

# Oracle® Solaris ゾーンの作成と使用

ORACLE®

Part No: E75320-01  
2018 年 8 月



**Part No: E75320-01**

Copyright © 2004, 2018, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

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

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

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

**ドキュメントのアクセシビリティについて**

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

**Oracle Supportへのアクセス**

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



# 目次

---

このドキュメントの使用法 .....	15
<b>1 非大域ゾーン構成の計画 .....</b>	<b>17</b>
現在のシステム設定の評価 .....	17
ゾーンのディスク容量要件 .....	17
ゾーンサイズの制限 .....	18
ゾーンとサービス管理機能 .....	18
ゾーンの SMF 依存関係 .....	19
ゾーンの SMF インスタンスのサービス状態について .....	20
ゾーンの SMF インスタンスのサービス状態 .....	20
SMF 目標サービスとゾーン .....	21
ゾーンのブート順序 .....	22
ゾーンの委任リスタータ .....	23
ゾーンのホスト名およびネットワーク要件の決定 .....	24
ゾーンのホスト名 .....	24
排他的 IP ゾーンのネットワークアドレス .....	25
共有 IP ゾーンのネットワークアドレス .....	25
ゾーンのファイルシステム構成 .....	26
ネイティブゾーンのシステム構成 .....	27
システム構成プロファイルとゾーン .....	28
対話式システム構成 (SCI) ツール .....	29
自動インストーラ (AI) のゾーン構成プロファイルの例 .....	30
非大域ゾーンのライブゾーン再構成の使用 .....	34
zonecfg ユーティリティー編集モード .....	34
非大域ゾーンのライブゾーン再構成について .....	35
ライブゾーン構成の一時的な変更 .....	36
ライブゾーン構成の永続的な変更 .....	36
ライブゾーン再構成の予行演習 .....	36
ライブゾーン構成の再ロード .....	37

<b>2 非大域ゾーンのインストール、停止処理、停止、アンインストール、クローニングについて</b> .....	39
ゾーンのインストールと管理の概要 .....	39
ゾーンがインストールされている Oracle Solaris 11.4 システムでの自動インストールおよびパッケージ .....	40
ゾーンのパッケージングの概要 .....	40
パッケージとゾーンについて .....	41
ゾーンがインストールされているシステムへのパッケージの追加について .....	42
ゾーン内でのパッケージの削除について .....	44
パッケージ情報の照会 .....	44
インストール済みのゾーンがあるシステムでのプロキシ構成 .....	44
ゾーンの並列更新 .....	46
ゾーンの状態がパッケージの操作に与える影響 .....	47
統合アーカイブからのゾーンの配備について .....	48
ゾーンのインストール方法 .....	48
作成後のゾーンの内容 .....	50
インストール後のゾーンの情報 .....	51
zoneadmd ゾーン管理デーモン .....	52
zsched ゾーンスケジューラプロセス .....	52
ゾーンアプリケーション環境 .....	52
ゾーンの停止処理、停止、リブート、およびアンインストールについて .....	53
ゾーンのシャットダウンについて .....	53
ゾーンの停止について .....	53
ゾーンのリブートについて .....	54
ゾーンのアンインストール .....	55
非大域ゾーンのクローニングについて .....	56
ゾーンのクローニングの再構成要件 .....	56
ゴールデンゾーンについて .....	56
ゾーンのクローニング方法 .....	57
ブート環境とゾーン .....	57
非大域ゾーンでの beadm コマンド .....	58
大域ゾーンと非大域ゾーンのブート環境 .....	58
共有ゾーンの状態 .....	59
代替ルートの指定 .....	60
孤立したゾーンブート環境とクローンについて .....	60
<b>3 非大域ゾーンへのログインについて</b> .....	63
zlogin コマンドについて .....	63

非大域ゾーンへのログイン方法 .....	64
ゾーンコンソールログイン .....	64
ゾーンへのユーザーログイン .....	65
ゾーンへのフェイルセーフログイン .....	65
ゾーンへのリモートログイン .....	65
対話型ゾーンログイン .....	66
非対話型ゾーンログイン .....	66
<b>4 Oracle Solaris ゾーンの管理について .....</b>	<b>67</b>
非大域ゾーンが含まれているシステムのセキュリティー対策 .....	67
ゾーン管理者向けの権利プロファイルの概要 .....	68
非大域ゾーン内の特権 .....	71
不変ゾーンについて .....	74
ゾーン内での IPsec アーキテクチャーの使用 .....	75
ゾーン内での Oracle Solaris 監査の使用 .....	75
非大域ゾーンが含まれているシステムのモニタリング .....	76
大域ゾーンの可視性とアクセス .....	76
ゾーン内でのプロセス ID の可視性 .....	77
ゾーンのシステム可観測性 .....	77
zonestat ユーティリティーを使用したアクティブなゾーンの統計情報の報告 .....	78
fsstat ユーティリティーを使用した非大域ゾーンのモニタリング .....	79
非大域ゾーン内での DTrace の実行 .....	79
ゾーン内のコアファイル .....	79
非大域ゾーンのノード名 .....	80
非大域ゾーンでの時間値の設定 .....	80
ファイルシステムと非大域ゾーン .....	81
ゾーンでの NFS サーバーの実行 .....	81
-o nosuid マウントオプション .....	82
ゾーン内でのファイルシステムのマウント .....	82
ゾーン内でのファイルシステムのアンマウント .....	83
NFS クライアントとして機能する非大域ゾーン .....	84
ファイルシステムのトラバース .....	84
セキュリティーの制限およびファイルシステムの動作 .....	84
ゾーン内で mknod を使用する際の制限 .....	87
大域ゾーンから非大域ゾーンにアクセスする際の制限 .....	87
排他的 IP 非大域ゾーンにおけるネットワーク .....	88
排他的 IP ゾーンにおけるネットワークの概要 .....	88

排他的 IP ゾーンの区分化 .....	89
排他的 IP データリンクインタフェース .....	89
同一システム上の排他的 IP ゾーン間の IP トラフィック .....	89
排他的 IP ゾーンにおけるパケットフィルタファイアウォール .....	90
排他的 IP ゾーン内の IP ネットワークマルチパス .....	90
共有 IP 非大域ゾーンにおけるネットワーク .....	90
共有 IP ゾーンの区分化 .....	90
共有 IP ネットワークインタフェース .....	91
同一システム上の共有 IP ゾーン間の IP トラフィック .....	91
共有 IP ゾーンでの Oracle Solaris パケットフィルタ .....	92
共有 IP ゾーン内の IP ネットワークマルチパス .....	92
非大域ゾーンでのデバイスの使用 .....	93
/dev および /devices 名前空間 .....	93
排他使用のデバイス .....	94
デバイスドライバの管理 .....	94
非大域ゾーンで動作しないか、変更されるユーティリティー .....	94
非大域ゾーンでのアプリケーションの実行 .....	95
非大域ゾーンでのリソース管理 .....	96
非大域ゾーンで使用されるリソース制御 .....	96
ゾーンがインストールされているシステムでの公平配分スケジューラ .....	97
ゾーンがインストールされているシステムでの拡張アカウントینگ .....	97
非大域ゾーンの移行および変換 .....	98
solaris ゾーンのコールド移行について .....	98
ゾーンの変換について .....	103
zonep2vchk ツールを使用した大域から非大域への変換準備 .....	105
非大域ゾーンが含まれているシステムのバックアップ方法 .....	109
非大域ゾーンが含まれているシステムのバックアップ手法 .....	109
非大域ゾーン内でバックアップするデータの決定 .....	110
非大域ゾーンの復元について .....	112
非大域ゾーンが含まれているシステムで使用されるコマンド .....	112
ゾーンの管理およびモニタリング用のコマンド .....	112
リソース上限制御用のコマンド .....	113
ゾーン用に変更されたコマンド .....	113
<b>5 非大域ゾーン構成の作成と変更 .....</b>	<b>117</b>
非大域ゾーン構成のタスクマップ .....	117



ゾーン構成 .....	118
▼ 構成プロファイルを使用してゾーンを構成する方法 .....	118
▼ zonecfg ユーティリティを使用してゾーンを構成する方法 .....	119
▼ 非大域ゾーンの構成を表示する方法 .....	125
ゾーン構成の変更 .....	127
▼ 例: ゾーン構成内のリソースを変更する方法 .....	127
▼ ゾーン構成内のプロパティをクリアする方法 .....	128
▼ ゾーンの名前を変更する方法 (zoneadm) .....	129
▼ ゾーンの名前を変更する方法 (zonecfg) .....	130
▼ 専用のデバイスをゾーンに追加する方法 .....	131
▼ 大域ゾーンの CPU シェアを設定する方法 .....	131
▼ ゾーンの特権を変更する方法 .....	132
既存の共有 IP ゾーンを排他的 IP ゾーンに変更する .....	133
ゾーン構成を元に戻す、または削除する .....	133
▼ ゾーン構成の変更を元に戻す方法 .....	133
▼ アンインストールされたゾーン構成を削除する方法 .....	134
<b>6 非大域ゾーンのインストール、ブート、クローニング、移動、およびアンインストール .....</b>	<b>137</b>
ゾーンのインストールとブート .....	137
▼ 構成済みのゾーンをインストールする方法 .....	138
▼ インストールされた非大域ゾーンの UUID を取得する方法 .....	141
▼ installed 状態の非大域ゾーンに unavailable または incomplete のマークを付ける方法 .....	142
▼ installed 状態のゾーンを ready 状態に移行する方法 .....	144
▼ ゾーンのブート方法 .....	144
▼ ゾーンをシングルユーザーモードでブートする方法 .....	145
同一システム上でのゾーンのクローニングと移動 .....	146
▼ 同一システム上で非大域ゾーンをクローンする方法 .....	146
ゴールデンゾーンの作成とクローニング .....	148
▼ 共有ストレージ上にないゾーンを移動する方法 .....	150
非大域ゾーンのアンインストールと削除 .....	151
▼ ゾーンをアンインストールする方法 .....	151
▼ システムから非大域ゾーンを削除する方法 .....	153
<b>7 非大域ゾーンへのログイン .....</b>	<b>155</b>
ゾーンへのログイン .....	155
▼ ゾーン構成のためにゾーンコンソールにログインする方法 .....	156

▼ ゾーンコンソールへのログイン方法 .....	157
▼ 対話型モードを使用してゾーンにアクセスする方法 .....	157
▼ 非対話型モードを使用してゾーンにアクセスする方法 .....	158
▼ フェイルセーフモードを使用してゾーンに入る方法 .....	158
現在のゾーンの名前の表示 .....	159
非大域ゾーンの終了 .....	159
<b>8 非大域ゾーンのライブゾーン再構成 .....</b>	<b>161</b>
ライブゾーン再構成の実行 .....	161
▼ 実行中のゾーンのライブ構成を検査する方法 .....	161
▼ ライブゾーン構成の影響をプレビューする方法 .....	162
▼ ライブゾーン構成の永続的な構成変更を行う方法 .....	162
▼ 実行中のゾーンに一時的な変更を加える方法 .....	163
▼ ライブゾーン構成の変更のコミット中の障害から回復する方法 .....	164
データセットのライブゾーン再構成 .....	165
▼ ZFS データセットを実行中のゾーンに追加する方法 .....	165
▼ 委任されたデータセットを実行中のゾーンに追加する方法 .....	167
▼ 実行中のゾーンから dataset リソースを削除する方法 .....	168
▼ solaris ゾーンからデータセットをエクスポートする方法 .....	169
<b>9 Oracle Solaris ゾーンの移行 .....</b>	<b>171</b>
非大域ゾーンの移行を実行するための root 以外のユーザーの承認 .....	171
▼ ユーザーが個々のゾーンのコールド移行を実行することを承認する 方法 .....	172
コールド移行を使用した solaris ゾーンの移行 .....	173
▼ solaris ゾーンをコールド移行する方法 .....	173
使用不可能なシステムからのゾーンの移行 .....	175
共有ストレージを使用していない非大域ゾーンのアーカイブと移動 .....	176
▼ 統合アーカイブを使用して非大域ゾーンを新しいシステムに移動す る方法 .....	176
<b>10 システムの Oracle Solaris ゾーン への変換 .....</b>	<b>179</b>
Oracle Solaris 大域ゾーンの solaris 非大域ゾーンへの変換 .....	179
▼ zonep2vchk によるソースシステムのチェック方法 .....	180
▼ システムイメージのアーカイブを作成する方法 .....	180
▼ ターゲットシステムで変換済みゾーンを構成する方法 .....	182
▼ ターゲットシステムにアーカイブから変換済みゾーンをインストー ルする方法 .....	183

<b>11 Oracle Solaris ゾーン管理</b> .....	185
ゾーン管理者への制限された権利の割り当て .....	185
▼ ゾーン管理者の権利を割り当ててゾーンへのアクセスを制限する方 法 .....	186
非大域ゾーンへの DTrace 特権の追加 .....	186
▼ 非大域ゾーンに DTrace 特権を追加する方法 .....	187
ゾーンのシャットダウン、停止、およびリブート .....	187
▼ ゾーンをシャットダウンする方法 .....	187
▼ ゾーンを停止およびリブートする方法 .....	188
ゾーンの委任リスタータの管理 .....	189
実行中の非大域ゾーンでのファイルシステムのマウント .....	190
▼ LOFS を使用してファイルシステムをマウントする方法 .....	191
▼ ZFS データセットを非大域ゾーンに委任する方法 .....	191
大域ゾーン内で非大域ゾーンでの特定のファイルシステムへのアクセスを追 加する .....	193
▼ 非大域ゾーンの CD または DVD メディアへのアクセスを追加する 方法 .....	193
非大域ゾーンの SMF サービスのステータスの報告 .....	194
非大域ゾーンのリソースの使用状況の報告 .....	195
ゾーンごとのファイルシステム統計情報の報告 .....	198
非大域ゾーンでの IP ネットワークマルチパスの使用 .....	199
▼ IP ネットワークマルチパス機能を共有 IP 非大域ゾーンに拡張する 方法 .....	199
排他的 IP 非大域ゾーンでのデータリンクの管理 .....	200
ゾーンがインストールされているシステムでの公平配分スケジューラの使 用 .....	201
▼ 大域ゾーンで FSS 配分を一時的に設定する方法 .....	202
ゾーンのバックアップと再作成 .....	202
<b>12 不変ゾーンの構成と管理</b> .....	205
不変ゾーンの構成 .....	205
MWAC セキュリティポリシーの設定 .....	205
MWAC セキュリティポリシーのゾーンリソースの例外 .....	207
不変ゾーン内の SMF サービス .....	207
不変非大域ゾーンの管理 .....	209
非大域ゾーンが不変かどうかの判断 .....	209
非大域不変ゾーンの管理の方法 .....	210
トラステッドパドメインを使用した不変ゾーンの管理 .....	210
不変ゾーンを書き込み可能にして管理する .....	213

不変大域ゾーン .....	214
不変大域ゾーンの構成 .....	214
不変大域ゾーンの保守 .....	214
<b>13 その他の Oracle Solaris ゾーン の問題のトラブルシューティング .....</b>	<b>217</b>
許可されているパッケージの不一致ためインストールが失敗する .....	217
排他的 IP ゾーンがデバイスを使用しているために <code>dladm reset-linkprop</code> が失敗する .....	218
ゾーン構成内に不正な特権セットが指定されている .....	218
ゾーンが停止しない .....	218
<b>14 共有ストレージ上の Oracle Solaris ゾーン .....</b>	<b>221</b>
ストレージ URI を使用した共有ストレージリソースについて .....	222
ローカルデバイス URI .....	222
論理ユニット URI .....	222
iSCSI URI .....	223
ストレージ URI と共有ストレージリソースの管理 .....	224
Oracle Solaris ゾーン への共有ストレージリソースの割り当て .....	226
ゾーンの <code>storage</code> プロパティ .....	226
<code>rootzpool</code> リソース .....	227
<code>zpool</code> リソース .....	227
ゾーンの名前変更 .....	228
ゾーン構成に関する制限事項 .....	228
共有ストレージリソースでの Oracle Solaris ゾーン の自動 ZFS ストレージ プール管理 .....	228
<code>unavailable</code> ゾーン状態について .....	230
追加の <code>zoneadm</code> サブコマンドオプション .....	230
ゾーンのインストール、クローニング、接続用のオプション .....	230
ゾーンのアンインストール用オプション .....	231
<code>rootzpool</code> リソースの移動の制限 .....	233
共有ストレージリソースでホストされる Oracle Solaris ゾーン の実装 .....	233
共有ストレージ上のゾーンに関する <code>zpool</code> 構成の考慮事項 .....	233
共有ストレージ上のゾーンのサンプルシナリオ .....	234
共有ストレージリソースでホストされる Oracle Solaris ゾーン の移行 .....	244
共有ストレージゾーン構成に対する既存のゾーンの移動 .....	245
▼ 共有ストレージ構成にゾーンを移動する方法 .....	246
▼ 共有ストレージ構成からゾーンを移動する方法 .....	247
▼ 別の共有ストレージにゾーンを移動する方法 .....	248

---

▼ インストール済みゾーンに ZFS ストレージプールを追加または移動 する方法 .....	249
共有ストレージリソースについてのゾンドキュメントリファレンス .....	250
用語集 .....	253
索引 .....	257



## このドキュメントの使用方法

---

- **概要** – Oracle Solaris ゾーンの機能と関連リソース管理機能を設定および使用する方  
法について説明します
- **対象読者** – 技術者、システム管理者、および認定サービスプロバイダ
- **前提知識** – Oracle Solaris 環境の管理経験。仮想化環境の経験も推奨されます。

## 製品ドキュメントライブラリ

この製品および関連製品のドキュメントとリソースは <http://www.oracle.com/pls/topic/lookup?ctx=E75431-01> で入手可能です。

## フィードバック

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





# 1

## 非大域ゾーン構成の計画

---

この章では、システムにゾーンを構成する前に実行する必要がある操作について説明します。内容は次のとおりです。

- 17 ページの「現在のシステム設定の評価」
- 18 ページの「ゾーンとサービス管理機能」
- 24 ページの「ゾーンのホスト名およびネットワーク要件の決定」
- 26 ページの「ゾーンのファイルシステム構成」
- 27 ページの「ネイティブゾーンのシステム構成」
- 34 ページの「非大域ゾーンのライブゾーン再構成の使用」

ゾーン構成プロセスの概要については、『Oracle Solaris Zones Configuration Resources』の第 1 章、「Non-Global Zone Configuration Command and Resources」を参照してください。

カーネルゾーン内の非大域ゾーンの管理については、『Oracle Solaris カーネルゾーンの作成と使用』の「カーネルゾーン内の非大域ゾーンの管理」を参照してください。

### 現在のシステム設定の評価

次に、ゾーンの使用に関連したシステムの主な考慮事項を示します。

- 各ゾーン内部で稼働するアプリケーションの性能要件。
- 各ゾーン内部で固有のファイルを保持するディスク容量がどれだけ利用可能か。

### ゾーンのディスク容量要件

ゾーンが消費可能なディスク容量に関する制限はありません。大域管理者または適切な承認を持つゾーン管理者が、容量の制限を担当します。大域管理者は、非大域ゾーンのルートファイルシステムを保持するのに十分なローカルストレージまたは共有ス

トレージがあることを確認する必要があります。小規模な単一プロセッサシステムでも、同時に稼働する多数のゾーンをサポートできます。

非大域ゾーンにインストールされるパッケージの特性は、ゾーンの容量要件に影響を与えます。また、パッケージの数も1つの要因です。

ディスク要件は、大域ゾーンに現在インストールされているパッケージとインストール済みのソフトウェアによって使用されるディスク容量によって決定されます。

1つのゾーンでは、ゾーンあたりに最低 150M バイトの空きディスク容量を必要とします。ただし、大域ゾーンが標準のすべての Oracle Solaris パッケージでインストールされている場合、必要な空きディスク容量は通常 500M バイトから 1G バイトです。さらに多くのソフトウェアが追加される場合、この数値は増える可能性があります。

ゾーンごとに 40M バイトの追加 RAM が推奨されていますが、十分なスワップ空間のあるシステムでは、これは必須ではありません。

## ゾーンサイズの制限

ZFS データセットによって支援される zonepath を持つゾーンでは、ZFS データセットの割り当て制限を使用してゾーンサイズを制限できます。zonepath データセットにアクセスできる管理者は、データセットの quota および reservation プロパティを変更して、各ゾーンが消費できる最大のディスク容量を制御できます。これらのプロパティについては、[zfs\(8\)](#) のマニュアルページで説明されています。

また、管理者は、ZFS ボリュームを固定サイズで作成して、そのボリュームのデータセット内にゾーンをインストールすることもできます。このボリュームは、その内部にインストールされたゾーンのサイズを制限します。

## ゾーンとサービス管理機能

Oracle Solaris サービス管理機能 (SMF) フレームワークは、システムサービスおよびアプリケーションサービスを管理します。SMF は、システムの操作に不可欠な重要なシステムサービスを管理し、データベースや Web サーバーなどのアプリケーションサービスも管理します。SMF は、ハードウェアまたはソフトウェアの障害が発生した場合でも、不可欠なシステムサービスおよびアプリケーションサービスが継続して実行できるように確保して、システムの可用性を高めます。

ゾーンの SMF サービスには、ゾーンのブート順序の優先順位付け、並列でブートするゾーンの数のしきい値設定、および障害管理アーキテクチャー (FMA) との統合の提供を行うメカニズムがあります。

## ゾーンの SMF 依存関係

ゾーンの依存関係を構築する一連の Oracle Solaris サービス管理機能 (SMF) 障害管理リソース識別子 (FMRI) は、`smf-dependency` リソースで定義されます。依存関係は、`name`、`fmri`、および `grouping` プロパティで構成されます。`svc:/system/zones:default` サービスの依存関係は次のとおりです。

`require_all` 依存関係

```
svc:/system/filesystem/local:default
svc:/milestone/name-services:default
svc:/milestone/devices:default
svc:/milestone/network:default
```

`optional_all` 依存関係

```
svc:/system/pools:default
svc:/system/zones-monitoring:default
```

ゾーンの依存関係を構築する一連の SMF FMRI は、`zonecfg` コマンドの `smf-dependency` リソースで定義されます。

依存関係の設定の詳細については、『[Oracle Solaris Zones Configuration Resources](#)』の第 1 章、「[Non-Global Zone Configuration Command and Resources](#)」を参照してください。

### 例 1 ゾーンの SMF 依存関係の設定

依存関係の説明:

- `require_all: svc:/application/list:default`
- `require_any: svc:/system/zones/zone:appfirewall`
- `require_any: svc:/3rdparty/my-firewall:default`
- `exclude_all: svc:/system/zones/zone:dataload`

```
global$ pfbash zonecfg -z my-zone
zonecfg:my-zone> add smf-dependency
zonecfg:my-zone:smf-dependency> set fmri=svc:/application/list:default
zonecfg:my-zone:smf-dependency> end
zonecfg:my-zone> add smf-dependency
zonecfg:my-zone:smf-dependency> set name=firewall
zonecfg:my-zone:smf-dependency> set fmri=svc:/system/zones/zone:appfirewall
zonecfg:my-zone:smf-dependency> set grouping=require_any
zonecfg:my-zone:smf-dependency> end
zonecfg:my-zone> add smf-dependency
zonecfg:my-zone:smf-dependency> set name=firewall
zonecfg:my-zone:smf-dependency> set fmri=svc:/3rdparty/my-firewall:default
zonecfg:my-zone:smf-dependency> end
zonecfg:my-zone> add smf-dependency
```

```
zonecfg:my-zone:smf-dependency> set fmri=svc:/system/zones/zone:dataload
zonecfg:my-zone:smf-dependency> set grouping=exclude_all
zonecfg:my-zone:smf-dependency> end
zonecfg:my-zone> exit
```

smf-dependency リソースの詳細は、『Oracle Solaris ゾーン構成リソース』および [zonecfg\(8\)](#) のマニュアルページを参照してください。

---

**注記** - 特定のゾーンのインフラストラクチャーの設定中にエラーが発生すると、そのゾーンは maintenance 状態になります。ゾーンに対して失敗した zoneadm boot または zoneadm attach 操作を、ゾーンの委任リスタータに再試行させるには、そのゾーンインスタンスに対して svcadm clear コマンドを使用してください。

---

## ゾーンの SMF インスタンスのサービス状態について

ゾーンの SMF インスタンスのサービス状態は、ゾーンの milestone/goals 状態によって決定されます。milestone/goals 目標サービスは、システムが起動して実行されていると判断できる、明確な地点を提供します。

ゾーンリスタータには、milestone/goals サービスをサポートしている各非大域ゾーンについて、そのサービスの状態が通知されます。各非大域ゾーンのゾーン SMF インスタンスの状態は、ゾーンの milestone/goals の状態に一致します。

ゾーンの委任リスタータに、milestone/goals を無視して zoneadm boot コマンドの成功に基づいてゾーン SMF インスタンスをオンラインに移行させるように指示できません。ブート操作が失敗した場合、ゾーン SMF インスタンスは保守状態になります。このスイッチは、ゾーンの委任リスタータサービスの下で、次の SMF プロパティグループによって制御されます。

```
PG name=config type=application
   config/track-zone-goals = true | false
```

## ゾーンの SMF インスタンスのサービス状態

svc.zones コマンドでは、サービス管理業務の一環として、管理対象ゾーンの SMF インスタンスごとに状態マシンを実装します。このマシンの状態は、[smf\(7\)](#) のマニュアルページで説明されている状態で構成されます。これらの状態の説明を次に示します。

### degraded

ゾーンの SMF インスタンスは online 状態に到達しましたが、zoneadmd は複数回失敗しました。デーモンは実行されていません。

**disabled**

ゾーンは停止されており、ゾーンリスタータによってブートできません。

**maintenance**

**maintenance** 状態の原因を調べるには、`svcs -vx` コマンドの出力を使用してください。**maintenance** 状態にあるゾーンの SMF インスタンスでは、無効化イベントで **maintenance** 状態はクリアされません。次の理由で、ゾーンの SMF インスタンスは **maintenance** 状態になっている可能性があります。

- ブートの試行に失敗しました。
- ゾーンがブート不可の状態です
- 管理アクションなしでゾーンの **milestone/goals** 依存関係を満たすことができません。

**offline**

ゾーンの SMF インスタンスは有効ですが、その SMF 依存関係は満たされていません。

**online**

ゾーンが実行中であり、**milestone/goals** に達しました。**milestone/goals** をサポートしていないゾーンでは、ゾーンがブートし始めるとすぐに **online** 状態に達します。

**milestone/goals** は、システムが起動して実行されていると判断できる、明確な地点を提供します。**milestone/goals** の依存関係は、システムにとってミッションクリティカルなサービスに合致するように構成されるべきです。**milestone/goals** のデフォルトの依存関係は次のとおりです。

```
svc:/milestone/multi-user-server:default
```

**milestone/goals** の詳細は、『[Oracle Solaris 12 でのシステムサービスの開発](#)』の「[Creating a Goal Service](#)」を参照してください。

**uninitialized**

対応するゾーンはまだ `svc.zones` で処理されていません。

## SMF 目標サービスとゾーン

目標サービスは、ゾーンが想定どおりに機能するために実行する必要のある、一連のサービスを定義します。システム起動時にこれらのサービスのいずれかが実行されていない場合、目標サービスはユーザーに通知します。

**milestone/goals** 目標サービスは、システムが起動して実行されていると判断できる、はっきりと明確に定義された地点を提供します。**milestone/goals** の依存関係

は、システムにとってミッションクリティカルなサービスを表すように構成されるべきです。milestone/goals のデフォルトの依存関係は `svc:/milestone/multi-user-server:default` です。

システム管理者は、目標サービスを使用することで、システムが起動し、その本来の目的で機能しているときに実行されているべき一連の期待されるサービスを定義できます。目標サービスは、`goals svcadm` サブコマンドを使用して管理されます。

目標サービスは、そのすべての依存関係を介入なしで満たすことを期待されています。依存関係が満たされるために管理者の介入が必要な場合、目標サービスは `maintenance` 状態に置かれます。maintenance 状態にある目標サービスは、その依存関係が満たされると、その状態から自動的に抜けます。

通常、実際の作業を行うサービスには `goal service` 設定を使用するべきではありません。

`general/goal-service=` プロパティを `true` に設定すると、サービスは目標サービスとして動作するように設定されます。

目標サービスは動的に有効にされるサービスに依存するべきではありません。動的に有効にされるサービスがあると、依存関係が別のサービスで有効にされるまで、目標サービスは `maintenance` 状態になります。目標サービスの依存関係を設定するには、`svcadm` サブコマンドの目標を使用できます。

## ゾーンのブート順序

ゾーンのブート順序は 2 ステップで設定します。

1. ブート順序は、ゾーンの SMF インスタンスの SMF 依存関係によって部分的に決定されます。

たとえば、ZoneA が ZoneB の前にブートするようにするには、ZoneA を ZoneB に依存するものとして追加します。[zonectfg\(8\)](#) のマニュアルページの `smf-dependency` プロパティを参照してください。

2. 各ゾーンのブート優先順位を割り当てることで、ブート順序をさらに制御します。ゾーンの SMF 依存関係が満たされると、ゾーンは構成されている `boot-priority` リソースプロパティに従ってブートします。受け入れられる値は、`high`、`normal`、および `low` です。デフォルト値は `normal` です。

『[Oracle Solaris Zones Configuration Resources](#)』の「[boot-priority Global Property](#)」および『[Oracle Solaris Zones Configuration Resources](#)』の「[smf-dependency Resource Type](#)」を参照してください。

## ゾーンの委任リスタータ

非大域ゾーンは、ゾーンの委任リスタータサービスによって管理されます。ゾーンの委任リスタータは、ゾーンサービスの障害管理リソース識別子 (FMRI) `svc:/system/zones:default` の下で実行されます。

このセクションでは、ゾーンの委任リスタータに関する次の情報を提供します。

- [23 ページの「ゾーンの委任リスタータの動作のしくみ」](#)
- [24 ページの「ゾーンの委任リスタータのプロパティ」](#)

### ゾーンの委任リスタータの動作のしくみ

各 SMF サービスインスタンスは、リスタータによって管理されます。リスタータは、インスタンス構成を取得し、実行環境を提供します。すべてのリスタータに共通する情報については、[smf\\_restarter\(7\)](#) のマニュアルページを参照してください。

`install`、`attach`、または `clone` ゾーン操作によって、ゾーン SMF インスタンスが FMRI `svc:/system/zones/zone:zonename` で作成されます。`uninstall` および `detach` ゾーン操作によって、ゾーン SMF インスタンスがシステムから削除されます。

起動時に、ゾーンの委任リスタータは、SMF インスタンスを必要とする非大域ゾーンのゾーン SMF インスタンスを作成します。ゾーンにゾーン SMF インスタンスがすでに存在するが、ゾーンの状態がインスタンス `configured` より低い場合、リスタータは既存の SMF インスタンスを削除します。

ゾーンの委任リスタータは、各ゾーンに必要なインフラストラクチャーを設定します。リスタータは起動時に次のステップを実行します。

- 各ゾーンの `zoneadmd` デーモンを開始し、必要に応じて再起動します。`configured` より高い状態にある各ゾーンでは、`zoneadmd` が実行中になります。
- `installed` 状態にあるすべてのゾーンに対して `zoneadm boot` コマンドを実行し、`unavailable` 状態にあるすべてのゾーンに対して `zoneadm attach` コマンドを実行します。詳細は、[zoneadm\(8\)](#) のマニュアルページを参照してください。

ゾーンの委任リスタータは、ゾーンサービスメソッドのスクリプトを提供します。ゾーンの委任リスタータは、既存のゾーンサービス FMRI の下で実行されます。

## ゾーンの委任リスタートのプロパティ

ゾーンの委任リスタートは、ゾーンで同時に実行されるブートまたはシャットダウン、および一時停止または再開の数を制限できます。制限を設定するには、`svccfg` および `svcadm` コマンドで、ゾーンの委任リスタートの次に示すプロパティを設定します。

`config/concurrent-boot-shutdown`

ゾーンで同時に実行されるブートまたはシャットダウンの最大数。

`config/concurrent-suspend-resume`

ゾーンで同時に実行される中断操作または再開操作の最大数。

デフォルトでは、ゾーンの委任リスタートはすべてのゾーンのタスクを並列で実行します。これらのプロパティが存在しないか、値が指定されていないか、値が 0 に設定されている場合、ゾーンリスタートが同時に実行するゾーン操作の数に制限はありません。

ゾーンの SMF インスタンスが `online` 状態または `maintenance` 状態に達すると、非大域ゾーンのブートプロセスは完了とみなされます。ゾーンの SMF 状態の詳細は、[20 ページの「ゾーンの SMF インスタンスのサービス状態について」](#) を参照してください。

ゾーンの委任リスタートの詳細は、[`svc.zones\(8\)` のマニュアルページ](#) を参照してください。

## ゾーンのホスト名およびネットワーク要件の決定

ゾーンのホスト名を決定する必要があります。排他的 IP ゾーンの内部で、大域ゾーンに対して行なったようにアドレスを構成します。

### ゾーンのホスト名

NIS または DNS ネームサービス、あるいは LDAP ディレクトリサービスを使用している場合は、サーバー上に存在するデータベース (`hosts.byname` など) にホスト情報が格納されます。

ネームサービス用にローカルファイルを使用する場合は、`/etc/inet/hosts` ファイル内で `hosts` データベースが保持されます。ゾーンネットワークインタフェースのホスト名の解決は、`/etc/inet/hosts` 内のローカル `hosts` データベースで行われます。



あるいは、共有 IP ゾーンの場合、ゾーンの構成時に IP アドレス自体を直接指定することで、ホスト名の解決を不要にできます。詳細は、[hosts\(5\)](#) および [nodename\(5\)](#) のマニュアルページを参照してください。『Oracle Solaris 12 でのネットワークコンポーネントの構成と管理』の第 3 章、「[Configuring and Administering IP Interfaces and Addresses in Oracle Solaris](#)」を参照してください。

## 排他的 IP ゾーンのネットワークアドレス

排他的 IP ゾーンの内部で、大域ゾーンと同様の方法でアドレスを構成します。DHCP および IPv6 ステートレスアドレスの自動構成を使用してアドレスを構成することもできます。

排他的 IP ゾーンの詳細は、『[Oracle Solaris Zones Configuration Resources](#)』の「[About Zone Network Interfaces](#)」を参照してください。IP アドレス構成については、『[Configuring and Managing Network Components in Oracle Solaris 11.4](#)』の第 3 章、「[Configuring and Administering IP Interfaces and Addresses in Oracle Solaris](#)」を参照してください。

## 共有 IP ゾーンのネットワークアドレス

ネットワーク接続を必要とする各共有 IP ゾーンには、1 つ以上の一意の IP アドレスがあります。IPv4 と IPv6 の両方のアドレスがサポートされます。

### IPv4 のゾーンネットワークアドレス

IPv4 を使用している場合、アドレスを取得してゾーンに割り当てます。アドレスをゾーンに割り当てる場合、CIDR 表記 (192.0.2.0/24 など) を使用してアドレスを指定できます。

共有 IP ゾーンの場合は、ゾーンの構成時に IP アドレス自体を直接指定することで、ホスト名の解決を不要にできます。

詳細は、[hosts\(5\)](#)、[netmasks\(5\)](#)、および [nodename\(5\)](#) を参照してください。

### IPv6 のゾーンネットワークアドレス

IPv6 を使用している場合、アドレスを手動で構成する必要があります。通常、次の 2 種類のアドレスを最小限構成する必要があります。

リンクローカルアドレス	<p>リンクローカルアドレスの書式は、<code>fe80::64</code> ビットインタフェース <code>ID/10</code> です。/10 は、接頭辞の長さが 10 ビットであることを示します。</p>
グローバルユニキャストアドレス	<p>大域ユニキャストアドレスは、管理者がサブネットごとに構成した 64 ビット接頭辞および 64 ビットのインタフェース ID に基づきます。接頭辞は、IPv6 を使用するよう構成されている同一サブネット上の任意のシステムで、<code>ipadm show-addr</code> コマンドを実行して取得できます。</p> <p>通常、64 ビットのインタフェース ID は、システムの MAC アドレスから取得されます。ゾーンで使用するために、次の規則を使用して、大域ゾーンの IPv4 アドレスから一意の代替アドレスを取得できます。</p> <p>16 ビットのゼロ:IPv4 アドレスの上位 16 ビット:IPv4 アドレスの下位 16 ビット:ゾーンで一意の番号</p> <p>大域ゾーンの IPv4 アドレスは 192.0.2.10 と想定しています。このアドレスは次のように 16 進数に変換されます。</p> <ul style="list-style-type: none"> <li>■ 192 = c0</li> <li>■ 0 = 0</li> <li>■ 2 = 2</li> <li>■ 10 = 0a</li> </ul> <p>したがって、ゾーン固有の番号 1 を使用した非大域ゾーンの適正なリンクローカルアドレスは <code>fe80::c00:20a:1/10</code> になります。</p> <p>そのサブネットで使用中の大域接頭辞が <code>2001:DB8:aabb:ccdd/32</code> である場合、同じ非大域ゾーン用の一意の大域ユニキャストアドレスは <code>2001:DB8:aabb:ccdd::c00:20a:1/32</code> です。IPv6 アドレスを構成する際、接頭辞の長さを指定する必要があることに注意してください。</p>

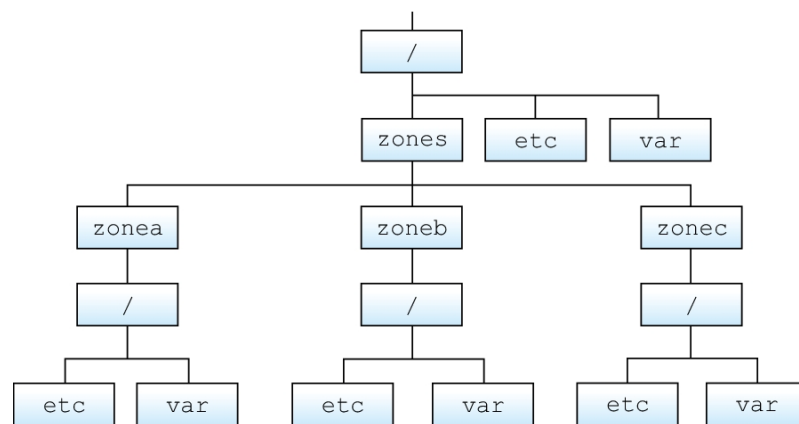
リンクローカルおよび大域ユニキャストアドレスの詳細は、[ipadm\(8\)](#) および [inet6\(4P\)](#) のマニュアルページを参照してください。

## ゾーンのファイルシステム構成

仮想プラットフォームを設定する際、実行するマウントを多数指定できます。ループバック仮想ファイルシステム (LOFS) を使用してファイルシステムをゾーンにループバックマウントする場合、`nodevices` オプションを指定してファイルシステムをマウントする必要があります。`nodevices` オプションの詳細は、[81 ページの「ファイルシステムと非大域ゾーン」](#)を参照してください。

LOFS を使用すると、代替パス名を使用してファイルにアクセスできるように、新しい仮想ファイルシステムを作成できます。非大域ゾーンでは、ループバックマウントにより、ファイルシステム階層がゾーンのルート下に複製されているように見えます。ゾーン内では、ゾーンのルートから始まるパス名を使ってすべてのファイルにアクセスできるようになります。LOFS マウントではファイルシステムの名前空間が保持されます。

図 1 ループバックマウントされたファイルシステム



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

## ネイティブゾーンのシステム構成

システム上のゾーンを一覧表示するとき、ネイティブゾーンは `solaris` ゾーンとして表示されます。詳しい説明については、[ネイティブゾーン](#) を参照してください。

**注記** - Oracle Solaris 11 でのネイティブゾーンの表示は、Oracle Solaris 10 の表示と多少異なります。

```
s10-system$ zoneadm list -cv
ID NAME          STATUS    PATH          BRAND    IP
0 global         running  /             native   shared
- zone_1        installed -             native   shared

s11-system$ zoneadm list -cv
ID NAME          STATUS    PATH          BRAND    IP
0 global         running  /             solaris  shared
- zone_1        installed -             solaris  shared
- kz_zone_2     installed -             solaris-kz excl
```

Oracle Solaris のインスタンスごとに、一定量の構成を次のいずれかの方法で実行する必要があります。

- システム構成プロファイルを使用して、インストールの一環として。概要については、[28 ページの「システム構成プロファイルとゾーン」](#)を参照してください。
- 対話式システム構成 (SCI) ツールを使用して、インストール後の初回ブート時に。概要については、[29 ページの「対話式システム構成 \(SCI\) ツール」](#)を参照してください。

システム構成は `sysconfig` ユーティリティーによって実行されます。 `sysconfig` ユーティリティーについては、『[Oracle Solaris 12 システムの自動インストール](#)』の第 9 章、『[Unconfiguring or Reconfiguring an Oracle Solaris Instance](#)』および `sysconfig(8)` のマニュアルページで説明されています。 `solaris` ゾーンのシステム構成の詳細は、『[Oracle Solaris ゾーン構成リソース](#)』および `solaris(7)` のマニュアルページを参照してください。

## システム構成プロファイルとゾーン

ゾーンを `configured` 状態から `installed` 状態にできる各操作では、システム構成プロファイルを使用してシステム構成を実行できます。システム構成プロファイルを受け入れる `zoneadm` 操作は次のとおりです。

```
zoneadm -z zone install -c {file.xml | directory}
zoneadm -z zone attach -c {file.xml | directory}
zoneadm -z zone clone -c {file.xml | directory} source-zone
```

`zoneadm` コマンドの `-c` オプションはブランド固有です。 `-c` オプションを使用して `solaris` ゾーンを構成する方法の詳細は、[zones\(7\)](#) のマニュアルページを参照してください。

-c *directory* 形式のオプションを使用する場合、指定するディレクトリにはファイルやサブディレクトリを含めることができます。この形式の -c オプションを使用する場合は、次のガイドラインと要件に従ってください。

- 各ファイルは、プロファイルを含む有効な SMF サービスバンドルである必要があります。サービスバンドルの要件の詳細は、[service\\_bundle\(5\)](#) のマニュアルページを参照してください。
- 各ファイルの名前は .xml 拡張子で終わる必要があります。
- ディレクトリの入れ子の深さは任意ですが、システムの制限に従います。
- -c オプションに指定されたディレクトリ内の特定のディレクトリは、SMF リポジトリ内の特別なレイヤーにプロファイルを配置します。このようなサブディレクトリの名前は enterprise、site、node、および sysconfig で、enterprise-profile、site-profile、node-profile、および sysconfig-profile SMF レイヤーに対応します。これらの特殊ディレクトリのいずれにも含まれていないプロファイルは、sysconfig-profile レイヤーで適用されます。SMF レイヤーの詳細は、『[Oracle Solaris 11.4 でのシステムサービスの管理](#)』の「[リポジトリレイヤー](#)」を参照してください。

インストール時または初回ブート時にゾーンの構成を必要としない場合が 1 つあります。この例外は、-r オプションで作成された統合アーカイブである回復用アーカイブからゾーンがインストールされた場合です。そのようなゾーンには、統合アーカイブの作成時に存在していたシステム構成が保持されます。ただし、回復用アーカイブのインストール時にシステム構成が指定された場合、システムは指定されたプロファイルで再構成されます。対話式の再構成を強制的に実行するには、`/usr/share/auto_install/sc_profiles/enable_sci.xml` プロファイルを指定します。回復用アーカイブの配備および例については、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』の「[How to Deploy a Zone From a Unified Archive](#)」を参照してください。

システム構成プロファイルは `sysconfig create-profile` コマンドを使用して生成できます。

- `solaris` ゾーンで実行された場合、`sysconfig create-profile` コマンドは、`solaris` ゾーンに関連する構成プロパティのサブセットを含むプロファイルを生成します。
- 大域ゾーンまたはカーネルゾーンで生成されたプロファイルは、`solaris` ゾーンでも使用できます。プロファイルに `solaris` ゾーンに関連しないプロパティが含まれている場合、それらは暗黙のうちに無視されます。

## 対話式システム構成 (SCI) ツール

対話式システム構成 (SCI) ツールを使用すると、新規にインストールされた Oracle Solaris インスタンスに構成パラメータを指定できます。

-c *profile.xml* オプションを指定しない `sysconfig configure` コマンドは、システムを構成解除します。次に、SCI ツールを表示して管理者に確認してから、`/etc/svc/profile/site/scit_profile.xml` にその構成を書き込みます。次に、このツールはこの情報を使用してシステムを構成します。

`sysconfig create-profile` コマンドは、管理者に確認してから、SMF プロファイルファイルを作成します。出力場所が指定されていない場合、デフォルトでは、SCI ツールはファイルに `/system/volatile/profile/scit_profile.xml` という名前を付けます。パラメータには、システムのホスト名、タイムゾーン、ユーザーと root のアカウント、ネームサービスが含まれます。

このツール内でナビゲートするには、次のようにします。

- 各画面の下部に一覧表示されたファンクションキーを使用して画面を移動し、ほかの操作を実行します。キーボードにファンクションキーがない場合や、キーが反応しない場合は、Esc キーを押します。画面下部の説明が変わり、ナビゲーションやほかの機能の各 Esc キーが表示されます。
- 選択を変更するか、または入力フィールド間で移動するには、上矢印/下矢印キーを使用します。

詳細は、『[Oracle Solaris 12 システムの自動インストール](#)』の第 9 章、「[Unconfiguring or Reconfiguring an Oracle Solaris Instance](#)」、および `sysconfig(8)` と `solaris(7)` のマニュアルページを参照してください。

## 自動インストール (AI) のゾーン構成プロファイルの例

デフォルトのゾーン用 AI マニフェストは `/usr/share/auto_install/manifest/zone_default.xml` です。デフォルトとは異なるが同じパッケージを含んでいるゾーン構成を自動的にインストールするには、ゾーンのシステム構成プロファイルのカスタマイズします。異なるパッケージを必要とするゾーンの場合は、[43 ページの「ゾーンのデフォルトの AI マニフェストを改訂する方法」](#)を参照してください。

このセクションでは、特定のゾーン構成について、自動インストール (AI) のシステム構成プロファイルの例を示します。

- [例2「自動インストール \(AI\) を使用する排他的 IP ゾーン」](#)
- [例3「LDAP や DNS を使用しない排他的 IP ゾーン」](#)
- [例4「DHCP を使用し DNS を使用しない排他的 IP ゾーン」](#)

---

ヒント - `pwhash` コマンドを使用して、次のサンプルプロファイルに必要なパスワードとハッシュを作成します。詳細は、[pwhash\(1\)](#) のマニュアルページを参照してください。

---

例 2 自動インストーラ (AI) を使用する排他的 IP ゾーン

```
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="sysconfig">
  <service version="1" type="service" name="system/config-user">
    <instance enabled="true" name="default">
      <property_group type="application" name="root_account">
        <propval type="astring" name="login" value="root"/>
        <propval type="astring" name="password" value="$5$KeNRy1zU
$1qzy9rIsNloUhfVJFIWmVewE75aB5/EBA77kY7EP6F0abcde"/>
        <propval type="astring" name="type" value="role"/>
      </property_group>
      <property_group type="application" name="user_account">
        <propval type="astring" name="login" value="admin1"/>
        <propval type="astring" name="password" value="$5$/g353K5q$V8Koe/XuAeR/
zpBvpLsgVIqPrvc.9z0hYFYoyoBkE37uvxyz"/>
        <propval type="astring" name="type" value="normal"/>
        <propval type="astring" name="description" value="admin1"/>
        <propval type="count" name="gid" value="10"/>
        <propval type="astring" name="shell" value="/usr/bin/bash"/>
        <propval type="astring" name="roles" value="root"/>
        <propval type="astring" name="profiles" value="System Administrator"/>
        <propval type="astring" name="sudoers" value="ALL=(ALL) ALL"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/timezone">
    <instance enabled="true" name="default">
      <property_group type="application" name="timezone">
        <propval type="astring" name="localtime" value="UTC"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/environment">
    <instance enabled="true" name="init">
      <property_group type="application" name="environment">
        <propval type="astring" name="LC_ALL" value="C"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/identity">
    <instance enabled="true" name="node">
      <property_group type="application" name="config">
        <propval type="astring" name="nodename" value="my-zone"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/keymap">
    <instance enabled="true" name="default">
      <property_group type="system" name="keymap">
        <propval type="astring" name="layout" value="US-English"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/console-login">
    <instance enabled="true" name="default">
      <property_group type="application" name="ttymon">
        <propval type="astring" name="terminal_type" value="vt100"/>
      </property_group>
    </instance>
  </service>
</service_bundle>
```

例 3 LDAP や DNS を使用しない排他的 IP ゾーン

```
<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="sysconfig">
  <service version="1" type="service" name="system/config-user">
    <instance enabled="true" name="default">
      <property_group type="application" name="root_account">
        <propval type="astring" name="login" value="root"/>
        <propval type="astring" name="password" value="$5$m80R3zqK
$0x5XGubRJdi4zj0JzNSmVJ3Ni4opDOGpxi2nK/GGzmC"/>
        <propval type="astring" name="type" value="normal"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/timezone">
    <instance enabled="true" name="default">
      <property_group type="application" name="timezone">
        <propval type="astring" name="localtime" value="UTC"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/environment">
    <instance enabled="true" name="init">
      <property_group type="application" name="environment">
        <propval type="astring" name="LC_ALL" value="C"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/identity">
    <instance enabled="true" name="node">
      <property_group type="application" name="config">
        <propval type="astring" name="nodename" value="my-zone"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/keymap">
    <instance enabled="true" name="default">
      <property_group type="system" name="keymap">
        <propval type="astring" name="layout" value="US-English"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/console-login">
    <instance enabled="true" name="default">
      <property_group type="application" name="ttymon">
        <propval type="astring" name="terminal_type" value="vt100"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="network/install">
    <instance enabled="true" name="default">
      <property_group type="application" name="install_ipv4_interface">
        <propval type="astring" name="address_type" value="static"/>
        <propval type="net_address_v4" name="static_address" value="192.0.2.13/24"/>
        <propval type="astring" name="name" value="net0/v4"/>
        <propval type="net_address_v4" name="default_route" value="192.0.2.1"/>
      </property_group>
      <property_group type="application" name="install_ipv6_interface">
        <propval type="astring" name="stateful" value="yes"/>
        <propval type="astring" name="stateless" value="yes"/>
        <propval type="astring" name="address_type" value="addrconf"/>
        <propval type="astring" name="name" value="net0/v6"/>
      </property_group>
    </instance>
  </service>
</service_bundle>
```



```

    </instance>
  </service>
  <service version="1" type="service" name="system/name-service/cache">
    <instance enabled="true" name="default"/>
  </service>
  <service version="1" type="service" name="network/dns/client">
    <instance enabled="false" name="default"/>
  </service>
</service_bundle>

```

**例 4** DHCP を使用し DNS を使用しない排他的 IP ゾーン

```

<!DOCTYPE service_bundle SYSTEM "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<service_bundle type="profile" name="sysconfig">
  <service version="1" type="service" name="system/config-user">
    <instance enabled="true" name="default">
      <property_group type="application" name="root_account">
        <propval type="astring" name="login" value="root"/>
        <propval type="astring" name="password" value="$5$Iq/.A.
K9$RQyt6RqsAY8TgnuxL9i0/84QwgIQ/nqck8QsTQdvMylmnop"/>
        <propval type="astring" name="type" value="normal"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/timezone">
    <instance enabled="true" name="default">
      <property_group type="application" name="timezone">
        <propval type="astring" name="localtime" value="UTC"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/environment">
    <instance enabled="true" name="init">
      <property_group type="application" name="environment">
        <propval type="astring" name="LC_ALL" value="C"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/identity">
    <instance enabled="true" name="node">
      <property_group type="application" name="config">
        <propval type="astring" name="nodename" value="my-zone"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/keymap">
    <instance enabled="true" name="default">
      <property_group type="system" name="keymap">
        <propval type="astring" name="layout" value="US-English"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="system/console-login">
    <instance enabled="true" name="default">
      <property_group type="application" name="ttymon">
        <propval type="astring" name="terminal_type" value="sun-color"/>
      </property_group>
    </instance>
  </service>
  <service version="1" type="service" name="network/dns/client">
    <instance enabled="false" name="default"/>
  </service>

```

</service\_bundle>

## 非大域ゾーンのライブゾーン再構成の使用

非大域ゾーンの実行中にそのゾーンのライブ構成を再構成したり、そのゾーンのライブ構成に関して報告したりするには、ライブゾーン再構成を使用します。

このセクションでは、次の情報について説明します。

- [34 ページの「zonecfg ユーティリティー編集モード」](#)
- [35 ページの「非大域ゾーンのライブゾーン再構成について」](#)
- [36 ページの「ライブゾーン構成の一時的な変更」](#)
- [36 ページの「ライブゾーン構成の永続的な変更」](#)
- [36 ページの「ライブゾーン再構成の予行演習」](#)
- [37 ページの「ライブゾーン構成の再ロード」](#)

### zonecfg ユーティリティー編集モード

zonecfg ユーティリティー編集モードを使用して、ゾーンに対して構成の変更を行います。格納済みの永続的構成でも、実行中のライブ構成でも変更できます。zonecfg ユーティリティーは、ライブ再構成で使用する場合、次の編集モードをサポートします。

#### デフォルトモード

安定したストレージに保存された永続的なゾーン構成を作成、変更、および一覧表示します。デフォルトモードで変更したパラメータは、変更を行なった時点では実行中のゾーンに影響しません。デフォルトモードは、ゾーン構成を保守するための主要な方法です。このモードには下位互換性があります。

デフォルトモードで行なった変更を実行中のゾーンで有効にするには、次のいずれかの zoneadm コマンドを発行する必要があります。

- `zoneadm apply` コマンドは、更新済みの永続的ゾーン構成を実行中のゾーンに適用されるようにロードするために使用します。
- `zoneadm reboot` コマンドはゾーンをリブートし、更新済みの永続的ゾーン構成を読み取るために使用します。

#### ライブモード

実行中のライブゾーン構成を取得、検査、および編集します。ライブモードは実行中のゾーンにのみ使用可能です。ライブモードで変更したパラメータ

は、`commit` サブコマンドを使用して、ライブゾーン構成でそれらを有効にした直後に有効になります。

ライブモードで行われた変更は一時的です。変更は、ゾーンが次回リブートするまで有効です。詳細は、[36 ページの「ライブゾーン構成の一時的な変更」](#)を参照してください。

ライブゾーン構成の変更を永続的にするには、`zoneadm apply` コマンドで、永続的ゾーン構成に変更を適用します。詳細は、[36 ページの「ライブゾーン構成の永続的な変更」](#)を参照してください。

ライブモードを有効にするには、`zonecfg` コマンドで `-r` オプションを使用し、これは永続的ゾーン構成ではなくライブゾーン構成を取得します。

```
global$ pfexec zonecfg -z zonename -r
```

デフォルトモードとまったく同様に、`zonecfg -r` コマンドを操作できます。`zonecfg` サブコマンドの完全なセットと対話型モードとバッチモードの両方がサポートされます。

ライブ構成ではすべてのリソースを再構成できるとは限りません。サポートされるリソースのタイプとプロパティについては、『[Oracle Solaris ゾーン構成リソース](#)』の「[Live Zone Reconfiguration Support or Restriction for Resource Types and Global Properties](#)」を参照してください。

## 非大域ゾーンのライブゾーン再構成について

実行中の非大域ゾーンに対して次のタスクを実行するには、ライブゾーン再構成を使用します。

- 現在のライブゾーン構成のレポートを作成し検査する
- ライブゾーン構成を一時的に変更する
- 永続的な構成に行なった変更をライブゾーン構成に適用する

ライブゾーン再構成を管理するには、`zonecfg` および `zoneadm` コマンドを使用します。ゾーン構成の一時的な変更または永続的な変更ができます。一時的な変更は、次のリブートまで有効です。変更を永続的構成に適用する場合は、リブートする必要はありません。

次の構成変更を行う場合、ダウンタイムなくゾーン内でサービスを使用できます。

- リソース制御の変更
- ネットワーク構成の変更
- CPU リソースプールの変更
- ファイルシステムの追加または削除
- 仮想および物理デバイスの追加または削除
- `dataset` リソースの追加または削除

ライブゾーン構成時にサポートされるリソースのタイプとプロパティについては、『Oracle Solaris ゾーン構成リソース』の「[Live Zone Reconfiguration Support or Restriction for Resource Types and Global Properties](#)」を参照してください。

## ライブゾーン構成の一時的な変更

実行中のゾーンの構成に一時的な変更のみを加える必要がある場合があります。たとえば、デバイスのメンテナンスの実行中に一時的にゾーンからリソースを削除したり、特別な目的でリソースを一時的に割り当てるが、ゾーンの実行全体には適用したりしない場合があります。そのようなリソースの削除や追加は、ゾーンのリブート時やリソースが使用できなくなった場合に障害が発生することがあるため、永続的なゾーン構成で行うべきではありません。

ライブモードで一時的に変更したパラメータは、`zonecfg commit` コマンドの発行直後に有効になります。これらの変更は、ゾーンが次回リブートするまで有効です。

## ライブゾーン構成の永続的な変更

永続的ゾーン構成からライブゾーン構成に変更を適用するには、`zoneadm apply` コマンドを使用します。実行中のゾーンで変更を有効にするために、リブートする必要はありません。たとえば、[162 ページの「ライブゾーン構成の永続的な構成変更を行う方法」](#)を参照してください。

## ライブゾーン再構成の予行演習

ライブゾーン構成に対する変更を有効にする前に、`zonecfg commit` および `zoneadm apply` コマンドに次のオプションを使用して、それらの変更の影響をテスト実行できます。

- n 予行演習モード。このコマンドは構成への変更の影響を表示しますが、実行中のゾーンに変更を適用しません。予行演習モードは、ライブゾーン構成に影響を与える `zonecfg commit` または `zoneadm apply` コマンドを発行した場合に実行されるアクションをプレビューするために使用します。
- q 非出力モード。このモードは、すべてのシステムメッセージを抑制し、ステータスコードだけを返します。

## ライブゾーン構成の再ロード

デフォルトモードまたはライブモードでの構成の変更中に、実行中のゾーンの構成が外部で変更された場合、`zonecfg commit` コマンドがエラーを返します。これが発生する可能性があるシナリオには、別の管理者による構成の変更、リソース制御の変更、ネットワーク管理コマンドを使用したゾーンのネットワークパラメータの変更などがあります。

ライブ再構成中に実行中のゾーンの構成が変更された場合、`zonecfg reload` サブコマンドを使用して、外部の構成変更をロードします。

- デフォルトモードで `zonecfg reload` コマンドを発行した場合、このコマンドはユーザーが行なったコミットされていないすべての変更を破棄し、永続的ストレージから構成を再ロードします。
- ライブモードで `zonecfg reload` コマンドを発行した (-r オプション) 場合、このコマンドはコミットされていないすべての変更を破棄し、実行中のゾーンの最新のライブ構成を取得します。

構成が再ロードされたら、構成の変更を繰り返して、コミットできます。

ゾーン構成の再ロードの手順については、[164 ページの「ライブゾーン構成の変更のコミット中の障害から回復する方法」](#)を参照してください。



## 非大域ゾーンのインストール、停止処理、停止、アンインストール、クローニングについて

---

この章では、Oracle Solaris オペレーティングシステムへのゾーンのインストールについて説明します。また、仮想プラットフォームおよびアプリケーション環境を管理する 2 つのプロセス `zoneadm` および `zsch` についても説明します。ゾーンの停止、リブート、クローニング、およびアンインストールに関する情報も提供します。

この章で扱う内容は、次のとおりです。

- 39 ページの「ゾーンのインストールと管理の概要」
- 40 ページの「ゾーンがインストールされている Oracle Solaris 11.4 システムでの自動インストールおよびパッケージ」
- 48 ページの「統合アーカイブからのゾーンの配備について」
- 48 ページの「ゾーンのインストール方法」
- 50 ページの「作成後のゾーンの内容」
- 51 ページの「インストール後のゾーンの情報」
- 53 ページの「ゾーンの停止処理、停止、リブート、およびアンインストールについて」
- 56 ページの「非大域ゾーンのクローニングについて」
- 57 ページの「ブート環境とゾーン」

手順については、第6章「非大域ゾーンのインストール、ブート、クローニング、移動、およびアンインストール」を参照してください。

### ゾーンのインストールと管理の概要

`zoneadm` コマンド (`zoneadm(8)` のマニュアルページに記載されています) は、非大域ゾーンをインストールおよび管理するために使用される主なツールです。

- `zoneadm` コマンドを使用する操作は、大域ゾーンから実行します。
  - ゾーン管理が `root` 以外のユーザーに割り当てられている場合、別のゾーンのコピーを作成するサブコマンドには、`solaris.zone.clonefrom/source-zone` の承認が必要です。
- 

ゾーンのインストールおよび検証の手順については、[第6章「非大域ゾーンのインストール、ブート、クローニング、移動、およびアンインストール」](#) および `zoneadm(8)` のマニュアルページを参照してください。`zoneadm list` コマンドでサポートされるオプションについては、`zoneadm(8)` のマニュアルページも参照してください。

ゾーンの構成手順については、[第1章「非大域ゾーン構成の計画」](#) および `zonecfg(8)` のマニュアルページを参照してください。ゾーンの状態については、『[Introduction to Oracle Solaris Zones](#)』の「[Non-Global Zone State Model](#)」に記載されています。

ゾーンの Oracle Solaris 監査レコードを生成する場合は、非大域ゾーンをインストールする前に [75 ページの「ゾーン内の Oracle Solaris 監査の使用」](#) を参照してください。

## ゾーンがインストールされている Oracle Solaris 11.4 システムでの自動インストールおよびパッケージ

AI クライアントインストールの一環として、非大域ゾーンのインストールおよびシステム構成を指定できます。このリリースでは、Image Packaging System (IPS) がサポートされています。このセクションでは、ゾーンをインストールするときに、IPS パッケージングを使用してオペレーティングシステムをインストールおよび保守する方法について説明します。

このセクションは、Oracle Solaris 大域ゾーンのインストールおよびパッケージングのドキュメント、『[Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新](#)』の [第3章、「ソフトウェアパッケージのインストールおよび更新」](#) を補足するものです。

### ゾーンのパッケージングの概要

`solaris` パッケージングリポジトリは、ゾーン環境の管理に使用されます。

`pkg` コマンドを使用してシステムを新しいバージョンの Oracle Solaris にアップグレードすると、ネイティブゾーン、つまり `solaris` ゾーンは自動的に更新されます。



イメージパッケージングシステム (IPS) ([pkg\(7\)](#)) のマニュアルページに記載) は、パッケージのインストール、アップグレード、削除などのソフトウェアのライフサイクル管理のために提供されるフレームワークです。IPS を使用すると、ソフトウェアパッケージの作成、パッケージリポジトリの作成と管理、および既存のパッケージリポジトリのミラー化を行うことができます。IPS がインストールされているシステムからインターネットにアクセスできる場合、クライアントは `solaris` パッケージングリポジトリにアクセスしてソフトウェアをインストールできます。

ゾーン管理者は、ここに記載されている制限の範囲内でパッケージングツールを使用して、非大域ゾーンにインストールされたすべてのソフトウェアを管理できます。

- パッケージが大域ゾーンにインストールされている場合、非大域ゾーンでは大域ゾーンのシステムリポジトリサービスからパッケージをインストールすることができ、そのパッケージをインストールするためにネットワークを使用する必要はありません。そのパッケージが大域ゾーンにインストールされていない場合、ゾーンはゾーンプロキシサービスを使用してパブリッシャーにアクセスし、大域ゾーンを使用して、ネットワーク経由でパッケージをインストールする必要があります。
- 大域管理者または適切な権利を持つゾーン管理者は、システムのすべてのゾーン内のソフトウェアを管理できます。
- 非大域ゾーンのルート (`/`) ファイルシステムは、Oracle Solaris パッケージングツールを使用することで、大域ゾーンから管理できます。Oracle Solaris パッケージングツールは、共通パッケージ (バンドル) 製品、スタンドアロン (別パッケージ) 製品、およびサードパーティー製品を管理するために、非大域ゾーン内でサポートされます。
- パッケージツールは、ゾーン対応の環境で動作します。このツールを使用すると、パッケージを非大域ゾーンにもインストールできます。

---

**注記** - 特定のパッケージ操作が行われている間、ゾーンはこの種のほかの操作から一時的にロックされます。システムはまた、要求された操作について管理者に確認してから処理を続行することがあります。

---

## パッケージとゾーンについて

`solaris` ブランドゾーンにインストールされているソフトウェア ([solaris\(7\)](#) のマニュアルページに記載) は、大域ゾーンにインストールされているソフトウェアと互換性がある必要があります。`pkg` コマンドを使用すると、この互換性が自動的に適用されます。大域ゾーンで `pkg update` コマンドを実行してソフトウェアを更新すると、ゾーンも更新されて、ゾーンと大域ゾーンの同期が維持されます。非大域ゾーンと大域ゾーンには、異なるソフトウェアをインストールできます。ゾーンで `pkg` コマンドを使用して、ゾーン内のソフトウェアを管理することもできます。

`pkg update` コマンド (FMRI の指定なし) を大域ゾーンで実行すると、`pkg` はシステムの非大域ゾーンと大域ゾーンのすべてのソフトウェアを更新します。

`pkg install` の試行インストール (予行演習とも呼ばれる) 機能を Oracle Solaris ゾーンで使用できます。試行インストールを実行するには、`pkg-install -n` コマンド構文を使用します。試行インストール中にシステムから拒否メッセージが生成された場合、問題の可能な解決策について、『Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新』の「パッケージをインストールできない」を参照してください。

ゾーンパッケージバリエーションを使用すると、パッケージ内の各種のコンポーネントが、大域ゾーンまたは非大域ゾーンのいずれかのみインストールされるように明確にタグ付けされます。指定されるパッケージには、非大域ゾーンにインストールされないようにタグ付けされたファイルを含めることができます。

非大域ゾーンのインストール時に、大域ゾーンの Oracle Solaris パッケージの一部だけが完全にレプリケートされます。たとえば、非大域ゾーン内では、Oracle Solaris カーネルパッケージの多くは必要ありません。すべての非大域ゾーンは、大域ゾーンと同一のカーネルを共有します。

詳細は、『Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新』の「非大域ゾーンの操作」および『Oracle Solaris 12 システムの自動インストール』を参照してください。

---

**注記** - 非大域ゾーンを含むシステム上で大域ゾーンを更新すると、ゾーンに関するパッケージダウンロード情報が 2 回表示されているように見える場合があります。しかし、パッケージがダウンロードされるのは一度だけです。

---

## ゾーンがインストールされているシステムへのパッケージの追加について

パッケージをインストールするには、`pkg install` コマンドを使用します。パッケージの依存性は IPS で自動的に処理されます。

```
$ pfexec pkg install package-name
```

- 大域ゾーンで `pkg install` コマンドを使用すると、パッケージは大域ゾーンのみ追加されます。パッケージはほかのゾーンには伝達されません。
- 非大域ゾーンで `pkg install` コマンドを使用すると、パッケージはその非大域ゾーンのみ追加されます。特定の非大域ゾーン内にパッケージを追加するには、ゾーン管理者としてそのゾーンに入り、`pkg install` コマンドを実行します。

## カスタム AI マニフェストを使用したゾーンへのパッケージの追加

自動インストール (AI) マニフェストを改訂して、インストール時にゾーンにソフトウェアを追加する処理を自動化できます。指定したパッケージと、それらが依存す

るパッケージがインストールされます。パッケージのデフォルトのリストは AI マニフェストから取得されます。デフォルトのゾーン用 AI マニフェストは `/usr/share/auto_install/manifest/zone_default.xml` です。

パッケージの特定方法および操作方法については、『[Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新](#)』を参照してください。

## ▼ ゾーンのデフォルトの AI マニフェストを改訂する方法

次の手順では、AI マニフェストをカスタマイズしてゾーンをインストールし、その変更を適用する方法を示します。この手順は大域ゾーンで実行します。

---

**注記** - デフォルトでは、`solaris-small-server` から最小限の `vim-core` のみがインストールされます。

---

始める前に このタスクを完了するには、Install Service Management 権利プロファイルを割り当てられている必要があります。root 役割は、このタスクを実行するための権利を持っています。詳細は、『[Oracle Solaris 12 システムの自動インストール](#)』の「[Using Rights Profiles to Install Oracle Solaris](#)」を参照してください。

1. デフォルトの AI マニフェストをファイルの編集場所にコピーし、書き込み可能にします。

```
$ cp /usr/share/auto_install/manifest/zone_default.xml ~/zonename-ai.xml
$ chmod 644 ~/zonename-ai.xml
```

2. マニフェストを編集します。

次の例は、`software_data` セクションに追加された、`mercurial` および `vim` パッケージのエントリを示しています。

```
<software_data action="install">
  <name>pkg:/group/system/solaris-small-server</name>
  <name>pkg:/developer/versioning/mercurial</name>
  <name>pkg:/editor/vim</name>
</software_data>
```

3. ゾーンをインストールします。

```
$ zoneadm -z zonename install -m ~/zonename-ai.xml
```

システムによって、次のような出力が表示されます。

```
A ZFS file system has been created for this zone.
Progress being logged to /var/log/zones/zoneadm.20111113T004303Z.my-zone.installnonglobal

Image: Preparing at /system/zones/my-zone/root.

Install Log: /system/volatile/install.15496/install_log
AI Manifest: /tmp/manifest.xml.XfawpE
```

```
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Zonename: my-zone
Installation: Starting ...

      Creating IPS image
      Installing packages from:
        solaris
          origin: http://localhost:1008/
solaris/54453f3545de891d4daa841ddb3c844fe8804f55/

DOWNLOAD          PKGS          FILES          XFER (MB)
Completed          169/169      34047/34047    185.6/185.6

PHASE              ACTIONS
Install Phase      46498/46498

PHASE              ITEMS
Package State Update Phase  169/169
Image State Update Phase    2/2
Installation: Succeeded
...
```

## ゾーン内でのパッケージの削除について

ゾーンがインストールされているシステムからパッケージを削除するには、`pkg uninstall` コマンドを使用します。

```
$ pfexec pkg uninstall package_name
```

## パッケージ情報の照会

ゾーンがインストールされているシステムでソフトウェアパッケージデータベースを照会するには、`pkg info` コマンドを使用します。

大域ゾーンでこのコマンドを使用すると、大域ゾーン内だけでソフトウェアパッケージデータベースを照会できます。非大域ゾーンでこのコマンドを使用すると、非大域ゾーン内だけでソフトウェアパッケージデータベースを照会できます。

## インストール済みのゾーンがあるシステムでのプロキシ構成

ゾーンからパッケージリポジトリに接続するときは、2つのオプションがあります。永続的なプロキシを作成するか、またはパッケージのインストール時にプロキシを指定できます。

- イメージに永続的なプロキシを設定する場合は、『[Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新](#)』の第 5 章、「インストールされるイメージの構成」の説明に従って `--proxy` オプションを使用します。
- インストール時にプロキシを指定する場合は、`system-repository` サービスのプロパティを変更して、`http_proxy` および `https_proxy` 環境変数と同じプロキシを使用するようにします。[pkg\(1\)](#) のマニュアルページを参照してください。

大域ゾーンに構成されたりリポジトリへのアクセスは、`system-repository` サービスを使用して非大域ゾーンに提供されます。大域ゾーンの起点に関するプロキシの更新は、`system-repository` の構成に対して自動的に行われます。次の例が示すように、この方法を使用する場合、`system-repository` SMF サービスに対する変更は必要ありません。

#### 例 5 大域ゾーンでのプロキシの構成

```
$ pfexec pkg set-publisher --proxy http://www-proxy -g http://pkg-server pub
```

プロキシが 80 以外のポートで接続を受け入れる場合を除き、ポートを指定する必要はありません。システム上にゾーンが存在する場合は、システムリポジトリサービスが再起動され、プロキシを使用して `pkg-server` へのアクセスが提供されます。

## https\_proxy および http\_proxy を使用した system-repository プロキシのオーバーライド

イメージにプロキシを設定すること、および `system-repository` サービスプロキシのみを設定することをお勧めします。`pkg` コマンドを実行する環境には、`https_proxy` および `http_proxy` 環境変数を設定するようにしてください。

`system-repository` SMF サービスによって使用されるプロキシを構成して、大域ゾーンのパブリッシャーに構成されているプロキシをオーバーライドすることもできます。`system-repository` プロキシは、`config/http_proxy` または `config/https_proxy` SMF プロパティを使用して設定できます。

詳細は、『[Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新](#)』および 46 ページの「[https\\_proxy および http\\_proxy を設定して大域ゾーンのプロキシをオーバーライドする方法](#)」の手順を参照してください。

この手順は、IPS パブリッシャーリポジトリに直接接続していない内部サブネット上の `system-repository` サービスにプロキシを設定します。この構成は、大域ゾーンで `pkg` コマンドによって構成されたすべてのプロキシをオーバーライドします。非大域ゾーンは、HTTP を介して `system-repository` と通信します。`system-repository` は、大域ゾーンに構成されたそのリポジトリ用のプロトコルを使用してパブリッシャーにアクセスします。

この構成によって、solaris 非大域ゾーンから大域ゾーンに設定されているパブリッシャーにも接続できるようになります。solaris ゾーンへの再帰的 pkg 操作は成功します。

## ▼ https\_proxy および http\_proxy を設定して大域ゾーンのプロキシをオーバーライドする方法

次の手順は、http\_proxy および https\_proxy 環境変数と SMF サービスプロパティを使用して、大域ゾーンと非大域ゾーンから IPS リポジトリにアクセスできるようにする方法を示しています。

これらの環境変数は、ユーザーが非大域ゾーンから pkg コマンドを実行してシステムパブリッシャーの URI (Universal Resource Identifier) に接続しないかぎり、起点に設定されたプロキシ構成をオーバーライドします。その場合、コマンドは system-repository を経由します。

解決可能なホスト名を使用することもできます。

### 1. 大域ゾーンの shell にプロキシを設定します。

プロキシを設定すると、pkg コマンドがプロキシサーバー経由でパブリッシャーに到達できるようになります。これは、https または http URL を使用し、大域ゾーンの system-repository を経由しない pkg 操作に影響を与えます。pkg コマンドの詳細は、[pkg\(1\)](#) のマニュアルページを参照してください。

```
$ export http_proxy=http://192.0.2.0
$ export https_proxy=http://192.0.2.0
```

### 2. システム上の solaris ゾーンが大域ゾーンから直接アクセスできる構成済みのシステムパブリッシャーを使用できるようにします。

```
$ svccfg -s system-repository:default setprop config/http_proxy = http://192.0.2.0
$ svccfg -s system-repository:default setprop config/https_proxy = http://192.0.2.0
```

### 3. ライブ SMF リポジトリ内で変更を有効にします。

```
$ svcadm refresh system-repository
```

### 4. 設定が機能していることを確認します。

```
$ svcprop -p config/http_proxy system-repository
$ svcprop -p config/https_proxy system-repository
```

## ゾーンの並列更新

ゾーンは、順次ではなく並列的に更新されるように構成できます。並列更新により、システム上のすべてのゾーンを更新するのに必要な時間が大幅に短縮されます。

詳細と構成例については、『[Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新](#)』の「[複数の非大域ゾーンの同時更新](#)」を参照してください。

## ゾーンの状態がパッケージの操作に与える影響

次のゾーン状態のリストは、各ゾーン状態がパッケージの操作に与える影響を示しています。

### configured

パッケージツールを実行できます。ソフトウェアはまだインストールされていません。

### incomplete

ゾーン上で `zoneadm` コマンドが動作している場合は、パッケージツールを使用しないでください。

ゾーン上で `zoneadm` プロセスが動作していない場合は、パッケージ操作を実行しても安全ですが、このゾーン内のソフトウェアは変更されず、ゾーン内のソフトウェアは依存性の解決に影響を与えません。

### installed

パッケージツールを実行できます。

`zoneadm -z zonename install` コマンドが完了した直後にも、ゾーンは `installed` 状態に移行されます。

### ready

パッケージツールを実行できます。

### running

パッケージツールを実行できます。

### unavailable

ゾーン内のソフトウェアイメージにアクセスできません。ソフトウェアイメージは変更されず、依存性の解決にも影響を与えません。

非大域ゾーンのストレージにアクセスできない場合、または非大域ゾーンのイメージが大域ゾーンのイメージと同期していない場合、そのゾーンは `unavailable` 状態に移行します。この状態移行は、非大域ゾーンによって大域ゾーンのパッケージ操作がブロックされることを防止します。

`unavailable` 状態の原因となった問題を修正したあとで、更新を可能にする `solaris` ブランドのいずれかの `attach` オプションを使用してゾーンを接続する必要がある場合があります。たとえば、大域ゾーンと使用不可の状態になっている非大域ゾーンの間で重要なソフトウェアのバージョンを同期するには、次のコマンドが必要な場合があります。

```
$ pfexec zoneadm -z zonename attach -u
```

## 統合アーカイブからのゾーンの配備について

統合アーカイブを使用して、次の方法でゾーンの作成と配備を行うことができます。

- 統合アーカイブから Oracle Solaris ゾーンを直接作成および配備します。
- `zonecfg` コマンドでアーカイブをテンプレートとして使用して、新しいゾーン構成を作成します。その後、`zoneadm` コマンドを使用してゾーンをインストールします。
- アーカイブから新規ゾーン構成を作成したあと、そのゾーンをイメージパッケージングシステム (IPS) やその他の方法でインストールします。
- 既存のゾーン構成を使用し、アーカイブを使用してインストールします。そのため、ゾーンの構成とゾーンのインストールで異なるアーカイブをソースとすることもできます。

詳細および手順については、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』の「[How to Deploy a Zone From a Unified Archive](#)」を参照してください。

## ゾーンのインストール方法

solaris ブランドのインストーラは、次に示すゾーンのインストール方法をサポートしています。

- solaris **パブリッシャーの起点**は、大域ゾーン内にインストールされているものと同じソフトウェアを最低限含んでいる必要があります。リポジトリには古いソフトウェアまたは新しいソフトウェアを含めることもできますが、大域ゾーンにインストールされているものと同じソフトウェアを含む必要があります。
- Oracle Solaris リリースが稼働しているインストール済みシステムまたは solaris 非大域ゾーンの**統合アーカイブファイルイメージ**。
- **ゾーンブート環境 (ZBE)** (`zoneadm install -z zbe` コマンドを使用)。システムは、必要に応じてパッケージ更新を実行します。

solaris ゾーンでは、次のインストーラオプションがサポートされています。詳細は、[zones\(7\)](#) および [zoneadm\(8\)](#) のマニュアルページを参照してください。例のコマンド行については、[138 ページの「構成済みのゾーンをインストールする方法」](#)を参照してください。

```
-a unified-archive
```

非大域ゾーンまたは非大域ゾーンの統合アーカイブのパスまたは file、http、https URI。



-a *archive* オプションを使用すると、必要に応じてパッケージの更新が行われます。必要な場合は、`zoneadm attach` サブコマンドを使用して、ゾーンをその元のホストに再接続できます。

-c *{profile | dir}*

構成時に適用する、プロファイルまたはプロファイルのディレクトリを指定します。

- プロファイルまたはプロファイルのディレクトリを指定しないと、システムの構成に対話式システム構成ツールが使用されます。
- プロファイルを指定すると、構成手順が非対話的に実行されます。プロファイルは、`.xml` ファイル拡張子を持つ必要があります。
- ディレクトリを指定した場合は、そのディレクトリ内のすべてのプロファイルが有効で、正しい形式の構成プロファイルである必要があります。ディレクトリに `enterprise`、`site`、`node`、または `sysconfig` という名前のサブディレクトリが含まれている場合、そのサブディレクトリ内のプロファイルは指定されたレイヤーで適用されます。

-d *path*

インストール済みシステムまたは非大域ゾーンのルートディレクトリのパス。必要に応じて、パッケージの更新が行われます。*path* にハイフン (-) を指定すると、`zonepath` にシステムイメージがすでに配置されているとみなされます。-d オプションと -a オプションは両立しません。

-m *manifest*

AI マニフェストは、ゾーンのインストール方法を定義する XML ファイルです。ファイルの引数は絶対パスで指定する必要があります。

-p

ゾーンのインストール後にシステム ID を保持します。-p オプションと -u オプションは両立しません。

-s

サイレントインストールします。-s オプションと -v オプションは両立しません。

-u

必要に応じて、すべてのパッケージを最新のバージョンに更新して、大域ゾーンにインストールされているパッケージとの互換性を持たせます。

-u

インストール後にゾーンを構成解除し、ゾーンブート時に新しい構成を求めるプロンプトを表示します。-p オプションと -u オプションは両立しません。

-v

インストールプロセスの詳細情報を出力します。-s オプションと -v オプションは両立しません。

-x *extended-option*

- **force-zpool-import** -x オプションとともに使用すると、使用中と見られる zpool が強制的にインポートされます。  
ストレージオブジェクトに既存のパーティション、zpool、または UFS ファイルシステムが含まれる場合は、install が失敗し、エラーメッセージが表示されます。zoneadm install に対して -x オプションを使用すると、インストールが続き、既存のデータが上書きされます。このオプションは、zpool create -f コマンドに似ています。
- **force-zpool-create-all** -x オプションとともに使用すると、すべての zpool リソースが強制的に作成されます。
- **force-zpool-create=zpoolname** - このオプションを特定の zpool または zpool のセットに制限する場合に使用します。使用方法については、[zoneadm\(8\)](#) のマニュアルページを参照してください。

## 作成後のゾーンの内容

このセクションの内容は、既存のゾーンのクローニングにではなく、初期のゾーン構築に適用されます。

ゾーンは、zoneadm install -m コマンドに渡されるマニフェストによって指定されたパッケージを使用してインストールされます。マニフェストが指定されない場合、zone\_default.xml マニフェストは pkg:/group/system/solaris-small-server を使用します。

新しいゾーンはデフォルトの solaris 構成とログを保持し、これらは、zoneadm install -s コマンドに渡されたシステム構成プロファイル、およびすべての zonecfg add net エントリに指定されたネットワーク接続情報によってのみ変更されます。

システムリポジトリ、ゾーンの構成されたパブリッシャー、および大域ゾーンとの同期が維持されたパッケージについては、[40 ページの「ゾーンがインストールされている Oracle Solaris 11.4 システムでの自動インストールおよびパッケージ」](#)を参照してください。

ゾーンのルートファイルシステムに必要とされるファイルは、システムによりゾーンのルートパス内にインストールされます。

正常にインストールされたゾーンは、ブートと初期ログインの準備が整っています。

ゾーンのインストール時に、次の場所からデータが参照またはコピーされます。

- インストールされていないパッケージ
- CD および DVD 内のデータ
- ネットワークインストールイメージ

また、大域ゾーンに存在可能な、次のタイプの情報は、インストール中のゾーンにはコピーされません。

- /etc/passwd ファイル内の新規または変更されたユーザー
- /etc/group ファイル内の新規または変更されたグループ
- DHCP アドレスの割り当てなどのネットワークサービスの構成
- sendmail などのネットワークサービスのカスタマイズ
- ネームサービスなどのネットワークサービスの構成
- 新規または変更された crontab、プリンタ、およびメールファイル
- システムログ、メッセージ、およびアカウントファイル

デフォルトでは、Oracle Solaris 監査が使用されます。必要に応じて行うファイル変更については、75 ページの「ゾーン内での Oracle Solaris 監査の使用」を参照してください。

ゾーンの状態が `installed` から `ready` に移行する際、構成ファイルで指定されたリソースが追加されます。システムにより、一意のゾーン ID が割り当てられます。ファイルシステムがマウントされ、ネットワークインタフェースが設定され、デバイスが構成されます。`ready` 状態に移行すると、仮想プラットフォームでユーザープロセスの実行を開始する準備が整います。

`ready` 状態では、仮想プラットフォームを管理するため、`zsched` および `zoneadmd` プロセスが開始されます。

- `zsched` プロセスは、`sched` プロセスに類似したシステムスケジューリングプロセスです。ゾーンに関連付けられたカーネルリソースの追跡に使用されます。
- `zoneadmd` プロセスは、ゾーン管理デーモンです。

`ready` 状態のゾーンには、そのゾーン内で実行中のユーザープロセスは存在しません。ゾーンを `running` 状態にするには、そのゾーン内で少なくとも 1 つのプロセスが実行中である必要があります。詳細は、[init\(8\)](#) のマニュアルページを参照してください。

## インストール後のゾーンの情報

`zoneadmd` デーモンはゾーンを管理し、`zsched` プロセスはゾーンによるカーネルの使用を追跡し、管理者は `sysconfig` コマンドを使用してゾーンのアプリケーション環境を設定します。

## zoneadmd ゾーン管理デーモン

ゾーン管理デーモン `zoneadmd` は、ゾーンの仮想プラットフォーム管理用の主要なプロセスです。このデーモンは、ゾーンのブートおよび停止処理の管理も担当します。システム上のアクティブな (準備完了、稼働中、または停止処理中の) ゾーンごとに、1つの `zoneadmd` プロセスが存在します。

`zoneadmd` デーモンは、ゾーン構成での指定に従ってゾーンを設定します。このプロセスには、次の処理が含まれます。

- ゾーン ID を割り当てて、`zsched` システムプロセスを開始する
- ゾーン規模のリソース制御を設定する
- ゾーン構成の指定に従ってゾーンのデバイスを準備する
- ネットワークインタフェースを設定する
- ループバックおよび従来のファイルシステムをマウントする
- ゾーンコンソールデバイスをインスタンス化および初期化する

`zoneadmd` デーモンは、まだ実行されていない場合は `zoneadm` コマンドによって自動的に起動されます。そのため、このデーモンが何らかの理由で実行されていない場合は、ゾーンを管理するための `zoneadm` コマンドのいずれかの呼び出しによって `zoneadmd` デーモンが再起動されます。

詳細は、[zoneadmd\(8\)](#) のマニュアルページを参照してください。

## zsched ゾーンスケジューラプロセス

アクティブなゾーンとは、`ready`、`running`、または `shutting_down` 状態のゾーンのことです。すべてのアクティブなゾーンには、カーネルプロセス `zsched` が関連付けられています。ゾーンのために処理を実行するカーネルスレッドは、`zsched` プロセスにより所有されています。`zsched` プロセスにより、ゾーンサブシステムがカーネルスレッドをゾーンごとに追跡することが可能になります。

## ゾーンアプリケーション環境

ゾーンアプリケーション環境の作成には、`zoneadm` コマンドが使用されます。

ゾーンの内部構成は、`sysconfig` インタフェースを使用して指定します。内部構成では、使用するネームサービス、デフォルトのロケールおよびタイムゾーン、ゾーンの

ルートパスワード、およびほかのアプリケーション環境特性を指定します。ゾーンのデフォルトロケールおよびタイムゾーンは、大域設定には関係なく構成できます。

sysconfig インタフェースについては、『[Oracle Solaris 12 システムの自動インストール](#)』の第9章、「[Unconfiguring or Reconfiguring an Oracle Solaris Instance](#)」および `sysconfig(8)` と `zones(7)` のマニュアルページで説明されています。

## ゾーンの停止処理、停止、リブート、およびアンインストールについて

大域ゾーンの場合と同様に、`shutdown` はスクリプトを実行してゾーンを正常にシャットダウンするのに対し、`halt` はより高速に、ただしあまりクリーンではない方法でゾーンをシャットダウンします。リブート時にはいくつかのオプションがあります。アンインストールを実行すると、ゾーンが完全に削除されます。

### ゾーンのシャットダウンについて

`zoneadm shutdown -c` コマンドは、ゾーンを正常にシャットダウンします。このアクションは、ゾーン内で `init 0` コマンドを実行するのと同様です。`-r` オプションも指定すると、ゾーンはリブートされます。サポートされるブートオプションについては、[54 ページの「ゾーンのブートオプション」](#)を参照してください。

`svc:/system/zones` サービスは、大域ゾーンのシャットダウン時に、`zoneadm shutdown` コマンドを使用してゾーンを正常にシャットダウンします。

`shutdown` サブコマンドはゾーンが正常に停止処理されるまで待機します。このアクションが妥当な時間内に完了しない場合は、`zoneadm halt` コマンドを使用すると、そのゾーンを強制的に停止できます。

### ゾーンの停止について

`zoneadm halt` コマンドは、ゾーン内で実行されているすべてのプロセスを終了し、仮想プラットフォームを削除します。これにより、ゾーンは `installed` 状態に戻れます。すべてのプロセスが終了し、デバイスが構成解除され、ネットワークインタフェースが破棄され、ファイルシステムがアンマウントされ、カーネルデータ構造が破棄されます。

`halt` コマンドにより、ゾーン内部の停止処理スクリプトが実行されることはありません。ゾーンを正常にシャットダウンするには、[53 ページの「ゾーンのシャットダウン」](#)を参照してください。

[トダウンについて](#)を参照してください。別の方法として、ゾーンにログインして `shutdown` コマンドを実行することもできます。[187 ページの「ゾーンをシャットダウンする方法](#)」を参照してください。

## ゾーンのリブートについて

`zoneadm reboot` コマンドはゾーンをリブートします。ゾーンのリブート時に、ゾーン ID が変更されます。

## ゾーンの自動ブート

大域ゾーンのブート時にゾーンを自動的にブートするには、ゾーンの構成内の `autoboot` グローバルプロパティを `true` に設定します。デフォルトの設定は `false` です。

---

**注記** - ゾーンを自動的にブートするには、ゾーンサービス `svc:/system/zones:default` も有効になっている必要があります。このサービスは、デフォルトで有効になっています。

---

`pkg update` 操作時の `autoboot` の設定については、[40 ページの「ゾーンのパッケージングの概要](#)」を参照してください。

## ゾーンのブートオプション

`zoneadm boot` および `zoneadm reboot` コマンドでは、次のブートオプションを使用できます。

### -i *alt-init*

最初のプロセスとなる代替実行可能ファイルを選択します。`alt-init` 値は、実行可能ファイルへの有効なパスでなければなりません。デフォルトの最初のプロセスについては、[init\(8\)](#) のマニュアルページで説明されています。

### -m *smf-options*

SMF のブート動作を制御します。SMF サービスオプションは、システムのブートに使用されるサービスを決定します。復元オプションとメッセージオプションという、2 種類のオプションがあります

- 復元オプションは次のとおりです。

**debug**

標準のサービス別出力と、ログに記録されるすべての `svc.startd` メッセージを出力します。

**milestone=*milestone***

指定された SMF *milestone* にブートします。有効なマイルストーンは、`none`、`single-user`、`multi-user`、`multi-user-server`、および `all` です。

- メッセージオプションは、ブート中に表示されるメッセージの種類と数を決定します。メッセージオプションは次のとおりです。

**quiet**

標準のサービス別出力と、管理者の介入を必要とするエラーメッセージを出力します。

**verbose**

標準のサービス別出力と、詳細情報を提供するメッセージを出力します。

**-s**

マイルストーン `svc:/milestone/single-user:default` に対してのみブートします。このマイルストーンは、`init` のレベル `s` と同等です。

使用例については、[144 ページの「ゾーンのブート方法」](#) および [145 ページの「ゾーンをシングルユーザーモードでブートする方法」](#) を参照してください。

Oracle Solaris サービス管理機能 (SMF) および `init` については、[『Oracle Solaris 11.4 でのシステムサービスの管理』](#) および `svc.startd(8)` と `init(8)` のマニュアルページを参照してください。

## ゾーンのアンインストール

ゾーンのルートファイルシステム内のすべてのファイルをアンインストールする場合に、`zoneadm uninstall` コマンドを使用します。`-F (force)` オプションを合わせて指定しない限り、処理を続行する前に、コマンドプロンプトにより実行の確認が求められます。



---

**注意** - `uninstall` アクションは元に戻すことができません。

---

## 非大域ゾーンのクローニングについて

クローニングを使用すると、システムの既存の構成済みおよびインストール済みゾーンをコピーして、新しいゾーンを同一のシステム上に迅速にプロビジョニングできます。

- ゾーンのクローニングは、ゾーンのインストールほど時間がかかりません。ゴールデンゾーンと呼ばれるバリエーションは、ブートされることのないゾーンですが、将来のクローン作成に使用されます。
- 新規ゾーンには、パッケージの追加やファイルの変更など、ソースゾーンをカスタマイズする過程で加えられた変更がすべて含まれます。

## ゾーンのクローニングの再構成要件

クローンされたゾーンには、次のようにいくつかの再構成が必要です。

- 少なくとも、複数のゾーンで同一であってはならないコンポーネントに対しては、グローバルプロパティとリソースをリセットする必要があります。さらに、共有 IP ゾーンの場合は、各 net リソースの IP アドレスが異なっている必要があります。
- 排他的 IP ゾーンの場合は、各 net リソースの physical プロパティが異なっている必要があります。
- 通常、アプリケーション固有の構成はクローンで再構成する必要があります。たとえば、ゾーン内にデータベースインスタンスがあり、そのゾーンをクローニングする場合は、クローン内のデータベースインスタンスを、別のインスタンスとして自身を認識するように再構成しなければならない可能性があります。

## ゴールデンゾーンについて

定期的にクローンする特定のゾーンがある場合は、そのゾーンのクローン(ゴールデンゾーンと呼ばれる)を作成し、それを使用して将来のクローンを作成できます。または、ゴールデンゾーンとしてのみ使用する新しいゾーンを、必要な構成で作成することもできます。

ゴールデンゾーンは、その有用性を維持するために、ブートされることがありません。したがって、クローニング時にゴールデンゾーンをシャットダウンする必要はありません。ゴールデンゾーンはブートされたことがないため、その作成に使用されたスナップショットからほとんど変わっていません。そのため、ゴールデンゾーンのコストはディスク容量 1 メガバイト未満であると考えられます。新しいブート環境が作



成されると、更新されたパッケージで使用されるディスク容量のコストのみで、ゴールデンゾーンは新しいブート環境を取得します。

## ゾーンのクローニング方法

次の方法のいずれかを使用すると、ゾーンをクローンできます。

- **zoneadm clone コマンド** – この方法は、少数のゾーンをクローニングする場合にもっとも役立ちます。

クローニング元の `zonpath` とクローニング先の `zonpath` が両方とも ZFS 上にあり、同じプールに含まれる場合、`zoneadm clone` コマンドは自動的に ZFS を使用してゾーンをクローンします。ZFS クローンを作成する場合、データが変更されるまでデータは実際にはコピーされません。したがって、最初のクローンにかかる時間はごくわずかです。`zoneadm` コマンドは、ソース `zonpath` の ZFS スナップショットを取得して、ターゲット `zonpath` を設定します。ZFS クローンの名前には、宛先ゾーンの `zonpath` が使用されます。

---

注記 - ZFS クローンを使用してソースを複製できる場合でも、ZFS クローンは行わず ZFS の `zonpath` をコピーするように指定することができます。

---

詳細は、[146 ページの「同一システム上で非大域ゾーンをクローンする方法」](#) および [148 ページの「ゴールデンゾーンを作成する方法」](#) を参照してください。

- **統合アーカイブファイル** – この方法は、大規模な配備で複数のゾーンをクローンする必要がある場合にもっとも役立ちます。この方法でのゾーンのクローニングには、次のタスクが必要です。

1. **統合アーカイブファイルの作成**。統合アーカイブファイルには、すべてのゾーンを含めることも、選択したゾーンを含めることもできます。
2. `zonecfg` および `zoneadm` コマンドを使用した、**1つまたは複数の新しいゾーンの構成およびインストール**。参照ソースとしてアーカイブを使用して新しいゾーンを作成すると、新しいゾーンは元のシステム構成を模倣します。

詳細は、『Oracle Solaris 12 でのシステム復旧とクローン』の第 1 章、「Oracle Solaris System Recovery and Cloning (Overview)」を参照してください。

## ブート環境とゾーン

このセクションでは、ブート環境 (BE)、共有ゾーン、ゾーンの代替ルート、および孤立したゾーンについて説明します。

## 非大域ゾーンでの beadm コマンド

beadm コマンドは、ブート環境 (BE と呼ばれる) の管理に使用されます。beadm コマンドは、ルートデータセットおよびルートデータセットの下にネストされたすべてのデータセットを含むブート環境構造に対してタスクを実行します。

beadm コマンドは非大域ゾーン内でサポートされています。beadm コマンドは、ゾーンの情報を表示しません。ブート環境のゾーンの変更内容を確認するには、大域ゾーンから zoneadm コマンドを使用してください。たとえば、システム上の現在のゾーンのリストを表示するには、zoneadm list コマンドを使用します。詳細は、[beadm\(8\)](#) のマニュアルページを参照してください。

## 大域ゾーンと非大域ゾーンのブート環境

大域ゾーンと非大域ゾーンの両方にブート環境が含まれています。非大域ゾーン内の各ブート環境は、大域ゾーン内の親ブート環境に関連付けられています。大域ゾーンブート環境はそれぞれ、インストールされている各 solaris ゾーン内の 1 つまたは複数のゾーンブート環境 (ZBE) に関連付けられています。ゾーンが切り離された場合、大域ゾーンへの参照は残ります。これにより、大域ゾーンブート環境と非大域ゾーンブート環境の間の関連付けを維持しながら、ゾーンを同じ大域ゾーンに再接続できます。

大域ゾーンブート環境が非アクティブの場合、関連する非大域ゾーンブート環境はブート不可能です。ただし、大域ゾーンのその親ブート環境からブートすると、関連する非大域ゾーンのブート環境はブート可能になります。

ゾーンを接続または移行するとき、システムで適切な ZBE を選択できるようにするために、-z オプションを使用して接続する ZBE を名前指定するか、またはいくつかの -x オプションの 1 つを使用できます。これらのオプションについては、[zones\(7\)](#) のマニュアルページを参照してください。

### -x deny-zbe-clone オプション

このオプションは、ゾーンブート環境 (ZBE) のクローンを拒否し、ZBE を更新してアクティブなブート環境としてマウントします。

デフォルトでは、次のいずれかが当てはまる場合は、選択されたゾーンブート環境がクローニングされ、そのクローンが接続されます。元のゾーンブート環境は引き続き存在します。

- 選択されたゾーンブート環境が別の大域ゾーンブート環境に関連付けられている。
- 選択されたゾーンブート環境が、いずれの大域ゾーンブート環境にも関連付けられていない孤立したブート環境である。

solaris ゾーンの接続時または移行時に `-x deny-zbe-clone` オプションを使用すると、クローニングは行われず、選択されたゾーンブート環境がインプレースで更新され、アクティブなブート環境としてマウントされます。

---

注記 `-x deny-zbe-clone` オプションは、アクティブでないゾーンブート環境にのみ使用できます。ブート環境を非アクティブにマークするには、『[Oracle Solaris 12 ブート環境の作成と管理](#)』の第4章、「[Administering Boot Environments](#)」を参照してください。

---

## **-x destroy-orphan-zbes オプション**

`attach` または `migrate` 操作時に、このオプションは、既存のどの大域ゾーンにも関連付けられていないすべてのゾーンブート環境を破棄します。現在接続している孤立したブート環境は、クローンされ更新されたあとで破棄されます。

## **-x force-zbe-clone オプション**

このオプションは、選択されたゾーンブート環境 (ZBE) をクローンし、クローン ZBE をアクティブなブート環境として接続します。元の ZBE は残ります。

ゾーンを接続するとき、デフォルトでは、現在アクティブな大域ゾーンブート環境に関連付けられているゾーンブート環境が選択されます。このオプションは、この動作をオーバーライドして、別の大域ゾーンブート環境を選択します。

## **共有ゾーンの状態**

ゾーンの状態はホストシステムのすべてのブート環境 (BE) 間で共有されます。ゾーンの状態が変更された場合、現在ブートされている BE のみでなく、すべての BE に影響します。また、ゾーン状態を `installed` から `configured` に変更することも含め、すべての状態に影響します。ゾーンは、アンインストールされた場合、すべての BE からアンインストールされます。アンインストールされたゾーンを回復する唯一の方法は、既存のバックアップデータから回復することです。

## 代替ルートの指定

代替ルート (ブート環境) を指定するには、`zoneadm` コマンドと `-R root` オプションを使用します。このオプションは、`list` および `mark` サブコマンドとの組み合わせでのみ使用できます。詳細は、[zoneadm\(8\)](#) のマニュアルページを参照してください。

## 孤立したゾーンブート環境とクローンについて

別のホストに移行された非大域ゾーンには、既存の大域ゾーンに関連付けられていないゾーンブート環境が集積されることがあります。大域ゾーンに関連付けられていない非大域ゾーンブート環境は、孤立したブート環境と呼ばれます。

孤立したゾーンブート環境が役立つのは、そのゾーンを元のソースシステムで再接続する予定がある場合のみです。以前に接続されていた大域ゾーンに戻す予定がない場合、孤立したゾーンブート環境は破棄してかまいません。

孤立したブート環境をゾーンに接続するように選択すると、孤立したブート環境はクローンされます。このクローニングされたブート環境が更新され、その親はアクティブな大域ブート環境に再設定されます。デフォルトでは、元の孤立したブート環境は引き続き存在します。オプションを指定すると、ブート環境を異なる方法で処理できます。

## beadm コマンドによる孤立したブート環境の特定

孤立したブート環境は、`beadm list` の出力で、Active 列の 0 フラグによって示されます。次の出力例では、`solaris-0`、`solaris-1`、および `solaris-2` ブート環境が孤立しています。

```
$ beadm list
BE           Active Mountpoint Space  Policy Created
--           -
solaris-0 !RO -      3.40M static 2014-01-17 15:04
solaris-1 !RO -      3.85M static 2014-02-09 18:11
solaris-2 !RO -      39.0K static 2014-02-10 04:54
solaris-3 !R  -      2.19G static 2014-02-12 16:56
solaris-4 !R  -      3.0K  static 2014-02-19 19:18
solaris-5 NR  /      2.43G static 2014-03-20 04:38
```

孤立しているとして示されているブート環境は、`beadm destroy` コマンドで破棄できます。孤立したすべてのブート環境を破棄するには、`-o` オプションを含めます。

ブート環境の詳細は、『[Oracle Solaris 12 ブート環境の作成と管理](#)』および [beadm\(8\)](#) のマニュアルページを参照してください。

## 孤立したゾーンブート環境とクローンの zoneadm オプション

次の -x オプションは、ゾーンの zoneadm attach 中にゾーンブート環境 (ZBE) を管理するために使用されます。zoneadm migrate コマンドでも同じオプションが同様に使用され、移行時にターゲットシステムで接続を実行するときに適用されます。

- destroy-orphan-zbes
- force-zbe-clone
- deny-zbe-clone

---

注記 - force-zbe-clone と deny-zbe-clone は、相互に排他的なオプションです。

オプション attach-last-booted-zbe および destroy-orphan-zbes は、force-zbe-clone オプションまたは deny-zbe-clone オプションとともに使用できます。

---



# ◆◆◆ 第 3 章

## 非大域ゾーンへのログインについて

---

この章で扱う内容は、次のとおりです。

- [63 ページの「zlogin コマンドについて」](#)
- [64 ページの「非大域ゾーンへのログイン方法」](#)

手順については、[第7章「非大域ゾーンへのログイン」](#)を参照してください。zlogin コマンドのオプションについては、[zlogin\(1\)](#)のマニュアルページを参照してください。

### zlogin コマンドについて

zlogin コマンドは、大域ゾーンから running 状態または ready 状態にある任意のゾーンにログインするために使用します。

---

**注記** - running 状態にないゾーンへのログインに使用できるのは、zlogin コマンドと -c オプションだけです。

---

zlogin コマンドは、大域管理者または大域ゾーン内で操作している、適切な承認を持つユーザーだけによって使用できます。詳細は、[zlogin\(1\)](#)のマニュアルページを参照してください。

ゾーン管理が root 以外のユーザーに割り当てられている場合、承認 solaris.zone.manage/zonename が必要です。スラッシュ文字 (/) が前に付いた特定の zonename の接尾辞はオプションです。省略すると、承認はすべてのゾーンと一致します。

-c オプションを使用してゾーンコンソールに接続しない限り、zlogin を使用してゾーンにログインすると、新しいタスクが開始されます。1つのタスクを2つのゾーンで実行することはできません。

[158 ページの「非対話型モードを使用してゾーンにアクセスする方法」](#)に記述されているとおり、ゾーン内での実行を指示するコマンドを指定することで、zlogin コマンドを非対話型モードで使用できます。ただし、このコマンドおよびこのコマンドの処理対象となるファイルは、いずれも NFS 上に存在してはなりません。開かれて

いるファイルのいずれか、またはそのアドレス空間のいずれかの部分が NFS 上に存在する場合、コマンドは失敗します。アドレス空間には、コマンドの実行可能ファイル自体またはリンクされたライブラリが含まれます。

## 非大域ゾーンへのログイン方法

このセクションでは、コンソールを使用したログイン、通常のコマンドラインでのログインと管理権限の引き受け、リモートログイン、対話型および非対話型ログイン、フェイルセーフログインについて説明します。

### ゾーンコンソールログイン

各ゾーンは仮想コンソール `/dev/console` を保持します。コンソール上で操作を実行することを、コンソールモードと呼びます。ゾーンが `installed` 状態の場合、ゾーンへのコンソールログインを使用できます。ゾーンコンソールは、システム上のシリアルコンソールに非常に似ています。コンソールへの接続は、ゾーンをリブートしても持続します。コンソールモードと `ssh` ログインとの違いを理解するには、[65 ページの「ゾーンへのリモートログイン」](#)を参照してください。

ゾーンコンソールへのアクセスには、`zlogin` コマンドと `-c` (大文字) オプション、および `zonename` を使用します。ゾーンを `running` 状態にする必要はありません。また、`-d` オプションを使用することもできます。このオプションは、ゾーンが停止した場合に、ゾーンがコンソールから切断されるよう指定します。このオプションは、`-c` オプションとともにのみ指定できます。ゾーン内部のプロセスが、コンソールを開いてメッセージを書き込むことができます。`zlogin -c` プロセスが終了すると、別のプロセスがコンソールにアクセスできるようになります。`zlogin` オプションの詳細については、[`zlogin\(1\)` のマニュアルページ](#)を参照してください。

権利プロファイルが使用中の場合、ゾーンコンソールへのアクセスには承認 `solaris.zone.manage/zonename` が必要です。スラッシュ文字 (`/`) が前に付いた特定の `zonename` の接尾辞はオプションです。省略すると、承認はすべてのゾーンと一致します。ゾーンにアクセスするための権利プロファイルの詳細については、[185 ページの「ゾーン管理者への制限された権利の割り当て」](#)を参照してください。

ブート時に対話式システム構成 (SCI) ツールを起動するには、`sysconfig configure` コマンドを発行します。SCI ツールの詳細については、[29 ページの「対話式システム構成 \(SCI\) ツール」](#) および [`sysconfig\(8\)` のマニュアルページ](#)を参照してください。

手順については、[157 ページの「ゾーンコンソールへのログイン方法」](#) および [156 ページの「ゾーン構成のためにゾーンコンソールにログインする方法」](#)を参照してください。



## ゾーンへのユーザーログイン

ユーザーはユーザー名を使用して、または root ユーザーとしてゾーンにログインできます。

- ユーザー名を使ってゾーンにログインする場合は、`zlogin` コマンドと `-l` オプション、ユーザー名、および `zonename` を使用します。たとえば、大域ゾーンの管理者は、`zlogin` に `-l` オプションを指定することで、通常のユーザーとして非大域ゾーンにログインできます。

```
global$ zlogin -l username zonename
```

- ユーザー `root` でログインするには、オプションを指定せずに `zlogin` を使用します。



**注意** - セキュリティーのため、root ユーザーの代わりに root 役割を使用して、ゾーンにアクセスし、管理します。185 ページの「ゾーン管理者への制限された権利の割り当て」および『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護』の「ユーザー権管理」を参照してください。

## ゾーンへのフェイルセーフログイン

ログインの問題が発生し、`zlogin` または `zlogin -c` を使用してゾーンにアクセスできない場合、代替方法が提供されます。`zlogin` コマンドと `-S (safe)` オプションを使用することで、ゾーンに入ることができます。このモードは、ほかのログイン方法が成功しなかったときに、損傷を受けたゾーンを回復する場合にのみ使用してください。この最小環境では、ゾーンログインが失敗した理由を診断できる場合があります。

158 ページの「フェイルセーフモードを使用してゾーンに入る方法」を参照してください。

## ゾーンへのリモートログイン

Oracle Solaris では、インストール時にデフォルトでのセキュリティー強化 (SBD) 機能が自動的に実装されます。この機能では、Secure Shell が Oracle Solaris システムへの有効な唯一のリモートログインです。ssh を使用して非大域ゾーンに入ります。詳細は、[ssh\(1\)](#) のマニュアルページを参照してください。

## 対話型ゾーンログイン

対話型モードでは、ゾーン内部で使用する新しい仮想端末が割り当てられます。コンソールデバイスへの排他的なアクセスが許可されるコンソールモードとは異なり、対話型モードでは、いつでも任意の数の `zlogin` セッションを開くことができます。対話型モードが有効になるのは、発行するコマンドが含まれていない場合です。エディタなどの端末デバイスを必要とするプログラムは、このモードで正常に動作します。

ゾーン管理が `root` 以外のユーザーに割り当てられており、ゾーン管理者が対話形式でログインしている場合、ゾーン管理者にはゾーンに対する承認 `solaris.zone.login/zonename` が割り当てられている必要があります。ゾーン内では、パスワード認証が実行されます。

[157 ページの「対話型モードを使用してゾーンにアクセスする方法」](#)を参照してください。

## 非対話型ゾーンログイン

非対話型モードは、ゾーンを管理するシェルスクリプトを実行する場合に使用します。非対話型モードでは、新しい仮想端末は割り当てられません。ゾーン内部で実行されるコマンドを指定すると、非対話型モードが有効になります。

非対話型ログインの場合、またはパスワード承認を省略する場合は、`solaris.zone.manage/zonename` の承認が必要です。

手順については、[158 ページの「非対話型モードを使用してゾーンにアクセスする方法」](#)を参照してください。

## Oracle Solaris ゾーンの管理について

---

この章で扱う内容は、次のとおりです。

- 67 ページの「非大域ゾーンが含まれているシステムのセキュリティー対策」
- 76 ページの「非大域ゾーンが含まれているシステムのモニタリング」
- 80 ページの「非大域ゾーンでの時間値の設定」
- 81 ページの「ファイルシステムと非大域ゾーン」
- 88 ページの「排他的 IP 非大域ゾーンにおけるネットワーク」
- 90 ページの「共有 IP 非大域ゾーンにおけるネットワーク」
- 93 ページの「非大域ゾーンでのデバイスの使用」
- 95 ページの「非大域ゾーンでのアプリケーションの実行」
- 96 ページの「非大域ゾーンでのリソース管理」
- 98 ページの「非大域ゾーンの移行および変換」
- 109 ページの「非大域ゾーンが含まれているシステムのバックアップ方法」
- 112 ページの「非大域ゾーンが含まれているシステムで使用されるコマンド」

カーネルゾーンでの solaris ゾーンの管理については、『Oracle Solaris カーネルゾーンの作成と使用』の「カーネルゾーン内の非大域ゾーンの管理」および『Oracle Solaris カーネルゾーンの作成と使用』の「複数の MAC アドレスをカーネルゾーンに追加する方法」を参照してください。

### 非大域ゾーンが含まれているシステムのセキュリティー対策

Oracle Solaris では、大域ゾーンと非大域ゾーンに同じようなセキュリティー対策を施しています。IPsec と IKE によってネットワークを保護し、権利と監査によってリソースの不正使用を防ぎ、不変ゾーンによって管理のセキュリティーを強化しています。次の各セクションでは、非大域ゾーンへのこれらのセキュリティー機能の実装について説明します。

## ゾーン管理者向けの権利プロファイルの概要

root 役割には、すべての管理権利が含まれています。デフォルトでは、大域ゾーン管理者 (root) はすべての非大域ゾーンを管理できます。

セキュリティのため、root 役割は、信頼できるユーザーが必要な権利だけでゾーンを管理できるように権利を委任する必要があります。Oracle Solaris の権利は特権コマンドと承認であり、個々の責任分野の権利プロファイルにまとめられます。Oracle Solaris では、ゾーン向けにいくつかの権利プロファイルを提供しています。

root 役割には、ゾーン内で管理権利を制限する別の方法も含まれています。各非大域ゾーンには admin リソースが含まれています。root では、admin リソース内にユーザー名と必要な承認を指定することで、そのゾーンへのアクセスを制限できます。サイトのセキュリティ対策として責務を分離する必要がある場合、この方法によって root で非大域ゾーンにアクセスできないようにすることが可能です。

root 役割では権利プロファイルをユーザーに直接割り当てることも、管理役割を作成し、権利プロファイルをその役割に割り当てることもできます。ユーザーは役割を割り当てられると、その役割を引き受けてゾーンを管理する必要があります。

Oracle Solaris の権利とゾーン向けの admin リソースの詳細は、次を参照してください。

- 『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティ保護』の第1章、「権利を使用したユーザーとプロセスの制御について」
- 『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティ保護』の「割り当てられている管理権利の使用」
- 『Oracle Solaris Zones Configuration Resources』の「admin Resource Type」

## root 以外のユーザーへのゾーンを管理する権利の割り当て

さまざまな理由から、大域ゾーンの管理者はゾーンを管理する権利を選ばれたユーザーに割り振ることが必要になる場合があります。これらの理由には次があります。

- root がほかの管理操作に専念できるようにするため
- ゾーンの管理にスペシャリストを割り当てるため
- 攻撃のエントリーポイントの数を減らすため
- 責務の分離を可能にして root の権限が制限されるようにするため
- PCI-DSS や HIPAA コンプライアンスなど、サイトのセキュリティ要件を実装するため

ゾーンの管理を委任するために root 役割で割り当てられる権利には、次の権利プロファイルが含まれます。

### ゾーンのクールド移行

管理者が `installed` 状態または `suspended` 補助状態のゾーンを移行できるようにします。

ゾーンのクールド移行プロファイルを使用して、ゾーン管理者がクールド移行およびウォーム移行を実行できますが、ライブ移行は実行できません。

`admin` リソースタイプの `auths` プロパティが管理対象ゾーンに構成されている場合、この権利プロファイルはゾーンの移行に十分ではありません。さらに、ゾーン管理者が `admin` リソースタイプの `user` プロパティに指定されており、`solaris.zone.migrate.cold` 承認も割り当てられている必要があります。ログインが制限されている場合、ゾーン管理者に `solaris.zone.login` 承認も割り当てられている必要があります。

### ゾーン構成

管理者がホスト上の任意のゾーン構成を作成、変更、または削除できるようにします。

ゾーン構成権利プロファイルを使用して、ゾーン管理者はゾーンを構成できます。移行されるゾーンでは、ゾーンの構成がまだターゲットシステムに存在していない場合、移行を完了するために、管理者にターゲットシステム上のこの権利プロファイルが付与されている必要があります。ゾーン構成権利プロファイルには、`zonecfg` コマンドのみが含まれます。

`admin` リソースタイプの `auths` プロパティが管理対象ゾーンに構成されている場合、この権利プロファイルはゾーンの構成に十分ではありません。さらに、ゾーン管理者が `admin` リソースタイプの `user` プロパティに指定されており、`solaris.zone.config` 承認も割り当てられている必要があります。ログインが制限されている場合、ゾーン管理者に `solaris.zone.login` 承認も割り当てられている必要があります。

### ゾーン管理

管理者が既存のゾーンを管理できるようにします。

ゾーン管理権利プロファイルには、`zlogin` および `zoneadm` コマンドが含まれます。

`admin` リソースタイプの `auths` プロパティが管理対象ゾーンに構成されている場合、この権利プロファイルはゾーンの管理に十分ではありません。さらに、ゾーン管理者が `admin` リソースタイプの `user` プロパティに指定されており、ゾーンにログインして管理するために `solaris.zone.*` 承認も割り当てられている必要があります。

### ゾーンの移行

管理者が任意のタイプのゾーンを移行できるようにします。

ゾーン移行権利プロファイルを使用して、ゾーン管理者はインストール済みまたは実行中のゾーンの移行を実行できます。このプロファイルが割り当てられているゾーン管理者はライブ移行、ウォーム移行、またはクールド移行を実行できま

す。ゾーン移行権利プロファイルには、`zoneadm` および `zonecfg` コマンドが含まれます。

`admin` リソースの `auths` プロパティが管理対象ゾーンに構成されている場合、この権利プロファイルはゾーンの移行に十分ではありません。さらに、ゾーン管理者が `admin` リソースに `user` として指定されており、`solaris.zone.migrate` 承認も割り当てられている必要があります。ログインが制限されている場合、ゾーン管理者に `solaris.zone.login` 承認も割り当てられている必要があります。

### ゾーンのセキュリティ

管理者がホスト上に、Trusted Extensions で構成されたゾーンを含む任意のゾーン構成を作成、変更、または削除できるようにします。

ゾーンセキュリティ権利プロファイルには、`zonecfg` または `txzonemgr` コマンドとすべての `solaris.zone.*` 承認が含まれます。割り当て対象者はゾーン管理を委任できます。txzonemgr については、『[Trusted Extensions 構成と管理](#)』の「[Creating Labeled Zones](#)」を参照してください。

`admin` リソースタイプの `auths` プロパティが管理対象ゾーンに構成されている場合、この権利プロファイルはゾーンの作成、ログイン、および構成に十分ではありません。さらに、ゾーン管理者が `admin` リソースタイプの `user` プロパティに指定されており、`solaris.zone.*` 承認も割り当てられている必要があります。

## ゾーンを管理する権利を制限する場合の一般的なコマンド

権利プロファイルの内容を確認したり、権利プロファイルを割り当てたりするには、次のコマンドを使用します。

- 権利プロファイルの内容を一覧表示します。

```
$ profiles -p "rights-profile-name" info
```

- Zone 権利プロファイル内のコマンドに付与されている権利または特権を表示します。

```
$ getent exec_attr | grep Zone
```

このコマンドでは、「Zone」という語を含む権利プロファイル内の特権コマンドを検索します。

- 権利プロファイルをユーザーに割り当てます。

```
# usermod -K profiles+="rights-profile-name" username
```

## 非大域ゾーン内の特権

プロセスは、特権のサブセットに制限されています。特権を制限することで、ほかのゾーンに影響を及ぼす可能性がある操作がゾーンで実行されないようにします。特権セットにより、特権が付与されたユーザーがゾーン内で実行可能な機能が制限されます。

特定のゾーンで利用可能な特権のリストを表示するには、`ppriv` ユーティリティを使用します。ゾーン内からは `ppriv -l` コマンドを使用します。大域ゾーンからは、このコマンドにゾーン名を追加します (`ppriv -l zonename`)。

次の表に、Oracle Solaris の特権すべて、およびゾーン内での各特権のステータスを示します。省略可能な特権は、デフォルト特権セットの一部ではありませんが、`limitpriv` プロパティを使って指定できます。必須の特権が、生成される特権セットに含まれている必要があります。禁止された特権を、生成される特権セットに含めることはできません。

表 1 ゾーン内のオプションの特権

特権	ステータス	注意事項
<code>cpc_cpu</code>	オプション	特定の <code>cpc</code> カウンタへのアクセスを可能にします
<code>dtrace_proc</code>	オプション	<code>fasttrap</code> 、 <code>pid</code> 、および <code>plockstat</code> プロバイダ用
<code>dtrace_user</code>	オプション	<code>profile</code> および <code>syscall</code> プロバイダ用
<code>file_flag_set</code>	オプション	ファイルを変更できないようにするため
<code>graphics_access</code>	オプション	<code>ioctl</code> による <code>agpgart_io</code> へのアクセスを可能にします
<code>graphics_map</code>	オプション	<code>mmap</code> による <code>agpgart_io</code> へのアクセスを可能にします
<code>proc_clock_highres</code>	オプション	高解像度タイマー用
<code>proc_prioctl</code>	オプション	スケジューリングの制御および <code>prioctl</code> プロバイダ用
<code>sys_ipc_config</code>	オプション	IPC メッセージキューバッファのサイズを大きくするため

表 2 ゾーン内の禁止された特権

特権	ステータス	注意事項
<code>dtrace_kernel</code>	禁止	現在、未サポート
<code>proc_zone</code>	禁止	現在、未サポート
<code>sys_config</code>	禁止	現在、未サポート
<code>sys_devices</code>	禁止	現在、未サポート
<code>sys_dl_config</code>	禁止	現在、未サポート
<code>sys_linkdir</code>	禁止	現在、未サポート

特権	ステータス	注意事項
sys_net_config	禁止	現在、未サポート
sys_res_config	禁止	現在、未サポート
sys_smb	禁止	現在、未サポート
sys_suser_compat	禁止	現在、未サポート

表 3 ゾーン内の必須のデフォルトの特権

特権	ステータス	注意事項
file_read	必須、デフォルト	モード/ACL によってプロセスの読み取り権限が許可されているファイルまたはディレクトリの読み取りに必要
file_write	必須、デフォルト	モード/ACL によってプロセスの書き込み権限が許可されているファイルまたはディレクトリの書き込みに必要
net_access	必須、デフォルト	ネットワークエンドポイントを開くために必要
proc_exec	必須、デフォルト	init の起動に必要
proc_fork	必須、デフォルト	init の起動に必要
sys_flow_config	必須、排他的 IP ゾーンではデフォルト	フローの構成に必要
sys_ip_config	必須、排他的 IP ゾーンではデフォルト 共有 IP ゾーンでは禁止	ゾーンのブートおよび IP ネットワークの初期化に必要
sys_iptables	必須、排他的 IP ゾーンではデフォルト 共有 IP ゾーンでは禁止	IP トンネルリンクの構成に必要
sys_mount	必須、デフォルト	必須ファイルシステムのマウントに必要

表 4 ゾーン内のオプションのデフォルトの特権

特権	ステータス	注意事項
cmi_access	デフォルト	CMI セグメントに使用
cmi_owner	デフォルト	CMI セグメントの表示と変更で使用
contract_event	デフォルト	契約ファイルシステムで使用
contract_identity	デフォルト	プロセス契約テンプレートのサービス FMRI の設定用
contract_observer	デフォルト	UID に依存しない契約観察に使用
file_audit	デフォルト	ファイル単位の監査用
file_chown	デフォルト	ファイル所有権の変更用
file_chown_self	デフォルト	所有するファイルの所有者/グループの変更用



特権	ステータス	注意事項
file_dac_execute	デフォルト	モード/ACL に依存しない実行アクセス用
file_dac_read	デフォルト	モード/ACL に依存しない読み取りアクセス用
file_dac_search	デフォルト	モード/ACL に依存しない検索アクセス用
file_dac_write	デフォルト	モード/ACL に依存しない書き込みアクセス用
file_link_any	デフォルト	所有者に依存しないリンクの設定用
file_owner	デフォルト	所有者に依存しない other によるアクセス用
file_setid	デフォルト	setid、setgid、setuid ファイルの権限の変更用
ipc_dac_owner	デフォルト	モードに依存しない IPC 書き込みアクセス用
ipc_dac_read	デフォルト	モードに依存しない IPC 読み取りアクセス用
ipc_dac_write	デフォルト	モードに依存しない IPC 書き込み権限用
ipc_owner	デフォルト	モードに依存しない other によるアクセス用
kstat_manage	デフォルト	kstats の管理用
kstat_rd_sensitive	デフォルト	機密性の高い kstats の読み取りを許可します
net_icmpaccess	デフォルト	ICMP パケットアクセス (ping) 用
net_observability	デフォルト	ネットワークトラフィックの受信用 (トラフィックの送信は禁止されています)
net_rawaccess	排他的 IP ゾーンではデフォルト	raw PF_INET/PF_INET6 パケットアクセスを許可します
proc_audit	デフォルト	監査レコードの生成用
proc_chroot	デフォルト	root ディレクトリの変更用
proc_info	デフォルト	プロセスの検査を許可します
proc_lock_memory	デフォルト	メモリーのロック用 (shmctl および mlock)  ゾーンに割り当てられている場合は、ゾーンがすべてのメモリーをロックするのを防ぐために zone.max-locked-memory リソース制御の設定も検討してください。
proc_owner	デフォルト	所有者に依存しないプロセス制御用
proc_session	デフォルト	セッションに依存しないプロセス制御用
proc_setid	デフォルト	ユーザー/グループ ID の任意設定を許可します
proc_taskid	デフォルト	呼び出し元へのタスク ID の割り当てを許可します
sys_acct	デフォルト	アカウントの管理用
sys_admin	デフォルト	単純なシステム管理タスク用
sys_audit	デフォルト	監査の管理用
sys_nfs	デフォルト	NFS クライアントのサポート用
sys_ppp_config	排他的 IP ゾーンではデフォルト  共有 IP ゾーンでは禁止	PPP (sppp) インタフェースの作成および削除用、PPP トンネル (sppptun) の構成用
sys_resource	デフォルト	リソース制限の変更用

特権	ステータス	注意事項
sys_share	デフォルト	ファイルシステムの共有に必要。割り当てられていない場合、ゾーン内での NFS 共有は禁止されます。
sys_time	デフォルト	ゾーンの時間の設定用 (xntp)

次の表に、Oracle Solaris Trusted Extensions の特権すべて、および各特権のゾーン内のステータスを示します。省略可能な特権は、デフォルト特権セットの一部ではありませんが、limitpriv プロパティを使って指定できます。

**注記** - Trusted Extensions の特権が解釈されるのは、システムが Trusted Extensions を使って構成されている場合だけです。

表 5 ゾーン内での Oracle Solaris Trusted Extensions の特権のステータス

Trusted Extensions の特権	ステータス	注意事項
file_downgrade_sl	オプション	高いラベルから低いラベルへのファイルのダウングレードを可能にします
file_upgrade_sl	オプション	高いラベルへのファイルのアップグレードを可能にします
net_bindmlp	デフォルト	マルチレベルポート (MLP) へのバインドを可能にします
net_mac_aware	デフォルト	NFS を使用した読み取りの可能にします
sys_trans_label	オプション	プロセスのラベルの範囲外にあるラベルの変換を可能にします

非大域ゾーンで利用できる特権を変更する場合は、132 ページの「[ゾーンの特権を変更する方法](#)」を参照してください。

特権の詳細は、ppriv(1) のマニュアルページおよび『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティ保護』の「[プロセス権管理](#)」を参照してください。

## 不変ゾーンについて

不変ゾーンは、読み取り専用の root ファイルシステムを持つゾーンです。このゾーンは、非大域ゾーンにも大域ゾーンにもできます。読み取り専用の root では、ゾーンの構成が維持されます。また、実行時環境に追加の制限を加えることにより、ゾーンのセキュアな実行時の境界が拡張されます。保守操作は可能ですが、保守を行うにあたっては、管理者は慎重な手順でゾーンにアクセスする必要があります。システムバ

イナリまたはシステム構成への変更は、必須書き込みアクセス制御 (MWAC) セキュリティポリシーによってブロックされます。

MWAC を使用すると、SMF プロパティ `file-mac-profile` を介してファイルシステム書き込み権限を適用できます。MWAC セキュリティポリシーを指定するには、`zonecfg` コマンドで `file-mac-profile` 値を変更します。このポリシーはカーネルで適用されます。大域ゾーンは非大域ゾーンの MWAC ポリシーの対象ではないため、大域ゾーンではインストール、イメージの更新、および保守のために非大域ゾーンのファイルシステムに書き込むことができます。

ゾーンが `ready` 状態になると、MWAC ポリシーがダウンロードされます。このポリシーはゾーンのブート時に有効になります。インストール後のアセンブリと構成を行うには、一時的な書き込み可能ルートファイルシステムのブート処理を使用します。ゾーンの MWAC 構成への変更は、ゾーンをリブートしたときにのみ反映されます。

## ゾーン内での IPsec アーキテクチャーの使用

IP データグラムの保護を提供する IPsec (Internet Protocol Security Architecture) は、排他的 IP ゾーン内で使用できます。IPsec については、『[Securing the Network in Oracle Solaris 11.4](#)』の「[IPsec Reference](#)」に説明されています。IKE (Internet Key Exchange) プロトコルを使用して、認証および暗号化に必要な鍵材料が自動的に管理されます。

詳細は、[ipsecconf\(8\)](#) および [ipseckey\(8\)](#) のマニュアルページを参照してください。

### 共有 IP ゾーン内の IP セキュリティーアーキテクチャー

IPsec は、大域ゾーン内で使用できます。ただし、非大域ゾーン内では、IPsec で IKE を使用することはできません。そのため、大域ゾーンで Internet Key Exchange (IKE) プロトコルを使用して、非大域ゾーンの IPsec キーおよび IPsec ポリシーを管理する必要があります。構成中の非大域ゾーンに対応するソースアドレスを使用します。

### 排他的 IP ゾーン内の IP セキュリティーアーキテクチャー

IPsec は、排他的 IP ゾーン内で使用できます。

## ゾーン内での Oracle Solaris 監査の使用

Oracle Solaris 監査はすべてのゾーン内でデフォルトで有効になっています。

監査には、ゾーンを実行しているシステムでの次の2つの監査モデルが用意されています。

- すべてのゾーンを大域ゾーンから一元的に監査します。このモデルは、すべてのゾーンが大域ゾーンによって管理される場合(たとえば、ゾーンを使用してサービスの隔離を実現する場合)に使用されます。
- 各ゾーンを大域ゾーンから独立して監査します。このモデルは、各ゾーンが別々に管理される場合(たとえば、ゾーンごとにサーバーの統合を実現する場合)に使用されます。

詳細は、次を確認してください。

- 『[Managing Auditing in Oracle Solaris 11.4](#)』の第1章、「[About Auditing in Oracle Solaris](#)」
- 『[Managing Auditing in Oracle Solaris 11.4](#)』の「[Auditing on a System With Oracle Solaris Zones](#)」
- 『[Managing Auditing in Oracle Solaris 11.4](#)』の「[Configuring the Audit Service in Zones](#)」
- [auditconfig\(8\)](#)、[auditreduce\(8\)](#)、[usermod\(8\)](#)、および [user\\_attr\(5\)](#) の各マニュアルページ

## 非大域ゾーンが含まれているシステムのモニタリング

通常、大域ゾーンでは非大域ゾーンとそのプロセスを表示できますが、非大域ゾーンでは互いのプロセスを表示することはできません。ゾーンが含まれていないシステムと同様に、リモートで表示できる情報の制限、コアファイルの作成、時間の設定、DTraceの実行、および統計の表示を行うことができます。

## 大域ゾーンの可視性とアクセス

大域ゾーンは、システムのデフォルトゾーンとしても、システム規模の管理制御用ゾーンとしても機能します。この二重の役割と関係のある管理上の問題が存在します。ゾーン内部のアプリケーションはほかのゾーン内のプロセスおよびほかのシステムオブジェクトにアクセスするため、管理操作の効果が予期したものよりも大きい場合があります。たとえば、サービスの停止スクリプトでは、プロセスを終了させるためのシグナルの送信にしばしば `pkill` が使用されます。大域ゾーンからこの種のスクリプトを実行すると、システム内の該当するすべてのプロセスが、ゾーンに関係なくシグナルを送信します。

多くの場合、システム規模の範囲が必要になります。たとえば、システム規模のリソース使用状況をモニターする場合、システム全体のプロセス統計情報を表示する必

必要があります。大域ゾーンのアクティビティだけのビューには、システムリソースの一部または全体を共有可能なシステム内のほかのゾーンからの関連情報が欠落しています。この種のビューは、CPUなどのシステムリソースがリソース管理機能を使用して厳密に区分されていない場合、特に重要です。

このため、大域ゾーン内のプロセスから、非大域ゾーン内のプロセスおよびほかのオブジェクトを監視できます。これにより、この種のプロセスがシステム規模の可観測性を備えることが可能になります。ほかのゾーン内のプロセスを制御したりシグナルを送信したりする機能は、`PRIV_PROC_ZONE` 特権により制限されます。この特権は、特権のないプロセスに設定された制限をオーバーライドできるため、`PRIV_PROC_OWNER` に類似しています。この場合の制限は、大域ゾーン内の特権のないプロセスはほかのゾーン内のプロセスにシグナルを送信したり制御したりすることはできない、というものです。これは、プロセスのユーザー ID が一致するか、動作しているプロセスが `PRIV_PROC_OWNER` 特権を保持している場合でも適用されます。`PRIV_PROC_ZONE` 特権を、そうでなければ特権の付与されたプロセスから削除して、大域ゾーンへの操作に制限できます。

`zoneidlist` を使用したプロセスの照合については、[pgrep\(1\)](#) および [pkill\(1\)](#) のマニュアルページを参照してください。

## ゾーン内でのプロセス ID の可視性

同一ゾーン内のプロセスだけが、`kill` や `priocntl` コマンドなどの、プロセス ID を指定するシステムコールインタフェースを介して表示されます。詳細は、[kill\(1\)](#) および [priocntl\(1\)](#) のマニュアルページを参照してください。

## ゾーンのシステム可観測性

大域ゾーンで `ps` コマンドを実行する場合、ユーザーおよびグループ名は大域ゾーンのネームサービスを使用して解決されます。システムの非大域ゾーンで実行しているプロセスは、大域ゾーンのネームサービスに一致したユーザーおよびグループ名を表示します。大域ゾーンでのこれらの名前は、非大域ゾーンのネームサービスで構成された名前とは異なる可能性があります。

`ps` コマンドに、ゾーンに関する次の変更が加えられました。

- 出力形式の指定には `-o` オプションを使用します。このオプションを使用すると、プロセスのゾーン ID またはプロセスを実行中のゾーンの名前を出力できます。
- 指定されたゾーン内のプロセスだけをリスト表示するには、`-zzonelist` オプションを使用します。ゾーンの指定には、ゾーン名またはゾーン ID を使用できます。このオプションは、大域ゾーン内でコマンドを実行する場合にのみ有効です。

- プロセスに関連するゾーンの名前を出力するには `-z` オプションを使用します。列見出し `ZONE` の下に名前が出力されます。

詳細は、[ps\(1\)](#) のマニュアルページを参照してください。

`-z zonename` オプションが、次の Oracle Solaris ユーティリティに追加されました。このオプションを使用して情報をフィルタ処理し、指定したゾーンだけを含めることができます。

- `ipcs -z zonename - ipcs(1)` のマニュアルページを参照してください
- `pgrep -z zonename - pgrep(1)` のマニュアルページを参照してください
- `ptree -z zonename - proc(1)` のマニュアルページを参照してください
- `prstat -z zonename - prstat(8)` のマニュアルページを参照してください

コマンドに加えられたすべての変更の一覧は、[表8](#)を参照してください。

## zonestat ユーティリティを使用したアクティブなゾーンの統計情報の報告

`zonestat` ユーティリティを使用する場合は、[zonestat\(1\)](#) のマニュアルページおよび [195 ページ](#) の「非大域ゾーンのリソースの使用状況の報告」を参照してください。

`zonestat` ユーティリティは、現在実行中のゾーンの CPU、メモリー、およびリソース制御の使用効率について報告します。`zonestat` ユーティリティは、指定された間隔で一連のレポートを出力します。必要に応じて、このユーティリティは 1 つ以上のサマリーレポートを出力できます。

`zonestat` ユーティリティは、排他的 IP ゾーン内のネットワーク帯域幅の使用効率についても報告を生成します。排他的 IP ゾーンには、独自の IP に関連付けられた状態と 1 つ以上の専用のデータリンクが保持されます。

非大域ゾーンから実行すると、そのゾーンから参照可能なプロセッサセットのみが報告されます。非大域ゾーンの出力には、すべてのメモリーリソースおよび制限リソースが含まれます。

非大域ゾーンで `zonestat` サービスを使用するには、大域ゾーンの `zonestat` サービスがオンラインである必要があります。各非大域ゾーンの `zonestat` サービスは、大域ゾーンの `zonestat` サービスからシステム構成および使用効率データを読み取ります。

`zonestated` システムデーモンは、システムブート時に開始されます。このデーモンは、ゾーンによるシステムリソースの使用をモニターするほか、`psrset` プロセッサ

セット、プールプロセッサセット、リソース制御設定などのゾーンおよびシステム構成情報をモニターします。構成できるコンポーネントはありません。

## fsstat ユーティリティを使用した非大域ゾーンのモニタリング

fsstat ユーティリティは、ゾーン単位で kstats (アグリゲーションを含む) を収集して出力します。デフォルトでは、このユーティリティは実行中のすべてのゾーンの集計を報告します。fstype ごとの kstat がそれぞれのゾーン用に生成されます。大域ゾーンの kstat は、その排他的なアクティビティを報告します。大域ゾーンでは、システム上のすべてのゾーンの kstat を表示できます。非大域ゾーンでは、ユーティリティが実行されているゾーンに関連付けられた kstats のみが表示されます。非大域ゾーンでは、ほかのゾーンのファイルシステムアクティビティをモニターできません。

詳細は、[fsstat\(8\)](#) のマニュアルページおよび [198 ページの「ゾーンごとのファイルシステム統計情報の報告」](#) を参照してください。

## 非大域ゾーン内での DTrace の実行

dtrace\_proc 特権と dtrace\_user 特権のみを必要とする DTrace プログラムは、非大域ゾーンで実行できます。非大域ゾーンで使用できる特権のセットにこれらの特権を追加するには、zonecfg limitpriv プロパティを使用します。手順については、[187 ページの「非大域ゾーンに DTrace 特権を追加する方法」](#) を参照してください。

dtrace\_proc によってサポートされるプロバイダは、fasttrap と pid です。dtrace\_user によってサポートされるプロバイダは、profile と syscall です。DTrace のプロバイダおよびアクションの有効範囲は、ゾーンに制限されます。

詳細は、[71 ページの「非大域ゾーン内の特権」](#) も参照してください。

## ゾーン内のコアファイル

coreadm コマンドを使用して、異常終了するプロセスにより生成されるコアファイルの名前と場所を指定できます。%z 変数を指定することで、プロセスが実行されたゾー

ンの `zonename` を含むコアファイルパスを生成できます。パス名は、ゾーンのルートディレクトリに対する相対パスです。

詳細は、[coreadm\(8\)](#) および [core\(5\)](#) のマニュアルページを参照してください。

## 非大域ゾーンのノード名

ノード名は、システム名のローカルソースです。ノード名はゾーン名などのように一意である必要があります。ゾーンのノード名は、ゾーン管理者が設定できます。たとえば、大域ゾーンでゾーン名を設定します。

```
$ hostname kzone1
```

ホスト名を表示するには、`hostname` コマンドを引数なしで入力します。例:

```
$ hostname  
kzone1
```

## 非大域ゾーンでの時間値の設定

非大域ゾーンの時間値には、大域ゾーン内の値と異なる時間値を設定できます。ゾーンで時間を設定する機能は、デフォルトの `sys_time` 権限を通じて利用できます。この特権により、非大域ゾーンのプロセスは、そのゾーンの `global-time` プロパティの値 (設定されている場合) に応じて、仮想ゾーン時間またはシステム時間のどちらかを設定できます。非大域ゾーンで異なる時間を設定する機能は、大域ゾーンで行われた時間の変更から独立しているわけではありません。大域ゾーンで時間が変更された場合は、非大域ゾーンの時間が同じ量で相殺されます。

NTP (Network Time Protocol) はどのゾーンからでも実行でき、それにより、そのコマンドが実行されているゾーンだけが影響を受けます。時間が異なる非大域ゾーンを含むシステム全体で NTP を実行している場合は、大域ゾーンで NTP を実行してください。大域ゾーンで NTP を実行すると、オフセットでのみ動作している非大域ゾーンのすべてのクロックが同期されます。大域ゾーンで NTP によるクロック調整を通じた時間の変更を行うと、その効果は非大域ゾーンにも及びます。

`global-time` が `false` に設定されているゾーンの内で NTP を実行した場合は、`ntp_adjtime` および `adjtime` システムコールを使ってゾーンの時間を修正できません。`global-time` が `false` に設定されている場合、NTP は時間の設定を通じてクロックを指定された値に調整することでタイミングを合わせ、第 2 レベルの精度で時間の同期を維持します。NTP は、時間の同期を維持するためにゾーンの時間を前後に調整できます。



`global-time` が `true` に設定されているゾーンの内側で NTP を実行した場合は、`ntp_adjtime` および `adjtime` システムコールを使ってシステムクロックを調整できます。その後、ゾーンの内側で NTP を実行してシステム時間の正確な同期を維持する機能が保持されます。

`global-time` プロパティの詳細は、『Oracle Solaris ゾーン構成リソース』を参照してください。時間の設定については、[date\(1\)](#) のマニュアルページを参照してください。権限の詳細は、[71 ページの「非大域ゾーン内の特権」](#)を参照してください。`adjtime` および `ntp_adjtime` の詳細は、[adjtime\(2\)](#) および [ntp\\_adjtime\(2\)](#) のマニュアルページを参照してください。

## ファイルシステムと非大域ゾーン

このセクションでは、ゾーンがインストールされている Oracle Solaris システムでファイルシステムを使用する場合の問題について説明します。これらの問題には、NFS サーバーまたはクライアントとしてのゾーンの使用、ファイルシステムのマウントとマウント済みファイルシステムのトラバース、ゾーンに固有のファイルシステム制限などです。

各ゾーンは、ゾーンの `root` と呼ばれるディレクトリをルートとする、ファイルシステム階層の独自セクションを保持します。ゾーン内のプロセスは、ファイルシステム階層内のゾーンルートよりも下位にあるファイルにのみアクセスできます。ゾーン内で `chroot` ユーティリティを使用できますが、プロセスをゾーン内のルートパスに制限する場合だけです。`chroot` の詳細は、[Unresolved link to "chroot8"](#) のマニュアルページを参照してください。

## ゾーンでの NFS サーバーの実行

ゾーン内に NFS 共有を作成するには、そのゾーン内に NFS サーバーパッケージ `svc:/network/nfs/server:default` をインストールする必要があります。

ゾーン内での NFS 共有を防止するため、ゾーン構成で `sys_share` 特権を禁止できます。[表1](#)を参照してください。

次の制限事項があります。

- ゾーンをまたぐ LOFS マウントはゾーンから共有できません。
- ゾーン内にマウントされたファイルシステムは、大域ゾーンから共有できません。
- リモートダイレクトメモリーアクセス (RDMA) を使用した NFS は、ゾーンでサポートされません。

- Oracle Sun Cluster HA for NFS (HANFS) のフェイルオーバーは、ゾーンでサポートされません。

詳細は、『[Oracle Solaris 12 ネットワークサービスの紹介](#)』を参照してください。

## -o nosuid マウントオプション

mount ユーティリティーで -o nosuid オプションを指定する場合、次の機能を利用できます。

- nosuid オプションを使用してマウントされたファイルシステム上の setuid バイナリに基づくプロセスは、setuid バイナリの権限では動作しません。プロセスは、バイナリを実行するユーザーの権限で動作します。  
たとえば、ユーザーが root の所有する setuid バイナリを実行する場合、プロセスはそのユーザーの権限で動作します。
- ファイルシステム内のデバイス特殊エントリを開くことはできません。この動作は、nodevices オプションを指定する場合と同じです。

[mount\(8\)](#) のマニュアルページに説明されているように、このファイルシステム固有のオプションは、mount ユーティリティーでマウントできるすべての Oracle Solaris ファイルシステムで使用できます。これらのファイルシステムの一覧については、このガイドの [82 ページの「ゾーン内でのファイルシステムのマウント」](#) を参照してください。マウント機能についても説明します。-o nosuid オプションの詳細は、『[Managing Network File Systems in Oracle Solaris 11.4](#)』の第7章、「[Accessing Network File Systems](#)」を参照してください。

## ゾーン内でのファイルシステムのマウント

ファイルシステムをゾーン内部でマウントする場合、nodevices オプションが適用されます。たとえば、ゾーンに、UFS ファイルシステムに対応するブロック型デバイス (/dev/dsk/c0t0d0s7) および raw デバイス (/dev/rdisk/c0t0d0s7) へのアクセスが許可される場合、ゾーン内部からマウントを行うと、ファイルシステムのマウントで自動的に nodevices オプションが適用されます。この規則は、zonecfg 構成を使用して指定されたマウントには適用されません。

次の表で、非大域ゾーン内でファイルシステムをマウントする場合のオプションを説明します。これらの補助的なマウント方法の実行手順については、[118 ページの「ゾーンの構成」](#) および [190 ページの「実行中の非大域ゾーンでのファイルシステムのマウント」](#) を参照してください。

/usr/lib/fstype/mount 内にマウントバイナリが存在する場合、表に含まれない任意のファイルシステムタイプを構成内で指定できます。

非大域ゾーンの内部から HSFS と NFS 以外のファイルシステムタイプをマウントするには、`zonecfg fs-allowed` プロパティを使用して、構成にファイルシステムタイプも追加します。

デフォルト以外のファイルシステムマウントを許可すると、ゾーン管理者によってシステムが危険にさらされることがあります。

ファイルシステム	非大域ゾーン内のマウントオプション
AutoFS	マウントに <code>zonecfg</code> を使用できません。ゾーン内部からマウントすることは可能です。
CacheFS	非大域ゾーン内では使用できません。
FDFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
HSFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
LOFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
MNTFS	マウントに <code>zonecfg</code> を使用できません。ゾーン内部からマウントすることは可能です。
NFS	マウントに <code>zonecfg</code> を使用できません。ゾーン内で現在サポートされているバージョンである V2、V3、および V4 を、ゾーン内からマウントできます。
PCFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
PROCFS	マウントに <code>zonecfg</code> を使用できません。ゾーン内部からマウントすることは可能です。
TMPFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
UDFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
UFS	<p><code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。</p> <p><b>注記</b> - <code>quota</code> コマンド (<a href="#">quota(8)</a> のマニュアルページを参照) は、<code>zonecfg</code> コマンドの <code>fs</code> リソースを使用して追加された UFS ファイルシステムの割り当て制限情報の取得には使用できません。</p> <p><code>fs</code> リソースを使用する場合は、<code>system/file-system/ufs</code> パッケージを大域ゾーンにインストールする必要があります。<code>zonecfg</code> コマンドを使用して非大域ゾーンで UFS ファイルシステムを使用する場合は、インストール後にこのパッケージをゾーンにインストールするか、AI マニフェストスクリプトを使用してインストールする必要があります。</p> <pre>global\$ pfexec pkg -R /system/zones/zonename/root install file-system/ufs</pre>
VxFS	<code>zonecfg</code> を使用してマウントできます。ゾーン内からマウントできます。
ZFS	<code>zonecfg dataset</code> および <code>fs</code> リソースタイプを使用してマウントできます。

詳細は、[119 ページ](#)の「`zonecfg` ユーティリティを使用してゾーンを構成する方法」と [190 ページ](#)の「実行中の非大域ゾーンでのファイルシステムのマウント」、および [mount\(8\)](#) のマニュアルページを参照してください。

## ゾーン内でのファイルシステムのアンマウント

ファイルシステムをアンマウントできるかどうかは、どの管理者がそのファイルシステムを最初にマウントしたかによって決まります。`zonecfg` コマンドを使用してゾ

ンを構成するときにファイルシステムが指定されている場合は、大域ゾーンがそのマウントの所有者となるので、非大域ゾーンの管理者はそのファイルシステムをアンマウントすることはできません。非大域ゾーンの `/etc/vfstab` ファイルにマウントを指定するなどの方法でファイルシステムが非大域ゾーンからマウントされている場合には、その非大域ゾーンの管理者はそのファイルシステムをアンマウントできます。

## NFS クライアントとして機能する非大域ゾーン

ゾーンは、NFS クライアントとしても機能できます。NFS version 2、version 3、および version 4 プロトコルがサポートされています。これらの NFS バージョンについては、『[Managing Network File Systems in Oracle Solaris 11.4](#)』の「[Features of the NFS Service](#)」を参照してください。

デフォルトのバージョンは、NFS version 4 です。次のいずれかの方法を使用して、クライアント上でほかの NFS バージョンを有効にできます。

- `sharectl` コマンドを使用してプロパティを設定します - 『[Managing Network File Systems in Oracle Solaris 11.4](#)』の「[How to Select Different Versions of NFS on a Client](#)」および `sharectl(8)` のマニュアルページを参照してください。
- バージョンマウントを手動で作成します - この方法により、`sharectl` 設定がオーバーライドされます。『[Managing Network File Systems in Oracle Solaris 11.4](#)』の「[How to Select Different Versions of NFS on a Server](#)」を参照してください。

## ファイルシステムのトラバース

ゾーンのファイルシステム名前空間は、大域ゾーンからアクセス可能な名前空間の一部です。大域ゾーン内の特権のないプロセスが、非大域ゾーンのファイルシステム階層をトラバースすることはできません。これには、次のような理由があります。

- `root` だけが、ゾーンルートの親ディレクトリを所有、読み込み可能、書き込み可能、および実行可能に指定する
- `/proc` によりエクスポートされたディレクトリへのアクセスが制限される

別のゾーン用にマウントされた AutoFS ノードへのアクセスを試みても、失敗します。大域管理者は、その子孫がほかのゾーンに含まれる自動マップを保持してはいけません。

## セキュリティの制限およびファイルシステムの動作

ゾーン内部から特定のファイルシステムをマウントする場合、適用されるセキュリティの制限が存在します。ほかのファイルシステムは、ゾーン内でマウントされ

たときに特有の動作を行います。変更されたファイルシステムの一覧を、次に示します。

#### AutoFS

AutoFS は、適切なファイルシステムを自動的にマウントするクライアント側のサービスです。クライアントが現在マウントされていないファイルシステムにアクセスしようとする、AutoFS ファイルシステムはその要求に介入し、`automountd` を呼び出して要求されたディレクトリをマウントします。ゾーン内で確立された AutoFS マウントは、そのゾーンだけで有効です。大域ゾーンを含むほかのゾーンからそのマウントにアクセスすることはできません。ゾーンが停止またはリブートすると、マウントは削除されます。AutoFS の詳細は、『[Managing Network File Systems in Oracle Solaris 11.4](#)』の「[How Autofs Works](#)」を参照してください。

各ゾーンは、`automountd` の独自コピーを実行します。自動マップおよびタイムアウトは、ゾーン管理者により制御されます。大域ゾーンから非大域ゾーンの AutoFS マウントポイントを横断的に使用して、別のゾーン内でマウントをトリガーすることはできません。

別のマウントがトリガーされると、カーネル内で特定の AutoFS マウントが作成されます。この種のマウントは、一括してマウントまたはアンマウントする必要があるため、通常の `umount` インタフェースを使用して削除することはできません。この機能がゾーンのシャットダウン用であることに注意してください。

#### MNTFS

MNTFS は、ローカルシステムのマウント済みファイルシステムのテーブルに読み取り専用アクセスを提供する仮想ファイルシステムです。非大域ゾーン内から `mnttab` を使用して表示可能なファイルシステムのセットは、ゾーン内でマウントされたファイルシステムセットおよびルート (`/`) のエントリで構成されます。`/dev/rdsk/c0t0d0s0` などの、ゾーン内からアクセス不可能な特殊なデバイスを保持するマウントポイントは、マウントポイントと同じ特殊なデバイスセットを保持します。システム内のすべてのマウントが、大域ゾーンの `/etc/mnttab` テーブルから表示可能になります。MNTFS の詳細は、『[Managing Network File Systems in Oracle Solaris 11.4](#)』の「[Mounting File Systems](#)」を参照してください。

#### NFS

ゾーン内部で確立された NFS マウントは、そのゾーンでのみ有効です。大域ゾーンを含むほかのゾーンからそのマウントにアクセスすることはできません。ゾーンが停止またはリブートすると、マウントは削除されます。

ゾーン内部からの NFS マウントは、`nodevices` オプションを使用してマウントされたかのように動作します。

`nfsstat` コマンドの出力は、コマンドが実行されたゾーンにのみ関係があります。たとえば、コマンドが大域ゾーン内で実行される場合、大域ゾーンに関する情報だけが出力されます。`nfsstat` コマンドの詳細は、[nfsstat\(8\)](#) のマニュアルページを参照してください。

## PROCFS

PROCFS とも呼ばれる `/proc` ファイルシステムは、プロセスの可視性とアクセス制限、およびプロセスのゾーン関連性に関する情報を提供します。`/proc` では、同じゾーン内のプロセスだけを表示できます。

大域ゾーン内のプロセスから、非大域ゾーン内のプロセスおよびほかのオブジェクトを監視できます。これにより、この種のプロセスがシステム規模の可観測性を備えることが可能になります。

ゾーン内部からは、`procfs` マウントは `nodevices` オプションを使用してマウントされたかのように動作します。`procfs` の詳細は、[proc\(5\)](#) のマニュアルページを参照してください。

## LOFS

LOFS でマウント可能なファイルシステムの範囲は、ゾーンで表示可能なファイルシステム部分に限定されています。このため、ゾーン内での LOFS マウントには制限はありません。

## UFS、UDFS、PCFS、およびその他のストレージに基づいたファイルシステム

`zonecfg` コマンドを使用して、UFS などの `fsck` バイナリを保持するストレージに基づいたファイルシステムを構成する場合、ゾーン管理者は `raw` パラメータを指定する必要があります。このパラメータは、`/dev/rdisk/c0t0d0s7` などの `raw` (文字) デバイスを示します。`zoneadm` デーモンは、`fsck` コマンドを `preen` モード (`fsck -p`) で自動的に実行します。このコマンドは、ファイルシステムをマウントする前に、非対話的にファイルシステムを検査および修正します。`fsck` が失敗した場合、`zoneadm` を使用してゾーンを `ready` 状態にすることはできません。`raw` により指定されるパスを、相対パスにすることはできません。

`/usr/lib/fs/fstype/fsck` 内で `fsck` バイナリを提供しないファイルシステムで、デバイスを `fsck` に指定することは誤りです。また、そのファイルシステムに `fsck` バイナリが存在する場合に、デバイスを `fsck` に指定しないことも誤りです。

詳細は、[52 ページの「zoneadm ゾーン管理デーモン」](#)、および [fsck\(8\)](#) のマニュアルページを参照してください。

## ZFS

デフォルトのデータセット ([『Oracle Solaris Zones Configuration Resources』の「File Systems Mounted in Zones」](#)に記載)に加えて、`zonecfg` コマンドに `add dataset` リソースを付けて実行することにより、ZFS データセットを非大域ゾーンに追加できます。データセットは非大域ゾーンでマウントされ可視になり、大域ゾーンでも可視になります。ゾーン管理者は、そのデータセット内のファイルシステムの作成と破棄、およびデータセットのプロパティの変更を行うことができます。

`zfs` の `zoned` 属性は、データセットが非大域ゾーンに追加されたかどうかを示します。

```
$ pfexec zfs get zoned dataset
```

NAME	PROPERTY	VALUE	SOURCE
dataset	zoned	on	local

データセットリソースを使用して非大域ゾーンに委任された各データセットには、別名が付けられます。データセットのレイアウトはゾーン内では表示されません。別名が付けられた各データセットは、プールのようにゾーンに表示されません。データセットのデフォルトの別名は、データセット名の最後のコンポーネントです。たとえば、委任されたデータセット `tank/sales` にデフォルトの別名が使用された場合、ゾーンには `sales` という仮想 ZFS プールが表示されます。別名が別の値になるようにカスタマイズするには、データセットリソース内の別名プロパティを設定します。

各非大域ゾーンの `zonepath` データセット内には、`rpool` というデータセットが存在します。すべての非大域ゾーンで、ゾーンのこの `rpool` データセットは `rpool` として別名が付けられます。

```
zonename$ zfs list -o name,zoned,mounted,mountpoint
NAME                ZONED  MOUNTED  MOUNTPOINT
rpool                on     no       /rpool
rpool/ROOT           on     no       legacy
rpool/ROOT/solaris  on     yes      /
rpool/export         on     no       /export
rpool/export/home   on     no       /export/home
```

データセットの別名は、ZFS プールと同じ名前制限の対象となります。この制限については、[zpool\(8\)](#) のマニュアルページに記載されています。

大域ゾーンのデータセットを共有する場合は、`zonecfg` コマンドとともに `add fs` サブコマンドを使用して、LOFS マウントした ZFS ファイルシステムを追加できます。データセットのプロパティの設定および制御は、大域管理者または適切な承認を付与されたユーザーが担当します。

ZFS の詳細は、『[Oracle Solaris 11.4 での ZFS ファイルシステムの管理](#)』の第 10 章、「[Oracle Solaris ZFS の高度なトピック](#)」を参照してください。

## ゾーン内で `mknod` を使用する際の制限

`mknod` コマンドを使用して、非大域ゾーン内で特殊ファイルを作成することはできません。

## 大域ゾーンから非大域ゾーンにアクセスする際の制限

非大域ゾーンのインストール後に、システムのバックアップユーティリティ以外のコマンドを使用して、大域ゾーンからそのゾーンに直接アクセスしてはいけません。また、非大域ゾーンを未知の環境に公開した後は、そのゾーンがセキュアであると考えすることはできません。たとえば、公開されたネットワーク上に配置されたゾーンについて考えてみましょう。この場合、ゾーンのセキュリティが低下し、ファイルシ

ステムの内容が変更される可能性があります。セキュリティーが低下した可能性がある場合、大域管理者はゾーンを信頼できないものとして扱うべきです。

次の両方の条件が当てはまる場合、`-R` オプションまたは `-b` オプション (またはこれと同等なオプション) を使って代替ルートを指定可能なコマンドを使用してはいけません。

- コマンドが大域ゾーン内で実行される。
- 代替ルートが非大域ゾーン内のいずれかのパスを参照している。現在動作中のシステムの非大域ゾーンからの相対パスの場合や、代替ルート内の大域ゾーンからの相対パスの場合を含む。

たとえば、`pkgadd -R root-path` コマンドを、非大域ゾーンのルートパスを保持する大域ゾーンから実行する場合はこの条件に該当します。

- `-R` を代替ルートパスとともに使用するコマンドには、`auditreduce`、`metaroot`、`pkg`、`syseventadm` などがあります。
- `-b` を代替ルートパスとともに使用するコマンドには、`add_drv` や `useradd` などがあります。

## 排他的 IP 非大域ゾーンにおけるネットワーク

排他的 IP ゾーン内のネットワークは、大域ゾーン内のネットワークと似ています。排他的 IP ネットワークの管理はそのゾーン内で行われるものもあれば、大域ゾーンから行われるものもあります。

## 排他的 IP ゾーンにおけるネットワークの概要

排他的 IP ゾーンは、IP 関連の状態を独自に保持します。ゾーンの構成時に、専用のデータリンクセットがゾーンに割り当てられます。

パケットは、物理リンクに送信されます。その後、Ethernet スイッチや IP ルーターなどのデバイスがパケットを宛先へ転送します。この送信先は、送信元と同じシステム上の別のゾーンである可能性もあります。

仮想リンクの場合、パケットは最初に仮想スイッチに送信されます。宛先のリンクが同じデバイス (同じ物理リンクまたは `etherstub` 上の VNIC など) を経由する場合、パケットは宛先の VNIC に直接送信されます。それ以外の場合、パケットは VNIC のベースとなる物理リンクに送信されます。

排他的 IP 非大域ゾーンで使用できる機能については、『[Oracle Solaris Zones Configuration Resources](#)』の「[Exclusive-IP Non-Global Zones](#)」を参照してください。



## 排他的 IP ゾーンの区分化

排他的 IP ゾーンはそれぞれ個別の TCP/IP スタックを持っているため、下位のデータリンク層まで分離されます。排他的 IP ゾーンには、1 つ以上のデータリンク名が大域管理者によって割り当てられます。データリンク名は、NIC または NIC 上の VLAN の場合があります。ゾーン管理者は、大域ゾーンの場合と同じ柔軟性とオプションで、これらのデータリンクの IP を構成できます。

## 排他的 IP データリンクインタフェース

1 つのデータリンク名は、1 つのゾーンだけに割り当てる必要があります。

`dladm show-link` コマンドを使用すると、次の例のように、実行中のゾーンに割り当てられているデータリンクを表示できます。

```
sol1-t2000-10{jdoe}1: dladm show-link
LINK          CLASS    MTU    STATE  OVER
vsw0          phys    1500  up     --
net0          phys    1500  up     --
netg2         phys    1500  up     --
netg1         phys    1500  up     --
netg3         phys    1500  up     --
zoneA/net0    vnic    1500  up     net0
zoneB/net0    vnic    1500  up     net0
aggr1         aggr    1500  up     net2 net3
vnic0         vnic    1500  up     net1
zoneA/vnic0   vnic    1500  up     net1
vnic1         vnic    1500  up     net1
zoneB/vnic1   vnic    1500  up     net1
vnic3         vnic    1500  up     aggr1
vnic4         vnic    1500  up     aggr1
zoneB/vnic4   vnic    1500  up     aggr1
```

詳細は、[dladm\(8\)](#) のマニュアルページを参照してください。

## 同一システム上の排他的 IP ゾーン間の IP トラフィック

排他的 IP ゾーン間での IP パケットの内部ループバックはありません。すべてのパケットが下位のデータリンクまで送信されます。これは通常、パケットがネットワークインタフェース上に送信されることを意味します。その後、Ethernet スイッチや IP ルーターなどのデバイスがパケットを宛先へ転送します。この送信先は、送信元と同じシステム上の別のゾーンである可能性もあります。

## 排他的 IP ゾーンにおけるパケットフィルタファイアウォール

排他的 IP ゾーンでは、大域ゾーンのものと同じパケットフィルタ (PF) 機能を使用できます。排他的 IP ゾーンでの PF の構成方法も大域ゾーンと同じです。

## 排他的 IP ゾーン内の IP ネットワークマルチパス

IP ネットワークマルチパス (IPMP) は、同一の IP リンク上に複数のインタフェースを保持するシステムで、物理インタフェースの障害検出および透過的なネットワークアクセスフェイルオーバーを提供します。IPMP も、複数のインタフェースを保持するシステムについて、パケットの負荷分散を提供します。

データリンク構成は、大域ゾーン内で行われます。最初に、`zonecfg` を使用して、複数のデータリンクインタフェースをゾーンに割り当てます。複数のデータリンクインタフェースを同じ IP サブネットに接続する必要があります。その後、排他的 IP ゾーン内からゾーン管理者が IPMP を構成することができます。

## 共有 IP 非大域ゾーンにおけるネットワーク

Oracle Solaris システムにインストールしたゾーンは、ネットワーク経由で互いに通信できます。ゾーンは別個の結合または接続を保持します。また、独自のサーバーデーモンを実行できます。これらのデーモンは、同一のポート番号で競合することなく待機できます。IP スタックは、着信接続の IP アドレスを考慮に入れることで競合を解決します。IP アドレスにより、ゾーンが識別されます。

## 共有 IP ゾーンの区分化

共有 IP はデフォルトではありませんが、このタイプはサポートされます。

ゾーンをサポートするシステム内の IP スタックは、ゾーン間のネットワークトラフィックの分離を実装します。IP トラフィックを受け取るアプリケーションは、同じゾーンに送信されたトラフィックの受信だけを実行できます。

システム上の各論理インタフェースは、特定のゾーンに所属します。デフォルトは、大域ゾーンです。`zonecfg` ユーティリティ経由でゾーンに割り当てられた論理ネットワークインタフェースは、ネットワーク経由での通信に使用されます。各ストリームおよび接続は、それを開いたプロセスのゾーンに所属します。

上位層ストリームと論理インタフェース間の結合は、制限されます。ストリームが確立できるのは、同一ゾーン内の論理インタフェースへの結合だけです。同様に、論理インタフェースからのパケットを渡すことができるのは、論理インタフェースと同じゾーン内の上位層ストリームに対してだけです。

各ゾーンは、独自のバインドセットを保持します。アドレスが使用中であるため、各ゾーンは、同一のポート番号で待機する同じアプリケーションを、バインドが失敗することなく稼働可能です。各ゾーンでは、次のような各種ネットワークサービスの固有のバージョンを実行できます。

- 完全な構成ファイルを保持するインターネットサービスデーモン ([inetd\(1M\)](#)) のマニュアルページを参照してください)
- [sendmail\(sendmail\(1M\)\)](#) のマニュアルページを参照)
- [apache](#)

大域ゾーン以外のゾーンは、ネットワークへのアクセスが制限されています。標準の TCP および UDP ソケットインタフェースが利用可能ですが、[SOCK\\_RAW](#) ソケットインタフェースは [ICMP \(Internet Control Message Protocol\)](#) に制限されています。ICMP は、ネットワークのエラー状況を検出および報告したり、[ping](#) コマンドを使用するのに必要です。

## 共有 IP ネットワークインタフェース

ネットワーク接続を必要とする非大域ゾーンには、それぞれ 1 つ以上の専用 IP アドレスがあります。これらのアドレスは、ゾーン内に配置可能な論理ネットワークインタフェースに関連付けられています。[zonecfg](#) により構成されるゾーンネットワークインタフェースは、ブート時に自動的に設定されてゾーン内に配置されます。[ipadm](#) コマンドを使用すると、ゾーンの稼働中に論理インタフェースを追加または削除できます。インタフェース構成およびネットワーク経路を変更できるのは、大域管理者または適切な承認を付与されたユーザーのみです。

非大域ゾーン内では、そのゾーンのインタフェースのみが [ipadm](#) コマンドで表示されます。

詳細は、[ipadm\(1M\)](#) および [if\\_tcp\(4P\)](#) のマニュアルページを参照してください。

## 同一システム上の共有 IP ゾーン間の IP トラフィック

指定された IP 宛先への使用可能な経路がルーティングテーブルにある場合、共有 IP ゾーンはその宛先に到達できます。ルーティングテーブルを表示するには、ゾーン内から [netstat](#) コマンドに [-r](#) オプションを付けて実行します。IP 転送ルールは、別のゾーンまたは別のシステムの IP 宛先でも同じです。

## 共有 IP ゾーンでの Oracle Solaris パケットフィルタ

Oracle Solaris のパケットフィルタ (PF) 機能は、着信パケットを取り込み、それをシステムへの入力およびシステムからの出力のために評価するネットワークファイアウォールです。PF には、ステートフルパケット検査が用意されています。IP アドレスやポート番号だけでなく、受信しているネットワークインタフェースごとにパケットを一致させることができます。PF は OpenBSD パケットフィルタ (PF) バージョン 5.5 に基づいています。詳細は、『[Securing the Network in Oracle Solaris 11.4](#)』の第 4 章、「[Oracle Solaris Firewall](#)」を参照してください。

『[Securing the Network in Oracle Solaris 11.4](#)』の第 5 章、「[Configuring the Firewall in Oracle Solaris](#)」で説明しているようにループバックフィルタリングをオンにすることによって、非大域ゾーンで PF を有効にできます。

## 共有 IP ゾーン内の IP ネットワークマルチパス

Oracle Solaris では、IP ネットワークマルチパス (IPMP) によって複数のインタフェースを同一の IP リンク上でグループ化することにより、継続的なネットワーク可用性が確保されます。これらのベースとなるインタフェースは、そのいずれかに障害が発生した場合でもネットワークを引き続き利用できるように互いにバックアップします。IPMP も、複数のインタフェースを保持するシステムについて、パケットの負荷分散を提供します。

IPMP は、次の方法で Oracle Solaris に実装されます。

- net0、net1、net2 などの複数のインタフェースから、IPMP インタフェース `ipmp0` が構成されます。
- IPMP インタフェース `ipmp0` は、データアドレスと呼ばれる複数の IP アドレスで構成されます。これらのアドレスは、ネットワークトラフィックのホストに使用されます。
- IP アドレスは、ベースとなるインタフェース `netN` 上に直接構成することもできます。これらのアドレスは、ネットワークトラフィックには使用されませんが、ベースとなるインタフェースに障害が発生しているかどうかを判断する障害検出に使用されます。したがって、ベースとなるインタフェース上のこれらの IP アドレスは検査用アドレスと呼ばれます。

データアドレスは `ipmp0` に存在するので、ベースとなる IP インタフェースの 1 つに障害が発生した場合でも、ネットワークを引き続き利用できます。トラフィックは、`ipmp0` の別のアドレスを使用して引き続き送られます。

すべてのネットワーク構成タスクと同様に、IPMP は大域ゾーンで構成します。そのあとで非大域ゾーンに機能を拡張します。機能の拡張は、IPMP インタフェースのデータアドレスの 1 つをゾーンに割り当てることによって行います。

指定された非大域ゾーンで `ipadm` コマンドを使用すると、ゾーンに関連するインタフェースだけが表示されます。

詳細は、次を確認してください。

- [199 ページの「IP ネットワークマルチパス機能を共有 IP 非大域ゾーンに拡張する方法」](#)
- 『[Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.4](#)』の第2章, 「[About IPMP Administration](#)」
- [119 ページの「zonecfg ユーティリティを使用してゾーンを構成する方法」](#)

## 非大域ゾーンでのデバイスの使用

あるゾーンのプロセスが別のゾーンで実行中のプロセスに干渉することがないように、ゾーン内部で利用可能なデバイスセットには制限が課されています。たとえば、ゾーン内のプロセスが、カーネルメモリーおよびルートディスクの内容を変更することはできません。このため、デフォルトでは、特定の擬似デバイスのみがゾーン内で使用しても安全であるとみなされます。zonecfg ユーティリティを使用すると、利用可能なデバイスを特定のゾーンに追加できます。

### /dev および /devices 名前空間

devfs(4FS) のマニュアルページに記載された devfs ファイルシステムは、Oracle Solaris システムで /devices を管理するために使用されます。この名前空間内の各要素は、ハードウェアデバイス、仮想デバイス、またはネクサスデバイスへの物理パスを表します。名前空間には、デバイスツリーが反映されます。したがって、ファイルシステムは、ディレクトリおよびデバイス特殊ファイルの階層により生成されます。

デバイスは相対 /dev 階層に従ってグループ化されます。たとえば、大域ゾーンの /dev の下のすべてのデバイスは、大域ゾーンのデバイスとしてグループ化されます。非大域ゾーンの場合、デバイスはゾーンのルートパスの下の /dev ディレクトリでグループ化されます。各グループは、/dev ディレクトリの下にマウントされた /dev ファイルシステムのインスタンスです。したがって、大域ゾーンのデバイスは /dev の下にマウントされ、my-zone という非大域ゾーンのデバイスは /my-zone/root/dev の下にマウントされます。

/dev ファイル階層は、dev ファイルシステムによって管理されます。



**注意** - /devices パス名に依存するサブシステムは、非大域ゾーンで実行できません。/dev パス名を使用するようにサブシステムを更新する必要があります。



**注意** - 非大域ゾーンに `/dev/zvol` 内のデバイスと一致するデバイスリソースがある場合、非大域ゾーン内で名前空間の競合が発生する可能性があります。詳細は、[dev\(4FS\)](#) のマニュアルページを参照してください。

## 排他使用のデバイス

デバイスを特定のゾーンに割り当てるが必要な場合があります。特権のないユーザーがブロックデバイスにアクセスできるようにすると、これらのデバイスの使用が許可されて、システムパニックやバスリセットなどの不具合が生じる場合があります。この種の割り当てを行う前に、次の点を考慮してください。

- SCSI テープデバイスを特定のゾーンに割り当てる前に、[sgen\(4D\)](#) のマニュアルページを参照してください。
- 物理デバイスを複数のゾーンに配置する場合、ゾーン間に隠れたチャンネルが作成される場合があります。大域ゾーンアプリケーションでこの種のデバイスを使用すると、非大域ゾーンによるデータ整合性の損失や、データの破壊が発生する危険があります。

## デバイスドライバの管理

非大域ゾーンで、[modinfo\(8\)](#) のマニュアルページに記載された `modinfo` コマンドを使用すると、ロードされているカーネルモジュールの一覧を確認できます。

カーネル、デバイス、およびプラットフォームの管理に関係した大半の操作は、非大域ゾーンの内部では機能しません。これは、プラットフォームハードウェア構成を変更すると、ゾーンのセキュリティーモデルに違反するためです。これらの操作には、次のことが含まれます。

- ドライバの追加および削除
- カーネルモジュールの明示的な読み込みおよび読み込み解除
- 動的再構成 (DR) 操作の開始
- 物理プラットフォームの状態に影響を与える機能の使用

## 非大域ゾーンで動作しないか、変更されるユーティリティー

ユーティリティーには、非大域ゾーン内で動作できないもの、ゾーン内での使用に合わせて変更できるもの、使用時にセキュリティーを考慮する必要があるものなどがあります。

## 非大域ゾーンで動作しないユーティリティー

次のユーティリティーは、通常は使用できないデバイスに依存しているため、ゾーン内では動作しません。

- `add_drv` ([add\\_drv\(8\)](#) のマニュアルページを参照)
- `disks` ([Unresolved link to " disks8"](#) のマニュアルページを参照)
- `prtconf` ([prtconf\(8\)](#) のマニュアルページを参照)
- `prtdiag` ([prtdiag\(8\)](#) のマニュアルページを参照)
- `rem_drv` ([rem\\_drv\(8\)](#) のマニュアルページを参照)

## SPARC: 非大域ゾーンでの使用に合わせて変更されたユーティリティー

`eeprom` ユーティリティーをゾーン内で使用すると、設定を表示できますが、設定を変更することはできません。詳細は、[eeprom\(8\)](#) および [openprom\(4D\)](#) のマニュアルページを参照してください。

## セキュリティーを考慮することにより許可されるユーティリティー

`allowed-raw-io` が有効にされている場合、次のユーティリティーをゾーンで使用できます。セキュリティーに関する考慮事項を評価する必要があります。デバイスを追加する場合、制限およびセキュリティーの懸念事項については、[93 ページの「非大域ゾーンでのデバイスの使用」](#)、[95 ページの「非大域ゾーンでのアプリケーションの実行」](#)、および [71 ページの「非大域ゾーン内の特権」](#) を参照してください。

- `cdrecord` ([cdrecord\(1\)](#) のマニュアルページを参照)。
- `cdrw` ([cdrw\(1\)](#) のマニュアルページを参照)。
- `rmformat` ([rmformat\(1\)](#) のマニュアルページを参照)。

## 非大域ゾーンでのアプリケーションの実行

通常は、すべてのアプリケーションを非大域ゾーンで実行できます。ただし、次のタイプのアプリケーションは、この環境に適さない場合があります。

- システム全体に影響を与える特権付きの操作を実行するアプリケーション。大域システムクロックを設定したり、物理メモリーをロックダウンする操作が、これに該当します。

- /dev/kmem などの、非大域ゾーン内に存在しない特定のデバイスに依存するいくつかのアプリケーション。

## 非大域ゾーンでのリソース管理

このセクションでは、リソース制御、公平配分スケジューラ (FSS)、および拡張アカウントティングについて説明します。

### 非大域ゾーンで使用されるリソース制御

ゾーン内でのリソース管理機能の使用に関する追加情報については、『[Oracle Solaris 12 でのリソースの管理](#)』のこの機能について説明した章を参照してください。

リソース管理の章に記載されたリソース制御および属性はすべて、大域ゾーンおよび非大域ゾーンの /etc/project ファイル、NIS マップ、または LDAP ディレクトリサービスで設定できます。指定されたゾーンの設定は、そのゾーンにのみ影響を及ぼします。異なるゾーン内で自動実行中のプロジェクトは、ゾーンごとに別個の制御セットを保持できます。たとえば、大域ゾーン内のプロジェクト A を `project.cpu-shares=10` に設定し、非大域ゾーン内の Project A を `project.cpu-shares=5` に設定できます。それぞれが該当するゾーン内でのみ機能する、rcapd のインスタンスをシステム上で複数実行できます。

ゾーン内部のプロジェクト、タスク、およびプロセスを制御するため、ゾーン内で使用するリソースの制御および属性は、プールおよびゾーン規模のリソース制御に関する追加要件に従います。

1 つの非大域ゾーンにはリソースプールを 1 つだけ関連付けることができますが、プールは特定のゾーンに排他的に割り当てる必要はありません。1 つのプール内のリソースを複数の非大域ゾーンが共有してもかまいません。ただし、十分な特権を付与されたプロセスを使って、大域ゾーン内のプロセスを任意のプールにバインドすることが可能です。リソースコントローラ `poolld` は、大域ゾーン内だけで動作します。大域ゾーン内には、リソースコントローラが動作するプールが複数存在します。`poolstat` ユーティリティを非大域ゾーンで実行すると、そのゾーンに関連付けられているプールの情報だけが表示されます。非大域ゾーンで引数なしで `pooladm` コマンドを実行すると、そのゾーンに関連付けられているプールの情報だけが表示されます。

ゾーン規模のリソース制御が `project` ファイルで設定されている場合、そのリソース制御は有効にはなりません。ゾーン規模のリソース制御は、`zonecfg` ユーティリティを使って設定されます。



## ゾーンがインストールされているシステムでの公平配分スケジューラ

このセクションでは、ゾーンで公平配分スケジューラ (FSS) を使用して CPU 使用量のバランスを取る方法について説明します。

### 大域ゾーンまたは非大域ゾーン内の FSS 配分分割

ゾーンの FSS CPU 配分は、階層的で、大域ゾーンおよび非大域ゾーンの配分は、ゾーン規模のリソース制御 `zone.cpu-shares` を使って大域管理者が設定します。次に、そのゾーン内のプロジェクトごとにリソース制御 `project.cpu-shares` を定義して、ゾーン規模の制御で設定された配分をさらに分割できます。

`zonectfg` コマンドを使用してゾーンに配分を割り当てる方法については、[131 ページの「大域ゾーンの CPU シェアを設定する方法」](#)を参照してください。`project.cpu-shares` の詳細は、『[Administering Resource Management in Oracle Solaris 11.4](#)』の「[Available Resource Controls](#)」を参照してください。配分を一時的に設定する方法を示す手順例については、[201 ページの「ゾーンがインストールされているシステムでの公平配分スケジューラの使用」](#)も参照してください。

### ゾーン間の配分均衡

`zone.cpu-shares` を使用して、FSS 配分を大域ゾーンと非大域ゾーンに割り当てることができます。FSS がシステムのデフォルトのスケジューラになっている場合、配分が割り当てられていないときは、各ゾーンにはデフォルトで 1 つの配分が付与されます。システムに 1 つの非大域ゾーンが存在し、`zone.cpu-shares` を使ってこのゾーンに 2 つの配分を付与する場合、これにより非大域ゾーンが大域ゾーンとの関連で受ける CPU の比率が定義されます。2 つのゾーン間の CPU 比率は 2:1 です。

## ゾーンがインストールされているシステムでの拡張アカウンティング

拡張アカウンティングサブシステムを大域ゾーンで実行した場合、非大域ゾーンを含むシステム全体の情報が収集および報告されます。大域管理者は、ゾーンごとのリソース消費を決定することもできます。

拡張アカウンティングサブシステムを使用すると、プロセスおよびタスクに基づいたアカウンティングに対応した異なるアカウンティング設定およびファイルを、ゾーン

単位で指定することが可能になります。exacct レコードに、プロセスの場合はゾーン名 EXD PROC ZONENAME のタグを付け、タスクの場合はゾーン名 EXD TASK ZONENAME のタグを付けることができます。アカウントングレコードは、大域ゾーンのアカウンティングファイルおよびゾーン単位のアカウンティングファイルに書き込まれます。EXD TASK HOSTNAME、EXD PROC HOSTNAME、および EXD HOSTNAME レコードには、大域ゾーンのノード名の代わりに、プロセスまたはタスクが実行されたゾーンの `uname -n` 値が含まれます。

## 非大域ゾーンの移行および変換

移行とは、ゾーンを新しい場所に移動させることです。変換とは、広く配備することのできるアーカイブにゾーンを変換することです。Oracle Solaris には、ゾーンの変換を支援する `zonep2vchk` ツールが用意されています。

## solaris ゾーンのコールド移行について

ゾーンの移行は、あるホストシステムから別のホストシステムに既存のゾーン構成とデータを転送するプロセスです。共有ストレージ上のゾーンに対してのみゾーンの移行を実行できます。ゾーンの移行には、コールド、ウォーム、またはライブを指定できます。ただし、solaris ゾーンはホストのカーネルに依存しているため、新しいホストへの移行中にゾーンを実行することはできません。したがって、solaris ゾーンの移行ではウォーム移行とライブ移行はサポートされていません。

コールド移行は、ゾーンの移行中にソースホストでゾーンが動作していないときに生じます。コールド移行中は、ゾーンがストレージから切り離され、少量のデータが新しいホストに移動されます。ゾーンが新しいホストに接続されると、共有ストレージがゾーンに接続され、同じパスを使用してゾーンデータがアクセスされます。コールド移行には、ゾーンを切り離してから接続するまでの時間とほぼ同じ時間がかかります。

共有ストレージの詳細は、[第14章「共有ストレージ上の Oracle Solaris ゾーン」](#)を参照してください。ゾーンの移行およびゾーンの移行タイプの詳細は、『[Introduction to Oracle Solaris Zones](#)』の「[About Zone Migration](#)」を参照してください。

## ゾーン移行の構成の要件

移行前に、構成とホストの要件、およびパッケージの更新を考慮する必要があります。

- **ゾーンの状態** - 移行を開始するときは、ゾーンの状態が `installed` になっている必要があります。移行後も、新しいホスト上でのゾーンの状態は `installed` のままです。
- **共有ストレージ** - ゾーンが共有ストレージ上にない場合は、ゾーンの変換と同様に、統合アーカイブを使用してゾーンを移動できます。103 ページの「[ゾーンの変換について](#)」を参照してください。
- **ユーザーの承認** - ゾーン移行プロファイルまたはゾーン構成プロファイルを持つユーザーまたは役割が移行を行う必要があります。root 以外の特定のユーザーが移行を行うことを承認できます。詳細は、171 ページの「[非大域ゾーンの移行を実行するための root 以外のユーザーの承認](#)」を参照してください。

**ゾーンのリソース** – `solaris` ゾーンの構成で、次の移行されるリソースの要件が満たされている必要があります。

**device** リソースタイプ

`device` リソースはストレージ URI で指定する必要があります。match プロパティは設定できません。

**fs、dataset** リソースタイプ

`fs` リソースと `dataset` リソースはローカルファイルシステムまたはローカルデバイスを参照するため、コールド移行では使用できません。

**npiv:over-hba** リソースタイププロパティ

`npiv:over-hba` プロパティは、ゾーンがターゲットシステム上にある場合のみ設定できます。これは物理コントローラ番号と結び付けられ、ソースシステムとターゲットシステムの間で整合性が保たれない可能性があるため、移植可能な設定ではありません。ただし、移行前にゾーン構成がターゲットシステム上にある場合は、コントローラ番号が矛盾しないようにシステムを設定してあると考えられるため、移行は許可されるべきです。

**rootzpool、zpool** リソースタイプ

`rootzpool` リソースと `zpool` リソースは、`iscsi` および `lu` の URI を使用して共有ストレージリソースとともに指定する必要があります。

## zoneadm migrate コマンド

`zoneadm` コマンドは、移行用の `migrate` サブコマンドとともに使用します。`solaris` ゾーンでのこのコマンドのフォーマットは次のとおりです。

```
zoneadm -z zonename migrate [-nq] [-t auto] [-u | -U] [-z ZBE] \ [-x destroy-orphan-zbes |
force-zbe-clone | deny-zbe-clone | attach-last-booted-zbe | attach-matched-zbe | attach-
last-mounted-zbe] ] \ ssh|rads|radg://user@host:port
```

`zoneadm migrate` コマンドのオプションには次のものがあります。

-n

移行の非実行の予行演習を行います。この予行演習では、両方のシステムから共有ストレージリソースにアクセスできることを確認します。

ゾーン構成にターゲットホストとの互換性があるかどうかはチェックされません。必要に応じて、ゾーンの移行後、ゾーンをブートする前にターゲットホストで構成を調整できます。

-q

非出力モード。移行操作中にステータスが報告されないように指定します。

-t auto

実行する移行のタイプを指定します。solaris ゾーンの場合は、auto 値(コールド移行の指定)のみがサポートされています。追加のオプションがブランド固有のオプションとして使用できる場合があります。

-u | -U

-u オプションと -U オプションは相互に排他的です。

- -u オプションは、接続後に大域ゾーンと一致するように、移行されたゾーン内で必要な最低数のパッケージを更新します。更新の試行中に新しいゾーンブート環境 (ZBE) が作成されることがあります。
- -U オプションは、接続後に大域ゾーンと一致するように、移行されたゾーン内のすべてのパッケージを更新します。更新の試行中に新しい ZBE が作成されることがあります。

-x extended-option

ゾーンをターゲットホストに接続するときに実行する拡張オプションを指定します。使用可能な拡張オプションは次のとおりです。

**attach-last-booted-zbe**

最後にブートされた ZBE を接続します。

**attach-last-mounted-zbe**

最新の変更を含む、ターゲットホストの zonepath に直前にマウントされた ZBE を接続します。これは、-z ZBE 選択オプションまたは -x オプションがどちらも指定されなかった場合のデフォルトの動作です。

**attach-matched-zbe**

ターゲットホスト上の現在の大域ゾーン BE と一致する ZBE を接続します。選択された ZBE が現在の大域 BE とは異なる大域 BE に関連付けられている場合は、その ZBE のクローンが作成され、接続されます。-x オプションを使用すると、ZBE の接続時の ZBE クローン処理用に次のいずれかのオプションを指定できます。

**deny-zbe-clone**

選択されたゾーンブート環境のクローニングをオーバーライドします。このオプションでは、デフォルトの動作が ZBE のクローニングである場合に、クローニングは行われず、選択された ZBE がゾーンに強制的に接続されます。それ以外の場合は無効です。

**solaris** ゾーンの接続中に `-x deny-zbe-clone` オプションを使用した場合、クローニングは行われず、選択されたゾーンブート環境に所定の更新が行われて、アクティブなブート環境としてマウントされます。この ZBE は孤立したブート環境になることがあります。

**destroy-orphan-zbes**

どの大域ゾーンにも関連付けられていないゾーンブート環境をすべて破棄します。

**force-zbe-clone**

選択されたゾーンブート環境のクローニングを強制的に行います。その後、新しくクローニングされたブート環境が、ゾーンへの接続対象として選択されます。

**solaris** ゾーンの接続中に `-x force-zbe-clone` オプションを使用した場合、選択されたゾーンブート環境がクローニングされ、そのクローンがアクティブなブート環境としてマウントされます。この ZBE は孤立したブート環境になることがあります。

ZBE の接続とクローニングの詳細は、[60 ページの「孤立したゾーンブート環境とクローンについて」](#)を参照してください。

**-z ZBE**

接続および更新の対象となる特定のゾーンブート環境の名前を指定します。指定された ZBE が別の大域ゾーンに関連付けられている場合は、指定された ZBE がクローニングされ、クローニングされた ZBE が接続されます。

接続する ZBE が指定されない場合は、現在の大域 ZBE に関係なく、`zonpath` に直近にマウントされた ZBE が接続されます。

`-z ZBE` オプションの代わりに `-x` オプションを使用して、ZBE の接続動作を指定することもできます。

**ssh | rads | radg://user@host:port**

ゾーンをターゲットホストに移行するために使用するスキーム、ユーザー名、およびホスト名を含む RAD URI を指定します。ssh スキームは Secure Shell を使用し、rads スキームは TLS を使用します。radg スキームは Generic Security Services API (GSS-API) を使用します。RAD クライアントおよびターゲットホストが Kerberos 用に構成されている場合は、`radg` を指定します。

ホスト名のみを指定した場合、スキームのデフォルト値は `rads` になり、`user` のデフォルト値は現在のユーザーになり、`port` のデフォルト値は標準 RAD ポート 12302 になります。

詳細は、『[Remote Administration Daemon Client User's Guide](#)』の「[Connecting in Python to a RAD Instance by Using a URI](#)」を参照してください。

zoneadm migrate コマンドの詳細は、[zoneadm\(8\)](#) のマニュアルページを参照してください。

## ゾーンの移行とさまざまなホスト構成

ゾーン移行のソースホストとターゲットホストに関する次のガイドラインを確認してください。

- ゾーンを同一のシステムに移行しているときに、すべてのストレージ参照で両方のホストからアクセス可能なストレージ URI が使用される場合、移行された構成はそのまま互換性が確保されるべきです。ターゲットシステムが同一でない場合は、移行後に、新しいシステムに合うようにターゲットシステムの構成を変更する必要があります。
- ゾーンストレージがローカルの場合、zoneadm migrate コマンドを使用できません。代わりに、次のいずれかを実行できます。
  - ローカルストレージデバイスがブートに必要な場合は、それらをゾーン構成から削除したあと、zoneadm migrate を使用します。
  - 『[Oracle Solaris ゾーン作成と使用](#)』の「[共有ストレージ構成にゾーンを移動する方法](#)」の説明に従ってストレージを共有ストレージに変換したあと、zoneadm migrate を使用します。
  - [176 ページ](#)の「[共有ストレージを使用していない非大域ゾーンのアーカイブと移動](#)」の説明に従って、ゾーンを移行するのではなくアーカイブして、新しいシステムに配備しなおします。
- 移行前にターゲットホストでゾーンを構成する場合は、ターゲットホストでのゾーン構成のバージョンがゾーンのブートに使用されます。
- 移行前にターゲットホストでゾーンを構成しない場合は、ゾーン構成がソースホストからエクスポートされて、ターゲットホストにインポートされます。移行を行うユーザーには、ターゲットホストでゾーン構成を作成するために、ゾーン構成の権利プロファイルと solaris.zone.config の承認が必要です。詳細は、[171 ページ](#)の「[非大域ゾーンの移行を実行するための root 以外のユーザーの承認](#)」を参照してください。

## ゾーンの移行とパッケージ更新

solaris ゾーンには、それが attached 状態であっても、常にそのストレージがマップされ、zpool がインポートされます。これは、大域ゾーンが更新されたときにゾーン内のパッケージを更新できるようにするために必要です。移行中に solaris ゾーンがターゲットシステムに接続されると、ゾーン内のパッケージが大域ゾーンに照らし合

わせて検証され、パッケージレベルが異なる場合は、`-u` または `-U` 更新オプションを使用してゾーンの更新を試みるよう指定していないかぎり、接続に失敗します。

`-u` または `-U` 更新オプションを使用した移行では、大域ゾーンと一致するように `solaris` ゾーン内のパッケージの更新を試みます。その試行中に新しい **ZBE** が作成されることがあります。

`zoneadm migrate` に `-x attach-*-zbe` オプションが指定されていない場合、宛先システムでの接続によって、直近にマウントされた ZBE がクローニングの対象として選択されます。その後、クローニングされた ZBE が接続されます。

## ゾーンの変換について

ゾーンの変換は、Oracle Solaris 大域ゾーンまたは非大域ゾーンのアーカイブを作成し、それを Oracle Solaris 大域ゾーンまたは非大域ゾーンとして配備するプロセスです。ソースゾーンのタイプ (大域または非大域) は、配備されるゾーンタイプ (大域または非大域) と同じ場合と異なる場合があります。

実行できるゾーンの変換には次の種類があります。

- **大域ゾーンから非大域ゾーン**。物理から仮想への変換 (P2V 変換) と呼ばれます。
- **非大域ゾーンから大域ゾーン**。仮想から物理への変換と呼ばれます。
- **非大域ゾーンから非大域ゾーン**。仮想から仮想への変換 (V2V 変換) と呼ばれます。この変換は、共有ストレージを使用していないために、`zoneadm migrate` コマンドで移行できないゾーンの移行に役立ちます。

Oracle Solaris では、いくつかの仮想化テクノロジーをサポートしています。たとえば、論理ドメインとゾーンはどちらも仮想 Oracle Solaris インスタンスです。論理ドメインからゾーンへの変換は、V2V 変換とみなすことができます。ただし、論理ドメインは大域ゾーンで動作しているため、大域から非大域への変換 (P2V 変換) でもあります。このため、Oracle Solaris 11 でのゾーンの変換のモデルは、物理対仮想ではなく、大域対非大域の観点から説明されます。

Oracle Solaris 11.4 リリースでは、ゾーン間の変換に統合アーカイブのみを使用できません。統合アーカイブの詳細は、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』を参照してください。

ゾーンの変換を実装するには、変換対象の Oracle Solaris インスタンスをアーカイブし、そのアーカイブを新しいゾーンまたはシステムに配備します。

Oracle Solaris 11.2 より前の Oracle Solaris 11 リリースが動作しているホストシステムでは、こうした以前のリリースに記載されているように、ゾーンの変換にレガシーアーカイブを使用する必要があります。

## Oracle Solaris 統合アーカイブを使用したゾーンの変換

統合アーカイブファイルには、ゾーン構成とゾーンデータの両方が含まれています。つまり、ターゲットシステムで、ゾーンをアーカイブから構成およびインストールできます。zonecfg コマンドを使用すると、統合アーカイブファイルから直接新しいゾーンを構成でき、zoneadm コマンドを使用すると、統合アーカイブファイルから直接新しいゾーンをインストールできます。システムおよびゾーンのクローニングや回復のための使用法を含む統合アーカイブの詳細は、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』を参照してください。

ゾーンの変換の全体的なプロセスは次のとおりです。

1. ソースホストと、新しいゾーンが置かれるターゲットホストの両方が要件を満たしていることを確認します。詳細は、[104 ページの「ソースホストとターゲットホストでのゾーンの変換の要件」](#)を参照してください。
2. 大域ゾーンから非大域ゾーンに変換する場合は、zonep2vchk ツールを使用して、変換されたゾーンで障害を引き起こす可能性のある問題をすべて特定します。詳細は、[105 ページの「zonep2vchk ツールを使用した大域から非大域への変換準備」](#)を参照してください。
3. archiveadm コマンドを使用して大域ゾーンのアーカイブを作成します。
4. zonecfg コマンドを使用して、ターゲットのゾーン構成を作成します。
5. zoneadm コマンドを使用してアーカイブからゾーンをインストールするか、installadm コマンドを使用してアーカイブからシステムをインストールします。

詳細は、[zonep2vchk\(8\)](#)、[archiveadm\(8\)](#)、[zfs\(8\)](#)、[zonecfg\(8\)](#)、[zoneadm\(8\)](#)、および [zones\(7\)](#) の各マニュアルページを参照してください。

## ソースホストとターゲットホストでのゾーンの変換の要件

solaris ゾーンの変換には、次の要件があります。

- **アーキテクチャー** - ソースシステムとターゲットシステムでは、同じ命令セットアーキテクチャー (ISA) を使用する必要があります。たとえば、両方とも SPARC ベースのシステムにするか、両方とも x86 ベースのシステムにする必要があります。
- **Oracle Solaris バージョン** - ターゲットシステム上の大域ゾーンでは、元のソースホスト上のリリース以上の Oracle Solaris リリースが動作している必要があります。
  - 大域ゾーンから非大域ゾーンに変換する前に、大域ゾーンで実行されているソフトウェアに非大域ゾーンとの互換性があることを確認します。zonep2vchk(8) ツールでは、非大域ゾーンへの変換前に大域ゾーンの構成を評価します。



- 非大域ゾーンから大域ゾーンまたはカーネルゾーンに変換する前に、ソースホストとターゲットホストの Oracle Solaris オペレーティングシステムのバージョンに互換性があることを確認する必要があります。そのほかの互換性を確認する必要はありません。それらのホストシステムに互換性があれば、非大域ゾーンで実行されているソフトウェアはすべて大域ゾーンまたはカーネルゾーンでも実行できます。
- **Oracle Solaris パッケージ** - ゾーンが正常に動作することを保証するには、元のソースホストにインストールされている必須オペレーティングシステムパッケージと同じかまたはそれ以降のバージョンのものが、ターゲットシステムにインストールされている必要があります。たとえば、ソースホストで Support Repository Update (SRU) が実行されている場合は、ターゲットホストでもその SRU またはそれ以降の SRU が実行されている必要があります。  
サードパーティー製品のパッケージなど、ほかのパッケージは異なってもかまいません。

## 非大域ゾーンが含まれているシステムからのゾーンの変換

大域ゾーンから非大域ゾーンへの変換を行うと、その大域ゾーン内にある既存の solaris ゾーンやカーネルゾーンはすべて、ゾーンの変換後に使用できなくなります。大域ゾーンとカーネルゾーンだけにはほかのゾーンを含めることができます。

---

**ヒント** - 大域ゾーンを変換する前に solaris ゾーンやカーネルゾーンを保持するには、個々のゾーンごとに別個の回復用アーカイブを作成します。次に、新しいシステムでそれらのアーカイブからゾーンを配備しなおすことができます。たとえば、最初に非大域ゾーン用のアーカイブを作成できます。次に、大域ゾーンをアーカイブできます。

---

詳細は、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』の第 1 章、「[Oracle Solaris System Recovery and Cloning \(Overview\)](#)」を参照してください。

## zonep2vchk ツールを使用した大域から非大域への変換準備

zonep2vchk ツールでは、非大域ゾーンへの変換前に大域ゾーンの構成を評価します。このツールの主なドキュメントは、[zonep2vchk\(8\)](#) のマニュアルページです。このセクションでは、次の内容について説明します。

- [106 ページの「zonep2vchk ツールについて」](#)
- [107 ページの「zonep2vchk ツールで行われるチェック」](#)

- 108 ページの「zonep2vchk ツールでハイライトされる問題」
- 108 ページの「変換されたゾーン用のゾーン構成テンプレート」

## zonep2vchk ツールについて

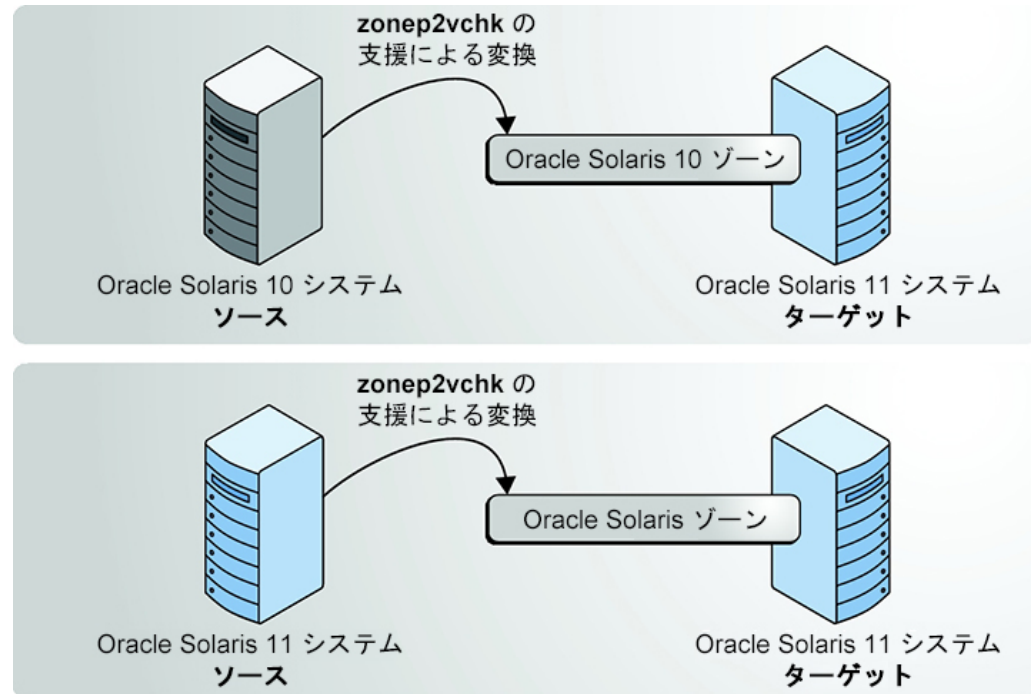
zonep2vchk ツールでは、大域ゾーン (物理インスタンス) で、それが非大域ゾーン (仮想インスタンス) に変換されるときに発生する可能性のある問題を評価します。zonep2vchk ツールは、`0` の実効ユーザー `id` で実行できます。

zonep2vchk ツールの機能は次のとおりです。

- ソースシステムの構成内の問題領域を特定します
- 必要になる手動の再構成作業を最小限に抑えます
- Oracle Solaris 大域ゾーンから、同じ Oracle Solaris リリース上の非大域ゾーンへの変換をサポートします
- 元のシステムイメージ内の複雑なネットワーク構成 (複数の IP インタフェース、IP マルチパス、VLAN など) をサポートします

zonep2vchk ツールを使用すると、Oracle Solaris の大域ゾーンから非大域ゾーンへの変換に役立ちます。このツールでは、システムを評価して、`solaris` ブランドゾーンへの変換を困難にしたり妨げたりする可能性のある問題にフラグを立て、変換を可能にするために取るべきアクションを提案します。このツールには、新しい `solaris` ゾーンを構成するためのテンプレートも用意されています。

図 2 zonep2vchk による評価のシナリオ



## zonep2vchk ツールで行われるチェック

zonep2vchk オプションを使用すると、何種類かのチェックを実行できます。

- **基本分析**では、大域から非大域への変換によって影響を受ける可能性のある使用中の Oracle Solaris 機能をチェックします。オプションを指定しない場合は、これがデフォルトです。

ほかのオプションと組み合わせるときは、基本チェックに `-b` オプションを使用します。

- **静的分析**では、ゾーン内で機能しない可能性のあるシステムおよびライブラリコールがないかどうかバイナリを検査します。

英小文字の `-s` オプションを使用すると、分析対象のバイナリへのパスを指定でき、英大文字の `-s` オプションを使用すると、分析対象のファイルやディレクトリを一覧表示するファイルへのパスを指定できます。

- **実行時分析**では、現在実行中のアプリケーションに、ゾーン内で機能しない可能性のある操作がないかどうかを検査します。  
-r オプションを指定された時間 (時間、分、または秒) とともに使用します。また、-x オプションを使用すると、Ctrl-C を入力するなどのシグナル割り込みを送信するまで分析できます。

詳細は、[180 ページの「zonep2vchk によるソースシステムのチェック方法」](#)、および [zonep2vchk\(8\)](#) のマニュアルページを参照してください。

## zonep2vchk ツールでハイライトされる問題

zonep2vchk ツールを使用して各種チェックを実行すると、次の主なカテゴリの情報が生成されます。

- 特定のゾーンの構成または大域ゾーン内の構成変更によって対処できる問題
- ゾーン内で実行できない機能の特定

たとえば、アプリケーションでシステムクロックを設定する場合、その機能を solaris ゾーンで有効にするには、適切な特権をそのゾーンに追加します。ただし、カーネルメモリーにアクセスするアプリケーションを solaris ゾーンで実行することはできません。出力では、これらの 2 つのクラスの問題を区別します。

zonep2vchk ツールは、デフォルトでは人間が読める形式のメッセージを出力します。メッセージを機械解析可能な形式で出力するには、-P オプションを使用します。詳細は、[zonep2vchk\(8\)](#) のマニュアルページを参照してください。

## 変換されたゾーン用のゾーン構成テンプレート

zonep2vchk ツールには、zonecfg コマンドで用いるためのテンプレートの作成に使用できる -c オプションが備わっています。このテンプレートは、大域ゾーンのリソースの一部を非大域ゾーンに適したものになるように構成します。

テンプレートを作成する前に、大域ゾーンで分析チェックを実行して、必要な変更を行うことができます。テンプレートを作成するには、コマンドを実行してその出力をファイルに送ります。次に、そのファイルを zonecfg コマンドの入力として使用して、ゾーン構成の開始ポイントを作成します。例:

```
global$ pfbash zonep2vchk -c > myzone.config
global$ zonecfg -z myzone -f myzone.config
```

## 非大域ゾーンが含まれているシステムのバックアップ方法

非大域ゾーンを個別にバックアップしたり、大域ゾーンからシステム全体をバックアップしたりできます。

### 非大域ゾーンが含まれているシステムのバックアップ手法

ゾーン内から、データベースまたはアプリケーションのみをバックアップしたり、ゾーン全体をバックアップしたりできます。大域ゾーンから非大域ゾーンをバックアップすることもできます。バックアップは簡単に復元されます。

#### ループバックファイルシステムのディレクトリのバックアップ

複数のコピーを防止するために、ループバックファイルシステム (lofs) のバックアップおよび復元は大域ゾーンからのみ行います。非大域ゾーンのループバックファイルシステムをバックアップしないでください。

非大域ゾーン内から読み取り/書き込みループバックファイルシステムをバックアップおよび復元する場合、これらのファイルシステムは大域ゾーンおよび読み取り/書き込みマウントされているほかのすべてのゾーンからも書き込むことができます。

#### 大域ゾーンからのシステムのバックアップ

次のような場合は、大域ゾーンからバックアップを実行することをお勧めします。

- 非大域ゾーンの構成をアプリケーションデータと一緒にバックアップする場合。
- 障害から回復することがもっとも重要である場合。使用しているゾーンのルートファイルシステムおよびそれらの構成データ、使用している大域ゾーン内のデータなど、システム上のすべてまたはほぼすべての情報を復元する必要がある場合。このような場合は、大域ゾーンでバックアップを実行してください。
- 市販のネットワークバックアップソフトウェアを使用する場合。

---

**注記** - ネットワークバックアップソフトウェアを使用するときには、可能であれば、継承された lofs ファイルシステムはすべてスキップするように構成することをお勧めします。バックアップは、ゾーンとそのアプリケーションがバックアップ対象のデータを休止させた状態のときに、行うことをお勧めします。

---

## システム上の非大域ゾーンを個別にバックアップ

次のような場合は、非大域ゾーン内からバックアップを実行することをお勧めします。

- 非大域ゾーンの管理者が、重大度の低い障害から回復する機能、またはゾーンに固有のアプリケーションデータまたはユーザーデータを復元する機能を必要とする場合。
- ゾーン内で動作する特定のアプリケーションまたはサービスのバックアップソフトウェアを使用する場合。ディレクトリパスやインストール済みソフトウェアなどのアプリケーション環境が大域ゾーンと非大域ゾーンとの間で異なっている場合には、バックアップソフトウェアを大域ゾーンから実行するのが困難な場合があります。

アプリケーションが非大域ゾーンごとのバックアップスケジュールに基づいてスナップショットを実行し、そのバックアップデータを大域ゾーンからエクスポートした書き込み可能なディレクトリに格納できる場合には、大域ゾーンの管理者は大域ゾーンからバックアップする処理の中でそれら個々のバックアップデータを個別に選択できます。

## Oracle Solaris 統合アーカイブのバックアップの作成

zonecfg および zoneadm コマンドを使用すると、統合アーカイブファイルから直接、新しいゾーンを構成およびインストールできます。統合アーカイブファイルには、ゾーン構成とゾーンデータの両方が含まれています。宛先システムで、ゾーンをアーカイブから構成およびインストールできます。

システムおよびゾーンのクローニングや回復のための使用方法を含む統合アーカイブの詳細は、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』を参照してください。

## 非大域ゾーン内でバックアップするデータの決定

非大域ゾーン内のデータをすべてバックアップすることも、アプリケーションデータのみをバックアップすることもできます。

## アプリケーションデータのみのバックアップ

アプリケーションデータがファイルシステムの特定の場所に格納されている場合には、このデータだけを定期的にバックアップすることもできます。ゾーンのルートファイルシステムはそれほど頻繁には変更されないため、頻繁にバックアップする必要がない場合もあります。

アプリケーションファイルがどこに格納されているかを確認する必要があります。アプリケーションファイルは次のディレクトリに格納されている可能性があります。

- ユーザーのホームディレクトリ
- /etc (構成データファイルの場合)
- /var

アプリケーション管理者がデータの格納場所を認識している場合には、ゾーンごとに書き込み可能ディレクトリを割り当てるように、システムを作成することもできます。バックアップがゾーンごとに格納されるので、大域管理者または適切な承認を付与されたユーザーは、その場所をシステム上のバックアップ対象の1つとして選択することができます。

## 一般的なデータベースバックアップ操作

データベースアプリケーションデータがデータベース固有のディレクトリに存在していない場合には、次の規則が適用されます。

- 最初にデータベースが安定した状態にあることを確認します。

データベースが休止している必要があります。内部バッファのデータがディスクにフラッシュされるためです。大域ゾーンからバックアップを開始する前に、非大域ゾーン内のデータベースが安定した状態になっていることを確認してください。
- ゾーンごとに、ファイルシステム機能を使用してデータのスナップショットを作成してから、そのスナップショットを大域ゾーンから直接バックアップします。

このようにすると、バックアップのための時間が短縮されるだけでなく、すべてのゾーンのクライアントやモジュールをバックアップする必要がなくなります。

## テープによるバックアップ

非大域ゾーンだけが使用するファイルシステムについては、そのゾーンにとって都合のよい時間帯に、アプリケーションがわずかに休止している時間を利用して、スナップショットを作成することもできます。それらのスナップショットは、アプリケーションがサービスに戻ったあとに大域ゾーンからバックアップしてテープに格納できます。

このスナップショット方法により、次の利点が得られます。

- 必要なテープデバイスが少なくすみずみます。
- 非大域ゾーンの間で調整する必要がなくなります。
- デバイスを直接ゾーンに割り当てる必要がないため、セキュリティ機能が向上します。
- 大域ゾーンでシステム管理を続けることができるので、通常はこの方法をお勧めします。

## 非大域ゾーンの復元について

大域管理者または適切な承認を付与されたユーザーは、大域ゾーンから実行したバックアップを復元するときには、関係するゾーンを再インストールしてから、そのゾーンのファイルを復元できます。これは、次のことを想定しています。

- 復元するゾーンの構成が、バックアップしたときと同じ構成であること。
- バックアップしてからゾーンを復元するまでの間に、大域ゾーンをアップデートしていないこと。

これらの前提を満たしていない場合は、一部のファイルが復元によって上書きされ、手作業でマージしなければならないことがあります。

---

**注記** - 大域ゾーンのすべてのファイルシステムが失われた場合には、大域ゾーンのすべてのファイルを復元すると、非大域ゾーンも復元されます。ただし、非大域ゾーンの各ルートファイルシステムがバックアップに含まれている必要があります。

---

## 非大域ゾーンが含まれているシステムで使用されるコマンド

このセクションには、ゾーンで使用されるコマンドの説明とマニュアルページへのリンクが記載されています。

### ゾーンの管理およびモニタリング用のコマンド

表6に示すコマンドは、Oracle Solaris ゾーンへの主要な管理インタフェースを提供します。

---

**注記** - zoneadmd デーモンは、ゾーンの仮想プラットフォームを管理する基本プロセスです。このデーモンは、プログラミングインタフェースの構成要素ではありません。詳細は、[zoneadmd\(8\)](#) のマニュアルページを参照してください。

---

表 6 ゾーンの管理およびモニターに使用するコマンド

マニュアルページリファレンス	説明
<a href="#">getzoneid(3C)</a>	ゾーン ID と名前とのマッピングに使用されます
<a href="#">zcons(4D)</a>	ゾーンコンソールのデバイスドライバ
<a href="#">zlogin(1)</a>	非大域ゾーンへのログインに使用されます



マニュアルページリファレンス	説明
<a href="#">zoneadm(8)</a>	システムでのゾーンの管理に使用されます
<a href="#">zonecfg(8)</a>	ゾーン構成の設定に使用されます
<a href="#">zonename(1)</a>	現在のゾーンの名前を出力します
<a href="#">zones(7)</a>	ゾーン機能の説明を提供します
<a href="#">zonestat(1)</a>	ゾーンのリソース使用率の監視に使用されます

## リソース上限制御用のコマンド

次の表に示すコマンドは、リソース上限デーモンとともに使用されます。

表 7 rcapd とともに使用されるコマンド

マニュアルページリファレンス	説明
<a href="#">rcapadm(8)</a>	リソース上限デーモンを構成します。構成済みのリソース上限デーモンの現在のステータスを表示します。リソース上限制御を有効または無効にします。
<a href="#">rcapd(8)</a>	リソース上限デーモン。
<a href="#">rcapstat(1)</a>	上限が定義されたプロジェクトのリソース使用効率をモニターします。

## ゾーン用に変更されたコマンド

次の表で示すコマンドは、ゾーンがインストールされている Oracle Solaris システムで使用できるように変更されています。これらのコマンドには、ゾーンに固有のオプションが用意されています。指定するオプションによって異なる情報が表示されます。コマンドは、マニュアルページのセクション別に記載されています。

表 8 ゾーンがインストールされている Oracle Solaris システムで使用するために変更されたコマンド

マニュアルページリファレンス	説明
<a href="#">audit_syslog(7)</a>	zonename 監査ポリシーが設定されている場合に使用される in zonename フィールドが追加されました。
<a href="#">auditconfig(8)</a>	zonename トークンが追加されました。
<a href="#">auditreduce(8)</a>	-z zone-name オプションが追加されました。ゾーンの監査ログを取得する機能が追加されました。
<a href="#">cmn_err(9F)</a>	ゾーンパラメータが追加されました。
<a href="#">core(5)</a>	n_type: NT_ZONE_NAME が追加されました。このエントリには、プロセスが実行されていたゾーンの名前を示す文字列が含まれます。
<a href="#">coreadm(8)</a>	プロセスが実行されたゾーンを識別するための変数 %z が追加されました。

マニュアルページリファレンス	説明
<a href="#">ddi_cred(9F)</a>	cr の指し示すユーザー資格からゾーン ID を返す、 <code>crgetzoneid()</code> が追加されました。
<a href="#">df(8)</a>	すべての可視ゾーン内のマウントを表示する <code>-Z</code> オプションが追加されました。このオプションは、非大域ゾーンでは効果がありません。
<a href="#">dladm(8)</a>	デフォルトのコマンド出力にゾーン列を追加する <code>-z</code> オプションが <code>show</code> サブコマンドに追加されました。ゾーン列は、現在、リソースが割り当てられているゾーンを示します。
<a href="#">dlstat(8)</a>	デフォルトのコマンド出力にゾーン列を追加する <code>-z</code> オプションが <code>show</code> サブコマンドに追加されました。ゾーン列は、現在、リソースが割り当てられているゾーンを示します。
<a href="#">fsstat(8)</a>	<p>ゾーン別のファイルシステムアクティビティについて報告する <code>-z</code> オプションが追加されました。複数の <code>-z</code> オプションを使用して、選択したゾーンのアクティビティをモニターできます。このオプションは、<code>fstypes</code> ではなく <code>mountpoints</code> のみをモニターする場合に使用しても効果はありません。</p> <p>指定した <code>fstypes</code> のファイルシステムアクティビティのすべてのゾーンにわたる集計を報告する <code>-A</code> オプションが追加されました。これは、<code>-z</code> または <code>-Z</code> オプションがどちらも使用されなかった場合のデフォルトの動作です。<code>-A</code> オプションは、<code>fstypes</code> ではなく <code>mountpoints</code> のみをモニターする場合に使用しても効果はありません。</p> <p><code>-A</code> オプションを <code>-z</code> または <code>-Z</code> オプションとともに使用すると、指定した <code>fstypes</code> のすべてのゾーンにわたる集計が別個の行に表示されます。</p> <p>システム上のすべてのゾーンのファイルシステムアクティビティを報告する <code>-Z</code> オプションが追加されました。このオプションは、<code>-z</code> オプションとともに使用しても効果はありません。このオプションは、<code>fstypes</code> ではなく <code>mountpoints</code> のみをモニターする場合に使用しても効果はありません。</p>
<a href="#">getloadavg(3C)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効な場合、 <code>PS_MYID</code> の <code>psetid</code> を使用して呼び出した場合と動作は同じになります。
<a href="#">getpriority(3C)</a>	ゾーン ID が、指定可能なターゲットプロセスに追加されました。ゾーン ID が <code>EINVAL</code> のエラー説明に追加されました。
<a href="#">if_tcp(4P)</a>	ゾーンの <code>ioctl()</code> 呼び出しが追加されました。
<a href="#">iostat(8)</a>	プール機能が有効な非大域ゾーン内で実行すると、ゾーンのバインド先プールのプロセッサセット内のプロセッサ情報だけが提供されます。
<a href="#">ipcrm(1)</a>	<code>-z zone</code> オプションが追加されました。このオプションは、大域ゾーン内でコマンドを実行する場合にのみ有用です。
<a href="#">ipcs(1)</a>	<code>-z zone</code> オプションが追加されました。このオプションは、大域ゾーン内でコマンドを実行する場合にのみ有用です。
<a href="#">kstat(8)</a>	大域ゾーンで実行すると、すべてのゾーンの <code>kstat</code> が表示されます。非大域ゾーンで実行すると、一致する <code>zoneid</code> を持つ <code>kstat</code> だけが表示されます。
<a href="#">mpstat(8)</a>	プール機能が有効な非大域ゾーン内で実行すると、ゾーンのバインド先プールのプロセッサセット内のプロセッサの行だけが表示されます。
<a href="#">Unresolved link to "nnd8"</a>	大域ゾーン内で使用された場合、すべてのゾーンの情報を表示します。排他的 IP ゾーンの TCP/IP モジュールに対して <code>nnd</code> を実行すると、そのゾーンの情報が表示されます。
<a href="#">netstat(8)</a>	現在のゾーンのみを表示します。

マニュアルページリファレンス	説明
<a href="#">nfsstat(8)</a>	現在のゾーンのみ統計情報を表示します。
<a href="#">p_online(2)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効であるが、プロセッサがゾーンのバインド先プールのプロセッサセット内でない場合、エラーが返されます。
<a href="#">pgrep(1)</a>	-z <i>zoneidlist</i> オプションが追加されました。このオプションは、大域ゾーン内でコマンドを実行する場合にのみ有用です。
<a href="#">pkginfo(5)</a>	ゾーンのサポート内でオプションパラメータおよび環境変数が提供されるようになりました。
<a href="#">poolbind(8)</a>	<i>zoneid</i> リストが追加されました。リソースプールとともにゾーンを使用する方法については、『 <a href="#">Administering Resource Management in Oracle Solaris 11.4</a> 』の「 <a href="#">Resource Pools Used in Zones</a> 」も参照してください。
<a href="#">ppriv(1)</a>	-l オプションとともに使用する式 <i>zone</i> が追加されました。これを使用すると、現在のゾーン内で使用可能なすべての特権が一覧表示されます。また、 <i>zone</i> に -v オプションを指定して、冗長出力を取得できます。
<a href="#">prioctl(1)</a>	<i>idlist</i> と -i <i>idtype</i> でゾーン ID を使用することで、プロセスを指定できます。prioctl -i <i>zoneid</i> コマンドを使用すると、実行中のプロセスを非大域ゾーン内の別のスケジューリングクラスに移動できます。
<a href="#">prioctl(2)</a>	P_ZONEID <i>id</i> 引数が追加されました。
<a href="#">priv_str_to_set(3C)</a>	呼び出し側のゾーン内部で使用可能なすべての特権セットで、「zone」文字列が追加されました。
<a href="#">privileges(7)</a>	プロセスによる追跡またはほかのゾーン内のプロセスへのシグナル送信を可能にする PRIV_PROC_ZONE が追加されました。zones(7) を参照してください。
<a href="#">proc(1)</a>	ptree だけに -z <i>zone</i> オプションが追加されました。このオプションは、大域ゾーン内でコマンドを実行する場合にのみ有用です。
<a href="#">proc(5)</a>	ゾーン内で実行中のプロセスに関する情報を取得する機能が追加されました。
<a href="#">processor_info(2)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効であるが、プロセッサがゾーンのバインド先プールのプロセッサセット内でない場合、エラーが返されます。
<a href="#">prstat(8)</a>	-z <i>zoneidlist</i> オプションが追加されました。また、-Z オプションも追加されました。  プール機能が有効な非大域ゾーンで実行した場合、ゾーンのバインド先プールのプロセッサセット内のプロセッサだけを対象にして、プロセスが使用した最新の CPU 時間の比率が表示されます。  -a、-t、-T、-J、および -Z の各オプションを指定すると、出力にはサイズ列の代わりにスワップ列が表示されます。報告されるスワップは、ゾーンのプロセスと tmpfs マウントで消費されるスワップの合計量です。この値により、各ゾーンで予約されているスワップをモニタリングしやすくなり、適切な zone.max-swap 設定を選択できます。
<a href="#">ps(1)</a>	-o オプションが使用する、認識される format 名リストに、zonename と zoneid が追加されました。  指定したゾーン内のプロセスだけを一覧表示するため、-z <i>zonelist</i> が追加されました。ゾーンの指定には、ゾーン名またはゾーン ID を使用できます。このオプションは、大域ゾーン内でコマンドを実行する場合にのみ有用です。  プロセスに関連するゾーンの名前を出力するため、-Z が追加されました。名前は、追加された列ヘッダー ZONE の下に出力されます。

マニュアルページリファレンス	説明
<a href="#">pset_bind(2)</a>	<i>idtype</i> として P_ZONEID が追加されました。P_MYID 仕様の選択肢にゾーンが追加されました。EINVAL エラー説明内の有効な <i>idtype</i> リストに P_ZONEID が追加されました。
<a href="#">pset_getloadavg(3C)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効であるが、プロセッサがゾーンのバインド先プールのプロセッサセット内でない場合、エラーが返されます。
<a href="#">pset_info(2)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効であるが、プロセッサがゾーンのバインド先プールのプロセッサセット内でない場合、エラーが返されます。
<a href="#">pset_list(2)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効であるが、プロセッサがゾーンのバインド先プールのプロセッサセット内でない場合、エラーが返されます。
<a href="#">pset_setattr(2)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効であるが、プロセッサがゾーンのバインド先プールのプロセッサセット内でない場合、エラーが返されます。
<a href="#">psrinfo(8)</a>	非大域ゾーン内で実行した場合、ゾーンで表示可能なプロセッサの情報だけが表示されます。
<a href="#">renice(1)</a>	有効な引数をリスト表示するため、-i オプションとともに使用する <i>zoneid</i> が追加されました。
<a href="#">sar(1)</a>	プール機能が有効な非大域ゾーン内で実行する際、-b、-c、-g、-m、-p、-u、-w、および -y オプションを指定すると、ゾーンのバインド先プールのプロセッサセット内のプロセッサの値だけが表示されます。
<a href="#">sysconf(3C)</a>	呼び出し側が非大域ゾーン内にあり、プール機能が有効な場合、 <code>sysconf(_SC_NPROCESSORS_CONF)</code> および <code>sysconf(_SC_NPROCESSORS_ONLN)</code> は、ゾーンのバインド先プールのプロセッサセット内の合計プロセッサ数およびオンラインのプロセッサ数を返します。
<a href="#">sysinfo(2)</a>	PRIV_SYS_CONFIG が PRIV_SYS_ADMIN に変更されました。
<a href="#">Unresolved link to "traceroute8"</a>	使用方法が変更されました。非大域ゾーン内から指定した場合、-F オプションは効果がありません。理由は、「断片化しない」というビットが常に設定されているためです。
<a href="#">ucred_get(3C)</a>	<code>ucred_getzoneid()</code> 関数が追加されました。この関数は、プロセスのゾーン ID を返します。ただし、ゾーン ID を取得できなかった場合は -1 を返します。
<a href="#">umount(2)</a>	<i>file</i> が参照しているファイルが絶対パスでない場合、ENOENT が返されます。
<a href="#">vmstat(8)</a>	プール機能が有効な非大域ゾーン内で実行すると、ゾーンのバインド先プールのプロセッサセット内のプロセッサ統計情報だけが報告されます。-p オプション指定時の出力と、 <code>page</code> 、 <code>faults</code> 、および <code>cpu</code> 報告フィールドに適用されます。

## 非大域ゾーン構成の作成と変更

この章では、ゾーンの構成方法、ゾーン構成の変更方法、およびシステムからゾーン構成を削除する方法について説明します。この章の内容は次のとおりです。

- [117 ページの「非大域ゾーン構成のタスクマップ」](#)
- [118 ページの「ゾーンの構成」](#)
- [127 ページの「ゾーン構成の変更」](#)
- [133 ページの「ゾーン構成を元に戻す、または削除する」](#)

バックグラウンド情報については、次を確認してください。

- [第1章「非大域ゾーン構成の計画」](#)
- 『Oracle Solaris Zones Configuration Resources』の第1章、「Non-Global Zone Configuration Command and Resources」

### 非大域ゾーン構成のタスクマップ

次のタスクマップでは、非大域ゾーン構成のライフサイクルにおける主要なタスクを説明しているセクションへのリンクを示します。

タスク	説明	参照先
非大域ゾーンを構成します。	<p>zonecfg ユーティリティを使用してゾーンの作成、構成の検証および確定を行います。ゾーン構成プロファイルを使用して、システム上の複数のゾーンを構成およびブートすることもできます。</p> <p>非大域ゾーンの構成を表示するには、zonecfg ユーティリティを使用できます。</p>	<a href="#">118 ページの「ゾーンの構成」</a>
ゾーン構成を変更します。	これらの手順を使用して、ゾーン構成内のリソースタイプを変更するか、ゾーン名などのグローバルプロパティタイプを変更する	<a href="#">127 ページの「ゾーン構成の変更」</a>

タスク	説明	参照先
	か、またはゾーンに専用のデバイスを追加します。	
ゾーン構成を元に戻すか削除します。	zonecfg コマンドと revert サブコマンドを使用して、ゾーン構成に対して行なったりソース設定を取り消すか、ゾーン構成を削除します。	133 ページの「ゾーン構成を元に戻す、または削除する」
ゾーン構成を削除します。	zonecfg コマンドと delete サブコマンドを使用して、システムからゾーン構成を削除します。	134 ページの「アンインストールされたゾーン構成を削除する方法」

## ゾーンの構成

非大域ゾーン構成の作成、必要な情報がすべて存在していることの検証、およびゾーン構成の確定には、zonecfg ユーティリティを使用します。zonecfg ユーティリティは、大域ゾーンのリソース管理設定を永続的に指定する場合にも使用できます。

**ヒント** - zonecfg ユーティリティを使用してゾーンを構成する際、revert サブコマンドを使用して、リソースの設定を元に戻すことができます。133 ページの「ゾーン構成の変更を元に戻す方法」を参照してください。

このセクションでは、次の手順について説明します。

- 118 ページの「構成プロファイルを使用してゾーンを構成する方法」
- 119 ページの「zonecfg ユーティリティを使用してゾーンを構成する方法」
- 125 ページの「非大域ゾーンの構成を表示する方法」

詳細は、zonecfg(8) のマニュアルページを参照してください。

### ▼ 構成プロファイルを使用してゾーンを構成する方法

ゾーン構成プロファイルを作成するには、この手順を実行します。次に、このプロファイルを zlogin コマンドに指定して、初回ブート時にゾーンを構成します。

**始める前に** ゾーンのどのグローバルプロパティおよびリソースを構成するかを特定し、どの設定を指定するかを決定します。ゾーンで構成できる zonecfg のグローバルプロパティおよびリソースタイプについては、『Oracle Solaris Zones Configuration Resources』の「Zone Configuration Data」および zonecfg(8) のマニュアルページを参照してください。



**注意** - 必要なすべてのデータを提供する必要があります。データが欠落したプロファイルを提供すると、ゾーンはデータが欠落した状態で構成されます。この構成によって、ユーザーがログインできなくなったり、ネットワークを実行できなくなったりする可能性があります。

### 1. ゾーン管理者になります。

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

### 2. プロファイルを作成します。

『Oracle Solaris 12 システムの自動インストール』の第9章、「Unconfiguring or Reconfiguring an Oracle Solaris Instance」、および `sysconfig(8)` と `zones(7)` のマニュアルページを参照してください。

```
# sysconfig create-profile -o /path/profile.xml
```

#### ■ 排他的 IP ゾーンの場合:

```
# sysconfig create-profile -o /path/profile.xml
```

#### ■ 共有 IP ゾーンの場合:

```
# sysconfig create-profile -o /path/profile.xml -g location,identity,naming_services,users
```

次の手順 ゾーンのインストール、クローン、または接続の操作中に、この作成したプロファイルを使用します。例:

```
global$ pfbash zoneadm -z zonename install -c /path/profile.xml
```

構成ファイルが使用されている場合、システムは `zlogin` コマンドによる最初のログイン時にコンソール上で対話式システム構成 (SCI) ツールを開始しません。

## ▼ zonecfg ユーティリティーを使用してゾーンを構成する方法

この手順では、いくつかのグローバルプロパティおよびリソースタイプを設定し、新しい非大域ゾーンを構成する方法を示します。

**注記** - 代わりにゾーン構成ファイルを作成し、`zlogin` コマンドを使用してゾーンの初回ブート時にゾーン構成を作成する場合は、118 ページの「構成プロファイルを使用してゾーンを構成する方法」に進みます。

非大域ゾーンの作成時に指定する必要があるパラメータは、ゾーン名のみです。zonepath パラメータは割り当てられます。そのほかのリソースおよびプロパティはオプションです。オプションのリソースには、dedicated-cpu リソースと capped-cpu リソースのどちらを使用するかを決めるなど、選択肢の中から選ぶ必要があるものもあります。

---

**ヒント** - zonecfg コマンドは、複数のサブコマンドもサポートします。次に示すように、同じシェル呼び出しで引用符で囲み、セミコロンで区切ります。例:

```
global$ pfbash zonecfg -z kzone1 "create ; set autoboot=true"
```

共有 IP ゾーンの場合、zonecfg net リソースで静的アドレスを割り当てできるだけです。このアドレスは、コマンド行で指定することはできません。

---

始める前に ゾーンのどのグローバルプロパティおよびリソースを構成するかを特定し、それらの値を決定します。ゾーンで構成できるグローバルプロパティおよびリソースタイプについては、『[Oracle Solaris Zones Configuration Resources](#)』の「[Zone Configuration Data](#)」および [zonecfg\(8\)](#) のマニュアルページを参照してください。

#### 1. ゾーン管理者になります。

詳細については、[68 ページ](#)の「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

#### 2. 選択したゾーン名を使用して、ゾーン構成を開始します。

```
global$ pfbash zonecfg -z zonename
```

このゾーンの初回構成時には、次のシステムメッセージが表示されます。

```
zonename: No such zone configured
Use 'create' to begin configuring a new zone.
```

#### 3. 新しいゾーン構成を作成します。

このステップでは、デフォルト設定を使用します。

```
zonecfg:zonename> create
create: Using system default template 'SYSdefault'
zonecfg:zonename> info
  zonename: zonename
  zonepath.template: /system/zones/{zonename}
  zonepath: /system/zones/zonename
```

#### 4. (オプション) autoboot 値を設定します。

true に設定すると、大域ゾーンのブート時にこのゾーンが自動的にブートします。デフォルト値は false です。



---

**注記** - ゾーンを自動的にブートできるためには、`svc:/system/zones:default` サービスが有効になっている必要があります。このサービスは、デフォルトで有効になっています。

---

```
zonecfg:zonename> set autoboot=true
```

**5. (オプション) ゾーンの永続的なブート引数を設定します。**

```
zonecfg:zonename> set bootargs="-m verbose"
```

**6. (オプション) 1 つ以上の CPU をこのゾーン専用に割り当てます。**

CPU の数または数の範囲を設定し、必要に応じて重要性も設定します。

```
zonecfg:zonename> add dedicated-cpu
zonecfg:zonename:dedicated-cpu> set ncpus=N
zonecfg:zonename:dedicated-cpu> set importance=N
zonecfg:zonename:dedicated-cpu> end
```

**7. (オプション) 権限のデフォルトセットを修正します。**

設定する特権をコンマ区切りリストで指定します。この設定は、システムクロックを設定する機能をデフォルトの特権セットに追加します。

```
zonecfg:zonename> set limitpriv="privileges-list"
```

**8. (オプション) スケジューリングクラスを FSS に設定します。**

```
zonecfg:zonename> set scheduling-class=FSS
```

**9. (オプション) メモリー上限を追加します。**

このステップでは、メモリーの上限、スワップメモリーの上限、およびロックされたメモリーの上限の値を設定します。

---

**注記** - `capped-memory` リソースを使用するには、`resource-cap` パッケージが大域ゾーン内にインストールされている必要があります。

---

```
zonecfg:zonename> add capped-memory
zonecfg:zonename:capped-memory> set physical=value
zonecfg:zonename:capped-memory> set swap=value
zonecfg:zonename:capped-memory> set locked=value
zonecfg:zonename:capped-memory> end
```

**10. (オプション) ファイルシステムを追加します。**

このステップでは、マウントポイント、ゾーンのマウントポイント、およびファイルシステムタイプを設定します。

```
zonecfg:zonename> add fs
zonecfg:zonename:fs> set dir=mountpoint
zonecfg:zonename:fs> set special=zone-mountpoint
zonecfg:zonename:fs> set type=fs-type
```

```
zonecfg:zonename:fs> end
```

複数のファイルシステムを追加するには、このステップを繰り返します。

**11. (オプション) hostid グローバルプロパティを設定します。**

```
zonecfg:zonename set hostid=ID
```

**12. (オプション) ゾーンで使用するストレージプールに ZFS データセットを追加します。**

dataset リソースには、ZFS ファイルシステムのパスを指定する必要があります。ゾーン管理者は、子ファイルシステムとその子孫のクローンを作成できます。ゾーン管理者は、データセットのプロパティを変更したり、圧縮や暗号化を制御したりできます。

```
zonecfg:zonename> add dataset
zonecfg:zonename> set name=storage-pool/dataset
zonecfg:zonename> end
```

**13. (オプション) 自動 VNIC を持つ排他的 IP ゾーンを作成します。**

作成されるリンクの基礎となるリンクとして auto を指定します。zoneadmd デーモンは、ゾーンがブートするたびに VNIC が作成されるリンクを自動的に選択します。データリンクを選択するときは、IPoIB リンクはスキップされます。

```
zonecfg:zonename> set ip-type=exclusive
zonecfg:zonename> add anet
zonecfg:zonename:anet> set lower-link=auto
zonecfg:zonename:anet> end
```

**14. (オプション) デバイスを追加します。**

```
zonecfg:zonename> add device
zonecfg:zonename:device> set match=/dev/sound/*
zonecfg:zonename:device> end
```

複数のデバイスを追加するには、このステップを繰り返します。

**15. (オプション) InfiniBand (IB) 診断ツール以外の OFUV (Open Fabrics User Verbs) のコンポーネントに対する OFUV デバイスを追加します。**

---

注記 - IB 診断ツールは、非大域ゾーンではサポートされていません。追加されたデバイスは、OFUV のコンポーネント (「Verb」や rdma\_cm など) とともに使用できます。

---

```
zonecfg:zonename> add device
zonecfg:zonename:device> set match=infiniband/ofs/*
zonecfg:zonename:device> end
```

複数のデバイスを追加するには、このステップを繰り返します。

**16. (オプション) IB 診断ツール以外の OFUV のコンポーネントに対する OFUV デバイスを追加します。**

複数のデバイスを追加するには、このステップを繰り返します。

---

**注記** - IB 診断ツールは、非大域ゾーンではサポートされていません。追加されたデバイスは、OFUV のコンポーネント (rdma\_cm など) とともに使用できます。

---

```
zonecfg:zonename> add device
zonecfg:zonename:device> set match=infiniband/hca/*
zonecfg:zonename:device> end
```

17. (オプション) **format** コマンドによるディスクのラベル付けを可能にするには、ディスクまたは LUN 全体をゾーンに委任し、**allow-partition** プロパティを設定します。

```
zonecfg:zonename> add device
zonecfg:zonename:device> set match=/dev/*dsk/cNtXdY*
zonecfg:zonename:device> set allow-partition=true
zonecfg:zonename:device> end
```

複数のデバイスを追加するには、このステップを繰り返します。

18. (オプション) ディスクに対して **uscsi** 操作を可能にするには、**allow-raw-io** プロパティを設定します。




---

**注意** - ゾーンでディスクに対する **uscsi** 操作の実行を可能にすると、ゾーンはディスクと同じバスに接続されている、すべての他のデバイスにもアクセスできます。したがって、この機能を有効にするとセキュリティリスクが生じ、同じバス上のリソースを使用する大域ゾーンまたはほかのゾーンに対する攻撃が可能になります。詳細は、[uscsi\(4I\)](#)を参照してください。

---

```
zonecfg:zonename> add device
zonecfg:zonename:device> set match=/dev/*dsk/cNtXdY*
zonecfg:zonename:device> set allow-raw-io=true
zonecfg:zonename:device> end
```

複数のデバイスを追加するには、このステップを繰り返します。

19. (オプション) グローバルプロパティ名を使用して、ゾーン規模のリソース制御を追加します。

```
zonecfg:zonename> set property=value
```

複数のリソース制御を追加するには、このステップを繰り返します。

20. (オプション) コメントを追加します。

```
zonecfg:zonename> add attr
zonecfg:zonename:attr> set name=comment
zonecfg:zonename:attr> set type=string
zonecfg:zonename:attr> set value="comment"
zonecfg:zonename:attr> end
```

21. ゾーン構成を確認します。

```
zonecfg:zonename> verify
```

## 22. 変更を確定して終了します。

```
zonecfg:zonename> commit
zonecfg:zonename> exit
```

確定された変更は、ゾーンの次回ブート時に有効になります。

プロンプトで `commit` コマンドを明示的に入力しなくても、`exit` を入力するか EOF が発生すると、`commit` の実行が自動的に試みられます。

### 例 6 zonecfg ユーティリティーを使用して新しいゾーンを構成する

この例では、システムのデフォルトテンプレート `SYSdefault` を使用して、`my-zone` という名前の新しいゾーンを構成します。

```
global$ pfbash zonecfg -z my-zone
my-zone: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:my-zone> create
create: Using system default template 'SYSdefault'
zonecfg:my-zone> info
  zonename: my-zone>
  zonepath.template: /system/zones/{zonename}
  zonepath: /system/zones/my-zone
zonecfg:my-zone> set autoboot=true
zonecfg:my-zone> set bootargs="-m verbose"
zonecfg:my-zone> add dedicated-cpu
zonecfg:my-zone:dedicated-cpu> set ncpus=1-2
zonecfg:my-zone:dedicated-cpu> set importance=10
zonecfg:my-zone:dedicated-cpu> end
zonecfg:my-zone> set limitpriv="default,sys_time"
zonecfg:my-zone> set scheduling-class=FSS
zonecfg:my-zone> add capped-memory
zonecfg:my-zone:capped-memory> set physical=1g
zonecfg:my-zone:capped-memory> set swap=2g
zonecfg:my-zone:capped-memory> set locked=500m
zonecfg:my-zone:capped-memory> end
zonecfg:my-zone> add fs
zonecfg:my-zone:fs> set dir=/usr/local
zonecfg:my-zone:fs> set special=/opt/local
zonecfg:my-zone:fs> set type=lofs
zonecfg:my-zone:fs> end
zonecfg:my-zone> set hostid=80f0c086
zonecfg:my-zone> add dataset
zonecfg:my-zone> set name=tank/sales
zonecfg:my-zone> end
zonecfg:my-zone> set ip-type=exclusive
zonecfg:my-zone> add anet
zonecfg:my-zone:anet> set lower-link=auto
zonecfg:my-zone:anet> end
zonecfg:my-zone> add device
zonecfg:my-zone:device> set match=/dev/sound/*
zonecfg:my-zone:device> end
zonecfg:my-zone> add device
zonecfg:my-zone:device> set match=infiniband/ofs/*
zonecfg:my-zone:device> end
zonecfg:my-zone> add device
zonecfg:my-zone:device> set match=infiniband/hca/*
zonecfg:my-zone:device> end
zonecfg:my-zone> add device
zonecfg:my-zone:device> set match=/dev/*dsk/c2t40d3*
```

```

zonecfg:my-zone:device> set allow-partition=true
zonecfg:my-zone:device> end
zonecfg:my-zone> add device
zonecfg:my-zone:device> set match=/dev/*dsk/c2t40d3*
zonecfg:my-zone:device> set allow-raw-io=true
zonecfg:my-zone:device> end
zonecfg:my-zone> set max-sem-ids=10485200
zonecfg:my-zone> add attr
zonecfg:my-zone:attr> set name=comment
zonecfg:my-zone:attr> set type=string
zonecfg:my-zone:attr> set value="This is my work zone."
zonecfg:my-zone:attr> end
zonecfg:my-zone> verify
zonecfg:my-zone> commit
zonecfg:my-zone> exit

```

次の手順 ゾーン構成を検証するために、次の手順に進みます。

ゾーン構成に問題がなければ、ゾーンをインストールします。137 ページの「ゾーンのインストールとブート」に進みます。

## ▼ 非大域ゾーンの構成を表示する方法

info サブコマンドは、デフォルト値以外の値に明示的に設定されているプロパティのみ表示します。info -a サブコマンドは、使用可能なすべてのプロパティを表示します。

1. ゾーン管理者になります。  
詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。
2. ゾーンのプロパティを表示します。
  - ゾーン内で明示的に設定されているすべてのプロパティを表示します。

```
global$ pfbash zonecfg -z zonename info
```

たとえば、このコマンドは、変更されているゾーンリソースを表示します。

```

global$ pfbash zonecfg -z zone1
zonecfg:zone1> info
zonename: zone1
brand: solaris
autoshutdown: shutdown
anet 0:
linkname: net0
configure-allowed-address: true

```

この出力はエクスポートに役立ちます。例7「ゾーン構成のエクスポート」を参照してください。

- ゾーンのすべてのリソースについて、すべてのプロパティを表示します。

```
global$ pfbash zonecfg -z zonename info -a
```

リストについては、例8「ゾーンのすべてのリソースプロパティを表示する」を参照してください。

#### 例 7 ゾーン構成のエクスポート

この例では、zone1 ゾーンの構成情報を標準出力に出力します。export サブコマンドでは、デフォルト値に一致しているプロパティはエクスポートされません。

```
global$ pfexec zonecfg -z zone1
zonecfg:zone1>export
create -b
set brand=solaris
set autoshutdown=shutdown
add anet
set linkname=net0
set configure-allowed-address=true
end
```

#### 例 8 ゾーンのすべてのリソースプロパティを表示する

この例では、zone1 ゾーンのすべてのリソースについて、すべてのプロパティを表示します。

```
$ pfexec zonecfg -z zone1 info -a
zonename: zone1
zonepath.template: /system/zones/{zonename}
zonepath: /system/zones/zone1
brand: solaris
autoboot: false
autoshutdown: shutdown
bootargs:
file-mac-profile:
pool:
limitpriv:
scheduling-class:
ip-type: exclusive
hostid:
tenant:
boot-priority: high
fs-allowed:
anet:
linkname: net0
lower-link: auto
allowed-address:
configure-allowed-address: true
defrouter:
allowed-dhcp-cids:
link-protection: mac-nospoof
mac-address: auto
auto-mac-address:
mac-prefix:
mac-slot:
vlan-id:
```

```
priority:
rxrings:
txrings:
mtu:
maxbw:
bwshare:
rxfanout:
vsi-typeid:
vsi-vers:
vsi-mgrid:
etsbw-lcl:
cos:
pkey:
linkmode:
evs:
vport:
ring-group: auto
smf-dependency:
fmri: svc:/application/list:default
```

- 次の手順
- 構成を変更する必要がある場合は、[127 ページの「ゾーン構成の変更」](#)に進みます。
  - ゾーン構成に問題がなければ、[137 ページの「ゾーンのインストールとブート」](#)に進みます。

## ゾーン構成の変更

このセクションでは、リソースのクリアーと設定、ゾーンの名前の変更、ゾーンの特権の変更など、ゾーンのプロパティを変更する次の手順について説明します。

### ▼ 例: ゾーン構成内のリソースを変更する方法

この手順では、`zonecfg` コマンドを使用して `rctl` リソースを設定する方法を示します。

**1. ゾーン管理者になります。**

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

**2. 変更するゾーンを選択します。**

```
global$ pfbash zonecfg -z zonename
```

**3. 変更するリソースを選択します。**

```
zonecfg:zonename> select rctl name=zone.cpu-shares
```

4. リソースの現在の値を削除します。

```
zonecfg:zonename:rctl> remove value (priv=privileged,limit=20,action=none)
```

5. リソースの新しい値を追加します。

```
zonecfg:zonename:rctl> add value (priv=privileged,limit=30,action=none)
```

6. 改定された rctl の指定を終了します。

```
zonecfg:zonename:rctl> end
```

7. 変更を確定して終了します。

```
zonecfg:zonename> commit  
zonecfg:zonename> exit
```

確定された変更は、ゾーンの次回ブート時に有効になります。

## ▼ ゾーン構成内のプロパティをクリアする方法

グローバルプロパティまたはリソースタイププロパティを、そのデフォルト値または値なしにリセットするには、この手順を実行します。

1. ゾーン管理者になります。

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. 変更するゾーンを選択します。

```
global$ pfbash zonecfg -z zonename
```

3. リソースプロパティをクリアするには、そのリソーススコープを選択します。

```
zonecfg:zonename> select resource
```

4. 変更するプロパティをクリアします。

- プロパティをそのデフォルト値にリセットするには、clear サブコマンドを使用します。

```
zonecfg:zonename:resource> clear property
```

- プロパティを値なしに変更するには、set サブコマンドを使用します。

```
zonecfg:zonename:resource> set property=""
```

5. リソーススコープ内にいる場合は、スコープから抜けます。



```
zonecfg:zonename:resource> exit
```

## 6. 変更を確定して終了します。

```
zonecfg:zonename> commit
zonecfg:zonename> exit
```

確定された変更は、ゾーンの次回ブート時に有効になります。

## ▼ ゾーンの名前を変更する方法 (zoneadm)

installed または configured 状態の solaris ゾーンの名前を zoneadm コマンドを使用して変更するには、この手順を実行します。

---

**注記** - 名前の変更は、solaris-kz ブランドゾーンではサポートされていません。

---

または、zonecfg コマンドを使用すると、configured 状態の solaris ゾーンの名前を変更できます。130 ページの「[ゾーンの名前を変更する方法 \(zonecfg\)](#)」を参照してください。

---

**注記** - ゾーン構成に %{zonename} トークンが使用されている場合、そのトークンが使用されている場所では新しいゾーン名が自動的に更新されます。ゾーン構成のトークンの詳細は、『[Oracle Solaris Zones Configuration Resources](#)』の「[zonecfg Templates](#)」を参照してください。

---

### 1. ゾーン管理者になります。

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

### 2. 名前を変更するゾーンが running 状態の場合は、ゾーンを停止します。

それ以外の場合は次の手順に進みます。

```
global$ zoneadm -z zonename halt
```

停止されたゾーンは installed 状態になります。

### 3. ゾーンが installed または configured 状態であることを確認します。

```
global$ zoneadm list -v
ID NAME          STATUS    PATH                                BRAND  IP
  0 global        running   /                                    solaris shared
- zonename      configured /system/zones/zonename          solaris excl
```

### 4. ゾーンの名前を変更します。

---

**注記** - installed 状態のゾーンのコンソールはすべて、ゾーンの名前が変更されると切り離されます。

---

```
global$ zoneadm -z zonename rename new-zonename
```

新しい名前はすぐに有効になります。

5. 新しいゾーン名を確認します。

```
global$ zoneadm list -cv
```

## ▼ ゾーンの名前を変更する方法 (zonecfg)

configured 状態の solaris ゾーンの名前を zonecfg ユーティリティを使用して変更するには、この手順を実行します。

---

**注記** - 名前の変更は、solaris-kz ブランドゾーンではサポートされていません。

---

または、zoneadm コマンドを使用すると、installed または configured 状態の solaris ゾーンの名前を変更できます。129 ページの「[ゾーンの名前を変更する方法 \(zoneadm\)](#)」を参照してください。

---

**注記** - ゾーン構成に %{zonename} トークンが使用されている場合、そのトークンが使用されている場所では新しいゾーン名が自動的に更新されます。ゾーン構成のトークンの詳細は、『[Oracle Solaris Zones Configuration Resources](#)』の「[zonecfg Templates](#)」を参照してください。

---

1. ゾーン管理者になります。

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. ゾーンが configured 状態であることを確認します。

```
global$ zoneadm list -civ
ID NAME          STATUS  PATH                                BRAND  IP
0 global         running /                                     solaris shared
- zonename      configured /system/zones/zonename          solaris excl
```

3. 変更するゾーンを選択します。

```
global$ zonecfg -z zonename
```

4. ゾーンの名前を変更します。

```
zonecfg:zonename> set zonename=new-zonename
```

5. 変更を確定して終了します。

```
zonecfg:zonename> commit  
zonecfg:zonename> exit
```

確定された変更は、ゾーンの次回ブート時に有効になります。

## ▼ 専用のデバイスをゾーンに追加する方法

非大域ゾーン構成に専用のデバイスを追加するには、次の手順を実行します。この手順では、例としてスキャン用デバイス `/dev/scsi/scanner/c3t4*` を使用して、`zonecfg` ユーティリティの構文を示しています。

1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

2. 変更するゾーンを選択します。

```
global$ zonecfg -z zonename
```

3. デバイスを追加します。

```
zonecfg:zonename> add device
```

4. デバイス照合を設定します。

```
zonecfg:zonename:device> set match=device-name
```

5. デバイスの指定を終了します。

```
zonecfg:zonename:device> end
```

6. `zonecfg` ユーティリティを終了します。

```
zonecfg:zonename> exit
```

## ▼ 大域ゾーンの CPU シェアを設定する方法

大域ゾーンの CPU シェアを永続的に設定するには、この手順を実行します。

1. ゾーン管理者になります。

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

2. **resource-mgmt サービスが有効になっていることを確認します。**

```
$ svcs svc:/system/resource-mgmt:default
STATE          STIME    FMRI
disabled       Jan_08   svc:/system/resource-mgmt:default
```

このサービスがオンラインでない場合は、有効にしてから再度確認します。

```
$ pfbash svcadm enable resource-mgmt:default
$ svcs svc:/system/resource-mgmt:default
STATE          STIME    FMRI
online         Jan_12   svc:/system/resource-mgmt:default
```

3. **global ゾーンを指定して zonecfg ユーティリティーを起動します。**

```
global$ zonecfg -z global
```

4. **大域ゾーンの CPU シェア数を設定します。**

```
zonecfg:global> set cpu-shares=N
```

5. **変更を確定して終了します。**

```
zonecfg:zonename> commit
zonecfg:zonename> exit
```

確定された変更は、ゾーンの次回ブート時に有効になります。

## ▼ ゾーンの特権を変更する方法

非大域ゾーンの特権を変更するには、この手順を実行します。ゾーンに対して指定できるデフォルトの Oracle Solaris 特権およびオプションの特権は、71 ページの「非大域ゾーン内の特権」に示されています。

1. **ゾーン管理者になります。**

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

2. **変更するゾーンを選択します。**

```
global$ zonecfg -z zonename
```

3. **特権セットを改訂します。**

このエントリ例は、非大域ゾーン内からファイルに immutable、nounlink、または appendonly ファイル属性を設定する機能を追加します。

```
zonecfg:zonename> set limitpriv="default,file_flag_set"
```

#### 4. 変更を確定して終了します。

```
zonecfg:zonename> commit
zonecfg:zonename> exit
```

確定された変更は、ゾーンの次回ブート時に有効になります。

## 既存の共有 IP ゾーンを排他的 IP ゾーンに変更する

共有 IP タイプのゾーンを排他的 IP ゾーンに更新できます。ほとんどの場合は、`zonecfg net` リソースを `anet` リソースで置き換え、`ip-type` を `exclusive` に変更します。

大域ゾーンの IPMP グループの一部であるインタフェースを使用している共有 IP ゾーンがすでに存在する場合は、DLMP アグリゲーションの使用に切り替えてください。アグリゲーションを作成するには、`dladm create-aggr` サブコマンドを使用します。

1. 大域ゾーンで、古い IPMP インタフェース上に DLMP アグリゲーションを作成します。
2. `zonecfg anet` リソースを作成します。ここで、`lower-link` プロパティは DLMP アグリゲーションを指します。

`dladm` の詳細は、[dladm\(8\)](#) のマニュアルページを参照してください。『[Oracle Solaris 12 でのネットワーク管理の計画](#)』も参照してください。

## ゾーン構成を元に戻す、または削除する

アクティブな `zonecfg` セッションにいる間は、ゾーン構成の設定を元に戻すことができます。`configured` 状態にあるゾーンは削除できます。

### ▼ ゾーン構成の変更を元に戻す方法

この手順例は、リソース設定の変更を、まだその `zonecfg` ユーティリティーセッションにいる間に、すぐに取り消す方法を示しています。

始める前に この手順では、すでにゾーン管理者であることと、アクティブな `zonecfg` ユーティリティーセッションにいることを前提としています。

1. `info` と入力して構成を表示します。

```
zonecfg:zonename> info
```

次のような構成の net リソースセグメントが表示されます。

```
...
fs:
    dir: /tmp
    special: swap
    type: tmpfs
net:
    address: 192.0.2.0
    physical: eri0
device
    match: /dev/pts/*
...
```

**2. net リソースアドレスを削除します。**

```
zonecfg:zonename> remove net address=192.0.2.0
```

**3. net リソースのエントリが削除されたことを確認します。**

```
zonecfg:zonename> info
...
fs:
    dir: /tmp
    special: swap
    type: tmpfs
device
    match: /dev/pts/*
...
```

**4. revert と入力し、プロンプトに yes と回答します。**

```
zonecfg:zonename> revert
Are you sure you want to revert (y/[n])? y
```

**5. net リソースアドレスがふたたび存在することを確認します。**

```
zonecfg:zonename> info
...
fs:
    dir: /tmp
    special: swap
    type: tmpfs
net:
    address: 192.0.2.0
    physical: eri0
device
    match: /dev/pts/*
...
```

## ▼ アンインストールされたゾーン構成を削除する方法

configured 状態のゾーン構成をシステムから削除するには、この手順を実行します。

---

**注記** - この手順は、インストール済みまたは実行中のゾーンには使用できません。インストール済みまたは実行中のゾーンを削除するには、代わりに [153 ページの「システムから非大域ゾーンを削除する方法」](#) に進んでください。

---

1. **ゾーン管理者になります。**

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#) を参照してください。

2. **ゾーンが `configured` 状態であることを確認します。**

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND    IP
0   global    running  /                                       solaris  shared
-   zonename  configured /system/zones/zonename              solaris  excl
```

3. **ゾーン構成を削除します。**

次のいずれかの方法を使用します。

- **`zonecfg delete` コマンドのシステムプロンプトに `yes` と回答することによって、ゾーンを対話形式で削除します。**

```
global$ pfbash zonecfg -z zonename delete
Are you sure you want to delete zone zonename (y/[n])? y
```

- **削除を強制的に実行するには、`zonecfg delete` コマンドに `-F` オプションを使用します。**

ゾーンは、先にユーザーに確認を求めることなく、すぐに削除されます。

```
global$ pfbash zonecfg -z zonename delete -F
```





## 非大域ゾーンのインストール、ブート、クローニング、移動、およびアンインストール

---

この章では、非大域ゾーンのインストールおよびブート方法について説明します。クローニングを使って同一のシステムにゾーンをインストールする方法についても説明します。ゾーンの停止、リブート、移動、アンインストールなどのインストールに関連するほかのタスクについて説明します。

この章で扱う内容は、次のとおりです。

- 137 ページの「ゾーンのインストールとブート」
- 146 ページの「同一システム上でのゾーンのクローニングと移動」
- 151 ページの「非大域ゾーンのアンインストールと削除」

ゾーンのインストールおよび関連する操作に関する一般的な情報については、[第2章「非大域ゾーンのインストール、停止処理、停止、アンインストール、クローニングについて」](#)を参照してください。

### ゾーンのインストールとブート

非大域ゾーンのインストールタスクを実行するには、`zoneadm` コマンドを使用します。ゾーンのインストールを実行するには、大域管理者または適切な権利を持つゾーン管理者である必要があります。

このセクションでは、次の手順について説明します。

- 138 ページの「構成済みのゾーンをインストールする方法」
- 141 ページの「インストールされた非大域ゾーンの UUID を取得する方法」
- 142 ページの「`installed` 状態の非大域ゾーンに `unavailable` または `incomplete` のマークを付ける方法」
- 144 ページの「`installed` 状態のゾーンを `ready` 状態に移行する方法」
- 144 ページの「ゾーンのブート方法」
- 145 ページの「ゾーンをシングルユーザーモードでブートする方法」

## ▼ 構成済みのゾーンをインストールする方法

構成済みの非大域ゾーンをインストールするには、この手順を実行します。インストールオプションについては、[48 ページの「ゾーンのインストール方法」](#)を参照してください。

ゾーンのインストール中に、2つの場所にログが記録されます。ログの内容は同じです。

- 大域ゾーンの `/var/log/zones/` ディレクトリ
- インストールされたゾーンの `/system/zones/zonename/root/var/log/zones/` ディレクトリ。

始める前に ゾーンは、構成済みで、独自の ZFS データセット上に存在する必要があります。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. (オプション) 構成済みのゾーンを検証します。

```
global$ zoneadm -z zonename verify
```

共有ストレージ上で構成されているゾーンの場合は、構成されているどの `zpool` リソースもシステム上でまだオンラインになっていないことが検証されます。

- エラーメッセージが表示されない場合は、ゾーンのインストールに進むことができます。
- エラーメッセージが表示され、ゾーンの検証に失敗した場合は、メッセージに従って修正を行い、コマンドを再度実行してください。検証で表示される可能性のあるメッセージの例を次に示します。

```
WARNING: /system/zones/zonename does not exist, so it could not be verified.
When 'zoneadm install' is run, 'install' will try to create
/system/zones/zonename, and 'verify' will be tried again,
but the 'verify' may fail if:
the parent directory of /system/zones/zonename is group or other-writable
or
/system/zones/zonename overlaps with any other installed zones
or
/system/zones/zonename is not a mountpoint for a zfs file system.
```

### 3. 構成済みのゾーンをインストールします。

次のインストール方法のいずれかを使用します。

ゾーンをインストールします。

```
global$ pfbash zoneadm -z zonename install
```

リポジトリからゾーンをインストールします

```
global$ pfbash zoneadm -z zonename install -m manifest -c {profile | dir}
```

アーカイブイメージからゾーンをインストールします。

```
global$ pfbash zoneadm -z zonename install -a archive -s -u
```

ディレクトリからゾーンをインストールします。

```
global$ pfbash zoneadm -z zonename install -d path -p -v
```

このゾーンに対して ZFS ファイルシステムが作成されたことが表示されます。ゾーンのルートファイルシステムに必要なファイルおよびディレクトリがゾーンのルートパスにインストールされる際、さまざまなメッセージが表示されます。zonepath ZFS の ZFS データセットが自動的に作成されます。ゾーンのパスの親ディレクトリもデータセットである必要があり、データセットでない場合はファイルシステムの作成が失敗します。

---

**注記** - ゾーンのインストールが失敗する場合は、この手順の最後の「トラブルシューティング」セクションを参照してください。

---

#### 4. インストールが完了したら、ゾーンが **installed** 状態であることを確認します。

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS      PATH                                     BRAND      IP
0   global    running     /                                         solaris    shared
-   zonename installed  /system/zones/zonename                 solaris    excl
```

#### 例 9 異なるレイヤーにあるプロファイルを使用したゾーンのインストール

この例では、既存のプロファイル zone1.xml および myappconfig.xml を使用してゾーンをインストールします。この例では、ディレクトリ階層 zone1/node および zone1/sysconfig を作成することによって、プロファイルが異なるレイヤーにインストールされるようになります。これにより、通常 sysconfig ユーティリティで管理される構成が sysconfig-profile レイヤーに配置され、ほかの構成は node-profile レイヤーに保持されます。あとで sysconfig unconfigure コマンドがゾーン内で実行された場合、myappconfig.xml プロファイルに指定された構成は削除されませんが、zone1.xml プロファイルに指定された構成は削除されます。

```
global$ pfbash mkdir zone1 zone1/node zone1/sysconfig
global$ mv zone1.xml zone1/sysconfig/zone1.xml
global$ cp myappconfig.xml zone1/node/myappconfig.xml

global$ zoneadm -z zone1 install -c zone1
```

**注意事項** ゾーンのインストールが失敗した場合の回復手順は次のとおりです。

- ゾーンのインストールが中断または失敗した場合、ゾーンは `incomplete` 状態のままになります。 `zoneadm uninstall -F` コマンドを使用して、ゾーンを `configured` 状態にリセットしてください。
- エラーメッセージが表示され、ゾーンのインストールに失敗した場合は、次のステップを実行してください。

### 1. ゾーンの状態を確認します。

次に出力例を示します。

```
global$ zoneadm list -cv
  ID NAME           STATUS    PATH                               BRAND  IP
  0 global          running   /                                   solaris shared
- zonename         configured /system/zones/zonename          solaris excl
```

### 2. 状態が `incomplete` と表示されている場合は、ゾーンをアンインストールします。

```
global$ pfexec zoneadm -z zonename uninstall
```

### 3. エラーメッセージで指定されている修正を行います。

### 4. `zoneadm install` コマンドを再度実行してゾーンをインストールします。

- ストレージオブジェクトに既存のパーティション、zpool、または UFS ファイルシステムが含まれる場合は、`install` 操作が失敗し、エラーメッセージが表示されます。次の手順を実行します。

### 1. ゾーンを `uninstalled` 状態にします。

```
global$ pfexec zoneadm -z zonename uninstall
```

### 2. インストールを続行し、既存のデータを上書きします。

`zoneadm -z zonename install` コマンドで、次のいずれかの形式の `-x` オプションを使用します。`-x` オプションは、`zpool create -f` コマンドに似ています。

```
-x force-zpool-import
-x force-zpool-create=zpoolname [...]
-x force-zpool-create=zpoolname1, zpoolname2, zpoolname3
-x force-zpool-create-all
```

`-x` オプションの使用方法の詳細は、[zoneadm\(8\)](#) のマニュアルページを参照してください。

**参照** この手順では、デフォルトの最小ネットワーク構成を使用してゾーンをインストールします。ゾーンへのログイン時に、オープンなネットワーク構成に切り替えることも、個別のサービスを有効または無効に設定することもできます。デフォルトの最小ネットワーク構成およびサービスの管理については、『[Oracle Solaris 11.4 でのシステムサービスの管理](#)』の第3章、「サービスの管理」を参照してください。

- 次の手順
- (オプション) インストールされた非大域ゾーンの汎用一意識別子を取得するには、[141 ページの「インストールされた非大域ゾーンの UUID を取得する方法」](#)を参照してください。
  - ゾーンが `installed` 状態のまま使用不可になった場合は、回復手順について [142 ページの「installed 状態の非大域ゾーンに `unavailable` または `incomplete` のマークを付ける方法」](#)を参照してください。
  - ゾーンを `ready` 状態に移行するには、[144 ページの「installed 状態のゾーンを `ready` 状態に移行する方法」](#)を参照してください。
  - ゾーンをブートして `running` 状態にするには、[144 ページの「ゾーンのブート方法」](#)または [145 ページの「ゾーンをシングルユーザーモードでブートする方法」](#)を参照してください。

## ▼ インストールされた非大域ゾーンの UUID を取得する方法

ゾーンのインストール時に、汎用一意識別子 (UUID) がゾーンに割り当てられます。インストールされたゾーンの UUID を表示するには、この手順を実行します。

1. **ゾーン管理者になります。**  
詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。
2. **インストールされたゾーンの UUID を表示します。**  
次に出力例を示します。UUID は、そのゾーンについて 5 番目に表示されるフィールドです。

```
global$ pfexec zoneadm list -cp
0:global:running:/::solaris:shared:-:none:
3:zonename:running:/system/volatile/zones/zonename/zonepath:95180a6d-fab2-4363-ee33-81ba6e84a84f:solaris-kz:excl:R:solaris-kz:
...
```

### 例 10 特定のゾーンの UUID を取得する

この例では、ゾーン `test_zone` の UUID を取得し、そのゾーンの UUID のみを表示するように出力を制限します。

```
$ zoneadm list -cp | grep test_zone | cut -f 5 -d:
95180a6d-fab2-4363-ee33-81ba6e84a84f
```

### 例 11 コマンド内で UUID を使用する

この例では、ゾーン `test_zone` の UUID を `zoneadm list` コマンド内で使用します。-u オプションで UUID を指定します。

```
global$ zoneadm -z test_zone -u 95180a6d-fab2-4363-ee33-81ba6e84a84f list -v
ID NAME          STATUS    PATH                                BRAND  IP
3  test_zone      running  -                                    solaris-kz excl
```

zoneadm list コマンド内に -u オプションと -z オプションの両方が存在する場合、最初に UUID に基づいてマッチングが行われます。指定した UUID のゾーンが見つかった場合はそのゾーンが使用され、-z オプションは無視されます。指定した UUID のゾーンが見つからなかった場合、システムはゾーン名で検索を実行します。

## ゾーンの UUID について

ゾーンをアンインストールすることも、同名のゾーンを内容を変えて再インストールすることもできます。ゾーンの内容を変更せずにゾーンの名前を変更することも可能です。こうした理由から、UUID はゾーン名よりも信頼性が高くなります。

参照 詳細は、[zoneadm\(8\)](#) および [libuuid\(3LIB\)](#) を参照してください。

- 次の手順
- ゾーンが installed 状態のまま使用不可になった場合は、回復手順について [142 ページの「installed 状態の非大域ゾーンに unavailable または incomplete のマークを付ける方法」](#) を参照してください。
  - ゾーンを ready 状態に移行するには、[144 ページの「installed 状態のゾーンを ready 状態に移行する方法」](#) を参照してください。
  - ゾーンをブートして running 状態にするには、[144 ページの「ゾーンのブート方法」](#) または [145 ページの「ゾーンをシングルユーザーモードでブートする方法」](#) を参照してください。

## ▼ installed 状態の非大域ゾーンに unavailable または incomplete のマークを付ける方法

システムに加えられた管理上の変更のためにゾーンが使用不可になるか、矛盾が生じた場合、installed 状態のゾーンを、状況に応じて unavailable または incomplete 状態に変更できます。

- たとえば、ストレージへの接続が失われたときなど、ゾーンが一時的に使用不可になった場合は、ゾーンの状態を unavailable に変更します。問題が解決されたあとで、zoneadm attach コマンドを使用してゾーンの状態を installed に戻します。
- たとえば、ルートプールが破損したときなど、ゾーンが永続的に使用不可になった場合は、ゾーンの状態を incomplete に変更します。ゾーンに incomplete のマークが付けられたあとは、ゾーンをアンインストールすることによって configured 状態に戻すことが唯一の選択肢となります。

---

注記 - ゾーンへの incomplete のマーク付けは、取り消すことができません。

---

始める前に ゾーンの状態を unavailable に変更するか incomplete に変更するかを決定します。

**1. ゾーン管理者になります。**

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

**2. ゾーンが installed 状態であることを確認します。**

次に出力例を示します。

```
global$ zoneadm -z zonename list -cv
ID  NAME      STATUS    PATH                                BRAND  IP
-   zonename  installed /system/zones/zonename            solaris excl
```

**3. ゾーンの状態をマークします。**

新しい状態を unavailable または incomplete として指定します。

```
global$ pfexec zoneadm -z zonename mark new-state
```

**4. 状態の変更を確認します。**

次に出力例を示します。

```
global$ zoneadm -z zonename list -cv
ID  NAME      STATUS    PATH                                BRAND  IP
-   zonename  new-state /system/zones/zonename            solaris excl
```

- 次の手順
- ゾーンに unavailable のマークを付けた場合は、ゾーンが使用不可になる原因となった問題を解決します。その後、`zoneadm -z zonename attach` コマンドを実行して、ゾーンを installed 状態に戻します。次に、次のアクションのいずれかを実行します。
    - ゾーンを ready 状態に移行するには、144 ページの「installed 状態のゾーンを ready 状態に移行する方法」を参照してください。
    - ゾーンをブートして running 状態にするには、144 ページの「ゾーンのブート方法」または 145 ページの「ゾーンをシングルユーザーモードでブートする方法」を参照してください。
  - ゾーンに incomplete のマークを付けた場合は、ゾーンをアンインストールして configured 状態にします。151 ページの「ゾーンをアンインストールする方法」を参照してください。

## ▼ installed 状態のゾーンを ready 状態に移行する方法

ゾーンを ready 状態に移行すると、仮想プラットフォームでユーザープロセスの実行を開始する準備が整います。ready 状態のゾーンには、内部で実行中のユーザープロセスはまだ存在しません。

---

注記 - ゾーンをブートしてすぐに使用する場合は、代わりに [144 ページの「ゾーンのブート方法」](#) または [145 ページの「ゾーンをシングルユーザーモードでブートする方法」](#) に進んでください。ゾーンのブート時に、ready 状態への移行が自動的に行われます。

---

1. ゾーン管理者になります。  
詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#) を参照してください。

2. ゾーンが **installed** 状態であることを確認します。  
次に出力例を示します。

```
global$ zoneadm -z zonename list -cv
ID NAME STATUS PATH BRAND IP
- zonename installed /system/zones/zonename solaris excl
```

3. ゾーンを **ready** 状態に移行します。

```
global$ pfexec zoneadm -z zonename ready
```

4. ゾーンの状態を確認します。  
次に出力例を示します。システムにより一意のゾーン ID 1 がゾーンに割り当てられています。

```
global$ zoneadm -z zonename list -cv
ID NAME STATUS PATH BRAND IP
1 zonename ready /system/zones/zonename solaris excl
```

次の手順 ゾーンをブートして running 状態にします。[144 ページの「ゾーンのブート方法」](#) または [145 ページの「ゾーンをシングルユーザーモードでブートする方法」](#) を参照してください。

## ▼ ゾーンのブート方法

ゾーンをブートして running 状態にします。ready 状態または installed 状態からブートできます。ブートした installed 状態のゾーンは、ready 状態から running



状態に透過的に移行します。running 状態のゾーンに対してはゾーンへのログインが可能です。

---

**注記** - 代わりに、ゾーンをブートしてシングルユーザーモードにするには、[145 ページの「ゾーンをシングルユーザーモードでブートする方法」](#)を参照してください。

---

1. **ゾーン管理者になります。**  
詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

2. **ゾーンをブートします。**

```
global$ pfexec zoneadm -z zonename boot
```

3. **ブートが完了したら、ステータスを確認します。**  
次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                BRAND  IP
0   global    running  /                                    solaris shared
1   zonename  running  /system/zones/zonename            solaris excl
```

#### 例 12 大域ゾーンからゾーンのブート引数を指定する

この例では、ゾーン my-zone を大域ゾーンからブートし、-m verbose ブートオプションの使用を指定します。

```
global$ pfexec zoneadm -z my-zone boot -- -m verbose
```

#### 例 13 非大域ゾーンからゾーンのブート引数を指定する

この例では、ゾーン my-zone をゾーン内からリブートし、-m verbose ブートオプションの使用を指定します。

```
my-zone$ reboot -- -m verbose
```

次の手順    ゾーンにログインして初期内部構成を実行する方法については、[第3章「非大域ゾーンへのログインについて」](#)のガイドラインを参照してから、[第7章「非大域ゾーンへのログイン」](#)を参照してください。

## ▼ ゾーンをシングルユーザーモードでブートする方法

---

**注記** - ゾーンをマルチユーザーモードでブートするには、[144 ページの「ゾーンのブート方法」](#)を参照してください。

---

1. **ゾーン管理者になります。**

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. **ゾーンをシングルユーザーモードでブートします。**

```
global$ pfexec zoneadm -z zonename boot -- -s
```

次の手順 ゾーンにログインして初期内部構成を実行する方法については、[第3章「非大域ゾーンへのログインについて」](#) および [第7章「非大域ゾーンへのログイン」](#) を参照してください。

## 同一システム上でのゾーンのクローニングと移動

ソース zonename からターゲット zonename にデータをコピーすることによってシステム上に新しいゾーンをプロビジョニングするには、クローニングを使用します。

クローニング元の zonename とクローニング先の zonename が両方とも ZFS 上にあり、同じプールに含まれる場合、zoneadm clone コマンドは自動的に ZFS を使用してゾーンをクローンします。ただし、ZFS の zonename のコピーは行い、ZFS のクローンは行わないように指定することもできます。

クローニングのみに使用する、ゴールデンゾーンと呼ばれるゾーンを作成できます。クローニングにゴールデンゾーンを使用する利点については、56 ページの「[ゴールデンゾーンについて](#)」を参照してください。

このセクションでは、次の内容について説明します。

- [146 ページの「同一システム上で非大域ゾーンをクローンする方法」](#)
- [148 ページの「ゴールデンゾーンの作成とクローニング」](#)
- [150 ページの「共有ストレージ上にないゾーンを移動する方法」](#)

共有ストレージ上のゾーンの移動については、245 ページの「[共有ストレージゾーン構成に対する既存のゾーンの移動](#)」を参照してください。

### ▼ 同一システム上で非大域ゾーンをクローンする方法

同一システム上にある別のゾーンのクローンである新規ゾーンを作成するには、この手順を実行します。

始める前に [56 ページの「ゾーンのクローニングの再構成要件」](#) の要件を参照してください。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. クローンされるソースゾーンを停止します。

```
global$ pfbash zoneadm -z source-zone halt
```

### 3. 新規ゾーンを作成します。

ソースゾーンをゾーンテンプレートとして指定します。

```
global$ zonecfg -z new-zone 'create -t source-zone'
```

```
-t source-zone
```

新規ゾーンに複製するゾーン構成を指定します。

### 4. 新規ゾーンの構成をソースゾーンからクローンします。

- 新規ゾーンの構成を変更する必要がない場合は、`zoneadm clone` コマンドを使用します。

```
global$ zoneadm -z new-zone clone source-zone
```

- 新規ゾーンの構成を変更する必要がある場合は、`zonecfg` ユーティリティーを使用して、クローン構成にそれらの変更を加えます。

複数のゾーンで同一であってはならないコンポーネントに対して、異なるプロパティとリソースを設定します。共有 IP ゾーンの場合は、各 `net` リソースの IP アドレスを変更する必要があります。排他的 IP ゾーンの場合は、各 `net` リソースの `physical` プロパティを変更する必要があります。

```
global$ zonecfg -z new-zone
zonecfg:new-zone> create -t source-zone
(必要な構成変更をクローンに加えます)
zonecfg:new-zone> exit
```

### 5. ソースゾーンをクローニングして、新規ゾーンをインストールします。

```
global$ zoneadm -z new-zone clone source-zone
Cloning zonepath /system/zones/source-zone...
```

既存のストレージオブジェクトが原因で `clone` 操作が失敗する場合は、クローン操作を強制的に実行します。この手順の最後にある「トラブルシューティング」の手順を参照してください。

### 6. 新規ゾーンが作成されたことを確認します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
0   global    running  /                                       solaris shared
-   source-zone installed /system/zones/source-zone            solaris excl
-   new-zone  installed /system/zones/new-zone                solaris excl
```

**例 14** システム構成プロファイルのクローンされたゾーンへの適用

この例では、ソースゾーン `my-zone` のクローンとして新規ゾーン `zone1` を作成します。コマンドでは、ソースゾーンからエクスポートされた構成ファイル `/path/config.xml` を指定します。

```
$ zoneadm -z zone1 clone -c /path/config.xml my-zone
```

**注意事項** 既存のストレージオブジェクトが原因で `clone` が失敗する場合は、クローン操作を強制的に実行します。

ストレージオブジェクトに既存のパーティション、`zpool`、または `UFS` ファイルシステムが含まれていた場合は、`clone` が失敗し、エラーメッセージが表示されます。操作を続行し、既存のデータを上書きするには、`zoneadm clone` に対して適切な `-x` オプションを使用します。

---

**注記** `--force` サブコマンドを使用する前に、ソースゾーンをアンインストールする必要があります。

---

```
-x force-zpool-import
-x force-zpool-create=zpoolname
-x force-zpool-create=zpoolname1, zpoolname2, zpoolname3
-x force-zpool-create-all
```

- `-x force-zpool-create=` オプションは、`zpool create -f` コマンドに似ています。
- `-x force-zpool-create=zpoolname` オプションは、複数回使用できます。

## ゴールデンゾーンの作成とクローニング

ゴールデンゾーンを作成したあとで、多数のクローンを作成できます。

### ▼ ゴールデンゾーンを作成する方法

既存のゾーンの構成に基づくゴールデンゾーンを作成するには、この手順を実行します。または、[119 ページの「zonecfg ユーティリティを使用してゾーンを構成する方法」](#)に記載されている次の手順によって新規ゾーンを作成します。ただし、そのゾーンをブートしないでください。

---

**注記** - 実際に使用するゾーンをクローンするには、代わりに [146 ページの「同一システム上で非大域ゾーンをクローンする方法」](#)を参照してください。

---

1. ゾーン管理者になります。

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

2. 新規ゾーンを作成し、ソースゾーンをそのテンプレートとして指定します。

```
global$ zonecfg -z myzone-golden 'create -t source-zone'
```

-t source-zone            新規ゾーンに複製するゾーン構成を指定します。

3. ソースゾーンをシャットダウンします。

次のステップに進む前に、ゾーンが running 状態ではなくなったことを確認します。

```
global$ zoneadm -z source-zone shutdown
global$ zoneadm list -cv
ID NAME STATUS PATH BRAND IP
0 global running / solaris shared
- source-zone installed /system/zones/source-zone solaris excl
```

4. ゴールデンゾーンをソースゾーンからクローンします。

- ゴールデンゾーンの構成をソースゾーンから変更する必要がない場合は、zoneadm clone コマンドを使用します。

```
global$ zoneadm -z myzone-golden clone source-zone
```

- ゴールデンゾーンの構成を変更する必要がある場合は、zonecfg ユーティリティを使用して、クローンにそれらの変更を加えます。

複数のゾーンで同一であってはならないコンポーネントに対して、異なるプロパティとリソースを設定します。共有 IP ゾーンの場合は、各 net リソースの IP アドレスを変更する必要があります。排他的 IP ゾーンの場合は、各 net リソースの physical プロパティを変更する必要があります。

```
global$ zonecfg -z myzone-golden
zonecfg:myzone-golden> create -t source-zone
(必要な構成変更をクローンに加えます)
zonecfg:myzone-golden> exit
```

5. ソースゾーンをブートします。

```
global$ zoneadm -z source-zone boot
```

例 15 ゴールデンゾーンを作成する

この例では、ソースゾーン z1 をクローニングすることによってゴールデンゾーン z1-golden を作成します。ゾーン z1 は、クローン操作のために一時的にシャットダウンされたあと、ブートされてサービスに戻ります。ゴールデンゾーンはブートされません。

```
global$ zonecfg -z z1-golden 'create -t z1'
```

```
global$ zoneadm -z z1 shutdown
global$ zoneadm -z z1-golden clone z1
global$ zoneadm -z z1 boot
```

次の手順 ゴールデンゾーンに基づく新規ゾーンを作成するには、[150 ページの「ゴールデンゾーンをクローンする方法」](#)を参照してください。

## ▼ ゴールデンゾーンをクローンする方法

ゴールデンゾーンのクローンを作成するには、この手順を実行します。ゴールデンゾーンはブートされることがないため、そのクローンを作成する前にゴールデンゾーンを停止する必要はありません。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. ゴールデンゾーンに基づく新規ゾーンを構成します。

```
global$ zonecfg -z new-zone 'create -t golden-zone'
```

-t *golden-zone*            新規ゾーンに複製するゾーン構成を指定します。

### 3. 新規ゾーンをゴールデンゾーンからクローンします。

```
global$ zoneadm -z new-zone clone golden-zone
```

#### 例 16 ゴールデンゾーンのクローニング

この例では、ゴールデンゾーン z1-golden のクローニングによってゾーン z2 を作成します。

```
global$ zonecfg -z z2 'create -t z1-golden'
global$ zoneadm -z z2 clone z1-golden
```

## ▼ 共有ストレージ上にないゾーンを移動する方法

zonepath を変更してゾーンを同じシステムの新しい場所に移動するには、この手順を実行します。ゾーンは、先に停止する必要があります。zonepath の通常の基準については、『[Oracle Solaris Zones Configuration Resources](#)』の「[Zone Global Properties](#)」で説明されています。

共有ストレージ上にないゾーンを移動するときは、次の制限を守ってください。

- ほかの BE 内に存在しているゾーンを移動することはできません。最初にそれらの BE を削除でき、またそのゾーンをクローニングすることによって新しいパスに新しいゾーンを作成することもできます。
- rootzpool リソースを使用するゾーン構成の場合、zonepath プロパティはデフォルトの zonepath テンプレート値に基づき、不変で固定されています。zonepath を取得されたデフォルト値とは異なる値に手動で設定することは、このゾーン構成では禁止されています。

共有ストレージ上のゾーンを移動するには、代わりに [245 ページの「共有ストレージゾーン構成に対する既存のゾーンの移動」](#) を参照してください。

#### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#) を参照してください。

#### 2. 移動するゾーンを停止します。

```
global$ zoneadm -z zonename halt
```

#### 3. ゾーンを新しいゾーンパスに移動します。

```
global$ zoneadm -z zonename move /new-zonepath/zonename
```

#### 4. パスを検証します。

```
global$ zoneadm list -cv
ID  NAME      STATUS   PATH
0   global    running /
-   zonename  installed /new-zonepath/zonename
                                solaris  excl
```

## 非大域ゾーンのアンインストールと削除

ゾーンのアンインストールおよび非大域ゾーンの削除は、どちらもゾーンを破棄します。アンインストールが中断した場合は、ゾーンを再構築できる可能性があります。

### ▼ ゾーンをアンインストールする方法

稼働中でないゾーンをアンインストールするには、この手順を実行します。uninstall 操作は、稼働中のゾーンに対しては無効です。



- ゾーンのルートファイルシステム内のファイルすべてを削除したあとで、操作を元に戻すことはできません。
- `zonepath` として独自の ZFS ファイルシステムを保持しているゾーンをアンインストールすると、その ZFS ファイルシステムは破棄されます。

1. ゾーン管理者になります。

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. ゾーンが **running** 状態でないことを確認します。

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
 0  global   running  /                                       solaris shared
-  zonename installed /system/zones/zonename                solaris excl
```

3. ゾーンをアンインストールします。

必要に応じて `-F` オプションを含めると、アクションをすぐに強制的に実行できます。`-F` オプションが指定されていない場合、システムにより確認を求めるメッセージが表示されます。

```
global$ zoneadm -z zonename uninstall
```

4. ゾーンが **installed** 状態ではなくなったことを確認します。

次に出力例を示します。

```
global$ zoneadm -z zonename list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
-  zonename configured /system/zones/zonename                solaris excl
```

- 注意事項
- ゾーンをアンインストールが中断した場合、ゾーンは `incomplete` 状態のままになります。`zoneadm uninstall` コマンドを使用して、ゾーンを `configured` 状態にリセットしてください。
  - `zonepath` が削除されない場合、そのゾーンが別のブート環境内にインストールされていることを示している可能性があります。特定の `zonepath` を保持するインストール済みゾーンを持つブート環境が存在するとき、`zonepath` やその `zonepath` データセット内に存在するさまざまなデータセットは削除されません。ブート環境の詳細は、[beadm\(8\)](#) のマニュアルページを参照してください。



## ▼ システムから非大域ゾーンを削除する方法

システムからゾーンを完全に削除するには、この手順を実行します。ゾーンはどの状態でもかまいません。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. ゾーンの状態を確認します。

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS   PATH                                BRAND  IP
0   global    running  /                                   solaris shared
-   zonename running  /system/zones/zonename            solaris excl
```

- ゾーンの状態が `shutting_down` または `down` の場合は、状態遷移の完了を待ってから、このステップを繰り返します。
- ゾーンの状態が `running` の場合は、次のステップに進みます。
- ゾーンの状態が `installed`、`ready`、`unavailable`、または `incomplete` の場合は、[ステップ 4](#) にスキップします。
- ゾーンの状態が `configured` またはそれより低い状態の場合は、[ステップ 5](#) にスキップします。

### 3. ゾーンをシャットダウンします。

```
global$ zoneadm -z zonename shutdown
zonename
```

### 4. ゾーンをアンインストールします。

このステップは、ゾーンのルートファイルシステムを削除します。

```
global$ zoneadm -z zonename uninstall [-F]
```

アクションを強制的に実行する必要がある場合は、`-F` オプションを使用します。`-F` オプションが使用された場合、コマンドは確認を求めずに処理を続行します。

### 5. ゾーン構成を削除します。

```
global$ zonecfg -z zonename delete [-F]
```

アクションを強制的に実行する必要がある場合は、`-F` オプションを使用します。`-F` オプションが使用された場合、コマンドは確認を求めずに処理を続行します。

6. 削除されたゾーンがリストに表示されなくなったことを確認します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH
   0  global    running  /
                                BRAND  IP
                                solaris shared
```

## 非大域ゾーンへのログイン

---

この章では、インストール済みのゾーンの構成を完了して大域ゾーンからゾーンにログインし、ゾーンを停止処理する手順について説明します。また、この章では、`zonename` コマンドを使用して現在のゾーンの名前を出力する方法についても説明します。

- [156 ページの「ゾーン構成のためにゾーンコンソールにログインする方法」](#)
- [157 ページの「ゾーンコンソールへのログイン方法」](#)
- [157 ページの「対話型モードを使用してゾーンにアクセスする方法」](#)
- [158 ページの「非対話型モードを使用してゾーンにアクセスする方法」](#)
- [158 ページの「フェイルセーフモードを使用してゾーンに入る方法」](#)
- [159 ページの「現在のゾーンの名前の表示」](#)
- [159 ページの「非大域ゾーンの終了」](#)

ゾーンへのログイン処理の概要については、[第3章「非大域ゾーンへのログインについて」](#)を参照してください。

## ゾーンへのログイン

大域ゾーンから実行中状態または `ready` 状態にある任意のゾーンにログインするには、`zlogin` コマンドを使用します。詳細は、[`zlogin\(1\)` のマニュアルページ](#)を参照してください。

次の手順で説明されているように、ゾーンへのログインはさまざまな方法で実行できます。[65 ページの「ゾーンへのリモートログイン」](#)で説明されているように、リモートでログインすることも可能です。

## ▼ ゾーン構成のためにゾーンコンソールにログインする方法

システム構成プロファイルが `zoneadm clone`、`attach`、または `install` コマンドに渡されると、この構成プロファイルがシステムを構成するために使用されます。`clone`、`attach`、または `install` 操作時にプロファイルが提供されなかった場合、ゾーンの初回のブートで、コンソールで対話式システム構成 (SCI) ツールが起動します。

システム構成プロファイルによって構成されていないゾーンをブートし、構成するには、この手順を実行します。

---

**注記** - 構成情報用の最初のプロンプトが表示されないことを避けるために、この手順では、ゾーンが 2 回目のセッションでブートされる前に、`zlogin` が実行されるように 2 つの端末ウィンドウを使用します。

---

1. **ゾーン管理者になります。**  
詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. **ゾーンにログインします。**

```
global$ pfexec zlogin -C zonename
```

3. **別の端末ウィンドウからゾーンをブートします。**

```
global$ pfbash zoneadm -z zonename boot
```

次のような出力が、`zlogin` 端末ウィンドウに表示されます。

```
[NOTICE: Zone booting up]
```

4. **新たにインストールしたゾーン用の構成パラメータに関する一連の質問に回答します。**

パラメータには、システムのホスト名、タイムゾーン、ユーザーと `root` アカウント、およびネームサービスが含まれます。SCI ツールは、デフォルトでは `/system/volatile/scit_profile.xml` に SMF プロファイルファイルを生成します。

5. **ブート処理の完了後、ゾーンが `running` 状態になったことを確認します。**

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
0   global    running  /                                         solaris shared
1   zonename  running  /system/zones/zonename                 solaris excl
```

**注意事項** 最初の SCI ツール画面が表示されない場合、`Enter` または `Ctrl-L` を入力して、SCI ツール画面をリフレッシュします。

次の手順    ゾーン構成が完了したら、構成のバックアップコピーを作成します。ゾーンを再作成する必要がある場合、バックアップを使用できます。例37「ゾーン構成のコピーの出力」に進みます。

## ▼ ゾーンコンソールへのログイン方法

構成済みゾーンのコンソールにログインするには、このタスクを実行します。

---

注記 - 構成が実行されていない場合、`zlogin` コマンドと `-c` オプションを使用すると対話式システム構成 (SCI) ツールが開始されます。代わりに、[156 ページの「ゾーン構成のためにゾーンコンソールにログインする方法」](#)に進み、ゾーンを構成します。

---

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. ゾーンのコンソールにログインします。

```
global$ pfexec zlogin -c -d zonename
```

### 3. ゾーンコンソールが表示されたら、root 役割を引き受けます。

```
zonename console login: root
Password:
zonename$
```

## ▼ 対話型モードを使用してゾーンにアクセスする方法

対話型モードでは、ゾーン内部で使用する新しい仮想端末が割り当てられます。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. 大域ゾーンから、ゾーンにログインします。

```
global$ pfexec zlogin zonename
```

次のような出力が表示されます。

```
[Connected to zone 'zonename' pts/2]
Last login: Wed Jul 3 16:25:00 on console
```

3. **exit** と入力して、接続を閉じます。

次のようなメッセージが表示されます。

```
[Connection to zone 'zonename' pts/2 closed]
```

## ▼ 非対話型モードを使用してゾーンにアクセスする方法

ゾーン内部で実行されるコマンドを指定すると、非対話型モードが有効になります。非対話型モードでは、新しい仮想端末は割り当てられません。

---

注記 - コマンドおよびコマンドの処理対象のファイルは、いずれも NFS 上に存在できません。

---

1. **ゾーン管理者になります。**

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. **大域ゾーンからゾーンにログインして、コマンド名を入力します。**

```
global$ zlogin zonename command  
          コマンド出力
```

- 例 17 非対話型ログインを使用したゾーンへのアクセスとゾーンでのコマンドの発行

この例では、単一の `zlogin` コマンドを発行して、`my-zone` ゾーンにログインし、`zoneadm` コマンドを実行して、現在のゾーンの名前を表示します。

```
global$ pfbash zlogin my-zone zonename  
my-zone
```

## ▼ フェイルセーフモードを使用してゾーンに入る方法

ゾーンへの接続が拒否された場合、`zlogin` コマンドと `-s` オプションを使用して、ゾーン内の最小環境に入ることができます。

1. **ゾーン管理者になります。**

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. **-s オプションを使用して、ログインします。**

```
global$ pfexec zlogin -s my-zone
```

## 現在のゾーンの名前の表示

`zonename` コマンドを発行して、現在のゾーンの名前を表示します。次に、大域ゾーン内で `zonename` コマンドを実行した場合の出力例を示します。

```
global$ zonename
global
```

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

## 非大域ゾーンの終了

接続に適切なコマンドを使用して、非大域ゾーンから切断します。

- ゾーンの非仮想コンソールを終了します。

```
zonename$ exit
```

- ゾーン of 仮想コンソールから切断します。

チルド (~) 文字とピリオドを入力します。次に出力例を示します。

```
zonename$ ~.
[Connection to zone 'zonename' pts/6 closed]
```

- Secure Shell (ssh) セッションを終了します。

また、ssh のデフォルトのエスケープシーケンスも ~ で、これは ssh セッションを終了させます。

```
zonename$ ~~.
```





## 非大域ゾーンのライブゾーン再構成

---

非大域ゾーンの実行中にそのゾーンのライブ構成を再構成したり、そのゾーンのライブ構成に関して報告したりするには、ライブゾーン再構成を使用します。

この章で扱う内容は、次のとおりです。

- 161 ページの「ライブゾーン再構成の実行」
- 165 ページの「データセットのライブゾーン再構成」

solaris ゾーンおよびカーネルゾーンのリソースおよびプロパティに対するライブゾーン再構成のサポートを示す表を表示するには、『Oracle Solaris ゾーン構成リソース』の「Live Zone Reconfiguration Support or Restriction for Resource Types and Global Properties」を参照してください。

### ライブゾーン再構成の実行

このセクションでは、一般的なライブゾーン再構成タスクを実行する次の手順について説明します。

- 161 ページの「実行中のゾーンのライブ構成を検査する方法」
- 162 ページの「ライブゾーン構成の影響をプレビューする方法」
- 162 ページの「ライブゾーン構成の永続的な構成変更を行う方法」
- 163 ページの「実行中のゾーンに一時的な変更を加える方法」
- 164 ページの「ライブゾーン構成の変更のコミット中の障害から回復する方法」

#### ▼ 実行中のゾーンのライブ構成を検査する方法

実行中のゾーンの構成を表示し、エクスポートするにはこの手順を実行します。

1. 管理者になります。  
詳細は、68 ページの「ゾーン管理者向けの権利プロファイルの概要」を参照してください。

2. ライブゾーン構成に関する情報を表示します。

```
global$ pfbash zonecfg -z zonename -r info
```

3. (オプション) ライブ構成をエクスポートします。

```
global$ zonecfg -z zonename -r export -f exported.cfg
```

## ▼ ライブゾーン構成の影響をプレビューする方法

ライブゾーン構成の変更の最終コミットメントを行う前に、それらの変更を確認するには、この手順を実行します。

始める前に 次のドキュメントを確認してください。

- [36 ページの「ライブゾーン再構成の予行演習」](#)
- 『Oracle Solaris Zones Configuration Resources』の付録 A, 「Resource Types and Global Properties That Support Live Zone Reconfiguration,」

1. 管理者になります。

詳細は、[68 ページの「ゾーン管理者向けの権利プロファイルの概要」](#)を参照してください。

2. zonecfg ユーティリティーをライブモードで起動し、目的のゾーン変更を構成します。

```
global$ pfbash zonecfg -z zonename -r
zonecfg:zonename>          ゾーン構成の変更を行います
```

3. 再構成によって実行されるアクションを表示します。

-n オプションによりゾーンの変更の実際のコミットメントが妨げられます。

```
zonecfg:zonename> commit -n
```

次の手順 プレビューした変更をライブゾーン構成に適用するには、-n オプションを使用せずに、zonecfg commit コマンドを発行します。

## ▼ ライブゾーン構成の永続的な構成変更を行う方法

ゾーンのリブート後でも維持されるライブゾーン構成の変更を行うには、この手順を実行します。

始める前に 次のドキュメントを確認してください。

- [34 ページの「zonecfg ユーティリティー編集モード」](#)

- 36 ページの「ライブゾーン構成の永続的な変更」
- 『Oracle Solaris Zones Configuration Resources』の付録 A, 「Resource Types and Global Properties That Support Live Zone Reconfiguration,」

1. 管理者になります。

詳細は、68 ページの「ゾーン管理者向けの権利プロファイルの概要」を参照してください。

2. デフォルトモードでゾーンを変更します。

```
global$ pfbash zonecfg -z zonename "zonecfg-commands"
```

1 つ以上の zonecfg サブコマンドを指定します。

形式については、例18「ライブ構成の永続的な変更」および zonecfg(8) のマニュアルページを参照してください。

3. ライブ構成に変更を適用します。

```
global$ pfbash zoneadm -z zonename apply
```

例 18 ライブ構成の永続的な変更

この例では、zonecfg コマンドを指定して、cpu-shares リソースを設定し、pool リソース設定をクリアーして、実行中のゾーン zone1 に anet リソースを追加し、変更を永続的ゾーン構成に適用しています。

```
global$ pfbash zonecfg -z zone1 "set cpu-shares=4;clear pool;add anet;"
global$ zoneadm -z zone1 apply
zone1: Checking: set property cpu-shares=4
zone1: Checking: clear property pool
zone1: Checking: add anet linkname=myanet0
zone1: Applying changes
```

## ▼ 実行中のゾーンに一時的な変更を加える方法

実行中のゾーンのライブ構成を一時的に変更してから、永続的な構成をリストアし、変更を元に戻すには、この手順を実行します。

始める前に 次のドキュメントを確認してください。

- 34 ページの「zonecfg ユーティリティー編集モード」
- 36 ページの「ライブゾーン構成の一時的な変更」
- 『Oracle Solaris Zones Configuration Resources』の付録 A, 「Resource Types and Global Properties That Support Live Zone Reconfiguration,」

1. 管理者になります。

詳細は、[68 ページ](#)の「[ゾーン管理者向けの権利プロファイルの概要](#)」を参照してください。

**2. ゾーン構成を変更します。**

このサンプルコマンドは、ライブモードでディスクを追加し、コマンド出力を表示します。

---

**注記** - `zonecfg commit` コマンドは必要ありません。 `zonecfg` ユーティリティーは、コマンドの終了時に変更をコミットします。

---

```
global$ pfbash zonecfg -z zonename -r 'add device; set match=/dev/rdisk/cNtXdY*;end;'  
zone zonename: Checking: Adding device match=/dev/rdisk/cNtXdY*  
zone zonename: Applying the changes
```

**3. (オプション) 構成の変更が不要になった場合は、ゾーンを永続的ゾーン構成に戻します。**

このサンプルコマンドは一時的な構成変更を削除します。

```
global$ zoneadm -z zonename apply  
zone 'zonename': Checking: Removing device match=/dev/rdisk/cNtXdY*  
zone 'zonename': Applying changes
```

または、ゾーンをリブートしてライブゾーン構成の変更を破棄し、永続的ゾーン構成に戻すことができます。

**注意事項** `commit` 操作がエラーを報告した場合、[164 ページ](#)の「[ライブゾーン構成の変更のコミット中の障害から回復する方法](#)」を参照してください。

## ▼ ライブゾーン構成の変更のコミット中の障害から回復する方法

ライブゾーン構成の編集集中に、実行中のゾーンの構成が外部で変更されることがあります。この競合が発生すると、`zonecfg commit` コマンドがエラーを返します。

ゾーン構成を再ロードしてエラーを修正し、更新されたバージョンを表示して再度編集するには、この手順を実行します。

**1. 管理者になります。**

詳細は、[68 ページ](#)の「[ゾーン管理者向けの権利プロファイルの概要](#)」を参照してください。

**2. `reload` サブコマンドを発行し、正しい構成変更を行い、変更をコミットします。**

このステップでは、まだ一時的な構成変更のコミットに失敗した `zonecfg` セッション中であるとしています。

```
zonecfg:zonename> reload
zonecfg:zonename> temporary-configuration-changes
zonecfg:zonename> commit
```

#### 例 19 失敗した一時的ゾーン構成変更からの回復

次の例では、一時的な構成変更、変更が失敗したことのエラーメッセージ、回復ステップ、変更が成功したことを確認する出力を示しています。

```
ライブモードでの構成変更の例
global$ pfbash zonecfg -z zone1 -r
zonecfg:zone1> add anet;set linkname=anet1;set lower-link=net1;end
zonecfg:zone1> commit
zone1: error: Failed to commit. The live configuration of the zone
'zone1' changed externally.

構成を再ロードします
zonecfg:zone1> reload
Are you sure you want to reload (y/[n])? y

以前に試みた構成変更を繰り返します
zonecfg:zone1> add anet;set linkname=anet1;set lower-link=net1;end
zonecfg:zone1> commit

コマンド出力に構成変更が成功したことが示されます
zone1: Checking: add anet linkname=anet1
zone1: Applying changes
```

## データセットのライブゾーン再構成

次の場合に、データセットのライブゾーン再構成を使用します。

- 大域ゾーンで、実行中の非大域ゾーンから dataset リソースを削除するため。
- 非大域ゾーンで、新しい仮想 zpool をインポートする場合。zpool は、ライブゾーン再構成によって dataset リソースを追加した帯域ゾーン管理者が、ゾーンで使用できます。

このセクションでは、次の内容について説明します。

- [165 ページの「ZFS データセットを実行中のゾーンに追加する方法」](#)
- [167 ページの「委任されたデータセットを実行中のゾーンに追加する方法」](#)
- [168 ページの「実行中のゾーンから dataset リソースを削除する方法」](#)
- [169 ページの「solaris ゾーンからデータセットをエクスポートする方法」](#)

### ▼ ZFS データセットを実行中のゾーンに追加する方法

ZFS データセットを実行中のゾーンに追加するには、この手順を実行します。

---

注記 - この手順のすべてのコマンドは大域ゾーンから発行します。

---

1. 管理者になります。

詳細は、68 ページの「ゾーン管理者向けの権利プロファイルの概要」を参照してください。

2. 現在のデータセット情報を表示します。

```
global$ pfbash zonecfg -z zonename info dataset
dataset:
  name: rpool/export/dataset1
  alias: dataset1
dataset:
  name: rpool/export/dataset2
  alias: dataset2
```

3. ゾーンの現在のライブ構成とデータセットを検査します。

```
global$ zonecfg -z zonename -r
zonecfg:zonename> info dataset
dataset 0:
  name: rpool/export/dataset2
  alias: dataset2
dataset 1:
  name: rpool/export/dataset1
  alias: dataset1
zonecfg:zonename> exit
```

4. 新しいデータセットを作成します。

```
global$ zfs create rpool/export/new-dataset
global$ zfs unmount rpool/export/new-dataset
```

5. 新しい dataset リソースを実行中の solaris ゾーンに追加します。

追加を確認し、変更をコミットして、zonecfg ユーティリティーを終了します。

```
global$ zonecfg -z zonename -r
データセットを追加します
zonecfg:zonename> add dataset
zonecfg:zonename:dataset> set name=rpool/export/new-dataset
zonecfg:zonename:dataset> end
追加したデータセットを確認します
zonecfg:zonename> info dataset
dataset 0:
  name: rpool/export/dataset2
  alias: dataset2
dataset 1:
  name: rpool/export/dataset1
  alias: dataset1
dataset 2:
  name: rpool/export/new-dataset
  alias: new-dataset
変更をコミットします
zonecfg:zonename> commit
zone 'zonename': Checking: Adding dataset name=rpool/export/new-dataset
zone 'zonename': Checking: Adding dataset alias=new-dataset
```

```
zone 'zonename': Applying the changes
ユーティリティを終了します
zonecfg:zonename> exit
```

---

注記 - データセットはまだ永続的な保存されたゾーン構成に追加されず、ライブゾーン構成にのみ追加されます。

---

6. 永続的なゾーン構成をライブゾーン構成に一致するように変更します。

```
global$ zoneadm -z zonename apply
```

7. 新しいデータセットが永続的なゾーン構成に追加されたことを確認します。

```
global$ zonecfg -z zonename info dataset
dataset:
  name: rpool/export/new-dataset
  alias: new-dataset
dataset:
  name: rpool/export/dataset1
  alias: dataset1
dataset:
  name: rpool/export/dataset2
  alias: dataset2
```

## ▼ 委任されたデータセットを実行中のゾーンに追加する方法

委任されたデータセットを実行中のゾーンに追加するには、この手順を実行します。

1. 管理者になります。

詳細は、[68 ページの「ゾーン管理者向けの権利プロファイルの概要」](#)を参照してください。

2. 新しいデータセットを実行中のゾーンに追加します。

```
global$ pfbash zonecfg -z zonename -r
zonecfg:zonename> add dataset
zonecfg:zonename:dataset> set name=rpool/export/new-dataset
zonecfg:zonename:dataset> end
zonecfg:zonename> commit
zone 'zonename': Checking: Adding dataset name=rpool/export/new-dataset
zone 'zonename': Applying the changes
zonecfg:zonename> exit
```

3. 新しいデータセットがライブゾーン構成に追加されたことを確認します。

```
global$ zonecfg -z zonename -r info dataset
dataset:
  name: rpool/export/new-dataset
  alias: new-dataset
...
```

4. (オプション) 永続的なゾーン構成をライブゾーン構成に一致するように変更します。

```
global$ zoneadm -z zonename apply
```

## ▼ 実行中のゾーンから dataset リソースを削除する方法

実行中のゾーンから ZFS データセットを削除するには、この手順を実行します。

1. 管理者になります。

詳細は、[68 ページ](#)の「ゾーン管理者向けの権利プロファイルの概要」を参照してください。

2. ライブゾーン構成のデータセット情報を表示します。

```
global$ pfbash zonecfg -z zonename -r info dataset
...
dataset:
  name: rpool/export/dataset1
  alias: dataset1
dataset:
  name: rpool/export/dataset2
  alias: dataset2
...
```

3. 実行中のゾーンからデータセットを削除し、変更を確認して、変更をライブゾーン構成にコミットします。

```
global$ zonecfg -z zonename -r
zonecfg:zonename> remove dataset name=rpool/export/dataset1
zonecfg:zonename> info dataset
dataset 0:
  name: rpool/export/dataset2
  alias: dataset2
...
zonecfg:zonename> commit
zone 'zonename': Checking: Removing dataset name=rpool/export/dataset1
zone 'zonename': Applying the changes
zonecfg:zonename> exit
```

4. ライブゾーン構成のデータセット情報を表示します。

データセットがライブゾーン構成から削除されたことを確認します。

```
global$ zonecfg -z zonename -r info dataset
dataset:
  name: rpool/export/dataset2
  alias: dataset2
```

5. (オプション) 永続的なゾーン構成をライブゾーン構成に一致するように変更します。

```
global$ zoneadm -z zonename apply
```



## ▼ solaris ゾーンからデータセットをエクスポートする方法

この手順は、移行や移動のためにデータセットをエクスポートします。データセットが使用中の場合、エクスポートは成功しないことがあります。詳細については、『Oracle Solaris 11.4 での ZFS ファイルシステムの管理』の「ZFS ストレージプールをエクスポートする」を参照してください。

### 1. 管理者になります。

詳細は、68 ページの「ゾーン管理者向けの権利プロファイルの概要」を参照してください。

### 2. (オプション) エクスポート可能なゾーンデータセットを表示します。

#### ■ 大域ゾーンからのゾーンデータセットを表示します。

```
global$ pfbash zonecfg -z zonename info dataset
dataset:
  name: rpool/export/dataset1
  alias: dataset1
dataset:
  name: rpool/export/dataset2
  alias: dataset2
```

#### ■ ゾーンにログインし、データセット情報を表示します。

```
$ zlogin zonename
zonename$ zpool list
NAME      SIZE  ALLOC   FREE   CAP  DEDUP  HEALTH  ALTRROOT
dataset1  68G   21.5G  46.5G  31%  1.00x  ONLINE  -
dataset2  68G   21.5G  46.5G  31%  1.00x  ONLINE  -
rpool     68G   21.5G  46.5G  31%  1.00x  ONLINE  -

zonename zfs list -t all|grep ds
dataset1                               31K  45.4G   31K  /dataset1
dataset2                               31K  45.4G   31K  /dataset2
zonename mount -v|grep ds
dataset1 on /dataset1 type zfs read/write/setuid/nodevices/rstchown/nonbmand/exec/xattr/atime/
zone=zonename/sharezone=1/dev=3350020 on Thu Jun 11 08:40:30 2015
dataset2 on /dataset2 type zfs read/write/setuid/nodevices/rstchown/nonbmand/exec/xattr/atime/
zone=zonename/sharezone=1/dev=3350021 on Thu Jun 11 08:40:30 2015
```

### 3. 大域ゾーンから、データセットをエクスポートします。

```
global$ zpool export dataset1
```

### 4. 実行中の zpool を一覧表示します。

エクスポートされたデータセットは表示されません。

```
global$ zpool list
```

NAME	SIZE	ALLOC	FREE	CAP	DEDUP	HEALTH	ALTROOT
<i>dataset2</i>	68G	21.5G	46.5G	31%	1.00x	ONLINE	-
rpool	68G	21.5G	46.5G	31%	1.00x	ONLINE	-

## Oracle Solaris ゾーンの移行

---

ゾーンの移行は、特定のホストシステムから既存のゾーンを別のシステム上のゾーンに転送します。この章では、solaris ゾーンを移行するために使用するコールドゾーン移行方法について説明します。

この章で扱う内容は、次のとおりです。

- 171 ページの「非大域ゾーンの移行を実行するための root 以外のユーザーの承認」
- 173 ページの「コールド移行を使用した solaris ゾーンの移行」
- 175 ページの「使用不可能なシステムからのゾーンの移行」
- 176 ページの「共有ストレージを使用していない非大域ゾーンのアーカイブと移動」

### 非大域ゾーンの移行を実行するための root 以外のユーザーの承認

大域ゾーン管理者は root 以外のユーザーに移行を実行する権利を割り当てられます。権利プロファイルの名前と詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

個々のゾーンに対してプロファイルとそれらに関連付けられる承認を制限するには、ゾーン構成で、admin リソースに auths プロパティを設定します。admin リソースの auths プロパティを auths=migrate に設定して、ユーザーがゾーンに対してすべてのタイプの移行を実行できるようにするか、または auths=migrate.cold に設定して、コールド移行とウォーム移行のみを有効にします。詳細については、『Oracle Solaris Zones Configuration Resources』の「admin Resource Type」を参照してください。

---

**注記** - ゾーン構成で auths プロパティを設定すると、ユーザーに適切なプロファイルが自動的に有効になります。

---

## ▼ ユーザーが個々のゾーンのクールド移行を実行することを承認する方法

ゾーン移行管理者に特定のゾーンのクールド移行を委任するには、この手順を実行します。

始める前に この手順では、ユーザーにゾーン構成を作成、変更、および削除する権利と承認がすでに割り当てられているものとします。例21「ユーザーがシステム上のゾーンを構成することの承認」を参照してください。

### 1. root 役割になります。

詳細は、『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

### 2. admin リソースの auths および user プロパティを設定します。

```
global$ zonecfg -z zonename
zonecfg:zonename> add admin
zonecfg:zonename:admin> set user=username
zonecfg:zonename:admin> set auths=migrate.cold
zonecfg:zonename:admin> end
zonecfg:zonename> commit
```

`username` はこのゾーンに対してのみ承認されています。

### 3. 非大域ゾーンおよび大域ゾーンのゾーン管理者の権利を確認します。

この例では、`jdoe` はユーザーで、ゾーン名は `zone1` です。

```
global$ zonecfg -z zone1 info admin
admin:
    user: jdoe
    auths: migrate.cold

global$ auths jdoe
solaris.admin.wusb.read, solaris.mail.mailq, solaris.network.autoconf.read, solaris.zone.migrate.cold/zonename

global$ profiles jdoe
jdoe:
Zone Cold Migration
Basic Solaris User
All
```

#### 例 20 ユーザーがシステム上のすべてのゾーンを移行することの承認

この例では、ユーザー `jdoe` が、ソースシステム上の任意のゾーンのクールド移行を実行するための承認を設定します。

```
global$ usermod -P +"Zone Migration" -A +solaris.zone.migrate jdoe
```

承認とプロファイルを確認します。

```
global$ auths jdoe
solaris.admin.wusb.read,solaris.mail.mailq,solaris.network.autoconf.read,solaris.zone.
migrate
global$ profiles jdoe
jdoe:
Zone Migration
Basic Solaris User
All
```

#### 例 21 ユーザーがシステム上のゾーンを構成することの承認

この例では、ユーザー `jdoe` に、任意のゾーン構成の作成、変更、削除に必要なプロファイルと承認を割り当て、次に割り当てを確認します。この割り当ては、ユーザーがコールド移行を実行するために必要です。

```
global$ usermod -P +"Zone Configuration" -A +solaris.zone.config jdoe

global$ auths jdoe
solaris.admin.wusb.read,solaris.mail.mailq,solaris.network.autoconf.read,solaris.zone.
config

global$ profiles jdoe
jdoe:
Zone Configuration
Basic Solaris User
All
```

## コールド移行を使用した solaris ゾーンの移行

コールド移行を使用して、共有ストレージを使用するように構成されている solaris ゾーンを移行できます。コールド移行では、`installed` 状態の実行中でないゾーンは切り離され、リブートする準備ができていない別のホストに移動され、接続されます。

### ▼ solaris ゾーンをコールド移行する方法

共有ストレージを使用する solaris ゾーンをコールド移行するには、この手順を実行します。

---

**注記** - ゾーンがローカルストレージを使用している場合は、コールド移行を使用してゾーンを移行できません。代わりに、統合アーカイブを使用してゾーンを移行します。[176 ページの「統合アーカイブを使用して非大域ゾーンを新しいシステムに移動する方法」](#)に進みます。

---

- 始める前に
- [98 ページの「非大域ゾーンの移行および変換」](#)で、コールド移行の要件を確認します。

- ゾーンをコールド移行するために必要な権利と権限があることを確認します。171 ページの「[非大域ゾーンの移行を実行するための root 以外のユーザーの承認](#)」を参照してください。

1. ゾーン管理者になります。

詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。

2. 移行するゾーンが実行中の場合は、シャットダウンします。

```
source-host$ pfbash zoneadm -z zonename shutdown
```

3. (オプション) 状態が `installed` であることを確認します。

例:

```
global$ zoneadm -z zone1 list -v
ID NAME          STATUS   PATH   BRAND   IP
- zone1         installed -      solaris excl
```

4. (オプション) `ssh://URI` を使用して、ターゲットホストに接続する予定がある場合は、**Secure Shell** のプロンプトなし認証をテストします。

ターゲットホストで、ssh 経由で `date` などのコマンドを実行します。

```
global1$ ssh global2 date
Mon Mar 9 13:22:40 PDT 2015
```

パスワードの入力を求められた場合は、対話型認証のないログインを有効にするように鍵ペアを構成していません。

『[Oracle Solaris 12 での Secure Shell アクセスの管理](#)』の「[How to Generate a Public/Private Key Pair for Use With Secure Shell](#)」を参照してください。

5. (オプション) 移行の予行演習を実行して、条件が適切に設定されていることを確認します。

```
source-host$ zoneadm -z zonename migrate -n ssh://user@target-host
```

例:

```
global$ zoneadm -z zone1 migrate -n ssh://global2
zoneadm: zone 'zone1': Importing zone configuration.
zoneadm: zone 'zone1': Attaching zone.
zoneadm: zone 'zone1': Dry-run migration successful.
zoneadm: zone 'zone1': Cleaning up.
```

6. 移行を実行します。

```
source-host$ zoneadm -z zonename migrate ssh://user@target-host
```

例:

```
global$ zoneadm -z zone1 migrate ssh://global2
```

```
zoneadm: zone 'zone1': Importing zone configuration.
zoneadm: zone 'zone1': Attaching zone.
zoneadm: zone 'zone1': Migration successful.
```

## 7. (オプション) ターゲットホストでゾーンをブートします。

```
target-host$ zoneadm -z zonename boot
```

### 例 22 ローカルストレージによるコールド移行の予行演習の失敗

この例では、ゾーンが実行していないことを確認しています。ゾーン zone2 でローカルストレージが使用されているために、コールド移行の予行演習が失敗します。

```
root@global3 $ zoneadm list -cv
ID NAME          STATUS      PATH                                BRAND  IP
0  global         running    /                                  solaris shared
-  zone2          installed  -                                  solaris excl
root@global3 $ zoneadm -z zone2 migrate -n ssh://global5
zoneadm: zone 'zone2': configuration check failed: The storage property
dev:/dev/zvol/dsk/rpool/VARSHARE/zones/zone2/disk0 is not a shared storage URI.
```

### 例 23 構成の変更後のコールド移行の成功

この例では、ゾーン zone2 のゾーン構成からのブートしていないローカルストレージデバイスの削除とその後の移行の成功を示しています。使用しているゾーン構成は移行中のゾーンからのもので、ターゲットホストにインポートされます。

```
root@global3 $ zonecfg -z zone2 'remove device id=1;commit;exit'
root@global3 $ zoneadm -z zone2 migrate ssh://global5
zoneadm: zone 'zone2': Importing zone configuration.
zoneadm: zone 'zone2': Attaching zone.
zoneadm: zone 'zone2': Migration successful.
```

## 使用不可能なシステムからのゾーンの移行

非大域ゾーンをホストするシステムが使用不可能になる場合があります。ただし、ゾーンがまだ使用可能な共有ストレージ上にある場合は、ゾーンの zonepath を新しいホストに移動して、まだゾーンを新しいホストに正常に移行できる可能性があります。

SAN などのいくつかの場合には、zonepath データを実際には移動できないこともあります。SAN の場合は、zonepath が新規ホストに表示されるように、再構成が実行されるだけです。

使用不可能なゾーンが適切に切り離されなかったため、まず zonecfg コマンドを使用して、新しいホスト上にゾーンを作成する必要があります。新しいホストにゾーンが作成されたら、ゾーンをホストに接続できます。

## 共有ストレージを使用していない非大域ゾーンのアーカイブと移動

solaris ゾーンがそのゾーンデータに共有ストレージを使用していない場合、`zoneadm migrate` コマンドを使用して移行できません。代わりに、ゾーンをアーカイブしてから、新しいホストにゾーンを配備します。

### ▼ 統合アーカイブを使用して非大域ゾーンを新しいシステムに移動する方法

共有ストレージ上にない solaris ゾーンを変換するには、この手順を実行します。

---

注記 - ゾーンが共有ストレージ上に構成されている場合は、[245 ページの「共有ストレージゾーン構成に対する既存のゾーンの移動」](#)に進みます。

---

1. 管理者になります。  
詳細は、[68 ページの「ゾーン管理者向けの権利プロファイルの概要」](#)を参照してください。
2. 移行するソースシステムの非大域ゾーンの回復用のアーカイブを作成します。  

```
source$ pfbash archiveadm create -r -z zonename zone-archive-name.uar
```
3. ソースシステムから非大域ゾーンをアンインストールするか、ゾーンの `autoboot` プロパティを `false` に設定します。
  - ゾーンをアンインストールするには、次のコマンドを実行します。  

```
source$ zoneadm -z zonename uninstall
```
  - `autoboot` プロパティを無効にするには、次のコマンドを実行します。  

```
source$ zonecfg -z zonename set autoboot=false
```
4. ターゲットホストでアーカイブを使用して、非大域ゾーンを構成します。  

```
target$ pfbash zonecfg -z zonename create -a /path/zone-archive-name.uar
```
5. アーカイブから、ターゲットホストにゾーンをインストールします。  

```
target$ zoneadm -z zonename install -a zone-archive-name.uar
```
6. ターゲットホストでインストールしたゾーンをブートします。



```
target$ zoneadm -z zonename boot
```

参照 統合アーカイブの作成および配備の詳細は、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』の第2章、「[Working With Unified Archives](#)」を参照してください。



## システムの Oracle Solaris ゾーン への変換

---

この章では、ゾーンの変換について説明し、物理システムをゾーンに変換し、異なるタイプのゾーン間で変換する方法を示します。ゾーンの変換は、次のプロセスで使用されます。

- 大域ゾーンから非大域ゾーンへの変換 (物理から仮想 (P2V) への変換とも呼ばれます)
- 非大域ゾーンから大域ゾーンへの変換
- solaris ゾーンのカーネルゾーンへの変換

変換には Oracle Solaris 統合アーカイブを使用したデータ転送が含まれます。ゾーン移行とともに、アーカイブを使用したゾーンの変換は、サーバー統合、作業負荷の再バランス、障害回復のために既存のシステムを移動する計画の一部になることがあります。

バックグラウンド情報については、次を確認してください。

- [103 ページの「ゾーンの変換について」](#)
- 『[Introduction to Oracle Solaris Zones](#)』 の 「[About Zone Migration](#)」

### Oracle Solaris 大域ゾーンの solaris 非大域ゾーンへの変換

このセクションでは、統合アーカイブを使用して、既存の Oracle Solaris 11.4 システムを Oracle Solaris 11.4 システム上の solaris ブランドゾーンに変換する手順を説明します。

- [180 ページの「zonep2vchk によるソースシステムのチェック方法」](#)
- [180 ページの「システムイメージのアーカイブを作成する方法」](#)
- [182 ページの「ターゲットシステムで変換済みゾーンを構成する方法」](#)
- [183 ページの「ターゲットシステムにアーカイブから変換済みゾーンをインストールする方法」](#)

## ▼ zonep2vchk によるソースシステムのチェック方法

ソースシステムでこの手順を実行します。

始める前に [103 ページの「ゾーンの変換について」](#) および [105 ページの「zonep2vchk ツールを使用した大域から非大域への変換準備」](#) で計画情報とガイドラインを確認します。

1. **ゾーン管理者になります。**  
詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#) を参照してください。

2. **変換によって影響を受ける可能性がある使用中の Oracle Solaris 機能をチェックする基本分析を実行します。**

-b オプションは基本分析を指定します。

```
source$ zonep2vchk -b
```

3. **アプリケーションファイルの静的分析を実行します。**

-s オプションは静的分析を指定します。このオプションには、パスをコンマ区切りのリストとして指定します。次のコマンド例は、ELF バイナリの 2 つのアプリケーションパスで、ゾーン内の操作に影響を与える可能性があるシステムコールとライブラリコールがないかどうかを検査します。

```
source$ zonep2vchk -s /opt/myapp/bin,/opt/myapp/lib
```

4. **ゾーン内で正常に実行できなかったプロセスを探す実行時検査を実行します。**

-r オプションは実行時検査を指定します。次のコマンド例は 2 時間の実行時検査を実行します。

```
source$ zonep2vchk -r 2h
```

5. **テンプレート zonecfg 構成を生成します。**

指定するテンプレート名は、.config 拡張子で終わる必要があります。

```
source$ zonep2vchk -c > /path/filename.config
```

この構成には、ソースホストの物理リソースとネットワーク構成に基づくリソース制限とネットワーク構成が含まれます。

次の手順 [180 ページの「システムイメージのアーカイブを作成する方法」](#) に進みます。

## ▼ システムイメージのアーカイブを作成する方法

solaris ゾーンに変換する大域ゾーンのファイルシステムをアーカイブするには、この手順を実行します。

**1. ゾーン管理者になります。**

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

**2. ソースシステムに、カーネルゾーンを含む非大域ゾーンがインストールされていないことを確認します。**

非大域ゾーンを含む回復用のアーカイブは、solaris ゾーンへのインストールに失敗することがあります。

```
global$ zoneadm list -cv
  ID NAME           STATUS    PATH           BRAND    IP
  0 global          running  /              solaris  shared
```

システムに非大域ゾーンまたはカーネルゾーンがある場合は、[153 ページの「システムから非大域ゾーンを削除する方法」](#)の手順に従って、それらを削除します。

**3. ネットワークストレージ上に大域ゾーンの回復用のアーカイブを作成します。**

作成する回復用のアーカイブの完全修飾パスを指定して、ファイル名を .uar 拡張子で終わらせます。

```
source$ pfbash archiveadm create -r -z global recovery-archive.uar
```

**参照** archiveadm コマンドで生成した統合アーカイブには、変換済みゾーンの開始ポイントとして機能するゾーン構成が含まれます。ターゲットシステムに最適になるようにゾーンを構成するために、追加の構成が必要になることがあります。

詳細については、『[Oracle Solaris 12 でのシステム復旧とクローン](#)』および [archiveadm\(8\)](#) のマニュアルページを参照してください。

**注意事項**

**ヒント** - 一部の配備では、オペレーティングシステムおよびアプリケーションのデータが個別の ZFS プールに分別されます。この場合、統合アーカイブ以外のバックアップと回復用のシステムが、アプリケーションのデータの保護に使用されます。ルートのみアーカイブを作成する必要がある場合があります。ルートのみアーカイブには、ルートプール (一般に rpool) のみが含まれます。アクションは、ルートプールでない各 ZFS プールに --exclude-dataset を使用する場合と同じです。

```
source$ pfbash archiveadm create -z global --recovery --root-only /archives/server1.uar
```

**次の手順** [182 ページの「ターゲットシステムで変換済みゾーンを構成する方法」](#)に進みます。

## ▼ ターゲットシステムで変換済みゾーンを構成する方法

zonep2vchk ツールによって生成されたテンプレート zonecfg スクリプトは、ソースシステムの構成のうち、ターゲットシステムのゾーン構成でサポートされる必要がある部分を定義します。ゾーンを完全に構成するには、この手順を実行して、ターゲットシステムに固有の追加情報を提供します。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. アーカイブからゾーンを構成します。

```
target$ zonecfg -z zonename
Use 'create' to begin configuring a new zone.
zonecfg:zonename> create -a /path/config-file.uar
zonecfg:zonename> info
```

### 3. ゾーンのストレージリソースを追加します。

このコマンド例は、rootzpool の共有ストレージを構成する方法を示しています。これは統合アーカイブを使用して配備されるゾーンの構成時に推奨されます。詳細については、[第14章「共有ストレージ上の Oracle Solaris ゾーン」](#)を参照してください。

```
zonecfg:zonename> add rootzpool
zonecfg:zonename:rootzpool> add storage iscsi://zfssa/luname.naa.600144F0DBF8AF19000052E820D60003
zonecfg:zonename:rootzpool> end
```

zonecfg ユーティリティーをまだ終了しないでください。

### 4. ターゲットシステムの全域ゾーンから、現在のリンク構成を表示して、それが正しいことを確認します。

```
target$ dladm show-link
target$ dladm show-phys
target$ ipadm show-addr
```

---

**注記** - デフォルトでは、zonecfg テンプレートはソースシステム上で構成されたすべての物理ネットワークインタフェースに対して anet リソースを使用してネットワーク構成を定義します。ターゲットシステムは、ゾーンのブート時に各 anet リソースに対して自動的に VNIC を作成します。VNIC を使用することで、複数のゾーンが同じ物理ネットワークインタフェースを共有できるようになります。anet リソースの下位リンク名は、zonecfg コマンドによって *change-me* に初期設定されます。このフィールドは手動でターゲットシステム上のいずれかのデータリンクの名前に設定する必要があります。VNIC の下位リンクとして有効な任意のリンクを指定できます。

---

### 5. ゾーン構成を確定して終了します。

```
zonecfg:zonename> commit
zonecfg:zonename> exit
```

次の手順 [183 ページの「ターゲットシステムにアーカイブから変換済みゾーンをインストールする方法」](#)に進みます。

## ▼ ターゲットシステムにアーカイブから変換済みゾーンをインストールする方法

変換済みゾーンをインストールするには、この手順を使用します。この手順では、インストール中に元のシステム構成は変更されません。

1. **ゾーン管理者になります。**  
詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。
2. ソースシステム上に作成した統合アーカイブファイルを使用してゾーンをインストールします。

```
target$ zoneadm -z zonename install -a config-file.uar
```





## Oracle Solaris ゾーン管理

---

この章では、一般的な管理タスクおよびその使用例を示します。

- 185 ページの「ゾーン管理者への制限された権利の割り当て」
- 186 ページの「非大域ゾーンへの DTrace 特権の追加」
- 187 ページの「ゾーンのシャットダウン、停止、およびリブート」
- 189 ページの「ゾーンの委任リスタータの管理」
- 190 ページの「実行中の非大域ゾーンでのファイルシステムのマウント」
- 193 ページの「大域ゾーン内で非大域ゾーンでの特定のファイルシステムへのアクセスを追加する」
- 194 ページの「非大域ゾーンの SMF サービスのステータスの報告」
- 195 ページの「非大域ゾーンのリソースの使用状況の報告」
- 198 ページの「ゾーンごとのファイルシステム統計情報の報告」
- 199 ページの「非大域ゾーンでの IP ネットワークマルチパスの使用」
- 200 ページの「排他的 IP 非大域ゾーンでのデータリンクの管理」
- 201 ページの「ゾーンがインストールされているシステムでの公平配分スケジューラの使用」
- 202 ページの「ゾーンのバックアップと再作成」

一般的なゾーン管理のトピックについては、[第4章「Oracle Solaris ゾーン管理について」](#)を参照してください。

### ゾーン管理者への制限された権利の割り当て

このセクションでは、権利と承認をユーザーに割り当てることで、各ユーザーが実行できるアクセスとアクションおよび各ユーザーがそれらを実行できるゾーンを制限する方法について説明します。ゾーンの権利の割り当てに関する詳細およびガイドラインについては、[68 ページの「ゾーン管理者向けの権利プロファイルの概要」](#)を参照してください。

システム上で実行されているアプリケーションを保護する Oracle Solaris の機能については、『[Oracle Solaris 11.4 Security and Hardening Guidelines](#)』の「[Protecting and Isolating Applications](#)」を参照してください。

## ▼ ゾーン管理者の権利を割り当ててゾーンへのアクセスを制限する方法

特定のゾーンのアカウントに管理権利を割り当てるには、この手順を実行します。

- 始める前に
- [68 ページの「ゾーン管理者向けの権利プロファイルの概要」](#)のガイドラインを確認してください。
  - 大域ゾーンに役割名またはユーザー名が存在することを確認してください。

1. ゾーン管理者として、大域ゾーンから、指定されたゾーン内で管理権利が割り当てられるユーザーまたは役割を指定します。

```
global$ pfbash zonecfg -z zonename
zonecfg:zonename> add admin
zonecfg:zonename:admin> set user=account-name
zonecfg:zonename:admin> set auths=login,manage
zonecfg:zonename:admin> end
zonecfg:zonename> commit
```

2. 非大域ゾーン内で *account-name* に必要な承認が付与されていることを確認します。

```
global$ zonecfg -z zone info admin
admin:
    user: account-name
    auths: login,manage
```

これで、指定されたゾーンに *account-name* アカウントでログインし、そのゾーンを管理できるようになります。

- 次の手順
- 認証された権利プロファイルの割り当て方法や役割を作成して割り当てる方法などの例については、次を参照してください。

- 『[Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護](#)』の「[ユーザーへの権利の割り当て](#)」
- [68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)

## 非大域ゾーンへの DTrace 特権の追加

非大域ゾーンに DTrace 特権を追加すると、[79 ページの「非大域ゾーン内での DTrace の実行」](#)で説明しているように DTrace 機能を使用できるようになります。

ヒント - root よりも権利が制限されているユーザーとして DTrace を実行するには、適切なゾーンの権利プロファイルだけでなく DTrace ツールキットの権利プロファイルも割り当てます。

## ▼ 非大域ゾーンに DTrace 特権を追加する方法

1. root 役割になります。  
詳細は、『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。
2. 大域ゾーンから、`dtrace_proc` 特権と `dtrace_user` 特権を非大域ゾーンに追加します。  
要件に応じて、どちらか一方の特権を追加することも、両方の特権を追加することもできます。

```
global# zonecfg -z zonename
zonecfg:zonename> set limitpriv="default,dtrace_proc,dtrace_user"
zonecfg:zonename> exit
```

3. ゾーンをブートします。  
`global# zoneadm -z zonename boot`
4. ゾーンにログインします。  
`global# pfexec zlogin zonename`
5. DTrace プログラムを実行できることを確認します。  
`zonename# dtrace -l`

## ゾーンのシャットダウン、停止、およびリブート

ゾーンの停止またはゾーンのシャットダウンでは、システムの停止およびシャットダウンと同じコマンドを使用します。違うのは、ゾーンの停止およびシャットダウンが名前で行われることです。詳細およびガイドラインについては、53 ページの「ゾーンの停止処理、停止、リブート、およびアンインストールについて」を参照してください。

## ▼ ゾーンをシャットダウンする方法

ゾーンを正常にシャットダウンするには、この手順を実行します。

始める前に [53 ページの「ゾーンのシャットダウンについて」](#) のガイドラインを確認してください。

1. **ゾーン管理者になります。**

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#) を参照してください。

2. **システムで実行中のゾーンを一覧表示します。**

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
 0  global    running  /                                       solaris shared
 1  zonename  running  /system/zones/zonename                solaris  excl
```

3. **ゾーンをシャットダウンします。**

- ユーザーにログオフするよう警告せずにゾーンをシャットダウンするには、`zoneadm shutdown` コマンドを使用します。

```
global$ pfbash zoneadm -z zonename shutdown
```

- システムがシャットダウンされる前にローカルユーザーとリモートユーザーにログオフするよう警告するには、`zlogin shutdown` コマンドを使用します。

```
global$ pfbash zlogin zonename shutdown -g grace-period -i0
```

4. **(オプション) システムで実行中のゾーンを一覧表示し、ゾーンがシャットダウンされていることを確認します。**

```
global$ zoneadm list -cv
```

## ▼ ゾーンを停止およびリブートする方法

ゾーンを停止し、ゾーンのアプリケーション環境と仮想プラットフォームの両方を削除するには、この手順を実行します。

---

**注記** - ゾーンを正常にシャットダウンするには、[187 ページの「ゾーンをシャットダウンする方法」](#) に進みます。

---

始める前に [53 ページの「ゾーンの停止について」](#) のガイドラインを確認してください。

1. **ゾーン管理者になります。**

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#) を参照してください。

## 2. システムで実行中のゾーンを一覧表示します。

次に出力例を示します。

```
global$ pfbash zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
0   global    running  /                                       solaris shared
1   zonename  running  /system/zones/zonename                solaris excl
```

## 3. ゾーンを停止します。

```
global$ zoneadm -z zonename halt
```

## 4. ゾーンが **running** 状態でなくなったことを確認します。

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
0   global    running  /                                       solaris shared
-   zonename  installed /system/zones/zonename                solaris excl
```

ゾーンが正しく停止しない場合は、[218 ページ](#)の「**ゾーンが停止しない**」でトラブルシューティングのヒントを参照してください。

## 5. ゾーンをリブートします。

```
global$ pfbash zoneadm -z zonename reboot
```

## 6. ゾーンが **running** 状態であることを確認します。

次に出力例を示します。

```
global$ zoneadm list -cv
ID  NAME      STATUS    PATH                                     BRAND  IP
0   global    running  /                                       solaris shared
2   zonename  running  /system/zones/zonename                solaris excl
```

---

**注記** - 出力には、ゾーン ID が変更されていることが示されます。通常、リブートするとゾーン ID は変更されます。

---

# ゾーンの委任リスタータの管理

このセクションでは、ゾーンの委任リスタータを管理する手順について説明します。ゾーンの委任リスタータについては、[23 ページ](#)の「**ゾーンの委任リスタータ**」、および [svc.zones\(8\)](#) のマニュアルページを参照してください。すべてのリスタータに共通した情報については、[smf\\_restarter\(7\)](#) のマニュアルページを参照してください。

- `svc:/system/zones:default` が非大域ゾーン用のリスタータであるかどうかを表示するには、`svcs -R` コマンドを使用します。

```
global$ svcs -R svc:/system/zones:default
STATE      STIME      FMRI
online     12:11:12  svc:/system/zones/zone:zonename
```

- ゾーンの委任リスタータのプロパティを設定するには、次のコマンドを使用します。

```
global$ pfbash svccfg -s svc:/system/zones:default setprop config/property=value
global$ svcadm refresh svc:/system/zones:default
```

ゾーンの委任リスタータのプロパティは次のとおりです。

`config/concurrent-boot-shutdown`

ゾーンで同時に実行されるブートまたはシャットダウンの最大数。

`config/concurrent-suspend-resume`

ゾーンで同時に実行される中断操作または再開操作の最大数。

ゾーンの委任リスタータのプロパティに関する詳細は、[svc.zones\(8\)](#) のマニュアルページを参照してください。

## 実行中の非大域ゾーンでのファイルシステムのマウント

実行中の非大域ゾーン内でファイルシステムをマウントするために適切な権利を割り当てる必要があります。これで、raw デバイスとブロック型デバイスを非大域ゾーンにインポートできます。権利の詳細は、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

それらのデバイスがインポートされたら、ディスクにアクセスできます。その後、ディスク上に新しいファイルシステムを作成して、次のいずれかのアクションを実行できます。

- ファイルシステムを手動でマウントします
- ファイルシステムがゾーンのブート時にマウントされるように、`/etc/vfstab` 内に配置します

大域ゾーンのファイルシステムを非大域ゾーンにマウントすることもできます。

大域ゾーンのファイルシステムを非大域ゾーンにマウントする前に、その非大域ゾーンが `ready` 状態になっているか、ブートされる必要があります。それ以外の場合、次回ゾーンをブートまたは準備完了状態にしようとするとき失敗します。また、大域ゾーンから非大域ゾーンにマウントされたファイルシステムは、ゾーンが停止されるときにアンマウントされます。

## ▼ LOFS を使用してファイルシステムをマウントする方法

LOFS マウントを使用して大域ゾーンと非大域ゾーンの間でファイルシステムを共有するには、この手順を実行します。

1. ゾーン管理者になります。

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

2. `zonecfg` ユーティリティーを起動します。

```
global$ pfexec zonecfg -z zonename
```

3. ファイルシステムを構成に追加します。

```
zonecfg:zonename> add fs
```

4. ファイルシステムのマウントポイントを設定します。

```
zonecfg:zonename:fs> set dir=/mount-point
```

5. 大域ゾーンのファイルシステムがそのゾーンにどのようにマウントされるかを指定します。

```
zonecfg:zonename:fs> set special=/export/mount-point
```

6. LOFS ファイルシステムタイプを設定します。

```
zonecfg:zonename:fs> set type=lofs
```

7. 指定を終了します。

```
zonecfg:zonename:fs> end
```

8. 構成を検証および確定して、ユーティリティーを終了します。

```
zonecfg:zonename> verify  
zonecfg:zonename> commit  
zonecfg:zonename> exit
```

## ▼ ZFS データセットを非大域ゾーンに委任する方法

ZFS データセットを非大域ゾーンに委任するには、この手順を実行します。

## 1. ゾーン管理者になります。

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

## 2. 大域ゾーンから、新しい ZFS ファイルシステムを作成します。

```
global$ pfbash zfs create pool/file-system
```

## 3. (オプション) ファイルシステムの mountpoint プロパティを設定します。

```
global$ zfs set mountpoint=/path/file-system pool/file-system
```

mountpoint の設定は必須ではありません。mountpoint プロパティを指定しない場合、データセットはデフォルトでゾーン内の /alias にマウントされます。mountpoint プロパティと canmount プロパティをデフォルト以外の値にすると、zfs(8) のマニュアルページで説明されているように、この動作が変更されます。

## 4. このファイルシステムの mountpoint プロパティのソースが、現在 local であることを確認します。

```
global$ zfs get mountpoint pool/file-system
NAME          PROPERTY  VALUE          SOURCE
pool/file-system  mountpoint /path/file-system  local
```

## 5. このファイルシステムを委任するか、別名が付いたデータセットを指定します。

## ■ ファイルシステムをゾーンに委任するには、次のコマンドを使用します。

```
$ zonecfg -z zonename
zonecfg:zonename> add dataset
zonecfg:zonename:dataset> set name=pool/file-system
zonecfg:zonename:dataset> end
```

## ■ 別名が付いたデータセットを指定するには、次のコマンドを使用します。

```
$ zonecfg -z zonename
zonecfg:zonename> add dataset
zonecfg:zonename:dataset> set name=pool/file-system
zonecfg:zonename:dataset> set alias=delegated
zonecfg:zonename:dataset> end
```

## 6. ゾーンをリブートして、プール内のすべてのファイルシステムの zoned プロパティを表示します。

```
global$ zfs get -r zoned pool
NAME          PROPERTY  VALUE  SOURCE
pool          zoned    off    default
pool/file-system  zoned    on     default
```

pool/file-system の zoned プロパティが on に設定されています。この ZFS ファイルシステムは、非大域ゾーンに委任されてゾーンにマウントされ、ゾーン管理者の制御下にあります。ZFS は、zoned プロパティを使用して、データセットがある時点で非大域ゾーンに委任されたことを示します。



## 大域ゾーン内で非大域ゾーンでの特定のファイルシステムへのアクセスを追加する

非大域ゾーンの CD または DVD メディアへの読み取り専用アクセスを追加できます。CD または DVD のコンテンツを非大域ゾーンで利用できるようにするためにループバックファイルシステム `lofs` が使用されます。それらのメディアは非大域ゾーンにマウントされます。その後、CD または DVD を使用して製品を非大域ゾーンにインストールできます。この手順では、`jes_05q4_dvd` という DVD を使用します。

### ▼ 非大域ゾーンの CD または DVD メディアへのアクセスを追加する方法

非大域ゾーンのメディアへのアクセスを追加するには、この手順を実行します。

1. **root** 役割になります。

詳細は、『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. メディアを挿入します。
3. 大域ゾーンから、ドライブにメディアが入っているかどうかを確認します。

```
global# volcheck
```

4. DVD が自動マウントされているかどうかをテストします。

```
global# ls /cdrom
```

次に出力例を示します。

```
cdrom  cdrom1  software_dvd
```

5. **ro,nodevices** オプションを使用してファイルシステムをループバックマウントします。

これらのマウントオプションは、非大域ゾーン用にファイルシステムを読み取り専用、デバイスなしでマウントするよう指定します。

```
global# zonecfg -z zonename
zonecfg:zonename> add fs
zonecfg:zonename:fs> set dir=/cdrom
zonecfg:zonename:fs> set special=/cdrom
zonecfg:zonename:fs> set type=lofs
zonecfg:zonename:fs> add options [ro,nodevices]
zonecfg:zonename:fs> end
```

```
zonecfg:zonename> commit
zonecfg:zonename> exit
```

6. 非大域ゾーンをリブートします。

```
global# zoneadm -z zonename reboot
```

7. ステータスを確認します。

次に出力例を示します。

```
global# zoneadm list -v
ID  NAME      STATUS    PATH                                     BRAND  IP
0   global    running   /                                       solaris shared
1   zonename  running   /system/zones/zonename               solaris  excl
```

8. 非大域ゾーンにログインします。

```
global# zlogin zonename
```

9. DVD-ROM がマウントされているかを確認します。

次に出力例を示します。

```
zonename$ ls /cdrom
cdrom  cdrom1  software_dvd
```

10. (オプション) 製品のインストールガイドの説明に従って、DVD からソフトウェア製品をインストールします。

11. 非大域ゾーンから抜けます。

```
zonename$ exit
global#
```

---

ヒント - /cdrom ファイルシステムを非大域ゾーンに残すこともできます。マウントするときには、常に CD-ROM ドライブの現在の内容が反映されます。つまり、ドライブが空の場合は、ディレクトリは空になります。

---

12. (オプション) /cdrom ファイルシステムを非大域ゾーンから削除します。

```
global# zonecfg -z zonename
zonecfg:zonename> remove fs dir=/cdrom
zonecfg:zonename> commit
zonecfg:zonename> exit
```

## 非大域ゾーンの SMF サービスのステータスの報告

非大域ゾーンのステータスを、そのゾーンにログインしているときに確認したり、ログイン後に確認したりできます。

**例 24** ログイン時に非大域ゾーンの SMF サービスのステータスを確認する

このコマンドでは、管理者としてログインしているときに、ゾーン `my-zone` に含まれているすべてのサービス (無効なサービスも含む) を表示します。

```
global$ zlogin my-zone svcs -a
```

**例 25** ゾーン内から SMF サービスのステータスを確認する

この例では、管理者としてゾーン `my-zone` にログインしたあと、すべてのサービス (無効なサービスを含む) を表示します。

```
global$ zlogin my-zone
my-zone$ svcs -a
```

## 非大域ゾーンのリソースの使用状況の報告

`zonestat` ユーティリティでは、現在実行中のゾーンの CPU、メモリー、ネットワーク、およびリソース制御の利用状況について報告します。このユーティリティの詳細は、[zonestat\(1\)](#) のマニュアルページを参照してください。

`zonestat` のネットワークコンポーネントでは、PHYS、AGGR、Etherstub、および SIMNET データリンク上の仮想ネットワーク (VNIC) リソースの使用量がゾーン別に表示されます。ブリッジやトンネルなど、ほかのデータリンクに関する情報を入手するには、[dladm\(8\)](#) および [dlstat\(8\)](#) のマニュアルページで説明されているネットワークユーティリティを使用します。

ゾーンの統計情報を表示するために、非大域ゾーン内ですべての `zonestat` オプションおよびリソースタイプを呼び出すことができます。

```
root@zoneA:~# zonestat -z global -r physical-memory 2
```

---

**注記** - 非大域ゾーン内の `zonestat` のユーザーは、システムリソースを使用しているほかのゾーンを認識しません。`zonestat` ユーティリティが非大域ゾーン内で使用されると、ほかのすべてのゾーン (大域ゾーンを含む) のリソース使用の総量が、大域ゾーンで使用されたものとして報告されます。

---

次の各例では、`my-zone` にログインし、`my-zone` のゾーン管理権利が割り当てられているものとします。

- [例26「非大域ゾーンの CPU とメモリーの利用状況の表示」](#)
- [例27「非大域ゾーンのデフォルトのプロセッサセットの報告」](#)
- [例28「非大域ゾーンの総使用量と最大使用量の報告」](#)

例29「大域ゾーンからの排他的 IP ゾーンのネットワーク帯域幅の表示」では、大域ゾーンからいくつかのゾーンに関するレポートを作成します。リソースの使用状況に関するレポートの詳細は、[zonestat\(1\)](#) のマニュアルページを参照してください。

例 26 非大域ゾーンの CPU とメモリーの利用状況の表示

```
my-zone$ zonestat -z global -r physical-memory 5
Collecting data for first interval...
Interval: 1, Duration: 0:00:05
PHYSICAL-MEMORY          SYSTEM MEMORY
mem_default              2046M
                          ZONE  USED %USED  CAP  %CAP
                          [total] 1020M 49.8%  -  -
                          [system] 782M 38.2%  -  -
                          global 185M 9.06%  -  -

Interval: 2, Duration: 0:00:10
PHYSICAL-MEMORY          SYSTEM MEMORY
mem_default              2046M
                          ZONE  USED %USED  CAP  %CAP
                          [total] 1020M 49.8%  -  -
                          [system] 782M 38.2%  -  -
                          global 185M 9.06%  -  -

...
```

例 27 非大域ゾーンのデフォルトのプロセッサセットの報告

次のコマンドでは、1 分間にわたって 2 秒 1 回、my-zone のデフォルトの pset リソースプール要素について報告します。

```
my-zone$ zonestat -r default-pset 1 1m
Collecting data for first interval...
Interval: 1, Duration: 0:00:01
PROCESSOR_SET           TYPE ONLINE/CPUS  MIN/MAX
pset_default            default-pset  2/2        1/-
                          ZONE  USED  PCT  CAP  %CAP  SHRS  %SHR  %SHRU
                          [total] 0.02 1.10%  -  -  -  -  -
                          [system] 0.00 0.19%  -  -  -  -  -
                          global 0.01 0.77%  -  -  -  -  -
                          zone1 0.00 0.07%  -  -  -  -  -
                          zone2 0.00 0.06%  -  -  -  -  -

...
Interval: 60, Duration: 0:01:00
PROCESSOR_SET           TYPE ONLINE/CPUS  MIN/MAX
pset_default            default-pset  2/2        1/-
                          ZONE  USED  PCT  CAP  %CAP  SHRS  %SHR  %SHRU
                          [total] 0.06 3.26%  -  -  -  -  -
                          [system] 0.00 0.18%  -  -  -  -  -
                          global 0.05 2.94%  -  -  -  -  -
                          zone1 0.00 0.06%  -  -  -  -  -
                          zone2 0.00 0.06%  -  -  -  -  -
```

## 例 28 非大域ゾーンの総使用量と最大使用量の報告

次のコマンドでは、10 秒間隔で 3 分間にわたって通知なしにモニターしたあと、my-zone の総使用量と最大使用量について報告します。

```
my-zone$ zonestat -q -R total,high 10s 3m 3m
Report: Total Usage
  Start: Fri Aug 26 07:32:22 PDT 2011
  End: Fri Aug 26 07:35:22 PDT 2011
  Intervals: 18, Duration: 0:03:00
SUMMARY
  Cpus/Online: 2/2   PhysMem: 2046M  VirtMem: 3069M
  ---CPU----  --PhysMem--  --VirtMem--  --PhysNet--
  ZONE  USED %PART  USED %USED  USED %USED  PBYTE %PUSE
[total] 0.01 0.62% 1020M 49.8% 1305M 42.5% 14 0.00%
[system] 0.00 0.23% 782M 38.2% 1061M 34.5% - -
  global 0.00 0.38% 185M 9.06% 208M 6.77% 0 0.00%
  test2 0.00 0.00% 52.4M 2.56% 36.6M 1.19% 0 0.00%

Report: High Usage
  Start: Fri Aug 26 07:32:22 PDT 2011
  End: Fri Aug 26 07:35:22 PDT 2011
  Intervals: 18, Duration: 0:03:00
SUMMARY
  Cpus/Online: 2/2   PhysMem: 2046M  VirtMem: 3069M
  ---CPU----  --PhysMem--  --VirtMem--  --PhysNet--
  ZONE  USED %PART  USED %USED  USED %USED  PBYTE %PUSE
[total] 0.01 0.82% 1020M 49.8% 1305M 42.5% 2063 0.00%
[system] 0.00 0.26% 782M 38.2% 1061M 34.5% - -
  global 0.01 0.55% 185M 9.06% 207M 6.77% 0 0.00%
  test2 0.00 0.00% 52.4M 2.56% 36.6M 1.19% 0 0.00%
```

## 例 29 大域ゾーンからの排他的 IP ゾーンのネットワーク帯域幅の表示

次のコマンドでは、システムの排他的 IP ゾーンのネットワーク帯域幅の利用状況を表示します。ここでは、使用中のデータリンク帯域幅 (VNIC として) の大きさと、各ネットワークデバイスのゾーンごとの利用状況が表示されます。

```
global$ zonestat -r network 1 1
Collecting data for first interval...
Interval: 1, Duration: 0:00:01

NETWORK-DEVICE          SPEED          STATE          TYPE
aggr1                    2000mbps      up             AGGR
  ZONE  TOBYTE  MAXBW  %MAXBW  PRBYTE  %PRBYTE  POBYTE  %POBYTE
  global 1196K  -      -      710K    0.28%   438K    0.18%

net0                      1000mbps      up             PHYS
  ZONE  TOBYTE  MAXBW  %MAXBW  PRBYTE  %PRBYTE  POBYTE  %POBYTE
[total] 7672K  -      -      6112K  4.89%   1756K   1.40%
  global 5344K  100m* 42.6%  2414K  1.93%   1616K   1.40%
  zoneB 992K   100m  15.8%  1336K  0.76%   140K    0.13%
  zoneA 1336K  50m   10.6%  950K   1.07%   0       0.00%

net1                      1000mbps      up             PHYS
  ZONE  TOBYTE  MAXBW  %MAXBW  PRBYTE  %PRBYTE  POBYTE  %POBYTE
  global 126M   -      -      63M    6.30%   63M    6.30%

etherstub1                n/a           n/a           ETHERSTUB
  ZONE  TOBYTE  MAXBW  %MAXBW  PRBYTE  %PRBYTE  POBYTE  %POBYTE
```

```
[total] 3920K    -    -    0    -    0    -
global  1960K  100M*  1.96%  0    -    0    -
zoneA   1960K    50M   3.92%  0    -    0    -
```

---

ヒント - ヒント: 特定の VNIC を表示するには、-x オプションを追加します。

---

## ゾーンごとのファイルシステム統計情報の報告

次の例は、非大域ゾーンのファイルシステムアクティビティを報告する方法を示しています。ほかの例については、[fsstat\(8\)](#) のマニュアルページを参照してください。

**例 30**            特定のゾーンでのアクティビティのモニタリング

このコマンドでは、指定されたゾーンのファイルシステムアクティビティについて報告します。個々の -z オプションは、モニター対象の各ゾーンを示します。

```
global$ pfexec fsstat -z ZoneA -z ZoneB zfs tmpfs
new name name attr attr lookup rddir read read write write
file remov chng get set ops ops ops bytes ops bytes
 93 82 6 163K 110 507K 148 69.7K 67.9M 4.62K 13.7M zfs:ZoneA
248 237 158 188K 101 612K 283 70.6K 68.6M 4.71K 15.2M zfs:ZoneB
12.0K 1.90K 10.1K 35.4K 12 60.3K 4 25.7K 29.8M 36.6K 31.0M tmpfs:ZoneA
12.0K 1.90K 10.1K 35.6K 14 60.2K 2 28.4K 32.1M 36.5K 30.9M tmpfs:ZoneB
```

---

ヒント - すべてのゾーン (大域ゾーンを含む) について報告するには、-z *zonename* オプションではなく、-Z オプション (英大文字) を使用します。

---

**例 31**            すべてのゾーンのゾーンごとの fstype 統計情報の表示

この例は、すべてのゾーン (大域ゾーンを含む) の集約されたファイルシステムアクティビティを報告する方法を示しています。管理者は、システムで実行中の各ゾーンの、tmpfs タイプと zfs タイプのファイルシステムに関するレポートを要求します。

```
global$ pfexec fsstat -A -Z zfs tmpfs
new name name attr attr lookup rddir read read write write
file remov chng get set ops ops ops bytes ops bytes
360K 1.79K 20.2K 4.20M 1.02M 25.0M 145K 5.42M 2.00G 1.07M 8.10G zfs
359K 1.48K 20.1K 4.04M 1.02M 24.5M 144K 5.31M 1.88G 1.06M 8.08G zfs:global
 93 82 6 74.8K 107 250K 144 54.8K 60.5M 4.61K 13.7M zfs:ZoneA
248 237 158 90.2K 101 336K 283 53.0K 58.3M 4.71K 15.2M zfs:ZoneB
60.0K 41.9K 17.7K 410K 515 216K 426 1022K 1.02G 343K 330M tmpfs
49.4K 38.1K 11.0K 366K 489 172K 420 968K 979M 283K 273M tmpfs:global
5.28K 1.90K 3.36K 21.9K 12 21.7K 4 25.7K 29.8M 29.9K 28.3M tmpfs:ZoneA
5.25K 1.90K 3.34K 22.1K 14 21.6K 2 28.4K 32.1M 29.8K 28.2M tmpfs:ZoneB
```

## 非大域ゾーンでの IP ネットワークマルチパスの使用

IP ネットワークマルチパス (IPMP) は、大域ゾーンと同じ方法で、排他的 IP ゾーン内に構成されます。IPMP を使用するためには、排他的 IP ゾーンに `net` リソースが少なくとも 2 つ必要です。IPMP は、これらのデータリンク上のゾーン内から構成されます。

1 つの IPMP グループに 1 つ以上の物理インタフェースを構成できます。IPMP を構成すると、IPMP グループのインタフェースに障害が発生していないかどうかをシステムが自動的にモニターします。グループのインタフェースに障害が発生した場合や、保守のためにインタフェースが削除された場合、IPMP は自動的に、そのインタフェースの IP アドレスを移行して処理を継続します。フェイルオーバーされたアドレスは、障害が発生したインタフェースの IPMP グループ内の機能中のインタフェースが受け取ります。

IPMP のフェイルオーバーコンポーネントは、接続を保持し、既存の接続の切断を防止します。さらに、IPMP は、ネットワークトラフィックを自動的に IPMP グループ内のインタフェースのセットに分散することによって、ネットワークパフォーマンス全体を向上させます。このプロセスは負荷分散と呼ばれます。

また、IPMP のサポートを大域ゾーンからその共有 IP ゾーンに拡張することもできます。

- 排他的 IP ゾーン向けに IPMP グループを構成するには、『[Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.4](#)』の「[Configuring IPMP Groups](#)」の手順に従います。
- 共有 IP ゾーンについては、[199 ページ](#)の「[IP ネットワークマルチパス機能を共有 IP 非大域ゾーンに拡張する方法](#)」の手順に従います。

### ▼ IP ネットワークマルチパス機能を共有 IP 非大域ゾーンに拡張する方法

大域ゾーン内で IPMP を構成し、IPMP 機能を非大域ゾーンに拡張する場合に、ここで説明する手順を使用します。

ゾーンの構成時に、各アドレスつまり論理インタフェースを非大域ゾーンと関連付ける必要があります。手順については、『[Oracle Solaris Zones Configuration Resources](#)』の「[zonecfg and the Global Zone](#)」および [119 ページ](#)の「[zonecfg ユーティリティーを使用してゾーンを構成する方法](#)」を参照してください。

この手順を実行すると、次のことが達成されます。

- `net0` カードと `net1` カードは、インタフェースが `ipmp0` である IPMP グループと一緒に構成されます。

- `imp0` のデータアドレス `192.0.2.0` が非大域ゾーン `my-zone` と関連付けられます。
- `net0` カードがゾーンの物理インターフェースとして設定されます。

稼働中のゾーンで、`ipadm` コマンドを使用して関連付けを行うことができます。詳細は、91 ページの「共有 IP ネットワークインターフェース」および `ipadm(1M)` のマニュアルページを参照してください。

**1. ゾーン管理者になります。**

詳細については、68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」を参照してください。

**2. 大域ゾーンで、IPMP グループを構成します。**

『Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.4』の「Configuring IPMP Groups」の手順に従います。

**3. `zonecfg` コマンドを使用してゾーンを構成します。**

`net` リソースを構成するときに、アドレス `192.0.2.0` および物理インターフェース `net0` をゾーン `zonename` に追加します。

```
zonecfg:my-zone> add net
zonecfg:my-zone:net> set address=192.0.2.0
zonecfg:my-zone:net> set physical=net0
zonecfg:my-zone:net> end
```

非大域ゾーン `my-zone` 内に `net0` のみが表示されるようになります。

**`net0` に連続して障害が発生した場合**

`192.0.2.0` が `imp0` に割り当てられているので、`net0` に連続して障害が発生した場合、このアドレスは引き続き `net1` から利用できます。アドレスの可用性はまた、`my-zone` にも適用されます。ただし、`net0` に障害が発生した場合、`net1` が `my-zone` の可視インターフェースになります。

## 排他的 IP 非大域ゾーンでのデータリンクの管理

このセクションの各例では、排他的 IP ゾーンでデータリンクを管理する方法について説明します。

**例 32**                    ゾーンへのデータリンクの割り当ての表示

この例では、`net0` リンクがゾーン `vz1-100` に割り当てられます。



```
global$ dladm show-linkprop
LINK      PROPERTY      PERM VALUE      EFFECTIVE      DEFAULT      POSSIBLE
net2      speed          r-  10          10            10           --
...
vz1-100/net0 autopush  rw  --          --            --           --
vz1-100/net0 zone      rw  vz1-100      vz1-100      --           --
...
```

**例 33** デバイスのデータリンク名と物理的な場所の表示

この例では、デバイスごとにデータリンク名と物理的な場所の情報が表示されます。デバイスの物理的な場所は、LOCATION フィールドに表示されます。

```
global$ dladm show-phys -L
LINK      DEVICE      LOCATION
net0      net0        MB
net1      net1        MB
net2      net2        MB
net3      net3        MB
net4      ibp0        MB/RISER0/PCIE0/PORT1
net5      ibp1        MB/RISER0/PCIE0/PORT2
net6      eoib2       MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
net7      eoib4       MB/RISER0/PCIE0/PORT2/cloud-nm2gw-2/1A-ETH-2
```

**例 34** 排他的 IP ゾーンへの一時的なデータリンクの割り当て

```
global$ dladm set-linkprop -t -p zone=zonename temp-dl
LINK      PROPERTY      PERM VALUE      DEFAULT      POSSIBLE
temp-dl  zonename      rw  zoneA          --           --
```

データリンクが必要なくなったら、管理者はそれを削除し、そのデータリンクが削除されていることを確認します。

```
global$ dladm reset-linkprop -t -p zonename datalink
LINK      PROPERTY      PERM VALUE      DEFAULT      POSSIBLE
temp-dl  zonename      rw  zoneA          --           --
```

```
global$ dladm show-linkprop -p zonename
LINK      PROPERTY      PERM VALUE      DEFAULT      POSSIBLE
zonename  zonename      rw  zoneA          --           --
```

---

ヒント - 実行中のゾーンでそのデバイスが使用中の場合、再割り当ては失敗し、エラーメッセージが表示されます。218 ページの「排他的 IP ゾーンがデバイスを使用しているために dladm reset-linkprop が失敗する」を参照してください。

---

## ゾーンがインストールされているシステムでの公平配分スケジューラの使用

このセクションでは、FSS をゾーンに使用する方法について説明します。詳細は、[prctl\(1\)](#) のマニュアルページを参照してください。

## ▼ 大域ゾーンで FSS 配分を一時的に設定する方法

大域ゾーンには、デフォルトで 1 つの配分が付与されます。そのデフォルトの割り当てを一時的に変更するには、この手順を実行します。

1. **ゾーン管理者になります。**  
詳細については、68 ページの「[root 以外のユーザーへのゾーンを管理する権利の割り当て](#)」を参照してください。
2. **大域ゾーンに配分を割り当てます。**

```
global$ prctl -n zone.cpu-shares -v N -r -i zone global
```

`-v value`                      新しい値 (この場合は CPU 配分の新しい数) を指定します。

`-i zone idtype`                ゾーン名またはゾーン ID を指定します。

3. **大域ゾーンに割り当てられた配分の数を確認します。**

```
global$ prctl -n zone.cpu-shares -i zone global
zone: 0: global
NAME      PRIVILEGE      VALUE      FLAG      ACTION      RECIPIENT
zone.cpu-shares
  usage          1
  privileged      1      -      none      -
  system        65.5K      max      none      -
```

**注記** - `prctl` コマンドで指定された制限は永続的なものではありません。システムがリブートされると、制限は無効になります。リブート後もデフォルトの割り当てを使用しないようにする場合は、`prctl` コマンドによって以前に割り当てた配分を設定しなおす必要があります。

ゾーンの配分を永続的に設定する方法については、119 ページの「[zonecfg ユーティリティーを使用してゾーンを構成する方法](#)」および 131 ページの「[大域ゾーンの CPU シェアを設定する方法](#)」を参照してください。

### 例 35 ゾーンの zone.cpu-shares 値の動的な変更

```
$ prctl -n zone.cpu-shares -r -v 2 -i zone ZoneA
```

## ゾーンのバックアップと再作成

このセクションでは、ゾーン内のファイルとゾーン構成ファイルをバックアップする手順、および保存した構成ファイルを使用してゾーンを作成しなおす手順について説明します。

---

**注記** - ゾーンが共有ストレージ上にあるシステムでは、各ゾーンまたは非大域ゾーンを別々にアーカイブして配備します。『Oracle Solaris 12 でのシステム復旧とクローン』の「How to Archive a System That Contains Zones on Shared Storage」を参照してください。

---

**例 36**                    統合アーカイブを使用した、ゾーンが含まれているシステムのバックアップ

次の例では、ゾーン my-zone の回復用アーカイブを /net/server/my-zone-archive.uar に作成します。

```
host1$ pfbash archiveadm create -r -z my-zone /net/server/my-zone-archive.uar
```

**例 37**                    ゾーン構成のコピーの出力

管理者は、sysidtool の質問に答えたあと、最初のログイン時に非大域ゾーン構成のバックアップファイルを作成します。管理者は、あとでこのバックアップを使用してゾーンを作成しなおすように計画します。

```
global$ pfbash zonecfg -z my-zone export > /net/server/my-zone-configtemplate.txt
```

**例 38**                    非大域ゾーンの再作成

管理者は、これらの手順に従って、例37「ゾーン構成のコピーの出力」および例36「統合アーカイブを使用した、ゾーンが含まれているシステムのバックアップ」で作成した各ファイルから my-zone を作成し直します。

1. 構成ファイルを使用してゾーンを構成します。

```
host2$ pfbash zonecfg -z my-zone -f /net/server/my-zone-configtemplate.txt
```

2. アーカイブを使用してゾーンをインストールします。

```
host2$ zoneadm -z my-zone install -a /net/server/my-zone-archive.uar
```



# ◆◆◆ 第 12 章

## 不変ゾーンの構成と管理

---

この章では、ゾーンのセキュリティーポリシーを指定して、不変ゾーンを構成する方法および管理アクセスを構成して、不変ゾーンを保守する方法について説明します。

---

**注記** - すべてのゾーン (物理大域ゾーン、非大域ゾーン、仮想大域ゾーン (カーネルゾーンと呼ばれる)) を不変にできます。

---

この章で扱う内容は、次のとおりです。

- [205 ページの「不変ゾーンの構成」](#)
- [209 ページの「不変非大域ゾーンの管理」](#)
- [214 ページの「不変大域ゾーン」](#)

### 不変ゾーンの構成

変更可能ゾーンと不変ゾーンは、`zonecfg` コマンドの `file-mac-profile` プロパティーで指定するそれらの MWAC セキュリティーポリシーによって区別されます。

### MWAC セキュリティーポリシーの設定

デフォルトで `file-mac-profile` プロパティーは設定されず、ゾーンには書き込み可能ルートデータセットがあります。

`file-mac-profile` のいくつかの値は、ゾーン内からのすべてまたは一部の実行時環境へのアクセスを制限します。none 以外のすべてのプロファイルでは、`/var/pkg` ディレクトリとその内容がゾーンの内部から読み取り専用になります。none MWAC セキュリティーポリシーは、未設定の MWAC セキュリティーポリシーと同等です。

次の MWAC 値は、ゾーン内からのすべてまたは一部の実行時環境へのアクセスを制限します。

#### dynamic-zones

カーネルゾーンの大域ゾーンを含む大域ゾーンに有効です。カーネルゾーンおよび非大域ゾーンの作成と破棄を許可します。

`fixed-configuration` と同等ですが、カーネルゾーンと非大域ゾーンを作成および破棄する機能を追加します。

`flexible-configuration` に似ていますが、`dynamic-zones` は、`/etc` ディレクトリ内のファイルに書き込みできません。

#### fixed-configuration

システム構成コンポーネントを格納するディレクトリを除く `/var/*` ディレクトリへの更新を許可します。

- IPS パッケージ (新しいパッケージを含む) はインストールできません。
- 持続的に有効な SMF サービスは固定されます。
- SMF マニフェストはデフォルトの場所から追加できません。
- 構成ファイルのロギングおよび監査はローカルに行うことができます。 `syslog` および監査の構成は固定されます。

#### flexible-configuration

`/etc/*` ディレクトリ内のファイルの変更、ルートのホームディレクトリへの変更、`/var/*` ディレクトリへの更新を許可します。この構成は、Oracle Solaris 10 ガイド『Oracle Solaris のシステム管理 (Oracle Solaris コンテナ: 資源管理と Oracle Solaris ゾーン)』に記述されている Oracle Solaris 10 native 疎ルートゾーンにもっとも近い機能を提供します。

- IPS パッケージ (新しいパッケージを含む) はインストールできません。
- 持続的に有効な SMF サービスは固定されます。
- SMF マニフェストはデフォルトの場所から追加できません。
- 構成ファイルのロギングおよび監査はローカルに行うことができます。 `syslog` および監査の構成は変更できます。

#### strict

読み取り専用ファイルシステム、例外なし。

- IPS パッケージはインストールできません。
- 持続的に有効な SMF サービスは固定されます。
- SMF マニフェストはデフォルトの場所から追加できません。
- 構成ファイルのロギングおよび監査は固定されます。データはリモートからのみロギングできます。

- このプロファイルによる不変ゾーン内での NFS サーバーの実行はサポートされていません。fixed-configuration プロファイルを使用して NFS サーバーを実行する必要があります。

#### 例 39 大域ゾーンの MWAC セキュリティポリシーの設定

この例では、ゾーンセキュリティ権利プロファイルが割り当てられ、不変大域ゾーンを作成しています。このゾーン内で、ゾーン管理者はカーネルゾーンおよび非大域ゾーンを作成および破棄できます。それ以外の場合、ゾーンは不変です。

```
global$ zonecfg -z global set file-mac-profile=dynamic-zones
```

MWAC セキュリティポリシーが設定され、不変ゾーンをリブートすると、ゾーンが self-assembly-complete マイルストーンに達するまで、一時的な読み取り/書き込みでブートし、その後読み取り専用モードでリブートします。

## MWAC セキュリティポリシーのゾーンリソースの例外

zonecfg add dataset コマンドによってゾーンに追加したデータセットは MWAC ポリシーの対象となりません。ゾーンは追加されたデータセットを完全に制御できません。プラットフォームのデータセットは表示されますが、ゾーンが読み取り/書き込みでブートされないかぎり、そのデータとプロパティは読み取り専用です。

zonecfg add fs コマンドによってゾーンに追加したファイルシステムは MWAC ポリシーの対象となりません。ポリシーを保持するには、ファイルシステムを読み取り専用でマウントします。

## 不変ゾーン内の SMF サービス

不変ゾーンでは、そのサービスで必要とされる承認がある場合でも、デフォルトで SMF サービスを永続的に変更することはできません。サービスを有効または無効にするか、プロパティの値を変更するには、-t (一時) オプションを使用する必要があります。ゾーンのリブート時に変更が元に戻されます。-t オプションを使用しない場合、「権限がありません」エラーメッセージが表示されます。

不変ゾーンのリブート間で保持されるサービスの変更を行うには、[210 ページの「トラステッドパストメインを使用した不変ゾーンの管理」](#)に説明されているとおり、トラステッドパストメイン (TPD) で操作している必要があります。

す。TPD で操作している場合、必要な承認を持つすべてのサービスに対して永続的な変更を行えます。

一部のサービスは、それらのメソッドによって起動されたプロセスが TPD で実行できるように構成できます。trusted\_path 属性が true に設定されているサービスは TPD で実行します。



**注意** - trusted\_path を true に設定すると、システムの不変性が低下します。不変ゾーンを作成したときに必要としたセキュリティが低下します。

特に、ゾーンの外部からアクセス可能なサービスの trusted\_path を true に設定しないでください。具体的には、SSH などの対話型ログインサービスはトラステッドパス上にあってはなりません。

RAD API は Oracle Solaris 承認をチェックし、実行できることを制限するため、212 ページの「[RAD を使用して不変ゾーンへのリモート管理アクセスを有効にする方法](#)」に説明されている rad:remote サービスはこの注意事項の例外です。rad:remote サービスは対話型シェルを提供せず、ホストの TPD に到達するためにセキュアトランスポートに対して、強力な認証および承認 (ユーザーの user\_attr エントリの tpd=yes) を使用します。

puppet:agent サービスは上記の注意事項のもう 1 つの例外です。Puppet を使用してシステムを更新することは、対話型ログインによって更新するより信頼できません。puppet:master インスタンスはトラステッドパス上にある必要がないことに注意してください。

次の例に、サービスが TPD で実行するように、trusted\_path の値を設定する方法を示します。

#### 例 40 Puppet サービスのトラステッドパスへの追加

TPD で実行するサービスに trusted\_path が設定されているかどうかを確認します。

```
# svcprop -p method_context/trusted_path puppet:agent
svcprop: Couldn't find property group 'method_context/trusted_path' for instance
'svc:/application/puppet:agent'.
```

上記のコマンドに示されているサービスは、trusted\_path 属性が設定されていないため、TPD で実行していません。trusted\_path の値が false であった場合も同じことが当てはまります。

puppet:agent サービスが TPD で実行するように構成するには、trusted\_path を true に設定します。この属性は、このサービスに存在しないため、値に加えて属性タイプを指定する必要があります。属性の設定後、サービスのリフレッシュと再起動を行います。次のコマンドには、検証ステップが含まれます。

```
# svcfg -s puppet:agent
svc:/application/puppet:agent> setprop method_context/trusted_path = boolean: true
```



```

svc:/application/puppet:agent> listprop method_context/trusted_path
method_context/trusted_path boolean    true
svc:/application/puppet:agent> refresh
svc:/application/puppet:agent> restart
svc:/application/puppet:agent> exit
# svcprop -p method_context/trusted_path puppet:agent
true

```

次の出力は、puppet:agent サービスによって起動された puppet プロセスに関する情報を示しています。TPD フラグの説明については、[getpflags\(2\)](#) のマニュアルページを参照してください。

```

$ svcs -p puppet:agent
STATE      STIME      FMRI
online     15:05:50   svc:/application/puppet:agent
           15:05:50     7008 puppet

$ ppriv 7008
7008:      /usr/ruby/2.1/bin/ruby /usr/sbin/puppet agent --logdest /var/log/puppet
flags = PRIV_PROC_TPD|PRIV_TPD_UNSAFE|PRIV_TPD_KILLABLE
  E: all
  I: basic
  P: all
  L: all

```

## 不変非大域ゾーンの管理

このセクションでは、非大域ゾーンが不変かどうかを判断する方法に関する情報を提供します。ゾーンが不変の場合、管理にはいくらかのセットアップが必要です。

---

**注記** - 管理アクセスを構成しない場合、ディスク上の非大域ゾーンは大域ゾーンからのみ管理できます。実行中の非大域ゾーン内では、不変ゾーンの実行時状態のみを管理できます。そのため、実行中のゾーンの MWAC ポリシーの変更は一時的になります。詳細については、[207 ページの「不変ゾーン内の SMF サービス」](#)を参照してください。

---

## 非大域ゾーンが不変かどうかの判断

大域ゾーンからの `zoneadm list -p` コマンドの解析可能な出力には、R/W 列と `file-mac-profile` 列が表示されます。次の出力で、`fixed-configuration` 実行中ゾーンの `testzone2` および `testzone3` は読み取り専用で、`testzone1` 実行中ゾーンは読み取り/書き込み `fixed-configuration` ゾーンです。

```

global$ zoneadm list -p
0:global:running:/:UUID:solaris:shared:-:none
5:testzone2:running:/export/zones/testzone2:UUID \
:solaris:shared:R:fixed-configuration
12:testzone3:running:/export/zones/testzone3:UUID \
:solaris:shared:R:fixed-configuration

```

```
13:testzone1:running:/export/zones/testzone1:UUID \  
    :solaris:excl:w:fixed-configuration  
-:testzone:installed:/export/zones/testzone:UUID \  
    :solaris:excl:-:fixed-configuration
```

## 非大域不変ゾーンの管理の方法

管理者は、管理操作用に不変ゾーンへのアクセスを明示的に構成する必要があります。よりセキュアな方法では、承認が必要であり、有効な MWAC ポリシーとアクセスのモードに依存します。セキュアな方法ではトラステッドパスを使用します。簡単にセキュアでない方法は、ゾーンを短時間変更可能にし、変更を行ってから、ゾーンを不変としてリポートすることです。

- **トラステッドパスドメインを使用する** - コンソールにアクセスでき、コンソールとユーザーがトラステッドパスドメイン (TPD) にアクセスするように構成する必要があります。

[210 ページの「トラステッドパスドメインを使用した不変ゾーンの管理」](#)を参照してください。

---

注記 - flexible-configuration ポリシーを除き、その他の3つの MWAC ポリシーは、不変ファイルのみにアクセスし、変更できる safe モードを適用します。

---

- **ゾーン全体を一時的に書き込み可能にする** - ターミナルウィンドウにアクセスでき、zoneadm または zlogin コマンドの実行が承認されている必要があります。zoneadm メソッドは小さなすばやい修正に役立ちます。不変ゾーンを一時的に書き込み可能としてブートし、変更して、リポートします。

zlogin メソッドは保護されたファイルの編集とパッケージの更新に役立ちます。zlogin セッション中、ゾーンは書き込み可能です。このメソッドは、コンソールログインで使用できません。

[213 ページの「不変ゾーンを書き込み可能にして管理する」](#)を参照してください。

## トラステッドパスドメインを使用した不変ゾーンの管理

Oracle Solaris では、不変ゾーンの管理のために不変ゾーンに入る4つの方法が用意されています。2つの方法は、[210 ページの「非大域不変ゾーンの管理の方法」](#)に説明するように、ゾーン全体を一時的に書き込み可能にします。より安全なモードでは、トラステッドパスを使用します。そこではトラステッドパスの一部としてマークされたプロセスのみを変更でき、ファイルやほかのゾーンプロセスは不変のままにな

ります。トラステッドパスで実行するプロセスは、トラステッドパスドメイン (TPD) の一部として記述されます。

不変ゾーンでは、特定のコアシステムプロセスが TPD の一部としてマークされます。たとえば、`init`、`svc.configd`、`svc.startd` など多数のシステムデーモンが TPD で実行されます。TPD プロセスへの管理アクセスが付与されている場合、すべての TPD 以外のプロセスが書き込み不可能なままであるため、不変ゾーンの構成を安全に変更できます。

不変ゾーンを管理するには、コンソール経由でローカルに、または信頼できる rad 接続によってリモートでトラステッドパスを使用します。

- ローカル管理を有効にするには、ILOM、シリアル接続、またはグラフィカルコンソールを通じてコンソールにアクセスできることを確認する必要があります。TPD に入るには、自分も信頼されているトラステッドパスによって保護されているコンソールにログインします。

手順については、[211 ページの「コンソールから不変ゾーンへの管理アクセスを有効にする方法」](#)を参照してください。

- リモート管理デーモン (RAD) によってリモート管理を有効にするには、トラステッドパスを使用して RAD プロセスを保護し、自分も信頼されている必要があります。

手順については、[212 ページの「RAD を使用して不変ゾーンへのリモート管理アクセスを有効にする方法」](#)を参照してください。

## ▼ コンソールから不変ゾーンへの管理アクセスを有効にする方法

ゾーンを不変のままにし、管理者がコンソールから TPD 内のプロセスとファイルにアクセスできるようにするには、このタスクを実行します。

1. **root 役割になります。**

詳細は、『[Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

2. **大域ゾーンで `tpdlogin` PAM モジュールを構成して、コンソールへのアクセスを制限します。**

手順については、『[Managing Authentication in Oracle Solaris 11.4](#)』の「[How to Restrict Access to the Trusted Path Domain](#)」を参照してください。

3. **コンソールログイン SMF サービスを TPD で実行するように変更します。**

```
# svccfg -s console-login:default
svc:/system/console-login:default> setprop start/trusted_path = true
svc:/system/console-login:default> refresh
svc:/system/console-login:default> exit
```

4. **(オプション) コンソールで `trusted_path` 属性が設定されていることを確認します。**

```
# svcprop -p start/trusted_path console-login:default
true
```



注意 - コンソールで `trusted_path` 属性を設定する場合、`/etc/security/tpdusers` ファイルでコンソールへのアクセスを制限し、未承認ユーザーによるログインを防ぎます。ステップ 2 で未承認ログインが妨げられているはずですが。

---

5. コンソールログインサービスを再起動します。

```
# svcadm restart console-login:default
```

6. `/etc/security/tpdusers` のいずれかのユーザーとして不変ゾーンにログインします。

- コンソールにログインし、トラステッドパスログインプロンプトに回答します。
- 物理コンソールで、セキュアアテンションキーシーケンスを入力して、トラステッドパスログインプロンプトを起動します。
  - Stop-A (SPARC)
  - F1-A (x86)

ログイン後、TPD にあるファイルとプロセスを管理できます。さらに、役割を引き受け、その役割で不変ゾーンを管理することもできます。

## ▼ RAD を使用して不変ゾーンへのリモート管理アクセスを有効にする方法

リモートアクセスデーモン (RAD) を使用して、不変ゾーンにアクセスできるようにするには、このタスクを実行します。`rad:remote` SMF サービスが有効になり、RAD 接続に対して TPD 認証が実行されます。

1. `root` 役割になります。

詳細は、『Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護』の「割り当てられている管理権利の使用」を参照してください。

2. `rad:remote` SMF サービスを TPD で実行するように変更します。

```
# svccfg -s rad:remote
svc:/system/rad:remote> setprop method_context/trusted_path = true
svc:/system/rad:remote> refresh
svc:/system/rad:remote> exit
```

3. `rad:remote` サービスに `trusted_path` 属性が設定されていることを確認します。

```
# svcprop -p method_context/trusted_path rad:remote
true
```

#### 4. リモート RAD サービスを再起動します。

```
# svcadm restart rad:remote
```

#### 5. 1 人または複数の管理者が RAD 接続で TPD にアクセスできるようにします。

```
$ usermod -K tpd=yes username
```

これらの管理者は、RAD 上でリモートで不変ゾーンにログインできるようになります。

参照 RAD アクセスについては、『[Managing User Accounts and User Environments in Oracle Solaris 11.4](#)』の第3章、「[Managing User Accounts Interactively](#)」を参照してください。

TPD で実行する SMF サービスを追加するには、[例40「Puppet サービスのトラステッドパスへの追加」](#)を参照してください。

## 不変ゾーンを書き込み可能にして管理する

`zoneadm boot` サブコマンドには、帯域ゾーン管理者が、書き込み可能なルートファイルシステムまたは一時的に書き込み可能なルートファイルシステムによって、不変ゾーンを手動でブートできるようにする2つのオプションがあります。ゾーンは次のリポートまでのみ書き込み可能モードになります。

-w

書き込み可能なルートファイルシステムを持つゾーンを手動でブートします。

-W

一時的に書き込み可能なルートファイルシステムを持つゾーンを手動でブートします。システムは、`self-assembly-complete` マイルストーンに達すると自動的にリポートされます。リポートすると、ゾーンがふたたび MWAC ポリシーの制御下に置かれます。このオプションは、ゾーンが MWAC ポリシー `none` を持っている場合に許可されます。

-w および -W オプションは、不変ゾーンではないゾーンでは無視されます。

`zlogin` コマンドは、不変ファイルの編集や新しいパッケージの追加などのアクションに対し2つのオプションを提供します。これらのオプションを使用するには、`solaris.zone.manage/zonename` の承認が必要です。

-T

トラステッドパス属性 `PRIV_PROC_TPD` を設定して不変ゾーンに入ります。このセッションでは、通常は変更できないゾーン内のファイルを変更できます。保護されていないファイルはセッションで読み取ることはできません。

-U

-T オプションと同じプロセスを安全ではないモードで実行します。安全でないモードでは、保護されていないファイルを変更できます。このオプションは、flexible-configuration MWAC セキュリティポリシーを使用したゾーンに使用します。

---

注記 - これらのオプションはコンソールログインで使用できず、不変ゾーンではないゾーンに対して無視されます。

---

## 不変大域ゾーン

不変大域ゾーンは、不変非大域ゾーンを帯域ゾーンに拡張します。

### 不変大域ゾーンの構成

不変大域ゾーンの構成は、不変非大域ゾーンの構成に似ています。205 ページの「MWAC セキュリティポリシーの設定」に説明するように、MWAC セキュリティポリシーを zonecfg コマンドで設定します。ゾーン構成をコミットしたあと、ブート情報が書き込まれ、ブートアーカイブが更新されます。大域ゾーンはすぐに不変になります。リブートは必要ありません。

次の情報は、不変大域ゾーンに固有のものであります。

- 大域ゾーンで DHCP を使用してネットワークインタフェースを設定している場合、flexible-configuration MWAC ポリシーを選択する必要があります。
- rpool データセットは制限されます。  
zonecfg add dataset コマンドを使用することにより、無制限のサブデータセットを追加できます。不変大域ゾーンは、無制限のデータセットでのみゾーンを実行できます。無制限のデータセットのすべての子も無制限です。

### 不変大域ゾーンの保守

大域ゾーンを保守するもっともセキュアな方法は、トラステッドパスを使用することです。トラステッドパスはコンソールでのみ使用可能なため、ILOM、シリアル接続、またはグラフィカルコンソールを通じてコンソールにアクセスできることを確認してください。

不変になるようにシステムを構成したら、トラステッドパスでコンソールログインを構成します。手順については、[211 ページの「コンソールから不変ゾーンへの管理アクセスを有効にする方法」](#)を参照してください。コンソールログインを構成したら、root アカウントはゾーンにログインできず、管理できなくなります。トラステッドパスを使用することを承認されているユーザーとしてログインする必要があります。ログイン後、役割を引き受けることができます。

不変大域ゾーン内で `pkg update` コマンドを実行すると、初回ブートは読み取り/書き込みになります。必要な自己アセンブリ手順を実行するために、これらのアクセス権がシステムで必要です。自己アセンブリ手順の実行が終わると、システムは再度不変になります。





## その他の Oracle Solaris ゾーンの問題のトラブルシューティング

---

この章では、次のゾーントラブルシューティング情報について説明します。

- 217 ページの「許可されているパッケージの不一致ためインストールが失敗する」
- 218 ページの「排他的 IP ゾーンがデバイスを使用しているために `dladm reset-linkprop` が失敗する」
- 218 ページの「ゾーン構成内に不正な特権セットが指定されている」
- 218 ページの「ゾーンが停止しない」

### 許可されているパッケージの不一致ためインストールが失敗する

solaris ゾーンのインストールを試みた場合の次のエラーメッセージは、大域ゾーンと非大域ゾーン間のパブリッシャーの不一致を示しています。

```
The following pattern(s) did not match any allowable packages. Try  
using a different matching pattern, or refreshing publisher information:
```

大域ゾーンは現在、インストールされているシステムソフトウェアと同じバージョンを含まないパブリッシャーによって構成されている可能性があります。非大域ゾーンをインストールするには、solaris パブリッシャーの起点として設定するリポジトリが、非大域ゾーンのインストール場所となる大域ゾーン内にインストールされているものと同じシステムソフトウェアを最低限含んでいる必要があります。

詳細については、『Oracle Solaris 11.4 でのソフトウェアの追加とシステムの更新』の「非大域ゾーンをインストールできない」を参照してください。

## 排他的 IP ゾーンがデバイスを使用しているために `dladm reset-linkprop` が失敗する

次のエラーメッセージは、`dladm reset-linkprop` コマンドを使用する試みが失敗したことを示しています。

```
dladm: warning: cannot reset link property 'zone' on 'net0': operation failed
```

例34「排他的 IP ゾーンへの一時的なデータリンクの割り当て」を参照してください。実行中のゾーン `exc1` がそのデバイスを使用しています。

値をリセットするには、次を実行します。

1. 次のコマンドを入力します。

```
global$ ipadm delete-ip net0
```

2. `dladm` コマンドを再実行します。

## ゾーン構成内に不正な特権セットが指定されている

ゾーンの特権セットに許可されない特権が含まれる場合、必須の特権が欠落している場合、または不明な特権名が含まれる場合、ゾーンの検証、準備、またはブートの試行は失敗し、次のようなエラーメッセージが表示されます。

```
zonecfg:zone5>set limitpriv="basic"
global$ zoneadm -z zone5 boot
required privilege "sys_mount" is missing from the zone's privilege set
zoneadm: zone zone5 failed to verify
```

問題を修正するには、[132 ページの「ゾーンの特権を変更する方法」](#)を参照してください。

## ゾーンが停止しない

ゾーンに関連付けられたシステム状態を配備できない場合、停止処理は途中で失敗します。このため、`running` と `installed` の間で、ゾーンが中間状態のままになります。この状態では、アクティブなユーザープロセスやカーネルスレッドは存在せず、何も作成できません。停止操作が失敗した場合は、手動で処理を完了する必要があります。

障害のもっとも一般的な原因は、システムがすべてのファイルシステムをアンマウントできないことです。システム状態が破棄される Oracle Solaris システムの従来のシャットダウンとは異なり、ゾーンでは、ゾーンのブートまたは操作時に実行されたマウントがゾーンの停止後に残らないようにする必要があります。zoneadm コマンドによりゾーン内で実行中のプロセスが存在しないことが確認されても、大域ゾーン内のプロセスによりゾーン内のファイルが開かれた場合には、アンマウント操作が失敗することがあります。

[proc\(1\)](#) のマニュアルページ ([pfiles\(1\)](#) のマニュアルページを参照) および [fuser\(8\)](#) のマニュアルページに説明されているツールを使用して、これらのプロセスを見つけて、適切なアクションを実行します。これらのプロセスが処理されたら、zoneadm halt コマンドを再実行して、完全にゾーンを停止します。



## 共有ストレージ上の Oracle Solaris ゾーン

---

共有ストレージ上の Oracle Solaris ゾーンを使用すると、ゾーン内の共有ストレージリソースに透過的にアクセスおよび管理できます。これらの自動化された機能により、Oracle Solaris システムでのゾーンとそれに対応する共有ストレージリソースの配備、管理、および移行が簡単になります。

対応する共有ストレージリソースについて、ホストに依存しない形式でゾーン構成に記述できます。この機能を使用したゾーンインストールは、共有ストレージデバイスでホストされる専用の ZFS ストレージプールにカプセル化されます。

Oracle Solaris ゾーン フレームワークは自動的に、共有ストレージリソースを構成および構成解除します。さまざまなゾーン管理アクティビティ全体で必要になる ZFS ストレージプール管理タスクがすべて自動的に実行されます。

この章では、次の内容について説明します。

- 222 ページの「ストレージ URI を使用した共有ストレージリソースについて」
- 224 ページの「ストレージ URI と共有ストレージリソースの管理」
- 226 ページの「Oracle Solaris ゾーン への共有ストレージリソースの割り当て」
- 228 ページの「共有ストレージリソースでの Oracle Solaris ゾーン の自動 ZFS ストレージプール管理」
- 230 ページの「unavailable ゾーン状態について」
- 230 ページの「追加の zoneadm サブコマンドオプション」
- 233 ページの「共有ストレージリソースでホストされる Oracle Solaris ゾーン の実装」
- 244 ページの「共有ストレージリソースでホストされる Oracle Solaris ゾーン の移行」
- 245 ページの「共有ストレージゾーン構成に対する既存のゾーンの移動」
- 250 ページの「共有ストレージリソースについてのゾンドキュメントリファレンス」

## ストレージ URI を使用した共有ストレージリソースについて

ストレージ Universal Resource Identifier (URI) は、ホストに依存しない形式で共有ストレージリソースを記述するために使用されます。ストレージ URI はさまざまなノード全体でストレージオブジェクトを一意に識別します。URI は、インターネットで一般的に使用される周知の原則や URI の形式に従います。次のストレージ URI を使用できます。

### ローカルデバイス URI

ローカルデバイスストレージ URI タイプは、そのローカルデバイスパスでストレージデバイスを記述します。パスは `/dev` 名前空間内のデバイスを示している必要があります。これらのデバイスは通常、特定のシステムに一意である直接接続ストレージ (DAS) リソースであり、そのデバイス名とパスは一般に移植できません。ただし、これらはまた、複数のノードにわたる統合名前空間を `/dev` の下にすでに用意しているテクノロジーの共有ストレージリソースを指すこともできます。

次に URI 構文のリストを示します。

```
dev:local-path-under-/dev
dev:///path-with-dev
dev:absolute-path-with-dev
```

次の例は、URI 構文を使用する方法を示しています。

```
dev:dsk/c0t0d0s0
dev:///dev/dsk/c0t0d0
dev:/dev/dsk/c0t0d0
dev:chassis/SYS/HD1/disk
dev:dsk/c0t60A98000564C303132302D6F72613939d0
```

ローカルデバイスストレージ URI は、ディスク全体を示す場合も、特定のスライスまたはパーティションを示す場合もあります。ただし、通常、ZFS ストレージプールではスライスまたはパーティションを使用することはお勧めしません。

### 論理ユニット URI

論理ユニット URI タイプは、ファイバチャネル (FC) またはシリアル接続 SCSI (SAS) ストレージデバイスを記述します。これは、そのデバイス ID (WWN) に基づいた論理ユニット (LU) を示します。論理ユニットストレージ URI は常にディスク全体を表します。

次に URI 構文のリストを示します。

```
lu:luname.naa.ID
lu:luname.eui.ID
```

```
lu:initiator.naa.ID,target.naa.ID,luname.naa.ID
lu:initiator.naa.ID,target.naa.ID,luname.eui.ID
```

次の例は、URI 構文を使用する方法を示しています。

```
lu:luname.naa.5000c5000288fa25
lu:luname.eui.0021280001cf80f6
lu:initiator.naa.2100001d38089fb0,target.naa.2100001d38089fb0,luname.naa.5000c5000288fa25
lu:initiator.naa.2100001d38089fb0,target.naa.2100001d38089fb0,luname.eui.0021280001cf80f6
```

luname のみの URI 形式では、ID は論理ユニット名を示します。

initiator,target,luname の形式では、initiator は発信元ポートを、target はターゲットポートを示します。これらが組み合わさって、論理ユニットへのパスが指定されます。2 番目の構文の論理ユニット名は、最初の構文 (luname のみの形式) の URI の luname と一致する必要があります。

luname のみの URI と組み合わせてマルチパスを使用することをお勧めします。マルチパスが無効で、luname のみの URI を使用する場合は指定した論理ユニットへのランダムなパスが選択されます。パスがランダムに割り当てられないようにするために、initiator,target,luname の URI 形式を使用して論理ユニットへの特定のパスを選択できます。マルチパスが有効で、initiator,target,luname の URI 形式を使用している場合、マルチパスフレームワークによって、論理ユニットにアクセスするために使用するパスが制御され、URI はアクセスパスではなく、ユニットを識別するためにのみ使用されます。

## iSCSI URI

iSCSI URI タイプは、iSCSI ネットワークベースのストレージプロトコルを使用してアクセスされるストレージデバイスを記述します。これは常にディスク全体を示します。

iSCSI 修飾名 (IQN) の詳細については、[suri\(7\)](#) のマニュアルページを参照してください。

次に URI 構文のリストを示します。

```
iscsi:///luname.naa.ID
iscsi:///luname.eui.ID
iscsi://host[:port]/luname.naa.ID
iscsi://host[:port]/luname.eui.ID
iscsi:///target.IQN,lun.LUN
iscsi://host[:port]/target.IQN,lun.LUN
```

次の例は、URI 構文を使用する方法を示しています。

```
iscsi:///luname.eui.0021280001cf80f6
iscsi:///luname.naa.600144f03d70c80000004ea57da10001
iscsi://[::1]/luname.naa.600144f03d70c80000004ea57da10001
iscsi://127.0.0.1/luname.naa.600144f03d70c80000004ea57da10001
iscsi://127.0.0.1:3260/luname.naa.600144f03d70c80000004ea57da10001
iscsi://hostname:3260/luname.eui.0021280001cf80f6
iscsi://hostname:3260/luname.naa.600144f03d70c80000004ea57da10001
```

```
iscsi://[::1]/target.iqn.com.sun:02:d0f2d311-f703,lun.2
```

luname のみの URI 形式および ID の説明については、[222 ページの「論理ユニット URI」](#) を参照してください。

オプションで、hostname[:port] 権限セクションには、認証されていない SendTargets 検出アドレスを使用して iSCSI イニシエータを自動的に構成するための情報が記されます。IPv6 アドレスは、角括弧 ([]) で囲む必要があります。

静的または iSNS ベースの構成の場合か、認証を使用する場合にかぎり、luname のみの URI 形式を使用します。これらの関数は、使用する前に、ゾーンフレームワークの外部で構成する必要があります。

iSER トランスポートを介して iSCSI を使用する場合は、iSCSI ストレージの URI は透過的にこれも許可します。iSER を使用するには、ターゲットと発信元は、InfiniBand (IB) パーティションリンクに割り当てられたアドレスを通過する必要があります。大域ゾーンで、dladm create-part コマンドを使用して IB パーティションを設定し、ipadm create-ip コマンドと ipadm create-addr コマンドを使用して IB パーティションにターゲットと発信元のアドレスを割り当てる必要があります。このアドレスは、続いて、ターゲットの検出アドレスを指定するために、iSCSI ストレージ URI の権限セクションで使用されます。

InfiniBand (IB) ハードウェアが存在し、InfiniBand Reliable-Connected (RC) 接続を確立できる場合、iSER 対応の発信元は、iSER 対応のターゲットへの iSER 接続を使用します。RC 接続を確立できない場合、接続は IP ベースの接続を使用して確立されます。

## ストレージ URI と共有ストレージリソースの管理

ストレージ URI、またはストレージ URI に基づいた管理者共有ストレージリソースを生成および検証するには、suriadm コマンドを使用します。

suriadm コマンドを使用して、既存のシステムデバイスパスに基づいて、手動で作成したストレージ URI を検証することも、自動的にストレージ URI を作成することもできます。ストレージ URI タイプに応じて、suriadm コマンドを使用して、対応するストレージサブシステムを構成および構成解除できます。ストレージ URI が与えられた場合は、同じコマンドで、ストレージ URI に記述されたストレージオブジェクトの可能なデバイスインスタンスを特定することもできます。次の例は一般的な使用事例について説明しています。詳細は、[suriadm\(8\)](#) のマニュアルページを参照してください。

**例 41** suriadm parse 構文を使用したストレージ URI の検証

次の例は、suriadm コマンドを使用してストレージ URI を検証する方法を示します。これらの例では、ストレージ URI の文字列が解析されており、その対応するプロパ



ティーが表示されます。検証が終わると、ストレージ URI は後から `zonecfg` コマンドまたは `suriadm` コマンドのどちらかで使用できます。

```
root@initiator:~# suriadm parse iscsi://target/lunname.naa.600144F035FF850000050C884E50001
PROPERTY      VALUE
uri-type      iscsi
hostname      target
port          -
lunname       naa.600144F035FF850000050C884E50001

root@host:~# suriadm parse dev:/dev/dsk/c4t1d0
PROPERTY      VALUE
uri-type      dev
path          /dev/dsk/c4t1d0
```

**例 42** `suriadm lookup-uri` 構文を使用した、デバイスパスに基づくストレージ URI の生成

次の例は、`suriadm` コマンドを使用してストレージ URI を作成する方法を示します。この例では、既存のローカルデバイスパスに基づいて、`suriadm` コマンド出力には、`suriadm` コマンドか `zonecfg` コマンドのどちらかで後から使用できる有効なストレージ URI が示されます。

```
root@target:~# suriadm lookup-uri -t iscsi /dev/dsk/c0t600144F035FF850000050C884E50001d0
iscsi://target/lunname.naa.600144f035ff850000050c884e50001

root@host:~# suriadm lookup-uri /dev/dsk/c4t1d0
dev:dsk/c4t1d0

root@host:~# suriadm lookup-uri /dev/dsk/c0t600144F0DBF8AF190000510979640005d0
lu:lunname.naa.600144f0dbf8af190000510979640005
lu:initiator.naa.10000000c9991d8c,target.naa.21000024ff3ee89f,lunname.naa.600144f0dbf8af190000510979640005
dev:dsk/c0t600144F0DBF8AF190000510979640005d0
```

**例 43** `suriadm map` 構文を使用した iSCSI ベースのストレージリソースの構成

次の例は、`suriadm map` コマンドを使用して iSCSI ベースのストレージリソースを構成する方法を示しています。この例では、iSCSI ストレージ URI に対し、`suriadm` が iSCSI イニシエータの送信先検出アドレスを構成し、iSCSI ターゲットを表すローカルデバイスをインスタンス化します。`mapped-dev` ストレージ URI プロパティからのローカルデバイスパスは、現在、`zpool` コマンド、`format` コマンド、`mkfs` コマンドなどのユーティリティーで使用できます。

```
root@initiator:~# suriadm map iscsi://target/lunname.naa.600144F035FF850000050C884E50001
PROPERTY      VALUE
mapped-dev     /dev/dsk/c0t600144F035FF850000050C884E50001d0s0
```

**例 44** `suriadm lookup-mapping` 構文を使用した構成済みストレージリソースの検索

この例では、コマンドは、指定されたストレージ URI に現在関連付けられているローカルシステムデバイスを表示するために使用されます。

```
root@initiator:~# suriadm lookup-mapping iscsi://target/lunname.naa.600144F035FF8500000050C884E50001
PROPERTY      VALUE
mapped-dev    /dev/dsk/c0t600144F035FF8500000050C884E50001d0s0
```

**例 45** suriadm unmap を使用した iSCSI ベースのストレージリソースの構成解除

suriadm コマンドは、iSCSI ストレージ URI に対して、iSCSI イニシエータの送信先 検出アドレスを削除し、共有ストレージリソースを構成解除します。

```
root@initiator:~# suriadm unmap iscsi://target/lunname.naa.600144F035FF8500000050C884E50001
root@initiator:~# suriadm lookup-mapping iscsi://target/lunname.naa.600144F035FF8500000050C884E50001
Failed to lookup mapping for URI: "iscsi://target/lunname.naa.
600144F035FF8500000050C884E50001": No such logical
unit name found: "naa.600144F035FF8500000050C884E50001"
```

## Oracle Solaris ゾーン への共有ストレージリソースの割り当て

ストレージ URI を使用してストレージオブジェクトの場所を記述することにより、 ゾーン構成で共有ストレージリソースをゾーンに割り当てます。

2つのゾーン構成リソースタイプ (rootzpool および zpools) とプロパティタイプ (storage) は、特定の Oracle Solaris ゾーンに共有ストレージリソースを割り当てるた めに使用します。これらは zonecfg で構成および保守されます。

### ゾーンの storage プロパティ

共有ストレージリソースを使用するときに、storage プロパティは、ストレージ URI を使用して、ホストに依存しない形式でストレージオブジェクトの場所を定義 します。現在、次のストレージ URI が Oracle Solaris の Oracle Solaris ゾーン フレーム ワークでサポートされています。

- dev: ローカルデバイスパスストレージ URI、DAS
- iscsi: iSCSI ストレージ URI
- lu: ファイバチャネル (FC) およびシリアル接続 SCSI (SAS)

ストレージプロパティは、rootzpool または zpools リソーススコープ内から次の zonecfg サブコマンドを使用して管理されます。

```
zonecfg:zonename:zpool> add storage URI-string
zonecfg:zonename:zpool> remove storage URI-string
```

## rootzpool リソース

rootzpool リソースは、ゾーンの専用 ZFS ストレージプールです。ゾーンインストール全体が、専用 ZFS ストレージプールにカプセル化されます。この ZFS ストレージプールは共有ストレージリソースから構成されます。

rootzpool リソースは少なくとも 1 つのストレージプロパティを指定する必要があります。複数のストレージプロパティは、冗長な ZFS ストレージプール構成を記述するように指定できます。rootzpool リソースはゾーン構成ごとに 1 つに限られることに注意してください。

rootzpool リソースの ZFS ストレージプール名は、`zonename_rpool` として自動的に割り当てられます。この名前は変更できません。rootzpool リソースは、グローバルリソーススコープから次の `zonecfg` サブコマンドを使用することによって管理されます。

```
zonecfg:zonename> add rootzpool
zonecfg:zonename:rootzpool> add storage URI-string
zonecfg:zonename:rootzpool> end

zonecfg:zonename> remove rootzpool

zonecfg:zonename> select rootzpool storage=URI-string

zonecfg:zonename> info rootzpool
```

## zpool リソース

zpool リソースは、Oracle Solaris ゾーンに委任された共有ストレージリソースから構成される ZFS ストレージプールを記述します。zpool リソースは少なくとも 1 つのストレージプロパティを指定します。複数のストレージプロパティは、冗長な ZFS ストレージプール構成を記述するように指定できます。ゾーン構成に対して複数の zpool リソースを定義できます。

zpool リソースの ZFS ストレージプール名は、ゾーン名と指定の名前プロパティを組み合わせる (`zonename_name` など) ことによって割り当てられます。名前プロパティの場合、`zonecfg` は、文字列が ZFS ストレージ名と ZFS データセット名として適格であることを検証します。文字列 `rpool` は許可されておらず、このプロパティには使用できません。

zpool リソースは、グローバルリソーススコープから次の `zonecfg` サブコマンドを使用して管理されます。

```
zonecfg:zonename> add zpool
zonecfg:zonename:zpool> add storage URI-string
zonecfg:zonename:zpool> set name=name-string
zonecfg:zonename:zpool> end
```

```
zonecfg:zonename> remove zpool
zonecfg:zonename> remove zpool name=name-string
zonecfg:zonename> remove zpool storage=URI-string

zonecfg:zonename> select zpool storage=URI-string

zonecfg:zonename> info zpool
zonecfg:zonename> info zpool name=name-string
zonecfg:zonename> info zpool storage=URI-string
```

## ゾーンの名前変更

zoneadm コマンドを使用すると、configured または installed のどちらかの状態のゾーンの名前を変更できます。さらに、zonecfg コマンドを使用すると、configured 状態のゾーンの名前を変更できます。詳細については、[129 ページの「ゾーンの名前を変更する方法 \(zoneadm\)」](#) および [130 ページの「ゾーンの名前を変更する方法 \(zonecfg\)」](#) を参照してください。

## ゾーン構成に関する制限事項

既存の共有ストレージリソースの場所の説明が変わったときに rootzpool または zpool リソース内でストレージリソース URI を変更するには、zonecfg remove storage *old-URI* コマンド構文のあとに add storage *new-URI* コマンドを続けて使用します。

## 共有ストレージリソースでの Oracle Solaris ゾーン の自動 ZFS ストレージプール管理

Oracle Solaris での共有ストレージサポート機能でゾーンを使用する場合、ゾーンフレームワークは、特定のゾーンの rootzpool リソースまたは zpool リソースに関連付けられたすべての ZFS ストレージプールを自動的に管理します。

ゾーンおよびストレージリソースの移行を容易にするため、rootzpool リソースまたは zpool リソースに関連付けられた ZFS ストレージプールは、永続的にはシステムで構成されません。これらは、システムのグローバル ZFS ストレージプールリポジトリ /etc/zfs/zpool キャッシュには表示されません。

ゾーンをインストールまたはクローンすると、ゾーンフレームワークは対応する ZFS ストレージプールを作成します。rootzpool または zpool リソースごとに zonecfg

`add storage` コマンドを使用すると複数の共有ストレージリソースを構成できます。この場合、デフォルトで、ミラー化された ZFS ストレージプールが作成されます。

暗号化などのカスタム構成や、`raidz` や `raidz2` などの異なる冗長性レベルを使用することによって、ZFS ストレージプールをあらかじめ作成できます。最初にゾーン構成内で必要なすべての共有ストレージリソースを構成したあと、`zpool` ユーティリティを使用して、事前に作成された ZFS ストレージプールを再度エクスポートします。ゾーンフレームワークは最初に、ゾーンのインストールおよびクローニング中に、この事前に作成された ZFS ストレージプールをインポートして使用しようと試みます。

`zoneadm attach` 構文を使用するときに、ゾーンフレームワークはまず共有ストレージリソースをすべて構成し、次に構成済みの ZFS ストレージプールをすべてインポートします。

`zoneadm detach` コマンド構文の場合、ゾーンフレームワークは、まず構成済み ZFS ストレージプールをすべてエクスポートし、次に共有ストレージリソースをすべて構成解除します。

`zoneadm uninstall` コマンドを使用してゾーンをアンインストールする場合、ゾーンフレームワークはデフォルトで、まず構成済み ZFS ストレージプールをすべてエクスポートし、次に共有ストレージリソースをすべて構成解除します。ただし、`zoneadm uninstall` コマンドで `-x force-zpool-destroy` オプションを使用することによって、ZFS ストレージプールを破棄するように明示的に要求できます。

システムブート中に、ゾーン SMF サービス `svc:/system/zones:default` が実行されます。SMF サービスは、共有ストレージリソースを構成し、`rootzpool` リソースまたは `zpool` リソースを含む `installed` 状態のすべてのゾーンに対して ZFS ストレージプールをインポートします。この段階の障害はすべて、対応する SMF サービスログファイル `/var/svc/log/system-zones:default.log` に記録されます。

`rootzpool` リソースまたは `zpool` リソースで構成された ZFS ストレージプールは、ゾーン内部の仮想化された ZFS ストレージプールとして表示されますが、ゾーン管理者が直接管理することはできません。

実行中のゾーンで使用されている間、大域ゾーンで ZFS ストレージプールを破棄またはエクスポートすることはできません。

大域ゾーンから、ゾーンフレームワークで管理される ZFS ストレージプールでの管理アクションに対して `zpool` コマンドを使用できます。管理アクションには、特定のデバイスのオンライン化、オフライン化、障害の発生したデバイスの交換、デバイスの追加または削除などがあります。ゾーン構成を同期し続けるために、使用されている共有ストレージリソースに加えられた変更を反映するように、対応するストレージ URI を適宜更新します。

## unavailable ゾーン状態について

Oracle Solaris では、unavailable のゾーン状態は、ゾーンがインストールされているがブートできないことを示します。この状態は、`zoneadm list -p` コマンド構文の出力に表示されます。

```
root@initiator:~# zoneadm list -cp
0:global:running:/::solaris:shared::-none
-:iscsi:unavailable:/iscsi:a0a4ba0d-9d6d-cf2c-cc42-f123a5e3ee11:solaris:excl::-:
```

共有ストレージリソースを含むインストール済みゾーンは、ゾーンフレームワークが共有ストレージリソースの構成中の障害や、ZFS ストレージプール管理中の障害に遭遇した場合、unavailable 状態に入ることがあります。

unavailable の状態からゾーンを移すには、最初に共有ストレージリソースの接続やゾーンの構成ミスに関連する問題を特定し、おそらく修正する必要があります。次に `zoneadm attach` コマンドを使用して、ゾーンを適切に再接続し、installed のゾーン状態に移すことができます。 `zoneadm uninstall` コマンドで、ゾーンをアンインストールし、ゾーンを configured 状態に戻すことも可能です。

ゾーン状態の詳細は、『[Introduction to Oracle Solaris Zones](#)』の「[Non-Global Zone State Model](#)」を参照してください。

## 追加の zoneadm サブコマンドオプション

該当する場合は、zoneadm CLI サブコマンドは、ZFS ストレージプール管理専用のオプションで拡張されています。

## ゾーンのインストール、クローニング、接続用のオプション

zoneadm の `install`、`clone`、および `attach` サブコマンドには次のオプションがあります。

- [231 ページの「-x force-zpool-import オプション」](#)
- [231 ページの「-x force-zpool-create= オプション」](#)
- [231 ページの「-x force-zpool-create-all オプション」](#)

## -x force-zpool-import オプション

このオプションは、ゾーン構成で指定されたすべての zpool リソースに適用され、(たとえば、別のシステムによって) 使用されているように見える可能性のあるすべての ZFS ストレージプールを強制的にインポートするようゾーンフレームワークに指示します。これは、zpool import -f コマンドの動作を模倣します。

このオプションは、たとえば、zoneadm detach コマンドや zoneadm uninstall コマンドを使用して ZFS ストレージプールが別のシステムに適切にエクスポートされなかった場合に、ゾーンの移行中に新しいシステムに ZFS ストレージプールを簡単にインポートできるようにします。これらの状況において、特定のゾーンに対して構成されたすべての ZFS ストレージプールに、強制インポートを適用する必要があります。

## -x force-zpool-create= オプション

このオプションは、既存の古いプールに対して、またはほかの目的で現在使用されているように見えるデバイス上に新しい ZFS ストレージプールを強制的に作成するように、ゾーンフレームワークに対して指示します。これは、zpool create -f コマンド構文の動作を模倣します。

このオプションの範囲は、たとえば、zpool 名のコンマ区切りのリストで指定されたり、オプションを複数回使用して指定された ZFS ストレージプールに限られます。

```
-x force-zpool-create=rpool,pool2,pool3
-x force-zpool-create=pool1 -x force-zpool-create=pool2
```

ここで使用される ZFS ストレージプール名は、ゾーン構成内の対応する zpool リソースの名前プロパティです。rootzpool リソースによって記述された ZFS ストレージプールを指定するには、rpool の名前を使用します。

## -x force-zpool-create-all オプション

このオプションは、ゾーン構成で指定されたすべての zpool リソースおよび rootzpool リソースに対して新しい ZFS ストレージプールを強制的に作成するように、ゾーンフレームワークに指示します。

## ゾーンのアンインストール用オプション

zoneadm uninstall サブコマンドには次のオプションがあります。

- [232 ページの「-x force-zpool-destroy=zpoolname オプション」](#)
- [232 ページの「-x force-zpool-destroy-all オプション」](#)

- 232 ページの「-x force-storage-destroy-all オプション」

## -x force-zpool-destroy=zpoolname オプション

このオプションは、ゾーンをアンインストールするときに、ZFS ストレージプールを破棄するように、ゾーンフレームワークに指示します。これは、`zpool destroy poolname` コマンドの動作を模倣します。このオプションには *zpoolname* リストを指定します。

```
force-zpool-destroy=zpoolname{,zpoolname,zpoolname,...}
```

このオプションの範囲は、`zpool` 名のコンマ区切りリストで指定されるか、またはこのオプションを複数回使用して指定される ZFS ストレージプールに制限されません。次に例を示します。

```
-x force-zpool-destroy=rpool,pool2,pool3
-x force-zpool-destroy=pool1 -x force-zpool-destroy=pool2
```

名前引数は、目的の `zpool` リソースについてゾーン構成で指定された名前プロパティに対応します。名前 `rpool` は、`rootzpool` リソースに関連付けられた ZFS ストレージプールを指定するために使用されます。

## -x force-zpool-destroy-all オプション

このオプションは、ゾーンをアンインストールするときに、ゾーン構成で指定されたすべての `zpool` リソースおよび `rootzpool` リソースの ZFS ストレージプールを破棄するように、ゾーンフレームワークに指示します。

ゾーンのアンインストール時に ZFS ストレージプールを破棄するようにゾーンフレームワークに強制するオプションは、慎重に使用する必要があります。現在アクティブなブート環境 (BE) でゾーンをアンインストールする場合でも、現在非アクティブなブート環境に所属しているほかのゾーンブート環境 (ZBE) が、この ZFS ストレージプール内に存在する可能性があります。`rootzpool` リソースに関連付けられた ZFS ストレージプールを破棄すると、`installed` 状態でゾーンを保持するすべてのブート環境で、ゾーンが利用できなくなります。ゾーンブート環境の詳細については、[beadm\(8\)](#) のマニュアルページを参照してください。

## -x force-storage-destroy-all オプション

ストレージを破棄する場合は、このオプションを指定します。すべてのストレージ URI タイプがこの操作をサポートしているわけではないことに注意してください。ファイルまたは NFS ストレージ URI によって参照されているストレージを破棄でき



ますが、dev、iSCSI、または lu ストレージ URI によって参照されているストレージは破棄できません。

## rootzpool リソースの移動の制限

rootzpool リソースで構成されたゾーンの場合、zoneadm move コマンドは、ZFS ストレージプールのパス名を変更できません。ただし、zoneadm move コマンドは、ZFS ストレージプールの基となるストレージデバイスを変更できます。

# 共有ストレージリソースでホストされる Oracle Solaris ゾーンの実装

次のセクションでは、共有ストレージリソースでの Oracle Solaris ゾーン の配備について説明します。

## 共有ストレージ上のゾーンに関する zpools 構成の考慮事項

ZFS ストレージプールを構成する場合、ミラー、raidz、raidz2 などの冗長なプール構成を使用することができます。共有ストレージリソースをホストするバックエンドストレージで、たとえばハードウェア RAID の形式で障害許容力およびデータ保護が確保されている場合でも、冗長な構成によって、ZFS ストレージプールは固有の自己回復機能を使用できるようになります。

Oracle Solaris ゾーン を共有ストレージリソースでホストする冗長な ZFS ストレージプール構成を使用する 2 つの方法があります。

- ミラー化した ZFS ストレージプール構成を使用できます。これはデフォルトで、rootzpool リソースまたは zpools リソースのストレージプロパティをゾーン構成で複数指定した場合、自動的に作成されます。
- または、zpools create コマンドで、raidz や raidz2 などのより複雑な ZFS ストレージプール構成をあらかじめ作成しておくこともできます。zpools export を使用して、新しく作成した ZFS ストレージプールを再度エクスポートする必要があることに注意してください。次に、zonecfg コマンドを使用して、ゾーン構成で、すべての対応するストレージリソース URI を rootzpool リソースまたは zpools リソースに追加する必要があります。ゾーンフレームワークは次に、新しい ZFS ストレージプールを作成しようと試みるのではなく、ゾーンのインストールまたはクローン中にこの ZFS ストレージプールをインポートします。

## 共有ストレージ上のゾーンのサンプルシナリオ

このセクションでは、共有ストレージリソースに Oracle Solaris ゾーン を配備する例を示します。

### 例 46 iSCSI ベースの共有ストレージデバイスを使用した Oracle Solaris ゾーン

この演習では、iSCSI ターゲット経由で共有ストレージを提供するサンプル構成を Oracle Solaris サーバー上で設定します。ゾーンは Oracle Solaris が動作している 2 番目のサーバーに構成およびインストールされ、iSCSI ベースの共有ストレージリソースを使用してゾーンをホストします。

最初に、次の `pkg install` コマンドのいずれかを使用して、対応するパッケージをインストールします。最初のコマンドは、マルチプロトコルストレージサーバーグループパッケージ全体をインストールします。`itadm(8)` および `stmfadm(8)` のマニュアルページに説明されているように、2 番目のコマンドは、一般的なマルチプロトコル SCSI ターゲット (COMSTAR) フレームワーク内で iSCSI のターゲットサポートだけをインストールします。

```
root@target:~# pkg install group/feature/storage-server
root@target:~# pkg install system/storage/iscsi/iscsi-target
```

次に、このサーバーから iSCSI ターゲットをエクスポートするためのバックイングストアを作成します。`zfs` コマンドを使用して、4 つの iSCSI ターゲット論理ユニット用のバックイングストアとして、ターゲットサーバーの `rpool/export` データセット内に格納される 4 つの ZFS ボリューム (サイズはそれぞれ 10G バイト) を作成します。

```
root@target:~# zfs create -V 10G rpool/export/zonevol1
root@target:~# zfs create -V 10G rpool/export/zonevol2
root@target:~# zfs create -V 10G rpool/export/zonevol3
```

バックイングストアを設定したあと、`stmfadm` コマンドを使用して、各 ZFS ボリュームのターゲット論理ユニットを作成します。これにより、それぞれに対応したデバイス ID (WWN) が与えられ、後からストレージ URI でクライアントホストでの iSCSI ターゲット検出に使用されます。

```
root@target:~# stmfadm create-lu /dev/zvol/rdisk/rpool/export/zonevol1
Logical unit created: 600144F035FF8500000050C884E50001
root@target:~# stmfadm create-lu /dev/zvol/rdisk/rpool/export/zonevol2
Logical unit created: 600144F035FF8500000050C884E80002
root@target:~# stmfadm create-lu /dev/zvol/rdisk/rpool/export/zonevol3
Logical unit created: 600144F035FF8500000050C884EC0003
```

`stmfadm list-lu` 構文で、構成された論理ユニットを表示できます。

```
root@target:~# stmfadm list-lu
LU Name: 600144F035FF8500000050C884E50001
LU Name: 600144F035FF8500000050C884E80002
LU Name: 600144F035FF8500000050C884EC0003
LU Name: 600144F035FF8500000050C884EC0004
```

`stmfadm list-lu -v` 構文で、構成済みの論理ユニットの詳細を照会できます。

```

root@target:~# stmfadm list-lu -v
LU Name: 600144F035FF850000050C884E50001
  Operational Status      : Online
  Provider Name           : sbd
  Alias                    : /dev/zvol/rdisk/rpool/export/zonevol1
  View Entry Count        : 0
  Data File                : /dev/zvol/rdisk/rpool/export/zonevol1
  Meta File                : not set
  Size                     : 10737418240
  Block Size               : 512
  Management URL           : not set
  Software ID              : not set
  Vendor ID                : SUN
  Product ID               : COMSTAR
  Serial Num                : not set
  Write Protect            : Disabled
  Write Cache Mode Select : Enabled
  Writeback Cache          : Enabled
  Access State             : Active

```

論理ユニットを iSCSI イニシエータで使用できるようにするには、`stmfadm add-view` コマンドで、ターゲットサーバーに論理ユニットビューを追加します。

```

root@target:~# stmfadm add-view 600144F035FF850000050C884E50001
root@target:~# stmfadm add-view 600144F035FF850000050C884E80002
root@target:~# stmfadm add-view 600144F035FF850000050C884EC0003

```

次に、ターゲットサーバー上で iSCSI ターゲットを構成します。最初に `svcadm enable` を使用して、iSCSI ターゲット SMF サービスを有効にします。

```

root@target:~# svcadm enable -r svc:/network/iscsi/target:default

```

次に、`itadm create-target` を使用して iSCSI ターゲット自体を作成します。

```

root@target:~# itadm create-target
Target iqn.1986-03.com.sun:02:b62a8291-b89e-41ba-9aef-e93836ad0d6a successfully created

```

`itadm list-target` または `stmfadm list-target` のどちらかを使用して、構成済みの iSCSI ターゲットに関する詳細をクエリーできます。

```

root@target:~# itadm list-target -v
TARGET NAME                                STATE    SESSIONS
iqn.1986-03.com.sun:02:b62a8291-b89e-41ba-9aef-e93836ad0d6a  online  0
  alias:                                     -
  auth:                                       none (defaults)
  targetchapuser:                             -
  targetchapsecret:                           unset
  tpg-tags:                                    default

root@target:~# stmfadm list-target -v
Target: iqn.1986-03.com.sun:02:b62a8291-b89e-41ba-9aef-e93836ad0d6a
  Operational Status      : Online
  Provider Name           : iscsit
  Alias                    : -
  Protocol                 : iSCSI
  Sessions                 : 0

```

これにより、iSCSI ターゲットストレージを提供するサンプルサーバー上で実行されるタスクが完了します。

次の手順は、iSCSI 経由で提供されるこの共有ストレージを使用して、2 番目のサーバー上でゾーンを構成およびインストールするためのものです。

最初の手順として、iSCSI イニシエータとして選択されたクライアントサーバー上で対応するパッケージをインストールします。

```
root@initiator:~# pkg install pkg:/system/storage/iscsi/iscsi-initiator
```

次に、zonecfg コマンドを使用して、rootzpool リソースおよび zpoo1 リソースでゾーンを構成します。共有ストレージリソースとして構成した 3 つの iSCSI ターゲット論理ユニットを使用して、そのゾーンをホストします。ターゲットサーバー上で、以前に suriadm で取得した iSCSI ストレージ URI を使用します。

```
root@initiator:~# zonecfg -z iscsi
Use 'create' to begin configuring a new zone.
zonecfg:iscsi> create
create: Using system default template 'SYSdefault'
zonecfg:iscsi> add rootzpool
zonecfg:iscsi:rootzpool> add storage iscsi://target/luname.naa.600144F035FF8500000050C884E50001
zonecfg:iscsi:rootzpool> end
zonecfg:iscsi> add zpool
zonecfg:iscsi:zpool> set name=data
zonecfg:iscsi:zpool> add storage iscsi://target/luname.naa.600144F035FF8500000050C884E80002
zonecfg:iscsi:zpool> add storage iscsi://target/luname.naa.600144F035FF8500000050C884EC0003
zonecfg:iscsi:zpool> end
zonecfg:iscsi> commit
zonecfg:iscsi> exit
```

zoneadm install を使用してゾーンをインストールします。

```
root@initiator:~# zoneadm -z iscsi install
Configured zone storage resource(s) from:
  iscsi://target/luname.naa.600144F035FF8500000050C884E50001
Created zone zpool: iscsi_rpool
Configured zone storage resource(s) from:
  iscsi://target/luname.naa.600144F035FF8500000050C884E80002
  iscsi://target/luname.naa.600144F035FF8500000050C884EC0003
Created zone zpool: iscsi_data
Progress being logged to /var/log/zones/zoneadm.20130125T112209Z.iscsi.install
Image: Preparing at /iscsi/root.

AI Manifest: /tmp/manifest.xml.pmai7h
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Zonename: iscsi
Installation: Starting ...

      Creating IPS image
Startup linked: 1/1 done
      Installing packages from:
        solaris
          origin: http://pkg.oracle.com/solaris/release/
DOWNLOAD
Completed          PKGS      FILES      XFER (MB)   SPEED
                   183/183    33556/33556  222.2/222.2  3.4M/s

PHASE              ITEMS
Installing new actions 46825/46825
Updating package state database      Done
Updating image state                  Done
Creating fast lookup database        Done
Installation: Succeeded
```

Note: Man pages can be obtained by installing pkg:/system/manual  
done.

Done: Installation completed in 266.487 seconds.

Next Steps: Boot the zone, then log into the zone console (zlogin -C)  
to complete the configuration process.

Log saved in non-global zone as /iscsi/root/var/log/zones/  
zoneadm.20130125T112209Z.iscsi.install  
root@initiator:~#

ゾーンのインストールが完了したら、`zoneadm list` コマンドを使用して、ゾーンが正しくインストールされたことを確認します。

```
root@initiator:~# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:installed:/iscsi:a0a4ba0d-9d6d-cf2c-cc42-f123a5e3ee11:solaris:excl:-:
```

`zpool` コマンドを使用して、このゾーンに関連付けられた、新しく作成した ZFS ストレージプールを監視します。

```
root@initiator:~# zpool list
NAME          SIZE  ALLOC  FREE  CAP  DEDUP  HEALTH  ALTRoot
iscsi_data    9.94G  83.5K  9.94G  0%   1.00x  ONLINE  -
iscsi_rpool   9.94G  436M   9.51G  4%   1.00x  ONLINE  -
```

```
root@initiator:~# zpool status -v iscsi_rpool
pool: iscsi_rpool
state: ONLINE
scan: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM
iscsi_rpool	ONLINE	0	0	0
c0t600144F035FF850000050C884E50001d0	ONLINE	0	0	0

```
root@initiator:~# zpool status -v iscsi_data
pool: iscsi_data
state: ONLINE
scan: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM
iscsi_data	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c0t600144F035FF850000050C884E80002d0	ONLINE	0	0	0
c0t600144F035FF850000050C884EC0003d0	ONLINE	0	0	0

ゾーンのインストール全体がこの ZFS ストレージプール内に含まれます。このゾーンの ZFS データセットレイアウトは次のとおりです。

```
root@initiator:~# zfs list -t all|grep iscsi
iscsi_data          83.5K  9.78G   31K  /iscsi_data
iscsi_rpool        436M   9.36G   32K  /iscsi
iscsi_rpool/rpool  436M   9.36G   31K  /rpool
iscsi_rpool/rpool/ROOT  436M   9.36G   31K  legacy
iscsi_rpool/rpool/ROOT/solaris  436M   9.36G  390M  /iscsi/root
iscsi_rpool/rpool/ROOT/solaris@install  64K    -    390M  -
iscsi_rpool/rpool/ROOT/solaris/var  46.1M  9.36G  45.4M  /iscsi/root/var
iscsi_rpool/rpool/ROOT/solaris/var@install  644K    -  45.4M  -
```

```

iscsi_rpool/rpool/VARSHARE          31K  9.36G  31K  /var/share
iscsi_rpool/rpool/export            62K  9.36G  31K  /export
iscsi_rpool/rpool/export/home      31K  9.36G  31K  /export/home

```

iSCSI ベースの共有ストレージリソースでホストされる新しいゾーンは正しくインストールされており、`zoneadm boot` を使用してブートできるようになっています。

ゾーンがブートされたあと、ゾーン管理者は、仮想化された ZFS データセットおよびストレージプールをゾーン内から監視します。

```

root@iscsi:~# zpool list
NAME      SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALROOT
data     9.94G   85K  9.94G   0%  1.00x  ONLINE  -
rpool    9.94G  449M  9.50G   4%  1.00x  ONLINE  -

root@iscsi:~# zpool status -v
pool: data
state: ONLINE
scan: none requested
config:

      NAME                                STATE      READ WRITE CKSUM
data   data                                ONLINE      0     0     0
      mirror-0
      c0t600144F035FF850000050C884E80002d0  ONLINE      0     0     0
      c0t600144F035FF850000050C884EC0003d0  ONLINE      0     0     0

pool: rpool
state: ONLINE
scan: none requested
config:

      NAME                                STATE      READ WRITE CKSUM
rpool  rpool                                ONLINE      0     0     0
      c0t600144F035FF850000050C884E50001d0  ONLINE      0     0     0

root@iscsi:~# zfs list -t all
NAME                                     USED  AVAIL  REFER  MOUNTPOINT
data                                     85K  9.78G  31K    /data
rpool                                    464M  9.33G  31K    /rpool
rpool/ROOT                              464M  9.33G  31K    legacy
rpool/ROOT/solaris                      464M  9.33G  416M   /
rpool/ROOT/solaris@install              1.83M  -      390M   -
rpool/ROOT/solaris/var                  46.2M  9.33G  45.6M  /var
rpool/ROOT/solaris/var@install          674K  -      45.4M  -
rpool/VARSHARE                          39K   9.33G  39K    /var/share
rpool/export                            96.5K  9.33G  32K    /export
rpool/export/home                       64.5K  9.33G  32K    /export/home
rpool/export/home/user                  32.5K  9.33G  32.5K  /export/home/user

```

#### 例 47 DAS ストレージデバイスを使用した Oracle Solaris ゾーン の例

この演習では、直接接続したローカルストレージデバイスを使用して、Oracle Solaris 上でゾーンを構成しインストールします。この方法は通常、異なるホスト間で移植できないことに注意してください。

最初に、`format` コマンドを使用して、利用可能なローカルディスクを検出します。次に、`suriadm lookup-uri` を使用して、ゾーン構成内で使用される対応するストレージ URI を構築します。

```

root@host:~# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  1. c4t1d0 <SEAGATE-ST336704LSUN36G-0326-33.92GB>
     /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@1,0
  2. c4t2d0 <FUJITSU-MAT3073NC-0104-68.49GB>
     /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@2,0
  3. c4t3d0 <SEAGATE-ST336704LSUN36G-0326-33.92GB>
     /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@3,0
  4. c4t4d0 <FUJITSU-MAW3073NC-0103-68.49GB>
     /pci@0,0/pci1022,7450@a/pci17c2,20@4/sd@4,0

```

```

root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t1d0
dev:dsk/c4t1d0
root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t2d0
dev:dsk/c4t2d0
root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t3d0
dev:dsk/c4t3d0
root@host:~# suriadm lookup-uri -t dev /dev/dsk/c4t4d0
dev:dsk/c4t4d0

```

これらのストレージ URI を使用して、rootzpool および zpool リソース (どちらもミラー化された ZFS ストレージプールを示す) でゾーンを構成します。

```

root@host:~# zonecfg -z disk
Use 'create' to begin configuring a new zone.
zonecfg:disk> create
create: Using system default template 'SYSdefault'
zonecfg:disk> add rootzpool
zonecfg:disk:rootzpool> add storage dev:dsk/c4t1d0
zonecfg:disk:rootzpool> add storage dev:dsk/c4t3d0
zonecfg:disk:rootzpool> end
zonecfg:disk> add zpool
zonecfg:disk:zpool> set name=dpool
zonecfg:disk:zpool> add storage dev:dsk/c4t2d0
zonecfg:disk:zpool> add storage dev:dsk/c4t4d0
zonecfg:disk:zpool> end
zonecfg:disk> commit
zonecfg:disk> exit

```

次にゾーンをインストールします。

```

root@host:~# zoneadm -z disk install
Created zone zpool: disk_rpool
Created zone zpool: disk_dpool
Progress being logged to /var/log/zones/zoneadm.20130213T132236Z.disk.install
Image: Preparing at /disk/root.

AI Manifest: /tmp/manifest.xml.r0a0he
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Zonename: disk
Installation: Starting ...

Creating IPS image
Startup linked: 1/1 done
Installing packages from:
solaris
origin: http://pkg.oracle.com/solaris/release/
DOWNLOAD          PKGS          FILES          XFER (MB)   SPEED
Completed          183/183       33556/33556    222.2/222.2  2.0M/s

PHASE              ITEMS
Installing new actions 46825/46825
Updating package state database Done

```

```

Updating image state           Done
Creating fast lookup database Done
Installation: Succeeded

```

Note: Man pages can be obtained by installing pkg:/system/manual done.

Done: Installation completed in 308.358 seconds.

Next Steps: Boot the zone, then log into the zone console (zlogin -C) to complete the configuration process.

Log saved in non-global zone as /disk/root/var/log/zones/zoneadm.20130213T132236Z.disk.install

ゾーンのインストール後、次の2つの新しいZFSストレージプールがオンラインになります。

```

root@host:~# zpool list
NAME      SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
disk_dpoo 68G   83.5K  68.0G   0%  1.00x  ONLINE  -
disk_rpool 33.8G  434M   33.3G   1%  1.00x  ONLINE  -

```

```

root@host:~# zpool status -v disk_rpool
pool: disk_rpool
state: ONLINE
scan: none requested
config:

```

NAME	STATE	READ	WRITE	CKSUM
disk_rpool	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c4t1d0	ONLINE	0	0	0
c4t3d0	ONLINE	0	0	0

```

root@host:~# zpool status -v disk_dpoo
pool: disk_dpoo
state: ONLINE
scan: none requested
config:

```

NAME	STATE	READ	WRITE	CKSUM
disk_dpoo	ONLINE	0	0	0
mirror-0	ONLINE	0	0	0
c4t2d0	ONLINE	0	0	0
c4t4d0	ONLINE	0	0	0

ゾーンのインストール全体がこのZFSストレージプール内に含まれます。このゾーンのZFSデータセットレイアウトは次のとおりです。

```

root@host:~# zfs list -t all|grep disk
disk_dpoo          83.5K 66.9G 31K /disk_dpoo
disk_rpool         434M 32.8G 32K /disk
disk_rpool/rpool  433M 32.8G 31K /rpool
disk_rpool/rpool/ROOT 433M 32.8G 31K legacy
disk_rpool/rpool/ROOT/solaris 433M 32.8G 389M /disk/root
disk_rpool/rpool/ROOT/solaris@install 63K - 389M -
disk_rpool/rpool/ROOT/solaris/var 43.8M 32.8G 43.2M /disk/root/var
disk_rpool/rpool/ROOT/solaris/var@install 584K - 43.2M -
disk_rpool/rpool/VARSHARE 31K 32.8G 31K /var/share
disk_rpool/rpool/export 62K 32.8G 31K /export

```



```
disk_rpool/rpool/export/home                31K 32.8G    31K /export/home
```

ローカルデバイスストレージリソースでホストされる新しいゾーンは正しくインストールされており、`zoneadm boot` コマンドを使用してブートできるようになっています。

ゾーンがブートされたあと、ゾーン管理者は、仮想化された ZFS データセットおよびストレージプールをゾーン内部から監視できます。

```
root@disk:~# zpool list
NAME      SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
dpool    68G   83.5K  68.0G   0%  1.00x  ONLINE  -
rpool    33.8G   472M  33.3G   1%  1.00x  ONLINE  -

root@disk:~# zpool status -v
pool: dpool
state: ONLINE
scan: none requested
config:

      NAME      STATE      READ WRITE CKSUM
      dpool     ONLINE      0     0     0
      mirror-0  ONLINE      0     0     0
      c4t2d0    ONLINE      0     0     0
      c4t4d0    ONLINE      0     0     0

pool: rpool
state: ONLINE
scan: none requested
config:

      NAME      STATE      READ WRITE CKSUM
      rpool     ONLINE      0     0     0
      mirror-0  ONLINE      0     0     0
      c4t1d0    ONLINE      0     0     0
      c4t3d0    ONLINE      0     0     0

root@disk:~# zfs list -t all
NAME                                     USED  AVAIL  REFER  MOUNTPOINT
dpool                                   83.5K 66.9G   31K    /dpool
rpool                                   465M 32.8G   31K    /rpool
rpool/ROOT                              465M 32.8G   31K    legacy
rpool/ROOT/solaris                      465M 32.8G  416M    /
rpool/ROOT/solaris@install              5.60M -       389M    -
rpool/ROOT/solaris/var                  43.9M 32.8G  43.3M    /var
rpool/ROOT/solaris/var@install          618K -       43.2M    -
rpool/VARSHARE                          39K 32.8G   39K    /var/share
rpool/export                            96.5K 32.8G   32K    /export
rpool/export/home                       64.5K 32.8G   32K    /export/home
rpool/export/home/user                  32.5K 32.8G  32.5K    /export/home/user
```

**例 48** ファイバチャネルベースのストレージデバイスを使用した Oracle Solaris ゾーン

この演習では、ファイバチャネルを介して提供される共有ストレージデバイスを使用して、Oracle Solaris 上でゾーンを構成しインストールします。

最初に、`fcinfo lu` コマンドを使用することによって、現在ホストから確認できるファイバチャネルの論理ユニットを検出します。

```

root@host:~# fcinfo lu -v
OS Device Name: /dev/rdsk/c0t600144F0DBF8AF190000564C9AB9001Bd0s2
HBA Port WWN: 21000024ff3ee788
Controller: /dev/cfg/c2
Remote Port WWN: 21000024ff3ee89e
LUN: 0
State: active/optimized
Vendor: SUN
Product: ZFS Storage 7120
Device Type: Disk Device
Unformatted capacity: 102400.000 MBytes

```

suriadm lookup-uri を使用して、デバイスパスに基づいてストレージ URI を構築します。クエリーで LU 全体を表すストレージ URI を取得できるように、デバイス名のスライス部分を削除します。

```

root@host:~# suriadm lookup-uri /dev/dsk/c0t600144F0DBF8AF190000564C9AB9001Bd0
dev:dsk/c0t600144F0DBF8AF190000564C9AB9001Bd0
lu:luname.naa.600144f0dbf8af190000564c9ab9001b
lu:initiator.naa.21000024ff3ee788,target.naa.21000024ff3ee89e,luname.naa.600144f0dbf8af190000564c9ab9001b

```

表示された 3 つの URI から、ゾーン構成で使用する、luname のみの形式の論理ユニットストレージ URI を選択します。

```

root@host:~# zonecfg -z fc
Use 'create' to begin configuring a new zone.
zonecfg:fc> create
create: Using system default template 'SYSdefaultzonecfg:fc> add rootzpool
zonecfg:fc:rootzpool> add storage lu:luname.naa.600144f0dbf8af190000564c9ab9001b
zonecfg:fc:rootzpool> end
zonecfg:fc> commit
zonecfg:fc> exit

```

ゾーンをインストールします。

```

root@host:~# zoneadm -z fc install
Created zone zpool: fc_rpool
Progress being logged to /var/log/zones/zoneadm.20130214T045957Z.fc.install
Image: Preparing at /fc/root.

```

```

AI Manifest: /tmp/manifest.xml.K9aaow
SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
Zonename: fc
Installation: Starting ...

```

```

Creating IPS image
Startup linked: 1/1 done
Installing packages from:
solaris
origin: http://pkg.oracle.com/solaris/release/
DOWNLOAD PKGS FILES XFER (MB) SPEED
Completed 190/190 34246/34246 231.3/231.3 7.2M/s

PHASE ITEMS
Installing new actions 48231/48231
Updating package state database Done
Updating image state Done
Creating fast lookup database Done
Installation: Succeeded

```

Note: Man pages can be obtained by installing pkg:/system/manual

done.

Done: Installation completed in 104.318 seconds.

Next Steps: Boot the zone, then log into the zone console (zlogin -C)

to complete the configuration process.

Log saved in non-global zone as /fc/root/var/log/zones/zoneadm.20130214T045957Z.fc.install

ゾーンのインストール後、次の新しい ZFS ストレージプールがオンラインになります。

```
root@host:~# zpool list
NAME      SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
fc_rpool  39.8G  441M   39.3G   1%  1.00x  ONLINE  -
```

```
root@host:~# zpool status -v fc_rpool
pool: fc_rpool
state: ONLINE
scan: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM
fc_rpool	ONLINE	0	0	0
c0t600144F0DBF8AF190000564C9AB9001Bd0	ONLINE	0	0	0

ゾーンのインストール全体がこの ZFS ストレージプール内に含まれます。ゾーンには次の ZFS データセットレイアウトがあります。

```
root@host:~# zfs list -t all|grep fc
fc_rpool                               440M  38.7G   32K  /fc
fc_rpool/rpool                         440M  38.7G   31K  /rpool
fc_rpool/rpool/ROOT                    440M  38.7G   31K  legacy
fc_rpool/rpool/ROOT/solaris            440M  38.7G  405M  /fc/root
fc_rpool/rpool/ROOT/solaris@install    67K   -      405M  -
fc_rpool/rpool/ROOT/solaris/var        34.3M  38.7G  33.6M  /fc/root/var
fc_rpool/rpool/ROOT/solaris/var@install 665K   -     33.6M  -
fc_rpool/rpool/VARSHARE                 31K   38.7G   31K  /var/share
fc_rpool/rpool/export                  62K   38.7G   31K  /export
fc_rpool/rpool/export/home              31K   38.7G   31K  /export/home
```

ファイバチャネルターゲットから提供される共有ストレージでホストされる新しいゾーンは、正しくインストールされました。これで、`zoneadm boot` を使用してこのゾーンをブートできます。

ゾーンがブートされたあと、ゾーン管理者は、仮想化された ZFS データセットおよびストレージプールをゾーン内部から監視できます。

```
root@fc:~# zpool list
NAME      SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool    39.8G  451M   39.3G   1%  1.00x  ONLINE  -
```

```
root@fc:~# zpool status -v
pool: rpool
state: ONLINE
scan: none requested
config:
```

NAME	STATE	READ	WRITE	CKSUM
------	-------	------	-------	-------

```

rpool                                     ONLINE      0      0      0
c0t600144F0DBF8AF190000564C9AB9001Bd0 ONLINE      0      0      0

root@fc:~# zfs list -t all
NAME                                     USED  AVAIL  REFER  MOUNTPOINT
rpool                                     467M  38.7G  31K    /rpool
rpool/ROOT                               467M  38.7G  31K    legacy
rpool/ROOT/solaris                       467M  38.7G  430M   /
rpool/ROOT/solaris@install               1.90M  -      405M   -
rpool/ROOT/solaris/var                   34.4M  38.7G  33.7M  /var
rpool/ROOT/solaris/var@install           703K  -      33.6M  -
rpool/VARSHARE                           39K   38.7G  39K    /var/share
rpool/export                             96.5K  38.7G  32K    /export
rpool/export/home                        64.5K  38.7G  32K    /export/home
rpool/export/home/user                   32.5K  38.7G  32.5K  /export/home/user

```

## 共有ストレージリソースでホストされる Oracle Solaris ゾーン の移行

ゾーン構成のストレージ URI を使用した共有ストレージでホストされるゾーンの移行は、明瞭で単純なプロセスです。このプロセスに必要な唯一の CLI は、`zoneadm` コマンドと `zonecfg` コマンドです。ストレージリソースまたは ZFS ストレージプールを含むゾーンの移行には、追加コマンドを実行する必要はありません。

### 例 49 iSCSI 共有ストレージに基づくゾーンの移行

この例では、iSCSI 共有ストレージに基づいて、現在のホストから新しいホストにゾーンを移行するための手順を示します。

```

root@initiator:~# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:installed:/iscsi:a0a4ba0d-9d6d-cf2c-cc42-f123a5e3ee11:solaris:excl:-:

```

最初の手順として、現在のホストのゾーンを切り離します。ゾーンは、`installed` 状態から `configured` 状態に移ります。すべての ZFS ストレージプールはエクスポートされ、共有ストレージリソースは自動的に構成解除されます。

```

root@initiator:~# zoneadm -z iscsi detach
Exported zone zpool: iscsi_rpool
Unconfigured zone storage resource(s) from:
  iscsi://target/lunname.naa.600144F035FF8500000050C884E50001
Exported zone zpool: iscsi_data
Unconfigured zone storage resource(s) from:
  iscsi://target/lunname.naa.600144F035FF8500000050C884E80002
  iscsi://target/lunname.naa.600144F035FF8500000050C884EC0003

```

```

root@initiator:~# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:configured:/iscsi::solaris:excl:-:

```

現在のホストで必要な最後の手順として、`zonecfg export` を使用してゾーン構成をエクスポートし、作成されたファイルを新しいホストに転送します。

```
root@initiator:/# zonecfg -z iscsi export -f /export/iscsi.cfg
```

新しいホストで、zonecfg コマンドを使用して、保存したファイルからゾーン構成を最初にインスタンス化します。ゾーンは、configured 状態になります。

```
root@newhost:/# zonecfg -z iscsi -f /export/iscsi.cfg
```

```
root@newhost:/# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:configured:/iscsi::solaris:excl:-:
```

次に、zoneadm attach を使用して、新しいホスト上でゾーンを接続します。ゾーンは installed 状態に移ります。すべての共有ストレージリソースが構成され、対応する ZFS ストレージプールが自動的にインポートされます。

```
root@newhost:/# zoneadm -z iscsi attach
Configured zone storage resource(s) from:
    iscsi://target/luname.naa.600144F035FF8500000050C884E50001
Imported zone zpool: iscsi_rpool
Configured zone storage resource(s) from:
    iscsi://target/luname.naa.600144F035FF8500000050C884E80002
    iscsi://target/luname.naa.600144F035FF8500000050C884EC0003
Imported zone zpool: iscsi_data
Progress being logged to /var/log/zones/zoneadm.20130214T145001Z.iscsi.attach
Installing: Using existing zone boot environment
    Zone BE root dataset: iscsi_rpool/rpool/ROOT/solaris
    Cache: Using /var/pkg/publisher.
    Updating non-global zone: Linking to image /.
Processing linked: 1/1 done
    Updating non-global zone: Auditing packages.
No updates necessary for this image.

    Updating non-global zone: Zone updated.
    Result: Attach Succeeded.
Log saved in non-global zone as /iscsi/root/var/log/zones/
zoneadm.20130214T145001Z.iscsi.attach
```

```
root@newhost:/# zoneadm list -cp
0:global:running:/::solaris:shared:-:none
-:iscsi:installed:/iscsi:a19fbb45-4af3-670f-c58e-ee48757c75d6:solaris:excl:-:
```

ゾーンは、新しいホストに移行されており、zoneadm boot を使用してブートする準備が整っています。

このプロセスは基本的に、Oracle Solaris でサポートされている 3 つの異なるタイプのストレージ URI で同じです。iSCSI ベースまたはファイバチャネルベースのストレージリソースの場合、新しいホストが、同じ論理ユニットおよびターゲットポートにアクセスできることも確認する必要があります。

## 共有ストレージゾーン構成に対する既存のゾーンの移動

既存のファイルシステムベースのゾーンインストールを共有ストレージベースのゾーン構成に移動できます。さらに、共有ストレージリソースからファイルシステム構成にインストール済みゾーンを移動することもできます。

---

**注記** - このプロセスには、ストレージ間でのゾーンインストールの移動が含まれ、これらのアクションは `installed` 状態のゾーンに対してのみ可能です。

---

Oracle Solaris 11.4 では、`rootzpool` リソースを含むゾーンのゾーンパスは変更できません。solaris ブランドでサポートしている特定のデフォルトについては、[solaris\(7\)](#) のマニュアルページを参照してください。

次の手順では、これらのゾーン構成の変更を行う方法を説明します。

- [246 ページの「共有ストレージ構成にゾーンを移動する方法」](#)
- [247 ページの「共有ストレージ構成からゾーンを移動する方法」](#)
- [248 ページの「別の共有ストレージにゾーンを移動する方法」](#)
- [249 ページの「インストール済みゾーンに ZFS ストレージプールを追加または移動する方法」](#)

共有ストレージに対するもの以外のゾーンの移動については、[150 ページの「共有ストレージ上にないゾーンを移動する方法」](#)に進みます。

## ▼ 共有ストレージ構成にゾーンを移動する方法

ローカルファイルシステムから共有ストレージに solaris ゾーンを移動するには、この手順を実行します。zoneadm move コマンドは、ゾーンで使用する新しい ZFS ストレージプールを作成するか、または既存のストレージプールをインポートするオプションを提供します。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. ゾーンが `installed` 状態であることを確認します。

```
global# pfbash zoneadm list -cv
ID NAME          STATUS  PATH                                BRAND  IP
0 global         running /                                     solaris shared
- zonename      installed /system/zones/zonename          solaris excl
```

### 3. 移動するゾーンが `running` 状態の場合は、ゾーンを停止します。

ゾーンが停止したら、ゾーンが `installed` 状態になったことを確認します。

```
global$ zoneadm -z zonename halt
global# zoneadm list -cv
```

### 4. ゾーンを共有ストレージに移動するために再構成します。

```
global$ zoneadm -z zonename move -p URI [-p URI] [-x extended-option ...]
```

`-p URI`

ストレージ URI を指定します。指定する追加の URI ごとに個別の `-p` オプションを使用します。ストレージ URI の詳細については、[suri\(7\)](#) のマニュアルページを参照してください。

`-x {force-zpool-import | force-zpool-create=rpool | create-size}`

ゾーンで使用する ZFS ストレージプールを作成するかインポートするかをオプションで指定します。`-x` オプションは、操作ごとに 1 つの引数のみを受け付けます。

`force-zpool-import`

既存の ZFS ストレージプールをインポートします。

`force-zpool-create=rpool`

ユーザー指定の名前 *rpool* で新しい ZFS ストレージプールを作成することを指定します。

`create-size=size`

指定の *size* の新しいストレージ URI オブジェクトを作成することを指定します。値に 1M、1G、および 1T などの形式指定子を含めないかぎり、デフォルトで *size* の値はバイト単位になります。

再構成したゾーンは `installed` 状態のままになります。

## ▼ 共有ストレージ構成からゾーンを移動する方法

共有ストレージを使用するゾーン (ZOSS) を、ローカルファイルシステムを使用する構成に移動するには、この手順を実行します。zoneadm move コマンドは、指定した新しいゾーンパスに対応する新しいターゲット ZFS データセットを作成します。ゾーンが使用しなくなった ZFS ストレージプールをオプションで構成解除するか、削除できます。

### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

### 2. ゾーンが `installed` 状態であることを確認します。

```
global$ pfbash zoneadm list -cv
ID NAME          STATUS    PATH                                BRAND  IP
0 global         running  /                                    solaris shared
- zonename     installed /system/zones/zonename solaris excl
```

### 3. 移動するゾーンが **running** 状態の場合は、ゾーンを停止します。

ゾーンが停止したら、ゾーンが **installed** 状態になったことを確認します。

```
global$ pfbash zoneadm -z zonename halt
global$ zoneadm list -cv
```

### 4. ローカルファイルシステムに移動するためにゾーンを再構成します。

```
global$ zoneadm -z zonename move -x remove-rootzpool [-x extended-option ...]
```

```
-x {remove-rootzpool | force-zpool-destroy=rpool | force-storage-destroy-all}
```

共有ストレージからゾーンを移動したあとに、ゾーンの **rootzpool** リソースを処理するには、拡張オプションを指定します。-x オプションは、操作あたり 1 つの拡張オプションのみを受け付けますが、1 つのコマンドで複数の -x オプションを指定できます。

**remove-rootzpool**

共有ストレージからゾーンを移動するために必要です。ゾーン構成からの **rootzpool** リソースの削除を指定します。

**force-zpool-destroy=rpool**

**rootzpool** リソースで指定された ZFS ストレージプールをエクスポートしてから破棄することを指定します。

**force-storage-destroy-all**

**rootzpool** リソースで指定された ZFS ストレージプール内のストレージオブジェクトを破棄することを指定します。

再構成したゾーンは **installed** 状態のままになります。

## ▼ 別の共有ストレージにゾーンを移動する方法

別の共有ストレージにゾーンを移動するには、この手順を実行します。それ以外の場合、ゾーン構成は同じままになります。

始める前に ゾーンの移動先の共有ストレージがすでに存在し、**zpool** を含むことを確認します。

#### 1. ゾーン管理者になります。

詳細については、[68 ページの「root 以外のユーザーへのゾーンを管理する権利の割り当て」](#)を参照してください。

#### 2. ゾーンが **installed** 状態であることを確認します。

```
global$ pfbash zoneadm list -cv
ID NAME          STATUS    PATH          BRAND  IP
0 global         running  /             solaris shared
```



```
- zonename          installed /system/zones/zonename  solaris  excl
```

3. 移動するゾーンが **running** 状態の場合は、ゾーンを停止します。  
ゾーンが停止したら、ゾーンが **installed** 状態になったことを確認します。

```
global$ zoneadm -z zonename halt
global$ zoneadm list -cv
```

4. ゾーンが使用する共有ストレージの場所を変更します。

```
global$ zoneadm -z zonename move -p URI [-p URI ...] \
-x remove-rootzpool [-x extended-option ...]
```

**-p URI**

ストレージ URI を指定します。指定する追加の URI ごとに個別の **-p** オプションを使用します。ストレージ URI の詳細については、[suri\(7\)](#) のマニュアルページを参照してください。

**-x {remove-rootzpool | force-zpool-destroy=rpool | force-storage-destroy-all}**

共有ストレージからゾーンを移動したあとに、ゾーンの **rootzpool** リソースを処理するには、拡張オプションを指定します。**-x** オプションは、操作あたり 1 つの拡張オプションのみを受け付けますが、1 つのコマンドで複数の **-x** オプションを指定できます。

**remove-rootzpool**

共有ストレージからゾーンを移動するために必要です。ゾーン構成からの **rootzpool** リソースの削除を指定します。

**force-zpool-destroy=rpool**

**rootzpool** リソースで指定された ZFS ストレージプールをエクスポートすることを指定します。

**force-storage-destroy-all**

**rootzpool** リソースで指定された ZFS ストレージプール内のストレージオブジェクトを破棄することを指定します。

## ▼ インストール済みゾーンに ZFS ストレージプールを追加または移動する方法

次のいずれかのタスクを実行するには、この手順を実行します。

- 既存の ZFS ストレージプールをインストール済みゾーンに追加します。
- インストール済みゾーンから別のインストール済みゾーンに ZFS ストレージプールを移行します。

始める前に 追加または移動する ZFS ストレージプールが作成されていることを確認します。

ホストシステムで、ストレージ URI を使用できることを確認します。詳細については、[224 ページの「ストレージ URI と共有ストレージリソースの管理」](#) および [suriadm\(8\)](#) のマニュアルページを参照してください。

**1. root 役割になります。**

詳細は、『[Oracle Solaris 11.4 でのユーザーとプロセスのセキュリティー保護](#)』の「[割り当てられている管理権利の使用](#)」を参照してください。

**2. ゾーンをシャットダウンします。**

```
global$ pfbash zoneadm -z zonename shutdown
```

**3. ゾーンを切り離します。**

```
global$ zoneadm -z zonename detach
```

ゾーンは configured 状態に移ります。

**4. 新しい zpool リソースおよび対応する共有ストレージリソースをゾーン構成に追加します。**

```
global$ zonecfg -z zonename
zonecfg:zonename> add zpool
zonecfg:zonename:zpool> add storage URI-string
zonecfg:zonename:zpool> set name=name-string
zonecfg:zonename:zpool> end
```

**5. ゾーンを再接続します。**

```
global$ zoneadm -z zonename attach
```

ゾーンフレームワークは、共有ストレージリソースを構成し、既存の ZFS ストレージプールをインポートして、インストール済みゾーンに割り当てます。ゾーンは installed 状態に移ります。

**6. ゾーンをブートします。**

```
global$ zoneadm -z zonename boot
```

ゾーンは running 状態に移ります。

## 共有ストレージリソースについてのゾーンドキュメントリファレンス

共有ストレージ上のゾーンの詳細は、次の資料を参照してください。

- [beadm\(8\)](#) のマニュアルページ

- [fcinfo\(8\)](#) のマニュアルページ
- [iscsiadm\(8\)](#) のマニュアルページ
- [iser\(4D\)](#) のマニュアルページ
- [itadm\(8\)](#) のマニュアルページ
- [sasinfo\(8\)](#) のマニュアルページ
- [stmfadm\(8\)](#) のマニュアルページ
- [suri\(7\)](#) のマニュアルページ
- [suriadm\(8\)](#) のマニュアルページ
- [solaris\(7\)](#) のマニュアルページ
- [zones\(7\)](#) のマニュアルページ
- [zoneadm\(8\)](#) のマニュアルページ
- [zonecfg\(8\)](#) のマニュアルページ
- [zpool\(8\)](#) のマニュアルページ
- 共有リソースへのゾーンの構成に関する追加情報について、Oracle Solaris 情報ライブラリの『[Oracle Solaris ゾーン構成リソース](#)』



## ゾーンの用語集

---

<b>アカウント の拡張</b>	Oracle Solaris オペレーティングシステムで、タスクまたはプロセスに基づくリソース消費量を柔軟に記録できる方法。
<b>局所スコープ</b>	制御値を超えようとしているプロセスに対して行われる局所アクション。
<b>公平配分 スケジューラ</b>	公平さを基準に CPU 時間を割り当てるスケジューリングクラス (FSS と呼ぶ)。配分は、システムの CPU リソースのうちプロジェクトに割り当てる部分を定義します。
<b>作業セット サイズ</b>	作業セットのサイズ。作業セットとは、プロジェクトの作業負荷がその処理サイクル中にアクティブに使用するページのことです。
<b>作業負荷</b>	アプリケーションまたはアプリケーショングループのプロセスすべての合計。
<b>上限制御</b>	システムリソース使用率を規制するプロセス。
<b>常駐セット サイズ</b>	常駐セットのサイズ。常駐セットとは、物理メモリーに常駐するページのものです。
<b>スカナ</b>	あまり使用されていないページを識別するカーネルスレッド。メモリーが少ない状況では、スカナは最近使用されていないページを再生します。
<b>静的プール 構成</b>	管理者が、リソースプール機能に関してシステムを構成する方法を表現したもの。
<b>ゾーン 管理者</b>	ゾーン管理者の権利は、非大域ゾーンに対してのみ有効です。 <a href="#">大域管理者</a> も参照してください。
<b>ゾーン 状態</b>	非大域ゾーンのステータス。ゾーンの状態は、構成済み、不完全、インストール済み、準備完了、使用不可、実行中、またはシャットダウンのいずれかになります。
<b>大域 管理者</b>	root ユーザーまたは root 役割を持つ管理者。大域ゾーンにログインすると、大域管理者または適切な承認が付与されたユーザーは、システム全体をモニターしたり制御したりできます。 <a href="#">ゾーン管理者</a> も参照してください。

<b>大域スコープ</b>	システム上のすべてのリソース制御のリソース制御値に適用されるアクション。
<b>大域ゾーン</b>	すべての Oracle Solaris システムに含まれるゾーン。非大域ゾーンを使用しているときには、大域ゾーンはシステムのデフォルトゾーンであると同時に、システム規模の管理制御に使用されるゾーンでもあります。  <a href="#">非大域ゾーン</a> も参照してください。
<b>タスク</b>	リソース管理において、長時間にわたる作業の集合を表すプロセスの集まり。各タスクは1つのプロジェクトに関連付けられます。
<b>データリンク</b>	OSI プロトコルスタックのレイヤー 2 インタフェース。システム内で STREAMS DLPI (v2) インタフェースとして表されます。このインタフェースは、TCP/IP などのプロトコルスタックで接続できます。Oracle Solaris 10 ゾーンでは、データリンクは物理インタフェース、アグリゲーション、または VLAN でタグ付けされたインタフェースです。データリンクは、物理インタフェースとして参照されることもあります (たとえば、NIC または VNIC を指す場合)。
<b>デフォルトプール</b>	プールが有効に設定される際にシステムにより作成されるプール。  <a href="#">リソースプール</a> も参照してください。
<b>デフォルトプロセッサセット</b>	プールが有効に設定される際にシステムにより作成されるプロセッサセット。  <a href="#">プロセッサセット</a> も参照してください。
<b>動的構成</b>	ある時点における、指定されたシステムのリソースプールフレームワーク内部のリソース配置に関する情報。
<b>動的再構成</b>	SPARC ベースのシステムで、システムの稼働中にハードウェアを再構成する機能。DR とも呼ばれます。
<b>ネイティブゾーン</b>	solaris ブランドの非大域ゾーン。labeled や solaris10 などのその他のブランド、または solaris-kz ブランドのゾーンはネイティブゾーンではありません。  実行中のネイティブゾーン内のオペレーティングシステムレベルは常に、大域ゾーンのレベルと一致しています。同じカーネルが使用されるため、システムコール変換は実行されません。  ネイティブゾーンは、Oracle Solaris 11 では solaris ブランドが付けられ、Oracle Solaris 10 では native ブランドが付けられています。
<b>ネームサービスデータベース</b>	LDAP コンテナとローカルファイルの両方のためのネームサービス情報データベース。
<b>非大域ゾーン</b>	Oracle Solaris オペレーティングシステムの単一インスタンス内に作成された仮想オペレーティングシステム環境。Oracle Solaris ゾーンソフトウェア区分技術を使用して、

オペレーティングシステムサービスが仮想化されます。非大域ゾーンと大域ゾーンは同じカーネルを共有しているため、非大域ゾーンのパッケージと大域ゾーンのパッケージは同じものです。

<b>非大域ゾーン管理者</b>	ゾーン管理者を参照してください。
<b>プール</b>	リソースプールを参照してください。
<b>プールデーモン</b>	リソースの動的割り当てが必要な場合にアクティブになる <code>poold</code> システムデーモン。
<b>不変ゾーン</b>	読み取り専用ルートを持つように構成されたゾーン。
<b>ブランド</b>	BrandZ 機能のインスタンス。アプリケーションの実行に使用される、ネイティブでないオペレーティング環境が含まれている非大域ゾーンを提供します。
<b>ブランドゾーン</b>	ネイティブでないアプリケーションを非大域ゾーンで実行するための隔離された環境。
<b>プロジェクト</b>	ネットワーク全体の関連作業に対する管理識別子。
<b>プロセッサセット</b>	互いに素である CPU のグループ。各プロセッサセットには、0 以上のプロセッサを含めることができます。プロセッサセットは、リソースプール構成内でリソース要素として表されます。 <code>pset</code> リソースプール要素を使用して構成されます。
<b>メモリー上限実行しきい値</b>	リソース上限デーモンが上限を制限するときのシステム上における物理メモリーの使用効率 (パーセンテージ)。
<b>リソース</b>	アプリケーションの動作を変更する目的で操作されるコンピューティングシステムの側面。
<b>リソース管理</b>	アプリケーションが利用可能なシステムリソースをどのように使用するかを制御する機能。
<b>リソース上限デーモン</b>	リソース上限が定義されたプロジェクト内で動作するプロセスが消費する物理メモリーを規制するデーモン。
<b>リソース消費者</b>	基本的には Oracle Solaris のプロセス。プロジェクトやタスクなどのプロセスモデルエンティティにより、集計済みのリソース消費に関して考察できます。
<b>リソース制御</b>	プロセスごと、タスクごと、またはプロジェクトごとのリソース消費量に対する制限。
<b>リソースセット</b>	プロセスをバインド可能なリソース。たいていの場合、カーネルサブシステムにより構築され、ある種の区分化を提供するオブジェクトを指して使用されます。リソースセットの例には、スケジューリングクラスやプロセッサセットが含まれます。

<b>リソースパーティション</b>	リソースの排他的な一部。リソースのパーティションすべての合計は、実行中の単一の Oracle Solaris インスタンスで利用可能なリソースの総量を表します。
<b>リソースプール</b>	マシンのリソース区分化に使用する構成メカニズム。リソースプールは、区分化可能なリソースグループ間の関係を表します。
<b>ロックされたメモリー</b>	ページング不可能なメモリー。
<b>cap</b>	システムリソース使用率に対する規制。
<b>FSS</b>	<a href="#">公平配分スケジューラ</a> を参照してください。
<b>Oracle Solaris ゾーン</b>	オペレーティングシステムサービスを仮想化し、アプリケーションを実行するための分離およびセキュリティー保護された環境を提供するソフトウェア区分技術。
<b>RSS</b>	<a href="#">常駐セットサイズ</a> を参照してください。
<b>ZBE</b>	ゾーンブート環境。



# 索引

---

## あ

- アプリケーション
  - ゾーンおよび, 95
  - ゾーンでのバックアップ, 110
- アンインストール
  - ゾーン, 151
- 移行
  - root 以外のユーザーによる, 171
  - 共有ストレージ上のゾーン, 244
  - ゾーン, 171
- 移行コマンド, 98
- 一時的なライブゾーン再構成, 36
  - 概要, 35
- 委任リスタタータ 参照 ゾーンの委任リスタタータ
- インストール
  - インストールログファイル, 138
  - トラブルシューティング, 140
- 永続的なライブゾーン再構成
  - 概要, 35
  - ライブ構成への適用, 36

## か

- カーネルゾーン
  - 不変にする, 205
- 回復用アーカイブ
  - 作成, 176
- 拡張アカウントティング
  - ゾーンおよび, 97
- 仮想 zpool
  - エクスポート, 169
- 管理
  - 共有ストレージリソース, 224
  - ゾーンの委任リスタタータ, 189
  - 不変ゾーン, 209

- 書き込み可能にする, 213
- 共有ストレージ
  - unavailable ゾーン状態および, 230
  - URI, 221, 222
  - zoneadm オプション, 230
  - 管理, 224
  - 自動 ZFS, 228
  - ゾーンの移行, 244
  - ゾーンの移動, 245
  - ゾーンのホスト, 233
  - ゾーンへの割り当て, 226
- 共有ストレージ上のゾーン
  - コマンドリファレンス, 250
- 共有ストレージ上のゾーン (ZOSS), 221
  - unavailable ゾーン状態および, 230
  - zoneadm サブコマンドオプション, 230
  - 移行, 244
  - 共有ストレージに対する移動, 245
  - 共有ストレージの割り当て, 226
  - サポートされる名前変更方法, 228
  - 実装, 233
  - 自動 ZFS ストレージ, 228
  - ドキュメントリファレンス, 250
- 共有 IP ゾーンを排他的 IP ゾーンに変更, 133
- クリアー
  - プロパティ, 128
- クローン
  - ZFS, 146
- 権限
  - ゾーンにない, 218
  - トラブルシューティング, 218
- 権利
  - ゾーンおよび, 185
- 権利プロファイル
  - DTrace ツールキット, 187

- 使用することの利点, 68
- ゾーン管理者向けの概要, 68
- ゾーンの管理および, 68
- ゾーンの管理に使用, 70
- コアファイル
  - ゾーンおよび, 79
- 更新
  - ゾーンを並列で, 46
- 構成
  - 削除, 134
  - ゾーン, 119
  - ゾーンのプロキシ, 44
  - 不変大域ゾーン, 214
  - 変更を元に戻す, 133
- 構成済みゾーンの検証, 138
- 公平配分スケジューラ (FSS)
  - ゾーンおよび, 97, 201
- ゴールデンゾーン
  - 概要, 56
  - クローニング, 150
  - 作成, 148
  - 説明, 148
- ゴールデンゾーンのクローニング, 148, 150
  - 参照 ゾーンのクローニング
  - 説明, 148
- ゴールデンゾーンの作成
  - 説明, 148
- コールド移行
  - ゾーン, 173
  - 定義済み, 98
- コマンド
  - ゾーン, 112
- コマンドリファレンス
  - 共有ストレージ上のゾーン, 250
- 孤立したブート環境, 60
- コンソールログインモード, 64
  
- さ
- サービス管理機能 (SMF)
  - 依存関係, 19
  - インスタンスのサービス状態, 20, 20
  - 概要, 18
  - ゾーンの委任リスタータ, 23
  - 表示, 189
  - プロパティの設定, 190
  - プロパティの説明, 24
  - ブート順序, 22
  - 目標サービス, 21
- 再作成
  - ゾーン, 203
- 削除
  - ゾーン, 153, 153
  - ゾーン構成, 134
  - データセットリソース, 168
- 作成
  - 回復用アーカイブ, 176
  - ゴールデンゾーン, 148
  - ゴールデンゾーンのクローン, 150
  - ゾーン, 119
  - ゾーン構成プロファイル, 118
- 時間値
  - ゾーンおよび, 80
- システム構成プロファイル
  - 概要, 28
  - 例, 30
- 状態
  - ゾーンの SMF インスタンスのサービス, 20
- 初期ゾーンログインおよび構成, 156
- シングルユーザーモード
  - ゾーンのブート, 145
- スナップショット
  - ZFS, 146
- セキュリティポリシー
  - MWAC, 74, 205
- 設定
  - 大域ゾーンの CPU シェア, 131
- 設定解除
  - プロパティを値なしに, 128
- ゾーン
  - admin リソースおよび, 68, 186
  - admin リソースを使用して管理, 68, 186
  - beadm のサポート, 58
  - CD へのアクセス, 193
  - DTrace の実行, 79
  - DVD へのアクセス, 193
  - file-mac-profile 値の設定, 207
  - fsstat アクティビティ, 198
  - fstype の統計情報, 198
  - IPMP, 199

- IPsec, 75
- MWAC セキュリティポリシーの設定, 207
- MWAC ポリシーのリソースの例外, 207
- NFS サーバー, 81
- Oracle Solaris 監査, 75
- ready 状態, 144
- SMF サービスのステータス, 194
- solaris、更新, 41
- solaris、パッケージ, 41
- zonecfg コマンド, 34
- zonep2vchk ツール, 105
- zonestat コマンド, 197
- zonestat による統計情報, 195
- アクティブなモニタリング, 78
- アプリケーション, 95
- アプリケーションのバックアップ, 110
- 移行, 104, 171
- 一般的な管理, 185
- 可観測性, 77
- 拡張アカウントティング, 97
- 共有 BE, 59
- 共有 IP を排他的 IP に変更, 133
- 共有ストレージ URI, 222
- 権利および, 185
- 権利プロファイル, 68, 70
- 権利プロファイルを使用して管理, 70
- コアファイル, 79
- 公平配分スケジューラ (FSS), 97, 201
- 個々に再作成, 203
- コマンド, 112
- 孤立したブート環境, 60
- サイズの制限, 18
- 時間値, 80
- 使用されるコマンド, 112
- 使用不可能なシステムからの移行, 175
- 使用例, 185
- セキュリティポリシーの設定, 205
- 大域ゾーンでプロセスが表示可能, 76
- 大域不変の構成, 214
- 大域不変の保守, 214
- ディスク容量, 17
- データベースのバックアップ, 111
- データリンクの管理, 200
- デバイスの使用, 93
- 特権, 71
- トラブルシューティング, 217
- 内容のバックアップ, 110
- 名前の表示, 159
- 名前の変更
  - zoneadm の使用, 129
  - zonecfg の使用, 130
- ネットワークアドレス, 25
- ネットワーク、共有 IP, 90
- ネットワーク帯域幅の統計情報, 197
- ネットワーク、排他的 IP, 88
- ノード名, 80
- バックアップ, 109
- パッケージの削除, 44
- パッケージの追加, 42
- パッケージング, 40
- ファイルシステム, 81
- ファイルシステムのマウント, 190
- ファイルのバックアップ, 202
- ブート環境, 57
- ブート引数, 145
- ブート不可能な環境に対する beadm アクション, 58
- 復元, 112
- 不変ゾーン, 74
- 不変大域, 214
- 不変の管理, 209
- 不変への設定, 205
- プロキシ構成, 44
- プロセス ID の可視性, 77
- 並列で更新, 46
- モニタリング, 79
- ライブ再構成, 34
- リソース制御, 96
- リムーバブルメディアへのアクセス, 193
- ログイン手順, 155
- ゾーンアプリケーション環境
  - 説明, 52
- ゾーン移行権利プロファイル, 171
- ゾーン構成
  - 表示, 125
- ゾーン構成プロファイル
  - 作成, 118
- ゾーンコンソールログイン, 157
- ゾーンサイズの制限, 18
- ゾーンシステム構成, 156

- ゾーン状態
    - マーク付け, 142
  - ゾーン状態のマーク付け, 142
  - ゾーン内での DTrace の実行, 79, 186
  - ゾーン内の特権, 71
  - ゾーンのアンインストール
    - 概要, 55
  - ゾーンの移行, 104
    - 共有ストレージを使用しない, 103
  - ゾーンの一覧表示, 138
  - ゾーンの移動
    - 共有ストレージから, 247
    - 共有ストレージ上, 245
    - 共有ストレージに対する, 246
    - 同一システム上, 150
    - 別の共有ストレージに対する, 248
  - ゾーンの委任リスタータ
    - displaying, 189
    - 概要, 23
    - 管理, 189
    - 動作のしくみ, 23
    - プロパティ
      - 説明, 24
    - プロパティの設定, 190
  - ゾーンのインストール, 138
    - 概要, 39, 48
    - タスク, 137, 137
  - ゾーンの管理
    - 概要, 39
  - ゾーンのクローニング, 146, 146
    - 参照 ゴールデンゾーンのクローニング
      - zoneadm clone コマンド, 57
    - 概要, 56
    - ゴールデンゾーン, 56
    - 再構成要件, 56
    - 説明, 57
    - 統合アーカイブ, 57
  - ゾーンの構築
    - 概要, 50
  - ゾーンのシステム構成, 27
  - ゾーンのシャットダウン, 53, 187, 187
    - 参照 ゾーンの停止
    - 概要, 53
  - ゾーンの終了, 159
  - ゾーンの停止, 53, 188, 188
    - 参照 ゾーンシャットダウン
      - 概要, 53
    - ゾーンの名前の変更
      - zoneadm の使用, 129
      - zonecfg の使用, 130
    - ゾーンのブート, 144, 144
      - 参照 ゾーンのリブート
      - シングルユーザーモード, 145
      - タスク, 137
    - ゾーンの並列更新, 46
    - ゾーンのホスト名, 24
    - ゾーンのリブート, 188
      - 参照 ゾーンのリブート
      - 概要, 54
    - ゾーンの利用状況
      - 報告, 197
    - ゾーンへのデータの取り込み
      - 概要, 50
    - ゾーン名
      - 表示, 159
    - ゾーンログイン
      - zlogin コマンド, 65
      - 概要, 64
      - 対話型モード, 66, 157
      - 非対話型モード, 66
      - フェイルセーフモード, 65
      - リモート, 65
- ## た
- 大域ゾーン
    - 不変にする, 205
  - 大域ゾーンのプロキシ, 45
  - 大域ゾーンのプロキシのオーバーライド, 45
  - 対話型モードログイン, 157
  - 対話式システム構成 (SCI) ツール
    - 概要, 29
    - 画面が表示されないトラブルシューティング, 156
  - 追加
    - 専用のデバイス, 131
  - 停止
    - トラブルシューティング, 218
  - 停止および再起動
    - ゾーン, 188

- データセット
  - 実行中のゾーンへの追加, 165
  - ゾーンからの削除, 168
  - ライブ再構成, 165
- データリンク
  - ゾーンおよび, 200
- テスト実行
  - ライブゾーン再構成, 36
- デバイス
  - ゾーンへの追加, 131
- デバイスの使用
  - ゾーンおよび, 93
- 統合アーカイブ
  - 概要, 48
  - ゾーンのクローニング
    - 概要, 57
- 特権セット
  - ゾーンに対する変更, 132
- トラステッドパストメイン
  - 不変ゾーンアクセス, 210
  - 不変ゾーンおよび SMF サービス, 207
- トラブルシューティング
  - SCI ツール画面が表示されない, 156
  - インストール失敗, 140
  - ゾーン, 217
- 取り消し 参照 元に戻る
- な**
- 名前変更
  - 共有ストレージ上のゾーン, 228
- ネットワーク
  - ゾーン内の排他的 IP, 88
- ネットワークアドレス
  - 排他的 IP ゾーン, 25
- ネットワーク、共有 IP, 90
- ネットワーク帯域幅
  - ゾーンの統計情報, 197
  - 排他的 IP ゾーン, 197
- ノード名
  - ゾーンおよび, 80
- は**
- 排他的 IP ゾーン
  - 帯域幅の利用状況の報告, 197
  - ネットワークアドレス, 25
  - ネットワーク帯域幅, 197
- バックアップ
  - ゾーン, 109, 202
  - ゾーン内のデータベース, 111
  - ゾーンの内容, 110
- パッケージ
  - インストールのトラブルシューティング, 217
- パッケージのインストール
  - トラブルシューティング, 217
- 汎用一意識別子 (UUID)
  - 表示, 141
- 非大域ゾーン
  - beadm コマンドと, 58
  - 不変にする, 205
- 非対話型モードログイン, 158
- 必須書き込みアクセス制御 参照 MWAC 表示
  - 現在のゾーンの名前, 159
  - すべてのリソースプロパティ, 126
  - ゾーン構成情報, 125
  - ゾーンの委任リスタータ, 189
  - 汎用一意識別子 (UUID), 141
- ファイルシステム
  - LOFS を使用してマウント, 191
  - ZFS データセットの委任, 191
  - 実行中のゾーンでのマウント, 190
  - ゾーンおよび, 81
- ファイルシステム構成, 26
- ブートオプション
  - 概要, 54
- ブート環境
  - 共有ゾーンの状態の影響, 59
  - ゾーンと, 57
- ブート順序, 22
- ブート引数とゾーン, 145
- ブート不可能なブート環境
  - ゾーン, 58
- フェイルセーフモードログイン, 158
- 復元
  - ゾーン, 112
- 不変ゾーン
  - add dataset リソースポリシー, 207
  - add fs リソースポリシー, 207

- dynamic-zones, 206
  - file-mac-profile プロパティ, 205
  - fixed-configuration MWAC ポリシー, 206
  - flexible-configuration MWAC ポリシー, 206
  - none MWAC ポリシー, 205
  - SMF コマンドおよび, 207
  - strict MWAC ポリシー, 206
  - zlogin コマンドによるアクセス, 213
  - zlogin コマンドの承認, 213
  - カーネルゾーン, 205
  - 書き込み可能でブート, 213
  - 管理, 209, 210
  - 構成, 205
  - 情報, 74
  - セキュリティポリシーの設定, 205
  - 大域, 214
  - 大域ゾーン, 205
  - トラステッドパストメイン, 210
  - 非大域ゾーン, 205
  - 不変ゾーンのブート, 213
  - プロキシ構成
    - ゾーンと, 44
  - プロキシの構成, 45
  - プロセス
    - ゾーン内での可視性, 77
    - 大域ゾーンで表示可能, 76
  - プロセッサセット
    - 情報の報告, 196
  - プロパティ
    - 値なしに設定解除, 128
    - クリアー, 128
    - ゾーンの委任リスタータ
      - 設定, 190
      - 説明, 24
    - デフォルトにリセット, 128
  - プロパティの設定
    - ゾーンの委任リスタータ, 190
  - プロファイル
    - ゾーンのシステム構成
      - 概要, 28
      - 例, 30
  - 変換
    - zonep2vchk の使用, 105
  - 変更
    - ゾーンの特権セット, 132
    - リソース, 127
  - 保守
    - 不変大域ゾーン, 214
  - ホスト名
    - ゾーン, 24
- ま**
- マウント
    - 実行中のゾーンでのファイルシステム, 190
  - メディア
    - ゾーンでのアクセス, 193
  - メモリー
    - 利用状況の報告, 196
  - 目標サービス
    - サービス管理機能 (SMF), 21
  - 元に戻す
    - ゾーン構成の変更, 133
  - モニタリング
    - アクティブなゾーンの統計情報, 78
    - ゾーンの統計情報, 79
- や**
- 役割
    - 管理者ゾーンへの権利の割り当て, 185
  - 要件
    - クローン作成時の再構成, 56
  - 読み取り専用ゾーンのルート 参照 不変ゾーン
- ら**
- ライブ移行
    - root 以外のユーザーによる, 171
  - ライブ再構成
    - データセット, 165
  - ライブゾーン構成の再ロード, 37
  - ライブゾーン再構成
    - 一時的な変更, 36
    - 概要, 35
    - 構成の再ロード, 37
    - データセットの追加, 165

- 手順, 161
  - テスト実行, 36
  - 非大域ゾーンとカーネルゾーン, 34
  - 予行演習, 36
  - リセット
    - プロパティをデフォルトに, 128
  - リソース
    - MWAC ポリシーの例外, 207
    - 変更, 127
    - 利用状況の報告, 195
  - リソース制御
    - ゾーンおよび, 96
  - リモートゾーンログイン, 65
  - 例
    - ゾーン構成プロファイル
      - DHCP を使用し DNS を使用しない排他的 IP ゾーン, 33
      - LDAP も DNS も使用しない排他的 IP ゾーン, 32
      - 自動構成を使用する排他的 IP ゾーン, 31
    - 大域ゾーンを不変にする, 207
  - ログイン, 155
    - ゾーンコンソール, 157
    - ゾーンシステム構成, 156
    - 非対話型モード, 158
    - フェイルセーフモード, 158
    - リモートゾーン, 65
  - ログインの概要, 63
  - ログファイル
    - ゾーンのインストール, 138
- わ**
- 割り当て
    - 権利, 185
- A**
- add dataset リソースポリシー
    - MWAC セキュリティポリシーの例外, 207
  - add fs リソースポリシー
    - MWAC セキュリティポリシーの例外, 207
  - admin リソース
    - ゾーンの管理のために使用, 68, 186
  - archiveadm コマンド
    - 回復用アーカイブの作成, 176
  - autoboot グローバルプロパティ
    - 概要, 54
- B**
- beadm コマンド
    - ゾーン
      - 非大域ゾーンでの, 58
- C**
- CD
    - ゾーンでのアクセス, 193
  - cpu-shares グローバルプロパティ
    - 大域ゾーンでの設定, 131
  - CPU
    - 利用状況の報告, 196
  - CPU シェア
    - 大域ゾーンでの設定, 131
  - create サブコマンド
    - archiveadm コマンドの例, 176
- D**
- /dev/console ゾーンコンソール
    - 概要, 64
  - dladm reset-linkprop エラーメッセージ
    - トラブルシューティング, 218
  - dtrace\_proc, 79, 186
  - dtrace\_user, 79, 186
  - DTrace ツールキットの権利プロファイル, 187
  - DVD
    - ゾーンでのアクセス, 193
  - dynamic-zones
    - ゾーン MWAC ポリシー, 206
- F**
- file-mac-profile
    - MWAC セキュリティポリシーの設定, 207

**fixed-configuration**  
ゾーン MWAC ポリシー, 206

**flexible-configuration**  
ゾーン MWAC ポリシー, 206

**force-storage-destroy-all**, 232

**force-zpool-create-all** オプション, 50

**force-zpool-create=** オプション, 50

**force-zpool-destroy-all**, 232

**force-zpool-import** オプション, 50

**fsstat** による統計情報, 198

**fsstat** ユーティリティー, 79, 198

**fstype** の統計情報  
ゾーンおよび, 198

**I**

**info** サブコマンド  
archiveadm コマンドの例, 176

**IPMP**  
ゾーンおよび, 199

**IPsec**  
ゾーン内での使用, 75

**L**

**limitpriv**  
トラブルシューティング, 218

**LOFS**  
ファイルシステムのマウント, 191

**M**

**MWAC**  
セキュリティポリシーの例外, 207  
ゾーンのセキュリティポリシー, 205  
定義, 74, 205

**N**

**NFS** サーバー, 81

**none**  
ゾーン MWAC ポリシー, 205

**O**

**Oracle Solaris** 監査  
ゾーン内での使用, 75

**P**

**pset** リソースプール要素  
情報の報告, 196

**R**

**ready** 状態  
ゾーンの移行, 144

**S**

**SCI** ツール 参照 対話式システム構成 (SCI) ツール

**SMF** 参照 サービス管理機能 (SMF)  
不変ゾーンおよび, 207

**SMF** サービス  
ゾーンでのステータス, 194

**solaris.zone.login** 承認  
対話型モード, 66

**solaris.zone.manage** 承認  
非対話型モード, 66

**solaris** ゾーン  
手動同期, 40

**solaris** ゾーン移行, 173

**solaris** ゾーンの移行, 173

**strict**  
ゾーン MWAC ポリシー, 206

**svc:/system/zones:default** サービス  
依存関係, 19

**T**

**TPD** 参照 トラストドパドメイン

**U**

**unavailable** ゾーン状態  
共有ストレージおよび, 230



Universal Resource Identifier (URI) 参照 URI  
URI

共有ストレージ, 222

## Z

### ZFS

クローン, 146

スナップショット, 146

### ZFS ストレージプール

ゾーンへの追加, 249

別のゾーンへの移動, 249

ZFS ストレージプールの移動, 249

ZFS ストレージプールの追加, 249

### ZFS データセット

ゾーンへの委任, 191

### zlogin コマンド

概要, 63

ゾーンコンソール, 157

対話型モード, 157

非対話型モード, 158

フェイルセーフモード, 158

zoneadm boot コマンド, 144, 145

ブートオプション, 54

zoneadm clone コマンド

概要, 57

zoneadm halt コマンド

概要, 53

zoneadm install コマンド, 138

概要, 48

zoneadm mark コマンド, 142

zoneadm migrate コマンド, 99

zoneadm ready コマンド, 144

zoneadm reboot コマンド

概要, 54

ブートオプション, 54

zoneadm shutdown コマンド, 187

概要, 53

zoneadm uninstall コマンド, 55, 151

zoneadm verify コマンド, 138

zoneadm コマンド

概要, 39

共有ストレージオプション, 230

ゾーンの名前の変更, 129

zoneadmd デーモン, 52

zonecfg delete コマンド

アンインストールされたゾーン, 134

インストール済みのゾーン, 153

zonecfg info コマンド

構成情報の表示, 125

すべてのリソースプロパティの表示, 126

zonecfg revert コマンド, 133

zonecfg コマンド, 34

ゾーンの名前の変更, 130

zonecfg ユーティリティ

ゾーンの構成, 119

デフォルトモード, 34

ライブモード, 34

zonename コマンド

現在のゾーンの名前の表示, 159

zonep2vchk

変換ツール, 105

zonepath グローバルプロパティ

ZFS に自動的に作成, 138

zonestat コマンド

ネットワーク帯域幅, 197

zonestat ユーティリティ, 78, 195

ZOSS 参照 共有ストレージ上のゾーン (ZOSS)

zsched プロセス, 52

