

Oracle® Solaris 11.1 でのサービスと障害 の管理

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

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

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

U.S. GOVERNMENT END USERS:

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

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

OracleおよびJavaはOracle Corporationおよびその関連企業の登録商標です。その他の名称は、それぞれの所有者の商標または登録商標です。

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

このソフトウェアまたはハードウェア、そしてドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても一切の責任を負いかねます。

目次

はじめに	7
1 サービスの管理 (概要)	9
このリリースの SMF について	9
SMF について	10
SMF を使用する利点	11
SMF の概念	11
SMF サービス	11
SMF の依存関係	12
サービス識別子	13
サービスの状態	14
SMF マニフェスト	14
SMF プロファイル	15
サービス構成リポジトリ	16
SMF 管理レイヤー	16
SMF リポジトリのバックアップ	18
SMF スナップショット	18
SMF サービスのエラーログ	18
SMF 管理およびプログラミングインタフェース	19
SMF コマンド行管理ユーティリティ	19
サービス管理構成ライブラリインタフェース	20
SMF コンポーネント	20
SMF マスターリスタータデーモン	20
SMF 委任リスタータ	21
SMF プロパティとプロパティグループ	21
サービス構成リポジトリ内の情報の管理	21
SMF 情報の表示	22
SMF 情報の変更	22

SMF 情報の削除	23
SMF とブート	23
SMF の互換性	24
実行レベル	24
実行レベルとマイルストーンの使い分け	26
システムの実行レベルを確認する	26
/etc/inittab ファイル	27
システムが実行レベル3になると実行される処理	28
2 サービスの管理(タスク)	29
SMF サービスのモニタリング	29
サービスのモニタリング(タスクマップ)	29
▼ サービスのステータスを一覧表示する方法	30
▼ サービスのカスタマイズを一覧表示する方法	31
▼ 特定のサービスに依存するサービスを表示する方法	31
▼ 特定のサービスが依存するサービスを表示する方法	32
▼ SMF 遷移イベントの通知を設定する方法	33
SMF サービスの管理	34
SMF サービス(タスクマップ)	34
RBAC 権利プロファイルを使って SMF を管理する	35
▼ SMF サービスを作成する方法	35
▼ サービスインスタンスを無効にする方法	36
▼ サービスインスタンスを有効にする方法	37
▼ サービスを再起動する方法	38
▼ 保守状態のサービスを復元する方法	38
▼ SMF プロファイルを作成する方法	39
▼ SMF プロファイルを手動で作成する方法	40
▼ SMF プロファイルを適用する方法	41
SMF サービスの構成	41
SMF サービスの構成(タスクマップ)	41
▼ SMF サービスプロパティの変更方法	42
▼ 1つのサービスの複数プロパティを変更する方法	42
▼ ファイルで構成されているサービスの変更方法	43
▼ サービスの環境変数を変更する方法	44
▼ inetd 制御サービスのプロパティを変更する方法	45

▼ サービスのカスタマイズを削除する方法	46
▼ inetd 制御サービスのコマンド行引数を変更する方法	47
▼ inetd.conf エントリを変換する方法	47
実行制御スクリプトの使用	48
実行制御スクリプトの使用 (タスクマップ)	48
▼ 実行制御スクリプトを使用して従来のサービスを起動または停止する方法	49
▼ 実行制御スクリプトを追加する方法	49
▼ 実行制御スクリプトを無効にする方法	50
▼ 実行制御スクリプトを SMF サービスに変換する方法	51
サービス管理機能のトラブルシューティング	52
SMF のトラブルシューティング (タスクマップ)	52
▼ 起動していないサービスをデバッグする方法	53
▼ 破壊されたりポジトリを修復する方法	53
▼ どのサービスも起動しないでブートする方法	56
▼ 詳細モードでブートする方法	56
▼ ブート中に <code>system/filesystem/local:default</code> サービスで障害が発生した場合に <code>sulogin</code> プロンプトを強制的に表示する方法	57
3 障害管理の使用	59
障害管理の概要	59
障害および欠陥の通知	61
障害または欠陥に関する情報の表示	62
▼ 障害の発生したコンポーネントに関する情報を表示する方法	62
▼ CPU がオフラインかどうかを特定する方法	64
▼ 欠陥のあるサービスに関する情報を表示する方法	65
障害または欠陥の修復	66
<code>fmadm replaced</code> コマンド	66
<code>fmadm repaired</code> コマンド	67
<code>fmadm acquit</code> コマンド	67
障害管理のログファイル	67
障害の統計情報	68
索引	69

はじめに

『Oracle Solaris 11.1 でのサービスと障害の管理』は、Oracle Solaris システム管理に関する重要な情報を提供するドキュメントセットの一部です。このガイドでは、Oracle Solaris のサービス管理機能 (SMF) と、障害管理アーキテクチャー (FMA) に焦点を当てています。

注 - この Oracle Solaris のリリースでは、SPARC および x86 系列のプロセッサアーキテクチャーを使用するシステムをサポートしています。サポートされるシステムについては、[Oracle Solaris OS: Hardware Compatibility Lists](#) を参照してください。このドキュメントでは、プラットフォームにより実装が異なる場合は、それを特記します。

Oracle サポートへのアクセス

Oracle のお客様は、My Oracle Support を通じて電子的なサポートを利用することができます。詳細は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> を参照してください。聴覚に障害をお持ちの場合は、<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> を参照してください。

表記上の規則

次の表では、このドキュメントで使用される表記上の規則について説明します。

表 P-1 表記上の規則

字体	説明	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 machine_name% you have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	machine_name% su Password:

表 P-1 表記上の規則 (続き)

字体	説明	例
<i>aabbcc123</i>	プレースホルダ: 実際使用する特定の名前または値で置き換えます。	ファイルを削除するには、 <code>rm filename</code> と入力します。
AaBbCc123	書名、新しい単語、および強調する単語を示します。	『ユーザーズガイド』の第 6 章を参照してください。 キャッシュは、ローカルに格納されるコピーです。 ファイルを保存しないでください。 注: いくつかの強調された項目は、オンラインでは太字で表示されます。

コマンド例のシェルプロンプト

Oracle Solaris OS に含まれるシェルで使用する、UNIX のシステムプロンプトとスーパーユーザープロンプトを次に示します。コマンド例で、シェルプロンプトは、一般ユーザーまたは特権を持つユーザーのどちらがコマンドを実行するのかを示します。

表 P-2 シェルプロンプト

シェル	プロンプト
Bash シェル、Korn シェル、および Bourne シェル	\$
Bash シェル、Korn シェル、および Bourne シェルのスーパーユーザー	#
C シェル	machine_name%
C シェルのスーパーユーザー	machine_name#

サービスの管理 (概要)

この章では、Oracle Solaris のサービス管理機能 (SMF) の概要について説明します。また、実行レベルに関する情報も提供します。

この章の内容は次のとおりです。

- 10 ページの「SMF について」
- 11 ページの「SMF の概念」
- 19 ページの「SMF 管理およびプログラミングインタフェース」
- 20 ページの「SMF コンポーネント」
- 24 ページの「SMF の互換性」
- 24 ページの「実行レベル」
- 27 ページの「/etc/inittab ファイル」

SMF に関連する手順については、29 ページの「サービスのモニタリング (タスクマップ)」を参照してください。実行レベルに関連する手順については、48 ページの「実行制御スクリプトの使用 (タスクマップ)」を参照してください。

このリリースの SMF について

このリリースには次の新しい SMF 機能が追加されました。

- `svccfg editprop` コマンドを使用した SMF プロパティ編集が、次の改善により強化されました。
 - インスタンスが選択されたときに合成ビューが表示されます。また、たとえばインスタンスまたはサービスのいずれかといった、プロパティが定義されている場所も表示されます。
 - デフォルトでは、`editprop` サブコマンドは、メソッド、依存関係、または SMF テンプレート定義などの SMF インフラストラクチャプロパティを表示しません。

詳細は、42 ページの「1つのサービスの複数プロパティを変更する方法」および Oracle Solaris 11 Update 1 システムの `svccfg(1M)` のマニュアルページを参照してください。

- `svccprop` コマンドが変更され、SMF テンプレート定義が出力に表示されなくなりました。必要に応じてテンプレート定義を含めるための `-a` オプションがあります。詳細は、Oracle Solaris 11 Update 1 システムの `svccprop(1)` のマニュアルページを参照してください。
- `svcbundle` コマンドがこのリリースに追加されました。このコマンドにより、SMF マニフェストおよびプロファイルを作成できます。詳細は、35 ページの「SMF サービスを作成する方法」、39 ページの「SMF プロファイルを作成する方法」および Oracle Solaris 11 Update 1 システムの `svcbundle(1M)` のマニュアルページを参照してください。

SMF について

SMF により、アプリケーションおよびシステムサービスの管理が容易になります。フレームワークインフラストラクチャーによって、従来の UNIX 起動スクリプト、`init` 実行レベル、および構成ファイルが補強されます。SMF は、アプリケーション間またはサービス間の関係を定義し、依存するサービスを必要なときに自動的に再起動できるようにするためのメカニズムを提供します。各サービスを管理するために必要な情報はサービス構成リポジトリに格納されます。このリポジトリは、各サービスを管理するための簡易な手段を提供します。

SMF では、管理者がサービスに対して呼び出すことのできる一連のアクションを定義します。`svcadm` コマンドによって手動で操作できるこれらのアクションには、有効化、無効化、リフレッシュ、再起動、マークなどがあります。各サービスは、管理アクションを実行するサービスリスタータによって管理されます。通常、アクションを実行する場合、リスタータはサービスに対していくつかのメソッドを実行します。各サービスのメソッドは、サービス構成リポジトリで定義されます。リスタータは、これらのメソッドを使って、サービスをある状態から別の状態へ移行できます。

サービス構成リポジトリでは、フォールバックができるように、各サービスが正常に起動されたときにサービスごとのスナップショットを取ります。また、リポジトリを使用すると、一貫した永続的な方法でサービスを有効または無効にしたり、サービスの状態を一貫して表示したりできます。この機能は、サービスの構成に関する問題を修正するのに役立ちます。

SMF を使用する利点

SMF によって提供されるフレームワークにより、サーバー上で実行中のすべてのサービスの管理が容易になります。具体的には、フレームワークによって次のことが提供されます。

- 失敗したサービスを、それが管理者エラーまたはソフトウェアバグの結果として失敗したか、修正不能なハードウェアエラーの影響によるかどうかにかかわらず、依存関係の順序に従って自動的に再起動します。依存関係の順序は依存関係ステートメントによって定義されます。
- サービスオブジェクトを `svcs` コマンドで表示でき、`svcadm` および `svccfg` コマンドで管理できます。また、`svcs -p` を使用すると、SMF サービスとレガシーの `init.d` スクリプトの両方について、サービスとプロセスの間の関係を表示することもできます。

`svcadm` コマンドは、サービスを有効または無効にすることも可能です。これらの変更は、アップグレードおよびリブート後も持続します。`-t` オプションを使用した場合、変更は一時的です。

- `svcs -x` を使用することで、サービスが実行されない理由の説明を提供することにより、サービスに関するデバッグおよび質問の作成が簡単になります。また、この処理はサービスごとの個別および永続的なログファイルによって容易になります。
- 管理者が `root` 以外のユーザーにタスクを安全に委任するための能力が高まり、これにはプロパティを変更したり、システム上のサービスを有効化、無効化、または再起動したりする能力が含まれます。
- サービスの依存関係に従ってサービスを並列で起動することにより、大規模システムで高速にブートします。シャットダウン中は逆の処理が実行されます。
- 可能な場合には既存の管理方法との互換性を維持します。たとえば、顧客および ISV から提供されるほとんどの `rc` スクリプトは、通常どおり機能します。

SMF の概念

このセクションでは、SMF フレームワーク内で使われる用語とその定義をいくつか紹介します。これらの用語は、ドキュメント全体で使用されます。SMF の概念を理解するには、これらの用語を理解する必要があります。

SMF サービス

SMF フレームワークでの基本的な管理単位は「サービスインスタンス」です。各 SMF サービスはシステム上で複数回実行でき、わずかに異なる構成の要因となります。これらの異なる構成は、サービスインスタンスと呼ばれます。各インスタンス

はサービスの特定の構成です。たとえば、Web サーバーはサービスです。ポート 80 で待機するよう構成された Web サーバーデーモンはインスタンスです。Web サーバーサービスの各インスタンスには、異なる構成要件を設定できます。サービスにはシステム全体の構成要件が設定されていますが、各インスタンスでは必要に応じて特定の要件をオーバーライドできます。1 つのサービスの複数のインスタンスは、サービスオブジェクトの子オブジェクトとして管理されます。

サービスは、`in.dhcpd` や `nfsd` などの標準の長年続いているシステムサービスだけではなく、ISV アプリケーションを含むさまざまなシステムエンティティーも表します。また、次のような今まであまり使われなかったエンティティーを表すこともできます。

- 物理ネットワークデバイス
- IP アドレス構成
- カーネル構成情報
- マルチユーザー実行レベルなど、システムの `init` 状態に対応するマイルストーン

一般に、サービスはアプリケーションやほかのサービス(ローカルやリモート)に機能リストを提供するエンティティーです。サービスは、暗黙的および明示的に宣言されたローカルサービスのリストに依存しています。

マイルストーンは特別な種類のサービスです。マイルストーンサービスは、システム即応性のレベルを表します。たとえば、実行レベルは SMF ではマイルストーンによって表されます。また、マイルストーンを使用すると、ネームサービスの場合の `svc:/milestone/name-services:default` や `sysconfig` サービスの場合の `svc:/milestone/config:default` のようなサービスグループの即応性を示すこともできます。

SMF の依存関係

依存関係は、サービス間の関係を定義します。これらの関係により、すべてのサービスを再起動するのではなく、障害の影響を直接受けているサービスのみを再起動することにより、障害を的確に封じ込めることができます。依存関係により、スケラブルで再現可能な初期化プロセスも実現します。最後に、正確な依存関係の定義によって、すべての独立したサービス群を並列的に起動できるため、並列性の高い最新のマシン群をシステムのスタートアップで利用することが可能になります。

サービスの再起動動作は、依存関係ごとの `restart_on` 属性によって定義されます。依存するサービスがエラーや別の理由で停止した場合、あるいはリフレッシュされた場合に停止するようサービスを構成できます。このプロセスによってサービスが停止されたあと、依存するサービスが起動するとすぐに、サービスは自動的に再起動されます。たとえば、`ssh` サービスには、`network/ipfilter` サービスに対する依存関係があります。`restart_on` 属性は `error` に設定されてお

り、この場合、`network/ipfilter` サービスがエラーによって停止すると、`ssh` サービスが停止され、自動的に再起動されます。ほかの種類のイベントが発生した場合、`ssh` サービスは停止されません。

サービス識別子

各サービスインスタンスの名前は、障害管理リソース識別子 (FMRI) によって付けられます。FMRI には、サービス名とインスタンス名が含まれます。たとえば、`rlogin` サービスの FMRI は `svc:/network/login:rlogin` となり、ここで、`network/login` はサービスを、`rlogin` はサービスインスタンスをそれぞれ示します。

次の FMRI の形式はどれも同じです。

- `svc://localhost/system/system-log:default`
- `svc:/system/system-log:default`
- `system/system-log:default`

また、多くの SMF コマンドでは、あいまいさがない場合に、省略したサービス名またはインスタンス名を使用できます。たとえば、長い形式ではなく `system-log` を直接使用できます。どの FMRI 形式が適切かについては、[svcadm\(1M\)](#) または [svcs\(1\)](#) などの SMF コマンドのマニュアルページを参照してください。

サービス名には、各サービスの目的をわかりやすく表す接頭辞が含まれます。これらの接頭辞には、`application`、`device`、`milestone`、`network`、`system` などの名前が含まれます。`site` 接頭辞はサイト固有のカスタマイズ用に予約されているため、`svc:/site/service-name` という名前のサービスは、Oracle Solaris リリースで配布されるサービスと競合しないことを意味します。

また、従来の `init.d` スクリプトは、`svc` ではなく `lrc` で始まる FMRI で表現され、たとえば `lrc:/etc/rc2_d/S47pppd` となります。レガシーサービスの、システムブート中の初期起動時間は `svcs` コマンドを使用して表示されます。ただし、SMF を使用してこれらのサービスを管理することはできません。

初期のシステム配備中、`/etc/inetd.conf` に記述されたサービスは自動的に SMF サービスに変換されます。これらのサービスの FMRI は多少異なります。変換された `inetd` サービスの構文は次のとおりです。

```
network/service-name/protocol
```

また、RPC プロトコルを使用するサービスの構文は次のとおりです。

```
network/rpc-service-name/rpc_protocol
```

ここで、`service-name` は `/etc/inetd.conf` に定義されている名前であり、`protocol` はそのサービスに使用されるプロトコルです。`inetconv` コマンドを使用して、初期システム配備後に `inetd.conf` エントリを変換することができます。

サービスの状態

svcs コマンドは、サービスインスタンスの状態、開始時間、および FMRI を表示します。各サービスの状態は次のいずれかになります。

- **degraded** – サービスインスタンスは有効ですが、限られた能力で実行されていません。
- **disabled** – サービスインスタンスは無効で、実行されていません。
- **legacy_run** – 従来のサービスは SMF によって管理されませんが、監視することはできます。この状態は従来のサービスでのみ使用されます。
- **maintenance** – サービスインスタンスに、管理者が解決しなければならないエラーが発生しました。
- **offline** – サービスインスタンスは有効ですが、サービスが実行されていないか、利用できる状態がありません。
- **online** – サービスインスタンスは有効で、正常に起動されました。
- **uninitialized** – この状態は、すべてのサービスの構成が読み込まれる前の初期状態です。

遷移中のインスタンスの状態にはアスタリスク「*」が付加されます。疑問符「?」は、状態が存在しないか、認識されない場合に表示されます。

SMF マニフェスト

SMF マニフェストは、サービスと一連のインスタンスを記述する XML ファイルです。マニフェストをインポートすることにより、サービスのプロパティとインスタンスがサービス構成リポジトリにロードされます。SMF マニフェストの詳細な内容については、[service_bundle\(4\)](#) のマニュアルページを参照してください。マニフェストを簡単に作成するためのツールの説明については、[svcbundle\(1M\)](#) のマニュアルページも参照してください。

マニフェストの推奨される場所は `/lib/svc/manifest` です。この場所に格納されたマニフェストは、ブートプロセスの間、どのサービスが起動するよりも前に `svc:/system/early-manifest-import:default` サービスによってインポートおよびアップグレードされます。インポートプロセスを早期に実行することにより、サービスが起動するよりも先に、最新のマニフェストに含まれる情報がリポジトリに確実に取り込まれます。その他の時点では `svcadm restart manifest-import` コマンドを実行することによって、これらのマニフェストから情報をインポートできます。`/var/svc/manifest` は互換性目的で引き続き利用できますが、この場所にあるマニフェストは、`svc:/filesystem/minimal:default` インスタンスがオンラインになるまでの間、すなわち `/var` がマウントされるまでの間、インポートもアップグレードもされません。

Oracle またはサードパーティーのソフトウェアベンダーによって配布されるマニフェストに変更を加えないでください。アップグレード時にカスタマイズがすべて失われるため、`/lib/svc/manifest` および `/var/svc/manifest` 内のマニフェストを直接編集しないでください。代わりに、サイトプロファイルを作成してサービスをカスタマイズするか、`svccfg` または `inetadm` コマンドを使用してプロパティを直接操作してください。`/lib/svc/manifest/site` および `/var/svc/manifest/site` ディレクトリも、サイト固有の用途のために予約されています。Oracle Solaris リリースは、これらのディレクトリにはマニフェストを配布しません。

Oracle Solaris 11 リリースでは、複数のマニフェストを使用して単一のサービスを記述することができます。これはたとえば、サービスの既存のマニフェストを変更することなく、サービスの新しいインスタスを定義するために役立ちます。同じサービスまたはインスタスについて、同じ管理レイヤー内の同じプロパティが複数のマニフェストで定義されている場合、SMF は、どの値を使用するかを決定できません。このような競合が検出された場合、インスタスは `maintenance` 状態に置かれます。レイヤーに関する詳細は、[16 ページの「SMF 管理レイヤー」](#)を参照してください。

SMF プロファイル

SMF プロファイルは、システムによって配布されるサービスおよびインスタスのカスタマイズに使用する XML ファイルです。プロファイルにより、複数のスクリプトではなく 1 つのファイルを使用してカスタマイズを実行できます。また、配布時またはインストール時に構成をカスタマイズすることも可能になります。

すべての構成は、プロファイルを使用してカスタマイズできます。

ローカルカスタマイズは、名前の接尾辞が `.xml` のファイルに記述し、このファイルを `/etc/svc/profile/site` ディレクトリに置く必要があります。このディレクトリ内のすべてのカスタマイズは、システムのブート時、または `svcadm restart manifest-import` コマンドの実行時に適用されます。

マニフェストと同様に、`/etc/svc/profile/site` 内のファイル間で競合する構成定義は競合として扱われ、影響を受けるインスタスは `maintenance` 状態に置かれます。

システムプロファイルもインストール中に適用されます。`/etc/svc/profile/generic.xml` 内のシステムプロファイルへの変更が必要になることはほとんどありません。詳細は、[smf_bootstrap\(5\)](#) のマニュアルページを参照してください。

プロファイルの使用方法については、[41 ページの「SMF プロファイルを適用する方法」](#)を参照してください。

サービス構成リポジトリ

サービス構成リポジトリには、永続的な構成情報と SMF 実行時サービスデータが格納されます。リポジトリは、ローカルメモリーとローカルファイルの間で割り当てられます。サービス構成リポジトリは、SMF インタフェースを使ってのみ操作または照会できます。リポジトリの操作やアクセスの方法については、[svccfg\(1M\)](#) および [svccprop\(1\)](#) のマニュアルページを参照してください。サービス構成リポジトリデーモンについては、[svc.configd\(1M\)](#) のマニュアルページで扱われています。サービス構成ライブラリについては、[libscf\(3LIB\)](#) のマニュアルページに記載されています。

リポジトリ内のプロパティは、サービスまたはインスタスのどちらかに対して定義できます。サービスに対して設定されるプロパティは、そのサービスのすべてのインスタスによって共有されます。インスタスに対して設定されるプロパティはそのインスタスのみによって使用され、サービスのプロパティをオーバーライドできます。

`svccfg` コマンドは、プロパティの生のビューを提供し、プロパティがサービスとインスタスのどちらに対して定義されているかを正確に示します。`svccfg` コマンドを使用してサービスを表示する場合、インスタスのプロパティを見ることはできません。インスタスを表示する場合、サービスのプロパティを見ることはできません。`svccprop` コマンドは、インスタスの合成ビューを提供し、インスタスのプロパティとサービスのプロパティの両方が単一のプロパティ名前空間に結合されます。サービスインスタスが起動されるときは、そのプロパティの合成ビューが使用されます。

すべての SMF 構成変更は、Oracle Solaris 監査フレームワークを使用してログに記録することができます。詳細は、『[Oracle Solaris 11.1 の管理: セキュリティーサービス](#)』の「[監査サービスの構成\(タスクマップ\)](#)」を参照してください。

SMF 管理レイヤー

Oracle Solaris 11 リリースで、プロパティ、プロパティグループ、インスタス、およびサービスのソースを記録する情報がサービス構成リポジトリに追加されました。ユーザーはこの情報によって、どのデータが管理者によるカスタマイズであり、どのデータがソフトウェアとともに配布されたかを判別できます。

エンティティのソースを識別しやすくするために、次のレイヤーが定義されています。

- `admin` レイヤーには、SMF コマンドを使用して、または `libscf(3LIB)` API を呼び出すことによって行われるすべての変更が含まれます。
- `site-profile` レイヤーには、`/etc/svc/profile/site` ディレクトリ内のファイル、またはレガシーの `/etc/svc/profile/site.xml` および `/var/svc/profile/site.xml` プロファイルからの値が含まれます。
- `system-profile` レイヤーには、システムプロファイルの場所である `/etc/svc/profile/generic.xml` および `/etc/svc/profile/platform.xml` からの値が含まれます。
- `manifest` レイヤーには、システムマニフェストディレクトリである `/lib/svc/manifest` または `/var/svc/manifest` からの値が含まれます。

プロパティ名ごとに1つのプロパティを想定する既存のクライアントとの互換性を維持するため、またオーバーライド用のポリシーを作成するために、レイヤー化は単純なオーバーライド動作を備えます。`admin` レイヤーが優先されます。`admin` レイヤーにプロパティの値がある場合、この値がサービスによって使用される値となります。そうでない場合、`site-profile` レイヤー、`system-profile` レイヤー、そして最後に `manifest` レイヤーの順に確認が行われます。この動作により、ローカルのカスタマイズが、システムのインストール時に提供された値よりも優先されます。

これらのレイヤーはシステムによって自動的に管理されます。管理者がリポジトリに直接加える変更は、`admin` レイヤーのみに記録されます。ほかのレイヤーは、標準の場所にファイルを配置するか、標準の場所のファイルを削除することによってのみ、変更されます。ファイルの内容に応じてプロパティがリポジトリに配置されるとき、そのプロパティについての情報には、内容の起源であるファイルの名前が含まれます。

管理者が `svccfg` または `libscf` の呼び出しを使用して下位レイヤーを直接変更することはできません。`svccfg delete`、`svccfg delpg`、または `svccfg delprop` コマンドを使用するとき、エンティティは完全に削除されるのではなく非表示になります。通常、ユーザーは削除されたエンティティを見ることはできませんが、非表示のエンティティは必要に応じて、`svccfg listcust` コマンドを使用して明示的に探査、また `svccfg delcust` コマンドを使用して非表示を解除することができます。非表示のエンティティを探査することによって、管理者は非表示が解除されたときに構成がどのように表示されるかを確認でき、動作中のシステムに悪影響を及ぼすことなく変更を行うことができます。

`svccfg listprop` コマンドには、これらのレイヤーの探査を有効にするオプションがあります。たとえば `svccfg listprop -l all` は、すべてのレイヤーと各レイヤーでの値を出力します。加えて、`svccfg listcust` コマンドを使用して、カスタマイズのみを一覧表示することができます。

SMF リポジトリのバックアップ

SMF では、次に示すリポジトリのバックアップを自動的に行います。

- ブートバックアップは、システムを起動するたびに、リポジトリに対する最初の変更が行われる直前に行われます。
- サービスが新しいマニフェストをインポートしたか、アップグレードスクリプトを実行した場合、`manifest_import` のバックアップは、`svc:/system/early-manifest-import:default` または `svc:/system/manifest-import:default` が完了したあとに発生します。

タイプごとに4つのバックアップがシステムによって管理されます。必要に応じて、もっとも古いバックアップから削除されます。バックアップは `/etc/svc/repository -type-YYYYMMDD_HHMMSSWS` という名前で格納され、ここで、`YYYYMMDD` (年、月、日) と `HHMMSS` (時、分、秒) は、バックアップが行われた日時です。時間は24時間形式で表されます。

エラーが発生した場合は、これらのバックアップからリポジトリを復元できません。そのためには、`/lib/svc/bin/restore_repository` コマンドを使用します。詳細は、53 ページの「破壊されたリポジトリを修復する方法」を参照してください。

SMF スナップショット

サービス構成リポジトリ内のデータには、編集可能な構成情報のほかにスナップショットもあります。各サービスインスタンスに関するデータは、スナップショットに格納されます。標準のスナップショットは、次のとおりです。

- `initial` – 目録の最初のインポート時に取られる
- `running` – `svcadm refresh` の実行時に取られます
- `start` – 最後に起動が成功したときに取られる

SMF サービスは常に `running` スナップショットを使って実行します。このスナップショットが存在しない場合は、自動的に作成されます。

`svccfg` コマンドは、現在のプロパティ値を変更するために使用します。それらの値は、`svcadm refresh` コマンドを実行して、実行中のスナップショットにそれらの値を統合した時点で、サービスから認識できるようになります。`svccfg` コマンドを使用すると、別のスナップショット内のインスタンス構成を参照したり、またはその構成に戻したりすることもできます。

SMF サービスのエラーログ

サービスまたはそのメソッドが生成するエラーなどのサービス固有情報に加えて、有効化アクションや開始時間などについての情報が、サービスインスタンスご

とに、`/var/svc/log` 内の個別ファイルにログとして記録されます。サービスのログファイルの名前を調べるには、`svcs -x service` コマンドを実行します。

SMF はデフォルトで、管理者による介入が必要な場合 (たとえば、サービスが `maintenance` 状態になった場合) にのみ、`syslog` プログラムとコンソールにログメッセージを書き込みます。ほかのオプションも使用可能ですが、めったに使用されません。可能性のあるその他の構成については、`svc.startd(1M)` のマニュアルページを参照してください。

エラーログに加えて、FMA イベントの発生時、あるいは、サービスが `service` 状態に (またはその状態から別の状態に) 遷移したときに通知を出すよう、SMF サービスを構成することができます。これらの通知では、SNMP (Simple Network Management Protocol) または SMTP (Simple Mail Transfer Protocol) を使用できます。SMF 通知の設定については、33 ページの「SMF 遷移イベントの通知を設定する方法」を参照してください。

SMF 管理およびプログラミングインタフェース

このセクションでは、SMF の使用時に利用できるインタフェースについて簡単に説明します。

SMF コマンド行管理ユーティリティー

SMF には、SMF とやりとりしたり、標準の管理タスクを実行したりする 1 連のコマンド行ユーティリティーが用意されています。SMF の管理には、次のユーティリティーを使用できます。

表 1-1 サービス管理機能のユーティリティー

コマンド名	機能
<code>svcadm</code>	サービスインスタンスの有効化、無効化、再起動などの一般的なサービス管理タスクを実行できます。
<code>svcbundle</code>	新しい SMF マニフェストおよびプロファイルを自動的に生成します。
<code>svccfg</code>	サービス構成リポジトリの内容を直接、表示および操作できます。注: このコマンドによって表示される内容は、選択されたエンティティーだけについての内容であって、サービスおよびインスタンスのプロパティーの組み合わせについての内容ではありません。
<code>svccprop</code>	シェルスクリプトでの使用に適した出力形式で、サービス構成リポジトリからプロパティー値を取り出します。注: このコマンドによって表示される内容は合成されており、サービスとインスタンスの両方に設定されているプロパティーを含みます。

表 1-1 サービス管理機能のユーティリティー (続き)

コマンド名	機能
svcs	サービス構成リポジトリ内のすべてのサービスインスタンスのサービス状態を詳細に表示できます。
inetadm	inetd によって制御されるサービスを監視または構成できます。

サービス管理構成ライブラリインタフェース

SMF には、`svc.configd` デーモンを介してサービス構成リポジトリとのやりとりを行うための一連のプログラミングインタフェースとして、`libscf(3LIB)` API が用意されています。このデーモンは、ローカルのリポジトリデータストアに対するすべての要求を判定します。サービス構成リポジトリ内のサービスとの最低レベルのやりとりの手段として、1 連の基本的なインタフェースが定義されています。これらのインタフェースを使うと、トランザクションやスナップショットなどのサービス構成リポジトリのすべての機能にアクセスできます。

多くの開発者は、SMF とやりとりするための一般的なタスクのみを必要としています。これらのタスクは、便利な機能として基本サービスの上位で実装され、実装にかかる負荷を軽減しています。

SMF コンポーネント

SMF には、マスターリスタータデーモンと委任リスタータがあります。加えて、個々のサービスまたはサービスインスタンスは構成データをプロパティーに格納できます。これらのプロパティーはプロパティーグループに整理され、管理が容易になっています。

SMF マスターリスタータデーモン

`svc.startd` デーモンは、マスタープロセスのスタータおよびリスタータです。このデーモンは、システム全体のサービス依存関係を管理する役割を担っています。適切な実行レベルで適切な `/etc/rc*.d` スクリプトを起動することは、以前は `init` の役割でしたが、現在はこのデーモンの役割です。まず、`svc.startd` はサービス構成リポジトリに格納されている情報を取り出します。次に、サービス依存関係が満たされたときにそのサービスを起動します。また、失敗したサービスの再起動や、依存関係が満たされなくなったサービスの停止も行います。デーモンはプロセス消滅などのオペレーティングシステムイベントを使用して、サービスの状態を追跡します。

SMF 委任リスタータ

一部のサービスは、起動時に共通の動きが見られます。これらのサービス間に共通性を持たせるために、委任リスタータがこれらのサービスに対する責任を負うことがあります。また、より複雑な再起動やアプリケーション固有の再起動を行えるようにする場合にも委任リスタータを使用できます。委任リスタータは、別のメソッド群をサポートできますが、マスターリスタータと同じサービス状態をエクスポートします。リスタータの名前は、サービスとともに格納されます。委任リスタータの例には、インターネットサービスを常に実行しておくのではなく、要求に応じて起動できる `inetd` があります。各サービスの定義済みリスタータは、`svcs -l` コマンドを使用して表示できます。

SMF プロパティとプロパティグループ

サービス構成リポジトリ内のすべての情報は、プロパティグループによってグループ化される一連のプロパティとして格納されます。プロパティグループは、さまざまな種類のサービス情報をグループ化するために使用されます。プロパティグループは、個々のサービスだけでなくシステムにも適用できます。一般的なシステム定義のプロパティグループには、次のものがあります。

- `general` - サービスインスタンスについての情報を格納します。インスタンスが有効かどうかを定義する `general/enabled` プロパティを含みます。
- `restarter` - サービスのリスタータによって格納される実行時の情報を格納します。サービスの現在の状態を示す `restarter/state` プロパティを含みます
- `start` - 起動方法の定義を格納します。サービスを起動するためにどのプログラムを実行するかを定義する `start/exec` プロパティを含みます。

他の一般的なプロパティグループに、`config` があります。`config` プロパティグループはシステムで定義されないため、サービス開発者がサービスデータを保持するために使用できます。プロパティとプロパティグループの詳細は、[smf\(5\)](#)のマニュアルページを参照してください。

サービス構成リポジトリ内の情報の管理

SMF を使用して情報を表示、変更、または削除する方法は多数あります。このセクションでは、用途ごとにどの方法が最適かについて説明します。

SMF 情報の表示

svccfg および svcprop コマンドを使用して、サービス構成リポジトリ内の情報を表示できます。これらのコマンドの完全な説明については、[svccfg\(1M\)](#) および [svcprop\(1\)](#) のマニュアルページを参照してください。

- **svcprop** – 実行中のスナップショットでプロパティグループまたはプロパティに割り当てられている値を一覧表示します。このコマンドはサービスのデータとサービスインスタンスのデータを結合するため、包括的な、あるいは「合成された」データの表示が得られます。
- **svccfg listpg** – 選択されたサービス、サービスインスタンス、またはスナップショット内のプロパティグループについての情報を一覧表示します。すべてのプロパティグループの名前、種類、およびフラグが一覧表示されます。
- **svccfg listprop** – 選択されたサービス、サービスインスタンス、またはスナップショット内のプロパティとプロパティグループについての情報を一覧表示します。プロパティグループの場合、名前、種類、およびフラグが一覧表示されます。プロパティの場合、名前、種類、および値が一覧表示されます。
 - **-l layer_name** – サービスまたはサービスインスタンスの指定されたレイヤー内のプロパティとプロパティグループを一覧表示します。レイヤー名に **all** を使用すると、そのサービスのすべてのレイヤーとプロパティが一覧表示されます。個々のサービスおよびサービスインスタンスは別々に表示されることに注意してください。
 - **-f** – プロパティの起源であるファイル名を一覧表示します。
 - **-o** – 表示するフィールドを選択します。
- **svccfg listcust – site-profile** または **admin** レイヤーでのサイトカスタマイズを一覧表示します。選択されたサービスまたはサービスインスタンスの非表示エントリも一覧表示します。
 - **-M** – 非表示エントリのみを一覧表示します。
 - **-L** – すべてのローカルカスタマイズを表示します。これには、管理者によるカスタマイズとサイトプロファイルによるカスタマイズの両方が含まれます。

SMF 情報の変更

サービスまたはサービスインスタンスについての情報は、マニフェスト、プロファイル、または **svccfg** コマンドを使用して変更できます。**svccfg** コマンドを使用して行う変更はすべて、**admin** レイヤーに記録されます。次に示すのは、**svccfg** コマンドを使用して情報を変更するために使用できるオプションの一部です。

- **addpg** – 選択されたサービスまたはサービスインスタンスにプロパティグループを追加します
- **addpropvalue** – 既存のプロパティに追加の値を割り当てます

- `setenv` - サービスまたはサービスインスタンスの環境変数を設定します
- `setprop` - 選択されたサービス内の指定されたプロパティの値を設定します
- `setnotify` - ソフトウェアイベントおよびFMA イベントクラスの通知パラメータを設定します

SMF情報の削除

サービスまたはサービスインスタンスについての情報は、`svccfg` コマンドを使用して削除できます。次に示すのは、`svccfg` コマンドを使用して情報を削除するために使用できるオプションの一部です。

- `delcust` - 選択されたサービスの管理者によるカスタマイズをすべて削除します
- `delpropvalue` - 指定された文字列に一致するすべてのプロパティ値を削除します
- `unsetenv` - サービスまたはサービスインスタンスの環境変数を削除します

リポジトリから情報を削除するときの結果は、エンティティが非表示になるか、エンティティが削除されるかの2つのうちのどちらかです。admin レイヤーのみで定義されている情報は削除されます。マニフェストまたはプロファイルが起源の情報は非表示になり、標準のコマンドでは情報が表示されなくなります。情報が非表示とされるのは、削除を取り消す必要がある場合に情報を容易に利用可能にするためです。

SMFとブート

システムをブートするとき、ブートするマイルストーン、または記録されるエラーメッセージのレベルを次のようにして選択できます。

- 次のコマンドを使用して、ブートする特定のマイルストーンを選択できます。

```
ok boot -m milestone=milestone
```

デフォルトは `all` で、有効化されているすべてのサービスを起動します。 `none` も役に立ちます。これは、`init`、`svc.startd`、および `svc.configd` のみを起動し、サービスを手動で起動できるデバッグ環境を提供します。 `none` マイルストーンの使用手順については、[56 ページの「どのサービスも起動しないでブートする方法」](#)を参照してください。

実行レベルに相当する `single-user`、`multi-user`、および `multi-user-server` も使用可能ですが、一般的には使用されません。特に `multi-user-server` は、そのマイルストーンに依存しない一切のサービスを起動しないため、重要なサービスが起動されない可能性があります。

- 次のコマンドを使用して、`svc.startd` のログのレベルを選択できます。

```
ok boot -m logging-level
```

選択できるログレベルには `quiet` と `verbose` があります。ログレベルの詳細は、18 ページの「SMF サービスのエラーログ」を参照してください。

SMFの互換性

標準のサービスの多くは SMF によって管理されていますが、実行レベルの移行に対しては `/etc/rc*.d` 内にあるスクリプトが引き続き実行されます。以前のリリースに含まれていた `/etc/rc*.d` スクリプトの大半は、SMF の一環として削除されました。残りのスクリプトを引き続き実行できることにより、SMF を使用するようにサービスを変換しなくてもサードパーティーのアプリケーションを追加できます。

さらに、`/etc/inittab` エントリも引き続き `init` コマンドによって処理されます。また、修正するパッケージに対しては `/etc/inetd.conf` が使用可能です。初期のシステム配備中、`/etc/inetd.conf` に記述されたサービスは自動的に SMF サービスに変換されます。その後の追加はすべて、`inetconv` コマンドを使用して変換できます。従来の実行サービスのステータスは表示できますが、ほかの変更は一切 SMF でサポートされていません。この変換機能を使用するアプリケーションは、SMF が提供する高精度の障害の封じ込めによるメリットを受けられません。`inetd` の最新バージョンは、初期ブート後に変換するエントリを `/etc/inetd.conf` から探しません。

SMF を利用するために変換されるアプリケーションは、このセクションで示すメカニズムを利用する必要がなくなりました。

実行レベル

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

このリリースには 8 つの実行レベルがあります (次の表を参照)。デフォルトの実行レベルは、実行レベル 3 として `/etc/inittab` ファイルに指定されています。

表 1-2 Oracle Solaris の実行レベル

実行レベル	init 状態	タイプ	目的
0	電源切断状態	電源切断	オペレーティングシステムをシャットダウンして、システムの電源を安全に落とせるようにします。
s または S	シングルユーザー状態	シングルユーザー	一部のファイルシステムがマウントされ使用可能な状態で、シングルユーザーとして動作します。

表 1-2 Oracle Solaris の実行レベル (続き)

実行レベル	init 状態	タイプ	目的
1	システム管理状態	シングルユーザー	すべての使用可能なファイルシステムにアクセスします。ユーザーログインは使用できません。
2	マルチユーザー状態	マルチユーザー	通常の運用に使用します。複数のユーザーがシステムとすべてのファイルシステムにアクセスできます。NFS および SMB サーバーデーモンを除く、すべてのデーモンが動作します。
3	NFS リソースを共有できるマルチユーザーレベル	マルチユーザー	NFS および SMB リソースを共有する通常の運用に使用します。これがデフォルトの実行レベルです。
4	マルチユーザー状態 (予備)	マルチユーザー	デフォルトでは構成されていませんが、顧客用に使用できます。
5	電源切断状態	電源切断	オペレーティングシステムをシャットダウンしてシステムの電源を安全に落とせるようにします。可能であれば、この機能をサポートしているシステムでは電源を自動的に切断します。
6	リブート状態	リブート	システムをシャットダウンして実行レベル 0 にしたあと、NFS および SMB リソースを共有するマルチユーザーレベル (または <code>inittab</code> ファイルに指定されたデフォルトのレベル) でリブートします。

また、`svcadm` コマンドを使用してシステムの実行レベルを変更することもできます。その場合は、実行するときのマイルストーンを選択してください。次の表は、各マイルストーンに対応する実行レベルを示しています。

表 1-3 実行レベルと SMF マイルストーン

実行レベル	SMF マイルストーンの FMRI
5	<code>milestone/single-user:default</code>
2	<code>milestone/multi-user:default</code>
3	<code>milestone/multi-user-server:default</code>

実行レベルとマイルストーンの使い分け

一般に、マイルストーンまたは実行レベルを変更する手順はめったに実行されません。必要な場合、実行レベルを変更するために `init` コマンドを使用するとマイルストーンも変更され、これは使用するのに適切なコマンドです。 `init` コマンドは、システムのシャットダウンにも役立ちます。

ただし、 `none` マイルストーンを使用してシステムをブートすることは、スタートアップの問題のデバッグ時に非常に役立つ可能性があります。 `none` マイルストーンと同等の実行レベルはありません。具体的な手順については、 [56 ページの「どのサービスも起動しないでブートする方法」](#) を参照してください。

システムの実行レベルを確認する

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

```
$ who -r
```

システムの現在の実行レベルを調べるには、 `who -r` コマンドを使用します。

例 1-1 システムの実行レベルを確認する

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

```
$ who -r
.      run-level 3  Dec 13 10:10  3  0 S
$
```

who -r コマンドの出力	説明
run-level 3	現在の実行レベルを識別します
Dec 13 10:10	実行レベルが最後に変更された日時
3	これも現在の実行レベルを識別します
0	最後にリポートしてからシステムがこの実行レベルになった回数
S	以前の実行レベルを識別します

/etc/inittab ファイル

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

- init プロセスが再起動すること
- 起動、モニターするプロセス、および停止時に再起動するプロセス
- システムが新しい実行レベルに移行したとき行う処理

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

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

次の表で、inittab エントリのフィールドについて説明します。

表 1-4 inittab ファイルのフィールドの説明

フィールド	説明
<i>id</i>	エントリに固有の(一意の)識別子。
<i>rstate</i>	このエントリが適用される実行レベルのリスト。
<i>action</i>	プロセスフィールドに指定されたプロセスの実行方法。指定できる値は、 <code>sysinit</code> 、 <code>boot</code> 、 <code>bootwait</code> 、 <code>wait</code> 、および <code>respawn</code> です。 ほかの <code>action</code> キーワードについては、 inittab(4) のマニュアルページを参照してください。
<i>process</i>	実行するコマンドまたはスクリプトを定義します。

例 1-2 デフォルトの inittab ファイル

次の例では、リリースでインストールされるデフォルトの inittab ファイルを示します。そのあとに、この例の出力の各行についての説明も示します。

```
ap::sysinit:/usr/sbin/autopush -f /etc/iu.ap          (1)
smf::sysinit:/lib/svc/bin/svc.startd    >/dev/msglog 2<>/dev/msglog </dev/console    (2)
p3:s1234:powerfail:/usr/sbin/shutdown -y -i5 -g0 >/dev/msglog 2<>/dev/...    (3)
```

1. STREAMS モジュールを初期化します
2. SMF 用のマスターリスタータを初期化します
3. 電源障害の場合のシャットダウンを指定します

システムが実行レベル3になると実行される処理

1. `init` プロセスが起動され、`svc:/system/environment:init` SMF サービスで定義されたプロパティーを読み取って、環境変数を設定します。
2. `init` プロセスは `inittab` ファイルを読み取り、次の処理を行います。
 - a. `action` フィールドが `sysinit` になっているすべてのプロセスエントリを実行して、ユーザーがログインする前に特別な初期設定処理がすべて行われるようにします。
 - b. 起動アクティビティーを `svc.startd` に渡します。

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

サービスの管理 (タスク)

この章では、サービス管理機能 (Service Management Facility: SMF) の管理とモニターに必要なタスクについて説明します。また、実行レベルのスクリプトの管理についても説明します。この章の内容は次のとおりです。

- 29 ページの「SMF サービスのモニタリング」
- 34 ページの「SMF サービスの管理」
- 41 ページの「SMF サービスの構成」
- 48 ページの「実行制御スクリプトの使用」
- 52 ページの「サービス管理機能のトラブルシューティング」

SMF サービスのモニタリング

次の各タスクは、SMF サービスのモニター方法を示したものです。

サービスのモニタリング (タスクマップ)

次のタスクマップに、SMF サービスをモニターするために必要な手順を示します。

タスク	説明	参照先
サービスインスタンスのステータスを表示します。	実行中のすべてのサービスインスタンスのステータスを表示します。	30 ページの「サービスのステータスを一覧表示する方法」
サービスに対するカスタマイズを表示します。	サービスに対する admin レイヤーでのカスタマイズを表示します。	31 ページの「サービスのカスタマイズを一覧表示する方法」
特定のサービスに依存するサービスを表示します。	指定されたサービスに依存するサービスを表示します。	31 ページの「特定のサービスに依存するサービスを表示する方法」

タスク	説明	参照先
特定のサービスが依存するサービスを表示します。	指定されたサービスが依存するサービスを表示します。この情報があれば、特定のサービスが起動できない原因を特定しやすくなります。	32 ページの「特定のサービスが依存するサービスを表示する方法」
重要なイベントに関する通知を構成します。	SNMP または SMTP を使用して、サービス状態の変更または FMA イベントの通知を送信します。	33 ページの「SMF 遷移イベントの通知を設定する方法」

▼ サービスのステータスを一覧表示する方法

この手順は、実行中のサービスを表示する場合に使用できます。

- **svcs** コマンドを実行します。

オプションを指定しないでこのコマンドを実行した場合、FMRI で指定されたサービスのステータスレポートが表示されます。

```
$ svcs -l FMRI
```

例 2-1 sendmail サービスのステータスを表示する

この例では、依存するサービスを持つサービスのステータスを表示しています。また、**-p** オプションを使用して、プロセス ID、開始時間、およびコマンド実行に関する情報を表示しています。

```
$ svcs -lp network/smtp:sendmail
fmri          svc:/network/smtp:sendmail
name          sendmail SMTP mail transfer agent
enabled       true
state         online
next_state    none
state_time    Tue Aug 09 19:25:54 2011
logfile       /var/svc/log/network-smtp:sendmail.log
restarter     svc:/system/svc/restarter:default
contract_id   114
manifest      /etc/svc/profile/generic_limited_net.xml
manifest      /lib/svc/manifest/network/smtp-sendmail.xml
dependency    require_all/refresh file://localhost/etc/mail/sendmail.cf (online)
dependency    optional_all/refresh file://localhost/etc/nsswitch.conf (online)
dependency    optional_all/none svc:/system/filesystem/autofs (online)
dependency    require_all/none svc:/system/filesystem/local (online)
dependency    require_all/none svc:/network/service (online)
dependency    require_all/refresh svc:/milestone/name-services (online)
dependency    optional_all/refresh svc:/system/identity:domain (online)
dependency    optional_all/none svc:/system/system-log (online)
process       101077 /usr/lib/sendmail -bd -q15m
```

例 2-2 すべてのサービスのステータスを表示する

次のコマンドを実行すると、システムにインストールされているすべてのサービスが一覧表示されます。その際、各サービスのステータスも表示されます。このコマンドでは、有効になっているサービスだけでなく、無効になっているサービスも表示されます。

```
$ svcs -a
```

例 2-3 すべての通知パラメータを表示する

次のコマンドは、すべての FMA イベント通知パラメータと SMF 状態遷移パラメータを一覧表示します。

```
$ svcs -n
```

例 2-4 inetd によって制御されるサービスのステータスを表示する

次のコマンドを実行すると、inetd によって制御されるサービスが一覧表示されます。サービスごとに、FMRI のほかに、実行状態と有効になっているかどうかが表示されます。

```
$ inetadm
```

▼ サービスのカスタマイズを一覧表示する方法

- ローカルのカスタマイズを一覧表示します。
このコマンドを実行すると、選択したサービスの admin レイヤーでの変更がすべて表示されます。

```
% /usr/sbin/svccfg -s FMRI listcust
```

▼ 特定のサービスに依存するサービスを表示する方法

この手順は、指定されたサービスに依存するサービスインスタンスを確認する方法を示しています。

- 特定のサービスに依存するサービスを表示します。

```
$ svcs -D FMRI
```

例 2-5 multiuser マイルストーンに依存するサービスインスタンスを表示する

次の例は、multiuser マイルストーンに依存するサービスインスタンスを確認する方法を示しています。

```
$ svcs -D milestone/multi-user
STATE          STIME    FMRI
disabled       Aug_09   svc:/application/time-slider:default
disabled       Aug_09   svc:/application/management/net-smnp:default
online         Aug_09   svc:/system/intrd:default
online         Aug_09   svc:/system/boot-config:default
online         Aug_09   svc:/milestone/multi-user-server:default
```

▼ 特定のサービスが依存するサービスを表示する方法

この手順は、指定されたサービスインスタンスが依存するサービスを確認する方法を示しています。

- 特定のサービスが依存するサービスを表示します。

```
$ svcs -d FMRI
```

例 2-6 multiuser マイルストーンが依存するサービスインスタンスを表示する

次の例では、multiuser マイルストーンが依存するサービスインスタンスを表示しています。

```
$ svcs -d milestone/multi-user:default
STATE          STIME    FMRI
disabled       Aug_09   svc:/network/nfs/client:default
disabled       Aug_09   svc:/network/smb/client:default
disabled       Aug_09   svc:/system/mdmonitor:default
disabled       Aug_09   svc:/application/print/server:default
online         Aug_09   svc:/system/resource-controls:default
online         Aug_09   svc:/system/metasync:default
online         Aug_09   svc:/system/rmtmpfiles:default
online         Aug_09   svc:/system/utmp:default
online         Aug_09   svc:/system/name-service/cache:default
online         Aug_09   svc:/system/device/audio:default
online         Aug_09   svc:/network/rpc/bind:default
online         Aug_09   svc:/milestone/name-services:default
online         Aug_09   svc:/network/iscsi/initiator:default
online         Aug_09   svc:/milestone/single-user:default
online         Aug_09   svc:/system/filesystem/local:default
online         Aug_09   svc:/network/inetd:default
online         Aug_09   svc:/system/cron:default
online         Aug_09   svc:/system/filesystem/autofs:default
online         Aug_09   svc:/system/filesystem/ufs/quota:default
online         Aug_09   svc:/system/power:default
online         Aug_09   svc:/system/system-log:default
```

```

online      Aug_09   svc:/system/system-log:default
online      Aug_09   svc:/system/auditd:default
online      Aug_09   svc:/network/smtp:sendmail

```

▼ SMF 遷移イベントの通知を設定する方法

この手順を実行すると、いずれかのサービスまたは選択したサービスの状態が変化するたびに、システムが電子メール通知を生成します。SMTP と SNMP のいずれかの使用を選択できます。他の何らかの理由で SNMP がすでに構成されている場合は、通常、SNMP のみを選択します。

デフォルトでは、保守の遷移に関して SNMP トラップが送信されます。SNMP を使用してモニターする場合は、他の状態遷移に関して追加のトラップを構成できません。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『[Oracle Solaris 11.1 の管理: セキュリティーサービス](#)』の「[割り当てられている管理権限を使用する方法](#)」を参照してください。

- 2 通知パラメータを設定します。

次の例は、SMF および FMA イベントに関する通知パラメータを設定する方法と、通知パラメータを一覧表示および削除する方法を示しています。

例 2-7 すべての SMF サービス状態イベントに関する通知を構成する

次のコマンドを実行すると、サービスが maintenance 状態に移行したときに電子メールを送信する通知が作成されます。

```
# /usr/sbin/svccfg setnotify -g maintenance mailto:sysadmins@example.com
```

from オプションに指定した状態で開始し、to オプションに指定した状態で終了するサービスを選択することもできます。SMF の有効な状態には、degraded、maintenance、offline、および online があります。-g all オプションを使用すると、すべての状態遷移イベントに関して電子メールを生成できます。詳細は、[smf\(5\)](#) のマニュアルページの「通知パラメータ」セクションを参照してください。

例 2-8 個々のサービスに関する通知の構成

次のコマンドを実行すると、switch サービスが online 状態に移行したときに電子メールを送信する通知が作成されます。

```
# /usr/sbin/svccfg -s svc:/system/name-service/switch:default setnotify to-online \
mailto:sysadmins@example.com
```

例 2-9 FMA イベントに関する通知の構成

次のコマンドを実行すると、FMA の問題が修復されたときに SNMP メッセージを送信する通知が作成されます。

```
# /usr/sbin/svccfg setnotify problem-repaired snmp:
```

FMA イベントのクラスには、`problem-diagnosed`、`problem-updated`、`problem-repaired`、および `problem-resolved` があります。詳細は、[smf\(5\)](#) のマニュアルページの「通知パラメータ」セクションを参照してください。

例 2-10 通知設定を一覧表示する

次のコマンドを実行すると、FMA サービスによって診断される新しい問題に関する通知設定が表示されます。SMF サービス状態遷移イベントの通知設定を表示するには、`listnotify` にイベントクラスではなくサービス状態を含めるか、または引数を 1 つも含めないようにします。

```
# /usr/sbin/svccfg listnotify problem-diagnosed
```

例 2-11 通知設定を削除する

次のコマンドを実行すると、`switch` サービスの `online` サービス状態への遷移に関連する通知設定が削除されます。サービス状態の代わりに FMA イベントクラスを使用できます。

```
# /usr/sbin/svccfg -s svc:/system/name-service/switch:default delnotify to-online
```

SMF サービスの管理

このセクションでは、SMF サービスの管理について説明します。

SMF サービス(タスクマップ)

次のタスクマップに、SMF サービスを管理するために必要な手順を示します。

タスク	説明	参照先
サービスを作成します。	SMF サービスを通じてスクリプトを管理できるようにします。	35 ページの「SMF サービスを作成する方法」
サービスインスタンスを無効にします。	実行中のサービスを停止し、サービスが再び起動しないようにします。	36 ページの「サービスインスタンスを無効にする方法」

タスク	説明	参照先
サービスインスタンスを有効にします。	特定のサービスを起動します。また、以降のリポート時にサービスが再起動されます。	37 ページの「サービスインスタンスを有効にする方法」
特定のサービスを再起動します。	別個のコマンドを使用してサービスを無効化および有効化することなしにサービスを再起動します。	38 ページの「サービスを再起動する方法」
保守状態のサービスを復元します。	保守状態のサービスをクリーンアップして再起動する方法を示します。	38 ページの「保守状態のサービスを復元する方法」
プロファイルを作成します。	すべての変更を実行するために、svccfg コマンドを実行しなくてもカスタマイズを簡単に配備できるプロファイルを作成します。 カスタマイズを簡単に配備するためのプロファイルを手動で作成します。	39 ページの「SMF プロファイルを作成する方法」 40 ページの「SMF プロファイルを手動で作成する方法」
プロファイルを適用します。	プロファイルの情報を使用して、必要に応じてサービスの無効化、有効化、またはカスタマイズを行います。	41 ページの「SMF プロファイルを適用する方法」

RBAC 権利プロファイルを使って SMF を管理する

RBAC 権利プロファイルを使用して、root アクセス権を付与していないユーザーに一部の SMF サービスの管理を許可できます。権利プロファイルには、そのユーザーが実行できるコマンドを定義します。SMF のために次のプロファイルが作成されています。

- **Service Management:** サービスを追加、削除、または変更できます。
- **Service Operator:** 再起動やリフレッシュなど、サービスインスタンスの状態の変更を要求できます。

承認については、[smf_security\(5\)](#) のマニュアルページを参照してください。権利プロファイルを割り当てる手順については、『[Oracle Solaris 11.1 の管理: セキュリティサービス](#)』の「[ユーザーのセキュリティー属性を変更する方法](#)」を参照してください。

▼ SMF サービスを作成する方法

管理者は、システムのブート中に作業を実行するスクリプトを記述することがよくあります。この手順では、スクリプトをサポートする SMF の利点を使用する方法を示します。

始める前に スクリプトが必要とするサービスモデルを確認します。スクリプトが何らかの作業を実行し、長期間実行するプロセスを開始せずに終了するのであれば、ほとんどの場合は一時的なサービスです。スクリプトが長時間実行するデーモンを開始してから終了する場合、それは契約サービスです。サービスモデルについての詳細は、`svc.startd(1M)` のマニュアルページの期間セクションを参照してください。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 開始メソッドスクリプトを `/lib/svc/method/ex_svc` にコピーします。

- 3 一時的なマニフェストを作成します。

この例で、サービス名は `site/ex_svc` で、一時的なサービスであるため、停止メソッドは不要です。

```
# svcbundle -o /tmp/ex_svc.xml -s service-name=site/ex_svc \  
-s start-method=/lib/svc/method/ex_svc -s model=transient
```

- 4 マニフェストへの追加変更を行います。

`/tmp/ex_svc.xml` 内のマニフェストが、必要な内容であることを確認します。

- 5 マニフェストを適切なディレクトリにコピーします。

```
# cp /tmp/ex_svc.xml /lib/svc/manifest/site/ex_svc.xml
```

- 6 マニフェストをインポートし、サービスを開始します。

```
# svcadm restart svc:/system/manifest-import
```

例 2-12 生成されたマニフェストを自動的にインストールする

マニフェストを `/lib/svc/manifest/site` に自動的にインストールするには、下に示すように `-i` オプションを付けて `svcbundle` コマンドを使用します。このコマンドはステップ 3 のコマンドの代わりに実行でき、他のステップを実行する必要はありません。

```
# svcbundle -i -s service-name=site/ex_svc \  
-s start-method=/lib/svc/method/ex_svc -s model=transient
```

▼ サービスインスタンスを無効にする方法

サービスを無効にするには、次の手順に従います。サービスのステータス変化はサービス構成リポジトリ内に記録されます。サービスがいったん無効になると、その無効な状態はリブート後も維持されます。サービスを実行中の状態に戻す唯一の方法は、そのサービスを有効にすることです。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 無効にするサービスに依存するサービスの有無を確認します。
必要とするサービスがこのサービスに依存していた場合、このサービスを無効にできません。

```
# svcs -D FMRI
```
- 3 サービスを無効にします。

```
# svcadm disable FMRI
```

▼ サービスインスタンスを有効にする方法

サービスを有効にするには、次の手順に従います。サービスのステータス変化はサービス構成リポジトリ内に記録されます。サービスを有効にすると、システムのリブートまで有効な状態が持続しますが、サービスが起動するのはすべての依存関係が満たされた場合だけです。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 サービスを有効にします。

```
# svcadm enable FMRI
```
- 3 サービスが有効になっていることを確認します。

```
# svcs -x FMRI
```
- 4 サービスの依存関係を有効にします。
必要な依存関係が無効になっている場合は、次のコマンドを使用して有効にします。

```
# svcadm enable -r FMRI
```

例 2-13 シングルユーザーモードでサービスを有効にする

次のコマンドは `rpcbind` を有効にします。 `-t` オプションを指定すると、サービスが一時モードで起動され、サービスリポジトリが変更されないため、この変更はリブートまで持続しません。 `-r` オプションは、指定されたサービスのすべての依存関係を再帰的に有効にします。

```
# svcadm enable -rt rpc/bind
```

▼ サービスを再起動する方法

構成が変更されたなどの理由により現在実行中のサービスを再起動する必要がある場合、サービスの停止時と起動時に個々のコマンドを入力することなく、サービスの再起動を行えます。サービスの無効化と有効化を別々に行う必要があるのは、サービスの無効化後かつ有効化前に変更を反映する必要がある場合だけです。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 特定のサービスを再起動します。

```
# svcadm restart FMRI
```

▼ 保守状態のサービスを復元する方法

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 サービスが保守中である理由を特定します。

```
# svcs -x FMRI
```

指示されたログファイルまたはマニュアルページを参照して、エラーの内容を特定します。

- 3 そのサービスに依存するプロセスのうち、停止していないものが存在しないかどうかを確認します。

通常、サービスインスタンスが保守状態になっている場合、そのインスタンスに関連付けられたすべてのプロセスが停止しています。ただし、処理を進める前にそのことを確認する必要があります。次のコマンドは、選択された FMRI の契約 ID を一覧表示します。

```
# svcs -o CTID FMRI
```

- 4 (省略可能) 残っているプロセスをすべて強制終了します。

svcs コマンドで表示されたすべてのプロセスに対して、この手順を繰り返します。

```
# pkill -9 -c CTID
```

- 5 サービスを復元します。

```
# svcadm clear FMRI
```

▼ SMF プロファイルを作成する方法

プロファイルとは、SMF サービスの一覧とそれぞれを有効にするかどうかを示す XML ファイルのことです。プロファイルは、多くのサービスを一度に有効または無効にするために使用されます。プロファイルは、プロパティー値の設定、プロパティー値の追加、およびサービスやサービスのインスタンスの作成にも使用されます。すべてのサービスをプロファイルに含める必要はありません。効果的なプロファイルを作成するには、各プロファイルには有効または無効にする必要のあるサービスだけを含めるようにします。

svc:/system/rmtmpfiles サービスは、/tmp ディレクトリをブート時にクリーンアップする役割を担っています。デフォルトでは、rmtmpfiles サービスは /var/tmp をクリーンアップしません。ブートプロセス中に /var/tmp をクリーンアップする場合、options/clean_vartmp プロパティーを true に設定することによって動作を変更できます。これを複数のシステムに対して実行する場合、もっとも簡単な方法は、プロファイルを作成し、それをこれらのシステムの /etc/svc/profile/site に配置することです。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 プロファイルを作成します。
このコマンドにより、/tmp/rmtmpfiles.xml に新しいプロファイルが作成され、これをあとで必要に応じて /etc/svc/profiles/site にコピーできます。

```
# svcbundle -o /tmp/rmtmpfiles.xml -s service-name=system/rmtmpfiles \
-s bundle-type=profile -s service-property=options:clean_vartmp:boolean:true
```
- 3 プロファイルを適切なディレクトリにコピーします。

```
# cp /tmp/rmtmpfiles.xml /etc/svc/profile/site/rmtmpfiles.xml
```
- 4 manifest-import サービスを再起動し、プロファイルをシステムに適用します。

```
# svcadm restart svc:/system/manifest-import
```

例 2-14 プロファイルの作成中に SMF プロファイルを自動的にインストールする

生成されたプロファイルをインストールするには -i オプションを使用します。コマンドはプロファイルを /etc/svc/profile/site/rmtmpfiles.xml に書き込みます。また、コマンドは manifest-import サービスを再起動してプロファイルを適用します。

```
# svcbundle -i -s service-name=system/rmtmpfiles \
-s bundle-type=profile -s service-property=options:clean_vartmp:boolean:true
```

▼ SMF プロファイルを手動で作成する方法

プロファイルとは、SMF サービスの一覧とそれぞれを有効にするかどうかを示す XML ファイルのことです。プロファイルは、多くのサービスを一度に有効または無効にするために使用されます。プロファイルは、プロパティ値の設定、プロパティ値の追加、およびサービスやサービスのインスタンスの作成にも使用されます。すべてのサービスをプロファイルに含める必要はありません。効果的なプロファイルを作成するには、各プロファイルには有効または無効にする必要のあるサービスだけを含めるようにします。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『[Oracle Solaris 11.1の管理: セキュリティーサービス](#)』の「[割り当てられている管理権限を使用する方法](#)」を参照してください。

- 2 プロファイルを作成します。

この例では、`svccfg` コマンドを使用して、現在のシステムに対するローカルのカスタマイズを表すプロファイルを作成しています。代わりに、既存のプロファイルのコピーを作成し、それを編集することもできます。

```
# svccfg extract> profile.xml
```

同一のシステムが多数ある場合や、あとで復元できるようにシステム構成のアーカイブを作成する場合は、この手順を使用して単一の SMF プロファイルを作成することもできます。

- 3 `profile.xml` ファイルを編集して、必要な変更を加えます。

- a. `service_bundle` 宣言内のプロファイルの名前を変更します。

この例では、`profile` という名前に変更しています。

```
# cat profile.xml
...
<service_bundle type='profile' name='profile'
  xmInss::xi='http://www.w3.org/2003/XInclude'
...

```

- b. このプロファイルで管理すべきでないサービスをすべて削除します。

各サービスについて、サービスを説明する 3 行を削除します。各サービスの説明は、`<service` で始まり、`</service` で終わります。この例は、LDAP クライアントサービスの行を示しています。

```
# cat profile.xml
...
<service name='network/ldap/client' version='1' type='service'>
  <instance name='default' enabled='true' />
</service>

```

- c. このプロファイルで管理すべきサービスをすべて追加します。

上記の 3 行の構文を使用して各サービスを定義する必要があります。

- d. 必要に応じて、選択したサービスの **enabled** フラグを変更します。

この例では、sendmail サービスを無効にしています。

```
# cat profile.xml
...
<service name='network/smtp' version='1' type='service'>
  <instance name='sendmail' enabled='false' />
</service>
...
```

- 4 必要に応じて、この新しいプロファイルを適用します。
手順については、41 ページの「SMF プロファイルを適用する方法」を参照してください。

▼ SMF プロファイルを適用する方法

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 プロファイルを適用します。
この例では、profile.xml を適用しています。

```
# svccfg apply profile.xml
```

SMF サービスの構成

次の各タスクは、SMF サービスの構成方法を示したものです。特に、サービスプロパティーの変更方法と、サービスまたはサービスインスタンスに関するその他の構成情報を示します。

SMF サービスの構成 (タスクマップ)

次のタスクマップに、SMF サービスを構成するために必要な手順を示します。

タスク	説明	参照先
サービスを変更します。	指定されたサービスインスタンスのサービスプロパティーを変更します。	42 ページの「SMF サービスプロパティーの変更方法」
	指定されたサービスインスタンスの複数のプロパティーを変更します。	42 ページの「1つのサービスの複数プロパティーを変更する方法」

タスク	説明	参照先
	<p>サービスインスタンスの構成ファイルを変更します。</p> <p>サービスインスタンスの環境変数を変更します。</p> <p>サービスに対する <code>admin</code> レイヤーでのカスタマイズを削除します。</p>	<p>43 ページの「ファイルで構成されているサービスの変更方法」</p> <p>44 ページの「サービスの環境変数を変更する方法」</p> <p>46 ページの「サービスのカスタマイズを削除する方法」</p>
inetd サービスを変更します。	<p>inetd によって制御されるサービスの構成プロパティを変更します。</p> <p>inetd によって制御されるサービスの起動オプションを変更します。</p>	<p>45 ページの「inetd 制御サービスのプロパティを変更する方法」</p> <p>47 ページの「inetd 制御サービスのコマンド行引数を変更する方法」</p>
inetd.conf エントリを変換します。	inetd サービスを、SMF でモニター可能な従来の実行サービスに変換します。	47 ページの「inetd.conf エントリを変換する方法」

▼ SMF サービスプロパティの変更方法

この手順は、サービスを起動できるユーザーを特定するプロパティの変更方法を示しています。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 **start/user** プロパティに割り当てられた値を変更します。
最初に、FMRI を指定して適切なサービスを特定します。次に、サービスを起動する UID を割り当てます。

```
# svccfg -s FMRI
svc:/service: setprop start/user = astring: newlogin
```
- 3 サービスをリフレッシュします。

```
# svcadm refresh FMRI
```

▼ 1つのサービスの複数プロパティを変更する方法

この手順では、`setprop` サブコマンドで必要な多くの情報を入力することなく複数のサービスプロパティを変更する方法を示します。この例では、いくつかの DNS サービスプロパティが変更されます。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 プロパティー値を変更します。

`editprop` サブコマンドは、指定されたサービスのプロパティーグループとプロパティーを一時ファイルに収集し、`VISUAL` または `EDITOR` 環境変数によって名前が指定されるプログラムが起動されてファイルが編集されます。詳細は、`svccfg(1M)` のマニュアルページを参照してください。

```
# svccfg -s network/dns/client editprop

##
## The value used to construct the "nameserver" directive in resolv.conf(4).
##
setprop config/nameserver = net_address: (192.168.1.10 192.168.1.11)

##
## The value used to construct the "search" directive in resolv.conf(4).
##
# setprop config/search = astring: "example.com sales.example.com"
この時点でファイルを保存しないでください。
```

- 3 変更内容をすべてのインスタンスに適用します。

ファイルを保存してエディタを終了するときこれらの変更を適用するには、ファイルの下部にある `refresh` 行のコメントを解除します。

```
## Uncomment to apply these changes to all instances of this service.
refresh
```

▼ ファイルで構成されているサービスの変更方法

次の手順は、`inetd` サービスによって管理されていないサービスの構成を変更する方法を示します。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 必要に応じて構成ファイルを変更します。

多くのサービスは、起動方法などの構成情報を定義するための設定ファイルを1つ以上持っています。それらのファイルはサービス実行中でも変更可能です。ファイルの内容がチェックされるのはサービス起動時です。

- 3 サービスを再起動します。

```
# svcadm restart FMRI
```

例 2-15 新しい NTP サーバーを追加する

NTP クライアントをサポートするために新しい NTP サーバーを追加するには、そのサーバー用の新しいエントリを `/etc/inet/ntp.conf` ファイルに追加します。次に、NTP サービスを再起動します。この例は、`ntp.conf` ファイルの内容とサービスの再起動方法を示しています。

```
# cat /etc/inet/ntp.conf
.
.
server ntpserver1.example.com
server ntpserver2.example.com
# svcadm restart svc:/network/ntp:default
```

▼ サービスの環境変数を変更する方法

この手順は、デバッグに役立つよう cron 環境変数を変更する方法を示します。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『[Oracle Solaris 11.1 の管理: セキュリティーサービス](#)』の「[割り当てられている管理権限を使用する方法](#)」を参照してください。

- 2 サービスが実行されていることを確認します。

```
# svcs system/cron
STATE          STIME    FMRI
online         Dec_04   svc:/system/cron:default
```

- 3 環境変数を設定します。

次の例では、環境変数 `UMEM_DEBUG` および `LD_PRELOAD` を設定しています。setenv サブコマンドについては、[svccfg\(1M\)](#) のマニュアルページを参照してください。

```
# svccfg -s system/cron:default setenv UMEM_DEBUG default
# svccfg -s system/cron:default setenv LD_PRELOAD libumem.so
```

- 4 サービスのリフレッシュと再起動を行います。

```
# svcadm refresh system/cron
# svcadm restart system/cron
```

- 5 正しく変更されたことを確認します。

```
# pargs -e 'pgrep -f /usr/sbin/cron'
100657: /usr/sbin/cron
envp[0]: LOGNAME=root
envp[1]: LD_PRELOAD=libumem.so
envp[2]: PATH=/usr/sbin:/usr/bin
envp[3]: SMF_FMRI=svc:/system/cron:default
envp[4]: SMF_METHOD=/lib/svc/method/svc-cron
envp[5]: SMF_RESTARTER=svc:/system/svc/restarter:default
envp[6]: TZ=GB
envp[7]: UMEM_DEBUG=default
```

▼ inetd 制御サービスのプロパティを変更する方法

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 特定のサービスのプロパティを一覧表示します。
このコマンドを実行すると、FMRI で指定されたサービスのすべてのプロパティが表示されます。

```
# inetadm -l FMRI
```
- 3 サービスのプロパティを変更します。
inetd 制御サービスの各プロパティは、プロパティ名とその値によって定義されます。値を指定せずにプロパティ名を指定した場合、そのプロパティはデフォルト値にリセットされます。特定のサービスのプロパティの詳細については、そのサービスのマニュアルページを参照してください。

```
# inetadm -m FMRI property-name=value
```
- 4 プロパティが変更されたことを確認します。
プロパティを再度一覧表示し、適切に変更されていることを確認します。

```
# inetadm -l FMRI
```
- 5 変更が適用されていることを確認します。
プロパティに対する変更が期待どおりに適用されていることを確認します。

例 2-16 telnet の tcp_trace プロパティを変更する

次の例は、telnet の tcp_trace プロパティを true に設定する方法を示しています。telnet コマンドの実行後の syslog 出力の結果から、変更が適用されていることがわかります。

```
# inetadm -l svc:/network/telnet:default
SCOPE      NAME=VALUE
           name="telnet"
.
.
default inherit_env=TRUE
default tcp_trace=FALSE
default tcp_wrappers=FALSE
# inetadm -m svc:/network/telnet:default tcp_trace=TRUE
# inetadm -l svc:/network/telnet:default
SCOPE      NAME=VALUE
           name="telnet"
.
.
```

```

.
default inherit_env=TRUE
      tcp_trace=TRUE
default tcp_wrappers=FALSE
# telnet localhost
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
login: root
Password:
.
.
.
Last login: Mon Jun 21 05:55:45 on console
Sun Microsystems Inc. SunOS 5.10 s10_57 May 2004
# ^D
Connection to localhost closed by foreign host.
# tail -1 /var/adm/messages
Jun 21 06:04:57 yellow-19 inetd[100308]: [ID 317013 daemon.notice] telnet[100625]
      from 127.0.0.1 32802

```

▼ サービスのカスタマイズを削除する方法

プロパティ、プロパティグループ、およびサービスインスタンスからカスタマイズを削除できます。

- ローカルのカスタマイズを削除します。
このコマンドを実行すると、選択したサービスの admin レイヤーでの変更がすべて削除されます。

```
% /usr/sbin/svccfg -s FMRI delcust
```

例 2-17 プロパティからカスタマイズを削除する

この例では、サービスからカスタマイズを削除した影響について説明します。最初に、rmtmpfiles サービスの clean_vartmp プロパティを設定することでブートプロセス中に /var/tmp が消去されるようするカスタマイズを適用します。特権ユーザータイプとして次のコマンドを実行します。

```
# svccfg -s svc:/system/rmtmpfiles setprop options/clean_vartmp = true
# svcadm refresh svc:/system/rmtmpfiles
```

次に、プロパティが変更されていることを確認します。

```
$ svcprop -p options/clean_vartmp svc:/system/rmtmpfiles
true
```

カスタマイズを確認するには、次のコマンドを入力します。

```
$ svccfg -s svc:/system/rmtmpfiles listcust
options/clean_vartmp    boolean    admin    true
```

```
$ svcprop -p options/clean_vartmp -l all svc:/system/rmtmpfiles
options/clean_vartmp boolean manifest false
options/clean_vartmp boolean admin true
```

カスタマイズを削除するには、特権ユーザーとして次のコマンドを入力します。

```
# svccfg -s svc:/system/rmtmpfiles delcust options/clean_vartmp
Deleting customizations for property: options/clean_vartmp
```

ここでカスタマイズを確認するためのコマンドを実行すると、次のようになります。

```
$ svccfg -s svc:/system/rmtmpfiles listcust
$ svcprop -p options/clean_vartmp -l all svc:/system/rmtmpfiles
options/clean_vartmp boolean manifest false
```

▼ inetd 制御サービスのコマンド行引数を変更する方法

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 特定のサービスの **exec** プロパティを表示します。
このコマンドを実行すると、FMRI で指定されたサービスのすべてのプロパティが表示されます。grep コマンドを追加することで、サービスの exec プロパティに出力を制限しています。

```
# inetadm -l FMRI|grep exec
```
- 3 サービスの **exec** プロパティを変更します。
exec プロパティに設定される *command-syntax* は、サービス起動時に実行されるコマンド文字列を定義します。

```
# inetadm -m FMRI exec="command-syntax"
```
- 4 プロパティが変更されたことを確認します。
プロパティを再度一覧表示し、適切に変更されていることを確認します。

```
# inetadm -l FMRI
```

▼ inetd.conf エントリを変換する方法

次の手順では、inetd.conf エントリを SMF サービス目録に変換します。inetd に依存するサードパーティーのアプリケーションをシステムに追加するたびに、この手順を実行する必要があります。また、/etc/inetd.conf 内のエントリに構成変更を行う必要がある場合にも、この手順を実行してください。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 **inetd.conf** エントリを変換します。
inetconv コマンドは、選択されたファイル内の各エントリをサービス目録に変換します。

```
# inetconv -i filename
```

例 2-18 /etc/inet/inetd.conf エントリを SMF サービス目録に変換する

```
# inetconv -i /etc/inet/inetd.conf
```

実行制御スクリプトの使用

次の手順は、実行制御スクリプトを使用してレガシーサービスを停止または開始する方法を示しています。また、実行制御スクリプトを追加または削除する手順も示しています。

実行制御スクリプトの使用 (タスクマップ)

次のタスクマップには、実行制御スクリプトの使用に関する複数の手順が含まれています。各行には、タスク、そのタスクを行う場面の説明、およびタスクへのリンクが含まれます。

タスク	説明	参照先
レガシーサービスを停止または起動します。	実行制御スクリプトを使用してサービスを起動または停止します。	49 ページの「実行制御スクリプトを使用して従来のサービスを起動または停止する方法」
実行制御スクリプトを追加します。	実行制御スクリプトを作成し、それを /etc/init.d ディレクトリに追加します。	49 ページの「実行制御スクリプトを追加する方法」
実行制御スクリプトを無効にします。	ファイル名を変更することで実行制御スクリプトを無効にします。	50 ページの「実行制御スクリプトを無効にする方法」
SMF を使用するよう実行制御スクリプトを変換します。	実行制御スクリプトを SMF マニフェストに置き換えることにより、サービスを SMF サービスで管理できるようにします。	51 ページの「実行制御スクリプトを SMF サービスに変換する方法」

▼ 実行制御スクリプトを使用して従来のサービスを起動または停止する方法

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 システムサービスを停止します。

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

```
# /etc/init.d/filename start
```
- 4 サービスが停止または起動されたことを確認します。

```
# pgrep -f service
```

例 2-19 実行制御スクリプトを使用してサービスを起動または停止する

たとえば、次のように入力することによって PPPoE サーバーデーモンを停止できます。

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

次に、PPPoE サーバーデーモンを再開するには、次のように入力します。

```
# /etc/init.d/pppd start
# pgrep -f ppp
101748
# pgrep -f ppp -d, | xargs ps -fp
  UID    PID  PPID   C   STIME TTY          TIME CMD
daemon 101748    1    0   Sep 01 ?        0:06 /usr/lib/inet/pppoed
```

▼ 実行制御スクリプトを追加する方法

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

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

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 スクリプトを `/etc/init.d` ディレクトリに追加します。

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

- 3 適切な `rcn.d` ディレクトリへのリンクを作成します。

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

- 4 スクリプトのリンクが指定のディレクトリ内にあることを確認します。

```
# ls /etc/init.d/*filename /etc/rc2.d/*filename /etc/rcn.d/*filename
```

例 2-20 実行制御スクリプトを追加する

次の例は、xyz サービスの実行制御スクリプトを追加する方法を示しています。

```
# cp xyz /etc/init.d
# chmod 0744 /etc/init.d/xyz
# chown root:sys /etc/init.d/xyz
# cd /etc/init.d
# ln xyz /etc/rc2.d/S99xyz
# ln xyz /etc/rc0.d/K99xyz
# ls /etc/init.d/*xyz /etc/rc2.d/*xyz /etc/rc0.d/*xyz
```

▼ 実行制御スクリプトを無効にする方法

実行制御スクリプトを無効にするには、ファイル名の先頭に下線 (`_`) を付けてファイル名を変更します。下線またはドット (`.`) で始まるファイルは実行されません。接尾辞を追加してファイルをコピーすると、両方のファイルが実行されます。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 新しいファイルの先頭に下線 (`_`) を付けて、スクリプト名を変更します。

```
# cd /etc/rcn.d
# mv filename_filename
```

- 3 スクリプトの名前が変更されたことを確認します。

```
# ls _*
_filename
```

例 2-21 実行制御スクリプトを無効にする

次の例は、S99datainit スクリプトの名前を変更する方法を示しています。

```
# cd /etc/rc2.d
# mv S99datainit _S99datainit
# ls _*
_S99datainit
```

▼ 実行制御スクリプトを **SMF** サービスに変換する方法

この手順では、既存の実行制御スクリプトをサポートする SMF の利点を使用する方法を示します。詳細は、`svcbundle` (1M) のマニュアルページを参照してください。

始める前に 最初に、スクリプトが必要とするサービスモデルを確認します。スクリプトが何らかの作業を実行し、長期間実行するプロセスを開始せずに終了するのであれば、ほとんどの場合は一時的なサービスです。スクリプトが長時間実行するデーモンを開始してから終了する場合、それは契約サービスです。サービスモデルについての詳細は、`svc.startd(1M)` のマニュアルページの期間セクションを参照してください。

次に、サービスに対して適切な実行レベルを確認します。これは、`/etc/rc3.d` などのスクリプトがリンクされているディレクトリを表示することで確認できます。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 一時的なマニフェストを作成します。
この例で、サービス名は `ex_svc` です。これは契約サービスで、レベル 3 で実行します。

```
# svcbundle -o /tmp/ex_svc.xml -s service-name=ex_svc
-s start-method=/lib/svc/method/ex_svc -s model=transient
```
- 3 マニフェストへの追加変更を行います。
`/tmp/ex_svc.xml` 内のマニフェストが、必要な内容であることを確認します。
- 4 マニフェストを適切なディレクトリにコピーします。

```
# cp /tmp/ex_svc.xml /lib/svc/manifest/site/ex_svc.xml
```
- 5 既存のサービスを停止します。

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

- 6 実行制御スクリプトを無効にします。
該当する `rc n.d` ディレクトリからの実行制御スクリプトへのリンクを削除します。
- 7 マニフェストをインポートし、サービスを開始します。

```
# svcadm restart svc:/system/manifest-import
```

サービス管理機能のトラブルシューティング

次の各作業は、SMF サービスのトラブルシューティングまたは修正の方法を示したものです。これらの手順の一部では、システムのブート方法を変更するためにブートパラメータを変更する方法も示します。

SMF のトラブルシューティング (タスクマップ)

次のタスクマップには、システムの問題をトラブルシューティングするために使用できる複数の手順が含まれています。各行には、タスク、そのタスクを行う場面の説明、およびタスクへのリンクが含まれます。

タスク	説明	参照先
実行中でないサービスをデバッグする	サービスが実行しない原因に関する情報を SMF サービスから収集します。	53 ページの「起動していないサービスをデバッグする方法」
破壊されたりポジトリを修復する	破壊されたりポジトリを交換してシステムが完全にブートするようにする際に実行する手順。	53 ページの「破壊されたりポジトリを修復する方法」
どのサービスも起動しないでブートする	ブートプロセス中にハングアップするシステムをデバッグするために、すべてのサービスを実行させずにシステムをブートします。	56 ページの「どのサービスも起動しないでブートする方法」
より多くのブートメッセージを生成する	デバッグを支援するために、ブートプロセス中に多くのメッセージを作成する手順。	56 ページの「詳細モードでブートする方法」
ローカルファイルシステムが使用できない場合にブートプロセスを停止する	ローカルファイルシステムがマウントされていない場合に、ログインサービスを起動することで、問題を調査できるようにするための手順。	57 ページの「ブート中に <code>system/filesystem/local:default</code> サービスで障害が発生した場合に <code>sulogin</code> プロンプトを強制的に表示する方法」

▼ 起動していないサービスをデバッグする方法

この手順では、印刷サービスが無効になっています。

- 1 管理者になるか、**Service Management** 権利プロファイルを含む役割になります。
詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。
- 2 エラーのために実行されていないサービスに関する情報を要求します。

```
# svcs -xv
svc:/application/print/server:default (LP Print Service)
  State: disabled since Wed 13 Oct 2004 02:20:37 PM PDT
  Reason: Disabled by an administrator.
  See: http://support.oracle.com/msg/SMF-8000-05
  See: man -M /usr/share/man -s 1M lpsched
Impact: 2 services are not running:
  svc:/application/print/rfc1179:default
  svc:/application/print/ipp-listener:default
```

-x オプションを指定すると、影響を受けたサービスインスタンスに関する情報も表示されます。

- 3 サービスを有効にします。
svcadm enable application/print/server

▼ 破壊されたリポジトリを修復する方法

この手順は、破壊されたリポジトリをデフォルトのリポジトリと交換する方法を示しています。リポジトリデーモン svc.configd が起動すると、構成リポジトリの整合性チェックが行われます。このリポジトリは、/etc/svc/repository.db に格納されています。このリポジトリは、次のいずれかが原因で破壊されることがあります。

- ディスク障害
- ハードウェアのバグ
- ソフトウェアのバグ
- 過失によるファイルの上書き

整合性チェックに失敗すると、svc.configd デーモンは次のようなメッセージをコンソールに書き出します。

```
svc.configd: smf(5) database integrity check of:
```

```
  /etc/svc/repository.db
```

```
failed. The database might be damaged or a media error might have
prevented it from being verified. Additional information useful to
your service provider is in:
```

```
/system/volatile/db_errors
```

The system will not be able to boot until you have restored a working database. `svc.startd(1M)` will provide a `sulogin(1M)` prompt for recovery purposes. The command:

```
/lib/svc/bin/restore_repository
```

can be run to restore a backup version of your repository. See <http://support.oracle.com/msg/SMF-8000-MY> for more information.

その後 `svc.startd` デーモンが終了し、`sulogin` が起動して保守作業を行えるようになります。

1 sulogin プロンプトで root パスワードを入力します。

`sulogin` コマンドでは、`root` ユーザーはシステム保守モードに入ってシステムを修復できます。

2 次のコマンドを実行します:

```
# /lib/svc/bin/restore_repository
```

このコマンドを実行すると、破壊されていないバックアップの復元に必要な手順が示されます。SMF では、システムの重要な局面でリポジトリのバックアップが自動的に作成されます。詳細は、[18 ページの「SMF リポジトリのバックアップ」](#)を参照してください。

`/lib/svc/bin/restore_repository` コマンドが起動すると、次のようなメッセージが表示されます。

```
See http://support.oracle.com/msg/SMF-8000-MY for more information on the use of
this script to restore backup copies of the smf(5) repository.
```

```
If there are any problems which need human intervention, this script will
give instructions and then exit back to your shell.
```

書き込み権を付けてルート (/) ファイルシステムをマウントしたあと、またはシステムがローカルゾーンである場合は、復元するリポジトリのバックアップを選択するよう求められます。

```
The following backups of /etc/svc/repository.db exists, from
oldest to newest:
```

```
... list of backups ...
```

バックアップには、バックアップのタイプとバックアップが作成された時間に基づいて名前が付けられています。`boot` で始まっているのは、システムのブート後、リポジトリに対して最初の変更が行われる前に作成されたバックアップです。`manifest import` で始まっているのは、`svc:/system/manifest-import:default` のプロセス終了後に作成されたバックアップです。バックアップ時間は、`YYYYMMDD_HHMMSS` 形式で記録されます。

3 適切な応答を入力します。

通常は、最新のバックアップオプションを選択します。

Please enter either a specific backup repository from the above list to restore it, or one of the following choices:

CHOICE	ACTION
-----	-----
boot	restore the most recent post-boot backup
manifest_import	restore the most recent manifest_import backup
-seed-	restore the initial starting repository (All customizations will be lost, including those made by the install/upgrade process.)
-quit-	cancel script and quit

Enter response [boot]:

復元するバックアップを指定しないでEnterを押した場合は、[]で囲まれたデフォルトの応答が選択されます。-quit-を選択すると、restore_repository スクリプトが終了して、シェルスクリプトに戻ります。

注--seed-を選択すると、seedリポジトリが復元されます。このリポジトリは、初期インストールとアップグレード時に使用する目的で作成されたものです。seedリポジトリを回復の目的で使用するのには、最後の手段にしてください。

復元するバックアップを選択すると、妥当性検証が行われ、その整合性がチェックされます。なんらかの問題があると、restore_repository コマンドによってエラーメッセージが表示され、別のバックアップを選択するよう促されます。有効なバックアップを選択すると、次の情報が表示され、最終確認を入力するよう促されます。

After confirmation, the following steps will be taken:

```

svc.startd(1M) and svc.configd(1M) will be quiesced, if running.
/etc/svc/repository.db
-- renamed --> /etc/svc/repository.db_old_YYYYMMDD_HHMMSS
/system/volatile/db_errors
-- copied --> /etc/svc/repository.db_old_YYYYMMDD_HHMMSS_errors
repository_to_restore
-- copied --> /etc/svc/repository.db
and the system will be rebooted with reboot(1M).

```

Proceed [yes/no]?

4 yes と入力して障害を修復します。

restore_repository コマンドが表示されたアクションをすべて実行すると、システムがリポートします。

▼ どのサービスも起動しないでブートする方法

サービスの起動に関する問題が発生した場合、ブート時にシステムがハングアップする可能性があります。この手順は、そうした問題のトラブルシューティングを行う方法を示しています。

- 1 どのサービスも起動しないでブートします。

このコマンドを実行すると、`svc.startd` デーモンはすべてのサービスを一時的に無効にし、コンソール上で `sulogin` を起動します。

```
ok boot -m milestone=none
```

- 2 システムに `root` としてログインします。

- 3 すべてのサービスを有効にします。

```
# svcadm milestone all
```

- 4 ブートプロセスがどこでハングアップするのかを確認します。

ブートプロセスがハングアップしたら、動作していないサービスを確認するために、`svcs -a` を実行します。`/var/svc/log` のログファイル内でエラーメッセージの有無を確認します。

- 5 問題が解決したら、すべてのサービスが起動していることを確認します。

- a. 必要なサービスがすべてオンラインになっていることを確認します。

```
# svcs -x
```

- b. `console-login` サービスの依存関係に問題がないことを確認します。

このコマンドを使えば、コンソール上の `login` プロセスが実行されるかどうかを確認できます。

```
# svcs -l system/console-login:default
```

- 6 通常のブートプロセスを継続します。

▼ 詳細モードでブートする方法

デフォルトでは、ブート中に表示されるメッセージのレベルは非表示モードに設定されており、この場合はサービスが起動してエラーが発生したときにメッセージが生成されます。システムのブート時に発生する問題をトラブルシューティングするとき、より多くのメッセージを表示したい場合もあります。この手順では、すべてのエラーメッセージが表示されるようにシステムをブートする方法を示します。

- 詳細モードでブートします。

```
# boot -m verbose
```

例 2-22 ブートメッセージをログに保管する

`boot` コマンドに `-m debug` オプションを指定すると、各サービスおよび各サービス起動についてのすべてのメッセージがログファイルに記録されます。

▼ ブート中に **system/filesystem/local:default** サービスで障害が発生した場合に **sulogin** プロンプトを強制的に表示する方法

システムのブートに必要なでないローカルファイルシステムは、`svc:/system/filesystem/local:default` サービスによってマウントされます。これらのファイルシステムのいずれかがマウントできない場合、このサービスは保守状態になります。システムの起動は続行され、`filesystem/local` に依存しないサービスは起動されます。依存関係を介して起動されるサービスで、オンラインになるために `filesystem/local` を必要とするものは、起動されません。

サービスで障害が発生した場合に、システムの起動の続行を許可する代わりに、ただちに `sulogin` プロンプトを表示するようシステムの構成を変更するには、次の手順に従ってください。

1 `system/console-login` サービスを変更します。

```
# svccfg -s svc:/system/console-login
svc:/system/console-login> addpg site,filesystem-local dependency
svc:/system/console-login> setprop site,filesystem-local/entities = fmri: svc:/system/filesystem/local

svc:/system/console-login> setprop site,filesystem-local/grouping = astring: require_all

svc:/system/console-login> setprop site,filesystem-local/restart_on = astring: none

svc:/system/console-login> setprop site,filesystem-local/type = astring: service

svc:/system/console-login> end
```

2 サービスをリフレッシュします。

```
# svcadm refresh console-login
```

注意事項 `system/filesystem/local:default` サービスで障害が発生したときは、`svcs -vx` コマンドを使用して障害を特定するようにしてください。障害が修正されたら、次のコマンドでエラー状態をクリアし、システムのブートを続行できるようにします。

```
svcadm clear filesystem/local.
```


障害管理の使用

Oracle Solaris OS には、予測的自己修復が可能なシステムおよびサービスを構築して配備するためのアーキテクチャーが含まれています。障害管理アーキテクチャー (FMA) の中核となるサービスは、ハードウェアおよびソフトウェアのエラーに関するデータを受け取り、ベースとなる問題を自動的に診断し、障害の発生したコンポーネントをオフラインにすることによって対応します。

この章の内容は次のとおりです。

- 59 ページの「障害管理の概要」
- 61 ページの「障害および欠陥の通知」
- 62 ページの「障害または欠陥に関する情報の表示」
- 66 ページの「障害または欠陥の修復」
- 67 ページの「障害管理のログファイル」
- 68 ページの「障害の統計情報」

障害管理の概要

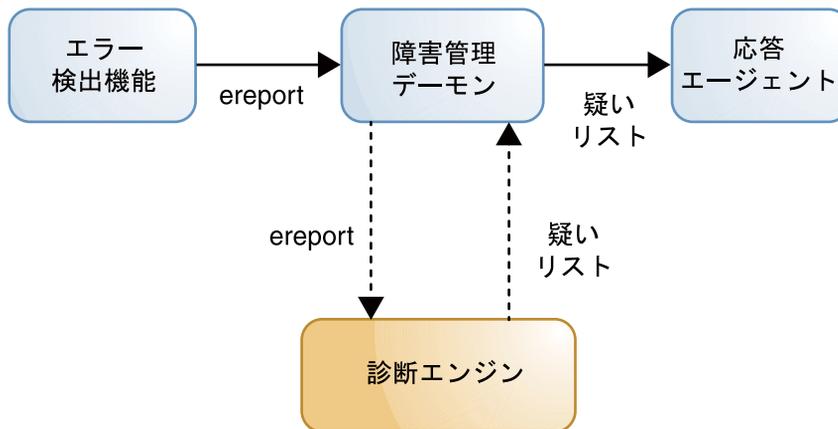
Oracle Solaris の障害管理機能は、耐障害性のあるエラーハンドラ、構造化されたエラー遠隔測定、自動化された診断ソフトウェア、応答エージェント、および構造化されたメッセージングを構築するためのアーキテクチャーを備えています。障害管理には、CPU、メモリー、I/O の各サブシステム、Oracle Solaris ZFS、増え続ける一連のデバイスドライバ、その他の管理スタックなど、ソフトウェアスタックの多くの部分に関与します。

FMA の目的は、Oracle Solaris システム上で発生する問題の解決を支援することです。問題は、以前機能していた何かが機能しなくなったことを意味する障害である場合があります。問題は、一度も正しく機能しなかったことを意味する欠陥である場合もあります。一般に、ハードウェアには障害と欠陥の両方が発生する可能性があります。しかし、ソフトウェアのほとんどの問題は欠陥であるか、または構成上の問題が原因です。

高レベルの障害管理スタックには、エラー検出機能、診断エンジン、および応答エージェントが含まれています。エラー検出機能は、その名前が示すように、システム内のエラーを検出し、必要な対応をただちに実行します。エラー検出機能は、明確に定義されたエラーレポート (*ereport*) を診断エンジンに対して発行します。診断エンジンは、*ereport* を解釈し、システムに障害や欠陥が存在するかどうかを判定します。このような判定を行うときに、診断エンジンは問題の原因である可能性があるリソースまたはリソースのセットを示す疑いリストを発行します。リソースには、現場交換可能ユニット (FRU)、ラベル、または自動システム再構成ユニット (ASRU) が関連付けられている場合と、関連付けられていない場合があります。ASRU は、FRU が交換されるまでは問題を軽減するためにサービスからただちに除外してもかまいません。

疑いリストに複数の疑いが含まれる場合 (たとえば、診断エンジンが疑いを1つだけに特定できない場合など) は、それぞれの疑いに主要な疑いである確率が割り当てられます。リスト内の確率を合計すると 100 パーセントになります。疑いリストは応答エージェントによって解釈されます。応答エージェントは、疑いリストに基づいていくつかの操作を試行します。応答には、メッセージの記録、CPU ストランドのオフライン化、メモリーページの取得、I/O デバイスの廃棄などが含まれます。

エラー検出機能、診断エンジン、および応答エージェントは、障害管理デーモン *fmd* によって接続されます。*fmd* は、次の図に示すように、さまざまなコンポーネント間のマルチプレクサとして機能します。



障害管理デーモンは、それ自身が SMF の制御下にあるサービスです。このサービスはデフォルトで有効になっており、ほかの SMF サービスと同じように制御されます。詳細は、*smf(5)* のマニュアルページを参照してください。

FMA サービスと SMF サービスは、必要に応じて相互にやりとりします。一部のハードウェアの問題が原因で、SMF がサービスを停止したり、再起動したりすることがあります。また、一部の SMF エラーが原因で、FMA が欠陥を報告することもあります。

障害および欠陥の通知

多くの場合、障害管理デーモンとの最初のやりとりは、障害または欠陥が診断されたことを示すシステムメッセージです。メッセージはコンソールと `/var/adm/messages` ファイルの両方に送信されます。障害管理デーモンから送信されるすべてのメッセージで、次の形式が使用されます。

```
1 SUNW-MSG-ID: SUN4V-8001-8H, TYPE: Fault, VER: 1, SEVERITY: Minor
2 EVENT-TIME: Wed Aug 24 21:56:03 UTC 2011
3 PLATFORM: SUNW,T5440, CSN: -, HOSTNAME: bur419-61
4 SOURCE: cpumem-diagnosis, REV: 1.7
5 EVENT-ID: 7b83c87c-78f6-6a8e-fa2b-d0cf16834049
6 DESC: The number of integer register errors associated with this thread has
7 exceeded acceptable levels.
8 AUTO-RESPONSE: The fault manager will attempt to remove the affected thread
9 from service.
10 IMPACT: System performance may be affected.
11 REC-ACTION: Use 'fmadm faulty' to provide a more detailed view of this
12 event. Please refer to the associated reference document at
13 http://support.oracle.com/msg/SUN4V-8001-8H for the latest service procedures and
14 policies regarding this diagnosis.
```

診断された問題が通知されたら、必ず推奨されたナレッジ記事で詳細を参照してください。具体例については、上記の 13 行目を参照してください。ナレッジ記事には、ユーザーまたはサービスプロバイダが 11 行目に記載されていること以外に取るべき追加アクションが含まれている場合があります。

障害管理エラーイベントの通知は、Simple Network Management Protocol (SNMP) または Simple Mail Transfer Protocol (SMTP) を使用して構成できます。手順については、[33 ページの「SMF 遷移イベントの通知を設定する方法」](#)を参照してください。

さらに、Oracle Auto Service Request を構成して、特定のハードウェアの問題が発生したときに Oracle サービスを自動的に要求することもできます。この機能の詳細については、[Oracle Auto Service Request の製品ページ](#)を参照してください。このページのドキュメントリンクを開くと、『Oracle ASR Quick Installation Guide』および『Oracle ASR Installation and Operations Guide』へのリンクがあります。

障害または欠陥に関する情報の表示

障害や欠陥の情報を表示したり、関係する FRU を特定したりするための方法としては、`fmadm faulty` コマンドを推奨します。ただし、`fmdump` コマンドもサポートされています。`fmdump` はシステム上の問題の履歴ログを表示するために使用されることが多く、`fmadm faulty` はアクティブな問題を表示するために使用されます。



注意 - 管理アクションは、`fmdump` コマンドの出力ではなく `fmadm faulty` の出力に基づいて決定してください。ログファイルには、障害や欠陥と見なすべきでないエラー文が含まれていることがあります。

▼ 障害の発生したコンポーネントに関する情報を表示する方法

- 1 管理者になります。

詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 コンポーネントに関する情報を表示します。

```
# fmadm faulty
```

生成されるテキストの説明については、次の例を参照してください。

例 3-1 障害の発生した CPU が 1 つある場合の `fmadm` の出力

```
1 # fmadm faulty
2 -----
3 TIME                EVENT-ID                MSG-ID                SEVERITY
4 -----
5 Aug 24 17:56:03 7b83c87c-78f6-6a8e-fa2b-d0cf16834049  SUN4V-8001-8H  Minor
6
7 Host                : bur419-61
8 Platform            : SUNW,T5440          Chassis_id : BEL07524BN
9 Product_sn         : BEL07524BN
10
11 Fault class        : fault.cpu.ultraSPARC-T2plus.ireg
12 Affects           : cpu:///cpuid=0/serial=1F95806CD1421929
13                   : faulted and taken out of service
14 FRU                : "MB/CPU0" (hc:///product-id=SUNW,T5440;server-id=bur419-61:\
15                   : serial=3529;part=541255304/motherboard=0/cpuboard=0)
16                   : faulty
17 Serial ID.        : 3529
18                   : 1F95806CD1421929
19
20 Description        : The number of integer register errors associated with this thread
21                   : has exceeded acceptable levels.
```

```

22
23 Response      : The fault manager will attempt to remove the affected thread from
24                 service.
25
26 Impact       : System performance may be affected.
27
28 Action       : Use 'fmdm faulty' to provide a more detailed view of this event.
29                 Please refer to the associated reference document at
30                 http://support.oracle.com/msg/SUN4V-8001-8H for the latest service
31                 procedures and policies regarding this diagnosis.

```

もっとも重要なのは、影響を受ける FRU に関するデータが表示された 14 行目です。可読形式の場所の文字列が引用符で囲んで表示されています ("MB/CPU0")。引用符で囲まれた値は、物理ハードウェアに付けられたラベルと一致させるためのものです。FRU は、障害管理リソース識別子 (FMRI) 形式でも表示されます。FMRI には、ホスト名やシャーシのシリアル番号など、障害を含むシステムに関する記述プロパティが含まれています。FRU をサポートするプラットフォームでは、FRU の FMRI に FRU のパート番号とシリアル番号も含まれています。

「Affects」行 (12 - 13 行目) には、障害または欠陥とその関連する状態の影響を受けるコンポーネントが示されます。この例では、1 つの CPU ストランドが影響を受け、「faulted and taken out of service (障害が発生して使用不可)」となっています。

この `fmdm faulty` コマンドの出力では、FRU の説明に続いて、16 行目に状態が「faulty」として表示されています。「Action」セクションには、通常の `fmdm` コマンドの参照に代わって (または追加して) ほかのアクションが含まれる場合もあります。

例 3-2 複数の障害がある場合の `fmdm` の出力

```

1  # fmdm faulty
2  -----
3  TIME                EVENT-ID                MSG-ID                SEVERITY
4  -----
5  Sep 21 10:01:36    d482f935-5c8f-e9ab-9f25-d0aaafec1e6c  PCIEX-8000-5Y  Major
6
7  Fault class       : fault.io.pci.device-invreq
8  Affects          : dev:///pci@0,0/pci1022,7458@11/pci1000,3060@0
9                   dev:///pci@0,0/pci1022,7458@11/pci1000,3060@1
10                  ok and in service
11                  dev:///pci@0,0/pci1022,7458@11/pci1000,3060@2
12                  dev:///pci@0,0/pci1022,7458@11/pci1000,3060@3
13                  faulty and taken out of service
14  FRU              : "SLOT 2" (hc:///.../pciexrc=3/pciexbus=4/pciexdev=0)
15                   repair attempted
16                   "SLOT 3" (hc:///.../pciexrc=3/pciexbus=4/pciexdev=1)
17                   acquitted
18                   "SLOT 4" (hc:///.../pciexrc=3/pciexbus=4/pciexdev=2)
19                   not present
20                   "SLOT 5" (hc:///.../pciexrc=3/pciexbus=4/pciexdev=3)
21                   faulty

```

```

22
23   Description   : The transmitting device sent an invalid request.
24
25   Response     : One or more device instances may be disabled
26
27   Impact      : Possible loss of services provided by the device instances
28                 associated with this fault
29
30   Action       : Use 'fmadm faulty' to provide a more detailed view of this event.
31                 Please refer to the associated reference document at
32                 http://support.oracle.com/msg/PCIEX-8000-5Y for the latest service
33                 procedures and policies regarding this diagnosis.

```

この `fmadm faulty` コマンドの出力では、FRUの説明に続いて、21行目に状態が「faulty」として表示されています。ほかの状況で表示される可能性がある状態値として、15行目の SLOT 2 に対する「acquitted」や17行目の SLOT 3 に対する「repair attempted」があります。

例 3-3 fmdump コマンドを使用して障害を表示する

コンソールメッセージやナレッジ記事の中には、古い `fmdump -v -u UUID` コマンドを使用して障害情報を表示するように指示するものがあります。 `fmadm faulty` コマンドが推奨されますが、次の例に示すように `fmdump` コマンドも動作します。

```

1  % fmdump -v -u 7b83c87c-78f6-6a8e-fa2b-d0cf16834049
2  TIME                UUID                                SUNW-MSG-ID EVENT
3  Aug 24 17:56:03.4596 7b83c87c-78f6-6a8e-fa2b-d0cf16834049 SUN4V-8001-8H Diagnosed
4    100% fault.cpu.ultraSPARC-T2plus.ireg
5
6      Problem in: -
7      Affects: cpu:///cpuid=0/serial=1F95806CD1421929
8      FRU: hc://:product-id=SUNW,T5440:server-id=bur419-61:\
9      serial=9999:part=541255304/motherboard=0/cpuboard=0
10     Location: MB/CPU0

```

影響を受けた FRU に関する情報も表示されますが、3行(8-10行目)に分かれています。「Location」の文字列には、可読形式の FRU の文字列が表示されます。「FRU」の行には、正式な FMRI が表示されます。 `fmdump` コマンドでは、 `-m` オプションを指定しないかぎり、重要度、説明テキスト、およびアクションが表示されません。詳細は、 [fmdump\(1M\)](#) のマニュアルページを参照してください。

▼ CPU がオフラインかどうかを特定する方法

- CPU に関する情報を表示します。

```

% /usr/sbin/psrinfo
0   faulted   since 05/13/2011 12:55:26
1   on-line   since 05/12/2011 11:47:26

```

faulted 状態は、障害管理の応答エージェントによって CPU がオフラインにされたことを示します。

▼ 欠陥のあるサービスに関する情報を表示する方法

- 1 管理者になります。

詳細は、『Oracle Solaris 11.1 の管理: セキュリティーサービス』の「割り当てられている管理権限を使用する方法」を参照してください。

- 2 欠陥に関する情報を表示します。

```
# fmadm faulty
-----
TIME                EVENT-ID                MSG-ID                SEVERITY
-----
May 12 22:52:47    915cb64b-e16b-4f49-efe6-de81ff96fce7    SMF-8000-YX          major

Host                : parity
Platform            : Sun-Fire-V40z      Chassis_id           : XG051535088
Product_sn          : XG051535088

Fault class         : defect.sunos.smf.svc.maintenance
Affects             : svc:///system/intrd:default
                    faulted and taken out of service
Problem in          : svc:///system/intrd:default
                    faulted and taken out of service

Description         : A service failed - it is restarting too quickly.

Response            : The service has been placed into the maintenance state.

Impact              : svc:/system/intrd:default is unavailable.

Action              : Run 'svcs -xv svc:/system/intrd:default' to determine the
                    generic reason why the service failed, the location of any
                    logfiles, and a list of other services impacted. Please refer to
                    the associated reference document at
                    http://support.oracle.com/msg/SMF-8000-YX for the latest service procedures
                    and policies regarding this diagnosis.
```

- 3 欠陥のあるサービスに関する情報を表示します。

fmadm の出力の「Action」セクションに示された指示に従います。

```
# svcs -xv svc:/system/intrd:default
svc:/system/intrd:default (interrupt balancer)
  State: maintenance since Wed May 12 22:52:47 2010
Reason: Restarting too quickly.
  See: http://support.oracle.com/msg/SMF-8000-YX
  See: man -M /usr/share/man -s 1M intrd
  See: /var/svc/log/system-intrd:default.log
Impact: This service is not running.
```

この問題の詳しい修正手順については、ナレッジ記事 SMF-8000-YX を参照してください。

障害または欠陥の修復

障害管理によってシステム内のコンポーネントに障害が検出されたら、それを修復する必要があります。修復は、暗黙的と明示的のいずれかの方法で行われます。

暗黙的修復は、コンポーネントに障害管理デーモンが追跡できるシリアル番号情報がある場合に、障害の発生したコンポーネントを交換または削除したときに行われます。多くの SPARC システムでは、FMRI にシリアル番号情報が含まれているため、障害管理デーモンはコンポーネントが交換またはその他の手段(たとえば、ブラックリストへの登録など)によって運用から削除されたことを判定できます。このような検出が行われると、障害管理デーモンは影響を受けたリソースを `fmadm faulty` の出力に表示しなくなります。このリソースは、デーモンの内部リソース キャッシュ内に保持され、障害イベントから 30 日が経過した時点で消去されます。

暗黙的修復が適用されないシステムもあります。FMRI にシャーシ ID が存在しても、FRU のシリアル番号情報を使用できない場合があります。そのため、障害管理デーモンは FRU の交換を検出できず、明示的修復が必要になります。

`fmadm` コマンドを使用して、障害を明示的に `repaired` (修復済み) として指定します。このコマンドでは、次の 4 つの構文が修復に関連付けられています。

- `fmadm replaced fmri | label`
- `fmadm repaired fmri | label`
- `fmadm acquit fmri | label`
- `fmadm acquit uuid [fmri | label]`

これら 4 つのコマンドでは、FMRI や UUID を引数として指定することもできますが、ラベルを引数として使用することを推奨します。FRU に複数の障害がある場合は、FRU を一回だけ交換する必要があります。ラベルに対して `fmadm replaced` コマンドを発行すると、FRU がその状態で未処理のケースに反映されます。

`fmadm replaced` コマンド

`fmadm replaced` コマンドを使用して、疑いのある FRU が交換または削除されたことを示すことができます。

FRU が交換された(シリアル番号が変更された)ことをシステムが自動的に検出すると、この検出はコマンド行に `fmadm replaced` を入力した場合と同じように処理されます。`fmadm replaced` コマンドは、FRU が交換された(シリアル番号が変更された)ことを `fmd` が自動的に確認できる場合は許可されません。

FRU が交換ではなく削除されたことをシステムが自動的に検出した場合、現在の動作は変更されません。この疑いは「not present (存在しない)」として表示されますが、永続的に削除されたものとは見なされず、障害イベントから 30 日が経過した時点で消去されます。

fmadm repaired コマンド

FRU の交換ではなく、何らかの物理的な修復が行われて問題が解決したときは、`fmadm repaired` コマンドを使用します。このような修復の例として、カードの再装着や曲がったピンの修正などがあります。

fmadm acquit コマンド

`acquit` (赦免) オプションは、多くの場合、そのリソースが原因ではないことがわかったときに使用します。赦免は、追加のエラーイベントが発生して診断が絞り込まれたときに、暗黙的に行われることもあります。

交換は修復より優先され、交換と修復はどちらも赦免より優先されます。したがって、コンポーネントを赦免してからコンポーネントを修復することはできませんが、すでに修復されているコンポーネントを赦免することはできません。

あるケースが修復された (`FMD_CASE_REPAIRED` 状態に移行し、`list.repaired` イベントが生成された) と見なされるのは、その UUID が赦免されるか、またはすべての疑いが修復、交換、削除、または赦免されたときです。

通常は、`fmd` が複数の要素を含む疑いリスト内の疑いを自動的に赦免するか、サポートサービスから手動で赦免するように指示されます。FMRI またはラベルで赦免する必要があるのは、そのリソースが原因として疑われている現在のすべてのケースで、そのリソースが疑いでないことがわかった場合だけです。ただし、FRU をあるケースについて手動で赦免し、ほかのすべてのケースで疑いのままにするには、次のオプションを使用して UUID と FMRI の両方または UUID とラベルの両方を指定します。

```
fmadm acquit uuid [fmri|label]
```

障害管理のログファイル

障害管理デーモン (`fmd`) は、複数のログファイルに情報を記録します。これらのログファイルは、`/var/fm/fmd` に格納され、`fmdump` コマンドを使用して表示します。詳細は、[fmdump\(1M\)](#) のマニュアルページを参照してください。

- `errlog` ログファイルには、`ereport` からなるインバウンドの遠隔測定情報が記録されます。
- 情報イベントは2つのログファイルに記録されます。`infolog_hival` は重要度の高いイベント用で、`infolog` はその他のすべての情報イベントを収集します。
- `fltlog` ログファイルには、障害の診断および修復イベントが記録されます。



注意 - 管理アクションは、ログファイルの内容ではなく `fmadm faulty` の出力に基づいて決定してください。ログファイルには、障害や欠陥と見なすべきでないエラー文が含まれていることがあります。

ログファイルのローテーションは自動的に実行されます。詳細は、`logadm(1M)` のマニュアルページを参照してください。

障害の統計情報

障害管理デーモン (`fmd`) とそのモジュールの多くは、統計情報を追跡します。`fmstat` コマンドはこれらの統計情報を報告します。オプションを指定せずに `fmstat` を実行すると、ロードされたモジュールのイベント、処理時間、およびメモリー使用量の高レベルの概要が表示されます。例:

```
# fmstat
module          ev_recv ev_acpt wait  svc_t  %w  %b  open solve  memsz  bufisz
cpumem-retire   1       0  0.0  403.5  0   0   0    0    419b   0
disk-transport  0       0  0.0  500.6  0   0   0    0    32b    0
eft             0       0  0.0   4.8   0   0   0    0    1.4M  43b
fmd-self-diagnosis 0       0  0.0   4.7   0   0   0    0    0      0
io-retire       0       0  0.0   4.5   0   0   0    0    0      0
snmp-trapgen    0       0  0.0   4.5   0   0   0    0    32b    0
sysevent-transport 0       0  0.0 1444.4 0   0   0    0    0      0
syslog-msgs     0       0  0.0   4.5   0   0   0    0    0      0
zfs-diagnosis   0       0  0.0   4.7   0   0   0    0    0      0
zfs-retire      0       0  0.0   4.5   0   0   0    0    0      0
```

この出力の各列については、`fmstat(1M)` のマニュアルページを参照してください。`open` 列と `solve` 列は、診断エンジンでのみ作成および解決される障害管理のケースにのみ適用されます。これらの列は、応答エージェントなどのほかのモジュールには無関係です。

`-m module` オプションを使用して、個々のモジュールに関する統計情報を表示できます。この構文は、通常、値がゼロの統計値を抑制する `-z` オプションと共に使用されます。例:

```
# fmstat -z -m cpumem-retire
NAME VALUE      DESCRIPTION
cpu_flts 1          cpu faults resolved
```

この例は、`cpumem-retire` という応答エージェントによって CPU をオフラインにする要求が正常に処理されたことを示しています。

索引

A

acquit オプション, fmadm コマンド, 67
addpg オプション, svccfg コマンド, 22-23
addpropvalue オプション, svccfg コマンド, 22-23
admin レイヤー, 説明, 16-17
all マイルストーン
 説明, 23-24
 例, 56
ASR, FMA, 61
ASRU, 定義, 59-61
Auto Service Request, FMA, 61

B

boot コマンド
 all マイルストーン, 56
 デバッグモード (-m debug), 57
 詳細モード (-m verbose), 56-57
 none マイルストーン, 56

C

config プロパティグループ, 説明, 21

D

debug ログレベル, SME, 18-19
degraded SMF サービス状態, 説明, 14
delcust サブコマンド, svccfg コマンド, 46-47
delcust サブコマンド, svccfg コマンド, 23

delpropvalue サブコマンド, svccfg コマンド, 23
disabled SMF サービス状態, 説明, 14

E

editprop サブコマンド, svccfg コマンド, 43
errlog ログファイル, 67-68
/etc/init.d ディレクトリ, 50
/etc/inittab ファイル, 27
 エントリの説明, 27
 デフォルトの例, 27
/etc/rc*.d, SME, 24
/etc/svc/profile/site プロファイル, 15

F

fltlog ログファイル, 67-68
FMA
 概要, 59-61
 障害の統計情報, 68
 障害または欠陥の修復, 66-67
 情報の表示, 62-65
 通知, 61
 通知パラメータ, 34
 ログファイル, 67-68
fmadm コマンド
 オプション, 66
 概要, 66-67
 例, 62-65
fmdump コマンド
 FMA のログファイル, 67-68

fmdump コマンド (続き)

例, 64

fmd デーモン, 概要, 59-61

FMRI, 説明, 13

fmstat コマンド, 例, 68

FRU, 定義, 59-61

G

general プロパティグループ, 説明, 21

I

inetadm コマンド

説明, 20

例, 45-46

inetd サービス

SMF への変換, 47-48

コマンド行引数の変更, 47

プロパティの変更, 45-46

infolog_hival ログファイル, 67-68

infolog ログファイル, 67-68

initial スナップショット, 説明, 18

init 状態, 「実行レベル」を参照

-i オプション, svcbundle コマンド, 36

L

legacy_run SMF サービス状態, 説明, 14

/lib/svc/bin/restore_repository コマンド,

例, 54

/lib/svc/manifest ファイル, 概要, 14-15

listcust オプション, svccfg コマンド, 22

listpg オプション, svccfg コマンド, 22

listprop オプション, svccfg コマンド, 22

logadm コマンド, FMA, 68

M

maintenance SMF サービス状態, 説明, 14

manifest レイヤー, 説明, 16-17

N

none マイルストーン

説明, 23-24

例, 56

O

offline SMF サービス状態, 説明, 14

online SMF サービス状態, 説明, 14

P

psrinfo コマンド, 例, 64

Q

quiet ログレベル, SMF, 18-19

R

RBAC 権利プロファイル, SMF サービス, 35

repaired オプション, fmadm コマンド, 67

replaced オプション, fmadm コマンド, 66

restarter プロパティグループ, 説明, 21

restore_repository コマンド, 例, 54

running スナップショット, 説明, 18

S

setenv オプション, svccfg コマンド, 22-23

Simple Mail Transfer Protocol (SMTP), FMA, 61

Simple Network Management Protocol (SNMP),

FMA, 61

site-profile レイヤー, 説明, 16-17

SMF

/etc/rc*.d, 24

RBAC 権利プロファイル, 35

委任リスタータ, 21

エラーログ, 18-19

概要, 9

SMF (続き)

- コマンド, 19–20
 - 情報の削除, 23
 - 情報の追加, 22–23
 - 情報の非表示, 23
 - 情報の表示, 22
 - 新機能, 9–10
 - タスク, 29–57
 - 通知パラメータ, 33
 - トラブルシューティング, 52–57
 - ブート, 23–24
 - プロパティ, 21
 - ライブラリインタフェース, 20
 - リスタータデーモン, 20
- SMTP, FMA, 61
- SNMP, FMA, 61
- start スナップショット, 説明, 18
- start プロパティグループ, 説明, 21
- svc.startd デーモン, 説明, 20
- svcadm コマンド, 説明, 19
- svcbundle コマンド
- 説明, 19
 - プロファイルの作成, 39
 - マニフェストの作成, 36
- svccfg コマンド
- delcust サブコマンド, 46–47
 - editprop サブコマンド, 43
 - 説明, 19, 22
- svcprop コマンド
- 説明, 19, 22
- svcs コマンド
- 説明, 20
 - 例, 30
- system-profile レイヤー, 説明, 16–17

U

- uninitialized SMF サービス状態, 説明, 14
- unsetenv サブコマンド, svccfg コマンド, 23
- /usr/bin/svcprop コマンド, 説明, 19
- /usr/bin/svcs コマンド

 - 説明, 20
 - 例, 30

- /usr/lib/fm/fmd/fmd デーモン, 概要, 59–61

- /usr/sbin/fmadm コマンド, 例, 62–65
- /usr/sbin/fmdump コマンド

 - FMA のログファイル, 67–68
 - 例, 64

- /usr/sbin/fmstat コマンド, 例, 68
- /usr/sbin/inetadm コマンド, 説明, 20
- /usr/sbin/logadm コマンド, FMA, 68
- /usr/sbin/psrinfo コマンド, 例, 64
- /usr/sbin/svcadm コマンド

 - 説明, 19

- /usr/sbin/svccfg コマンド

 - 説明, 19, 22

V

- /var/fm/fmd ログファイル, 67–68
- /var/svc/manifest ファイル, 概要, 14–15
- verbose ログレベル, SMF, 18–19

W

- who コマンド, 26

い

- 依存 (SMF), 表示, 32–33
- 依存関係 (SMF)

 - 説明, 12
 - 表示, 31–32

- 一覧表示, 「表示」を参照
- 委任リスタータ, 説明, 21

え

- エラーログ (SMF), 説明, 18–19

か

- 確認, システムの実行レベル (方法), 26
- カスタマイズ (SMF), 表示, 31

環境変数, SMF での変更, 44
管理, サービス (SMF), 34-41
管理レイヤー (SMF), 「レイヤー (SMF)」を参照

け

欠陥 (FMA)
修復, 66-67
情報の表示, 62-65
通知, 61
現場交換可能ユニット, 「FRU」を参照
権利プロファイル, SMF サービス, 35

こ

構成, サービス (SMF), 41-48
構成リポジトリ (SMF), 「リポジトリ」を参照
コマンド (SMF), 一覧, 19-20

さ

サービス (SMF)
管理, 34-41
構成, 41-48
構成ファイル
変更, 43-44
説明, 11
モニタリング, 29-34
有効化, 37
サービス管理機能, 「SMF」を参照
サービス構成リポジトリ, 「リポジトリ」を参照
サービスステータス, 表示, 30-31
サービスのカスタマイズ (SMF), 削除, 46-47
サービスの起動と停止, 49
サービスの状態 (SMF), 説明, 14
再起動, サービス (SMF), 38
削除
「削除」を参照
SMF 情報, 23
サービスのカスタマイズ (SMF), 46-47

作成

SMF プロファイル
自動的, 39
手動, 40-41
SMF マニフェスト, 35-36

し

実行制御スクリプト
SMF サービスへの変換, 51-52
サービスの起動と停止, 49
使用したタスク, 48-52
追加 (方法), 50
無効にする (方法), 50
実行レベル
NFS を使用したマルチユーザー
システムが実行レベル 3 になると実行される処理, 28
確認 (方法), 26
対応する SMF マイルストーン, 25
定義, 24
自動システム再構成ユニット, 「ASRU」を参照
修復
FMA の障害または欠陥, 66-67
破壊されたりポジトリ, 53-55
障害 (FMA)
修復, 66-67
情報の表示, 62-65
通知, 61
障害管理アーキテクチャー, 「FMA」を参照
障害管理リソース識別子, 「FMRI」を参照
新機能, SMF, 9-10
シングルユーザー実行レベル, 説明, 24

す

スナップショット (SMF), 説明, 18

せ

設定, 通知パラメータ, 33-34

た

タスク, SMF, 29-57

つ

追加

SMF 情報, 22-23

実行制御スクリプト (方法), 50

通知パラメータ

設定, 33-34

表示, 31

て

適用, プロファイル (SMF), 41

デバッグ, 「トラブルシューティング」を参照

デフォルトの実行レベル, 定義, 24

電源切断実行レベル, 説明, 24

と

トラブルシューティング

SMF サービス, 52-57

無効になったサービス, 53

は

破壊されたりポジトリ, 修復, 53-55

ひ

非表示, SMF, 23

表示

「表示」を参照

FMA の情報, 62-65

SMF 情報, 22

カスタマイズ (SMF), 31

サービスステータス (SMF), 30-31

ふ

ブート

エラーログ (SMF), 23-24

サービスを起動させない, 56

詳細モード, 56-57

マイルストーン (SMF), 23-24

復元, サービス (SMF), 38

プロパティ (SMF)

説明, 21

単一の変更, 42

複数の変更, 42-43

プロパティグループ (SMF), 説明, 21

プロファイル (SMF)

作成, 39

説明, 15

適用, 41

へ

変換

inetd サービスから SMF への, 47-48

実行制御スクリプト, SMF サービスへの, 51-52

変更

「変更」を参照

inetd サービスのプロパティ, 45-46

SMF 環境変数, 44

SMF 情報, 22-23

コマンド行引数, 47

サービス構成ファイル, 43-44

単一プロパティ (SMF), 42

複数プロパティ (SMF), 42-43

ま

マイルストーン (SMF)

説明, 12

対応する実行レベル, 25

ブート, 23-24

マニフェスト (SMF)

作成, 35-36

説明, 14-15

マルチユーザー実行レベル, 説明, 25

む

無効化, サービス (SMF), 36-37
無効にする, 実行制御スクリプト (方法), 50
無効になったサービス, トラブルシューティング, 53

も

モニタリング, サービス (SMF), 29-34

ゆ

有効化, サービス (SMF), 37

ら

ライブラリインタフェース, SMF, 20

り

リスタータ (SMF), 21
説明, 10
リスタータデーモン (SMF), 説明, 20
リブート実行レベル, 説明, 25
リポジトリ (SMF)
説明, 10, 16
バックアップ, 18

れ

レイヤー (SMF), 説明, 16-17