYSTEM-S-NORMAL, normal successful completion
Privileges used:      SYSPRV
```

---

例 9-6 に、要約レポートを示します。レポートを作成する ANALYZE/AUDIT コマンドでは、使用する監査ログ・ファイルの名前に置き換えます。

#### 例 9-6: 監査ログ・ファイルのイベントの要約

---

```
$ ANALYZE/AUDIT/SUMMARY SYS$MANAGER:SECURITY.AUDIT$JOURNAL
Total records read:      9701      Records selected:      9701
Record buffer size:     1031
Successful logins:       542      Object creates:         1278
Successful logouts:      531      Object accesses:        3761
Login failures:          35       Object deaccesses:      2901
Breakin attempts:        2        Object deletes:         301
System UAF changes:      10       Volume (dis)mounts:     50
Rights db changes:        8        System time changes:    0
Netproxy changes:         5        Server messages:        0
Audit changes:            7        Connections:            0
Installed db changes:     50       Process control audits: 0
Sysgen changes:           9        Privilege audits:       91
NCP command lines:       120
```

---

### 9.5.4 会話形式での監査分析ユーティリティの使用

ターミナルに出力を送信する場合は、監査ログ・ファイルを会話形式で分析できます。リスト表示中の任意の時点で Ctrl/C を押すことにより、表示されているレポートを中断できます。これにより、自動的に完全なリスト表示が開始され、Command> プロンプトが表示されます。コマンド・モードでは、レポート内で先に進んだり前のレコードに戻って、レコードを詳細に調べることができます。

Command> プロンプトでは、『*OpenVMS* システム管理ユーティリティ・リファレンス・マニュアル』の任意の ANALYZE/AUDIT コマンドを入力して、分析基準の変更、監査レポート内での位置の変更、または完全な表示と簡略表示の切り替えを行うことができます。監査レポートの表示に戻るには、CONTINUE コマンドを入力します。

### 9.5.5 レポートの調査

監査ログ・ファイルの定期的な分析で、(実際の侵入、侵入の試み、ログイン失敗の繰り返しなどの疑わしいセキュリティ・イベントにより)システムのセキュリティが危険にさらされている疑いが見つかった場合には、セキュリティ監査ログ・ファイルをより詳しく調査することによって、セキュリティ・イベントの発生源を調査できます。

たとえば、前日の監査レポートの定期的な調査中に、例 9-7 に示すセキュリティ・イベントが確認されたとします。

例 9-7: 監査レポートにある疑わしい活動の 特定

Date / Time	Type	Subtype	Node	Username	ID	Term
-----						
:						
:						
26-OCT-2000 16:06:09.17	LOGFAIL	REMOTE	BOSTON	KOVACS	5BC002EA	_RTA14:
26-OCT-2000 16:06:220.01	LOGFAIL	REMOTE	BOSTON	KOVACS	5BC002EA	_RTA14:
26-OCT-2000 16:06:34.17	LOGFAIL	REMOTE	BOSTON	KOVACS	5BC002EA	_RTA14:
26-OCT-2000 16:06:450.50	LOGFAIL	REMOTE	BOSTON	KOVACS	5BC002EA	_RTA14:
26-OCT-2000 16:07:12.39	LOGIN	REMOTE	BOSTON	KOVACS	5BC002EA	_RTA14:
26-OCT-2000 16:23:42.45	SYSUAF	SYSUAF_ADD	BOSTON	KOVACS	5BC002EA	_RTA14:
:						
:						

例 9-7 のレポートに表示されているセキュリティ・イベントは、ユーザ Kovacs が、ログインに 4 回失敗した後でシステムにログインしたことを示しています。ユーザ Kovacs はログインしてすぐに、システム・ユーザ登録ファイル (SYSUAF.DAT) に新しいアカウントを作成しています。

この時点で、この行動が正常か異常かを判断する必要があります。そのためには、システムに新しいユーザ・アカウントを追加する権限がユーザ Kovacs にあるかどうかを考慮します。システムのセキュリティが危険にさらされていると考えられる場合は、次のコマンドを使用してセキュリティ監査ログ・ファイルからより詳細なレポートを生成し、システムが損害を受けているかどうかを判断します。

```
$ ANALYZE/AUDIT/FULL/SINCE=01-JUN-2003:16:06
```

この例のコマンドは、ユーザ Kovacs が最初にシステムへのログインを試みた時点から監査ログ・ファイルに書き込まれたすべてのセキュリティ監査イベントの完全なレポートを生成します。完全な形式のレポートでは、監査ログ・ファイルにある各レコードの全データが表示されます。完全なレポートを使用することで、例 9-8 に示すように、ローカルの KOVACS のアカウントにログインした遠隔ユーザの名前と、ログイン元のノードを判定することができます。

#### 例 9-8: 疑わしいレコードの調査

---

```
.
.
.
Security alarm (SECURITY) and security audit (SECURITY) on BOSTON,
                                system id: 20011
Auditable event:                Remote interactive login failure
Event time:                    01-JUN-2003 16:06:09.17
PID:                           5BC002EA
Username:                      KOVACS
Terminal name:                 _RTA14:
Remote nodename:               NACHWA           Remote node id:       7300
Remote username:               FOLLEN
Status:                        %LOGIN-F-INVPWD, invalid password
.
.
.
Security alarm (SECURITY) and security audit (SECURITY) on BOSTON,
                                system id: 20011
Auditable event:                Remote interactive login
Event time:                    01-JUN-2003 16:07:120.39
PID:                           5BC002EA
Username:                      KOVACS
Terminal name:                 _RTA14:
Remote nodename:               NACHWA           Remote node id:       7300
Remote username:               FOLLEN
```

---

例 9-8 に表示されている情報は、ログインの失敗とその後のログイン成功が、遠隔ノード NACHWA からユーザ Follen により行われたことを示しています。次のステップは、セキュリティ・イベントがユーザ Follen に起因するものか、FOLLEN のアカウントを利用して遠隔ノード NACHWA に侵入した誰かに起因するものかを判断します。

## 9.6 監査サブシステムの管理

この節では、監査システムの管理方法を説明します。管理タスクには、次の作業が含まれます。

- 監査サーバ・プロセスのスタートアップの有効化および無効化

- スタートアップにおける，オペレーティング・システムが監査を開始するポイントの変更
- プロセス中断のきっかけとなる未処理メッセージの数の指定
- メモリの枯渇に対する監査サーバの対応の指定
- メッセージの正確なタイムスタンプ設定の維持
- システム監査バッファからディスクへのメッセージ転送の調整
- システム監査ログに定期的に割り当てられるディスク容量の指定

### 9.6.1 監査サーバにより実行されるタスク

オペレーティング・システムは，システム・スタートアップ時に独立プロセスとして監査サーバを作成し，次のタスクを実行します。

- SYS\$COMMON:[SYS\$MGR] での，クラスタ・ワイド・セキュリティ監査ログ・ファイル (SECURITY.AUDIT\$JOURNAL) の作成
- ログ・ファイルへのセキュリティ・イベントの記録と，セキュリティ・クラス・メッセージの受信が有効になっている任意のオペレータ・ターミナルへのアラーム配信の制御
- サイト定義のセキュリティ・イベント・セットの監査の有効化
- ディスクとメモリの資源の監視
- セキュリティ監査特性のデータベースの維持

監査サーバは，オペレータ通信マネージャ (OPCOM) に情報メッセージとエラー・メッセージを送信します。OPCOM はこれらのメッセージをオペレータ・ターミナルにブロードキャストし，メッセージをオペレータ・ログ・ファイルに書き込みます。

例 9-9 に，監査サーバの初期の動作値を示します。これらの設定は，監査サーバ・データベースである，SYS\$COMMON:[SYS\$MGR] の VMS\$AUDIT\_SERVER.DAT に保存されます。DCL の SET AUDIT コマンドを使用してセキュリティ監査の特性を変更するたびに，監査サーバ・データベースが更新されます。また，システムのリブートのたびに，システムはこのデータベースから監査の値を取得します。

#### 例 9-9: 監査サーバのデフォルトの特性

---

```
$ SHOW AUDIT/ALL
List of audit journals:
Journal name:          SECURITY
Journal owner:         (system audit journal)
Destination:          SYS$COMMON:[SYS$MGR] SECURITY.AUDIT$JOURNAL
Monitoring:            enabled
Warning thresholds,    Block count:      100    Duration:    2 00:00:00.0
Action thresholds,     Block count:      25    Duration:    0 00:30:00.0
```

### 例 9-9: 監査サーバのデフォルトの特性 (続き)

---

```
Security auditing server characteristics:
  Database version:      4.4
  Backlog (total):      100, 200, 300
  Backlog (process):    5, 2
  Server processing intervals:
    Archive flush:      0 00:01:00.00
    Journal flush:      0 00:05:00.00
    Resource scan:      0 00:05:00.00
  Final resource action: purge oldest audit events

Security archiving information:
  Archiving events:      none
  Archive destination:

System security alarms currently enabled for:
  ACL
  Authorization
  Breakin:      dialup,local,remote,network,detached
  Logfailure:   batch,dialup,local,remote,network,subprocess,detached,server

System security audits currently enabled for:
  ACL
  Authorization
  Breakin:      dialup,local,remote,network,detached
  Logfailure:   batch,dialup,local,remote,network,subprocess,detached,server
```

---

## 9.6.2 監査サーバのスタートアップの無効化と再有効化

オペレーティング・システムはすべてデフォルトで監査サーバ・プロセスと OPCOM を起動します。

システムの物理メモリまたはディスク・ストレージ領域が特に限定されていて、かつセキュリティ関連イベントのログ記録が重要でない場合は、システム・スタートアップ・プロシージャから監査サーバと OPCOM のプロセスを削除できます。ただし、削除の前に、クラスタ・オブジェクトのサポートには監査サーバが必要である点に注意してください(第 11 章を参照)。次の例に、システム管理ユーティリティ (SYSMAN) を使用して、これらのプロセスを削除する方法を示します。

```
$ SET PROCESS/PRIVILEGES=(OPER,BYPASS)
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> STARTUP SET DATABASE STARTUP$STARTUP_VMS
SYSMAN> STARTUP DISABLE FILE VMS$CONFIG-050_OPCOM.COM/NODE=*
SYSMAN> STARTUP DISABLE FILE VMS$CONFIG-050_AUDIT_SERVER.COM /NODE=*
SYSMAN> EXIT

$ SET PROCESS/PRIVILEGES=(NOOPER,NOBYPASS)
```

監査サーバ・プロセスを削除し、システム上のセキュリティ監査をシャット・ダウンするには、クラスタの各ノードで次のコマンドを入力します。

```
$ SET AUDIT/ALARM/AUDIT/DISABLE=ALL/CLASS=*
$ SET AUDIT/SERVER=EXIT
```

システム上のセキュリティ監査と OPCOM を再起動するには、次の DCL コマンド行を入力します。

```
$ @SYS$SYSTEM:STARTUP OPCOM
$ @SYS$SYSTEM:STARTUP AUDIT_SERVER
```

以降のすべてのシステムのブート時に OPCOM と監査サーバ・プロセスを起動するには、システム・スタートアップ・プロシージャに加えた編集をすべて元に戻します。次の SYSMAN コマンドを使用します。

```
$ SET PROCESS/PRIVILEGES=(OPER,BYPASS)
$ RUN SYS$SYSTEM:SYSMAN
SYSMAN> STARTUP SET DATABASE STARTUP$STARTUP_VMS
SYSMAN> STARTUP ENABLE FILE VMS$CONFIG-050_OPCOM.COM/NODE=*
SYSMAN> STARTUP ENABLE FILE VMS$CONFIG-050_AUDIT_SERVER.COM -
_SYSMAN> /NODE=*

SYSMAN> EXIT
```

```
$ SET PROCESS/PRIVILEGES=(NOOPER,NOBYPASS)
```

SYSMAN の詳細については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照してください。

### 9.6.3 スタートアップにおける、オペレーティング・システムが監査を開始するポイントの変更

通常、オペレーティング・システムは SYSTARTUP\_VMS.COM の実行の直前に、監査イベント・メッセージの送信を開始します。しかし、スタートアップ時に監査イベント・メッセージの受信を必要としないサイトでは、論理名 SYS\$AUDIT\_SERVER\_INHIBIT を再定義することにより、この動作を変更できます。

オペレーティング・システムによるセキュリティ・イベント・メッセージの配信開始ポイントを変更するには、SYS\$MANAGER:SYLOGICALS.COM コマンド・プロシージャに、次の行を追加します。

```
$ !
$ DEFINE /SYSTEM /EXECUTIVE SYS$AUDIT_SERVER_INHIBIT yes
$ !
```

システム管理者は、SYSTARTUP\_VMS 終了時点など、システム・スタートアップの別の段階を選択して監査を開始することができます。ただし、システムへの一般のログインを許可する前（つまり、すべての SET LOGINS/INTERACTIVE コマン



ドの前) には、必ず監査を開始してください。監査メッセージの配信を開始するには、適切なコマンド・ファイルに次の行を追加します。

```
$ !  
$ SET AUDIT/SERVER=INITIATE  
$ !
```

#### 9.6.4 プロセス中断のきっかけとなる未処理メッセージの数の指定

監査サーバがメッセージの流入を制御している場合を除き、ある条件下では、メモリ不足になる可能性があります。非常に遅い入出力デバイス、ディスク領域の問題、または突然のメッセージの大量発生により、メッセージをディスクに書き込むサーバの能力が対応できなくなることがあります。メモリの枯渇を防ぐために、監査サーバは未処理メッセージの総数を常時監視し、アクティブな各プロセスにより生成されるメッセージの数を数えます。サーバは、ディスクに記録可能な量を超えるイベントを受信した場合、監査イベントを生成しているプロセスに対して、フロー制御の適用を開始します。

##### 9.6.4.1 メッセージのフロー制御

メッセージの量は、プロセス単位で制御されます。表 9-7 に、フロー制御の 3 つの段階を示します。

表 9-7: 監査イベント・メッセージのフロー制御

制御の段階	メッセージの合計バックログ (デフォルト)	プロセス・バックログの上限 (デフォルト)
1	100	5
2	200	2
3	300	なし

1. メモリ内に 100 個のメッセージがある場合は、オペレーティング・システムは 5 つ以上の未処理メッセージを持つプロセスをすべて一時中断します。プロセスの全メッセージがログ・ファイルに書き込まれれば、プロセスは処理を再開できます。
2. メモリ内に 200 個のメッセージがある場合は、全メッセージがディスクに書き込まれるまで、オペレーティング・システムは 2 つ以上のメッセージを送信したプロセスをすべて一時中断します。
3. メモリ内に 300 個のメッセージがある場合は、全メッセージがディスクに書き込まれるまで、メモリ内にメッセージを持つすべてのプロセスが一時中断されます。

SET AUDIT コマンドに /BACKLOG 修飾子を指定することで、メッセージを制御するためのサイト固有の値を設定することができます。たとえば、次のコマンドを

使用すると、キュー内に 125 個の未処理メッセージがあり、かつ生成側プロセスに 8 個の未処理メッセージが発生した時点で、オペレーティング・システムがメッセージ流入の制御を開始するように、アクションしきい値を上げます。

```
$ SET AUDIT/BACKLOG=(TOTAL=(125,250,350),PROCESS=(8,4) )
```

#### 9.6.4.2 プロセスの一時中断の防止

当然ながら、オペレーティング・システムはいくつかの重要なプロセスは一時中断しません。リアルタイム・プロセスと、次のプロセスはすべて一時中断を免除されています。

---

CACHE_SERVER	CLUSTER_SERVER
CONFIGURE	DFS\$COM_ACP
DNS\$ADVER	IPCACP
JOB_CONTROL	NETACP
NET\$ACP	OPCOM
REMACP	SHADOW_SERVER
SMISERVER	SWAPPER
TP_SERVER	VWS\$DISPLAYMGR
VWS\$EMULATORS	

---

プロセスを一時中断の対象から除外するには、プロセス識別子 (PID) をプロセス除外リストに追加します。次の形式の SET AUDIT コマンドを使用します。

```
SET AUDIT/EXCLUDE=process-id
```

プロセスがシステムからログアウトしても、プロセス (PID) はプロセス除外リストから自動的に削除されない点に注意してください。除外リストからプロセスを削除するには、SET AUDIT/NOEXCLUDE コマンドを使用します。オペレーティング・システムによって除外されているプロセスは削除できません。

#### 9.6.5 メモリ不足への対応

除外リスト上のプロセス (9.6.4.2 項を参照) があまりに多くの監査メッセージを生成するために監査サーバがメモリ不足になった場合、監査サーバはデフォルトの動作として、メモリが使用できるようになるまで古いイベント・メッセージを削除します。これにより、最新のメッセージが保存されます。

監査サーバには、メモリ不足に陥った場合に使用できる次の代替策もあります。

オプション	説明
Crash	監査サーバがメモリ不足になった場合、システムをクラッシュさせます。
Ignore_New	メモリが使用できるようになるまで、新しいイベント・メッセージを無視します。新しいイベント・メッセージは失われますが、メモリ内のイベント・メッセージは保存されます。
Purge_Old (デフォルト)	最新メッセージのために、メモリが使用できるようになるまで、古いイベント・メッセージを削除します。

監査サーバのデフォルトの動作を変更し、古いメッセージをパージするのではなく、新しい監査メッセージをすべて無視するよう監査サーバに指示するには、次のコマンドを入力します。

```
$ SET AUDIT/SERVER=FINAL_ACTION=IGNORE_NEW
```

監査サーバは、仮想メモリの上限 (PGFLQUOTA) が 20,480 ページに制限された状態で動作します。これは、システムにインストールされているページ・ファイルのサイズによりさらに制限される場合があります。AUTOGEN を実行することで、ページ・ファイルのサイズを調整できます。AUTOGEN は、ページ・ファイルの問題を検出すると、必ず自動的にサイズをリセットして、問題を解消します。

### 9.6.6 メッセージの正確なタイムスタンプ設定の維持

発生順序が重要であるセキュリティ・イベントのセットを監査している場合、クラスタ内のすべての時計が同期している必要があります。これにより、クラスタにおける全ノードのメッセージのタイムスタンプが、イベントの発生順序を厳密に反映するようになります。

クラスタ構成内の各ノードは独立して時刻を維持するため、時間の経過とともにクラスタの時刻にずれが生じる可能性があります。時刻のずれを防止するには、定期的に SYSMAN コマンドの CONFIGURATION SET TIME を使用します。

『OpenVMS システム管理ユーティリティ・リファレンス・マニュアル』に、時計の同期を 1 秒以内に保つために 1 時間ごとに実行可能なコマンド・プロシージャの例を示します。

### 9.6.7 ディスクへのメッセージ転送の調整

監査サーバはメモリ内にセキュリティ・イベント・メッセージを保存し、メッセージのグループを、バッファからディスク上の監査ログ・ファイルに定期的に転送します。通常、監査サーバは監査メッセージを 5 分ごとに転送し、アーカイブされたメッセージ (9.4.3.1 項を参照) を毎分 1 回転送します。高いセキュリティが必要と

される一部の環境と、システム上で大量の監査メッセージが生成される場合を除いて、このデフォルトで十分なはずです。

高いセキュリティが必要なサイトでは、ログ転送操作の間隔を変更することにより、通常よりも高い頻度でディスクにイベント・メッセージを転送できます。たとえば、次のコマンドを使用して、監査サーバが2分ごとに監査ログ・ファイルにイベント・メッセージを書き込むように、監査サーバの特性を変更します。

```
$ SET AUDIT/INTERVAL=JOURNAL_FLUSH=00:02
```

ただし、メッセージの転送が頻繁に行われれると、監査サーバ・プロセスに関連付けられているシステム・バッファへのメッセージの保存よりも、入出力操作の方が多くなるため、システム性能が影響を受けます。

直ちにすべての監査メッセージを強制的にログ・ファイルに書き込むには、次のコマンドを入力します。

```
$ SET AUDIT/SERVER=FLUSH
```

### 9.6.8 監査ログ・ファイル用のディスク領域の割り当て

監査サーバは、セキュリティ監査ログ・ファイルに割り当てられているディスク領域を常時監視して、イベント・メッセージ用に十分な領域があることを確認します。使用可能なブロックが不足してくると、監査サーバは監査ログ・ファイルを拡張します。ディスク資源の制約により、サーバがログ・ファイルに対して、これ以上ブロックを割り当てることができない場合、サーバは次のいずれかの措置を取ります。

- オペレータ・ターミナルに警告メッセージを送信することによって警告を發します。これは、使用できるディスク・ブロックが100以下の場合にデフォルトで行われます。

次のコマンドは、使用できるブロックが150個になった時点で警告が出るように、デフォルトを変更します。

```
$ SET AUDIT /JOURNAL=SECURITY /THRESHOLD=WARNING=150
```

- 監査レコードを生成しているプロセスを一時中断する措置を取ります。一部のプロセスはこの措置の対象外です(9.6.4.2項を参照)。ログ・ファイルに対する資源の監視が有効になっている場合、プロセスが一時中断されるのは、使用できるディスク・ブロックが25以下の場合です。

アクションしきい値を50ブロックに変更するには、次のコマンドを入力します。

```
$ SET AUDIT /JOURNAL=SECURITY /THRESHOLD=ACTION=50
```

しきい値は、ブロックまたはデルタ時間として表現します。デルタ時間に平均の領域消費速度をかけることで、ブロックの数が算出されます。ブロックと時間のしきい値の最大値が、アクティブなしきい値として使用されます。

## 9.6.9 監査機能におけるエラー処理

OpenVMS のセキュリティ監査機能により消費される資源は、記録されるシステム・イベントの数とタイプによって異なります。監査機能に関連して、次の3つの異なるエラー状態が発生し得ます。

- 監査サーバのメモリ不足。9.6.5 項に、この状況に対応するためのさまざまな方法を説明しています。
- 監査ログ・ファイルを保存するディスクの領域不足。
- 遠隔ログ・ファイル(アーカイブ・ファイル)用のネットワーク接続の切断。

この節では、ディスク領域を監視し、アーカイブ・ファイルにログを記録する際の、監査システムのデフォルトの動作を説明します。

### 9.6.9.1 ディスク監視の無効化

監視サーバは監査ログ・ファイルを監視し、着信イベント・メッセージ用に十分な領域を確保するために、定期的にディスク・ブロック割り当てを事前に拡張します。ディスク領域が不足すると、サーバはまずオペレータ・メッセージによって警告を発してから、一部の生成側プロセスを一時中断する措置を取ります(9.6.8 項を参照)。明確な理由なく多数のプロセスが中断されている場合は、おそらく監査ディスクがいっぱいであることが原因です。ディスク領域の問題を是正したら、SET AUDIT/SERVER=RESUME コマンドを使用して、(次回の資源のスキャンを待たずに)中断されていたプロセスを再開できます。

次のコマンドを入力すると、資源の監視を完全に無効にすることができます。

```
$ SET AUDIT/JOURNAL=SECURITY/RESOURCE=DISABLE
```

ただし、ディスク資源の監視を無効にすると、手遅れになるまで、警告メッセージを受信する機会がなくなります。9.6.4 項で説明しているように、監査サーバは生成する監査が多すぎるプロセスの一時中断を開始します。また、メモリ不足になった場合は、9.6.5 項で説明しているように、メッセージの無視、古いメッセージのページ、また場合によってはシステムをクラッシュさせるという措置を取ります。

ディスク領域が再び使用可能になると、監査サーバはログ・ファイルを拡張し、中断されていたプロセスを再開します。

### 9.6.9.2 遠隔ログ・ファイルへのリンクの喪失

遠隔ログ・ファイルに監査メッセージを書き込む場合、9.4.3.1 項で説明しているように、ローカル・ノードと遠隔ノードの間のリンクに障害が発生する場合があります。この障害が発生すると、監査サーバは全オペレータ・ターミナルに警告メッセージをブロードキャストし、接続されるまで1分ごとにリンクの再確立を試みます。



## システムのセキュリティ侵害

セキュリティ・ポリシーの策定，およびそのポリシーを実装するための適切なセキュリティ対策の選択に加えて，サイトでは，システム，サイト，ネットワークに対する侵害行為に対処するための手順を確立し，テストする必要があります。その手順は，次の2つの領域を対象に作成します。

- 侵害の疑いがある場合，または侵害が確認された場合の適切な対応。サイトのガイドラインは，サイトのセキュリティを強化する（侵害行為拡大のあらゆる可能性を排除する）かどうか，侵害者を捕まえるための事前対策を講じるかどうか，そして刑事訴訟や民事訴訟を起こすための証拠を収集するかどうかを決めるための判断基準とならなければなりません。決定事項については，それぞれ個別にルールやガイドラインを設定します。
- 侵害の疑いまたは侵害のある場合に必要となる可能性のある，サイト外部の連絡先と資源。たとえば，企業によっては，地域や都道府県，国の各種関連機関（該当する場合のみ）や最寄りの電話会社（セキュリティ部門），HP サポート・グループについてすぐに連絡が取れるようにしておく必要があります。<sup>1</sup>

この章では，システムに対する進行中の攻撃またはすでに発生した攻撃を検出する方法，およびとりうる対抗策について説明します。

### 10.1 システム攻撃の形態

セキュリティ管理者は，定期的にシステムを監視して，セキュリティ侵害の恐れがないか確認しなければなりません。よくあるシステム攻撃の形態は次のとおりです。

- アクセス回線の探索
- パスワードの探索
- 侵入行為
- ユーザ登録ファイル (UAF) のレコードの改ざんまたはねつ造
- 本来与えられていない特権の付与や盗用
- 外見上は無害ながらも，ユーザ・パスワードを盗んだり，システムに損害を与えたりすることを目的とするソフトウェア（トロイの木馬ソフトウェア）のインストール

<sup>1</sup> HP サポート・グループには，米国にある Software Security Response Team (SSRT) の他，European Security Program Office (ESPO) などがあります。

- 特権アカウントへのアクセスを目的とした，コマンド・プロシージャやプログラムへのウィルスの混入
- ディスクに対するスカベンジング
- 別のノードに対するゲートウェイとしてのノードの使用

## 10.2 問題の兆候

システムに脆弱性が存在し攻撃を受けている可能性がある場合，最初の兆候は次から得られる情報によって気づくと考えられます。

- ユーザからの報告
- システムの監視。たとえば，次のような兆候が考えられます。
  - アプリケーションや通常のプロセスにおける，説明のつかない変化や動作
  - OPCOM または監査サーバから送られてきた，説明のつかないメッセージ
  - システム登録データベースに登録されているユーザ・アカウントへの，説明のつかない変更（特権の変更，保護設定，優先順位，クォータ）

### 10.2.1 ユーザからの報告

ユーザによってシステム・セキュリティの問題が発見されることは少なくありません。ユーザは，次のような状況で管理者に連絡することが考えられます。

- ファイルが無くなっている。
- ユーザが実行した覚えのない正常なログインや説明のつかないログイン失敗など，最後のログイン・メッセージが説明のつかない内容になっている。
- ユーザがログインできない。この場合は，前回の正常なログイン以降ユーザ・パスワードが変更された可能性があるなど，何らかの不正な操作が行われた恐れがあります。
- 侵入回避機能が有効になっているようで，ユーザがログインできない。
- SHOW USERS コマンドの報告では別のターミナルからログインしているはずのユーザが，実際にはログインしていない。
- ユーザが開始したことのないプロセスについて，ジョブが切断されたことを示すメッセージがログイン時に表示される。
- ユーザのディレクトリに，該当ユーザが作成した覚えのないファイルが存在する。
- ユーザ・ファイルの保護または所有権について，説明のつかない変更が見つかっている。
- ユーザがリストを要求していないにもかかわらず，そのユーザ名で作成されたリストが存在する。



- ダイアルアップ回線など，利用可能なリソースが突然減少している。

上記のいずれかが報告されたら，速やかに対処します。まず，報告された状況が間違いないかどうかを確認します。間違いない場合は，原因を突き止めて解決策を探します。

## 10.2.2 システムの監視

6.7 節に，システムに対するセキュリティ侵害の可能性の有無を判別するのに役立つ作業のリストを示します。次のリストは，前述のリストの作業を実施しているときに明らかになる可能性のある，警戒すべき兆候をまとめたものです。

- SHOW USERS の報告に，現在ログインしているはずのないユーザが表示されている。
- システムの負荷や性能に，説明のつかない変化が見られる。
- メディアやプログラムのリストが消失していたり，物理的セキュリティが低下している兆候が見られる。
- 施錠されているファイル・キャビネットが不正に開けられ，権限を持つユーザのリストが紛失している。
- システムの実行可能イメージ・ライブラリである [SYSEXEC] または [SYSLIB] に，見覚えのないソフトウェアがある。
- MONITOR SYSTEM レポートを見ると，見覚えのないイメージが実行されている。
- DCL の SHOW USER コマンドを入力すると，権限のないユーザの名前が表示される。登録ユーティリティ (AUTHORIZE) で SHOW コマンドを使って生成したリストを調べたところ，システムへのアクセス権限がユーザに与えられていた。
- 権限を与えていない代理ユーザが見つかった。
- 会計レポートを見ると，最近，通常ではあり得ない量の処理が行われており，外部からのアクセスが原因と見られる。
- 説明のつかないバッチ・ジョブがバッチ・キューに登録されていた。
- SHOW DEVICE コマンドを入力したところ，説明のつかないデバイス割り当てが存在する。
- 通常ではあり得ない時間帯に大量の処理が発生している。
- 重要なファイルの保護コードやアクセス制御リスト (ACL) が変更されている。識別子が追加されているか，識別子の保持者がライト・リストに追加されている。
- 離職率が高い，または従業員の士気が低下している。

上記の状態はいずれもさらなる調査を必要とします。すでに問題が発生していることを示すものもあれば、簡単に説明のつくものもあるかもしれません。また、重大な問題につながる可能性を示すものもあります。

## 10.3 システムの定期的な監視

OpenVMS には、システムの活動を体系的に監視するための仕組みが数多く備わっています。システムを監視するための仕組みは、次に示すように、手動のもの、または、ユーザの作成したコマンド・プロシージャを使用するものなどが多数存在します。

- 会計情報ユーティリティ (ACCOUNTING)
- 登録ユーティリティ (AUTHORIZE)
- インストール・ユーティリティ (INSTALL)
- システム管理ユーティリティ (SYSMAN)

これらのユーティリティを適切に使用することで、設定を確認し、問題発生の警告を受け取り、対処することができます。この節では、システム監視機能の中でも最も重要な ACCOUNTING と ANALYZE/AUDIT について説明します。

### 10.3.1 システムの会計記録

会計情報ユーティリティ (ACCOUNTING) のレポートを調べることで、リソースの平常時の利用パターンを把握することができます。レポートを得るには、ユーティリティ・イメージ SYS\$SYSTEM:ACC.EXE を実行します。実行結果のデータ・ファイルは SYS\$MANAGER:ACCOUNTNG.DAT です。ACCOUNTING レポートに、問題の初期の兆候が記録されている可能性があるので調べます。次の点を確認します。

- 見覚えのないユーザ名
- 特定の時間帯や曜日における異常な活動など、普段見られない利用パターン
- 通常では考えにくい量のリソースの使用
- ネットワーク・ノードやリモート・ターミナルなど、普段見られないログイン元

### 10.3.2 セキュリティ監査の実施

セキュリティ管理者は、DCL の SET AUDIT コマンドを使用して監査対象のイベント・カテゴリを有効にして、セキュリティに関わる活動をオペレーティング・システムに報告させることができます。Audit Analysis ユーティリティ (ANALYZE/AUDIT) を使用することで、セキュリティ監査ログ・ファイルに収集されたイベント・メッセージを定期的に調べることができます。詳細については、第 9 章 を参照してください。

OpenVMS は、イベント・メッセージを監査ログ・ファイルに記録したり、オペレータ・ターミナル宛に送信したりできます。イベントを監査情報として報告させるか、アラームとして伝わるようにするかを、次の方法で指定します。

- 通常、セキュリティ関連のイベントについてはアラームではなく監査を有効にします。これは、監査レコードをシステム・セキュリティ監査ログに記録しておけば、まとめて調べたり、あとで参照するためにアーカイブしておけるためです。監査メッセージは、単独ではあまり多くの情報を伝えない可能性があります。量が多ければセキュリティ違反のパターンが浮かび上がります。たとえば、オブジェクトに対するアクセスを監査すれば、アクセスの時間、アクセス対象オブジェクトの種類、その他のシステム情報についてパターンが見られ、時間帯ごとのシステムの利用状況を総合的に把握することができます。

ファイル、デバイス、ボリュームへのアクセスの失敗について監査を有効にするには、次のコマンドを入力します。

**\$ SET AUDIT/AUDIT/ENABLE=ACCESS=FAILURE/CLASS=(FILE,DEVICE,VOLUME)**

このコマンドを実行すると、失敗したアクセスを示すイベントがセキュリティ監査ログに記録されますが、オペレータ・ターミナルにアラームは送信されません。

- リアル・タイム・イベントや、侵入行為、システム・ユーザ登録ファイル (SYSUAF.DAT) の改ざんなど、即時の確認が必要なイベントについてセキュリティ・アラームを有効にします。たとえば、既知ファイル・リストに対する変更とシステム時刻の変更についてアラームを有効にするには、次のコマンドを入力します。

**\$ SET AUDIT/ALARM/ENABLE=(INSTALL,TIME)**

このコマンドを実行すると、オペレータ・ターミナルにイベント・メッセージが送信されます。アラームのハードコピーを保存しておくには、ハードコピー・オペレータ・ターミナルを使用するか、またはイベントをアラームおよび監査の両方として有効にします。

セキュリティ監査はシステムの性能に影響を与えるため、もっとも重要なイベントについてのみ監査を有効にします。次に示すセキュリティ監査措置は、重要性が高く、システム・コストが低いものから順に掲載してあります。

1. ログインの失敗と侵入についてセキュリティ監査を有効にします。これは、部外者による詮索行為（および部内者によるアカウント探索）を検出するのに最適の方法です。セキュリティを施す必要のあるサイトはすべて、これらのイベントについてアラームを有効にします。
2. ログインについてセキュリティ監査を有効にします。リモート・ユーザやダイヤルアップ・ユーザなど、より疑わしいアクセス元からの正常ログインの監査

は、使用中のアカウントの追跡に最も適しています。特権アカウントにログインするユーザが身元の偽装が可能になる前に監査レコードが書き込まれます。

3. ファイル・アクセスの失敗 (ACCESS=FAILURE) についてのセキュリティ監査を有効にします。この方法は、あらゆるファイル保護違反を監査するため、詮索行為の把握に適した方法です。
4. ACL ベースのファイル・アクセス監査を実施して、重要なシステム・ファイルに対する書き込みアクセスを検出します。監査対象にする必要のある最も重要なファイルについては、表 10-1 を参照してください。(表 9-2 は、ACL でセキュリティ関連のエントリを作成する方法の例です)。これらのファイルへのアクセスの成功のみを監査して侵入行為を検出したり、アクセスの失敗も監査して詮索行為を検出することもできます。

表 10-1 に示したファイルの一部は、通常のシステム動作時にも書き込みが行われます。たとえば、SYSUAF.DAT はログインがあるたびに書き込みが行われ、SYSMGR.DIR はシステム・ブート時に書き込みが行われます。

**表 10-1: ACL ベースの監査が有効なシステム・ファイル**

デバイスおよびディレクトリ	ファイル名
SYS\$SYSTEM	AUTHORIZE.EXE
	F11BXQP.EXE
	LOGINOUT.EXE
	DCL.EXE
	JOBCTL.EXE
	SYSUAF.DAT
	NETPROXY.DAT
	RIGHTSLIST.DAT
	STARTUP.COM
	VMS\$OBJECTS.DAT
SYS\$LIBRARY	SECURESHR.EXE
	SECURESHRP.EXE
SYS\$MANAGER	VMS\$AUDIT_SERVER.DAT
	SY*.COM
	VMSIMAGES.DAT
SYS\$SYSROOT	[000000]SYSEXEC.DIR
	[000000]SYSLIB.DIR

表 10-1: ACL ベースの監査が有効なシステム・ファイル (続き)

デバイスおよびディレクトリ	ファイル名
	[000000]SYS\$LDR.DIR
	[000000]SYSMGR.DIR

5. システム・パラメータまたは既知ファイル・リストの改ざんに対するセキュリティ監査を有効にします (/ENABLE=(SYSGEN,INSTALL))。
6. ファイル・アクセス (書き込みアクセスまたはあらゆる種類のアクセス) の特権の利用を監査します。キーワード ACCESS=(SYSPRV,BYPASS,READALL,GRPPRV) を使用して、セキュリティ監査を実装します。メールの配信やオペレータによるバックアップなど、通常のシステム操作において特権が使用されることが多いため、このクラスの監査処理では、監査に伴う出力が大量になる可能性があります。

9.3 節 では、推奨される監査内容の組み合わせについて取り上げています。

## 10.4 セキュリティ侵害への対処

セキュリティ侵害に対処する場合、侵害が実際に発生したか、その試みがあったかに関係なく、セキュリティ管理者は 4 つの段階を踏むことになります。

1. 問題の検出
2. 実行者の特定
3. セキュリティ違反拡大の防止
4. 損害の修復

以下の節では、侵入の試みがあった場合と侵入を許した場合の両方について、この 4 つの段階を説明します。

どの段階においても、実行者を捕まえたり起訴したりする必要がある場合に備えて、情報やデータを証拠として保全するように要員を教育しておきます。

### 10.4.1 失敗に終わった侵入行為

失敗に終わった侵入行為には、パスワードの推測やファイルを閲覧しようとした行為も含まれます。

#### 10.4.1.1 侵入行為の検出

通常、次に示す情報から侵入行為を検出します。

- 説明のつかないログインの失敗に関するユーザからの報告
- システムの異常な活動やダイアルアップ回線が塞がっている状態

- ログイン失敗，侵入の試み，ファイル保護違反についてのセキュリティ・アラーム
- 侵入データベースの調査

#### 10.4.1.2 実行者の特定

ファイル監査を有効にすることで，ファイル閲覧者の特定が簡単になります。ただし，閲覧行為がネットワーク内の別のノードから開始されている場合は，ファイル保護違反のあった時刻に該当するネットワーク・サーバのログ・ファイル (NETSERVER.LOG) を調査する必要があります。リモート・ノードのセキュリティ管理者と連携して調査を行います。

パスワードを推測しようとしている人物の特定は，ファイル保護違反の場合よりもはるかに困難です。ダイアルアップ回線を使用したアクセスのように，アクセス元が匿名の場合には特に困難です。通常，実行者の特定と侵入の防止は両立が難しく，どちらかを優先する必要があります。システムへの侵入を試みる部外者を確実に特定するには，実行者を特定できるまで侵入の試みを許すことが唯一の方法であることがほとんどです。

#### 10.4.1.3 侵入行為の防止

この種の攻撃に対する防止段階では，侵入試行者が実際にシステムにアクセスできないようにし，以後の試行もより困難となるようにします。

##### パスワードの推測

パスワードの推測が成功する可能性を低くするには，次の対策をとるようにします。

- 適切なパスワードを選択するようユーザを指導します。パスワード・ジェネレータの使用を検討します (7.3.2.4 項 参照)。
- システムへの入り口において，システム・パスワードを有効にする。システム・パスワードは，ユーザの立場からすると多少不便になりますが，詮索行為の拡大からシステムを保護する最適な方法です。すでにシステム・パスワードを有効にしていた場合には，新しいパスワードに変更します (7.3.1.2 項)。
- 侵入試行者が侵入に成功した場合のイベントを把握するため，正常ログインの監査を有効にします (10.3.2 項)。

##### ファイルの閲覧

ファイルの閲覧が成功する可能性を低くするには，次の対策をとるようにします。

- 侵害実行者を特定できる場合は，サイトの方針に従って措置を講じます。
- ファイルに十分な保護を施すことの重要性をユーザに伝え，ユーザ・ファイルの保護状態を調査することを検討します。

- ネットワーク内の他のノードからのファイル閲覧が継続的に発生する場合は、デフォルトの FAL アカウントを削除し、代理ログイン・アカウントを通じて個々のユーザにアクセス権を与えるようにします (12.3.2 項 参照)。

## 10.4.2 成功した侵入

成功したセキュリティ侵害には、パスワードの推測の成功、情報やシステム・リソースの盗みや改ざん、有害なソフトウェアのシステムへの配置などが含まれます。侵入を許した場合、侵害実行者の技能や意図によっては、修復にかなりの時間が必要となる可能性があります。

### 10.4.2.1 成功した侵害実行者の特定

侵害実行者の特定は、侵入への対処の中でも最も困難であることが少なくありません。まず、侵害実行者が登録ユーザなのかそうでないかを明らかにする必要があります。登録ユーザかどうかによって、とるべき予防措置の性質が決まってきます。ただし、部内者と部外者の区別が困難な場合があります。

#### 侵害実行者の特定と予防措置との間のトレードオフ

侵害実行者の特定と以後の攻撃に対する防止措置のどちらを優先するかを決めなければならない場合があります。侵入行為のあった最初の段階で得られたデータでは、侵害実行者をはっきりと特定できないことがよくあります。侵害実行者を特定することが重要な場合、侵入行為を分析するために、引き続き侵入を許すことが必要な場合があります。この場合は、監査内容を増やします。追加情報を得るために、セキュリティ管理者の管理が及んでいるシステム・プロシージャ (SYLOGIN.COM など) にわなを仕掛けるのは 1 つの方法です。また、ファイルが損傷した場合に備えて、即座に復旧できるようシステム・バックアップを作成する頻度も増やします。

#### 部外者の特定

外部からの侵入者を特定することは非常に困難です。侵入者が交換式の通信手段を使用している場合 (ダイヤルアップ回線や公衆データ・ネットワークなど) は特に困難です。DECnet for OpenVMS ソフトウェアには、アクセス元のノードまでネットワークをたどって操作を追跡するのに役立つさまざまな機能が備わっています。ローカル・ターミナルが関係している場合は、物理的な監視が有効な場合もあります。

交換式の通信手段が関係している場合、コンピュータ・セキュリティにおける大きな問題の 1 つとなるのが電話システムそのものです。電話回線または公衆データ・ネットワークによる接続をたどるのは、非常に時間がかかります。電話システムをたどって侵入者を追跡する作業は、月単位の時間を要する可能性があり、警察当局の協力が必要になります。複数の長距離電話サービスを経由している場合、協力を要請する会社の数が増加するため、問題がますます複雑になります。

したがって、外部の侵入者の特定は、継続的かつ重大な金銭的損害を被っている場合でなければ通常は割に合いません。多くの場合、問題の再発を防止する対策に集中した方が有益といえます。

#### 10.4.2.2 システムのセキュリティ保護

侵入を許したあと、システムをセキュリティで保護するために必要な措置は、その侵入の性質と侵入元によって異なります。この節では、講じるべき措置について優先順位の高い順番に紹介します。

1. SYSUAF.DAT、NETPROXY.DAT、NET\$PROXY.DAT および RIGHTSLIST.DAT が損傷している場合は、バックアップを使用して復旧します。または、ファイル・リストを生成して詳細に調査し、不適切なエントリがないかどうか、特権が追加されていないか、UIC が変更されていないかを確認します。SYSUAF.DAT が最初に変更されたのがいつかがはっきりわからない場合は、バックアップ・コピーを使用するか既存のファイルをそのまま使用するかに関係なく、SYSUAF.DAT を注意深く調べます。すべての登録ファイルが安全な状態となるよう、必要な措置を講じます。
2. 侵害実行者は、ファイルの閲覧またはネットワーク内の他のノードからアクセスすることによって、パスワードを発見し、ほとんど使用されていない個人用のアカウントを使用している可能性があります。そこで、アカウントのパスワードを変更し、ユーザ本人を呼び出して新しいパスワードを知らせます。少なくとも、特権アカウントのパスワードはすべて変更します。新しいパスワードとして、すべてのアカウントに同じパスワードを設定しないようにしてください。
3. システムをセキュリティ保護するための措置を講じていても、高度な知識を持った侵入者は、後でシステムへアクセスするための足がかりをすでに埋め込んでいる可能性があります。そのため、場合によっては、バックアップまたは OpenVMS ディストリビューション・キットから、OpenVMS のコンポーネントを選択して復旧する必要があります。侵入者が部外者の場合、システムのすべてのエントリを検証する LOGINOUT.EXE と NETACP.EXE という 2 つのコンポーネントが重要です。

ただし、侵入者が登録ユーザの場合は、バックアップ・コピーを使用してすべてのシステム・ファイルを復旧します。登録ユーザは、エグゼクティブ (SYS.EXE)、ファイル・システム (F11BXQP.EXE)、DCL、およびその他のシステム・ファイルに挿入する、さまざまな種類の不正なソフトウェア・パッチ (トラップ・ドア) を使用することが可能です。侵入者は、特権を持つユーザが使用する可能性のあるソフトウェアやコマンド・プロシージャの中に、有害なソフトウェアを埋め込んでいる恐れがあります。このため、システムが安全であることを確かなものとするには、バックアップを使用してファイルをまるごと復旧する必要があります。特権を使用してインストールされたイメージ (レ



イヤード・プロダクトのものも含む)もトラップ・ドアに使用される可能性があるため、これらのイメージもインストールし直します。別の方法として、攻撃の明白な標的の信頼できるコピーを復旧し、疑わしいイベントを捕捉するために一定期間監査を強化するという方法もあります。

4. 再発を防ぐため、システム・パスワードの利用やパスワード生成、監査の拡大、ファイル保護の強化など、さらなるセキュリティ機能の実装を検討します。

#### **10.4.2.3 侵入を許したあとの復旧**

侵入を許したあとは破壊されたファイルを復元します。システムのデータ全体を復旧するか、または発見した問題点を個別に解決するかのどちらが適切かを判断します。システムに仕掛けられ、依然として存在している可能性のあるウイルスやトロイの木馬のためにパスを作成するような、ファイル保護に対する改ざんを見つけて修復します。



## クラスタのセキュリティ保護

この章では、クラスタ・システムのセキュリティ管理者が考慮すべき事項について説明します。クラスタ・システムとは、さまざまなコンピュータ間で、ディスク、資源、および共通のオペレーティング・システムの共用が可能なハードウェアとソフトウェアを使用するシステムのことを指します。VAX プロセッサで構成されるクラスタは、VAXcluster 環境を構成している、と言います。一方、Alpha プロセッサと VAX プロセッサの両方を含むクラスタは、OpenVMS Cluster 環境を構成している、と言います。クラスタに対するセキュリティ保護を適切に行うには、『*HP OpenVMS Cluster システム*』の内容に精通している必要があります。

『*HP OpenVMS Cluster システム*』では、クラスタ管理者が行うべき作業を説明しています。クラスタ管理者の仕事は、システム管理者の仕事と同じですが、クラスタ管理者は、多数のノードに対して変更を適用する必要があります。クラスタを担当するセキュリティ管理者は、クラスタ管理者と同じトレーニングを受け、同じスキルを持つことが要求されます。クラスタのサイトによっては、一人でセキュリティ管理者とクラスタ管理者の両方の役割を担う場合があります。また、クラスタ管理チームの他に、1 名または複数名のセキュリティ管理者を置くサイトもあります。

セキュリティ管理者とクラスタ管理者の仕事に分けているサイトでは、それぞれの役目の間での調整、協力、連絡が非常に重要となります。これまでの章と同様、この章でもセキュリティ管理者という呼び名は、他に担当している役目に関係なく、システムのセキュリティに対する責任を負う人を指します。

### 11.1 クラスタの概要

クラスタ・システムは、スケーラビリティと可用性が高く、セキュリティ保護された均一なコンピューティング環境を提供します。登録ユーザのセットが 1 つのみ存在し、クラスタの任意のメンバでこれらのユーザがプロセスを実行できることが重要です。

均一なコンピューティング環境を実現するため、クラスタは、クラスタの全メンバで次のコンポーネントが動作していることを前提とします。

- ロック・マネージャ・システム・サービス (\$ENQ/\$DEQ) (分散アプリケーション構築のためのフレームワークを提供)
- ファイルおよびレコードの管理サブシステム (ロック・マネージャを通じた連携)
- バッチ・サービスおよびプリント・サービス

- プロセス制御システム・サービス
- セキュリティ監査システム

クラスタ内では、ユーザの登録データおよびオブジェクトのセキュリティ・プロファイルがすべてのノードの間で整合性がとれていなければなりません。これは、特定のオブジェクトに対する特定のユーザのアクセス要求が発生したときに、各クラスタ・メンバがアクセス制御について同じ判断をする必要があるためです。11.2 節 および 11.3 節 では、単一のセキュリティ管理領域を実現する方法について説明しています。

## 11.2 共通環境の構築

クラスタ内では、各ノードが共通の登録情報を利用してアクセス制御を仲介します。単一のセキュリティ管理領域モデルの場合、権限を有する個人に代わって処理を行うプロセスは、クラスタの管理対象となっているオブジェクトへのアクセスを要求します。すると、調整を行っているノードは、そのノードが保持している共通の登録データベースのコピーと、アクセスを要求されているオブジェクトのセキュリティ・プロファイルと比較して、アクセス要求に対する応答を決定します。このモデルでは、登録情報とオブジェクトのセキュリティ・プロファイルがクラスタ内のすべてのノードの間で整合性がとれている場合のみセキュリティが適用されます。

クラスタ内のデータの整合性を確保するためには、サイトでは次の作業が必要です。

- 共通のデータ・セットを維持する ( 11.2.1 項, 11.2.2 項, および 11.2.3 項を参照)
- システム・パラメータの変更を一括して実行する

LGI システム・パラメータを変更する場合は、システム管理ユーティリティ (SYSMAN) を使用します ( 11.8 節参照)。

### 11.2.1 必須の共通システム・ファイル

単一のセキュリティ管理領域を確立するもっとも簡単な方法は、クラスタにマウントされている 1 台以上のディスクに、表 11-1 に示す各ファイルを 1 つだけ置くことです。あるノードで必須ファイルを作成すると、ただちにそのファイルをそのノード以外のすべてのクラスタ・メンバ上でも作成するか、共用ファイルとして参照できるようにする必要があります。クラスタが複数のシステム・ディスクで構成されている場合は、システム論理名を使用して、各ファイルのコピーが 1 つのみ存在するようにできます。

表 11-1 に示したファイルには、同期をとる必要のあるデータが含まれています。これらのファイルの複数のバージョンを維持するサイトの場合、11.2.3 項の説明に従ってデータの同期をとります。

表 11-1: クラスタ内で一本化する必要の必須システム・ファイル

ファイル	説明
NETOBJECT.DAT	DECnet オブジェクト・データベースが格納されています。このファイルには、既知の DECnet サーバ・アカウントとパスワードの一覧などが含まれています。
NETPROXY.DAT NET\$PROXY.DAT	ネットワーク代理データベースが格納されています。このファイルは、登録ユーティリティ (AUTHORIZE) を使用して管理します。
QMAN\$MASTER.DAT	マスタ・キュー・マネージャ・データベースが格納されています。このファイルには、すべての共用バッチ・キューと共用プリント・キューに関するセキュリティ情報が格納されています。2 つ以上のノードが共用キュー・システムに参加する場合、このファイルのコピーを 1 つだけ共用ディスク上に配置します。
RIGHTSLIST.DAT	ライト識別子データベースが格納されています。このファイルは、AUTHORIZE および各種ライト識別子システム・サービスが管理します。
SYSALF.DAT	システムの自動ログイン・ファイルが格納されています。このファイルは、システム管理ユーティリティ (SYSMAN) を使用して管理します。
SYSUAF.DAT	システム・ユーザ登録ファイルが格納されています。このファイルは、AUTHORIZE を使用して管理し、ユーザ登録情報設定 (\$SETUAI) システム・サービスを使用して変更が可能です。
SYSUAF.DAT	予備のシステム・ユーザ登録ファイルが格納されています。このファイルは SYSUAF.DAT のバックアップです。SYSUAFALT システム・パラメータを使用して有効にします。
VMS\$OBJECTS.DAT	クラスタの管理対象であるオブジェクトのデータベースが格納されています。このファイルには、クラスタの管理対象であるすべてのオブジェクトのセキュリティ・プロファイルが含まれています。

## 11.2.2 推奨される共通システム・ファイル

表 11-2 に示すファイルをクラスタのすべてのメンバで共用する必要はありませんが、これらのファイルに含まれるデータは完全に同期をとっておくことをお勧めします。表 11-3 では、これらのファイルの同期をとる方法と、同期がとれていない場合に生じる可能性のある問題についてまとめてあります。

一部の推奨ファイルは、要求した場合にのみ作成され、構成によっては存在しない場合があります。あるノードの必須ファイルが存在しなくてもよいのは、他のすべてのノードに存在しない場合だけです。あるノードで必須ファイルを作成すると、

ただちにそのファイルをそのノード以外のすべてのクラスタ・メンバ上でも作成するか、共用ファイルとして参照できるようにする必要があります。

表 11-2: 共通化が推奨されるシステム・ファイル

ファイル	説明
VMS\$AUDIT_SERVER.DAT	有効になっているセキュリティ監査イベントやシステム・セキュリティ監査ログ・ファイルの場所など、セキュリティ監査関連の情報が格納されています。
VMS\$PASSWORD_HISTORY.DATA	システム・パスワードの履歴データベースが格納されています。このファイルは、SET PASSWORD ユーティリティを使用して管理します。
VMSMAIL_PROFILE.DATA	システム・メール・データベースが格納されています。このファイルは、メール・ユーティリティ (MAIL) を使用して管理します。このファイルは、すべてのシステム・ユーザのメール・プロファイルに加え、システムで 사용되는すべてのメール転送アドレスのリストが記録されています。
VMS\$PASSWORD_DICTIONARY.DATA	システム・パスワード・ディクショナリが格納されています。システム・パスワード・ディクショナリとは、アカウント・パスワードとして使用できない、英語の単語や句のリストです。
VMS\$PASSWORD_POLICY	サイト固有のパスワード・フィルタが格納されています。このファイルは、セキュリティ管理者またはシステム管理者が作成し、インストールします。(パスワード・フィルタの詳細については、7.3.3.3 項を参照してください。)

### 11.2.3 複数のバージョンが存在するファイルの同期

共用ファイルを使用することが、単一のセキュリティ管理領域を実現する唯一の方法というわけではありません。サイトによっては、クラスタ内の別々のノード上に 1 つまたは複数の共用ファイルの複数のコピーを配置することを要件としている場合もあります。クラスタ内の各ノードが利用できるセキュリティ情報が同一であれば、これらのサイトでは、単一のセキュリティ管理領域が実現されていることとなります。

表 11-3 に、同期の必要なファイル、ファイルを更新するタイミング、同期がとれていない場合に発生しうる問題を示します。

表 11-3: 複数バージョンの必須クラスタ・ファイルの使用

ファイル	必要な同期	同期がとれていない 場合の影響
VMS\$AUDIT_SERVER.DAT	SET AUDIT コマンドを実行した後に必ず更新します。	監査対象領域の分割の可能性
NETOBJECT.DAT	NCP SET OBJECT コマンドまたは DEFINE OBJECT コマンドを実行した後、すべてのバージョンを更新します。	説明のつかないネットワーク・ログインの失敗や不正なネットワーク・アクセス
NETPROXY.DAT NET\$PROXY.DAT	AUTHORIZE 代理コマンドを実行した後、すべてのバージョンを更新します。	説明のつかないネットワーク・ログインの失敗や不正なネットワーク・アクセス
RIGHTSLIST.DAT	識別子または保持者に何らかの変更があった場合には、すべてのバージョンを更新します。	不正なシステム・アクセスの可能性および保護されているオブジェクトへの不正アクセス
SYSALF.DAT	SYSMAN ALF コマンドを実行した場合は、すべてのバージョンを更新します。	説明のつかないログインの失敗や不正なシステム・アクセス
SYSUAF.DAT	表 11-4のフィールドが各ユーザ・レコードについて同期されるよう、すべてのバージョンを更新します。	説明のつかないログインの失敗や不正なシステム・アクセスの可能性
SYSUAFALT.DAT	このファイルの登録レコードに何らかの変更があった場合に、すべてのバージョンを更新します。	説明のつかないログインの失敗や不正なシステム・アクセスの可能性
VMS\$OBJECTS.DAT	クラスタの管理対象オブジェクトのセキュリティ・プロファイルに何らかの変更があった場合に、またはクラスタの管理対象オブジェクトが新規に作成された場合に、すべてのバージョンを更新します。(詳細については、11.5 節を参照してください。)	保護オブジェクトへの不正なアクセスの可能性

表 11-3: 複数バージョンの必須クラスタ・ファイルの使用 (続き)

ファイル	必要な同期	同期がとれていない 場合の影響
VMSMAIL_PROFILE.DATA	メール転送パラメータに何らかの変更があった場合は、すべてのバージョンを更新します。	情報の不正な公開の可能性
VMS\$PASSWORD_HISTORY.DATA	パスワードの変更があった場合は、すべてのバージョンを更新します。	システム・パスワード・ポリシー違反の可能性
VMS\$PASSWORD_DICTIONARY.DATA	サイト固有の追加があった場合は、すべてのバージョンを更新します。	システム・パスワード・ポリシー違反の可能性
VMS\$PASSWORD_POLICY	すべてのノードに共通バージョンをインストールします。	システム・パスワード・ポリシー違反の可能性

### 11.3 登録データの同期

クラスタ上では、ユーザ登録データのすべての要素が共用データベースに存在する必要があります。この登録要素には、システム・ユーザ登録ファイル (SYSUAF.DAT とそのバックアップの SYSUAFALT.DAT)、ライト・データベース (RIGHTSLIST.DAT)、ネットワーク登録ファイル (NETPROXY.DAT)、およびネットワーク・オブジェクト・データベース (NETOBJECTS.DAT) など、すべての OpenVMS システムに存在するものや、必要に応じて作成される自動ログイン・ファイル (SYSALF.DAT) などがあります。

クラスタの安全性を確保するには、すべてのノードの登録データの同期がとれていなければなりません。サイトでこれらのファイルの複数のバージョンを維持する場合、データの同期をとる必要があります。各ユーザは、すべてのノードで同じ UIC、グループ番号、および識別子のセットが定義されている必要があります。特権とアクセス権の同期も重要です。共用ディスクは、保護の度合いがもっとも低いノードと同じ水準で保護されます。クラスタ内の各ノードで登録ファイルを個別に維持する場合は、システム・ユーザ登録ファイル (SYSUAF.DAT) のすべてのコピーについて、ユーザ特権が同じになるようにします。表 11-4 に、各ノードで同一になっていなければならない SYSUAF.DAT のフィールドです。



表 11-4: 同期をとる必要のある **SYSUAF.DAT** のフィールド

内部名	\$SETUAI アイテム・コード
UAF\$R_DEF_CLASS	UAI\$_DEF_CLASS
UAF\$Q_DEF_PRIV	UAI\$_DEF_PRIV
UAF\$B_DIALUP_ACCESS_P	UAI\$_DIALUP_ACCESS_P
UAF\$B_DIALUP_ACCESS_S	UAI\$_DIALUP_ACCESS_S
UAF\$B_ENCRYPT	UAI\$_ENCRYPT
UAF\$B_ENCRYPT2	UAI\$_ENCRYPT2
UAF\$Q_EXPIRATION	UAI\$_EXPIRATION
UAF\$L_FLAGS	UAI\$_FLAGS
UAF\$B_LOCAL_ACCESS_P	UAI\$_LOCAL_ACCESS_P
UAF\$B_LOCAL_ACCESS_S	UAI\$_LOCAL_ACCESS_S
UAF\$B_NETWORK_ACCESS_P	UAI\$_NETWORK_ACCESS_P
UAF\$B_NETWORK_ACCESS_S	UAI\$_NETWORK_ACCESS_S
UAF\$B_PRIME_DAYS	UAI\$_PRIMEDAYS
UAF\$Q_PRIV	UAI\$_PRIV
UAF\$Q_PWD	UAI\$_PWD
UAF\$Q_PWD2	UAI\$_PWD2
UAF\$Q_PWD_DATE	UAI\$_PWD_DATE
UAF\$Q_PWD2_DATE	UAI\$_PWD2_DATE
UAF\$B_PWD_LENGTH	UAI\$_PWD_LENGTH
UAF\$Q_PWD_LIFETIME	UAI\$_PWD_LIFETIME
UAF\$B_REMOTE_ACCESS_P	UAI\$_REMOTE_ACCESS_P
UAF\$B_REMOTE_ACCESS_S	UAI\$_REMOTE_ACCESS_S
UAF\$R_MAX_CLASS	UAI\$_MAX_CLASS
UAF\$R_MIN_CLASS	UAI\$_MIN_CLASS
UAF\$W_SALT	UAI\$_SALT
UAF\$L_UIC	適用なし

自動ログイン・ファイルを作成し、登録ファイルおよびライト・データベースと  
 いっしょに共用登録データベースに自動ログイン・ファイルを格納する場合は、  
 SYSMAN を使用します。クラスタ・システムの場合、自動ログイン・ファイルに  
 は、ターミナル名の接頭辞としてクラスタ・ノード名を含める必要があります。た

例えば、WILLOW というノード上の TTA0 ターミナルは、WILLOW\$TTA0 のように表現します。SYSMAN の概要については 11.8 節 を参照してください。

## 11.4 監査ログ・ファイルの管理

監査サーバ・データベースである VMS\$AUDIT\_SERVER.DAT には、監査対象のイベントに関する情報、監査ログ・ファイルの場所、および監査に伴う資源消費量の監視に使用する情報が格納されています。

監査ログ・ファイルは SYS\$COMMON:[SYSMGR] にあります。監査ログ・ファイルをシステム・ディスクとは別の場所にリダイレクトする場合、クラスタのすべてのノードで完全に同じようにリダイレクトすることが重要です。監査ログ・ファイルのリダイレクトには、SET AUDIT/JOURNAL=SECURITY/DESTINATION=ファイル名 というコマンドを使用します。このコマンドに指定するファイル名は、ノードごとに固有の名前ではなく、クラスタのすべてのノードで同じ名前に帰着するファイル名を指定します。詳しい手順については、『*HP OpenVMS Cluster システム*』を参照してください。

## 11.5 オブジェクトの保護

単一のセキュリティ管理領域とは、特定のオブジェクトに対して、特定のユーザからアクセス要求が発生した場合に、その領域内の各クラスタ・メンバがアクセス制御について同じ判断を下さなければならない範囲のことです。OpenVMS は、ファイル、キューのほか、クラスタの管理対象であるその他のオブジェクト（デバイス、ディスクやテープのボリューム、資源ドメインなど）に対して、このレベルの保護機能を提供します。表 11-5 に、各オブジェクト・クラスの動作の要約と、各オブジェクトがセキュリティ・プロファイルを格納する場所を示します。各オブジェクト・クラスの説明については、第 5 章 を参照してください。

表 11-5: クラスタにおけるオブジェクトの動作

クラス	クラスタにおける可視範囲	プロファイルの場所
ケーパビリティ	ローカル・ノードからのみ可視。	ローカル・ノードに格納。
デバイス	一部はクラスタ全体から可視。	VMS\$OBJECTS に格納されているプロファイル。
ファイル	クラスタ全体から可視。	ファイル・ヘッダに格納。
グローバル・セクション	ローカル・ノードからのみ可視。	ローカル・ノードに格納。
論理名テーブル	ローカル・ノードからのみ可視。	ローカル・ノードに格納。
キュー	クラスタ全体から可視。	ジョブ・コントローラ・キュー・データベースに格納 (表 11-1 参照)。
資源ドメイン	クラスタ全体から可視。	VMS\$OBJECTS に格納。

表 11-5: クラスタにおけるオブジェクトの動作 (続き)

クラス	クラスタにおける可視範囲	プロファイルの場所
セキュリティ・クラス	クラスタ全体から可視。	VMS\$OBJECTS に格納。
ボリューム	クラスタ全体から可視にすることが可能。	ボリュームに格納。

## 11.6 プロファイルおよび監査情報の格納

監査サーバは、クラスタ全体から可視となるオブジェクトのセキュリティ要素を VMS\$OBJECTS.DAT (場所は SYS\$COMMON:[SYSEXEC]) というデータベース内に作成し、管理します。このオブジェクト・データベースは、ノードごとに固有のファイル名ではなく、クラスタのすべてのノードで同じファイルに帰着するファイル名を指定することで、クラスタ内の各ノードに存在するようにします。

システムのブートのたびに論理名が設定されるようにするには、SYSECURITY.COM を使用して論理名を指定します。SYSECURITY.COM コマンド・プロシージャは、監査サーバが起動する前に定義する必要があります。

このオブジェクト・データベースには、次の情報が格納されています。

- すべてのオブジェクトの監査およびアラームの設定 (DCL の SET AUDIT コマンドを使用して設定)
- すべてのセキュリティ・プロファイルのテンプレート (第 5 章 参照)
- すべての資源ドメイン・オブジェクト、すべてのセキュリティ・クラス・オブジェクト、およびクラスタの管理対象の全デバイスのセキュリティ・プロファイル (11.5 節参照)

このデータベースは、特性が変更されるたびに更新され、更新後の情報は、クラスタに参加するすべてのノードでオブジェクトについて同じ情報を共用できるよう配布されます。

オブジェクト・サーバが存在せず、クラスタ・データベースである VMS\$OBJECTS.DAT が更新できない場合は、セキュリティ・プロファイルの変更や保護プロジェクトの作成はできません。ただし、SECURITY\_POLICY システム・パラメータを修正することで、ローカル・ノードでの保護オブジェクトのセキュリティ・プロファイルの変更 (第 4 ビット) やローカル・ノード上での保護オブジェクトの作成 (第 5 ビット) を行えるようにすることは可能です。

## 11.7 クラスタ全体での侵入検出

クラスタ全体での侵入検出では、あらゆる種類の攻撃からの保護措置をクラスタ全体に行き渡るようにします。各システムからの侵入データや情報を統合し、クラスタ全体を 1 つのまとまりとして保護します。

クラスタ内のメンバ・システムで SECURITY\_POLICY システム・パラメータを設定することで、不正侵入の試みと侵入イベントの状態のデータベースをローカルまたはクラスタ全体で管理できます。

SECURITY\_POLICY の第 7 ビットがクリアされている場合、システムが攻撃を受けるか、何らかの侵入イベントが記録されると、すべてのクラスタ・メンバに通知されます。あるシステムに記録されたイベントに基づき、クラスタ内の別のシステムが制限措置を講じることができます。(たとえば、ログインを試みるユーザをより詳細に監視し、ログインの再試行可能な回数と時間を制限する措置などが考えられます。この場合、再試行の上限回数が時間制限を上回ると、そのユーザはログインできなくなります。)

システム・サービスの \$DELETE\_INTRUSION、\$SCAN\_INTRUSION、および \$SHOW\_INTRUSION についての詳細は、『*HP OpenVMS System Services Reference Manual*』を参照してください。

DCL の DELETE INTRUSION コマンドおよび SHOW INTRUSION の詳細については、『*OpenVMS DCL ディクショナリ*』を参照してください。

## 11.8 システム管理ユーティリティの使用

システム管理ユーティリティ (SYSMAN) は、クラスタの管理に伴うセキュリティ管理者の作業を支援するツールです。SYSMAN のノードとクラスタの一元管理機能によって、SYSMAN を実行するローカル・ノードから、管理対象環境のすべてのノードを対象にシステム管理作業ができます。

SYSMAN を使用するには、ローカル・ノードの OPER 特権と、リモート・ノードに対する OPER 特権の許可が必要です。管理者のアカウントでクラスタ内で作業をしている場合は、SYSMAN にパスワードを入力する必要はありません。オペレーティング・システムは、論理リンク接続、またはパスワードが必要となるユーティリティの操作を監査します。

SYSMAN を使用するシステム管理者は、各ノードで論理名を同じ名前に設定するように注意する必要があります。

## 11.9 クラスタ所属の管理

クラスタ・システムでは、グループ番号とクラスタ・パスワードを使用することによって、拡張された同一の LAN (Local Area Network) に個別の複数のクラスタ・システムを共存できるようにするほか、権限のないコンピュータによるクラスタへの予期しないアクセスを回避できるようになります。グループ番号は、LAN 上の各クラスタ・システムを一意に識別するためのものです。クラスタ・パスワードは、同一 LAN 上で同じグループ番号を偶然使用してしまったクラスタの整合性を確保

するための追加のチェック手順としての働きがあります。パスワードは、グループ番号を知った侵入者によるクラスタへの参加を防ぐ働きもあります。

クラスタのグループ番号とパスワードは、暗号化され、クラスタ登録ファイルである `SYS$COMMON:[SYSEXEC]CLUSTER_AUTHORIZE.DAT` に格納されます。このファイルは、オペレーティング・システムのインストール時に、ローカル・エリア・クラスタまたは混合インターコネクト・クラスタをセットアップするよう指定した場合に作成されます。その後、インストール・プロシージャによって、クラスタのグループ番号とパスワードの指定を求められます。

通常は、`CLUSTER_AUTHORIZE.DAT` ファイル内のレコードを会話形式で変更する必要はありません。ただし、セキュリティ侵害の疑いがある場合は、必要に応じてクラスタ・パスワードを変更します。クラスタ・パスワードの変更には、`SYSMAN` を使用します。`CLUSTER_AUTHORIZE.DAT` には、`SYSPRV` 特権を持つユーザだけがアクセスできます。グループ番号またはパスワードを変更した場合は、必ずクラスタ全体をリブートしてください。

複数のシステム・ディスクを使用する構成の場合、それぞれのディスクに `CLUSTER_AUTHORIZE.DAT` のコピーを置く必要があります。`SYSMAN` を実行してすべてのコピーを更新する必要があります。

次のコマンド・シーケンスは、`SYSMAN` を使用してクラスタ・パスワードを変更する場合の出力例です。

```
SYSMAN> SET CLUSTER_AUTHORIZATION/GROUP_NUMBER=65353
SYSMAN> SET ENVIRONMENT/CLUSTER/NODE21
SYSMAN> SET PROFILE /PRIVILEGE=SYSPRV
SYSMAN> CONFIGURATION SET CLUSTER_AUTHORIZATION/PASSWORD=HOOVER
%SYSMAN-I-CAFOLDGROUP, existing group will not be changed
%SYSMAN-I-GRPNOCHG, Group number not changed
%SYSMAN-I-CAFREBOOT, cluster authorization file updated
The entire cluster should be rebooted.
```

## 11.10 クラスタ・ノード間での DECnet の使用

クラスタ環境では、さまざまな資源共用モデル（ファイルとボリューム、ディスクとテープ・デバイス、バッチ・キューとプリント・キューなど）が使用できるため、通常は、DECnet ソフトウェアを利用してクラスタの別のノードに直接アクセスする必要がありません。それでも、資源がクラスタ全体で均一に共用されないことがあります。これは特に、サテライトのディスク・ボリュームやテープ・ボリュームへのクラスタによるアクセスを制限する、混合インターコネクト・クラスタ構成やローカル・エリア・クラスタ構成の場合に当てはまります。このような場合は、DCL の `SET HOST` コマンドを使用するか、何らかのネットワーク・アクセス手段を用いて、他のクラスタ・メンバからサテライトの資源にアクセスする必要があります。代理ログインによるネットワーク・アクセスの詳細については、12.3 節を参照してください。



## ネットワーク環境におけるセキュリティ

ネットワーク環境におけるセキュリティは、単一のシステムからなる環境でのセキュリティと比較して、求められる慎重さの度合いは高くなります。また、ネットワーク環境に共通して存在する運用上の複雑さや制御の分散化が理由で、セキュリティの確立が非常に困難です。ネットワークの規模が大きくなればなるほど、多数のノードのセキュリティ管理者間における調整やコミュニケーションの問題もそれだけ難しくなります。

現在のネットワーク技術の制限が原因で、ネットワーク・サイトにおいて達成可能であると見込めるセキュリティのレベルには限度があります。発生しうる問題に注意を払うことは、ネットワーク内のセキュリティの弱点を拡大しかねない運用を避けることにつながります。この章では、ネットワークにおけるセキュリティ問題が生じる領域を明らかにし、適切な運用の実現に役立つ情報を紹介します。

次に示すものも含め、OpenVMS システムのネットワーク・ソフトウェア・オプションの詳細については、『*OpenVMS* システム管理者マニュアル』を参照してください。

- HP TCP/IP Services for OpenVMS
- DECnet-Plus for OpenVMS (DECnet Phase V)
- DECnet for OpenVMS (DECnet Phase IV)

### 12.1 ネットワーク・セキュリティの管理

ネットワーク・ソフトウェアは、次に示すように、さまざまなレベルでネットワークへのアクセスを規制します。

- ネットワークへアクセスするための特権

何らかのネットワーク処理を実行するネットワーク・ユーザはすべて TMPMBX 特権と NETMBX 特権を持っていなければなりません。特権ユーザは、TMPMBX と NETMBX の他に、さまざまな特権を持っています。

- アクセス制御

ネットワークに属しているノードへ接続するには、ユーザには明示的なアクセス情報、代理アカウント、アプリケーション・アカウント、およびデフォルトの DECnet アカウントが必要です。(12.2 節参照)

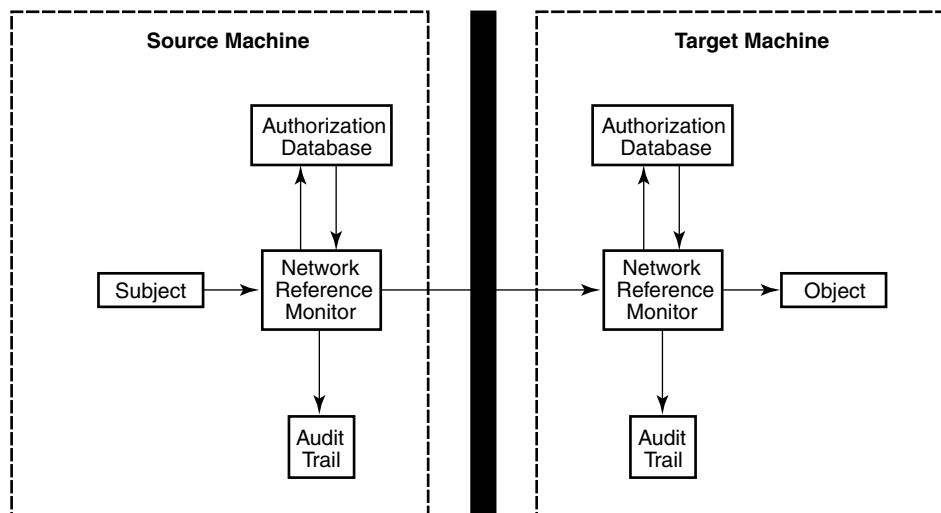
- 同期回線または非同期回線を経由してローカル・ノードを遠隔ノードに接続するために必要なルーティング初期化パスワード (12.5 節参照)

### 12.1.1 セキュリティ確保のための要件

ネットワーク環境におけるセキュリティを確保するには、3つの重要な要件があります。

- 共通のセキュリティ・ポリシー**  
 ソース・マシン上の開始側プロセスと、開始側プロセスに代わってターゲット・マシン上で処理を行うプロセスが対応している必要があります (図 12-1 参照)。この対応関係は、2つのリファレンス・モニタで管理し、ターゲット・マシン (最終的にオブジェクトを保護するマシン) のセキュリティ・ポリシーと整合していなければなりません。リファレンス・モニタの説明については、第2章を参照してください。
- 共用アクセス制御情報**  
 ターゲット・マシン上の登録データベースには、ソース・マシン上の開始側プロセスに対応する何らかのアクセス許可 (アカウントや代理など) を設定する必要があります。
- 保護されている回線、通信線、ターミナル、およびプロセッサ**  
 ローカルと遠隔のサブジェクトとの間の対応関係を確実に確立して認証できるように、2つのリファレンス・モニタ (ソースとターゲット) を結ぶ保護された通信手段が必要です。

図 12-1: ネットワークにおける参照モニタ



VM-1002A-AI



## 12.1.2 ネットワークにおける監査

セキュリティ管理者は、SET AUDIT コマンドを使用して特定のイベント・クラスを有効にすることで、ネットワークの状態を監査することができます。監査できる内容は次のとおりです。

- NCP コマンドの使用。各 NCP コマンド行を、実行完了時の状態も含め、監査できます。
- 特権の使用。ネットワーク環境の場合、特権の使用の大半は、運用時ネットワーク・データベースの変更にともなう OPER 特権の使用に関連しています。
- 接続の開始と終了

DECnet for OpenVMS を実行する VAX システムでは、各ネットワーク接続について、4 種類の監査が行われます。

1. 接続を開始するソース・ノードは、最初のイベント・メッセージを記録します。
2. 着信開始メッセージを受信するターゲット・ノードは、2 つ目のイベントを記録します。
3. 3 つ目のイベント・メッセージは、接続を終了した側のノードが記録します。
4. 最後のイベント・メッセージは、接続を切断された側のノードが記録します。

着信ネットワーク接続の場合、監査メッセージには、接続を開始したユーザを示す遠隔ユーザ名が含まれています。発信論理リンク接続の場合、遠隔論理リンク識別子は常に 0 となります。

## 12.2 アクセス制御の階層

DECnet ノードは、遠隔 DECnet ノードへの接続を試みる時、遠隔 DECnet ノードに対してアクセス制御情報を送信します。アクセス制御情報は、いくつかの情報源から取得できます。次のリストは、優先順位の高いものから順に示した、アクセス制御の階層です。

1. ローカル・ノード上のネットワーク・ユーザは、明示的にアクセス制御情報を提供できます。ローカル・ノード上のユーザがアクセス情報を提供する場合、遠隔ノードはこのアクセス制御情報を使用します。明示的なアクセス制御については、12.2.1 項を参照してください。
2. ローカル・ノードは、ローカル・ノードまたはアプリケーションについて発信代理アクセスが有効になっているかどうかを調べます。代理が有効になっていれば、ローカル・ノードは、接続要求の中に開始ユーザ名を含めて送信しま

す。代理が遠隔ノードでも有効になっている場合、開始ユーザが代理アクセスの権限を持っているかどうかを DECnet ソフトウェアが判断します。代理アクセス制御については、12.2.2 項および 12.3 節を参照してください。

3. 遠隔ノードは、アクセス制御が指定されておらず、代理も有効になっていないと判断した場合、構成データベースを調べます。構成データベースにアプリケーション・ユーザ名が含まれていれば、その名前が使用されます。デフォルト・アプリケーション・アカウントについては、12.2.3 項および 12.4 節を参照してください。
4. 構成データベースにデフォルト・アプリケーション・ユーザ名がない場合、遠隔ノードは、デフォルトの非特権 DECnet ユーザ名情報が含まれていないかどうか構成データベースを調べます。含まれていれば、遠隔ノードは、そのデフォルトの非特権 DECnet ユーザ名を使用します。デフォルトの DECnet アカウントについては、12.4 節を参照してください。

最終的に上記の情報源から情報が得られない場合、接続は失敗します。

### 12.2.1 明示的アクセス制御の使用

ユーザは、明示的アクセス制御情報を指定することにより、遠隔ノード上で DCL コマンドおよび NCP コマンドを実行できます。明示的アクセス制御情報は、ユーザ名とパスワードが含まれており、遠隔システムの特定のアカウントへのアクセスを可能にします。明示的アクセス制御情報を指定するには、標準的な OpenVMS ノード指定を使用する方法と、NCP コマンドを使用する方法があります。

- OpenVMS ノード指定の場合、アクセス制御文字列は、次に示すように、遠隔アカウントのユーザ名とユーザのパスワードを二重引用符で囲んで指定します。

```
NODE"username password"::disk:[directory]file.typ
```

次の例では、Puterman というユーザがアクセス制御文字列を使用して、BIONEWS.MEM というファイルをコピーしています。

```
$ COPY WALNUT"PUTERMAN A25D3255"::BIONEWS.MEM BIONEWS.MEM
```

- 遠隔ノード上で NCP コマンドを実行するには、ユーザ名とパスワードを指定することでコマンドを実行できます。

次の例では、TORONTO という遠隔ノードの MAIL アプリケーションに関するすべての特性情報を表示しています。

```
NCP> TELL TORONTO USER A_JOHNSTON PASSWORD XZZOQ87 SHOW OBJECT-  
_NCP> MAIL CHARACTERISTICS
```

### 12.2.2 代理ログインの使用

代理ログインを使用すると、遠隔ノードにログインしたユーザが、アクセス制御情報を指定せずにローカル・ノードの特定のアカウントに自動的にログインできるよ

うになります。代理ログインは、会話型ログインとは異なります。代理ログインの場合、コピー操作など、特定のネットワーク・アクセス操作を実行できます。一方、会話型ログインの場合、ユーザが会話型の操作を実行するには、ユーザ名とパスワードを入力する必要があります。

ローカル・ノードに代理ログインを設定するには、遠隔ユーザは、ローカルのユーザ名にマッピングされるデフォルトの代理アカウントをローカル・ノード上に持っている必要があります。代理ログインした遠隔ユーザは、ローカル・ユーザ名と同じファイル・アクセス権、ライト、特権を継承します。代理ログインのケーパビリティを利用して、セキュリティを高めることができます。なぜなら、代理ログインでは、ネットワーク経由で送信されるノード指定、および、コマンド・プロセスに格納されるノード指定に、明示的アクセス制御情報を指定する必要性を最小限に抑えることができるためです。

ネットワーク・アプリケーションにも代理ログイン・アクセスを割り当てることができます。

評価済みの構成でアクセス制御文字列を使用することは許可されていません。評価済み構成では、代理ログイン・アカウントを使用してください。

### 12.2.3 デフォルト・アプリケーション・アカウントの使用

ネットワーク・アプリケーション専用のアクセス制御のもう 1 つの形態に、アクセス制御情報を送信しない遠隔ノードからの着信接続で使用されるデフォルト・アカウント情報があります。遠隔ノードがアクセス制御情報を指定しないため、ローカル・ノードは、アプリケーションについて指定した接続用のデフォルト情報を使用します。

次のコマンドを使用して、アプリケーションに使用するデフォルトのアクセス制御情報をネットワーク構成データベースに格納できます。

```
NCP> SET OBJECT FAL USER JILL
```

## 12.3 代理アクセス制御

12.2.2 項では、代理ログインの概念について説明しました。代理アクセスの許可は、異なるノードまたはグループに属するユーザがシステム上のファイルを共有したいけれども、パスワードの発行や、ディレクトリとファイルの保護を W:RWE に設定するのを控えたい場合に与えることができます。代理ログインでは、ネットワーク経由でファイルをコピーするときに、パスワードをコマンドに含める必要がありません。また、ファイル転送のためにファイルにワールド読み込みアクセスを許可する必要もありません。ユーザは、次に示す形式で DCL の COPY コマンドをデフォルトの代理アカウントに対して入力します。

```
COPY remotenode::file-spec file-spec
```

デフォルト以外のアカウントから代理アクセスを使用して、ネットワーク経由でファイルをコピーするには、次に示すように、DCL コマンドのアクセス制御文字列の中に代理アカウントの名前を記述します。

```
COPY remotenode"proxyacct"::file-spec file-spec
```

### 12.3.1 代理アクセスに関する特別なセキュリティ対策

代理アクセスは、関係するシステムの登録データベースどうしの選択的なマージです。そのため、そのセキュリティは、最もセキュリティの弱いノードと同じ水準です。

代理アクセスでは、ネットワーク経由でパスワードを送信せずに済みますが、パーソナル・コンピュータが、許可されているノードのいずれかになりすまして代理ログインを迂回することは可能です。このため、次の手続きを実施します。

- 機密性の高いデータに対する着信代理アクセスは有効にしないようにします。
- 非特権代理アカウントを設定します。アカウントに特権が必要な場合は、その特権でシステムに損害を与えられないことを確認します。この手法は、あるノードが侵入を許した場合でも、システム間の防御壁として機能します。代理ログインは、他のノードの非特権アカウントに対してのみ許可を与えるため、ネットワーク内のあるシステムが侵入を許しても、その損害が広がるのを抑えるのに役立つ場合があります。サイトに高いセキュリティ要件が設定されている場合は、特権ユーザ名に対してネットワーク・アクセスや遠隔アクセスを許可しないようにしてください。
- 代理アクセスを、ネットワーク上に常に、またはほぼ常に存在しているノードに対してのみ許可するようにします。侵入者にとっては、ネットワークから切断されているノードになりすます方が簡単なためです。代理アクセスの使用と、アクセス制御文字列（パスワードを含む）がネットワークを流れる状態とのバランスをとる必要があります。ノードになりすますことが可能なネットワーク上のワークステーションまたはパーソナル・コンピュータは、ネットワーク・メッセージを監視して、パスワードを盗むことも可能です。最終的には、ローカル・ネットワークに接続されるすべてのノードに一定レベルの信頼性が備わるようにする必要があります。
- ユーザの登録は慎重に行います。ユーザの登録については、遠隔サイトのセキュリティ管理者から正式な登録要求を受け取るのが理想的です。
- 代理アカウント用のログイン・コマンド・プロシージャを調べます。キャプティブ・アカウントのログイン・コマンド・プロシージャについては、7.2.4.2 項に示した推奨事項に必ず従うようにします。ログイン・コマンド・プロシージャは、代理アカウントの所有者以外のユーザが所有する、確実に保護されているディレクトリに配置するようにします。その代理アカウントを使用するユーザには、書き込みアクセスを禁止します。

### 12.3.2 代理データベースの設定

遠隔ユーザからの接続要求にアクセス制御情報が含まれていない場合、代理アクセスを認めるには次の条件を満たす必要があります。

- ターゲット・ノード上の代理データベースには、遠隔ソース・ノードのノード同義語とソース・ユーザ名に一致する、ソース・ノードのノード同義語とソース・ユーザ名の組み合わせが格納されていなければなりません。たとえば、例 12-1 では、セキュリティ管理者は KMahogany の代理を追加しています。KMahogany は Birch というノードから代理アカウントにアクセスしなければなりません。
- ターゲット・ノードのユーザ登録ファイルには、代理データベースに格納されているターゲット・ソース・ユーザ名に一致するソース・ユーザ名が格納されていなければなりません。例 12-1 では、ノード Birch の SYSUAF.DAT ファイルに KMahogany のユーザ登録レコードが含まれていると仮定しています。
- 着信代理アクセスは、構成データベースの中でターゲット・ノードについて有効にしなければなりません。(12.3.2.1 項参照。)
- 着信代理アクセスは、構成データベースの中でターゲット・アプリケーションについて有効にしなければなりません(12.3.2.1 項参照)。
- 発信代理アクセスは、発信元のノード上で、そのノード自体と、代理アクセスを使用する予定のアプリケーションすべてについて有効にしなければなりません。

代理ログインの使用の制御は、ローカル・ノードでできます。AUTHORIZE を使用して、パーマネント代理データベースの作成、変更を行います。

デフォルトのネットワーク代理登録ファイルは、NET\$PROXY.DAT です。ただし、AUTHORIZE は、互換性、多くのレイヤード・プロダクトのサポート、および DECnet for OpenVMS (Phase IV) ノード名の変換に使用するために NETPROXY.DAT ファイルも維持しています。

それぞれのネットワーク代理エントリでは、1 つの遠隔ユーザをローカル・ノード上の複数の代理ユーザ名 (1 つのデフォルト代理ユーザ名と最大 15 の追加代理ユーザ名) にマッピングできます。1 つのノードとログイン名から複数の代理アカウントにアクセスする予定の場合は、どの代理アカウントをデフォルトにするかを指定します。代理データベースのエントリは、*nodename::username* または *nodename::[group,member]* という形式でユーザを表します。

たとえば、ローカル・ノードに代理ファイルを作成し、Boston という遠隔ノード上の Martin というユーザをローカル・ノードの Allen というユーザにマッピングする、デフォルト代理エントリを追加するには、次のコマンドを入力します。

```

$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE

UAF> CREATE/PROXY
UAF> ADD/PROXY BOSTON::MARTIN ALLEN/DEFAULT
UAF> EXIT

```

同様に、遠隔ノードのシステム管理者は、そのノードの特定のアカウントに対する代理アクセスが可能なネットワーク・ユーザの代理データベースを作成し、管理することができます。表 12-1は、代理データベースの管理に使用する AUTHORIZE コマンドをまとめたものです。

表 12-1: ネットワーク代理アクセスの管理に使用する AUTHORIZE コマンド

コマンド	引数	説明
ADD/PROXY	node::remoteuser localuser[,...]	指定したユーザについて代理アクセスを可能にします。
CREATE/PROXY		ネットワーク代理登録ファイルを作成します。
LIST/PROXY		すべての代理アカウントとそれらのアカウントへの代理アクセスが可能なすべての遠隔ユーザを一覧にしたリスト・ファイルを作成します。
MODIFY/PROXY	node::remoteuser	指定したユーザの代理アクセスの設定を変更します。
REMOVE/PROXY		指定したユーザの代理アクセスを削除します。
SHOW/PROXY	* node::remoteuser	指定したユーザについて許可されている代理アクセスを表示します。

### 12.3.2.1 着信代理アクセスの有効化および無効化

ローカル・ノードに対する代理アクセスおよび特定のアプリケーションに対する代理アクセスは制御できます。

#### ノードへの代理アクセスの制御

ローカル・ノードへの代理接続を認めるには、次に示すコマンドを使用して、エグゼキュータ・データベースに格納されている着信代理属性を設定します。

```
NCP>SET EXECUTOR INCOMING PROXY ENABLE
```

ローカル・ノードへの代理接続を拒否するには、次のコマンドを使用して、発信代理属性を設定します。

NCP>**SET EXECUTOR INCOMING PROXY DISABLE**

ノードへの代理アクセスが無効になっている場合、システムはすべての代理接続要求を無視します。

発信ノードについても、接続要求メッセージで代理データが伝送されるように、同様の手順を実行する必要があります。ノード、および代理を使用する予定のあるすべてのアプリケーションの両方について、代理属性を設定します。たとえば、次のようにして設定します。

NCP>**SET EXECUTOR OUTGOING PROXY ENABLE**

NCP>**SET OBJECT MAIL PROXY BOTH**

NCP>**SET OBJECT MAIL PROXY INCOMING**

NCP>**SET OBJECT MAIL PROXY OUTGOING**

一般に、発信代理接続を有効にする方法は、接続メッセージに開始ユーザ名が挿入されるため、よい方法といえます。これは、ターゲット・ノードがオブジェクトに対して代理アクセスを有効にしていない場合であっても同様です。そうしておけば、ユーザ名がターゲット・ノードの会計情報ログおよび監査ログに記録されます。ただし、ごく一部の DECnet アプリケーションは、接続メッセージのための領域をユーザ名でなくアプリケーション情報に使用するなど、非代理形式の接続メッセージを使用するため、発信代理接続が有効になっていると機能しないことがあります。

#### アプリケーションへの代理アクセスの制御

特定のアプリケーションへの代理アクセスを許可するには、ノードとアプリケーションの両方について代理アクセスを有効にする必要があります。加えて、アプリケーションの名前を SET OBJECT コマンドを使用して指定します。たとえば、NML というアプリケーションへの代理アクセスを有効にするには、次のコマンドを実行します。

NCP>**SET EXECUTOR INCOMING PROXY ENABLE**

NCP>**SET OBJECT NML INCOMING PROXY ENABLE**

アプリケーションへの代理アクセスを無効にするには、SET OBJECT コマンドを使用してアプリケーションを指定し、着信代理属性を無効に設定します。たとえば、FAL というアプリケーションへの代理アクセスを無効にするには、次のコマンドを実行します。

NCP>**SET OBJECT FAL INCOMING PROXY DISABLE**

アプリケーションへの着信代理アクセスが有効になっていて、ノードへの代理アクセスが無効に設定されている場合、当該システムはそのアプリケーションへのすべての代理アクセスを無視します。

#### 12.3.2.2 代理アクセスの削除

システムへの代理アクセスは、不要になったら削除します。代理アクセスを削除するには、AUTHORIZE を起動し、次のコマンドを入力します。

```
UAF> REMOVE/PROXY BOSTON::MARTIN
```

### 12.3.2.3 代理アカウントの作成手順

他のノードの1人以上のユーザが使用する代理アカウントをローカル・ノード上に設定するには、次の手順を実行する必要があります。アカウントを作成するときは、12.3.1 項に示したセキュリティ・ガイドラインを参照してください。

1. アカウントの目的、名前、アクセスを許可するネットワーク・ユーザを指定します。
2. 必要な場合は、AUTHORIZE を使用してローカル・アカウントを作成します。アカウントが既にある場合は、/NOINTERACTIVE、/NOBATCH、および /NETWORK の制限と指定が済んでいることを確認します。
3. アカウントの特権を確認します。通常、代理ログイン・アカウントには特権を与えないようにします。
4. 必要な場合は、ネットワーク代理登録ファイルを AUTHORIZE ユーティリティの CREATE/PROXY コマンドを使用して作成します。(通常はシステムによって自動的に作成されます。)
5. AUTHORIZE の ADD/PROXY コマンドを使用して、代理アカウントへのアクセスが必要なすべての遠隔ユーザを指定します。
6. ディレクトリのデフォルトの保護設定を確認し、必要に応じてカスタマイズします。
7. ADD コマンドの /LGICMD 修飾子に指定されているログイン・コマンド・プロシージャがあれば、それらを確認します。キャプティブ・アカウントの場合、該当するログイン・コマンド・プロシージャが 7.2.4.2 項に示した推奨事項に従っていることを確認してください。ログイン・コマンド・プロシージャは、代理アカウントの所有者以外のユーザが所有する、確実に保護されているディレクトリに配置するようにします。その代理アカウントを使用するユーザによる書き込みアクセスを禁止するようにします。
8. 遠隔ノードのセキュリティ管理者に対し、その遠隔ノードのどのユーザにローカル・ノードへのアクセスを許可したかを通知します。

### 12.3.3 代理アカウントの例

例 12-1は、WALNUT という名前のノードのセキュリティ管理者が、GENACCESS という名称の汎用アクセス・アカウントを作成する場合の例です。同時に、この管理者は、BIRCH というノードの KMahogany、PSumac、および WPine という3人のユーザと、WILLOW というノード RDogwood と WCherry という2人のユーザに対して、代理ログインを許可する手順を実行します。この時点では、ネットワーク代理登録ファイルは存在していないものとします。



## 例 12-1: 代理アカウントの例

---

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF>ADD GENACCESS /PASSWORD=WHYNADGUM/UIC=[236,043] -
_UAF> /DEVICE=STAFFDEV/DIRECTORY=[GENACCESS] -
_UAF> /OWNER="Security Mgmt"/ACCOUNT=SEC -
_UAF> /FLAGS=(DISWELCOME,DISNEWMAIL,GENPWD,DISMAIL) -
_UAF> /NOBATCH/NOINTERACTIVE/MAXDETACH=8 -
_UAF> /LGICMD=LOGIN/MAXACCTJOBS=10

%UAF-I-ADDMSG, user record successfully added
%UAF-I-RDBADDMSGU, identifier GENACCESS value [000236,000043]
added to rights database
%UAF-I-RDBADDMSGU, identifier SEC value [000236,177777] added to
rights database
UAF>CREATE/PROXY
UAF>ADD/PROXY BIRCH::KMAHOGANY GENACCESS/DEFAULT
%UAF-I-NAFADDMSG, proxy from OMNI:.BOSTON.BIRCH::KMAHOGANY to
GENACCESS added
UAF>ADD/PROXY BIRCH::PSUMAC GENACCESS/DEFAULT
%UAF-I-NAFADDMSG, proxy from OMNI:.BOSTON.BIRCH::PSUMAC to
GENACCESS added
UAF>ADD/PROXY BIRCH::WPINE GENACCESS/DEFAULT
%UAF-I-NAFADDMSG, proxy from OMNI:.BOSTON.BIRCH::WPINE to
GENACCESS added
UAF>ADD/PROXY WILLOW::RDOGWOOD GENACCESS/DEFAULT
%UAF-I-NAFADDMSG, proxy from OMNI:.BOSTON.WILLOW::RDOGWOOD to
GENACCESS added
UAF>ADD/PROXY WILLOW::WCHERRY GENACCESS/DEFAULT
%UAF-I-NAFADDMSG, proxy from OMNI:.BOSTON.WILLOW::WCHERRY to
GENACCESS added
UAF>SHOW/PROXY *::*
Default proxies are flagged with a (D)

OMNI:.BOSTON.BIRCH::KMAHOGANY
GENACCESS (D)

OMNI:.BOSTON.BIRCH :::PSUMAC
GENACCESS (D)

OMNI:.BOSTON.BIRCH :::WPINE
GENACCESS (D)

OMNI:.BOSTON.WILLOW :::RDOGWOOD
GENACCESS (D)

OMNI:.BOSTON.WILLOW :::WCHERRY
GENACCESS (D)

UAF>EXIT
{messages}
$ DIRECTORY/SECURITY SYS$STAFF:[000000]GENACCESS.DIR
:
:
```

#### 例 12-1: 代理アカウントの例 (続き)

---

```
$ DIRECTORY/SECURITY SYS$STAFF:[GENACCESS]LOGIN.COM  
:  
:
```

---

## 12.4 DECnet アプリケーション (オブジェクト) アカウントの使用

ネットワーク・オブジェクトとは、DECnet ネットワーク内のノード間の通信が可能なシステム・プログラムおよびユーザの作成したアプリケーションのことで、システムへのアクセスが許可されているネットワーク・オブジェクトのセットを特定し、各オブジェクトに対して適切なアクセス制御を設定する必要があります。次の機能を使用できます。

- DECnet オブジェクト・アカウント

このアカウントは、システムに自動的に設定される特定のネットワーク・オブジェクト (MAIL など) 専用のアカウントです。デフォルトの DECnet アカウントと比較して、オブジェクトへの遠隔アクセスに対するより詳細な管理が可能です。(たとえば、遠隔ノード名またはユーザ名に応じて、オブジェクトへのアクセスを許可または拒否するログイン・コマンド・プロシージャを使用するキャプティブ・アカウントをオブジェクトに設定できます。)

- デフォルトの DECnet アカウント

このアカウントは、すべてのネットワーク・オブジェクトに対してシステムへの汎用アクセスを許可します。外部との接続やダイアルアップ回線がないサイト内に配置されているシステムで構成される LAN など、セキュリティ要件が高くないシステムに適したアカウントです。

デフォルトの DECnet ユーザ名を使用する場合、ユーザはユーザ名とパスワードを入力しなくても、特定のネットワーク操作 (別々のノードのユーザ同士の電子メール交換など) を実行できます。デフォルトの DECnet ユーザ名は、アクセス制御情報が指定されていない場合のファイル操作にも使用されます。たとえば、デフォルトの DECnet ユーザ名を使用することで、ファイル保護がワールド・アクセスに設定されているローカル・ファイルに遠隔ユーザがアクセスすることができます。遠隔ユーザによるローカル・ノードへのアクセスを望まない場合は、デフォルトの DECnet ユーザ名を作成しないようにします。デフォルトの DECnet アカウントの削除については、12.4.3 項を参照してください。

### 12.4.1 ネットワーク・オブジェクトのまとめ

ネットワーク・オブジェクトに適用するアクセス制御を決める前に、OpenVMS オペレーティング・システムが提供するネットワーク・オブジェクトの機能を理

解する必要があります。この節では、最も一般的なネットワーク・オブジェクトについて説明します。

## **FAL**

ファイル・アクセス・リスナ (FAL) とは、遠隔ファイル・アクセス機能のことです。具体的には、ローカル・ノードのファイルに対する遠隔ファイル・アクセス要求を受け取って処理するイメージのことを指します。

一般 *FAL* アクセスは、よほどの理由がない限り使用しないでください。無制限アクセスでは、ワールド・アクセス許可が設定されているすべてのファイルに対する一般ネットワーク・アクセスが可能になります。また、ワールド・ライト・アクセスが設定されているディレクトリでの遠隔ユーザによるファイルの作成が可能になります。

セキュリティ要件が高いサイト、またはアクセスが想定されるすべてのユーザを認知することが困難なサイトでは、*FAL* アカウントを作成するべきではありません。このようなサイトでは、ユーザのアクセスを制御するために、特定の目的のための代理アカウントを設定することができます (12.3 節参照)。

## **MAIL**

MAIL は、OpenVMS システムにおいてパーソナル・メール・サービスを提供するイメージです。一部の例外を除いて、MAIL オブジェクトにはシステムへの一般アクセスを許可します。

## **MIRROR**

MIRROR は、特定の形式のループバック・テストに使用するイメージです。たとえば、MIRROR は、UETP テスト・パッケージの DECnet フェーズで実行されます。

## **MOM**

MOM とは、保守操作モジュール (Maintenance Operations Module) のことです。MOM イメージは、無人システムをダウンライン・ロードし、OpenVMS ノードからターゲット・ノードにオペレーティング・システム・ファイル・イメージのコピーを転送します。MOM オブジェクトは、システム・インストール時に設定されます。

## **NML**

NML とは、ネットワーク管理リスナ (Network Management Listener) のことです。NML へのアクセスが可能な遠隔ユーザは、NCP TELL コマンドを使用して、DECnet データベースからネットワーク情報を収集して報告することができます。

## PHONE

PHONE とは、遠隔 OpenVMS システム上のユーザとの間でオンラインの会話を可能にするイメージです。PHONE に対してデフォルトの DECnet アクセスを許可する場合、ネットワーク内の誰もが現在ローカル・システムにログインしているユーザの一覧を取得でき、そのユーザ名の一覧を使用してログインを試みる可能性もあるので注意が必要です。

## TASK

TASK オブジェクトでは、デフォルトの DECnet アカウントを介した任意のコマンド・プロシージャ（侵入の際に使用される恐れのあるものも含まれます）のシステム上での実行が許可されます。

システムに対するデフォルトの DECnet アクセスを許可しない場合、または TASK オブジェクトへのデフォルトの DECnet アクセスを無効にしている場合は、アクセス制御文字列または代理アクセスを使用することで、遠隔ユーザが作成したコマンド・プロシージャ（タスク）をシステム上で実行させることが可能です。

## VPM

VPM とは、仮想性能モニタ・サーバ (Virtual Performance Monitor Server) のことです。VPM へのアクセスは、モニタ・ユーティリティ (MONITOR) のクラスタ監視機能を使用する場合に必要です。

### 12.4.2 手作業でのネットワーク・オブジェクトの設定

NETCONFIG.COM コマンド・プロシージャは、システム上のネットワーク・オブジェクトを自動的に設定します。また、NETCONFIG\_UPDATE.COM コマンド・プロシージャは、ネットワーク・オブジェクトを自動的に更新します。

これらのコマンド・プロシージャを使用しない場合は、次の手順を実行することで、特定のオブジェクトに対するネットワーク・アクセスを許可することができます。

1. 各ネットワーク・オブジェクトについて、最上位のディレクトリを作成し、一意の所有者 UIC とグループ UIC を指定します。たとえば、次のコマンド・シーケンスは、MAIL オブジェクトのための最上位ディレクトリをシステム・ディスク上に作成します。

```
$ SET DEFAULT SYS$SPECIFIC:[000000]
$ CREATE/DIRECTORY [MAIL$SERVER]/OWNER_UIC=[376,374]
```

表 12-2 は、コマンド・プロシージャの NETCONFIG.COM および NETCONFIG\_UPDATE.COM が、個々のネットワーク・オブジェクトのためのアカウントを作成するために使用する、ディレクトリ名、ユーザ名、および UIC の一覧です。一貫性を保つために、手作業でネットワーク・オブジェクト・アカウントを作成するときも同じ情報を使用してください。

MOM オブジェクトは、インストール時にオペレーティング・システムによって作成されます。

- 2. AUTHORIZE を使用して、オブジェクト用のアカウントを作成し、自動生成されたパスワードを使用します。(指定するユーザ名とパスワードは、ネットワーク・データベースのオブジェクトに指定するパスワード(手順 3)と同じものにする必要があります。)

たとえば、次のコマンド・シーケンスは MAIL オブジェクト用のアカウントを設定します。

```
$ RUN SYS$SYSTEM:AUTHORIZE
_UAF> ADD MAIL$SERVER/OWNER=MAIL$SERVER DEFAULT -
_UAF> /PASSWORD=MDU1294B/UIC=[376,374]/ACCOUNT=DECNET -
_UAF> /DEVICE=SYS$SPECIFIC: /DIRECTORY=[MAIL$SERVER] -
_UAF> /PRIVILEGE=(TMPMBX,NETMBX) /DEFPRIVILEGE=(TMPMBX,NETMBX) -
_UAF> /FLAGS=(RESTRICTED,NODISUSER,NOCAPTIVE) /LGICMD=NL: -
_UAF> /NOBATCH /NOINTERACTIVE
```

AUTHORIZE ユーティリティの SHOW MAIL\$SERVER コマンドを実行すると、例 12-2に示すように、MAIL オブジェクトのネットワーク・アカウント設定が表示されます。

- 3. NCP の DEFINE コマンドを使用して、アカウントのユーザ名とパスワードを、ネットワーク・データベース内の指定したオブジェクトに関連付けます。次のように指定します。

```
$ RUN SYS$SYSTEM:NCP
_NCP> DEFINE OBJECT MAIL USER MAIL$SERVER PASSWORD MDU1294B
_NCP> EXIT
```

- 4. 各ネットワーク・オブジェクトごとに、手順 1 から手順 3 を繰り返します。
- 5. ネットワーク・オブジェクトの設定が終わったら、エグゼキュータ・データベースからデフォルトの DECnet アクセスを削除し、SYSUAF からデフォルトの DECnet アカウントを削除します( 12.4.3 項参照)。
- 6. 最後にシステムをリブートして、パーマネント・エグゼキュータとオブジェクト・データベースに加えた変更を、実行中のシステムに反映させます。

表 12-2 は、ネットワーク・オブジェクトのデフォルト設定です。

表 12-2: ネットワーク・オブジェクトのデフォルト設定

オブジェクト名	ディレクトリおよびユーザ (アカウント) 名	UIC
FAL	FAL\$SERVER	[376,373]
MAIL	MAIL\$SERVER	[376,374]
MIRROR	MIRRO\$SERVER <sup>a</sup>	[376,367]

表 12-2: ネットワーク・オブジェクトのデフォルト設定 (続き)

オブジェクト名	ディレクトリおよびユーザ (アカウント) 名	UIC
\$MOM	VMS\$COMMON:[MOM\$SYS- TEM] <sup>b</sup>	[376,375]
NML	NML\$SERVER	[376,371]
PHONE	PHONE\$SERVER	[376,372]
VPM	VPM\$SERVER	[376,370]

<sup>a</sup>AUTHORIZE では、ユーザ名が 12 文字以下という制限があるため、MIRRO\$SERVER への MIRROR オブジェクト・アカウントのユーザ名(およびディレクトリ名)を縮める必要があります。

<sup>b</sup>MOM には、関連付けられているユーザ名はありません。

#### 例 12-2: MAIL\$SERVER アカウントの UAF レコード

```

Username: MAIL$SERVER          Owner:  MAIL$SERVER
Account: MAIL$SERVER DEFAULT   UIC:    [376,374] ([DECNET,MAIL$SERVER])
CLI:    DCL                     Tables:
Default: SYS$SPECIFIC:[MAIL$SERVER]
LGICMD:
Login Flags:  Restricted
Primary days:  Mon Tue Wed Thu Fri Sat Sun
Secondary days:
Primary  000000000001111111112222  Secondary 000000000001111111112222
Day Hours 012345678901234567890123  Day Hours 012345678901234567890123
Network:  ##### Full access #####          ##### Full access #####
Batch:    ----- No access -----          ----- No access -----
Local:    ----- No access -----          ----- No access -----
Dialup:   ----- No access -----          ----- No access -----
Remote:   ----- No access -----          ----- No access -----
Expiration:          (none)      Pwdminimum: 6   Login Fails: 0
Pwdlifetime:        (none)      Pwdchange:  (none)
Last Login:         (none) (interactive), (none) (non-interactive)
Maxjobs:           0  Fillm:      16  Bytln:       12480
Maxacctjobs:       0  Shrfillm:   0  Pbytln:        0
Maxdetach:         0  BIoIm:     12  JTquota:      1024
Prclm:             0  DIOIm:      6  WSdef:        180
Prio:              4  ASTIm:     16  WSquo:        200
Queprio:           0  TQElm:     10  WSextent:      0
CPU:              (none) Enqlm:    20  Pgflquo:    25600

Authorized Privileges:
    TMPMBX NETMBX
Default Privileges:
    TMPMBX NETMBX

```

### 12.4.3 システムへのデフォルトの DECnet アクセスの削除

デフォルトの DECnet アカウントは、厳しいセキュリティ要件を必要としないシステムに適しています (12.4 節参照)。中～高レベルのセキュリティ要件を必要とするサイトでは、個々のネットワーク・オブジェクトについてアカウントの設定が完了したら、システムへのデフォルトの DECnet アクセスを削除してください。

---

#### 注意

---

この節で説明したようにデフォルトの DECNET アカウントの削除を実行する前に、NCP の SHOW KNOWN OBJECTS コマンドと登録ユーティリティ (AUTHORIZE) を使用して、すべてのネットワーク・オブジェクトのネットワーク・アカウント、およびネットワーク・オブジェクトを使用するレイヤード・プロダクトのネットワーク・アカウントが、システム・ユーザ登録ファイル (SYSUAF.DAT) に設定されていることを確認します。設定されていない場合、ネットワーク・オブジェクト、およびネットワーク・オブジェクトを使用するレイヤード・プロダクトが想定どおりに動作しない場合があります。

デフォルトの DECnet アクセスを削除するには、ネットワーク構成データベース内の DECNET アカウントへのアクセスを削除し、SYSUAF から DECNET アカウントを削除します。

#### デフォルトの DECnet アクセスの削除

次の NCP コマンドを実行して、ネットワーク・エグゼキュータ・データベースからデフォルトの DECnet アクセスを削除します。

```
NCP> DEFINE EXECUTOR NONPRIVILEGED USER DEFAULT_DECNET
NCP> PURGE EXECUTOR NONPRIVILEGED PASSWORD
```

最初のコマンドに含まれている DEFAULT\_DECNET というユーザは、監査のためにのみ指定する、存在しないユーザ・アカウントです。(存在しない DEFAULT\_DECNET アカウントを使用してシステムへのアクセスが試みられるたびに、ネットワーク・ログイン失敗のメッセージがセキュリティ監査ログ・ファイルに書き込まれます。)

#### DECNET アカウントの削除

AUTHORIZE を使用して、次に示すように、SYSUAF から DECNET アカウントを削除します。

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> REMOVE DECNET
UAF> EXIT
```

[DECNET] ディレクトリ構造内に存在するすべてのファイルを削除します。

#### 運用時構成データベースの変更

変更をすぐに反映するには、次に示す NCP コマンドを使用して運用時データベースを変更します。

```
NCP>SET EXECUTOR NONPRIVILEGED USER DEFAULT_DECNET  
NCP>CLEAR EXECUTOR NONPRIVILEGED PASSWORD
```

### 12.4.4 遠隔オブジェクト接続の特権要件の設定

特定の特権を選択することで、ネットワーク設定時に指定した DECnet オブジェクトの使用を制御できます。この場合、特権 DECnet オブジェクトへ接続する操作、および発信 DECnet オブジェクトを使用する操作は特権操作となります。

たとえば、次のコマンドを実行すると、遠隔オブジェクトである MAIL への DECnet 接続を開始するユーザは、OPER 特権と SYSNAM 特権を持っていないとしないという要件が設定されます。

```
NCP>DEFINE OBJECT MAIL OUTGOING CONNECT PRIVILEGES OPER,SYSNAM
```

この仕組みは、特定の DECnet アプリケーションへのアクセスを特権ユーザまたは特権プログラムに限定する便利な方法です。ただし、これが有効性を発揮するには、特権の要件をネットワーク内のすべてのノードに一貫して適用する必要があります。

## 12.5 ルーティング初期化パスワードの指定

ポイント・ツー・ポイント接続は、同期回線および非同期回線を経由する接続のことです。ポイント・ツー・ポイント接続では、特にダイアルアップ回線経由の場合、ルーティング初期化パスワードを使用することで、開始ノードがローカル・ノードへの接続を許可されているかどうかを確認することができます。ポイント・ツー・ポイント・サーキットのそれぞれの側で、他方のノードに送信するベリファイアを設定するとともに、他方のノードから受け取るべきベリファイアを指定できます。接続確立の前に、それぞれのノードで、指定したベリファイアが相手側のノードから受け取ったかどうかを確認します。

パスワードの使用は、ポイント・ツー・ポイント接続では任意ですが、動的非同期接続では必須です。遠隔ノードが動的非同期接続（通常、電話による通話の間のみ維持される接続）を要求してきたときにセキュリティを高めるために、動的非同期接続を要求する側のノードはパスワードを送信しますが、ログイン要求を受信する側のノードは、接続を要求するノードに対してはパスワードを明らかにできないようになっています。接続を要求する側のノードのネットワーク・アドレス、ノード名、およびパスワードは、ローカル・システムのルーティング登録データと一致していなければなりません。



## 12.5.1 動的非同期接続の確立

動的非同期 DECnet 接続は、2 つのノードを結ぶ一時的な接続のことで、通常、モデムを利用して、電話回線経由で行われます。接続の両端の回線は、ターミナル回線から動的非同期 DECnet 回線に切り換えることができます。動的非同期回線の設定は、動的接続の確立時に DECnet によって自動的に行われます。動的非同期接続は通常、電話による通話が行われている間のみ維持されます。

---

### 注意

---

OpenVMS ノードに対する動的非同期接続は、DECnet 非同期 DDCMP プロトコルをサポートする任意のノードから開始できます。

---

OpenVMS ノードでは、ローカル・ノードにおいてネットワークを有効にする (手順 3) 前に、動的非同期接続プロセスの手順 1 および手順 2 を実行できます。以後の手順 (手順 4 以降) は、必ず回線が DECnet に切り替わっているときに実行する必要があります。

以下に示す手順に従って、動的非同期 DECnet 接続を確立します。この手順では、ローカルの OpenVMS ノードが接続を開始し、DECnet 用にターミナル回線を有効にします。接続は、NETMBX 特権を持つアカウントのある OpenVMS に対して行う必要があります。また、この手順は、動的非同期 DECnet 接続を正常に確立するために、遠隔の OpenVMS ノードのシステム管理者が実行しなければならない処理も示しています。

1. SYSTEM アカウントにログインし、次のコマンドを会話形式で入力します。または、システムをブートする前に SYS\$MANAGER:SYSTARTUP\_VMS.COM コマンド・プロシージャの中で指定します。これらのコマンドは、非同期ドライバの NODRIVER (NOA0) をロードし、DYN SWITCH ソフトウェアをインストールします。

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT NOA0/NOADAPTER
SYSGEN> EXIT
$ INSTALL:=$SYS$SYSTEM:INSTALL
$ INSTALL/COMMAND
INSTALL> CREATE SYS$LIBRARY:DYN SWITCH/SHARE -
_ /PROTECT/HEADER/OPEN
INSTALL> EXIT
```

遠隔の OpenVMS ノードのシステム管理者も上記のコマンドを入力する必要があります。

また、遠隔の OpenVMS ノードのシステム管理者は、以下に示すコマンドも入力する必要があります。これらのコマンドを実行すると、切り換え対象のターミナル回線の仮想ターミナルが使用できるようになり、ターミナル回線に

DISCONNECT 特性が設定されます。(仮想ターミナル機能により、使用している物理ターミナルが切断された場合でも、プロセスが継続して実行されます。)

```
$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT VTA0/NOADAPTER/DRIVER=TTDRIVER
SYSGEN> EXIT
$ SET TERMINAL/EIGHT_BIT/PERMANENT/MODEM/DIALUP -
_$ /DISCONNECT device-name:
```

*Device-name* は、動的非同期接続の接続先ターミナル・ポートの名前です。

2. 動的非同期ダイヤルアップ接続の発信側で、必要な送信パスワードを指定します。送信パスワードは、接続開始時に遠隔ノードに送信されるパスワードのことです。NCP を使用して、遠隔ノードの送信パスワードを指定するコマンドを入力します。パスワードには、1 ~ 8 文字の英数字を使用できます。スペースは使用できません。次のようにコマンドを指定します。

```
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE node-id TRANSMIT PASSWORD password
NCP> EXIT
```

*Node-id* は、ローカル・ノードが接続を確立しようとしている相手先の遠隔ノードの名前です。

次の例では、ローカル・ノードのノード名は LOCALA、送信パスワードは PASSA です。そして動的非同期ダイヤルアップ接続を確立しようとしている接続先の遠隔ノードの名前は REMOTC です。

```
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE REMOTC TRANSMIT PASSWORD PASSA
NCP> EXIT
```

動的非同期 DECnet ダイヤルアップ接続を作成する相手となる遠隔ノードごとに、コマンドを個別に使用して送信パスワードを指定する必要があります。

接続先の相手側ノードのシステム管理者は、接続元であるローカル・ノードが受信するパスワード(受信パスワード)として同じパスワードを指定する必要があります。遠隔側のシステム管理者は、INBOUND ROUTER パラメータまたは INBOUND ENDNODE パラメータを指定して、動的接続を開始するノードのタイプ(ルータまたはエンド・ノード)を指定する必要があります。次は、遠隔ノード側の管理者が入力するコマンドです。

```
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE node-id -
_ RECEIVE PASSWORD password INBOUND node-type
NCP> EXIT
```

たとえば、ローカル・ノードである LOCALA がエンド・ノードで、送信パスワードが PASSA の場合、REMOTC の管理者は次のコマンドを実行します。

```
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE LOCALA RECEIVE PASSWORD PASSA INBOUND ENDNODE
NCP> EXIT
```

3. 以下の手順は、必ず両方のノードで DECnet が実行されている状態で行います。まだその状態になっていない場合は、次のコマンドを入力して、ネットワークを有効にします。また、遠隔のシステム管理者にも同じ準備をしてもらうよう依頼します。

```
$ @SYS$MANAGER:STARTNET
```

動的非同期接続の手順を開始する前に、すでにネットワークが動作していた場合は、次のコマンドを入力して、パーマネント・データベースのエントリを運用時データベースに入力します。

```
$ RUN SYS$SYSTEM:NCP
NCP> SET NODE node-id ALL
NCP> EXIT
```

4. 以降の手順は、NETMBX 特権を持つ OpenVMS ユーザであれば実行できます。ローカルの OpenVMS システムにログインし、ローカルのターミナルに次の DCL コマンドを入力して、プロセスをターミナル・エミュレータ (遠隔ターミナルをローカル・ターミナル接続のように見せる機能) として機能させます。

SET HOST/DTE device-name:

*device-name* は、モデムが接続されているローカル・ターミナル・ポートの名前です。両方のシステムで、自動ダイヤル機能が備わっているモデムを使用する場合は、必要に応じて SET HOST/DTE コマンドに /DIAL 修飾子を追加し、遠隔ノードのモデムに自動でダイヤルすることができます。次のように指定します。

SET HOST/DTE/DIAL=number device-name:

5. 自動ダイヤル機能を使用しない場合は、遠隔ノードに手動でダイヤルアップします。
6. ダイヤルアップ接続が確立し、遠隔の OpenVMS システムのウェルカム・メッセージを受信したら、遠隔ノードの自分のアカウントにログインします。
7. 遠隔ノードの自分のアカウントにログインしている状態で、次のコマンドを入力し、回線が自動的に DECnet 回線に切り替わるようにします。

```
$ SET TERMINAL/PROTOCOL=DDCMP/SWITCH=DECNET
```

次のメッセージは、DECnet 接続が確立された状態であることを示すメッセージです。

```
%REM-S-END - control returned to local-nodename::
$
```

通信リンクを確立できたかどうかを調べるには、ローカル・システムで次のコマンドを入力します。

```
$ RUN SYS$SYSTEM:NCP
NCP> SHOW KNOWN CIRCUITS
NCP> EXIT
```

コマンドの実行結果には、回線に接続されている非同期デバイスに応じて、TT または TX という二モニックで示されるサーキットの一覧が表示され、ON の状態であることを示されます。

ローカル・ターミナルの画面に DCL プロンプトが表示されたら、非同期 DECnet 接続経由で遠隔ノードとの間のやり取りを開始できます。

8. ターミナル回線を DECnet 回線に自動的に切り換える方法 (前述の手順 7 の方法) の代わりに、手動で回線を切り換えることもできます。OpenVMS ソフトウェアを実行していないノードから OpenVMS ノードに対して動的接続を開始する場合は、手動による切り換えが必要です。OpenVMS システムから開始する場合は、手動による切り換えは任意です。OpenVMS ソフトウェアを実行していないノードから接続を開始するときは、システム固有の手順に従って、ターミナル・エミュレーション機能を使用して遠隔の OpenVMS ノードにログインします。

遠隔ノードへログインしたら、手動による切り換えを行うには 2 つの手順が必要です。

- a. 遠隔の OpenVMS ノード上の自分のアカウントを使用して、手順 7 で説明した SET TERMINAL コマンドを指定します。ただし、ここでは /MANUAL 修飾子を追加します。

```
$ SET TERMINAL/PROTOCOL=DDCMP/SWITCH=DECNET/MANUAL
```

遠隔のシステムが回線を DECnet の回線に切り換わることを示す、次のメッセージを遠隔ノードから受信します。

```
%SET-I-SWINPRG The line you are currently logged over is becoming  
a DECnet line
```

- b. ターミナル・エミュレータを終了して、手動で DECnet 回線に切り換えます。切り換えの手順は、ログインしているオペレーティング・システムによって異なります。

次の例は、動的接続を開始する OpenVMS ユーザの場合の切り換え手順です。

- ターミナル・エミュレータを終了するには、OpenVMS システム上で、バックスラッシュ (\) キーと Ctrl キーを同時に押します。
- 次のコマンドを入力して、ターミナル回線を手動で DECnet 回線に切り換えます。

```
$ SET TERMINAL/PROTOCOL=DDCMP TTA0:
```

TTA0 は、ローカル・ノード上のターミナル・ポートの名前です。

- NCP コマンドを入力して、ターミナル・ポートの TTA0 に接続されている回線とサーキットを手動で有効にします (次の例を参照)。

```

$ RUN SYS$SYSTEM:NCP
NCP> SET LINE TT-0-0 RECEIVE BUFFERS 4 -
_ LINE SPEED 2400 STATE ON
NCP> EXIT

```

これで、ローカルの OpenVMS ノード上で非同期 DECnet が開始します。

9. 動的非同期接続は、次のいずれかの方法で終了できます。

- 電話回線の接続を切断します。
- NCP を実行し、非同期の回線またはサーキットのいずれかを無効にします。接続の終了に使用できる 2 つのコマンド（回線とサーキット）は次のとおりです。

```

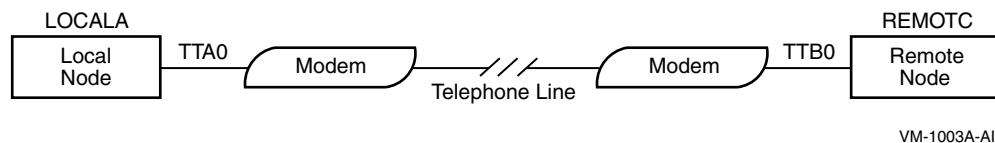
$ RUN SYS$SYSTEM:NCP
NCP> SET LINE dev-c-u STATE OFF
NCP> SET CIRCUIT dev-c-u STATE OFF
NCP> EXIT

```

遠隔ノードで上記のいずれかの NCP コマンドが入力されると、回線はただちにターミナル・モードに戻ります。ローカルの OpenVMS ノード（接続を開始した側）でコマンドが入力されると、遠隔側の回線とサーキットは有効な状態がおよそ 4 分間継続し、その後回線はターミナル・モードに戻ります。

図 12-2 は、動的非同期接続が確立する様子を示したものです。接続の両端で入力する必要のあるコマンドは、例 12-3 のとおりです。

図 12-2: 典型的な動的非同期接続



例 12-3: 動的非同期接続のコマンドの例

ローカル OpenVMS ノード (LOCALA) と遠隔 OpenVMS ノード (REMOTC) の両方で実行するコマンド:

```

$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT NOA0/NOADAPTER
SYSGEN> EXIT
$ INSTALL:=$SYS$SYSTEM:INSTALL
$ INSTALL/COMMAND
INSTALL> CREATE SYS$LIBRARY:DYN SWITCH/SHARE/PROTECT/HEADER/OPEN
INSTALL> EXIT

```

遠隔ノード (REMOTC) で実行するコマンド:

```

$ RUN SYS$SYSTEM:SYSGEN
SYSGEN> CONNECT VTA0/NOADAPTER/DRIVER=TTDRIVER
SYSGEN> EXIT

```

### 例 12-3: 動的非同期接続のコマンドの例 (続き)

---

```
$ SET TERMINAL/EIGHT_BIT/PERMANENT/MODEM/DIALUP/DISCONNECT TTB0:
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE LOCALA RECEIVE PASSWORD PASSA INBOUND ENDNODE
NCP> SET NODE LOCALA ALL
NCP> EXIT
ローカル・ノード (LOCALA) で実行するコマンド:
$ RUN SYS$SYSTEM:NCP
NCP> DEFINE NODE REMOTC TRANSMIT PASSWORD PASSA
NCP> SET NODE REMOTC ALL
NCP> EXIT
$ SET HOST/DTE/DIAL=8556543 TTA0:
! REMOTC に自動的にダイヤルアップした後,
! REMOTC の自分のアカウントにログインする。
$ SET TERMINAL/PROTOCOL=DDCMP/SWITCH=DECNET
%REM-S-END - control returned to LOCALA:
$
```

---

## 12.6 ネットワークにおけるファイルの共有

ユーザが、パスワードを共有したり、ファイルやディレクトリの保護コードを変更してワールド・カテゴリに対する読み込みアクセス権または実行アクセス権を付与したりしないようにします。BYPASS 特権または READALL 特権の付与は慎重に行います。

ネットワーク環境において、ファイルを臨時に共有する場合は、メール・ユーティリティを使用するのが最も簡単です。この場合、共有相手にファイルをメールで送信するため、パスワードを知らせる必要がなく、その相手以外のユーザはファイルにアクセスできません。ただし、共有対象のファイルにアクセスしたいときに、ファイルの所有者に依頼し、応答を待たなければならないという短所があります。共有ファイルへのアクセスが頻繁に発生する状態が継続する場合は、ディレクトリとファイルに対して、代理アカウントや ACL を設定する方が効率的です。

### 12.6.1 メール・ユーティリティの使用

ユーザがテキスト・ファイルを別のユーザに転送する最も簡単な方法は、メール・ユーティリティ (MAIL) を実行し、相手ユーザにファイルを送信することです。この方法は、パスワードを明かす必要がなく、ファイルの保護も変更されないため、比較的安全です。受信側のユーザは、MAIL の EXTRACT/NOHEADER コマンドに新しいファイル名を指定するだけで、自分のディレクトリにファイルのコピーを作成できます。コピーしたファイルには、受信ユーザのデフォルトの保護が自動的に設定されます。続いて、受信ユーザは、MAIL の DELETE コマンドを使用して、メール・ファイルからファイルを削除します。

## 12.6.2 ローカル・ユーザおよび遠隔ユーザのアカウントの設定

ネットワーク管理者は、特定の作業のために、ローカル・ノード上のディレクトリに対するアクセスを外部ノードの何人かのユーザに許可する必要があることがあります。そのため、代理アカウントを作成し、その1つの代理アカウントに対して外部ユーザのアクセスを許可する代理アクセスを追加します(12.3.2.3 項参照)。このアカウントのディレクトリに置かれているファイルを共用する必要のあるローカル・ユーザがいる場合は、それらのユーザに必要なアクセス権を与え、部外者によるアクセスから守るために、ディレクトリとファイルにACLを適用します。

ある企業が、全従業員がアクセス可能な、営業の最新情報を蓄積した集中リポジトリを必要としているとします。

1. ファイルを格納するノード (BNORD) のセキュリティ管理者が、SALES\_READER という特別なアカウントを作成します。SALES\_READER アカウントは、メール機能が無効になっているキャプティブ・アカウントとして設定されます。デフォルトのディレクトリは、[SALESINFO] で、次に示す保護コードがデフォルトで設定されています。

```
(S:RWED,O:RWED,G:R,W)
```

この保護コードにより、ホーム・ノードである BNORD 上の SALES\_READER と同じグループに属するユーザは、ファイルの読み込みが可能になります。さらに、システム・カテゴリまたは所有者カテゴリに属しているユーザのみ、または同等のアクセスを認める特権を持つユーザのみが、当該ディレクトリ内のファイルを更新できます。ACL を使用して、アクセスをさらに詳細に定義します(手順 3 参照)。

2. セキュリティ管理者は、AUTHORIZE の ADD/PROXY コマンドを使用して、外部ユーザのための代理アクセスを追加します。たとえば、代理アクセスを、DEXTER というノードの Jackson というユーザと、BANGOR というノードの Goodwin というユーザにまで拡張するには、次のコマンドを使用します。

```
UAF> ADD/PROXY DEXTER::JACKSON SALES_READER/DEFAULT
UAF> ADD/PROXY BANGOR::GOODWIN SALES_READER/DEFAULT
```

3. 後になってホーム・ノード BNORD の他のユーザもアクセスの必要があり、当該ユーザが SALES\_READER と同じグループに属していないことがわかった場合は、[SALESINFO] ディレクトリ内のファイルに ACL を追加できます。たとえば、R. Grant がすべてのファイルに対する制御アクセス、J. Martinez はすべてのファイルに対する読み込みアクセスを必要としているとします。次の2つの DCL コマンドを実行することで、ディレクトリに対する ACL を定義した後、その ACL を既存のすべてのファイルに継承させることができます。

```
$ SET SECURITY/ACL=-
_$( (IDENTIFIER=R_GRANT,ACCESS=CONTROL), -
_$( (IDENTIFIER=J_MARTINEZ,ACCESS=READ)) -
```

```

_$( (IDENTIFIER=R_GRANT,OPTIONS=DEFAULT,ACCESS=CONTROL), -
_$( IDENTIFIER=J_MARTINEZ,OPTIONS=DEFAULT,ACCESS=READ)) -
_$( [000000]SALESINFO.DIR
$ SET SECURITY/DEFAULT *.*;*

```

### 12.6.3 複数アカウントに対する遠隔ユーザの許可

少数の外部ユーザが、それぞれ異なる理由から、特別な保護が設定されているファイルへのアクセスを必要とする場合は、複数の代理アカウントへのアクセスを設定し、広範な ACL を適用します。

たとえば、数多くの支社を持つ大規模な企業の場合、個別のファイル共用目的のために複数の代理アカウントを作成することが考えられます。中央の本社で、東部地域の 2 つのノードの 2 名の主要ユーザに対し、LEVIGRAY というプロジェクト・コード名のプロジェクト・ファイルに対する読み込みアクセス権と書き込みアクセス権、そして BETSEYHARLOW というプロジェクトのファイルへの読み込み専用アクセス権を与えることを考えているとします。同時に、西部地域のユーザで、LEVIGRAY ファイルへの読み込みアクセス権と、BETSEYHARLOW ファイルへの読み込みおよび書き込みアクセス権が必要な 3 名のユーザも存在するとします。中央本社の 2 名のユーザだけが、LEVIGRAY ファイルへの完全なアクセス権を持ち、そして本部の別の 2 名のユーザが BETSEYHARLOW ファイルへの完全なアクセス権を持ちます。作業のために、この状況を表形式でまとめると、例 12-4 のようになります。

例 12-4: 保護ファイルのネットワークにおける共用

---

	CENTRL::PROJ:[DESGN_PROJECTS] へのアクセス要件	
	所有者は [DESIGNERS,MGR]	
ユーザとノード	サブディレクトリ LEVI プロジェクト・ファイル LEVIGRAY*.*	サブディレクトリ BETSEY プロジェクト・ファイル BETSEYHARLOW*.*
FRISCO::ALBION	R	RW
FRISCO::ELTON	R	RW
LA::IRVING	R	RW
CENTRL::DIANTHA	RWED	NONE
CENTRL::BRITTANIA	RWED	NONE
CENTRL::ALBERT	NONE	RWED
CENTRL::DELIA	NONE	RWED
BOS::AYLMER	RW	R
WASH::LAVINA	RW	R

---



次に示す解決方法では、CENTRL というノード上の 4 つのローカル・アカウントに加え、5 つの代理アカウントを使用し、ディレクトリ、サブディレクトリ、およびファイルに対して ACL を使用します。

1. 本部のセキュリティ管理者は、AUTHORIZE を使用して、遠隔ユーザの Albion, Elton, Irving, Aylmer, Lavina の 5 名に対して CENTRL ノード上に新規の代理アカウントを作成します。これらのアカウントは、キャプティブ・アカウントとし、メールは使用不可、そしてネットワーク・アクセスのみに限定します。また、これらのアカウントは、CLI テーブルを使用して DCL のサブセットに限定します。各ユーザのデフォルトのディレクトリは、[DESGN\_PROJECTS] とします。セキュリティ管理者は、ファイルの想定用途に合わせて DESIGNERS グループにこれらのアカウントを入れることにします。

Diantha, Brittainia, Albert, および Delia というユーザについてはすでにアカウントが存在しています。これらのアカウントは必ずしも同じグループには属する必要はありません。これらのユーザには、作業に使用するデバイスとディレクトリを通知します。

2. 次に、代理レコードをネットワーク代理登録ファイルに追加します。次の AUTHORIZE コマンドを使用します。

```
UAF> ADD/PROXY FRISCO::ALBION ALBION/DEFAULT
UAF> ADD/PROXY FRISCO::ELTON ELTON/DEFAULT
UAF> ADD/PROXY LA::IRVING IRVING/DEFAULT
UAF> ADD/PROXY BOS::AYLMER AYLMEER/DEFAULT
UAF> ADD/PROXY WASH::LAVINA LAVINA/DEFAULT
```

3. CENTRL ノードのセキュリティ管理者は、[DESGN\_PROJECTS] の最上位ディレクトリに ACL を適用します。次の DCL コマンドを使用します。

```
$ SET SECURITY/ACL=(DEFAULT_PROTECTION,S:RWED,O,G,W) -
_$ [000000]DESGN_PROJECTS.DIR
```

この ACL を適用することにより、システム・カテゴリに属さないユーザは、当該ディレクトリとそのサブディレクトリに格納されているファイルに対して、BYPASS 特権を持っている場合を除き、UIC ベースのアクセスはできなくなります。実のところ、この制限は DESIGNERS グループの 5 名のユーザにも適用されます。目標は、特定のユーザのグループにのみアクセスを許可する ACL をすべてのファイルに適用することです。この保護コードが、当該ディレクトリとそのサブディレクトリに格納されるすべてのファイルに継承されるのが理想的です。(さらなる保護のためファイルに対して設定される ACL は、実際にいずれかのユーザがファイルへのアクセスを要求する場合に優先して適用されます。)

4. [DESGN\_PROJECTS] の下に、次の 2 つのサブディレクトリが作成されます。
  - [DESGN\_PROJECTS.LEVI]

- [DESGN\_PROJECTS.BETSEY]

5. セキュリティ管理者は、ACL エディタを使用して、最上位ディレクトリの ACL に次の ACE を追加します。

```
DESGN_PROJECTS.DIR
```

```
(IDENTIFIER=DIANTHA, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=BRITTANIA, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=ALBERT, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=DELIA, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=AYLMER, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=LAVINA, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=ALBION, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=ELTON, OPTIONS=PROTECTED, ACCESS=EXECUTE)
(IDENTIFIER=IRVING, OPTIONS=PROTECTED, ACCESS=EXECUTE)
```

これらの保護 ACE により、選択した 9 名のユーザのみが最上位ディレクトリにアクセス可能となります。ACL によって最上位ディレクトリへの書き込みまたは削除のアクセスを許可されているユーザはいないため、最上位ディレクトリとそのサブディレクトリは、ファイルの削除と名前変更からは基本的に保護されます。(もちろん、システム・カテゴリのユーザは、UIC ベースの保護機能を通じて、読み込みと削除のアクセスが可能です。)

6. 次に、セキュリティ管理者は、サブディレクトリに対する ACL を作成します。それぞれのサブディレクトリに必要な ACE は次のとおりです。

```
[DESGN_PROJECTS]LEVI.DIR
```

```
(IDENTIFIER=DIANTHA, OPTIONS=PROTECTED, ACCESS=READ+WRITE+EXECUTE+CONTROL)
(IDENTIFIER=DIANTHA, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE+EXECUTE
+DELETE+CONTROL)
(IDENTIFIER=BRITTANIA, OPTIONS=PROTECTED, ACCESS=READ+WRITE+EXECUTE+CONTROL)
(IDENTIFIER=BRITTANIA, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE+EXECUTE
+DELETE+CONTROL)
(IDENTIFIER=AYLMER, OPTIONS=PROTECTED, ACCESS=READ+WRITE)
(IDENTIFIER=AYLMER, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE)
(IDENTIFIER=LAVINA, OPTIONS=PROTECTED, ACCESS=READ+WRITE)
(IDENTIFIER=LAVINA, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE)
(IDENTIFIER=ALBION, OPTIONS=PROTECTED, ACCESS=READ)
(IDENTIFIER=ALBION, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ)
(IDENTIFIER=ELTON, OPTIONS=PROTECTED, ACCESS=READ)
(IDENTIFIER=ELTON, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ)
(IDENTIFIER=IRVING, OPTIONS=PROTECTED, ACCESS=READ)
(IDENTIFIER=IRVING, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ)
```

```
[DESGN_PROJECTS]BETSEY.DIR
```

```
(IDENTIFIER=ALBERT, OPTIONS=PROTECTED, ACCESS=READ+WRITE+EXECUTE+CONTROL)
(IDENTIFIER=ALBERT, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE+EXECUTE
+DELETE+CONTROL)
(IDENTIFIER=DELIA, OPTIONS=PROTECTED, ACCESS=READ+WRITE+EXECUTE+CONTROL)
(IDENTIFIER=DELIA, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE+EXECUTE
```

```

+DELETE+CONTROL)
( IDENTIFIER=ALBION, OPTIONS=PROTECTED, ACCESS=READ+WRITE)
( IDENTIFIER=ALBION, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE)
( IDENTIFIER=ELTON, OPTIONS=PROTECTED, ACCESS=READ+WRITE)
( IDENTIFIER=ELTON, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE)
( IDENTIFIER=IRVING, OPTIONS=PROTECTED, ACCESS=READ+WRITE)
( IDENTIFIER=IRVING, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ+WRITE)
( IDENTIFIER=AYLMER, OPTIONS=PROTECTED, ACCESS=READ)
( IDENTIFIER=AYLMER, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ)
( IDENTIFIER=LAVINA, OPTIONS=PROTECTED, ACCESS=READ)
( IDENTIFIER=LAVINA, OPTIONS=DEFAULT+PROTECTED, ACCESS=READ)

```

上記のどちらの ACL にも，識別子ごとに 2 つの ACE が指定されています。最初の ACE は，サブディレクトリへのアクセスを制御するためのものです。この ACE は，サブディレクトリ保護のために削除アクセスを拒否するもので，サブディレクトリ内に作成されるどのファイルにも継承されません。それぞれの識別子の 2 つめの ACE は，Default 属性が含まれているため，サブディレクトリに追加されたすべてのファイルに自動的に継承されます。さらには，Protected 属性により，すべての ACE が，一部の処理によるものを除いて，削除から保護されます。

ここまでで，すべての基本作業は完了です。時間の経過とともに，サブディレクトリにファイルが追加されていきます。そのため，Washington にいるユーザ Lavina が次の DCL コマンドを入力すると，ファイル LEVIGRAYMEM3.MEM は WASH というノードで印刷されます。

```
$ COPY CENTRL::LEVIGRAYMEM3.MEM LP:
```

しかし，ユーザ Lavina がこのファイルを編集しようとするとき，このユーザは ACL によって書き込みアクセスを拒否されるため，編集操作は失敗します。

この構想に多数のユーザが関わっていた場合は，ユーザに追加の識別子を付与するのが有効です。たとえば，LEVI サブディレクトリのファイルに対する読み込みアクセスを許可するユーザに，LEVI\_READER という識別子を与えます。識別子を追加することで，ACL を短縮できます。



## 保護サブシステムの使用

OpenVMS オペレーティング・システムでは、セキュリティ制御の大部分はユーザ ID に基づいています。ファイルやデバイスなどの保護オブジェクトは、ユーザ単位または複数のユーザからなるグループ単位でアクセスできます。オブジェクトの ACL または保護コードが、必要なアクセス権限をユーザに与えている場合、そのユーザは利用可能な任意のソフトウェアを使用してそのオブジェクトを使用できます。OpenVMS のオブジェクト保護の説明については、第 4 章 を参照してください。

保護サブシステムでは、通常のアクセス制御で保護されたアプリケーションが、当該サブシステムに属するオブジェクトの門番の役割を果たします。ユーザは、門番として機能するアプリケーションを実行しない限り、サブシステムのオブジェクトにはアクセスできません。ユーザがアプリケーションを実行すると、ユーザのプロセス・ライト・リストが識別子を取得し、サブシステムが所有するオブジェクトへのアクセスが許可されます。ユーザがアプリケーションを終了すると、これらの識別子と、それらによってユーザが得たオブジェクトへのアクセス権が直ちに失われます。

この章では、保護サブシステムの詳細と、その構築方法を説明します。

### 13.1 保護サブシステムの利点

保護サブシステムの利用には、次のような利点があります。

- 保護サブシステムでは、OpenVMS の従来のアクセス制御にない、条件に基づくデータへのアクセス制御が可能な仕組みを利用できます。従来は、ユーザに特権を付与することで、保護コードやアクセス制御リスト (ACL) の適用を回避していました。しかし、このような特権を与えることは、ユーザに幅広いアクセス権限を与えることになります。さまざまな特権に設定されている権限の詳細については、付録 A を参照してください。保護サブシステムによって、個々のユーザが広範囲に特権を使用することを回避できます。
- 保護サブシステムは、特権付きでイメージをインストールする代替の手段となります。特権付きの安全なイメージを作成するには一定の技術が必要です。不適切なイメージを作成してしまうと、システムのセキュリティが損なわれる恐れがあります。
- 保護サブシステムは、保護された共用可能イメージ (ユーザ記述のシステム・サービス) の作成に代わる手段を提供します。

- 保護サブシステムでは、非特権ユーザがセキュリティ管理者の支援をそれほど受けずに保護サブシステムを管理できるため、システム管理の負担が軽減されます。システム管理要件の詳細については、13.5 節を参照してください。

## 13.2 保護サブシステムの適用範囲

保護サブシステムには、データベースから一般的なシステム管理に至るまで、さまざまな適用対象があります。

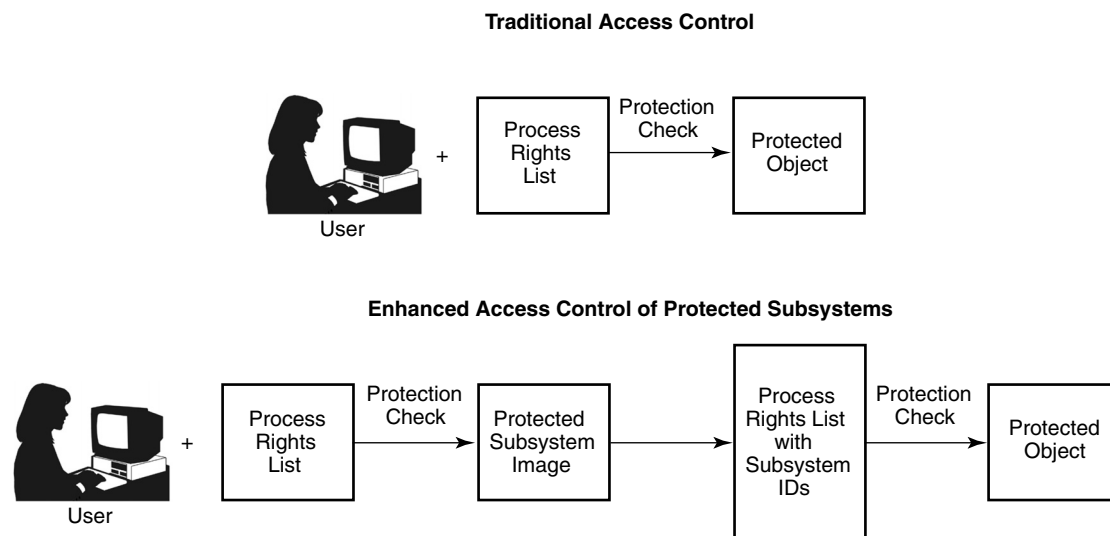
たとえば、グループのメンバ全員が利用できる、グループ・メンバ・リストなどが考えられます。このリストには、グループ・メンバの名前、住所、管理番号、関心事項が記載されています。メンバ・リストを保護サブシステムとして設定すると、グループのすべてのメンバが、特定の情報を参照したり、特定の種類の情報を更新したりできます。

また保護サブシステムによって、共用の場所にあるプリンタに機密情報が送信される危険性の問題なども解決できます。たとえば、データに機密情報が含まれていないかをチェックするアプリケーションを作成できます。機密ファイルは制限された場所のプリンタに送信し、共用のファイルは利用可能な任意のプリンタに送信することなどが可能になります。アプリケーションの実行アクセス権を持っているユーザであれば、制限されているプリンタを使用できますが、保護サブシステムを通じてのみとなります。

## 13.3 保護サブシステムの仕組み

保護サブシステムは、アプリケーションであり、そのアプリケーションを実行するプロセスに1つ以上の識別子を割り当てます。ユーザがサブシステムを実行している間は、ユーザのプロセス・ライト・リストにこれらの識別子が維持されます。図 13-1は、保護サブシステムが従来のアクセス制御にどのようにアクセス制御の層を付け加えるかを示します。

図 13-1: 通常のアクセス制御と保護サブシステムとの違い



VM-1004A-AI

アプリケーションに対する実行アクセス権を持つユーザは、サブシステムにアクセスできます。サブシステムにアクセスすると、ユーザはデータ・ファイルやサブシステムの他の資源を使用して作業ができます。

サブシステムが消費する資源(ファイルやプリンタなど)はそれぞれ異なる方法で保護できるため、サブシステムは複数の識別子を持つことができます。

サブシステム識別子の保有は、ユーザがアプリケーションを実行している間に限定されています。ユーザがアプリケーションを終了すると、識別子はプロセス・ライト・リストから削除されます。サブシステム識別子は、ユーザが Ctrl/Y を入力するか、DCL コマンドの SPAWN を使用してサブプロセスを作成しようとした場合にも、ライト・リストから削除されます。この点では、サブシステム識別子の使用は、特権付きでインストールされているイメージの操作と同じと言えます。

次の識別子は、セキュリティ・サブシステム用に予約されているため、ユーザには割り当てないでください。

- SECSRV\$CLIENT
- SECSRV\$COMMUNICATION
- SECSRV\$OBJECT

## 13.4 設計に関する検討事項

保護サブシステム用のアプリケーションを開発する場合は、/DEBUG 識別子および /TRACEBACK 識別子を付けずにアプリケーション・イメージをリンクしなければなりません。

この種のサブシステムは多くの場合、特権の必要性があらかじめ排除されていますが、アプリケーションを特権付きでインストールすることは可能です。たとえば、パーマネント・グローバル・セクションを作成するために PRMGBL 特権を必要とするアプリケーションや、セキュリティ監査レコードをシステムのセキュリティ監査ログ・ファイルに送るために AUDIT 特権が必要なアプリケーションがあります。よほどの理由がない限り、ALL カテゴリに属する特権付きで保護サブシステム・アプリケーションをインストールすることは避けてください。このカテゴリには、BYPASS、CMKRNL、SYSPRV など、OpenVMS のアクセス制御をユーザが無効にできる特権が含まれています。OpenVMS の特権の一覧については表 8-2 を、特権の説明については付録 A を参照してください。

サブシステム設計者は、サブシステムが意図どおりに動作するのに必要な識別子のリストを作成する必要があります。次にサブシステム設計者は、セキュリティ管理者に 13.5 節に示す準備をするよう依頼します。

## 13.5 システム管理の要件

非特権ユーザでも保護サブシステムを作成し、管理することはできますが、サブシステム用に必要な識別子を作成する最初の段階と、保護システムが格納されているボリュームをマウントする最終段階は、セキュリティ管理者が担当する必要があります。

セキュリティ管理者は、次の作業を行う必要があります。

1. サブシステム用の識別子を作成し、それぞれに Subsystem 属性を設定します。Subsystem 属性は、識別子の保持者にサブシステムを管理する権限を与える属性です。
2. Subsystem 属性を持つサブシステム識別子を、サブシステムの管理者の役割を果たす人物に付与します。これによりサブシステム管理者は、サブシステム識別子を、サブシステムを構成するイメージに割り当てることができるようになります。
3. サブシステム管理者に、アプリケーション・イメージへの制御アクセス権を与えます。サブシステム管理者は、サブシステム ACE をイメージ ACL に追加するために制御アクセス権が必要です。
4. 保護サブシステムの管理対象となる既存の資源に対する制御アクセス権をサブシステム管理者に割り当てます。

サブシステム管理者には、主要なシステム資源に対する制御アクセス権が必要になることがあります。オブジェクトを対象とした ACL により、サブシステム管理者のアクセス権がこれらの資源のみに制限されます。この方法は、SYSPRV 特権付きでイメージをインストールするよりは危険性が低いと言えます。



次は、ユーザがメンバ・リストを管理できるように、識別子と必要なアプリケーション・アクセス権を設定する方法の例です。

#### 例 13-1: メンバ・リスト管理用の識別子とアプリケーション・アクセス権の設定

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF>ADD/IDENTIFIER MEMBERS_SUBSYSTEM- [1]
_UAF> /ATTRIBUTES=(SUBSYSTEM,RESOURCE)
UAF>GRANT/IDENTIFIER MEMBERS_SUBSYSTEM - [2]
_UAF> /ATTRIBUTES=(SUBSYSTEM,RESOURCE) LOUIS
UAF>GRANT/IDENTIFIER MEMBERS_SUBSYSTEM -
_UAF> /ATTRIBUTES=(SUBSYSTEM,RESOURCE) WU
$ SET SECURITY/ACL=(IDENTIFIER=MEMBERS_SUBSYSTEM,- [3]
_$ ACCESS=CONTROL) MEMBER_LIST.EXE
```

1. AUTHORIZE を使用して、MEMBERS\_SUBSYSTEM というサブシステム識別子を作成します。この識別子には Subsystem 属性が設定されています。
2. Louis と Wu を識別子の保持者に設定し、この 2 名がサブシステムを管理できるようにします。
3. Louis と Wu に対して MEMBER\_LIST.EXE への制御アクセス権を割り当てます。

MEMBERS\_SUBSYSTEM というサブシステム識別子を作成し、Resource 属性を設定します。これにより、サブシステムにアクセスする個人ではなく、MEMBERS\_SUBSYSTEM 識別子にディスク領域を割り当てることができます。(Resource 属性を使用するときは、ディレクトリに対して適切な ACL を設定するように注意してください [ 8.8.1.2.3 項参照]。)

## 13.6 サブシステムの構築

サブシステム管理者は、13.5 節の手順に従って、必要な識別子とアクセス権を設定された後は、サブシステム・イメージに対して必要な ACE を追加することができます。サブシステム・イメージを構築するには、アプリケーション・イメージに適用されるサブシステム ACE と、サブシステムが管理するオブジェクトに適用される識別子 ACE の 2 種類の ACE が必要です。そのため、サブシステムを構築するには、次の手順を実行する必要があります。

1. アプリケーション・イメージの ACL 内にサブシステム識別子を含むサブシステム ACE を作成します。サブシステム ACE の形式は、次のとおりです。

```
(SUBSYSTEM,{IDENTIFIER=identifier[,ATTRIBUTES=attributes]})
```

2. サブシステムが管理するオブジェクトに対するアクセス権を付与します。サブシステムに帰属するさまざまなオブジェクトの ACL に識別子 ACL を追加する必要があります。それぞれの識別子 ACE には、次の形式で記述されたサブシステム識別子の 1 つが含まれています。

```
(IDENTIFIER=identifier, ACCESS=access-type[+...])
```

次の例では、サブシステム管理者が、DCL の SET SECURITY コマンドを使用して、サブシステムを構成するイメージにサブシステム識別子を関連付けています。まず、サブシステム管理者は、MEMBERS\_SUBSYSTEM 識別子を持つサブシステム ACE を、MEMBER\_LIST.EXE アプリケーション・イメージの ACL に追加します。

```
$ SET SECURITY/ACL=(SUBSYSTEM,IDENTIFIER=MEMBERS_SUBSYSTEM,-  
_ $ ATTRIBUTES=RESOURCE) MEMBER_LIST.EXE
```

次に、MEMBERS\_SUBSYSTEM サブシステム識別子を持つ識別子 ACE を、サブシステムが管理するデータ・ファイルに追加します。

```
$ SET SECURITY/ACL=(IDENTIFIER=MEMBERS_SUBSYSTEM,-  
_ $ ACCESS=READ+WRITE) MEMBER_DATA*.DAT
```

DCL の SHOW SECURITY コマンドを使用して、ファイルのセキュリティ属性を表示できます。次に例を示します。

```
$ SHOW SECURITY MEMBER_LIST.EXE
```

MEMBER\_LIST.EXE object of class FILE

```
Owner: [STAFF]  
Protection: (System: RWED, Owner: RWED, Group, World: RE)  
Access Control List:  
(SUBSYSTEM, IDENTIFIER=MEMBERS_SUBSYSTEM, ATTRIBUTES=RESOURCE)
```

```
$ SHOW SECURITY MEMBER_DATA*.DAT
```

MEMBER\_DATA\_1.DAT object of class FILE

```
Owner: MEMBERS_SUBSYSTEM  
Protection: (System: RWED, Owner: RWED, Group, World)  
Access Control List: (IDENTIFIER=MEMBERS_SUBSYSTEM, ACCESS=READ+WRITE)
```

MEMBER\_DATA\_2.DAT object of class FILE

```
Owner: MEMBERS_SUBSYSTEM  
Protection: (System: RWED, Owner: RWED, Group, World)  
Access Control List: (IDENTIFIER=MEMBERS_SUBSYSTEM,  
ACCESS=READ+WRITE)
```

## 13.7 トラストッド・ボリュームにおける保護サブシステムの有効化

SECURITY 特権を持つユーザは、MOUNT コマンドに /SUBSYSTEM 修飾子を指定することにより、ボリュームにおけるサブシステムを有効にすることができます。デフォルトでは、サブシステムはシステム・ディスクについてのみ有効になっています。その他のディスクについては、ボリュームをマウントするたびにサブシステムを有効にする必要があります。

次の例では、セキュリティ管理者が、MOUNT コマンドに /SUBSYSTEM 修飾子を指定して、DUA0 というデバイスにおけるサブシステム ACE の処理を有効にしています。このディスクには、MEMBERS\_SUBSYSTEM 識別子が設定されたサブシステムが含まれているものと仮定します。

```
$ MOUNT /SUBSYSTEM /SYSTEM DUA0: DOC WORK8
```

サブシステム ACE の処理は、DCL の SET VOLUME /SUBSYSTEM コマンドを使用することで、動的に有効/無効を切り換えることができます。このコマンドは、MOUNT コマンドを使用してマウントされないシステム・ディスクの場合に特に有用です。

サブシステムをマウントするユーザは、マウントするボリュームに何が含まれているかを把握している必要があります。把握していない場合、そのオペレータまたはシステム管理者は、システム・セキュリティを不用意に無効にしてしまう可能性があります。たとえば、あるクラスタにおいて特権を持つユーザが、サブシステム識別子を有するアプリケーションをボリュームに置き、別のクラスタの無警戒のオペレータにそのボリュームをマウントするよう要請することが簡単にできてしまいます。アプリケーションは、適切なサブシステム識別子を有しているため、権限のないサブシステムでメンバを装うことができます。このため、信用できるユーザのボリュームのみマウントするか、またはサブシステムを有効にしてボリュームをマウントする前に、サブシステム ACE が含まれているかどうかボリュームを徹底的に検索します。

## 13.8 ユーザへのアクセス権の付与

サブシステムのメイン・アプリケーション・イメージに対する実行アクセス権を持つユーザはすべて、サブシステムがアクセスを許可している場合に、サブシステムの管理下にあるデータ・ファイルやその他のオブジェクトを利用できます。ただし、サブシステムの管理者は、次の方法でサブシステムのオブジェクトへのアクセスを禁止できます。

- サブシステム管理者は、サブシステムに帰属する資源のうち、すべてのメンバにはアクセスを許可したくない資源について特別な識別子を作成し、これらの資源に ACE を追加することができます。
- ACE の中で表現を組み合わせることで、条件に応じてアクセス権を指定できます。たとえば、次に示す ACE は、MEMBERS\_SUBSYSTEM を実行している MEMBERS\_ADMIN にアクセスを許可しますが、MEMBERS\_ADMIN のみ、および MEMBERS\_SUBSYSTEM 識別子を有する他のユーザに対してはアクセスを許可しません。

```
(ID=MEMBERS_SUBSYSTEM+MEMBERS_ADMIN, ACCESS=READ+WRITE)
```

ユーザがサブシステムのアプリケーション・イメージを実行している間は、そのユーザのプロセス・ライト・リストに、そのユーザの通常の識別子だけでなく、サ

ブシステム識別子も含まれていることを忘れないでください。ただし、ユーザがアプリケーションに対して割り込み操作を行うか、または終了すると、そのユーザのプロセス・ライト・リストに含まれるサブシステム識別子は直ちに消滅し、サブシステム内のオブジェクトへのアクセス権を失います。サブシステム識別子は、デフォルトでは、サブプロセスが生成されたときに継承されません。

## 13.9 保護サブシステムの例

R. D. Taylor Inc. というサプライ品製造の会社が、購買部門と仕入先勘定部門のために保護サブシステムを設定することにしました。これらの部門は、別々に分かれています。サプライヤからの購買状況を記録する共通のデータベースを共用しています。

同社の在庫が、設定した数量を下回ると、購買部門は必要なサプライ品の注文するよう指示されます。購買担当者は、(必要であれば) サプライヤを探し、注文番号を割り当て、注文を出します。

商品が到着すると、受領と品質管理を担当する部門が、到着したものと注文内容とをつき合わせて確認し、品質基準を満たしているかどうかを調べ、合格したものを在庫に加えます。在庫処理が済むと、その情報が仕入先勘定部門に送られ、そこで請求書の処理が行われます。

仕入先勘定部門の管理者は、請求書と注文書とをつき合わせて確認し、サプライヤへの毎週の支払金額を計算するために支払いプログラムを実行します。支払い情報はデータベースに記録され、小切手は会社の小切手用紙をセットしたプリンタで印刷されます。

サブシステムを使用することで、同社は 2 つの目的をかなえることができます。

- 購買担当者に、社のデータベースに記録されている注文の参照およびデータベースへの注文の記録を行うライトを与え、仕入先勘定部門の担当者にはサプライヤの請求書を調べるライトを与えます。これらの作業を行う購買担当者は、SUPPLIERS\_ORDERS 識別子を有します。仕入先勘定部門の担当者は、ACCOUNTS\_PAYABLE 識別子を有します。

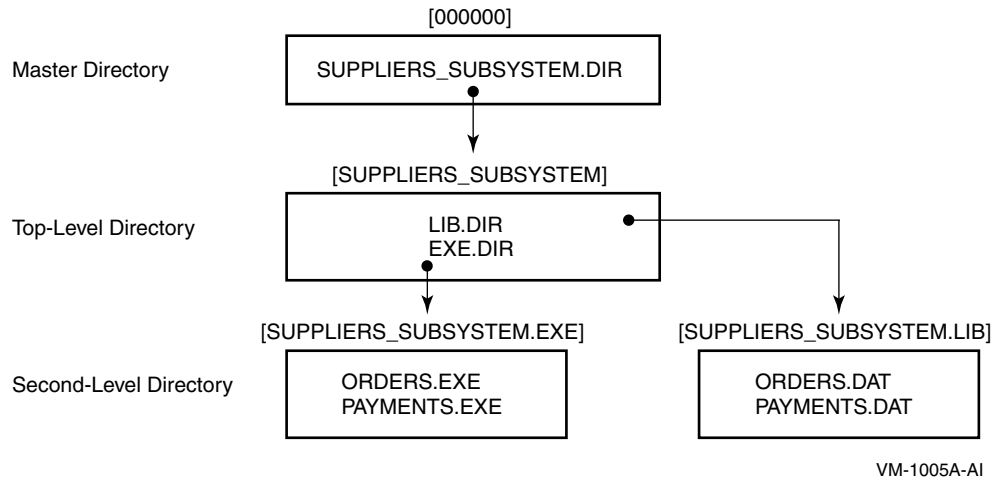
これらの従業員は、ORDERS.EXE を実行して、サプライヤの情報を更新します。このプログラムは、ORDERS.DAT にデータを格納します。

- プログラムは、仕入先勘定部門の信頼できる管理者に、データベースの更新、支払金額の計算、および小切手の印刷を行うライトを与えます。(小切手の印刷には、同社の小切手用紙をセットした 1 台のプリンタを使用します。) 仕入先勘定部門の管理者は、ACCOUNTS\_PAYABLE 識別子を有します。

仕入先勘定部門の管理者は、PAYMENTS.EXE を実行して上記の作業を行います。このプログラムは、完了した支払いを PAYMENTS.DAT データ・ファイルに記録します。

同社は、McGrey をサブシステムの設計および管理の担当者に任命しました。図 13-2 は、Taylor 社のサブシステムのディレクトリ構造です。例 13-6 は、このサブシステムを実装するため、McGrey が作成したコマンド・プロシージャです。

図 13-2: Taylor 社のサブシステムのディレクトリ構造



### 13.9.1 最上位ディレクトリの保護

McGrey は、ユーザが、必要な識別子を有している場合にのみサブシステムにアクセスできるようなディレクトリ構造を実装しました。具体的には、購買担当者は SUPPLIERS\_ORDERS 識別子を、仕入先勘定部門の管理者は ACCOUNTS\_PAYABLE 識別子を有します。サブシステム管理者として、McGrey は SUPPLIERS\_SUBSYSTEM 識別子を有します。

最上位ディレクトリの SUPPLIERS\_SUBSYSTEM.DIR には、次の例に示す保護が設定されています。

#### 例 13-2: SUPPLIERS\_SUBSYSTEM.DIR の保護

```

$ DIRECTORY/SECURITY SYS$SYSDEVICE:[000000]SUPPLIERS_SUBSYSTEM.DIR
Directory SYS$SYSDEVICE:[000000]
SUPPLIERS_SUBSYSTEM.DIR;1
      SUPPLIERS_SUBSYSTEM      (RWE,RWE,,)      [1]
(CREATOR,ACCESS=NONE) [2]
(DEFAULT_PROTECTION,SYSTEM:RWED,OWNER:RWED,GROUP:,WORLD:) [3]
(IDENTIFIER=SUPPLIERS_SUBSYSTEM,ACCESS=READ+WRITE+CONTROL) [4]
(IDENTIFIER=SUPPLIERS_ORDERS,ACCESS=EXECUTE) [5]
(IDENTIFIER=ACCOUNTS_PAYABLE,ACCESS=EXECUTE) [6]
(IDENTIFIER=*,ACCESS=NONE) [7]
(IDENTIFIER=SUPPLIERS_SUBSYSTEM,OPTIONS=DEFAULT,ACCESS=READ+WRITE+CONTROL) [8]
(IDENTIFIER=SUPPLIERS_ORDERS,OPTIONS=DEFAULT,ACCESS=EXECUTE)
(IDENTIFIER=ACCOUNTS_PAYABLE,OPTIONS=DEFAULT,ACCESS=EXECUTE)
(IDENTIFIER=*,OPTIONS=DEFAULT,ACCESS=NONE)
  
```

Total of 1 file.

- このディレクトリ保護コードは、システム・カテゴリと所有者カテゴリに属するユーザに対して、読み込み、書き込み、および実行のアクセス権を与えますが、グループおよびワールド・ユーザに対してはアクセス権を与えません。そのため、グループとワールド・ユーザは、ACL をととしてアクセス権を得なければなりません。

### 例 13-2: SUPPLIERS\_SUBSYSTEM.DIR の保護 (続き)

2. 作成者 ACE により、このディレクトリにファイルを作成するユーザが、作成したファイルへの特別なアクセス権を持たないことが保証されます。作成者 ACE の詳細については、8.8.1.2 項を参照してください。
3. デフォルト保護 ACE は、ディレクトリに作成されたファイルへの、グループおよびワールド・ユーザによるアクセスを拒否します。
4. McGrey は SUPPLIERS\_SUBSYSTEM というサブシステム識別子を有します。この ACE は、McGrey に読み込み、書き込み、および制御のアクセス権を与えます。これにより、McGrey はサブシステムのディレクトリとイメージを管理できます。
5. SUPPLIERS\_ORDERS 識別子の保有者は、実行アクセス権を持っているため、サブディレクトリに置かれているファイルにアクセスできます。
6. ACCOUNTS\_PAYABLE 識別子の保有者は、実行アクセス権を持っているため、サブディレクトリに置かれているファイルにアクセスできます。
7. 他の識別子を有するユーザには、アクセス権が与えられません。
8. McGrey は Default 属性をすべての識別子 ACE に追加し、ここでそれらを記述しています。そのため、すべての識別子 ACE がサブディレクトリの ACL に継承されます。

## 13.9.2 サブシステムのディレクトリの保護

サブシステムのユーザは、サブシステム・イメージである ORDERS.EXE と PAYMENTS.EXE にアクセスする必要があるため、EXE.DIR ディレクトリには最上位ディレクトリと同じ保護が設定されています。もう1つのディレクトリである LIB.DIR にアクセスする必要があるのは、サブシステム・イメージと McGrey だけであるため、このディレクトリにはより厳しい保護が設定されています。

### 例 13-3: SYS\$SYSDEVICE:[SUPPLIERS\_SUBSYSTEM] の保護

```
$ DIRECTORY/SECURITY SYS$SYSDEVICE:[SUPPLIERS_SUBSYSTEM...]
Directory SYS$SYSDEVICE:[SUPPLIERS_SUBSYSTEM]

EXE.DIR;1          SUPPLIERS_SUBSYSTEM    (RWE,RWE,,)    [1]
(CREATOR,ACCESS=NONE)
(DEFAULT_PROTECTION,SYSTEM:RWED,OWNER:RWED,GROUP:,WORLD:)
(IDENTIFIER=SUPPLIERS_SUBSYSTEM,ACCESS=READ+WRITE+CONTROL)
(IDENTIFIER=SUPPLIERS_ORDERS,ACCESS=EXECUTE)
(IDENTIFIER=ACCOUNTS_PAYABLE,ACCESS=EXECUTE)
(IDENTIFIER=*,ACCESS=NONE)
(IDENTIFIER=SUPPLIERS_SUBSYSTEM,OPTIONS=DEFAULT,ACCESS=READ+WRITE+CONTROL)
(IDENTIFIER=SUPPLIERS_ORDERS,OPTIONS=DEFAULT,ACCESS=EXECUTE)
(IDENTIFIER=ACCOUNTS_PAYABLE,OPTIONS=DEFAULT,ACCESS=EXECUTE)
(IDENTIFIER=*,OPTIONS=DEFAULT,ACCESS=NONE)
LIB.DIR;1          SUPPLIERS_SUBSYSTEM    (RWE,RWE,,)    [2]
(CREATOR,ACCESS=NONE)
(DEFAULT_PROTECTION,SYSTEM:RWED,OWNER:RWED,GROUP:,WORLD:)
(IDENTIFIER=SUPPLIERS_SUBSYSTEM,ACCESS=READ+WRITE+CONTROL)
(IDENTIFIER=*,ACCESS=NONE)
(IDENTIFIER=SUPPLIERS_SUBSYSTEM,OPTIONS=DEFAULT,ACCESS=READ+WRITE+CONTROL)
(IDENTIFIER=*,OPTIONS=DEFAULT,ACCESS=NONE)

Total of 2 files.
:
:
```

### 例 13-3: SYSSYSDEVICE:[SUPPLIERS\_SUBSYSTEM] の保護 (続き)

---

1. [SUPPLIERS\_SUBSYSTEM.EXE] には、13.9.1 項 に示した親ディレクトリと同じ保護コードと ACL が設定されています。サブシステムのユーザは、このディレクトリに格納されているプログラムを実行する必要があります。
  2. [SUPPLIERS\_SUBSYSTEM.LIB] には、同じ保護コードが設定されていますが、サブシステム管理者とサブシステム・イメージだけがアクセスを必要とするため、ACL はより厳しいものになっています。
- 

## 13.9.3 イメージおよびデータ・ファイルの保護

次の例に示すように、アクセスが必要な同社の担当者は、サブシステムのイメージである ORDERS.EXE と PAYMENTS.EXE にアクセスできます。ただし、データ・ファイルを更新できるのはそれらのイメージだけです。

### 例 13-4: サブシステムの ORDERS.EXE イメージおよび PAYMENTS.EXE イメージへのアクセス

---

Directory SYSSYSDEVICE:[SUPPLIERS\_SUBSYSTEM.EXE]

```
ORDERS.EXE;1  SUPPLIERS_SUBSYSTEM  (RWED,RWED,,)  [1]
  (SUBSYSTEM,IDENTIFIER=SUPPLIERS_SUBSYSTEM,  ATTRIBUTES=RESOURCE)
  (IDENTIFIER=SUPPLIERS_SUBSYSTEM,  ACCESS=READ+WRITE+CONTROL)
  (IDENTIFIER=ACCOUNTS_PAYABLE,ACCESS=EXECUTE)
  (IDENTIFIER=*,ACCESS=NONE)
PAYMENTS.EXE;1  SUPPLIERS_SUBSYSTEM  (RWED,RWED,,)  [2]
  SUBSYSTEM,IDENTIFIER=SUPPLIERS_SUBSYSTEM,  ATTRIBUTES=RESOURCE)
  (IDENTIFIER=SUPPLIERS_SUBSYSTEM,  ACCESS=READ+WRITE+CONTROL)
  (IDENTIFIER=ACCOUNTS_PAYABLE,ACCESS=EXECUTE)
  (IDENTIFIER=*,ACCESS=NONE)
```

Total of 2 files.

Directory SYSSYSDEVICE:[SUPPLIERS\_SUBSYSTEM.LIB] [3]

```
ORDERS.DAT;1  SUPPLIERS_SUBSYSTEM  (RWED,RWED,,)
  (IDENTIFIER=SUPPLIERS_SUBSYSTEM,  ACCESS=READ+WRITE)
  (IDENTIFIER=*,ACCESS=NONE)
PAYMENTS.DAT;1  SUPPLIERS_SUBSYSTEM  (RWED,RWED,,)
  (IDENTIFIER=SUPPLIERS_SUBSYSTEM,  ACCESS=READ+WRITE)
  (IDENTIFIER=*,ACCESS=NONE)
```

Total of 2 files.

Grand total of 3 directories, 6 files.

1. SUPPLIERS\_ORDERS 識別子または ACCOUNTS\_PAYABLE 識別子を有するすべてのサブシステム・ユーザが ORDERS.EXE を実行できます。
2. サブシステム・イメージ、および ACCOUNTS\_PAYABLE 識別子の所有者のみが PAYMENTS.EXE を実行できます。

### 例 13-4: サブシステムの **ORDERS.EXE** イメージおよび **PAYMENTS.EXE** イメージへのアクセス (続き)

---

3. サブシステム用のデータ・ファイルは [SUPPLIERS\_SUBSYSTEM.LIB] にあります。ここに配置されているファイルには、サブシステム・イメージと McGrey のみがアクセスできます。
- 

## 13.9.4 プリンタの保護

小切手用のプリント・キューにもディレクトリやイメージと同等の保護が必要です。小切手用プリンタへのアクセスは、サブシステムおよび ACCOUNTS\_PAYABLE 識別子の両方を有する唯一の人物である、信頼できる管理者に限定されます。例 13-5は、信頼できる管理者だけがプリント・キューへジョブを送信できるようにプリント・キューが保護されていることを示します。

### 例 13-5: キューの保護

---

```
$ SHOW SECURITY/CLASS=QUEUE TTA1
TTA1 object of class QUEUE
  Owner: [SYSTEM]
  Protection: (System: M, Owner: D, Group, World)
  Access Control List:
    (IDENTIFIER=SUPPLIERS_SUBSYSTEM+ACCOUNTS_PAYABLE, -
     ACCESS=READ+SUBMIT+MANAGE+DELETE)
    (IDENTIFIER=*, ACCESS=NONE)
```

---

## 13.9.5 サブシステム構築のためのコマンド・プロシージャ

例 13-6は、R. D. Taylor 社のサブシステムの作成に使用したコマンド・プロシージャです。

### 例 13-6: サブシステム・コマンド・プロシージャ

---

```
$ SET NOON
$ OLD_PRIV = F$SETPRV("NOALL,SYSPRV,CMKRN,OPER")
$ OLD_DEFAULT = F$ENVIRONMENT("DEFAULT")
$
$ ON CONTROL_Y THEN GOTO LEAVE
$
$ IF P1 .EQS. "REMOVE" THEN GOTO CLEANUP
$ IF P1 .EQS. "VERIFY" THEN SET VERIFY
$!
$! サブシステム識別子と、2 つの異なる作業を行う担当者のための
$! 識別子を作成する。
$!
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
ADD/IDENTIFIER SUPPLIERS_SUBSYSTEM/ATTRIBUTES=(RESOURCE,SUBSYSTEM)
ADD/IDENTIFIER SUPPLIERS_ORDERS
ADD/IDENTIFIER ACCOUNTS_PAYABLE
!
! サブシステム管理者の McGrey にサブシステム識別子を 付与する。
```



## 例 13-6: サブシステム・コマンド・プロシージャ (続き)

```
!  
GRANT/IDENTIFIER SUPPLIERS_SUBSYSTEM MCGREY/ATTRIBUTE=(RESOURCE,SUBSYSTEM)  
$!  
$! プリント・キューを設定する。  
  
$!  
$ INITIALIZE/QUEUE/START TTA1  
$ SET SECURITY/ACL=( -  
    (ID=SUPPLIERS_SUBSYSTEM+ACCOUNTS_PAYABLE,ACCESS=READ+SUBMIT+MANAGE+DELETE), -  
    (ID=*,ACCESS=NONE) )/PROTECTION=(G,W)/CLASS=QUEUE TTA1:  
  
$!  
$! サブシステムを格納するルートのディレクトリを作成する。  
$!  
$!  
$! McGrey としてログインしているものとする。  
$!  
$ SET RIGHTS_LIST/ENABLE SUPPLIERS_SUBSYSTEM/ATTRIBUTE=(RESOURCE,SUBSYSTEM)  
$ SET DEFAULT SYS$SYSDEVICE:[SUPPLIERS_SUBSYSTEM]  
$!  
$! イメージとデータ・ファイルのためのディレクトリを作成する。  
$!  
$ CREATE/DIR [SUPPLIERS_SUBSYSTEM.EXE]/PROTECTION=(G,W)  
$ CREATE/DIR [SUPPLIERS_SUBSYSTEM.LIB]/PROTECTION=(G,W)  
$ SET SECURITY/ACL=( (ID=SUPPLIERS_ORDERS,ACCESS=EXECUTE), -  
    (ID=ACCOUNTS_PAYABLE,ACCESS=EXECUTE), -  
    (ID=SUPPLIERS_ORDERS,OPTIONS=DEFAULT,ACCESS=EXECUTE), -  
    (ID=ACCOUNTS_PAYABLE,OPTIONS=DEFAULT,ACCESS=EXECUTE) )/DELETE -  
    [SUPPLIERS_SUBSYSTEM]LIB.DIR  
  
$!  
$! サブシステム・イメージの作成をエミュレートする。  
$!  
$ SET DEFAULT [.EXE]  
$ CREATE ORDERS.MAR  
    .ENTRY START,0  
    $setpri_s pri=#0  
10$: BRB 10$  
    ret  
    .END START  
  
$ MACRO ORDERS  
$ LINK ORDERS  
$ SET SECURITY/PROTECTION=(W:RWED) ORDERS.MAR;*,.OBJ;*  
$ DELETE ORDERS.MAR;*,.OBJ;*  
$ COPY ORDERS.EXE PAYMENTS.EXE  
$!  
$! イメージに適切な保護を適用する。  
$!  
$ SET SECURITY/ACL=(ID=SUPPLIERS_ORDERS,ACCESS=EXECUTE)/DELETE PAYMENTS.EXE  
$ SET SECURITY/ACL=(SUBSYSTEM,ID=SUPPLIERS_SUBSYSTEM,ATTRIBUTES=RESOURCE) ORDERS.EXE  
$ SET SECURITY/ACL=(SUBSYSTEM,ID=SUPPLIERS_SUBSYSTEM,ATTRIBUTES=RESOURCE) PAYMENTS.EXE  
$!  
$! アプリケーションによって使用されるデータ・ファイルを作成して保護する。  
$!  
$ SET DEFAULT [-.LIB]  
$ CREATE ORDERS.DAT  
$ CREATE PAYMENTS.DAT  
$ SET SECURITY/ACL=( (ID=SUPPLIERS_SUBSYSTEM,ACCESS=READ+WRITE), -  
    (ID=*,ACCESS=NONE) ) ORDERS.DAT  
$ SET SECURITY/LIKE=(NAME=ORDERS.DAT) PAYMENTS.DAT  
$!  
$! ディレクトリ構造とキューの保護を表示する。  
$!  
$ SET DEFAULT 'OLD_DEFAULT'  
$ DEFINE SYS$OUTPUT SUBSYS.LIS  
$ DIRECTORY/SECURITY SYS$SYSDEVICE:[000000]SUPPLIERS_SUBSYSTEM.DIR  
$ DIRECTORY/SECURITY SYS$SYSDEVICE:[SUPPLIERS_SUBSYSTEM...]  
  
$ SHOW SECURITY/CLASS=QUEUE TTA1
```

### 例 13-6: サブシステム・コマンド・プロシージャ (続き)

---

```
$ DEASSIGN SYSS$OUTPUT
$
$ LEAVE:
$ IF P1 .EQS. "VERIFY" THEN SET NOVERIFY
$ SET DEFAULT 'OLD_DEFAULT'
$ SET PROC/PRIV=('OLD_PRIV')
$ EXIT
$
$ CLEANUP:
$ SET PROC/PRIV=BYPASS
$ SET DEFAULT SYSS$SYSDEVICE:[000000]
$ DELETE [SUPPLIERS_SUBSYSTEM...]*.*
$ DELETE [SUPPLIERS_SUBSYSTEM]EXE.DIR;
$ DELETE [SUPPLIERS_SUBSYSTEM]LIB.DIR;
$ DELETE SUPPLIERS_SUBSYSTEM.DIR;
$ STOP/QUE/NEXT TTA1
$ DELETE/QUEUE TTA1
$ GOTO LEAVE
```

---

## 特権の割り当て

特権は、特定のシステム機能の使用範囲を、登録ユーザのために作成されたプロセスに限定する働きがあります。この制限により、オペレーティング・システムのコード、データ、資源の一貫性が守られ、その結果としてユーザ・サービスの一貫性が守られます。特権を個々のユーザに与える場合は、必ず次に示す 2 つの要素を慎重に検討してからにします。

- システムを中断させることなく特権を使用するだけの技能と経験をそのユーザが持っているかどうか
- 特権を必要とする妥当性がそのユーザにあるかどうか

特権は、特権を有するユーザがシステムに与え得る損害に応じて、次に示す 7 つのカテゴリに分類されます。

- None: 特権なし
- Normal: システムを有効に使用するための最低限の特権
- Group: 同一グループのメンバに影響が及ぶ可能
- Devour: システム全体のクリティカルではない資源を消費する可能性
- System: 通常のシステム操作に影響が及ぶ可能性
- Objects: 保護オブジェクト（ファイル、デバイス、論理名テーブル、グローバル・セクションなど）のセキュリティを損なう可能性
- All: システムを制御する可能性

ユーザの特権は、そのユーザの UAF レコードに、64 ビットの特権マスクの形で記録されます。ユーザがシステムにログインすると、ユーザの特権が、ユーザのプロセスのヘッダに格納されます。このようにして、そのユーザの特権は、ユーザに対して作成されるプロセスに渡されます。ユーザは、DCL の SET PROCESS/PRIVILEGES コマンドを使用して、ユーザに許可される特権の有効/無効の切り換え、およびユーザが実行するイメージで利用できる特権の詳細の制御が可能です。さらには、SETPRV 特権を持つユーザは任意の特権を有効にすることができます。

表 8-2 表 8-2 に、カテゴリごとに特権を分け、それぞれの簡単な概要を説明します。以降の節では、OpenVMS システムで利用できるすべての特権について詳しく説明します。各節のタイトルは対象となる特権のカテゴリ (Normal, Devour など)

を示します。この付録では、それぞれの特権について、特権によって与えられるケーパビリティと特権を与えるべきユーザについて説明します。

## A.1 ACNT 特権 (Devour)

ACNT 特権を有するプロセスは、RUN (Process) コマンドを使用できます。また、会計情報管理が無効になっているプロセスの作成にプロセス作成 (\$CREPRC) システム・サービスを使用できます。会計情報管理が無効になっているプロセスとは、資源の利用状況が現在の会計情報ログ・ファイルに記録されないプロセスのことです。

## A.2 ALLSPOOL 特権 (Devour)

ALLSPOOL 特権を有するユーザ・プロセスは、デバイス割り当て (\$ALLOC) システム・サービスを実行するか、DCL の ALLOCATE コマンドを使用して、スプールされたデバイスを割り当てることができます。

\$ALLOC システム・サービスは、デバイスの独占的利用のために、デバイスの割り当て、もしくは予約をします。マウントされた、共用可能なデバイスを割り当ててはできません。

この特権は、スプールされたデバイスに対して、論理入出力操作または物理入出力操作を実行する必要のあるユーザにのみ付与します。通常、スプールされたデバイスを割り当てる特権は、シンビオントにのみ付与されます。

## A.3 ALTPRI 特権 (System)

ALTPRI 特権を有するユーザ・プロセスは、次のことが可能です。

- 自身の基本優先順位の引き上げ
- ターゲット・プロセスの基本優先順位の設定
- バッチ・ジョブまたはプリント・ジョブの優先順位の変更

基本優先順位を上げるには、優先順位設定 (\$SETPRI) システム・サービスまたは DCL の SET PROCESS/PRIORITY コマンドを実行します。一般に、このシステム・サービスにより、プロセスは、自身の基本優先順位を設定したり、別のプロセスの基本優先順位を設定できるようになります。ただし、あるプロセスが別のプロセスの優先順位を設定できるのは、次のいずれかの条件に該当する場合だけです。

- \$SETPRI システム・サービスを呼び出すプロセスの UIC が、ターゲット・プロセスの UIC と同じであること。
- 呼び出し元のプロセスが、ターゲット・プロセスに対する制御特権 (GROUP または WORLD) を有する。

ALTPRI 特権を有するプロセスは、自身よりも優先度の高い独立プロセスを作成できます。このようなプロセスを作成するには、プロセス作成 (\$CREPRC) システム・サービスまたは DCL の RUN/PRIORITY コマンドに省略可能な引数を指定します。

また、ALTPRI 特権を有するユーザは、ジョブのスケジューリング (\$SNDJBC) の優先順位を、MAXQUEPRI システム・パラメータで設定されたものより大きい値に調整できます。

この特権を広く付与しないでください。条件を満たさないユーザが基本優先順位を無制限に設定できるようになると、プロセスの公平かつ秩序立ったスケジューリングが容易に混乱させられる可能性があります。

## A.4 AUDIT 特権 (System)

AUDIT 特権を有するソフトウェアは、\$AUDIT\_EVENT、\$CHECK\_PRIVILEGE、\$CHKPRO または \$CHECK\_ACCESS の 4 つのシステム・サービスのいずれかを使用して、システム・セキュリティ・監査ログ・ファイルに監査レコードを追加できます。さらに、\$AUDIT\_EVENT システム・サービスにより、監査メッセージのすべての構成要素を指定することができます。そのため、AUDIT 特権により、オペレーティング・システムまたはユーザ・プロセスに起因すると考えられるイベントをログに記録することが可能となります。

この特権は、監査メッセージをシステム監査ログ・ファイルに追加する必要のある、信頼できるイメージに対してのみ付与します。この特権を有するユーザが、NSA\$M\_INTERNAL フラグを設定した状態で無効なイベントを記録しようとして、システム障害を引き起こす可能性があります。

## A.5 BUGCHK 特権 (Devour)

BUGCHK 特権を有するプロセスは、ユーザ・モード、スーパバイザ・モード、互換性モードからのバグチェック・エラー・ログ・エントリの作成 (EXE\$BUG\_CHECK)、またはシステム・エラー・ロガーへのメッセージの送信 (\$SNDERR) が可能です。この特権は、Bugcheck 機能を使用する、HP 製のシステム・ソフトウェアにのみ付与してください。

## A.6 BYPASS 特権 (All)

BYPASS 特権を有するユーザ・プロセスは、UIC ベースの保護、アクセス制御リスト (ACL) による保護、および強制アクセス制御の適用を回避して、すべての保護オブジェクトにフルにアクセスできます。BYPASS 特権により、システムへの無制限のアクセスが可能になります。実行可能な操作には次のものがあります。

- すべてのユーザ登録レコード (SYSUAF.DAT) の変更
- すべてのライト識別子と保持者レコード (RIGHTSLIST.DAT) の変更

- すべてのネットワーク代理レコード (NETPROXY.DAT または NET\$PROXY.DAT [VAX のみ]) の変更
- すべての DECnet オブジェクトのパスワードとアカウント (NETOBJECT.DAT) の変更
- すべてのボリュームのすべてのファイルへの無制限アクセス

この特権は、あらゆるオブジェクト保護を無効にするため、付与するときは特に注意してください。十分テスト済みで信頼性の高いプログラムやコマンド・プロシージャの場合にのみ使用することをお勧めします。SYSPRV 特権は、アクセスのチェックを行いながらすべてのオブジェクトへのアクセスを最終的には付与するため、会話型のアクセスに適しています。READALL 特権は、バックアップ操作に適しています。

BYPASS 特権を有するプロセスは、次の作業を実行することができます。

作業	インタフェース
ファイル・システム操作の実行	
ファイル所有権の変更	SET SECURITY/OWNER, F11BXQP への \$QIO 要求
削除のマークが設定されたファイルへのアクセス	F11A ACP または F11BXQP に対する \$QIO 要求
非アクセス・ロックされているファイルへのアクセス	F11A ACP または F11BXQP に対する \$QIO 要求
新規作成ファイルに対する所有者 ACE の作成の無効化	F11BXQP への \$QIO 要求
ディレクトリのファイル・ヘッダに含まれているディレクトリ・ビットのクリア	F11BXQP への \$QIO 要求
拡張ヘッダの操作	F11BXQP への \$QIO 要求
ボリューム・ロックの獲得または解放	F11BXQP への \$QIO 要求
ボリュームに対する強制マウント検証	F11BXQP への \$QIO 要求
アクセス・ロック・ビットが設定されていないファイル・アクセス・ウィンドウの作成	F11BXQP への \$QIO 要求
ボリューム・ロックに対するヌル・ロック・モードの指定	F11BXQP への \$QIO 要求
ロックされているファイルへのアクセス	F11BXQP への \$QIO 要求
ボリューム上のディスク使用量制限の有効化または無効化	F11BXQP への \$QIO 要求
ネットワーク・データベースの操作	

作業	インタフェース
パーマネント・ネットワーク・データベース・レコードの表示	NCP
パーマネント DECnet オブジェクト・パスワードの表示	NCP
運用時 DECnet オブジェクト・パスワードの表示	NCP
任意アクセス制御または強制アクセス制御の調整	
ユーザ登録レコードの読み込み	\$GETUAI
ユーザ登録レコードの変更	\$SETUAI
メールボックスの保護設定の変更	メールボックス・ドライバ (MBDRIVER) に対する \$QIO 要求
共用メモリ・メールボックスの保護設定の変更	メールボックス・ドライバ (MBXDRIVER) に対する \$QIO 要求
任意オブジェクト保護または強制オブジェクト保護の適用回避	\$CHKPRO
その他	
磁気テープの初期化	\$INIT_VOL
InfoServer システムのアンロード	InfoServer システム (DADDRIVER) への \$QIO 要求

## A.7 CMEXEC 特権 (All)

CMEXEC 特権を有するユーザ・プロセスは、エグゼクティブ・モードへのモード変更 (\$CMEXEC) のシステム・サービスを実行できます。

プロセスは、このシステム・サービスを使用して、自身のアクセス・モードをエグゼクティブ・モードに変更し、指定したルーチンを実行して、システム・サービス呼び出しの前のアクセス・モードに戻れます。エグゼクティブ・モードでは、プロセスはカーネル・モードへのモード変更 (\$CMKRNL) のシステム・サービスを実行する許可を得ます。

この特権は、オペレーティング・システムのデータ構造や内部機能のうち、保護が設定され、機密性の高いものへのアクセスが必要なユーザに対してのみ付与します。条件に満たないユーザが機密性の高いデータ構造や機能に無制限にアクセスできるようになると、オペレーティング・システムや他のユーザへのサービスが容易に混乱させられる可能性があります。たとえば、システム障害、システムおよびユーザのすべてのデータの破壊、機密情報の漏洩などの混乱が考えられます。

## A.8 CMKRNL 特権 (AII)

CMKRNL 特権を有するユーザ・プロセスは、カーネル・モードへのモード変更 (\$CMKRNL) のシステム・サービスを実行できます。

プロセスは、このシステム・サービスを使用して、自身のアクセス・モードをカーネル・モードに変更し、指定したルーチンを実行して、システム・サービス呼び出しの前のアクセス・モードに戻れます。カーネル・モードでは、プロセスは任意のシステム特権を有効にすることができます。

CMKRNL および SYSNAM の両方の特権を有するプロセスは、システム時間を設定できます。

この特権は、特権命令を実行する必要があるユーザ、またはオペレーティング・システムのデータ構造や機能のうち、保護が設定され、機密性の高いものへのアクセスが必要なユーザに対してのみ付与します。条件に満たないユーザが特権命令を無制限に使用したり、機密性の高いデータ構造や機能に無制限にアクセスできるようになると、オペレーティング・システムや他のユーザへのサービスが容易に混乱させられる可能性があります。たとえば、システム障害、システムおよびユーザのすべてのデータの破壊、機密情報の漏洩などの混乱が考えられます。

CMKRNL 特権を有するプロセスは、次の作業を実行することができます。

作業	インタフェース
マルチプロセッサ操作の変更	START/CPU , STOP/CPU
システム全体の RMS デフォルト値の変更	SET RMS/SYSTEM
カーネル・モードのプロセスの一時停止	SET PROCESS/SUSPEND=KERNEL
別のプロセスのライト・リストまたはそのプロセスの非動的識別子属性の変更	SET RIGHTS_LIST
属性を変更した識別子の付与	SET RIGHTS/ATTRIBUTE
システム・ライト・リストの変更	SET RIGHTS_LIST/SYSTEM
プロセス UIC の変更	SET UIC
インターロックされたキューの再試行回数の変更	Ethernet 802 ドライバ (DEBNA/NI) への \$QIO 要求
デバイス割り込みベクタへの接続	割り込みベクタ (CONINTERR) への \$QIO 要求
Genbyte モードでの回線の開始または変更	同期通信回線 (XGDRIVER) への \$QIO 要求
ポート・コマンド・レジスタにおけるスピン・ウェイト時間の設定	Ethernet 802 ドライバ (DEBNA) への \$QIO 要求
既知のイメージ・リストの変更	INSTALL



作業	インタフェース
次のアイテム・コードの処理 SJC\$_ACCOUNT_NAME アイテム SJC\$_UIC SJC\$_USERNAME	ジョブ・コントローラ・システム・サービス (\$SNDJBC) への送信
ディスク使用量制限のない独立プロセスの作成	RUN/DETACHED, \$CREPRC
実行中のシステムの内部の調査	ANALYZE/SYSTEM

## A.9 DIAGNOSE 特権 (Objects)

DIAGNOSE 特権を有するプロセスは、オンライン診断プログラムを実行したり、エラー・ログ・ファイルに書き込まれるすべてのメッセージの取得とコピーを行うことができます。

DIAGNOSE 特権を有するプロセスは、次の作業も実行することができます。

作業	インタフェース
対応する診断バッファへの \$QIO 要求を送信	\$QIO
インターロックされたキューの再試行回数の変更	Ethernet 802 ドライバ (DEBNA/NI) への \$QIO 要求
ポート・コマンド・レジスタにおけるスピン・ウェイト時間の設定	Ethernet 802 ドライバ (DEBNA) への \$QIO 要求
Diagnostic and Utilities Protocol (DUP) クラス・ドライバへのアクセス	SET HOST/HSC が使用する DUP クラス・ドライバ (FYDRIVER) への \$QIO 要求
SCSI 汎用クラス・ドライバの特別なパススルー機能の実行	SCSI ドライバ (GKDRIVER) に対する \$QIO 要求
診断バッファの処理	TU58 磁気テープ (TUDRIVER) への \$QIO 要求

## A.10 DOWNGRADE 特権 (All)

DOWNGRADE 特権を有するプロセスは、強制アクセス制御を操作できます。この特権により、プロセスは、Bell and LaPadula 制限 (\*) プロパティに違反して、秘密性の低いオブジェクトへの書き込みを行うことができます。<sup>1</sup> この特権は、Security Enhancement Service ソフトウェア (SEVMS) など、高度なセキュリティ製品用に予約済みです。

<sup>1</sup> 書き込みに対する制限の名称。マルチレベル・セキュリティでは、信用されないソフトウェアによる書き込みを一切禁止することが要件となっています。

## A.11 EXQUOTA 特権 (Devour)

EXQUOTA 特権により、指定のディスク・ボリュームにあるユーザのファイルが占有している領域が、そのボリュームで当該ユーザに対して設定されている使用制限 (UIC によって決まる値) を超えることができます。

## A.12 GROUP 特権 (Group)

GROUP 特権を有するユーザ・プロセスは、次に示すプロセス制御システム・サービスを実行することにより、同じグループ内の他のプロセスに働きかけることができます。

- プロセス一時停止 (\$SUSPND)
- プロセス再開 (\$RESUME)
- プロセス削除 (\$DELPRI)
- 優先順位設定 (\$SETPRI)
- ウェイクアップ (\$WAKE)
- スケジューリングされたウェイクアップ (\$SCHDWK)
- ウェイクアップ取り消し (\$CANWAK)
- 強制終了 (\$FORCEX)

GROUP 特権を有するユーザ・プロセスは、同じグループ内の別のプロセスを制御できます。ユーザのプロセスは、ジョブ取得/プロセス情報 (\$GETJPI) システム・サービスを実行することで、同じグループ内の他のプロセスを調査することができます。GROUP 特権を有するプロセスは、グループ内の他のプロセスに対して SET PROCESS コマンドを実行できます。

GROUP 特権は、プロセス自身が作成したサブプロセスや、そのプロセスの UIC を有する他の独立プロセスに対する制御や調査を行う場合には必要ありません。ただし、この特権は、UIC グループの他のメンバのプロセスや操作を制御する必要があるユーザには付与する必要があります。

## A.13 GRPNAM 特権 (Devour)

GRPNAM 特権を有するユーザ・プロセスは、システム論理名テーブルに対する任意アクセス制御の適用を回避できます。そして、論理名作成 (\$CRELNM) システム・サービスおよび論理名削除 (\$DELLNM) システム・サービスを使用して、プロセスが属しているグループの論理名テーブルに名前を挿入 (または論理名テーブルから名前を削除) することができます。

加えて、特権プロセスは、DCL の ASSIGN コマンドおよび DEFINE コマンドを実行してグループの論理名テーブルに名前を追加したり、DCL の DEASSIGN コマンドを実行してテーブルから名前を削除したりできます。この特権では、グループ・メンバ間でボリュームを共用するときに、DCL の MOUNT コマンドおよび

DISMOUNT コマンドで /GROUP 修飾子を使用することができます (システム・サービスの \$MOUNT および \$DISMOUNT でも同様)。

この特権は、システムのすべてのユーザには付与しないようにします。この特権により、ユーザ・プロセスがグループ論理名を無制限に作成できるようになるためです。条件を満たさないユーザがグループ論理名を無制限に作成できると、システムの動的メモリが過剰に使用され、システムの性能が低下する可能性があります。また、GRPNAM 特権を有するプロセスが、SYS\$SYSTEM など、よく使用される論理名の定義を作成することで、同一グループ内の他のプロセスの動作に影響を与える可能性があります。

## A.14 GRPPRV 特権 (Group)

プロセスのグループがオブジェクト所有者のグループと一致する場合、GRPPRV 特権を有するプロセスは、オブジェクトのシステム保護フィールドが提供するアクセス権を獲得します。また、GRPPRV 特権を有するプロセスは、DCL の SET SECURITY コマンドを使用して、そのプロセスのグループと同じ所有者グループの任意のオブジェクトの保護と所有権を変更できます。

この特権は、グループ管理者の役割を果たすユーザに対してのみ付与します。この特権を、オブジェクトの保護と所有権を変更する必要のない、条件を満たさないユーザに付与すると、そのユーザが、グループの UAF レコードの値を、グループ管理者と同じ値に書き換えてしまう可能性があります。また、資源の割り当てを増やしたり、権限を与えられている特権を付与したりすることも可能となってしまいます。

GRPPRV 特権を有するプロセスは、次の作業を実行することができます。

作業	インタフェース
オブジェクト所有権の変更	SET SECURITY/OWNER, F11BXQP への \$QIO 要求
ユーザ登録レコードの読み込みまたは変更	\$GETUAI, \$SETUAI

作業	インタフェース
ファイル・システム操作 <ul style="list-style-type: none"> <li>新規作成されたファイルに対する所有者 ACE の作成の無効化</li> <li>ディレクトリのファイル・ヘッダに含まれる ディレクトリ・ビットのクリア</li> <li>ボリューム・ロックの獲得または解放</li> <li>ボリュームに対する強制マウント検証</li> <li>アクセス・ロック・ビットが設定されていないファイル・アクセス・ウィンドウの作成</li> <li>ボリューム・ロックに対するヌル・ロック・モードの指定</li> <li>ロックされているファイルへのアクセス</li> <li>ボリューム上のディスク使用量制限の有効化または無効化</li> </ul>	F11BXQP への \$QIO 要求

## A.15 IMPERSONATE 特権 (AII) (旧名称 DETACH)

プロセスは、MAXJOBS および MAXDETACH で設定されている制限を上回っていません。IMPERSONATE 特権がなくてもそのプロセスの UIC を持った独立プロセスを作成できます。ただし、IMPERSONATE 特権は、プロセスが独立プロセスに対して異なる UIC を指定したい場合に有用です。IMPERSONATE 特権を持っていれば、独立プロセスに指定できる UIC に制限はありません。このため、独立プロセスがアクセス可能なファイル、ディレクトリなどのオブジェクトについても制限がありません。また、IMPERSONATE 特権により、プロセスは、制限値が設定されない独立プロセスを作成できます。プロセスは、プロセス作成 (\$CREPRC) システム・サービスを実行することで、独立プロセスを作成できます。

さらに、IMPERSONATE 特権により、DCL の RUN/DETACH コマンドを使用して、信頼できるサーバ・プロセスを作成できます。信頼できるプロセスは、通常のシステム・セキュリティ監査ポリシーの適用対象から除外されます。

独立プロセスは、そのプロセスを作成したユーザがシステムからログアウトした後もそのまま残ります。

IMPERSONATE 特権は、以前は DETACH 特権と呼ばれていました。後方互換性を確保するため、コマンド行で DETACH を指定した場合でも、コマンドはそのまま正常に動作します。

## A.16 IMPORT 特権 (Objects)

IMPORT 特権を有するプロセスは、強制アクセス制御を操作できます。IMPORT 特権により、プロセスは、ラベルのないテープ・ボリュームをマウントできます。この特権は、SEVMS など、高度なセキュリティ製品用に予約済みです。

## A.17 LOG\_IO 特権 (All)

LOG\_IO 特権を有するユーザ・プロセスは、入出力要求キュー登録 (\$QIO) システム・サービスを実行して、論理レベルの入出力操作を実行できます。LOG\_IO 特権は、パーマネント・ターミナル特性の設定など、特定のデバイス制御機能にも必要です。LOG\_IO および SYSNAMA も有する、典型的な NETMBX 特権と TMPMBX 特権を有するプロセスは、Phase IV ネットワーク設定プロシージャである NICONFIG.COM を使用して、Ethernet を再設定できます。

通常、プロセスの入出力要求は、OpenVMS Record Management Services (RMS) などの入出力パッケージを使用することで、間接的に処理されます。しかし、入出力操作をより詳細に制御したり、入出力操作の効率を向上させたりするために、上級ユーザの中には、プロセスとシステム入出力ドライバ・プログラムとの間のインタフェースを直接扱うことを好むものもいます。それには、\$QIO を実行します。多くの場合、対象となる操作は、論理レベルの入出力操作です。論理レベルの機能は LOG\_IO 特権なしでも、/FOREIGN 修飾子を使用してマウントされているデバイス、および非ファイル構造デバイスに対して実行できます。

この特権では、ファイル構造化のメリットに関係なく、選択したボリュームの任意の場所にあるデータにプロセスがアクセスできるため、必要なユーザのみに付与してください。この特権を必要としない条件に満たないユーザに付与すると、オペレーティング・システムおよび他のプロセスのためのサービスが簡単に混乱させられる可能性があります。たとえば、システム・デバイス上の情報の破壊、ユーザ・データの破壊、機密情報の漏洩などの混乱が考えられます。

LOG\_IO 特権を有するプロセスは、次の作業も実行することができます。

作業	インタフェース
非ファイル構造のプライベート・デバイスに対する物理入出力呼び出しの発行	\$QIO
次のターミナル属性の変更 HANGUP SET_SPEED SECURE_SERVER	SET TERMINAL (または TTDRIVER) /[NO]HANGUP /[NO]SET_SPEED /[NO]SECURE_SERVER

## A.18 MOUNT 特権 (Normal)

MOUNT 特権を有するユーザ・プロセスは、マウント・ボリューム QIO 機能を実行できます。この機能は、HP が提供するシステム・ソフトウェアに限定して使用してください。

## A.19 NETMBX 特権 (Normal)

NETMBX 特権を有するプロセスは、DECnet コンピュータ・ネットワークに関連する機能を実行できます。たとえば、プロセスがターミナル回線を非同期 DECnet プロトコルに切り換えたり、チャンネルをネットワーク・デバイスに割り当てたりできます。この特権は、ネットワークへのアクセスが必要な一般ユーザに付与します。

## A.20 OPER 特権 (System)

OPER 特権を有するプロセスは、Operator Communication Manager (OPCOM) プロセスを使用して、ユーザの要求への応答、ログインしているすべてのターミナルへのメッセージのブロードキャストを行えるほか、ターミナルをオペレータのターミナルとして指定し、これらターミナルに表示するメッセージの種類を指定したり、オペレータのメッセージのログ・ファイルの初期化や管理を行ったりできます。さらに、この特権により、ユーザは、デバイスのスプール、あらゆるキューの作成と管理、非ファイル構造のデバイスすべての保護と所有権の変更を行うことができます。

この特権は、システムのオペレータにのみ付与します。オペレータとは、一般ユーザの要求に応えるユーザで、システムの周辺機器の管理（テープ・リールのマウント、プリンタ用紙の交換など）、およびシステム運用に関わるその他の日常的な業務を行います。非特権ユーザは、コンソール・ターミナルにログインすることで、オペレータに対する要求（テープのマウントなど）に応えることができます。

OPER 特権を有するプロセスは、次の作業を実行することができます。

作業	インタフェース
デバイスの保護設定の変更	SET PROTECTION/DEVICE
デバイスの所有権の変更	SET PROTECTION/DEVICE/OWNER

作業	インタフェース
システム管理ユーティリティへのアクセス	SYSMAN
オペレータ作業の実行	
ブロードキャスト応答の発行	REPLY, \$SNDOPR
システム・オペレータ要求の取り消し	REPLY/ABORT, \$SNDOPR
システム・オペレータ・ログ・ファイルの初期化	\$SNDOPR
未処理のシステム・オペレータ要求に対する応答	REPLY/TO, REPLY/PENDING, REPLY/INITIALIZE_TAPE, \$SNDOPR
システム・オペレータ要求の発行	REQUEST, \$SNDOPR
システム・オペレータ・クラスの有効化	REPLY/ENABLE, \$SNDOPR, \$SNDMSG
システム・オペレータ・クラスの無効化	REPLY/DISABLE, \$SNDOPR
ブロードキャスト・メッセージの送信	\$BRKTHRU, \$BRDCST
オペレータ・ログへのイベントの書き込み	\$SNDOPR
システム・オペレータ・ログの初期化	REPLY/LOG, \$SNDOPR
現在のオペレータ・ログのクローズ	REPLY/NOLOG, \$SNDOPR
オペレータへのメッセージの送信	REPLY, \$SNDOPR
自動起動の有効化または無効化	\$SNDJBC (SJC\$_DISABLE_AUTO_START, SJC\$_ENABLE_AUTO_START)
すべてのキューの停止	\$SNDJBC (SJC\$_STOP_ALL_QUEUES_ON_NODE)
デバイスの特性の変更	
デバイスの使用の可否の変更	SET DEVICE/[NO]AVAILABLE
デバイス・デュアル・ポート接続の変更	SET DEVICE/[NO]DUAL_PORT
デバイス・エラー・ログの変更	SET DEVICE/[NO]ERROR_LOGGING
デバイス・スプーリングの変更	SET DEVICE/[NO]SPOOLED
デフォルトの曜日定義の変更	
デフォルトの曜日タイプを PRIMARY に設定	SET DAY/PRIMARY
デフォルトの曜日タイプを SECONDARY に設定	SET DAY/SECONDARY
曜日タイプを DEFAULT に戻す	SET DAY/DEFAULT
ログイン制限の変更または無効化	
会話型ログイン制限の変更	SET LOGIN/INTERACTIVE
ネットワーク・ログイン制限の変更	SET LOGIN/NETWORK

作業	インタフェース
バッチ・ログイン制限の変更	SET LOGIN/BATCH
キューの作成および変更	
キューへの任意アクセス制御の適用回避	
キューの作成	\$SNDJBC (SJC\$_CREATE_QUEUE)
キュー特性の定義	\$SNDJBC (SJC\$_DEFINE_CHARACTERISTICS)
フォームの定義	\$SNDJBC (SJC\$_DEFINE_FORM)
特性の削除	\$SNDJBC (SJC\$_DELETE_CHARACTERISTICS)
フォームの削除	\$SNDJBC (SJC\$_DELETE_FORM)
バッチ・プロセスの基本優先順位の設定	\$SNDJBC (SJC\$_BASE_PRIORITY)
ジョブ・スケジュールの優先順位の設定	\$SNDJBC (SJC\$_PRIORITY)
会計情報管理の開始	SET ACCOUNTING/ENABLE , \$SNDJBC (SJC\$_START_ACCOUNTING)
会計情報管理の停止	SET ACCOUNTING/DISABLE , \$SNDJBC (SJC\$_STOP_ACCOUNTING)
<i>LAT</i> デバイスの操作	
LAT 要求情報メッセージの送信	LAT ポート・ドライバ (LTDRIVER) への \$QIO 要求
LAT サービスの静的レートの設定	LAT ポート・ドライバ (LTDRIVER) への \$QIO 要求
最新の LAT 応答メッセージ・バッファの読み取り	LAT ポート・ドライバ (LTDRIVER) への \$QIO 要求
ポート・タイプの「専用」から「アプリケーション」への変更	LAT ポート・ドライバ (LTDRIVER) への \$QIO 要求
ポート・タイプの「アプリケーション」から「専用」への変更	LAT ポート・ドライバ (LTDRIVER) への \$QIO 要求
テープ操作の変更	
ファイル・ウィンドウ・マッピング・ポイントの数の指定	MOUNT/WINDOWS , \$MOUNT
代替 ACP が設定されたボリュームのマウント	MOUNT/PROCESSOR , \$MOUNT
代替キャッシュ制限が設定されたボリュームのマウント	MOUNT/CACHE , \$MOUNT



作業	インタフェース
テープ・コントローラの書き込みキャッシュの変更	MOUNT/CACHE , \$MOUNT
ODS1 ディレクトリの FCB キャッシュ制限の変更	SET VOLUME/ACCESSED , MOUNT/ACCESSED , \$MOUNT
ネットワーク操作の実行	
エグゼキュータ状態が制限されている間のオブジェクトへの接続	
ネットワーク・イベント・ログ・バッファの読み込み	NETACP
ネットワーク運用時データベースの変更	NETACP
パーマネント・データベースへの更新アクセス	DECnet/NML
DECnet サークットへの接続	DECnet ダウンライン・ローディングおよびループバック・クラス・ドライバ (NDDRIVER) への \$QIO 要求
パーマネント DECnet サービス・パスワードの表示	NCP
運用時 DECnet サービス・パスワードの表示	NCP
ターミナルによる特性変換の制御	
ターミナル・フォールバック・テーブルのロード	TFU , ターミナル・フォールバック・ドライバ (FBDRIVER) への \$QIO 要求
ターミナル・フォールバック・テーブルのアンロード	TFU , ターミナル・フォールバック・ドライバ (FBDRIVER) への \$QIO 要求
システムのデフォルト・ターミナル・フォールバック・テーブルの作成	TFU , ターミナル・フォールバック・ドライバ (FBDRIVER) への \$QIO 要求
クラスタ操作の制御	
想定ボート変更の要求	SET CLUSTER/EXPECTED_VOTES
デバイスの MSCP サービスの要求	SET DEVICE/SERVED
クォーラム変更の要求	SET CLUSTER/QUORUM
フェールオーバー・リストへのアダプタの追加	DEBNI BI バス NI ドライバ (EFDRIVER) への \$QIO 要求
フェールオーバー・リストからのアダプタの削除	DEBNI BI バス NI ドライバ (EFDRIVER) への \$QIO 要求

作業	インタフェース
アダプタを現在のアダプタに設定	DEBNI BI バス NI ドライバ (EFDRIIVER) への \$QIO 要求
新規アダプタのテスト間隔の設定	DEBNI BI バス NI ドライバ (EFDRIIVER) への \$QIO 要求

他の特権と組み合わせることで、OPER 特権を有するプロセスは、次の作業を実行できます。

特権	作業	インタフェース
OPER および CMKRNL	プライベート ACP が設定されたボリュームのマウント	MOUNT/PROCESSOR , \$MOUNT
OPER および LOG_IO	システム時間の設定	SET TIME , \$SETIME
OPER および SYSNAM	キュー・マネージャの起動または終了	START/QUEUE/MANAGER , STOP/QUEUE/MANAGER , \$SNDJBC
OPER および VOLPRO	ブランク・テープの初期化、またはブランク・テープ初期化中のアクセス・チェックの無効化	\$INIT_VOL , MOUNT , \$MOUNT

## A.21 PFNMAP 特権 (All)

PFNMAP 特権を有するユーザ・プロセスは、ページ・フレーム番号 (PFN) のグローバル・セクションを作成し、ページまたはレジスタの利用者が誰であるかに関係なく、作成した PFN グローバル・セクションを物理メモリのページまたは入出力デバイス・レジスタにマッピングすることができます。このような特権プロセスは、システム・サービスの \$DGBLSC を使用して、PFN ベースのグローバル・セクションを削除することもできます。

この特権を付与するときは注意が必要です。条件に満たないユーザが物理メモリに自由にアクセスできるようになると、オペレーティング・システムや他のプロセスへのサービスが容易に混乱させられる可能性があります。たとえば、システム障害、システムおよびユーザのすべてのデータの破壊、機密情報の漏洩などの混乱が考えられます。

## A.22 PHY\_IO 特権 (All)

PHY\_IO 特権を有するユーザ・プロセスは、入出力要求キュー登録 (\$QIO) システム・サービスを実行して、物理レベルの入出力操作を実行できます。

通常、プロセスの入出力要求は、OpenVMS Record Management Services (RMS) などの入出力パッケージを使用することで、間接的に処理されます。しかし、入

出力操作をより詳細に制御したり，アプリケーションの効率を向上させたりするために，上級ユーザの中には，プロセスとシステム入出力ドライバ・プログラムとの間のインタフェースを直接扱うことを好むものもいます。それには，\$QIO システム・サービスを実行します。多くの場合，対象となる操作は，物理レベルの入出力操作です。

PHY\_IO 特権は，必要なユーザにのみ付与します。LOG\_IO 特権の場合よりもさらに慎重にします。この特権を必要としない条件に満たないユーザに付与すると，オペレーティング・システムおよび他のユーザのためのサービスが簡単に混乱させられる可能性があります。たとえば，システム・デバイス上の情報の破壊，ユーザ・データの破壊，機密情報の漏洩などの混乱が考えられます。

PHY\_IO 特権を有するプロセスは，次の作業も実行することができます。

作業	インタフェース
特定のシャドウ・セット・メンバ・ユニットへのアクセス	\$ASSIGN, \$QIO
ウォッチポイントの作成または削除	SMP ウォッチポイント・ドライバ (WPDRIVER) に対する \$QIO 要求
LTA デバイスのサーバ/ポート (IO\$TTY_PORT!IO\$M_LT_MAPPORT) へのマッピング	LAT ポート・ドライバ (LTDRIVER) への \$QIO 要求
以下の入出力要求を発行 <ul style="list-style-type: none"> <li>論理入出力要求</li> <li>IO\$M_MSCPMODIFS 修飾子を指定した論理入出力要求または仮想入出力要求</li> <li>非ファイル構造のプライベート・デバイスに対する物理入出力</li> </ul>	\$QIO
次のターミナル属性の変更 HANGUP SET_SPEED SECURE_SERVER	SET TERMINAL またはターミナル・ドライバ (TTDRIVER) /[NO]HANGUP /[NO]SET_SPEED /[NO]SECURE_SERVER
DEBNA/NI デバイス・ドライバに対する IO\$_ACCESS (診断) 機能の実行	同期通信回線 (XGDRIVER) への \$QIO 要求
Ethernet プロミスカス・モード・リスニングの有効化	
Ethernet コモン・ドライバに対する IO\$_ACCESS (診断) 機能の実行	

## A.23 PRMCEB 特権 (Devour)

PRMCEB 特権を有するユーザ・プロセスは、コモン・イベント・フラグ・クラスタ関連付け (\$ASCEFC) システム・サービスまたはコモン・イベント・フラグ・クラスタ削除 (\$DLCEFC) システム・サービスを実行することによって、パーマネント・コモン・イベント・フラグ・クラスタの作成や削除が行えます。コモン・イベント・フラグ・クラスタにより、連携するプロセス同士が相互に通信して、処理を同期させることができます。

この特権の付与は慎重に行ってください。パーマネント・コモン・イベント・フラグ・クラスタは、明示的に削除されないと、システムの動的メモリ内の領域を占有したままになり、システムの性能が低下する可能性があります。

## A.24 PRMGBL 特権 (Devour)

PRMGBL 特権を有するユーザ・プロセスは、セクションの作成とマップ (\$CRMPSC) システム・サービスまたはグローバル・セクション削除 (\$DGBLSC) システム・サービスを実行することによって、パーマネント・グローバル・セクションの作成または削除が行えます。さらに、この特権 (CMKRNL 特権と SYSGBL 特権も必要) を有するプロセスは、インストール・ユーティリティ (INSTALL) を使用できます。

グローバル・セクションとは、同時に複数のプロセスの仮想アドレス空間内にマッピングできる共用可能な構造のことです。すべてのプロセスが、同じコードまたはデータを参照します。グローバル・セクションは、リエントラント・サブルーチンまたはデータ・バッファに使用されます。

この特権の付与は慎重に行ってください。パーマネント・グローバル・セクションは、明示的に削除されないと、限られた資源であるグローバル・セクションとグローバル・ページ・テーブル内の空間を占有したままになります。

## A.25 PRMMBX 特権 (Devour)

PRMMBX 特権を有するユーザ・プロセスは、メールボックス作成とチャネル割り当て (\$CREMBX) システム・サービスまたはメールボックス削除 (\$DELMBX) システム・サービスを実行することによって、パーマネント・メールボックスの作成または削除が行えます。この特権では、\$CREMBX サービスを使用して一時的メールボックスを作成することもできます。

メールボックスとは、レコード指向入出力デバイスであるかのように扱われる仮想メモリ内のバッファのことです。メールボックスは、一般的なプロセス間通信に使用されます。

PRMMBX は、システムのすべてのユーザには付与しないでください。パーマネント・メールボックスは、メールボックス作成プロセスが削除されても自動的に

削除されません。そのため、システムの動的メモリの一部が使用され続けてしまいます。システムの動的メモリが不足してくると、システムの性能が低下するので注意してください。

## A.26 PSWAPM 特権 (System)

PSWAPM 特権を有するユーザ・プロセスは、プロセス・スワップ・モード設定 (\$SETSWM) システム・サービスを実行することによって、プロセスをバランス・セットからスワップ・アウトするかどうかを制御できます。プロセスは、自身をバランス・セット内にロックしたり (スワップの無効化のため)、バランス・セット内での自身に対するロックを解除したり (スワップの有効化のため) するのにこの特権が必要です。

この特権により、プロセスは、プロセス作成 (\$CREPRC) システム・サービスにオプションの引数を指定して実行するか、プロセスの作成に DCL の RUN コマンドを使用して /NOSWAPPING 修飾子を指定することによって、バランス・セット内でロックされる (スワップ・モードが無効になる) プロセスを作成することができます。さらに、プロセスは、メモリ内にページをロック (\$LCKPAG) システム・サービスを使用して、物理メモリ内のページまたはページ範囲をロックすることができます。

この特権は、性能上の理由からメモリ内にプロセスをロックする必要のあるユーザにのみ付与します。通常、これに該当するのはリアルタイム・プロセスです。条件を満たさないプロセスが、バランス・セット内のプロセスを自由にロックできるようになると、物理メモリが不必要に占有され、その結果、システム性能が低下する可能性があります。

## A.27 READALL 特権 (Objects)

READALL 特権を有するプロセスは、オブジェクトの読み込みを禁止する既存の制限の適用を回避できます。ただし、書き込みや削除が可能な BYPASS 特権とは異なり、READALL 特権ではオブジェクトの読み込みのみが許可され、バックアップ日など、バックアップ関連のファイル特性の更新が可能です。バックアップ操作についての詳細は、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』および『*OpenVMS システム管理者マニュアル*』を参照してください。

READALL 特権は、ボリュームのバックアップに十分な特権となるように考えられています。したがって、この特権は、システム・バックアップを実行するオペレータに付与します。

READALL 特権を有するプロセスは、次の作業を実行することができます。

作業	インタフェース
ユーザ登録レコードの読み込み	\$GETUAI
パーマネント・ネットワーク・データベース・レコードの表示	NCP

## A.28 SECURITY 特権 (System)

SECURITY 特権を有するプロセスは、システム・パスワードの変更 (DCL の SET PASSWORD/SYSTEM コマンドを使用)、システム・アラームと監査設定の変更 (DCL の SET AUDIT コマンドを使用) といったセキュリティ関連の機能を実行できます。この特権では、ユーザ・プロセスが SET AUDIT を使用して監査サーバ・プロセスの開始と停止が行えるだけでなく、SET AUDIT を使用して監査データベースの特性 (監査サーバ、システム監査ジャーナル、セキュリティ・アーカイブ・ファイル、資源モニタリング、監査モード、アラーム・モード、障害モードなどの特性) を変更することもできます。

この特権は、セキュリティ管理者にのみ付与します。条件を満たさないユーザがこの特権を獲得すると、そのユーザが、システムのセキュリティ機構を弱体化させ、システム・パスワードの不適切な設定によりユーザを締め出し、セキュリティ監査を無効にする可能性があります。

SECURITY 特権を有するプロセスは、次の作業も実行することができます。

作業	インタフェース
システム監査ログ・ファイル、監査サーバの設定などのシステム監査情報の表示	SHOW AUDIT
Hidden 属性の ACE の表示	SHOW SECURITY
システム侵入リストの表示またはレコードの削除	SHOW INTRUSION , DELETE/INTRUSION
セキュリティ・オペレータ・ターミナルの有効化	REPLY/ENABLE=SECURITY , \$SNDOPR
ボリューム上の保護サブシステムの有効化	MOUNT/SUBSYSTEM , \$MOUNT , SET VOLUME/SUBSYSTEM

## A.29 SETPRV 特権 (All)

SETPRV 特権を有するユーザ・プロセスは、オプションの引数を指定してプロセス作成 (\$CREPRC) システム・サービスを実行するか、または DCL の RUN コマンドを実行してプロセスを作成することによって、自身よりも上位の特権を有するプロセスを作成できます。この特権を有するプロセスは、DCL の SET PROCESS/PRIVILEGES コマンドを実行することで、任意の特権を獲得することもできます。

SETPRV 特権では、プロセスが任意またはすべての特権を有効にできるため、他の特権と同様に、十分に注意して特権を付与します。

### A.30 SHARE 特権 (All)

SHARE 特権を有するプロセスは、入出力チャンネル割り当て (\$ASSIGN) システム・サービスを使用して、別のプロセスに割り当てられているデバイスや非共用デバイスに対して、チャンネルを割り当てることが可能です。

この特権は、プリント・シンピオントなど、システム・プロセスにのみ付与します。それ以外の対象に付与すると、条件を満たさないユーザが、他のユーザが所有するデバイスの操作を妨げる恐れがあります。

### A.31 SHMEM 特権 (Devour)

SHMEM 特権を有するユーザ・プロセスは、適切な PRMGBL 特権、PRMMBX 特権、SYSGBL 特権、および TMPMBX 特権を持っていれば、複数のプロセッサが共用するメモリにグローバル・セクションとメールボックス (パーマネントおよび一時的のどちらも可) を作成できます。ローカル・メモリの場合と同様に、マルチポート・メモリにおける一時メールボックスに必要な容量は、プロセスに対するバッファード入出力バイト数の制限 (BYTLM) と照合して計算されます。

この特権を有するユーザ・プロセスは、コモン・イベント・フラグ・クラスタ関連付け (\$ASCEFC) システム・サービスまたはコモン・イベント・フラグ・クラスタ関連付け解除 (\$DACEFC) システム・サービスを実行することによって、共用メモリ内でイベント・フラグ・クラスタの作成や削除が行えます。

### A.32 SYSGBL 特権 (Files)

SYSGBL 特権を有するユーザ・プロセスは、セクションの作成とマップ (\$CRMPSC) システム・サービスまたはグローバル・セクション削除 (\$DGBLSC) システム・サービスを実行することによって、システム・グローバル・セクションの作成または削除が行えます。さらに、この特権 (CMKRNL 特権と PRMGBL 特権も必要) を有するプロセスは、インストール・ユーティリティ (INSTALL) を使用できます。

この特権を付与するときは注意が必要です。システム・グローバル・セクションは、限られた資源であるグローバル・セクションとグローバル・ページ・テーブル内に領域が必要です。

### A.33 SYSLOCK 特権 (System)

SYSLOCK 特権を有するユーザ・プロセスは、ロック要求キュー登録 (\$ENQ) システム・サービスを使用してシステム全体の資源をロックしたり、ロック情報取得 (\$GETLKI) システム・サービスを使用してシステム資源に関する情報を取得したりできます。

この特権は、システム全体のネームスペースにおいて、資源をロックするプログラムを実行する必要があるユーザに付与します。ただし、この特権を付与するときは注意が必要です。SYSLOCK 特権を有するユーザは、すべてのシステム・ソフトウェアとユーザ・ソフトウェアの同期を妨げる可能性があります。

## A.34 SYSNAM 特権 (All)

SYSNAM 特権を有するユーザ・プロセスは、システム論理名テーブルに対する任意アクセス制御の適用を回避できます。そして、論理名作成 (\$CRELNM) システム・サービスおよび論理名削除 (\$DELLNM) システム・サービスを使用して、システム論理名テーブルに名前を挿入したり、システム論理名テーブルから名前を削除したりできます。この特権を有するプロセスは、DCL の ASSIGN コマンドと DEFINE コマンドを使用してシステム論理名テーブルに名前を追加したり、DEASSIGN コマンドを使用してシステム論理名テーブルから名前を削除したりできます。追加および削除は、ユーザ・モード、エグゼクティブ・モードのいずれでも可能です。

適切なマウント・コマンドまたはディスマウント・コマンド、あるいはシステム・サービスを使用して、システム・ボリュームのマウント、またはシステム・ボリュームやグループ・ボリュームのディスマウントを行うには、SYSNAM 特権が必要です。

この特権は、システム論理名 (ユーザ・デバイス、ライブラリ・ディレクトリ、システム・ディレクトリの名前など) を定義する必要があるシステム・オペレータまたはシステム・プログラマにのみ付与します。SYSNAM 特権を有するプロセスは、SYS\$SYSTEM や SYSUAF などの重要なシステム論理名を定義し直して、システムの制御権を握ることができる点に注意してください。

SYSNAM 特権を有するプロセスは、次の作業も実行することができます。

作業	インタフェース
MAIL 保守レコードへのアクセス	MAIL
MAIL 転送レコードの変更	MAIL
ネットワーク・オブジェクトの宣言	NETACP
IPC の関連付けの作成	\$IPC
CMKRNL を使用した、システム・ライト・リストに対する識別子の追加と削除	SET RIGHTS_LIST/SYSTEM, \$GRANTID, \$REVOKID

## A.35 SYSPRV 特権 (All)

SYSPRV 特権を有するプロセスは、システム保護フィールドによって保護されているオブジェクトにアクセスしたり、オブジェクトの所有者 (UIC)、UIC ベースの保護コード、ACL の読み込みと変更ができます。オブジェクトがシステム・アクセス



から保護されている場合でも、SYSPRV 特権を有するプロセスは、このオブジェクトの保護を変更して、オブジェクトにアクセスできます。SYSPRV 特権を有するプロセスは、システム・ユーザ登録ファイル (SYSUAF.DAT) 内のエントリについて、追加、変更、削除を行うことができます。

この特権を付与するときは注意が必要です。通常、この特権は、システム管理者とセキュリティ管理者にのみ付与します。条件を満たさないユーザがシステム・アクセス・ライトを獲得すると、オペレーティング・システム、およびその他へのサービスが容易に混乱させられる可能性があります。たとえば、システム障害、システムおよびユーザのすべてのデータの破壊、機密情報の漏洩などの混乱が考えられます。

SYSPRV 特権を有するプロセスは、次の作業も実行することができます。

作業	インタフェース
ファイルの有効期限の変更	SET FILE/EXPIRATION
インターロックされたキューの再試行回数の変更	Ethernet 802 ドライバ (DEBNA/NI) への \$QIO 要求
ポート・コマンド・レジスタにおけるスピン・ウェイト時間の設定	Ethernet 802 ドライバ (DEBNA) への \$QIO 要求
メール・メッセージ内の FROM フィールドの設定	MAIL ルーチン
MAIL 保守レコードへのアクセス	MAIL
MAIL データベース・レコードの変更または削除	MAIL
ローカル・エリア・クラスタのグループ番号とパスワードの変更	SYSMAN の CLUSTER_AUTHORIZE コンポーネント
トランザクション回復の実行、コーディネータとしてのトランザクションへの参加、トランザクションの遷移	DECdtm ソフトウェア

グループ UIC がシステム・パラメータの MAXSYSGRP の値以下のプロセスは、SYSPRV を暗黙に有します。SYSPRV 特権または暗黙の SYSPRV 特権を有するプロセスは、次の作業も実行できます。

作業	インタフェース
磁気テープの初期化	\$INIT_VOL
新規作成ファイルに対する所有者 ACE の作成の無効化	F11BXQP への \$QIO 要求
ディレクトリのファイル・ヘッダに含まれているディレクトリ・ビットのクリア	F11BXQP への \$QIO 要求、SET FILE/NODIRECTORY
ボリューム・ロックの獲得または解放	F11BXQP への \$QIO 要求

作業	インタフェース
ボリュームに対する強制マウント検証	F11BXQP への \$QIO 要求
アクセス・ロック・ビットが設定されていないファイル・アクセス・ウィンドウの作成	F11BXQP への \$QIO 要求
ボリューム・ロックに対するヌル・ロック・モードの指定	F11BXQP への \$QIO 要求
ロックされているファイルへのアクセス	F11BXQP への \$QIO 要求
ボリュームにおけるディスク制限の無効化	F11BXQP への \$QIO 要求
ボリュームにおけるディスク制限の有効化	F11BXQP への \$QIO 要求

## A.36 TMPMBX 特権 (Normal)

TMPMBX 特権を有するユーザ・プロセスはメールボックス作成とチャンネル割り当て (\$CREMBX) システム・サービスを実行することによって、一時メールボックスを作成できます。

メールボックスとは、レコード指向入出力デバイスであるかのように扱われる仮想メモリ内のバッファのことです。メールボックスは、一般的なプロセス間通信に使用されます。明示的に削除する必要があるパーマネント・メールボックスとは異なり、一時メールボックスは、どのプロセスからも参照されなくなった時点で自動的に削除されます。

この特権は、プロセス間通信がスムーズに行われるようにするために、システムのすべてのユーザに付与します。一時メールボックスの作成を許可することでシステム性能が低下することはまずありません。一時メールボックスの数が、システムの動的メモリの利用に対する制限 (BYTLM クォータ) によって制御されているためです。

## A.37 UPGRADE 特権 (All)

UPGRADE 特権を有するプロセスは、強制アクセス制御を操作できます。この特権を有するプロセスは、Biba 制限 (\*) プロパティに違反して、より機密性の高いオブジェクトへの書き込みを行うことができます。この特権は、SEVMS など、高度なセキュリティ製品用に予約済みです。

## A.38 VOLPRO 特権 (Objects)

VOLPRO 特権を有するユーザ・プロセスは、次のことが可能です。

- ユーザ自身の UIC とは異なる所有者 UIC を使用した、以前に使用されているボリュームの初期化
- 別のユーザが所有するテープまたはディスク・ボリュームに設定されている有効期限の無効化

- /FOREIGN 修飾子を使用した，別のユーザが所有する Files-11 ボリュームのマウント
- ボリュームに対する所有者 UIC 保護の無効化

VOLPRO 特権では，ユーザのプロセスがマウントまたは初期化することのできるボリュームのみを制御できます。/SYSTEM 修飾子を使用してマウントされているボリュームは，プロセスが SYSNAM 特権も有する場合を除き，VOLPRO 特権を有するプロセスからは安全です。

VOLPRO 特権を付与するときは特に注意が必要です。条件を満たさないユーザがボリュームの保護を無効にできると，オペレーティング・システム，およびその他へのサービスが混乱させられる可能性があります。たとえば，データベースの破壊や機密情報の漏洩などの混乱が考えられます。

VOLPRO 特権を有するプロセスは，次の作業を実行することができます。

作業	インタフェース
ボリュームのディスマウント	DISMOUNT/ABORT，\$DISMOU
ボリュームの初期化	\$INIT_VOL
フォーリン・マルチボリューム磁気テープ・セットのマウント	MOUNT/MULTI_VOLUME
ボリューム・ラベルまたはアクセシビリティの無効化	\$MOUNT
ブランク・テープの初期化	REPLY/BLANK_TAPE，\$SNDOPR
ファイル・アクセス・エラーが発生した後の磁気テープ初期化中のアクセスの無効化	\$INIT_VOL
エラー発生時のボリュームに対する書き込みロックの無効化	\$MOUNT
以前のシャドウ・セット・メンバの書き込み保護の無効化	\$MOUNT
ボリュームの有効期限，保護，所有権の無効化	\$MOUNT

## A.39 WORLD 特権 (System)

WORLD 特権を有するユーザ・プロセスは，次に示すプロセス制御システム・サービスを実行することにより，同じグループ内のプロセスかどうかに関係なく，他のプロセスに働きかけることができます。

プロセス一時停止 (\$SUSPND)  
 プロセス再開 (\$RESUME)  
 プロセス削除 (\$DELPRI)  
 優先順位設定 (\$SETPRI)

ウェイクアップ (\$WAKE)  
スケジューリングされたウェイクアップ (\$SCHDWK)  
ウェイクアップ取り消し (\$CANWAK)  
強制終了 (\$FORCEX)

ユーザのプロセスは、ジョブ取得/プロセス情報 (\$GETJPI) システム・サービスを実行することで、自身のグループ外の他のプロセスを調査することができます。WORLD 特権を有するプロセスは、他のすべてのプロセスに対して SET PROCESS コマンドを実行できます。WORLD 特権を有するプロセスは、ロック情報取得 (\$GETLKI) システム・サービスを使用して、別のグループに属するプロセスが保持しているロックに関する情報を取得することもできます。

プロセスは、自身が作成したサブプロセスの制御、およびこれらのサブプロセスの調査のために、特別な権限を得る必要はありません。同じグループ内の他のプロセスへの働きかけ、および調査のために必要な特権は、GROUP 特権だけです。ただし、WORLD 特権は、自身のグループ以外のプロセスへの働きかけや調査を行う必要があるユーザに付与します。

## OpenVMS システム・ファイルの保護

付録 B では、OpenVMS のシステム・ファイルとその保護コードの一覧を示します。これにより OpenVMS のシステム・ファイルを定期的に監視し、改ざんがないことを保証できるようになります。B.1 節では、ファイルに割り当てられている保護コードと所有権を特定します。B.2 節には、OpenVMS メディアで提供されるシステム・ファイルの一覧を示します。

OpenVMS のシステム・ファイルを保護する方法については、第 8 章、特に 8.9.2 項の「システム・ファイルの保護」を参照してください。

### B.1 標準の所有権と保護

システム (SYSTEM) は、1 つを除き、OpenVMS のすべてのシステム・ファイルを所有します。ディレクトリ MOM\$SYSTEM は、UIC [376,375] が所有します。

表 B-1 に示されているファイルを除き、SYS\$SYSDEVICE:[VMS\$COMMON] にあるすべてのファイルの保護コードは S:RWED,O:RWED,G:RWED,W:RE になります。

ディレクトリ VMS\$COMMON.DIR と SYS\$SYSDEVICE:[SYSx.DIR] にあるファイルの保護コードは S:RWE,O:RWE,G:RE,W:RE になります。

表 B-1: 標準の OpenVMS システム・ファイル保護の例外

ファイル	保護
[VMS\$COMMON]	
DECW\$DEFAULTS.DIR	MOM\$SYSTEM.DIR S:RWE,O:RWE,G:RE,W:RE
SYS\$KEYMAP.DIR	SYS\$LDR.DIR
SYS\$STARTUP.DIR	SYSCBI.DIR
SYSERR.DIR	SYSEXE.DIR
SYSFONT.DIR	SYSHLP.DIR
SYSLIB.DIR	SYSMAINT.DIR
SYSMGR.DIR	SYSMSG.DIR
SYSTEST.DIR	SYSUPD.DIR
VUE\$LIBRARY.DIR	
[VMS\$COMMON.SYS\$KEYMAP]	

表 B-1: 標準の OpenVMS システム・ファイル保護の例外 (続き)

DECW.DIR		S:RWE,O:RWE,G:RE,W:RE
[VMS\$COMMON.SYS\$KEYMAP.DECW]		
SYSTEM.DIR	USER.DIR	S:RWE,O:RWE,G:RE,W:RE
[VMS\$COMMON.SYSEXEXE]		
ISL_LVAX_061.SYS	ISL_SVAX_061.SYS	S:RWED,O:RWED,G:RE,W:RE
NETPROXY.DAT		S:RWE,O:RWE,G:RWE,W
NET\$PROXY.DAT		S:RWE,O:RWE,G:RWE,W
MSGHLP\$MAIN.EXE		S:RE,O:RE,G:RE,W:RE
RIGHTSLIST.DAT		S:RWED,O:RWED,G:R,W
SYSUAF.DAT		S:RWED,O:RWED,G,W
SYVMS\$OBJECTS.DAT- SUAF.DAT		S:RWE,O:RWE,G:RE,W
[VMS\$COMMON.SYSFONT]		
DECW.DIR	PS_FONT_METRICS.DIR	S:RWE,O:RWE,G:RE,W:RE
VWS.DIR	XDPS.DIR	
[VMS\$COMMON.SYSFONT]		
DECW.DIR	PS_FONT_METRICS.DIR	S:RWE,O:RWE,G:RE,W:RE
VWS.DIR	XDPS.DIR	
[VMS\$COMMON.SYS- FONT.DECW]		
100DPI.DIR	75DPI.DIR	S:RWE,O:RWE,G:RE,W:RE
COMMON.DIR	CURSOR16.DIR	
CURSOR32.DIR	USER_100DPI.DIR	
USER_75DPI.DIR	USER_COMMON.DIR	
USER_CURSOR16.DIR	USER_CURSOR32.DIR	
[VMS\$COMMON.SYSHLP]		
DECW.DIR	VMSDOC.DIR	S:RWE,O:RWE,G:RE,W:RE
MSGHLP\$ENGLISH.EXE		S:RE,O:RE,G:RE,W:RE
EXAMPLES.DIR		S:RWE,O:RWE,G:RE,W:RE
[VMS\$COMMON.SYSLIB]		
CDA\$ACCESS.EXE	DECW\$DWTLIBSHR.EXE	S:RW,O:RWED,G:R,W:R
DECW\$PRINTWGT- SHR.EXE	DECW\$XLIBSHR.EXE	
MSGHLP\$ENGLISH.EXE	MSGHLP\$SHARE.EXE	S:RE,O:RE,G:RE,W:RE
VMS\$PASSWORD_DIC TIONARY.DATA		S:RE,O:RE,G,W

表 B-1: 標準の OpenVMS システム・ファイル保護の例外 (続き)

XDPS\$DPSBIND- INGSSHR.EXE	XDPS\$DP- SCLIENTSHR.EXE	S:RW,O:RWED,G:R,W:R
XDPS\$DPSLIBSHR.EXE	XNL\$SHR.EXE	
[VMS\$COMMON.SYSMGR]		
SECURITY.AUDIT\$JOUR- NAL		S:RWED,O:RWED,G:RE,W
VMS\$AUDIT_SERVER.DAT		S:RWE,O:RWE,G:RE,W
WELCOME.TEMPLATE	WELCOME.TXT	S:RWED,O:RWED,G:RE,W:RE
[VMS\$COMMON.VUE\$LIBRARY]		
SYSTEM.DIR	USER.DIR	S:RWE,O:RWE,G:RE,W:RE

## B.2 OpenVMS システム・ファイルの一覧

次の節では、システム・ファイルの一覧を、DCL の DIRECTORY コマンドで生成される順番で示します。

### B.2.1 最上位ディレクトリのファイル

クラスタ・システムの最上位ディレクトリ VMS\$COMMON のファイルには、次のファイルが含まれています。

```
Directory SYS$SYSDEVICE:[VMS$COMMON]

DECW$DEFAULTS.DIR;1      MOM$SYSTEM.DIR;1
SYS$KEYMAP.DIR;1         SYS$LDR.DIR;1
SYS$STARTUP.DIR;1        SYSCBI.DIR;1
SYSERR.DIR;1             SYSEXEC.DIR;1
SYSPFONT.DIR;1           SYSHLP.DIR;1
SYSLIB.DIR;1             SYSMINT.DIR;1
SYSMGR.DIR;1             SYMSG.DIR;1
SYSTEST.DIR;1            SYSUPD.DIR;1
VUE$LIBRARY.DIR;1

Total of 17 files.

Directory SYS$SYSDEVICE:[VMS$COMMON.DECW$DEFAULTS]

SYSTEM.DIR;1             USER.DIR;1

Total of 2 files.
```

### B.2.2 SYS\$KEYMAP のファイル

ディレクトリ SYS\$KEYMAP には、例 B-1 に示すファイルが含まれています。

## 例 B-1: SYS\$KEYMAP のファイル

---

```
Directory SYS$SYSDEVICE:[VMS$COMMON.SYS$KEYMAP]

DECW.DIR;1

Total of 1 file.

Directory SYS$SYSDEVICE:[VMS$COMMON.SYS$KEYMAP.DECW]

SYSTEM.DIR;1          USER.DIR;1

Total of 2 files.
```

---

## B.2.3 SYS\$LDR のファイル

ディレクトリ SYS\$LDR には、例 B-2 に示すファイルが含まれています。

### 例 B-2: SYS\$LDR のファイル

---

```
Directory SYS$SYSDEVICE:[VMS$COMMON.SYS$LDR]

CLASS_SCHEDULER.EXE;1
CONINTERR.EXE;1
CRDRIVER.EXE;1
CSDRIVER.EXE;1
CVDRIIVER.EXE;1
DBDRIVER.EXE;1
DDIF$RMS_EXTENSION.EXE;1
DLDRIVER.EXE;1
DQDRIVER.EXE;1
DSDRIVER.EXE;1
DVDRIIVER.EXE;1
DYDRIVER.EXE;1
ECDRIVER.EXE;1
EPDRIVER.EXE;1
ESDRIVER.EXE;1
ESS$LADDRIVER.EXE;1
ESS$MADDRIVER.EXE;1
EVENT_FLAGS_AND_ASTS.EXE;1
EXDRIVER.EXE;1
EZDRIVER.EXE;1
FCDRIVER.EXE;1
FQDRIVER.EXE;1
FXDRIVER.EXE;1
GAADRIVER.EXE;1
GBBDRIVER.EXE;1
GCBDRIVER.EXE;1
GEBDRIVER.EXE;1
GFBDRIVER.EXE;1
IKDRIVER.EXE;1
IMDRIVER.EXE;1
IO_ROUTINES.EXE;1
LCDRIVER.EXE;1
LMF$GROUP_TABLE.EXE;1

CNDRIVER.EXE;1
CPULOQ.A.EXE;1
CS9AQDRIVER.EXE;1
CTDRIVER.EXE;1
CWDRIVER.EXE;1
DDDRIVER.EXE;1
DKDRIVER.EXE;1
DMDRIVER.EXE;1
DRDRIVER.EXE;1
DUDRIVER.EXE;1
DXDRIVER.EXE;1
DZDRIVER.EXE;1
EFDRIVER.EXE;1
ERRORLOG.EXE;1
ESS$DADDRIVER.EXE;1
ESS$LASTDRIVER.EXE;1
ETDRIVER.EXE;1
EXCEPTION.EXE;1
EXEC_INIT.EXE;1
FBDRIVER.EXE;1
FPEMUL.EXE;1
FTDRIVER.EXE;1
FYDRIVER.EXE;1
GABDRIVER.EXE;1
GCADRIVER.EXE;1
GDDRIVER.EXE;1
GECDRIVER.EXE;1
GKDRIVER.EXE;1
IMAGE_MANAGEMENT.EXE;1
INDRIVER.EXE;1
LADDRIVER.EXE;1
LIDRIVER.EXE;1
LOCKING.EXE;1
```



## 例 B-2: SYS\$LDR のファイル (続き)

---

LOGICAL_NAMES.EXE;1	LPDRIVER.EXE;1
LTDRIVER.EXE;1	MBXDRIVER.EXE;1
MESSAGE_ROUTINES.EXE;1	MKDRIVER.EXE;1
NDDRIVER.EXE;1	NET\$CSMACD.EXE;1
NET\$FDDI.EXE;1	NETDRIVER.EXE;1
NODRIVER.EXE;1	PADRIVER.EXE;1
PAGE_MANAGEMENT.EXE;1	PBDRIVER.EXE;1
PDDRIVER.EXE;1	PEDRIVER.EXE;1
PIDRIVER.EXE;1	PKBDRIVER.EXE;1
PKCDRIVER.EXE;1	PKIDRIVER.EXE;1
PKNDRIVER.EXE;1	PKRDRIVER.EXE;1
PKSDRIVER.EXE;1	PKXDRIVER.EXE;1
PRIMITIVE_IO.EXE;1	PROCESS_MANAGEMENT.EXE;1
PUDRIVER.EXE;1	PWDRIVER.EXE;1
RECOVERY_UNIT_SERVICES.EXE;1	RMS.EXE;1
RTTDRIVER.EXE;1	RXDRIVER.EXE;1
SECURITY.EXE;1	SHDRIVER.EXE;1
SNAPSHOT_SERVICES.EXE;1	SODRIVER.EXE;1
SYS\$CLUSTER.EXE;1	SYS\$IPC_SERVICES.EXE;1
SYS\$NETWORK_SERVICES.EXE;1	SYS\$SCS.EXE;1
SYS\$TRANSACTION_SERVICES.EXE;1	SYS\$UTC_SERVICES.EXE;1
SYS.EXE;2	SYSDEVICE.EXE;1
SYSGETSYI.EXE;1	SYSLDR_DYN.EXE;1
SYSLICENSE.EXE;1	SYSLOA1202.EXE;1
SYSLOA1302.EXE;1	SYSLOA1303.EXE;1
SYSLOA1701.EXE;1	SYSLOA410.EXE;1
SYSLOA41D.EXE;1	SYSLOA41W.EXE;1
SYSLOA420.EXE;1	SYSLOA42D.EXE;1
SYSLOA42S.EXE;1	SYSLOA42W.EXE;1
SYSLOA43.EXE;1	SYSLOA43D.EXE;1
SYSLOA43S.EXE;1	SYSLOA43W.EXE;1
SYSLOA440.EXE;1	SYSLOA46.EXE;1
SYSLOA49.EXE;1	SYSLOA520.EXE;1
SYSLOA60.EXE;1	SYSLOA600.EXE;1
SYSLOA640.EXE;1	SYSLOA64D.EXE;1
SYSLOA650.EXE;1	SYSLOA65D.EXE;1
SYSLOA660.EXE;1	SYSLOA66D.EXE;1
SYSLOA670.EXE;1	SYSLOA67D.EXE;1
SYSLOA690.EXE;1	SYSLOA69D.EXE;1
SYSLOA700.EXE;1	SYSLOA70D.EXE;1
SYSLOA730.EXE;1	SYSLOA750.EXE;1
SYSLOA780.EXE;1	SYSLOA790.EXE;1
SYSLOA8NN.EXE;1	SYSLOA8PS.EXE;1
SYSLOA8SS.EXE;1	SYSLOA9AQ.EXE;1
SYSLOA9CC.EXE;1	SYSLOA9RR.EXE;1
SYSLOAUV1.EXE;1	SYSLOAUV2.EXE;1
SYSLOAWS1.EXE;1	SYSLOAWS2.EXE;1
SYSLOAWS.D.EXE;1	SYSTEM_DEBUG.EXE;1
SYSTEM_PRIMITIVES.EXE;1	SYSTEM_PRIMITIVES_MIN.EXE;1
SYSTEM_SYNCHRONIZATION.EXE;1	SYSTEM_SYNCHRONIZATION_MIN.EXE;1
SYSTEM_SYNCHRONIZATION_SPC.EXE;1	SYSTEM_SYNCHRONIZATION_UNI.EXE;1
TFDRIVER.EXE;1	TMDRIVER.EXE;1
TSDRIVER.EXE;1	TTDRIVER.EXE;1

## 例 B-2: SYS\$LDR のファイル (続き)

---

TUDRIVER.EXE;1	TVDRIVER.EXE;1
VAXCLUSTER_CACHE.EXE;1	VAXEMUL.EXE;1
VBSS.EXE;1	VECTOR_PROCESSING.EXE;1
VMS\$SYSTEM_IMAGES.DATA;1	VVIEF_BOOTSTRAP.EXE;1
WORKING_SET_MANAGEMENT.EXE;1	WPDRIVER.EXE;1
WSDRIVER.EXE;1	XADRIVER.EXE;1
XDDRIVER.EXE;1	XEDRIVER.EXE;1
XFDRIVER.EXE;1	XIDRIVER.EXE;1
XMDRIVER.EXE;1	XQDRIVER.EXE;1
XTDRIVER.EXE;1	YCDRIVER.EXE;1
YEDRIVER.EXE;1	YFDRIVER.EXE;1
YIDRIVER.EXE;1	

Total of 195 files.

---

## B.2.4 SYS\$STARTUP および SYS\$ERR のファイル

ディレクトリ SYS\$STARTUP および SYS\$ERR には、例 B-3 に示すファイルが含まれています。

### 例 B-3: SYS\$STARTUP および SYS\$ERR のファイル

---

Directory SYS\$SYSDEVICE:[VMS\$COMMON.SYS\$STARTUP]

DEBUG\$STARTUP.COM;1	DECDTM\$SHUTDOWN.COM;1
DECDTM\$STARTUP.COM;1	DNS\$CLERK_STARTUP.COM;1
DNS\$CLERK_STOP.COM;1	ESS\$LAD_STARTUP.COM;1
ESS\$LAD_STARTUP.DAT;1	ESS\$LAST_STARTUP.COM;1
ESS\$LAST_STARTUP.DAT;1	ESS\$STARTUP.COM;1
IPC\$STARTUP.COM;1	LAT\$CONFIG.COM;1
LAT\$STARTUP.COM;1	LICENSE_CHECK.EXE;1
VMS\$BASEENVIRON-050_LIB.COM;1	VMS\$BASEENVIRON-050_SMISERVER.COM;1
VMS\$BASEENVIRON-050_VMS.COM;1	VMS\$CONFIG-050_AUDIT_SERVER.COM;1
VMS\$CONFIG-050_CACHE_SERVER.COM;1	VMS\$CONFIG-050_CSP.COM;1
VMS\$CONFIG-050_ERRFMT.COM;1	VMS\$CONFIG-050_JOBCTL.COM;1
VMS\$CONFIG-050_LMF.COM;1	VMS\$CONFIG-050_OPCOM.COM;1
VMS\$CONFIG-050_SECURITY_SERVER.COM;1	VMS\$CONFIG-050_SHADOW_SERVER.COM;1
VMS\$CONFIG-050_VMS.COM;1	VMS\$DEVICE_STARTUP.COM;1
VMS\$INITIAL-050_CONFIGURE.COM;1	VMS\$INITIAL-050_LIB.COM;1
VMS\$INITIAL-050_VMS.COM;1	VMS\$LAYERED.DAT;1
VMS\$LPBEGIN-050_STARTUP.COM;1	VMS\$PHASES.DAT;1
VMS\$VMS.DAT;1	

Total of 35 files.

Directory SYS\$SYSDEVICE:[VMS\$COMMON.SYS\$ERR]

ERRSNAP.COM;1

### 例 B-3: SYS\$STARTUP および SYS\$ERR のファイル (続き)

---

Total of 1 file.

---

## B.2.5 SYSEXE のファイル

ディレクトリ SYS\$EXE には、例 B-4 に示すファイルが含まれています。

### 例 B-4: SYSEXE のファイル

---

Directory SYS\$SYSDEVICE:[VMS\$COMMON.SYSEXE]

ACC.EXE;1	ACLED.T.EXE;1
AGEN\$FEEDBACK.EXE;1	ANALAUDIT.EXE;1
ANALIMDMP.EXE;1	ANALYZBAD.EXE;1
ANALYZOBJ.EXE;1	ANALYZRMS.EXE;1
AUDIT_SERVER.EXE;1	AUTHORIZE.EXE;1
BACKUP.EXE;1	BADBLOCK.EXE;1
BOOT58.EXE;1	BOOTBLOCK.EXE;1
CDU.EXE;1	CHECKSUM.EXE;1
CIA.EXE;1	CLUE.EXE;1
CONFIGURE.EXE;1	CONVERT.EXE;1
CONVERT_PROXY.EXE;1	COPY.EXE;1
CREATE.EXE;1	CREATEFDL.EXE;1
CSP.EXE;1	CVTNAFV5.EXE;1
DBLMSGMGR.EXE;1	DCL.EXE;1
DCLDEF.STB;1	DECDTMDEF.STB;1
DECW\$DWT_DECNET.EXE;1	DECW\$DWT_FONT_DAEMON.EXE;1
DECW\$DWT_STARTXTDRIVER.EXE;1	DECW\$FONTCOMPILER.EXE;1
DECW\$MKFONTDIR.EXE;1	DECW\$SERVER_MAIN.EXE;1
DECW\$SETSHODIS.EXE;1	DELETE.EXE;1
DIFF.EXE;1	DIRECTORY.EXE;1
DISKQUOTA.EXE;1	DISMOUNT.EXE;1
DNS\$ADVER.EXE;1	DNS\$ANALYZE.EXE;1
DNS\$SOLICIT.EXE;1	DSRINDEX.EXE;1
DSRTOC.EXE;1	DTEPAD.EXE;1
DTR.COM;1	DTRECV.EXE;1
DTSEND.EXE;1	DUMP.EXE;1
EDF.EXE;1	EDT.EXE;1
ERF.EXE;1	ERFADPTR.EXE;1
ERFBRIEF.EXE;1	ERFBUS.EXE;1
ERFCNTRL.EXE;1	ERFCVAX.EXE;1
ERFDISK.EXE;1	ERFDISK2.EXE;1
ERFMISC.EXE;1	ERFMSCP.EXE;1
ERFNVAX.EXE;1	ERFRLTIM.EXE;1
ERFSCSI.EXE;1	ERFSUMM.EXE;1
ERFTAPE.EXE;1	ERFUVAX.EXE;1
ERFV14.EXE;1	ERFV9000.EXE;1
ERFVAX7XX.EXE;1	ERFVX8200.EXE;1
ERFVX8600.EXE;1	ERFVX87XX.EXE;1
ERFXRP.EXE;1	ERRFMT.EXE;1
ERRSNAP.EXE;1	ESS\$ISL_VMSLOAD.EXE;1

#### 例 B-4: SYSEXEC のファイル (続き)

---

ESS\$LADCP.EXE;1	ESS\$LASTCP.EXE;1
EVL.COM;1	EVL.EXE;1
EXCHANGE\$NETWORK.EXE;1	EXCHANGE.EXE;1
F11AACP.EXE;1	F11BXQP.EXE;1
F11CACP.EXE;1	F11DACP.EXE;1
FAL.COM;1	FAL.EXE;1
FILESERV.EXE;1	HLD.COM;1
HLD.EXE;1	HSCPAD.EXE;1
IMGDEF.STB;1	INIT.EXE;1
INPSMB.EXE;1	INSTALL.EXE;1
IPCACP.EXE;1	IPCPDEF.STB;1
ISL_LVAX_061.SYS;1	ISL_SVAX_061.SYS;1
JBC\$COMMAND.EXE;1	JBC\$JOB_CONTROL.EXE;1
LALOAD.EXE;1	LALoader.EXE;1
LATACP.EXE;1	LATCP.EXE;1
LATSYM.EXE;1	LIBRARIAN.EXE;1
LINK.EXE;1	LMCP.EXE;1
LMF\$LICENSE.LDB;1	LMF\$LURT.DAT;1
LMF.EXE;1	LOGINOUT.EXE;1
LTPAD.EXE;1	MACRO32.EXE;1
MAIL.COM;1	MAIL.EXE;1
MAILEDIT.COM;1	MAIL_SERVER.EXE;1
MESSAGE.EXE;1	MIRROR.COM;1
MIRROR.EXE;1	MOM.COM;1
MOM.EXE;1	MONITOR.EXE;1
MSCP.EXE;1	MSGHLP\$MAIN.EXE;1
MTAAACP.EXE;1	NCP.EXE;1
NCS.EXE;1	NET\$NAME_SERVER.EXE;1
NETACP.EXE;1	NETDEF.STB;1
NETSERVER.COM;1	NETSERVER.EXE;1
NICONFIG.COM;1	NICONFIG.EXE;1
NML.COM;1	NML.EXE;1
OPCCRASH.EXE;1	OPCOM.EXE;1
PATCH.EXE;1	PCSI\$MAIN.EXE;1
PHONE.COM;1	PHONE.EXE;1
PRTSMB.EXE;1	QMAN\$QUEUE_MANAGER.EXE;1
QUEMAN.EXE;1	RECLAIM.EXE;1
RECOVER.EXE;1	REMACP.EXE;1
RENAME.EXE;1	REPLY.EXE;1
REQSYSDEF.STB;1	REQUEST.EXE;1
RIGHTSLIST.DAT;1	RMS.STB;1
RMSDEF.STB;1	RMSREC\$SERVER.EXE;1
RTB.EXE;1	RTPAD.EXE;1
RUNDET.EXE;1	RUNOFF.EXE;1
SA_STARTUP.COM;1	SCSDEF.STB;1
SDA.EXE;1	SDLNPARSE.EXE;1
SEARCH.EXE;1	SECURITY_SERVER.EXE;1
SET.EXE;1	SETAUDIT.EXE;1
SETFILENOMOVE.COM;1	SETFILENOMOVE.EXE;1
SETP0.EXE;1	SETRIGHTS.EXE;1
SETSHOSECUR.EXE;1	SETSHOSECUR.EXE;1
SETWATCH.EXE;1	SHADOW_SERVER.EXE;1
SHOW.EXE;1	SHUTDOWN.COM;1

## 例 B-4: SYSEXE のファイル (続き)

---

SHWCLSTR.EXE;1	SMGBLDTRM.EXE;1
SMGMAPTRM.EXE;1	SMGTERMS.TXT;1
SMISERVER.EXE;1	SMPUTIL.EXE;1
SNAPSHOT\$DRIVER.DAT;1	SNAPSHOT\$IMAGE.DAT;1
SNAPSHOT\$LOADED_IMAGES.DAT;1	SNAPSHOT\$WATCHDOG.EXE;1
SNAPSHOT.EXE;1	SORTMERGE.EXE;1
STABACCOP.EXE;1	STABACKUP.EXE;1
STACONFIG.EXE;1	STANDCONF.EXE;1
STARTUP.COM;1	STASYSGEN.EXE;1
STOPREM.EXE;1	SUBMIT.EXE;1
SUCCESS.COM;1	SUMSLP.EXE;1
SYS.MAP;1	SYS.STB;1
SYSBOOT.EXE;1	SYSBOOT_XDELTA.EXE;1
SYSDEF.STB;1	SYSGEN.EXE;1
SYSINIT.EXE;1	SYSMAN.EXE;1
SYSUAF.DAT;1	SYSUAF.TEMPLATE;1
TECO32.EXE;1	TERMTABLE.EXE;1
TERMTABLE.TXT;1	TERTIARY_VMB.EXE;1
TFF\$MASTER.DAT;1	TFU.EXE;1
TMSCP.EXE;1	TPSERV.EXE;1
TPU.EXE;1	TYPE.EXE;1
UNLOCK.EXE;1	UTC\$CONFIGURE_TDF.EXE;1
VERIFY.EXE;1	VMB.EXE;1
VMB9AQ.EXE;1	VMOUNT.EXE;1
VMS\$CREATE_SYSDIRS.COM;1	VMS\$FILE_ATTRIBUTES.DAT;1
VMS\$IMAGE_VERSION.DAT;1	VMS\$INSTALL_UPG_DATA.COM;1
VMS\$OBJECTS.DAT;1	VMSHELP.EXE;1
VMSPARAMS.DAT;1	VPM.EXE;1
WP.EXE;1	WRITEBOOT.EXE;1
XFLOADER.EXE;1	

Total of 245 files.

---

## B.2.6 SYSHLP のファイル

ディレクトリ SYSHLP には、例 B-5 に示すファイルが含まれています。

### 例 B-5: SYSHLP のファイル

---

Directory SYS\$SYSDEVICE:[VMS\$COMMON.SYSHLP]

ACLED.T.HLB;1	ANALAUDIT\$HELP.HLB;1
ANLRMSHLP.HLB;1	CLUE.HLB;1
DBG\$HELP.HLB;1	DBG\$HELP.PS;1
DBG\$HELP.TXT;1	DBG\$UIHELP.HLB;1
DISKQUOTA.HLB;1	DTEHELP.HLB;1
DTSDTR.HLB;1	EDFHLP.HLB;1
EDTHELP.HLB;1	EDTVT100.DOC;1
EDTVT52.DOC;1	ESS\$LADCP.HLB;1
ESS\$LASTCPHELP.HLB;1	EVE\$HELP.HLB;1
EVE\$KEYHELP.HLB;1	EXAMPLES.DIR;1
EXCHNGHLP.HLB;1	HELPLIB.HLB;1

## 例 B-5: SYSHLP のファイル (続き)

---

```

INSTALHLP.HLB;1
LMCP$HLB.HLB;1
MAILHELP.HLB;1
MSGHLP$LIBRARY.MSGHLP$DATA;1
OPENVMSDOC_SURVEY.TXT;1
PCSI$DCLHELP.HLP;1
PHONEHELP.HLB;1
SHWCLHELP.HLB;1
SYSMANHELP.HLB;1
TFF$TFUHELP.HLB;1

UAFHELP.HLB;1
WP.HLB;1
XA_PROFILE.TXT;1

Total of 47 files.

Directory SYS$SYSDEVICE:[VMS$COMMON.SYSHLP.EXAMPLES]

ADDRIVER.MAR;1
AUDSRV_LISTENER.B32;1
BACKUSER.COM;1
BIND_MAIN.EXE;1

CBDRIVER.MAR;1
CDROM_AUDIO.EXE;1
CLU_MOUNT_DISK.COM;1
CONNECT.COM;1
DB_REQUESTER.C;1
DB_SERVER.C;1

DECDTM$EXAMPLE1.COM;1
DECDTM$EXAMPLE2.C;1
DECW.DIR;1
DISKMOUNT.H;1

DISKMOUNT_CREATE_DAT.COM;1
DOD_ERAPAT.MAR;1
DRCOPY.PRM;1
DRMAST.MAR;1
DRSLAVE.FOR;1
DTE_DF03.MAR;1
EVE$ADVANCED.TPU;1
EVE$CONSTANTS.TPU;1

EVE$SCORE.TPU;1
EVE$EDIT.TPU;1
EVE$EXTEND.TPU;1
EVE$FILE.TPU;1
EVE$HELP.TPU;1
EVE$MASTER.FILE;1

EVE$MOUSE.TPU;1
EVE$PARSER.TPU;1
EVE$SYNONYMS.TPU;1
EVE$VERSION.DAT;1
EVE$WILDCARD.TPU;1
EVE$WPS.TPU;1
GBLSECUFO.MAR;1

HASH_PASSWORD.MAR;1
LABCHNDEF.FOR;1
LABIOACQ.FOR;1
LABIOCIN.OPT;1
LABIOCOMP.COM;1
LABIOLINK.COM;1
LABIOSAMP.FOR;1
LABIOSTAT.FOR;1

LATCP$HELP.HLB;1
MACRO$DWCI.HLB;1
MNRHELP.HLB;1
NCPHELP.HLB;1
PATCHHELP.HLB;1
PCSI$MUIHELP.DECW$BOOK;1
SDA.HLB;1
SYSGEN.HLB;1
TECO.HLB;1
TPUHELP.HLB;1

VMSDOC.DIR;1
XA_PROFILE.PS;1

ADDUSER.COM;1
AUDSRV_LISTENER.MAR;1
BIND.CLD;1
BIND_READ_ME.TXT;1

CDROM_AUDIO.C;1
CLASS.C;1
CMA_STDIO.H;1
DAYLIGHT_SAVINGS.COM;1
DB_REQUESTER.MAR;1
DB_SERVER.MAR;1

DECDTM$EXAMPLE1.FOR;1
DECDTM$EXAMPLE2.COM;1
DISKMOUNT.C;1
DISKMOUNT_CHILD.C;1

DISK_DRIVER.MAR;1
DOD_ERAPAT_LNK.COM;1
DRCOPYBLD.COM;1
DRMASTER.FOR;1
DRSLV.MAR;1
DTE_DF112.MAR;1
EVE$BUILD.TPU;1
EVE$CONSTANTS.UIL;1

EVE$DECWINDOWS.TPU;1
EVE$EDT.TPU;1
EVE$EXTRAS.TPU;1
EVE$FORMAT.TPU;1
EVE$INTERNATIONALIZATION.TPU;1
EVE$MENUS.TPU;1

EVE$OPTIONS.TPU;1
EVE$SHOW.TPU;1
EVE$TERMINALS.TPU;1
EVE$WIDGETS_MOTIF.UIL;1
EVE$WINDOWS.TPU;1
FULL_DUPLEX_TERMINAL.MAR;1
GKTEST.C;1

HASH_PASSWORD_LNK.COM;1
LABIO.OPT;1
LABIOCIN.MAR;1
LABIOCOMP.FOR;1
LABIOCON.FOR;1
LABIOPEAK.FOR;1
LABIOSEC.FOR;1
LABIOSTRT.COM;1
```

## 例 B-5: SYSHLP のファイル (続き)

---

```
LABMBXDEF.FOR;1
LANETH.MAR;1
LAVC$BUILD.COM;1
LAVC$START_BUS.MAR;1
LBRDEMO.COM;1
LBRMAC.MAR;1
LOGGER.EXE;1
LPATEST.FOR;1
MAGNETIC_TAPE.MAR;1
MONITOR.COM;1
MSCPMOUNT.COM;1
PKVDRIIVER.MAR;1
PREFER.CLD;1

QKDRIVER.MAR;1
QSDRIVER.MAR;1
RECOVERY_UNIT_SERVICES_.ADA;1
RESTUSER.COM;1
RMSJNL_EXAMPLE.COB;1
RMSJNL_EXAMPLE.EXE;1

RUFEXAMPLE.ADA;1
RUFEXAMPLE.BAS;1
RUFEXAMPLE.COB;1
RUFEXAMPLE.EXE;1
RUFEXAMPLE.PAS;1
SKDRIVER.MAR;1
SUBMON.COM;1

TDRIVER.MAR;1
USING_BACKUP.DECW$BOOK;1
USING_BACKUP.TXT;1
USSLNK.COM;1
USSTSTLNK.COM;1
VME_PIOTEST.C;1
VMS$PASSWORD_POLICY.B32;1
VMS_DEPENDABILITY_CHECKLIST.PS;1

XADRIVER.MAR;1
XAMESSAGE.MAR;1
XATEST.FOR;1

Total of 160 files.

Directory SYS$SYSDEVICE:[VMS$COMMON.SYSHLP.EXAMPLES.DECW]

DECW$FONT_ALIAS_CHARTER.DAT;1
DECW$FONT_ALIAS_FILENAMES.DAT;1
DECW$FONT_ALIAS_LUCIDA.DAT;1
DECW$TRANSPORT_EXAMPLE.EXE;1
FONTS.ALIAS;1
XPORT_EXAMPLE.B32;1

DECW$FONT_ALIAS_CHARTER_100DPI.DAT;1
DECW$FONT_ALIAS_KANJI.DAT;1
DECW$FONT_ALIAS_LUCIDA_100DPI.DAT;1
DEMO_XPORT_BUILD.COM;1
XPORTEXAMPLEDEF.R32;1
XPORT_EXAMPLE_XFER.MAR;1

Total of 12 files.

Directory SYS$SYSDEVICE:[VMS$COMMON.SYSHLP.VMSDOC]

VMSDOC_GLOSSARY.TXT;1
VMSDOC_OVERVIEW.TXT;1

VMSDOC_MASTER_INDEX.TXT;1

Total of 3 files.
```

---

## B.2.7 SYSLIB のファイル

ディレクトリ SYSLIB には、例 B-6 に示すファイルが含まれています。

### 例 B-6: SYSLIB のファイル

---

```
Directory SYS$SYSDVICE:[VMS$COMMON.SYSLIB]

ACLEDIT.TPU;1
ACLEDTSHR.EXE;1
BASRTL.EXE;1
BLAS1RTL.EXE;1
CDDSHR.EXE;1

CMA$LIB_SHR.EXE;1
CMA$OPEN_RTL.EXE;1
CMA$TIS_SHR.EXE;1
CMA_CONFIG.H;1
CMA_LIBRARY.H;1
CMA_TIS.H;1

CONVSHR.EXE;1
CXXL$011_SHR.EXE;1
DBG$HA_KERNEL.EXE;1
DBGSSISHR.EXE;1
DCLTABLES.EXE;1
DEBUG.EXE;1
DEBUGISHR.EXE;1
DECC$SHR.EXE;1

DECCRTL.OLB;1
DECW$DRIVER.MLB;1
DECW$FONTCOMPILER.CLD;1
DECW$SECURITY.EXE;1
DECW$SERVER_DDX_GA.EXE;1
DECW$SERVER_DDX_GC.EXE;1
DECW$SERVER_DDX_GF.EXE;1

DECW$SESSIONSHRP.EXE;1
DECW$SVEXT_D2DX_EXTENSIONS.EXE;1
DECW$SVEXT_MULTI_BUFFERING.EXE;1
DECW$SVEXT_X3D_PEX_GB.EXE;1
DECW$SVEXT_X3D_PEX_GE.EXE;1
DECW$SVEXT_X3D_PEX_STP_UCODE.EXE;1
DECW$SVEXT_XIE.EXE;1
DECW$TRANSPORT_DECNET.EXE;1

DECW$TRANSPORT_LOCAL.EXE;1
DECW$XLIBSHR.EXE;2
DECW$XPORTCOM.MAR;1
DECW$XPORTDEF.H;1
DECW$XPORTDEF.R32;1
DECW$XPORTMSG.R32;1

DELTA.OBJ;1
DNS$CLIENT.EXE;1
DNS$SHARE.EXE;1
DNSDEF.FOR;1
DNSDEF.MAR;1
DNSDEF.PLI;1
DNSMSG.BAS;1

DNSMSG.H;1
DNSMSG.PAS;1
DNSMSG.R32;1
DTE_DF112.EXE;1
DTI$SHARE.EXE;1
DYN SWITCH.EXE;1
ENCRYP SHR.EXE;1

ACLEDT$SECTION.TPU$SECTION;1
ADARTL.EXE;1
BASRTL2.EXE;1
CDA$ACCESS.EXE;2
CLIMAC.REQ;1

CMA$OPEN_LIB_SHR.EXE;1
CMA$RTL.EXE;1
CMA.H;1
CMA_CRTLX.H;1
CMA_PX.H;1
COBRTL.EXE;1

CRF$SHR.EXE;1
DBG$HA.UID;1
DBG$HA_MAIN.EXE;1
DBLRTL.EXE;1
DCXSHR.EXE;1
DEBUGSHR.EXE;1
DECC$EMPTY.EXE;1
DECCCURSE.OLB;1

DECCRTL.G.OLB;1
DECW$DWTLIBSHR.EXE;2
DECW$PRINTWGTSHR.EXE;2
DECW$SECURITY_VMS.EXE;1
DECW$SERVER_DDX_GB.EXE;1
DECW$SERVER_DDX_GE.EXE;1
DECW$SERVER_DIX.EXE;1

DECW$SVEXT_ADOBE_DPS_EXTENSION.EXE;1
DECW$SVEXT_DEC_XTRAP.EXE;1
DECW$SVEXT_X3D_PEX.EXE;1
DECW$SVEXT_X3D_PEX_GB_UCODE.EXE;1
DECW$SVEXT_X3D_PEX_STP.EXE;1
DECW$SVEXT_X3D_PEX_VCFB.EXE;1
DECW$TRANSPORT_COMMON.EXE;1
DECW$TRANSPORT_LAT.EXE;1

DECW$TRANSPORT_TCPIP.EXE;1
DECW$XPORTCOM.H;1
DECW$XPORTCOM.R32;1
DECW$XPORTDEF.MAR;1
DECW$XPORTMAC.R32;1
DELTA.EXE;1

DISMNTSHR.EXE;1
DNS$RTL.EXE;1
DNSDEF.BAS;1
DNSDEF.H;1
DNSDEF.PAS;1
DNSDEF.R32;1
DNSMSG.FOR;1

DNSMSG.MAR;1
DNSMSG.PLI;1
DTE_DF03.EXE;1
DTE_DMCL.EXE;1
DTKSHR.EXE;1
EDTSHR.EXE;1
EPC$FACILITY.TLB;1
```



## 例 B-6: SYSLIB のファイル (続き)

---

```
EPC$SHR.EXE;1
ERFCOMMON.EXE;1
ERFLIB.TLB;1
ERFSHR2.EXE;1
EVE$WIDGETS_MOTIF.UID;1
EXC_HANDLING.H;1
FORDEF.FOR;1
FORRTL.EXE;1
IMAGELIB.OLB;1

INIT$SHR.EXE;1
LAT$SHR.EXE;1
LIB.MLB;1
LIBDEF.FOR;1
LIBRTL2.EXE;1
MAILSHR.EXE;1
MMEDEF.H;1
MMESHR.EXE;1

MSGHLP$ENGLISH.EXE;1
MTHDEF.FOR;1
NC$LIBRARY.NLB;1
NISCS_LAA.EXE;1
NMLSHR.EXE;1
PCSI$MOTIFSHR.EXE;1

PLIRTL.EXE;1
PTD$SERVICES_SHR.EXE;1
PTHREAD_EXC.H;1
SCNRTL.EXE;1
SDATP$SHARE.EXE;1
SECURESHR.EXE;1
SIGDEF.FOR;1
SMGSHR.EXE;1
SMI$SHR.EXE;1

SORTSHR.EXE;1
STARLET.MLB;1
STARLET.REQ;1
SUMSHR.EXE;1
TECOSHR.EXE;1
TPAMAC.REQ;1

TPU$DEBUG.TPU;1
TPU.DAT;1
TRACE.EXE;1
UTIL$SHARE.EXE;1
VAXC$EMPTY.EXE;1
VAXC2DECC.EXE;1
VAXCG2DECC.EXE;1

VAXCRTL.OLB;1
VAXCRTLG.OLB;1
VECTOR_EMULATOR.EXE;1
VMESUPPORT.MLB;1
VMS$PASSWORD_DICTIONARY.DAT;1
VMSDEBUGUIL.UID;1
VMTHRTL.EXE;1
XDPS$DPSCLIENTSHR.EXE;2
XDPS$MASTERDPSVM.DAT;1
XNL$SHR.EXE;2

Total of 217 files.
```

```
EPMS$SRVSHR.EXE;1
ERFCTL$SHR.EXE;1
ERFSHR.EXE;1
EVE$SECTION.TPU$SECTION;1
EVE.DAT;1
FDLSHR.EXE;1
FORIOSDEF.FOR;1
FORRTL2.EXE;1
IMGDMP.EXE;1

IPC$SHARE.EXE;1
LBRSHR.EXE;1
LIB.REQ;1
LIBRTL.EXE;1
LIBRTL_INSTRUMENTED.EXE;1
MAILSHRP.EXE;1
MMMSG.H;1
MOUNTSHR.EXE;1

MSGHLP$SHARE.EXE;1
MTHRTL.EXE;1
NCSSHR.EXE;1
NISCS_LOAD.EXE;1
PASRTL.EXE;1
PCSI$SHR.EXE;1

PPLRTL.EXE;1
PTHREAD.H;1
RPGRTL.EXE;1
SCRSHR.EXE;1
SDA_EXTEND_VECTOR.EXE;1
SECURESHRP.EXE;1
SMB$SRVSHR.EXE;1
SMI$OBJSHR.EXE;1
SNAPSHOT$SHARE.EXE;1

SPISHR.EXE;1
STARLET.OLB;1
STARLETSD.TLB;1
TC$LIBRARY.OLB;1
TFFSHR.EXE;1
TPU$CCTSHR.EXE;1

TPU$MOTIFSHR.EXE;1
TPUSHR.EXE;1
UISSHR.EXE;1
UVMTHRTL.EXE;1
VAXC$LCL.OPT;1
VAXCCURSE.OLB;1
VAXCRTL.EXE;1

VAXCRTLG.EXE;1
VBLAS1RTL.EXE;1
VME$LIBRARY.OLB;1
VMS$FORMAT_AUDIT_SYSTEM.EXE;1
VMSDEBUGCUSTUIL.UID;1
VMSRTL.EXE;1
XDPS$DPSBINDINGSSHR.EXE;2
XDPS$DPSLIBSHR.EXE;2
XFDEF.FOR;1
```

## B.2.8 SYSMGR のファイル

ディレクトリ SYSMGR には、例 B-7 に示すファイルが含まれています。

### 例 B-7: SYSMGR のファイル

---

```
Directory SYS$SYSDEVICE:[VMS$COMMON.SYSMGR]

AGEN$NEW_NODE_DEFAULTS.DAT;1
AGEN$NEW_SATELLITE_DEFAULTS.DAT;1
AGENPARAMS.EXE;1
CLUSTER_CONFIG.COM;1
DECW$DEVICE.COM;1
DECW$DEVICE_GF.COM;1

DECW$PRIVATE_SERVER_SETUP.TEMPLATE;1
DECW$STARTSERVER.COM;1
DNS$CHANGE_DEF_FILE.COM;1
DNS$CLIENT_STOP.COM;1
LAT$SYSTARTUP.COM;1
LIB$DT_STARTUP.COM;1
LOGIN.COM;1
LPA11STRT.COM;1

MAKEROOT.COM;1
RTTLOAD.COM;1
SMISERVER.COM;1
SNAPSHOT$NEW_DISK.COM;1
SNAPSHOT$SYSHUTDOWN.TEMPLATE;1
STARTNET.COM;1
SYCONFIG.TEMPLATE;1

SYLOGICALS.TEMPLATE;1
SYLOGIN.TEMPLATE;1
SYPAGSWPFILES.TEMPLATE;1
SYSECURITY.TEMPLATE;1
SYSHUTDOWN.TEMPLATE;1
SYSTARTUP_VMS.COM;1
TFF$STARTUP.COM;1
VMS$AUDIT_SERVER.DAT;1
VMSIMAGES.DAT;1
WELCOME.TXT;1

AGEN$NEW_NODE_DEFAULTS.TEMPLATE;1
AGEN$NEW_SATELLITE_DEFAULTS.TEMPLATE;1
ALFMAINT.COM;1
DBLSTRUP.COM;1
DECW$DEVICE_GE.COM;1
DECW$DEVICE_GG.COM;1

DECW$RGB.DAT;1
DECW$STARTXTERMINAL.COM;1
DNS$CLIENT_STARTUP.COM;1
EDTINI.TEMPLATE;1
LAT$SYSTARTUP.TEMPLATE;1
LOADNET.COM;1
LOGIN.TEMPLATE;1
LTLOAD.COM;1

NETCONFIG.COM;1
SECURITY.AUDIT$JOURNAL;1
SNAPSHOT$CLEANUP.COM;1
SNAPSHOT$SYCLEANUP.TEMPLATE;1
SNAPSHOT.COM;1
SYCONFIG.COM;1
SYLOGICALS.COM;1

SYLOGIN.COM;1
SYPAGSWPFILES.COM;1
SYSECURITY.COM;1
SYSHUTDOWN.COM;1
SYSTARTUP_V5.COM;1
SYSTARTUP_VMS.TEMPLATE;1
UTC$CONFIGURE_TDF.COM;1
VMS$IMAGES_MASTER.DAT;1
WELCOME.TEMPLATE;1

Total of 61 files.
```

---

## B.2.9 SYSMGR のファイル

ディレクトリ SYSMGR には、例 B-8 に示すファイルが含まれています。

### 例 B-8: SYSMGR のファイル

---

```
Directory SYS$SYSDEVICE:[VMS$COMMON.SYSMSG]

ADAMSG.EXE;1
CXXL$MSG_SHR.EXE;1
DBLRTLMSG.EXE;1

DNS$MSG.EXE;1
FILMNTMSG.EXE;1
LMF_MESSAGE.EXE;1
PASMSG.EXE;1

CLIUTLMSG.EXE;1
DBGTBKMSG.EXE;1
DECW$TRANSPORTMSG.EXE;1

EPC$MSG.EXE;1
LMCP$MSG.EXE;1
NETWRKMSG.EXE;1
PLIMSG.EXE;1
```

### 例 B-8: SYSMMSG のファイル (続き)

---

PPLMSG.EXE;1	PRGDEVMSG.EXE;1
RPGMSG.EXE;1	SCNMSG.EXE;1
SHRIMGMSG.EXE;1	SORTMSG.EXE;1
SYSMGTMSG.EXE;1	SYSMMSG.EXE;1
TECOMSG.EXE;1	TPUMSG.EXE;1
VAXCMSG.EXE;1	VMSINSTAL_LANGUAGE.COM;1
VMSLICENSE_LANGUAGE.COM;1	VVIEFMSG.EXE;1

Total of 28 files.

---

## B.2.10 SYSTEST のファイル

ディレクトリ SYSTEST には、例 B-9 に示すファイルが含まれています。

### 例 B-9: SYSTEST のファイル

---

Directory SYS\$SYSDEVICE: [VMS\$COMMON.SYSTEST]

DECDTM_IVP.EXE;1	TCNTRL.CLD;1
UETCDRO00.EXE;1	UETCLIG00.COM;1
UETCLIG00.DAT;1	UETCLIG00.EXE;1
UETCOMS00.EXE;1	UETDISK00.EXE;1
UETDMPF00.EXE;1	UETDNET00.COM;1
UETDNET00.DAT;1	UETDRIW00.EXE;1
UETDR7800.EXE;1	UETFORT01.DAT;1
UETFORT01.EXE;1	UETFORT02.EXE;1
UETFORT03.EXE;1	UETINIT00.EXE;1
UETINIT01.EXE;1	UETLOAD00.DAT;1
UETLOAD02.COM;1	UETLOAD03.COM;1
UETLOAD04.COM;1	UETLOAD05.COM;1
UETLOAD06.COM;1	UETLOAD07.COM;1
UETLOAD08.COM;1	UETLOAD09.COM;1
UETLOAD10.COM;1	UETLOAD11.COM;1
UETLPAK00.EXE;1	UETMA7800.EXE;1
UETMEMY01.EXE;1	UETNETS00.EXE;1
UETP.COM;1	UETPHAS00.EXE;1
UETR SXFOR.EXE;1	UETSUPDEV.DAT;1
UETTAPPE00.COM;1	UETTAPPE00.EXE;1
UETTTYS00.EXE;1	UETUNAS00.EXE;1
UETVECTOR.COM;1	UETVECTOR.EXE;1

Total of 44 files.

---

## B.2.11 SYSUPD のファイル

ディレクトリ SYSUPD には、例 B-10 に示すファイルが含まれています。

### 例 B-10: SYSUPD のファイル

---

```
Directory SYS$SYSDEVICE:[VMS$COMMON.SYSUPD]

AUTOGEN.COM;1
CONSCOPY.COM;1
DECW$KITBLD.DAT;1
DECW$MKFONTDIR.COM;1

DECW$OBSOLETE.IDX;1
DECW$TAILOR_ON.TEMPLATE;1
INSTALLED_PRDS.COM;1
NETCONFIG_UPDATE.COM;1
PCSI$CREATE_NETWORK_OBJECT.COM;1
PCSI$DELETE_ACCOUNT.COM;1
PCSI$DELETE_RIGHTS_IDENTIFIER.COM;1

REGISTER_PRIVILEGED_IMAGE.COM;1
SPKITBLD.COM;1
STABACKIT.COM;1
TAILOR_ON.TEMPLATE;1
VMS$ROLLING_UPGRADE.COM;1
VMSINSTAL.COM;1
VMSKITBLD.COM;1
VMSKITBLD.IDX;1
VMS$TAILOR.EXE;1
VMS_VERSION_OVERRIDE.DAT;1
VVIEF$INSTAL.COM;1

BOOTUPD.COM;1
CREATE_IDX.EXE;1
DECW$KITBLD.IDX;1
DECW$OBSOLETE.DAT;1

DECW$TAILOR.EXE;1
DXCOPY.COM;1
LIBDECOMP.COM;1
PCSI$CREATE_ACCOUNT.COM;1
PCSI$CREATE_RIGHTS_IDENTIFIER.COM;1
PCSI$DELETE_NETWORK_OBJECT.COM;1
PCSI$REGISTER_PRODUCT.COM;1

SETDEFBOO.COM;1
STABACKIT-TABLE.DAT;1
SWAPFILES.COM;1
UPDATE_CONSOLE.COM;1
VMS$SYSTEM_IMAGES.COM;1
VMSINSTAL_LMFGROUPS.COM;1
VMSKITBLD.DAT;1
VMSLICENSE.COM;1
VMSUPDATE.COM;1
VVIEF$DEINSTAL.COM;1

Total of 43 files.
```

---

## B.2.12 VUE\$LIBRARY のファイル

ディレクトリ VUE\$LIBRARY には、例 B-11 に示すファイルが含まれています。

### 例 B-11: VUE\$LIBRARY のファイル

---

```
Directory SYS$SYSDEVICE:[VMS$COMMON.VUE$LIBRARY]

SYSTEM.DIR;1                                [SYSTEM]      (RWE,RWE,RE,RE)
USER.DIR;1                                  [SYSTEM]      (RWE,RWE,RE,RE)

Total of 2 files.

Directory SYS$SYSDEVICE:[VMS$COMMON.VUE$LIBRARY.SYSTEM]

MACRO$DWCI.EXE;1                             [SYSTEM]      (RWED,RWED,RWED,RE)
MACRO$DWCI.UID;1                             [SYSTEM]      (RWED,RWED,RWED,RE)

Total of 2 files.

Grand total of 35 directories, 2055 files.
```

---

## C2 環境における OpenVMS システムの運用

この付録では、C2 環境で OpenVMS オペレーティング・システムを運用する方法について説明します。C2 とは、オペレーティング・システムのセキュリティについて、米国政府が定めた格付けのことで、OpenVMS VAX および OpenVMS Alpha は、Division C のクラス 2 システムの基準を満たすオペレーティング・システムとされています (C.1.1 項参照)。この付録で使用する用語は、米国政府の評価基準で使用されている用語に由来します。

National Computer Security Center (NCSC) により評価が行われた OpenVMS のバージョンは、「Evaluated Products List」に掲載されています。このリストは以下から入手できます。

National Computer Security Center  
9800 Savage Road  
Fort George G. Meade  
Maryland 20755-6000

同じ情報が、下記の Web サイトでも提供されています。

<http://www.radium.NCSC.mil/tpep/epl/index.HTML>

OpenVMS VAX Version 6.1 および OpenVMS AXP Version 6.1 が提供するセキュリティ保護は、National Computer Security Center (NCSC) が、「Department of Defense Trusted Computer System Evaluation Criteria」(1985年 12月発行)に規定されている要件に照らして評価しています。OpenVMS VAX Version 6.1 および OpenVMS AXP Version 6.1 は、C2 の格付けを獲得しています。

### C.1 C2 システムについて

この節では、C2 システムの要件を紹介し、この要件を満たすシステムをサポートするために OpenVMS 製品について提供しているドキュメントについて説明します。

#### C.1.1 C2 環境の定義

C2 環境とは、信頼できるコンピュータ・システムに関する米国国防総省の基準を満たしている環境であり、OpenVMS オペレーティング・システムについての政府の評価の対象となった TCB (トラステッド・コンピューティング・ベース) のハードウェア・コンポーネントとソフトウェア・コンポーネントのみで構成されている環境のことを指します。

C2 システムの基準は、Department of Defense Computer Security Center が発行している『*Department of Defense Trusted Computer System Evaluation Criteria*』(DOD 5200.28-STD) に定義されています。定義されている主な内容は次のとおりです。

- アクセス制御。個々のユーザに加え、複数のユーザで構成されるグループの識別が可能
- ユーザを明確に特定するログイン手順を通じたユーザ確認
- セキュリティ関連イベントの監査
- 再割り当ての前にオブジェクトが消去されるようにすることを目的とした、資源の分離

#### C.1.1.1 ドキュメント

トラステッド・ファシリティ・マニュアルは、システム管理者を対象としていません。C2 トラステッド・ファシリティ・マニュアルには次のものが含まれます。

- このマニュアルの第 5 章～第 13 章および付録
- 『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』の「監査分析ユーティリティ」の節
- 『*OpenVMS AXP Version 6.1 Upgrade and Installation Manual*』
- 『*OpenVMS VAX Version 6.1 Upgrade and Installation Manual*』
- 『*OpenVMS AXP Version 6.1 Release Notes*』
- 『*OpenVMS AXP Version 6.1 Release Notes Addendum*』
- 『*OpenVMS VAX Version 6.1 Release Notes*』
- 『*OpenVMS VAX Version 6.1 Release Notes Addendum*』

本書の Part 1「セキュリティの概要」および Part 2「一般ユーザのためのセキュリティ」は、セキュリティ機能に関するユーザ向けの手引きとなっており、すべてのユーザが参照できるようにする必要があります。

## C.2 C2 システム向けのトラステッド・コンピューティング・ベース (TCB)

米連邦政府によるコンピュータ・システムの評価は、トラステッド・コンピューティング・ベース (TCB) を C.1.1 項に要約されている基準に照らして行われます。TCB は、セキュリティ・ポリシーを実施する、コンピュータ・ハードウェアとオペレーティング・システムの組み合わせのことです。

## C.2.1 TCB に含まれるハードウェア

VAX プロセッサのアーキテクチャは設計上、競合するプログラム同士がお互いのデータに干渉できないようになっています。VAX ハードウェアでは、あるプログラムが別のプログラムのメモリに干渉できないようになっています。

本書に記載されているセキュリティ機能は、評価されたハードウェア構成に含まれるすべての VAX プロセッサ、およびサポートされているすべてのマス・ストレージと通信デバイスに適用されます。『*Final Evaluation Report, Digital Equipment Corporation, OpenVMS VAX and SEVMS Version 6.1*』には、評価済みハードウェアがすべて掲載されています。

## C.2.2 TCB に含まれるソフトウェア

OpenVMS オペレーティング・システムでは、TCB は OpenVMS のほとんどを網羅しています。OpenVMS の TCB には、エグゼクティブやファイル・システムの全体、ユーザ・モードで実行されないその他すべてのシステム・コンポーネント（デバイス・ドライバ、RMS、DCL など）、特権を使ってインストールされた大部分のシステム・プログラム、およびシステム管理者が TCB に関連するデータを保守するために使用するその他の各種ユーティリティが含まれます。

ユーザの便宜を図るため、OpenVMS は、オペレーティング・システム本体以外のものと一緒に出荷されます。OpenVMS ソフトウェア・パッケージには、セーブ・セットや、OpenVMS オペレーティング・システムでよく実行されるレイヤード・プロダクト向けの有用なイメージなどが含まれています。ただし、C2 システムとして評価済みなのはベースとなる OpenVMS オペレーティング・システムのみです。DECwindows ソフトウェアやDisplay PostScript® のサポートなど、レイヤード・プロダクトは C2 評価の対象外です。このため、C2 格付けの対象は、表 C-1 に示すソフトウェアを実行する OpenVMS VAX システムまでは及びません。これらのソフトウェア・コンポーネントが対象外であるということが、これらが安全でないということではまったくありません。単に、評価の対象となったシステムには含まれていなかったということです。このようなソフトウェアを導入した場合、ベース・システムは、その使用について認可を得る必要があります。

表 C-1: C2 の評価済みシステムに含まれていないソフトウェア

ソフトウェア	機能	説明
DECwindows ソフトウェア	ウィンドウ化インタフェース	DECwindows は、レイヤード・プロダクトです。DECwindows は C2 要件を満たすよう設計されていますが、評価はまだ行われていません。

表 C-1: C2 の評価済みシステムに含まれていないソフトウェア (続き)

ソフトウェア	機能	説明
DECdns 分散ネーム・サービス	クライアント・サポート	DECdns ソフトウェアは、レイヤード・プロダクトであるサーバ・ソフトウェアを必要とします。クラスタは、DECdns に関係なく DECnet 接続が可能です。
HP DECamsd ソフトウェア	監視および診断	HP DECamsd ソフトウェアは、評価済み構成の範囲外です。
LASTport プロトコルおよび LASTport/DISK プロトコル	プロトコル・サポート	クラスタ・システムのシステム安全性の範囲外にある HP の Infoserver プロダクトは、これらのプロトコルに依存します。
LAT プロトコル	プロトコル・サポート	LAT プロトコルは、評価済みの構成の範囲外にある、DECserver ターミナル・サーバへの接続に使用されます。
DECnet/OSI フルネーム	プロトコル・サポート	OpenVMS オペレーティング・システム内の DECnet/OSI (Phase V) ノード名の使用をサポートします。この機能の使用は、C2 評価済み構成には含まれていません。
HSM (Hierarchial Shelving Manager)	ストレージ・サポート	File Shelving は、レイヤード・プロダクトです。File Shelving ファシリティ (HSM) の使用は、C2 評価済み構成ではサポートされていません。
MME (Media Management Extension)	クライアント・サポート	Media Management Extension (MME) では、ストレージ・メディア・プログラムを使用できます。メディア管理の使用は、C2 評価済み構成には含まれていません。
OpenVMS Management Station		OpenVMS Management Station は、OpenVMS 向けの PC ベースのシステム管理ツールです。OpenVMS Management Station は、C2 評価済み構成ではまだ検証されていません。
アクセス制御文字列	遠隔ノード上のファイルへのアクセス	評価済み構成では、アクセス制御文字列ではなく、代理アカウントを使用すべきです。



### C.2.3 オブジェクトの保護

OpenVMS オペレーティング・システムは、情報を格納しているオブジェクトへのアクセスを制御します。保護オブジェクトには、ODS-2 や ODS-5 のディスク・ファイル、コモン・イベント・フラグ・クラスタ、デバイス、グループやシステムのすべてのグローバル・セクション、論理名テーブル、キュー、資源ドメイン、ODS-2 や ODS-5 のディスク・ボリュームなどがあります。ケーパビリティ・オブジェクトおよびセキュリティ・クラス・オブジェクトでは、任意アクセス保護をフルに利用できますが、これらは C2 評価基準に基づくオブジェクトではありません。

第 4 章 および第 5 章 では、オブジェクトの保護と、すべての新規オブジェクトに UIC、保護コード、および場合によっては ACL が設定されるようにするためにオペレーティング・システムに備わっているテンプレート・プロファイルについて説明しています。4.4.7 項「ファイルの継承方式の設定」、4.5.6 項「ディレクトリ構造に対するデフォルトの保護コードの提供」、および 8.8 節、では特に、新規作成オブジェクトに対するデフォルトの保護を設定する方法について説明しています。

グローバル・セクションおよびメールボックス・オブジェクトに割り当てられているデフォルトの保護は、他のオブジェクトに割り当てられているものと比較すると緩やかです。これは、一部のソフトウェア製品では、メールボックス・オブジェクトとグローバル・セクション・オブジェクトがデフォルトでは制限が他のオブジェクトよりも緩い保護設定で作成されていることを前提としているためです。これらのオブジェクト用のテンプレート・プロファイルを変更して、保護設定を厳しくすることは可能ですが、ソフトウェア製品によっては悪影響の出る場合もあります。

デフォルトの保護を変更するには、変更対象のオブジェクトと任意の既存オブジェクトの両方のテンプレート・プロファイルを変更する必要があります。たとえば、次のコマンドを実行すると、デバイス・クラスの MAILBOX テンプレートを変更できます。

```
$ SET SECURITY/CLASS=SECURITY_CLASS/PROFILE=TEMPLATE=MAILBOX -  
_$/PROTECTION=(S:RWPL,O:RWPL,G,W) DEVICE
```

オペレーティング・システムは、この値をすべての新規メールボックスに適用するようになります。ただし、既存の各メールボックスの保護も、SET SECURITY コマンドを使用して、厳しい設定にする必要があります。次に例を示します。

```
$ SET SECURITY/CLASS=DEVICE -  
_$/PROTECTION=(S:RWPL,O:RWPL,G,W) mailbox_name
```

セキュリティ・テンプレートに指定されているデフォルトのオブジェクト保護設定は、システムをシャットダウンまたはリブートをまたいで保持されます。そのため、システムを自動的にリブートすることにより、リブート後に作成されたオブジェクトはすべて、新しいデフォルト保護が設定された状態で作成されるようになります (オブジェクト作成者が別の保護を指定した場合を除く)。

## C.2.4 TCB の保護

OpenVMS TCB を構成するコードとデータは、ファイルに格納され、一部は稼働中のオペレーティング・システムのアドレス空間にあります。これらのコードやデータは、ファイル・アクセス制御やメモリ・ページ保護によって保護されます。メモリ・ページの保護は、オペレーティング・システムが実行時に設定するため、通常、システム管理者は気にする必要はありません。

### C.2.4.1 ファイルの保護

TCB を構成するファイルは、オペレーティング・システムをインストールすると正しく保護されます。ただし、十分な特権を持っているユーザであれば、保護設定を変更することが可能です。このマニュアルの付録 B では、オペレーティング・システムのファイルの正しいファイル保護について説明しています。

OpenVMS オペレーティング・システムをインストールするときは、サイト固有のファイルを除き、システム・ファイルに変更を加えることを避けます。これは、ベース・オペレーティング・システムのセキュリティを維持する上で重要です。

### C.2.4.2 信頼できるユーザに付与する特権

一部の特権は、その保持者が通常のファイル・アクセス制御やメモリ・アクセス制御を直接または間接的に回避することを可能にします。そのため、このような特権は、システム管理者やセキュリティ管理者、または信頼できるユーザ以外の人物には付与しないようにします。Objects, All, System, および Group の 4 つのカテゴリの特権は、信頼できるユーザに対してのみ適しています。それぞれのカテゴリにどの特権が属しているかについては、表 8-2 を参照してください。個々の特権については、付録 A で詳しく説明しています。

Objects カテゴリおよび All カテゴリに属する特権の保持者は、信頼性の低いユーザからの TCB の隔離を無効にすることが可能です。System カテゴリの特権の保持者は、通常のシステム運用を妨げ、サービスの拒否を引き起こすことができますが、オブジェクト・アクセス制御に違反することはできません。System カテゴリの一部の特権では、結果的にアクセス制御の適用を回避することが可能です。

Group カテゴリの特権の保持者は、同じグループ内の他のユーザの操作を妨げることが可能です。特に、GRPPRV 特権の保持者は、保持者のグループ内での通常のアクセス制御に違反することが可能です。これは、GRPPRV 特権によって、同じグループ UIC を有するサブジェクトが所有するオブジェクトへのアクセスが（保護コードのシステム・フィールドによって）許可されるためです。

信頼できるユーザは全員、自分が実行する操作のあらゆる影響を認識している必要があります。特に、信頼できるユーザは、ある操作において使用される可能性のあるすべてのソフトウェア製品について把握している必要があります。信頼できる

ユーザの特権では、本来なら OpenVMS のセキュリティ・ポリシーによって禁止される操作を、信頼の低いソフトウェアが実行できてしまうからです。

#### C.2.4.3 信頼の低いユーザに付与する特権

信頼の低いユーザは、Normal カテゴリおよび Devour カテゴリに属する特権を保持できます (GRPNAM は除く)。ただし、Devour カテゴリに属する特権を付与するときは注意が必要です。このカテゴリの特権の保持者は、無制限に資源を消費することが可能なため、サービス拒否を引き起こしたり、システム上の他のユーザの操作を妨げる可能性があります。表 C-2 に、信頼の低いユーザに付与される特権です。

表 C-2: 信頼の低いユーザに付与する特権

カテゴリ	特権	許可されている操作
Normal	NETMBX TMPMBX	ネットワーク接続の作成、一時メールボックスの作成
Devour	ACNT ALLSPOOL BUGCHK EXQUOTA PRMCEB PRMGBL PRMMBX SHMEM	会計情報管理の無効化、スプールされたデバイスの割り当て、バグチェック・エラー・ログ・エントリの作成、ディスク制限の超過、パーマメント・コモン・イベント・フラグ・クラスタの作成/削除、パーマメント・グローバル・セクションの作成、パーマメント・メールボックスの作成、共用メモリ内の構造の作成/削除

#### C.2.4.4 物理的セキュリティ

物理的セキュリティおよび環境面のセキュリティは、システムの安全な運用にとって非常に重要です。TCB のすべての物理コンポーネントには、十分な保護を設定する必要があります。保護が不十分な場合、不正ユーザによってシステムのセキュリティが脅かされる可能性があるからです。以下のように、TCB のセキュリティを脅かす可能性のある運用方法や機能は、C2環境では使用しないでください。

- 公共の場所にはコンソール・ターミナルを置かないようにします。コンソール・ターミナルは必ず物理的なセキュリティを確保する必要があります。コンソール・ターミナルが、CPU、ひいてはシステムの運用を制御するからです。
- コンソールにパスワード機能が備わっている場合に、コンソール・パスワードを無効なままにしておかないようにします。(コンソール・パスワード機能は、一部の VAXstation 3100、それ以降のほとんどのモデル、および評価済みの Alpha モデルに備わっています。) コンソール・パスワードにより、不正な人物がコマンドを使用して別のメディアからブートしたり、会話型ブートを実行したりするのを防止できるほか、メモリの変更を防ぐことができます。

- モデムの使用は許可しないようにします。モデムは、信頼できるシステムへの入口となり、システム・セキュリティが脅かされる可能性が非常に高くなります。
- 遠隔診断機能を有効のままにしておかないようにします。遠隔診断機能も、信頼できるシステムへの入口となります。診断機能のスイッチをオフの位置に切り換えることで、遠隔診断機能を無効にします。
- 認証カードの使用は許可しないようにします。これらのデバイスは、C2 の評価済み構成ではサポートされていません。
- クラスタの通信メディアに物理的にアクセスできないようにします。侵入者は、プロセッサやケーブルに物理的にアクセスすることができれば、システムに侵入することが可能です。

オペレーティング・システムは、デフォルトで、すべての通信インタフェースをワールド・アクセスから保護します。対象となる通信インタフェースには、CI や LAN デバイス (Ethernet や DSSI, FDDI, SCSI など) などがあります。CI インタフェースは、CI クラスタのメンバ間における信頼できるインタフェースであり、非特権ユーザはアクセスできません。非特権ユーザには、LAN デバイスへのアクセス権を付与しないでください。

- 信頼の低いユーザに HSC コンソールへのアクセスを許可しないようにします。HSC コンソールは、許可されている人だけが使用できる場所に置きます。そうすることでディスク・ボリュームのバックアップや復元など、慎重さが要求される操作を、信頼の低いユーザが実行しないようにします。
- ユーザが、他のユーザのプリンタ出力を読むことができないようにします。ユーザが自分のデータにのみアクセスできるようにプリンタの出力を保護します。
- 権限のないユーザのいる場所に、ディスク、テープ、CD などのストレージ・メディアを放置しないようにします。メディアを入手したユーザが、その内容を見たり、改ざんしたりする恐れがあります。

### C.2.5 C2 システムの設定

この節では、OpenVMS の機能を使用する上で課される C2 に関わる制限を説明します。次のトピックについて説明します。

- ユーザを確実に特定できることを保証するための要件
- 監査ログ・ファイルの適切な管理
- ターミナル、ボリューム、プリンタの適切な使用
- クラスタの要件
- システム・パラメータに関する必要な設定
- システム操作から除外されるコマンドとソフトウェア

### C.2.5.1 ユーザを確実に特定できることの保証

名前、UIC、およびパスワードを適切に使用することで、OpenVMS オペレーティング・システムが個々のユーザを確実に特定できることが保証されます。基本的な運用手順として、特権アカウントについては自動生成されたパスワードの使用を推奨します。以下のように、個別ユーザの特定が不確実になるような運用方法と機能は、C2環境では避けてください。

- 同じ UIC を複数のユーザに割り当てないようにします。UIC は、ユニバーサルな内部ユーザ識別子として使用されます。そのため、すべてのユーザに個別の UIC を割り当てなければなりません。
- パスワードのないオープン・アカウントは許可しないようにします。アカウントにパスワードが設定されていないと、その存在を知るすべてのユーザがそのアカウントを利用できてしまいます。システム管理者は、登録ユーティリティ (AUTHORIZE) でパスワードなしのアカウントを設定しないようにして、すべてのアカウントに 1 文字以上のパスワードを設定することによって、オープン・アカウントの存在を防止できます。
- グループ・アカウントは許可しないようにします。1 つのアカウントを複数ユーザで共用すると、個別ユーザの特定が不確実になります。ユーザごとに一意のアカウントを与える必要があります。
- ゲスト・アカウントを許可しないようにします。複数のユーザが共通のアカウントを利用して、システム上の資源にアクセスできるようになってしまうためです。ゲスト・アカウントが必要となる場面の大半は、特別な代理ログイン・アカウントを用意することで対処できます。
- 自動ログインを有効にしないようにします。自動ログイン機能 (ALF) は、特定のユーザではなく特定のターミナルにアカウントを関連付けます。そのため、個別ユーザの特定が不確実になります。
- グループ用のネットワーク代理アカウントを開始しないようにします。個別ユーザを確実に特定できるようにするためには、ネットワークの各ユーザには、そのユーザがアクセスするそれぞれのノードに一意のネットワーク代理アカウントを用意しなければなりません。該当するすべてのノードに同じユーザ名と UIC を割り当て、対応するアカウントの中で個別に代理を設定します。
- 代理アカウントに特権アクセスを付与しないようにします。
- ライト・データベースにおいて、DBG\$ENABLE\_SERVER 識別子を付与しないようにします (デバッグ・サーバの実行に必要な場合を除く)。
- オペレータによる HSC 操作をビデオ・ターミナルに記録しないようにします。オペレータの操作の記録にはハードコピー・プリンタを使用します。それにより、特定のシステム操作をその操作を実行した特定の人物に関連付けることができます。

- 評価済み構成におけるアクセス制御文字列の使用に課される制限について、確実にユーザに周知します。(SFUG の 3 ~ 15 ページ参照) 具体的には、評価済み構成において、アクセス制御文字列の使用は許可されません。評価済み構成では、代理ログイン・アカウントを使用します。
- オペレータがオペレータ・ログにサイン・インせずに HSC コンソールでいかなる作業も行うことがないようにします。サインイン・ログは、HSC コンソール操作を実行したユーザと実行した日時を追跡するために必要です。ログは、ハードコピー出力とともに、HSC 操作の記録として使用できます。

#### C.2.5.2 監査証跡の管理

セキュリティ監査システムでは、適切に管理を行えば、システム上のセキュリティに関わるか活動を追跡することができます。監査ログを使用して活動を追跡するには、情報の欠落がない正確な記録が残っていなければなりません。セキュリティ・イベント・メッセージは、セキュリティ監査ログ・ファイルや、セキュリティ・クラス・イベント・メッセージを受信するよう指定されたターミナルに記録できます。次に示すように、システムにおけるセキュリティ関連イベントの追跡機能に悪影響を及ぼす可能性のある運用方法は、C2 環境では使用しないでください。

- 監査サーバまたは OPCOM を無効にしないようにします。監査サーバは、監査イベント・メッセージを処理するため、実行しておく必要があります。OPCOM はアラームを生成するために必要です。
- 1 つのクラスタ内で複数の監査ログ・ファイルを使用しないようにします。システムがデフォルトで作成する、クラスタ全体を対象とする監査ログ・ファイルを使用します。クラスタ全体を対象とする監査ログ・ファイルがない場合、一定期間の間に個別のクラスタ・ノード上で発生したイベントどうしの正確な関係を示すことは困難です。
- ビデオ・ターミナルをセキュリティ・オペレータ・ターミナルとして使用しないようにします。セキュリティ・イベント・メッセージを受信するハードコピー・ターミナルを有効にする必要があります。
- セキュリティ・オペレータ・ターミナルを公共の場所に置かないようにします。権限を有するユーザだけがターミナルを使用できるよう、ターミナルの物理的な安全を確保します。
- 監査ログ・ファイルを無視しないようにします。すべての監査ログ・イベントについて、必ず、定期的にセキュリティ監査ログ・ファイルを調べてください。特に、監査内容について何らかの変更が加えられていないかを調べます。(SET AUDIT コマンドが使用された形跡があれば、何らかの変更が行われたことになります。) 監査ログ・ファイルは通常、権限のないユーザによる読み込みと変更からは保護されています。

- 監査ログ・ファイルの改ざんを防ぎます。システム・セキュリティ監査ログ・ファイルに対しては必ずセキュリティ監査 ACE を設定し、監査ログ・ファイルに対する変更と削除のすべての試みを監査の対象にします。

次に例を示します。

```
$ SET SECURITY SYS$MANAGER:SECURITY.AUDIT$JOURNAL -
_$ /ACL=( (ALARM=SECURITY,ACCESS=WRITE+DELETE+CONTROL+SUCCESS+FAILURE) ,-
_$ (AUDIT=SECURITY,ACCESS=WRITE+DELETE+CONTROL+SUCCESS+FAILURE) )
```

オペレーティング・システムは、デフォルトで ACL イベントを監査するように設定されています。この設定の状況については、DCL の SHOW AUDIT コマンドを使用して確認できます。必要があれば、ACL アラームと ACL 監査を再び有効にします。次のコマンドを使用します。

```
$ SET AUDIT/ALARM/AUDIT/ENABLE=ACL
```

- 信頼できるユーザが、監視のない状態で操作を行うことがないようにします。登録データベースに対する変更の監査を有効にするで、信頼できるユーザ (オペレータ、マネージャ、セキュリティ管理者など) の操作を監査するようにします。また、セキュリティ監査 ACE を、キャプティブ・ログイン・コマンド・プロシージャとそれらのプロシージャを格納するディレクトリに対して設定し、変更の有無を把握できるようにしておきます。

### C.2.5.3 オブジェクトの再利用

メモリ、またはボリュームやデバイスなどの保護オブジェクトを新規ユーザに割り当てる前に、それらに古いデータが残っていないことを必ず確認します。メモリ管理サブシステムは、システム・メモリ・ページの再利用を防ぐ機能を備えており、この機能を無効にすることはできません。以下のように、再割り当てを行う前にボリュームやターミナルから古いデータを削除する処理に悪影響を及ぼす可能性のある運用方法は、C2 環境では実施しないようにしてください。

- システム・ディスク・ボリュームに対するハイウォータ・マーク処理は無効にしないようにします。オペレーティング・システムのハイウォータ・マーク処理および削除時除去は、ディスク・ブロックの再利用を防止する機能です (8.9.5 項参照)。
- ユーザがログアウト後にターミナルの電源を入れたままにしておかないようにします。ログアウト・メッセージが消されるように、ユーザはターミナルの電源を切る必要があります。これは、ログアウト・メッセージからユーザ名、および場合によってはノード名が判明してしまうこともあるからです。また、ターミナルの電源を切ることにより、ターミナルの特性がリセットされ、メモリ・バッファがクリアされます。トロイの木馬プログラムの中には、ハードウェア・フレーム・バッファを使用するものや、最新のターミナルに組み込まれているアンサーバック機能を使用するものがあります。

- テープ・オペレーション担当者が外部でテープを消去するまでは、新規ユーザがテープ・ボリュームを再利用しないようにします。オペレーティング・システムには、テープ・ボリュームの再利用を防ぐ保護機能はありません。これは、OpenVMS オペレーティング・システムでは、テープ・ドライブがシングル・ユーザ・デバイスと見なされているためです。テープの保護は、ボリューム・レベルでのみ可能です。ボリューム全体に所有権と保護を設定できますが、ボリューム上の個々のファイルに対しては設定できません。

HP では、プリント・ジョブが互いを妨げないように、ジョブごとにプリンタをクリアすることを推奨しています。セキュリティ管理者は、デバイス制御ライブラリをプリント・キューに関連付けることで、各ジョブの開始時か終了時（またはその両方）に自動的にプリンタがリセットされるようにできます。適切なリセット・シーケンスについては、使用しているプリンタ付属のドキュメントをまず参照し、次に『*OpenVMS システム管理者マニュアル*』を参照して、リセット・シーケンスのライブラリへの追加と、キューのライブラリへの関連付けの方法について確認してください。

#### C.2.5.4 クラスタの設定

共通環境クラスタとして設定されている有効なクラスタ設定はすべて、OpenVMS のセキュリティ機能を完全にサポートします。以下のように、共通環境クラスタの状態でなくなる可能性のある運用方法と機能は、C2環境では使用しないでください。

#### 注意

OpenVMS クラスタは、VAX ノードと Alpha ノードで構成できます。

- 複数の登録データベースや監査ログ・ファイルを使用して運用をしないようにします。クラスタ・システムは、セキュリティと管理の対象としては単一の範囲と見なされており、共用の登録データベースと単独の監査ログ・ファイルを使用して運用しなければなりません。性能上の理由から複数のシステム・ディスクを使用している場合、システム管理者は、使用されるすべてのシステム・ファイルが同一であるようにする必要があります。

次に示すファイルは、クラスタのすべてのメンバで共用する必要があります。

NETOBJECT.DAT	NET\$PROXY.DAT
NETPROXY.DAT	QMAN\$MASTER.DAT
RIGHTSLIST.DAT	SYS\$QUEUE_MANAGER.QMAN\$QUEUES
SYSUAF.DAT	SYSUAFALT.DAT
VMS\$AUDIT_SERVER.DAT	VMSMAIL_PROFILE.DATA



VMS\$OBJECTS.DAT	VMS\$PASSWORD_DICTIONARY.DATA
VMS\$PASSWORD_HISTORY.DATA	VMS\$PASSWORD_POLICY.EXE

- 評価済みシステムの一部に含まれていないクラスタにはノードを接続しないようにします。評価済みの OpenVMS 構成には、単一のセキュリティ管理領域となっているクラスタ環境にバインドされている DECnet ソフトウェアが含まれています。物理的に接続されているすべてのノードが、評価済みシステムの一部となっていなければなりません。

#### C.2.5.5 システムのスタートアップと運用

C2 システムとは、この付録のガイドラインに従ってあらかじめ設定を行って出荷されているシステムです。システムを設定するときは、以下に示すガイドラインを遵守してください。

- セキュリティに影響を与えるパラメータは次の値に設定します。

システム・パラメータ	設定値	説明
LGI_CALLOUTS	0	LOGINOUT コールアウトの使用を無効にします。
LOAD_PWD_POLICY	0	サイト固有のパスワード・フィルタを無効にします。
MAXSYSGROUP	7	システム・カテゴリの最大 UIC 値を 1 桁の UIC に設定します。
NISCS_CONV_BOOT	0	会話形式のシステム・ブートストラップの使用を無効にします。
RMS_FILEPROT	65,280	ユーザのファイルにデフォルトの保護コード S:RWED,O:RWED,G,W を設定します。
SECURITY_POLICY	0	特定の未評価のオペレーティング・システム・コンポーネントを無効にします。
STARTUP_P1	"####"	スタートアップ・プロシージャの最小シーケンスを無効にします。

- CONNECT CONSOLE コマンドを使用してコンソール・ストレージ・デバイスに接続しないようにします (VAX 9000 システムの場合を除く)。VAX 9000 システムでは、SET SPU\_UPDATE OFF コンソール・コマンドを使用して、ストレージ・デバイスを隔離します。コンソール・サブシステムの中には、テープやディスクなど、システムや診断プログラムのロードに使用するストレージ・デバイスをサポートしているものもあります。しかし、オペレーティング・システムはコンソール・ストレージ・デバイス上でのデータの読み込みと書き込みもサポートしているため、システムからコンソール・ストレージ

ジ・デバイスを隔離する必要があります。このコマンドは、評価済み Alpha プラットフォームでは利用できません。

- FYDRIVER を指定してブートすることでコンソール・オペレーションを有効にすることのないようにします。FYDRIVER を指定すると、次の 2 つの DCL コマンドが実行できる状態になります。
  - SET HOST/HSC では、ユーザが OpenVMS ノードから特定の HSC コンソール・オペレーションを開始できます。
  - SET HOST/DUP は、DSSI デバイスの設定に使用されます。

システムのスタートアップ時に FYDRIVER をインストールして HSC デバイスやディスクを設定したり、必要な診断を実行したりする必要のある場合は、ミニマム・ブートを実行してから FYDRIVER をインストールすることで、これらのデバイスなどを設定できるようにします。設定を終えたら、システムをシャットダウンして、FYDRIVER なしでリブートします。

#### C.2.5.6 アクセス権変更後の指定サブジェクトの即時再認証の実行

システム管理者またはセキュリティ管理者は、信頼の低いサブジェクトに対して、いつでも再認証を求めることができます。この再認証は、サブジェクトのアクセス権が変更された場合に必要となることがあります。再認証の手順は次のとおりです。この手順は、信頼できるサブジェクトのみが実行できます。

---

##### 注意

---

この手順は、信頼の低いユーザが独立プロセスを作成できる非特権アプリケーションがシステム上に存在しないことを前提としています。

また、この手順は、信頼できるユーザや特権ユーザに対して再認証を求める場合や特権アプリケーションが使用されている場合には適していません。そのような場合には、再認証を確実に行うためにシステムをリブートする必要があります。

---

1. 登録ファイルに記録されている、サブジェクトの登録レコードを変更します。
2. 登録ファイルから、サブジェクト所有者の UIC を取得します。
3. SYSMAN ユーティリティを起動します。
4. SYSMAN ユーティリティを使用して、対象サブジェクトが所有するすべてのプロセスを特定します。
  - a. OpenVMS Cluster 環境の場合、SYSMAN 環境をクラスタ全体に設定します。OpenVMS Cluster 環境ではない場合は、この手順は省略してください。

- b. SYSMAN DO SHOW SYSTEM/FULL を使用して、システム上または OpenVMS クラスタ上のすべてのプロセスの一覧を取得します。このコマンドを実行すると、各プロセスの所有者 UIC とシステム PID も表示されます。この情報を記録しておきます。
5. SYSMAN ユーティリティから、各システムでサブジェクトが所有しているすべてのプロセスを停止します。

注: 手順 4 以降に対象サブジェクトが作成したプロセスは、新しいアクセス権が適用されるため、削除する必要はありません。したがって、これは再帰的な手順ではありません。

  - a. OpenVMS Cluster 環境では、SYSMAN 環境を設定し、1 つのノードのみを参照するようにします。OpenVMS Cluster 環境ではない場合は、この手順は省略してください。
  - b. システム上の削除対象プロセスごとに、手順 2 で取得した PID を調べ、SYSMAN DO STOP/ID=pid コマンドを使用してジョブを停止します。
  - c. 手順 a と手順 b を、クラスタのすべてのノードについて、該当するすべてのプロセスが停止されるまで繰り返します。

### C.3 C2 システム構築のためのチェックリスト

本付録のこれまでの節では、C2 環境で OpenVMS オペレーティング・システムを運用するための、米国政府の要件について説明しました。以下は、米国政府のセキュリティ要件を確認するためのチェックリストです。

#### システムのインストール

- 『*OpenVMS AXP Version 6.1 Upgrade and Installation Manual*』または『*OpenVMS VAX Version 6.1 Upgrade and Installation Manual*』の説明に従って、アップグレードではなく、フル・インストールを実行しましたか。

#### 評価済みコンポーネントの使用

- 使用しているハードウェアはすべて、評価済みハードウェア・リストに掲載されているものですか。(『*Final Evaluation Report, Digital Equipment Corporation, OpenVMS VAX and SEVMS Version 6.0*』参照)。
- DECdns, LASTport, LASTport/DISK, および LAT は除外しましたか。
- システム・ファイルの保護設定は、HP から納品されたときのままになっていますか。(付録 B 参照)。
- DECwindows ソフトウェアや他の特権レイヤード・プロダクトのインストールは避けましたか。

#### 個別ユーザの確実な特定

- 特権ユーザを教育して、それらのユーザが実行する可能性のある操作の影響を理解させましたか。
- 各ユーザに一意の UIC が設定されていますか。
- すべてのアカウントに 1 文字以上のパスワードが設定されていますか。
- ユーザごとにアカウントを個別に設定していますか。
- ゲスト・アカウントはすべて削除しましたか。
- 自動ログインはすべて無効にしましたか。
- 各ユーザに一意の代理が設定されていますか。
- 代理アカウントは、すべて非特権アカウントになっていますか。
- オペレータの HSC 操作は、ハードコピー・プリンタを使用して記録していますか。
- HSC コンソールにサインイン・ログは設定されていますか。また、このログを使用するようオペレータを教育しましたか。
- 評価済み構成におけるアクセス制御文字列の使用に課される制限について、確実にユーザに周知しましたか。

#### 監査レポート・システムの管理

- 監査サーバおよび OPCOM プロセスは実行されていますか。
- クラスタ全体で 1 つの監査ログ・ファイルを使用していますか。
- セキュリティ・オペレータ・ターミナルとしてハードコピー・ターミナルを使用していますか。
- セキュリティ・オペレータ・ターミナルは、権限のある担当のみが利用できる状態になっていますか。
- 定期的に監査ログ・ファイルを確認する作業を実践していますか。
- 監査ログ・ファイルには、監査用 ACE とアラーム用 ACE の両方が設定されていますか。
- Authorization イベント・クラスおよび ACL イベント・クラスは有効になっていますか。
- 監査用 ACE を、すべてのキャプティブ・ログイン・コマンド・プロシージャとそのホーム・ディレクトリに設定していますか。

#### ディスク、テープ、ターミナルの再利用

- システム・ディスク・ボリュームのハイウォータ・マーク処理は有効になっていますか。
- ログアウト後にターミナルの電源を切るようユーザに教育してありますか。

- テープを再利用する前に内容を消去する手順は用意していますか。

#### 単一のセキュリティ管理領域の構築

- クラスタには、次のファイルが 1 つだけ存在する状態になっていますか。

NETOBJECT.DAT	NET\$PROXY.DAT
NETPROXY.DAT	QMAN\$MASTER.DAT
RIGHTSLIST.DAT	SYS\$QUEUE_MANAGER.QMAN\$QUEUES
SYSUAF.DAT	SYSUAFALT.DAT
VMS\$AUDIT_SERVER.DAT	VMSMAIL_PROFILE.DATA
VMS\$OBJECTS.DAT	VMS\$PASSWORD_DICTIONARY.DATA
VMS\$PASSWORD_HISTORY.DATA	VMS\$PASSWORD_POLICY.EXE

- クラスタ内のノードはすべて、C2 構成の一部ですか。

#### システムの起動

- セキュリティに影響を与えるパラメータは次の値に設定されていますか。

LGI_CALLOUTS	0
LOAD_PWD_POLICY	0
MAXSYSGROUP	7
NISCS_CONV_BOOT	0
RMS_FILEPROT	65,280
SECURITY_POLICY	0
STARTUP_P1	"####"

- CONNECT CONSOLE コマンドは無効になっていますか。(VAX 9000 システムの場合、SET SPU\_UPDATE\_OFF コマンドは有効になっていますか)。
- システムから FYDRIVER を除外してありますか。



## アラーム・メッセージ

この付録では、各種システム・イベントを監査した結果通知されるアラーム・メッセージについて説明します。監査システムの説明については、第9章を参照してください。監査メッセージのレコード形式の説明については、『*OpenVMS システム管理ユーティリティ・リファレンス・マニュアル*』を参照してください。

アラーム・メッセージに含まれる情報は、イベントの種類によって異なります。どの場合でも、アラーム・メッセージには、Operator Communication Manager (OPCOM) のヘッダが含まれます。このヘッダには、アラーム・メッセージが送信された日時が記録されています。さらに、アラーム・メッセージには、アラーム・イベントの種類、アラーム・イベントが発生した日時、およびイベントを発生させたユーザ（ユーザ名と PID（プロセス識別）によって識別）が含まれています。その他に、アラームが示すイベントの種類に固有の情報が含まれています。

### オブジェクト・アクセスを通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード ACCESS を加えて指定することで、保護オブジェクトに対する正常アクセスとアクセスの失敗を監査できます。オブジェクト・タイプは /CLASS 修飾子を使用して指定します。オブジェクト監査の説明については、4.7 節「保護オブジェクトの監査」を参照してください。次に例を示します。

```
%%%%%%%%%% OPCOM 17-SEP-2001 10:13:20.46 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 19728
Auditable event:      Object access
Event time:          17-SEP-2001 10:13:20.09
PID:                 30200117
Process name:        Hobbit
Username:            GREG
Process owner:       [MTI,GREG]
Terminal name:       RTA1:
Image name:          DSA1:[GREG.TEST.ACCESS]ACCESS.EXE;50
Object class name:    COMMON_EVENT_CLUSTER
Object name:         FOO
Access requested:     READ
Deaccess key:         808E3380
Status:              %SYSTEM-S-NORMAL, normal successful completion
Privileges used:      none
```

また、GRPPRV、READALL、SYSPRV、BYPASS のいずれかの特権を使用して、アクセスについての監査を行うこともできます。

## ACL によって要求されるアラーム

オブジェクトの ACL にアラーム用 ACE または監査用 ACE を追加し，SET AUDIT コマンドの /ENABLE 修飾子にキーワード ACL を加えて指定して ACL イベントを有効にすることで，個々の保護オブジェクトに対する正常アクセスとアクセスの失敗を監査できます。次に例を示します。

```
%%%%%%%% OPCOM 12-NOV-2001 10:53:16.34 %%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) and security audit (SECURITY) on FNORD, system id: 19681
Auditable event:      Object deletion
Event information:    file deletion request (IO$_DELETE)
Event time:          12-NOV-2001 10:53:16.30
PID:                 20200158
Process name:        FNORD$RTA2
Username:            HUBERT
Process owner:       [LEGAL,HUBERT]
Terminal name:       RTA2:
Image name:          $1$DIA1:[SYS0.SYSCOMMON.][SYSEXEC]DELETE.EXE
Object class name:   FILE
Object owner:        [SYSTEM]
Object protection:   SYSTEM:RWE, OWNER:RWE, GROUP:, WORLD:
File name:           _$1$DIA3:[USERS.HUBERT.TMP]FOO.BAR;2
File ID:             (4134,20,0)
Access requested:    DELETE
Sequence key:        0005E05F
Status:              %SYSTEM-F-NOPRIV, insufficient privilege or object
protection violation
```

## 登録データベースの変更に起因するアラーム

セキュリティ・イベントの Authorization クラスは，デフォルトで有効になっています。ライト・データベース，システム・ユーザ登録ファイル，およびネットワーク代理登録ファイルに変更が加えられると，直ちに監査イベント・メッセージが生成されます。

ライト・データベースの変更は，新規データベースの作成，または識別子の追加，変更，削除などの操作によって生じます。監査サーバは，ユーザの識別子に変更があった場合にも報告を行います。アラーム・メッセージは，ライト・データベースの変更に使用されたイメージおよび変更の内容を示します。次に例を示します。

```
%%%%%%%% OPCOM 15-DEC-2001 12:27:17.44 %%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) and security audit (SECURITY) on LASSIE, system id: 19661
Auditable event:      Identifier modified
Event time:          15-DEC-2001 12:27:17.43
PID:                 00000113
Username:            SYSTEM
Image name:          LASSIE$DMA0:[SYS0.SYSCOMMON.][SYSEXEC]AUTHORIZE.EXE
Identifier name:      ROBINSON
Identifier value:     %X80010014      New attributes:  RESOURCE
```

システム・ユーザ登録ファイルまたはネットワーク・ユーザ登録ファイルに加えられた変更を報告するとき，監査サーバは，変更されたレコードと変更内容を含め，あらゆる変更点についても通知します。次に例を示します。

```
%%%%%%%% OPCOM 18-DEC-2001 19:53:25.99 %%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) and security audit (SECURITY) on LASSIE, system id: 19611
Auditable event:      System UAF record addition
```



```

Event time:          18-DEC-2001 19:53:25.98
PID:                20200B25
Username:           SYSTEM
Image name:         $1$DUS0:[SYS0.SYSCOMMON.][SYSEXEC]AUTHORIZE.EXE
Object name:        SYS$COMMON:[SYSEXEC]SYSUAF.DAT;2
Object type:        file
User record added:  COOPER
Fields modified:    FLAGS,PWDLIFETIME

```

次のアラーム・メッセージは、パスワードの変更によって生成されたアラームの例です。

```

%%%%%%%%%% OPCOM 26-SEP-2001 15:12:35.95 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) and security audit (SECURITY) on FNORD, system id:
20300
Auditable event:      System UAF record modification
Event time:          26-SEP-2001 15:12:35.92
PID:                 52C00119
Process name:        Hobbit
Username:            GREG
Process owner:       [RTB,GREG]
Terminal name:       RTA2:
Image name:          $99$DUA0:[SYS0.SYSCOMMON.][SYSEXEC]AUTHORIZE.EXE
Object name:         CLU$COMMON:<SYSEXEC>SYSUAF.DAT;1
Object type:         file
User record:         GREG
Password:            New:          7C5E4DA2 F19176AF
                    Original: 7C5E4DA2 F19176AF
Password date:       New:          0 00:00:00.00
                    Original: 26-SEP-2001 15:12

```

#### 侵入行為を通知するアラーム

侵入行為は、オペレーティング・システムがデフォルトで監査します。ダイアルアップ、ローカル、遠隔、ネットワーク、および独立プロセスからの侵入行為を監査します。侵入行為に使用されたパスワードは、セキュリティ・オペレータのターミナルには表示されませんが、セキュリティ監査ログ・ファイルには記録されるため、監査分析ユーティリティで表示できます。

このタイプのアラームは、侵入行為の種類、デバイス・ユーザ、侵入元(侵入の種類が遠隔かネットワークの場合)、および親ユーザ名(侵入の種類が独立プロセスによる場合)を通知します。次に例を示します。

```

%%%%%%%%%% OPCOM 7-DEC-2001 14:33:20.69 %%%%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) on LASSIE, system id: 19611
Auditable event:      Dialup interactive breakin detection
Event time:          7-DEC-2001 14:33:20.68
PID:                 00000052
Username:            SNIDELY
Terminal name:       _LTA13: (AV47C1/LC-2-10)

```

#### オブジェクトの作成を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード CREATE を加えて指定することで、オブジェクトの作成を監査できます。このタイプのアラームは、オブジェクトのクラスと名前を通知します。次に例を示します。

```
%%%%%%%%%% OPCOM 17-SEP-2001 10:13:20.29 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 19728
Auditable event:      Object creation
Event time:          17-SEP-2001 10:13:20.01
PID:                 30200117
Process name:        Hobbit
Username:            HUBERT
Process owner:       [SST,HUBERT]
Terminal name:       RTA1:
Image name:          DSA1:[HUBERT.TEST.ACCESS]ACCESS.EXE;50
Object class name:   COMMON_EVENT_CLUSTER
Object name:         FOO
Status:              %SYSTEM-S-NORMAL, normal successful completion
```

#### オブジェクトのアクセス解除を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード DEACCESS を加えて指定することで、オブジェクトからのプロセスのアクセス解除を監査できます。このタイプのアラームは、オブジェクトのクラスを通知します。次に例を示します。

```
%%%%%%%%%% OPCOM 17-SEP-2001 10:13:38.34 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 19728
Auditable event:      Object deaccess
Event time:          17-SEP-2001 10:13:38.31
PID:                 30200117
Object class name:   COMMON_EVENT_CLUSTER
Deaccess key:        808E3380
```

#### オブジェクトの削除を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード DELETE を加えて指定することで、オブジェクトの削除を監査することができます。このタイプのアラームは、オブジェクトのクラスと名前を通知します。次に例を示します。

```
%%%%%%%%%% OPCOM 17-SEP-2001 10:13:36.17 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 19728
Auditable event:      Object access
Event time:          17-SEP-2001 10:13:36.08
PID:                 30200117
Process name:        Hobbit
Username:            HUBERT
Process owner:       [MTI,HUBERT]
Terminal name:       RTA1:
Image name:          DSA1:[HUBERT.TEST.ACCESS]ACCESS.EXE;50
Object class name:   COMMON_EVENT_CLUSTER
Object name:         FOO
Access requested:    DELETE
Status:              %SYSTEM-S-NORMAL, normal successful completion
```

Privileges used: none

### インストール・ユーティリティの使用を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード INSTALL を指定することで、(イメージのインストールまたはインストール済みイメージの削除のための) インストール・ユーティリティの使用を監査できます。インストール・アラームは、操作の種類、その操作に影響を受けるイメージの名前、インストール操作によって設定されたフラグ、および使用された特権を通知します。次に例を示します。

```
%%%%%%%%% OPCOM 7-DEC-2001 12:37:49.69 %%%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) on LASSIE, system id: 19661
Auditable event: Installed file addition
Event time: 7-DEC-2001 12:37:49.68
PID: 00000113
Username: SYSTEM
Object name: LASSIE$DMA0:[SYS0.SYSCOMMON.][SYSEXEC]NCP.EXE;1
Object type: file
INSTALL flags: /OPEN/HEADER_RESIDENT/SHARED
```

### ログインを通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード LOGIN を加えて指定することで、正常なログインを監査できます。バッチ、ダイアルアップ、ローカル、遠隔、ネットワーク、サブプロセス、および独立プロセスによるログイン・クラスを監査できます。このタイプのアラームは、ログイン・クラス、使用デバイス、ログイン元(遠隔またはネットワークの場合)、親 PID(サブプロセスの場合)、および親ユーザ名(独立プロセスの場合)を通知します。次に例を示します。

```
%%%%%%%%% OPCOM 18-DEC-2001 18:49:40.09 %%%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) on LASSIE, system id: 19611
Auditable event: Batch process login
Event time: 18-DEC-2001 18:49:40.08
PID: 20002001
Username: LEWIS
```

### ログインの失敗を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード LOGFAILURE を加えて指定することで、ログインの失敗を監査できます。バッチ、ダイアルアップ、ローカル、遠隔、ネットワーク、サブプロセス、および独立プロセスによるログイン失敗クラスを監査できます。このタイプのアラームは、ログイン・クラス、使用デバイス、失敗理由の詳細を示すステータス・メッセージ、ログイン元(遠隔またはネットワークの場合)、親 PID(サブプロセスの場合)、および親ユーザ名(独立プロセスの場合)を通知します。次に例を示します。

```
%%%%%%%%% OPCOM 7-DEC-2001 12:48:43.50 %%%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) on LASSIE, system id: 19611
```

```

Auditable event:      Network login failure
Event time:           7-DEC-2001 12:48:43.49
PID:                  0000011D
Username:             DECNET
Remote nodename:      TIGER           Remote node id:      3218
Remote username:     PROBER
Status:               %LOGIN-F-INVPWD, invalid password

```

### ログアウトを通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード LOGOUT を加えて指定することで、ログアウトを監査できます。バッチ、ダイアルアップ、ローカル、遠隔、ネットワーク、サブプロセス、および独立プロセスによるログアウト・クラスを監査できます。このタイプのアラームは、ログアウト・クラス、使用デバイス、ログイン元（遠隔またはネットワークの場合）、および親 PID（サブプロセスの場合）を通知します。次に例を示します。

```

%%%%%%%%%% OPCOM 18-DEC-2001 19:14:22.03 %%%%%%%%%%%
Message from user AUDIT$SERVER on LASSIE
Security alarm (SECURITY) on LASSIE, system id: 19611
Auditable event:      Dialup interactive logout
Event time:           18-DEC-2001 19:14:22.02
PID:                  20200001
Username:             DANCER
Terminal name:        _TTA1:

```

### ボリュームのマウントおよびディスマウントを通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード MOUNT を加えて指定することで、マウント要求またはディスマウント要求を監査できます。このタイプのアラームは、ボリュームのマウントまたはディスマウントに使用されたイメージの名前、使用デバイス、操作を記録しているログ・ファイル、ボリューム名、ボリュームの UIC と保護コード、および操作時に設定されたフラグを通知します。次に例を示します。

```

%%%%%%%%%% OPCOM 18-DEC-2001 17:43:26.94 %%%%%%%%%%%
Message from user AUDIT$SERVER on CANINE
Security alarm (SECURITY) on CANINE, system id: 19681
Auditable event:      Volume mount
Event time:           18-DEC-2001 17:43:26.04
PID:                  00000038
Username:             HOBBIT
Image name:           CANINE$DUA0:[SYS0.SYSCOMMON.][SYSEXEC]VMOUNT.EXE;1
Object name:          _CANINE$MUA0:
Object type:          device
Object owner:         [DEVO,HOBBIT]
Object protection:    SYSTEM:RWEDC, OWNER:RWEDC, GROUP:RWEDC, WORLD:RWEDC
Logical name:         TAPE$DBACK1
Volume name:          DBACK1
Mount flags:          /OVERRIDE=IDENT/MESSAGE

```

### ネットワーク接続を通知するアラーム

VAX システムでは、DECnet for OpenVMS を利用して接続が確立されている場合に、ネットワーク内の他のノードとの論理リンクの作成と終了を監査できます。このタイプの監査を行うには、SET AUDIT コマンドの /ENABLE 修飾子にキーワード CONNECTION を加えて指定します。次に例を示します。

```
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 19681
Auditable event:      DECnet logical link deleted
Event time:          12-NOV-2001 10:54:25.01
PID:                 202002EB
Process name:        FAL_16729
Username:            HUBERT_N
Process owner:       [ACCOUNTS,HUBERT]
Image name:          $1$DIA1:[SYS0.SYSCOMMON.][SYSEXE]FAL.EXE
Remote nodename:     JPT
Remote node id:      19.130
Remote username:     HUBERT
DECnet logical link ID: 16729
DECnet object name:  FAL
DECnet object number: 17
Remote logical link ID: 35429
Status:              %SYSTEM-S-NORMAL, normal successful completion
```

#### プロセス制御システム・サービスの使用を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード PROCESS を加えて指定することで、\$CREPRC や \$GETJPI などのプロセス制御システム・サービスを監査できます。このタイプのアラームは、プロセスの制御に使用されたシステム・サービス、使用デバイス、プロセス名、およびプロセスのユーザ名を通知します。次に例を示します。

```
%%%%%%%%%% OPCOM 25-JUL-2001 16:07:09.20 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 20300
Auditable event:      Process suspended ($SUSPND)
Event time:          25-JUL-2001 16:07:08.77
PID:                 30C00119
Process name:        Hobbit
Username:            HUBERT
Process owner:       [LEGAL,HUBERT]
Terminal name:       RTA1:
Image name:          $99$DUA0:[SYS0.SYSCOMMON.][SYSEXE]SET.EXE
Status:              %SYSTEM-S-NORMAL, normal successful completion
Target PID:          30C00126
Target process name:  SMISERVER
Target username:     SYSTEM
Target process owner: [SYSTEM]
```

#### 特権の使用を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード PRIVILEGE を加えて指定することで、特権の使用を監査できます。このアラームは、使用された特権と使用目的を通知します。次に例を示します。

```
%%%%%%%%% OPCOM 17-SEP-2001 10:13:20.16 %%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 19728
Auditable event:          Privilege used
Event information:        PRMCEB used to create permanent common event flag
cluster ($ASCEFC)
Event time:               17-SEP-2001 10:13:20.01
PID:                      30200117
Process name:             Hobbit
Username:                 HUBERT
Process owner:            [MTI,HUBERT]
Terminal name:            RTA1:
Image name:               DSA1:[HUBERT.TEST.ACCESS]ACCESS.EXE;50
Event flag cluster name:  FOO
Privileges used:          PRMCEB
```

#### システム・パラメータの変更を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子にキーワード SYSGEN を加えて指定することで、システム・パラメータの変更を監査できます。このタイプのアラームは、現在処理中のパラメータとディスクに保存されているパラメータの両方について通知します。次に例を示します。

```
%%%%%%%%% OPCOM 25-JUL-2001 16:09:04.67 %%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 20300
Auditable event:          SYSGEN parameter set
Event time:               25-JUL-2001 16:09:04.65
PID:                      30C00119
Process name:             Hobbit
Username:                 HUBERT
Process owner:            [LEGAL,HUBERT]
Terminal name:            RTA1:
Image name:               $99$DUA0:[SYS0.SYSCOMMON.][SYSEXE]SYSGEN.EXE
Parameters write:         SYS$SYSROOT:[SYSEXE]VAXVMSSYS.PAR;68
Parameters inuse:         SYS$SYSROOT:[SYSEXE]VAXVMSSYS.PAR;68
NSA_PAGES:                New:          15
                          Original: 10
```

#### システム時間の変更を通知するアラーム

SET AUDIT コマンドの /ENABLE 修飾子に TIME を加えて指定することで、システム時間の変更を監査できます。このタイプのアラームは、変更前と変更後のシステム時間、変更を行ったユーザの名前、および使用デバイスについて通知します。次に例を示します。

```
%%%%%%%%% OPCOM 25-JUL-2001 16:08:25.23 %%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) on FNORD, system id: 20300
Auditable event:          System time recalibrated
```

```
Event time:          25-JUL-2001 16:08:25.21
PID:                 30C00119
Process name:        Hobbit
Username:            HUBERT
Process owner:       [LEGAL,HUBERT]
Terminal name:       RTA1:
Image name:          $99$DUA0:[SYS0.SYSCOMMON.][SYSEXEC]SET.EXE
New system time:     25-JUL-2001 16:08:25.19
Old system time:     25-JUL-2001 16:08:25.18
```

### *SET AUDIT* コマンドの実行を通知するアラーム

SET AUDIT の使用はすべて自動的に監査されます。この監査を無効にすることはできません。次のアラーム・メッセージは、SET AUDIT アラームの例です。

```
%%%%%%%%%% OPCOM 12-NOV-2001 10:54:11.91 %%%%%%%%%%%
Message from user AUDIT$SERVER on FNORD
Security alarm (SECURITY) and security audit (SECURITY) on FNORD, system id: 19681
Auditable event:      Security alarm state set
Event time:           12-NOV-2001 10:54:11.58
PID:                  20200158
Alarm flags:          ACL,AUTHORIZATION,CONNECTION
                      BREAKIN: (DIALUP,LOCAL,REMOTE,NETWORK,DETACHED)
                      LOGFAIL: (BATCH,DIALUP,LOCAL,REMOTE,NETWORK,
                                SUBPROCESS,DETACHED)
```





---

## 用語一覧

この用語一覧では、このガイドで使用しているセキュリティ関連用語の定義を示します。

### アクセス制御

サブジェクト(ユーザまたはプロセス)によるシステムやコンピュータ・システム内のオブジェクトの使用能力に制限を加えること。システムへのアクセスはユーザ名とパスワードの認証によって制御し、システム内の保護オブジェクトへのアクセスは保護コード、アクセス制御リスト、および特権によって制限します。

### アクセス制御エントリ (ACE)

アクセス制御リスト(ACL)内のエントリ。アクセス制御エントリには、識別子に加えて、識別子の保持者に対して許可または拒否するアクセス権、ディレクトリに対するデフォルトの保護、またはセキュリティの詳細情報を指定します。各オブジェクトのACLには、格納領域や性能に関する全般的な問題がない限り、任意の数のエントリを入れることができます。アクセス制御リスト、識別子も参照。

### アクセス制御リスト (ACL)

オブジェクトのユーザに対して許可または拒否するアクセスの種類を定義するリスト。アクセス制御リストは、すべての保護オブジェクト(ファイル、デバイス、論理名テーブルなど)に関して作成できます。1つのACLは、1つまたは複数のアクセス制御エントリ(ACE)と呼ばれるエントリで構成されます。アクセス制御エントリも参照。

### アクセス制御文字列

遠隔ログイン時に使用される文字列。遠隔アカウントのユーザ名とユーザのパスワードを二重引用符で囲ったもの。

### アクセス・マトリックス

一方の軸にサブジェクトを列挙し、もう一方の軸にオブジェクトを列挙した表。マトリックス内の交点は、それぞれ、あるサブジェクトがあるオブジェクトに対して許可されているアクセスを表します。

### アクセス・タイプ

保護オブジェクトに対する操作を実行するのに必要な権限。OpenVMSのセキュリティ・ポリシーでは、1つの操作を完了するのに複数の権限が必要な場合があります。最も一般的なアクセス対象オブジェクトであるファイルの場合は、読み込みアクセス権、書き込みアクセス権、実行アクセス権、削除アクセス権、または制御アクセス権が必要です。

## **ACE**

アクセス制御エントリを参照。

## **ACL**

アクセス制御リストを参照。

## **ACL** エディタ

ユーザによるアクセス制御リストの作成と管理を支援する OpenVMS のユーティリティ。アクセス制御リストも参照。

## アラーム

セキュリティ・アラームを参照。

## **ALF** ファイル

自動ログインを参照。

## 英数字形式の **UIC**

ユーザ識別コード (UIC) の形式の 1 つ。グループ名とメンバ名は、それぞれ最大 31 文字の英数字 (そのうち少なくとも 1 文字は英字) で構成されます。UIC のもう 1 つの形式である数値形式には、グループ番号とメンバ番号で構成されます。ユーザ識別コード、数値形式の *UIC* も参照。

## 属性

セキュリティにかかわる文脈においては、識別子または識別子の保持者の特性を表します。属性によって、識別子に与えられた権限を増強または制限できます。たとえば、Resource 属性が割り当てられた識別子を保持するユーザは、ディスク領域を識別子に割り当てることができます。

## 監査

セキュリティ監査を参照。

## 監査の実施

システムで発生したセキュリティ関連イベントを記録し、後でセキュリティ違反やシステムの不適切な使用の有無についてシステムの動作を調べることに。セキュリティ関連イベントには、ログイン、侵入、登録データベースの変更、保護オブジェクトへのアクセスなどの操作があります。イベントのメッセージは、オペレータ・ターミナルにアラームとして送信されるか、ログ・ファイルに監査レコードとして記録されます。セキュリティ監査、セキュリティ・アラームも参照。

## 監査証跡

監査ログ・ファイルに記録されることもあるセキュリティ関連処理のパターン。監査ログ・ファイルには、登録データベースによって要求されるアクセス操作 (成功、失敗にかかわらず) などのセキュリティ関連イベントのレコードが保持されています。セキュリティ監査も参照。

## 認証

ユーザがシステムを使い始めるときに、ユーザの身元を確認する行為。OpenVMS システム（およびその他の大部分の商用オペレーティング・システム）では、パスワードが主な認証メカニズムとして使用されます。パスワードも参照。

## 登録データベース

サブジェクトとオブジェクトのセキュリティ属性を格納するデータベース。リファレンス・モニタは、これらの属性に基づいて、許可されたアクセスの種類を（もしあれば）特定します。

## 登録ファイル

システム・ユーザ登録ファイルを参照。

## 自動ログイン

ユーザがユーザ名を指定しなくてもログインできるようにする機能。オペレーティング・システムは、ユーザ名をターミナル（またはターミナル・サーバのポート）に対応させ、その割り当て情報を SYS\$SYSTEM:SYSALF.DAT ファイル（自動ログイン・ファイル、ALF ファイルとも呼ぶ）に保存します。

## 侵害

システムのセキュリティ・ポリシーに違反してシステムの資源やオブジェクトにアクセスされる結果を招く、システム・セキュリティが損なわれた状態。

## 侵入行為

権限のないソースがシステムへのアクセス権を得ようとして行う行為。システムへの最初のアクセスはログインによって行われるので、侵入行為とは主に不正にログインしようとする行為を指します。これらの行為では、そのシステムにアカウントがあることがわかっているユーザのパスワードを、入手した情報に基づく推測や試行錯誤的な方法で入力することが中心になります。回避措置も参照。

## C2 システム

米国政府によるオペレーティング・システムのセキュリティ認定レベルの 1 つ。オペレーティング・システムがディビジョン C、クラス 2 システムの基準を満たしていることを表します。

## ケーパビリティ

システムによってアクセスが制御される資源。現時点で定義されているケーパビリティは、ベクタ・プロセッサだけです。

OpenVMS のセキュリティ・ポリシーでは、ベクタ・プロセッサが不正なアクセスから保護されます。操作には、使用アクセス権または制御アクセス権が必要です。

## キャプティブ・アカウント

ユーザをキャプティブ・ログイン・コマンド・プロシージャに限定するアカウントのタイプ。Ctrl/Y の使用が禁止されます。キャプティブ・コマンド・プロシージャ

でエラーが発生して、プロシージャが終了し、ユーザが DCL コマンド・レベルに戻されるときに、プロセスが削除されます。このタイプのアカウントは、ターンキー・アカウント、結合アカウントと同義です。

#### コモン・イベント・フラグ・クラスタ

連携するプロセス同士がイベント通知を相互に提示できるようにするために、32 個のイベント・フラグをセットにしたもの。

OpenVMS のセキュリティ・ポリシーでは、コモン・イベント・フラグ・クラスタが不正なアクセスから保護されます。操作には、使用アクセス権または関連アクセス権、削除アクセス権、または制御アクセス権が必要です。

#### 制御アクセス権

オブジェクトのセキュリティ・プロファイルを変更する権限。制御アクセス権は、ACL で明示的に付与され、保護コードで暗黙に付与されます。システム・カテゴリまたは所有者カテゴリに適格するすべてのユーザは、制御アクセス権を持っています。

#### 復号

エンコードされた情報を元のエンコードされていない形式に復元する処理。復号する情報は、暗号化によってエンコードされたものです。

#### Default 属性

ACE に追加するオプションで、ディレクトリ内に作成されるすべてのファイルの ACL に当該 ACE を含めることを指示します。作成されたファイルにこの ACE が継承されるときには、ACE から Default 属性が削除されます。Default 属性を持つ識別子用 ACE は、アクセスに影響を与えません。アクセス制御エントリ、識別子用 ACE も参照。

#### デバイス

プロセッサに接続された周辺機器のクラスの 1 つで、データを受信、保存、または伝送する機能をもつもの。

OpenVMS のセキュリティ・ポリシーでは、デバイスが不正なアクセスから保護されます。操作には、読み込みアクセス権、書き込みアクセス権、物理アクセス権、論理アクセス権、または制御アクセス権が必要です。

#### 任意アクセス制御

ユーザの選択によって適用される（つまり、必須ではない）セキュリティ制御。アクセス制御リスト (ACL) は、このような任意のセキュリティ機能の典型例です。任意制御の反対は強制制御です。

#### ディスク・スカベンジング

所有者が破棄するつもりだった情報をディスクから取り出す任意の手法。元の所有者が通常の方法ではアクセスできない情報であっても、磁氣的に記録された状

態で情報の大半が残っているため、何らかのスキャベンジング手法によってそれらを取り出され、利用される可能性があります。割り当て時除去、削除時除去、除去パターンも参照。

#### 暗号化

情報のコピーを入手してもその内容がすぐにはわからないように、情報をコードに変換する処理。暗号化した情報は、復号によってデコードされます。

#### 環境識別子

4 つある識別子のクラスの 1 つ。環境識別子は、システムの利用方法に応じてユーザのグループを識別するためにシステムによって付与されます。環境識別子はログイン・クラスに対応します。たとえば、ダイアルアップ経由でシステムにアクセスするすべてのユーザは、ダイアルアップ識別子を付与されます。識別子も参照。

#### 割り当て時除去

ファイルの拡張のために新しい領域を割り当てる時点で、常に除去パターンを適用する手法。新しい領域が除去パターンで除去されるので、意味のあるデータが残らず、後でその領域を読み取ろうとしても得られるのは除去パターンのみとなります。この手法は、ディスク・スキャベンジングを阻止するために使用されます。ディスク・スキャベンジング、削除時除去、除去パターン、ハイウォーター・マーク処理も参照。

#### 削除時除去

ファイルを削除またはパージした時点で、常に除去パターンを適用する手法。この手法は、ディスク・スキャベンジングを阻止するために使用されます。ディスク・スキャベンジング、割り当て時除去、除去パターンも参照。

#### 除去パターン

磁気媒体を上書きするとき、上書き対象領域に保存されていた情報を除去するために使用される文字列。

#### 回避措置

侵入が試みられている判断される場合にオペレーティング・システムによって実行される対応動作。オペレーティング・システムには、侵入行為が進行していることを検出するための一連の基準があります。通常、権限のないユーザがログインしようとしていることをオペレーティング・システムが検出すると、回避措置によってその侵入者によるログイン操作がすべて一定期間ロックアウトされます。

#### イベント・クラス

セキュリティ関連イベントのカテゴリ。オペレーティング・システムはデフォルトでいくつかのイベント・クラスを監査しますが、セキュリティ管理者は必要に応じて監査対象イベント・クラスを追加できます。

### イベント・メッセージ

セキュリティの観点からは、システムまたはシステム内の保護オブジェクトへのユーザ・アクセスに関する各種の通知を表します。オペレーティング・システムは、成功・失敗に関わらずイベントを記録するので、セキュリティ管理者はシステム上でセキュリティ関連処理がいつ発生したのかがわかります。

### 機能識別子

識別子を定義するアプリケーションの機能コードをバイナリ値の中に含む識別子。識別子も参照。

### ファイル

ユーザにとって意味のある構造で配置された一連のデータ要素。ファイルは、名前を付けて保存されたプログラム、データ、またはその両方であり、システムからアクセスされます。ファイルに対するアクセス権には、ファイルが変更されない読み込み専用アクセス権と、ファイルの内容を変更できる読み込み/書き込みアクセス権の2種類があります。ボリュームも参照。

OpenVMS のセキュリティ・ポリシーでは、ファイルが不正なアクセスから保護されます。操作には、読み込みアクセス権、書き込みアクセス権、実行アクセス権、削除アクセス権、または制御アクセス権が必要です。

### ファイルの暗号化

暗号化を参照。

### 汎用識別子

4 つある識別子のタイプの 1 つで、ユーザのグループを 1 つまたは複数指定するもの。汎用識別子は英数字で構成され、ユーザのグループの機能を表す、わかりやすい単語が使用されます。典型的な汎用識別子として、たとえば、給与計算アプリケーションの実行を許可されるすべてのユーザに対する PAYROLL、予約デスクのオペレータに対する RESERVATIONS、などが考えられます。識別子も参照。

### グローバル・セクション

システム内のすべてのプロセスが使用できる共用メモリ領域(たとえば、FORTRAN のグローバル・コモン)。グローバル・セクションは、ディスク・ファイルへのアクセスを提供したり(ファイルによるバックアップのあるグローバル・セクションと呼ばれる)、動的に作成されたストレージへのアクセスを提供したり(ページ・ファイルによるバックアップのあるグローバル・セクションと呼ばれる)、特定の物理メモリへのアクセスを提供したり(ページ・フレーム番号 (PFN) グローバル・セクションと呼ばれる)できます。グループ・グローバル・セクション、システム・グローバル・セクションも参照。

### グループ

システム内のユーザの集合。グループ UIC がオブジェクトのグループ UIC と一致するユーザは、保護コードによって付与されるアクセス権を得ます。グループ名

は、ユーザ識別コード (UIC) の最初のフィールドに設定されます ([グループ名, メンバ名])。

#### グループ・グローバル・セクション

同じグループ内のすべてのプロセスが使用できる共用可能なメモリ・セクション。

OpenVMS のセキュリティ・ポリシーでは、グループ・グローバル・セクションが不正なアクセスから保護されます。ファイルによるバックアップのあるセクションでの操作には、読み込みアクセス権、書き込みアクセス権、実行アクセス権、削除アクセス権、または制御アクセス権が必要です。それ以外のタイプのセクションでの操作には、読み込みアクセス権、書き込みアクセス権、実行アクセス権、または制御アクセス権が必要です。グローバル・セクション、システム・グローバル・セクションも参照。

#### グループ番号

ユーザ識別コード (UIC) の最初のフィールドに設定される番号またはそれと等価の英数字 ([グループ名, メンバ名])。

#### Hidden 属性

アクセス制御エントリに追加され、ACE を変更できるのはその ACE を追加したアプリケーションだけであることを示すオプション。Hidden 属性はどの ACE タイプにも有効ですが、その使用目的はアプリケーション用 ACE を隠すことにあります。アクセス制御エントリも参照。

#### ハイウォーター・マーク

書き込みがあった最高位のファイル・アドレスを示すマーク。ユーザはこれを越えて読み込むことができません。

#### ハイウォーター・マーク処理

ディスク・スキベンジングを阻止する手法の 1 つ。この手法では、ファイルの所有者がファイルの割り当て領域に書き込んだ上限 (ハイウォーター・マーク) を追跡します。これにより、現在の書き込み領域の上限の先に存在する情報はいずれかのユーザが破棄するつもりだった情報だと見なして、書き込み領域を超える読み込みを禁止します。オペレーティング・システムは、ハイウォーター・マーク処理の目的を達成するため、本来のハイウォーター・マーク処理と割り当て時除去の方法を組み合わせで使用します。割り当て時除去も参照。

#### 保持者

特定の識別子を所有するユーザ。ユーザとユーザが保持する識別子は、ライト・データベースに記録されます。オブジェクトにアクセスするユーザに対して識別子を保持することが要求される場合は、アクセス要求を処理する時点でシステムが必ず (ライト・データベースに基づいて作成される) プロセス・ライト・リストをチェックします。

## 識別子

ライト・データベースに記録されているユーザまたはユーザのグループを表す英数字の文字列。システムがアクセス要求をチェックする時点で使用します。識別子には環境、機能、汎用、および UIC の 4 種類があります。環境識別子、機能識別子、汎用識別子、資源識別子、UIC 識別子も参照。

## 識別子用 ACE

特定のユーザまたはユーザのグループに許可されるアクセスのタイプを制御するアクセス制御エントリ。

## ジャーナル

セキュリティに影響のあるイベント（ログイン、侵入、登録データベースの変更など）を記録する監査ログ・ファイルの名前。

## ロックされたパスワード

アカウントの所有者が変更できないパスワード。ロックされたパスワードは、システム管理者または SYSPRV 特権を持つユーザだけが変更できます。

## ログ

性能やシステムに関するイベントを記録したもの。

## 論理入出力アクセス権

一連の入出力操作を実行する権限で、論理ブロック・アドレスを使ってデバイス・レベルの入出力操作に直接アクセスすることを制限付きで許可するもの。

## 論理名テーブル

オペレーティング・システムまたは特定のグループに関する論理名とその等価名を格納した共用可能なテーブル。

OpenVMS のセキュリティ・ポリシーでは、論理名テーブルが不正なアクセスから保護されます。操作には、読み込みアクセス権、書き込みアクセス権、作成アクセス権、削除アクセス権、または制御アクセス権が必要です。

## ログイン

システムに対するユーザの認証とユーザのために実行されるプロセスの作成に関わる一連の処理。

## ログイン・クラス

ユーザがシステムにログインする方法。システム管理者は、ログイン・クラス（ローカル、ダイアルアップ、遠隔、またはネットワーク）に基づいてシステムへのアクセスを制御できます。

## 強制アクセス制御

システムがすべてのユーザに義務付けるセキュリティ制御。OpenVMS システム内には、強制制御の具体例はありません。このオペレーティング・システムでは、ア



アクセス制御はオプション (任意) です。OpenVMS のセキュリティ強化バージョンである SEVMS には、強制アクセス制御 (MAC) と強化されたセキュリティ監査機能が用意されており、これによって安全なスタンドアロンまたはクラスタ化された OpenVMS システムを実現できます。

## **NETPROXY**

ネットワーク代理登録ファイルを参照。

ネットワーク代理登録ファイル (**NETPROXY.DAT** または **NET\$PROXY.DAT(VAXのみ)**)

ネットワーク内の遠隔ノードからローカル・システムに接続することを許可された各ユーザのエントリを格納するファイル。

非任意制御

強制制御を参照。

非特権

TMPMBX と NETMBX 以外の特権を持たず、システム・パラメータである MAXSYSGROUP より大きい値のユーザ識別コード (UIC) を持つアカウントのタイプを示します。

## **Nopropagate** 属性

アクセス制御エントリに追加され、通常であれば ACE を継承するはずの操作 (SET SECURITY/LIKE など) によってその ACE をコピーできないことを示すオプション。アクセス制御エントリも参照。

## 数値形式の **UIC**

ユーザ識別コード (UIC) の形式の 1 つで、ユーザのグループ番号とメンバ番号を数値形式で指定するもの。グループ番号には 1 ~ 37776 の 8 進数を指定し、メンバ番号には 0 ~ 177776 の 8 進数を指定します。

## オブジェクト

システムによってアクセスが制御される、情報の受動的な格納場所。オブジェクトへのアクセスは、そこに格納されている情報へのアクセスを意味します。ケーパビリティ、コモン・イベント・フラグ・クラスタ、デバイス、ファイル、グループ・グローバル・セクション、論理名テーブル、キュー、資源ドメイン、セキュリティ・クラス、システム・グローバル・セクション、ボリュームも参照。

## オブジェクト・クラス

共通の特性を持つ一連の保護オブジェクト。たとえば、すべてのファイルはファイル・クラスに属し、すべてのデバイスはデバイス・クラスに属します。

#### オブジェクト・セキュリティ・プロファイル

アクセス要件を定義する一連のセキュリティ要素。これらの要素には、所有者 (UIC)、UIC に基づく保護コード、および (場合によっては) ACL が含まれます。アクセス制御リスト、所有者、保護コードも参照。

#### オープン・アカウント

パスワードを必要としないアカウント。

#### オペレータ・ターミナル

システム・オペレータが操作するターミナル。イベント・クラスが有効になっていれば、システムからターミナルにシステム・イベント・メッセージを送信できます。

#### 所有者

保護オブジェクトと同じユーザ識別コード (UIC) を持つユーザ。所有者は、所有するオブジェクトに対する制御アクセス権を必ず持っているため、そのオブジェクトのセキュリティ・プロファイルを変更できます。オペレーティング・システムが所有者によるアクセスを処理するとき、保護コードの所有者フィールド内のアクセス権が考慮されます。

#### パスワード

ユーザの身元を検証するため、アカウントへのアクセスが許可されていることを証明する形式の 1 つとして、ユーザがログイン時に入力する文字列。パスワードには、システム・パスワードとユーザ・パスワードがあります。ユーザ・パスワードには、第 1 パスワードと第 2 パスワードがあります。第 1 パスワード、第 2 パスワード、システム・パスワード、ユーザ・パスワードも参照。

#### 物理入出力アクセス権

一連の入出力機能を実行する権限で、物理ブロック・アドレスを使って保守モードを除くデバイス・レベルのすべての入出力操作にアクセスすることを許可するもの。

#### 第 1 パスワード

ユーザ・パスワードのタイプの 1 つで、ユーザに対して要求される最初のユーザ・パスワード。第 2 パスワードを要求するようにシステムを設定することもできます。第 1 パスワードと第 2 パスワードは、ユーザ登録ファイル内でユーザ名に対応している必要があります。第 2 パスワードも参照。

#### 特権

システムの資源や一貫性に影響を与えるシステム機能の使用を保護するための手段。システム管理者は、ユーザの必要に応じて特権を付与し、システムへのユーザ・アクセスを規制する手段としてユーザへの特権の付与を拒否します。

## プロセス・セキュリティ・プロファイル

システムがプロセスの作成時にプロセスに割り当て一連のセキュリティ要素。これらの要素には、プロセスの UIC のほかに、その識別子および特権がすべて含まれます。識別子、特権、ユーザ識別コードも参照。

## Protected 属性

アクセス制御エントリに追加され、ACE が偶発的な削除から保護されていることを示すオプション。このような ACE を削除するには、ACL エディタを使用するか、削除するときに明示的にその ACE を指定します。

## 保護オブジェクト

システムによってアクセスが制御される共用可能な情報を格納するオブジェクト。オブジェクトも参照。

## 保護サブシステム

アクセス制御が強化されたアプリケーション。アクセス制御が強化されたアプリケーションをユーザが実行している間は、ユーザのプロセス・ライト・リストに含まれている識別子によって、サブシステムが所有するオブジェクトへのアクセスが許可されます。ユーザがアプリケーションを終了すると、これらの識別子と（それに伴って）オブジェクトへのアクセス権が直ちに消滅します。

## 保護

オブジェクトの属性のうち、ユーザが実行できるアクセスのタイプを制限するもの。アクセス制御リスト、保護コード、ユーザ識別コードも参照。

## 保護コード

ユーザとオブジェクトの所有者との関係に基づいて、そのオブジェクトに関してユーザに許可するアクセスのタイプを定義するコード。このコードによって、システム権限を持つユーザ、所有者権限を持つユーザ、同じグループに属するユーザ、システム上のすべてのユーザ（ワールド・ユーザと呼ばれる）の計 4 種類のユーザが定義されます。グループ、所有者、システム、ワールドも参照。

## 代理ログイン

遠隔ノードのユーザが実質的にはローカル・ノード上のアカウントを所有しているかのようにローカル・ノードにログインできるようにするログインのタイプ。ただし、ユーザはアクセス制御文字列内にパスワードを指定しません。遠隔ユーザがアカウントを所有する場合と、遠隔ユーザが他のユーザとアカウントを共用する場合があります。

## 擬似デバイス

メールボックスのように、ユーザやシステムによって入出力デバイスとして扱われるが、特定の物理デバイスではないエントリ。

## キュー

処理されるジョブの集合。実行キューには、バッチ、ターミナル、サーバ、プリントの4種類があります。

OpenVMS のセキュリティ・ポリシーでは、キューが不正なアクセスから保護されます。操作には、読み込みアクセス権、登録アクセス権、管理アクセス権、削除アクセス権、または制御アクセス権が必要です。

## リファレンス・モニタ

サブジェクトを認証し、サブジェクトによるオブジェクトへのあらゆるアクセスに関してセキュリティ・ポリシーを実装および実施する、オペレーティング・システム内部の管理センター。

## Resource 属性

ライト・データベースに識別子を追加するとき、後でその識別子をユーザに付与するときに指定するオプション。Resource 属性が割り当てられた識別子を保持するユーザは、ディスク領域をその識別子に割り当てることができます。

## 資源ドメイン

OpenVMS の分散型ロック管理資源へのアクセスを制御するネームスペース。

OpenVMS のセキュリティ・ポリシーでは、資源ドメインが不正なアクセスから保護されます。操作には、読み込みアクセス権、書き込みアクセス権、ロック・アクセス権、または制御アクセス権が必要です。

## 資源識別子

Resource 属性が割り当てられた識別子。この識別子の保持者は、ディスク領域をこの識別子に割り当てることができます。

## 制限付きアカウント

安全なログイン・プロシージャを持つアカウントのタイプ。システムまたはプロセスのログイン・コマンド・プロシージャの実行中、ユーザは Ctrl/Y キー・シーケンスの使用を禁止されます。ログイン・コマンド・プロシージャの実行後は、ユーザに制御が渡されます。

## ライト・データベース

システムが識別子を定義し、識別子を識別子の保持者に関連付けるために管理および使用するデータを集めたもの。

## ライト識別子

識別子を参照。

## ライト・リスト

各プロセスに関連付けられ、各プロセスが保持するすべての識別子が入っているリスト。

## RWED

データ・ファイルやディレクトリ・ファイルへのアクセスのタイプ (読み込み (Read), 書き込み (Write), 実行 (Execute), 削除 (Delete)) を示す略語。

## 第 2 パスワード

ログイン時に第 1 パスワードを正しく入力した直後に要求されるユーザ・パスワード。第 1 パスワードと第 2 パスワードを異なるユーザに伝えれば、ログイン時に複数のユーザがいることを確認できます。あまり一般的ではありませんが、パスワード長を増やす手段として第 2 パスワードを要求し、合計の文字数によってパスワードの推測に時間がかかるようにするという使い方もあります。第 1 パスワードも参照。

## 安全ターミナル・サーバ

既にログアウトしているターミナルにのみログインできるように設計されたオペレーティング・システム・ソフトウェア。ユーザがターミナル上で Break キーを押すと、安全ターミナル・サーバは (有効になっていれば) ログインしているプロセスをすべて切断してからログインを開始することで応答します。ターミナルにログインしているプロセスがない場合は、ログインが直ちに開始されます。

## セキュリティ管理者

組織のセキュリティ・ポリシーの実装と管理を担当する人。システム管理者の職務を担当する人がこの役割を果たす場合もあります。システム管理者と同じスキルに加えて、オペレーティング・システムに装備されているセキュリティ機能に関する知識が要求されます。

## セキュリティ・アラーム

セキュリティ・イベントに関係するメッセージを受信するように設定されたオペレータ・ターミナルに送信されるメッセージ。セキュリティ・アラームは、セキュリティとの関わりで警告を出すに値すると指定したイベントが発生することによって発行されます。

## セキュリティ監査

セキュリティ監査ログ・ファイルに記録される監査メッセージ。これらのメッセージにより、セキュリティに影響するイベント (ログイン、侵入、登録データベースの変更など) の発生が報告されます。システム管理者は、セキュリティ違反やシステムの不適切な使用がなかったかどうかについてシステムの動作を調べるために、このログ・ファイルを使用します。

## セキュリティ監査の実施

監査の実施を参照。

## セキュリティ・クラス

メンバがすべてオブジェクト・クラスであるオブジェクト・クラス。各メンバには、そのオブジェクト・クラス用のオブジェクト・テンプレートと管理ルーチンが定義されます。

OpenVMS のセキュリティ・ポリシーでは、セキュリティ・クラスが不正なアクセスから保護されます。操作には、読み込みアクセス権、書き込みアクセス権、または制御アクセス権が必要です。

セキュリティ担当役員

セキュリティ管理者を参照。

セキュリティ・オペレータ・ターミナル

OPCOM からセキュリティ・オペレータに送信されるメッセージを受信するように設定されたターミナルのクラス。これらのメッセージは、セキュリティ・アラーム・メッセージです。通常、このようなターミナルには、保護された部屋に設置されたハードコピー・ターミナルを使用します。この出力によって、セキュリティ関連イベントと、そのイベントのソースを特定するための詳細情報のログが得られます。

セキュリティ・プロファイル

オブジェクトのアクセス要件とサブジェクトのアクセス権のいずれかを記述する一連の要素。オブジェクト・セキュリティ・プロファイル、プロセス・セキュリティ・プロファイルも参照。

ソーシャル・エンジニアリング

何も知らないユーザやオペレータの助けを借りて、コンピュータのシステムや資源に不正にアクセスしたり、それらに関する情報を得たりする行為。多くの場合、なりすましやその他の不正行為を伴います。

サブジェクト

情報にアクセスする、または情報へのアクセスを禁じられている当事者（ユーザ・プロセスまたはアプリケーション）。オペレーティング・システムは、共用可能な情報を含むオブジェクトへのアクセスを制御します。このため、サブジェクトはオブジェクトへのアクセスを許可されていなければなりません。プロセス・セキュリティ・プロファイルも参照。

システム

保護コードにかかわる文脈においては、システム内の 1 つのユーザ集合を意味します。システム・ユーザは、通常 1 ~ 10 (8 進数) の範囲の UIC を持っていますが、システム UIC の正確な範囲は MAXSYSGROUP システム・パラメータによって決まります。この他、システム・ユーザになるには、SYSPRV 特権を持つ方法や、所有者と同じグループに属して GRPPRV を保持する方法もあります。通常、システム・オペレータとシステム管理者はシステム・ユーザです。

システム定義識別子

環境識別子を参照。

システム・グローバル・セクション

システム内のすべてのプロセスが使用できる共用可能なメモリ・セクション。

OpenVMS のセキュリティ・ポリシーでは、システム・グローバル・セクションが不正なアクセスから保護されます。ファイルによるバックアップのあるセクションでの操作には、読み込みアクセス権、書き込みアクセス権、実行アクセス権、削除アクセス権、または制御アクセス権が必要です。それ以外のタイプのセクションでの操作には、読み込みアクセス権、書き込みアクセス権、実行アクセス権、または制御アクセス権が必要です。

#### システム・パスワード

特定のターミナルへのアクセスを制御するパスワード。システム・パスワードは、通常、ダイアルアップ回線やパブリック・ターミナル・ラインなど、不正使用の対象となる恐れがあるターミナルへのアクセスを制御するために使用します。権限を持つ人がシステム・パスワードを入力した後は、ユーザが自分のユーザ・パスワードを入力できます。ユーザ・パスワードも参照。

#### システム・ユーザ登録ファイル (SYSUAF.DAT)

システム管理者がシステムへのアクセスを許可したすべてのユーザのエントリを格納するファイル。各エントリには、システムを使用する個人に割り当てるユーザ名、パスワード、デフォルト・アカウント、ユーザ識別コード (UIC)、クォータ、制限、および特権が設定されます。

#### SYSUAF

システム・ユーザ登録ファイルを参照。

#### TCB

トラステッド・コンピューティング・ベースを参照。

#### テンプレート・プロファイル

クラスの新しいオブジェクトに適用されるセキュリティ要素のデフォルト・セット。オブジェクト・セキュリティ・プロファイルも参照。

#### 結合アカウント

キャプティブ・カウントを参照。

#### トラップ・ドア

オペレーティング・システム内で不正なソフトウェアを使用するか、ソフトウェアを不正に変更することによって、システムに設定されたセキュリティ・ポリシーに違反するアクセスを可能にすること。

#### トロイの木馬プログラム

実際には不正な（場合によっては有害な）目的を持ちながら、別の目的を持っているように見せかけることで保護された領域にアクセスするプログラム。権限を持つユーザがあるプログラムを使って正当な操作を実行すると、そのプログラムに組み込まれた不正なプログラム（トロイの木馬）が不正な機能を実行します。

### トラステッド・コンピューティング・ベース (TCB)

セキュリティ・ポリシーを実施するコンピュータ・ハードウェアとオペレーティング・システム・ソフトウェアの組み合わせ。

OpenVMS の TCB には、エグゼクティブやファイル・システムの全体、ユーザ・モードで実行されないその他すべてのシステム・コンポーネント (デバイス・ドライバ、RMS、DCL など)、特権を使ってインストールされた大部分のシステム・プログラム、およびシステム管理者が TCB に関連するデータを保守するために使用するその他の各種ユーティリティが含まれます。

### ターンキー・アカウント

キャプティブ・カウントを参照。

### UAF

システム・ユーザ登録ファイルを参照。

### UIC

ユーザ識別コードを参照。

### UIC 識別子

ユーザ識別コード (UIC) に基づく英数字形式の識別子。UIC 識別子は、大括弧が付いている場合と付いていない場合があります。識別子も参照。

### UIC 保護コード

保護コードを参照。

### ユーザ・カテゴリ

保護コード内にある 4 つのフィールドの 1 つ。保護コードでは、(a) 所有者、(b) 所有者と同じグループ UIC を共用するユーザ (グループ・カテゴリ)、(c) システム上のすべてのユーザ (ワールド・カテゴリ)、(d) システムの特権または権限を持つユーザ (システム・カテゴリ) の、計 4 種類のユーザに対してアクセス権が定義されます。コードのアクセス権の並びは、必ずシステム、所有者、グループ、ワールドの順となっています。

### ユーザ識別コード (UIC)

各ユーザに割り当てられる 32 ビットの値で、ユーザが属するシステム上のグループと、そのグループ内でユーザを一意に識別するための情報を規定したもの。UIC を指定するときは必ず大括弧で囲みますが、形式は英数字と数値のいずれかを選択できます。たとえば、[SALES,JONES] という UIC によって、Sales グループのメンバーである Jones が識別されます。ファイルのような保護オブジェクトにも UIC があります。ファイルの UIC は、ほとんどの場合それを作成したユーザに由来します。

### ユーザの無責任行為

ユーザが故意に、または誤ってコンピュータ・システムに著しい損害を与えること。



### ユーザ名

ユーザがシステムにログインするときに入力する名前。ユーザ名とパスワードの組み合わせにより、その人がシステムの有効なユーザであることが特定され、認証されます。パスワード、ユーザ・パスワードも参照。

### ユーザ・パスワード

システム・ユーザ登録ファイル内のユーザ・レコードに記録される文字列。ユーザがシステムへのアクセスを認証してもらうためにログインしようとするときは、パスワードとユーザ名を正確に入力する必要があります。ユーザ・パスワードには、第1パスワードと第2パスワードの2種類があります。これらの用語は、それぞれの入力の順番を表しています。第1パスワード、第2パスワード、システム・パスワードも参照。

### ユーザの侵入行為

ユーザがシステム・ソフトウェアやシステム管理の不具合に乗じてセキュリティ制御を突破し、コンピュータ・システムにアクセスすること。

### ユーザの詮索行為

ユーザがコンピュータ・システムの十分に保護されていない部分を不正に利用すること。

### ウイルス

システム上のファイルやアカウントへ不正にアクセスすることを唯一の目的としてシステム上に作成または配置されたコマンド・プロシージャまたは実行可能イメージ。ウイルスは、ファイル保護の不具合に乗じてユーザ・ファイルへのアクセスを獲得しようとします。アクセスに成功すると、ウイルスはファイルを変更して自分自身のコピーを埋め込みます。ユーザが知らずにウイルスの入ったコードを実行するたびに、ウイルスは保護が十分でない他のプロシージャやイメージに自分自身を増殖させます。ウイルスは、システムに損害を与えるため、特権アカウントから実行されるプロシージャに侵入する方法を見つけようとします。

### ボリューム

ディスクやテープなどの、ODS-2 形式または ODS-5 形式の大容量ストレージ媒体。ボリュームは、ファイルを格納するもので、デバイスにマウントすることができます。

OpenVMS のセキュリティ・ポリシーでは、ボリュームが不正なアクセスから保護されます。操作には、読み込みアクセス権、書き込みアクセス権、作成アクセス権、削除アクセス権、または制御アクセス権が必要です。

### ワールド

オブジェクトへのアクセス権が保護コードの最後のフィールドで規定されているユーザのカテゴリ。ワールド・カテゴリには、システム上のすべてのユーザやアプ

リケーション(システム・オペレータ, システム管理者, 所有者グループのユーザ, その他すべてのグループのユーザ)が含まれます。

#### ワーム

デフォルトのネットワーク・アクセスやセキュリティ上の既知の欠陥を利用して, ネットワーク内の多くのノードに自分自身を複製するプロシージャ。ワームによる一般的な影響は, 複製されたワームがコンピュータの能力やネットワークの帯域幅を使い切ってしまうために性能が著しく低下することです。既存のプログラムを変更し, ユーザのプログラム実行をきっかけにして広まるウイルスとは異なり, ワームは単独で存在し, 自分自身のプロセス・コンテキストで動作し, 自分自身の複製を起動します。

## 数字および記号

2 つのパスワードを使用した確認... 7-20

## A

## /ACCESS 修飾子，登録ユーティリ

ティ ..... 7-3

## Accounting ログ

セキュリティ・ツール ..... 10-4

## ACE (アクセス制御エントリ)

アラーム用 ACE ..... 4-41

置き換え ..... 4-31

監査用 ACE ..... 4-41

削除 ..... 4-31

作成 ..... 4-24

作成者 ACE ..... 5-15

サブシステム ACE. 13-4, 13-6, 13-7

重要ファイル ..... 3-22

順序 ..... 4-17, 4-26, 4-30

セキュリティ監査機能 ..... 4-31

タイプ ..... 4-23

追加 ..... 4-30

デフォルトの保護用 ACE..... 4-37

リスト内への挿入..... 4-30

## ACE 属性

Default ..... 4-28

Hidden ..... 4-29

None..... 4-25

Nopropagate ..... 4-32, 4-38

Protected ..... 4-31, 4-33, 4-38

## ACL (アクセス制御リスト) ..... 4-13,

4-23, 8-25

ACE の置き換え ..... 4-31

ACE の順序 ..... 4-17, 4-26, 4-30

ACL によって生成されるアラーム D-2

C2 システムにおける監査 ..... C-10

アクセスの許可 ..... 4-24

アクセス評価における優先順位 . 4-17

エントリの順序変更..... 4-30

オブジェクトのセキュリティ要素 4-11

管理の概要..... 8-3

キューに対するアクセス権 ..... 5-24

コピー ..... 4-32

削除 ..... 4-31

作成 ..... 4-24

新しいファイルへのデフォルトの割り

当て ..... 4-27

システム・プログラム・ファイル 8-35

設計 ..... 8-5

デフォルト ACL の復元 ..... 4-32

デメリット..... 8-5

特権による適用回避..... 4-39

特権の効果..... 4-18

ネットワークでのファイル共有 12-25

パフォーマンスへの影響 ..... 8-5

表示 ..... 4-14, 4-29

ファイルのデフォルト値の復元 . 4-38

ファイル保護の設定..... 8-19, 8-26

古い識別子の削除..... 8-9

変更 ..... 4-30

保護コード..... 4-25

保護コードによるやり取り ..... 4-37

## ACL エディタ

ACL の表示 ..... 4-14

ACL の変更 ..... 4-30

## ACL の編集 ..... 4-30, 4-33

## ACME (Authentication and

## Credentials Management

Extensions) ..... 7-39

## ACME\_SERVER プロセス ..... 7-39

## ACME エージェント ..... 7-40

## ACME エージェントの順序 ..... 7-40

## ACME サブシステム ..... 7-39

## ACNT 特権 ..... A-2

## ADD/IDENTIFIER コマンド，登録ユー

ティリティ ..... 8-8

## ADD/PROXY コマンド，登録ユーティリ

ティ ..... 12-10, 12-25

**ALF (自動ログイン機能)**..... 7-47  
     ALF ファイルを使用するためのクラス  
     タの要件 ..... 11-6  
**AUTOLOGIN フラグ** ..... 7-15  
**C2 システム** ..... C-9  
     セキュリティ問題としての自動ログイン・  
     アカウント..... 7-15  
**ALLSPOOL 特権** ..... A-2  
**ALTPRI 特権** ..... A-2  
**ANALYZE/AUDIT コマンド** ..... 9-24  
     修飾子の要約 ..... 9-25  
**APPEND コマンド, /PROTECTION 修  
 飾子** ..... 8-24  
**\$AUDIT\_EVENT システム・サービス,**  
     セキュリティ関連イベントの報告 9-11  
**AUDIT 特権** ..... A-3

## B

**Break キーおよび安全ターミナル・サー  
 バ**..... 7-48  
**BUGCHK 特権** ..... A-3  
**BYPASS 特権**  
     アクセス制御の変更..... 4-18, 4-39  
     制御アクセスに対する影響..... 4-40  
     説明 ..... A-3

## C

**C2 環境** ..... C-1  
**C2 セキュリティ・システム** . C-2, C-15  
     環境構築のためのチェックリスト C-15  
**C2 セキュリティ, システム**  
     オブジェクトの保護..... C-5  
     基準 ..... C-2  
     システムのスタートアップ..... C-13  
     システム・パラメータ ..... C-13  
     対象外のソフトウェア ..... C-3  
     ドキュメント ..... C-2  
     物理的セキュリティ要件 ..... C-7  
**C2 評価の対象外, OpenVMS**  
     Management Station..... C-4t  
**CDSA**..... 1-5  
**CDSA (Common Data Security**  
     Architecture)..... 1-5

**\$CHECK\_ACCESS システム・サービ  
 ス, セキュリティ監査** ..... 9-11  
**\$CHECK\_PRIVILEGE システム・サー  
 ビス, 特権の使用の報告** ..... 9-11  
**\$CHKPRO システム・サービス**  
     アクセス制御における役割..... 4-17  
     セキュリティ監査..... 9-11  
**/CLITABLES 修飾子**..... 7-11, 8-35  
**CLUSTER\_AUTHORIZE.DAT ファイ  
 ル**..... 11-11  
**CMEEXEC 特権**..... A-5  
**CMKRNL 特権** ..... A-6  
**CONNECT コマンド, /LOGOUT 修飾  
 子**..... 3-26  
**COPY コマンド**  
     /PROTECTION 修飾子 ..... 8-24  
     割り当てられるセキュリティ・プロファ  
     イル ..... 5-15  
**CREATE/PROXY コマンド, 登録ユー  
 ティリティ**..... 12-10  
**CREATE/RIGHTS コマンド, 登録ユー  
 ティリティ**..... 8-7  
**Ctrl/B キー・シーケンス**..... 3-18  
**Ctrl/Y キー・シーケンスと制限付きアカ  
 ウント** ..... 7-14

## D

**DBG\$ENABLE\_SERVER 識別子**  
     C2 システムにおける制限 ..... C-9  
**DCL コマンド**  
     ネットワーク操作における SET  
     HOST/DTE ..... 12-21  
     ネットワーク操作における SET  
     TERMINAL ..... 12-20  
**DCL テーブル, セキュリティのための変  
 更**..... 8-34  
**DDCMP (Digital Data Communica-  
 tions Message Protocol)**  
     非同期ドライバ ..... 12-19  
**DECamds**  
     C2 評価の対象外のソフトウェア C-4t  
**DECamds, C2 評価の対象外のソフト  
 ウェア** ..... C-4t  
**DECdns (Digital Distributed Name  
 Service)**

C2 評価の対象外 ..... C-3t

**DECdns** 分散ネーム・サービス, C2 評価  
の対象外 ..... C-3t

**DECnet**

C2 システムにおける制限 ..... C-13

INBOUND パラメータ ..... 12-20

クラスタ・ノードと DECnet... 11-11

削除 ..... 12-17

受信パスワード ..... 12-20

送信パスワード ..... 12-20

動的非同期接続 12-20, 12-21, 12-23

動的非同期接続のインストール 12-19

ネットワーク・オブジェクト . 12-14,  
12-15

非特権ユーザ名 ..... 12-12

**DECnet-Plus for OpenVMS**

フルネーム

C2 評価の対象外..... C-4t

**DECnet-Plus for OpenVMS** , C2 評価  
の対象外のフルネーム ..... C-4t

**DECwindows** 画面の消去 ..... 3-14,  
3-18, 3-25

**DECwindows** ソフトウェア

C2 評価の対象外 ..... C-3t

**DECwindows** ソフトウェア, C2 評価の  
対象外 ..... C-3t

**Default** 属性, **ACE** ..... 4-28

**DELETE** コマンド, **/ERASE** 修飾  
子..... 5-16

**DETACH** 特権 ..... A-10

**DIAGNOSE** 特権 ..... A-7

**DIRECTORY** コマンド

**/SECURITY** 修飾子 ..... 5-18

**DIRECTORY** コマンド, **/SECURITY** 修  
飾子 ..... 5-18

**DISFORCE\_PWD\_CHANGE** フラ  
グ..... 7-24

**DOWNGRADE** 特権..... A-7

**DSE** (データ・セキュリティ除去)  
適合理化 ..... 8-36

**Dynamic** 属性

識別子 ..... 8-10

## E

**/EXPIRATION** 修飾子 ..... 7-4

**EXQUOTA** 特権 ..... A-8

**EXTAUTH** フラグ ..... 7-32

## F

**F\$MODE** レキシカル関数..... 3-5

**FAL** (ファイル・アクセス・リスナ) に関  
する推奨事項 ..... 12-13

**/FLAGS=CAPTIVE** 修飾子 ..... 7-10

**/FLAGS=DISIMAGE** 修飾子 ..... 8-35

**/FLAGS=DISMAIL** 修飾子 ..... 7-46

**/FLAGS=DISNEWMAIL** 修飾子.. 7-46

**/FLAGS=DISPWDDIC** 修飾子.... 7-27

**/FLAGS=DISPWDHIS** 修飾子.... 7-27

**/FLAGS=DISRECONNECT** 修飾  
子..... 7-46

**/FLAGS=DISREPORT** 修飾子 ... 7-46

**/FLAGS=DISUSER** 修飾子 ..... 7-29

**/FLAGS=DISWELCOME** 修飾子 . 7-45

**/FLAGS=GENPWD** 修飾子 . 7-21, 7-26

**/FLAGS=LOCKPWD** 修飾子 ..... 7-26

**/FLAGS=PWD\_EXPIRED** 修飾子 7-23

**/FLAGS=RESTRICTED** 修飾子 .. 7-13

**FYDRIVER** , C2 システム..... C-13

## G

**GROUP** 特権 ..... A-8

**GRPNAM** 特権 ..... 5-22, A-8, C-7

**GRPPRV** 特権 ..... A-9

システム・ユーザの権限の付与 . 4-18,  
4-33

信頼できるユーザ..... C-6

制御アクセスの許可..... 4-39

説明 ..... A-9

保護メカニズムに対する影響 ... 4-39

## H

**Hidden** 属性..... 4-29

**Holder Hidden** 属性 ..... 8-11

**HSC** コンソール・ターミナル

C2 システムにおける制限 ..... C-9  
C2 システムの要件 ..... C-8  
**HSM (Hierarchical Shelving Manager)**  
C2 評価の対象外 ..... C-4t  
**HSM (Hierarchical Shelving Manager)**, C2 評価の対象外... C-4t

## I

**IMPERSONATE** 特権 ..... A-10  
**IMPORT** 特権 ..... A-11  
**/INHERIT\_SECURITY** 修飾子 ,  
    **RENAME** コマンド ..... 5-15  
**INITIALIZE** コマンド  
    **/ERASE** 修飾子 ..... 5-16  
**INITIALIZE** コマンド , **/ERASE** 修飾子 ..... 5-16, 8-36

## K

**Kerberos** ..... 1-7

## L

**LASTport** プロトコルおよび  
    **LASTport/DISK** プロトコル  
    C2 評価の対象外 ..... C-4t  
**LAT** プロトコル, C2 評価の対象外 C-4t  
**/LGICMD** 修飾子およびキャプティブ・ア  
    カウント ..... 7-10  
**LGI** システム・パラメータ ..... 7-52t  
    **LGI\_BRK\_DISUSER** ..... 7-53t  
    **LGI\_BRK\_LIM** ..... 7-53t  
    **LGI\_BRK\_TERM** ..... 7-53t  
    **LGI\_BRK\_TMO** ..... 7-53t  
    **LGI\_CALLOUTS** ..... C-13  
    **LGI\_HID\_TIM** ..... 7-53t  
    **LGI\_RETRY\_LIM** ..... 7-52t  
    **LGI\_RETRY\_TMO** ..... 7-52t  
    **LGI\_TWD\_TMO** ..... 7-52t  
    ログイン試行の制御 ..... 7-52  
**LINK** コマンド , **/NOTRACEBACK** 修飾子 ..... 8-18  
**LOAD\_PWD\_POLICY** システム・パラ  
    メータ ..... C-13

**/LOCAL\_PASSWORD** 修飾子 .... 7-32  
**LOCKPWD** フラグ ..... 3-5  
**LOG\_IO** 特権 ..... 5-9, A-11  
**LOGOUT** コマンド ..... 3-25  
    **/HANGUP** 修飾子 ..... 3-26

## M

### MAIL.EXE

    特権を与えての再インストール . 8-33  
**MAIL** オブジェクト , 推奨されるアクセ  
    ス設定 ..... 12-13  
    メール・ユーティリティ (**MAIL**)  
        通知メッセージの制御 ..... 7-46  
**MAXSYSGROUP** システム・パラメー  
    タ ..... 4-33, C-13  
**MFD** (マスタ・ファイル・ディレクト  
    リ) ..... 5-14  
**Microsoft ACME** エージェント... 7-40  
**MIRROR** オブジェクト ..... 12-13  
**MME (Media Management Extension)**  
    C2 評価の対象外 ..... C-4t  
**MME (Media Management Extension)** , C2 評価の対象  
    外 ..... C-4t  
**MOM** (保守操作モジュール) オブジェ  
    クト ..... 12-13  
**MOUNT** コマンド , アラーム ..... D-6  
**MOUNT** 特権 ..... A-12

## N

**Name Hidden** 属性 ..... 8-11  
**NCP** (ネットワーク制御プログラム)  
    データベース変更の監査 ..... 9-8  
**NET\$PROXY.DAT** ファイル ..... 12-7  
    監査 ..... 9-4  
**NETMBX** 特権 ..... A-12  
**NET PASSWORD** コマンド ..... 7-34  
**NETPROXY.DAT** ファイル ..... 12-7  
    監査 ..... 9-4  
    通常の保護 ..... 7-30  
**NISCS\_CONV\_BOOT** システム・パラ  
    メータ ..... C-13  
**NML** (ネットワーク管理リスナ) オブジェ  
    クト ..... 12-13

**No Access** 属性 ..... 8-12  
**None** 属性 (ACE) ..... 4-26  
**Nopropagate** 属性 .. 4-32, 4-38, 5-14

## O

**OPCOM** (オペレータ通信マネージャ),  
セキュリティ監査 ..... 9-31  
**OpenSSL** ..... 1-6  
**OpenVMS Cluster** 環境, 保護オブジェ  
クト ..... 11-8  
**OpenVMS Management Station**  
C2 評価の対象外 ..... C-4t  
**OpenVMS** クラスタ環境  
C2 システムにおける制限 ..... C-12  
監査ログ・ファイルの管理 ..... 11-8  
システム・ファイルの推奨事項 . 11-3  
システム・ファイルの要件 ..... 11-2  
セキュリティの考慮事項 ..... 11-1  
セキュリティの実装 ..... 11-10  
単一のセキュリティ管理領域の構  
築 ..... 11-2  
登録データの同期 ..... 11-6  
保護オブジェクト・データベース 11-9  
**OPER** 特権 ..... A-12  
アクセス制御の変更 ..... 4-18  
キューの管理 ..... 5-24  
キューへのアクセス ..... 4-40

## P

**PFMGBL** 特権 ..... 5-20  
**PFNMAP** 特権 ..... 5-20, A-16  
**PHONE** オブジェクト ..... 12-14  
**PHY\_IO** 特権 ..... 5-9, A-16  
**PIPE** コマンド, サブプロセス監査イベン  
トへの影響 ..... 9-15  
**PIPE** サブプロセス, 監査メッセージの分  
析 ..... 9-23  
**/PRCLM** 修飾子, **AUTHORIZE** におけ  
る ..... 7-10t  
**/PRIMEDAYS** 修飾子の例 ..... 7-2  
**PRMCEB** 特権 ..... 5-3, A-18  
**PRMGBL** 特権 ..... A-18

**PRMMBX** 特権 ..... 5-9, A-18  
**Protected** 属性 ..... 4-33, 4-38  
ACE の削除 ..... 4-31  
**PSWAPM** 特権 ..... A-19  
**PURGE** コマンド, **/ERASE** 修飾  
子 ..... 5-16  
**/PWDLIFETIME** 修飾子 ..... 7-23  
**/PWDMINIMUM** 修飾子 ..... 7-25

## R

**READALL** 特権 ..... 4-18, 4-39, A-19  
**RECALL** コマンド, **/ERASE** 修飾  
子 ..... 3-18  
**REMOVE/IDENTIFIER** コマンド, 登録  
ユーティリティ ..... 8-9  
**RENAME** コマンド  
**/INHERIT\_SECURITY** 修飾子 .. 5-15  
**Resource** 属性 ..... 8-12, 8-25  
**RIGHTSLIST.DAT** ファイル  
UIC の格納方法 ..... 4-5  
監査 ..... 9-4  
作成と保守 ..... 8-7  
**RMS\_FILEPROT** システム・パラメー  
タ ..... 5-14, 8-19, 8-23, C-13

## S

**SECSRV\$CLIENT**, 予約済みの識別  
子 ..... 13-3  
**SECSRV\$COMMUNICATION**, 予約済  
みの識別子 ..... 13-3  
**SECSRV\$OBJECT**, 予約済みの識別  
子 ..... 13-3  
**SECURITY.AUDIT\$JOURNAL** ファイ  
ル ..... 9-24  
**SECURITY\_POLICY** システム・パラ  
メータ ..... 11-9, C-13  
**SECURITY** 特権 ..... A-20  
隠された ACE ..... 4-29  
**SET AUDIT** コマンド  
**/EXCLUDE** 修飾子 ..... 9-34  
**/INTERVAL** 修飾子 ..... 9-36  
**/LISTENER** 修飾子 ..... 9-21

/SERVER 修飾子 .....	9-36	ファイルのデフォルト値の復元 .	4-38
/THRESHOLD 修飾子 .....	9-36	保護コードの変更 .....	4-36
新しいログ・ファイルのオープン	9-17	例 .....	12-25
アラーム .....	D-9	<b>SET TERMINAL</b> コマンド	
セキュリティ関連イベントの有効化 .....	9-3	/HANGUP 修飾子 .....	3-26
<b>SET FILE</b> コマンド, /ERASE 修飾子 .....	5-16	ネットワーク経由での使用 .....	12-20
<b>SET HOST/DTE</b> コマンド, ネットワーク経由での使用 .....	12-21	<b>SET VOLUME</b> コマンド	
<b>SET HOST</b> コマンド .....	3-6	/ERASE_ON_DELETE 修飾子 ..	5-16, 8-35
<b>SET PASSWORD</b> コマンド .....	3-12	/NOHIGHWATER_MARKING 修飾子 .....	5-17, 8-37
/GENERATE 修飾子 .....	3-12	/PROTECTION 修飾子 .....	8-20
/SECONDARY 修飾子 .....	3-14	<b>SET VOLUME</b> コマンド,	
自動パスワード生成 .....	3-12	/ERASE_ON_DELETE 修飾子 .....	5-16
<b>SET PROCESS</b> コマンド,		Set-Up キー .....	3-25
/PRIVILEGES 修飾子 ....	4-10, 8-14	<b>SET AUDIT</b> コマンド	
<b>SET PROTECTION/DEFAULT</b> コマンド .....	8-19	推奨監査アプリケーション .....	10-5
<b>SET SECURITY</b> コマンド		<b>SET PASSWORD</b> コマンド	
ACE の削除 .....	4-31	/GENERATE 修飾子 .....	7-25
/ACL 修飾子 .....	4-30	/SYSTEM/GENERATE 修飾子 ..	7-19
例 .....	8-24	/SYSTEM 修飾子 .....	7-19
削除 .....	4-31	<b>SETPRV</b> 特権 .....	A-20
識別子用 ACE の追加 .....	4-24	<b>SET SERVER ACME</b> コマンド...	7-43
ACE の削除 .....	4-31	<b>SET TERMINAL</b> コマンド	
ACE の置き換え .....	4-31	/DISCONNECT 修飾子 .....	7-46
ACL のコピー .....	4-32	/NOMODEM/SECURE 修飾子 ..	7-48
ACL の作成 .....	8-26	/SECURE 修飾子 .....	7-47
/AFTER 修飾子 .....	4-30	/SYSPWD 修飾子 .....	7-18
/CLASS=DEVICE 修飾子 .....	8-40	パスワード・グラバの防止 .....	7-48
/CLASS 修飾子 .....	4-15, 4-25	<b>SHARE</b> 特権 .....	A-21
/COPY_ATTRIBUTE 修飾子 ....	4-32	<b>SHMEM</b> 特権 .....	A-21
/DEFAULT 修飾子 .....	4-32, 12-25	<b>SHOW AUDIT</b> コマンド .....	9-4, 9-30
/DELETE 修飾子 .....	4-31	<b>SHOW PROCESS</b> コマンド .....	4-8
/LIKE 修飾子 .....	4-32	WORLD 特権 .....	8-19
/OWNER 修飾子 .....	4-15	<b>SHOW PROTECTION</b> コマンド .	5-14
/PROTECTION 修飾子 ....	4-15, 4-35	<b>SHOW SECURITY</b> コマンド .....	4-29
コードの変更 .....	4-36	オブジェクトのクラスの表示 ...	4-15
デバイス用の変更 .....	8-40	オブジェクトのセキュリティ・プロファイルの表示 .....	4-14
/REPLACE 修飾子 .....	4-31	サイトのデフォルトの表示 .....	8-28
オブジェクト・セキュリティ・プロファイルの変更 .....	4-15	<b>SHOW USERS</b> コマンドと切断されたジョブ .....	3-26
サイトのデフォルトの管理 .....	8-27	<b>SHOW/IDENTIFIER</b> コマンド, 登録ユーティリティ .....	8-7
デフォルトのファイル保護の設定	8-24		



**SHOW/RIGHTS** コマンド, 登録ユーティリティ ..... 8-7  
**SHOW INTRUSION** コマンド.... 7-51  
**SOGW** ユーザ・カテゴリの短縮形 4-33  
**SSL** ..... 1-6  
**SSL (Secure Sockets Layer)** ..... 1-6  
**STARTNET.COM** コマンド・プロシージャ ..... 12-21  
**STARTUP\_P1** システム・パラメータ..... C-13  
**Subsystem** 属性 ..... 8-13  
**SYS\$ACM** システム・サービス ... 7-39  
**SYS\$ANNOUNCE** 論理名 ..... 7-45  
**SYS\$NODE** 論理名 ..... 7-45  
**SYS\$PASSWORD\_HISTORY\_LIFE-TIME**..... 7-27  
**SYS\$PASSWORD\_HIS-TORY\_LIMIT** ..... 7-27  
**SYS\$SINGLE\_SIGNON** 論理名 .. 7-31  
**SYS\$SINGLE\_SIGNON** 論理名ビット..... 7-36  
**SYS\$WELCOME** 論理名 ..... 7-45  
**SYSALF, ALF** (自動ログイン機能) ファイル ..... 7-47  
**SYSECURITY.COM** コマンド・プロシージャ ..... 9-18  
**SYSGBL** 特権..... 5-20, A-21  
**SYSLCK** 特権..... 5-26, A-21  
**SYSNAM** 特権 ..... 5-22, A-22  
     アクセス制御の変更..... 4-18  
     キューの管理 ..... 5-24  
     システム動作の変更..... 4-10  
**SYSPRV** 特権..... 4-18, 4-39  
     システム・ユーザの権限の付与 . 4-33  
     必要とする作業 ..... A-22  
**SYSTARTUP\_VMS.COM** コマンド・プロシージャ..... 12-19  
**SYSUAF.DAT** ファイル  
     LOCKPWD フラグ..... 3-5  
     アカウントの有効期限 ..... 3-16  
     通常の保護..... 7-30  
     特権 ..... 8-14, A-1  
     特権の記録..... 4-10  
     パスワードの保存..... 2-4

変更とセキュリティ 監査 .. 3-24t, 9-8  
 変更の監査..... 9-4  
 ライト・データベースとの同期 ... 8-7  
 ログイン・クラスの制限 ..... 3-10  
**SYSUAF** (システム・ユーザ登録ファイル)  
     外部認証用のマーク付け ..... 7-32

## T

**TASK** オブジェクト..... 12-14  
**TCB** (トラステッド・コンピューティング・ベース) ..... C-2, C-6  
     ソフトウェア ..... C-3  
     対象外のソフトウェア ..... C-3  
     特権 ..... C-6  
     ハードウェア ..... C-3  
     ファイル保護 ..... C-6  
**TMPMBX** 特権 ..... A-24  
**TTY\_DEFCHAR2** システム・パラメータ  
     遠隔ログイン用のシステム・パスワードの有効化 ..... 7-19  
     仮想ターミナルの無効化 ..... 7-46  
**TTY\_TIMEOUT** システム・パラメータ,  
     再接続時間の設定 ..... 7-46

## U

**UAF** (ユーザ登録ファイル) ..... 3-2  
     LOCKPWD フラグ..... 3-5  
     MODIFY user/FLAG=AUDIT ..... 9-8, 9-15  
     監査の有効化 ..... 9-3, 9-8  
     監査の有効化による性能への影響 9-15  
     最終ログインの記録..... 3-21  
     通常の保護..... 7-30  
     特権 ..... 8-14, A-1  
     特権の記録..... 4-10  
     パスワードの保存..... 2-4  
     変更とセキュリティ 監査 .. 3-24t, 9-8  
     変更の監査..... 9-4  
     ライト・データベースとの同期 ... 8-7  
     ログイン・クラスの制限 ..... 3-10  
**UIC** グループ  
     設計 ..... 8-1

設計の制約 .....	8-3
ユーザ特権への影響 .....	8-1
<b>UIC 識別子</b>	
従業員の退職時の削除 .....	8-9
例 .....	4-8, 4-27
<b>UIC のメンバ番号</b> .....	4-5
<b>UIC (ユーザ識別コード)</b> .....	2-4, 4-4
C2 システム .....	C-9
英数字形式 .....	4-4
オブジェクト・アクセスの評価 ..	4-17
オブジェクトの UIC の変更 .....	4-12
格納 .....	4-5
クラス・システムにおける一意性の要件 .....	11-8
グループに関する制限事項 .....	4-5
形式 .....	4-4
作成に関するガイドライン .....	4-5
数値形式 .....	4-5
ゼロ .....	4-18
プロセス .....	4-6
ライト・データベースへの追加 ...	8-6
<b>UPGRADE 特権</b> .....	A-24

## V

<b>VMS\$OBJECTS.DAT</b> ファイル ..	11-9
<b>VMS ACME</b> エージェント .....	7-40
<b>VOLPRO</b> 特権 .....	5-29, A-24
<b>VT100</b> シリーズ・ターミナル	
画面の消去 .....	3-25
<b>VT200</b> シリーズ・ターミナル	
画面の消去 .....	3-25

## W

<b>WORLD</b> 特権 .....	A-25
SHOW PROCESS コマンドへの影響 .....	8-19

## あ

<b>アカウント</b>	
DECNET アカウントの削除 ....	12-17
DISUSER フラグを使用した無効化 .....	7-4
アクセスの監査 .....	3-21

<b>アプリケーション</b> .....	12-4
<b>安全なアカウントの設計</b> ....	6-7, 7-4
<b>オープン</b> .....	3-5
<b>会話型</b> .....	7-5
<b>期限切れの更新</b> .....	3-16
<b>キャプティブ</b> .....	7-6
<b>グループ</b> .....	C-9
<b>グループの代理</b> .....	C-9
<b>ゲスト</b> .....	7-15, C-9
<b>限定アクセス</b> .....	7-5
<b>最初のログイン</b> .....	3-2
<b>初期パスワード</b> .....	3-2
<b>制限付き</b> .....	3-5, 7-6
<b>第 2 パスワード</b> .....	3-3
<b>タイプ</b> .....	3-5, 7-5
<b>代理</b> .....	7-16
<b>特権</b> .....	7-8
<b>なりすまし</b> .....	10-6
<b>ネットワーク・オブジェクト</b> .	12-14, 12-15
<b>パスワードの有効期限</b> .....	3-15
<b>パスワードの有効期限切れ後のアクセス</b> .....	3-15
<b>パスワード要件</b> .....	3-5
<b>複数アカウントのパスワード</b> ...	3-17
<b>プロジェクト</b> .....	8-26
<b>プロジェクト識別子を使用するための設定</b> .....	8-25
<b>有効期間の設定</b> .....	7-3
<b>有効期限</b> .....	3-14, 3-16
<b>ユーザ・パスワード</b> .....	3-3
<b>アカウントの有効期限</b> .....	3-16
<b>アクセス</b>	
ACL .....	4-24
ACL の適用を回避する特権 ....	4-39
BYPASS 特権 .....	4-18
GRPPRV 特権 .....	4-18
READALL 特権 .....	4-18
SYSPRV 特権 .....	4-18
オブジェクト指向 .....	2-8
拒否 .....	4-37
クラス別の変更 .....	4-18
削除されたファイル・データ ...	5-17
サブジェクト指向 .....	2-8
システムによる判定方法 .....	4-17

性能への監査の影響 .....	9-15	監査 ACE .....	9-6
プロセスの監査 .....	9-8	監査イベント・メッセージの生成 ..	9-3
保護コード .....	4-12	作成者 ACE .....	8-13, 8-26
保護コードの適用を回避する特権	4-39	サブシステム ACE .....	8-10
アクセス・カテゴリ .....	4-33	アクセス制御文字列 .....	3-18, 12-4
アクセス制御		コマンド・プロシージャ .....	3-18
ACE の順序, 重要性 .....	4-26	情報の保護 .....	3-18
ACL .....	4-23, 4-26	第 2 パスワードの使用 .....	7-21
ACL によるアクセスの拒否 .....	4-25	パスワードの開示 .....	3-16
ACL の適用回避 .....	4-39	アクセス・タイプ	
NCP の使用 .....	12-4	ACL .....	4-28
アプリケーション .....	12-5	書き込み	
オブジェクト固有の考慮事項 ...	4-40	グローバル・セクション .....	5-19t
オブジェクト・セキュリティ・プロファ		セキュリティ・クラス .....	5-27t
イル .....	4-11	デバイス .....	5-5t
環境へのアクセスの制限 ...	4-6, 4-26	論理名テーブル .....	5-21
限定アクセス・アカウント .....	7-4	資源ドメイン .....	5-25t
識別子用 ACE .....	4-23, 4-28	ファイル .....	5-10, 5-12
識別子用 ACE の使用 .....	4-23, 4-28	ボリューム .....	5-28
セキュリティ・プロファイルの比		管理 .....	5-23t
較 .....	4-1	関連付け .....	5-3t
接続 .....	12-3	キュー .....	5-23
代理 .....	12-3	共用デバイス .....	5-5
着信接続に対するデフォルト ...	12-5	クラス依存性 .....	4-35
デバイス・アクセスの制限 .....	4-25	グローバル・セクション .....	5-19
デフォルト・アプリケーション・アカ		ケーパビリティ・クラス .....	5-1
ウント .....	12-4	コモン・イベント・フラグ・クラス	
ネットワーク .....	12-3	タ .....	5-3
ネットワーク環境 .....	12-1	削除	
ネットワーク環境における制御 ..	12-7	キュー .....	5-23t
ファイルのデフォルト値の割り当		コモン・イベント・フラグ・クラス	
て .....	4-27	タ .....	5-3t
保護オブジェクト .....	4-1	論理名テーブル .....	5-22
保護コードの処理規則 .....	4-17	ファイル .....	5-10
保護コードの適用回避 .....	4-39	ボリューム .....	5-28
保護コードのユーザ・カテゴリ ..	4-13	作成	
マトリックス .....	2-7	論理名テーブル .....	5-22
明示的 .....	12-3	ボリューム .....	5-28
ユーザのアクセス要求の評価 ...	4-17	資源ドメイン .....	5-25
ユーザのクラスの拒否 .....	8-5	実行	
ルーティング初期化パスワード	12-18	グローバル・セクション .....	5-19t
ログインの制限 .....	7-1	ファイル .....	5-10
アクセス制御エントリ (ACE)		制御 .....	4-35, 5-3t
アラーム ACE .....	9-6	一般的なオブジェクト .....	4-40

ファイル ..... 5-10  
 セキュリティ監査 ..... 3-24t  
 セキュリティ・クラス ..... 5-27  
 短縮形 ..... 4-34  
 ディレクトリ ..... 5-10  
 登録 ..... 5-23t  
 非共用デバイス ..... 5-5  
 ファイル ..... 5-10  
 物理入出力 ..... 5-5t  
 保護コード ..... 4-34, 4-35  
 ボリューム ..... 5-28  
 読み込み  
   キュー ..... 5-23t  
   グローバル・セクション ..... 5-19t  
   セキュリティ・クラス ..... 5-27t  
   デバイス ..... 5-5t  
   論理名テーブル ..... 5-21  
   資源ドメイン ..... 5-25t  
   ファイル ..... 5-10  
   ボリューム ..... 5-28  
   ロック ..... 5-25t  
   論理入出力 ..... 5-5t  
   論理名テーブル ..... 5-21  
 アプリケーション，アクセス制御の設  
   定 ..... 12-5  
 アラーム  
   セキュリティのために設定する . 3-22  
 アラーム ACE  
   使用法 ..... 9-6  
 アラーム・メッセージ ..... D-1  
   ACL イベント ..... D-2  
   INSTALL イベント ..... D-5  
   SET AUDIT の使用 ..... D-9  
   オブジェクト・アクセス・イベン  
   ト ..... D-1  
   オブジェクトのアクセス解除 .... D-4  
   オブジェクトの削除 ..... D-4  
   オブジェクトの作成 ..... D-4  
   時間の変更 ..... D-8  
   システム・パラメータの変更 .... D-8  
   侵入イベント ..... D-3  
   登録データベースの変更 ..... D-2  
   特権の使用 ..... D-7  
   ネットワーク接続 ..... D-6  
   プロセス制御イベント ..... D-7

ボリュームのマウント/ディスマウン  
   ト ..... D-6  
 ログアウト ..... D-6  
 ログイン ..... D-5  
 ログインの失敗 ..... D-5  
 アラーム用 ACE ..... 4-41  
   ACL 内での位置 ..... 4-30  
 暗号化 ..... 8-35  
 安全ターミナル・サーバ .... 3-17, 7-48  
   パスワード保護 ..... 3-17  
 安全なシステムのための保守作業 ... 6-9  
 アーカイブのフラッシュ ..... 9-35  
 アーカイブ・ファイル  
   遠隔の有効化 ..... 9-21  
   セキュリティ・イベント・メッセー  
   ジ  
   用 ..... 9-20  
   セキュリティ関連イベントの分析 9-21

## い

イベント許容度とセキュリティ・レベ  
   ル ..... 1-3  
 イメージ  
   インストール  
     セキュリティへの影響 ..... 8-18  
 イメージのインストール  
   サブシステム・イメージ .. 13-1, 13-3  
   セキュリティへの影響 .... 8-18, 13-1  
 インストール・ユーティリティ  
   (INSTALL)  
   アラーム ..... D-5  
   セキュリティへの影響 .... 8-18, 13-1  
   変更の監査 ..... 9-8  
 ログイン・プログラム，安全ターミナル・  
   サーバによる認証 ..... 3-17

## う

ウイルス ..... 8-31  
 ウェルカム・メッセージ ..... 3-7  
   セキュリティ上のデメリット ... 7-45  
 運用時データベース  
   ネットワーク ..... 12-21

## え

英数字形式の UIC ..... 4-4

エミュレータ	
ターミナル.....	12-22
遠隔識別子 .....	4-6t
遠隔診断, C2 システム要件 .....	C-8
遠隔リンクの喪失, セキュリティ・アーカイブ・ファイル .....	9-37
遠隔ログイン.....	3-6
システム・パスワード .....	7-19
ログアウト.....	3-25

## お

大文字小文字の区別	
パスワードとユーザ名 .....	7-34
オブジェクト.....	4-1
ACL.....	4-13
C2 システム .....	C-5
アクセス解除のアラーム .....	D-4
アクセス, セキュリティ・プロファイルの比較 .....	4-1
アクセスの監査 .....	4-40, 4-41, 9-8
アクセスを判定する規則 .....	4-17
オペレーティング・システムによって保護されるクラス .....	4-16, 5-1
監査するイベントの種類 .....	4-41
キュー .....	5-23
クラス .....	4-16
クラス固有のアクセス権変更 ...	4-40
クラス・テンプレートの変更 ...	8-29
クラスの指定 .....	4-15
クラスの詳細 .....	5-1
グローバル・セクション .....	5-19
ケーパビリティ・クラス .....	5-1
削除のアラーム .....	D-4
作成のアラーム .....	D-4
識別子用 ACE によるアクセスの制御 .....	4-24, 4-25
資源ドメイン .....	5-25
所有権の再割り当て.....	4-12
セキュリティ監査に必要な ACE の追加 .....	4-41
セキュリティ管理の概要 .....	5-1
セキュリティ・クラス .....	5-26

セキュリティ・プロファイル ...	4-11, 4-17
セキュリティ・プロファイルの表示 .....	4-14
セキュリティ・プロファイルの変更 .....	4-15
セキュリティ・モデルにおける ...	2-2
セキュリティ・モデルにおける役割 .....	2-4
セキュリティ要素のソース.....	4-12
単位で整理されたアクセス.....	2-8
デフォルトの保護と所有権の管理	8-19
デフォルトの保護と所有権の表示	8-27
保護オブジェクトの特性 .....	4-11
保護コード.....	4-12, 4-33
保護コードによるアクセスの許可	4-33
ボリューム.....	5-28
論理名テーブル .....	5-21
オブジェクト・クラス	
詳細 .....	5-1
セキュリティ属性.....	4-15
オブジェクトの永続性	
キュー .....	5-24
グローバル・セクション .....	5-21
ケーパビリティ・オブジェクト ...	5-2
コモン・イベント・フラグ・クラス	
タ .....	5-4
資源ドメイン .....	5-26
セキュリティ・クラス・オブジェクト	
ト .....	5-28
デバイス.....	5-10
ボリューム.....	5-29
論理名テーブル .....	5-23
オブジェクトの所有権	
オブジェクトのセキュリティ要素	4-11
規則の例外.....	4-12
再割り当て.....	4-12
資源識別子.....	5-12
ディレクトリのデフォルトの管理	8-27
適合 .....	4-12
デフォルトの管理.....	8-19, 8-23
ファイル.....	5-13
ファイル作成時の割り当て.....	8-20
ファイルのデフォルト値の復元 .	4-38

- 変更 ..... 4-12, 4-15
- 保護チェックにおけるゼロの UIC ..... 4-18
- オープン・アカウント ..... 3-5
- C2 システム ..... C-9
- キャプティブ・アカウント ..... 7-10t
- キャプティブの推奨事項 ..... 7-30
- オープンされたファイルと ACL のメモリ消費 ..... 8-5

## か

- 改ざん，システム・ファイルに対する操作の検出 ..... 10-6
- 外部認証 ..... 7-31
  - DECnet-Plus および NET\_CALLOUTS パラメータ ..... 7-38
  - DECnet-Plus の要件 ..... 7-38
  - /LOCAL\_PASSWORD 修飾子の使用 ..... 7-32
  - NET PASSWORD コマンド ..... 7-34
  - POP サーバでの接続の失敗 ..... 7-38
  - SYS\$SINGLE\_SIGNON 論理名ビットの指定 ..... 7-36
  - ネットワーク・ダウン時の無効化 ..... 7-32
  - パスワード検証 ..... 7-35
  - パスワードの設定 ..... 7-34
  - ユーザ・アカウントのマーク付け ..... 7-32
  - レイヤード・プロダクトおよびアプリケーションに対する影響 ..... 7-33
  - 論理名の定義 ..... 7-31
- 会話型アカウント ..... 7-5
- 会話型識別子 ..... 4-6t
- 会話型ログイン ..... 3-5
  - 遠隔 ..... 3-6
  - クラス ..... 3-6
  - システム・メッセージ ..... 3-7
  - ダイアルアップ ..... 3-6, 3-10
  - ローカル ..... 3-6
- 会話モード
  - プロセス ..... 3-5
- 書き込みアクセス
  - ACL による許可 ..... 4-28
  - グローバル・セクション ..... 5-19t
  - 資源ドメイン ..... 5-25t
  - セキュリティ・クラス ..... 5-27t
  - デバイス ..... 5-5t

- ファイル ..... 5-10, 5-12
- 保護コードによる許可 ..... 4-34
- ボリューム ..... 5-28
- 論理名テーブル ..... 5-21
- 仮想ターミナル ..... 7-46, 12-20
  - 使用禁止 ..... 3-7
  - 切断されたプロセス ..... 3-26
  - ログアウト ..... 3-26
  - ローカル・デバイス ..... 5-4
- 画面の消去 ..... 3-25, C-11
- 画面の消去，VT100 シリーズ・ターミナル ..... 3-25
- 画面の消去，VT200 シリーズ・ターミナル ..... 3-25
- 環境識別子 ..... 8-4
  - 識別子用 ACE ..... 4-26
  - 汎用識別子の条件指定 ..... 8-5
  - 例 ..... 4-6, 4-8, 4-27
- 監査
  - アプリケーション ..... 10-5
  - セキュリティ・イベント ..... 9-1
  - セキュリティ機能 ..... 10-5
- 監査 ACE
  - 使用法 ..... 9-6
- 監査サーバ・データベース ..... 9-30
- 監査証跡
  - セキュリティ・モデルにおける ... 2-2
- 監査分析ユーティリティ
  - (ANALYZE/AUDIT) 9-1, 9-21, 9-28
  - ASCII 出力 ..... 9-26
  - アーカイブ・ファイルの分析 ... 9-21
  - イベントを無視する条件 ..... 9-22
  - 概要 ..... 9-22
  - 会話型コマンド ..... 9-27
  - 起動 ..... 9-24
  - 出力のタイプ ..... 9-26
  - 前提条件 ..... 9-22
  - 日次レポートの作成 ..... 9-23
  - バイナリ出力 ..... 9-26
  - 分析の基準の決定 ..... 9-28
  - 例 ..... 9-28
  - レポート形式 ..... 9-25t
- 監査分析ユーティリティからの ASCII 出力 ..... 9-26
- 監査分析ユーティリティからのバイナリ出力 ..... 9-26

監査分析ユーティリティのコマンド・モード, 表示の操作 .....	9-27
監査用 ACE .....	4-41
監査リスナ・メールボックス	
監査イベント・メッセージの取得 .....	9-21
プログラムの例 .....	9-22
無効化 .....	9-22
監査サーバ・プロセス	
イベント・メッセージの配信の遅延 .....	9-32
エラー処理 .....	9-36, 9-37
管理 .....	9-29
最終的なサーバのアクション ...	9-35
実行されるタスク .....	9-30
ディスク転送レートの変更 .....	9-35
無効化 .....	9-31
メッセージのフロー制御 .....	9-33
メモリの制約 .....	9-34
有効化 .....	9-31
ログ・ファイルの事前拡張 .....	9-36
監視のガイドライン .....	6-9
管理アクセス .....	5-23t
関連付けアクセス .....	5-3t
カード・リーダ, デフォルトのセキュリティ要素 .....	5-7

## き

期限切れパスワード, システム・メッセージ .....	3-14
機能識別子 .....	4-7
機密ファイルとセキュリティ監査 ..	3-21
キャプティブ・アカウント ....	3-5, 7-9
Ctrl/Y キー・シーケンス .....	7-9
運用アカウントの例 .....	7-7
コマンド・プロシージャ .....	7-11
使用条件 .....	7-5
メールと配信通知の無効化 .....	7-46
ロックされたパスワード .....	7-10t
キュー	
ACL によるアクセス権 .....	5-24
OPER 特権によって付与されるアクセス権 .....	4-40
アクセスのタイプ .....	5-23

監査対象イベント .....	5-24
セキュリティ要素 .....	5-23
テンプレート・プロファイル ...	5-24
必要な特権 .....	5-24
プロファイルの保存 .....	5-24
保護オブジェクト .....	4-16t
保護コードによるアクセス権 ...	5-24
強制アクセス制御 ....	A-3, A-11, A-24
共用可能デバイス, 必要なアクセス権 .....	5-7
共用ファイル, クラスタ・システムにおける考慮事項 .....	11-7
緊急のアカウントと特権 .....	8-17

## <

### クラスタ環境

C2 システムにおける制限 .....	C-12
SYSMAN の要件 .....	11-10
監査ログ・ファイルの管理 .....	11-8
システム・ファイルの推奨事項 ..	11-3
システム・ファイルの要件 .....	11-2
セキュリティの考慮事項 .....	11-1
セキュリティの実装 .....	11-10
単一のセキュリティ管理領域の構築 .....	11-2
登録データの同期 .....	11-6
保護オブジェクト .....	11-8
保護オブジェクト・データベース ..	11-9
クラスタ管理者とセキュリティ管理	
者 .....	11-1
クラスタ全体での侵入検出 .....	11-9
クラスタでの同期化, タイムスタン	
プ .....	9-35
グループ	
UIC の設計 .....	8-1
設計 .....	8-7
組織化のガイドライン .....	8-1
グループ UIC 名 .....	4-4
グループ・アカウント, C2 システム ..	C-9
グループ番号	
UIC .....	4-5
クラスタ・システムにおける一意性の要件 .....	11-8

予約済み UIC .....	4-5
グループ番号とパスワード .....	11-11
グループ番号とパスワード, クラスタ向け のセットアップ .....	11-10
グループ・ユーザ (セキュリティ・カテ ゴリ) .....	4-13, 4-34
グローバル・セクション	
アクセスの制限 .....	5-20
アクセスのタイプ .....	5-19
監査対象イベント .....	5-20
グループ .....	4-16t
システム .....	4-17t
セキュリティ・プロファイルの再設 定 .....	5-21
セキュリティ要素 .....	5-19
デフォルトの保護 .....	C-5
テンプレート・プロファイル ...	5-20
必要な特権 .....	5-20

## け

形式	
UIC (ユーザ識別コード) .....	4-4
識別子用 ACE .....	4-24
セキュリティ監査 ACE .....	9-7t
保護コード .....	4-33
ライト識別子 .....	4-6
ゲスト・アカウント	
C2 システム .....	C-9
限定アクセス・アカウントとして	7-15
権限ベースのシステム .....	2-8
権限, ユーザ	
表示 .....	8-7
限定アクセス・アカウント .....	7-5
ケーパビリティ・オブジェクト	
アクセスのタイプ .....	5-1
テンプレート・プロファイル ....	5-2
プロファイルの再設定 .....	5-2
保護オブジェクト .....	4-16t
要素 .....	5-1
ケーパビリティ・ベースのシステム .	2-8

## こ

攻撃, システムの種類 .....	10-1
コマンド, 使用の制限 .....	8-34

コマンド・プロシージャ	
STARTNET.COM .....	12-21
SYSTARTUP_VMS.COM .....	12-19
アクセス制御文字列 .....	3-18
コモン・イベント・フラグ・クラスタ	
アクセスのタイプ .....	5-3
監査対象イベント .....	5-4
システムによるテンプレートの変 更 .....	5-3
セキュリティ・プロファイルの再設 定 .....	5-4
セキュリティ要素 .....	5-2
テンプレート・プロファイル ....	5-3
必要な特権 .....	5-3
保護オブジェクト .....	4-16t
コンソール・ターミナル	
C2 システム .....	C-13
C2 システムの要件 .....	C-7
HSC および C2 システム要件 ....	C-8
コンソール, パスワードの有効化 .	7-21
コンパイラ, <b>ACL</b> を用いた使用の制 限 .....	8-34

## さ

最終ログイン・メッセージ .....	3-21
無効化 .....	7-46
サイトのセキュリティ .....	1-4
サイン・オン, シングル .....	7-31
作業時間制限 .....	3-10
作業時間制限によるジョブの強制終 了 .....	3-10
作業の制限 .....	7-3
削除アクセス	
キュー	
ACL .....	5-23t
保護コード .....	5-23t
コモン・イベント・フラグ・クラス タ .....	5-3t
ファイル .....	5-10
保護コードによる許可 .....	4-34
ボリューム .....	5-28
論理名テーブル .....	5-22
削除時除去 .....	5-16, 8-35
C2 システム .....	C-11
作成アクセス	



ボリューム.....	5-28
論理名テーブル.....	5-22
作成者 <b>ACE</b> .....	5-15
資源識別子.....	8-13
例.....	8-26
サブシステム <b>ACE</b> .....	8-10,
13-4, 13-5, 13-7	
形式 .....	13-5
サブプロセス	
監査イベントの増加.....	9-15
監査メッセージの分析 .....	9-23
サーバ	
安全ターミナル .....	3-17
監査 .....	9-30
セキュリティ .....	7-53

## し

時間帯によるログイン制限 .....	3-10
識別子	
<b>ACE</b> .....	4-23
<b>UIC</b> .....	4-8
一意性の要件 .....	11-8
カスタマイズ .....	8-4
環境 .....	4-6, 4-8, 8-4
機能 .....	4-7
形式 .....	4-6
削除 .....	8-9
作成 .....	4-25
資源	
およびディレクトリの所有権 .....	8-20
使用の監査.....	9-8
セキュリティ監査レポート.....	4-8
タイプ .....	4-6
ディレクトリの所有者として ...	8-25
汎用 .....	4-8, 4-24
ファイルの所有者.....	5-12, 5-13
プロセス.....	4-1
プロセスの表示 .....	4-8
保護サブシステム.....	13-7
ユーザへの割り当て.....	8-8
予約 .....	13-3
ライト・データベースへの追加 ...	8-7
識別子としてのユーザ名 .....	2-4, 4-7t

識別子の <b>Dynamic</b> 属性 .....	8-10
識別子の属性.....	8-9, 8-13
<b>Dynamic</b> .....	8-10
<b>Holder Hidden</b> .....	8-11
<b>Name Hidden</b> .....	8-11
<b>No Access</b> .....	8-12
<b>Resource</b> .....	8-12
<b>Subsystem</b> .....	8-13
説明 .....	8-9
識別子用 <b>ACE</b> .....	4-23, 4-30, 13-5
<b>ACE</b> の順序.....	4-26
<b>ACL</b> への追加.....	4-30
<b>Default</b> 属性 .....	4-27
アクセス条件の設定.....	4-26
アクセスの拒否 .....	4-25
解釈 .....	4-24
形式 .....	4-24
作成 .....	4-24
汎用識別子の使用.....	4-24
保護サブシステム.....	13-5
資源識別子 .....	8-25
ファイルの所有者.....	5-13
資源ドメイン.....	4-16t
アクセスのタイプ.....	5-25
監査対象イベント.....	5-26
セキュリティ要素.....	5-25
テンプレート・プロファイル ...	5-25
必要な特権.....	5-26
プロファイルの保存.....	5-26
資源の監視 .....	9-37
無効化 .....	9-37
時刻	
クラスタ時刻の同期化 .....	9-35
システム時刻の変更の監査.....	9-8
システム	
アクセスの制御 .....	3-5
使用の制御.....	3-3t
システム管理者	
監査要件の評価 .....	9-12
システム管理ユーティリティ ( <b>SYSMAN</b> )	
<b>LGI</b> パラメータの変更 .....	11-2
クラスタ・セキュリティ・データの変 更 .....	11-11
クラスタの管理 .....	11-10

システム・サービス, 監査イベント情報..... 9-11

システム障害

ハードコピー出力の破棄..... 3-25

システム生成ユーティリティ (SYSGEN), パラメータ変更の監査..... 9-8

システム・パスワード..... 3-3t

ガイドライン..... 7-19

格納場所..... 7-19

推奨される変更頻度..... 7-24

設定..... 7-18

デメリット..... 7-20

入力..... 3-4

必要な最低限の長さ..... 7-25

変更..... 7-19

ログイン失敗の原因..... 3-9

システム・パラメータ

システム・ユーザの定義 (セキュリティ・カテゴリ)..... 4-40

切断されたプロセスの制御..... 7-46

必要な C2 設定..... C-13

変更の監査..... 9-8

システム・ファイル

ACL の追加..... 8-34

ACL の利点..... 10-6

Alpha のデフォルトの保護..... 8-32

監査における推奨事項..... 10-6

推奨ファイル..... 11-3

デフォルトの保護..... 8-32, B-1

必須ファイル..... 11-2

保護..... 8-32

保護コードと所有権..... B-1

システム・ユーザ (セキュリティ・カテゴリ)..... 4-13, 4-40

MAXSYSGROUP パラメータによる定義..... 4-33

資格..... 4-33

実行アクセス

グローバル・セクション..... 5-19t

ファイル..... 5-10

保護コードによる許可..... 4-34

自動ダイアル・プロトコル..... 12-21

自動パスワード生成..... 3-12

最小限の長さ..... 3-13

デメリット..... 3-13

例..... 3-13

ジャーナルのフラッシュ..... 9-35

受信パスワード..... 12-20

シュレダ..... 3-25

使用アクセス..... 5-2

除去パターン..... 5-16, 8-35

ジョブ・コントローラ

作業時間制限による影響..... 3-10

作業時間の制限の適用..... 7-3

ジョブの終了

作業時間制限による強制..... 3-10

所有者

ユーザ・アクセスのカテゴリ... 4-34

シングル・サイン・オン..... 7-31

侵入

検出..... 7-48

回避手順..... 3-11

イベントの報告..... 3-24

クラスタ全体..... 11-9

システム・パラメータ..... 7-51

除外期間の設定..... 7-52

データベース..... 7-49

二重パスワードによる対策... 7-20

試み..... 3-11

侵入行為..... 1-2, 3-11

回避..... 3-11

監査..... 9-4, 9-8

検出..... 7-48, 7-52

セキュリティ監査レポート..... 9-28

二重パスワードによる対策..... 7-20

侵入行為のアラーム..... D-3

侵入データベース..... 7-51

## す

数値形式の UIC..... 4-5

スプールされたデバイス, 必要なアクセス権..... 5-6

## せ

制御アクセス

獲得..... 4-17, 4-35, 4-40

キュー..... 5-23t

グローバル・セクション..... 5-19

コモン・イベント・フラグ・クラス

タ..... 5-3t

資源ドメイン .....	5-25	システム・コンソール上での無効	
制限 .....	4-40	化 .....	9-19
セキュリティ・クラス .....	5-27t	有効にするイベント.....	9-3, 9-15
デバイス.....	5-5t	セキュリティ・アーカイブ・ファイル	
ファイル.....	5-10	遠隔リンクの喪失.....	9-37
ボリューム.....	5-28	セキュリティ・オペレータ・ターミナ	
論理名テーブル .....	5-22	ル.....	9-19
制限付きアカウント .....	3-5, 7-13	セキュリティ監査 .....	3-21
使用条件.....	7-5t	C2 システムにおける制限 .....	C-10
設定 .....	7-6	アカウントおよびファイルへのアクセ	
プロセス生成の危険性 .....	7-10t	ス .....	3-21
生成パスワード .....	3-12	アーカイブ・ファイル .....	9-21
最小限の長さ .....	3-13	アーカイブ・ファイルへのイベント・	
初期パスワード .....	7-17	メッセージの送信 .....	9-20, 9-21
デメリット.....	3-13	一時中断からのプロセスの除外 .	9-34
長さ .....	7-25	イベントの精度 .....	4-41
要求 .....	7-21	イベント・メッセージの制御 ...	9-33
要件 .....	7-28	エラー処理.....	9-36, 9-37
例.....	3-13	オブジェクトへのアクセスに関する報	
性能		告 .....	4-40
セキュリティ監査の影響 .....	9-15	オブジェクトの使用に関する報告 .	4-8
セキュリティ		オペレータ・ターミナルへのイベント・	
オペレーティング・システムのモデ		メッセージの送信 .....	9-19
ル .....	2-1	監査サーバ・データベース.....	9-30
環境要因.....	1-4	監査サーバの管理.....	9-29
監査の管理.....	9-29	監査証跡.....	C-10
監査要件の評価 .....	9-12	監査の無効化 .....	9-31
クラスタ全体での侵入検出.....	11-9	監査の有効化 .....	9-31
システムによって保護されるオブジェ		監査リスナ・メールボックス ...	9-21
クト .....	4-16	監査ログ・ファイルの分析.....	9-22
性能への影響		キュー .....	5-24
監査 .....	9-15	クラスタ時刻の同期化 .....	9-35
ディスク上のデータの除去.....	5-16	グローバル・セクション .....	5-20
デフォルトの保護と所有権の管理	8-19	ケーパビリティ・オブジェクト ...	5-2
データ保護メカニズム .....	4-11	効果的な使用 .....	9-22
トロイの木馬プログラム .....	5-18	コモン・イベント・フラグ・クラス	
ハイウォータ・マーク処理.....	5-16	タ .....	5-4
ファイル・セキュリティの最適化	5-18	資源ドメイン .....	5-26
レベルの定義 .....	1-3	性能への影響 .....	9-15
セキュリティ・アラーム .....	3-22	セキュリティ・クラス・オブジェク	
イベントの有効化の例 .....	9-13	ト .....	5-27
監査ログ・ファイル.....	C-10	資源監視の無効化.....	9-37
起動するイベント.....	3-24	中程度のセキュリティ条件.....	9-14e
サンプル・メッセージ .....	9-1, D-1	ディレクトリ .....	5-15

デバイス .....	5-9	監査証跡 .....	2-6
デフォルトの特性 .....	9-30	サイトの要件の評価 .....	9-12
ファイル .....	3-21, 5-15	高いセキュリティ条件 .....	1-3, 9-14
ファイルへの ACE の追加 .....	3-22	中程度のセキュリティ条件 .....	9-13
ボリューム .....	5-29	デフォルトの監査イベント .....	2-7
メッセージ .....	3-23	中程度のセキュリティ条件 .....	1-3
メモリの制約 .....	9-34	低いセキュリティ条件 .....	1-3, 9-13
メールボックスへのイベント・メッセージの送信 .....	9-21	セキュリティ監査レポート ..	9-22, 9-28
有効にされたオブジェクト・クラス .....	4-41	疑わしい活動の分析 .....	9-24
論理名テーブル .....	5-22	完全な形式 .....	9-27e
セキュリティ監査 ACE		簡略形式 .....	9-26e
ACL 内での位置 .....	4-30	形式 .....	9-25t
セキュリティ監査イベント .....	3-24	作成 .....	9-22
アラームとしての有効化 .....	9-13	出力先 .....	9-26t
遠隔アーカイブ・ファイルへの送信 .....	9-20	詳細調査 .....	9-28
オペレータ・ターミナルへの送信 ..	9-19	スケジューリング .....	9-23
監査としての有効化 .....	9-13	定期的な検査 .....	9-23
監査ログ・ファイルへの送信 ...	9-16	内容の定義 .....	9-24, 9-26
クラス .....	9-8	要約形式 .....	9-27e
システム・サービス .....	9-11	ライト識別子 .....	4-9
すべてのクラスの無効化 .....	9-15	例 .....	9-26, 9-28
すべてのクラスの有効化 .....	9-14	セキュリティ監査ログ・ファイル ..	2-7, 3-23
セキュリティ上のニーズに基づく ..	9-12	C2 システム .....	C-10
デフォルトのクラス... 9-1, 9-3, 9-13		会話形式による分析 .....	9-27
特権監査の抑制 .....	9-9	作成 .....	9-17
ネットワーク .....	12-3	事前拡張 .....	9-37
表示 .....	9-4	説明 .....	9-16
プロセス制御監査の抑制 .....	9-10	ディスク領域の割り当て .....	9-36
報告 .....	9-3, 9-15	手順 .....	9-17
リスナ・メールボックスへの送信 ..	9-21	特徴 .....	9-17
例 .....	9-4	場所の変更 .....	9-18
セキュリティ監査イベント・メッセージ		報告するイベント .....	9-15
サーバへの配信の制御 .....	9-33	保守 .....	9-17
スタートアップ時の配信の遅延 ..	9-32	メッセージ転送レートの変更 ...	9-35
ディスク転送レートの変更 .....	9-35	メリット .....	9-15
無視する条件 .....	9-22	レコードの選択 .....	9-26
セキュリティ監査機能		セキュリティ管理 .....	6-1
クラスタの考慮事項 .....	11-8	SYSMAN の要件 .....	11-10
セキュリティ監査の実施 .....	10-5	監査ログ・ファイルの管理 .....	11-8
イベント・クラスの有効化 .....	9-3	クラスタ .....	11-2, 11-3
イベントの無効化 .....	9-3	クラスタのグループ番号の変更 ..	11-11
イベントの有効化 .....	9-2	クラスタ・パスワードの変更 ..	11-11
概要 .....	9-1	登録データの同期 .....	11-6
		保護オブジェクト	

クラスタの管理対象 .....	11-8
データベース .....	11-9
ポリシーの策定 .....	1-3, 6-1, 10-1
セキュリティ管理者	
C2 要件 .....	C-15
安全なシステムを維持するためのチェッ クリスト .....	6-9
クラスタ管理者とセキュリティ管理 者 .....	11-1
個人用カウント .....	6-5
システム・パスワード .....	3-3
必要な特権 .....	6-5
目標 .....	1-1
役割 .....	6-1
ユーザのトレーニング .....	3-27, 6-6
セキュリティ・カーネル, 定義 .....	2-3
セキュリティ機能	
アカウントの有効期間 .....	3-14, 3-16, 7-3
アクセス制御 .....	4-1, 7-1
安全ターミナル・サーバ .....	3-17, 7-47
監査 .....	3-22, 9-1, 10-5
作業時間制限 .....	3-10
削除時除去 .....	8-35
システム・パスワード .....	3-3, 3-9
自動パスワード生成 .....	3-12, 7-17
除去パターン .....	5-16
侵入検出 .....	3-11, 7-20
セキュリティ・アラーム .....	3-22
第 2 パスワード .....	3-4, 3-14
ダイアルアップの再試行 .....	3-10
代理アカウント .....	12-12
代理ログイン .....	3-19, 12-5
ハイウォータ・マーク処理 .....	8-36
パスワード .....	7-17, 7-31
パスワードの制限 .....	3-2, 7-17
パスワードの変更 .....	3-11
パスワードの有効期限 .....	3-14, 7-23
パスワードの要件 .....	7-25
パスワード保護 .....	3-16, 7-29
パスワード要件 .....	3-5
保護サブシステム .....	13-1
ログイン・クラスの制限 .....	3-10, 7-3
割り当て時除去 .....	8-36

セキュリティ・クラス・オブジェク ト .....	5-26, 5-28
アクセスのタイプ .....	5-27
監査対象イベント .....	5-27
定義 .....	4-16t
テンプレート・プロファイル ...	5-27
プロファイルの保存 .....	5-28
セキュリティ, クラスタ全体での侵入検 出 .....	11-9
セキュリティ・サーバ・プロセス ..	7-53
セキュリティ上の制限	
作業時間 .....	3-10
時間帯 .....	3-10
ログイン・クラス .....	3-10
セキュリティ侵害への対処 .....	10-7
セキュリティ制限	
キャプティブ・コマンド・プロシー ジャ .....	7-11
コマンドの使用 .....	8-34
時間帯 .....	7-3
シフト .....	7-2
操作モードに関する .....	7-3
セキュリティ・チェックリスト	
C2 システム .....	C-15
安全なシステムを維持するための ..	6-9
安全なシステムを設計するための ..	2-10
ユーザのための .....	3-27
ユーザのトレーニング用 .....	6-6
セキュリティ・ツールとしての	
Accounting ログ .....	10-4
セキュリティに対する攻撃, 形態 ....	1-1
セキュリティに対する攻撃, 形態 ..	10-1
セキュリティの環境要因 .....	1-4
セキュリティ・プロファイル	
アクセス評価 .....	4-17
新しいデバイスへの割り当て .....	5-7
オブジェクト .....	4-11
内容 .....	4-11, 4-14, 4-15
所有者要素 .....	4-12
ACL .....	4-13
ACL の削除 .....	4-31
クラス・テンプレートの変更 ..	8-29
保護コード .....	4-12, 4-33
キュー .....	5-24

クラスのデフォルトの表示 ..... 8-28  
 グローバル・セクション ..... 5-20  
 ケーパビリティ・オブジェクト ... 5-2  
 コモン・イベント・フラグ・クラス  
   タ ..... 5-3  
 資源ドメイン ..... 5-25  
 セキュリティ・クラス ..... 5-27  
 デバイス ..... 5-7  
 ファイル ..... 4-38, 5-10, 5-13  
 プロセス ..... 4-1  
   表示 ..... 4-8, 4-10  
   識別子 ..... 4-6  
   UIC ..... 4-4  
 変更の要件 ..... 4-17, 4-40  
 ボリューム ..... 5-29  
 ユーザ ..... 4-1  
   表示 ..... 4-8, 4-10  
   識別子 ..... 4-6  
   UIC ..... 4-4, 4-6  
 論理名テーブル ..... 5-22  
 セキュリティ・ホール, 処理 ..... 1-1  
 セキュリティ・モデル ..... 2-1  
 セキュリティ・モデルにおけるサブジェク  
   ト ..... 2-2, 2-3  
 セキュリティ問題  
   カテゴリ ..... 1-1  
   自動ログイン・アカウント, 削減 7-14  
   ディスク・スキャベンジング ... 5-16  
   電話システム ..... 10-9  
   ネットワーク・アクセス制御文字  
     列 ..... 3-18  
   ネットワークおよびダイアルアップ・  
     ユーザの匿名性 ..... 7-3  
   パスワード検出 ..... 3-13  
   ハードコピー・ターミナルの出力 3-25  
   ログアウト ..... 3-24, 3-25  
 セキュリティ問題としてのソーシャル・エ  
   ンジニアリング ..... 1-2  
 セキュリティ問題としてのユーザの侵入行  
   為 ..... 1-2  
 セキュリティ問題としてのユーザの詮索行  
   為 ..... 1-2  
 セキュリティ・レベル ..... 1-4  
   イベント監視 ..... 9-12  
   高 ..... 1-3, 3-21

低 ..... 1-3, 3-21  
 中 ..... 1-3  
 接続  
   監査 ..... 9-8  
   動的非同期接続の終了 ..... 12-23  
 接続の監査 ..... 9-8  
 切断ジョブ・メッセージ ..... 3-7  
 ゼロ UIC, 保護チェック ..... 4-18  
 詮索行為, セキュリティ問題 ..... 1-2  
 詮索行為の把握 ..... 10-6  
 詮索行為, 捕捉 ..... 10-3  
 セーブ・セット (BACKUP), 保護 8-38

## た

第 1 パスワード ..... 3-3t  
 第 2 パスワード ..... 3-3t  
   管理 ..... 7-20  
   期限切れのパスワードの変更 ... 3-15  
   最小限の長さ ..... 3-4  
   デメリット ..... 3-3t  
   入力 ..... 3-4  
   変更 ..... 3-14  
   メリット ..... 7-20  
   ログインの時間切れ ..... 3-4  
 ダイアルアップ回線  
   アクセスの制御 ..... 3-3t  
   公共の場での使用 ..... 3-26  
   接続のセキュリティ ..... 12-20  
   動的非同期接続における使用 .. 12-19  
 ダイアルアップ識別子 ..... 4-6t  
 ダイアルアップ・ログイン ..... 3-6  
   再試行 ..... 3-11  
   再試行の制御 ..... 7-46  
   失敗 ..... 3-10  
   接続の切断 ..... 3-26  
 タイムスタンプ  
   クラスタでの同期化 ..... 9-35  
 代理  
   アクセス制御  
     削除 ..... 12-9  
 代理アカウント ..... 3-19, 12-4, 12-12  
   C2 システム ..... C-9  
   キャプティブ・アカウント ..... 12-10  
   許可できる最大数 ..... 3-19  
   推奨される制限 ..... 12-10

制限付きアカウントとして..... 7-16  
 単一ユーザ..... 3-20  
 デフォルト..... 3-20  
 名前の指定..... 3-20  
 汎用アクセス ..... 3-20  
 複数からの選択 ..... 3-20  
 複数ユーザ..... 3-20  
 例..... 12-10, 12-27  
 代理アクセス..... 12-4  
   アクセス制御 ..... 12-3  
   アプリケーション..... 12-9  
   削除 ..... 12-9  
   代理データベースの設定 ..... 12-7  
   ノードへの代理アクセス ..... 12-8  
 代理アクセスの削除 ..... 12-9  
 代理データベース ..... 12-7  
   設定 ..... 12-7  
 代理ログイン..... 3-8, 3-19, 12-4  
   NET\$PROXY.DAT..... 12-7  
   NETPROXY.DAT ..... 12-7  
   アカウント..... 12-4  
   アクセス制御 ..... 12-4  
   セキュリティ上の利点 ..... 3-19  
   設定と管理..... 12-6  
   ネットワーク・アプリケーション 12-5  
 ターミナル  
   C2 システムにおける制限 ..... C-10  
   DECwindows 画面の消去 ..... 3-18  
   アクセスの制御 ..... 3-3t, 7-18  
   アクセスの制限 ..... 8-40  
   応答しない..... 3-4  
   仮想 ... 3-7, 3-26, 5-4, 7-46, 12-20  
   画面の消去..... 3-18, 3-25  
   システム・パスワード  
     要件..... 3-3  
   システム・パスワードの要件 ..... 3-3  
   システム・パスワードを必要とする ..... 3-9  
   使用制限..... 8-39  
   セキュリティ・アラーム ..... 9-19  
   セキュリティ・プロファイルの変更 ..... 5-8  
   セッションのログ取得 ..... 6-7  
   ダイアルアップ接続の切断..... 3-26

ダイアルアップ・ログイン..... 3-6  
 デフォルトのセキュリティ要素 ... 5-7  
 ハードコピー  
   出力の破棄 ..... 3-25  
 ハードコピー，出力の破棄..... 3-25  
 ポート ..... 12-21  
 モデム用の回線，セキュリティ . 8-40  
 ユーザ，C2 システム ..... C-11  
 ログアウトに関する考慮事項 ... 3-25  
 ターミナル・エミュレータ ..... 12-22  
 ターミナル回線 ..... 12-21

## ち

着信代理アクセス，有効化または無効化..... 12-8

## つ

通信デバイス  
   C2 システムの要件..... C-8  
   デフォルトのセキュリティ要素 ... 5-7  
 通知メッセージ ..... 3-4, 3-7e  
   セキュリティ上のデメリット ... 7-45

## て

ディスク  
   削除されたデータへのアクセス . 5-17  
   除去 ..... 5-17, 8-35  
   除去パターン ..... 5-16  
   セキュリティ・プロファイルの管理 ..... 5-8  
   デフォルトのセキュリティ要素 ... 5-7  
   ハイウォータ・マーク処理 5-16, 5-17  
   ファイル削除後の保護 ..... 5-16  
   保護  
     ファイル削除後..... 5-16  
   メッセージ転送レートの変更 ... 9-35  
   割り当て時除去 ..... 5-16, 5-17  
 ディスク・スカベンジング  
   防止 ..... 5-16  
   抑制 ..... 8-35  
 ディスク制限  
   識別子への割り当て..... 8-12

ユーザに対する制限.....	7-4	保護オブジェクト.....	4-16t
ディスクの除去 .....	8-35	デバッグ・サーバ識別子, C2 システムに おける制限.....	C-9
ディスクへのメッセージのフラッ シュ .....	9-36	デフォルトの所有権	
ディスク・ボリューム		ディレクトリ .....	8-27
アクセスの制御 .....	5-28	ファイル.....	8-23
制限 .....	7-4	保護オブジェクト.....	8-19, 8-27
保護 .....	5-28	デフォルトの保護	
ディスク領域		Alpha のシステム・ファイル....	8-32
識別子への割り当て.....	8-25	管理 .....	8-19
使用と割り当て .....	8-12	システム・ファイル.....	B-1
セキュリティ監査ログ・ファイルの要 件 .....	9-36	ディレクトリ .....	5-14
ディレクトリ		ファイル.....	5-13
ACL によるアクセス制御 .....	4-27	プロセス.....	8-19, 8-23
監査対象イベント.....	5-15	デフォルトの保護用 ACE.....	4-37, 8-19, 8-24
作成 .....	5-12	デフォルトのファイル保護の生成	5-13
所有権		例.....	12-27
デフォルトの設定 .....	8-19	テンプレート・デバイスのセキュリティ要 素.....	5-8
資源識別子による .....	8-26	データベース	
ファイルへのアクセス権の変更	8-20	運用時ネットワーク・データベー ス .....	12-21
セキュリティ・プロファイルの割り当 て .....	5-14	クラスタ・システムにおける登録情報 の同期.....	11-6
デフォルトのファイル保護の設定	4-27	権限 .....	8-7
必要なアクセス権.....	5-10, 5-11	登録 .....	2-5, 2-7
ファイルへのアクセス権の制御 .	8-20	保護オブジェクト.....	11-9
ファイルへのアクセスの制御 ...	4-28	テーブル	
ファイル保護の設定.....	8-19	セキュリティ・プロファイルの管 理 .....	5-8
デバイス		デフォルトのセキュリティ要素 ...	5-7
ACE によるアクセスの制御 .....	4-25		
C2 システムにおける再利用.....	C-11		
監査対象イベント.....	5-9		
スプールされた, 必要なアクセス 権 .....	5-6		
セキュリティ・プロファイルの変 更 .....	5-7		
セキュリティ要素.....	5-4		
ターミナルの設定.....	8-40		
デフォルトのセキュリティ要素 ...	5-7		
バックアップ・セーブ・セットの保 護 .....	8-38		
必要なアクセス権.....	5-5		
必要な特権.....	5-9		
テンプレート・セキュリティ・プロファ イル .....	5-7		
プロファイルの保存.....	5-10		

## と

同期, パスワード .....	7-36
動的非同期接続	
手動によるターミナル回線の切り換 え .....	12-22
セキュリティ .....	12-20
接続確立の手順 .....	12-19
接続の終了.....	12-23
接続例 .....	12-23
ターミナル回線の切り換え.....	12-19
ターミナル回線の自動切り換え	12-21
パスワード.....	12-20



ベリファイア .....	12-18	CMEXEC.....	A-5
登録アクセス.....	5-23t	CMKRNL .....	A-6
登録データベース .....	2-5, 2-7	DETACH .....	A-10
アクセス・マトリックス ....	2-7, 2-9	Devour カテゴリ .....	8-15, C-7
監査 .....	9-4	DIAGNOSE .....	A-7
クラスタ・システムにおける登録情報		DOWNGRADE.....	A-7
の同期.....	11-6	EXQUOTA .....	A-8
内容 .....	2-2	GROUP.....	A-8, A-9
変更の監査.....	9-8	Group カテゴリ .....	8-15, C-6
ユーザの追加 .....	7-4	GRPNAM .....	A-8, C-7
登録ユーティリティ ( <b>AUTHORIZE</b> )		GRPPRV .....	4-18, 4-33, 4-39, C-6
ADD/FLAG コマンド .....	7-32	IMPERSONATE .....	A-10
ADD/IDENTIFIER コマンド .....	8-8, 8-25	IMPORT.....	A-11
ADD/PROXY コマンド .	12-10, 12-25	LOG_IO .....	A-11
CREATE/PROXY コマンド.....	12-10	MOUNT .....	A-12
CREATE/RIGHTS コマンド.....	8-7	NETMBX.....	A-12
EXTAUTH フラグ .....	7-32	Normal カテゴリ.....	8-15, C-7
/GENERATE_PASSWORD 修飾		Objects カテゴリ .....	8-15, C-6
子 .....	7-17	OPER.....	4-40, A-12
GRANT/IDENTIFIER コマンド..	8-8, 8-25	PFNMAP.....	A-16
MODIFY/FLAG コマンド .....	7-32	PHY_IO .....	A-16
MODIFY/SYSTEM_PASSWORD コマ		PRMCEB.....	A-18
ンド .....	7-19	PRMGBL.....	A-18
REMOVE/IDENTIFIER コマンド .	8-9	PRMMBX .....	A-18
SHOW/IDENTIFIER コマンド....	8-7	PSWAPM.....	A-19
SHOW/RIGHTS コマンド .....	8-7	READALL.....	4-18, 4-39, A-19
登録ユーティリティの <b>MODIFY</b>		SECURITY.....	A-20
<b>user/FLAG=AUDIT</b> コマンド...	9-8, 9-15	SET PROCESS/PRIVILEGES ...	4-10
登録ユーティリティの <b>MODIFY/SYS-</b>		SETPRV.....	A-20
<b>TEM_PASSWORD</b> コマンド ...	7-19	SETPRV による有効化 .....	4-10
特権		SHARE .....	A-21
ACL の適用回避 .....	4-39	SHMEM.....	A-21
ACNT.....	A-2	SYSGBL.....	A-21
ALLSPOOL .....	A-2	SYSLCK.....	A-21
All カテゴリ .....	C-6	SYSNAM .....	A-22
ALTPRI .....	A-2	SYSPRV .....	4-18
AUDIT .....	A-3	アクセスの制御.....	4-40
BUGCHK .....	A-3	システム・ユーザの権限の付与	4-33
BYPASS .....	4-18, 4-39, A-3	必要とする作業.....	A-22
\$CHECK_PRIVILEGE を用いた使用の		保護メカニズムに対する影響 .	4-39
報告 .....	9-11	TMPMBX .....	A-24
		UAF レコード .....	4-10
		UAF レコードへの格納 .....	8-14
		UPGRADE .....	A-24

VOLPRO ..... A-24  
 WORLD ..... A-25  
 オブジェクト・アクセスに影響を与  
 える ..... 4-18  
 オブジェクト・アクセスへの影響 4-17  
 カテゴリ ..... 8-14, 8-15  
 キャプティブ・アカウント ..... 7-8  
 グループ UIC との関連 ..... 8-1  
 さまざまなユーザに関する推奨事  
 項 ..... 8-16  
 システム・カテゴリ ..... 8-15  
 使用の監査 ..... 3-24t, 9-8  
 信頼できないユーザ ..... C-7  
 信頼できるユーザ ..... C-6  
 セキュリティ管理者の要件 ..... 6-5  
 全カテゴリ ..... 8-15  
 定義 ..... 4-10  
 ネットワークの要件 ..... 12-1  
 ファイルの共用 ..... 12-24  
 プロセス ..... A-1  
 プロセスのデフォルト .... 4-10, 8-14  
 プロセスへの許可 ..... 4-10, 8-14  
 保護コードの適用回避 ..... 4-39  
 無効化 ..... 4-10  
 要件  
   キュー ..... 5-24  
   グローバル・セクション ..... 5-20  
   コモン・イベント・フラグ・クラス  
   タ ..... 5-3  
   デバイス ..... 5-9  
   論理名テーブル ..... 5-22  
   資源ドメイン ..... 5-26  
   ボリューム ..... 5-29  
 要約 ..... 8-15, A-2  
 特権アカウント ..... 7-8, 8-17  
 トレーニング  
   ユーザ用, セキュリティに対する重要  
   性 ..... 6-6  
 トロイの木馬プログラム .... 5-18, 8-30

## に

二重パスワード ..... 7-20  
 入出力操作, デバイスに必要なアクセス  
 権 ..... 5-5  
 入出力チャネル, 必要なアクセス権 . 5-5

任意アクセス制御 ..... A-8, A-22  
 認証, 外部 ..... 7-31  
 認証カード ..... 7-22  
   C2 システムの要件 ..... C-8

## ね

ネットワーク  
   INBOUND パラメータ ..... 12-20  
   アクセス制御 ..... 12-3  
   アプリケーションに対する代理ロギ  
   ン ..... 12-5  
 ネットワーク・アカウント  
   DECNET アカウントの削除 .... 12-17  
   ネットワーク・オブジェクト . 12-14,  
   12-15  
 ネットワーク・アクセス制御文字列 3-16,  
   3-18, 7-21, 12-4  
 ネットワーク, 運用時データベ  
 ース ..... 12-21  
 ネットワーク識別子 ..... 4-6t  
 ネットワーク・セキュリティ ..... 3-18,  
   12-1, 12-24  
   C2 システム ..... C-9  
   監査対象イベント ..... 12-3  
   制限 ..... 12-1  
   ネットワーク・オブジェクトの設  
   定 ..... 12-14, 12-15  
   要件 ..... 12-2  
 ネットワーク・データベース ..... 12-21  
 ネットワーク・ログイン ..... 3-5, 3-8

## の

ノード・タイプを指定する **INBOUND** パ  
 ラメータ ..... 12-20  
 ノードのタイプ ..... 12-20

## は

ハイウォータ・マーク処理 ..... 5-16,  
   5-17, 8-36  
   C2 システム ..... C-11  
   パフォーマンス ..... 8-37  
 媒体の初期化  
   必要なアクセス権 ..... 5-28

パス、デフォルトのセキュリティ要	
素.....	5-7
パスワード	
アカウントごとの一意性.....	3-17
アカウント作成時.....	3-2
誤り.....	3-7
ダイアルアップ中に入力できる回	
数.....	3-10
暗号化アルゴリズム.....	7-26
安全ターミナル・サーバ.....	3-17
安全な.....	3-1
安全な選択.....	3-1
エンコード.....	2-4
オープン・アカウント.....	3-5
危険.....	3-1
キャプティブ・アカウントの選択肢と	
してのヌル.....	7-10t
強制変更.....	3-15, 7-24
共用.....	3-16, 12-24
許容可能.....	3-2
クラスタ所属の管理.....	11-10
形式.....	3-1
検査	
サイト固有のフィルタの使用.....	7-28
辞書との照合.....	7-27
履歴リストとの照合.....	7-27
検出の回避.....	3-13, 10-7
コンソール	
C2 システムの要件.....	C-7
コンソール・パスワード.....	7-21
最小限の長さ.....	3-2, 3-11
最初の.....	3-2
最低限の長さ.....	7-25
再利用.....	3-1
システム.....	3-4, 7-18
辞書.....	3-2
ガイドライン.....	7-19
推奨される変更頻度.....	7-24
設定.....	7-18
デメリット.....	7-20
必要な最低限の長さ.....	7-25
変更.....	7-19
ログイン失敗の原因.....	3-9
事前に期限切れにする方法.....	7-18
自動生成.....	3-12
受信.....	12-20
初期.....	3-2, 7-17
新規.....	3-14
推測.....	3-2
制限.....	7-23
制限事項.....	3-2
生成.....	3-13, 7-17
送信.....	12-20
第 1.....	3-4, 7-17
第 2.....	3-3t, 7-20
入力.....	3-4
変更.....	3-14
管理.....	7-20
メリット.....	7-20
期限切れの変更.....	3-15
ダイアルアップの再試行 ..	3-10, 3-11
タイプ.....	3-2
代理ログイン.....	3-19
長さ.....	3-2, 7-25
二重.....	3-3t, 7-17
ネットワーク用パスワードの削除	12-6
パスワード・グラバ・プログラム	3-17
複数のシステム.....	3-17
変更.....	3-11
第 2.....	3-14
/NEW_PASSWORD 修飾子の使	
用.....	3-14
頻度に関するガイドライン ...	3-17
期限切れ.....	3-15
ログイン時.....	3-14
変更確認.....	3-12
変更しない.....	3-15
変更するタイミング.....	3-2
変更の理由.....	3-21, 3-24
有効期限....	3-12, 3-14, 3-15, 7-23
ユーザ.....	2-4, 3-3
ユーザのためのガイドライン .....	3-1
履歴リスト.....	3-2
ルーティングの初期化.....	12-18
ロック.....	7-26
ロックされた.....	7-10t
ロック済み.....	3-5

パスワード・グラバ・プログラム	3-17, 7-47
監査 ACE を使用した検出	9-7
パスワード・ジェネレータ	
使用の要件	7-26
初期パスワードの取得	7-17
パスワード同期	7-36
パスワードの有効期限	3-12, 3-14
パスワード保護	3-16, 7-29
パスワード履歴	7-27
バックアップ操作	
一般的な推奨事項	8-37
キャプティブ特権アカウントからの実行	7-8
バックアップ・ユーティリティ (BACKUP)	
一般的な推奨事項	8-37
キャプティブ特権アカウントからの実行	7-8
バッチ識別子	4-6t
バッチ・ジョブ	
作業時間制限による影響	3-10
登録	3-8
パスワード保護とカード・リーダー	3-17
バッチ・ログイン	3-8
パフォーマンス	
ACL の長さ	8-5
ハイウォータ・マーク処理	8-37
汎用識別子	4-24
形式	4-7t
設計の考慮事項	8-3
例	4-8, 4-27
ハードコピー出力	
破棄	3-25
ハードコピー出力の破棄, システム障害	3-25
ハードコピー・ターミナル, 出力に関する考慮事項	3-25

## ひ

非会話型ログイン	3-5, 3-8
クラス	3-8
ネットワーク	3-8
バッチ	3-8

非共用可能デバイス, 必要なアクセス権	5-7
必要なアクセス権	
キュー	5-23
共用可能デバイス	5-7
グローバル・セクション	5-19
ケーパビリティ・オブジェクト	5-2
コモン・イベント・フラグ・クラス	
タ	5-3
資源ドメイン	5-25
スプールされたデバイス	5-6
セキュリティ・クラス・オブジェクト	5-27
ディレクトリ	5-11
デバイスの割り当て	5-5
入出力チャンネル	5-5
非共用可能デバイス	5-7
非ファイル指向デバイス	5-6
ファイル	5-11
ファイル指向デバイス	5-5
ボリューム	5-6
論理名テーブル	5-21
必要な特権	
キュー	5-24
グローバル・セクション	5-20
コモン・イベント・フラグ・クラス	
タ	5-3
資源ドメイン	5-26
デバイス	5-9
ボリューム	5-29
論理名テーブル	5-22
非同期 DDCMP ドライバ	12-19
非同期接続, 動的	12-23
非ファイル指向デバイス, 必要なアクセス権	5-6, 5-7

## ふ

ファイル	
ACL によるアクセス制御	4-27
MAIL を使用した転送	12-24
アクセス	
ファイル識別子	5-12
割り当てられたディスク・ブロック	5-17
アクセスの監査	3-21, 4-40

アラームの適用 .....	3-21	ファイルの保護 .....	4-11, 5-10
暗号化 .....	8-35	デフォルト ACL の設定 .....	4-27
監査対象イベント .....	5-15	ファイル保護 .....	8-19
クラスタ・システムにおける共用	11-7	C2 システム .....	C-6
コピー		DCL コマンド .....	8-32
遠隔アカウントから .....	3-20	監査 .....	10-6
削除後のデータの保護 .....	5-16	フォーリン・ボリューム, 必要なアクセス	
作成		権 .....	5-6
要件 .....	5-12	復号 .....	8-35
ディレクトリの所有権に対する依存		物理的セキュリティ .....	1-4
関係 .....	8-20	C2 システム .....	C-7
識別子用 ACE によるアクセスの制		違反の兆候 .....	10-2
御 .....	4-24	システムへのアクセスの制限 .....	8-1
資源識別子により所有される ...	8-26	ファイルの暗号化 .....	8-35
資源識別子による所有 .....	5-14	ログアウト時 .....	3-24, 3-25
重要, 保護 .....	3-22	物理入出力アクセス .....	5-5t
所有権規則の例外 .....	4-12	フラッシュの間隔 .....	9-35
所有権の規則 .....	5-13	プリンタ	
セキュリティ監査 .....	3-21, 5-15	C2 システム .....	C-12
セキュリティ監査に必要な ACE の追		デフォルトのセキュリティ要素 ...	5-7
加 .....	3-22, 4-41	プロジェクト・アカウント .....	8-25
セキュリティの最適化 .....	5-18	設定 .....	8-26
セキュリティ・プロファイルの変		保護サブシステムとしてのプロジェク	
更 .....	5-14	ト・アカウント .....	13-2
セキュリティ・プロファイルの割り当		プロセス	
て .....	5-13, 8-20	UIC 識別子 .....	4-6
セキュリティ要素 .....	5-10	アクセス権 .....	4-1
代理アクセスに必要な保護 .....	3-20	一時中断 .....	9-34
ディスクからのデータの除去 ...	5-16	会話モード .....	3-5
ディレクトリのデフォルトの管理	8-27	監査 .....	9-8
デフォルトのセキュリティ要素の復		監査サーバ .....	9-30
元 .....	4-32	現在のプロセスのログアウト ...	3-26
デフォルトの保護 .....	4-37	異なる UIC を持つものの作成 .....	4-6
デフォルトの保護と所有権の設定	8-19	再接続 .....	3-7
ネットワーク環境における共用と交		除外リストへの追加 .....	9-34
換 .....	12-24, 12-29	制御するシステム・サービスの監	
必要なアクセス権 .....	5-10, 5-11	査 .....	9-8
デフォルトのセキュリティ・プロファ		セキュリティ・プロファイル .....	4-1
イルの復元 .....	4-38	接続の制限事項 .....	3-7
保護オブジェクト .....	4-16t	切断 .....	3-26
保護コードの割り当て .....	5-13	切断された .....	3-7
命名規則 .....	5-10	デフォルトの保護 .....	5-14
メールの保護 .....	5-18	デフォルトの保護の表示 .....	5-14
ファイル閲覧者 .....	3-22, 10-6, 10-8	特権により許可される処理 .....	8-14

特権の有効化 .....	4-10
プロセス・ライト識別子の表示 ...	4-8
ライト・リストの変更 .....	8-14
プロセス除外リスト .....	9-34
プロセスのスポン、制限付きアカウ ントにおけるセキュリティの考慮事 項.....	7-10t
プロセスへの再接続 .....	7-46
プロトコル	
autodial/master .....	12-21
プロトコル, <b>autodial/nomaster</b> .....	12-21
プロンプト, パスワード .....	3-4

## へ

エコーバック, パスワード .....	3-4
---------------------	-----

## ほ

### 保護

ACL ベース .....	8-25
UIC ベースのコード .....	4-12
オブジェクト .....	4-11
キュー .....	5-24
グローバル・セクション .....	5-20
ケーパビリティ .....	5-2
コマンド・プロシージャ .....	8-31
コモン・イベント・フラグ・クラス タ .....	5-3
削除されたデータ .....	5-16, 5-17
資源ドメイン .....	5-25
セキュリティ・クラス .....	5-27
デバイス .....	5-7
デフォルトの管理 .....	8-19, 8-23
保護サブシステムの利用 .....	13-1
ボリューム .....	5-29
論理名テーブル .....	5-22
保護オブジェクト・データベース .	11-9
保護コード .....	B-1
ACL とのやり取り .....	4-37
アクセス指定 .....	4-34
アクセス・タイプ .....	4-34
アクセス評価における優先順位 .	4-17
オブジェクトのセキュリティ要素	4-11
解釈 .....	4-13
カテゴリをチェックする順序 ...	4-35

キューに対するアクセス権 .....	5-24
形式 .....	4-33
識別子用 ACE .....	4-25
処理 .....	4-35
すべてのアクセスの拒否 .....	4-37
制御アクセスの許可 .....	4-35
定義 .....	2-6, 4-12
デフォルトのファイル保護	4-37, 8-23
特権による適用回避 .....	4-39
特権の効果 .....	4-18
ヌル・アクセス指定 .....	4-34
評価手順 .....	4-13
ファイル作成時の割り当て .....	8-20
ファイルのデフォルト値の復元 .	4-38
複数のユーザ・カテゴリ .....	4-35
変更 .....	4-36
ユーザ・カテゴリ .....	4-13
読み込み .....	4-35

### 保護サブシステム

構築 .....	13-5
サブシステム ACE .....	13-5
システム管理の要件 .....	13-4
設計の要件 .....	13-3
説明 .....	13-2, 13-8
適用範囲 .....	13-2
ファイル保護 .....	13-10, 13-11
プリンタの保護 .....	13-12
ボリュームのマウント .....	13-6
メリット .....	13-1
有効化 .....	13-6
ユーザ・アクセス .....	13-7
例 .....	13-8
保護チェック .....	4-17
オブジェクト・アクセス要求の評 価 .....	4-17
ゼロの UIC に関する例外 .....	4-17
保護の継承例 .....	12-27
保護のチェック	

所有権による影響 .....	8-20
ボリューム	

C2 システムにおける再利用 .....	C-11
アクセスのタイプ .....	5-28
監査対象イベント .....	5-29
セキュリティ要素 .....	5-28
テンプレート・プロファイル ...	5-29
データの除去 .....	8-36

必要なアクセス権 .....	5-6
必要な特権 .....	5-29
フォーリン	
必要なアクセス権 .....	5-6
プロファイルの保存 .....	5-29
保護 .....	5-28
保護オブジェクト .....	4-17t
マウントまたはデスマウントの監 査 .....	9-8
ボリュームのマウント	
セキュリティ監査 .....	3-24
必要なアクセス権 .....	5-28
保護サブシステム .....	13-6
ポート, ターミナル .....	12-21

## め

命名規則	
キュー .....	5-23
グローバル・セクション .....	5-19
ケーパビリティ・オブジェクト ...	5-1
コモン・イベント・フラグ・クラス タ .....	5-3
資源ドメイン .....	5-25
セキュリティ・クラス .....	5-26
デバイス .....	5-4
ファイル .....	5-10
論理名テーブル .....	5-21
メッセージ	
ウェルカム .....	3-7
監査 .....	9-2
最後の正常な会話型ログイン .....	3-7
最終ログイン情報の無効化 .....	7-46
最終ログインに関する表示の抑制	3-21
セキュリティ関連イベントの監査	3-23
通知 .....	3-7e
セキュリティ上のデメリット .	7-45
表示の抑制 .....	3-8
抑止 .....	7-45
ログイン .....	3-6
ログインの失敗 .....	3-21
メディアの初期化	
ACL を使用した制限 .....	8-34
メモリ消費, ACL による .....	8-5

メンバ <b>UIC</b> 名 .....	4-4
メール・ファイル, 推奨される保護	5-18
メールボックス	
監査イベント・メッセージ用 ...	9-16
セキュリティ・プロファイルの変 更 .....	5-8
デフォルトのセキュリティ要素 ...	5-7
デフォルトの保護 .....	C-5
必要な特権 .....	5-9
メール・ユーティリティ ( <b>MAIL</b> )	
テキスト・ファイルの転送 .....	12-24

## も

モデム .....	12-19
C2 システムの要件 .....	C-7

## ゆ

有効期限	
アカウント .....	3-16
第 2 パスワード .....	3-15
パスワード .....	3-15, 7-18
パスワードに関するシステム・メッセー ジ .....	3-14, 3-15
ユーザ	
ACE によるアクセス .....	4-24
C2 システム .....	C-9
アクセスの要求 .....	4-18
権限の表示 .....	8-7
システムの紹介 .....	6-6
信頼できない .....	C-7
信頼できる .....	C-6, C-11
セキュリティ・カテゴリ ..	4-13, 4-34
セキュリティ・プロファイル .....	4-1
デフォルトのオブジェクト保護の設 定 .....	8-19
特権の付与 .....	8-14
トレーニング .....	6-6
ファイル・セキュリティ .....	5-18
プロセス・ライト識別子の表示 ...	4-8
保護コードのカテゴリ .....	4-33
ユーザ・アカウント .....	6-7
セキュリティの考慮事項 .....	7-4
ユーザ記述のシステム・サービス	

保護サブシステムによる代替 ...	13-1
ユーザ登録	
アカウントの有効期限 .....	3-16
作業時間制限 .....	3-10
特権の使用.....	4-10
ログイン・クラスの制限 .....	3-10
ユーザ・トレーニング.....	6-6
ユーザのトレーニング, セキュリティに対する重要性.....	6-6
ユーザの無責任行為	
セキュリティ問題としての.....	1-1
防御手段としてのトレーニング ...	6-6
ユーザ名	
識別子 .....	2-4, 4-7t
ユーザ名マッピング .....	7-35

## よ

曜日によるログイン制限 .....	3-10
読み込みアクセス	
ACL による許可 .....	4-28
キュー	
ACL.....	5-23t
保護コード .....	5-23t
グローバル・セクション .....	5-19t
資源ドメイン .....	5-25t
セキュリティ・クラス .....	5-27t
デバイス.....	5-5t
ファイル.....	5-10
保護コードによる許可 .....	4-34
ボリューム.....	5-28
論理名テーブル .....	5-21
予約済み UIC グループ番号 .....	4-5

## ら

ライト識別子の保持者	
アクセスの許可 .....	4-24
識別子との関連付け.....	8-8
ライト・データベースからの削除 ..	8-8
レコードの表示 .....	8-7
ライト識別子の保持者を表示するレコー	
ド.....	8-7
ライト・データベース	
作成と保守.....	8-6
識別子と保持者の削除 .....	8-9

識別子の追加 .....	8-7
表示 .....	8-7
ユーザへの識別子の割り当て .....	8-8
ライト・リスト	
ケーパビリティ単位で整理されたアク	
セス .....	2-9
ライト・リスト, ケーパビリティ単位で整	
理されたアクセス.....	2-9

## り

リコール・バッファ .....	3-18
リスナ・デバイス, プログラムの例	
9-22	
リファレンス・モニタ.....	2-1
実装 .....	2-3
セキュリティにおける概念..	2-1, 2-7
ネットワークへの適用 .....	12-2
要件 .....	2-2
履歴.....	7-27

## る

ルーティング初期化パスワード...	12-18
-------------------	-------

## ろ

ログアウト	
セキュリティに関する考慮事項 ..	3-24, 3-25
切断されたプロセスから .....	3-26
ダイアルアップ接続の切断.....	3-26
必要となる状況の判断 .....	3-24
理由 .....	3-24
ログアウト・アラーム.....	D-6
ログアウトの監査 .....	9-8
ログイン	
安全ターミナル・サーバ..	3-17, 7-47
遠隔 .....	3-6
システム・パスワード.....	7-19
ログアウト .....	3-25
外部認証の使用 .....	3-6
会話型 .....	3-5
クラス.....	3-6
最後の.....	3-7
監査 .....	9-8
期限切れアカウント.....	3-16



許可される時間帯 .....	3-10	再試行 .....	3-11
最終の監視 .....	3-21	作業時間制限 .....	3-10
時間切れ .....	3-4	システム・パスワード .....	3-9
システム・パスワードを使用した制限 .....	7-19	侵入回避 .....	3-11
実行時のパスワード変更 .....	3-14	セキュリティ監査レポート .....	9-28
制御 .....	3-3t	ダイアルアップ・ログイン .....	3-10
制御するシステム・パラメータ .....	7-52	パスワード・グラバ・プログラム .....	3-17
セキュリティへの影響 .....	3-2	メッセージ .....	3-7, 3-21
ダイアルアップ .....	3-6	ログイン・クラスの制限 .....	3-10
パスワードの入力 .....	3-10	ログイン・メッセージ .....	3-6
ネットワーク .....	3-8	ウェルカム .....	3-7
パスワードの変更 .....	3-2	期限切れパスワード .....	3-14, 3-15
パッチ .....	3-8	最後の正常な会話型ログイン .....	3-7
非会話型 .....	3-5	最後の正常な非会話型ログイン ...	3-7
クラス .....	3-8	新規メール .....	3-7
最後の .....	3-7	制御 .....	7-46
フラグ .....	7-24	切断されたジョブ .....	3-7
プロセスのデフォルトの保護 ...	5-14	通知 .....	3-7e
無効		表示の抑制 .....	3-8, 3-21
作業時間制限によって .....	3-10	ログイン失敗回数 .....	3-7
侵入回避による .....	3-11	ログ取得	
ユーザのための ALF (自動ログイン機能) を使用した簡素化 .....	7-14	セキュリティ監査イベント .....	9-2, 9-16
ローカル .....	3-6	ターミナル・セッション .....	6-7
ログイン・アラーム .....	D-5	保護オブジェクトへのアクセス .....	4-40
有効化 .....	9-8	ロック・アクセス .....	5-25t
ログイン・クラス .....	3-5	論理入出力アクセス .....	5-5t
遠隔 .....	3-6	論理名	
会話型 .....	3-6	外部認証用の定義 .....	7-31
制限事項 .....	3-10	論理名テーブル	
ダイアルアップ .....	3-6	アクセスのタイプ .....	5-21
ネットワーク .....	3-8	監査対象イベント .....	5-22
パッチ .....	3-8	セキュリティ・プロファイルの再設定 .....	5-23
非会話型 .....	3-8	セキュリティ要素 .....	5-21
ローカル .....	3-6	テンプレート・プロファイル ...	5-22
ログイン・コマンド・プロシージャ		必要な特権 .....	5-22
制限付きアカウント用 .....	7-9, 7-11	保護オブジェクト .....	4-16t
適切な保護 .....	8-31	ローカル識別子 .....	4-6t
ログインの失敗			
アラーム .....	D-5		
監査 .....	9-8		
期限切れアカウント .....	3-16		
原因 .....	3-9		

## わ

ワイルドカード文字	
ADD/IDENTIFIER コマンド .....	8-8
SHOW/RIGHTS コマンド .....	8-7

割り当て時除去 ..... 5-16, 5-17  
ワークステーション  
画面の消去..... 3-25

デフォルトのセキュリティ要素 ... 5-7  
ワールド・ユーザ(セキュリティ・カテ  
リ) ..... 4-13, 4-34