

# ZENworks 2020

## コマンドラインユーティリ ティリファレンス

2019 年 10 月

## 保証と著作権

保証と著作権、商標、免責事項、保証、輸出およびその他の使用制限、米国政府の規制による権利、特許ポリシー、および FIPS コンプライアンスの詳細については、<https://www.microfocus.com/about/legal/> を参照してください。

---

# 目次

このガイドについて	5
<b>1 ZENworks コマンドラインユーティリティ</b>	<b>7</b>
zman.....	8
zeninfocollect .....	147
zac for Windows .....	149
zac for Linux.....	165
Macintosh 用の zac .....	195
<b>2 ZENworks 診断センター</b>	<b>207</b>
2.1 ZDC のダウンロードと抽出 .....	207
2.2 ZDC の実行 .....	208
<b>3 ZENworks コマンドラインユーティリティのトラブルシューティング</b>	<b>211</b>
<b>4 Zman で作業するためのガイドライン</b>	<b>217</b>



# このガイドについて

この『ZENworks コマンドラインユーティリティリファレンス』には、zman、zac、および zeninfocollect といった Micro Focus ZENworks ユーティリティを使用するための情報が含まれます。

このガイドの情報は、次のように構成されます。

- ◆ 7 ページの第 1 章「ZENworks コマンドラインユーティリティ」
- ◆ 207 ページの第 2 章「ZENworks 診断センター」
- ◆ 211 ページの第 3 章「ZENworks コマンドラインユーティリティのトラブルシューティング」
- ◆ 217 ページの第 4 章「Zman で作業するためのガイドライン」

## 対象読者

このガイドは、ZENworks 管理者を対象としています。

## フィードバック

本マニュアルおよびこの製品に含まれているその他のマニュアルについて、皆様のご意見やご要望をお寄せください。オンラインヘルプの各ページの下部にあるユーザコメント機能を使用してください。

## その他のマニュアル

ZENworks Configuration Management には、製品について学習したり、製品を実装したりするために使用できるその他のマニュアル (PDF 形式および HTML 形式の両方) も用意されています。その他のマニュアルについては、[ZENworks マニュアル Web サイト](#)を参照してください。



# 1 ZENworks コマンドラインユーティリティ

次のセクションでは、Micro Focus ZENworks コマンドラインユーティリティについて説明します。

- ◆ 8 ページの「zman(1)」
- ◆ 147 ページの「zeninfocollect(1)」
- ◆ 149 ページの「zac for Windows(1)」
- ◆ 165 ページの「zac for Linux(1)」
- ◆ 195 ページの「Macintosh 用の zac(1)」

# zman(1)

## 名前

zman - Asset Management、Configuration Management、Endpoint Security Management、および Full Disk Encryption といった Micro Focus ZENworks 製品を管理するためのコマンドラインインタフェース。

## 説明

zman コマンドラインインタフェースからは、ZENworks のほとんどの管理操作にすばやくアクセスできます。zman コマンドをスクリプトで使用することで、多数の ZENworks 管理タスクを自動化できます。

以降のセクションでは、ZENworks 製品用に提供されるすべてのコマンドについての情報を提供します。コマンドは、次の例外を除き、すべての製品に適用されます。

- ◆ **ZENworks Asset Management のコマンド** : ZENworks Asset Management にのみ適用されます。
- ◆ **ZENworks バンドルのコマンド** : ZENworks Configuration Management にのみ適用されます。
- ◆ **ZENworks イメージングのコマンド** : ZENworks Configuration Management にのみ適用されます。
- ◆ **ZENworks Endpoint Security Management のコマンド** : ZENworks Endpoint Security Management にのみ適用されます。
- ◆ **ZENworks Full Disk Encryption のコマンド** : ZENworks Full Disk Encryption にのみ適用されます。

## 構文

```
zman <カテゴリ>< コマンド>[- オプション][ 引数]
```

通常は、zman コマンドには、短い形式と長い形式があります。長い形式は、カテゴリ - アクションの形式で構成されます。たとえば、コマンド registration-list は、登録フォルダ内の登録すべてを一覧表示します。多くのコマンドがありますが、コマンドはカテゴリ (ワークステーション、サーバ、バンドル、ポリシー、管理者、登録など) ごとにまとめられており、これらに対して実行するアクション (リスト、作成、変更、削除) のほとんどは一般的なものであるため、ほとんどのコマンドは覚えやすいものばかりです。

各コマンドの短縮形では、長い形式の各単語の 1 文字を使用します。つまり、admin-list の短縮形は、al になります。このルールには、短い名前の固有性を保持するために、いくつか例外があります。たとえば、bc は bundle-create の短い形式で、bcp は、bundle-copy の短い形式です。

コマンドは、オプションと引数を取ります。オプションは、コマンドのどこでも指定できます。引数は位置に依存します。



## オプション

オプションには、スイッチオプションとフラグ付きオプションの2つのタイプがあります。どちらのオプションにも、短い形式と長い形式があります。短い形式の前には、1つのハイフンがあり、長い形式の前には、2つのハイフンがあります。たとえば、`-n`と`--namefilter`はそれぞれフラグ付きオプションの短い形式と長い形式で、`bundle-list` コマンドの結果の表示を名前でもフィルタする場合に使用します。

**スイッチ**: 値を取らないパラメータです。存在するだけで意味を持ちます。たとえば、`bundle-list` で `-recursive` スイッチを指定すると、サブフォルダの内容も表示するように指示したことになります。

**フラグされたオプション**: このパラメータは、オプションの値を設定するために使用されます。値は、オプションの短いフラグまたは長いフラグの後にスペースまたは等号記号を付けて指定します。例: `zman bundle-list -namefilter "bundle*" -r`

## 引数

コマンドの引数は通常、コマンド自体と同じ順序で並べます。たとえば、`server-add-bundle` の場合、サーバを最初に、バンドルを2番目に指定します。

## 使用方法

このセクションでは、一般的なコマンド形式と規則について説明しています。

### 管理者

`zman` のすべてのアクションは、管理者のアクセス制限によって管理されています。最初のインストール中に作成された最初の管理者アカウントは、すべてのオブジェクトに対する権限を持ちます。作成する追加の管理者アカウントには、デフォルトで読み取り専用権限が与えられています。これらのアカウントには、管理するオブジェクトに対する権限が明示的に与えられている必要があります。

## ヘルプ

zman コマンドラインインタフェースには複数レベルのヘルプが用意されています。スクロール可能な出力の場合は、オペレーティングシステムネイティブの more コマンドを使用してヘルプをパイプラインします。

- ◆ 一番上のレベルでは、zman で使用可能なカテゴリのコマンドすべてとコマンドのグローバルオプションが一覧表示されます。

例 : zman

- ◆ すべてのカテゴリレベルで、zman で使用可能なコマンドすべてがカテゴリ (ワークステーション、サーバ、バンドル、ポリシーなど) 別に整理されて一覧表示されます。

例 : zman -h

- ◆ 各カテゴリでは、そのカテゴリだけで使用可能なコマンドが一覧表示されます。

例 : zman policy /?

- ◆ 各コマンドレベルでは、特定のコマンドの使用法、詳しいヘルプ、および例が提供されます。

例 : zman policy-list --help

- ◆ -h、および --help は、使用可能なさまざまなヘルプオプションを表します。

## フォルダ

フォルダ引数を取るコマンドに対してフォルダが指定されていない場合、コマンドはルートフォルダをターゲットにします。フォルダパスを指定するには、スラッシュ (/) で区別されるルートからの各フォルダを一覧表示します。たとえば、ルートに folder1 という名前のフォルダがあり、そのフォルダの中に subfolder1 という名前のサブフォルダがある場合、このフォルダを folder1/subfolder1 として参照します。指定した各フォルダは、すでに存在する必要があります。

## 必須の引数とオプションの引数

括弧 ( ) で囲まれた引数は必須で、角括弧 [ ] で囲まれた引数はオプションです。

## 省略記号 (...)

省略記号は、コマンドが最後の引数タイプのエントリを複数受け入れることを示します。引数は、コマンドに応じてスペースまたはカンマで区切る必要があります。例を次に示します。

```
zman server-add-bundle [ オプション ] (サーバ) (バンドル) [...]
```

この省略記号は、server-add-bundle が複数のバンドルを受け入れることができることを示しています。コマンドレベルのヘルプでは、複数エントリの引数ヘルプは、より明確に [[ バンドル ] (バンドル) ... (バンドル) ] として指定されます。

最後の引数のエントリを複数持つすべてのコマンドには、-f|--folder オプションもあります。このオプションは、その引数のすべての項目が同じフォルダ内にある場合に便利です。項目のパスは、このフォルダパスを基準にして相対的に計算されます。例を次に示します。

```
zman server-add-bundle サーバ1バンドルフォルダ / バンドル1バンドルフォルダ / バンドル2
```

これは次のようにも指定できます。

```
zman server-add-bundle サーババンドル 1 バンドル 2 -- フォルダ バンドルフォルダ
```

### ZENworks オブジェクトのパス

必要な引数が ZENworks オブジェクトのパスである場合、コマンドヘルプで絶対パスを入力するように指定されていない限り、パスはオブジェクトのルートフォルダを基準とし、スラッシュ (/) で区切った相対パスとして指定する必要があります。例を次に示します。

```
zman server-add-bundle サーバフォルダ / サーババンドルフォルダ / バンドル 1
```

引数の絶対パスは次のようになります。

```
/ デバイス / サーバ / サーバフォルダ / サーバおよび / バンドル / バンドルフォルダ / バンドル 1
```

### ファイルへのオブジェクトのエクスポート

bundle-export-to-file、policy-export-to-file、および settings-export-to-file などの zman コマンドを使用すれば、それぞれバンドル、ポリシー、および設定の定義を、指定したファイルに XML フォーマットでエクスポートすることができます。ファイルは、指定された名前とファイル拡張子で作成されます。ファイル拡張子を指定しなかった場合には、デフォルトで .xml 拡張子が追加されます。これらのファイルを使用して、バンドル、ポリシー、および設定を作成することができます。

## zman サービス

zman をサービスとして実行できるようになりました。zman をサービスとして実行すると zman コマンドがサービスモードで実行され、コマンドの実行時間を短縮できます。

zman サービスコマンドは、シェルカテゴリ内で次のようにグループ化されています。

- ◆ [11 ページの「zman shell-start-as-service \(ssas\) \[オプション\]」](#)
- ◆ [11 ページの「zman shell-stop-service \(sss\)」](#)
- ◆ [11 ページの「zman shell-status \(ssts\)」](#)

### zman shell-start-as-service (ssas) [オプション]

zman サービスを開始します。

次のオプションを受け入れます。

-s または --servicePort: zman サーバが実行するポートを指定します。デフォルトポートは 2020 です。

### zman shell-stop-service (sss)

zman サービスを停止します。

### zman shell-status (ssts)

zman サービスの状態を表示します。

zman コマンドをサービスとして実行するには、次の操作を行う必要があります。

- ◆ zman サービスを開始する前に、zman admin-store-credential(asc) コマンドを実行して資格情報を保存します。zman サービスは、保存された資格情報を使用して、zman コマンドをサービスモードで実行します。
- ◆ サービスモードでファイルパスが必要な zman コマンドを実行する場合は、相対ファイルパスではなく、完全なファイルパスを指定する必要があります。デフォルトでは、作業ディレクトリは、zman サービスが開始される場所に初期化されます。
- ◆ zman をサービスとして実行している場合、サーバでシステム更新が完了した後にサービスを再起動してください。

次の zman コマンドは、zman サービスが起動し、実行している場合でも、別のプロセスとして実行されます。

zman shell-stop-service(sss)

zman admin-create(ac)

zman admin-store-credential(asc)

zman certificate-authority-export(cae)

zman certificate-authority-import(cai)

zman zenserver-backup(zsb)

zman zenserver-restore(zsr)

## コマンドのリスト

- ◆ [14 ページの「アセット管理コマンド」](#)
- ◆ [15 ページの「監査設定コマンド」](#)
- ◆ [16 ページの「管理者コマンド」](#)
- ◆ [19 ページの「バッチコマンド」](#)
- ◆ [20 ページの「バンドルコマンド」](#)
- ◆ [37 ページの「バンドル設定コマンド」](#)
- ◆ [39 ページの「証明書の権限コマンド」](#)
- ◆ [40 ページの「資格情報コマンド」](#)
- ◆ [41 ページの「データベースコマンド」](#)
- ◆ [41 ページの「展開コマンド」](#)
- ◆ [43 ページの「ディスカバリコマンド」](#)
- ◆ [44 ページの「動的グループコマンド」](#)
- ◆ [45 ページの「エンドポイントセキュリティ管理コマンド」](#)
- ◆ [47 ページの「Full Disk Encryption コマンド」](#)
- ◆ [49 ページの「ホットリストコマンド」](#)
- ◆ [50 ページの「Intel AMT コマンド」](#)
- ◆ [52 ページの「イメージングコマンド」](#)

- ◆ 52 ページの「インベントリコマンド」
- ◆ 54 ページの「ライセンスコマンド」
- ◆ 55 ページの「場所コマンド」
- ◆ 58 ページの「メッセージコマンド」
- ◆ 60 ページの「ネットワーク環境コマンド」
- ◆ 62 ページの「Micro Focus Customer Center 登録コマンド」
- ◆ 63 ページの「モバイルデバイスコマンド」
- ◆ 67 ページの「オブジェクトコマンド」
- ◆ 67 ページの「プライマリサーバコマンド」
- ◆ 68 ページの「ポリシーコマンド」
- ◆ 77 ページの「ポリシー設定コマンド」
- ◆ 78 ページの「クエリコマンド」
- ◆ 79 ページの「キューコマンド」
- ◆ 80 ページの「登録コマンド」
- ◆ 83 ページの「役割コマンド」
- ◆ 86 ページの「ルールセットコマンド」
- ◆ 92 ページの「サテライトコマンド」
- ◆ 99 ページの「サーバコマンド」
- ◆ 113 ページの「設定コマンド」
- ◆ 115 ページの「サブスクリプションコマンド」
- ◆ 118 ページの「システム更新 / 製品認識更新コマンド」
- ◆ 122 ページの「ユーザコマンド」
- ◆ 129 ページの「ワークステーションコマンド」
- ◆ 139 ページの「ZENworks サーバコマンド」
- ◆ 140 ページの「グローバルオプション」
- ◆ 141 ページの「セキュリティ」
- ◆ 141 ページの「ファイル」
- ◆ 142 ページの「終了コード」
- ◆ 145 ページの「ZeUS コマンド」

## アセット管理コマンド

これらのコマンドを使用して購入記録ファイルをロードし、コンプライアンスエンジンを更新します。ZENworks Asset Management にのみ適用されます。

- ◆ 14 ページの「`asset-management-import-contract|amic (-f|--ファイル名 = ファイルパス)`」
- ◆ 14 ページの「`asset-management-load-purchase-record (amlpr)`」
- ◆ 14 ページの「`asset-management-refresh-compliance-data (amrcd)`」

### `asset-management-import-contract|amic (-f|--ファイル名 = ファイルパス)`

契約をファイルにインポートします。コマンドによって、ローダが処理するインポートタスクがキューに登録されます。その後、ローダがファイルから契約をロードします。

コマンドは非同期です。つまり、zman はインポートの終了を待たずにすぐに戻ります。これはコマンドの実行に長時間かかることがあるためです。インポートの進行状況は、コマンドによって作成されるログファイルを表示することで確認できます。

次のオプションを受け入れます。

`-f, --filename=[ファイルパス]`: インポートするカンマ区切り (.csv または .txt) ファイルのフルパスを指定します。

### `asset-management-load-purchase-record (amlpr)`

購入記録ファイルをロードします。Purchase Record Loader は購入記録ファイルを検索して、プライマリサーバの ZENWORKS\_HOME/temp/PurchaseRecordImport/uploaded フォルダにロードします。ZENworks コントロールセンターから Purchase Record Loader を使用したことがない場合は、このフォルダを作成する必要があります。

このコマンドは非同期です。つまり、zman はコマンドの終了を待たずにすぐに戻ることを意味しています。これはコマンドの実行に長時間かかるためです。ZENworks コントロールセンターからこのプロセスのステータスを確認できます。

### `asset-management-refresh-compliance-data (amrcd)`

ソフトウェアコンプライアンスエンジンを実行して、ソフトウェアライセンスコンプライアンスデータを更新します。

このコマンドは非同期です。つまり、zman はコマンドの終了を待たずにすぐに戻ることを意味しています。これはコマンドの実行に長時間かかるためです。ZENworks コントロールセンターからこのプロセスのステータスを確認できます。

## 監査設定コマンド

監査環境設定の詳細のインポートとエクスポートに使用するコマンドです。監査コマンドは、長い形式では `audit-` から始まり、短い形式では `a` から始まります。次のコマンドは、「グローバルオプション」のセクションに一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- ◆ 15 ページの「`audit-settings-export-to-file|asetf [-s|--source= ソースデバイスまたはデバイスフォルダのパス] [-a|--auditEventStatus= イベントステータス] [-t|--type= タイプ] [-e|--effective] (XML ファイルパス) [< 設定名 > < 設定名 >... < 設定名 >]`」
- ◆ 15 ページの「`audit-settings-import-from-file (asiff) [ オプション ] (XML ファイルパス) [ 宛先デバイスまたはデバイスフォルダのパス ] [...]`」

`audit-settings-export-to-file|asetf [-s|--source= ソースデバイスまたはデバイスフォルダのパス] [-a|--auditEventStatus= イベントステータス] [-t|--type= タイプ] [-e|--effective] (XML ファイルパス) [< 設定名 > < 設定名 >... < 設定名 >]`

変更内容とエージェントの監査イベント設定を、ゾーンの XML ファイル、デバイスフォルダ、またはデバイスレベルにエクスポートします。

`-s, --source=< ソースデバイスまたはデバイスフォルダのパス >`: /Devices に対する、設定がエクスポートされる元となるデバイスまたはデバイスフォルダの相対パスで、指定しない場合、設定は管理ゾーンレベルからエクスポートされます。

`-a, --auditEventStatus=< イベントステータス >`: enabled、disabled、または all からなるイベントステータス。デフォルトは all です。

`-t, --type=< タイプ >`: イベントタイプとして、AgentEvents、ChangeEvents、または all を指定できます。デフォルト値は all です。

`-e, --effective`: 指定した場合、有効な設定が取得され、指定しない場合はソースパスで定義されていたり、無効になっている設定のみが取得されます。

(XML ファイルパス): 設定データが XML 形式で保存されるファイルです。ファイルが存在しない場合、新規ファイルが作成されます。

[設定名] [...]: エクスポートする設定の名前。指定しない場合、すべての設定がエクスポートされます。指定した場合は有効な設定が取得され、そうでない場合はソースパスで定義または上書された設定のみが取得されます。

`audit-settings-import-from-file (asiff) [ オプション ] (XML ファイルパス) [ 宛先デバイスまたはデバイスフォルダのパス ] [...]`

変更内容とエージェントの監査イベント設定を、ゾーンの XML ファイル、デバイスフォルダ、またはデバイスレベルからインポートします。

(XML ファイルパス): エクスポートされた設定情報が含まれる XML ファイルです。audit-settings-export-to-file (asetf) を使用して、設定情報を XML ファイルにエクスポートします。

[宛先デバイスまたはデバイスフォルダのパス] [...]: /Devices に対する、設定を設定する必要があるデバイスまたはデバイスフォルダの相対パス。指定しない場合、設定は管理ゾーンレベルで設定されます。

## 管理者コマンド

これらのコマンドは、管理者アカウントの作成と管理に使用されます。管理者コマンドは、長い形式では `admin-` で始まり、短い形式では `a` で始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- 16 ページの「`admin-clear-credential (acc)`」
- 16 ページの「`admin-create (ac) <管理者またはユーザ> [オプション]`」
- 17 ページの「`admin-delete (ad) <ZENworks 管理者名> [...]`」
- 17 ページの「`admin-list (al) [オプション]`」
- 17 ページの「`admin-list-roles (alr) <ZENworks 管理者名> [オプション]`」
- 17 ページの「`admin-rename (ar) <ZENworks 管理者名> (新しい名前)`」
- 17 ページの「`admin-rights-delete (ard) <ZENworks 管理者名> [オプション] [オブジェクトパス]`」
- 17 ページの「`admin-rights-info (ari) <ZENworks 管理者名> [オブジェクトパス] [オプション]`」
- 18 ページの「`admin-rights-set (ars) <ZENworks 管理者名> [権限カテゴリ] [オブジェクトパス] [...] [オプション]`」
- 18 ページの「`admin-store-credential (asc) <ZENworks 管理者名>`」

### `admin-clear-credential (acc)`

`admin-store-credential` コマンドを使用して、保存された ZENworks ユーザ名とパスワードをクリアします。コマンドの実行終了後は常に、安全のために、保存された資格情報を削除することをお勧めします。

### `admin-create (ac) <管理者またはユーザ> [オプション]`

ZENworks 管理者アカウントを作成します。

(管理者またはユーザ) - 管理者の名前またはユーザソースのユーザのフルパス。

次のオプションを受け入れます。

`-f, --fullname=[フルネーム]` - ZENworks 管理者のフルネーム。

`-a, --assign-same-rights` - このコマンドを実行する管理者と同じ権限を新しい管理者に与えます。

クリアテキストでパスワードを入力することを防ぐために、コマンドによって、管理者のパスワードを入力するようプロンプトが表示されます。管理者アカウントが特定のユーザに基づいている場合、ユーザソースで定義された資格情報と同じ資格情報が使用されるため、コマンドの使用時にパスワードの入力が求められることはありません。



#### **admin-delete (ad) <ZENworks 管理者名 > [...]**

ZENworks 管理者アカウントを削除します。

(ZENworks 管理者名) [...] - ZENworks 管理者の名前。引用符で囲む限り、ワイルドカード \* を使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

#### **admin-list (al) [ オプション ]**

ZENworks 管理者アカウントすべてを一覧表示します。

次のオプションを受け入れます。

-n, --namefilter=[ フィルタ文字列 ] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

#### **admin-list-roles (alr) <ZENworks 管理者名 > [ オプション ]**

ZENworks 管理者に割り当てられている役割を一覧表示します。

次のオプションを受け入れます。

-l, --longnames - 権限の名前を表示します。デフォルトでは、権限名は省略形式で表示されます。

#### **admin-rename (ar) <ZENworks 管理者名 > (新しい名前)**

ZENworks 管理者アカウントの名前を変更します。

#### **admin-rights-delete (ard) <ZENworks 管理者名 > [ オプション ] [ オブジェクトパス ]**

ZENworks 管理者アカウントに割り当てられる権限を削除します。

[ オブジェクトパス ] [...] - スラッシュ (/) から始まる ZENworks オブジェクトのフルパス。

次のオプションを受け入れます。

-c, --category=[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、zman admin-rights-set --help | more を実行します。

#### **admin-rights-info (ari) <ZENworks 管理者名 > [ オブジェクトパス ] [ オプション ]**

ZENworks 管理者アカウントに割り当てられる権限を表示します。

[ オブジェクトパス ] [...] - スラッシュ (/) から始まる ZENworks オブジェクトのフルパス。

次のオプションを受け入れます。

-c, --category=[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、zman admin-rights-set --help | more を実行します。

-e, --effective - 管理者の有効な権限を表示します。有効な権限には、管理者に割り当てられた役割から継承された権限が含まれます。このオプションが指定されない場合、管理者に直接割り当てられている権限が表示されます。

-l, --longnames - 権限の名前を表示します。デフォルトでは、権限名は省略形式で表示されます。

## **admin-rights-set (ars) <ZENworks 管理者名 > [ 権限カテゴリ ] [ オブジェクトパス ] [...] [ オプション ]**

ZENworks 管理者アカウントの権限を許可または却下します。

[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、`zman admin-rights-set --help | more` を実行します。

[ オブジェクトパス ] [...] - 権限を適用する ZENworks オブジェクトのフルパスで、スラッシュ (/) で始まります。各権限カテゴリ内で権限を割り当てることができる、許可されているオブジェクトタイプのオブジェクトのルートフォルダを表示するには、`zman admin-rights-modify --help | more` を実行します。

次のオプションを受け入れます。

-a, --allow=[ 権限 ] [...] - 許可されている権限の長い名前または短い名前のカンマ区切りリスト。各カテゴリの権限を表示するには、`zman admin-rights-set --help | more` を実行します。

-A, --assign-all - 権限カテゴリのすべての権限を許可します。

-d, --deny=[ 権限 ] [...] - 拒否する権限の長い名前または短い名前のカンマ区切りリスト。各カテゴリの権限を表示するには、`zman admin-rights-set --help | more` を実行します。

-D, --deny-all - 権限カテゴリのすべての権限を拒否します。

-f, --folder=[ フォルダ名 ] - フォルダのフルパス。このオプションを指定すると、オブジェクトのパスは、このフォルダを基準とした相対パスとして決定されます。これは、同じフォルダで複数のオブジェクトを指定するために使用できます。

-H, --help - 使用可能な権限カテゴリに関する詳しいヘルプおよび各カテゴリ内の権限名の長い形式と短い形式を表示します。カテゴリ名を指定した場合、そのカテゴリに関するヘルプが提供されます。

## **admin-store-credential (asc) <ZENworks 管理者名 >**

各コマンドに使用する ZENworks ユーザ名とパスワードを保存し、資格情報をコマンドごとに入力する必要がなくなります。資格情報はログインユーザのホームディレクトリに保存され、ログインユーザのためにのみ使用されます。Linux では、ユーザがホームディレクトリを持っていることを確認してください。他の ZENworks 管理者と同じマシログインを共有している場合は、コマンドの実行が終了したら、`admin-clear-credential` コマンドを使用して資格情報をクリアする必要があります。

## バッチコマンド

バッチコマンドは、テキストファイルから zman コマンドのリストを実行する場合に使用するユーティリティコマンドです。ZENworks へのログインは一度しか実行されず、すべてのコマンドごとに実行されるわけではないため、コマンドの実行が高速化されます。バッチコマンドは、長い形式では batch- から始まり、短い形式では b から始まります。

このコマンドは、「[グローバルオプション](#)」のセクションに一覧表示されるオプションフラグを受け入れます。さらに、コマンドと併せて一覧表示されている個々のオプションも受け入れます。

- ◆ [19 ページ](#)の「[batch-execute \(bex\) \(ファイルパス\) \[...\] \[オプション\]](#)」

### batch-execute (bex) (ファイルパス) [...] [オプション]

テキストファイルにある zman コマンドをバッチで実行します (Windows のバッチファイルと混同しないこと)。

(ファイルパス) [...] - 実行する zman コマンドを含むテキストファイルのパス。#、rem、REM、;、および // で始まる行はコメントとみなされ、処理されません。サンプルファイルは、BatchExecuteCommands.txt を参照してください：

**Windows の場合** : %ZENWORKS\_HOME%\Novell\Zenworks\share\zman\samples\batch

**Linux の場合** : /opt/novell/zenworks/share/zman/samples/batch

次のオプションを受け入れます。

-e, --exit-on-failure - コマンドが部分的または全体的に失敗した場合、コマンドの実行を停止し終了します。複数の引数を取るコマンドが、1 つ以上の引数を正常に処理できなかった場合、コマンドは部分的に失敗したとみなされます。たとえば、bundle-delete の場合、引数の 1 つが見つからないか削除できないと、コマンドは部分的に失敗したとみなされます。デフォルトでは、ファイル内のコマンドは連続して実行され、失敗はチェックされません。

例：

- ◆ zman コマンドを複数のファイルで実行する場合：

```
zman bex c:\filetxt c:\filetxt
```

ファイルは入力順で実行されます。

- ◆ zman コマンドを複数のファイルで実行するが、コマンドに失敗した場合は終了する場合：

```
zman bex /home/user1/file1 /home/user1/file2 -e
```

ファイルは入力順で実行されます。

## バンドルコマンド

バンドルの作成、変更、および管理を行うコマンドです。バンドルコマンドは、長い形式では bundle- から始まり、短い形式では b から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。これらのコマンドは ZENworks Configuration Management にのみ該当します。

- [22 ページ](#)の「`bundle-add-actions (baa)` (バンドルパス) (アクション XML ファイル) [...] [オプション]」
- [22 ページ](#)の「`bundle-assign (ba)` (デバイスまたはユーザタイプ) (バンドルまたはバンドルグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]」
- [24 ページ](#)の「`bundle-copy (bcp)` (バンドルパス) (バンドル名) [親フォルダ] [オプション]」
- [24 ページ](#)の「`bundle-create (bc)` (バンドル名) (バンドル XML ファイル) [親フォルダ] [オプション]」
- [25 ページ](#)の「`bundle-delete (bd)` (バンドルオブジェクトパス) [...] [オプション]」
- [25 ページ](#)の「`bundle-delete-version (bdv)` (バンドルパス) [バンドルバージョン番号] [オプション]」
- [26 ページ](#)の「`bundle-disable (bd1)` (バンドルパス) [...] [オプション]」
- [26 ページ](#)の「`bundle-enable (be1)` (バンドルパス) [...] [オプション]」
- [26 ページ](#)の「`bundle-export-actions (bea)` (バンドルパス) (XML ファイルパス) (アクションセットタイプ) [アクションの位置] [...]」
- [27 ページ](#)の「`bundle-export-to-file (betf)` (バンドルパス) (XML ファイルパス) [バンドルのバージョン] [-c|--export-content]」
- [27 ページ](#)の「`bundle-folder-create (bfc)` (フォルダ名) [親フォルダ] [オプション]」
- [27 ページ](#)の「`bundle-group-add (bga)` (バンドルグループパス) (バンドルパス) [...] [オプション]」
- [28 ページ](#)の「`bundle-group-create (bgc)` (グループ名) [親フォルダ] [オプション]」
- [28 ページ](#)の「`bundle-group-members (bgm)` (バンドルグループパス) [オプション]」
- [28 ページ](#)の「`bundle-group-remove (bgr)` (バンドルグループパス) (バンドルパス) [...] [オプション]」
- [28 ページ](#)の「`bundle-increment-version (biv)` (バンドルパス) [...] [オプション]」
- [28 ページ](#)の「`bundle-list (bl)` [オプション] [フォルダ]」

- ◆ 29 ページの「bundle-list-actions (bla) (バンドルパス) [アクションセットタイプ]
- ◆ 29 ページの「bundle-list-assignments (blas) (バンドルまたはバンドルグループ) [オプション]
- ◆ 29 ページの「bundle-list-groups (blg) (バンドルパス) [オプション]
- ◆ 29 ページの「bundle-list-version (blv) (バンドルパス)
- ◆ 29 ページの「bundle-move (bmv) (バンドルオブジェクトパス) [宛先フォルダパス]
- ◆ 30 ページの「bundle-remove-actions (bra) (バンドルパス) (アクションセットタイプ) [アクションの位置] [...]
- ◆ 30 ページの「bundle-rename (br) (バンドルオブジェクトパス) (新しい名前) [オプション]
- ◆ 30 ページの「bundle-relocate-actions (brla) (バンドルパス) (アクションセットタイプ) (ターゲットアクションセットタイプ) [< アクションの位置 >, < アクションの位置 >, ..., < アクションの位置 >]
- ◆ 31 ページの「bundle-reorder-actions (broa) (バンドルパス) (アクションセットタイプ) (現在の位置) (新しい位置)
- ◆ 31 ページの「bundle-sandbox-create (bsc) (バンドルパス) [バンドルのバージョン] [オプション]
- ◆ 31 ページの「bundle-sandbox-publish (bsp) (バンドルパス) [オプション]
- ◆ 32 ページの「bundle-sandbox-publish-to-newbundle (bsptn) (元のバンドルパス) (新しいバンドル名) [親フォルダ] [オプション]
- ◆ 33 ページの「bundle-sandbox-revert (bsr) (バンドルパス)
- ◆ 33 ページの「bundle-unassign (bua) (デバイスまたはユーザタイプ) (バンドルまたはバンドルグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]
- ◆ 33 ページの「bundle-update-assignment (bupa) (デバイスまたはユーザタイプ) (バンドルまたはバンドルグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]
- ◆ 35 ページの「bundle-view-advanced-deployment-status (bvads) (バンドルパス) [オプション]
- ◆ 36 ページの「bundle-view-assignment-schedule (bvas) (デバイスまたはユーザタイプ) (バンドルまたはバンドルグループ) (デバイスまたはユーザオブジェクトパス)
- ◆ 36 ページの「bundle-view-summary-totals (bvst) (バンドルパス) [オプション]
- ◆ 37 ページの「linux-export-actions (lea) (アクションタイプ) (アクション XML ファイルのパス) (コンテンツファイルの場所のパス) [オプション]
- ◆ 37 ページの「bundle-appdata-calculation (bac) (バンドルパスまたはバンドル GUID) [...]

## **bundle-add-actions (baa) (バンドルパス) (アクション XML ファイル) [...] [オプション]**

バンドルにアクションを追加します。

(アクション XML ファイル) [...] - 追加するアクションの情報が含まれた XML ファイル。アクションは、Distribution、Install、Launch、Verify、Uninstall、Terminate のいずれかのアクションセットにグループ化されます。XML ファイルには ActionSet 要素が含まれ、アクションセットに対して追加するアクションの情報が記述されます。バンドルの別のアクションセットにアクションを追加する場合、複数の XML ファイルを入力として指定できます。XML ファイルを作成するには、bundle-export-actions コマンドを使用して既存のバンドルのアクションをエクスポートします。

次のオプションを受け入れます。

-a, --actioninfo=[content and dependency for actions] - ファイルコンテンツまたは別のバンドルへの従属関係を持つアクションのコンテンツおよびバンドル従属関係に関する情報が含まれた XML ファイル。たとえば、MSI のインストールアクションの場合、インストールされる MSI ファイルがファイルコンテンツです。バンドルのインストールアクションは、別のバンドルを依存関係として取得するため、これらのアクションを正常に機能させるにはこのデータも指定する必要があります。XML フォーマットテンプレートについては、ActionInfo.xml を参照してください：

**Windows:** %ZENWORKS\_HOME%\Novell\Zenworks\share\zman\samples\bundles

**Linux:** /opt/novell/zenworks/share/zman/samples/bundles

---

**注:** アクションセットタイプに Uninstall を選択する場合、Linux バンドルと Linux パッチバンドルでは次のオプションはサポートされません：

DaysBeforeUninstallWhenNotUsed、DisplayUninstallMessage、

IgnoreUninstallAfterUnassociation、IgnoreChainedDependencies、および

AllowToCancelUninstall。

---

## **bundle-assign (ba) (デバイスまたはユーザタイプ) (バンドルまたはバンドルグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]**

バンドルまたはバンドルグループを 1 つまたは複数のデバイスまたはユーザオブジェクトに割り当てます。

(デバイスまたはユーザタイプ) - 有効な値は、device、server、workstation、および user です。

(デバイスまたはユーザオブジェクトパス) [...] - 指定したデバイスまたはユーザタイプのルートフォルダを基準にした、デバイスまたはユーザオブジェクトの相対パス。

次のオプションを受け入れます。

-f, --folder=[バンドルフォルダ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

-l, --icon-location=[アプリケーションロケーション XML ファイル] - バンドルアプリケーション用のアイコンを配置する場所を含む XML ファイル。XML ファイルフォーマットについては、IconLocation.xml を参照してください：

**Windows:** %ZENWORKS\_HOME%\Novell\Zenworks\share\zman\samples\bundles

**Linux:** /opt/novell/zenworks/share/zman/samples/bundles

-d, --distribution-schedule=[ 配布スケジュール XML ファイル ] - 配布スケジュールを含む XML ファイル。

-l, --launch-schedule=[ 起動スケジュール XML ファイル ] - 起動スケジュールを含む XML ファイル。

-a, --availability-schedule=[ 可用性スケジュール XML ファイル ] - 可用性スケジュールを含む XML ファイル。スケジュール XML ファイルテンプレートについては、XML ファイルを参照してください：

**Windows:** %ZENWORKS\_HOME%\Novell\Zenworks\share\zman\samples\schedules

**Linux:** /opt/novell/zenworks/share/zman/samples/schedules

i, --install-immediately - 配布直後にバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule, --distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。

-L, --launch-immediately - インストール後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule, --distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。

-n, --distribute-now - バンドルをただちに配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule および --distribute-on-device-refresh オプションは無視されます。--distribute-now、--distribute-on-device-refresh、および --distribution-schedule オプションは配布スケジュールの設定に使用し、同時に使用することはできません。最初に --distribute-now オプションが考慮され、次に --distribute-on-device-refresh オプションが考慮され、最後に --distribution-schedule が考慮されます。

-r, --distribute-on-device-refresh - デバイス更新時にバンドルを配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule オプションは無視されます。

-s, --launch-on-device-refresh - デバイス更新時にバンドルを起動するよう起動スケジュールを設定します。このオプションを指定すると、--launch-schedule オプションは無視されます。

-w, --wakeup-device-on-distribution - バンドルの配布中にデバイスがシャットダウンした場合、Wake-On-LAN を使用してデバイスを起動します。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule, --distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。

-B, --broadcast=[ ブロードキャストアドレス ][...] - Wake-On-LAN マジックパケットをブロードキャストするのに使用するアドレスのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。有効な IP アドレスが、有効な値です。

-S, --server=[/ デバイスを基準にしたプライマリサーバオブジェクトまたはプロキシサーバオブジェクトの相対パス ][...] - デバイスを起動するのに使用するプライマリまたはプロキシサーバオブジェクトのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。

- C, --retries=[ 再試行回数 ] - デバイスに Wake-On-LAN マジックパケットを送信する回数。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 0 ~ 5 の範囲である必要があります。デフォルト値は「1」です。
- T, --timeout=[ 再試行間隔 ] - 2 回の再試行の間隔。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 2 ~ 10 の範囲である必要があります (単位は分)。デフォルト値は 2 です。
- N, --app-installation-now-schedule - バンドルを直ちにインストールするようにアプリインストールスケジュールを設定します。このオプションを指定すると、-app-installation-schedule オプションは無視されます。
- M --app-installation-schedule=< アプリインストールスケジュール XML ファイル > - XML ファイルにはインストールスケジュールが含まれます。テンプレートのスケジュールの詳細については、次を参照してください：  
**Windows:** %ZENWORKS\_HOME%\Novell\ZENworks\share\zman\samples\schedules  
**Linux:** /opt/novell/zenworks/share/zman/samples/schedule
- c, --conflicts=[ バンドル競合解決順序 ] - バンドル競合を解決する方法を決定します。有効な値は userprecedence または 1、deviceprecedence または 2 です。userprecedence の場合、ユーザ関連のバンドルが最初に適用され、その後デバイス関連のバンドルが続きます。deviceprecedence の場合、デバイス関連のバンドルが最初に適用され、その後ユーザ関連のバンドルが続きます。このオプションを指定しなかった場合には、userprecedence がデフォルト値として使用されます。

#### **bundle-copy (bcp) (バンドルパス) (バンドル名) [親フォルダ] [オプション]**

バンドルをコピーします。

次のオプションを受け入れます。

- d, --desc=[ 説明 ] - バンドルの説明。
- i, --icon-file=[ バンドルアイコンファイル ] - バンドルアイコンとして使用するイメージが含まれるイメージファイルのパス。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。

#### **bundle-create (bc) (バンドル名) (バンドル XML ファイル) [親フォルダ] [オプション]**

新しいバンドルを作成します。

(バンドル XML ファイル) - エクスポートしたバンドル情報が含まれる XML ファイル。bundle-export-to-file (betf) を使用して、既存のバンドルの情報を XML ファイルにエクスポートします。ファイルを再使用する場合には、ZENworks コントロールセンターで作成したバンドルからテンプレート XML ファイルを作成することができます。サンプル XML ファイルについては、WindowsMSIBundle.xml を参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/bundles、Windows サーバでは *Installation\_directory:\Novell\Zenworks\share\zman\samples\bundles* に保存されています。

次のオプションを受け入れます。

- d, --desc=[ 説明 ] - バンドルの説明。
- a, --actioninfo=[content and dependency for actions] - ファイルコンテンツまたは別のバンドルへの従属関係を持つアクションのコンテンツおよびバンドル従属関係に関する情報が含まれた XML ファイル。たとえば、MSI のインストールアクションの場合、イ



インストールされる MSI ファイルがファイルコンテンツです。バンドルのインストールアクションは、従属関係として別のバンドルを取ります。これらのアクションを正常に機能させるには、このデータも指定する必要があります。XML 形式のテンプレートについては、ActionInfo.xml を参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/bundles、Windows サーバでは

*Installation\_directory:\Novell\Zenworks\share\zman\samples\bundles* に保存されています。

- i, --icon-file=[ バンドルアイコンファイル ] - バンドルアイコンとして使用するイメージが含まれるイメージファイルのパス。
- s, --create-as-sandbox=[ サンドボックスとして作成 ] - バンドルをサンドボックスとして作成します。
- x, --actions-as-in-xml - テンプレート XML ファイルにある新しいバンドルのデフォルトアクションを作成します。たとえば、[ 検証 ] アクションタイプで、デフォルトアクションの [ インストールを検証する ] をバンドルから削除し、バンドル情報を XML ファイルにインポートする場合、新しいバンドルでは、デフォルトアクションの [ インストールを検証する ] を除き、XML ファイルのすべてのデフォルトアクションが作成されます。このオプションを指定しない場合、XML ファイルにはないデフォルトアクションの [ インストールを検証する ] も含め、すべてのデフォルトアクションを使用してバンドルが作成されます。
- n, --create-with-new-grid - 新しい GUID を使用してポリシーオブジェクトを作成します。オプションが指定されない場合、ポリシー XML ファイル内のインポートされた GUID のポリシーオブジェクトが検証されます。インポートされた GUID が存在しない場合、新しい GUID が作成されるか、インポートされた GUID が保持されます。

---

**注:** アクションセットタイプに Uninstall を選択する場合、Linux バンドルと Linux パッチバンドルでは次のオプションはサポートされません:

DaysBeforeUninstallWhenNotUsed、DisplayUninstallMessage、IgnoreUninstallAfterUnassociation、IgnoreChainedDependencies、および AllowToCancelUninstall。

---

#### **bundle-delete (bd) ( バンドルオブジェクトパス ) [...][ オプション ]**

1 つまたは複数のバンドルオブジェクトを削除します。

( バンドルオブジェクトパス ) [...] - / バンドルを基準にした、バンドルオブジェクト ( バンドル、バンドルフォルダ、またはバンドルグループ ) の相対パス。引用符で囲む限り、ワイルドカード \* を使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

次のオプションを受け入れます。

- r, --recursive - フォルダ内のオブジェクトを再帰的に削除します。
- f, --folder=[ バンドルフォルダ ] - / バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

#### **bundle-delete-version (bdv) ( バンドルパス ) [ バンドルバージョン番号 ] [ オプション ]**

1 つまたは複数のバンドルバージョンを削除します。

[ バンドルバージョン番号 ] [...] - 削除するバンドルのバージョン。

次のオプションを受け入れます。

-a --all - バンドルの古いバージョンをすべて削除します。

-p, --previous - 指定されたバージョンより古いバージョンのバンドルをすべて削除します。

例：

バンドル zenbundle のバージョン 5 を削除する場合：

```
zman bdv zenbundle 5
```

バンドル zenbundle のすべてのバージョンを削除する場合：

```
zman bdv zenbundle -a
```

バンドル zenbundle のバージョン 3 以前のすべてのバージョンを削除する場合：

```
zman bdv zenbundle version -p 3
```

### **bundle-disable (bdl) (バンドルパス)[...][オプション]**

バンドルを無効にします。

次のオプションを受け入れます。

-f, --folder=[バンドルフォルダ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

### **bundle-enable (bel) (バンドルパス)[...][オプション]**

バンドルを有効にします。

次のオプションを受け入れます。

-f, --folder=[バンドルフォルダ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

### **bundle-export-actions (bea) (バンドルパス)(XML ファイルパス)(アクションセットタイプ)[アクションの位置][...]**

バンドルのアクションセットに追加したアクションをファイルにエクスポートします。このファイルを bundle-add-actions コマンドの入力として使用できます。

(アクションセットタイプ) - アクションセットのタイプ。このオプションを指定した場合、このアクションセットのアクションのみが一覧表示されます。有効な値は、Install、Launch、Verify、Uninstall、Terminate、および Preboot です。

---

**注** : Terminate アクションセットは、Linux バンドルおよび Linux パッチバンドルに対してはサポートされません。

---

[アクションの位置][...] - アクションセット内におけるアクションの位置のカンマ区切りリスト。最初のアクションが位置 1 になります。アクションセット内の特定のアクションの位置を参照するには、`bundle-list-actions` コマンドを使用します。これはオプションですが、指定しなかった場合は、指定したアクションセット内のすべてのアクションがエクスポートされます。

**bundle-export-to-file (betf) (バンドルパス)(XML ファイルパス)[バンドルのバージョン][-c|--export-content]**

バンドルの情報 (XML 形式) をファイルにエクスポートします。この XML ファイルはバンドル作成時の入力として使用します。

(XML ファイルパス) - バンドルの情報をエクスポートする XML ファイルの完全なパス。

[バンドルのバージョン] - エクスポートするバンドルのバージョン。このオプションを指定しない場合、バンドルの発行済みバージョンが対象となります。バンドルのサンドボックスバージョンをエクスポートするには、`sandbox` を指定します。

[-c|--export-content] - バンドル情報のエクスポート先である XML ファイルが格納されているディレクトリ内のサブディレクトリに、バンドルのコンテンツをエクスポートします。オプションを指定しない場合、バンドルコンテンツはエクスポートされません。

例 :

zenbundle という名前のサンドボックスバージョンのバンドルを、`bundle.xml` という名前の xml ファイルにエクスポートする場合 :

```
zman betf bundlefolder/zenbundle C:\bundles\bundle.xml sandbox
```

zenbundle という名前のバージョン 3 のバンドルを `bundle.xml` という名前の xml ファイルにエクスポートする場合 :

```
zman betf bundlefolder/zenbundle C:\bundles\bundle.xml 3
```

zenbundle という名前のバージョン 5 のバンドルを `bundle.xml` という名前の xml ファイルにエクスポートし、バンドルのコンテンツを `C:\bundles\bundle_content` ディレクトリにエクスポートする場合 :

```
zman betf bundlefolder/zenbundle C:\bundles\bundle.xml 5 -c
```

**bundle-folder-create (bfc) (フォルダ名)[親フォルダ][オプション]**

バンドルを格納するための新しいフォルダを作成します。

次のオプションを受け入れます。

--desc=[説明] - フォルダの説明。

**bundle-group-add (bga) (バンドルグループパス)(バンドルパス)[...][オプション]**

バンドルをバンドルグループに追加します。

次のオプションを受け入れます。

-f, --folder=[ バンドルフォルダ ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

#### **bundle-group-create (bgc) (グループ名) [親フォルダ] [オプション]**

バンドルグループを作成して、メンバーを追加します。

次のオプションを受け入れます。

--desc=[ 説明 ] - グループの説明。

-m, --members=[ バンドルパス ][...] - /バンドルを基準にした、バンドルの相対パス。

-f, --folder=[ バンドルフォルダ ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

#### **bundle-group-members (bgm) (バンドルグループパス) [オプション]**

バンドルグループのメンバーを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **bundle-group-remove (bgr) (バンドルグループパス) (バンドルパス) [...] [オプション]**

バンドルグループからバンドルを削除します。

次のオプションを受け入れます。

-f, --folder=[ バンドルフォルダ ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

#### **bundle-increment-version (biv) (バンドルパス) [...] [オプション]**

バンドルのバージョンを増やします。このコマンドは、ZENworks 11SP1 より前のバージョンに適用されます。ZENworks 11 SP1 以降では、このコマンドは bundle-sandbox-publish (bsp) コマンドで置き換えられます。

次のオプションを受け入れます。

-f, --folder=[ バンドルフォルダ ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

#### **bundle-list (bl) [オプション] [フォルダ]**

バンドルオブジェクトを一覧表示します。

次のオプションを受け入れます。

- r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。
- n, --namefilter=[ フィルタ文字列 ] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。
- c, --count - 結果の数が表示されます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

#### **bundle-list-actions (bla) (バンドルパス) [アクションセットタイプ]**

バンドルに追加されたアクションを一覧にします。

[アクションセットタイプ] - アクションセットのタイプ。このオプションを指定した場合、このアクションセットのアクションのみが一覧表示されます。有効な値は、Install、Launch、Verify、Uninstall、Terminate、および Preboot です。

---

**注 :** Terminate アクションセットは、Linux バンドルおよび Linux パッチバンドルに対してはサポートされません。

---

#### **bundle-list-assignments (blas) (バンドルまたはバンドルグループ) [オプション]**

バンドルのデバイスとユーザ割り当てを一覧表示します。

次のオプションを受け入れます。

-t, --typefilter=[ 割り当てのタイプ ] - 割り当てタイプでフィルタします。有効な値は、device および user です。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **bundle-list-groups (blg) (バンドルパス) [オプション]**

特定のバンドルがメンバーであるバンドルグループを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

例 :

bundle1 が属しているすべてのバンドルグループを、5 つずつ同時に一覧表示する場合 :

```
zman blg bundlefolder/bundle1 -s 5
```

#### **bundle-list-version (blv) (バンドルパス)**

バンドルのすべてのバージョンを一覧表示します。

例 :

バンドル zenbundle のすべてのバージョンを一覧表示する場合 :

```
zman blv zenbundle
```

#### **bundle-move (bmv) (バンドルオブジェクトパス) [宛先フォルダパス]**

バンドルオブジェクトを別のフォルダに移動します。

(バンドルオブジェクトパス): /Bundles を基準にした、バンドルオブジェクト (バンドル、バンドルフォルダ、またはバンドルグループ) の相対パス

[宛先フォルダパス] /Bundles を基準にした、バンドルオブジェクトが移動するバンドルフォルダの相対パス。

**bundle-remove-actions (bra) (バンドルパス) (アクションセットタイプ) [アクションの位置] [...]**

バンドルからアクションを削除します。

(アクションセットタイプ) - アクションが含まれるアクションセットのタイプ。有効な値は、Install、Launch、Verify、Uninstall、Terminate、および Preboot です。

---

**注:** Terminate アクションセットは、Linux バンドルおよび Linux パッチバンドルに対してはサポートされません。

---

[アクションの位置] [...] - アクションセット内におけるアクションの位置のカンマ区切りリスト。最初のアクションが位置 1 になります。アクションセット内の特定のアクションの位置を参照するには、bundle-list-actions コマンドを使用します。これはオプションですが、指定しなかった場合は、指定したアクションセット内のすべてのアクションが削除されます。

**bundle-rename (br) (バンドルオブジェクトパス) (新しい名前) [オプション]**

バンドルオブジェクトを名前変更します。

(バンドルオブジェクトパス): /Bundles に対するバンドルオブジェクト (バンドル、バンドルフォルダ、またはバンドルグループ) の相対パス。

(新しい名前): バンドルオブジェクトに付ける新しい名前。

-p, --publish [=発行] - このオプションが指定され、バンドルの表示名が編集された場合、バンドルは即座に発行されます。ただし、現在のバンドルの表示名が既存のバンドル名と異なる場合、またはバンドルにサンドボックスがすでに存在する場合は、このオプションを指定しても効果はありません。

**bundle-relocate-actions (brla) (バンドルパス) (アクションセットタイプ) (ターゲットアクションセットタイプ) [<アクションの位置>, <アクションの位置>, ..., <アクションの位置>]**

バンドルのソースアクションセットタイプからターゲットアクションセットタイプにアクションを移転します。

(バンドルパス) - /Bundles

を基準にした、バンドルのパス。(アクションセットタイプ) - ソースアクションを含むアクションセットのタイプ。有効な値は、Install、Launch、Verify、Uninstall、Terminate、および Preboot です。

[ターゲットのアクションセットタイプ] - ターゲットのアクションを含むアクションセットのタイプ。有効な値は、Install、Launch、Verify、Uninstall、Terminate、および Preboot です。

[アクションの位置] - アクションセット内における各アクションの位置のカンマ区切りリスト。最初のアクションが位置 1 になります。アクションセット内の特定のアクションの位置を識別するには、bundle-list-actions コマンドを使用します。これはオプションで、指定しなかった場合は、指定したアクションセット内のすべてのアクションが移転されます。

例:

バンドルの Launch アクションセットから Install アクションセットにすべてのアクションを移転するには: `zman brla bundlefolder/bundle1 Launch Install`

バンドルの Install アクションセットから Launch アクションセットに最初の 3 つのアクションを移転するには: `zman brla bundle1 Install Launch 1,2,3`

---

**注:** Terminate アクションセットは、Linux バンドルおよび Linux パッチバンドルに対してはサポートされません。

---

**bundle-reorder-actions (broa) (バンドルパス) (アクションセットタイプ) (現在の位置) (新しい位置)**

バンドルのアクションセット内にあるアクションを並べ替えます。

(アクションセットタイプ) - アクションが含まれるアクションセットのタイプ。有効な値は、Install、Launch、Verify、Uninstall、Terminate、および Preboot です。

(現在の位置) - アクションセットのアクションの現在の位置。最初のアクションが位置 1 になります。

(新しい位置) - アクションを移動する新しい位置。最初のアクションが位置 1 になります。

---

**注:** Terminate アクションセットは、Linux バンドルおよび Linux パッチバンドルに対してはサポートされません。

---

**bundle-sandbox-create (bsc) (バンドルパス) [バンドルのバージョン] [オプション]**

既存のバンドルバージョンからサンドボックスを作成します。

[バンドルのバージョン] - サンドボックスの作成に使用されるバンドルのバージョンを指定します。

次のオプションを受け入れます。

-f, --force =[強制] - 既存のサンドボックスを上書きします。

例:

バンドル zenbundle のバージョン 2 からサンドボックスを作成する場合:

```
zman bsc zenbundle 2
```

**bundle-sandbox-publish (bsp) (バンドルパス) [オプション]**

サンドボックスを発行して、発行済みバージョンを作成します。

次のオプションを受け入れます。

-f, --force =[強制] - サンドボックスのみのバージョンを持つ依存子バンドルがプライマリバンドルにある場合、このオプションを指定してプライマリバンドルと依存バンドルの両方の変更を発行する必要があります。このオプションを指定しないと、プライマリバンドルの発行も失敗します。

-c, --allChild =[すべての子] - サンドボックスバージョンを持つ依存子バンドルがプライマリバンドルにある場合、このオプションを指定してプライマリと依存子バンドルの両方の変更を発行できます。

-i, --incAllParent =[すべての親を増分]- プライマリバンドルに親バンドルがある場合、このオプションを使用して親バンドルのバージョンを上げ、プライマリバンドルへの新しく発行された変更を親バンドルに割り当てられたデバイスに適用できます。

-p, --forceParent =[親を強制]- プライマリバンドルに親バンドルがあり、親バンドルのバージョンを上げる場合、このオプションを指定して、すでにサンドボックスバージョンを持つ任意の親バンドルを発行する必要があります。

例:

サンドボックスバージョンのバンドル (zenbundle) を発行する場合:

```
zman bsp zenbundle
```

zenbundle1 という名前のサンドボックスバージョンのバンドルを、そのすべての依存バンドルと共に発行する場合:

```
zman bsp zenbundle1 -f -c
```

### **bundle-sandbox-publish-to-newbundle (bsptn) (元のバンドルパス) (新しいバンドル名) [親フォルダ] [オプション]**

サンドボックスを新しいバンドルに発行します。

次のオプションを受け入れます。

-s, --createAsSandbox =[サンドボックスとして作成]- バンドルをサンドボックスとして作成します。

-g, --groups -新しく作成したバンドルを、プライマリバンドルがメンバーになっているすべてのグループに追加します。

-d, --deviceAssignments -すべてのデバイス割り当てをプライマリバンドルから新しいバンドルにコピーします。

-u, --userAssignments -すべてのユーザ割り当てをプライマリバンドルから新しいバンドルにコピーします。

-f, --force =[強制]- サンドボックスのみのバージョンを持つ依存子バンドルがプライマリバンドルにある場合、このオプションを指定してプライマリバンドルと依存バンドルの両方の変更を発行する必要があります。このオプションを指定しないと、プライマリバンドルの発行も失敗します。

-c, --allChild =[すべての子]- サンドボックスバージョンを持つ依存子バンドルがプライマリバンドルにある場合、このオプションを指定してプライマリと依存子バンドルの両方の変更を発行できます。

-i, --incAllParent =[すべての親を増分]- プライマリバンドルに親バンドルがある場合、このオプションを使用して親バンドルのバージョンを上げ、プライマリバンドルへの新しく発行された変更を親バンドルに割り当てられたデバイスに適用できます。

-p, --forceParent =[親を強制]- プライマリバンドルに親バンドルがあり、親バンドルのバージョンを上げる場合、このオプションを指定して、すでにサンドボックスバージョンを持つ任意の親バンドルを発行する必要があります。



-n, -depsToNewFolder =[ 依存バンドルを新しいフォルダに発行 ] - 依存バンドルが新しいバンドルとして発行されるバンドルフォルダのパス。新しいバンドルが元の依存バンドルと同じ名前で発行されるため、このフォルダには、発行対象に選択されている依存バンドルを格納できません。

例：

zenbundle1 という名前のサンドボックスバージョンのバンドルを、zenbundle2 という名前の新しいバンドルとして発行する場合：

```
zman bsptn zenbundle1 zenbundle2 /Bundles/Folder1
```

zenbundle1 という名前のサンドボックスバージョンのバンドルを zenbundle2 という名前の新しいバンドルとして発行し、zenbundle1 の依存バンドルを新しいバンドルとして発行する場合：

```
zman bsptn zenbundle1 zenbundle2 /Bundles/Folder1 --depsToNewFolder=/Bundles/Folder2
```

#### **bundle-sandbox-revert (bsr) (バンドルパス)**

サンドボックスを削除して、バンドルの最新バージョンに戻します。

例：

バンドル zenbundle のサンドボックスを削除して、バンドルの最新バージョンに戻す場合：

```
zman bsr zenbundle
```

#### **bundle-unassign (bua) ( デバイスまたはユーザタイプ )( バンドルまたはバンドルグループ )( デバイスまたはユーザオブジェクトパス ) [ ... ] [ オプション ]**

バンドルまたはバンドルグループを 1 つまたは複数のデバイスまたはユーザオブジェクトから割り当て解除します。

( デバイスまたはユーザタイプ ) - 有効な値は、device、server、workstation、および user です。

( デバイスまたはユーザオブジェクトパス ) [ ... ] - 指定したデバイスまたはユーザタイプのルートフォルダを基準にした、デバイスまたはユーザオブジェクトの相対パス。

次のオプションを受け入れます。

-f, --folder=[ バンドルフォルダ ] - / バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

#### **bundle-update-assignment (bupa) ( デバイスまたはユーザタイプ )( バンドルまたはバンドルグループ )( デバイスまたはユーザオブジェクトパス ) [ ... ] [ オプション ]**

バンドルまたはバンドルグループとデバイスまたはユーザオブジェクト間の割り当てを更新します。

( デバイスまたはユーザタイプ ) - 有効な値は、device、server、workstation、および user です。

( デバイスまたはユーザオブジェクトパス ) [ ... ] - 指定したデバイスまたはユーザタイプのルートフォルダを基準にした、デバイスまたはユーザオブジェクトの相対パス。

次のオプションを受け入れます。

- f, --folder=[ デバイスまたはユーザフォルダ ] - ルートフォルダを基準にした、デバイスまたはユーザフォルダの相対パス。このオプションを指定すると、デバイスまたはユーザオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のデバイスまたはユーザオブジェクトを指定するのに使用できます。
- l, --icon-location=[ アプリケーションロケーション XML ファイル ] - バンドルアプリケーション用のアイコンを配置する場所を含む XML ファイル。XML ファイル形式については、IconLocation.xml を参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/bundles、Windows サーバでは *Installation\_directory:\Novell\Zenworks\share\zman\samples\ bundles* に保存されています。
- d, --distribution-schedule=[ 配布スケジュール XML または NoSchedule ] - 配布スケジュール、またはスケジュールを削除する必要がある場合に NoSchedule 値を指定する XML ファイル。
- l, --launch-schedule=[ 起動スケジュール XML ファイルまたは NoSchedule ] - 起動スケジュール、またはスケジュールを削除する必要がある場合に NoSchedule 値を指定する XML ファイル。
- a, --availability-schedule=[ 可用性スケジュール XML ファイルまたは NoSchedule ] - 可用性スケジュール、またはスケジュールを削除する必要がある場合に NoSchedule 値を指定する XML ファイル。スケジュール XML ファイルテンプレートについては、XML ファイルを参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/schedules、Windows サーバでは *Installation\_directory:\Novell\Zenworks\share\zman\samples\ schedules* に保存されています。
- i, --install-immediately=[yes または no] - 配布後ただちにバンドルをインストールするか、またはバンドルの割り当て中に設定された同じオプションを取り消します。有効な値は、true または yes、および false または no です。
- L, --launch-immediately=[yes または no] - インストール後ただちにバンドルを起動するか、またはバンドルの割り当て中に設定された同じオプションを取り消します。有効な値は、true または yes、および false または no です。
- n, --distribute-now=[yes または no] - バンドルをただちに配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule および --distribute-on-device-refresh オプションは無視されます。--distribute-now、--distribute-on-device-refresh、および --distribution-schedule オプションは配布スケジュールの設定に使用し、同時に使用することはできません。最初に --distribute-now オプションが考慮され、次に --distribute-on-device-refresh オプションが考慮され、最後に --distribution-schedule が考慮されます。
- r, --distribute-on-device-refresh=[yes または no] - デバイス更新時にバンドルを配布するよう配布スケジュールを設定するか、またはバンドルの割り当て中に設定された同じオプションを取り消します。有効な値は、true または yes、および false または no です。このオプションの値が true または yes の場合、--distribution-schedule オプションは無視され、以前設定された配布スケジュールが上書きされます。
- s, --launch-on-device-refresh=[yes または no] - デバイス更新時にバンドルを起動するよう起動スケジュールを設定します。このオプションを指定すると、--launch-schedule オプションは無視されます。

- w, --wakeup-device-on-distribution=[yes または no] - バンドルの配布中にデバイスがシャットダウンした場合、Wake-On-LAN を使用してデバイスを起動するか、またはバンドルの割り当て中に設定された同じオプションを取り消します。有効な値は、true または yes、および false または no です。
- B, --broadcast=[ブロードキャストアドレス][...] - Wake-On-LAN マジックパケットをブロードキャストするのに使用するアドレスのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。有効な IP アドレスが、有効な値です。
- S, --server=[/Devices を基準にしたプライマリサーバオブジェクトまたはプロキシサーバオブジェクトの相対パス][...] - デバイスを起動するのに使用するプライマリまたはプロキシサーバオブジェクトのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。
- C, --retries=[再試行回数] - デバイスに Wake-On-LAN マジックパケットを送信する回数。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 0 ~ 5 の範囲である必要があります。デフォルト値は「1」です。
- T, --timeout=[再試行間隔] - 2 回の再試行の間隔。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 2 ~ 10 の範囲である必要があります (単位は分)。デフォルト値は 2 です。
- M --app-installation-schedule=< アプリインストールスケジュール XML ファイル > - XML ファイルにはインストールスケジュールが含まれます。
- c, --conflicts=[バンドル競合解決順序] - バンドル競合の解決方法を決定します。有効な値は userprecedence または 1、deviceprecedence または 2 です。userprecedence の場合、ユーザ関連のバンドルが最初に適用され、その後デバイス関連のバンドルが続きます。deviceprecedence の場合、デバイス関連のバンドルが最初に適用され、その後ユーザ関連のバンドルが続きます。このオプションを指定しなかった場合には、userprecedence がデフォルト値として使用されます。

#### **bundle-view-advanced-deployment-status (bvads) (バンドルパス)[オプション]**

バンドルの展開ステータスの詳細を表示します。

次のオプションを受け入れます。

- d, --device=[デバイスパス] - 指定されたデバイスのみに対する配布ステータスを表示します。デバイスのパスは、/デバイスを基準にした相対パス。
- u, --user=[ユーザパス] - 指定したユーザのみの配布ステータスを表示します。ユーザのパスは、/ユーザを基準にした相対パス。デバイスも指定した場合、指定したデバイスにログインしている指定したユーザのステータス詳細が表示されます。
- n, --namefilter=[ターゲットデバイス名] - デバイスの名前でフィルタします。指定したフィルタに一致するオプションを表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。
- statusfilter=[ステータスタイプ][...] - バンドル配布およびインストールイベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。
- t, --typefilter=[ターゲットデバイスまたはユーザタイプ][...] - ターゲットのタイプでフィルタします。有効な値は、server、workstation、および user です。ターゲットタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

-v, --version[値]

有効な値は次のとおりです。

- ◆ 「published」 : このオプションを使用して、バンドルを展開する非テストデバイスの数を表示します。
- ◆ 「sandbox」 : このオプションを使用して、バンドルのサンドボックスバージョンを展開するテストデバイスの数を表示します。
- ◆ [version-of-the-object]: バージョン番号を使用して関連するバンドルバージョンのステータスを取得します。  
デフォルト値は「published」です。

-c, --status-calculation[値]

- ◆ 「consolidated」 : このオプションを使用して、このバンドルを展開するデバイスの合計数を表示します。
- ◆ 「version」 : このオプションを使用して、特定のバージョンのステータスを表示します。

デフォルト値は「version」です。

#### **bundle-view-assignment-schedule (bvas) (デバイスまたはユーザタイプ)(バンドルまたはバンドルグループ)(デバイスまたはユーザオブジェクトパス)**

デバイスまたはユーザオブジェクトに割り当てたバンドルの展開スケジュールを表示します。

(デバイスまたはユーザタイプ) - 有効な値は、device、server、workstation、および user です。

(デバイスまたはユーザオブジェクトパス) [...] - バンドルグループの割り当て先を基準にした、デバイスまたはユーザオブジェクトの相対パス。

#### **bundle-view-summary-totals (bvst) (バンドルパス) [オプション]**

特定のバンドルの展開ステータスのサマリを表示します。

次のオプションを受け入れます。

-v, --version[値]

有効な値は次のとおりです。

- ◆ 「published」 : このオプションを使用して、バンドルを展開する非テストデバイスの数を表示します。
- ◆ 「sandbox」 : このオプションを使用して、バンドルのサンドボックスバージョンを展開するテストデバイスの数を表示します。
- ◆ [version-of-the-object]: バージョン番号を使用して関連するバンドルバージョンのステータスを取得します。  
デフォルト値は「published」です。

-c, --status-calculation[ 値 ]

- ◆ 「consolidated」 : このオプションを使用して、このバンドルを展開するデバイスの合計数を表示します。
- ◆ 「version」 : このオプションを使用して、特定のバージョンのステータスを表示します。

デフォルト値は「version」です。

**linux-export-actions (lea) (アクションタイプ)(アクション XML ファイルのパス)(コンテンツファイルの場所のパス) [オプション]**

XML ファイルを作成してパッケージメタデータをエクスポートします。baa コマンドを使用してエクスポートされたコンテンツをサーバに追加します。

(アクションタイプ)-XML ファイルを作成するアクションを指定します。サポートするアクションタイプは、Install RPM と Distribute RPM です。

(アクション XML ファイルのパス)-生成されたアクション定義 XML ファイルの場所。  
(コンテンツファイルの場所のパス)-ローカルマシンの RPM コンテンツファイルの場所。

次のオプションを受け入れます。

-t, --target - パッケージの有効な Linux 配布パッケージターゲットを指定します。

**bundle-appdata-calculation (bac) (バンドルパスまたはバンドル GUID) [...]**

バンドルのアプリケーションデータを計算してキャッシュするローダアクションをキューに入れ、アクションが完全に実行されるまで待機します。キャッシュされたデータは、Web サービス応答の準備とパフォーマンス向上のために使用されます。

---

**注** : zman bac コマンドは、ZENworks Patch Management 内のパッチポリシーにのみ適用できます。

---

## バンドル設定コマンド

バンドル設定の設定と変更を使用するコマンドです。設定コマンドは、長い形式では settings- から始まり、短い形式では s から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- ◆ [38 ページの「bundle-settings-copy \(bscp\) \(ソースバンドルまたはバンドルフォルダパス\) \(宛先バンドルまたはバンドルフォルダパス\) \[...\] \[オプション\]」](#)
- ◆ [38 ページの「bundle-settings-export-to-file \(bsetf\) \[オプション\] \(XML ファイルパス\) \[設定名\] \[...\]」](#)
- ◆ [38 ページの「bundle-settings-revert \(bsrt\) \(ソースバンドルまたはバンドルフォルダパス\) \(設定名\) \[...\]」](#)
- ◆ [38 ページの「bundle-settings-set \(bss\) \(XML ファイルパス\) \[宛先バンドルまたはバンドルフォルダパス\] \[...\]」](#)

**bundle-settings-copy (bscp)** ( ソースバンドルまたはバンドルフォルダパス ) ( 宛先バンドルまたはバンドルフォルダパス ) [...][ オプション ]

設定をコピーします。

( ソースバンドルまたはバンドルフォルダパス ) - / バンドルを基準にした、設定のコピー元バンドルまたはバンドルフォルダの相対パス。

( 宛先バンドルまたはバンドルフォルダパス ) [...] - / バンドルを基準にした、設定のコピー先バンドルまたはバンドルフォルダの相対パス。

次のオプションを受け入れます。

-n, --names=[ 設定名 ] [...] - ソースパスからコピーされる設定の名前。指定しなかった場合、ソースパスで定義されたすべての設定がコピーされます。

**bundle-settings-export-to-file (bsetf)** [ オプション ] ( XML ファイルパス ) [ 設定名 ] [...]

設定データ ( XML 形式 ) をファイルにエクスポートします。設定の作成および変更用の入力として XML ファイルが使用されます。

( XML ファイルパス ) - 設定データが XML 形式で保存されるファイル。ファイルが存在しない場合、新しいファイルが作成されます。

[ 設定名 ] [...] - エクスポートする設定の名前。指定しなかった場合、すべての設定がエクスポートされます。

次のオプションを受け入れます。

-s, --source=[ ソースバンドルまたはバンドルフォルダパス ] - / バンドルを基準にした、設定のエクスポート元バンドルまたはバンドルフォルダの相対パス。指定しなかった場合、設定はゾーンからエクスポートされます。

-e, --effective - 指定した場合は有効な設定が取得されます。そうでない場合はソースパスで定義 / 上書きされた設定のみが取得されます。

**bundle-settings-revert (bsrt)** ( ソースバンドルまたはバンドルフォルダパス ) ( 設定名 ) [...]

親フォルダへの設定を取り消します。

( ソースバンドルまたはバンドルフォルダパス ) - / バンドルを基準にした、設定を元に戻すバンドルまたはバンドルフォルダの相対パス。

**bundle-settings-set (bss)** ( XML ファイルパス ) [ 宛先バンドルまたはバンドルフォルダパス ] [...]

さまざまなレベルで設定を行います。

( XML ファイルパス ) - エクスポートされた設定情報が含まれる XML ファイル。settings-export-to-file (setf) を使用して、設定情報を XML ファイルにエクスポートします。

次のオプションを受け入れます。

[ 設定名 ] [...] - エクスポートする設定の名前。指定しなかった場合、すべての設定がエクスポートされます。

-f, --force - すべての子 ( サブフォルダおよび個々のバンドル ) にこれらの設定の継承を強制します。

(XML ファイルパス): エクスポートされた設定情報が含まれる XML ファイル。bundle-setting-export-to-file (bsetf) を使用して、設定情報を XML ファイルにエクスポートします。

(宛先バンドルまたはバンドルフォルダパス): /Bundle を基準にした、設定の設定元バンドルまたはバンドルフォルダの相対パス。指定しない場合、設定は管理ゾーンレベルで設定されます。

## 証明書の権限コマンド

ZENworks サーバの証明書の権限ロールを管理するコマンドです。証明書の権限コマンドは、長い形式では certificate-authority- から始まり、短い形式では ca から始まります。

- ◆ 39 ページの「(certificate-authority-export/cae) (ファイルパス) [オプション]」
- ◆ 39 ページの「certificate-authority-import (certificate-authority-import/cai) (ファイルパス)」
- ◆ 39 ページの「certificate-authority-role-disable (card)」
- ◆ 39 ページの「certificate-authority-role-enable (care)」
- ◆ 39 ページの「certificate-authority-server (certificate-authority-server/cas)」

### (certificate-authority-export/cae) (ファイルパス) [オプション]

ゾーン証明書権限と資格情報のキーの組み合わせをファイルにエクスポートして、任意でローカルサーバの証明書の権限ロールを無効にします。

次のオプションを受け入れます。

-d, --disable-CA-role - ローカルサーバの証明書の権限ロールを削除します。

### certificate-authority-import (certificate-authority-import/cai) (ファイルパス)

ゾーン証明書権限と資格情報のキーの組み合わせをファイルからインポートして、ローカルサーバの証明書の権限ロールを有効にします。

### certificate-authority-role-disable (card)

ローカルサーバの証明書の権限ロールを無効にします。

### certificate-authority-role-enable (care)

ローカルサーバの証明書の権限ロールを有効にします。

### certificate-authority-server (certificate-authority-server/cas)

認証局の役割を持つサーバを表示します。

## 資格情報コマンド

これらのコマンドは、特定の ZENworks アクションに必要な資格情報およびリソースの認証が必要なタスクを管理するために使用されます。資格情報コマンドは、長い形式では `credentials-` から始まり、短い形式では `cr` から始まります。

- ◆ 40 ページの「`credentials-create (crc) (資格情報名) [フォルダ] (--userName) [--userPassword] [--desc]`」
- ◆ 40 ページの「`credentials-delete (crd) (資格情報パス)`」
- ◆ 40 ページの「`credentials-folder-create (crf) (フォルダ名) [親フォルダ] [オプション]`」
- ◆ 40 ページの「`credentials-list (crl) [フォルダ]`」
- ◆ 40 ページの「`credentials-move (crm) (資格情報名) [宛先フォルダパス]`」
- ◆ 40 ページの「`credentials-rename (crr) (資格情報パス) (新しい資格情報パス)`」
- ◆ 40 ページの「`credentials-update (cru) (-u | --userName= ユーザ名) [--userPassword= パスワード] [-d | --desc= 説明]`」

**`credentials-create (crc) (資格情報名) [フォルダ] (--userName) [--userPassword] [--desc]`**

資格情報を作成します。

次のオプションを受け入れます。

`-u, [--userName= ユーザ名]` - リソースにアクセスするユーザ名。

`--userPassword= パスワード` - リソースにアクセスするために使用するパスワード。

`-d, --desc` - 資格情報の説明。

**`credentials-delete (crd) (資格情報パス)`**

資格情報を削除します。

**`credentials-folder-create (crf) (フォルダ名) [親フォルダ] [オプション]`**

資格情報を格納するための新しいフォルダを作成します。

次のオプションを受け入れます。

`--desc=[ 説明]` - フォルダの説明。

**`credentials-list (crl) [フォルダ]`**

資格情報を一覧表示します。

**`credentials-move (crm) (資格情報名) [宛先フォルダパス]`**

資格情報を移動します。

**`credentials-rename (crr) (資格情報パス) (新しい資格情報パス)`**

資格情報の名前を変更します。

**`credentials-update (cru) (-u | --userName= ユーザ名) [--userPassword= パスワード] [-d | --desc= 説明]`**

資格情報を作成します。

次のオプションを受け入れます。



- u, --user - リソースにアクセスするためのユーザ名です。
- password - リソースにアクセスするために使用するパスワードです。
- d, --desc - 資格情報の説明。

## データベースコマンド

組み込み PostgreSQL データベースを管理するコマンドです。データベースコマンドは、長い形式では database- から始まり、短い形式では d から始まります。

---

**注:** これらのコマンドを使用して、外部 PostgreSQL データベースを管理することはできません。

---

- ◆ 41 ページの「[database-get-credentials-audit \(dgca\)](#) および [database-get-credentials \(dgc\)](#)」
- ◆ 41 ページの「[database-backup \(db\) \(backup directory\)](#)」

### database-get-credentials-audit (dgca) および database-get-credentials (dgc)

組み込みの PostgreSQL 監査データベースおよび組み込みの PostgreSQL データベースに接続するために使用された資格情報を取得します。

### database-backup (db) (backup directory)

組み込みの PostgreSQL データベースをバックアップします (データベースファイルがバックアップされるネットワークの場所)。

(バックアップディレクトリ) - データベースファイルがバックアップされるデータベースサーバまたはネットワークの場所のローカルディレクトリ。親ディレクトリがすでに存在し、十分なディスク容量があるようにします。ディレクトリ内のどの既存のデータベースファイルも上書きされます。

## 展開コマンド

展開タスクを実行するコマンドです。展開コマンドは、長い形式では deployment- から始まり、短い形式では dp から始まります。

- ◆ 41 ページの「[deployment-task-abort \(dpta\)](#) (展開タスク名) [...]」
- ◆ 42 ページの「[deployment-task-create \(dptc\)](#) (展開タスク名) (ユーザの資格情報ファイル) [IP アドレス /DNS 名] [...] [オプション]」
- ◆ 42 ページの「[deployment-task-delete \(dptd\)](#) (展開タスク名) [...]」
- ◆ 42 ページの「[deployment-task-list \(dptl\)](#) [オプション]」
- ◆ 43 ページの「[deployment-task-run-now \(dptrn\)](#) (展開タスク名) [...]」

### deployment-task-abort (dpta) (展開タスク名) [...]

展開タスクを中断します。

**deployment-task-create (dptc) ( 展開タスク名 ) ( ユーザの資格情報ファイル ) [IP アドレス /DNS 名 ]  
[...][ オプション ]**

展開タスクを作成し、IP アドレスまたは DNS 名を使用して、ZENworks Agent をデバイスに展開します。

( ユーザの資格情報ファイル ) - デバイスに接続するための資格情報を含むファイルのパス。ファイルの各行には、ユーザ名とパスワードをスペースで区切って指定する必要があります。たとえば、「管理者のパスワード」のようになります。

次のオプションを受け入れます。

- f, --file=[IP アドレスファイル ][...] - ZENworks Agent の展開先デバイスの IP アドレスまたは DNS 名を含むファイルパスのカンマ区切りリスト。ファイルには、カンマ区切り値 (CSV) 形式でデバイスの IP アドレスまたは DNS 名のリストが含まれているか、各行に IP アドレスまたは DNS 名が 1 つあります。
- s, --schedule=[ 起動スケジュール XML ファイル ] - 起動スケジュールを含む XML ファイルです。スケジュール XML ファイルテンプレートについては、XML ファイルを参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/schedules、Windows サーバでは *Installation\_directory:\Novell\Zenworks\share\zman\samples\*schedules に保存されています。スケジュールファイルまたは --run-now オプションが指定されていない場合、タスクの実行はスケジュールされません。
- r, --run-now - 展開タスクを作成後ただちに実行するようにスケジュールします。
- b, --reboot=[ 再起動オプション ] - エージェントの展開後にデバイスを再起動するタイミングを指定します。有効な値は、immediate および manual です。
- d, --desc=[ 説明 ] - 展開タスクの説明。
- p, --proxy=[Windows プロキシ ] - / デバイスを基準にした、Windows デバイスの相対パス。管理エージェントを Linux プライマリサーバから Microsoft Windows デバイスに展開するには、Microsoft Windows を実行している管理対象デバイスが必要です。Windows プライマリサーバから展開タスクを実行する場合には、このオプションは無視されます。
- t, --timeout=[ タイムアウト ] - プライマリサーバが Windows プロキシからの応答を待機する時間を秒数で指定します。
- k, --key=[Registration key] - デバイスを登録するための登録キー。

**deployment-task-delete (dptd) ( 展開タスク名 ) [...]**

展開タスクを削除します。

**deployment-task-list (dptl) [ オプション ]**

展開タスクとステータスを一覧表示します。

次のオプションを受け入れます。

- n, --namefilter=[ フィルタ文字列 ] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。
- c, --count - 結果の数が表示されます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

## deployment-task-run-now (dptrn) ( 展開タスク名 ) [...]

指定された展開タスクをスケジュールして、ただちに実行します。タスクの資格情報がデータストアに保存されている必要があります。

## ディスカバリコマンド

これらのコマンドは、デバイスのディスカバリ関連情報を表示する場合に使用します。ディスカバリコマンドは、長い形式では discovery- から始まり、短い形式では d から始まります。

次のコマンドはすべて、「**グローバルオプション**」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- ◆ [43 ページの「discovery-view-discovered-devices \(dvdd\) \[オプション\]」](#)
- ◆ [43 ページの「discovery-task-list \(dtl\) \[オプション\]」](#)
- ◆ [44 ページの「discovery-task-run-now \(dtrn\) \( 検出タスク名 \)」](#)

### discovery-view-discovered-devices (dvdd) [オプション]

検出されたデバイスを一覧表示します。

次のオプションを受け入れます。

- n, --nameFilter=[ フィルタ文字列 ] - 指定したフィルタに当てはまるデバイスを表示します。引用符で囲む限り、ワイルドカード \* を使用できます。
- t, --typefilter=[ タイプフィルタ ] - 指定したタイプに当てはまるオプションを表示します。有効な値は、server、workstation、printer、network、thinclient、other、unknown、および deployable です。このオプションを指定していない場合、すべてのタイプのデバイスが表示されます。
- o, --osfilter=[ オペレーティングシステム ] - 指定した OS がインストールされているデバイスを表示します。有効な値は、other、win9x、winnt、wince、win2k、win2k3、winxp、nw6、nw6\_5、nwoes、suse、sles、nld、rh\_es、および rh\_as です。指定しない場合、すべてのデバイスが表示されます。
- m, --management-status=[ 管理ステータス ] - 指定したステータスを持つデバイスを表示します。有効な値は、discovered、inventoried、managed、および retired です。指定しない場合、ステータスにかかわらずすべてのタイプのデバイスが表示されます。
- modefilter=[discovery mode] - 特定のディスカバリモードを使用して検出されたデバイスを表示します。有効な値は、IP、LDAP、csvimport、ZENworks-migration、および ZAM-migration です。指定しない場合、すべてのデバイスが表示されます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

### discovery-task-list (dtl) [オプション]

ディスカバリタスクとステータスを一覧表示します。

次のオプションを受け入れます。

- n, --namefilter=[ フィルタ文字列 ] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

## discovery-task-run-now (dtrn) ( 検出タスク名 )

指定した検出タスクをただちに実行します。

## 動的グループコマンド

これらのコマンドは、動的グループ、グループメンバを表示したり、グループをリフレッシュするために使用されます。動的グループコマンドは、長い形式では dynamic-group- から始まり、短い形式では dg から始まります。

- ◆ 44 ページの「dynamic-group-list (dgl) ( デバイスタイプ ) [ フォルダ ] [-n|--namefilter= フィルタ文字列 ] [-s|--scrollsize= スクロールサイズ ] [-r|--recursive] [-c|--count]」
- ◆ 44 ページの「dynamic-group-members (dgm) ( デバイスタイプ ) (dynamic group name) [path of the dynamic group folder] [-s|--scrollsize=scroll size]」
- ◆ 45 ページの「dynamic-group-refresh (dgr) ( デバイスタイプ ) [( ダイナミックグループの名前 ) ( ダイナミックグループの名前 )... ( ダイナミックグループの名前 )] [-f|--folder= ダイナミックグループフォルダのパスを含むフォルダ名 ] [-a|--all]」
- ◆ 45 ページの「dynamic-group-view-query (dgvq) ( ダイナミックグループフォルダのパス )」

**dynamic-group-list (dgl) ( デバイスタイプ ) [ フォルダ ] [-n|--namefilter= フィルタ文字列 ] [-s|--scrollsize= スクロールサイズ ] [-r|--recursive] [-c|--count]**

ダイナミックグループを一覧表示します。

次のオプションを受け入れます。

( デバイスタイプ ) - 有効な値は server および workstation です。

[ フォルダ ] - 指定したフォルダのコンテンツを一覧表示します。指定しなかった場合には、ルートフォルダの内容が表示されます。

[-n|--namefilter= フィルタ文字列 ] - 指定した名前に一致する結果が表示されます。引用符で囲む限り、ワイルドカード \* および ? を使用できます。

[-s|--scrollsize= スクロールサイズ ] - 一度に表示される結果の数。

[-r|--recursive] - サブフォルダを含め、結果を再帰的に一覧表示します。指定しなかった場合は、サブフォルダの内容は一覧表示されません。

[-c|--count] - 結果の数を表示します。

**dynamic-group-members (dgm) ( デバイスタイプ ) (dynamic group name) [path of the dynamic group folder] [-s|--scrollsize=scroll size]**

指定したダイナミックグループのメンバーを一覧表示します。

次のオプションを受け入れます。

( デバイスタイプ ) - 有効な値は server および workstation です。

( ダイナミックグループの名前 ) - 動的グループの名前です。

[ ダイナミックグループフォルダのパス ] - ダイナミックグループのパス。完全なパス、またはルートフォルダの関係パスのいずれかを指定できます。  
[-s|--scrollsize= スクロールサイズ ] - 一度に表示される結果の数。

**dynamic-group-refresh (dgr) ( デバイスタイプ ) [ ( ダイナミックグループの名前 ) ( ダイナミックグループの名前 ) ... ( ダイナミックグループの名前 ) ] [-f|--folder= ダイナミックグループフォルダのパスを含むフォルダ名 ] [-a|--all]**

指定したダイナミックグループを更新します。

次のオプションを受け入れます。

( デバイスタイプ ) - 有効な値は server および workstation です。

( ダイナミックグループの名前 ) - 動的グループの名前です。複数のグループを一覧表示できます。

[-f|--folder= ダイナミックグループフォルダのパスを含むフォルダ名 ] - フルパスを含む、ダイナミックグループフォルダの名前。

[-a|--all] - 既存の動的グループをすべてリフレッシュします。

**dynamic-group-view-query (dgvq) ( ダイナミックグループフォルダのパス )**

指定したダイナミックグループのフィルタを一覧表示します。

( ダイナミックグループフォルダのパス ) - ダイナミックグループのパス完全なパスを指定することも、ルートフォルダを基準にした相対パスを指定することもできます。

## エンドポイントセキュリティ管理コマンド

これらのコマンドは、セキュリティポリシーのインポートとエクスポート、および管理ゾーンのセキュリティポリシー暗号化キー (KMK) のエクスポートに使用されます。これらのコマンドは、ZENworks Endpoint Security Management にのみ適用されます。

次のすべてのコマンドでは、**グローバルオプション**セクションに一覧されるオプションフラグを受け入れます。ただし、esmpolicy-export-to-file コマンドと esmpolicy-import コマンドは、リモートで実行できないので、--host オプションを受け入れません。

- [45 ページの「esmpolicy-export-to-file \(epetf\) \(ポリシーパス\) \(XMLポリシーファイルパス\)」](#)
- [46 ページの「esmpolicy-export-kmk-to-file \(epektf\) \(ポリシー暗号化キーのファイルパス\)」](#)
- [46 ページの「esmpolicy-import \(epi\) \(ポリシー名\) \(ポリシー暗号化キーのファイルパス\) \(XMLポリシーファイルパス\) \[親フォルダ\]」](#)
- [47 ページの「esmpolicy-purge-effective-policies \(epep\) \[\(デバイスパス\) \(デバイスパス\) ... \(デバイスパス\)\] \[-b|--begin-date=年-月-日 時:分:秒\] \[-e|end-date=年=月-日 時:分:秒\] \[-u|--unregisteredDevices\]」](#)

**esmpolicy-export-to-file (epetf) (ポリシーパス) (XMLポリシーファイルパス)**

セキュリティポリシーの情報を暗号化された XML ポリシーファイルにエクスポートします。XML ポリシーファイルは、同じ管理ゾーンまたは別の管理ゾーンへのポリシーのインポートに使用できます。

注: ポリシーをインポートするには、暗号化された XML ポリシーファイルを復号できるように、管理ゾーンのセキュリティポリシー暗号化キー (KMK) も提供する必要があります。キーファイルの作成には、`esmpolicy-export-kmk-to-file` コマンドを使用します。

(ポリシーパス) - Policies ルートフォルダを基準にしたポリシーオブジェクトの相対パス (ファイル名を含む)。たとえば、FWpolicy1 や ESMpolicies/DEpolicy4 のように指定します。

(XML ポリシーファイルパス) - XML ポリシーファイルを保存するパス (ファイル名を含む)。ファイル名だけを指定した場合、ファイルは現在のディレクトリに保存されます。たとえば、firewallpolicy.xml または c:\firewallpolicy.xml のように指定します。

例:

```
zman epetf FWPolicy1 c:\FWpolicy1.xml
```

```
zman epetf ESMpolicies/DEpolicy4 DEpolicy4.xml
```

#### **esmpolicy-export-kmk-to-file (epektf) (ポリシー暗号化キーのファイルパス)**

管理ゾーンのセキュリティポリシー暗号化キー (KMK) をファイルにエクスポートします。キーファイルは、`esmpolicy-import` コマンドによるポリシーのインポート時に、XML ポリシーファイル (ゾーン内のポリシーからエクスポートされたファイル) を復号化するために必要です。

(ポリシー暗号化キーのファイルパス) - セキュリティポリシー暗号化キー (KMK) ファイルを保存するパス (ファイル名を含む)。ファイル名だけを指定した場合、ファイルは現在のディレクトリに保存されます。ファイルには、サポートされている任意のファイル名を使用します。拡張子は重要ではありません。任意の拡張子を使用できます。拡張子を使用しなくても構いません。たとえば、KMK.txt、key.xml、KMK、および decryption.file は、すべて有効なファイル名です。

例:

```
zman epektf c:\key.txt
```

```
zman epektf EncryptionKey.xml
```

#### **esmpolicy-import(epi) (ポリシー名) (ポリシー暗号化キーのファイルパス) (XML ポリシーファイルパス) [親フォルダ]**

`esmpolicy-export-to-file` コマンドで作成された暗号化 XML ファイルからセキュリティポリシーをインポートします。

(ポリシー名) - ポリシーオブジェクトに割り当てる名前。

(ポリシー暗号化キーのファイルパス) - ポリシーのエクスポート元となった管理ゾーンのセキュリティポリシー暗号化キー (KMK) ファイルのフルパス (ファイル名を含む)。このファイルは、暗号化された XML ファイルを復号するために必要です。キーファイルが現在のディレクトリにある場合は、ファイル名だけを指定します。

(XML ポリシーファイルパス) - 暗号化された XML ポリシーファイルのフルパス (ファイル名を含む)。ファイルが現在のディレクトリにある場合は、ファイル名だけを指定します。

[親フォルダ] - ポリシーオブジェクトを作成する Policies フォルダ。ルートフォルダでオブジェクトを作成する場合は、このオプションを無視してください。

例：

```
zman epi FWPolicy c:\key.txt c:\FWpolicy.xml
```

```
zman epi DEPolicy key.txt encryptionpolicy.xml esmpolicies/encryption
```

**esmpolicy-purge-effective-policies (epep) [( デバイスパス )( デバイスパス )...( デバイスパス )] [-b|--begin-date= 年 - 月 - 日 時 : 分 : 秒 ] [-e|end-date= 年 = 月 - 日 時 : 分 : 秒 ] [-u|--unregisteredDevices]**

有効なポリシーレポートのレコードを ZENworks データベースからパージします。次のオプションを使用すると、特定のデバイスまたは特定の期間のレポートレコードをターゲットにすることができます。

[( デバイスパス )( デバイスパス )... ( デバイスパス )]: 特定デバイスの有効なポリシーレポートレコードをパージするには、デバイスごとにフルパスを指定します。すべてのデバイスのレポートをパージする場合は、このオプションを無視してください。

このコマンドは次のオプションを受け入れます。

[-b|--begin-date= 年 - 月 - 日 時 : 分 : 秒 ]: 特定の日付から開始する有効なポリシーレポートレコードをパージするには、開始日を指定します。開始日またはそれ以降のタイムスタンプを持つすべてのレコードがパージされます。特定の期間を指定するには、このオプションと end-date オプションを併用してください。

[-e|--end-date=yyyy=MM-dd HH:mm:ss]: 特定の日付までの有効なポリシーレポートレコードをパージするには、終了日を指定します。終了日またはそれ以前のタイムスタンプを持つすべてのレコードがパージされます。特定の期間を指定するには、このオプションと begin-date オプションを併用してください。

[-u|--unregisteredDevices]: もうゾーンには登録されていないものの、まだ ZENworks データベースにレポートデータのあるデバイスの有効なポリシーレポートレコードをパージします。

例：

```
zman epep /Devices/Workstations/device1
```

```
zman epep /Devices/Workstations/device1 -b "2010-10-10 10:10:10" -e "2010-12-31 24:00:00"
```

```
zman epep -u
```

## Full Disk Encryption コマンド

これらのコマンドは、ZENworks Full Disk Encryption にのみ適用されます。これらのコマンドは、次にリストされたパラメータに加え、「グローバルオプション」セクションにリストされたオプションフラグを受け入れます。

- ◆ [48 ページの「fdpolicy-purge-eri \(fpe\) \[\( デバイスパス \)\( デバイスパス \)...\( デバイスパス \)\] \[-b|--begin-date=yyyy-MM-dd HH:mm:ss\] \[-e|end-date=yyyy=MM-dd HH:mm:ss\] \[-u|--unregisteredDevices\]」](#)
- ◆ [48 ページの「fdpolicy-export-to-file \(fpetf\) \( ポリシーパス \) \(XML ポリシーファイルパス\)」](#)

- ◆ 49 ページの「fdeppolicy-import (fpi) (ポリシー名) (ポリシー暗号化キーのファイルパス) (XML ポリシーファイルパス) [親フォルダ]」
- ◆ 49 ページの「fdeppolicy-export-kmk-to-file (fpektf) (ポリシー暗号化キーのファイルパス)」

**fdeppolicy-purge-eri (fpe) [(デバイスパス) (デバイスパス)...(デバイスパス)] [-b|--begin-date=yyyy-MM-dd HH:mm:ss] [-e|--end-date=yyyy-MM-dd HH:mm:ss] [-u|--unregisteredDevices]**

ZENworks データベースから緊急回復情報 (ERI) レコードをパージします。次のオプションを使用すると、特定のデバイスまたは特定の期間のレコードをターゲットにすることができます。

[(デバイスパス) (デバイスパス)...(デバイスパス)]: 特定デバイスの ERI レコードをパージするには、デバイスごとにフルパスを指定します。すべてのデバイスのレコードをパージする場合は、このオプションを無視してください。

[-b|--begin-date=yyyy-MM-dd HH:mm:ss]: 特定の日付から開始する ERI レコードをパージするには、開始日を指定します。開始日またはそれ以降のタイムスタンプを持つすべてのレコードがパージされます。特定の期間を指定するには、このオプションと end-date オプションを併用してください。

[-e|--end-date=yyyy-MM-dd HH:mm:ss]: 特定の日付までの ERI レコードをパージするには、終了日を指定します。終了日またはそれ以前のタイムスタンプを持つすべてのレコードがパージされます。特定の期間を指定するには、このオプションと begin-date オプションを併用してください。

[-u|--unregisteredDevices]: もうゾーンには登録されていないものの、まだ ZENworks データベースに ERI レコードのあるデバイスの ERI レコードをパージします。

例:

```
zman fpe /Devices/Workstations/device1
```

```
zman fpe /Devices/Workstations/device1 -b "2010-10-10 10:10:10" -e "2010-12-31 24:00:00"
```

```
zman fpe -u
```

**fdeppolicy-export-to-file (fpetf) (ポリシーパス) (XML ポリシーファイルパス)**

完全ディスク暗号化ポリシーの情報を暗号化された XML ポリシーファイルにエクスポートします。XML ポリシーファイルは、同じ管理ゾーンまたは別の管理ゾーンへのポリシーのインポートに使用できます。

注: ポリシーをインポートするには、暗号化された XML ポリシーファイルを復号できるように、管理ゾーンの完全ディスク暗号化ポリシー暗号化キー (KMK) も提供する必要があります。キーファイルの作成には、fdeppolicy-export-kmk-to-file コマンドを使用します。

(ポリシーパス) - Policies ルートフォルダを基準にしたポリシーオブジェクトの相対パス (ファイル名を含む)。たとえば、FDEpolicies/FDEpolicy4 です。

(XML ポリシーファイルパス) - XML ポリシーファイルを保存するパス (ファイル名を含む)。ファイル名だけを指定した場合、ファイルは現在のディレクトリに保存されます。たとえば、FDEpolicy.xml です。

例:

```
zman fpetf FDEPolicy1 c:\FDEpolicy1.xml
```



## **fdepolicy-import (fpi) (ポリシー名) (ポリシー暗号化キーのファイルパス) (XML ポリシーファイルパス) [親フォルダ]**

fdepolicy-export-to-file コマンドで作成された暗号化 XML ファイルから完全ディスク暗号化ポリシーをインポートします。

(ポリシー名) - ポリシーオブジェクトに割り当てる名前。

(ポリシー暗号化キーのファイルパス) - ポリシーのエクスポート元となった管理ゾーンの完全ディスク暗号化ポリシー暗号化キー (KMK) ファイルのフルパス (ファイル名を含む)。このファイルは、暗号化された XML ファイルを復号するために必要です。キーファイルが現在のディレクトリにある場合は、ファイル名だけを指定します。

(XML ポリシーファイルパス) - 暗号化された XML ポリシーファイルのフルパス (ファイル名を含む)。ファイルが現在のディレクトリにある場合は、ファイル名だけを指定します。

[親フォルダ] - ポリシーオブジェクトを作成する Policies フォルダ。ルートフォルダでオブジェクトを作成する場合は、このオプションを無視してください。

例:

```
zman fpi FDEPolicy c:\key.txt c:\FDEpolicy.xml
```

## **fdepolicy-export-kmk-to-file (fpektf) (ポリシー暗号化キーのファイルパス)**

管理ゾーンの完全ディスク暗号化ポリシー暗号化キー (KMK) をファイルにエクスポートします。キーファイルは、fdepolicy-import コマンドによるポリシーのインポート時に、XML ポリシーファイル (ゾーン内のポリシーからエクスポートされたファイル) を復号化するために必要です。

(ポリシー暗号化キーのファイルパス) - 完全ディスク暗号化ポリシー暗号化キー (KMK) ファイルを保存するパス (ファイル名を含む)。ファイル名だけを指定した場合、ファイルは現在のディレクトリに保存されます。ファイルには、サポートされている任意のファイル名を使用します。拡張子は重要ではありません。任意の拡張子を使用できません。拡張子を使用しなくても構いません。たとえば、KMK.txt、key.xml、KMK、および decryption.file は、すべて有効なファイル名です。

例:

```
zman fpektf c:\key.txt
```

```
zman fpektf EncryptionKey.xml
```

## **ホットリストコマンド**

このコマンドは、未確認の警告またはエラーを持つデバイスのリストを表示する場合に使用します。Hotlist コマンドは、長い形式では hotlist- から始まり、短い形式では hl から始まります。

さらに、コマンドと併せて一覧表示されている個々のオプションも受け入れます。

- ◆ [49 ページの「hotlist-view \(hlv\) \[オプション\]」](#)

### **hotlist-view (hlv) [オプション]**

未確認の警告またはエラーを持つデバイスのリストを表示します。

次のオプションを受け入れます。

- c, --count - ホットリスト項目の数のみを表示します。
- t, --type=[ エラータイプ ] - ホットリスト項目のエラータイプでフィルタします。有効な値は、Noncompliant、Critical、および Warning です。
- o, --operator=[ 比較演算子 ] - 指定した場合、この演算子はエラーカウントベースのフィルタリングで使用されます。有効な値は、>、>=、=、<、および <= です。システムが > および < をリダイレクション演算子と誤認識しないよう、これらの演算子は引用符で囲んでください。
- e, --errorcount=[ エラー数 ] - 指定したエラータイプの数を基準にフィルタします。比較演算子を指定しなかった場合、> がデフォルトで使用されます。エラータイプが指定されていない場合、この数は不適合エラー、重大エラー、および警告エラーの合計とみなされます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。警告またはエラーは、messages-acknowledge コマンドを使用して承認できます。デバイスの警告またはエラーすべてがいったん未承認になると、メッセージはホットリストには今後は表示されなくなります。

例：5 個より多い重大なエラーを含むデバイスをホットリストに表示するには：

```
zman hlv --type critical --operator ">" --error-count 5
```

## Intel AMT コマンド

これらのコマンドは、Intel AMT デバイス上のプレプロビジョニング、プロビジョニング、および管理操作の実行に使用されます。Intel AMT コマンドは、長い形式では iamt- で始まり、短い形式では英字 i で始まります。

ZENworks でサポートされている iAMT バージョンは、3、4、5、および 6 です。

- ◆ 50 ページの 「iamt-root-certificate-hash (irch)」
- ◆ 50 ページの 「iamt-create-mgmtcert (icmc) [ オプション ]」
- ◆ 51 ページの 「iamt-create-csr (icc) (isCA) (タイプ) [ オプション ]」
- ◆ 51 ページの 「iamt-clear-mgmtcert (idmc)」
- ◆ 51 ページの 「iamt-cacert-import (icai) [isclear] [ オプション ]」

### iamt-root-certificate-hash (irch)

ZENworks ルート証明書ハッシュを印刷します。

### iamt-create-mgmtcert (icmc) [ オプション ]

iAMT プロビジョニングサーバまたは管理サーバとして使用されているすべてのプライマリサーバ上で、.DER 形式の Intel AMT プロビジョニング証明書または Intel AMT 管理証明書を作成またはインポートします。

外部証明書をインポートする場合は、次のオプションを受け入れてください。

- p, --certpath - 外部 CA によって署名された Intel AMT プロビジョニング / 管理証明書のパスを指定します。証明書は、.DER 形式にする必要があります。

たとえば、外部証明書をインポートするには、次のコマンドを実行します。

```
zman icmc --certpath C:\Certificate.DER
```

#### **iamt-create-csr (icc) (isCA) (タイプ) [オプション]**

Intel AMT プロビジョニング / 管理サーバまたは外部 CA 環境内の AMT デバイスに対する証明書要求を作成します。

isCA - 有効な値は、True または False です。True の場合、証明書要求は、Intel AMT 管理ゾーンに証明書を生成します。

Type - 有効な値は、mgmt または device です。管理ゾーンまたは管理ゾーン内の特定デバイスに対する証明書要求が作成されます。

次のオプションを受け入れます。

-c, --country=[ 国名 ] - 国の名前を指定します。

-s, --state=[ 州名 ] - 州の名前を指定します。

-o, --organization=[ 組織名 ] - 組織の名前を指定します。

-u, --orgunit=[ 組織単位 ] - 組織の単位を指定します。

-n, --commonname=[ 共通名 ] - 証明書の共通名を指定します。

-d, --destination-folder=[ 宛先フォルダ ]: 証明書署名要求ファイルのコピー先フォルダのパスを指定します。

#### **iamt-clear-mgmtcert (idmc)**

Intel AMT プロビジョニング / 管理証明書をクリアします。

#### **iamt-cacert-import (icai) [isclear] [オプション]**

外部 CA またはその従属 CA をサーバ上でインポートします。

[isclear] - 有効な値は True または False です。true の場合は、インポートした外部 CA またはその従属 CA が削除されます。

外部 CA またはその従属 CA をインポートする場合は、次のオプションを受け入れません。

-i, --inform - 証明書フォーマットが PEM または DER のいずれであるか指定します。

-p, --cacertpath - 証明書パスを指定します。

## イメージングコマンド

このコマンドは、イメージング操作を実行するために使用されます。イメージングコマンドは、長い形式では `imaging-` から始まり、短い形式では `i` から始まります。

- ◆ 52 ページの「`imaging-apply-assigned-imaging-bundle (iaaib)` ( デバイスタイプ ) ( デバイス名 )」

### `imaging-apply-assigned-imaging-bundle (iaaib)` ( デバイスタイプ ) ( デバイス名 )

選択したデバイスに直接割り当てられている最新のプレブートバンドルを適用します。使用可能な直接割り当てがない場合、このコマンドでは、選択したデバイスの継承割り当てに属する最初のプレブートバンドルが適用されます。バンドルは、デバイスが次にプレブート作業を確認するときに適用されます。

次のオプションを受け入れます。

( デバイスタイプ ) - 有効な値は `server` および `workstation` です。

( デバイス名 ) - サーバまたはワークステーションオブジェクトの名前。

たとえば、割り当てられているイメージングバンドルを `ネームサーバ1` を使用してサーバに適用するには、次のコマンドを実行します。

```
zman iaaib server server1
```

デバイスオブジェクトがフォルダ内にある場合は、次のコマンドを使用します。

```
iaaib ( デバイスタイプ ) ( デバイスのフォルダ / パス ) ( デバイス名 )
```

ここで `folder/path of device` は、Workstation または Server フォルダ内のデバイスの相対パス。

ディレクトリ内に複数のワークステーションがあり、ワークステーション名の前にアルファベット数字 (`dev091`、`dev092`、...`dev099` など) が付いている場合は、Linux サーバから次のコマンドを使用して、割り当てられたイメージングバンドルを複数のデバイスに同時に適用します。

```
zman iaaib workstation folderx/dev{09{1,2,3,4,5,6,7,9}}
```

バンドルはデバイス `dev091`、`dev092`、などに適用されます。

## インベントリコマンド

スキャンを初期化して、管理対象デバイスにインベントリデータコレクションフォームを送信するために使用するコマンドです。インベントリコマンドは、長い形式では `inventory-` から始まり、短い形式では `i` から始まります。

- ◆ 53 ページの「`inventory-collection-wizard-run-now (icwrn)` ( デバイスオブジェクトパス ) [ ... ] [ オプション ]」
- ◆ 53 ページの「`(inventory-scan-now|isn)` ( < デバイスオブジェクトパス > < デバイスオブジェクトパス > ... < デバイスオブジェクトパス > ) [ オプション ]」
- ◆ 53 ページの「`inventory-export-localproducts|ielp` ( パス名 )」
- ◆ 53 ページの「`inventory-import-localproducts|iilp` ( パス名 ) [ `dokbmerge` ]」

### **inventory-collection-wizard-run-now (icwrn) ( デバイスオブジェクトパス ) [ ... ] [ オプション ]**

インベントリデータコレクションフォームを 1 つ以上のデバイスに送信します。

次のオプションを受け入れます。

-f, --folder=[ デバイスフォルダ ] - / デバイスを基準にした、デバイスフォルダの相対パス。このオプションを指定すると、デバイスオブジェクトのパスは、このフォルダを基準にした相対パスとして決定されます。これは、同じフォルダで複数のデバイスオブジェクトを指定するのに使用できます。

### **(inventory-scan-now|isn) (< デバイスオブジェクトパス > < デバイスオブジェクトパス > ... < デバイスオブジェクトパス > ) [ オプション ]**

1 つ以上のデバイスのインベントリスキャンを実行します。

次のオプションを受け入れます。

-f, --folder=[ デバイスフォルダ ] - / デバイスを基準にした、デバイスフォルダの相対パス。このオプションを指定すると、デバイスオブジェクトのパスは、このフォルダを基準にした相対パスとして決定されます。これは、同じフォルダで複数のデバイスオブジェクトを指定するのに使用できます。

### **inventory-export-localproducts|ielp ( パス名 )**

管理者が定義したローカル製品と製品の指紋をエクスポートします。ローカル製品をエクスポートできるのは CDLP 権限を持つ管理者だけです。

製品データをエクスポートするには、次のように指定します。

zman ielp { パス名 }。

例 :

管理者が定義したすべてのローカル製品を、ZENworks Configuration Management から ./output/lpexports.txt のファイルにエクスポートする例を示します。

zman ielp ./output/lpexports.txt.

### **inventory-import-localproducts|iilp ( パス名 ) [ dokbmerge ]**

管理者が定義したローカル製品と製品の指紋をインポートします。ローカル製品をインポートできるのは CDLP 権限を持つ管理者だけです。

dokbmerge - ローカル製品のインポート後に KB マージをトリガします。

製品データをインポートするには、次のように指定します。

zman iilp { パス名 } [-U ( 管理者 )] [-P ( パスワード )]。

例 :

管理者が定義したローカル製品を、./output/lpimports.txt のファイルから ZENworks Configuration Management にインポートする例を示します。

zman iilp ./output/lpimports.txt -U Administrator -P novell.

## ライセンスコマンド

これらのコマンドは、ZENworks サーバを有効化したり、ライセンス情報を表示したりする場合に使用します。ライセンスコマンドは、長い形式では `license-` から始まり、短い形式では `l` から始まります。

次のコマンドはすべて、「**グローバルオプション**」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- ◆ 54 ページの「`license-activate (la) (製品名) (ライセンスキー)`」
- ◆ 54 ページの「`license-deactivate (ld) (製品名)`」
- ◆ 54 ページの「`license-info (li)`」
- ◆ 54 ページの「`license-refresh (lr)`」
- ◆ 54 ページの「`license-show-active-components (lsac)`」

### **license-activate (la) (製品名) (ライセンスキー)**

ZENworks 製品 (Asset Inventory for Linux、ZENworks Configuration Management、ZENworks Asset Management、ZENworks Full Disk Encryption、ZENworks Endpoint Security Management) または ZENworks Suite を有効にします。

### **license-deactivate (ld) (製品名)**

ZENworks 製品を非アクティブ化します。引数として製品名だけが必要です。

### **license-info (li)**

ライセンス情報を表示します。

### **license-refresh (lr)**

通常 1 日 1 回更新される、保存されているライセンスキャッシュをシステムに更新させます。このコマンドを使用すると、評価版ライセンスの期限が切れ、キャッシュに反映されていない場合など、いつでもキャッシュを更新できます。

### **license-show-active-components (lsac)**

既知のすべての ZENworks Configuration Management コンポーネントと DataModel プラグインの現在のライセンスステータスを一覧にします (2 つのそれぞれのリスト)。

## 場所コマンド

場所の表示と管理を行うコマンドです。場所は、1つまたは複数のネットワーク環境から構成されます。ZENworks Agent は、その現在の環境が、場所に関連付けられたネットワーク環境に一致していると判断すると、場所に割り当てられたすべてのセキュリティポリシーおよび最近接サーバールールを適用します。

場所コマンドは、長い形式では `location-` から始まり、短い形式では `l` から始まります。次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- 55 ページの「`location-assign-device-folder (ladf)` (デバイスフォルダまたはデバイスオブジェクトパス) [`<場所オブジェクト> <場所オブジェクト>...<場所オブジェクト>`] [`-a|--allLocations`]
- 56 ページの「`location-audit-upload (lau)` (場所名) (状態)」
- 56 ページの「`location-create (lc)` (場所名) [`-d|--desc=<説明>`] [ネットワーク環境名] [...]
- 56 ページの「`location-delete (ldel)` (場所名) [...]
- 56 ページの「`location-delete-all-network-assignments (ldana)` (場所名)」
- 56 ページの「`location-delete-network-assignments (ldna)` (場所名) [ネットワーク環境名] [...]
- 57 ページの「`location-list (ll)` [オプション]
- 57 ページの「`location-list-network-assignments (llna)` (場所名)」
- 57 ページの「`location-list-relationships (llrel)` (場所名) [`-c` カウント]
- 57 ページの「`location-network-assign (lna)` (場所名) (ネットワーク環境名) [...]
- 57 ページの「`location-rename (lrn)` (古い名前意) (新しい名前)」
- 57 ページの「`location-response-recompute|lrr>` [`<デバイスフォルダまたはデバイスオブジェクトパス> <デバイスフォルダまたはデバイスオブジェクトパス>...<デバイスフォルダまたはデバイスオブジェクトパス>`] [`-f|--forceRecompute`]
- 58 ページの「`location-throttle (lt)` (場所名) (スロットルレート)」
- 58 ページの「`location-unknown-throttle (lut)` (スロットルレート)」

**`location-assign-device-folder (ladf)` (デバイスフォルダまたはデバイスオブジェクトパス) [`<場所オブジェクト> <場所オブジェクト>...<場所オブジェクト>`] [`-a|--allLocations`]**

特定のデバイスフォルダまたはデバイスオブジェクトに場所を割り当てます。

(デバイスフォルダまたはデバイスオブジェクトパス) - デバイスフォルダまたはデバイスオブジェクトの完全なパス。

[場所オブジェクト] [...] - 場所オブジェクトの名前。

`-a, --allLocations` - このパラメータを使用して、デバイスフォルダまたはデバイスオブジェクトにすべての場所およびネットワーク環境を割り当てます。

### **location-audit-upload (lau) (場所名) (状態)**

指定された場所への監査データのアップロードを設定します。

(場所名) - 場所の名前です。

(状態) - 次の状態から選択します。

- 1 を選択すると、関連付けられた場所にあるデバイスから、任意のサテライトサーバ (コレクションの役割) またはプライマリサーバに監査イベントデータをアップロードできるようになります。
- 2 を選択すると、関連付けられた場所にあるデバイスからは、どのサテライトサーバ (コレクションの役割) やプライマリサーバにも監査イベントデータをアップロードできなくなります。

### **location-create (lc) (場所名) [-d|--desc=<説明>] [ネットワーク環境名] [...]**

場所を作成し、(オプションで) 場所にネットワーク環境を割り当てます。

(場所名) - 新しい場所に割り当てる名前。

[ネットワーク環境名] - 場所に割り当てる既存のネットワーク環境の名前。複数のネットワーク環境を追加できます。場所の作成中のネットワーク環境の追加はオプションです。場所の作成後に、location-nwenv-assign コマンドを使用して、ネットワーク環境を追加することができます。

[ネットワーク環境名] - 場所に割り当てる既存のネットワーク環境の名前。複数のネットワーク環境を追加できます。場所の作成中のネットワーク環境の追加はオプションです。場所の作成後に、location-nwenv-assign コマンドを使用して、ネットワーク環境を追加することができます。

次のオプションを受け入れます。

-d |--desc= 説明 : 場所の説明を指定します。

-u |--uploadThrottle= スロットルレートのアップロードアップロードスロットルレートを指定します。

-a |--auditUpload= アップロードデータの監査 監査アップロードデータを有効にするには 0 を、無効にするには 1 を指定します。

### **location-delete (ldel) (場所名) [...]**

場所を削除します。

(場所名) - 削除する場所の名前。1 つまたは複数の場所名を指定できます。

### **location-delete-all-network-assignments (ldana) (場所名)**

場所からすべてのネットワーク環境を削除します。

(場所名) - 割り当てられたすべてのネットワーク環境を削除する場所の名前。

### **location-delete-network-assignments (ldna) (場所名) [ネットワーク環境名] [...]**

指定したネットワーク環境を場所から削除します。

(場所名) - ネットワーク環境を削除する場所の名前。

[ネットワーク環境名] - 削除するネットワーク環境の名前。複数のネットワーク環境を削除することができます。



### **location-list (ll) [ オプション ]**

場所を一覧表示します。

次のオプションを受け入れます。

-c, --count - 結果の数が表示されます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

### **location-list-network-assignments (llna) ( 場所名 )**

場所に割り当てられたネットワーク環境を一覧表示します。

( 場所名 ) - ネットワーク環境を表示する場所の名前。

次のオプションを受け入れます。

-c, --count - 結果の数が表示されます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

### **location-list-relationships (llrel) ( 場所名 ) [-c カウント]**

指定した場所に関連付けられたポリシーやバンドルなどの ZENworks オブジェクトを一覧表示します。

( 場所名 ) - ZENworks オブジェクトを表示する場所の名前。

次のオプションを受け入れます。

-c, --count - 結果の数が表示されます。

### **location-network-assign (lna) ( 場所名 ) ( ネットワーク環境名 ) [...]**

場所にネットワーク環境を追加します。

( 場所名 ) - ネットワーク環境を割り当てる場所の名前。

[ ネットワーク環境名 ] - 場所に追加する既存のネットワーク環境の名前。複数のネットワーク環境を追加できます。

### **location-rename (lrn) ( 古い名前意 ) ( 新しい名前 )**

場所の名前を変更します。

( 古い名前 ) - 名前を変更する場所の名前。

( 新しい名前 ) - 場所の新しい名前。

### **location-response-recompute|lrr> [< デバイスフォルダまたはデバイスオブジェクトパス >> デバイスフォルダまたはデバイスオブジェクトパス >...< デバイスフォルダまたはデバイスオブジェクトパス >] [-f|--forceRecompute]**

要求元デバイスが最新の変更を取得できるように、サーバに対して場所設定の応答の再計算を要求します。

[ デバイスフォルダまたはデバイスオブジェクトパス ][...]: デバイスフォルダまたはデバイスオブジェクトのパスを指定します。

-f, --forceRecompute: 場所設定のすべての応答を強制的に再計算します。

### location-throttle (lt) (場所名) (スロットルレート)

指定した場所で認識されたデバイスに、コンテンツを配布する際に使用する帯域幅スロットルレートを設定します。ZENworks サーバおよびネットワークシステムのパフォーマンスを最大化するために、高帯域幅環境と低帯域幅環境で別のスロットルレートセットを使用できます。スロットルレートは、優先順序の高いパッチおよびバンドルコンテンツがすばやく展開できるようにバンドルで上書きすることができます。

(場所名) - 帯域スロットルレートを定義する場所の名前。

(スロットルレート) - 毎秒あたりのキロビット (kbps) 数での最大帯域幅レート。

### location-unknown-throttle (lut) (スロットルレート)

未知の場所で認識されたデバイスに、コンテンツを配布する際に使用する帯域幅スロットルレートを設定します。

(スロットルレート) - 毎秒あたりのキロビット (kbps) 数での最大帯域幅レート。

## メッセージコマンド

メッセージの表示と管理を行うコマンドです。メッセージコマンドは、長い形式では message- から始まり、短い形式では m から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [58 ページ](#)の「(messages-acknowledge|ma) [オブジェクトパス] [オプション]」
- [59 ページ](#)の「(messages-delete|md) [オブジェクトパス] [オプション]」
- [59 ページ](#)の「messages-view (mv) [オプション] [オブジェクトパス]」

### (messages-acknowledge|ma) [オブジェクトパス] [オプション]

ZENworks 管理対象デバイス、バンドル、またはポリシーに関連するメッセージを確認します。

[オブジェクトパス] - メッセージが関連しているオブジェクトのフルパス。オブジェクトは、サーバ、ワークステーション、バンドル、またはポリシーのいずれかになります。

次のオプションを受け入れます。

- b, --begin-date=[メッセージがログされた日付] - この日付以降にログされたメッセージを確認します。日付の形式は、YYYY-MM-DD です。
- e, --end-date=[メッセージがログされた日付] - この日付以前にログされたメッセージを確認します。日付の形式は、YYYY-MM-DD です。
- a, --all - オブジェクトに対してログ記録されたメッセージをすべて確認します。オブジェクトが指定されていない場合、ログ記録されたすべてのメッセージが確認されます。

- l, --logID=[ ログ ID][...] - 確認するメッセージのログ ID のカンマ区切りリスト。ログ ID は、messages-view コマンドを使用して取得できます。
- L, --logID-file=[ ファイルパス ] - メッセージログ ID を含むファイルのパス。ファイルの各行には 1 つのメッセージログ ID が含まれている必要があります。ログ ID を入力するには、このオプションを --log-ID オプションの代わりに使用します。

#### **(messages-delete|md) [ オブジェクトパス ][ オプション ]**

ZENworks 管理デバイスに関連するメッセージを削除します。

[ オブジェクトパス ] - メッセージが関連しているオブジェクトのフルパス。オブジェクトは、サーバ、ワークステーション、バンドル、またはポリシーのいずれかになります。

次のオプションを受け入れます。

- b, --begin-date=[ メッセージがログされた日付 ] - この日付以降にログされたメッセージを削除します。日付の形式は、YYYY-MM-DD です。
- e, --end-date=[ メッセージがログされた日付 ] - この日付以前にログされたメッセージを削除します。日付の形式は、YYYY-MM-DD です。
- l, --logID=[ ログ ID][...] - 削除するメッセージのログ ID のカンマ区切りリスト。ログ ID は、messages-view コマンドを使用して取得できます。
- L, --logID-file=[ ファイルパス ] - メッセージログ ID を含むファイルのパス。ファイルの各行には 1 つのメッセージログ ID が含まれている必要があります。ログ ID を入力するには、このオプションを --log-ID オプションの代わりに使用します。

#### **messages-view (mv) [ オプション ][ オブジェクトパス ]**

ZENworks 管理デバイス、バンドル、またはポリシーに関連するメッセージのリストを表示します。

- S, --severity=[ 重大度 [...] ] - メッセージの重大度でフィルタします。有効な値は、critical、warning、および info です。
- t, --type=[ メッセージタイプ ] - メッセージのソースでフィルタします。有効な値は、server および client です。server でフィルタすると、ZENworks サーバによって生成されたメッセージのリストが表示されます。client でフィルタすると、そのデバイス上の ZENworks Agent によって生成されたメッセージのリストが表示されます。
- D, --date-condition=[ 日付条件 ] - 日付範囲に基づいてメッセージをフィルタします。有効な値は、before、since、および between です。日付を指定するには --begin-date オプションおよび --end-date オプションを指定します。日付条件が使用され、日付が指定されていない場合は、現在の日付がデフォルトで使用されます。
- b, --begin-date=[ メッセージがログされた日付 ] - メッセージがログされた日付。日付の形式は、YYYY-MM-DD です。このオプションは日付条件オプションとともに使用する必要があります。日付条件が指定されていない場合は、この日付以前のメッセージが表示されます。
- e, --end-date=[ メッセージがログされた日付 ] - メッセージがログされた日付。日付の形式は、YYYY-MM-DD です。日付条件オプションの値として between を指定する場合は、このオプションを指定する必要があります。
- a, --acknowledged - 確認されたメッセージでフィルタします。
- n, --not-acknowledged - 確認されていないメッセージでフィルタします。

- m, --messagefilter=[ ローカライズメッセージ ] - ローカライズされたメッセージでフィルタします。
- c, --count - メッセージ数のみ表示します。
- o, --sort-order=[ ソート順序 ] - 日付、重大度、および承認に基づいてメッセージをソートします。
- A, --asc - ソート済みリストのソート方向を指定します。デフォルトは降順です。
- d, --detailed - メッセージに関する詳細を返します。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

## ネットワーク環境コマンド

ネットワーク環境の表示と管理を行うコマンドです。ネットワーク環境は、環境を特定するサービス(ゲートウェイ、DNS サーバ、DHCP サーバなど)から構成されます。

ネットワーク環境は場所の基本要素です。ZENworks Agent は、その現在の環境が、場所に割り当てられたネットワーク環境に一致していると判断すると、場所に割り当てられたすべてのセキュリティポリシーおよび最近接サーバルールを適用します。

ネットワーク環境コマンドは、長い形式では `network-environment-` から始まり、短い形式では `ne` から始まります。次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [60 ページ](#)の「`network-environment-audit-upload (neau)` (ネットワーク環境名) (状態)」
- [61 ページ](#)の「`network-environment-create (nec)` (ネットワーク環境名) [-d|-desc=<説明>] [ネットワーク環境 XML ファイル]」
- [61 ページ](#)の「`network-environment-delete (ned)` (ネットワーク環境名) [...]」
- [61 ページ](#)の「`network-environment-export-to-file (neetf)` (ネットワーク環境名) (XML ファイルパス)」
- [61 ページ](#)の「`network-environment-list (nel)` [オプション]」
- [61 ページ](#)の「`network-environment-list-relationships (nelr)` (ネットワーク環境名) [-cr カウント]」
- [62 ページ](#)の「`network-environment-rename (ner)` (古い名前) (新しい名前)」
- [62 ページ](#)の「`network-environment-throttle (net)` (ネットワーク環境名) (スロットルレート)」

### `network-environment-audit-upload (neau)` (ネットワーク環境名) (状態)

指定されたネットワーク環境の監査データのアップロード設定を設定します。

(ネットワーク環境名) - ネットワーク環境の名前。

(状態) - 次の状態から選択します。

- 0 を選択すると、デバイスに関連付けられた場所に対して設定されている監査データのアップロード設定に戻すことができます。

- 1 を選択すると、関連付けられたネットワーク環境にあるデバイスから、任意のサテライトサーバ (コレクションの役割) またはプライマリサーバに監査イベントデータをアップロードできるようになります。
- 2 を選択すると、関連付けられたネットワーク環境にあるデバイスからは、どのサテライトサーバ (コレクションの役割) やプライマリサーバにも監査イベントデータをアップロードできなくなります。

**network-environment-create (nec) ( ネットワーク環境名 ) [-d] --desc=< 説明 > [ ネットワーク環境 XML ファイル ]**

指定した XML ファイルに含まれている情報からネットワーク環境を作成します。

( ネットワーク環境名 ) - ネットワーク環境に割り当てる名前。

-d, --desc=< 説明 > - ネットワーク環境の説明。

[ ネットワーク環境 XML ファイル ] - ネットワーク環境の作成に必要な情報が含まれた XML ファイルのパス名。XML ファイルを持っていない場合は、network-environment-export-to-file コマンドを使用して既存のネットワーク環境の情報をエクスポートします。その後、XML ファイルを新しいネットワーク環境ファイル作成用のテンプレートとして使用できます。

**network-environment-delete (ned) ( ネットワーク環境名 ) [...]**

ネットワーク環境を削除します。

( ネットワーク環境名 ) - 削除するネットワーク環境の名前。1 つまたは複数の名前を指定できます。

**network-environment-export-to-file (neetf) ( ネットワーク環境名 ) (XML ファイルパス)**

ネットワーク環境の情報を XML 形式のファイルにエクスポートします。

network-environment-create コマンドでは、ネットワーク環境は XML 形式のファイルで入力する必要があります。このファイルをテンプレートとして使用し、必要に応じて、新しいネットワーク環境を作成するときに使用する情報を含むように変更できます。

( ネットワーク環境名 ) - ファイルに情報をエクスポートするネットワーク環境の名前。

(XML ファイルパス) - エクスポートファイルのファイルパスおよび名前。

**network-environment-list (nel) [ オプション ]**

ネットワーク環境を一覧表示します。

次のオプションを受け入れます。

-c, --count - 結果の数が表示されます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示される結果の数。

**network-environment-list-relationships (nelr) ( ネットワーク環境名 ) [-cr カウント]**

指定したネットワーク環境に関連付けられたポリシーやバンドルなどの ZENworks オブジェクトを一覧表示します。

( ネットワーク環境名 ) - ZENworks オブジェクトを表示するネットワーク環境の名前。

次のオプションを受け入れます。

-c, --count - 結果の数が表示されます。

### network-environment-rename (ner) (古い名前)(新しい名前)

ネットワーク環境の名前を変更します。

(古い名前) - 名前を変更するネットワーク環境の名前。

(新しい名前) - ネットワーク環境の新しい名前。

### network-environment-throttle (net) (ネットワーク環境名)(スロットルレート)

特定のネットワーク環境にあるデバイスに、コンテンツを配布する際の帯域幅スロットルレートを設定します。ZENworks サーバおよびネットワークシステムのパフォーマンスを最大化するために、高帯域幅環境と低帯域幅環境で別のスロットルレートセットを使用できます。スロットルレートは、優先順序の高いパッチおよびバンドルコンテンツがすばやく展開できるようにバンドルで上書きすることができます。

(ネットワーク環境名) - 帯域幅スロットルレートを定義するネットワーク環境の名前。

(スロットルレート) - 毎秒あたりのキロビット (kbps) 数での最大帯域幅レート。

## Micro Focus Customer Center 登録コマンド

このコマンドを使用して、プライマリサーバを Micro Focus Customer Center に登録します。

Micro Focus Customer Center 登録コマンドは、長い形式では `ncc-register` から始まり、短い形式では `nccreg` から始まります。このコマンドは、「[グローバルオプション](#)」のセクションに一覧表示されるフラグを受け入れます。さらに、コマンドと併せて一覧表示されている個々のオプションも受け入れます。

- ◆ [62 ページの](#) 「`(ncc-register|nccreg) (abc@xyz.com) (登録キー) [プライマリサーバのパス] [-i|--nccConfigXML=ncc-config.xml] [-r|--reRegister]`」

`(ncc-register|nccreg) (abc@xyz.com) (登録キー) [プライマリサーバのパス] [-i|--nccConfigXML=ncc-config.xml] [-r|--reRegister]`

プライマリサーバを Micro Focus Customer Center に登録します。

`(abc@xyz.com)` - プライマリサーバを Micro Focus Customer Center のサーバに登録するための電子メールアドレス。

`(登録キー)` - プライマリサーバを Micro Focus Customer Center のサーバに登録するための登録キー。

`[プライマリサーバのパス]` - Micro Focus Customer Center のサーバに登録するためのプライマリサーバのパス。

`-i, --nccConfigXML= ncc-config.xml` - 製品、バージョン、アーキテクチャ、およびリリースなどの登録情報が含まれた XML ファイル。

`-r, --reRegister` - プライマリサーバを Micro Focus Customer Center のサーバに再登録します。

例:

ローカルプライマリサーバを Micro Focus Customer Center のサーバに登録するには、次のコマンドを実行します。

```
zman nccreg email reg-key
```

特定のプライマリサーバを Micro Focus Customer Center のサーバに登録するには、次のコマンドを実行します。

```
zman nccreg email reg-key server_path
```

ローカルプライマリサーバを Micro Focus Customer Center のサーバに再登録するには、次のコマンドを実行します。

```
zman nccreg email reg-key -r
```

特定のプライマリサーバを Micro Focus Customer Center のサーバに再登録するには、次のコマンドを実行します。

```
zman nccreg email reg-key server_path -r
```

## モバイルデバイスコマンド

これらのコマンドは、モバイルデバイスの管理に使用されます。モバイルデバイスコマンドは、長い形式では mobile- から始まり、短い形式では m から始まります。

- ◆ 64 ページの「mobile-folder-create (mfc) (モバイルオブジェクトパス) [...] [オプション]」
- ◆ 64 ページの「mobile-group-create (mgc) (グループ名) [親フォルダ] [オプション]」
- ◆ 64 ページの「mobile-move (mmv) (モバイルオブジェクトのパス) [宛先フォルダパス]」
- ◆ 64 ページの「mobile-rename (mr) (モバイルオブジェクトのパス) (新しい名前)」
- ◆ 64 ページの「mobile-delete (mdl) (モバイルオブジェクトのパス) [...] [オプション]」
- ◆ 64 ページの「mobile-group-add (mga) (モバイルグループのパス) (モバイルパス) [...] [オプション]」
- ◆ 65 ページの「mobile-list (ml) [フォルダ] [オプション]」
- ◆ 65 ページの「mobile-group-members (mgm) (モバイルグループパス) [オプション]」
- ◆ 65 ページの「mobile-group-remove (mgr) (モバイルグループのパス) (モバイルパス) [...] [オプション]」
- ◆ 65 ページの「mobile-list-groups (mlg) (モバイルパス) [オプション]」
- ◆ 65 ページの「mobile-add-policy (マップ) (モバイルオブジェクトのパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- ◆ 66 ページの「mobile-remove-policy (mrp) (モバイルオブジェクトのパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- ◆ 66 ページの「mobile-list-policies (mlp) (モバイルオブジェクトのパス) [オプション]」
- ◆ 66 ページの「mobile-reorder-policies (mrop) (モバイルオブジェクトのパス) (現在の位置) (新しい位置)」

- ◆ 66 ページの「mobile-view-all-policies-status (mvaps) (モバイルパス) [オプション]」
- ◆ 66 ページの「mobile-set-as-test (msat) (モバイルオブジェクトのパス)」
- ◆ 67 ページの「mobile-set-as-nontest (msan) (モバイルオブジェクトのパス)」

#### **mobile-folder-create (mfc) (モバイルオブジェクトパス) [...] [オプション]**

モバイルデバイスの新しいフォルダを作成します。

次のオプションを受け入れます。

--desc=[説明] - フォルダの説明。

#### **mobile-group-create (mgc) (グループ名) [親フォルダ] [オプション]**

モバイルグループを作成して、メンバーを追加します。

次のオプションを受け入れます。

--desc=[説明] - グループの説明。

-m, --members=[モバイルパス] [...] - /Devices/Mobile Devices へのモバイルデバイスの相対パスです。

-f, --folder=[モバイルフォルダ] - /Devices/Mobile Devices へのモバイルデバイスフォルダの相対パスです。このオプションを指定すると、モバイルデバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のモバイルデバイスオブジェクトを指定するのに使用できます。

#### **mobile-move (mmv) (モバイルオブジェクトのパス) [宛先フォルダパス]**

モバイルデバイスオブジェクトを別のフォルダに移動します。

#### **mobile-rename (mr) (モバイルオブジェクトのパス) (新しい名前)**

モバイルデバイスオブジェクトを名前変更します。

#### **mobile-delete (mdl) (モバイルオブジェクトのパス) [...] [オプション]**

1 つ以上のモバイルデバイスオブジェクトを削除します。

(モバイルオブジェクトのパス) [...] - /Devices/Mobile Devices へのモバイルデバイスオブジェクト (モバイルデバイス、モバイルデバイスフォルダ、またはモバイルデバイスグループ) への相対パスです。ワイルドカード \* は、引用符で囲まれている場合には、オブジェクト名に使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

次のオプションを受け入れます。

-r, --recursive - フォルダ内のオブジェクトを再帰的に削除します。

-f, --folder=[モバイルフォルダ] - /Devices/Mobile Devices へのモバイルデバイスフォルダの相対パスです。このオプションを指定すると、モバイルデバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のモバイルデバイスオブジェクトを指定するのに使用できます。

#### **mobile-group-add (mga) (モバイルグループのパス) (モバイルパス) [...] [オプション]**

モバイルデバイスをモバイルデバイスグループに追加します。



次のオプションを受け入れます。

-f, --folder=[ モバイルフォルダ ] - /Devices/Mobile Devices へのモバイルデバイスフォルダの相対パスです。このオプションを指定すると、モバイルデバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のモバイルデバイスオブジェクトを指定するのに使用できます。

#### **mobile-list (ml) [ フォルダ ] [ オプション ]**

モバイルデバイスオブジェクトを一覧表示します。

次のオプションを受け入れます。

-r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。

-n, --namefilter=[ フィルタ文字列 ] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。

-t, --typefilter=[ タイプフィルタ ][...] - 指定したオブジェクトタイプのカンマ区切りリストに一致する結果が表示されます。有効な値は、device、group、および folder です。

-c, --count - 結果の数が表示されます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

-f, --flagfilter=[ フラグフィルタ ][...] - 指定したフラグのカンマ区切りリストに一致する結果が表示されます。有効な値は、retired、notretired、test、non-test です。

#### **mobile-group-members (mgm) ( モバイルグループパス ) [ オプション ]**

モバイルデバイスグループまたは動的モバイルデバイスグループのメンバーを一連表示します。

次のオプションを受け入れます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **mobile-group-remove (mgr) ( モバイルグループのパス ) ( モバイルパス ) [...] [ オプション ]**

モバイルデバイスグループからモバイルデバイスを削除します。

次のオプションを受け入れます。

-f, --folder=[ モバイルフォルダ ] - /Devices/Mobile Devices へのモバイルデバイスフォルダの相対パスです。このオプションを指定すると、モバイルデバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のモバイルデバイスオブジェクトを指定するのに使用できます。

#### **mobile-list-groups (mlg) ( モバイルパス ) [ オプション ]**

任意のモバイルデバイスがメンバーであるグループを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **mobile-add-policy ( マップ ) ( モバイルオブジェクトのパス ) ( ポリシーまたはポリシーグループ ) [...] [ オプション ]**

ポリシーをモバイルデバイスオブジェクトに割り当てます。

次のオプションを受け入れます。

- c, --conflicts=[ ポリシー競合解決順序 ] - ポリシー競合の解決方法を決定します。有効な値は、userlast または 1、devicelast または 2、deviceonly または 3、useronly または 4 です。userlast の場合、デバイスに関連付けられたポリシーがまず適用され、それからユーザに関連付けられたポリシーが適用されます。devicelast の場合、ユーザに関連付けられたポリシーがまず適用され、それからデバイスに関連付けられたポリシーが適用されます。deviceonly の場合、ユーザに関連付けられたポリシーは無視されます。useronly の場合、デバイスに関連付けられたポリシーは無視されます。このオプションを指定しなかった場合には、userlast がデフォルト値として使用されます。
- e, --enforce-now - 割り当てられているすべてのデバイスに対してポリシーをすぐに強制します。
- f, --folder=[ ポリシーフォルダ ] - / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

**mobile-remove-policy (mrp) ( モバイルオブジェクトのパス ) ( ポリシーまたはポリシーグループ ) [...]**  
**[ オプション ]**

モバイルデバイスオブジェクトに割り当てられたポリシーを削除します。  
 次のオプションを受け入れます。

- f, --folder=[ ポリシーフォルダ ] - / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

**mobile-list-policies (mlp) ( モバイルオブジェクトのパス ) [ オプション ]**

モバイルデバイスオブジェクトに割り当てられたポリシーを一覧表示します。  
 次のオプションを受け入れます。

- a, --all - 有効および無効のポリシー両方を一覧表示します。
- e, --effective - 有効なポリシーのみ一覧表示します。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

**mobile-reorder-policies (mrop) ( モバイルオブジェクトのパス ) ( 現在の位置 ) ( 新しい位置 )**

モバイルデバイスに割り当てられたポリシーの順序を変更します。mobile-list-policies コマンドを使用して、割り当てられたポリシーの順序を取得します。

**mobile-view-all-policies-status (mvaps) ( モバイルパス ) [ オプション ]**

モバイルに割り当てられたポリシーの詳細な展開ステータスを表示します。  
 次のオプションを受け入れます。

- statusfilter=[ ステータスタイプ ] [...] - ポリシー適用イベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

**mobile-set-as-test (msat) ( モバイルオブジェクトのパス )**

モバイルデバイスをテストデバイスとして設定します。

例：

mobiledevice1 をテストデバイスとして設定する場合：

```
zman msat mobiledevice1
```

#### **mobile-set-as-nontest (msan) ( モバイルオブジェクトのパス )**

モバイルデバイスをテストデバイス以外として設定します。

例：

mobiledevice1 をテストデバイス以外として設定する場合：

```
zman msan mobiledevice1
```

## オブジェクトコマンド

ZENworks オブジェクトで操作するユーティリティまたは便利なコマンドです。オブジェクトコマンドは、長い形式では `object-` から始まり、短い形式では `o` から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [67 ページの「object-get-GUID \(ogg\) \( オブジェクトパス \)」](#)
- [67 ページの「object-get-name \(ogn\) \(ZENworks GUID\)」](#)
- [67 ページの「object-get-path \(ogp\) \(ZENworks GUID\)」](#)

#### **object-get-GUID (ogg) ( オブジェクトパス )**

ZENworks オブジェクトの GUID を取得します。

( オブジェクトパス )-スラッシュ (/) から始まる ZENworks オブジェクトのフルパス。

#### **object-get-name (ogn) (ZENworks GUID)**

ZENworks オブジェクトの名前を取得します。

(ZENworks GUID) - オブジェクトの ZENworks GUID で、32 文字から構成されます。

#### **object-get-path (ogp) (ZENworks GUID)**

指定した GUID を持つ ZENworks オブジェクトのパスを取得します。

(ZENworks GUID) - オブジェクトの ZENworks GUID で、32 文字から構成されます。

## プライマリサーバコマンド

プライマリサーバ上のコンテンツの管理に使用するコマンドです。プライマリサーバコマンドは、`primary-server-` から始まります。

- [68 ページの「primary-server-replicate-content \(psrc\) \( バンドルパス | ポリシーパス \)」](#)
- [68 ページの「primary-server-cleanup-content \(pscc\) \( バンドル \) \( バンドルパス | ポリシーパス \)」](#)

### primary-server-replicate-content (psrc) (バンドルパス | ポリシーパス)

プライマリサーバ上の指定されたバンドルまたはポリシーを複製します。

例:

バンドルのコンテンツを現在のサーバに複製するには、`zman psrc /Bundle/bundle1` を指定します。

ポリシーのコンテンツを現在のサーバに複製するには、`zman psrc /Policy/policy1` を指定します。

### primary-server-cleanup-content (pscc) (バンドル) (バンドルパス | ポリシーパス)

プライマリサーバ上の指定されたバンドルまたはポリシーのコンテンツをクリーンアップします。

例:

現在のサーバからバンドルのコンテンツをクリーンアップするには、`zman pscc /Bundle/bundle1` を指定します。

現在のサーバからポリシーのコンテンツをクリーンアップするには、`zman pscc /Policy/policy1` を指定します。

## ポリシーコマンド

ポリシーの作成、変更、および管理を行うコマンドです。ポリシーコマンドは、長い形式では `policy-` から始まり、短い形式では `p` から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [69 ページ](#)の「`policy-assign (pa)` (デバイスまたはユーザタイプ) (ポリシーまたはポリシーグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]
- [70 ページ](#)の「`policy-create (pc)` (ポリシー名) (ポリシーXMLファイル) [親フォルダ] [オプション]
- [70 ページ](#)の「`policy-copy (pcp)` (ポリシーパス) (ポリシー名) [親フォルダ] [オプション]
- [70 ページ](#)の「`policy-delete (pd)` (ポリシーオブジェクトのパス) [...] [オプション]
- [71 ページ](#)の「`policy-disable (pdl)` (ポリシーパス) [...] [オプション]
- [71 ページ](#)の「`policy-delete-version (pdv)` (ポリシーパス) [削除するポリシーのバージョン] [オプション]
- [71 ページ](#)の「`policy-enable (pel)` (ポリシーパス) [...] [オプション]
- [71 ページ](#)の「`policy-export-to-file (petf)` (ポリシーパス) (XMLファイルパス) [ポリシーのバージョン] [-c|--export-content]
- [72 ページ](#)の「`policy-folder-create (pfc)` (フォルダ名) [親フォルダ] [オプション]

- ◆ 72 ページの「policy-group-add (pga) ( ) (ポリシーパス) [...] [オプション]
- ◆ 72 ページの「policy-group-create (pgc) (グループ名) [親フォルダ] [オプション]
- ◆ 72 ページの「policy-group-members (pgm) (ポリシーグループのパス) [オプション]
- ◆ 73 ページの「policy-group-remove (pgr) (ポリシーグループのパス) (ポリシーパス) [...] [オプション]
- ◆ 73 ページの「policy-list (pl) [オプション] [フォルダ]
- ◆ 73 ページの「policy-list-assignments (plas) (ポリシーまたはポリシーグループ) [オプション]
- ◆ 73 ページの「policy-list-groups (plg) (ポリシーパス) [オプション]
- ◆ 73 ページの「policy-list-version (plv) (ポリシーパス)
- ◆ 73 ページの「policy-move (pmv) (ポリシーオブジェクトのパス) [宛先フォルダパス]
- ◆ 74 ページの「policy-sandbox-create (psc) (ポリシーパス) [ポリシーのバージョン] [オプション]
- ◆ 74 ページの「policy-sandbox-publish (psp) (ポリシーパス)
- ◆ 74 ページの「policy-sandbox-publish-to-newpolicy (psptn) (ポリシーパス) (ポリシー名) [親フォルダ] [オプション]
- ◆ 74 ページの「policy-sandbox-revert (psr) (ポリシーパス)
- ◆ 75 ページの「policy-rename (pr) (ポリシーオブジェクトのパス) (新しい名前) [オプション]
- ◆ 75 ページの「policy-unassign (pua) (デバイスまたはユーザタイプ) (ポリシーまたはポリシーグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]
- ◆ 75 ページの「policy-view-advanced-deployment-status (pvads) (ポリシーパス) [オプション]
- ◆ 76 ページの「policy-view-summary-totals (pvst) (ポリシーパス) [オプション]

**policy-assign (pa) (デバイスまたはユーザタイプ) (ポリシーまたはポリシーグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]**

ポリシーまたはポリシーグループを 1 つまたは複数のデバイスまたはユーザオブジェクトに割り当てます。

(デバイスまたはユーザタイプ) - 有効な値は、device、server、workstation、および user です。

(デバイスまたはユーザオブジェクトパス) [...] - 指定したデバイスまたはユーザタイプのルートフォルダを基準にした、デバイスまたはユーザオブジェクトの相対パス。

次のオプションを受け入れます。

- c, --conflicts=[ ポリシー競合解決順序 ] - ポリシーの競合の解決方法を決定します。有効な値は、userlast または 1、devicelast または 2、deviceonly または 3、useronly または 4 です。userlast の場合、デバイスに関連付けられたポリシーがまず適用され、それからユーザに関連付けられたポリシーが適用されます。devicelast の場合、ユーザに関連付けられたポリシーがまず適用され、それからデバイスに関連付けられたポリシーが適用されます。deviceonly の場合、ユーザに関連付けられたポリシーは無視されます。useronly の場合、デバイスに関連付けられたポリシーは無視されます。このオプションを指定しなかった場合には、userlast がデフォルト値として使用されます。
- e, --enforce-now - 割り当てられているすべてのデバイスに対してポリシーをただちに強制的に適用します。
- f, --folder=[ デバイスまたはユーザフォルダ ] - ルートフォルダを基準にした、デバイスまたはユーザフォルダの相対パス。このオプションを指定すると、デバイスまたはユーザオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のデバイスまたはユーザオブジェクトを指定するのに使用できます。

#### **policy-create (pc) (ポリシー名) (ポリシー XML ファイル) [親フォルダ] [オプション]**

ポリシーを作成します。

(ポリシー XML ファイル) - エクスポートしたポリシー情報が含まれる XML ファイル。policy-export-to-file (petf) を使用して、ポリシーの情報を XML ファイルにエクスポートします。ファイルを再使用する場合には、ZENworks コントロールセンターで作成したポリシーからテンプレート XML ファイルを作成することができます。

次のオプションを受け入れます。

- d, --desc=[ 説明 ] - ポリシーの説明。
- a, --actioninfo=[ ポリシーのファイルコンテンツ ] - ポリシーに関連付けられ、パッケージ化されたファイルコンテンツに関する情報が含まれた XML ファイル。たとえば、プリンタポリシーに対してインストールするドライバファイルはファイルコンテンツです。XML 形式のテンプレートについては、ActionInfo.xml ファイルを参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/policies、Windows サーバでは *Installation\_directory*:\Novell\Zenworks\share\zman\samples\policies に保存されています。
- s, --create-as-sandbox=[ サンドボックスとして作成 ] - ポリシーをサンドボックスとして作成します。
- n, --create-with-new-grid - 新しい GUID を使用してポリシーオブジェクトを作成します。オプションが指定されない場合、ポリシー XML ファイル内のインポートされた GUID のポリシーオブジェクトが検証されます。インポートされた GUID が存在しない場合は新しい GUID が作成され、存在する場合はインポートされた GUID が保持されます。

#### **policy-copy (pcp) (ポリシーパス) (ポリシー名) [親フォルダ] [オプション]**

ポリシーをコピーします。

次のオプションを受け入れます。

- d, --desc=[ 説明 ] - ポリシーの説明。

#### **policy-delete (pd) (ポリシーオブジェクトのパス) [...] [オプション]**

1 つまたは複数のポリシーオブジェクトを削除します。

(ポリシーオブジェクトのパス) [...] - /ポリシーを基準にした、ポリシーオブジェクト (ポリシー、ポリシーフォルダ、またはポリシーグループ) の相対パス。ワイルドカード \* は、引用符で囲まれている場合には、オブジェクト名に使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

次のオプションを受け入れます。

-r, --recursive - フォルダ内のオブジェクトを再帰的に削除します。

-f, --folder=[ポリシーフォルダ] - /ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **policy-disable (pdl) (ポリシーパス) [...] [オプション]**

ポリシーを無効にします。

次のオプションを受け入れます。

-f, --folder=[ポリシーフォルダ] - /ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **policy-delete-version (pdv) (ポリシーパス) [削除するポリシーのバージョン] [オプション]**

1 つまたは複数のポリシーバージョンを削除します。

例:

ポリシー zenpolicy のバージョン 3 を削除する場合:

```
zman pdv zenpolicy 3
```

ポリシー zenpolicy のすべてのバージョンを削除する場合:

```
zman pdv zenpolicy -a
```

ポリシー zenpolicy の指定されたバージョン以前のすべてのバージョンを削除する場合:

```
zman pdv zenpolicy version -p
```

#### **policy-enable (pel) (ポリシーパス) [...] [オプション]**

ポリシーを有効にします。

次のオプションを受け入れます。

-f, --folder=[ポリシーフォルダ] - /ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **policy-export-to-file (petf) (ポリシーパス) (XML ファイルパス) [ポリシーのバージョン] [-c|--export-content]**

ポリシーの情報 (XML 形式) をファイルにエクスポートします。この XML ファイルはポリシー作成時の入力として使用します。

(XML ファイルパス) - ポリシーの情報をエクスポートする XML ファイルの完全なパス。

[ポリシーのバージョン]- エクスポートするポリシーのバージョン。バージョンが指定されていない場合、発行されたバージョンのポリシーがエクスポートされます。ポリシーのサンドボックスバージョンをエクスポートするには、sandbox を指定します。

[-c|--export-content]- ポリシー情報のエクスポート先である XML ファイルが格納されているディレクトリ内のサブディレクトリに、ポリシーのコンテンツをエクスポートします。オプションを指定しない場合、ポリシーコンテンツはエクスポートされません。

例：

zenpolicy という名前のサンドボックスバージョンのポリシーを、policy.xml という名前の xml ファイルにエクスポートする場合：

```
zman petf policyfolder/zenpolicy C:\policies\policy.xml sandbox
```

zenpolicy という名前のバージョン 3 のポリシーを、policy.xml という名前の xml ファイルにエクスポートする場合：

```
zman petf policyfolder/zenpolicy C:\policies\policy.xml 3
```

zenpolicy という名前のバージョン 5 のポリシーを policy.xml という名前の xml ファイルにエクスポートし、ポリシーのコンテンツを C:\policies\ にエクスポートする場合：

```
zman petf policyfolder/zenpolicy C:\policies\policy.xml 5 -c
```

#### **policy-folder-create (pfc) (フォルダ名) [親フォルダ] [オプション]**

ポリシーを格納するための新しいフォルダを作成します。

次のオプションを受け入れます。

--desc=[説明]- フォルダの説明。

#### **policy-group-add (pga) () (ポリシーパス) [...] [オプション]**

ポリシーグループにポリシーを追加します。

次のオプションを受け入れます。

-f, --folder=[ポリシーフォルダ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **policy-group-create (pgc) (グループ名) [親フォルダ] [オプション]**

ポリシーグループを作成して、メンバーを追加します。

次のオプションを受け入れます。

--desc=[説明]- グループの説明。

-m, --members=[ポリシーパス] [...] - / ポリシーを基準にした、ポリシーの相対パス。

-f, --folder=[ポリシーフォルダ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **policy-group-members (pgm) (ポリシーグループのパス) [オプション]**

ポリシーグループのメンバーを一覧表示します。



次のオプションを受け入れます。

-s, --scrollsize=[スクロールサイズ]- 一度に表示させる結果の数。

#### **policy-group-remove (pgr) (ポリシーグループのパス) (ポリシーパス) [...] [オプション]**

ポリシーグループからポリシーを削除します。

次のオプションを受け入れます。

-f, --folder=[ポリシーフォルダ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **policy-list (pl) [オプション] [フォルダ]**

ポリシーオブジェクトを一覧表示します。

次のオプションを受け入れます。

-r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。

-n, --namefilter=[フィルタ文字列]- 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を引用符で囲まれている限りは使用できません。

-c, --count - 結果の数が表示されます。

-s, --scrollsize=[スクロールサイズ]- 一度に表示させる結果の数。

#### **policy-list-assignments (plas) (ポリシーまたはポリシーグループ) [オプション]**

ポリシーのデバイスとユーザ割り当てを一覧表示します。

次のオプションを受け入れます。

-t, --typefilter=[割り当てのタイプ]- 割り当てタイプでフィルタします。有効な値は、device および user です。

-s, --scrollsize=[スクロールサイズ]- 一度に表示される結果の数。

#### **policy-list-groups (plg) (ポリシーパス) [オプション]**

指定したポリシーがメンバーであるグループのリストを表示します。

次のオプションを受け入れます。

-s, --scrollsize=[スクロールサイズ]- 一度に表示させる結果の数。

#### **policy-list-version (plv) (ポリシーパス)**

ポリシーのすべてのバージョンを一覧表示します。

例:

ポリシー zenpolicy のすべてのバージョンを一覧表示する場合:

```
zman plv zenpolicy
```

#### **policy-move (pmv) (ポリシーオブジェクトのパス) [宛先フォルダパス]**

ポリシーオブジェクトを別のフォルダに移動します。

### **policy-sandbox-create (psc) (ポリシーパス) [ポリシーのバージョン] [オプション]**

既存のポリシーバージョンからサンドボックスを作成します。

[ポリシーのバージョン] - サンドボックスの作成に使用されるポリシーのバージョンを指定します。

次のオプションを受け入れます。

-f, --force =[強制] - 既存のサンドボックスを上書きします。

例:

ポリシー zenpolicy のバージョン 2 からサンドボックスを作成する場合:

```
zman psc zenpolicy 2
```

### **policy-sandbox-publish (psp) (ポリシーパス)**

サンドボックスを発行してポリシーの発行済みバージョンを作成します。

例:

ポリシー zenpolicy のサンドボックスを発行する場合:

```
zman psp zenpolicy
```

### **policy-sandbox-publish-to-newpolicy (psptn) (ポリシーパス) (ポリシー名) [親フォルダ] [オプション]**

サンドボックスバージョンのポリシーを新しいポリシーに発行します。

(ポリシーパス) - ポリシーのルートフォルダである /Policies を基準にしたポリシーオブジェクトの相対パス。

(ポリシー名) - 作成する新しいポリシーの名前。

[親フォルダ] - 新しいポリシーを作成するフォルダ。指定されていない場合、ポリシーがルートフォルダに作成されます。

次のオプションを受け入れます。

-s, --createAsSandbox =[サンドボックスとして作成] - ポリシーをサンドボックスとして作成します。

-g, --groups - 新しく作成したポリシーを、ソースポリシーがメンバーになっているすべてのグループに追加します。

-d, --deviceAssignments - すべてのデバイス割り当てをソースポリシーから新しく作成するポリシーにコピーします。

-u, --userAssignments - すべてのユーザ割り当てをソースポリシーから新しく作成するポリシーにコピーします。

例:

サンドボックスバージョンのポリシーを新しいポリシーに発行する場合:

```
zman psptn policy1 policy2 /Policies/Folder1
```

### **policy-sandbox-revert (psr) (ポリシーパス)**

サンドボックスを削除して、ポリシーの最新の発行済みバージョンに戻します。

例:

サンドボックスを削除して、ポリシーの最新の発行済みバージョンに戻す場合：

```
zman psr zenpolicy
```

#### **policy-rename (pr) (ポリシーオブジェクトのパス) (新しい名前) [オプション]**

ポリシーオブジェクトを名前変更します。

-p, --publish=[発行]- このオプションが指定され、バンドルの表示名が編集された場合、バンドルは即座に発行されます。ただし、現在のバンドルの表示名が既存のバンドル名と異なる場合、またはバンドルにサンドボックスがすでに存在する場合は、このオプションを指定しても効果はありません。

#### **policy-unassign (pua) (デバイスまたはユーザタイプ) (ポリシーまたはポリシーグループ) (デバイスまたはユーザオブジェクトパス) [...] [オプション]**

ポリシーまたはポリシーグループを1つまたは複数のデバイスまたはユーザオブジェクトから割り当て解除します。

(デバイスまたはユーザタイプ)- 有効な値は、device、server、workstation、および user です。

(デバイスまたはユーザオブジェクトパス) [...] - 指定したデバイスまたはユーザタイプのルートフォルダを基準にした、デバイスまたはユーザオブジェクトの相対パス。

次のオプションを受け入れます。

-f, --folder=[デバイスまたはユーザフォルダ]- ルートフォルダを基準にした、デバイスまたはユーザフォルダの相対パス。このオプションを指定すると、デバイスまたはユーザオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のデバイスまたはユーザオブジェクトを指定するのに使用できます。

#### **policy-view-advanced-deployment-status (pvads) (ポリシーパス) [オプション]**

ポリシーの展開ステータスの詳細を表示します。

次のオプションを受け入れます。

-d, --device=[デバイスパス]- 指定されたデバイスのみに対する配布ステータスを表示します。デバイスのパスは、/デバイスを基準にした相対パス。

-u, --user=[ユーザパス]- 指定したユーザのみの展開ステータスを表示します。ユーザのパスは、/ユーザを基準にした相対パス。デバイスも指定した場合、指定したデバイスにログインしている指定したユーザのステータス詳細が表示されます。

-n, --namefilter=[ターゲットデバイス名]- デバイスの名前でフィルタします。指定したフィルタに一致するオプションを表示します。ワイルドカードの \* および ? は、引用符で囲まれている限りは使用できます。

--statusfilter=[ステータスタイプ] [...] - ポリシー適用イベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

-t, --typefilter=[ターゲットデバイスまたはユーザタイプ] [...] - ターゲットのタイプでフィルタします。有効な値は、server、workstation、および user です。ターゲットタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ]- 一度に表示させる結果の数。

-v, --version[値]

有効な値は次のとおりです。

**published:** このオプションを使用して、ポリシーを展開する非テストデバイスの数を表示します。

「sandbox」：このオプションを使用して、ポリシーのサンドボックスバージョンを展開するテストデバイスの数を表示します。

**[version-of-the-object]:** バージョン番号を使用して関連するバンドルバージョンのステータスを取得します。

デフォルト値は「published」です。

-c, --status-calculation[ 値 ]

有効な値は次のとおりです。

**consolidated:** このオプションを使用して、このポリシーを展開するデバイスの合計数を表示します。

「version」：このオプションを使用して、特定のバージョンのステータスを表示します。

デフォルト値は「version」です。

#### **policy-view-summary-totals (pvst) (ポリシーパス) [オプション]**

特定のポリシーの展開ステータスのサマリを表示します。

次のオプションを受け入れます。

-v, --version[ 値 ]

有効な値は次のとおりです。

- 「published」：このオプションを使用して、ポリシーを展開する非テストデバイスの数を表示します。
- 「sandbox」：このオプションを使用して、ポリシーのサンドボックスバージョンを展開するテストデバイスの数を表示します。
- **[version-of-the-object]:** バージョン番号を使用して関連するバンドルバージョンのステータスを取得します。  
デフォルト値は「published」です。

-c, --status-calculation[ 値 ]

- 「consolidated」：このオプションを使用して、このポリシーを展開するデバイスの合計数を表示します。
- 「version」：このオプションを使用して、特定のバージョンのステータスを表示します。

デフォルト値は「version」です。

## ポリシー設定コマンド

ポリシー設定の設定と変更に使用するコマンドです。設定コマンドは、長い形式では settings- から始まり、短い形式では s から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [77 ページ](#)の「[policy-settings-copy \(pscp\)](#) (ポリシーソースのパス) (宛先ポリシーまたはポリシーフォルダパス) [...] [オプション]」
- [77 ページ](#)の「[policy-settings-export-to-file \(psetf\)](#) (XML ファイルパス) [設定名] [オプション] [...]」
- [78 ページ](#)の「[policy-settings-revert \(psrt\)](#) (ポリシーソースのパス) (設定名) [...]」
- [78 ページ](#)の「[policy-settings-set \(pss\)](#) [オプション] (XML ファイルパス) [宛先ポリシーまたはポリシーフォルダパス] [...]」

### **policy-settings-copy (pscp) (ポリシーソースのパス) (宛先ポリシーまたはポリシーフォルダパス) [...] [オプション]**

設定をコピーします。

(ソースポリシーまたはポリシーフォルダパス) - / ポリシーを基準にした、設定のコピー元ポリシーまたはポリシーフォルダの相対パス。

(宛先ポリシーまたはポリシーフォルダパス) [...] - / ポリシーを基準にした、設定のコピー先ポリシーまたはポリシーフォルダの相対パス。

次のオプションを受け入れます。

-n, --names=[設定名] [...] - ソースパスからコピーされる設定の名前。指定しなかった場合、ソースパスで定義されたすべての設定がコピーされます。

### **policy-settings-export-to-file (psetf) (XML ファイルパス) [設定名] [オプション] [...]**

設定データ (XML 形式) をファイルにエクスポートします。設定の作成および変更用の入力として XML ファイルが使用されます。

(XML ファイルパス) - 設定データが XML 形式で保存されるファイル。ファイルが存在しない場合、新しいファイルが作成されます。

[設定名] [...] - エクスポートする設定の名前。指定しなかった場合、すべての設定がエクスポートされます。

次のオプションを受け入れます。

s, --source=[ソースポリシーまたはポリシーフォルダパス] - / ポリシーを基準にした、設定のエクスポート元ポリシーまたはポリシーフォルダの相対パス。指定しなかった場合、設定はゾーンからエクスポートされます。

-e, --effective - 指定した場合は有効な設定が取得されます。そうでない場合はソースパスで定義 / 上書きされた設定のみが取得されます。

### **policy-settings-revert (psrt) (ポリシーソースのパス) (設定名) [...]**

親フォルダへの設定を取り消します。

(ソースポリシーまたはポリシーフォルダパス) - /ポリシーを基準にした、設定を元に戻すポリシーまたはポリシーフォルダの相対パス。

### **policy-settings-set (pss) [オプション] (XML ファイルパス) [宛先ポリシーまたはポリシーフォルダパス] [...]**

さまざまなレベルで設定を行います。

(XML ファイルパス) - エクスポートされた設定情報が含まれる XML ファイル。settings-export-to-file (setf) を使用して、設定情報を XML ファイルにエクスポートします。

次のオプションを受け入れます。

[設定名] [...] - エクスポートする設定の名前。指定しなかった場合、すべての設定がエクスポートされます。

-f, --force - すべての子 (サブフォルダおよび個々のポリシー) にこれらの設定の継承を強制します。

s, --source=[ソースポリシーまたはポリシーフォルダパス] - /ポリシーを基準にした、設定が必要なポリシーまたはポリシーフォルダの相対パス。指定しなかった場合、設定はゾーンレベルで設定されます。

## クエリコマンド

保存されている検索を一覧表示して実行する場合に使用するコマンドです。クエリコマンドは、長い形式では query- から始まり、短い形式では q から始まります。

- [78 ページの「query-list-savedsearches|qls \[-d|--detailed\]」](#)
- [78 ページの「query-run-savedsearch|qrs \(保存された検索\) \[フォルダパス\] \[-r|--recursive\] \[-s|--scrollsize=スクロールサイズ\]」](#)

### **query-list-savedsearches|qls [-d|--detailed]**

保存された検索を一覧表示します。

次のオプションを受け入れます。

[-d, --detailed] - 保存された検索のフィルタ基準を一覧表示します。

### **query-run-savedsearch|qrs (保存された検索) [フォルダパス] [-r|--recursive] [-s|--scrollsize=スクロールサイズ]**

保存された検索を実行します。

次のオプションを受け入れます。

(保存された検索) - 実行する、保存された検索。

[フォルダパス] - /デバイスを基準にした、検索を実行するデバイスフォルダの相対パス。デフォルトのフォルダは、/デバイスです。

[`-r, --recursive`] - このオプションは、ZENworks 11 SP3 で非推奨になりました。このオプションを指定しても無視されます。再帰的な値は、保存された検索から取得されません。

[`-s, --scrollsize=スクロールサイズ`] - 一度に表示される結果の数。

## キューコマンド

これらのコマンドは、キューを変更する必要がある場合に使用します。キューは、クライアント更新やコンテンツクリーンアップなどの非同期イベントを処理し、多くの場合には変更する必要がありません。キューコマンドは、長い形式では `queue-` から始まり、短い形式では `q` から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [79 ページの「`queue-flush \(qf\)` \( キューステータス \)」](#)
- [79 ページの「`queue-list \(ql\)` \[ サーバパス \] \[ オプション \]」](#)
- [79 ページの「`queue-reset \(qr\)` \[ サーバパス \]」](#)

### `queue-flush (qf)` ( キューステータス )

キューエントリを削除してキュー内容を消去します。

( キューステータス ) [...]- キューエントリのステータスでフィルタします。有効な値は、N、I、C、S、および F (New、In Progress、Cancelled、Success、および Failed) です。

### `queue-list (ql)` [ サーバパス ] [ オプション ]

キューエントリを一覧表示します。

[ サーバパス ] - / デバイス / サーバを基準にした、サーバの相対パス。特定のサーバに属するキューエントリのみ一覧表示します。

次のオプションを受け入れます。

`-t, --type=[ キューのアクションタイプ ]` - キューエントリのタイプでフィルタします。たとえば、`content.cleanup` はキューエントリのタイプです。

`-s, --status=[ キューステータス ]` - キューエントリのステータスでフィルタします。有効な値は、N、I、C、S、および F (New、In Progress、Cancelled、Success、および Failed) です。

### `queue-reset (qr)` [ サーバパス ]

キューをリセットし、失敗したキューエントリのステータスを New にセットします。

[ サーバパス ] - / デバイス / サーバを基準にした、サーバの相対パス。特定のサーバに属するキューエントリのみリセットします。

## 登録コマンド

登録の作成と変更を行うコマンドです。デバイスタイプを指定する必要があります。登録キーが関連付けられるデバイスフォルダとデバイスグループのタイプが決定されるので、登録デバイスのタイプも決定されます。登録コマンドは、長い形式では registration- から始まり、短い形式では r から始まります。

次のコマンドはすべて、「**グローバルオプション**」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- 80 ページの「registration-add-device-groups (radg) (デバイスタイプ) (登録キー) (デバイスグループパス) [...] [オプション]」
- 81 ページの「registration-create-key (rck) (デバイスタイプ) (登録キー) [登録フォルダパス] [オプション]」
- 81 ページの「registration-copy (rcp) (登録キー) (新しい名前)」
- 81 ページの「registration-delete (rd) (登録オブジェクトのパス) [...] [オプション]」
- 82 ページの「registration-folder-create (rfc) (フォルダ名) [親フォルダ] [オプション]」
- 82 ページの「registration-info (ri) (登録キー)」
- 82 ページの「registration-list (rl) [オプション] [フォルダ]」
- 82 ページの「registration-list-device-groups (rldg) (登録キー) (rd)」
- 82 ページの「registration-move (rmv) (登録オブジェクトのパス) [宛先フォルダパス]」
- 82 ページの「registration-remove-device-groups (rrdg) (デバイスタイプ) (登録キー) (デバイスグループパス) [...] [オプション]」
- 82 ページの「registration-rename (rr) (登録オブジェクトのパス) (新しい名前)」
- 82 ページの「registration-update-key (ruk) (デバイスタイプ) (登録キー) [オプション]」
- 83 ページの「registration-import-devices (rid) (CSV ファイルパス) (登録キー)」

**registration-add-device-groups (radg) (デバイスタイプ) (登録キー) (デバイスグループパス) [...] [オプション]**

任意のキーに登録するデバイスの、指定されたデバイスグループでメンバーシップを追加します。

(デバイスタイプ)-登録デバイスのタイプ。有効な値は、server および workstation です。



次のオプションを受け入れます。

-f, --folder=[ デバイスフォルダ ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。このオプションを指定すると、デバイスオブジェクトのパスは、このフォルダを基準にした相対パスとして決定されます。これは、同じフォルダで複数のデバイスオブジェクトを指定するのに使用できます。

#### **registration-create-key (rck) ( デバイスタイプ ) ( 登録キー ) [ 登録フォルダパス ] [ オプション ]**

新しい登録キーを作成します。

( デバイスタイプ ) - 登録デバイスのタイプ。有効な値は、server および workstation です。

次のオプションを受け入れます。

-f, --devicefolder=[ デバイスフォルダ ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。登録デバイスはこのフォルダに置かれます。

-g, --devicegroup=[ デバイスグループパス ][...] - デバイスグループのパスのカンマ区切りリスト。指定したパスは、指定したデバイスタイプのルートフォルダを基準にした相対パスである必要があります。登録デバイスはこれらのデバイスグループのメンバーになります。

--desc=[ 説明 ] - 登録キーの説明。

--site=[ サイト ] - デバイスが存在するサイト。登録デバイスにはこのサイト情報が入力されます。

--dept=[ 部署 ] - デバイスが使用される部署。登録デバイスにはこの部署情報が入力されます。

--loc=[ 場所 ] - デバイスの物理的な場所。登録デバイスにはこの場所情報が入力されます。

--limit=[ 制限 ] - このキーを登録デバイスに使用できる回数。

#### **registration-copy (rcp) ( 登録キー ) ( 新しい名前 )**

登録キーをコピーします。

( 登録キー ) - 登録キーの名前。

( 新しい名前 ) - コピーした登録キーの名前。

#### **registration-delete (rd) ( 登録オブジェクトのパス ) [...] [ オプション ]**

登録オブジェクトを削除します。

( 登録オブジェクトのパス ) [...] - / キーを基準にした、登録オブジェクト ( 登録キーまたはフォルダ ) の相対パス。ワイルドカード \* は、引用符で囲まれている場合には、オブジェクト名に使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

次のオプションを受け入れます。

-r, --recursive - 再帰的に、フォルダ内にオブジェクトを削除します。

-f, --folder=[ 登録フォルダ ] - / キーを基準にした、登録フォルダの相対パス。このオプションを指定すると、登録オブジェクトのパスは、このフォルダを基準にした相対パスとして決定されます。これは、同じフォルダで複数の登録オブジェクトを指定するのに使用できます。

**registration-folder-create (rfc) (フォルダ名) [親フォルダ] [オプション]**

新しい登録フォルダを作成します。

次のオプションを受け入れます。

--desc=[説明] - フォルダの説明。

**registration-info (ri) (登録キー)**

登録キーについての詳細情報を表示します。

**registration-list (rl) [オプション] [フォルダ]**

すべての登録オブジェクトを一覧表示します。

次のオプションを受け入れます。

-r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。

-n, --namefilter=[フィルタ文字列] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示される結果の数。

**registration-list-device-groups (rldg) (登録キー) (rd)**

登録キーに関連付けられるデバイスグループを一覧表示します。

**registration-move (rmv) (登録オブジェクトのパス) [宛先フォルダパス]**

登録オブジェクトを別の登録フォルダに移動します。

**registration-remove-device-groups (rrdg) (デバイスタイプ) (登録キー) (デバイスグループパス) [...] [オプション]**

デバイスグループと登録キーの関連付けを削除します。

(デバイスタイプ) - 登録デバイスのタイプ。有効な値は、server および workstation です。

次のオプションを受け入れます。

-f, --folder=[デバイスフォルダ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。このオプションを指定すると、デバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のデバイスオブジェクトを指定するのに使用できます。

**registration-rename (rr) (登録オブジェクトのパス) (新しい名前)**

登録オブジェクトを名前変更します。

**registration-update-key (ruk) (デバイスタイプ) (登録キー) [オプション]**

登録キーをアップデートします。

次のオプションを受け入れます。

-f, --devicefolder=[デバイスフォルダ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。登録デバイスは、このフォルダに置かれます。

--desc=[説明] - 登録キーの説明。

- site=[ サイト ]- デバイスが存在するサイト。登録デバイスにはこのサイト情報が入力されます。
- dept=[ 部署 ]- デバイスが使用される部署。登録デバイスにはこの部署情報が入力されます。
- loc=[ 場所 ]- デバイスの物理的な場所。登録デバイスにはこの場所情報が入力されません。
- limit=[ 制限 ]- デバイスを登録するためにこのキーを使用できる回数。
- unlimited - このキーはデバイスを登録するために無制限に使用できます。

#### registration-import-devices (rid) (CSV ファイルパス) (登録キー)

デバイス情報の指定に使用される CSV ファイルからデバイスをインポートします。

(CSV ファイルパス) - CSV ファイルの名前とパス。ファイル内の各エントリには、ワークステーションまたはサーバの詳細が含まれます。

ワークステーションまたはサーバのファイルエントリの形式は、バージョン、ホスト名、シリアル番号、および MAC アドレスになります。

ワークステーションまたはサーバのバージョンは WS\_1.0 です。

バージョンとホスト名の値は必須で、シリアル番号と MAC アドレスの値はオプションです。

(登録キー) - 登録キーの名前。

---

注: ファイルが UTF-8 形式であることを確認します。

---

次に例を示します。

devicesInfo という CSV ファイル内のデバイスを regkey という登録キーを使用して登録するには、zman rid c:\devicesInfo.csv regkey コマンドを実行します。

## 役割コマンド

これらのコマンドは、管理者アカウントを権限ベースで管理するための役割を作成および管理する場合に使用します。役割コマンドは、長い形式では role- から始まり、短い形式では ro から始まります。

- ◆ 84 ページの 「role-add-admins (rlaa) [役割名] [ZENworks 管理者名] [...] [権限カテゴリ] [オブジェクトパス] [...] [オプション]」
- ◆ 84 ページの 「role-create (rlc) (役割名)」
- ◆ 84 ページの 「role-delete (rld) (役割名) [...]」
- ◆ 84 ページの 「role-list (rll) [オプション]」
- ◆ 84 ページの 「role-list-admins (rlla) [役割名]」
- ◆ 85 ページの 「role-remove-admins (rlra) (役割名) (ZENworks 管理者名) [...] [権限カテゴリ] [オブジェクトパス] [...] [オプション]」
- ◆ 85 ページの 「role-rename (rlr) (役割名) (新しい名前)」
- ◆ 85 ページの 「role-rights-delete (rlrd) (役割名) [権限カテゴリ]」

- ◆ 85 ページの「role-rights-info (rlri) (役割名) [オプション]」
- ◆ 85 ページの「role-rights-set (rlrs) [役割名] [権限カテゴリ] [オプション]」

#### **role-add-admins (rlaa) [役割名] [ZENworks 管理者名] [...] [権限カテゴリ] [オブジェクトパス] [...] [オプション]**

役割を 1 つまたは複数の ZENworks 管理者に割り当て、役割の権限カテゴリを異なるオブジェクトのコンテキストに関連付けします。

[権限カテゴリ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、`zman role-rights-add-admins --help | more` を実行します。

[オブジェクトパス] [...] - スラッシュ (/) から始まる ZENworks オブジェクトのフルパス。そのオブジェクトに対して、権限カテゴリにグループ化された権限が強制的に適用されます。権限カテゴリに関連付けることができるオブジェクトタイプのルートフォルダを表示するには、`zman role-add-admins --help | more` を実行します。

次のオプションを受け入れます。

- f, --folder=[フォルダ名] - フォルダのフルパスです。このオプションを指定すると、オブジェクトのパスは、このフォルダに相対して決定されます。これで、同じフォルダに複数のオブジェクトを指定する場合に便利になります。
- H, --help - 利用可能な権限のカテゴリ、および権限を実行可能なオブジェクトタイプのルートフォルダに関する詳細なヘルプが表示されます。

#### **role-create (rlc) (役割名)**

ZENworks 管理者の役割を作成します。

#### **role-delete (rld) (役割名) [...]**

ZENworks 管理者の役割の名前を変更します。

(役割名) [...] - 役割の名前。引用符で囲まれている限り、ワイルドカード \* を使用できます。ワイルドカードを使用してオブジェクトを削除する場合には、注意してください。

#### **role-list (rll) [オプション]**

ZENworks 管理者の役割をすべてを一覧表示します。

次のオプションを受け入れます。

- n, --namefilter=[フィルタ文字列] - 指定した名前に一致する結果が表示されます。引用符で囲む限り、ワイルドカード \* および ? を使用できます。
- N, --name-only - 役割名だけを表示します。
- T, --name-and-type - 役割名と特権タイプだけを表示します。
- l, --longnames - 権限の名前を表示します。デフォルトでは、権限名は省略形式で表示されます。

#### **role-list-admins (rlla) [役割名]**

特定の役割、およびその役割の権限カテゴリと関連付けられたコンテキストを持つ ZENworks 管理者を一覧表示します。

**role-remove-admins (rlra) ( 役割名 ) (ZENworks 管理者名) [...] [ 権限カテゴリ ] [ オブジェクトパス ] [...] [ オプション ]**

1 つまたは複数の ZENworks 管理者からの役割の割り当てを削除する、またはオブジェクトコンテキストへの権限カテゴリの関連付けを削除します。

[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、zman role-rights-add-admins --help | more を実行します。

[ オブジェクトパス ] [...] - スラッシュ (/) から始まる ZENworks オブジェクトのフルパス。そのオブジェクトに対して、権限カテゴリにグループ化された権限が強制的に適用されます。権限カテゴリに関連付けることができるオブジェクトタイプのルートフォルダを表示するには、zman role-add-admins --help | more を実行します。

次のオプションを受け入れます。

-f, --folder=[ フォルダ名 ] - フォルダのフルパス。このオプションを指定すると、オブジェクトのパスは、このフォルダに相対して決定されます。これで、同じフォルダに複数のオブジェクトを指定する場合に便利になります。

**role-rename (rlr) ( 役割名 ) ( 新しい名前 )**

ZENworks 管理者の役割の名前を変更します。

**role-rights-delete (rlrd) ( 役割名 ) [ 権限カテゴリ ]**

ZENworks 管理者の役割に割り当てられる権限を削除します。

[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、zman role-rights-set --help | more を実行します。

**role-rights-info (rlri) ( 役割名 ) [ オプション ]**

ZENworks 管理者の役割に割り当てられる権限を表示します。

次のオプションを受け入れます。

-c, --category=[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、zman role-rights-set --help | more を実行します。

-l, --longnames - 権限の名前を表示します。デフォルトでは、権限名は省略形式で表示されます。

**role-rights-set (rlrs) [ 役割名 ] [ 権限カテゴリ ] [ オプション ]**

ZENworks 管理者の役割の権限を許可または拒否します。

[ 権限カテゴリ ] - 権限がグループ化されているカテゴリの名前。カテゴリのリストを確認するには、zman role-rights-set --help | more を実行します。

次のオプションを受け入れます。

-a, --allow=[ 権限 ] [...] - 許可されている権限の長い名前または短い名前のカンマ区切りリスト。各カテゴリの権限を表示するには、zman role-rights-set --help | more を実行します。

-A, --allow-all - 権限カテゴリの権限をすべて許可します。

-d, --deny=[ 権限 ] [...] - 拒否する権限の長い名前または短い名前のカンマ区切りリスト。各カテゴリの権限を表示するには、zman role-rights-set --help | more を実行します。

- D, --deny-all - 権限カテゴリの権限をすべて拒否します。
- u, --unset=[ 権限 ][...] - 設定解除する権限の長い名前または短い名前のカンマ区切りリスト。各カテゴリの権限を表示するには、zman role-rights-set --help | more を実行します。
- H, --help - 使用可能な権限カテゴリに関する詳しいヘルプおよび各カテゴリ内の権限名の長い形式と短い形式を表示します。カテゴリ名を指定した場合、そのカテゴリの権限に関するヘルプが提供されます。

## ルールセットコマンド

登録ルールセットの作成および変更を行うコマンドです。ルールセットは、登録キーを持たないZENworks サーバに新しいデバイスを登録するのに、適用されます。これらのコマンドでは、デバイスタイプを指定する必要があります。その上で、デバイスフォルダのタイプと、ルールセットが関連付けられるデバイスグループを決定するため、登録デバイスのタイプも決定します。ルールセットコマンドは、長い形式では ruleset- から始まり、短い形式では rs から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [87 ページ](#)の「ruleset-add-device-groups (rsadg) (デバイスタイプ) (ルールセット名または位置) (デバイスグループパス) [...] [オプション]」
- [87 ページ](#)の「ruleset-add-rule (rsar) (ルールセット名または位置) (追加モード) (ルールグループの位置) (ルールファイル)」
- [88 ページ](#)の「ruleset-create (rsc) (デバイスタイプ) (ルールセット名) (ルールファイル) [位置] [オプション]」
- [89 ページ](#)の「ruleset-copy (rscp) (ルールセット名または位置) (新しい名前) [位置]」
- [89 ページ](#)の「ruleset-export-rules-to-file (rsertf) (ルールセット名または位置) (XML ファイルパス)」
- [89 ページ](#)の「ruleset-delete (rsd) (ルールセット名または位置)」
- [89 ページ](#)の「ruleset-info (rsi) (ルールセット名または位置)」
- [90 ページ](#)の「ruleset-list (rsl)」
- [90 ページ](#)の「ruleset-list-device-groups (rsl dg) (ルールセット名または位置)」
- [90 ページ](#)の「ruleset-move (rsmv) (ルールセット名または位置) (新しい位置)」
- [90 ページ](#)の「ruleset-remove-device-groups (rsrdg) (デバイスタイプ) (ルールセット名または位置) (デバイスグループパス) [...] [オプション]」
- [90 ページ](#)の「ruleset-remove-rule (rsrr) (ルールセット名または位置) (ルールの位置) [...]」

- ◆ 91 ページの「ruleset-rename (rsr) (ルールセット名または位置) (新しい名前)」
- ◆ 91 ページの「ruleset-update (rsu) (デバイス) (ルールセット名または位置) [オプション]

#### ruleset-add-device-groups (rsadg) (デバイスタイプ) (ルールセット名または位置) (デバイスグループパス) [...] [オプション]

指定したルールセットを使用して、インポートしたデバイスに対して指定されているデバイスグループにメンバーシップを追加します。

(デバイスタイプ) - 登録デバイスのタイプ。有効な値は、server および workstation です。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

(デバイスグループパス) - デバイスタイプのルートフォルダを基準にした、デバイスグループの相対パスを指定します。

次のオプションを受け入れます。

-f, --folder=[デバイスフォルダ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。このオプションを指定すると、デバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のデバイスオブジェクトを指定するのに使用できます。

#### ruleset-add-rule (rsar) (ルールセット名または位置) (追加モード) (ルールグループの位置) (ルールファイル)

ルールセットにルールまたはルールグループを追加します。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

(追加モード) - ルールまたはルールグループを追加しているときの追加のモードを指定します。有効な値は、rule および rulegroup です。モードが rule の場合、入力ファイル内のルールグループすべてのルールが選択され、追加されます。

(ルールグループの位置) - ルールグループの位置を指定します。ルールグループを追加している場合、これは新しいグループが挿入される位置を表します。ルールを追加している場合は、新しいルールが追加されるルールグループ内の位置を表します。最後のルールグループを指すには、last と指定します。新しいルールグループにルールを追加する、またはリストの末尾にルールグループを追加するには、makenew と指定します。最初のルールグループは position 1 にあります。

(ルールファイル) - ルールセットに追加されるルールまたはルールグループの情報が含まれる XML ファイルのパス。XML ファイルは、ruleset-export-rules-to-file コマンドを使用して、既存のルールセットのルールをエクスポートすることによって作成できます。XML ファイルに指定された結合子は重要ではありません。または、ルールのみを追加する場合は、古い形式も使用できます。

ファイル形式については、AddRulesToRuleSet.txt を参照してください。このファイルは、Linux サーバでは /opt/novell/zenworks/share/zman/samples/rulesets、Windows サーバでは <Installation directory>:\Novell\Zenworks\share\zman\samples\rulesets にそれぞれ保存されています。

次に例を示します。

ルールグループをルールセット内の position 4 に追加する場合 :

```
zman rsar ruleset2 rulegroup 4 C:\RuleGroups.xml
```

ルールセットの 3 番目のルールグループにルールを追加する場合 :

```
zman rsar ruleset2 rule 3 C:\Rules.xml
```

ルールセットの最後のルールグループにルールを追加する場合 :

```
zman rsar ruleset2 rule last C:\Rules.xml
```

ルールセット内の新しいルールグループとしてルールを追加する場合 :

```
zman rsar ruleset2 rule makenew C:\Rules.xml
```

#### **ruleset-create (rsc) (デバイスタイプ) (ルールセット名) (ルールファイル) [位置] [オプション]**

登録キーを持たないデバイスを登録する場合に、適用する新しいルールセットを作成します。

(デバイスタイプ) - 登録デバイスのタイプ。有効な値は、server および workstation です。

(ルールセット名) - ルールセットの名前を指定します。

(ルールファイル) - 新しいルールセットのルールまたはルールグループの情報が含まれる XML ファイルのパス。XML ファイルは、ruleset-export-rules-to-file コマンドを使用して、既存のルールセットのルールをエクスポートすることによって作成できます。または、ルールのみを追加する場合は (結合子は AND)、古い形式も使用できます。

ファイル形式の例については、AddRulesToRuleSet.txt ファイルを参照してください。このファイルは、Linux サーバでは /opt/novell/zenworks/share/zman/samples/rulesets、Windows サーバでは *Installation\_directory*:\Novell\Zenworks\share\zman\samples\ rulesets にそれぞれ保存されています。

(位置) - 複数のルールセット内でのルールセットの位置。最初のルールセットは position 1 にあります。

次のオプションを受け入れます。

-f, --devicefolder=[デバイスフォルダ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。登録デバイスはこのフォルダに置かれます。

-g, --devicegroup=[デバイスグループパス][...] - デバイスグループのパスのカンマ区切りリスト。指定したパスは、指定したデバイスタイプのルートフォルダを基準にした相対パスである必要があります。登録デバイスは、これらのデバイスグループのメンバーになります。

--desc=[説明] - ルールセットの説明。

--site=[サイト] - デバイスが存在するサイト。登録デバイスにはこのサイト情報が入力されます。

--dept=[部署] - デバイスが使用される部署。登録デバイスにはこの部署情報が入力されます。

--loc=[場所] - デバイスの物理的な場所。登録デバイスにはこの場所情報が入力されます。

次に例を示します。



サーバ登録用のルールセットを作成する場合：

```
zman rsc server ruleset1 C:\RuleSet1.xml --devicefolder serverfolder --desc "サーバをインポートする  
ルール" --site "構築A" --dept Finance --loc Brazil
```

このルールセット内に定義されているルールに対して True となる登録デバイスは、指定したサイト、部署、および場所の値で /Devices/Servers/serverfolder フォルダに配置されます。

特定のワークステーショングループのメンバとなる可能性があるワークステーションを登録するためのルールセットを作成する場合：

```
zman rsc workstation ruleset3 --devicegroup wsgroup,wsgroup1,wsfolder/wsgroup2
```

デバイスグループとルールセットの関連付けは、ルールセットの作成後に ruleset-add-device-group コマンドを使用して行うこともできます。

#### **ruleset-copy (rscp) (ルールセット名または位置) (新しい名前) [位置]**

ルールセットをコピーします。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

(新しい名前) - コピーしたルールセットに付ける新しい名前。

(位置) - 複数のルールセット内でのルールセットの位置。最初のルールセットは position 1 にあります。

次に例を示します。

ルールセットを最初の位置にコピーする場合：

```
zman rscp ruleset3 newruleset 1
```

注：指定された新しい位置が 0、またはルールセット数より大きな値の場合、ルールセットはリストの先頭または末尾にそれぞれコピーされます。

#### **ruleset-export-rules-to-file (rsertf) (ルールセット名または位置) (XML ファイルパス)**

ルールセットのルールをファイルにエクスポートします。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

(XML ファイルパス) - ルールが XML 形式で保存されるファイルを指定します。

次に例を示します。

ルールセットからルールをエクスポートする場合：

```
zman rsertf ruleset1 C:\ExportedRuleSet.xml
```

#### **ruleset-delete (rsd) (ルールセット名または位置)**

ルールセットを削除します。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

#### **ruleset-info (rsi) (ルールセット名または位置)**

ルールセットに関する詳細情報を表示します。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

次に例を示します。

ルールセット MyRegRule に関する詳細情報を表示する場合：

```
zman rsi MyRegRule
```

#### **ruleset-list (rsl)**

ルールセットすべてを一覧表示します。

#### **ruleset-list-device-groups (rsl dg) (ルールセット名または位置)**

ルールセットに関連付けられるデバイスグループを一覧表示します。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

#### **ruleset-move (rsmv) (ルールセット名または位置) (新しい位置)**

ルールセットの位置を変更します。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

(新しい位置) - ルールセットの移動先の新しい位置を指定します。

次に例を示します。

```
zman rsmv ruleset3 1
```

注：指定された新しい位置が 0、またはルールセット数より大きな値の場合、ルールセットはリストの先頭または末尾にそれぞれコピーされます。

#### **ruleset-remove-device-groups (rsrdg) (デバイスタイプ) (ルールセット名または位置) (デバイスグループパス) [...] [オプション]**

デバイスグループとルールセットの関連付けを削除します。

(デバイスタイプ) - 登録デバイスのタイプ。有効な値は、server および workstation です。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

(デバイスグループパス) - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスグループの相対パス。

次のオプションを受け入れます。

-f, --folder=[デバイスフォルダ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。このオプションを指定すると、デバイスオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のデバイスオブジェクトを指定するのに使用できます。

#### **ruleset-remove-rule (rsrr) (ルールセット名または位置) (ルールの位置) [...]**

ルールセットからルールまたはルールグループを削除します。

(ルールセット名または位置) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

( ルールの位置 ) - 削除するルールの位置を指定します。最初のルールグループは G1、G1 内の最初の 2 つのルールは G1:1,2 で示されます。位置は、グループ、またはグループ内の特定のルールを示します。

次に例を示します。

ルールセットの 2 番目のルールグループを削除する場合：

```
zman rsrr ruleset2 G2
```

特定のルールグループおよび異なるルールグループの特定のルールを削除する場合：

```
zman rsrr ruleset2 G1:3 G2:2,5,6 G3 G6:4,9 G8
```

#### **ruleset-rename (rsr) ( ルールセット名または位置 ) ( 新しい名前 )**

ルールセットの名前を変更します。

( ルールセット名または位置 ) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

( 新しい名前 ) - ルールセットに付ける新しい名前を指定します。

#### **ruleset-update (rsu) ( デバイス ) ( ルールセット名または位置 ) [ オプション ]**

ルールセットをアップデートします。

( デバイスタイプ ) - 登録デバイスのタイプ。有効な値は、server および workstation です。

( ルールセット名または位置 ) - ルールセットの名前またはルールセット内での位置を指定します。最初のルールセットは position 1 にあります。

次のオプションを受け入れます。

-f, --devicefolder=[ デバイスフォルダ ] - 指定されたデバイスタイプのルートフォルダを基準にした、デバイスフォルダの相対パス。登録デバイスはこのフォルダに置かれます。

--desc=[ 説明 ] - ルールセットの説明。

--site=[ サイト ] - デバイスが存在するサイト。登録デバイスにはこのサイト情報が入力されます。

--dept=[ 部署 ] - デバイスが使用される部署。登録デバイスにはこの部署情報が入力されます。

--loc=[ 場所 ] - デバイスの物理的な場所。登録デバイスにはこの場所情報が入力されます。

次に例を示します。

サーバを登録するルールセットを更新する場合：

```
zman rsu server ruleset1 --devicefolder serverfolder1 --desc "サーバをインポートするルール" --site "構築B" --dept Transport --loc Brazil
```

インポートされたサーバがメンバになるサーバグループを変更するには、ruleset-add-device-group コマンドおよび ruleset-remove-device-group コマンドを使用します。

デバイスフォルダをワークステーションフォルダからサーバルートフォルダに更新する場合：

zman rsu server ruleset2

このコマンドによって、登録デバイスが配置されるフォルダが、/Devices/Workstations から /Devices/Servers に変更されます。

## サテライトコマンド

これらのコマンドは、サテライトを表示、作成、および削除する場合に使用します。サテライトコマンドは `satellite-server-` で開始します。

---

**注:** ZENworks 2020 以降のリリースから、32 ビットデバイスのサテライトサーバ役割へのプロモーションまたは既存の 32 ビットサテライトサーバへの役割の追加はサポートされていません。

---

- ◆ 93 ページの「`satellite-server-add-content-replication-setting` (ssacrs) (デバイスへのパス) (コンテンツタイプ) [-s|--scheduleFileName=ファイルパス] [-t|--throttleRate=スロットルレート] [-d|--duration=期間] [--noSchedule]」
- ◆ 93 ページの「`satellite-server-add-imagingrole|ssaimg` (デバイスへのパス) [親プライマリサーバ] [-p|--proxydhcp=プロキシDHCPサービス] [-s|--serverPort=サテライトサーバポート] (-r|--Content-Replication-Schedule=サテライトサーバのコンテンツレプリケーションスケジュール) [--force-port-change] [--force-parent-change]」
- ◆ 94 ページの「`satellite-server-create` (ssc) (デバイスへのパス) ((役割), (役割), ... (役割)) [親プライマリサーバ] [サテライトサーバポート] [サテライトサーバセキュアHTTPSポート] [--force-port-change] [--force-parent-change]」
- ◆ 95 ページの「`satellite-server-update-joinproxy-settings` (ssujs) (デバイスへのパス) [-j|--jpport=jpport] [-m|--maxconnections=maxConnections] [-c|--connectionCheckInterval=connectionCheckInterval]」
- ◆ 96 ページの「`satellite-server-delete` (ssd) (デバイスへのパス) [(役割), (役割), ... (役割)] [-f|--force]」
- ◆ 96 ページの「`satellite-server-configure-authentication-user-source` (sscaus) (デバイスへのパス) (ユーザソース) [ユーザ接続, ユーザ接続, ..., ユーザ接続]」
- ◆ 96 ページの「`satellite-server-edit-imagingrole` (sseimg) (デバイスへのパス) (プロキシDHCPサービスに対するアクション)」
- ◆ 97 ページの「`satellite-server-export-content` (ssec) (デバイスへのパス) (ディレクトリのエクスポート) [オプション]」
- ◆ 97 ページの「`satellite-server-export-content-replication-setting-schedule` (ssecrss) (デバイスへのパス) (コンテンツタイプ) (-s | --scheduleFileName = ファイルパス)」

- ◆ 97 ページの「satellite-server-list-authentication-settings (sslas) (デバイスへのパス)」
- ◆ 97 ページの「satellite-server-list-imagingservers (sslimg)」
- ◆ 97 ページの「satellite-server-list-content-replication-settings (sslcrs) (デバイスへのパス)」
- ◆ 97 ページの「satellite-server-list-roles (sslr) [管理対象デバイス]」
- ◆ 98 ページの「Satellite-server-list-servers (ssls)」
- ◆ 98 ページの「satellite-server-remove-content-replication-setting (ssrcrs) (デバイスへのパス) (コンテンツタイプ)」
- ◆ 98 ページの「satellite-server-remove-imagingle (ssrimg) (デバイスへのパス) [-r|--removeImageFiles]」
- ◆ 98 ページの「satellite-server-update-content-replication-setting (ssucrs) (デバイスへのパス) (コンテンツタイプ) (-s | --scheduleFileName= ファイルパス) [オプション]」

**satellite-server-add-content-replication-setting (ssacrs) (デバイスへのパス) (コンテンツタイプ) [-s|--scheduleFileName= ファイルパス] [-t|--throttleRate= スロットルレート] [-d|--duration= 期間] [--noSchedule]**

コンテンツタイプ複製スケジュールとスロットル設定をサテライトデバイスに追加します。

(デバイスへのパス) - 複製設定の追加先のサテライトデバイスの完全名。

(コンテンツタイプ) - サテライトデバイスの追加先の複製設定のコンテンツタイプの名前。

-s, --scheduleFileName=< ファイルパス > - XML 形式で保存された複製設定のスケジュールデータを含むファイルへのフルパス。

-t, --throttleRate=< スロットルレート > - サテライトデバイスに追加する複製設定のスロットルレート (kbps)。

-d, --duration=< 期間 > - コンテンツ複製の期間 (分単位)。

--noSchedule - コンテンツ複製のスケジュールを指定したくない場合は、このオプションを使用します。

**satellite-server-add-imagingle|ssaimg) (デバイスへのパス) [親プライマリサーバ] [-p|--proxydhcp= プロキシ DHCP サービス] [-s|--serverPort= サテライトサーバポート] [-r|--Content-Replication-Schedule= サテライトサーバのコンテンツレプリケーションスケジュール] [--force-port-change] [--force-parent-change]**

管理対象デバイスを、イメージングサーバの役割を持つサテライトサーバに昇格します。管理デバイスが既にサテライトサーバとして機能している場合、このコマンドは、イメージングサーバの役割を追加します。

次のオプションを受け入れます。

(デバイスへのパス) - /Devices に対する、管理対象デバイスの相対パスまたは管理対象デバイスの絶対パス。

[親プライマリサーバ] - /Devices に対するプライマリサーバの相対パス、または新しいサテライトサーバへの親サーバとしての役割を果たすプライマリサーバの絶対パス。

- p, --proxydhcp=< プロキシ DHCP サービス > - プロキシ DHCP サービスを開始または停止します。有効な値は、start および stop です。
- s, --serverPort=< サテライトサーバポート > - 管理対象デバイスの要求をリスンする際にサテライトサーバが使用する必要があるポート。指定しない場合、ポート 80 が使用されます。
- r, --Content-Replication-Schedule=< サテライトサーバのコンテンツレプリケーションスケジュール > - コンテンツレプリケーションスケジュールを含む XML ファイル。コンテンツレプリケーションスケジュールで、親プライマリサーバからサテライトサーバのコンテンツを更新する頻度を、日数、時間、および分で入力します。XML 形式のテンプレートについては、Linux サーバでは /opt/novell/zenworks/share/zman/samples/schedules、Windows サーバでは < インストールディレクトリ >:\Novell\Zenworks\share\zman\samples\schedules にある ContentReplicationSchedule.xml を参照してください。
- force-port-change - 管理対象デバイスがすでにサテライトサーバとして機能している場合は、このオプションによって、このデバイス上の全サテライトサーバコンポーネントが管理対象デバイス要求をリスンするポートの変更が許可されます。
- force-parent-change - 管理対象デバイスがすでにサテライトサーバとして機能している場合は、このオプションによって、デバイス上の全サテライトサーバコンポーネントで親プライマリサーバの変更が許可されます。

---

**注:** 管理対象デバイスをサテライトに昇格させる前に、管理対象デバイスにインストールされている ZENworks のバージョンがプライマリサーバにインストールされているバージョンと同一であることを確認してください。また、以下のデバイスはサテライトとして昇格することはできません。

- 以前のバージョンの ZENworks Agent がインストールされている管理対象デバイス。
  - ZENworks テストデバイス。
- 

**satellite-server-create (ssc) ( デバイスへのパス ) (( 役割 ), ( 役割 ), ... ( 役割 )) [ 親プライマリサーバ ] [ サテライトサーバポート ] [ サテライトサーバセキュア HTTPS ポート ] [--force-port-change] [--force-parent-change]**

サテライトの役割を設定できます。  
次のオプションを受け入れます。

( デバイスへのパス ) - サテライトとして作成する管理対象デバイスの名前。

( 役割 ) - 管理対象デバイスに追加するサテライトの役割。役割には、認証、コレクション、コンテンツ、および JoinProxy が含まれます。イメージングサテライトの役割を割り当てるか、または管理対象デバイスをイメージングサーバの役割を持つサテライトに昇格させる場合、このコマンドではなく、zman ssaimg コマンドを使用する必要があります。JoinProxy 役割設定を再構成する場合は、zman ssujs コマンドを使用する必要があります。

[ 親プライマリサーバ ] - サテライトサーバがコレクションデータをロールアップする先、またはコンテンツの取得元のプライマリサーバの完全な名前。管理対象デバイスがすでにサテライトの場合、このフィールドはオプションです。

[ サテライトサーバポート ] - サテライトサーバが、コレクションロールアップ要求をリスンするため、または親プライマリサーバからコンテンツを取得するために使用するポート。指定しない場合は、ポート 80 が使用されます。

- [サテライトサーバセキュア HTTPS ポート] - サテライトが認証セキュア HTTPS 要求をリスンするために使用するポート。指定しなかった場合、ポート 443 が使われます。
- [--force-port-change] - 管理デバイスがすでにサテライトとして機能している場合、このオプションを指定すると、このデバイス上のすべてのサテライトコンポーネントが要求をリスンする、またはコンテンツ取得のために使用するポートを変更することができます。
- [--force-parent-change] - 管理デバイスがすでにサテライトとして機能している場合、このオプションを指定すると、デバイス上のすべてのサテライトコンポーネントのプライマリサーバの親を変更することができます。

---

**注:** 管理対象デバイスをサテライトに昇格させる前に、管理対象デバイスにインストールされている ZENworks のバージョンがプライマリサーバにインストールされているバージョンと同一であることを確認してください。また、以下のデバイスはサテライトとして昇格することはできません。

- ◆ 以前のバージョンの ZENworks Agent がインストールされている管理対象デバイス。
- ◆ ZENworks テストデバイス。

---

非ルートユーザは、ZENworks ユーザグループに追加された場合のみ、Linux プライマリサーバ上でこのコマンド (zman ssc) を問題なく実行できます。非ルートユーザを ZENworks ユーザグループに追加するには、`usermod -A non-root_username` コマンドを使用します。

**satellite-server-update-joinproxy-settings (ssujs) (デバイスへのパス) [-j|--jport=jport] [-m|--maxconnections=maxConnections] [-c|--connectionCheckInterval=connectionCheckInterval]**

サテライトサーバで JoinProxy 設定を更新します。Windows または Linux の管理対象デバイスのみを JoinProxy サテライト役割に昇格できます。デフォルトでは、プライマリサーバには JoinProxy 役割が割り当てられています。

次のオプションを受け入れます。

(デバイスへのパス) - JoinProxy 設定を更新するサテライトサーバの完全名。

-j, --jport=<jport> - JoinProxy が接続を待機するポート。

-m, --maxconnections=<maxConnections> - JoinProxy への接続が許可されたデバイスの最大数。

-c, --connectionCheckInterval=<connectionCheckInterval> - JoinProxy が、デバイスが接続されたかどうかを定期的に確認する間隔。

---

**注:** ssujs コマンドを使用する場合、JoinProxy サテライトサーバは次のデフォルト値によって構成されます。

- ◆ ポート : 7019
- ◆ 最大接続数 : 100
- ◆ 最大接続間隔 : 1 分

### **satellite-server-delete (ssd) ( デバイスへのパス ) [( 役割 ),( 役割 ),...( 役割 )][ -f|--force ]**

デバイスからサテライトの役割を削除します。すべての役割が削除されるとデバイスは、ZENworks コントロールセンター内のサーバ階層リストから自動的に除去されます。

管理ゾーンが ZENworks プライマリサーバおよび ZENworks Configuration Management サテライトで構成されている場合は、サテライトから個別の役割を削除することはできません。サテライトを管理対象デバイスに降格させることしかできません。任意のバージョンの ZENworks Agent がインストールされたサテライトを削除できます。

次のオプションを受け入れます。

( デバイスへのパス ) - サテライトの役割を削除するサテライトサーバの完全名。

[ 役割 ] - デバイスから削除するサテライトの役割。役割を指定しない場合には、デバイスからすべてのサテライトの役割が削除されます。認証、コレクション、コンテンツ、および JoinProxy 役割を削除できます。イメージングの役割を削除するには、このコマンドではなく、satellite-server-remove-imagingrole (ssrimg) コマンドを使用します。

[ -f|--force ] - 指定された既存の役割をすべて削除しながら、サテライトサーバから管理対象デバイスへの降格を強制的に適用します。このオプションは、サテライトサーバが常にオフラインで、ZENworks サーバと通信できない場合にのみ使用してください。

### **satellite-server-configure-authentication-user-source (sscaus) ( デバイスへのパス ) ( ユーザソース )**

**[ ユーザ接続 , ユーザ接続 , ..., ユーザ接続 ]**

認証の役割を持つサテライトが使用する、ユーザソース接続を設定します。このコマンドを複数回実行して、同じサテライト上の複数のユーザソースに接続を設定し、コマンドの実行ごとに、それぞれ別のユーザソースを指定します。このコマンドを接続なしで実行し、指定したユーザソースをサテライトの認証の役割から削除します。

次のオプションを受け入れます。

( デバイスへのパス ) - 認証の役割のユーザソースを設定するサテライトの完全名。

( ユーザソース ) - 設定するユーザソースの名前。

[ ユーザ接続 ] - 指定されたユーザソースに対して、このサテライトがユーザを認証するとき使用する順に一覧されたユーザ接続。

### **satellite-server-edit-imagingrole (sseimg) ( デバイスへのパス ) ( プロキシ DHCP サービスに対するアクション )**

イメージングサテライトの Proxy DHCP Service を開始または停止します。

( デバイスへのパス ) - /Devices/Workstations に対するデバイスの相対パス。指定するデバイスは、イメージングの役割を持つサテライトデバイスである必要があります。

( プロキシ DHCP サービスに対するアクション ) - プロキシ DHCP サービスを開始または停止します。有効な値は、start および stop です。



### **satellite-server-export-content (ssec) ( デバイスへのパス )( ディレクトリのエクスポート )[ オプション ]**

コンテンツファイルをコンテンツリポジトリからエクスポートし、サテライトデバイスのコンテンツリポジトリに手動でインポートできるようにします。コンテンツを管理対象デバイスのコンテンツリポジトリにインポートするには、`zac cdp-import-content (cic)` コマンドを使用します。

( デバイスへのパス ) - コンテンツファイルをエクスポートするサテライトデバイスの完全名。

( ディレクトリのエクスポート ) - コンテンツファイルのエクスポート先のディレクトリの完全パス。

次のオプションを受け入れます。

`-a, --exportAll` - サテライトデバイスに割り当てられたすべてのコンテンツをエクスポートします ( デフォルトで、不足しているコンテンツのみがエクスポートされます )。

### **satellite-server-export-content-replication-setting-schedule (ssecrssh) ( デバイスへのパス )( コンテンツタイプ )( -s | --scheduleFileName = ファイルパス )**

コンテンツタイプ複製スケジュールをサテライトデバイスからファイルにエクスポートします。

( デバイスへのパス ) - 複製設定スケジュールのエクスポート元のサテライトデバイスの完全名。

( コンテンツタイプ ) - サテライトデバイスからエクスポートする複製設定のコンテンツタイプの名前。

`-s, --scheduleFileName =< ファイルパス >` - スケジュールデータのエクスポート先にするファイルのフルパス。

### **satellite-server-list-authentication-settings (sslas) ( デバイスへのパス )**

サテライトデバイスの認証の役割設定を一覧表示します。

( デバイスへのパス ) - 認証の役割設定の表示元のサテライトデバイスの完全名。

### **satellite-server-list-imagingservers (sslimg)**

イメージングの役割を持つサテライトを一覧表示します。

### **satellite-server-list-content-replication-settings (sslcrs) ( デバイスへのパス )**

サテライトのコンテンツ複製スケジュールおよびスロットルの設定を、コンテンツタイプごとに一覧表示します。

( デバイスへのパス ) - 複製設定の表示元のサテライトデバイスの完全名。

### **satellite-server-list-roles (sslr) [ 管理対象デバイス ]**

以前に定義されたサテライトの役割を一覧表示します

次のオプションを受け入れます。

[ 管理対象デバイス ] - サテライトの役割を一覧表示する管理対象デバイスの完全な名前。指定しない場合には、使用可能なすべてのサテライトの役割が表示されます。

### Satellite-server-list-servers (ssls)

サテライトと役割を一覧表示します。

### satellite-server-remove-content-replication-setting (ssrcrs) ( デバイスへのパス )( コンテンツタイプ )

コンテンツタイプ複製スケジュールとスロットル設定をサテライトデバイスから削除します。

( デバイスへのパス ) - 複製設定の削除元のサテライトデバイスの完全名。

( コンテンツタイプ ) - サテライトデバイスから削除する複製設定のコンテンツタイプ  
の名前。

### satellite-server-remove-imagingrole (ssrimg) ( デバイスへのパス ) [-r|--removeImageFiles]

管理対象デバイスから、サテライトのイメージングサーバの役割を削除します。

管理ゾーンが ZENworks プライマリサーバおよび ZENworks Configuration Management サテライトで構成されている場合は、サテライトから個別の役割を削除することはできません。サテライトを管理対象デバイスに降格させることしかできません。任意のバージョンの ZENworks Agent がインストールされたサテライトを削除できます。

( デバイスへのパス ) - /Devices/Workstations に対する管理対象デバイスの相対パス。指定するデバイスは、イメージングの役割を持つサテライトデバイスである必要があります。

-r, --removeImageFiles - 指定したデバイスからイメージファイルを削除します。

### satellite-server-update-content-replication-setting (ssucrs) ( デバイスへのパス )( コンテンツタイプ ) (-s| --scheduleFileName= ファイルパス ) [ オプション ]

コンテンツタイプ複製スケジュールとスロットル設定をサテライトデバイスに更新します。

( デバイスへのパス ) - 複製設定の更新先のサテライトデバイスの完全名。

( コンテンツタイプ ) - サテライトデバイスの更新先の複製設定のコンテンツタイプ  
の名前。

-s, --scheduleFileName=( ファイルパス ) - XML 形式で保存されたレプリケーション設定  
のスケジュールデータを含むファイルへのフルパス。

次のオプションを受け入れます。

-d, --duration=< 期間 > - コンテンツ複製の期間 ( 分単位 )。

-t, --throttleRate=< スロットルレート > - サテライトデバイスに更新する複製設定の  
スロットルレート (kbps)。

## サーバコマンド

サーバを管理するコマンドです。サーバコマンドは、長い形式では server- から始まり、短い形式では s から始まります。

次のコマンドはすべて、「**グローバルオプション**」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- 101 ページの「server-add-bundle (sab) (サーバオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]」
- 102 ページの「server-add-certificate (sacert) (サーバオブジェクトパス) (証明書ファイルパス) [-f]」
- 103 ページの「server-add-policy (sap) (サーバオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- 103 ページの「server-copy-relationships (scr) (サーバパス) (サーバオブジェクトパス) [...] [オプション]」
- 104 ページの「server-delete (sd) (サーバオブジェクトパス) [...] [オプション]」
- 104 ページの「server-folder-create (sfc) [フォルダ名] [親フォルダ] [オプション]」
- 104 ページの「server-group-add (sga) (サーバグループパス) [サーバパス] [...] [オプション]」
- 105 ページの「server-group-create (sgc) [グループ名] [親フォルダ] [オプション]」
- 105 ページの「server-group-members (sgm) (サーバグループパス) [オプション]」
- 105 ページの「server-group-remove (sgr) (サーバグループパス) (サーバパス) [...] [オプション]」
- 105 ページの「server-info (si) (サーバパス)」
- 105 ページの「server-list (sl) [オプション] [フォルダ]」
- 106 ページの「server-list-bundles (slb) (サーバオブジェクトパス) [オプション]」
- 106 ページの「server-list-groups (slg) (サーバパス) [オプション]」
- 106 ページの「server-list-policies (slp) (サーバオブジェクトパス) [オプション]」
- 106 ページの「server-move (smv) (サーバオブジェクトパス) [宛先フォルダパス]」
- 106 ページの「server-refresh (srf) (サーバオブジェクトパス) [...] [オプション]」
- 107 ページの「server-remove-bundle (srb) (サーバオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]」

- ◆ 107 ページの「server-remove-policy (srp) (サーバオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- ◆ 107 ページの「server-rename (sr) (サーバオブジェクトパス) (新しい名前)」
- ◆ 107 ページの「server-reorder-bundles (srob) (サーバオブジェクトパス) (現在の位置) (新しい位置)」
- ◆ 107 ページの「server-reorder-policies (srop) (サーバオブジェクトパス) (現在の位置) (新しい位置)」
- ◆ 107 ページの「server-retire (sret) (サーバオブジェクトパス)」
- ◆ 107 ページの「server-set-as-test (ssat) (サーバオブジェクトパス)」
- ◆ 108 ページの「server-role-zookeeper-add-server (srzas)」
- ◆ 108 ページの「server-role-zookeeper-remove-server (srzrs)」
- ◆ 108 ページの「server-role-zookeeper-list-cluster (srzlc)」
- ◆ 108 ページの「server-role-zookeeper-update-cluster (srzuc) [...] [オプション]」
- ◆ 108 ページの「server-role-kafka-configure-cluster (srkcc) [...] [オプション]」
- ◆ 109 ページの「server-role-kafka-add-broker (srkab)」
- ◆ 109 ページの「server-role-kafka-remove-broker (srkrb)」
- ◆ 109 ページの「server-role-kafka-reconfig-broker (srkrbc)」
- ◆ 109 ページの「server-role-kafka-update-cluster (srkuc) [...] [オプション]」
- ◆ 110 ページの「server-role-kafka-list-cluster (srklc)」
- ◆ 110 ページの「server-role-kafka-list-connectors (srklcn)」
- ◆ 110 ページの「server-role-kafka-restart-connectors (srkrcn) [...] [オプション]」
- ◆ 110 ページの「server-role-kafka-get-connector-config (srkgcc) [...] [オプション]」
- ◆ 110 ページの「server-role-kafka-reconfigure-connectors (srkccn) [...] [オプション]」
- ◆ 110 ページの「server-role-vertica-create-cluster (srvcc) [...] [オプション]」
- ◆ 111 ページの「server-role-vertica-add-server (srvas)」
- ◆ 111 ページの「server-role-vertica-remove-server (srvrs)」
- ◆ 111 ページの「server-role-vertica-list-cluster (srvlc)」
- ◆ 111 ページの「server-role-vertica-update-cluster (srvuc)」
- ◆ 111 ページの「server-role-vertica-get-credentials (srvgc)」
- ◆ 111 ページの「server-role-vertica-prepare-server (srvps)」

- ◆ 111 ページの「`server-set-as-nontest (ssan)` (サーバオブジェクトパス)」
- ◆ 112 ページの「`server-statistics (sst)` (サーバパス)」
- ◆ 112 ページの「`server-unretire (suret)` (サーバオブジェクトパス)」
- ◆ 112 ページの「`server-view-all-bundles-status (svabs)` (サーバパス) [オプション]」
- ◆ 112 ページの「`server-view-all-policies-status (svaps)` (サーバパス) [オプション]」
- ◆ 112 ページの「`server-wakeup (sw)` (サーバオブジェクトパス) [...] [オプション]」

**server-add-bundle (sab)** (サーバオブジェクトパス)(バンドルまたはバンドルグループ)[...][オプション]

バンドルをサーバオブジェクトに割り当てます。

次のオプションを受け入れます。

- f, --folder=[バンドルフォルダ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。
- l, --icon-location=[アプリケーションロケーション XML ファイル] - バンドルアプリケーション用のアイコンを配置する場所を含む XML ファイル。XML ファイル形式については、IconLocation.xml を参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/bundles、Windows サーバでは *Installation\_directory*:\Novell\Zenworks\share\zman\samples\ bundles に保存されています。
- d, --distribution-schedule=[配布スケジュール XML ファイル] - 配布スケジュールを含む XML ファイル。
- l, --launch-schedule=[起動スケジュール XML ファイル] - 起動スケジュールを含む XML ファイル。
- a, --availability-schedule=[可用性スケジュール XML ファイル] - 可用性スケジュールを含む XML ファイル。スケジュール XML ファイルテンプレートについては、XML ファイルを参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/schedules、Windows サーバでは *Installation\_directory*:\Novell\Zenworks\share\zman\samples\schedules に保存されています。
- D, --dry-run - 要求されたアクションをテストして表示しますが、実際には実行しません。
- i, --install-immediately - 配布後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- L, --launch-immediately - インストール後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。

- n, --distribute-now - バンドルをただちに配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule および --distribute-on-device-refresh オプションは無視されます。--distribute-now、--distribute-on-device-refresh、および --distribution-schedule オプションは配布スケジュールの設定に使用し、同時に使用することはできません。最初に --distribute-now オプションが考慮され、次に --distribute-on-device-refresh オプションが考慮され、最後に --distribution-schedule が考慮されます。
- r, --distribute-on-device-refresh - デバイス更新時にバンドルを配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule オプションは無視されます。
- s, --launch-on-device-refresh - デバイス更新時にバンドルを起動するよう起動スケジュールを設定します。このオプションを指定すると、--launch-schedule オプションは無視されます。
- w, --wakeup-device-on-distribution - バンドルの配布中にデバイスがシャットダウンした場合、Wake-On-LAN を使用してデバイスを起動します。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- B, --broadcast=[ブロードキャストアドレス][...] - Wake-On-LAN マジックパケットをブロードキャストするのに使用するアドレスのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。有効な IP アドレスが、有効な値です。
- S, --server=[/デバイス/基準にしたプライマリサーバオブジェクトまたはプロキシサーバオブジェクトの相対パス][...] - デバイスを起動するのに使用するプライマリまたはプロキシサーバのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。
- C, --retries=[再試行回数] - デバイスに Wake-On-LAN マジックパケットを送信する回数。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 0 ~ 5 の範囲である必要があります。デフォルト値は「1」です。
- T, --timeout=[再試行間隔] - 2 回の再試行の間隔。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 2 ~ 10 の範囲である必要があります (単位は分)。デフォルト値は「2」です。

**server-add-certificate (sacert) (サーバオブジェクトパス)(証明書ファイルパス)[-f]**

ZENworks 11 より前は、このコマンドを使用して、失効間近の既存の証明書を置き換えて、サーバに新しい有効な証明書を追加していました。ZENworks 11 SP4 の SSL 管理機能の導入により、このコマンドは廃止されました。

データベース内に存在しない証明書を追加するには、強制 (-f, --force) オプションとともにこのコマンドを使用できます。ただし、データベース内にすでにアクティブな証明書がある場合は、このコマンドを使用できません。

(サーバオブジェクトパス) - /デバイス/サーバを基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の相対パス。

(証明書ファイルパス)- DER形式の証明書ファイルのパス。

-f, --force - アクティブな証明書として証明書を強制的に追加します。このオプションは、アクティブな証明書がデータベース内で使用できない場合にのみ機能します。

**server-add-policy (sap) (サーバオブジェクトパス)(ポリシーまたはポリシーグループ)[...][オプション]**

ポリシーをサーバオブジェクトに割り当てます。

次のオプションを受け入れます。

-c, --conflicts=[ポリシー競合解決順序]- ポリシー競合の解決方法を決定します。有効な値は、userlast または 1、devicelast または 2、deviceonly または 3、useronly または 4 です。userlast の場合、デバイスに関連付けられたポリシーがまず適用され、それからユーザに関連付けられたポリシーが適用されます。devicelast の場合、ユーザに関連付けられたポリシーがまず適用され、それからデバイスに関連付けられたポリシーが適用されます。deviceonly の場合、ユーザに関連付けられたポリシーは無視されます。useronly の場合、デバイスに関連付けられたポリシーは無視されます。このオプションを指定しなかった場合には、userlast がデフォルト値として使用されます。

-e, --enforce-now - 割り当てられているすべてのデバイスに対してポリシーをすぐに強制します。

-f, --folder=[ポリシーフォルダ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

**server-copy-relationships (scr) (サーバパス)(サーバオブジェクトパス)[...][オプション]**

サーバから複数のサーバに関係をコピーします。サーバの関係には、バンドル割り当て、ポリシー割り当て、およびグループメンバーシップが含まれます。

次のオプションを受け入れます。

-r, --relationship-type: コピーする関係タイプ。有効な値は、[bundles]、[policies]、および [groups] です。オプション [bundles] は、ソースデバイスの関連付けられたすべてのバンドル割り当てをコピーします。オプション [policies] は、ソースデバイスの関連付けられたすべてのポリシー割り当てをコピーします。オプション [groups] は、ソースデバイスに関連付けられたグループのメンバーとしてターゲットデバイスを追加します。

-a, --assignment-options: 割り当てオプション。有効な値は、[assign-to-group-members]、[assign-to-folder-members]、および [assign-to-folder-members-recursively] です。オプション [assign-to-group-members] は、ターゲットグループのメンバーに割り当てをコピーします。オプション [assign-to-folder-members] は、ターゲットフォルダ内のデバイスに割り当てをコピーします。オプション [assign-to-folder-members-recursively] は、ターゲットフォルダおよびそのサブフォルダ内のデバイスに割り当てをコピーします。

-c, --conflict-resolution: 既存の割り当ての競合解決オプション。有効な値は、[delete-existing-assignments] および [replace-existing-assignments] です。オプション [delete-existing-assignments] は、ターゲットデバイスの既存のすべての割り当てを削除します。オプション [replace-existing-assignments] は、選択した割り当てでターゲットデバイスの既存の割り当てを置き換えます。

-g, --group-membership: ターゲットオブジェクトタイプ、グループおよびフォルダのグループメンバーシップオプション。有効な値は、[add-group-members]、[add-folder-members]、[add-folder-members-recursively]、および [delete-existing-group-membership] です。オプション [add-group-members] は、ソースデバイスがメンバーであるグループのメンバーとしてターゲットグループのメンバーを追加します。オプション [add-folder-members] は、ソースデバイスがメンバーであるグループのメンバーとしてターゲットフォルダ内にデバイスを追加します。オプション [add-folder-members-recursively] は、ソースデバイスがメンバーであるグループのメンバーとしてターゲットフォルダとそのサブフォルダ内にデバイスを追加します。オプション [delete-existing-group-membership] は、ターゲットデバイスがメンバーである既存のグループメンバーシップを削除します。

次に例を示します。

ソースサーバのすべての関係を複数のサーバタイプにコピーするには : zman server-copy-relationships serverfolder/sourceserver --relationship-type=bundles,policies,groups serverfolder/serverfolder/serverfolder1 serverfolder/servergroup

ソースサーバのポリシー割り当てをターゲットサーバフォルダ内のサーバに再帰的にコピーするには : zman scr serverfolder/sourceserver --relationship-type=policies serverfolder1 serverfolder2 -assignment-options=assign-to-folder-members-recursively

ソースサーバから割り当てをコピーしているときに、ターゲットサーバの競合バンドル割り当てを置き換えるには : zman scr serverfolder/sourceserver -r=bundles serverfolder/server1 -conflict-resolution=replace-existing-assignments

ソースサーバのグループにターゲットサーバグループのサーバを追加するには : zman scr serverfolder/sourceserver -r=groups servergroup -group-membership=add-group-members

#### **server-delete (sd) (サーバオブジェクトパス) [...][オプション]**

1 つまたは複数のサーバオブジェクトを削除します。

(サーバオブジェクトパス) [...]- / デバイス / サーバを基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の相対パス。引用符で囲む限り、ワイルドカード \* を使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

次のオプションを受け入れます。

-r, --recursive - 再帰的に、フォルダ内にオブジェクトを削除します。

-f, --folder=[サーバフォルダ]- / デバイス / サーバを基準にした、サーバフォルダの相対パス。このオプションを指定すると、サーバオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のサーバオブジェクトを指定するのに使用できます。

#### **server-folder-create (sfc) [フォルダ名][親フォルダ][オプション]**

サーバを格納するための新しいフォルダを作成します。

次のオプションを受け入れます。

--desc=[説明]- フォルダの説明。

#### **server-group-add (sga) (サーバグループパス) [サーバパス] [...][オプション]**

サーバをサーバグループに追加します。



次のオプションを受け入れます。

-f, --folder=[サーバフォルダ]- / デバイス / サーバを基準にした、サーバフォルダの相対パス。  
このオプションを指定すると、サーバオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のサーバオブジェクトを指定するのに使用できます。

#### **server-group-create (sgc) [グループ名] [親フォルダ] [オプション]**

サーバグループを作成して、メンバーを追加します。

次のオプションを受け入れます。

--desc=[説明]- グループの説明。

-m, --members=[サーバパス][...] - / デバイス / サーバを基準にした、サーバの相対パス。

-f, --folder=[サーバフォルダ]- / デバイス / サーバを基準にした、サーバフォルダの相対パス。  
このオプションを指定すると、サーバオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のサーバオブジェクトを指定するのに使用できます。

#### **server-group-members (sgm) (サーバグループパス) [オプション]**

サーバグループまたは動的サーバグループのメンバーを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[スクロールサイズ]- 一度に表示される結果の数。

#### **server-group-remove (sgr) (サーバグループパス) (サーバパス)[...][オプション]**

サーバをサーバグループから削除します。

次のオプションを受け入れます。

-f, --folder=[サーバフォルダ]- / デバイス / サーバを基準にした、サーバフォルダの相対パス。  
このオプションを指定すると、サーバオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のサーバオブジェクトを指定するのに使用できます。

#### **server-info (si) (サーバパス)**

サーバに関する詳細情報が一覧表示されます。

#### **server-list (sl) [オプション] [フォルダ]**

サーバオブジェクトを一覧表示します。

次のオプションを受け入れます。

-r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。

-n, --namefilter=[フィルタ文字列]- 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。

-t, --typefilter=[タイプフィルタ][...] - 指定したオブジェクトタイプのカンマ区切りリストに一致する結果が表示されます。有効な値は、device、group、および folder です。

-c, --count - 結果の数が表示されます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。  
-f, --flagfilter=[フラグフィルタ][...] - 指定したフラグのカンマ区切りリストに一致する結果が表示されます。有効な値は、retired、notretired、test、non-test です。

#### **server-list-bundles (slb) (サーバオブジェクトパス)[オプション]**

サーバオブジェクトに割り当てられたバンドルを一覧表示します。  
次のオプションを受け入れます。

-a, --all - 有効および無効のバンドル両方を一覧表示します。  
-e, --effective - 有効なバンドルのみ一覧表示します。  
-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

#### **server-list-groups (slg) (サーバパス)[オプション]**

特定のサーバがメンバーであるグループを一覧表示します。  
次のオプションを受け入れます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示される結果の数。

#### **server-list-policies (slp) (サーバオブジェクトパス)[オプション]**

サーバオブジェクトに割り当てられたポリシーを一覧表示します。  
次のオプションを受け入れます。

-a, --all - 有効および無効のポリシー両方を一覧表示します。  
-e, --effective - 有効なポリシーのみ一覧表示します。  
-s, --scrollsize=[scrollsize] - 一度に表示される結果の数です。

#### **server-move (smv) (サーバオブジェクトパス)[宛先フォルダパス]**

サーバオブジェクトを別のフォルダに移動します。

#### **server-refresh (srf) (サーバオブジェクトパス)[...][オプション]**

サーバ内の ZENworks Agent を更新します。

次のオプションを受け入れます。

-f, --folder=[サーバフォルダ] - / デバイス / サーバを基準にした、サーバフォルダの相対パス。このオプションが指定される場合は、サーバオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のサーバオブジェクトを指定するのに使用できます。

--server=<server/server folder/server group object path>[...] - /Devices/Servers を基準にした、プライマリサーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の相対パス。クイックタスク通知を送信するために使用します。

--notify=<minutes> - すべてのデバイスに通知する時間 (分)。

--expire=<minutes> - クイックタスクが作成されてから期限切れになるまでの時間 (分)。ただちに期限切れにするには、0 を指定します。

--currentserver - 現在のプライマリサーバにクイックタスクを追加します。

--anyserver - 任意のプライマリサーバにクイックタスクを追加します。

--expirenever - プライマリサーバでクイックタスクを期限切れにしません。

**server-remove-bundle (srb) (サーバオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]**

サーバオブジェクトに割り当てられたバンドルを削除します。  
次のオプションを受け入れます。

-f, --folder=[バンドルフォルダ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

**server-remove-policy (srp) (サーバオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]**

サーバオブジェクトに割り当てられたポリシーを削除します。  
次のオプションを受け入れます。

-f, --folder=[ポリシーフォルダ] - /ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用するのに便利です。

**server-rename (sr) (サーバオブジェクトパス) (新しい名前)**

サーバオブジェクトを名前変更します。

**server-reorder-bundles (srob) (サーバオブジェクトパス) (現在の位置) (新しい位置)**

サーバに割り当てられたバンドルの順序を変更します。server-list-bundles コマンドを使用して、割り当てられたバンドルの順序を取得します。

**server-reorder-policies (srop) (サーバオブジェクトパス) (現在の位置) (新しい位置)**

サーバに割り当てられたポリシーの順序を変更します。server-list-policies コマンドを使用して、割り当てられたポリシーの順序を取得します。

**server-retire (sret) (サーバオブジェクトパス)**

次の更新時に、ZENworks システムから、選択したサーバをリタイアさせます。デバイスのリタイアはデバイスの削除とは異なります。デバイスをリタイアしても、GUID は保持されます (これとは反対に、デバイスを削除すると、GUID も削除されます)。そのため、インベントリ情報はすべて保持されたままでアクセス可能ですが、ポリシーおよびバンドルの割り当てはすべて削除されます。後でデバイスを再度アクティブ化する場合、割り当ては復元されます。

次のオプションを受け入れます。

-i, --immediate - デバイスのリフレッシュで強制的にデバイスをすぐにリタイアします。

**server-set-as-test (ssat) (サーバオブジェクトパス)**

サーバをテストデバイスとして設定します。

例

server1 をテストデバイスとして設定する場合 :

```
zman ssat server1
```

#### **server-role-zookeeper-add-server (srzas)**

ZooKeeper 役割をプライマリサーバに割り当てます。

--servers - サーバのカンマ区切りリスト。/Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-role-zookeeper-remove-server (srzrs)**

Zookeeper クラスタからサーバを削除します。ゾーン内の少なくとも 1 つのサーバに ZooKeeper 役割が必要です。

--servers - /Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-role-zookeeper-list-cluster (srzlc)**

Zookeeper クラスタ内で現在設定されているサーバを一覧にします。

#### **server-role-zookeeper-update-cluster (srzuc) [...] [オプション]**

Zookeeper クラスタの共通プロパティを更新します。

次のオプションを受け入れます。

- l --leader-connect-port: フォロワーがクラスタ内のリーダーに接続するために使用するリーダーポートを定義します。デフォルトポートは 6790 です。ただし、6000 ~ 65535 の間の未使用ポートを指定できます。
- c --client-port: Zookeeper が着信接続用にリスンするポートを定義します。デフォルトポートは 6789 です。ただし、6000 ~ 65535 の間の未使用ポートを指定できます。
- e --leader-elect-port: クラスタ内のすべてのサーバがリーダーを選択するために使用する選択ポートを定義します。デフォルトポートは 6791 です。ただし、6000 ~ 65535 の間の未使用ポートを指定できます。
- t --tick-time: ハートビートおよびタイムアウトを調整するために Zookeeper によって使用されるミリ秒単位の基本タイムユニットである、単一ティックの長さを定義します。デフォルト値は 2000 です。
- i --init-limit: フォロワーがリーダーに接続可能な時間数をティック単位で定義します。デフォルト値は 10 です。
- s --sync-limit: フォロワーが Zookeeper と同期することが可能な時間数をティック単位で定義します。デフォルト値は 5 です。
- m --max-cluster-size: Zookeeper クラスタのサイズを定義します。デフォルト値は 3 です。
- x --max-connections: Zookeeper クラスタが受け入れることが可能なクライアントの最大接続数を指定します。デフォルト値は 60 です。

#### **server-role-kafka-configure-cluster (srkcc) [...] [オプション]**

Kafka クラスタを設定します。

次のオプションを受け入れます。

- c --replication count: トピックごとに作成されるコピー数を指定します。

- l --logRetentionBytes: そのサイズを超えると既存のデータが新しいデータで上書きされる許容可能な最大ログサイズを指定します。デフォルトでは、ログサイズは無制限です。
- t --zkSessionTimeout: Zookeeper セッションタイムアウトを指定します ( ミリ秒単位 )。デフォルト値は 30000 ミリ秒です。サーバがこの指定された期間内に ZooKeeper にハートビート信号を送らない場合、サーバは停止しているとみなされます。ハートビート要求は、サーバが Kafka クラスタに依然として接続されているかどうかを識別するのに役立ちます。
- r --retainDetectedLogsDuration: 削除したログを保持する最大時間を指定します。
- p --logCleanupPolicy: 許容される最大保持期間を超えるセグメントのデフォルトのクリーンアップポリシーを指定します。有効な値は「Delete」および「Compact」です。デフォルト値は「Delete」です。「Delete」ポリシーは、保持期間またはサイズ制限に達した場合に古いセグメントを削除します。「Compact」ポリシーは、トピック上のログ圧縮を有効にし、Kafka が単一トピックパーティションのデータのログ内に各メッセージキーの少なくとも最新の不明な値を常に保持するようにします。
- s --schemaregistryport: スキーマレジストリが実行中のポートを指定します。デフォルトポートは 8081 です。
- k, --kafkaport: Kafka がリスンするポートを指定します。デフォルトポートは 9093 です。
- x, --connectport: Kafka connect がリスンするポートを指定します。デフォルトポートは 8083 です。

#### **server-role-kafka-add-broker (srkab)**

ブローカを Kafka クラスタに追加します。

--servers - サーバのカンマ区切りリスト。/Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パスを指定できます。

-i --ignorewarning- true または false を指定します。このオプションは、外部 CA が特定される場合に表示されるクライアント認証証明書に関連した警告メッセージを無視するために使用されます。続行するには、このオプションを true に設定できます。

#### **server-role-kafka-remove-broker (srkrb)**

Kafka クラスタからブローカを削除します。

--servers - /Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-role-kafka-reconfig-broker (srkrcb)**

Kafka ブローカを再設定します。

--servers - サーバのカンマ区切りリスト。/Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-role-kafka-update-cluster (srkuc) [...] [オプション]**

Kafka クラスタ設定を更新します。

次のオプションを受け入れます。

- l --logRetentionBytes: そのサイズを超えると既存のデータが新しいデータで上書きされる許容可能な最大ログサイズを指定します。デフォルトでは、ログサイズは無制限です。
- t --zkSessionTimeout: Zookeeper セッションタイムアウトを指定します ( ミリ秒単位 )。デフォルト値は 30000 ミリ秒です。サーバがこの指定された期間内に ZooKeeper にハートビート信号を送らない場合、サーバは停止しているとみなされます。ハートビート要求は、サーバが Kafka クラスタに依然として接続されているかどうかを識別するのに役立ちます。
- r --retainDetectedLogsDuration: 削除したログを保持する最大時間を指定します。
- p --logCleanupPolicy: 許容される最大保持期間を超えるセグメントのデフォルトのクリーンアップポリシーを指定します。有効な値は「Delete」および「Compact」です。デフォルト値は「Delete」です。「Delete」ポリシーは、保持期間またはサイズ制限に達した場合に古いセグメントを削除します。「Compact」ポリシーは、トピック上のログ圧縮を有効にし、Kafka が単一トピックパーティションのデータのログ内に各メッセージキーの少なくとも最新の不明な値を常に保持するようにします。
- c, --replication count: 各トピックに対して作成されるコピー数を指定します。
- s --schemaregistryport: スキーマレジストリが実行中のポートを指定します。
- k, --kafkaport: Kafka がリスンするポートを指定します。
- x, --connectport: Kafka connect がリスンするポートを指定します。

#### **server-role-kafka-list-cluster (srklc)**

Kafka クラスタ設定を取得します。

#### **server-role-kafka-list-connectors (srklcn)**

Kafka コネクタのリストを取得します。

#### **server-role-kafka-restart-connectors (srkrnc) [...] [ オプション ]**

指定された Kafka コネクタを再起動します。

次のオプションを受け入れます。

- c --connectors: コネクタのカンマ区切りリストを指定します。

#### **server-role-kafka-get-connector-config (srkgcc) [...] [ オプション ]**

Kafka コネクタの設定の詳細を取得します。

次のオプションを受け入れます。

- c --connector: その設定詳細を取得するためにコネクタの名前を指定します。

#### **server-role-kafka-reconfigure-connectors (srkccn) [...] [ オプション ]**

指定された Kafka コネクタを再設定します。

次のオプションを受け入れます。

- c --connectors: コネクタのカンマ区切りリストを指定します。

#### **server-role-vertica-create-cluster (srvcc) [...] [ オプション ]**

1 つ以上のノードで Vertica クラスタを作成します。これは、Vertica 設定時に実行する必要がある最初のコマンドです。

--servers - サーバのカンマ区切りリスト。/Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

次のオプションを受け入れます。

-k --replication-factor: データベースに存在するデータのレプリカ数を指定します。クラスタ内のノード数に基づいた値を指定する必要がある、必要なノードの最小数は  $2k+1$  として測定されます。Vertica では 1 の K-safety ファクタを推奨します。

#### **server-role-vertica-add-server (srvas)**

サーバを Vertica クラスタに追加します。

--servers - サーバのカンマ区切りリスト。/Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-role-vertica-remove-server (srvrs)**

Vertica クラスタからサーバを削除します。

--servers - サーバのカンマ区切りリスト。/Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-role-vertica-list-cluster (srvlc)**

Vertica クラスタ内のサーバを一覧にします。

#### **server-role-vertica-update-cluster (srvuc)**

クラスタ内の K-safety ファクタを更新します。

次のオプションを受け入れます。

-k --replication-factor: クラスタ内に維持するレプリカの合計数を示すためにレプリケーションファクタを変更します。

#### **server-role-vertica-get-credentials (srvgc)**

Vertica データベース資格情報を取得します。

#### **server-role-vertica-prepare-server (srvps)**

バックアップおよび復元プロセス中にサーバを準備します。

(サーバオブジェクトパス) - /Devices/Servers を基準にした、サーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の GUID、DNS、または相対パス。

#### **server-set-as-nontest (ssan) (サーバオブジェクトパス)**

サーバをテストデバイス以外として設定します。

例

server1 をテストデバイス以外として設定する場合 :

```
zman ssan server1
```

### **server-statistics (sst) (サーバパス)**

ZENworks サーバ使用の統計が表示されます。

### **server-unretire (suret) (サーバオブジェクトパス)**

次のリフレッシュ時に選択されたサーバを再度有効にして、デバイスが以前所有していたすべてのポリシーとバンドルの割り当てを再割り当てします。

次のオプションを受け入れます。

-i, --immediate - デバイスのリフレッシュで強制的にデバイスをすぐにリタイア解除します。

### **server-view-all-bundles-status (svabs) (サーバパス) [オプション]**

サーバに割り当てられたバンドルの詳細な展開ステータスを表示します。

次のオプションを受け入れます。

--statusfilter=[ステータスタイプ][...] - バンドル配布およびインストールイベントのステータスに対するフィルタです。有効な値は S、R、C、A、F、および P (Success、Partial Success、Completed、Partial Complete、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

例

server1 に割り当てられたバンドルの展開状態を表示する場合：

```
zman svabs server1
```

バンドル展開が成功または保留状態の server1 に割り当てられたバンドルの展開状態を表示する場合：

```
zman svabs server1 --statusfilter S,P
```

### **server-view-all-policies-status (svaps) (サーバパス) [オプション]**

サーバに割り当てられたポリシーの詳細な展開ステータスを表示します。

次のオプションを受け入れます。

--statusfilter=[ステータスタイプ][...] - ポリシー適用イベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示される結果の数。

### **server-wakeup (sw) (サーバオブジェクトパス)[...][オプション]**

Wake-On-LAN を使用してサーバを起動します。

次のオプションを受け入れます。

-f, --folder=[サーバフォルダ] - / デバイス / サーバを基準にした、サーバフォルダの相対パス。このオプションが指定される場合は、サーバオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のサーバオブジェクトを指定するのに使用できます。



- b --broadcast=[ブロードキャストアドレス][...] - Wake-On-LAN マジックパケットをブロードキャストするのに使用するアドレスのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。有効な IP アドレスが、有効な値です。
- s, --server=[プライマリサーバオブジェクトまたはプロキシサーバオブジェクトのパスであり、/デバイスへの相対パスです][...] - デバイスを起動するのに使用するプライマリまたはプロキシサーバのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。
- C, --retries=[再試行回数] - デバイスに Wake-On-LAN マジックパケットを送信する回数。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 0 ~ 5 の範囲である必要があります。デフォルト値は「1」です。
- t, --timeout=[再試行間隔] - 2 回の再試行の間隔。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 2 ~ 10 の範囲である必要があります (単位は分)。デフォルト値は「2」です。

## 設定コマンド

設定の設定と変更を使用するコマンドです。設定コマンドは、長い形式では settings- から始まり、短い形式では s から始まります。

設定コマンドは ZENworks Endpoint Security Management の設定 (ゾーンポリシー設定とデータ暗号化キー設定) には適用されません。

次のコマンドはすべて、「**グローバルオプション**」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- [113 ページ](#)の「`settings-copy (scp)` (ソースデバイスまたはデバイスフォルダパス) (宛先デバイスまたはデバイスフォルダパス) [...] [オプション]」
- [114 ページ](#)の「`settings-export-to-file (setf)` [オプション] (XML ファイルパス) [設定名] [...]」
- [114 ページ](#)の「`settings-revert (srt)` (ソースデバイスまたはデバイスフォルダパス) (設定名) [...]」
- [114 ページ](#)の「`settings-set (ss)` (XML ファイルパス) [宛先デバイスまたはデバイスフォルダパス] [...]」
- [114 ページ](#)の「`settings-gcm-delete (sgd)` [...]」

**settings-copy (scp)** (ソースデバイスまたはデバイスフォルダパス) (宛先デバイスまたはデバイスフォルダパス) [...] [オプション]

設定をコピーします。

(ソースデバイスまたはデバイスフォルダパス) - /デバイスを基準にした、設定のコピー元デバイスまたはデバイスフォルダの相対パス。

(宛先デバイスまたはデバイスフォルダパス) [...] - /デバイスを基準にした、設定のコピー先デバイスまたはデバイスフォルダの相対パス。

次のオプションを受け入れます。

-n, --names=[ 設定名 ][...] - ソースパスからコピーされる設定の名前。指定しなかった場合、ソースパスで定義されたすべての設定がコピーされます。

#### **settings-export-to-file (setf) [ オプション ] (XML ファイルパス) [ 設定名 ] [...]**

設定データ (XML 形式) をファイルにエクスポートします。設定の作成および変更用の入力として XML ファイルが使用されます。

(XML ファイルパス) - 設定データが XML 形式で保存されるファイル。ファイルが存在しない場合、新しいファイルが作成されます。

[ 設定名 ] [...] - エクスポートする設定の名前。指定しなかった場合、すべての設定がエクスポートされます。

次のオプションを受け入れます。

-s, --source=[ ソースデバイスまたはデバイスフォルダパス ] - / デバイスを基準にした、設定のエクスポート元デバイスまたはデバイスフォルダの相対パス。指定しなかった場合、設定はゾーンからエクスポートされます。

-e, --effective - 指定した場合は有効な設定が取得されます。そうでない場合はソースパスで定義 / 上書きされた設定のみが取得されます。

#### **settings-revert (srt) ( ソースデバイスまたはデバイスフォルダパス ) ( 設定名 ) [...]**

親フォルダへの設定を取り消します。

( ソースデバイスまたはデバイスフォルダパス ) - / デバイスを基準にした、設定を取り消す必要があるデバイスまたはデバイスフォルダの相対パス。

#### **settings-set (ss) (XML ファイルパス) [ 宛先デバイスまたはデバイスフォルダパス ] [...]**

さまざまなレベルで設定を行います。

(XML ファイルパス) - エクスポートされた設定情報が含まれる XML ファイル。settings-export-to-file (setf) を使用して、設定情報を XML ファイルにエクスポートします。

次のオプションを受け入れます。

[ 設定名 ] [...] - エクスポートする設定の名前。指定しなかった場合、すべての設定がエクスポートされます。

-s, --source=[ ソースデバイスまたはデバイスフォルダパス ] - / デバイスを基準にした、設定が必要なデバイスまたはデバイスフォルダの相対パス。指定しなかった場合、設定はゾーンレベルで設定されます。

#### **settings-gcm-delete (sgd) [...]**

Google Cloud Messaging の設定値を削除します。変更を反映するには、ZENworks サービスを再起動する必要があります。

## サブスクリプションコマンド

これらのコマンドは、サブスクリプションの作成、変更、管理に使用されます。サブスクリプションコマンドは、長い形式では subscription- から始まり、短い形式では sr から始まります。

このコマンドは、「**グローバルオプション**」のセクションに一覧表示されるオプションフラグを受け入れます。さらに、コマンドと併せて一覧表示されている個々のオプションも受け入れます。

- 115 ページの「subscription-create (src) (サブスクリプション名) (サブスクリプション XML ファイル) [親フォルダ] [オプション]」
- 116 ページの「subscription-folder-create (srfc) (フォルダ名) [親フォルダ] [オプション]」
- 116 ページの「subscription-move (srmv) (サブスクリプションオブジェクトのパス) [宛先フォルダパス]」
- 116 ページの「subscription-replicate (srr) (サブスクリプション名) [宛先フォルダパス] [オプション]」
- 116 ページの「subscription-export (sre) (サブスクリプション名) (サブスクリプションデータ XML)」
- 116 ページの「subscription-delete (srd) (サブスクリプションオブジェクト) [...] [オプション]」
- 117 ページの「subscription-list (srl) [オプション]」
- 117 ページの「subscription-modify (srm) (サブスクリプション名) (サブスクリプションデータ XML) [親フォルダ] [オプション]」
- 117 ページの「subscription-status (srs) (サブスクリプション名) [オプション]」
- 117 ページの「subscription-channel-list (srcl) (サブスクリプション名) (レポートエンティティタイプ) [オプション]」
- 118 ページの「subscription-copy (srcp) (サブスクリプションパス) (サブスクリプション名) [親フォルダ] [オプション]」
- 118 ページの「subscription-rename (srrn) (サブスクリプションオブジェクトのパス) (新しい名前)」
- 118 ページの「subscription-proxy-password (srpp) [サブスクリプションプロキシパスワード]」

**subscription-create (src) (サブスクリプション名) (サブスクリプション XML ファイル) [親フォルダ] [オプション]**

新しいサブスクリプションを作成します。

(サブスクリプション XML ファイル) - エクスポートされたサブスクリプション情報が含まれる XML ファイル。subscription-export (sre) を使用して、既存のサブスクリプションの情報を XML ファイルにエクスポートします。ファイルを再使用する場合には、

ZENworks コントロールセンターで作成したサブスクリプションからテンプレート XML ファイルを作成することができます。サンプル XML ファイルは、nu.xml を参照してください

「Linux」 : /opt/novell/zenworks/share/zman/samples/subscriptions

「Windows」 : %ZENWORKS\_HOME%\novell\zenworks\share\zman\samples\subscriptions

次のオプションを受け入れます。

-d, --desc=[ 説明 ] - サブスクリプションの説明。

-s, --sysIDpath=[rhel- システム ID]

---

**注 :** このコマンドは Apple VPP サブスクリプションに対して実行できません。

---

#### **subscription-folder-create (srfc) (フォルダ名) [親フォルダ] [オプション]**

新しいサブスクリプションフォルダを作成します。

次のオプションを受け入れます。

--desc=[ 説明 ] - フォルダの説明。

#### **subscription-move (srmv) (サブスクリプションオブジェクトのパス) [宛先フォルダパス]**

サブスクリプションオブジェクトを別のフォルダに移動します。

#### **subscription-replicate (srr) (サブスクリプション名) [宛先フォルダパス] [オプション]**

サブスクリプションレプリケーションを開始します。

次のオプションを受け入れます。

-s, --schedule=[XML ファイル名] - サブスクリプションを開始するスケジュールを含む XML ファイルです。Eventsch.xml ファイルはサブスクリプションに適用されません。

-r, --fromRepository - コンテンツの複製元のリポジトリです。

---

**注 :** Apple VPP サブスクリプションの場合、このコマンドの実行中にアウトバウンドサーバは利用されません。そのため、このコマンドを実行するサーバがインターネットに接続されていることを確認する必要があります。

---

#### **subscription-export (sre) (サブスクリプション名) (サブスクリプションデータ XML)**

サブスクリプションの情報 (XML 形式) をファイルにエクスポートします。この XML ファイルはサブスクリプション作成時の入力として使用します。

#### **subscription-delete (srd) (サブスクリプションオブジェクト) [...] [オプション]**

1 つまたは複数のサブスクリプションオブジェクトを削除します。ステータスが「進行中」、「中止」、「割り当て済み」のサブスクリプションは削除できません。

次のオプションを受け入れます。

-r, --recursive - フォルダ内のオブジェクトを再帰的に削除します。

-b, --delete-bundles - 指定されたサブスクリプションで作成されたすべてのバンドルを削除します。

-f, --folder=[サブスクリプションフォルダ]- /サブスクリプションを基準にした、サブスクリプションフォルダの相対パス。このオプションを指定すると、サブスクリプションオブジェクトのパスは、このフォルダに対して決定されます。このオプションは、同じフォルダで複数のサブスクリプションオブジェクトを指定するのに使用できません。

---

**注:** このコマンドは Apple VPP サブスクリプションに対して実行できません。

---

#### subscription-list (srl) [オプション]

サブスクリプションオブジェクトを一覧表示します。

次のオプションを受け入れます。

- r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。
- n, --namefilter=[フィルタ文字列] - 指定した名前に一致する結果を表示します。ワイルドカード \* および ? は引用符で囲まれている限りは使用できます。
- c, --count - 結果の数が表示されます。
- s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。  
[フォルダ] - 指定したフォルダの内容を一覧表示します。フォルダを指定しないと、ルートフォルダの内容が一覧表示されます。

#### subscription-modify (srm) (サブスクリプション名) (サブスクリプションデータ XML) [親フォルダ] [オプション]

既存のサブスクリプションをエクスポートまたは変更します。

次のオプションを受け入れます。

- d, --desc=[説明] - サブスクリプションの説明。
- s, --sysIDpath=[rhel- システム ID]

---

**注:** このコマンドは Apple VPP サブスクリプションに対して実行できません。

---

#### subscription-status (srs) (サブスクリプション名) [オプション]

サブスクリプションのステータスを返します。

次のオプションを受け入れます。

- s, --suspend - レプリケーションを停止し、サブスクリプションのステータスを「中止」に変更します。
- c, --cancel - レプリケーションをキャンセルし、サブスクリプションステータスを「キャンセル」に変更します。

---

**注:** ドライ実行の実行中の中止タスクとキャンセルタスクは、サポートされていません。

---

#### subscription-channel-list (srcl) (サブスクリプション名) (レポートエンティティタイプ) [オプション]

バンドル、カタログ、パッケージなどのデータを一覧表示します。

次のオプションを受け入れます。

-r, --fromRepository - キャッシュをバイパスして、コンテンツの複製元のリポジトリからメタデータをダウンロードします。

-i, --ignoreFilters - フィルタが設定されていても無視して、コンテンツを一覧表示します。

**subscription-copy (srcp) (サブスクリプションパス)(サブスクリプション名)[親フォルダ][オプション]**

サブスクリプションをコピーします。

次のオプションを受け入れます。

-d, --desc=説明 - バンドルの説明。

**subscription-rename (srrn) (サブスクリプションオブジェクトのパス)(新しい名前)**

サブスクリプションの名前を変更します。

たとえば、サブスクリプション SLES10Update を SLED10Update に名前を変更するには、次のコマンドを実行します。

```
zman srrn /Subscription/SLES10Update /Subscription/SLED10Update
```

**subscription-proxy-password (srpp) [サブスクリプションプロキシパスワード]**

レプリケーション時にすべてのサブスクリプションで使用されるプロキシパスワードを設定します。このコマンドでは、プロキシパスワードの引数はオプションです。プロキシパスワードの引数を入力しない場合、パスワードを入力するようにコマンドから求められます。

## システム更新 / 製品認識更新コマンド

これらのコマンドを使用して、ZENworks へのシステム更新、および製品認識更新 (PRU) を管理します。これらのコマンドは、長い形式では system-update から始まり、短い形式では su から始まります。

- 119 ページの「[system-update-download \(sud\)](#) (システムアップデートまたは PRU の名前)」
- 119 ページの「[system-update-view-available \(suva\)](#) [オプション]」
- 119 ページの「[system-update-status \(sus\)](#) (システムアップデートまたは PRU の名前)」
- 119 ページの「[system-update-delete-update \(sudu\)](#) (システムアップデートまたは PRU の名前) [オプション]」
- 120 ページの「[system-update-cancel-download \(sucd\)](#) (システムアップデートまたは PRU の名前)」
- 120 ページの「[system-update-assign \(sua\)](#) (システム更新名) [デバイスパス] [...] [オプション]」
- 120 ページの「[system-update-list-assign \(sula\)](#) (デバイスパス)」

- ◆ 120 ページの「system-update-authorize (suaz) (システムアップデートまたは PRU の名前)」
- ◆ 120 ページの「system-update-baseline (sub) (システム更新名)」
- ◆ 120 ページの「system-update-import (sui) (更新のパス) [システムアップデートまたは PRU の名前] [-b, --checkbaseline]
- ◆ 121 ページの「system-update-activate-entitlement (suae) (メール ID) (起動コード)」
- ◆ 121 ページの「system-update-view-entitlement-status (suves)」
- ◆ 121 ページの「system-update-rebuild-packages (surp) (システム更新名)」
- ◆ 121 ページの「system-update-patch-file (supf) (システム更新名) (パッチのパス)」
- ◆ 121 ページの「(system-update-create-package | sucpc) (システムアップデート名) [デバイスパス] [-n|--packagename=スタンドアロンアップデートパッケージ名] [-p|--platform=OS プラットフォーム] [-a|--arch=アーキテクチャ]
- ◆ 122 ページの「zman 領域 [オプション]

#### **system-update-download (sud) (システムアップデートまたは PRU の名前)**

最新のシステムアップデートまたは PRU をダウンロードします。

(システムアップデートまたは PRU の名前) - システムアップデートまたは PRU の名前か UID。

#### **system-update-view-available (suva) [オプション]**

更新サーバまたは更新ファイル内で公開されている、利用可能な更新または PRU のすべてのリストが表示されます。

次のオプションを受け入れます。

- i, --importFile=[インポートファイル/ディレクトリのパス] - クエリへの重要なファイルへのフルパス。
- u, --url=[更新サーバの URL] - クエリするシステムアップデートサーバの URL で、http://servername:[port]/path という形式。
- f, --product=[製品コード] - アップデートをチェックする製品コード (たとえば zcm、pru)。
- v, --version=[製品のベースバージョン] - アップデートをチェックする製品のベースバージョン (たとえば、10.0、10.1)。

#### **system-update-status (sus) (システムアップデートまたは PRU の名前)**

指定したシステムアップデートまたは PRU のステータスを表示します。

(システムアップデートまたは PRU の名前) - システムアップデートまたは PRU の名前か UID。

#### **system-update-delete-update (sudu) (システムアップデートまたは PRU の名前) [オプション]**

指定したシステムアップデートまたは PRU をゾーンから削除します。

(システムアップデートまたは PRU の名前) - システムアップデートまたは PRU の名前か UID です。

次のオプションを受け入れます。

-f, --force - 更新を強制的に削除します。割り当てがサーバからのみ削除されます。デバイスを更新し、更新を表示すると、更新がそのデバイスに適用されます。十分注意し、更新が有効な状態でないことが確かな場合のみ行ってください。

**system-update-cancel-download (sucd) (システムアップデートまたは PRU の名前)**

システムアップデートまたは PRU のダウンロードをキャンセルします。

(システムアップデートまたは PRU の名前) - システムアップデートまたは PRU の名前か UID。

**system-update-assign (sua) (システム更新名) [デバイスパス] [...] [オプション]**

認定済みシステムアップデートをデバイスに割り当てます。

(システム更新名) - システムアップデート名または UID。

[デバイスパス] [...] - / デバイスを基準にした、デバイスの相対パス。

次のオプションを受け入れます。

-a, --all - すべてのデバイスに割り当てます。

**system-update-list-assign (sula) (デバイスパス)**

指定したデバイスまたはグループに割り当てられている更新をすべてリストにします。

(デバイスパス) - / デバイスを基準にした、デバイスの相対パス。

**system-update-authorize (suaz) (システムアップデートまたは PRU の名前)**

アップデートまたは PRU を認定します。

(システムアップデートまたは PRU の名前) - システムアップデートまたは PRU の名前か UID。

**system-update-baseline (sub) (システム更新名)**

指定した更新をベースライン更新として設定するようにスケジュールします。

(システム更新名) - システム更新名または UID。

**system-update-import (sui) (更新のパス) [システムアップデートまたは PRU の名前] [-b, --checkbaseline]**

システムアップデートまたは PRU を、ZIP ファイルまたはディレクトリからゾーンにインポートします。アップデートまたは PRU が ZENworks システムに入った後は、オンラインアップデートや PRU と同じ方法で管理することができます。

(更新のパス) - アップデートまたは PRU の ZIP ファイルまたはディレクトリへのフルパス。

(システムアップデートまたは PRU の名前) - システムアップデートまたは PRU の名前か UID。

(-b, --checkbaseline) - 更新をチェックし、該当する場合はベースラインにします。デフォルトは、更新をベースラインにしません。



### system-update-activate-entitlement (suae) ( メール ID ) ( 起動コード )

管理ゾーン内の ZENworks ソフトウェアのシステム更新エンタイトルメントをアクティブ化して、Micro Focus Customer Center サーバから最新バージョンの ZENworks システム更新と PRU(Product Recognition Update: 製品認識の更新)を受け取ります。

( メール ID ) - Micro Focus からの通信に使用される有効な電子メールアドレス。システム更新保守エンタイトルメントの購入に使用した電子メールアドレスを指定することをお勧めします。

( 起動コード ) - システム更新エンタイトルメントの起動コード。

---

**注:** エンタイトルメントをアクティブ化する前に、エンタイトルメントのアクティブ化に使用するプライマリサーバが [NCC サーバ \(https://secure-www.novell.com\)](https://secure-www.novell.com) と通信できることを確認してください。

---

### system-update-view-entitlement-status (suves)

システム更新のエンタイトルメントステータスを表示します。

### system-update-rebuild-packages (surp) ( システム更新名 )

指定された更新コンテンツを使用して、このサーバに展開パッケージを再構築します。

( システム更新名 ) - システムアップデート名または UID です。

-f, --force - アップデートに新しいパッケージが含まれていない場合でもパッケージの再構築を強制します。

### system-update-patch-file (supf) ( システム更新名 ) ( パッチのパス )

新バージョンのファイルが含まれるアップデートのパッチを適用します。

( システム更新名 ) - システムアップデート名または UID です。

( パッチのパス ) - パッチファイルが含まれる zip ファイルまたはフォルダのフルパスです。

### (system-update-create-package | sucpl) ( システムアップデート名 ) [ デバイスパス ] [-n|--packagename= スタンドアロンアップデートパッケージ名] [-p|--platform=OS プラットフォーム] [-a|--arch= アーキテクチャ]

指定されたスタンドアロンアップデートパッケージを作成します。このコマンドを使用すると、アップデート GUID とデバイス GUID を使用して特定のデバイスのパッケージを作成したり、プラットフォームとアーキテクチャの詳細を使用してプラットフォームのパッケージを作成することができます。

( システム更新名 ) - システムアップデート名または UID です。

[ デバイスパス ] - /Devices に対するデバイスの相対パス。

-n, --packagename=< スタンドアロンアップデートパッケージ名 > - スタンドアロンアップデートパッケージの名前。指定しない場合、指定されたデバイス名またはプラットフォームから名前が導出されます。

-p, --platform=< OS プラットフォーム > - スタンドアロンアップデートパッケージの OS プラットフォーム。有効な値は *Windows* です。

-a, --arch=< アーキテクチャ > - スタンドアロンアップデートパッケージのデバイスアーキテクチャ。有効な値は 32 または 64 です。

## zman 領域 [ オプション ]

管理対象デバイスでの有効な割り当てを計算するには、次のオプションを受け入れます。

-f、--強制された計算 – 管理対象デバイスに対する有効な割り当て計算を即時にトリガーします。

## ユーザコマンド

ZENworks で設定されるユーザソースに属するユーザを管理するコマンドです。ユーザソースは、e-Directory や Active Directory など、既存の LDAP の信頼されたソースから設定されず。ユーザコマンドは、長い形式では user- から始まり、短い形式では u から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- 123 ページの「`user-add-bundle (uab)` (ユーザオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]]」
- 124 ページの「`user-add-policy (uap)` (ユーザオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]]」
- 124 ページの「`user-container-create (ucc)` (ユーザソース) (ユーザコンテナ) (表示名)]]」
- 125 ページの「`user-group-add (uga)` (ZENworks ユーザグループパス) (ユーザパス) [...] [オプション]]」
- 125 ページの「`user-group-create (ugc)` (グループ名) [オプション]]」
- 125 ページの「`user-group-delete (ugd)` (ZENworks ユーザグループパス) [...]」
- 125 ページの「`user-group-members (ugm)` (ユーザグループパス) [オプション]]」
- 125 ページの「`user-group-remove (ugr)` (ZENworks ユーザグループパス) (ユーザパス) [...] [オプション]]」
- 125 ページの「`user-info (ui)` (ユーザパス)]]」
- 125 ページの「`user-list (ul)` [オプション] [フォルダ]]」
- 126 ページの「`user-list-bundles (ulb)` (ユーザオブジェクトパス) [オプション]]」
- 126 ページの「`user-list-groups (ulg)` (ユーザパス) [オプション]]」
- 126 ページの「`user-list-policies (ulp)` (ユーザオブジェクトパス) [オプション]]」
- 126 ページの「`user-logged-in-devices (ulid)` (ユーザパス)]]」
- 126 ページの「`user-primary-workstation (upws)` (ユーザパス)]]」
- 126 ページの「`user-remove-bundle (urb)` (ユーザオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]]」

- ◆ 127 ページの「user-remove-policy (urp) (ユーザオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- ◆ 127 ページの「user-reorder-policies (urop) (user object path) (current position) (new position)」
- ◆ 127 ページの「user-set-as-test (usat) (ユーザオブジェクトパス)」
- ◆ 127 ページの「user-set-as-nontest (usan) (ユーザオブジェクトパス)」
- ◆ 127 ページの「user-source-create (usc) (ユーザソース XML ファイルパス) [オプション]」
- ◆ 127 ページの「user-source-list (usl)」
- ◆ 127 ページの「user-source-list-connections (uslc) (ユーザソース)」
- ◆ 127 ページの「user-source-remove-connection (usrc) (ユーザソース) (接続名)」
- ◆ 127 ページの「user-source-trustedcert-add (usta) (証明書エイリアス) (証明書のファイルパス)」
- ◆ 128 ページの「user-source-trustedcert-list (ustl)」
- ◆ 128 ページの「user-source-trustedcert-remove, ustr (証明書エイリアス)」
- ◆ 128 ページの「user-source-add-connection (usac) (ユーザソース) (接続名) (サーバアドレス) (サーバポート) [オプション]」
- ◆ 128 ページの「user-view-all-bundles-status (uvabs) (ユーザパス) [オプション]」
- ◆ 128 ページの「user-view-all-policies-status (uvaps) (ユーザパス) [オプション]」

**user-add-bundle (uab) (ユーザオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]**

バンドルをユーザオブジェクトに割り当てます。

次のオプションを受け入れます。

- f, --folder=[ バンドルフォルダ ] - / バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。
- l, --icon-location=[ アプリケーションロケーション XML ファイル ] - バンドルアプリケーション用のアイコンを配置する場所を含む XML ファイル。XML ファイル形式については、IconLocation.xml を参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/bundles、Windows サーバでは *Installation\_directory*: \Novell\Zenworks\share\zman\samples\bundles に保存されています。
- d, --distribution-schedule=[ 配布スケジュール XML ファイル ] - 配布スケジュールを含む XML ファイル。
- l, --launch-schedule=[ 起動スケジュール XML ファイル ] - 起動スケジュールを含む XML ファイル。

- b, --install-schedule=[インストールスケジュール XML ファイル]- インストールスケジュールを含む XML ファイル。
- a, --availability-schedule=[ 可用性スケジュール XML ファイル ]- 可用性スケジュールを含む XML ファイル。スケジュール XML ファイルテンプレートについては、XML ファイルを参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/schedules、Windows サーバでは *Installation\_directory*:\Novell\Zenworks\share\zman\samples\schedules に保存されています。
- i, --install-immediately - 配布後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- L, --launch-immediately - インストール後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- n, --distribute-now - バンドルをただちに配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule および --distribute-on-device-refresh オプションは無視されます。--distribute-now、--distribute-on-device-refresh、および --distribution-schedule オプションは配布スケジュールの設定に使用し、同時に使用することはできません。最初に --distribute-now オプションが考慮され、次に --distribute-on-device-refresh オプションが考慮され、最後に --distribution-schedule が考慮されます。
- r, --distribute-on-device-refresh - デバイス更新時にバンドルを配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule オプションは無視されます。
- s, --launch-on-device-refresh - デバイス更新時にバンドルを起動するよう起動スケジュールを設定します。このオプションを指定すると、--launch-schedule オプションは無視されます。

**user-add-policy (uap) ( ユーザオブジェクトパス )( ポリシーまたはポリシーグループ ) [...][ オプション ]**

ポリシーをユーザオブジェクトに割り当てます。

次のオプションを受け入れます。

- e, --enforce-now - 割り当てられているすべてのデバイスに対してポリシーをただちに強制的に適用します。
- f, --folder=[ ポリシーフォルダ ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

**user-container-create (ucc) ( ユーザソース )( ユーザコンテナ )( 表示名 )**

ZENworks で設定されたユーザソースにユーザコンテナを追加します。

( ユーザコンテナ )- LDAP ソースのルートコンテキストに対して相対的な RDN( 相対式別名 ) 形式で追加されるユーザコンテナ。

( 表示名 )- ユーザコンテナの名前を表示します。

### **user-group-add (uga) (ZENworks ユーザグループパス) (ユーザパス) [...] [オプション]**

ZENworks ユーザグループにユーザを追加します。

次のオプションを受け入れます。

-f, --folder=[ ユーザフォルダ ] - /ユーザを基準にした、ユーザフォルダの相対パス。このオプションを指定すると、ユーザオブジェクトのパスは、このフォルダを基準にして決定されます。これは、同じフォルダで複数のユーザオブジェクトを指定するのに使用できます。

### **user-group-create (ugc) (グループ名) [オプション]**

ZENworks ユーザグループを作成して、メンバーを追加します。

次のオプションを受け入れます。

--desc=[ 説明 ] - グループの説明。

-m, --members=[ ユーザパス ] [...] - /ユーザを基準にした、ユーザの相対パス。

-f, --folder=[ ユーザフォルダ ] - /ユーザを基準にした、ユーザフォルダの相対パス。このオプションを指定すると、ユーザオブジェクトのパスは、このフォルダを基準にして決定されます。これは、同じフォルダで複数のユーザオブジェクトを指定するのに使用できます。

### **user-group-delete (ugd) (ZENworks ユーザグループパス) [...]**

1 つ以上の ZENworks ユーザグループを削除します。

(ZENworks ユーザグループパス) [...] - /ユーザを基準にした、ZENworks ユーザグループのパス。

### **user-group-members (ugm) (ユーザグループパス) [オプション]**

ZENworks ユーザグループまたは LDAP ディレクトリユーザグループのメンバーを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

### **user-group-remove (ugr) (ZENworks ユーザグループパス) (ユーザパス) [...] [オプション]**

ZENworks ユーザグループからユーザを削除します。

次のオプションを受け入れます。

-f, --folder=[ ユーザフォルダ ] - /ユーザを基準にした、ユーザフォルダの相対パス。このオプションを指定すると、ユーザオブジェクトのパスは、このフォルダを基準にして決定されます。これは、同じフォルダで複数のユーザオブジェクトを指定するのに使用できます。

### **user-info (ui) (ユーザパス)**

ユーザに関する詳細情報が一覧表示されます。

### **user-list (ul) [オプション] [フォルダ]**

ユーザオブジェクトを一覧表示します。

次のオプションを受け入れます。

- r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。
- n, --namefilter=[ フィルタ文字列 ] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。
- t, --typefilter=[ タイプフィルタ ][...] - 指定したオブジェクトタイプのカンマ区切りリストに一致する結果が表示されます。有効な値は、user、group、および folder です。
- c, --count - 結果の数が表示されます。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **user-list-bundles (ulb) ( ユーザオブジェクトパス ) [ オプション ]**

ユーザオブジェクトに割り当てられたバンドルを一覧表示します。  
次のオプションを受け入れます。

- a, --all - 有効および無効のバンドル両方を一覧表示します。
- e, --effective - 有効なバンドルのみ一覧表示します。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **user-list-groups (ulg) ( ユーザパス ) [ オプション ]**

特定のユーザがメンバーであるグループを一覧表示します。  
次のオプションを受け入れます。

- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **user-list-policies (ulp) ( ユーザオブジェクトパス ) [ オプション ]**

ユーザオブジェクトに割り当てられたポリシーを一覧表示します。

- a, --all - 有効および無効のポリシー両方を一覧表示します。
- e, --effective - 有効なポリシーのみ一覧表示します。
- s, --scrollsize=[ スクロールサイズ ] - 一度に表示させる結果の数。

#### **user-logged-in-devices (ulid) ( ユーザパス )**

ユーザがログインしているデバイスを表示します。

#### **user-primary-workstation (upws) ( ユーザパス )**

ユーザのプライマリワークステーション、ログイン数、プライマリワークステーションにユーザがログインしていた時間 (分) を一覧にします。

#### **user-remove-bundle (urb) ( ユーザオブジェクトパス ) ( バンドルまたはバンドルグループ ) [...] [ オプション ]**

ユーザオブジェクトに割り当てられたバンドルを削除します。  
次のオプションを受け入れます。

- f, --folder=[ バンドルフォルダ ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

**user-remove-policy (urp) ( ユーザオブジェクトパス )( ポリシーまたはポリシーグループ ) [ ... ] [ オプション ]**

ユーザオブジェクトに割り当てられたポリシーを削除します。  
次のオプションを受け入れます。

-f, --folder=[ ポリシーフォルダ ] - / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

**user-reorder-policies (urop) ( user object path ) ( current position ) ( new position )**

ユーザに割り当てられたポリシーの順序を変更します。

**user-set-as-test (usat) ( ユーザオブジェクトパス )**

ユーザをテストユーザとして設定します。

例

user1 をテストユーザとして設定する場合 :

```
zman usat user1
```

**user-set-as-nontest (usan) ( ユーザオブジェクトパス )**

ユーザをテストユーザ以外として設定します。

例

user1 をテストユーザ以外として設定する場合 :

```
zman usan user1
```

**user-source-create (usc) ( ユーザソース XML ファイルパス ) [ オプション ]**

ユーザソースを設定します。  
次のオプションを受け入れます。

-a, --accept-certificate - SSL モードを選択した場合に、ユーザソースが提示する証明書を受け入れます。このオプションは、スクリプトの場合にインタラクティブモードにならないようにするために設けられています。証明書を表示してから承認することをお勧めします。

**user-source-list (usl)**

設定されたユーザソースを一覧表示します。

**user-source-list-connections (uslc) ( ユーザソース )**

ユーザソースのサーバ接続を一覧表示します。

**user-source-remove-connection (usrc) ( ユーザソース ) ( 接続名 )**

ユーザソースからサーバ接続を削除します。

**user-source-trustedcert-add (usta) ( 証明書エイリアス ) ( 証明書のファイルパス )**

他の LDAP サーバ用に別の信頼する証明書を追加します。この証明書は、セキュアなユーザソースの接続を使用するとき、LDAP の検索要求の処理で参照できます。

(証明書エイリアス) - キーストアで証明書を特定するために使用するエイリアス。  
(証明書ファイルパス) - DER 形式の証明書ファイルのパス。

#### **user-source-trustedcert-list (ustl)**

セキュアなユーザソースの接続に使用する別の信頼する証明書を一覧表示します。

#### **user-source-trustedcert-remove, ustr (証明書エイリアス)**

セキュアなユーザソースの接続に使用する信頼する証明書を削除します。

(証明書エイリアス) - キーストアで証明書を特定するために使用するエイリアス。

#### **user-source-add-connection (usac) (ユーザソース) (接続名) (サーバアドレス) (サーバポート) [オプション]**

ユーザソースの新しいサーバ接続を追加します。

次のオプションを受け入れます。

-a, --accept-certificate - SSL モードを選択した場合に、ユーザソースが提示する証明書を受け入れます。このオプションは、スクリプトの場合にインタラクティブモードにならないようにするために設けられています。証明書を表示してから承認することをお勧めします。

-s, --use-ssl - 新しいサーバのアクセスに SSL を使用するように指定します。

#### **user-view-all-bundles-status (uvabs) (ユーザパス) [オプション]**

ユーザに割り当てられたバンドルの詳細な展開ステータスを表示します。

次のオプションを受け入れます。

--statusfilter=[ステータスタイプ][...] - バンドル配布およびインストールイベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

#### **user-view-all-policies-status (uvaps) (ユーザパス) [オプション]**

ユーザに割り当てられたポリシーの詳細な展開ステータスを表示します。

次のオプションを受け入れます。

--statusfilter=[ステータスタイプ][...] - ポリシー適用イベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。



## ワークステーションコマンド

ワークステーションを管理するコマンドです。ワークステーションコマンドは、長い形式では workstation- から始まり、短い形式では w から始まります。

次のコマンドはすべて、「[グローバルオプション](#)」のセクションで一覧表示されるオプションフラグを受け入れます。さらに、個別のオプションは、各コマンドでリストされたとおりに受諾されます。

- 130 ページの「workstation-add-bundle (wab) (ワークステーションオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]」
- 132 ページの「workstation-add-policy (wap) (ワークステーションオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- 132 ページの「workstation-copy-relationships (wcr) (ワークステーションパス) (ワークステーションオブジェクトパス) [...] [オプション]」
- 133 ページの「workstation-delete (wd) (ワークステーションオブジェクトパス) [...] [オプション]」
- 134 ページの「workstation-folder-create (wfc) (フォルダ名) [親フォルダ] [オプション]」
- 134 ページの「workstation-group-add (wga) (ワークステーショングループパス) (ワークステーションパス) [...] [オプション]」
- 134 ページの「workstation-group-create (wgc) (グループ名) [親フォルダ] [オプション]」
- 134 ページの「workstation-group-members (wgm) [オプション] (ワークステーショングループパス)」
- 134 ページの「workstation-group-remove (wgr) (ワークステーショングループパス) (ワークステーションパス) [...] [オプション]」
- 135 ページの「workstation-info (wi) (ワークステーションパス)」
- 135 ページの「workstation-list (wl) [フォルダ] [オプション]」
- 135 ページの「workstation-list-bundles (wlb) (ワークステーションオブジェクトパス) [オプション]」
- 135 ページの「workstation-list-groups (wlg) (ワークステーションパス) [オプション]」
- 135 ページの「workstation-list-policies (wlp) (ワークステーションオブジェクトパス) [オプション]」
- 136 ページの「workstation-move (wmv) (workstation object path) [destination folder path]」
- 136 ページの「workstation-refresh (wrf) (ワークステーションオブジェクトパス) [...] [オプション]」
- 136 ページの「workstation-remove-bundle (wrb) (ワークステーションオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]」

- ◆ 136 ページの「workstation-remove-policy (wrp) (ワークステーションオブジェクトパス) (ポリシーまたはポリシーグループ) [...] [オプション]」
- ◆ 136 ページの「workstation-rename (wr) (ワークステーションオブジェクトパス) (新しい名前)」
- ◆ 137 ページの「workstation-reorder-policies (wrop) (ワークステーションオブジェクトパス) (現在の位置) (新しい位置)」
- ◆ 137 ページの「workstation-retire (wret) (ワークステーションオブジェクトパス)」
- ◆ 137 ページの「workstation-set-as-test (wsat) (ワークステーションオブジェクトパス)」
- ◆ 137 ページの「workstation-set-as-nontest (wsan) (ワークステーションオブジェクトパス)」
- ◆ 137 ページの「workstation-unretire (wuret) (ワークステーションオブジェクトパス)」
- ◆ 137 ページの「workstation-view-all-bundles-status (wvabs) (ワークステーションパス) [オプション]」
- ◆ 138 ページの「workstation-view-all-policies-status (wvaps) (ワークステーションパス) [オプション]」
- ◆ 138 ページの「workstation-wakeup (ww) (ワークステーションオブジェクトパス) [...] [オプション]」

**workstation-add-bundle (wab) (ワークステーションオブジェクトパス) (バンドルまたはバンドルグループ) [...] [オプション]**

バンドルをワークステーションオブジェクトに割り当てます。

次のオプションを受け入れます。

- f, --folder=[バンドルフォルダ] - /バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。
- i, --icon-location=[アプリケーションロケーション XML ファイル] - バンドルアプリケーション用のアイコンを配置する場所を含む XML ファイル。XML ファイル形式については、IconLocation.xml を参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/bundles、Windows サーバでは *Installation\_directory:\Novell\Zenworks\share\zman\samples\bundles* に保存されています。
- d, --distribution-schedule=[配布スケジュール XML ファイル] - 配布スケジュールを含む XML ファイル。
- l, --launch-schedule=[起動スケジュール XML ファイル] - 起動スケジュールを含む XML ファイル。

- a, --availability-schedule=[ 可用性スケジュール XML ファイル ] - 可用性スケジュールを含む XML ファイル。スケジュール XML ファイルテンプレートについては、XML ファイルを参照してください。Linux サーバでは /opt/novell/zenworks/share/zman/samples/schedules、Windows サーバでは *Installation\_directory*:\Novell\Zenworks\share\zman\samples\schedules に保存されています。
- b, --install-schedule=[ インストールスケジュール XML ファイル ] - インストールスケジュールを含む XML ファイル。
- i, --install-immediately - 配布後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- L, --launch-immediately - インストール後ただちにバンドルをインストールします。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- n, --distribute-now - バンドルをただちに配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule および --distribute-on-device-refresh オプションは無視されます。--distribute-now、--distribute-on-device-refresh、および --distribution-schedule オプションは配布スケジュールの設定に使用しますが、互いに同時に使用することはありません。最初に --distribute-now オプションを検討してから、--distribute-on-device-refresh オプションと --distribution-schedule を使用してください。
- r, --distribute-on-device-refresh - デバイス更新時にバンドルを配布するよう配布スケジュールを設定します。このオプションを指定すると、--distribution-schedule オプションは無視されます。
- s, --launch-on-device-refresh - デバイス更新時にバンドルを起動するよう起動スケジュールを設定します。このオプションを指定すると、--launch-schedule オプションは無視されます。
- w, --wakeup-device-on-distribution - バンドルの配布中にデバイスがシャットダウンした場合、Wake-On-LAN を使用してデバイスを起動します。このオプションを使用するには、配布スケジュールも指定する必要があります。配布スケジュールは、--distribution-schedule、--distribute-now、または --distribute-on-device-refresh オプションを使用して指定することができます。
- D, --dry-run - 要求されたアクションをテストして表示しますが、実際には実行しません。
- B, --broadcast=[ ブロードキャストアドレス ][...] - Wake-On-LAN マジックパケットをブロードキャストするのに使用するアドレスのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。有効な IP アドレスが、有効な値です。
- S, --server=[/ デバイスを基準にしたプライマリサーバオブジェクトまたはプロキシサーバオブジェクトの相対パス ][...] - デバイスを起動するのに使用するプライマリまたはプロキシサーバオブジェクトのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。

- C, --retries=[再試行回数]- デバイスに Wake-On-LAN マジックパケットを送信する回数。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 0 ~ 5 の範囲である必要があります。デフォルト値は「1」です。
- T, --timeout=[再試行間隔]- 2 回の再試行の間隔。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 2 ~ 10 の範囲である必要があります (単位は分)。デフォルト値は「2」です。

#### **workstation-add-policy (wap) (ワークステーションオブジェクトパス)(ポリシーまたはポリシーグループ)[...][オプション]**

ポリシーをワークステーションオブジェクトに割り当てます。  
次のオプションを受け入れます。

- c, --conflicts=[ポリシー競合解決順序]- ポリシー競合の解決方法を決定します。有効な値は、userlast または 1、devicelast または 2、deviceonly または 3、useronly または 4 です。userlast の場合、デバイスに関連付けられたポリシーがまず適用され、それからユーザに関連付けられたポリシーが適用されます。devicelast の場合、ユーザに関連付けられたポリシーがまず適用され、それからデバイスに関連付けられたポリシーが適用されます。deviceonly の場合、ユーザに関連付けられたポリシーは無視されます。useronly の場合、デバイスに関連付けられたポリシーは無視されます。このオプションを指定しなかった場合には、userlast がデフォルト値として使用されます。
- e, --enforce-now - 割り当てられているすべてのデバイスに対してポリシーをすぐに強制します。
- f, --folder=[ポリシーフォルダ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

#### **workstation-copy-relationships (wcr) (ワークステーションパス)(ワークステーションオブジェクトパス)[...][オプション]**

ワークステーションからワークステーションに関係をコピーします。ワークステーションの関係にはバンドル割り当て、ポリシー割り当て、およびグループメンバーシップが含まれます。

次のオプションを受け入れます。

- r, --relationship-type: コピーする関係タイプ。有効な値は、**[bundles]**、**[policies]**、および **[groups]** です。オプション **[bundles]** は、ソースデバイスの関連付けられたすべてのバンドル割り当てをコピーします。オプション **[policies]** は、ソースデバイスの関連付けられたすべてのポリシー割り当てをコピーします。オプション **[groups]** は、ソースデバイスに関連付けられたグループのメンバーとしてターゲットデバイスを追加します。
- a, --assignment-options: 割り当てオプション。有効な値は、**[assign-to-group-members]**、**[assign-to-folder-members]**、および **[assign-to-folder-members-recursively]** です。オプション **[assign-to-group-members]** は、ターゲットグループのメンバーに割り当てをコピーします。オプション **[assign-to-folder-members]** は、ターゲットフォルダ内のデバイスに割り当てをコピーします。オプション **[assign-to-folder-members-recursively]** は、ターゲットフォルダおよびそのサブフォルダ内のデバイスに割り当てをコピーします。

- c, --conflict-resolution: 既存の割り当ての競合解決オプション。有効な値は、[delete-existing-assignments] および [replace-existing-assignments] です。オプション [delete-existing-assignments] は、ターゲットデバイスの既存のすべての割り当てを削除します。オプション [replace-existing-assignments] は、選択した割り当てでターゲットデバイスの既存の割り当てを置き換えます。
- g, --group-membership: ターゲットオブジェクトタイプ、グループおよびフォルダのグループメンバーシップオプション。有効な値は、[add-group-members]、[add-folder-members]、[add-folder-members-recursively]、および [delete-existing-group-membership] です。オプション [add-group-members] は、ソースデバイスがメンバーであるグループのメンバーとしてターゲットグループのメンバーを追加します。オプション [add-folder-members] は、ソースデバイスがメンバーであるグループのメンバーとしてターゲットフォルダ内にデバイスを追加します。オプション [add-folder-members-recursively] は、ソースデバイスがメンバーであるグループのメンバーとしてターゲットフォルダとそのサブフォルダ内にデバイスを追加します。オプション [delete-existing-group-membership] は、ターゲットデバイスがメンバーである既存のグループメンバーシップを削除します。

次に例を示します。

ソースワークステーションのすべての関係を複数のワークステーションにコピーするには：  
`zman workstation-copy-relationships workstationfolder/sourceworkstation --relationship-type=bundles,policies,groups workstationfolder/workstation workstationfolder/workstationfolder1 workstationfolder/workstationgroup`

ソースワークステーションのポリシー割り当てをターゲットワークステーションフォルダ内のワークステーションに再帰的にコピーするには：  
`zman wcr workstationfolder/sourceworkstation --relationship-type=bundlesworkstationfolder1 workstationfolder2 -assignment-options=assign-to-group-member,assign-to-folder-members`

ソースワークステーションから割り当てをコピーしているときにターゲットワークステーションの競合バンドル割り当てを置き換えるには：  
`zman wcr workstationfolder/sourceworkstation -r=policies workstationfolder/workstation -conflict-resolution=delete-existing-assignments`

ターゲットワークステーショングループのワークステーションをソースワークステーションのグループに追加するには：  
`zman wcr workstationfolder/sourceworkstation -r=groups workstationgroupgroup-membership=add-folder-members-recursively`

### **workstation-delete (wd) (ワークステーションオブジェクトパス)[...][オプション]**

1 つまたは複数のワークステーションオブジェクトを削除します。

(ワークステーションオブジェクトパス)[...]-/デバイス/ワークステーションを基準にした、ワークステーションオブジェクト(ワークステーション、ワークステーションフォルダ、またはワークステーショングループ)の相対パス。ワイルドカード\* は、引用符で囲まれている場合には、オブジェクト名に使用できます。オブジェクトを削除するのにワイルドカードを使用する場合は、注意します。

次のオプションを受け入れます。

-r, --recursive - フォルダ内のオブジェクトを再帰的に削除します。

-f, --folder=[ワークステーションフォルダ]-/デバイス/ワークステーションを基準にした、ワークステーションフォルダの相対パス。このオプションを指定すると、ワークステーションオブジェクトのパスは、このフォルダに関連して決定されます。これは、同じフォルダで複数のワークステーションオブジェクトを指定するのに使用できます。

#### **workstation-folder-create (wfc) (フォルダ名) [親フォルダ] [オプション]**

ワークステーションを格納するための新しいフォルダを作成します。

次のオプションを受け入れます。

--desc=[説明]- フォルダの説明。

#### **workstation-group-add (wga) (ワークステーショングループパス) (ワークステーションパス) [...] [オプション]**

ワークステーションをワークステーショングループに追加します。

次のオプションを受け入れます。

-f, --folder=[ワークステーションフォルダ]-/デバイス/ワークステーションを基準にした、ワークステーションフォルダの相対パス。このオプションを指定すると、ワークステーションオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のワークステーションオブジェクトを指定するのに使用できます。

#### **workstation-group-create (wgc) (グループ名) [親フォルダ] [オプション]**

ワークステーショングループを作成して、メンバーを追加します。

次のオプションを受け入れます。

--desc=[説明]- グループの説明。

-m, --members=[ワークステーションパス] [...] -/デバイス/ワークステーションを基準にした、ワークステーションの相対パス。

-f, --folder=[ワークステーションフォルダ]-/デバイス/ワークステーションを基準にした、ワークステーションフォルダの相対パス。このオプションを指定すると、ワークステーションオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のワークステーションオブジェクトを指定するのに使用できます。

#### **workstation-group-members (wgm) [オプション] (ワークステーショングループパス)**

ワークステーショングループまたは動的ワークステーショングループのメンバーを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[スクロールサイズ]- 一度に表示させる結果の数。

#### **workstation-group-remove (wgr) (ワークステーショングループパス) (ワークステーションパス) [...] [オプション]**

ワークステーションをワークステーショングループから削除します。

次のオプションを受け入れます。

-f, --folder=[ワークステーションフォルダ] - / デバイス / ワークステーションを基準にした、ワークステーションフォルダの相対パス。このオプションを指定すると、ワークステーションオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のワークステーションオブジェクトを指定するのに使用できます。

#### **workstation-info (wi) (ワークステーションパス)**

ワークステーションに関する詳細情報が一覧表示されます。

#### **workstation-list (wl) [フォルダ] [オプション]**

ワークステーションオブジェクトを一覧表示します。

次のオプションを受け入れます。

-r, --recursive - サブフォルダを含め、結果を再帰的に一覧表示します。このオプションを指定しなかった場合は、サブフォルダの内容は一覧表示されません。  
-n, --namefilter=[フィルタ文字列] - 指定した名前に一致する結果を表示します。引用符で囲む限り、ワイルドカード \* および ? を使用できます。  
-t, --typefilter=[タイプフィルタ][...] - 指定したオブジェクトタイプのカンマ区切りリストに一致する結果が表示されます。有効な値は、device、group、および folder です。  
-c, --count - 結果の数が表示されます。  
-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。  
-f, --flagfilter=[フラグフィルタ][...] - 指定したフラグのカンマ区切りリストに一致する結果が表示されます。有効な値は、retired、notretired、test、non-test です。

#### **workstation-list-bundles (wlb) (ワークステーションオブジェクトパス) [オプション]**

ワークステーションオブジェクトに割り当てられたバンドルを一覧表示します。

次のオプションを受け入れます。

-a, --all - 有効および無効のバンドル両方を一覧表示します。  
-e, --effective - 有効なバンドルのみ一覧表示します。  
-s, --scrollsize=[スクロールサイズ] - 一度に表示される結果の数。

#### **workstation-list-groups (wlg) (ワークステーションパス) [オプション]**

特定のワークステーションがメンバーであるグループを一覧表示します。

次のオプションを受け入れます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

#### **workstation-list-policies (wlp) (ワークステーションオブジェクトパス) [オプション]**

ワークステーションオブジェクトに割り当てられたポリシーを一覧表示します。

次のオプションを受け入れます。

-a, --all - 有効および無効のポリシー両方を一覧表示します。  
-e, --effective - 有効なポリシーのみ一覧表示します。  
-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

**workstation-move (wmv) (workstation object path) [destination folder path]**

ワークステーションオブジェクトを別のフォルダに移動します。

**workstation-refresh (wrf) (ワークステーションオブジェクトパス)[...][オプション]**

ワークステーション内の ZENworks Agent を更新します。

次のオプションを受け入れます。

-f, --folder=[ワークステーションフォルダ]- / デバイス / ワークステーションを基準にした、ワークステーションフォルダの相対パス。このオプションを指定すると、ワークステーションオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のワークステーションオブジェクトを指定するのに使用できます。

--server=<server/server folder/server group object path>[...] - /Devices/Servers を基準にした、プライマリサーバオブジェクト (サーバ、サーバフォルダ、またはサーバグループ) の相対パス。クイックタスク通知を送信するために使用します。

--notify=<minutes> - すべてのデバイスに通知する時間 (分)。

--expire=<minutes> - クイックタスクが作成されてから期限切れになるまでの時間 (分)。ただちに期限切れにするには、0 を指定します。

--currentserver - 現在のプライマリサーバにクイックタスクを追加します。

--anyserver - 任意のプライマリサーバにクイックタスクを追加します。

--expirenever - プライマリサーバでクイックタスクを期限切れにしません。

**workstation-remove-bundle (wrb) (ワークステーションオブジェクトパス)(バンドルまたはバンドルグループ)[...][オプション]**

ワークステーションオブジェクトに割り当てられたバンドルを削除します。

次のオプションを受け入れます。

-f, --folder=[バンドルフォルダ]- / バンドルを基準にした、バンドルフォルダの相対パス。このオプションを指定すると、バンドルオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のバンドルオブジェクトを指定するのに使用できます。

**workstation-remove-policy (wrp) (ワークステーションオブジェクトパス)(ポリシーまたはポリシーグループ)[...][オプション]**

ワークステーションオブジェクトに割り当てられたポリシーを削除します。

次のオプションを受け入れます。

-f, --folder=[ポリシーフォルダ]- / ポリシーを基準にした、ポリシーフォルダの相対パス。このオプションを指定すると、ポリシーオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のポリシーオブジェクトを指定するのに使用できます。

**workstation-rename (wr) (ワークステーションオブジェクトパス)(新しい名前)**

ワークステーションオブジェクトを名前変更します。



**workstation-reorder-policies (wrop) (ワークステーションオブジェクトパス) (現在の位置) (新しい位置)**

ワークステーションに割り当てられたポリシーの順序を変更します。workstation-list-policies コマンドを使用して、割り当てられたポリシーの順序を取得します。

**workstation-retire (wret) (ワークステーションオブジェクトパス)**

次の更新時に、ZENworks システムから、選択したワークステーションをリタイアさせます。デバイスのリタイアはデバイスの削除とは異なります。デバイスをリタイアしても、GUID は保持されます (これとは反対に、デバイスを削除すると、GUID も削除されます)。そのため、インベントリ情報はすべて保持されたままでアクセス可能ですが、ポリシーおよびバンドルの割り当てはすべて削除されます。後でデバイスを再度アクティブ化する場合、割り当ては復元されます。

次のオプションを受け入れます。

-i, --immediate - デバイスのリフレッシュで強制的にデバイスをすぐにリタイアします。

**workstation-set-as-test (wsat) (ワークステーションオブジェクトパス)**

ワークステーションをテストデバイスとして設定します。

例

workstation1 をテストデバイスとして設定する場合：

```
zman wsat workstation1
```

**workstation-set-as-nontest (wsan) (ワークステーションオブジェクトパス)**

ワークステーションをテストデバイス以外として設定します。

例

workstation1 をテストデバイス以外として設定する場合：

```
zman wsan workstation1
```

**workstation-unretire (wuret) (ワークステーションオブジェクトパス)**

次のリフレッシュ時に選択されたワークステーションを再度有効にして、デバイスが以前所有していたすべてのポリシーとバンドルの割り当てを再割り当てします。

次のオプションを受け入れます。

-i, --immediate - デバイスのリフレッシュで強制的にデバイスをすぐにリタイア解除します。

**workstation-view-all-bundles-status (wvabs) (ワークステーションパス) [オプション]**

ワークステーションに割り当てられたバンドルの詳細な展開ステータスを表示します。次のオプションを受け入れます。

--statusfilter=[ステータスタイプ][...] - バンドル配布およびインストールイベントのステータスに対するフィルタです。有効な値は S、R、C、A、F、および P(Success、Partial Success、Completed、Partial Complete、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

-s, --scrollsize=[スクロールサイズ] - 一度に表示させる結果の数。

例

workstation1 に割り当てられたバンドルの展開状態を表示する場合：

```
zman wvabs workstation1
```

バンドル展開が成功または保留状態の workstation1 に割り当てられたバンドルの展開状態を表示する場合：

```
zman wvabs workstation1 --statusfilter S,P
```

#### **workstation-view-all-policies-status (wvaps) (ワークステーションパス)[オプション]**

ワークステーションに割り当てられたポリシーの詳細な展開ステータスを表示します。次のオプションを受け入れます。

`--statusfilter=[ステータスタイプ][...]` - ポリシー適用イベントのステータスでフィルタします。有効な値は、S、F、および P (Success、Failure、および Pending) です。ステータスタイプのカンマ区切りリストを指定できます。

`-s, --scrollsize=[スクロールサイズ]` - 一度に表示させる結果の数。

#### **workstation-wakeup (ww) (ワークステーションオブジェクトパス)[...][オプション]**

Wake-On-Lan を使用してワークステーションを起動します。

次のオプションを受け入れます。

- `-f, --folder=[ワークステーションフォルダ]` - / デバイス / ワークステーションを基準にした、ワークステーションフォルダの相対パス。このオプションを指定すると、ワークステーションオブジェクトのパスは、このフォルダに対して決定されます。これは、同じフォルダで複数のワークステーションオブジェクトを指定するのに使用できます。
- `-b --broadcast=[ブロードキャストアドレス][...]` - Wake-On-LAN マジックパケットをブロードキャストするのに使用するアドレスのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。有効な IP アドレスが、有効な値です。
- `-s, --server=[プライマリサーバオブジェクトまたはプロキシサーバオブジェクトのパスであり、/ デバイスへの相対パスです][...]` - デバイスを起動するのに使用するプライマリまたはプロキシサーバのカンマ区切りリスト。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。
- `-C, --retries=[再試行回数]` - デバイスに Wake-On-LAN マジックパケットを送信する回数。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 0 ~ 5 の範囲である必要があります。デフォルト値は「1」です。
- `-t, --timeout=[再試行間隔]` - 2 回の再試行の間隔。このオプションは、Wake-On-LAN を使用してデバイスを起動する場合にのみ使用します。値は 2 ~ 10 の範囲である必要があります (単位は分)。デフォルト値は「2」です。

## ZENworks サーバコマンド

これらのコマンドは、ZENworks がインストールされているサーバを管理する場合に使用します。ZENworks サーバコマンドは、長い形式では `zenserver-` から始まり、短い形式では `zs` から始まります。

- ◆ 139 ページの「`zenserver-backup (zsb)` (ファイルパス) [パスフレーズ]」
- ◆ 139 ページの「`zenserver-restore (zsr)` (ファイルパス) [パスフレーズ]」

### `zenserver-backup (zsb)` (ファイルパス) [パスフレーズ]

このサーバを別の ZENworks Server に複製するために必要な設定ファイルをバックアップします。

(ファイルパス) - ZENworks Server 設定ファイルがバックアップされるファイルのパス。サーバに埋め込みデータベースが含まれている場合、`database-backup` コマンドを使用して、データベースをバックアップします。別のサーバインストールでサーバを復元するには、`zenserver-restore` コマンドを使用して設定ファイルを最初に復元してから、データベースを復元する必要があります。コマンドによって、バックアップファイルを暗号化するためのパスフレーズが求められます。指定したパスフレーズをメモしてください。サーバを復元するときに、これを入力する必要があります。

[パスフレーズ] - ZENworks サーバ設定ファイルがバックアップされるファイルのパス。サーバに埋め込みデータベースが含まれている場合、`database-backup` コマンドを使用して、データベースをバックアップします。別のサーバインストールでサーバを復元するには、`zenserver-restore` コマンドを使用して設定ファイルを最初に復元してから、データベースを復元する必要があります。バックアップファイルを暗号化するパスフレーズのコマンドプロンプト。

### `zenserver-restore (zsr)` (ファイルパス) [パスフレーズ]

クラッシュした ZENworks Server のバックアップされた設定ファイルを新しい ZENworks Server インストールに復元します。

(ファイルパス) - 復元されるバックアップされた設定ファイルを含むファイルのパス。サーバは復元されているサーバと同じ DNS 名と IP アドレスを持つ必要があります。コマンドによって、バックアップされたファイルを暗号化するためのパスフレーズが求められたら、ファイルがバックアップされた際にファイルを暗号化するために使用されたパスフレーズを入力してください。

[パスフレーズ] - ZENworks サーバ設定ファイルがバックアップされるファイルのパス。サーバに埋め込みデータベースが含まれている場合、`database-backup` コマンドを使用して、データベースをバックアップします。別のサーバインストールでサーバを復元するには、`zenserver-restore` コマンドを使用して設定ファイルを最初に復元してから、データベースを復元する必要があります。バックアップファイルを暗号化するパスフレーズのコマンドプロンプト。

## グローバルオプション

次のコマンドは、どの zman コマンドにも適用できます。

- ◆ 140 ページの 「-h、 --help」
- ◆ 140 ページの 「--usage」
- ◆ 140 ページの 「--terse」
- ◆ 140 ページの 「--debug」
- ◆ 140 ページの 「-V, --verbose」
- ◆ 140 ページの 「--quiet」
- ◆ 140 ページの 「-U, --User=< ユーザ名 >」
- ◆ 140 ページの 「-P, --Password=< パスワード >」
- ◆ 140 ページの 「---host=< ホスト >」
- ◆ 141 ページの 「--port=< ポート >」
- ◆ 141 ページの 「--cleartext」
- ◆ 141 ページの 「-R, --Redirect=< ファイルパス >」
- ◆ 141 ページの 「-G, --ignore-global-file」

### **-h, --help**

さまざまなレベルでヘルプを提供します。ヘルプの使用方法の詳細については、「[使用方法](#)」のセクションを参照してください。

### **--usage**

コマンドの構文を表示します。

### **--terse**

簡潔な出力を表示します。

### **--debug**

デバッグ出力を表示します。

### **-V, --verbose**

詳細な出力を有効にします。

### **--quiet**

出力を行いません。エラーメッセージだけが出力されます。

### **-U, --User=< ユーザ名 >**

ユーザ名を指定します。指定しなかった場合は、入力するように求められます。

### **-P, --Password=< パスワード >**

パスワードを指定します。指定しなかった場合は、入力するように求められます。

### **---host=< ホスト >**

接続先のホスト名または IP アドレスを指定します ( デフォルト : localhost)。

**--port=< ポート >**

サーバがリスンするポートを指定します ( デフォルト :443)。

**--cleartext**

デバッグのために SSL を無効化します。ポートがクリアテキストポートに設定されていない場合は、デフォルトでポート 80 に接続します。

**-R, --Redirect=< ファイルパス >**

画面の出力をファイルにリダイレクトします。コマンドラインリダイレクション演算子 (>) の代わりにこのオプションを使用すると、データは UTF-8 形式で保存され、英語以外の文字も保持されます。

**-G, --ignore-global-file**

zman-config.properties で指定されたグローバルオプションファイルを無視します。

## セキュリティ

各コマンドを実行するには、管理者の許可が必要です。コマンドごとに管理者資格情報を入力しなくて済むようにするには、admin-store-credential コマンドを使用して資格情報を保存します。資格情報は暗号化されて、ユーザがコマンドを実行しているオペレーティングシステムのホームフォルダに保存されます。資格情報はログオンしているユーザにのみ適用されます。コマンドをすべて実行し終わったら、admin-clear-credential コマンドを実行して、情報をクリアしておくのが安全です。

---

**重要:** ユーザ名とパスワードを入力するコマンドラインオプションもありますが、これらの使用は避けてください。Linux では、これらは ps のリストに表示されます。さらに Windows でも Linux でも、上矢印キーや下矢印キーを押すと、以前に入力したコマンドが表示されます。入力を求められたときに資格情報を入力するか、admin-store-credential コマンドを使用してください。

---

## ファイル

zman-config.properties は、zman を設定するためのプロパティファイルで、次の場所にあります。

- **Windows の場合:** %ZENWORKS\_HOME%\conf\zman\properties
- **Linux の場合:** /etc/opt/novell/zenworks/zman/properties

利用可能なオプションは、次のとおりです。

- **RENDERER\_SCROLLSIZE:** リストコマンドの出力をグローバルに制御します。結果が取得され、指定されたスクロールサイズの数で表示されます。この設定は、リストコマンドで --scrollsize オプションを使用することで、上書きできます。
- **GLOBAL\_OPTIONS\_FILE:** グローバルオプションは、ファイル内に保存し、このプロパティに対する値としてファイルのパスを指定することで、すべてのコマンドに追加できます。Window ファイルでは、\ のかわりに \\ を使用します。ファイルのコンテンツは、コマンドと同じように 1 行内にある必要があります。

例を次に示します。

```
GLOBAL_OPTIONS_FILE=C:\\zman\\globaloptions.txt
```

ファイルのサンプルのコンテンツは次のとおりです。

```
--host=123.34.45.56 --port=8080 -cleartext --verbose
```

このファイルに保存されているオプションをバイパスするには、`-ignore-global-file` オプションを使用します。`-U|--User` および `P|--Password` 以外のすべてのグローバルオプションは、グローバルオプションファイルに保存できます。`admin-store-credential` および `admin-clear-credential` は、管理者資格情報を各コマンドに追加する場合に使用します。管理者資格情報の使用のガイドラインについては、「[セキュリティ](#)」のセクションを参照してください。

- **GLOBAL\_OPTIONS\_HELP:** コマンドのグローバルオプションは、すべてのコマンドのヘルプで表示されます。グローバルオプションの表示をオフにするには、`GLOBAL_OPTIONS_HELP=false` を設定します。
- **DEBUG\_LEVEL:** Windows ではインストールドライブ `:\Novell\zenworks\logs`、Linux では `/var/opt/novell/log/zenworks` にある `zman.log` ファイルに記録されるデバッグメッセージの量を制御します。デフォルトのレベル 3 では、入力されたコマンドと、エラーが発生した場合の例外スタックトレースが記録されます。レベル 4 では、入力されたコマンド、コマンドの出力、および詳細デバッグメッセージが記録されます。

## 終了コード

`zman` は、0 から 255 の範囲で終了コードを返します。0 は、コマンドの実行の成功を示します。1 から 255 は、実行中のエラーを示します。

### 0 - 成功

解析エラーに対する終了コード：

- 1 - 不明な解析エラーが発生しました。
- 2 - 必須の引数が指定されていません。
- 3 - 不明なフラグが指定されています。
- 4 - 値はスイッチパラメータに関して指定されています。
- 5 - フラグ付きオプションに値が指定されていません。
- 6 - 無効な追加引数が指定されています。
- 7 - 無効な値が指定されています。
- 8 - 値を有効なタイプに変換できません。
- 9 - 不明なコマンド。
- 10 - 少なくとも 1 つのオプションが必要な場合に、オプションが入力されていません。たとえば、権限の変更には、`Assign` または `Revoke` オプションのうち少なくとも 1 つを指定する必要があります。

一般エラーに対する終了コード：

- 11 - 不明なエラーが発生しました。
- 12 - 従属する `zman.jar` ファイルが存在しないため、機能が利用できません。

- 13 - 認証に失敗しました。
- 14 - ユーザ名またはパスワードとして、空の文字列が指定されました、または、指定されたパスワードが、最小長さの要件を満たしていません。
- 15 - ZENworks ライセンスがの期限が切れています。
- 16 - 接続に失敗しました。データベースサーバがダウンしている可能性があります。
- 17 - ZENworks 管理者には、この操作を実行する十分な権限がありません。
- 18 - SSL 通信を確立するための証明書を読み込めません。
- 19 - このコマンドはローカルホストでのみ実行できます。

一般オブジェクト関連エラーに対する終了コード：

- 21 - 指定したオブジェクトが予期されたタイプではありません。たとえば、デバイスの代わりにバンドルが指定されています。
- 22 - オブジェクトが見つかりませんでした。
- 23 - 同じ名前を持つオブジェクトがすでに指定したフォルダに存在します。
- 24 - 指定したオブジェクトが指定したフォルダ以外のディレクトリにあります。
- 25 - 親フォルダはサブフォルダに移動できません。
- 26 - オブジェクトの名前を変更できません。
- 27 - オブジェクトを削除できません。
- 28 - 空でないフォルダを削除しようとしています。
- 29 - オブジェクトはすでにグループのメンバーです。
- 30 - オブジェクトはグループのメンバーではありません。
- 31 - オブジェクトはすでにデバイスまたはユーザに割り当てられています。
- 32 - オブジェクトはまだデバイスまたはユーザに割り当てられていません。
- 33 - オブジェクトタイプが、デバイスまたはユーザに割り当てるには無効です。
- 34 - デバイスへのオブジェクトの割り当てが許可されていません。たとえば、ユーザ固有のポリシーはデバイスに割り当てることができません。

その他の一般エラーに対する終了コード：

- 41 - 指定した名前が無効です。たとえば、オブジェクトの名前には、\* または % などの特殊文字を含めることはできません。
- 42 - スラッシュ (/) で始まる絶対パスが必要です。
- 43 - 無効なスケジュールが指定されています。
- 44 - XML ファイルに、予期された入力が含まれていませんでした。たとえば、エクスポート済みバンドル XML ファイルが必要であるにもかかわらず、エクスポート済みアクションセット XML ファイルが入力として指定された、などの場合です。
- 45 - XML ファイルが予期されたスキーマと同じではありませんでした。
- 46 - 無効な GUID が指定されています。
- 47 - 指定された値が最大長を超えています。
- 48 - 従属する引数が指定されていません。たとえば、バンドル配布時にただちにインストールするオプションがコマンドで指定されているにもかかわらず、配布スケジュールが指定されていません。

ファイル関連エラーに対する終了コード：

- 61 - ディレクトリが予期されています。
- 62 - ファイルが存在しないか、ディレクトリです。
- 63 - ファイルがすでに存在します
- 64 - ファイルを読み取り中にエラーが発生しました。
- 65 - ファイルへの書き込み中にエラーが発生しました。
- 66 - ファイルを作成中にエラーが発生しました。
- 67 - ディレクトリの作成中にエラーが発生しました。
- 68 - ファイルのコピー中にエラーが発生しました。
- 69 - ファイルのアップロード中にエラーが発生しました。
- 70 - ターゲットデバイスに十分なディスク容量がないためファイルをロードできません。

登録コマンドに固有の終了コード：

- 71 - 登録キーまたはルールセットに関連付けられているフォルダとグループが同じデバイスタイプではありません。
- 72 - ルールセットは少なくとも 1 つのルールを持つ必要があります。

バンドルコマンドおよびポリシーコマンドに共通の終了コード：

- 76 - コンテンツの作成に失敗しました。
- 77 - その名前のファイルがすでに存在するため、一時ディレクトリへのコンテンツファイルのコピーがキャンセルされました。
- 78 - ファイルコンテンツを持つバンドルまたはポリシーの作成をコンテンツサーバではないサーバに作成しようとしています。

バンドルコマンドに固有の終了コード：

- 81 - バンドルアイコンに指定されたファイルが有効なイメージファイルではありません。
- 82 - アクションセットに並べ替え、削除、またはエクスポートするアクションが含まれていません。
- 83 - アクションセットに並べ替え、削除、またはエクスポートするアクションが任意の位置に含まれていません。
- 84 - アクションセットがバンドルタイプでサポートされていません。たとえば、Preboot アクションセットは、Windows バンドルではサポートされていません。
- 85 - アクションがアクションセットでサポートされていません。たとえば、Verify バンドルアクションは、Install アクションセットではサポートされていません。
- 86 - アクションが削除できません。
- 87 - アクションタイプが単体です。同じアクションタイプを 2 度以上アクションセットに追加することはできません。
- 88 - アクションタイプは従属バンドルをサポートしません。たとえば、Install MSI アクションには従属バンドルパスを指定できません。
- 89 - 指定した従属バンドルパスがバンドルの循環従属関係になります。
- 90 - バンドルの割り当て時に Wake-On-LAN スケジュールを作成できませんでした。
- 91 - バンドルの割り当て解除時に Wake-On-LAN スケジュールを削除できませんでした。

デバイスコマンドとユーザコマンドに共通の終了コード：



101 - デバイスまたはユーザに関連付けられているバンドルまたはポリシーがありません。

ユーザコマンドに固有の終了コード：

106 - ユーザソース証明書の期限が切れています。

107 - ユーザソースに接続するには SSL が必要です。

ライセンスコマンドに固有の終了コード：

116 - ライセンスを有効にできません。

118 - サーバに認証局ロールがありません。

119 - サーバに認証局ロールを有効にするために必要なファイルがありません。

120 - 使用されている認証局は ZENworks Internal Certificate Authority ではありません。

管理者コマンドに固有の終了コード：

121 - 権限が権限カテゴリに属していません。

データベースコマンドに固有の終了コード：

126 - データベースが組み込みデータベースではありません。

127 - SQL ステートメントは無効です。

128 - SQL 例外が発生しました。

展開コマンドに固有の終了コード：

131 - 無効な IP アドレスまたはホスト名が指定されています。

132 - デバイスに接続するために必要な資格情報が指定されていません。

133 - 管理対象デバイス、またはすでに展開がスケジュールされているデバイスが指定されています。

レポートコマンドに固有の終了コード：

136 - レポートを削除できませんでした。

137 - レポート生成の要求を送信できませんでした。

138 - 無効なレポートドキュメントです。

## ZeUS コマンド

zman の zeus コマンドは、最新の ZeUS MSI をゾーンにインポートして、現在の ZeUS 環境設定を取得するために使用します。

- ◆ 145 ページの「[zeus-msi-config | zmc](#)」
- ◆ 146 ページの「[zeus-msi-import | zmi \(ZeUS Msi へのパス\) \[-f|--force\]](#)」
- ◆ 146 ページの「[zac zeus-refresh / zeus-ref](#)」

**zeus-msi-config | zmc**

前回インポートされた ZeUS msi の設定を指定します。

#### **zeus-msi-import | zmi (ZeUS Msi へのパス) [-f|--force]**

最新の ZeUS MSI をゾーンにインポートします。

(ZeUS Msi へのパス) - ZeUS ディレクトリへのフルパスを入力します。

次のオプションを受け入れます。

-f, --force - バージョンが同じ場合でも MSI を再インポートします。これにより、サーバ内にある現在の zeus 環境設定が上書きされます。

#### **zac zeus-refresh / zeus-ref**

システム更新がデバイスに割り当てられている場合、そのシステム更新を取得します。

メモ : これは ZMAN コマンドではありません。

# zeninfocollect(1)

## 名前

zeninfocollect - Micro Focus ZENworks の使用時に発生する可能性がある問題の解決に役立つ情報を収集できるユーティリティです。

## 構文

```
./zeninfocollect.py [ オプション ]
```

## 説明

デフォルトでは、zeninfocollect は、キャッシュ、サーバ、クライアント、設定、ハードウェア、およびパッケージのデータに加え、ログファイルも収集します。収集された情報は、Windows では zip ファイル、Linux では tarball ファイルにパッケージ化され、指定した場所に保存されます。

デフォルトでは、zeninfocollect ユーティリティはどのデバイスにもインストールされません。

zeninfocollect ユーティリティをダウンロードして、Windows 管理デバイスで実行するには、次の手順に従います。

- 1 ZENworksInfocollect\_Win.zip ファイルを  
ZENworks\_Server\novell\zenworks\install\downloads\tools ディレクトリからコピーします。または、ZENworks コントロールセンター（[環境設定] タブ > [設定タスク] > [ZENworks ツールのダウンロード] > [管理ツール]）からファイルをダウンロードします。
- 2 管理デバイスに ZENworksInfocollect\_Win.zip ファイルを解凍します。
- 3 管理デバイスのコマンドプロンプトで、.zip の内容を抽出したディレクトリに移動し、zeninfocollect.exe [オプション] を実行します。

zeninfocollect ユーティリティをダウンロードして、Python がインストールされた Linux 管理デバイスで実行するには、次の手順に従います。

- 1 ZENworksInfocollect\_Linux.zip ファイルを ZENworks\_Server/novell/zenworks/install/downloads/tools ディレクトリからコピーします。または、ZENworks（[環境設定] タブ > [設定タスク] > [ZENworks ツールのダウンロード] > [管理ツール]）からファイルをダウンロードします。
- 2 管理デバイスに ZENworksInfocollect\_Linux.zip ファイルを解凍します。
- 3 chmod +x zeninfocollect.py コマンドを実行してファイルを実行可能にします。
- 4 管理デバイスのコマンドプロンプトで、zip の内容を抽出したディレクトリに移動し、./zeninfocollect.py [オプション]。

## オプションのリスト

- ◆ 148 ページの 「--output-dir=*directory\_name*」
- ◆ 148 ページの 「-h, --help」
- ◆ 148 ページの 「-a, --no-cache」
- ◆ 148 ページの 「-c, --no-client」
- ◆ 148 ページの 「-d, --no-hardware」
- ◆ 148 ページの 「-l, --no-logs」
- ◆ 148 ページの 「-o, --no-config」
- ◆ 148 ページの 「-p, --no-packages」
- ◆ 148 ページの 「-s, --no-server」

### **--output-dir=*directory\_name***

出力アーカイブを含めるディレクトリ名を指定します。ディレクトリを指定しなかった場合、出力はユーザの一時ディレクトリ (Windows 管理デバイスでは %TEMP%、Linux 管理デバイスでは /tmp) に保存されます。

### **-h, --help**

ヘルプ情報を表示します。

### **-a, --no-cache**

キャッシュデータを収集しません。

### **-c, --no-client**

クライアントデータを収集しません。

### **-d, --no-hardware**

ハードウェアデータを収集しません。

### **-l, --no-logs**

ログを収集しません。

### **-o, --no-config**

設定データを収集しません。

### **-p, --no-packages**

パッケージデータを収集しません。パッケージデータには、ZENworks Configuration Management パッケージリポジトリ内にあるパッケージのすべてのバージョン情報が含まれています。

### **-s, --no-server**

サーバデータを収集しません。

# zac for Windows(1)

## 名前

zac - Windows 管理対象デバイスにインストールされて実行される Micro Focus ZENworks Agent のコマンドライン管理インターフェースです。

## 構文

*zac command options*

## 説明

zac ユーティリティは、ソフトウェアバンドルのインストールと削除、ポリシーの適用、およびデバイスの登録と登録解除など、ZENworks 管理デバイス上でコマンドライン管理機能を実行します。

## 使用方法

ほとんどのコマンドには、長い形式と短い形式があります。

- 長い形式 : add-reg-key
- 短い形式 : ark

両方の形式が使用できる場合、コマンドは以下のように表示されます。

add-reg-key (ark) 引数

コマンドを使用する場合は、長い形式または短い形式のいずれかのみを入力します。

zac add-reg-key 引数

zac ark 引数

引数は必須にもオプションにもできます。必須引数は、山括弧で囲まれます (< 引数 >)。オプションの引数は角括弧で囲まれます ([ 引数 ])。引数にスペースを含める場合には、次のように引用符で囲みます。

zac ark "arg 1"

## ヘルプコマンド

- [149 ページの「/h または --help」](#)

**/h または --help**

コマンドについての情報を表示します。

## コマンドのリスト

- ◆ 150 ページの「認証サテライトサーバコマンド」
- ◆ 151 ページの「サテライトの役割コマンド」
- ◆ 151 ページの「バンドルコマンド」
- ◆ 152 ページの「証明書コマンド」
- ◆ 153 ページの「コレクションロールアップコマンド」
- ◆ 153 ページの「コンテンツ配布コマンド」
- ◆ 155 ページの「データベースコマンド」
- ◆ 155 ページの「イメージングコマンド」
- ◆ 155 ページの「インベントリコマンド」
- ◆ 156 ページの「場所コマンド」
- ◆ 156 ページの「ログコマンド」
- ◆ 156 ページの「OS ターゲットコマンド」
- ◆ 157 ページの「パッチ管理コマンド」
- ◆ 157 ページの「ポリシーコマンド」
- ◆ 158 ページの「登録コマンド」
- ◆ 160 ページの「リモート管理コマンド」
- ◆ 160 ページの「ステータスコマンド」
- ◆ 164 ページの「システムアップデートコマンド」
- ◆ 164 ページの「ZENworks Endpoint Security Management のコマンド」

## 認証サテライトサーバコマンド

- ◆ 150 ページの「`authentication server reconfigure (asr) [-t all|config|casa] [-u ユーザ名] [-p パスワード]`」

**authentication server reconfigure (asr) [-t all|config|casa] [-u ユーザ名] [-p パスワード]**

有効化された認証サテライトを再設定します。

例：

設定ファイルをサーバから取得する場合：

```
zac asr -t config
```

CASA 署名証明書を再設定する場合：

```
zac asr -t casa
```

サテライト全体を再設定する場合：

```
zac asr -t all -u Administrator -p パスワード
```

ユーザ名とパスワードが必要な場合、コマンドラインで指定しないと入力を求められます。

## サテライトの役割コマンド

- 151 ページの「`satellite server reconfigure (ssr) [-t jetty] [-u username] [-p password]`」
- 151 ページの「`Import-satellite-cert (isc) [-pk <private-key.der>] [-c <signed-server-certificate.der>] [-ca<signing-authority-public-certificate.der>] [-ks <keystore.jks>] [-ksp<keystore-pass-phrase>] [-a <signed-cert-alias>] [-cp<signed-cert-passphrase>] [-u <username>] [-p <password>] [ -rc ]]`」

### **satellite server reconfigure (ssr) [-t jetty] [-u username] [-p password]**

Jetty Web サーバを再設定します。

Jetty Web サーバを再設定する場合：

```
zac ssr -t jetty -u Administrator -p password
```

### **Import-satellite-cert (isc) [-pk <private-key.der>] [-c <signed-server-certificate.der>] [-ca<signing-authority-public-certificate.der>] [-ks <keystore.jks>] [-ksp<keystore-pass-phrase>] [-a <signed-cert-alias>] [-cp<signed-cert-passphrase>] [-u <username>] [-p <password>] [ -rc ]**

外部署名証明書を使用してサテライトデバイスを設定します。

rc - 管理者に対して再設定が要求されないように、サテライトサーバの再設定を確認します。

## バンドルコマンド

- 151 ページの「`bundle-install (bin) <バンドル表示名 >`」
- 151 ページの「`bundle-launch (bln) <バンドル表示名 > [-noSelfHeal]`」
- 152 ページの「`bundle-list (bl)`」
- 152 ページの「`bundle-props (bp) <バンドル表示名 >`」
- 152 ページの「`bundle-refresh (br) <バンドル表示名または GUID>`」
- 152 ページの「`bundle-uninstall (bu) <バンドル表示名 >`」
- 152 ページの「`bundle-verify (bv) <バンドル表示名 >`」

### **bundle-install (bin) <バンドル表示名 >**

指定したバンドルをインストールします。利用可能なバンドルと、その表示名のリストを取得するには、`bundle-list` コマンドを使用します。

例：

```
zac bin バンドル 1
```

### **bundle-launch (bln) <バンドル表示名 > [-noSelfHeal]**

指定したバンドルを起動します。利用可能なバンドルと、その表示名のリストを取得するには、`bundle-list` コマンドを使用します。

表示名に基づいたバンドルの起動例：

```
zac bln バンドル 1
```

表示名に基づいてバンドルを起動し、起動アクションが失敗した場合の自己回復設定をオフ ( デフォルトでは自己回復がオンの状態 ) にした例 :

```
zac bln バンドル 1 -noSelfHeal
```

#### **bundle-list (bl)**

デバイスおよびログインユーザに割り当てられているバンドルのリストを表示します。

例 :

```
zac bl
```

#### **bundle-props (bp) <バンドル表示名 >**

指定したバンドルのステータス、バージョン、GUID、および要件情報を表示します。利用可能なバンドルと、その表示名のリストを取得するには、bundle-list コマンドを使用します。

例 :

```
zac bln バンドル 1
```

#### **bundle-refresh (br) <バンドル表示名または GUID>**

指定されたバンドルの情報を更新します。

例 :

```
zac br バンドル 1
```

#### **bundle-uninstall (bu) <バンドル表示名 >**

指定したバンドルをアンインストールします。インストールされたバンドルと、その表示名のリストを取得するには、bundle-list コマンドを使用します。

例 :

```
zac bu バンドル 1
```

#### **bundle-verify (bv) <バンドル表示名 >**

インストールされたバンドル (bundle display name で指定されたもの) を検証して、削除されたり壊れたりしたものがないことを確認します。インストールされたバンドルと、その表示名のリストを取得するには、bundle-list コマンドを使用します。

例 :

```
zac bv バンドル 1
```

## 証明書コマンド

- ◆ [152 ページの「cert-info \(ci\) \[CA 証明書のファイルパス\] \[-u <ユーザ名> -p <パスワード>\]」](#)

#### **cert-info (ci) [CA 証明書のファイルパス] [-u <ユーザ名> -p <パスワード>]**

既知の ZENworks サーバごとに公開鍵証明書情報を一覧表示するか、デバイスの信頼するストアに信頼するルート証明書を追加します。ファイルは、ASN.1 DER フォーマットにすることも、----BEGIN CERTIFICATE---- と ----END CERTIFICATE-- で区切って Base64 エンコードすることもできます。



例：

既知の ZENworks サーバごとに証明書を一覧表示するには、次のコマンドを実行します。

```
zac ci
```

デバイスの信頼するストアに信頼するルート証明書を追加するには、次のコマンドを実行します。

```
zac ci c:\certs\mytrustcacert.der -u myuser -p mypassword
```

## コレクションロールアップコマンド

- [153 ページの「collection-point \(cp\)」](#)
- [153 ページの「collection-upload-orphans \(cuo\)」](#)

### collection-point (cp)

コレクションの役割の状態と設定を表示します。

例：

```
collection-point [wake]
```

wake - コレクションを実行するモジュール ( インベントリ、MD ステータス、メッセージ送信側 ) を起動します。

### collection-upload-orphans (cuo)

サテライトデバイス上の孤立したファイルを検索して、親コレクションサーバにロールアップするか、またはロールアップ済みの場合は削除します。

このコマンドは %zenworks\_home%\work\collection の下のフォルダにファイルリストを構築し、コレクション統計データベースの各エントリの元のアップロード情報を検索しようとしています。

データベースにファイルのエントリがある場合は、ファイルがロールアップされていないことを示し、ファイルをロールアップします。エントリにファイルがロールアップ済みであることが示されている場合は、サテライトデバイスのファイルを削除します。データベースにファイルのエントリがない場合は、ファイルをロールアップします。このコマンドは、アップロードまたは削除されなかったファイルのリストも表示します。

このコマンドを実行する前に、zac crw コマンドを実行して保留中のファイルを親サーバに送信する必要があります。

例：

```
zac cuo
```

## コンテンツ配布コマンド

- [154 ページの「cdp-checksum \(cchk\) \[-l:< ログへのパス >\]」](#)
- [154 ページの「cdp-verify-content \(cvc\) \[-c\] \[-l:< ログへのパス >\]」](#)

- ◆ 154 ページの「`cdp-import-content (cic) < コンテンツパス > [-l:< ログへのパス >]`」
- ◆ 154 ページの「`wake-cdp (cdp) [replicate | cleanup]`」

#### **cdp-checksum (cchk) [-l:< ログへのパス >]**

各ファイルのチェックサムを計算することによって、サテライトのコンテンツを検証します。

オプションのログファイルには、チェックサム比較結果の詳細が記録されます。

例：

```
zac cchk -l:"C:\Program Files\Novell\ZENworks\logs\cchk.log"
```

#### **cdp-verify-content (cvc) [-c] [-l:< ログへのパス >]**

この CDP でのコンテンツ ID とその同期状態のリストを、プライマリサーバが想定しているものと比較します。

次のオプションを使用できます。

c - ローカルコンテンツのチェックサムを計算します。

例：

```
zac cvc -l:"C:\Program Files\Novell\ZENworks\logs\cvc.log"
```

#### **cdp-import-content (cic) < コンテンツパス > [-l:< ログへのパス >]**

`content-path` で指定されたディレクトリから失われたコンテンツをインポートし、`log-path` で指定されたファイルにログします。

例：

```
zac cic c:\import_source_directory -l:"C:\Program Files\Novell\ZENworks\logs\cic.log"
```

#### **wake-cdp (cdp) [replicate | cleanup]**

コンテンツ配布ポイントワーカースレッドを起動します。次のいずれかのオプションを使用できます。

replicate - 新しいコンテンツまたは変更されたコンテンツがあれば、コンテンツ配布ポイントの親 ZENworks サーバからダウンロードします。

cleanup - コンテンツの配布ポイントに保存されないようにするすべてのコンテンツを削除します。

例：

```
zac cdp
```

```
zac cdp replicate
```

このコマンドは、エージェントがサテライトとして昇格されている場合のみ、適用できます。

## データベースコマンド

- ◆ [155 ページの「statussender RollUp」](#)

### statussender RollUp

正常にステータスがロールアップされた後に更新されたステータスを、MDStatus データベースにロールアップします。デフォルトでは、ステータス送信側が 5 分ごとにステータスを自動的にロールアップします。

MDStatus データベースを使用して、管理対象デバイスから ZENworks サーバにバンドルとポリシーステータスをロールアップします。

## イメージングコマンド

- ◆ [155 ページの「file-system-guid \(fsg\) \[-d\] \[-r\]」](#)

### file-system-guid (fsg)[-d] [-r]

イメージ取得の準備のために、ファイルシステム内で、ワークステーション GUID の表示、削除、または復元を行います。

次に例を示します。

GUID 値を表示する場合：

```
zac fsg
```

ファイルシステムから GUID を削除し、conninfo.dat も削除する場合：

```
zac file-system-guid -d
```

ファイルシステムに GUID を復元する場合：

```
zac file-system-guid -r
```

GUID 値を表示する場合：

```
zac fsg
```

## インベントリコマンド

- ◆ [155 ページの「inventory \[scannow | cdf | -f scannow\]」](#)

### inventory [scannow | cdf | -f scannow]

インベントリスキャンを実行します。またはコレクションデータフォームを開きます。

例：インベントリスキャンを実行する

```
zac inv scannow
```

例：コレクションデータフォームを開く

```
zac inv cdf
```

例：フルスキャンを実行する

```
zac inv -f scannow
```

## 場所コマンド

### config-location (cl)

設定場所を表示します。設定場所によって、認証、設定、コンテンツ、およびコレクションのためにデバイスが接続する ZENworks サーバが指定されます。

例：

```
zac config-location
```

```
zac cl
```

## ログコマンド

- ◆ 156 ページの「`logger (log) [resetlog | level [MANAGED|ERROR|WARN|INFO|DEBUG] | managedlevel]`」

### logger (log) [resetlog | level [MANAGED|ERROR|WARN|INFO|DEBUG] | managedlevel]

ZENworks Agent のロガーの設定を変更または表示します。

次のオプションを使用できます。

resetlog - ログをリセットします。

level - レベルを指定せずにこのオプションを使用する場合、現在の管理ログのレベルを表示します。レベルを指定した場合には、ログレベルを指定したレベルに変更します。

managedlevel - ゾーンのグローバルログレベルを表示します。

ログファイルをリセットする例：

```
zac logger resetlog
```

現在のログレベルを表示する例：

```
zac logger level
```

ログレベルを DEBUG 以上に設定する例：

```
zac logger level DEBUG
```

## OS ターゲットコマンド

### ostarget (os)

ワークステーション OS または指定されたバージョン文字列に関連付けられている ostarget レコードを表示します。

例：

ワークステーションのバージョン文字列と対応する ostarget 情報を表示するには：

```
zac ostarget
```

特定のバージョン文字列の対応する ostarget 情報を表示するには：

```
zac ostarget "Windows XP Professional Service Pack 2 (ビルド 2600)"
```

## パッチ管理コマンド

- ◆ 157 ページの「`patch-scan (ps)`」
- ◆ 157 ページの「`patch-scan --quick (ps --quick)`」
- ◆ 157 ページの「`patch-apply-policy (pap)`」
- ◆ 157 ページの「`patch-quarantine-release (pqr)`」

### patch-scan (ps)

デバイスの現在のパッチ署名 (DAU) ファイルを使用して、パッチが適用されていないデバイスをスキャンします。その後、結果がサーバにアップロードされます。

パッチスキャンの実行例：

```
zac ps
```

### patch-scan --quick (ps --quick)

サーバに最新スキャン結果をアップロードします。新たな検出スキャンは実行されません。

クイックパッチスキャンの実行例：

```
zac ps --quick
```

### patch-apply-policy (pap)

すべてのパッチポリシーの最新バージョンを使ってデバイスを更新します。

パッチ適用ポリシーの実行例：

```
zac pap
```

### patch-quarantine-release (pqr)

コマンドが検疫から実行されるデバイスで検疫済みパッチをリリースし、パッチポリシースケジュールまたは修復スケジュールのいずれかから、一度限りのインストールを試行できるようにします。

パッチ検疫リリースの実行例：

```
zac pqr
```

## ポリシーコマンド

- ◆ 157 ページの「`policy-list (pl)`」
- ◆ 158 ページの「`policy-refresh (pr)`」

### policy-list (pl)

現在デバイスに強制的に適用されているポリシー (有効ポリシー) を一覧表示します。すべてのポリシー (有効ポリシーと有効でないポリシー) を一覧表示するには、`--all` オプションを使用します。

例：

```
zac pl
```

```
zac pl --all
```

## policy-refresh (pr)

デバイスとユーザに割り当てられているすべてのポリシーを適用します。

例：

```
zac pr
```

## 登録コマンド

- 158 ページの「add-reg-key (ark) <登録キー>」
- 158 ページの「register (reg) [-g] [-k <キー>] [-u <ユーザ名> -p <パスワード>] <ZENworks サーバアドレス : ポート>」
- 159 ページの「reregister (rereg) [-u <ユーザ名> -p <パスワード>] <新規 GUID>」
- 159 ページの「unregister (unr) [-f] [-s] [-a] [-u <ユーザ名> -p <パスワード>]」
- 160 ページの「reestablish-trust (retr) [-u <ユーザ名> -p <パスワード>]」

### add-reg-key (ark) <登録キー>

指定したキーを使用してデバイスを登録します。キーによる登録は追加的です。デバイスが以前キーで登録されていて、そのデバイスを新しいキーで登録する場合、デバイスは、両方のキーに関連付けられているすべてのグループの割り当てを受け取ります。

例：

```
zac ark キー 12
```

### register (reg) [-g] [-k <キー>] [-u <ユーザ名> -p <パスワード>] <ZENworks サーバアドレス : ポート>

デバイスを管理ゾーンに登録します。

このコマンドを実行するには、デバイスの登録を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

次のオプションを使用できます。

g - 同じ GUID を持つ複数のデバイスがある場合、新しい GUID とパスワードでデバイスに新しいデバイスオブジェクトを作成できます。このスイッチを使用してデバイスを登録すると、元のデバイスオブジェクトに割り当てられたすべての関連付け (ポリシーとバンドル) が削除されます。このオプションを使用して、プライマリサーバまたはサテライトデバイスに新しい GUID を作成することはできません。ローカルユーザがこのオプションを使用するにはローカル管理者の権限が必要です。

k - 指定されたレジストリキーを使用してデバイスを登録できます。

p - 管理ゾーン管理者のパスワードを指定できます。

u - 管理ゾーン管理者のユーザ名を指定できます。

例：

```
zac reg -k キー 1 https://123.456.78.90
```

```
zac reg -k key1 -u administrator -p novell https://zenserver.novell.com:8080
```

ポート番号は、ZENworks サーバが標準 HTTP ポート (80) を使用していない場合にのみ必要です。ユーザ名とパスワードを指定しなかった場合には、入力するように求められます。

#### **reregister (rereg)[-u <ユーザ名> -p <パスワード>] <新規 GUID>**

デバイスを現在のゾーンに登録し、既存のデバイスオブジェクトの GUID を割り当てます。現在関連付けられているデバイスオブジェクトは削除されます。

このコマンドを実行するには、デバイスの再登録を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

たとえば、ハードドライブの交換後にデバイスのイメージを作成した場合、デバイスには新しい GUID が与えられる可能性があります。しかし、reregister コマンドを使用すれば、ハードドライブを交換する前にデバイスが持っていた GUID を割り当てることができます。

例：

再登録してユーザ名とパスワードを指定する

```
zac reregister -u myuser -p mypassword eaa6a76814d650439c648d597280d5d4
```

再登録して、ユーザ名とパスワードを入力するようプロンプトを表示する

```
zac reregister eaa6a76814d650439c648d597280d5d4
```

---

**注：**対応するデバイスオブジェクトがサーバ上にすでに存在し、そのデバイスオブジェクトで更新が発生した場合は、-g と -k オプションは適用されません。

---

#### **unregister (unr) [-f] [-s] [-a] [-u <ユーザ名> -p <パスワード>]**

管理ゾーンからデバイスの登録を削除します。

このコマンドを実行するには、デバイスの登録解除を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

例：

サーバに接続できないときに、強制的にローカルでデバイスを登録解除する場合：

```
zac unr -f -u myuser -p mypassword
```

ローカルで登録解除して、ユーザ名とパスワードのプロンプトを抑止する場合：

```
zac unr -s
```

非同期で登録解除するには、-a オプションを使用します。このオプションを使用すると、サーバによってデバイスが非同期で削除されます。

-a、-f、-u、および -p パラメータはオプションです。-u および -p パラメータを指定しないと、ユーザ名とパスワードを入力するよう求められます。-f パラメータは ZENworks データベースを無視し、強制的にローカルでデバイスを登録解除します。このオプションは、デバイスオブジェクトが ZENworks データベースからすでに削除済みか、またはデバイスがデータベースに接続できない場合にのみ必要です。-a オプションを指定した場合、ZENworks サーバによって登録解除コールが速やかに返されますが、データベースからのデバイスオブジェクトの削除は、後で非同期で実行されます。デバイス削除が完了していないのにデバイスの登録を再度試行した場合、

ZENworks サーバがエラーを表示します。データベース内のデバイスに大量のデータが関連付けられている場合、デバイスの削除に長時間かかることがあります。サーバ上での実際のデバイス削除が長時間かかるため、エージェントの登録解除コマンドがタイムアウトする場合は、-a オプションを使用してください。

---

**注:** UNR コマンドを使用すると、データベースが過度に使用される可能性があります。これは、次のいずれかの理由による場合があります。

- ◆ サーバ上で UNR コマンドが実行されている。
  - ◆ ゾーンに多数の管理対象デバイスが含まれている。
  - ◆ 管理対象デバイスに膨大な量の履歴がある。
  - ◆ パッチ管理が有効である。
- 

**reestablish-trust (retr) [-u <ユーザ名> -p <パスワード>]**

現在の管理ゾーンとの信頼を再確立します。使用するユーザ名とパスワードはゾーン管理者のものであることが必要です。

例:

```
zac retr -u myuser -p mypassword
```

-u および -p パラメータはオプションです。-u および -p パラメータを指定しないと、ユーザ名とパスワードを入力するよう求められます。

## リモート管理コマンド

**request-remote-session, rrs**

Z アイコンがない場合でも、管理対象デバイスからのリモート管理セッションを要求します。このコマンドは、11.3.1 以降のバージョンの管理対象デバイスで使用できます。

例:

```
zac request-remote-session
```

```
zac rrs
```

## ステータスコマンド

- ◆ 161 ページの「[cache-clear \(cc\)](#)」
- ◆ 161 ページの「[dump-prop-pages \(dpp\) <ターゲットディレクトリ>](#)」
- ◆ 161 ページの「[get-settings \(gs\) <キー>](#)」
- ◆ 162 ページの「[refresh \(ref\) \[general | partial bundle <バンドル表示名> \[bypasscache\]\]](#)」
- ◆ 162 ページの「[set-proxy \(sp\) \[オプション\] <IP アドレス / ホスト名 : ポート>](#)」
- ◆ 162 ページの「[winproxy-refresh \(wpr\)](#)」
- ◆ 162 ページの「[zenhttp-status \(zhs\)](#)」
- ◆ 162 ページの「[info-collect \(zeninfo\) \[<ターゲットファイル>\] \[-q\]](#)」



- ◆ 163 ページの「zone-config (zc) [-1]」
- ◆ 163 ページの「statussender (sts)」

#### cache-clear (cc)

デバイスの ZENworks キャッシュをクリアします。これにより、キャッシュデータベースのすべてのエントリと、これらのエントリに関連付けられたキャッシュファイルが削除されます。

例：

```
zac cc
```

---

**注：**ZENworks 管理者が ZENworks Agent のセルフディフェンス機能を有効にしている場合、zac cc コマンドを実行する前に、上書きパスワードを入力する必要があります。上書きパスワードを入力しないと、次のメッセージが表示されます。

You do not have permission to clear the cache. Please contact your ZENworks administrator.

ZENworks 管理者から上書きパスワードを入手してください。上書きパスワードが設定されていない場合、コマンドを使用するには、ZENworks 管理者が上書きパスワードを設定する必要があります。パスワードを受け取ったら、次の操作を実行します。

1. システムトレイの ZENworks アイコン ( [Z] アイコン ) をダブルクリックし、[エージェント]([ステータス]の下にあります) をクリックし、[エージェントセキュリティ設定]セクションの[ポリシーの上書き]リンクをクリックして、[バージョン情報] ボックスを表示します。
  2. [ポリシーの上書き] をクリックし、上書きパスワードを入力して、[上書き] をクリックします。
  3. コマンドラインプロンプトに移動し、zac cc コマンドを実行します。
  4. キャッシュが正常にクリアされたら、[バージョン情報] ボックスに戻り、[ポリシーのロード] をクリックして、パスワードの上書きを無効にします。
- 

#### dump-prop-pages (dpp) < ターゲットディレクトリ >

ZENworks Icon のプロパティページに表示される HTML ページを、指定したターゲットディレクトリのファイルに出力します。

例：

```
zac dpp c:\temp
```

#### get-settings (gs) < キー >

設定は更新のたびに設定モジュールからローカルキャッシュにダウンロードされます。これは指定されたキーに関連付けられている有効な設定を返します。

例：

```
zac gs キー 1
```

すべての有効な ZENworks 設定のキーは、%ZENWORKS\_HOME%\cache\zmd\settings ディレクトリに保存されます。

リモート管理設定を一覧表示する例：

```
zac gs RemoteManagement
```

#### **refresh (ref)[general | partial bundle <バンドル表示名> [bypasscache]**

一般更新を呼び出してすべてのバンドル、ポリシー、登録、環境設定を更新します。  
部分更新を呼び出して、すべてのポリシー、登録、環境設定を更新します。

bypasscache を使用して、更新中にサーバキャッシュからデータが使用されないようにします。このオプションはテストまたはトラブルシューティングに有効です。

例：

```
zac ref general bypasscache
```

```
zac ref partial bypasscache
```

#### **set-proxy (sp) [ オプション ] <IP アドレス / ホスト名 : ポート>**

ZENworks サーバに直接接続する代わりに、接続するプロキシを指定します。

オプションは次のとおりです。

--default - 管理ゾーンのプロキシ設定によって上書きできるプロキシを設定します。

--clear - 現在のプロキシをクリアし、管理ゾーンのプロキシ設定を使用します。

例：

**IPv4:**

```
zac sp 123.456.78.90:2349 administrator novell
```

```
zac sp /default 123.456.78.90:2349
```

```
zac sp /clear
```

ユーザ名とパスワードを指定していない場合は、それらの入力を求められます。

#### **winproxy-refresh (wpr)**

このデバイスに割り当てられたプロキシ作業について、管理ゾーンに対してクエリを実行します。

例：

```
zac wpr
```

#### **zenhttp-status(zhs)**

登録ハンドラのポートとタグを一覧表示します。

例：

```
zac zhs
```

このコマンドは、エージェントがサテライトとして昇格される場合にのみ適用されません。

#### **info-collect (zeninfo) [<ターゲットファイル>] [-q]**

キャッシュデータ、環境設定データ、デバッグログ、製品インストール情報、更新時間、ステータスイベント、および基本システム情報を含む ZENworks サポート情報を収集します。情報は zip ファイルにパッケージ化され、指定した場所に保管されます。場所を指定しないと、Windows の場合は  $\${TEMP}\zeninfo-{\DateTime}.zip$ 、Linux の場合は

`${TMPDIR}\zeninfo-${DateTime}.zip` が使用されます。管理対象デバイスで問題が発生している場合は、Micro Focus Support が、問題解決に役立てるためにこのコマンドを実行し、結果の zip ファイルを Micro Focus に送信するよう依頼することがあります。

zeninfo プロセスをバックグラウンドで実行するには、次のコマンドを実行します。これらのコマンドは Linux でのみ適用できます。

```
zac zeninfo /tmp/zeninfo/ & echo $!> /tmp/zeninfo/zeninfo.pid
```

zeninfo プロセスを停止するには、次のコマンドを実行します。

```
kill `cat /tmp/zeninfo/zacinfo.pid`
```

次のオプションを使用できます。

q - 収集後にエクスプローラの起動をスキップします。

zeninfo コマンドは、ローカル管理者が実行できます。ローカル管理者以外がこのコマンドを実行すると、管理者の資格情報を入力するようにシステムから求められます。または、AllowZenInfoWithoutAdminPwd 文字列の値を True に設定することもできます。このように設定すると、任意のユーザが zeninfo コマンドを実行可能になります。AllowZenInfoWithoutAdminPwd 文字列の値を設定するには、次の手順に従います。

1. レジストリエディタを開きます。
2. HKLM\Software\Novell\ZCM\ に移動します。
3. AllowZenInfoWithoutAdminPwd 文字列の値を True に設定します。

---

**警告** : AllowZenInfoWithoutAdminPwd 文字列の値を True に設定すると、ローカル管理者以外のユーザも、取り扱いに細心の注意が必要な ZENworks Configuration Management 設定および環境設定情報を見ることが可能になります。

---

### zone-config (zc) [-l]

デバイスが環境設定情報を求めてアクセスしている ZENworks サーバの情報を表示します (設定サーバ)。または、設定サーバについての情報を一覧表示します。

例 :

```
zac zc
```

```
zac zc -l
```

### statussender (sts)

このコマンドは、プライマリサーバにステータス情報をロールアップします。前回ステータスがロールアップされてから更新された情報をロールアップするか、完全なステータス情報をロールアップすることができます。

例 :

最後に成功したステータスロールアップ以降に更新されたステータス情報をロールアップするには :

```
zac sts rollup
```

同じスレッド上のステータス情報をロールアップするには

```
zac sts rollup syn
```

完全なステータス情報をロールアップするには：

```
zac sts rollup full
```

---

**注：**ZENworks 2020 以降のリリースから、zac bsr コマンドが廃止されました。zac sts コマンドを実行して、バンドルステータス情報をロールアップします。

---

## システムアップデートコマンド

```
zac zeus-refresh / zeus-ref
```

システム更新がデバイスに割り当てられている場合、そのシステム更新を取得します。

## ZENworks Endpoint Security Management のコマンド

```
zac zesm-refresh / zesm-ref
```

管理対象デバイス上の ZENworks Endpoint Security Management キャッシュをリセットします。

# zac for Linux(1)

## 名前

zac - Linux 管理対象デバイスにインストールされて実行される Micro Focus ZENworks Agent のコマンドライン管理インタフェースです。

## 構文

*zac command options*

## 説明

zac ユーティリティは、ソフトウェアバンドルのインストールと削除、ポリシーの適用、およびデバイスの登録と登録解除など、ZENworks 管理デバイス上でコマンドライン管理機能を実行します。

## 使用方法

このセクションでは、一般的なコマンド形式と規則について説明しています。

ほとんどのコマンドは、長い形式と短い形式があります。

ロングフォーム : add-reg-key

ショートフォーム : ark

両方の形式が使用できる場合、コマンドは以下のように表示されます。

add-reg-key (ark) オプション

コマンドを使用する場合は、長い形式または短い形式のいずれかのみを入力します。

zac add-reg-key オプション

zac ark オプション

オプションにスペースが含まれている場合は、引用符で囲みます。

zac ark " オプション1"

### 必須の引数とオプションの引数

括弧 ( ) で囲まれた引数は必須で、角括弧 [ ] で囲まれた引数はオプションです。

### 省略記号 (...)

省略記号は、コマンドが最後の引数タイプのエントリを複数受け入れることを示します。引数は、コマンドに応じてスペースまたはカンマで区切る必要があります。例を次に示します。

zac <コマンド> [オプション] (サーバ) (バンドル) [...]

この省略記号は、<コマンド> が複数のバンドルを受け入れることができることを示しています。コマンドレベルのヘルプでは、複数エントリの引数ヘルプは、より明確に [(バンドル) (バンドル) ... (バンドル)] として指定されます。

## ヘルプコマンド

- 166 ページの 「/? または --help」
- 166 ページの 「COMMAND /?」
- 166 ページの 「/g または --group」

### **/? または --help**

システムのヘルプを表示します。

### **COMMAND /?**

指定したコマンドの説明を表示します。

### **/g または --group**

指定したグループのコマンドを表示します。

## コマンドのリスト

- 167 ページの 「認証サテライトサーバコマンド」
- 167 ページの 「バンドルコマンド」
- 172 ページの 「証明書コマンド」
- 172 ページの 「コレクションロールアップコマンド」
- 173 ページの 「コンテンツ配布コマンド」
- 175 ページの 「コアコマンド」
- 176 ページの 「システムアップデートコマンド」
- 176 ページの 「外部サービスコマンド」
- 178 ページの 「イメージングコマンド」
- 178 ページの 「インベントリコマンド」
- 178 ページの 「インベントリ専用コマンド」
- 179 ページの 「ローカルユーザ管理コマンド」
- 181 ページの 「場所コマンド」
- 181 ページの 「ログコマンド」
- 181 ページの 「パッケージ管理コマンド」
- 187 ページの 「パッチ管理コマンド」
- 188 ページの 「ポリシーコマンド」
- 189 ページの 「登録コマンド」

- 191 ページの「サテライトの役割コマンド」
- 192 ページの「ステータスコマンド」

## 認証サテライトサーバコマンド

- 167 ページの「`authentication server reconfigure (asr) [-t all|config|jetty|casa] [-u ユーザ名] [-p パスワード]`」
- 167 ページの「`import-authentication-cert(iac) [-pk <秘密鍵.der>] [-c <署名サーバ証明書.der>] [-ca <署名認証局パブリック証明書.der>] [-ks <キーストア.jks>] [-ksp <キーストアパスフレーズ>] [-a <署名証明書エイリアス>] [-ks <署名証明書パスフレーズ>] [-u ユーザ名] [-p パスワード] [-rc]`」

**authentication server reconfigure (asr) [-t all|config|jetty|casa] [-u ユーザ名] [-p パスワード]**

有効化された認証サテライトを再設定します。

例：

設定ファイルをサーバから取得する場合：

```
zac asr -t config
```

CASA 署名証明書を再設定する場合：

```
zac asr -t casa
```

Jetty Web サーバを再設定する場合：

```
zac asr -t jetty -u Administrator -p password
```

サテライト全体を再設定する場合：

```
zac asr -t all -u Administrator -p パスワード
```

ユーザ名とパスワードが必要な場合、コマンドラインで指定しないと入力を求められます。

**import-authentication-cert(iac)[-pk <秘密鍵.der>] [-c <署名サーバ証明書.der>] [-ca <署名認証局パブリック証明書.der>] [-ks <キーストア.jks>] [-ksp <キーストアパスフレーズ>] [-a <署名証明書エイリアス>] [-ks <署名証明書パスフレーズ>] [-u ユーザ名] [-p パスワード] [-rc]**

外部署名証明書を使用して認証サテライトデバイスを設定します。

rc - 管理者に対して再設定が要求されないように、認証サテライトサーバの再設定を確認します。

各サテライトには、独自のサーバ証明書と秘密鍵が必要です。外部証明書の生成の詳細については、『ZENworks サーバインストールガイド』を参照してください。

## バンドルコマンド

- 168 ページの「`bundle-install (bin) [オプション] [バンドル名 | GUID | パターン]`」
- 168 ページの「`bundle-launch (bln) [bundle display name] [-noSelfHeal]`」

- ◆ 169 ページの 「bundle-list (bl) [ オプション ]」
- ◆ 169 ページの 「bundle-lock-add (bla) [ オプション ] [ バンドル名 ] [...]
- ◆ 170 ページの 「bundle-lock-list (bl1)」
- ◆ 170 ページの 「bundle-lock-remove (blr) [ オプション ] [ バンドルインデックス ] [...]
- ◆ 170 ページの 「bundle-refresh (br) [bundle display name | guid]
- ◆ 170 ページの 「bundle-search (bse) [ オプション ] pattern]
- ◆ 171 ページの 「bundle-uninstall (bu) [bundle display name | guid]
- ◆ 171 ページの 「bundle-verify (bv) [bundle display name]

### **bundle-install (bin) [ オプション ] [ バンドル名 | GUID | パターン ]**

指定された正規表現または文字列、あるいはバンドル GUID に一致する表示名に基づいて指定されたバンドルをインストールします。bundle-list コマンドを使用して、使用できるバンドルおよびその表示名のリストを取得します。

次の正規表現がサポートされています。

正規表現	機能
<code>^abc</code>	「abc」から始まる文字列に一致します。
<code>[abc]</code>	「a」、「b」、または「c」を含む文字列に一致します。
<code>a*c</code>	0 回以上の「a」の後に「c」、「ac」、「aac」、「aaac」、または「c」が続く場合に一致します。

次のオプションを受け入れます。

-N, --dryrun - 要求されたアクションをテストして表示しますが、実際には実行しません。1 度だけバンドルバージョンでドライ実行を行うことができます。このオプションは、バンドルの [RPM のインストール] アクションにのみ適用できます。

バンドル名に基づいたバンドルのインストール例：

```
zac bin bundle1
```

### **bundle-launch (bln) [bundle display name] [-noSelfHeal]**

指定したバンドルを起動します。[起動] アクションセットのアクションが実行されません。

インストールされていないバンドルを起動しようとする場合、バンドルが最初にインストールされてから起動されます。

bundle-list コマンドを使用して、使用できるバンドルおよびその表示名のリストを取得します。

表示名に基づいたバンドルの起動例：

```
zac bln bundle1
```



表示名に基づいてバンドルを起動し、起動アクションが失敗した場合の自己回復設定をオフ ( デフォルトでは自己回復がオンの状態 ) にした例 :

```
zac bln bundle1 -noSelfHeal
```

### bundle-list (bl) [ オプション ]

デバイスおよびログインユーザに割り当てられているバンドルのリストを表示します。このコマンドの出力は列に表示され、管理対象デバイスに割り当てられている、または利用可能なすべてのバンドルの詳細が表示されます。[表示名]、[バージョン]、および [バンドルタイプ] 列は、バンドル表示名、バンドルバージョン番号またはバンドルがサンドボックスバージョンの場合はサンドボックスのバージョン番号、およびバンドルタイプをそれぞれ示します。[ステータス] 列には次の状態が表示されません。

使用可能 : バンドルがインストールされている場合はこの状態を表示します。

インストールされていません : 特定のバンドルバージョンがインストールされていない場合にこの状態を表示します。

ダウンロードされました : コンテンツがダウンロードされるが [インストール] アクションセットのアクションが実行されていない場合にこの状態を表示します。

このコマンドは次のオプションを受け入れます。

-a, --all - フィルタされたバンドルと、フィルタされていないバンドルの両方を一覧表示します。フィルタされたバンドルはチェーンバンドルです。デフォルトでは、フィルタされていないバンドルが表示されます。

-g, --show-guid - バンドル GUID を表示します。

すべてのバンドルを一覧表示する例 :

```
zac bl -a
```

### bundle-lock-add (bla) [ オプション ] [ バンドル名 ] [...]

アップグレードまたは削除されないように、指定された正規表現または文字列に一致する指定されたバンドルをロックします。

次の正規表現がサポートされています。

正規表現	機能
^abc	「abc」から始まる文字列に一致します。
[abc]	「a」、「b」、または「c」を含む文字列に一致します。
a*c	0 回以上の「a」の後に「c」、「ac」、「aac」、「aaac」、または「c」が続く場合に一致します。

次のオプションを受け入れます。

-li, --lock-install - バンドルをロックし、ロックに指定されていないバンドルがインストールされないようにします。

---

**注:** バンドルロックは [インストール] アクションセットにのみ適用できます。

---

-t, --type - ロックするバンドルタイプを指定します。指定できるタイプは linux と patch です。タイプを指定しないと、bundle タイプが選択されます。

**bundle-lock-list (bll)**

バンドルロックを一覧表示します。バンドルロックは ID 番号でソートされます。

例:

```
zac bll
```

**bundle-lock-remove (blr) [ オプション ] [ バンドルインデックス ] [...]**

番号で指定したバンドルロックを削除します。bundle-lock-list (bll) コマンドを使用して、各ロックの番号を見つけられます。

次のオプションを受け入れます。

-n - 名前で指定したバンドルロックを削除します。名前にパターンが含まれていないことを確認します。

--remove-all - すべてのバンドルロックを削除します。

バンドルロック abc を削除する例:

```
zac blr -n abc
```

すべてのバンドルロックを削除する例:

```
zac blr --remove-all
```

**bundle-refresh (br) [bundle display name | guid]**

指定されたバンドルの情報を更新します。

例:

```
zac br バンドル 1
```

**bundle-search (bse) [ オプション ] pattern**

指定された正規表現または文字列に一致するバンドルを検索します。

次の正規表現がサポートされています。

正規表現	機能
a+bc	「a」の後に「bc」が1回以上続く場合に一致します。 たとえば、1つ以上の「a」の後に「abc」または「aabc」と続く場合に一致します。
[a-z]\*	空の文字列、および名前にアルファベット文字だけを含む文字列に一致します。 たとえば、空の文字列 ""、または acb や xyz などの名前にアルファベット文字だけを含む文字列に一致します。
a*bc	0回以上の「a」の繰り返しの後に「bc」が続く場合に一致します。 たとえば、0回以上の「a」の後に「abc」、「aabc」、「aaabc」、または「bc」が続く場合に一致します。

このコマンドの出力は `zac bl` コマンドの出力と類似しています。

次のオプションを受け入れます。

`-i, --installed-only` - インストールされたバンドルのリストのみを検索します。

#### **bundle-uninstall (bu) [bundle display name | guid]**

指定したバンドルをアンインストールします。[元に戻す] アクションをサポートするアクションは、バンドルのアンインストール中に実行されます。バンドルのアンインストールが完了した後で、バンドルの状態が [インストールされていません] と表示されます。

`bundle-list` コマンドを使用して、インストールされたバンドルおよびその表示名のリストを取得します。

例：

```
zac bu バンドル 1
```

#### **bundle-verify (bv) [bundle display name]**

バンドルを再インストールし、[検証] アクションセットを実行して、削除されたファイルや破損したファイルがないことを確認します。`bundle-list` コマンドを使用して、インストールされたバンドルおよびその表示名のリストを取得します。

例：

```
zac bv バンドル 1
```

## 証明書コマンド

- ◆ 172 ページの「`cert-info (ci) [CA 証明書のファイルパス] [-u <ユーザ名> -p <パスワード>]`」

### `cert-info (ci) [CA 証明書のファイルパス] [-u <ユーザ名> -p <パスワード>]`

既知の ZENworks サーバごとに公開鍵証明書情報を一覧表示するか、デバイスの信頼するストアに信頼するルート証明書を追加します。ファイルは、ASN.1 DER フォーマットにすることも、`-----BEGIN CERTIFICATE-----` と `-----END CERTIFICATE--` で区切って Base64 エンコードすることもできます。

例：

既知の ZENworks サーバごとに証明書を一覧表示するには、次のコマンドを実行します。

```
zac ci
```

デバイスの信頼するストアに信頼するルート証明書を追加するには、次のコマンドを実行します。

```
zac ci c:\certs\mytrustcacert.der -u myuser -p mypassword
```

## コレクションロールアップコマンド

- ◆ 172 ページの「`collection-clients (crc)`」
- ◆ 172 ページの「`collection-point (cp)`」
- ◆ 172 ページの「`collection-queued (crq)`」
- ◆ 173 ページの「`collection-state (crs)`」
- ◆ 173 ページの「`collection-upload-orphans (cuo)`」
- ◆ 173 ページの「`collection-wake (crw)`」

### `collection-clients (crc)`

このコレクションにファイルをアップロードしたデバイスを表示します。

例：

```
zac crc
```

### `collection-point (cp)`

コレクションの役割の状態と設定を表示します。

例：

```
collection-point [wake]
```

`wake` - コレクションを実行するモジュール ( インベントリ、MD ステータス、メッセージ送信側 ) を起動します。

### `collection-queued (crq)`

収集したファイルのうち、ロールアップを行っていないファイルを表示します。

例：

```
zac crq
```

#### collection-state (crs)

サテライトコレクション役割の現在の状態を表示します。

例：

```
zac crs
```

#### collection-upload-orphans (cuo)

サテライトデバイス上の孤立したファイルを検索して、親コレクションサーバにロールアップするか、またはロールアップ済みの場合は削除します。

このコマンドは /var/opt/novell/zenworks/work/collection (collectionStatsDB フォルダ以外) の下のフォルダにファイルリストを構築し、コレクション統計データベースの各エントリの元のアップロード情報を検索しようとします。

データベースにファイルのエントリがある場合は、ファイルがロールアップされていないことを示し、ファイルをロールアップします。エントリにファイルがロールアップ済みであることが示されている場合は、サテライトデバイスのファイルを削除します。データベースにファイルのエントリがない場合は、ファイルをロールアップします。このコマンドは、アップロードまたは削除されなかったファイルのリストも表示します。

このコマンドを実行する前に、zac crw コマンドを実行して保留中のファイルを親サーバに送信する必要があります。

例：

```
zac cuo
```

#### collection-wake (crw)

収集したファイルを親サーバに送信します。

例：

```
zac crw
```

## コンテンツ配布コマンド

- 173 ページの「`cdp-checksum (cchk) [-l:< ログへのパス >]`」
- 174 ページの「`cdp-verify-content (cvc) [-c] [-l:< ログへのパス >]`」
- 174 ページの「`cdp-import-content (cic) <content path> [<path to log>]`」
- 174 ページの「`distribution-info (cdi) [オプション]`」
- 174 ページの「`distribution-state (cdis)`」
- 175 ページの「`distribution-wake (cdp) [replicate | cleanup]`」

#### cdp-checksum (cchk) [-l:< ログへのパス >]

各ファイルのチェックサムを計算することによって、サテライトのコンテンツを検証します。

オプションのログファイルには、チェックサム比較結果の詳細が記録されます。

例：

```
zac cchk -l:"/var/opt/novell/zenworks/logs/cchk.log"
```

#### **cdp-verify-content (cvc) [-c] [-l:< ログへのパス >]**

この CDP でのコンテンツ ID とその同期状態のリストを、プライマリサーバが想定しているものと比較します。

次のオプションを使用できます。

c - ローカルコンテンツのチェックサムを計算します。

例：

```
zac cvc -l:"/var/opt/novell/zenworks/logs/cvc.log"
```

#### **cdp-import-content (cic) <content path> [<path to log>]**

content-path で指定されたディレクトリから失われたコンテンツをインポートし、log-path で指定されたファイルにログします。

例：

```
zac cic /usr/content_Dir /var/opt/novell/zenworks/logs/cic.log
```

#### **distribution-info (cdi) [ オプション ]**

コンテンツの役割に関する統計情報を表示します。次のオプションを使用できます。

all - 「\*」の印が付いているすべてのオプションを実行します。

ss - 同期ステータスを表示します\*。

rs - コンテンツリポジトリのサイズを取得します\*。

da - このデバイスからデータを取得したデバイスを一覧表示します\*。

ne - コンテンツ配布エラー数を取得します\*。

ds - このデバイスが提供したデータサイズを取得します\*。

fs - 提供したファイル数を取得します\*。

ah [n] - アクセス履歴の n ( デフォルトは all ) 項目を表示します。

ch - 履歴を消去します。

eh <file> - アクセス履歴を <file> にエクスポートします。ここにはパスを含む必要があり、ファイル自体が存在してはなりません。

例：

```
zac cdi
```

```
zac cdi rs
```

#### **distribution-state (cnds)**

サテライトコンテンツの役割のステータスを表示します。役割が有効 / 無効か、ステータス ( 開始中、アクティブ、停止中、停止済み )、親 URL、レプリケーションスケジュールを表示します。

例：

```
zac cnds
```

## distribution-wake (cdp) [replicate | cleanup]

コンテンツ配布ポイントワークスレッドを起動します。次のいずれかのオプションを使用できます。

replicate - 新しいコンテンツまたは変更されたコンテンツがあれば、コンテンツ配布ポイントの親 ZENworks サーバからダウンロードします。

cleanup - コンテンツの配布ポイントに保存されないようにするすべてのコンテンツを削除します。

例 :

```
zac cdp
```

```
zac cdp replicate
```

## コアコマンド

- 175 ページの「`get-settings (gs) <キー>`」
- 175 ページの「`get-pref (get) <初期設定名>`」
- 175 ページの「`set-pref (set) <初期設定名> <初期設定の値>`」
- 176 ページの「`set-proxy (sp) [オプション] <IPアドレス / ホスト名 : ポート>`」

### get-settings (gs) <キー>

設定は更新のたびに設定モジュールからローカルキャッシュにダウンロードされます。これは指定されたキーに関連付けられている有効な設定を返します。

例 :

```
zac gs キー 1
```

リモート管理設定を一覧表示する例 :

```
zac gs RemoteManagement
```

すべての有効な ZENworks 設定キーは、`/var/opt/novell/zenworks/zmd/cache/settings/` ディレクトリに保存されます。

### get-pref (get) <初期設定名>

現在設定されている初期設定を表示します。

設定されているすべての初期設定を表示する例 :

```
zac get-pref
```

abc で始まる名前のすべての初期設定を表示した例 :

```
zac get-pref abc
```

### set-pref (set) <初期設定名> <初期設定の値>

初期設定値を設定します。

初期設定値 abc を 123 に設定した例 :

```
zac set-pref abc 123
```

---

**注:** 管理対象デバイスにプロキシサーバを設定する場合は、管理対象デバイスに proxy-host および proxy-port の両方の初期設定が設定されていることを確認します。プロキシサーバが認証を必要とする場合は、管理対象デバイスにはさらに proxy-username および proxy-password の初期設定が必要です。

---

**set-proxy (sp) [ オプション ] <IP アドレス / ホスト名 : ポート >**

ZENworks サーバに直接接続する代わりに、接続するプロキシを指定します。

オプションは次のとおりです。

--default - 管理ゾーンのプロキシ設定によって上書きできるプロキシを設定します。

--clear - 現在のプロキシをクリアし、管理ゾーンのプロキシ設定を使用します。

例:

**IPv4:**

```
zac sp 123.456.78.90:2349 administrator novell
```

```
zac sp /default 123.456.78.90:2349
```

```
zac sp /clear
```

ユーザ名とパスワードを指定していない場合は、それらの入力を求められます。

## システムアップデートコマンド

**zac zeus-refresh / zeus-ref**

システム更新がデバイスに割り当てられている場合、そのシステム更新を取得します。

## 外部サービスコマンド

外部サービスを管理するコマンドです。

- ◆ 176 ページの「external-service-add (esa) --type (type) (url) (Service Name) [Properties]」
- ◆ 177 ページの「external-service-delete (esd) (index | name | url | --all)」
- ◆ 177 ページの「external-service-list (esl)」
- ◆ 178 ページの「external-service-refresh (esr) (index | name | url | --all)」
- ◆ 178 ページの「external-service-state (ess) (enable | disable) (index | name | url | --all)」
- ◆ 178 ページの「external-service-types (est)」

**external-service-add (esa) --type (type) (url) (Service Name) [Properties]**

YUM、ZYPP、または RPM を含むローカルディレクトリなどの外部パッケージレポジトリを追加します。タイプ固有のプロパティは必ず -D で開始します。

zac est コマンドを使用して追加できるサービスタイプを一覧表示できます。



次のタイプをサポートしています。

- ◆ YUM: YUM タイプのレポジトリを追加します。次のプロパティを受け入れます。

-Dusername=value -Dpassword=value - このプロパティはオプションで、認証された YUM レポジトリを追加しているときに使用されます。

-DsyncExternal=true - このプロパティはオプションで、ローカルパッケージ管理ツールと同期するために使用されます。

たとえば、YUM サービスタイプを追加するには、次のコマンドを実行します。

```
zac esa --type YUM -Dusername=username -Dpassword=passwd https://nu.novell.com/sles11/sles11-yum-repo
```

- ◆ ZYPP: ZYPP タイプのレポジトリを追加します。次のプロパティを受け入れます。

-DsyncExternal=true - このプロパティはオプションで、ローカルパッケージ管理ツールと同期するために使用されます。

たとえば、ZYPP サービスタイプを追加するには、次のコマンドを実行します。

```
zac esa --type ZYPP http://zypp.source.com/sles11/x86_64 sles11-zypp-repo
```

- ◆ Mount: Mount タイプのレポジトリを追加します。Mount タイプは RPM を含むローカルディレクトリからサービスを追加します。次のプロパティを受け入れます。

-Drecursive=(Value) - このプロパティはオプションです。有効な値は true または false です。true の場合、RPM は指定した絶対パスの下のすべてのディレクトリから再帰的に追加されます。

-DsyncExternal=true - このプロパティはオプションで、ローカルパッケージ管理ツールと同期するために使用されます。

たとえば、Mount サービスタイプを使用してすべてのディレクトリから再帰的にパッケージを追加するには、次のコマンドを実行します。

```
zac esa --type mount -Drecursive=true file:///mnt/sles11-cd/i586 sles11-rpms
```

---

**注** : ZYPP または Mount サービスは Red Hat デバイス上で YUM パッケージ管理ツールと同期しません。

---

#### **external-service-delete (esd) (index | name | url | --all)**

指定された外部サービスを削除します。

--all オプションはすべての外部サービスを削除します。

#### **external-service-list (esl)**

すべての外部サービスを一覧表示します。

[ソース] 列には、zac コマンドを使用して追加された ZENworks エージェントが表示されるか、また外部サービスポリシーから追加された ZENworks サーバが表示されます。

[ステータス] 列には、外部サービスがアクティブであるか、または停止しているかが表示されます。ステータスが「停止」と表示されている外部サービスを更新することはできません。

[状態] 列には、外部サービスが有効であるか、または無効であるかが表示されます。

### **external-service-refresh (esr) (index | name | url | --all)**

指定された外部サービスを更新します。

--all オプションはすべてのサービスを更新します。

### **external-service-state (ess) (enable | disable) (index | name | url | --all)**

指定された外部サービスを有効または無効にします。

--all オプションはすべてのサービスの状態を変更します。

### **external-service-types (est)**

使用できるサービスタイプを一覧表示します。

## イメージングコマンド

- ◆ [178 ページの「file-system-guid \(fsg\) \[-d\] \[-r\]」](#)

### **file-system-guid (fsg)[-d] [-r]**

イメージ取得の準備のために、ファイルシステム内で、ワークステーション GUID の削除と復元を行います。

ファイルシステムから GUID を削除する場合の例

```
zac file-system-guid -d
```

ファイルシステムに GUID を復元する場合の例

```
zac file-system-guid -r
```

GUID 値をプリントする場合の例

```
zac fsg
```

## インベントリコマンド

- ◆ [178 ページの「inventory \(inv\) \[scannow | cdf\]」](#)

### **inventory (inv) [scannow | cdf]**

インベントリスキャンを実行します。またはコレクションデータフォームを開きます。

例：インベントリスキャンを実行する

```
zac inv scannow
```

例：コレクションデータフォームを開く

```
zac inv cdf
```

## インベントリ専用コマンド

インベントリ専用エージェント (IOA) でのみ使用するコマンドです。

### **zac ioa cfg <サーバ IP>**

Linux IOA デバイスを、指定されたサーバ IP に設定します。

例：インベントリ専用スキャンを実行する

```
zac ioa cfg 10.70.13.124
```

#### **zac ioa version**

インベントリ専用エージェント (IOA) のバージョンを表示します。

#### **inventory (inv) [scannow]**

インベントリスキャンを実行します。

インベントリスキャンの実行例：

```
zac inv scannow
```

## ローカルユーザ管理コマンド

ローカルユーザとその権限を管理するコマンドです。

- ◆ 179 ページの 「`user-add (ua) (username)`」
- ◆ 179 ページの 「`user-delete (ud) (username)`」
- ◆ 179 ページの 「`user-list (ul)`」
- ◆ 180 ページの 「`user-edit (ue) (username) [+/-Privilege]`」
- ◆ 180 ページの 「`user-privilege-export (upe) (.xml filename)`」
- ◆ 180 ページの 「`user-privilege-import (upi) (.xml filename)`」

#### **user-add (ua) (username)**

ローカルユーザを ZENworks 特権ストアに追加します。

例：

```
zac ua username
```

#### **user-delete (ud) (username)**

ローカルユーザを ZENwork 特権ストアから削除します。

例：

```
zac ud username
```

#### **user-list (ul)**

次のものを一覧表示します。

- ZENworks 特権ストアに追加されるローカルユーザのユーザ名と特権。
- ZENworks 特権ストアに追加されないローカルユーザ ( デフォルトユーザ ) の特権。デフォルトユーザは \* ( アスタリスク ) で表されます。

例：

```
zac ul
```

### user-edit (ue) (username) [+/-Privilege]

ローカルユーザの特権を編集します。新しい特権を追加するには+(プラス)を使用し、既存の特権を削除するには-(マイナス)を使用します。

デフォルトで、新しく作成されたローカルユーザには表示特権があります。特権ストアに追加されないローカルユーザは、\*(アスタリスク)で表されるデフォルトユーザに割り当てられた特権を継承します。ローカルユーザに割り当てられた特権を変更し、ユーザに次の特権を割り当てることができます。

特権	機能
表示	バンドル、ポリシー、設定などに関する情報を表示または一覧表示します。
インストール	バンドル、ポリシー、パッケージなどをインストールまたはアンインストールします。
設定	設定コマンドを実行します。
スーパーユーザ	スーパーユーザはすべての特権を割り当てられるため、すべての zac コマンドを実行します。デフォルトで、ルートユーザはスーパーユーザ特権を割り当てられ、変更することはできません。

例：

ZENworks 特権ストアのローカルユーザにすべてのバンドルとポリシーの表示、インストール、およびアンインストールを許可する場合は、次のコマンドを実行して、これらのユーザに表示およびインストール特権を割り当てることができます。

```
zac ue username +Install
```

ZENworks 特権ストアに追加されないローカルユーザにすべてのバンドルとポリシーの表示、インストール、およびアンインストールを許可する場合は、次のコマンドを実行して、\*ユーザに表示およびインストール特権を割り当てることができます。

```
zac ue '*' +Install
```

### user-privilege-export (upe) (.xml filename)

指定された .xml ファイルにローカルユーザとその特権のリストをエクスポートして、他の管理対象デバイス上でこのリストを再使用できるようにします。

例：

```
zac upe /tmp/privileges.xml
```

リストをエクスポートした後で、user-privilege-import (upi) コマンドを実行して、他の管理対象デバイスにインポートできます。

### user-privilege-import (upi) (.xml filename)

指定された .xml ファイルから他の管理対象デバイスにローカルユーザとその権限のリストをインポートします。このリストは、user-privilege-export (upe) コマンドを実行して、.xml ファイルにエクスポートできます。

例：

```
zac upi /tmp/privileges.xml
```

## 場所コマンド

- ◆ 181 ページの「`config-location (cl)`」

### `config-location (cl)`

設定場所を表示します。設定場所によって、認証、設定、コンテンツ、およびコレクションのためにデバイスが接続する ZENworks サーバが指定されます。

例：

```
zac config-location
zac cl
```

## ログコマンド

- ◆ 181 ページの「`logger [resetlog | level [MANAGED|ERROR|WARN|INFO|DEBUG] | managedlevel]`」

### `logger [resetlog | level [MANAGED|ERROR|WARN|INFO|DEBUG] | managedlevel]`

ロガーの環境設定を変更します。

次のオプションを使用できます。

`resetlog` - ログをリセットします。

`level` - レベルなしで使用した場合、現在管理されているログインレベルが表示されます。レベルを指定した場合には、指定したレベルにログレベルを変更します。

`managedlevel` - ゾーンのグローバルログレベルを表示します。

ログファイルをリセットする例：

```
zac logger resetlog
```

現在のログレベルを表示する例：

```
zac logger level
```

ログレベルを DEBUG 以上に設定する例：

```
zac logger level DEBUG
```

## パッケージ管理コマンド

- ◆ 182 ページの「`checkpoint-add (cpa) (name) [date]`」
- ◆ 182 ページの「`checkpoints-list`」
- ◆ 182 ページの「`checkpoint-remove (cpr) (name) [...]`」
- ◆ 182 ページの「`dist-upgrade (dup) [オプション] [バンドル名 | サービス名] [...]`」
- ◆ 182 ページの「`info (info) [オプション] pattern`」
- ◆ 183 ページの「`install (in) [オプション] (package name) [...]`」
- ◆ 184 ページの「`list-updates (lu) [オプション] [バンドル名 | サービス名] [...]`」
- ◆ 184 ページの「`lock-add (la) (name or pattern) [version]`」

- ◆ 185 ページの 「lock-delete (ld) (lock number) [...]」
- ◆ 185 ページの 「lock-list (ll)」
- ◆ 185 ページの 「packagehistory (ph) [ オプション ] [...]」
- ◆ 185 ページの 「remove (rm) [ オプション ] (package name) [...]」
- ◆ 186 ページの 「rollback (ro) [ オプション ] [ 日付またはチェックポイント ]」
- ◆ 186 ページの 「search (se) [ オプション ] [ パターン ]」
- ◆ 187 ページの 「update (up) [ オプション ] [ バンドル名 | サービス名 ] [...]」
- ◆ 187 ページの 「what-provides (wp) pattern」
- ◆ 187 ページの 「what-requires (wr) pattern」

#### **checkpoint-add (cpa) (name) [date]**

チェックポイントを追加します。日付が指定されていない場合、現在の日付にチェックポイントを追加します。

#### **checkpoints-list**

保存されたチェックポイントを一覧表示します。

#### **checkpoint-remove (cpr) (name) [...]**

指定されたチェックポイントを削除します。

#### **dist-upgrade (dup) [ オプション ] [ バンドル名 | サービス名 ] [...]**

ディストリビューションアップグレードを実行します。このコマンドは、パッケージアーキテクチャの変更、パッケージスプリット、パターンおよび製品のアップデートなどを処理する特殊なリゾルバルゴリズムを使用してすべてのパッケージの更新を行います。引数を指定しない場合は、ZENworks バンドルと外部サービスのすべてのパッケージがディストリビューションアップグレードの対象になります。

次のオプションを受け入れます。

-d, --download-only - パッケージのダウンロードのみを実行します。

-N, --dry-run - 要求されたアクションをテストして表示しますが、実際には実行しません。

-y, --no-confirm - 確認を求めません。

たとえば、SLES 10 SP2 から SLES 10 SP3 にディストリビューションをアップグレードできます。

#### **info (info) [ オプション ] pattern**

指定されたパターンに一致する指定されたパッケージのパッケージ情報を表示します。次の正規表現がサポートされています。

正規表現	機能
<code>^abc</code>	「abc」から始まる文字列に一致します。
<code>[abc]</code>	「a」、「b」、または「c」を含む文字列に一致します。
<code>a*c</code>	0回以上の「a」の後に「c」、「ac」、「aac」、「aaac」、または「c」が続く場合に一致します。

このコマンドは、パッケージの名前、バージョン、アーキテクチャ、および説明を表示します。

次のオプションを受け入れます。

`--provides` - パッケージおよび指定されたパターンに一致するパッケージが提供するファイルを一覧表示します。

`--requires` - パッケージおよび指定されたパターンに一致するパッケージが必要とするファイルを一覧表示します。

#### **install (in) [ オプション ] (package name) [...]**

指定されたパッケージをインストールします。特定のバージョンの特定のパッケージを指定する場合、正確なバージョンのパッケージがインストールされます。特定の名前を持つ特定のパッケージを指定する場合、最適なバージョンのパッケージがインストールされます。

**注:** パッケージをダウングレードする場合は、パッケージをダウングレードするバージョンを指定します。

次のオプションを受け入れます。

`-b, --bundle [bundle name]` - 指定されたバンドルからパッケージをインストールします。指定するバンドルは Linux 依存バンドルとすることもできます。ただし、Linux 依存バンドルをインストールするには、[パッケージの発行] を True に設定する必要があります。パッケージの発行に関する詳細については、「[Creating Linux Dependency Bundles \(Linux 依存関係バンドルの作成\)](#)」を参照してください。

`-d, --download-only` - パッケージを zen キャッシュディレクトリにダウンロードします。デフォルトで、zen キャッシュディレクトリは `/var/opt/novell/zenworks/zmd/cache` にあります。

`-N, --dry-run` - 要求されたアクションをテストして表示しますが、実際には実行しません。

`-y, --no-confirm` - 確認を求めません。

`-f, --freshen` - 以前のバージョンのパッケージが存在する場合のみ、パッケージをインストールします。

`-s, --service [service name]` - 指定されたサービスからパッケージをインストールします。

`-p, --pattern [pattern name]` - 指定されたパターンのパッケージをインストールします。ただし、インストールされるのはパターンの必須パッケージ部分だけです。

`--entire-bundle [bundle name] [...]` - 指定されたバンドルからすべてのパッケージをインストールします。バンドルは 1 つまたは複数指定できます。

--ignore-dependency - 依存関係をチェックしません。依存パッケージはスキップされ、パッケージのみがインストールされます。

-r, --regex-pattern < 正規表現パターン > [< 正規表現パターン >]- 指定された正規表現に一致する、すべてのパッケージをインストールします。

次の正規表現がサポートされています。

正規表現	機能
^abc	「abc」から始まる文字列に一致します。
[abc]	「a」、「b」、または「c」を含む文字列に一致します。
a*c	0回以上の「a」の後に「c」、「ac」、「aac」、「aaac」、または「c」が続く場合に一致します。

さらに、コマンドはインストールを正常に完了するための依存パッケージも一覧表示します。

#### **list-updates (lu) [ オプション ] [ バンドル名 | サービス名 ] [...]**

指定されたバンドルおよびサービスに使用できる更新を表示します。更新の表示対象として、1つのバンドルまたはサービスを指定することも、複数のバンドルまたはサービスを指定することもできます。バンドルまたはサービスを指定しないと、使用できるすべての更新が表示されます。

次のオプションを受け入れます。

-t, --type [type] - 指定されたサービスタイプからの更新を一覧表示します。

#### **lock-add (la) (name or pattern)[version]**

パッケージロックルールを追加します。このコマンドは、指定されたパッケージをロックし、指定された正規表現または文字列に一致すると、アップグレードまたは削除されません。ワイルドカードパターンやバージョン番号の関係で、パッケージ名を個別に指定できます。バージョン番号は、一重または二重引用符で囲って指定してください。



関係演算子	機能
=	特定のパッケージバージョンのみをロックします。
<	指定されたバージョンより古いパッケージのバージョンをすべてロックします (指定したバージョンは除く)。
>	指定されたバージョンより新しいパッケージのバージョンをすべてロックします (指定したバージョンは除く)。
<=	指定されたバージョンより古いパッケージのバージョンをすべてロックします (指定したバージョンも含む)。
>=	指定したバージョンより新しいパッケージのバージョンをすべてロックします (指定したバージョンも含む)。

特定のバージョンのパッケージをインストールする場合、このコマンドはパッケージバージョンがロックされているかどうかを先に確認し、ロックされていない場合のみパッケージバージョンをインストールします。たとえば、パッケージ「X 1.7」より新しいすべてのバージョンが `zac la X '>' 1.7` コマンドを使用してロックされているとします。 `zac in X (1.9)` コマンドを使用して X 1.9 パッケージをインストールしようとする、インストールは失敗します。

#### **lock-delete (ld) (lock number) [...]**

番号で指定したパッケージロックを削除します。 `lock-list (ll)` コマンドを使用して、各ロックの番号を見つけられます。

#### **lock-list (ll)**

パッケージロックを一覧表示します。

#### **packagehistory (ph) [ オプション ] [...]**

ZENworks を使用して実行されるパッケージトランザクション履歴を一覧表示します。このコマンドはパッケージ名、日付、操作、バージョン、およびリリースを表示します。

次のオプションフラグを受け入れます。

-n, --search-name [package name] - パッケージ名で検索します。

-a, --search-action [action name] - アクション名で検索します。有効なアクション名は、None、Install、Removal、Upgrade です。

-d, --days-back [days] - 遡って調査する最大日数 (デフォルトは 30)。

-c, --clean-history - パッケージ履歴をクリアします。

-o, --search-operation [operation] - 実行される操作に基づいて検索します。実行される操作には、インストール、削除、またはアップグレードを指定できます。

#### **remove (rm) [ オプション ] (package name) [...]**

指定されたパッケージを削除します。

次のオプションを受け入れます。

- entire-bundle [bundle name] - 指定されたバンドルからすべてのパッケージを削除します。
- N, --dry-run - 要求されたアクションをテストして表示しますが、実際には実行しません。
- y, --no-confirm - 確認を求めません。
- r, --regex-pattern < 正規表現パターン > [< 正規表現パターン >] - 指定された正規表現に一致する、すべてのパッケージを削除します。

次の正規表現がサポートされています。

正規表現	機能
^abc	「abc」から始まる文字列に一致します。
[abc]	「a」、「b」、または「c」を含む文字列に一致します。
a*c	0回以上の「a」の後に「c」、「ac」、「aac」、「aaac」、または「c」が続く場合に一致します。

さらに、このコマンドは、指定されたパッケージを正常に削除するための依存パッケージも一覧表示します。

#### rollback (ro) [オプション][日付またはチェックポイント]

ロールバックの初期設定を設定している場合は、パッケージトランザクションを指定した日時までロールバックします。デフォルトで、ロールバックの初期設定は無効になっています。ロールバックの初期設定は、`zac set rollback true` コマンドを実行して有効にすることができます。

特定のバージョンのパッケージにロールバックする場合は、そのバージョンのパッケージとその依存関係がサービスまたは割り当てられたバンドルの一部として利用できる必要があります。そのバージョンのパッケージまたはその依存関係が利用できない場合、ロールバックは失敗します。

また、このコマンドは、正常にロールバックするために必要な依存パッケージも一覧表示します。

次のオプションを受け入れます。

- d, --download-only - パッケージのダウンロードのみを実行します。
- N, --dry-run - 要求されたアクションをテストして表示しますが、実際には実行しません。
- y, --no-confirm - 確認を求めません。
- p, --package (package name) - ロールバックする必要があるパッケージ名です。

#### search (se) [オプション][パターン]

指定されたパターンに一致するパッケージを検索します。引数が提供されない場合、すべてのパッケージが表示されます。このコマンドは、ソース名、ソースタイプ、パッケージ名、バージョン、およびアーキテクチャを表示します。

次のオプションを受け入れます。

- b, --bundle [bundle name] - 指定されたバンドル内のパッケージを検索します。

- d, --search-in-description - パッケージの説明内を検索します。パッケージ名は検索しません。
- i, --installed-only - インストールされたパッケージのみを表示します。
- u, --uninstalled-only - アンインストールされたパッケージのみを表示します。
- s, --service [service name] - 指定されたサービス内のパッケージを検索します。
- t, --type [service type] - 指定されたサービスタイプに基づいて、パッケージを検索します。
- p, --pattern [pattern name] - 利用可能なパターンを表示します。
- sort-by-bundle - 名前ではなく、バンドルでパッケージをソートします。デフォルトのソート方法はパッケージ名です。
- show-nevra - パッケージの NEVRA 詳細を表示します。

#### update (up) [オプション][バンドル名 | サービス名][...]

インストールされたパッケージを更新します。パッケージの更新対象として、1つのバンドルまたはサービス名を指定することも、複数のバンドルまたはサービス名を指定することもできます。バンドルまたはサービス名が指定されない場合、すべてのパッケージが更新されます。

名前とアーキテクチャが同じだが、バージョンが異なるパッケージがインストールされている場合、上位バージョンのパッケージだけがアップグレードされます。

次のオプションを受け入れます。

- d, --download-only - パッケージのダウンロードのみを実行します。
- N, --dry-run - 要求されたアクションをテストして表示しますが、実際には実行しません。
- y, --no-confirm - 確認を求めません。
- t, --type [service-type] - 指定された外部サービスタイプからパッケージを更新します。

#### what-provides (wp) pattern

指定されたパターンに一致するライブラリ、プログラム、またはパッケージを提供するパッケージを表示します。\* ワイルドカードパターンのみがサポートされています。

末尾が a のパッケージを表示する例：

```
zac wp *a
```

#### what-requires (wr) pattern

指定されたパターンに一致するライブラリ、プログラム、またはパッケージを要求するパッケージを一覧表示します。

a で始まるパッケージを表示する例：

```
zac wr a*
```

## パッチ管理コマンド

- ◆ [188 ページの「patch-scan \(ps\)」](#)
- ◆ [188 ページの「patch-scan --quick \(ps --quick\)」](#)

- ◆ 188 ページの「[patch-apply-policy \(pap\)](#)」
- ◆ 188 ページの「[patch-quarantine-release \(pqr\)](#)」

#### **patch-scan (ps)**

デバイスの現在のパッチ署名 (DAU) ファイルを使用して、パッチが適用されていないデバイスをスキャンします。その後、結果がサーバにアップロードされます。

パッチスキャンの実行例：

```
zac ps
```

#### **patch-scan --quick (ps --quick)**

サーバに最新スキャン結果をアップロードします。新たな検出スキャンは実行されません。

クイックパッチスキャンの実行例：

```
zac ps --quick
```

#### **patch-apply-policy (pap)**

すべてのパッチポリシーの最新バージョンを使ってデバイスを更新します。

パッチ適用ポリシーの実行例：

```
zac pap
```

#### **patch-quarantine-release (pqr)**

コマンドが検疫から実行されるデバイスで検疫済みパッチをリリースし、パッチポリシースケジュールまたは修復スケジュールのいずれかから、一度限りのインストールを試行できるようにします。

パッチ検疫リリースの実行例：

```
zac pqr
```

## **ポリシーコマンド**

- ◆ 188 ページの「[policy-list \(pl\)](#)」
- ◆ 188 ページの「[policy-refresh \(pr\)](#)」

#### **policy-list (pl)**

現在デバイスに強制的に適用されているポリシー (有効ポリシー) を一覧表示します。すべてのポリシー (有効ポリシーと有効でないポリシー) を一覧表示するには、`--all` オプションを使用します。

例：

```
zac pl
```

```
zac pl --all
```

#### **policy-refresh (pr)**

デバイスとユーザに割り当てられているすべてのポリシーを適用します。

例：

```
zac pr
```

## 登録コマンド

- ◆ 189 ページの「add-reg-key (ark) <登録キー>」
- ◆ 189 ページの「register (reg) [-k <key>] [-u <ユーザ名> -p <パスワード>] <ZENworks サーバアドレス : ポート>」
- ◆ 189 ページの「reregister (rereg) [-u <ユーザ名> -p <パスワード>] <新規 GUID>」
- ◆ 190 ページの「unregister (unr) [-f] [-s] [-a] [-u <ユーザ名> -p <パスワード>]」
- ◆ 190 ページの「reestablish-trust (retr) [-u <ユーザ名> -p <パスワード>]」

### add-reg-key (ark) <登録キー>

指定したキーを使用してデバイスを登録します。キーによる登録は追加的です。デバイスが以前キーで登録されていて、そのデバイスを新しいキーで登録する場合、デバイスは、両方のキーに関連付けられているすべてのグループの割り当てを受け取ります。

例：

```
zac ark キー 12
```

### register (reg) [-k <key>] [-u <ユーザ名> -p <パスワード>] <ZENworks サーバアドレス : ポート>

デバイスを管理ゾーンに登録します。このコマンドを実行するには、デバイスの登録を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

例：

```
zac reg -k キー 1 https://123.456.78.90
```

```
zac reg -k key1 https://zenserver.novell.com:8080
```

ポート番号は、ZENworks サーバが標準 HTTP ポート (80) を使用していない場合にのみ必要です。ユーザ名とパスワードが指定されていない場合は、指定するようにプロンプトが表示されます。

---

**注：**対応するデバイスオブジェクトがサーバ上にすでに存在し、そのデバイスオブジェクトで更新が発生した場合は、-k オプションは適用されません。

---

### reregister (rereg)[-u <ユーザ名> -p <パスワード>] <新規 GUID>

デバイスを現在のゾーンに登録し、既存のデバイスオブジェクトの GUID を割り当てます。現在関連付けられているデバイスオブジェクトは削除されます。このコマンドを実行するには、デバイスの再登録を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

たとえば、ハードドライブの交換後にデバイスのイメージを作成した場合、デバイスには新しい GUID が与えられる可能性があります。しかし、reregister コマンドを使用すれば、ハードドライブを交換する前に同じ GUID を割り当てることができます。

例：

再登録してユーザ名とパスワードを指定する

```
zac reregister -u myuser -p mypassword eaa6a76814d650439c648d597280d5d4
```

再登録して、ユーザ名とパスワードを入力するようプロンプトを表示する

```
zac reregister eaa6a76814d650439c648d597280d5d4
```

#### **unregister (unr) [-f] [-s] [-a] [-u <ユーザ名> -p <パスワード>]**

管理ゾーンからデバイスの登録を削除します。このコマンドを実行するには、デバイスの登録解除を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

例：

サーバに接続できないときに、強制的にローカルでデバイスを登録解除する場合：

```
zac unr -f -u myuser -p mypassword
```

ローカルで登録解除して、ユーザ名とパスワードのプロンプトを抑止する場合：

```
zac unr -s
```

非同期で登録解除するには、-a オプションを使用します。このオプションを使用すると、サーバによってデバイスが非同期で削除されます。

-a、-f、-u、および -p パラメータはオプションです。-u および -p パラメータを指定しないと、ユーザ名とパスワードを入力するよう求められます。-a オプションを指定した場合、ZENworks サーバによって登録解除コールが速やかに返されますが、データベースからのデバイスオブジェクトの削除は、後で非同期で実行されます。デバイス削除が完了していないのにデバイスの登録を再度試行した場合、ZENworks サーバがエラーを表示します。データベース内のデバイスに大量のデータが関連付けられている場合、デバイスの削除に長時間かかることがあります。サーバ上での実際のデバイス削除が長時間かかるため、エージェントの登録解除コマンドがタイムアウトする場合は、-a オプションを使用してください。

---

**注：**UNR コマンドを使用すると、データベースが過度に使用される可能性があります。これは、次のいずれかの理由による場合があります。

- ◆ サーバ上で UNR コマンドが実行されている。
- ◆ ゾーンに多数の管理対象デバイスが含まれている。
- ◆ 管理対象デバイスに膨大な量の履歴がある。
- ◆ パッチ管理が有効である。

---

#### **reestablish-trust (retr) [-u <ユーザ名> -p <パスワード>]**

現在の管理ゾーンとの信頼を再確立します。使用するユーザ名とパスワードはゾーン管理者のものであることが必要です。

例：

```
zac retr -u myuser -p mypassword
```

-u および -p パラメータはオプションです。-u および -p パラメータを指定しないと、ユーザ名とパスワードを入力するよう求められます。

## サテライトの役割コマンド

- ◆ 191 ページの「`remint-satellite-cert (rsc)`」
- ◆ 191 ページの「`satellite-cachecfg (satcfg)`」
- ◆ 191 ページの「`satellite-clearcache (satcc)`」
- ◆ 191 ページの「`satellite-refreshcfg (satrc)`」
- ◆ 191 ページの「`satellite-roles (satr)`」
- ◆ 191 ページの「`imgsatsserver status (iss status)`」

### **remint-satellite-cert (rsc)**

サテライトの SSL 証明書を改変します。

例 : `zac rsc`

### **satellite-cachecfg (satcfg)**

キャッシュからサテライト役割を読み込みます。

例 :

`zac satcfg`

### **satellite-clearcache (satcc)**

キャッシュからサテライト設定情報を消去します。

例 :

`zac satcc`

### **satellite-refreshcfg (satrc)**

サテライト役割を更新します。

例 :

`zac satrc`

### **satellite-roles (satr)**

現在のサテライト役割とその状態を報告します。

例 :

`zac satr`

### **imgsatsserver status (iss status)**

現在のステータス、PXE サービスステータス、イメージファイル、イメージング統計を含む、デバイスのイメージング役割のプロパティを表示します。

**現在のステータス** : デバイスが現在、イメージング操作を実行するように環境設定されているかどうか表示します。デバイスがイメージングの役割を持つサテライトになると、現在のステータスは[アクティブ]になります。

**PXE サービスステータス** : Proxy DHCP Service がデバイスで有効になっているかどうか表示します。

**PXE 要求** : イメージングサーバが前回起動されてからイメージングサーバが受信した各種イメージング要求の数。これには、失敗した要求、拒否された要求、および他のイメージングサーバを参照した要求の数も含まれています。ソース、タイプ、日時、結果などの各要求に関する情報は、イメージングサーバにログされます。

**送信したイメージ** : イメージングサーバが前回起動されてから、イメージングサーバがイメージングクライアントに送信したイメージの数。この数に含まれるのは、このイメージングサーバから取得されたイメージのみです。

**受信したイメージ** : イメージングサーバが前回起動されてから、イメージングサーバが受信して、保存した新しいイメージの数。これには、クライアント参照を通じて受信されたイメージが含まれます。

例 :

```
zac iss status
```

## ステータスコマンド

- 192 ページの「`agent-properties (agp)`」
- 193 ページの「`cache-clear (cc)`」
- 193 ページの「`refresh (ref) [general [bypasscache] | partial [bypasscache] | bypasscache]`」
- 193 ページの「`zenhttp-status (zhs)`」
- 193 ページの「`zone-config (zc)`」
- 193 ページの「`get-settings (gs) <キー>`」
- 194 ページの「`statussender (sts)`」

### **agent-properties (agp)**

ZENworks Agent についての情報を提供します。次のフィールドには ZENworks Agent に関する情報が記載されています。

**デバイスアドレス**。デバイスの IP アドレスです。

**デバイス名**。デバイスのコンピュータ名。

**デバイス状態**。デバイスの状態 : 管理対象、未管理、廃棄、または不明。不明の状態はエラーの場合に限り表示されます。

**サーバとの前回の接続**。ZENworks Agent が ZENworks サーバと最後に通信した時刻。

**サーバとの次回の接続**。ZENworks Agent が ZENworks サーバと通信する予定の時刻。

**プライマリユーザ**。現在ログインしているユーザ。

例 :



```
zac agp
```

```
zac agent-properties
```

#### **cache-clear (cc)**

デバイスの ZENworks キャッシュをクリアします。これにより、キャッシュデータベースのすべてのエントリと、これらのエントリに関連付けられたキャッシュファイルが削除されます。

例：

```
zac cc
```

#### **refresh (ref)[general [bypasscache] | partial [bypasscache] | bypasscache]**

一般更新または部分更新を呼び出して、すべての登録と環境設定を更新します。

bypasscache を使用して、更新中にサーバキャッシュからデータが使用されないようにします。このオプションはテストまたはトラブルシューティングに有効です。

例：

```
zac ref general bypasscache
```

```
zac ref partial bypasscache
```

```
zac ref bypasscache
```

#### **zenhttp-status(zhs)**

登録ハンドラのポートとタグを一覧表示します。

例：

```
zac zhs
```

#### **zone-config (zc)**

デバイスが環境設定情報を求めてアクセスしている ZENworks サーバの情報を表示します (設定サーバ)。または、設定サーバについての情報を一覧表示します。

例：

```
zac zc
```

#### **get-settings (gs) < キー >**

設定は更新のたびに設定モジュールからローカルキャッシュにダウンロードされます。これは指定されたキーに関連付けられている有効な設定を返します。

例：

```
zac gs キー 1
```

有効な ZENworks 設定キーはすべて、/var/opt/novell/zenworks/cache/zmd/settings ディレクトリに保存されます。

リモート管理設定を一覧表示する例：

```
zac gs RemoteManagement
```

### **statussender (sts)**

このコマンドは、プライマリサーバにステータス情報をロールアップします。前回ステータスがロールアップされてから更新された情報をロールアップするか、完全なステータス情報をロールアップすることができます。

例：

最後に成功したステータスロールアップ以降に更新されたステータス情報をロールアップするには：

```
zac sts rollup
```

完全なステータス情報をロールアップするには：

```
zac sts rollup full
```

# Macintosh 用の zac(1)

## 名前

zac - Macintosh 管理対象デバイスにインストールされて実行される Micro Focus ZENworks Agent のコマンドライン管理インターフェースです。

## 構文

*zac command options*

## 説明

zac ユーティリティは、ソフトウェアバンドルのインストールと削除、ポリシーの適用、およびデバイスの登録と登録解除など、ZENworks 管理デバイス上でコマンドライン管理機能を実行します。

## 使用方法

このセクションでは、一般的なコマンド形式と規則について説明しています。

ほとんどのコマンドは、長い形式と短い形式があります。

ロングフォーム : add-reg-key

ショートフォーム :ark

両方の形式が使用できる場合、コマンドは以下のように表示されます。

add-reg-key (ark) オプション

コマンドを使用する場合は、長い形式または短い形式のいずれかのみを入力します。

zac add-reg-key オプション

zac ark オプション

オプションにスペースが含まれている場合は、引用符で囲みます。

zac ark " オプション1"

### 必須の引数とオプションの引数

括弧 ( ) で囲まれた引数は必須で、角括弧 [ ] で囲まれた引数はオプションです。

### 省略記号 (...)

省略記号は、コマンドが最後の引数タイプのエントリを複数受け入れることを示します。引数は、コマンドに応じてスペースまたはカンマで区切る必要があります。例を次に示します。

zman server-add-bundle [ オプション ] ( サーバ ) ( バンドル ) [...]

この省略記号は、server-add-bundle が複数のバンドルを受け入れることができることを示しています。コマンドレベルのヘルプでは、複数エントリの引数ヘルプは、より明確に [( バンドル ) ( バンドル ) ... ( バンドル )] として指定されます。

## コマンドのリスト

- 196 ページの「バンドルコマンド」
- 198 ページの「コアコマンド」
- 202 ページの「インベントリコマンド」
- 202 ページの「インベントリ専用コマンド」
- 203 ページの「パッチ管理コマンド」
- 204 ページの「サテライトの役割コマンド」
- 204 ページの「ステータスコマンド」
- 205 ページの「システムアップデートコマンド」

## バンドルコマンド

- 196 ページの「bundle-install (bin)」
- 197 ページの「bundle-launch (bln)」
- 197 ページの「bundle-list (bl)」
- 197 ページの「bundle-lock-add (bla) [ オプション ] [ バンドル名 ] [...]
- 197 ページの「bundle-lock-list (bl1)」
- 197 ページの「bundle-lock-remove (blr)」
- 198 ページの「bundle-props (bp)」
- 198 ページの「bundle-refresh (br)」
- 198 ページの「bundle-search (bse)」
- 198 ページの「bundle-uninstall (bu)」
- 198 ページの「bundle-verify (bv)」

### bundle-install (bin)

指定された正規表現または文字列、あるいはバンドル GUID に一致する表示名に基づいて指定されたバンドルをインストールします。

bundle-list コマンドを使用して、使用できるバンドルおよびその表示名のリストを取得します。

バンドルのインストール例

```
zac bin MacBundle
```

### **bundle-launch (bln)**

指定したバンドルを起動します。[起動] アクションセットのアクションが実行されます。インストールされていないバンドルを起動しようとする場合、バンドルが最初にインストールされてから起動されます。

bundle-list コマンドを使用して、使用できるバンドルおよびその表示名のリストを取得します。

表示名に基づいたバンドルの起動例：

```
zac bln MacBundle
```

### **bundle-list (bl)**

デバイスおよびログインユーザに割り当てられているバンドルのリストを表示します。このコマンドの出力は列に表示され、管理対象デバイスに割り当てられている、または利用可能なすべてのバンドルの詳細が表示されます。[表示名]、[バージョン]、および [バンドルタイプ] 列は、バンドル表示名、バンドルバージョン番号またはバンドルがサンドボックスバージョンの場合はサンドボックスのバージョン番号、およびバンドルタイプをそれぞれ示します。

[ステータス] 列には次の状態が表示されます。

使用可能：バンドルがインストールされている場合はこの状態を表示します。

インストールされていません：特定のバンドルバージョンがインストールされていない場合にこの状態を表示します。

ダウンロードされました：コンテンツがダウンロードされるが [インストール] アクションセットのアクションが実行されていない場合にこの状態を表示します。

すべてのバンドルを一覧表示する例：

```
zac bl
```

### **bundle-lock-add (bla) [オプション][バンドル名][...]**

アップグレードまたは削除されないように、指定された正規表現または文字列に一致する指定されたバンドルをロックします。

バンドルロックは [インストール] アクションセットにのみ適用できます。

次のオプションを受け入れます。

-li, --lock-install - バンドルをロックし、ロックに指定されていないバンドルがインストールされないようにします。

-t, --type - ロックするバンドルタイプを指定します。指定できるタイプは「MAC」と「Patch」です。タイプを指定しないと、bundle タイプが選択されます。

### **bundle-lock-list (bll)**

バンドルロックを一覧表示します。バンドルロックは ID 番号でソートされます。

バンドルを一覧表示する例：

```
zac bll
```

### **bundle-lock-remove (blr)**

番号で指定したバンドルロックを削除します。bundle-lock-list コマンドを使用して、各ロックの番号を見つけられます。

すべてのバンドルロックを削除する例：

```
zac blr --remove-all
```

#### **bundle-props (bp)**

バンドルのプロパティ情報を返します。

指定したバンドルのバンドルプロパティを取得する例：

```
zac bp MacBundle
```

#### **bundle-refresh (br)**

指定されたバンドルの情報を更新します。

バンドルの更新例：

```
zac br MacBundle
```

#### **bundle-search (bse)**

指定された Java 正規表現または文字列に一致するバンドルを検索します。

#### **bundle-uninstall (bu)**

指定したバンドルをアンインストールします。[元に戻す]アクションをサポートするアクションは、バンドルのアンインストール中に実行されます。バンドルのアンインストールが完了した後で、バンドルの状態が「インストールされていません」と表示されます。

bundle-list コマンドを使用して、インストールされたバンドルおよびその表示名のリストを取得します。

バンドルのアンインストール例：

```
zac bu MacBundle
```

#### **bundle-verify (bv)**

バンドルを再インストールし、[検証]アクションセットを実行して、削除されたファイルや破損したファイルがないことを確認します。bundle-list コマンドを使用して、インストールされたバンドルおよびその表示名のリストを取得します。

バンドルの検証例：

```
zac bv MacBundle
```

## コアコマンド

- ◆ 199 ページの「[add-reg-key \(ark\)](#)」
- ◆ 199 ページの「[agent-properties \(agp\)](#)」
- ◆ 199 ページの「[cache-clear \(cc\)](#)」
- ◆ 199 ページの「[cert-info \(ci\)](#)」
- ◆ 199 ページの「[config-location \(cl\)](#)」
- ◆ 200 ページの「[file-system-guid \(fsg\)](#)」
- ◆ 200 ページの「[get-pref \(get\)](#)」

- ◆ 200 ページの「get-settings (gs)」
- ◆ 200 ページの「logger [resetlog | level [MANAGED|ERROR|WARN|INFO|DEBUG] | managedlevel]」
- ◆ 201 ページの「reestablish-trust (retr)」
- ◆ 201 ページの「refresh (ref) [general | partial bundle <バンドル表示名> | bypasscache]」
- ◆ 201 ページの「register (reg)」
- ◆ 201 ページの「reregister (rereg)」
- ◆ 201 ページの「set-pref (set)」
- ◆ 201 ページの「set-proxy (sp) [ オプション ] <IP アドレス / ホスト名 : ポート >」
- ◆ 202 ページの「unregister (unr)」
- ◆ 202 ページの「zone-config (zc)」

#### **add-reg-key (ark)**

このコマンドは、指定されたレジストリキーを使用してデバイスを登録します。

例 : `zac add-reg-key`

#### **agent-properties (agp)**

このコマンドは、ZENworks Agent のバージョン情報を表示します。

例 : `zac agent-properties`

#### **cache-clear (cc)**

このデバイスのキャッシュをクリアします。

例 : `zac cc`

#### **cert-info (ci)**

既知の ZCM サーバごとに公開鍵証明書情報を一覧表示するか、コンピュータストアに信頼するルート証明書を追加します。

例 :

既知の ZENworks ユーザごとに証明書を一覧表示するには、次のように指定します。

`zac ci`

デバイスの信頼するストアに信頼するルート証明書を追加するには、次のコマンドを実行します。

`zac ci <場所> -u myuser -p mypassword`

#### **config-location (cl)**

現在の場所を表示します。

例 :

`zac config-location`

### file-system-guid (fsg)

イメージ取得の準備のために、ファイルシステム内で、ワークステーション GUID の削除と復元を行います。

例：

GUID 値を表示する場合：

```
zac fsg
```

ファイルシステムから GUID を削除し、conninfo.dat も削除する場合：

```
zac file-system-guid -d
```

ファイルシステムに GUID を復元する場合：

```
zac file-system-guid -r
```

GUID 値を印刷するオプションを指定しない場合：

```
zac fsg
```

### get-pref (get)

現在設定されている初期設定を表示します。

abc で始まる名前のすべての初期設定を表示した例：

```
zac get-pref abc
```

### get-settings (gs)

設定は更新のたびに設定モジュールからローカルキャッシュにダウンロードされます。これは指定されたキーに関連付けられている有効な設定を返します。

例：

```
zac gs キー 1
```

get-settings (gs) コマンドを使用して、さまざまなキーを使用する設定を一覧表示できます。詳細については、[ステータスコマンドセクションの get-settings \(gs\) <キー>](#) を参照してください。

### logger [resetlog | level [MANAGED|ERROR|WARN|INFO|DEBUG] | managedlevel]

ロガーの環境設定を変更します。

次のオプションを使用できます。

resetlog - ログをリセットします。

level - レベルなしで使用した場合、現在管理されているログインレベルが表示されます。レベルを指定した場合には、指定したレベルにログレベルを変更します。

managedlevel - ゾーンのグローバルログレベルを表示します。

ログファイルをリセットする例：

```
zac logger resetlog
```

現在のログレベルを表示する例：

```
zac logger level
```



ログレベルを DEBUG 以上に設定する例：

```
zac logger level DEBUG
```

#### reestablish-trust (retr)

このコマンドは、現在のゾーンを使用して信頼を再確立します。

現在の管理ゾーンとの信頼を再確立します。使用するユーザ名とパスワードはゾーン管理者のものである必要があります。

例：

```
zac retr -u myuser -p mypassword
```

-u および -p パラメータはオプションです。-u および -p パラメータを指定しないと、ユーザ名とパスワードを入力するよう求められます。

#### refresh (ref)[general | partial bundle <バンドル表示名> [bypasscache]

一般更新を呼び出してすべてのバンドル、ポリシー、登録、環境設定を更新します。部分更新を呼び出して、すべてのポリシー、登録、環境設定を更新します。

bypasscache を使用して、更新中にサーバキャッシュからデータが使用されないようにします。このオプションはテストまたはトラブルシューティングに有効です。

例：

```
zac ref general bypasscache
```

```
zac ref partial bypasscache
```

#### register (reg)

このコマンドは、指定されたゾーンにデバイスを登録します。

このコマンドを実行するには、デバイスの登録を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

例：

```
zac reg -k キー 1 https://123.456.78.90
```

#### reregister (rereg)

このコマンドは、デバイスを現在のゾーンに登録し、既存のデバイスオブジェクトの GUID を割り当てます。現在関連付けられているデバイスオブジェクトは削除されます。このコマンドを実行するには、デバイスの再登録を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

再登録してユーザ名とパスワードを指定する

```
zac reregister -u myuser -p mypassword eaa6a76814d650439c648d597280d5d4
```

#### set-pref (set)

初期設定値を設定します。

初期設定値 abc を 123 に設定した例：

```
zac set-pref abc 123
```

#### set-proxy (sp) [オプション] <IP アドレス / ホスト名 : ポート>

ZENworks サーバに直接接続する代わりに、接続するプロキシを指定します。

オプションは次のとおりです。

--default - 管理ゾーンのプロキシ設定によって上書きできるプロキシを設定します。

--clear - 現在のプロキシをクリアし、管理ゾーンのプロキシ設定を使用します。

例：

#### IPv4:

```
zac sp 123.456.78.90:2349 administrator novell
```

```
zac sp /default 123.456.78.90:2349
```

```
zac sp /clear
```

ユーザ名とパスワードを指定していない場合は、それらの入力を求められます。

#### unregister (unr)

このコマンドは、現在のゾーンからデバイスを登録解除します。

このコマンドを実行するには、デバイスの登録解除を試みているフォルダで、デバイスの [作成 / 削除] 権限を持っている必要があります。

例：

```
zac unr -u administrator -p novell
```

---

**注：**UNR コマンドを使用すると、データベースが過度に使用される可能性があります。これは、次のいずれかの理由による場合があります。

- ◆ サーバ上で UNR コマンドが実行されている。
  - ◆ ゾーンに多数の管理対象デバイスが含まれている。
  - ◆ 管理対象デバイスに膨大な量の履歴がある。
  - ◆ パッチ管理が有効である。
- 

#### zone-config (zc)

このコマンドはゾーン設定サーバを表示するか、現在のゾーン設定を一覧表示します。

```
zac zc
```

## インベントリコマンド

- ◆ [202 ページ](#)の「`inventory (inv) [scannow]`」

#### inventory (inv) [scannow]

インベントリスキャンを実行します。

インベントリスキャンの実行例：

```
zac inv scannow
```

## インベントリ専用コマンド

インベントリ専用エージェント (IOA) でのみ使用するコマンドです。

### zac ioa cfg <サーバ IP>

MAC IOA デバイスを、指定されたサーバ IP に設定します。

MAC インベントリ専用デバイスをサーバ <10.70.13.124> に登録する場合の例

```
zac ioa cfg 10.70.13.124
```

### zac ioa version

インベントリ専用エージェント (IOA) のバージョンを表示します。

### inventory (inv) [scannow]

インベントリスキャンを実行します。

インベントリスキャンの実行例：

```
zac inv scannow
```

## パッチ管理コマンド

- 203 ページの「`patch-scan (ps)`」
- 203 ページの「`patch-scan --quick (ps --quick)`」
- 203 ページの「`patch-apply-policy (pap)`」
- 203 ページの「`patch-quarantine-release (pqr)`」

### patch-scan (ps)

デバイスの現在のパッチ署名 (DAU) ファイルを使用して、パッチが適用されていないデバイスをスキャンします。その後、結果がサーバにアップロードされます。

パッチスキャンの実行例：

```
zac ps
```

### patch-scan --quick (ps --quick)

サーバに最新スキャン結果をアップロードします。新たな検出スキャンは実行されません。

クイックパッチスキャンの実行例：

```
zac ps --quick
```

### patch-apply-policy (pap)

すべてのパッチポリシーの最新バージョンを使ってデバイスを更新します。

パッチ適用ポリシーの実行例：

```
zac pap
```

### patch-quarantine-release (pqr)

コマンドが検疫から実行されるデバイスで検疫済みパッチをリリースし、パッチポリシースケジュールまたは修復スケジュールのいずれかから、一度限りのインストールを試行できるようにします。

パッチ検疫リリースの実行例：

```
zac pqr
```

## サテライトの役割コマンド

- ◆ 204 ページの「[satellite-cachecfg \(satcfg\)](#)」
- ◆ 204 ページの「[satellite-clearcache \(satcc\)](#)」
- ◆ 204 ページの「[satellite-refreshcfg \(satrc\)](#)」
- ◆ 204 ページの「[satellite-roles \(satr\)](#)」

### satellite-cachecfg (satcfg)

キャッシュからサテライト役割を読み込みます。

キャッシュからサテライト役割を読み取る例：

```
zac satcfg
```

### satellite-clearcache (satcc)

キャッシュからサテライト設定情報を消去します。

キャッシュからサテライト設定情報を消去する例：

```
zac satcc
```

### satellite-refreshcfg (satrc)

サテライト役割を更新します。

サテライト役割を更新する例：

```
zac satrc
```

### satellite-roles (satr)

現在のサテライト役割とその状態を報告します。

現在のサテライト役割と状態を報告する例：

```
zac satr
```

## ステータスコマンド

- ◆ 204 ページの「[zenhttp-status \(zhs\)](#)」

### zenhttp-status(zhs)

登録ハンドラのポートとタグを一覧表示します。

登録済みハンドラのポートとタグを一覧表示する例：

```
zac zhs
```

このコマンドは、エージェントがサテライトとして昇格される場合にのみ適用されません。

## システムアップデートコマンド

`zac zeus-refresh / zeus-ref`

システム更新がデバイスに割り当てられている場合、そのシステム更新を取得します。



# 2 ZENworks 診断センター

ZENworks が提供する ZENworks Diagnostic Center (ZDC) ツールを使用すると、システム更新を展開する前にプライマリサーバと管理ゾーンのヘルスを検証できます。

ZENworks Diagnostic Center (ZDC) は、プライマリサーバ上で一連の診断テストを実行し、プライマリサーバと管理ゾーンの状態を判断します。ZDC は次の検証機能から構成されます。

- ◆ **ZENworks System Files Verifier:** ZENworks が必要とする重要なシステムファイルの存在と整合性を検証します。
- ◆ **ZENworks Database Schema Verifier:** ZENworks が使用するデータベースのスキーマ ( テーブル、インデックス、データベースプロシージャなど ) の適切性を検証します。
- ◆ **ZENworks Content Verifier:** ZDC がインストールされて実行されているプライマリサーバ上にホストされているコンテンツの存在と整合性を検証します。

ZENworks 2017 のアップグレード前にデータベーススキーマを検証するには、ZENworks 2017 インストールメディアに含まれる ZENworks Diagnostic Center ツールを使用します。2017 にアップグレードした後でプライマリサーバと管理ゾーンのヘルスを検証する場合は、ZENworks の ZENworks コントロールセンターから ZENworks Diagnostic Center をダウンロードして使用します。詳細については、次のセクションを参照してください。

- ◆ [207 ページのセクション 2.1 「ZDC のダウンロードと抽出」](#)
- ◆ [208 ページのセクション 2.2 「ZDC の実行」](#)

## 2.1 ZDC のダウンロードと抽出

ツールを実行したいプライマリサーバに ZENworks Diagnostic Center をダウンロードして抽出する必要があります。

- 1 Web ブラウザを開いて、次のアドレスを入力します。

*Novell のダウンロード* Web サイト。

- 2 [ZCM <バージョン> ZENworks Diagnostic Center - 実行ファイルとパターンファイル] ページから、ZENworksDiagnosticCenter.zip ファイルを一時的な場所にダウンロードし、ZIP ファイルの内容を展開します。

ZENworks システムファイルと ZENworks データベーススキーマでは、現在の展開データと比較し、正常性を評価するために、良好と確認済みの展開から収集されたベースラインデータが必要です。

ZIP ファイルの data ディレクトリには、ベースラインデータファイルの `zdc_<バージョン>.zdc` があります。このファイルは、ZENworks Configuration Management の良好と確認済みの展開から収集されたベースラインデータを含みます。

ZDC を実行するには、ZDC 実行ファイルと設定ファイルが必要です。ZENworks Diagnostic Center を含むディレクトリは `$ZDC_Home` になります。

## 2.2 ZDC の実行

- 1 プライマリサーバのコンソールプロンプトで、\$ZDC\_Home ディレクトリに移動します。
- 2 適切なコマンドを実行します。
  - ◆ ZENworks システムファイル、ZENworks データベーススキーマ、および ZENworks コンテンツを検証するには、次のように指定します。

**Windows の場合** : zdc.bat verify -v systemfiles,zendatabase,zencontent -b <ZDC ベースラインデータファイルの完全なパス>

**Linux の場合** : ./zdc verify -v systemfiles,zendatabase,zencontent -b <ZDC ベースラインデータファイルの完全なパス>
  - ◆ ZENworks システムファイルのみを検証する場合は、次のように指定します。

**Windows の場合** : zdc.bat verify -v systemfiles -b <ZDC ベースラインデータファイルの完全なパス>

**Linux の場合** : ./zdc verify -v systemfiles -b <ZDC ベースラインデータファイルの完全なパス>
  - ◆ ZENworks データベーススキーマのみを検証する場合は、次のように指定します。

**Windows の場合** : zdc.bat verify -v zendatabase -b <ZDC ベースラインデータファイルの完全なパス>

**Linux の場合** : ./zdc verify -v zendatabase -b <ZDC ベースラインデータファイルの完全なパス>
  - ◆ ZENworks コンテンツのみを検証する場合は、次のように指定します。

**Windows の場合** : zdc.bat verify -v zencontent

**Linux の場合** : ./zdc verify -v zencontent

ZDC を実行する際は、次のコマンドオプションも指定できます。

- s: セッション名
- l: ZDC ログファイル名の完全なパス
- r: ZDC レポートディレクトリの完全なパス

ZDC は、診断テストを実行し、レポートを HTML 形式で生成します。このレポートは、Windows プライマリサーバの場合は \$ZDC\_Home\reports\timestamp ディレクトリ、Linux プライマリサーバの場合は \$ZDC\_Home/reports/timestamp ディレクトリに保存されます。レポートを表示するには、index.html(report ディレクトリ)を開きます。

診断テストの完全なログは、Windows プライマリサーバの場合は \$ZDC\_HOME\logs\timestamp.log に、Linux プライマリサーバの場合は \$ZDC\_HOME/logs/timestamp.log に保存されます。

---

**重要** : Microsoft SQL Server 上の ZENworks データベースで、ZDC を実行してスキーマの差異を確認します。zDevice テーブルのシリアル番号列に対して一意の制約が作成されていない場合、管理ゾーンに接続されているデバイスに重複するシリアル番号が割り当てられることがあります。詳細については、「[Troubleshooting the database migration failure, duplication of serial numbers in the zDevice table](#)」を参照してください。

---



## 監査データベースの検証

ZDC ツールを使用して、ZENworks の監査データベーススキーマを検証する手順は次のとおりです。

- ◆ **Windows の場合** : `zdc.bat verify -v auditdatabase -b <ZDC ベースラインデータファイルの完全なパス>` コマンドを実行します。
- ◆ **Linux の場合** : `./zdc verify -v auditdatabase -b <ZDC ベースラインデータファイルの完全なパス>` コマンドを実行します。

1 つのインスタンスでコマンドを実行して、監査データベースと ZENworks データベースを検証することはできません。これらのデータベースは、別々に検証する必要があります。

監査用の ZDC レポートは、`reports\timestamp` ディレクトリにあります。このレポートは、`index.html` ファイルから参照できます。



# 3 ZENworks コマンドラインユーティリティのトラブルシューティング

次のセクションでは、Micro Focus ZENworks コマンドラインユーティリティを使用中に発生する可能性がある問題の解決方法について説明します。

- 211 ページの「コマンドラインリダイレクションオペレータの > または >> などを使用してコマンドの出力がファイルにリダイレクトされるときに zman がハングする」
- 212 ページの「system-update-staging-group-create コマンドに「stage group name」オプション引数を渡さなかった場合、間違った成功メッセージが表示される」
- 212 ページの「zman ssaimg コマンドを使用してサテライトに追加したイメージングの役割がただちにサーバに適用されない」
- 212 ページの「zman コマンドは、ZENworks Linux サーバの非 root ユーザには機能しません。」
- 213 ページの「非ルートユーザは、zman ssc コマンドを使用してサテライト役割を設定できません。」
- 214 ページの「collectionStatsDB データベースが破損していると、zac cuo コマンドは、サテライトサーバからプライマリサーバにエージェント監査ファイルをアップロードしません。」
- 214 ページの「一部の zman コマンドは、OutOfMemory エラーが発生して失敗します。」
- 215 ページの「サテライトの役割を削除するために使用する zman コマンド ssd が失敗します。」
- 215 ページの「コンテンツの複製スケジュールが短時間に設定されている場合に RHEL サテライトサーバ上で zac コマンドがハングする」

## コマンドラインリダイレクションオペレータの > または >> などを使用してコマンドの出力がファイルにリダイレクトされるときに zman がハングする

ソース : ZENworks; zman

考えられる原因 : zman で各コマンドを実行するには、ZENworks 管理者のユーザ名とパスワードが必要です。コマンドで資格情報を指定しなかった場合には、同じものを入力するように要求されます。ただし、コマンドがファイルにリダイレクトされている場合、ユーザ名とパスワードのプロンプトもファイルにリダイレクトされるため、資格情報を入力するよう要求するプロンプトがユーザに表示されません。その結果、zman は、資格情報の入力を待ち続けて、ハングしたのと同じような状態になります。

アクション: -U|--User および -P|--Password オプションを使用して、コマンドの一部として資格情報を指定します。

アクション: admin-store-credential コマンドを使用して、資格情報を保存し、コマンドを実行します。

アクション: > や >> などのリダイレクション演算子の代わりに、-R|--Redirect オプションを使用して、出力をファイルにリダイレクトします。

## system-update-staging-group-create コマンドに「stage group name」オプション引数を渡さなかった場合、間違った成功メッセージが表示される

ソース: ZENworks; zman

説明: system-update-staging-group-create コマンドにステージグループ名オプション引数を渡さなかった場合、zman は誤ったメッセージを表示します。

```
Successfully created the object "null"
```

アクション: 新たに作成されたステージンググループを含む、ステージンググループのリストを表示するには、コンソールプロンプトで system-update-staging-group-list (susgl) コマンドを実行します。

## zman ssaimg コマンドを使用してサテライトに追加したイメージングの役割がただちにサーバに適用されない

ソース: ZENworks; zman

説明: zman ssaimg コマンドを使用してイメージングの役割を設定した場合、役割はデバイスにただちに追加されます。役割がただちに追加されない場合は、次回のデバイス更新スケジュールのときにのみ追加されます。

アクション: 役割をただちにデバイスに適用したい場合には、次のいずれかの方法で、手動でデバイスを更新する必要があります。

- ◆ コマンドプロンプトで、zman workstation-refresh|wrf コマンドを実行します。管理デバイスがサーバの場合には、zman server-refresh|srf コマンドを実行します。
- ◆ 管理対象デバイスで ZENworks アイコンを右クリックし、[[更新]] をクリックします。

## zman コマンドは、ZENworks Linux サーバの非 root ユーザには機能しません。

ソース: ZENworks; zman

説明： ZENworks Linux サーバ上で非 root ユーザが zman コマンドを実行すると、次のエラーメッセージが表示されます。

zman コマンドを実行する十分な権限がありません。Linux の非ルートユーザである場合、zman コマンドを実行するには zmanusers グループに属している必要があります。

アクション： zman コマンドを実行する前に、非 root ユーザを zmanusers グループに追加します。

- 1 ZENworks サーバに root としてログインします。
- 2 サーバのコンソールプロンプトで、次のようにコマンドを実行します。
  - ◆ ユーザがすでに Linux デバイスに存在する場合は、次のコマンドを実行します。

```
groupmod -A non-rootusername zmanusers
```
  - ◆ ユーザが Linux デバイスに存在しない場合は、次のコマンドを実行します。

```
useradd -G zmanusers non-rootusername
```
- 3 非 root ユーザとして ZENworks サーバにログインします。

**非ルートユーザは、zman ssc コマンドを使用してサテライト役割を設定できません。**

ソース： ZENworks; zman

説明： 非ルートユーザが Linux プライマリサーバ上で zman ssc コマンドを使用してサテライト役割を設定しようとする、次のエラーメッセージが表示されます。

```
The role you specified (Satellite_role_name) is not a valid satellite server role. Valid roles are:
```

```
ZoneConfig.ROLE_COLLECTION
ZoneConfig.ROLE_CONTENT
ZoneConfig.ROLE_AUTHENTICATION
```

アクション： 次の操作を実行してください：

- 1 zmanusers グループに非ルートが追加されていることを確認します。
  - 1a ZENworks サーバに root としてログインします。
  - 1b サーバのコンソールプロンプトで、次のようにコマンドを実行します。
    - ◆ ユーザがすでに Linux デバイスに存在する場合は、次のコマンドを実行します。

```
groupmod -A non-rootusername zmanusers
```

- ◆ ユーザが Linux デバイスに存在しない場合は、次のコマンドを実行します。

```
useradd -G zmanusers non-rootusername
```

- 2 usermod -A non-root\_username コマンドを使用して、非ルートユーザを ZENworks ユーザグループに追加します。その後、非ルートユーザはサテライト役割を設定できます。

## collectionStatsDB データベースが破損していると、zac cuo コマンドは、サテライトサーバからプライマリサーバにエージェント監査ファイルをアップロードしません。

ソース : ZENworks; zac

説明 : エージェント監査ファイルがサテライトサーバにアップロードされる際、collectionStatsDB データベース内の監査ファイルへの参照は保持されます。データベースが破損している場合、すべての参照が失われ、ファイルは孤立することになります。そのため、エージェント監査ファイルをプライマリサーバにアップロードするために zac cuo コマンドを実行すると、エラーが発生します。

アクション : エージェント監査ファイルをプライマリサーバにアップロードするには、サテライトサーバで次の手順を実行してください。

- 1 エージェントサービスを停止します。
- 2 孤立した監査ファイルへの参照が含まれている /var/opt/novell/zenworks/work/collection/collectionStatsDB フォルダを削除します。
- 3 エージェントサービスを起動します。
- 4 zac cuo コマンドを実行します。

## 一部の zman コマンドは、OutOfMemory エラーが発生して失敗します。

ソース : ZENworks; zman

説明 : 一部の zman コマンドは、OutOfMemory エラーが発生して失敗する可能性があります。

アクション : zman の最大ヒープサイズを増やすには、次のように指定します。

**Windows の場合 :** %ZENWORKS\_HOME%\conf\zman\properties\zman-config.properties ファイルで、JVM\_STARTUP\_OPTIONS の最大ヒープサイズ値を 128m から 512m に増やします。

**Linux:** /opt/novell/zenworks/bin/zman ファイルで、MEM\_OPTS の最大ヒープサイズ値を 256m から 512m に増やします。

## サテライトの役割を削除するために使用する zman コマンド `ssd` が失敗します。

ソース : ZENworks; zman

説明 : zman コマンド `ssd` が、次のエラーにより失敗します。

An internal error occurred. Please check the zman log for more ( 内部エラーが発生しました。詳細については、zman ログを確認してください)

サテライトから役割を削除するための zman コマンド `ssd` を使用すると、役割が削除されますが、サテライト更新クイックタスクの送信には失敗します。

考えられる原因 : これはサテライト用のサテライト更新クイックタスクがすでに処理中であるか、これから処理されるためです。

アクション : なし

## コンテンツの複製スケジュールが短時間に設定されている場合に RHEL サテライトサーバ上で `zac` コマンドがハングする

ソース : ZENworks、zac

説明 : RHEL サテライトサーバ上で `zac` コマンドを実行する際に、プロセスが無期限にハングします。

考えられる原因 : コンテンツの複製スケジュールが短時間に設定されています。たとえば、5分です。

---

注 : コンテンツの複製スケジュールを 12 時間に設定することをお勧めします。

---

アクション : ZENworks コントロールセンターにログインして、ゾーンのシステム変数 `enable.cdp.cleanup.thread.wait.time` を追加し、値を `True` に設定します。





# 4 Zman で作業するためのガイドライン

Micro Focus ZENworks の zman コマンドラインユーティリティで作業を行う場合には、次のガイドラインに従ってください。

- bundle-create や policy-create などのコマンドを使用してエクスポートされる XML ファイルに、ñ、ë、Ä などの発音記号付きや拡張 ASCII 文字が含まれている場合、エディタでファイルを開くには、UTF-8 エンコーディングを使用する必要があります。ファイルに何らかの変更を加えた場合には、UTF-8 形式で保存する必要があります。
- 拡張 ASCII 文字を含むコマンド出力を Windows 上のファイルにリダイレクトしたい場合、コマンドラインのリダイレクション演算子 (>) を使用することはできません。これは、MS-DOS プロンプトのコードページは、大部分のロケールでファイルに書き出す場合に使用されるコードページとは異なっているからです。

たとえば、英語、フランス、ドイツ語、スペイン語などの西ヨーロッパ言語の場合、DOS プロンプトのコードページは cp437 または cp850 ですが、Windows オペレーティングシステムの他の部分では cp1252 を使用しています。

出力をファイルにリダイレクトし、拡張 ASCII 文字を正しく保持するには、-R|--Redirect グローバルオプションを使用してください。

- PuTTY などのクライアントを使用して Windows マシンから Linux サーバに接続する場合には、変換に使用する文字セットを UTF-8 に設定してください。これにより、標準 ASCII 文字以外の文字の変換が適切に行われます。

PuTTY における UTF-8 への変換に文字セットを設定する

1. PuTTY クライアントを開きます。
2. PuTTY 設定ウィンドウで、[Windows] > [変換] をクリックします。
3. [Received Data Assumed to be in Which Character Set] ドロップダウンリストで [UTF-8] を選択します。

---

**注:** コマンドの入出力に a ~ z または A ~ Z の文字しか含まれていない場合には、この設定は必要ありません。

---

- デフォルトでは、zman はサーバのデフォルトのロケールを使用します。次の手順を実行して、zman で特定の言語を使用することができます。
  - Windows では、ZENworks\_Installation\_directory\Novell\ZENworks\conf\zman\properties\zman-config.properties の JVM\_STARTUP\_OPTIONS プロパティの値として、適切なユーザ言語とファイルエンコーディングを指定します。

たとえば、英語で zman を実行するには、JVM\_STARTUP\_OPTIONS の値を -Duser.language=en -Dfile.encoding=cp850 に設定します。ファイルを編集した後で、chcp 850 コマンドを使用して、コマンドプロンプトのコードページを cp850 に変更します。

- ◆ Linux では、`/opt/novell/zenworks/bin/zman` で、`/opt/novell/zenworks/lib/java/bin/java` の後に `-Duser.language=<language>` を追加します。

たとえば、英語で `zman` を実行するには、`zman` スクリプトファイルを以下のように変更します。

```
/opt/novell/zenworks/lib/java/bin/java -Duser.language=en -  
Djava.library.path="${LD_LIB_PATH}" .....
```