



**Hewlett Packard  
Enterprise**

# Smart Update Manager 8.5.0 CLI および対 話式 CLI ガイド

## 摘要

このガイドでは、SUM CLI モード、対話式 CLI モード、および入力ファイルモードを使用して、HPE ProLiant および Integrity サーバーにファームウェアアップデートを適用し、HPE ProLiant、HPE BladeSystem、HPE Synergy、および HPE Apollo のサーバーとインフラストラクチャにシステムソフトウェアアップデートを適用する方法について説明します。このガイドは、Microsoft Windows、Windows Server、Linux、Smart コンポーネント、VMware の構成および操作、ならびにアップデートの実行に伴うデータ消失の危険性について理解している担当者を対象にしています。

部品番号: 814105-594  
発行: 2019 年 12 月  
版数: 1

## ご注意

本書の内容は、将来予告なしに変更されることがあります。Hewlett Packard Enterprise 製品およびサービスに対する保証については、当該製品およびサービスの保証規定書に記載されています。本書のいかなる内容も、新たな保証を追加するものではありません。本書の内容につきましては万全を期しておりますが、本書中の技術的あるいは校正上の誤り、脱落に対して、責任を負いかねますのでご了承ください。

本書で取り扱っているコンピューターソフトウェアは秘密情報であり、その保有、使用、または複製には、Hewlett Packard Enterprise から使用許諾を得る必要があります。FAR 12.211 および 12.212 に従って、商業用コンピューターソフトウェア、コンピューターソフトウェアドキュメンテーション、および商業用製品の技術データ (Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items) は、ベンダー標準の商業用使用許諾のもとで、米国政府に使用許諾が付与されます。

他社の Web サイトへのリンクは、Hewlett Packard Enterprise の Web サイトの外に移動します。Hewlett Packard Enterprise は、Hewlett Packard Enterprise の Web サイト以外の情報を管理する権限を持たず、また責任を負いません。

## 商標

Microsoft<sup>®</sup>および Windows<sup>®</sup>は、米国および/またはその他の国における Microsoft Corporation の登録商標または商標です。

UNIX<sup>®</sup>は、The Open Group の登録商標です。

Linux<sup>®</sup>は、Linus Torvalds の米国およびその他の国における登録商標です。

# 目次

<b>SUM CLI について.....</b>	<b>5</b>
SUM および対話式 CLI コマンドラインオプションについて.....	5
SUM CLI モード.....	5
アップデートのスケジュール.....	6
リモートノードでの Linux の root 認証情報.....	6
Linux の sudo 認証情報を使用するための前提条件.....	6
root 認証情報での SSH キーファイルの使用.....	7
<b>SUM の CLI コマンドと入力ファイルコマンドの概要.....</b>	<b>9</b>
コマンドライン構文.....	9
コマンドラインの説明.....	9
入力ファイルの概要.....	10
入力ファイルのセクション.....	11
<b>SUM CLI コマンドタスクの例.....</b>	<b>13</b>
CLI コマンドの例について.....	13
入力ファイルコマンドの発行.....	13
ローカルホストの更新.....	13
リモートノードの更新.....	14
1つのベースラインでの2つのノードのアップデート.....	15
ノード上のソフトウェアのみの更新.....	15
ノード上のファームウェアのみの更新.....	16
ノードへの最新コンポーネントの展開.....	16
ノード上での旧バージョンのコンポーネントの強制的な展開.....	17
2つのソフトウェアコンポーネントの展開.....	17
sudo を使用したリモートの Linux ノードの更新.....	18
Superdome X パーティションのアップデート.....	18
スーパーユーザーの認証情報を使用したノードの更新.....	19
SSH キーを使用した Linux ノードのアップデート.....	20
<b>CLI のパラメーター.....</b>	<b>22</b>
管理エージェントインストールのパラメーター.....	22
SUM CLI パラメーター.....	23
ヘルプ.....	23
インストールパラメーター.....	23
エラーを無視するパラメーター.....	28
再起動パラメーター.....	31
実行パラメーターのテスト.....	32
ノードパラメーター.....	32
ログファイルのパラメーター.....	40
レポートパラメーター.....	41
入力ファイルのパラメーター.....	43
Superdome X サーバーのコマンド.....	44
リターンコード.....	45
Windows Smart コンポーネントのリターンコード.....	46
Linux Smart コンポーネントのリターンコードとファームウェア RPM のリターンコード.....	46

Linux ソフトウェア RPM のリターンコード.....	47
VMware ESXi Smart コンポーネントのリターンコード.....	47

## 入力ファイルのパラメーターを使用した CLI..... 49

入力ファイルにおけるコンポーネント固有の構成.....	49
入力ファイルでのコンポーネントの構成.....	50
入力ファイルのエンコードについて.....	50
エラーのレポート.....	50
入力ファイルのパラメーター.....	50
SDR からの SUM およびコンポーネントのダウンロード.....	63

## 対話式 CLI モードでのコマンドの実行..... 64

ノードの展開.....	64
activate.....	64
add.....	65
configure.....	69
delete.....	71
deploy.....	71
findavailablenodes.....	73
generatereports.....	74
getattributes.....	76
getbaselines.....	76
getcomponentlogs.....	76
getcurrentlyinstalledversions.....	78
getenginestatus.....	79
getlogs.....	80
Getneededupdates.....	82
getnodes.....	83
inventory.....	85
login.....	86
setattributes.....	88
shutdownengine.....	96
コンポーネントのステージング.....	98
stageonly.....	98
validate baselines.....	98

## Web サイトおよびサポート..... 100

Web サイト.....	100
サポートと他のリソース.....	100
Hewlett Packard Enterprise サポートへのアクセス.....	100
アップデートへのアクセス.....	101
リモートサポート (HPE 通報サービス).....	101
保証情報.....	102
規定に関する情報.....	102
ドキュメントに関するご意見、ご指摘.....	102

# SUM CLI について

## SUM および対話式 CLI コマンドラインオプションについて

SUM の CLI モードと入力ファイルモードを使用して、ローカルおよびリモートのノードにベースラインからのアップデートを展開できます。

CLI モードでは、1 つのコマンドラインにすべてのパラメーターを追加し、ユーザーが操作することなく複数のノードでシーケンス全体を実行します。この方法では、`silent` コマンドが必要です。1 つのノードをアップデートする場合は、この方法を使用します。

入力ファイルモードを使用してすべてのパラメーターを 1 つのテキストファイルに追加し、このテキストファイルの名前を `inputfile` コマンドのパラメーターとして指定して、SUM を呼び出します。この方法では、`silent` コマンドが必要です。複数のノードをアップデートする場合は、この方法を使用します。

SUM の CLI モードと入力ファイルモードは、プロセス指向のツールです。SUM はコマンドを終了した後、その設定や結果を保存しません。いくつかの手順を実行し、その作業を後のために保存する必要がある場合は、GUI または対話式 CLI モードを使用します。

対話式 CLI モードでは、SUM GUI で使用できるほとんどのコマンドをスクリプト化して実行できます。対話式 CLI モードから使用できる機能は、次のとおりです。

- ・ ノードおよびベースラインの追加
- ・ ノードの展開
- ・ ログファイルの収集
- ・ レポートの生成

---

**注記:** アップデートの実行に対話式 CLI を使用しないでください。対話式 CLI のアクセスはサポートされず、SUM の将来のリリースで廃止される予定です。

---

## SUM CLI モード

使用しているサーバー環境に最も適した SUM の CLI モードを選択してください。

### CLI

ノードの数 : 1~3

長所 :

1 つのコマンドでノードに対するすべての指示を与えます。

### 入力ファイル CLI

ノードの数 : 1 - 50

---

**注記:** 10 個の VMware ノードのみを使用できます。

---

長所 :

- ・ ベースラインとノードのすべての情報が含まれているファイルを作成します。
- ・ 後で使用するためにファイルを保存します。
- ・ 入力ファイルを呼び出す CLI コマンドを発行します。

#### 対話式 CLI

ノードの数 : 1 - 50

## アップデートのスケジュール

標準のオペレーティングシステムツールを使用して、CLI コマンドの実行スケジュールを設定します。たとえば、Windows ではタスクスケジューラを使用し、Linux システムでは cron を使用します。必ず、コマンドに `s` (サイレント) パラメーターを含めるようにしてください。タスクのスケジュールについて詳しくは、オペレーティングシステムのドキュメントを参照してください。

---

**注記:** 入力ファイルでは、スケジューリングパラメーターをサポートしています。

---

## リモートノードでの Linux の root 認証情報

root 認証情報を提供するか、sudo 権限のあるユーザーを提供するか、リモートノードで root 以外の認証情報と root 認証情報を提供することができます。

スーパーユーザーの機能を使用するには、ユーザーをすべての root 特権を持つスーパーユーザーとして構成します。また、root ユーザーとともに root 以外のユーザーを使用してコンポーネントをアップデートすることもできます。

## Linux の sudo 認証情報を使用するための前提条件

- ・ 次のいずれかを指定します。
  - ユーザー名とパスワード
  - ユーザー名と SSH キーのファイルパス (PEM 形式)
- ・ sudo ユーザーに `/var/tmp` ディレクトリへの書き込みアクセスを提供します。
- ・ sudo ユーザーについては、`/etc/sudoers` ファイルにユーザーを追加します。以下に、各ユーザーの権限と指定値を示します。
  - ユーザー : `<Sudo_user>` (ユーザーアカウントの実際の名前)
  - 権限 : ALL
  - 仕様 : ALL
- ・ sudo コマンドの実行時にシステムが root ユーザーのパスワードではなく sudo ユーザーのパスワードを求めるように、`/etc/sudoers` ファイルのエントリを編集します。
- ・ 以下に、`/etc/sudoers` でコメント化または削除する権限を示します。
  - ユーザー : ALL
  - 権限 : ALL
  - 仕様 : ALL

---

**注記:** このオプションは必ずすべてのシステム上でデフォルトの `targetpw` を指定して使用してください。

---

- ・ ログイン認証情報の SSH キーを `sudo` と組み合わせて使用するときには、システムがユーザーパスワードを要求しないように、`/etc/sudoers` ファイルのエントリーを次のように編集します。
  - ユーザー : `sudo_user`
  - 権限 : `ALL`
  - 仕様 : `NOPASSWD:ALL`

## root 認証情報での SSH キーファイルの使用

SUM では、パスワードを使用するか、SSH キーを指定することでログインできます。

### 手順

1. まず、SUM を実行しているノードで `ssh-keygen -t rsa` と入力します。SUM は、DSA 暗号キー形式もサポートします。DSA 暗号キーを作成するには、`ssh-keygen -t dsa` と入力します。パスフレーズはオプションです。
2. `.ssh/id_rsa.pub` ディレクトリ内にキーを保存します。root ユーザーの場合は、`/root/.ssh` ディレクトリを使用します。
3. キーを開き、次のテキストがあるかキーの最上部を調べて、秘密キーが PEM 形式であることを確認します。

```
----- BEGIN RSA PRIVATE KEY -----
```
4. 必要に応じて、更新するノードで、ルートレベルに `.ssh` ディレクトリを作成します。ディレクトリでの権限レベルを 700 に設定します。
5. 更新するノードに `.ssh/authorized_keys` ファイルがあるかどうかを確認します。このファイルが存在しない場合は、このファイルを作成または追加します。 `authorized_keys` ファイルに対する権限を 640 に設定します。

---

**注記:** ファイルを追加すれば、より多くのユーザーが秘密キーの使用を承認されます。

---

6. ホストノード上の `.ssh/id_rsa.pub` の内容をコピーし、リモートノード上の `.ssh/authorized_keys` ファイルに貼り付けます。
7. ホストノードと、更新するノードの間の SSH パスを開きます。
  - a. `ssh root@10.0.0.1` と入力します。このパラメーターにより、シェルのパスが開きます。
  - b. `ssh root@10.0.0.1 uname` と入力します。このパラメーターにより、コマンドが実行され結果が戻されます。
8. SUM には、`id_rsa` (秘密キーファイル) へのアクセスと、オプションのパスフレーズが必要です。

---

**注記:** その他のアプリケーションで PEM 形式のパブリックキーが必要な場合は、`ssh-keygen -e -f id_rsa.pub > id_rsa_pub.pem` と入力して変換できます。キーを開き、`---BEGIN SSH2 PUBLIC KEY---`がある PEM 形式であることを確認します。

---

Windows システムで秘密キーを作成するには、PuTTY や PuTTY Key Generator (PuTTYGen) などのアプリケーションを使用します。詳しくは、<https://www.digitalocean.com/community/tutorials/how-to-create-ssh-keys-with-putty-to-connect-to-a-vps> を参照してください。

ファイルを PEM 形式にエクスポートするには、**Conversion > Export OpenSSH key** を選択します。



# SUM の CLI コマンドと入力ファイルコマンドの概要

## コマンドライン構文

SUM の CLI モードでは、1つのコマンドラインにすべてのパラメーターを入力する必要があります。**Enter** キーを押すと、SUM はシーケンス全体を実行します。すべての CLI コマンドの実行には、コマンドに `s` (サイレント) パラメーターが必要で、`/s` (Windows) または `--s` (Linux) が必要です。

❗ **重要:** コマンドラインモードは、ダブルバイト文字セットをサポートしていません。ダブルバイト文字セットを使用してコマンドラインに入力されたコマンドは、正しく表示されません。

SUM の一般的なコマンドライン構文は、以下のとおりです。

Windows では、各引数の前にスラッシュ (`/`) を付けます。

```
smartupdate /s /use_location <baseline_directory>
```

Linux では、各引数の前に2つのハイフン (`--`) を付けます。

```
smartupdate --s --use_location <baseline_directory>
```

**注記:** `/target` と Onboard Administrator (エンクロージャーファームウェア) の IP アドレスの後に OA をアップデートする場合は、ユーザー名と OA 管理者の認証情報を指定する必要があります。ユーザー ID は、管理者相当の ID を使用する必要があり、オペレーターやユーザーレベルの ID は使用できません。

## コマンドラインの説明

### `smartupdate`

SUM を起動します。

### `--s (Linux) /s (Windows)`

CLI モードを開始します。

その他のインストールパラメーターについては、[インストールパラメーター](#)を参照してください。

### `--target (Linux) /target (Windows)`

ターゲットノードの詳細を指定します。

その他のノードパラメーターについては、[ノードパラメーター](#)を参照してください。

### `--on_failed_dependency (Linux) /on_failed_dependency (Windows)`

このパラメーターは、依存関係の問題を処理する方法を SUM に指示します。

エラーを無視するその他のパラメーターについては、[エラーを無視するパラメーター](#)を参照してください

### `--r (Linux) /r (Windows)`

更新を展開した後にノードを再起動するかどうかを SUM に指示します。

その他の再起動パラメーターについては、[再起動パラメーター](#)を参照してください

--v (Linux) /v (Windows)

冗長ログの設定を使用するように SUM に指示します。

その他のログパラメーターについては、[ログファイルのパラメーター](#)を参照してください。

--report (Linux) /report (Windows)

SUM レポートを生成します。

その他のレポートパラメーターについては、[レポートパラメーター](#)を参照してください。

--h (Linux) /h (Windows)

SUM CLI のヘルプファイルを起動します。

## 入力ファイルの概要

入力ファイルには、構成の詳細とターゲットノード情報を含めます。アップデート用のユーザー認証情報とリモートホスト情報を追加できます。

- ❶ **重要:** 入力ファイルはプレーンテキストファイルです。セキュリティを強化するため、認証情報はファイルから省略し、コマンドラインを使用して認証情報を SUM に渡してください。コマンドラインでユーザー認証情報を指定する場合は、入力ファイル内のすべてのノードで同じユーザー ID とパスワードを使用する必要があります。

入力ファイルで SUM CLI を使用すると、ノードをアップデートする .txt ファイルスクリプトを作成できます。入力ファイルを作成するには、プレーンテキストエディターを使用します。これはマークアップ言語と似ており、ヘッダーとトレーラーを一致させる必要があります。リストの値を取ることができるパラメーターには、カンマをリストの区切り文字として使用します。

スクリプトファイルを作成したら、`inputfile <filename>`パラメーターとして SUM コマンドラインにスクリプトファイルを追加します。たとえば入力ファイル `update.in` を実行するには、次のように入力します。

- Linux : `smartupdate --inputfile update.in`
- Windows : `smartupdate /inputfile update.in`

入力ファイルが SUM 実行可能ファイルと同じ場所に保存されていない場合は、そのファイルの場所へのフルパスを追加できます。スペースの入っているパスを指定するには、`<filename>`フィールドを二重引用符で囲みます。

**注記:** ファイアウォールポートを開くための入力ファイルパラメーターはありません。ファイアウォールを開く場合は、パラメーターとして `--open_firewall` (Linux) または `/open_firewall` (Windows) を、入力ファイルを呼び出すコマンドに追加します。以下に例を示します。

- Linux : `smartupdate --inputfile update.in --open_firewall --s`
- Windows : `smartupdate /inputfile update.in /open_firewall /s`

入力ファイルを編集せずに入力ファイルを変更するには、入力ファイルにすべてのコマンドラインオプションを使用します。コマンドラインオプションは、入力ファイル内のコマンドよりも優先されます。

# 入力ファイルのセクション

入力ファイルには、構成およびターゲット情報セクションが含まれています。入力ファイルでは、テキストの1行ごとに1つのパラメーターがサポートされます。

```
#Input file sample
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
SOURCEPATH = C:\SPP
FORCEALL = YES
REBOOTALLOWED = YES
REBOOTDELAY = 30
REBOOTMESSAGE = "Install complete, server will reboot in 30 seconds"
[TARGETS]
HOST = 192.168.1.1
UID = user
PWD = password
[END]
[TARGETS]
HOST = 192.168.1.2
HOST = 192.168.1.2
UID = user2
PWD = password2
[END]
```

## 構成セクション

このセクションでは、入力ファイルのすべてのパラメーターを割り当てます。構成セクションの終了部から TARGETS セクションを開始します。

シャープ (#) 記号で始まる行は注記です。1行に複数のシャープ記号 (#) を使用することはできません。

```
#Input file sample
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
SOURCEPATH = C:\SPP
FORCEALL = YES
REBOOTALLOWED = YES
REBOOTDELAY = 30
REBOOTMESSAGE = "Install complete, server will reboot in 30 seconds"
```

## ターゲットセクション

このセクションは、アップデートする各ノードを定義します。

```
[TARGETS]
HOST = 192.168.1.1
UID = user
PWD = password
[END]
```

## ターゲットセクション

```
[TARGETS_GROUP_HOME]
HOST = 192.168.1.2
HOST = 192.168.1.2
UID = user2
```

```
PWD = password2
```

```
[END]
```

複数のノードで同じユーザー認証情報を使用する場合、同じ TARGETS セクションにすべてのノードをリストします。

[TARGETS] タイトルにテキストを追加して、ターゲットの詳細を示すことができます。

# SUM CLI コマンドタスクの例

## CLI コマンドの例について

次のタスクは、CLI コマンドおよび入力ファイルを構成する方法の例を示します。使用できるパラメーターについて詳しくは、[SUM CLI のパラメーターと入力ファイルのパラメーター](#)を参照してください。

## 入力ファイルコマンドの発行

### 手順

1. コマンドラインウィンドウを開きます。
2. SUM が配置されているディレクトリに移動します。
3. 使用しているオペレーティングシステムに応じて、次のコマンドを入力します。

- ・ Linux : `./smartupdate --inputfile <path:/inputfile.txt>`
- ・ Windows : `smartupdate /inputfile <path:\inputfile.txt>`

---

**注記:** 入力ファイルにパラメーター `SILENT = YES` が含まれていない場合は、CLI に `s` (サイレント) パラメーターを含めます。サイレントパラメーターを指定しない場合、SUM は入力ファイルを処理できません。

---

## ローカルホストの更新

次の例では、ローカルホストにベースライン内のすべての更新を適用します。

### CLI モードで SUM と同じディレクトリにあるベースラインを使用する

- ・ Linux : `./smartupdate --silent`
- ・ Windows : `smartupdate /silent`

### CLI モードで SUM と異なるディレクトリにあるベースラインを使用する

- ・ Linux : `./smartupdate --silent --use_location <directorypath>`
- ・ Windows : `smartupdate /silent /use_location <directorypath>`

### 入力ファイルモードで SUM と同じディレクトリにあるベースラインを使用する

```
SILENT = YES
[TARGETS]
HOST = localhost
UID = <userid>
PWD = <password>
[END]
```

SUM では、Windows ドメイン (たとえば、`domain1/userid1`) がサポートされています。

## 入力ファイルモードで SUM と異なるディレクトリにあるベースラインを使用する

```
SILENT = YES
SOURCEPATH = <baseline_path>
[TARGETS]
HOST = localhost
UID = <userid>
PWD = <password>
[END]
```

## リモートノードの更新

次の例では、リモートノードにベースライン内のすべての更新を適用します。

### CLI モードで SUM を実行しているディレクトリと同じディレクトリからベースラインを使用する

- Linux: `./smartupdate --s --target <ip_address> --targettype <type> --user <userid> --passwd <password>`
- Windows: `smartupdate /s /target <ip_address> /targettype <type> /user <userid> /passwd <password>`

SUM では、Windows ドメイン（たとえば、domain1/userid1）がサポートされています。

### CLI モードで SUM と異なるディレクトリにあるベースラインを使用する

- Linux: `./smartupdate --s --target <ip_address> --targettype <type> --user <userid> --passwd <password> --use_location <directorypath>`
- Windows: `smartupdate /s /target <ip_address> /targettype <type> /user <userid> /passwd <password> /use_location <directorypath>`

SUM では、Windows ドメイン（たとえば、domain1/userid1）がサポートされています。

### 入力ファイルモードで SUM を実行しているディレクトリと同じディレクトリからベースラインを使用する

```
SILENT = YES
ONFAILEDDEPENDENCY = OMITCOMPONENT
TARGETTYPE = <TYPE>
[TARGETS]
HOST = <node_ip_address>
UID = <userid>
PWD = <password>
[END]
```

### 入力ファイルモードで SUM と異なるディレクトリにあるベースラインを使用する

```
SILENT = YES
SOURCEPATH = <baseline_directory>
ONFAILEDDEPENDENCY = OMITCOMPONENT
TARGETTYPE = <TYPE>
[TARGETS]
HOST = <node_ip_address>
UID = <userid>
PWD = <password>
[END]
```

# 1つのベースラインでの2つのノードのアップデート

CLI モードで2つのノードにベースライン内のすべてのアップデートを適用する

- **Linux**: `./smartupdate --s --target <ip_address_1> --targettype <type_1> --user <userid_1> --passwd <password_1> --target <ip_address_2> --targettype <type_2> --user <userid_2> --passwd <password_2>`
- **Windows**: `smartupdate /s /target <ip_address_1> /targettype <type_1> /user <userid_1> /passwd <password_1> /target <ip_address_2> /targettype <type_2> /user <userid_2> /passwd <password_2>`

---

**注記:** これらの例は、SUM を実行しているディレクトリと同じディレクトリにあるベースラインを使用します。別のベースラインを使用する場合は、そのベースラインのディレクトリパスを指定した次のコマンドを含めます。

```
/use_location <baseline_directory_path>
```

---

入力ファイルモードで2つのノードにベースライン内のすべてのアップデートを適用する

```
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY = OMITCOMPONENT
[TARGETS]
HOST = <ipaddress_1>
TARGETTYPE = <type_1>
UID = <userid_1>
PWD = <password_1>
[END]
[TARGETS]
HOST = <ipaddress_2>
TARGETTYPE = <type_2>
UID = <userid_2>
PWD = <password_2>
[END]
```

---

**注記:** 両方のノードで同じユーザー認証情報を使用する場合、それらのノードにはユーザー認証情報を1回だけ指定します。2番目の TARGETS エントリーを作成する代わりに、最初のノードエントリーに2番目のノードの IP アドレスを入力します。以下に例を示します。

```
[TARGETS]
HOST = <ipaddress_1>
HOST = <ipaddress_2>
TARGETTYPE = <type_1>
UID = <userid_1>
PWD = <password_1>
[END]
```

---

## ノード上のソフトウェアのみの更新

これらの例を使用すると、SUM はリモートノード上のソフトウェアのみを更新します。

CLI モードで SUM と同じディレクトリにあるベースラインから1つのノードのソフトウェアをアップデートする

- Linux: `./smartupdate --s --target <ip_address> --user <userid> --passwd <password> --softwareonly`
- Windows: `smartupdate /s /target <ip_address> /user <userid> /passwd <password> /softwareonly`

---

**注記:** 別のベースラインを使用する場合は、次のコマンドを含めます。

- Linux: `--use_location <baseline_directory>`
- Windows: `/use_location <baseline_directory>`

---

### 入力ファイルモードで SUM と同じディレクトリにあるベースラインから 1 つのノードのソフトウェアをアップデートする

```
SILENT = YES
SOFTWAREONLY = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY = OMITCOMPONENT
[TARGETS]
HOST = <ip_address>
UID = <userid>
PWD = <password>
[END]
```

この例は、SUM を実行しているディレクトリと同じディレクトリに配置されているベースラインを使用します。

別のベースラインを使用するには、入力ファイルに以下を追加します。

```
SOURCEPATH = <directory_path>
```

## ノード上のファームウェアのみの更新

ノード上のファームウェアのみを更新するには、ソフトウェアのみを展開した例に以下の変更を加えます。

- CLI モード - コマンド `softwareonly` を削除し、`romonly` を追加します。
- 入力ファイルモード - コマンド `SOFTWAREONLY` を削除し、`ROMONLY` を追加します。

## ノードへの最新コンポーネントの展開

これらの例を使用すると、SUM は、複数のベースラインを含むディレクトリから最新の更新のみを適用します。

### CLI モードでの最新コンポーネントの展開

- Linux: `./smartupdate --s --target <ip_address> --user <userid> --passwd <password>`
- Windows: `smartupdate /s /target <ip_address> /user <userid> /passwd <password>`

### 入力ファイルモードでの最新コンポーネントの展開

```
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
```



```
SKIPTARGET = NO
ONFAILEDDEPENDENCY = OMITCOMPONENT
[TARGETS]
HOST = <ipaddress_1>
TARGETTYPE = <type_1>
UID = <userid_1>
PWD = <password_1>
[END]
```

## ノード上での旧バージョンのコンポーネントの強制的な展開

以下の例では、SUM は、旧バージョンのコンポーネントのソフトウェア展開をノードで強制的に実行します。

### CLI モードでの強制的な展開

- Linux : `./smartupdate --f:bundle --softwareonly --use_location <baseline_directory_path> --s`
- Windows : `smartupdate /f:bundle /softwareonly /use_location <baseline_directory_path> /s`

### 入力ファイルモードでの強制的な展開

```
SILENT = YES
SOFTWAREONLY = YES
FORCEBUNDLE= YES
ONFAILEDDEPENDENCY = OMITCOMPONENT
[TARGETS]
HOST = <ip_address>
UID = <userid>
PWD = <password>
[END]
```

## 2つのソフトウェアコンポーネントの展開

以下の例では、ローカルホストに2つのコンポーネントを展開します。また、コンポーネントがすでにインストールされており、それがソフトウェアコンポーネントである場合、再書き込みまたはダウングレードを強制的に実行します。

---

**注記:** この例では、指定されたコンポーネントがソフトウェアでない場合、SUM はそのコンポーネントを展開しません。

---

### CLI モードでの2つのコンポーネントの展開

以下の例のいずれかを選択します。両方の例でタスクを実行します。

- Linux オプション 1 : `./smartupdate --f:software hponcfg-5.2.0-0.x86_64.rpm hp-smh-templates-10.6.1-1481.4.noarch.rpm --s`
- Linux オプション 2 : `./smartupdate --c hponcfg-5.2.0-0.x86_64.rpm --c hp-smh-templates-10.6.1-1481.4.noarch.rpm --f:software --s`
- Windows オプション 1 : `smartupdate /f:software cp008097.exe cp008257.exe /s`
- Windows オプション 2 : `smartupdate /c cp008097.exe /c cp008257.exe /f:software /s`

## 入力ファイルモードでの2つのコンポーネントの展開

```
SILENT = YES
FORCESOFTWARE = YES
COMPONENTSLIST= cp008097.exe, cp008257.exe
ONFAILEDDEPENDENCY = OMITCOMPONENT
[TARGETS]
HOST = localhost
UID = <userid>
PWD = <password>
[END]
```

## sudo を使用したリモートの Linux ノードの更新

以下の例では、SUM は `userid` を使用してリモートの Linux ノードにログインし、`sudo` 機能を使用してコマンドを実行します。SUM は、SUM を実行しているディレクトリ内のベースラインを使用します。

### sudo を使用した CLI モードでのリモート Linux ノードのアップデート

- Linux: `./smartupdate --target <ip_address> --targettype linux --username <userid> --passwd <password> --use_sudo --use_location <baseline_directory_path> --silent`
- Windows: `smartupdate /target <ip_address> /targettype linux /username <userid> /passwd <password> /use_sudo /use_location <baseline_directory_path> /silent`

### sudo を使用した入力ファイルモードでのリモート Linux ノードのアップデート

```
SILENT = YES
SOURCEPATH = <baseline_path>
ONFAILEDDEPENDENCY = OMITCOMPONENT
TARGETTYPE = LINUX
[TARGETS]
HOST = <ip_address>
USESUDO = YES
UID = <userid>
PWD = <password>
[END]
```

### sudo を使用した CLI モードでの複数のリモート Linux ノードのアップデート

- Linux: `./smartupdate --s --f --target 10.0.1.15 --user sudouser1 --passwd password1 --targettype linux --use_sudo --target 10.0.1.17 --username sudouser2 --passwd password2 --targettype linux --use_sudo --use_location <baseline_directory_path>`
- Windows: `smartupdate /s /f /target 10.0.1.16 /user sudouser1 /passwd password1 /targettype linux /use_sudo /target 10.0.1.17 /username sudouser2 /passwd password2 /targettype linux /use_sudo /use_location C:\SPP\Swpackages`

## Superdome X パーティションのアップデート

以下の例では、SUM は、Superdome X エンクロージャーの `npar3` とブレード `1/1` の各パーティションにアップデートを展開し、パーティションを再起動してアクティブ化します。

### CLI モードでのパーティションのアップデート

- **Linux** : `./smartupdate -silent --target <ip_address> --user <userid> --password <password> --ignore_warnings --update_type PARTITIONS --devicelist npar3,blade1/1 --rebootlist npar3 --use_location <baseline_directory_path>`
- **Windows** : `smartupdate /silent /target 16.17.18.19 /user Administrator /password 12345678 /ignore_warnings /update_type PARTITIONS /devicelist npar3,blade1/1 /rebootlist npar3 /use_location C:\Users\Administrator\Desktop\SDX\3.7.60`

### 入力ファイルモードでのパーティションのアップデート

```
SILENT= YES
SOURCEPATH = <baseline_directory_path>
IGNOREWARNINGS = YES
[TARGET]
HOST = <ip_address>
UID = <userid>
PWD = <password>
UPDATETYPE = PARTITIONS
DEVICELIST=npar3,blade1/1
REBOOTLIST=npar3
[END]
```

## スーパーユーザーの認証情報を使用したノードの更新

以下の例では、SUM は、リモートノードにログインし、スーパーユーザーの認証情報を使用してリモートノードを更新します。

### CLI モードでのスーパーユーザーの認証情報の使用

- **Linux** : `./smartupdate --target <ip_address> --targettype linux --username <userid> --passwd <password> --silent --su_username <su_userid> --su_password <su_password>`
- **Windows** : `smartupdate /target <ip_address> /targettype linux /username <userid> /passwd <password> /silent /su_username <su_userid> /su_password <su_password>`

### 入力ファイルモードでのスーパーユーザーの認証情報の使用

#### 1つのノードのアップデート

```
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY=Omitcomponent
SUUSERNAME = <su_userid>
SUPASSWORD = <su_password>
[TARGETS]
HOST = <ip_address>
UID = <userid>
PWD = <password>
[END]
```

#### 同じスーパーユーザー認証情報を使用する2つのノードの更新

```
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY=Omitcomponent
```

```

SUUSERNAME = <su_userid>
SUPASSWORD = <su_password>
[TARGETS]
HOST = <ip_address_1>
UID = <userid_1>
PWD = <password_1>
[END]
[TARGETS]
HOST = <ip_address_2>
UID = <userid_2>
PWD = <password_2>
[END]

```

### 異なるスーパーユーザー認証情報を使用する2つのノードの更新

```

SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY=Omitcomponent
[TARGETS]
HOST = <ip_address_1>
UID = <userid_1>
PWD = <password_1>
SUUSERNAME = <su_userid_1>
SUPASSWORD = <su_password_1>
[END]
[TARGETS]
HOST = <ip_address_2>
UID = <userid_2>
PWD = <password_2>
SUUSERNAME = <su_userid_2>
SUPASSWORD = <su_password_2>
[END]

```

## SSH キーを使用した Linux ノードのアップデート

次の例では、SUM で SSH キーを使用してログインし、リモート Linux ノードをアップデートすることができません。SUM は、DSA および RSA 暗号化キー形式もサポートします。次のコマンドを実行して、秘密鍵を生成します。

```
ssh-keygen -t -dsa または ssh-keygen -t -rsa
```

### CLI モードでの SSH キーの使用

- Linux : `./smartupdate --target <ip_address> --targettype linux --username <userid> --use_sshkey privatekeyfile=/privatekeyfile/location passphrase=12345 --silent`
- Windows : `smartupdate /target <ip_address> /targettype linux /username <userid> /use_sshkey privatekeyfile=c:\privatekeyfile\location passphrase=12345 /silent`

### 入カファイルモードでの SSH キーの使用

#### 1つのノードのアップデート

```

SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY=Omitcomponent
USE_SSHKEY = YES

```

```
[TARGETS]
HOST = <ip_address>
UID = <userid>
PRIVATEKEYFILE = c:\<keyfile_directory>
PASSPHRASE = <passphrase_string>
[END]
```

### 同じキーファイルと SSHKEY を使用する 2 つのノードのアップデート

```
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY=Omitcomponent
USE_SSHKEY=YES
PRIVATEKEYFILE = c:\<keyfile_directory>
PASSPHRASE = <passphrase_string>
[TARGETS]
HOST = <ip_address_1>
UID = <userid_1>
[END]
[TARGETS]
HOST = <ip_address_2>
UID = <userid_2>
[END]
```

### 異なる SSH キーファイルとパスフレーズを使用する 2 つのノードのアップデート

```
SILENT = YES
IGNOREERRORS = ServerNotFound,BadPassword, FailedDependencies
SKIPTARGET = NO
ONFAILEDDEPENDENCY=Omitcomponent
USE_SSHKEY=YES
[TARGETS]
HOST = <ip_address_1>
UID = <userid_1>
PRIVATEKEYFILE = c:\<keyfile_directory>
PASSPHRASE = <passphrase_string>
[END]
[TARGETS]
HOST = <ip_address_2>
UID = <userid_2>
PRIVATEKEYFILE = c:\<keyfile_directory>
PASSPHRASE = <passphrase_string>
[END]
```

# CLIのパラメーター

## 管理エージェントインストールのパラメーター

管理エージェントがインストールされているかどうかを確認するには、SUM を使用してファームウェア詳細レポートを作成します。以下に例を示します。

- Linux : \$ ./smartupdate --firmware\_report --s
- Windows : C:\ smartupdate /firmware\_report /s

---

**注記:** ノードにすでに管理エージェントがインストールされている場合は、新しいバージョンが公開されている場合は、SUM は管理エージェントをアップグレードします。

たとえば、ノードに SNMP がインストールされている場合に `use_ams` パラメーターを使用すると、AMS がインストールされ、SNMP がアップグレードされます。

---

以下に、特定のコマンドを発行したときに SUM によって展開されるエージェントを示します。

**管理エージェントのパラメーターを指定しない場合 :**

Gen8 以降のサーバーがインストールされたエージェント : AMS

`--no_mgmt (Linux) /no_mgmt (Windows)`

このパラメーターにより、SNMP、AMS、および WBEM エージェントがすべてインストールまたはアップグレードされなくなります。システムにインストールされている管理エージェントには影響がありません。

Gen8 および Gen9 サーバーがインストールされたエージェント : なし

`--use_ams (Linux) /use_ams (Windows)`

Gen8 以降のサーバーがインストールされたエージェント : AMS

`--use_snmp (Linux) /use_snmp (Windows)`

---

**注記:** Gen10 サーバーでは、SNMP エージェントがサポートされていません。

`/use_wmi (Windows のみ)`

Gen8 以降のサーバーがインストールされたエージェント : WBEM

---

**注記:** Gen10 サーバーでは、WBEM エージェントがサポートされていません。

`--use_ams --use_snmp (Linux) /use_ams /use_snmp (Windows)`

Gen8 および Gen9 サーバーがインストールされたエージェント : AMS および SNMP

`/use_ams /use_wmi (Windows)`

Gen8 および Gen9 サーバーがインストールされたエージェント : AMS および WBEM

`/use_wmi /use_snmp (Windows)`

Gen8 および Gen9 サーバーがインストールされたエージェント : WBEM および SNMP

`/use_ams /use_wmi /use_snmp (Windows)`

Gen8 および Gen9 サーバーがインストールされたエージェント : AMS、SNMP、および WBEM

Gen9 のみ : SUM は、SNMP エージェントがインストールされるときに SMH テンプレートをインストールします。

# SUM CLI パラメーター

SUM は、次のコマンドライン引数を認識します。--romonly、--softwareonly など、一部の引数は一緒に使用することはできません。

## ヘルプ

CLI のヘルプを表示するには、`smartupdate --s --h` と入力します。

### 構文

--h[elp] または --? (Linux)

/h[elp] または /? (Windows)

### 説明

コマンドラインのヘルプ情報を表示します。

## インストールパラメーター

次のセクションでは、アップデートのインストール時に使用できる属性について説明します。

## 強制

### 構文

--f[orce] (Linux)

/f[orce] (Windows)

### 説明

現在インストールされているソフトウェアを再書き込みまたはダウングレードします。このパラメーターの役割は、`f:software` と同じです。

### オプション

#### **f[orce]:bundle**

選択されているバンドルに含まれ、現在インストールされているコンポーネントを再書き込みまたはダウングレードします。

#### **f[orce]:rom**

現在インストールされ、適用可能なファームウェアコンポーネントを再書き込みまたはダウングレードします (ファームウェアにのみ適用)。

#### **f[orce]:software**

現在インストールされ、選択されているソフトウェアコンポーネントを再書き込みまたはダウングレードします。

#### **f[orce]:all**

現在インストールされ、適用可能なソフトウェアおよびファームウェアコンポーネントを再書き込みまたはダウングレードします。

## ダウングレード

### 構文

--g or --downgrade (Linux)

/g or /downgrade (Windows)

### 説明

現在インストールされているバージョンよりも低いバージョンを利用できる、インストール対象のコンポーネントを選択します。このコマンドを発行するとアップグレードも行われます。rewrite と組み合わせることができます。

## 再書き込み

### 構文

```
--e or --rewrite (Linux)
/e or /rewrite (Windows)
```

### 説明

インストールされているバージョンよりも低いバージョンを利用できる、インストール対象のコンポーネントを選択します。このコマンドを発行するとアップグレードも行われます。downgrade と組み合わせることができます。

## ROM 展開のみ

### 構文

```
--romonly (Linux)
/romonly (Windows)
```

### 説明

インストール対象とみなされるコンポーネントをファームウェアコンポーネントのみに制限します。softwareonly と一緒に使用することはできません。softwareonly または romonly を使用しない場合、すべてのコンポーネントがインストール対象となる可能性があるかとみなされます。

## ソフトウェア展開のみ

### 構文

```
--softwareonly (Linux)
/softwareonly (Windows)
```

### 説明

インストール対象とみなされるコンポーネントをソフトウェアコンポーネントのみに制限します。romonly と一緒に使用することはできません。softwareonly または romonly を使用しない場合、すべてのコンポーネントがインストール対象であるとみなされます。

## 使用するネットワークポート

### 構文

```
--port <n> (Linux)
/port <n> (Windows)
```

### 説明

SUM が内部 HTTP サーバー用として使用するポートを 63001 から指定のポートに変更します。

## SSL ポート

### 構文

```
--ssl_port <n> (Linux)
/ssl_port <n> (Windows)
```

### 説明



SUM が内部 HTTPS サーバー用として使用するポートを 63002 から指定のポートに変更します。

## ファイアウォールのオープン

### 構文

```
--open_firewall (Linux)  
/open_firewall (Windows)
```

### 説明

SUM に、外部アクセスに SUM で使用される HTTP および HTTPS ポートを開くことを許可します。リモートノード機能およびリモートブラウザアクセスのためにポートを開くときにも使用します。

SUM は、ip\_tables に DROP コマンドが含まれていても、ノードに更新を展開します。open\_firewall を使用すると、DROP ルールはバイパスされます。

入力ファイルを呼び出すコマンドにこのパラメーターを含めます。SUM 入力ファイルは、ファイアウォールを開くパラメーターをサポートしていません。

## サイレント展開

### 構文

```
--s[ilent] (Linux)  
/s[ilent] (Windows)
```

### 説明

GUI や対話式 CLI の出力を伴わないサイレントインストールが行われます。データはすべてログファイルに書き込まれます。express\_install パラメーターは silent パラメーターと組み合わせて使用しないでください。

### 使用方法

SUM では、CLI コマンドにサイレントパラメーターが必要です。

## 展開するコンポーネントの指定

### 構文

```
--c[omponent]<component_to_install>または<component1_to_install>  
<component2_to_install> (Linux)  
/c[omponent]<component_to_install>または<component1_to_install>  
<component2_to_install> (Windows)
```

### 説明

インストールするコンポーネントを指定します。

c[omponent] パラメーターを使用する場合は、パラメーターごとに 1 つの構成要素を指定します。複数の c パラメーターを、同じコマンドラインの個々のコンポーネントに含めることができます。

- ・ --c[omponent] 引数を使用しない場合、複数のコンポーネントをスペースで区切り、コマンドライン上の他のすべての引数の後に指定します。
- ・ デフォルトでは、SUM は、コマンドに記載されている順序でコンポーネントを展開します。SUM は、依存関係に基づいて展開の順序を変更します。

- ・ 複数のコンポーネント (--c[omponent]<component\_to\_install>を参照) とバンドルを同一コマンドラインに指定します。1つのコマンドラインにコンポーネントとバンドルが混在する場合、フィルタースイッチによりインストールされるコンポーネントとバンドルが制御されます。
- ・ SUM は、コンポーネント名のみを使用します。完全なディレクトリパスを入力しても、SUM はパスを無視します。--use\_location および c を使用する場合、SUM は、デフォルトのレポジトリと指定されたディレクトリの両方をチェックします。

5つより多くの構成要素を指定する場合は、複数のコマンドを発行するか、入力ファイルを使用します。

#### 例

```
smartupdate --silent <other_params> --save_install_set --install_set_name Test1
--c[omponent]<component1_to_install> (Linux)

/smartupdate /silent <other_params> /save_install_set /install_set_name Test1 /
c[omponent]<component1_to_install> (Windows)
```

## インストールするバンドルの定義

### 構文

```
--b[undle] <bundlename> (Linux)
/b[undle] <bundlename> (Windows)
```

### 説明

この引数は、インストールするバンドルを指定します。

複数のコンポーネントとバンドル (b[undle] <bundlename>を参照) を同一コマンドラインに指定します。1つのコマンドラインにコンポーネントとバンドルが混在する場合、フィルタースイッチによりインストールされるコンポーネントとバンドルが制御されます。

## ベースラインの場所の定義

### 構文

```
--use_location "[filepath]" (Linux)
/use_location "filepath\"file_share" (Windows)
```

### 説明

SPP およびコンポーネントを含むディレクトリまたはファイル共有が SUM で使用されるように指定します。SUM は、ターゲットの **file\_share** がマッピング済みファイル共有または UNC 形式であることを要求します。

この引数を指定しない場合は、SUM は、SUM を含むディレクトリをデフォルトで使用します。

ログインしたアカウントでこの場所にアクセスできることを確認してください。この場所が UNC ファイル共有である場合は、unc\_username と unc\_password を使用してファイル共有の認証情報を指定する必要があります。この認証情報を指定しない場合、SUM は現在の認証情報を使用して共有にアクセスします。

SUM では、CLI モードでの http ベースラインがサポートされていません。

## コンポーネント構成のインポート

### 構文

```
--import_configuration <directory_path> (Linux)
/import_configuration <directory_path> (Windows)
```

### 説明

このパラメーターは、指定したベースライン内のコンポーネントのコンポーネント構成設定を含むディレクトリを示します。ベースライン内のコンポーネント構成は、インポートしたコンポーネント構成によって上書きされます。

## UNC ユーザー認証情報

### 構文

/unc\_username (Windows)

### 説明

ファイル共有にアクセスするためのユーザー名認証情報。SUM では、Windows ドメイン（たとえば、domain1/userid1）がサポートされています。

## UNC パスワード認証情報

### 構文

/unc\_password (Windows)

### 説明

ファイル共有にアクセスするためのパスワード認証情報。

## 管理エージェントコンポーネントなし

### 構文

--no\_mgmt (Linux)

/no\_mgmt (Windows)

### 説明

AMS、SNMP、および WBEM Provider の管理コンポーネントが自動的に選択されていないことを示します。サイレントモードでは、SUM は、どの管理コンポーネントも更新しません。

管理オプションについて詳しくは、[管理エージェントインストールのパラメーター](#)を参照してください。

## SNMP エージェントコンポーネントの使用

### 構文

--use\_snmp (Linux)

/use\_snmp (Windows)

### 説明

Gen8 および Gen9 サーバー：SUM が自動的に SNMP コンポーネントをインストール対象に選択します。

Gen10 サーバー：SUM は適用外として SNMP コンポーネントを一覧表示します。

### 制限

Linux および Gen9 のみ：SUM が SNMP RPM および smh\_templates RPM を選択します。

## WMI エージェントコンポーネントの使用

### 構文

/use\_wmi

### 説明

SUM が自動的に WBEM コンポーネントをインストール対象に選択します。

### 制限

Windows の場合のみ

## AMS エージェントコンポーネントの使用

### 構文

```
--use_ams (Linux)  
/use_ams (Windows)
```

### 説明

(Gen8 以降のみ) SUM が自動的に AMS コンポーネントをインストール対象に選択します。

## 高速インストール

### 構文

```
--express_install (Linux)  
/express_install (Windows)
```

### 説明

GUI を起動し、Localhost ガイド付きアップデートを自動モードで開始します。このパラメーターは、`silent` パラメーターを使用しているときは無視されます。

## エラーを無視するパラメーター

以下の属性は、SUM がエラーを処理する方法を決定します。

SUM では、これらの属性はコマンドに含まれているすべてのノードに適用されます。

## TPM のバイパス

### 構文

```
--tpmbypass または--ignore_tpm (Linux)  
/tpmbypass または/ignore_tpm (Windows)
```

### 説明

ターゲット iLO 5、Windows、Linux、または VMware ESXi で TPM を有効にした場合は、警告メッセージを無視し、コンポーネントのインストールを続行します。TPM について詳しくは、Smart Update Manager ユーザーガイドを参照してください。

---

**注記:** このパラメーターは、CLI コマンドの末尾で指定します。

---

## 警告の無視

### 構文

```
--ignore_warnings (Linux)  
/ignore_warnings (Windows)
```

### 説明

SUM がノードに関する警告を受けた後でも、インストールの続行が許可されます。一部の警告は、次のとおりです。

- ・ Serviceguard クラスターのアクティブメンバー このオプションを指定せず、ノードが Serviceguard クラスターのアクティブなメンバーである場合、ノードはインストールフェーズまたは展開フェーズに入りません。
- ・ TPM の警告
- ・ Superdome X 固有の警告
- ・ 空きディスク領域の減少
- ・ HPE OneView で管理されるシステム
- ・ 保留中 iLO インストールキューの警告
- ・ 例外タスクキューの警告
- ・ iLO 5 および iSUT の警告

## インストールの続行とエラーの無視

### 構文

```
--continue_on_error <error> (Linux)
/continue_on_error <error> (Windows)
```

### 説明

インストールが継続され、エラーが無視されます。

### オプション

有効な値は、次のとおりです。

#### All

任意の種類のエラーを返すリモートノードをバイパスし、他のノードで展開を続行します。

#### ServerNotFound

ServerNotFound オプションを使用すると、ファームウェアまたはソフトウェアを複数のリモートホストに同時に展開する場合に、アクティブではない、または利用できないリモートホストをバイパスすることができます。

#### BadPassword

BadPassword オプションを使用すると、提供された認証応報が誤っていると報告されたリモートノードをバイパスし、他のノードの処理を続行することができます。

#### FailedDependencies

FailedDependencies オプションを使用すると、失敗した依存関係のあるリモートノードをバイパスし、インストール準備ができて他のノードの処理を続行することができます。このパラメーターは、`on_failed_dependency:Force` または `:OmitComponent` を使用して上書きすることができます。

#### iLOCACEnabled

iLOCACEnabled オプションを使用して、iLO で CAC モードが有効になっているノードをバイパスし、他のノードで続行します。iLO 管理者認証情報は必要ありません。

#### iLOHighSecurityMode

iLOHighSecurityMode オプションを使用して、iLO で高セキュリティモードが有効になっているノードをバイパスし、他のノードで続行します。iLO 管理者認証情報は必要ありません。

### **CHIFSignFailure**

CHIFSignFailure オプションを使用して、CHIF ドライバーコンポーネントの署名検証に失敗したときにノードをバイパスし、他のノードで続行します。

### **CHIFInstallFailure**

CHIFInstallFailure オプションを使用して、CHIF ドライバーのインストールに失敗したときにノードをバイパスし、他のノードで続行します。

## **既存の接続の無効化**

### **構文**

```
--override_existing_connection (Linux)  
/override_existing_connection (Windows)
```

### **説明**

SUM に、進行中のセッションを無効化してリモートノードのインストールフレームワークを再初期化するよう指示します。

このパラメーターを指定しないと、SUM は、SUM セッションがリモートノードですでに実行されている場合に、このノードをスキップします。

## **失敗した依存状態**

### **構文**

```
--on_failed_dependency (Linux)  
/on_failed_dependency (Windows)
```

### **説明**

このパラメーターには、失敗した依存関係が発生したときにコンポーネントを省略するか、ホストへのインストールをスキップするオプションがあり、SUM はインストールを試行しません。

### **オプション**

#### **OmitHost (デフォルト)**

ターゲットが障害状態に設定されるため、インストールは試行されません。

#### **OmitComponent**

影響を受けるコンポーネントの選択が解除され、失敗した依存状態が発生していないアップデートによってプロセスが進められます。

#### **Force**

依存状態に障害があっても、すべてのアップデートが試行されます。

### **サンプル**

```
--on_failed_dependency:OmitHost (Linux)  
/on_failed_dependency:OmitHost (Windows)  
--on_failed_dependency:OmitComponent (Linux)  
/on_failed_dependency:OmitComponent (Windows)  
--on_failed_dependency:Force (Linux)  
/on_failed_dependency:Force (Windows)
```

## 再起動パラメーター

以下の属性は、SUM が再起動を実行する方法を決定します。

### 再起動

#### 構文

`--r[eboot]` (Linux)

`/r[eboot]` (Windows)

#### 説明

サーバー（リモートインストールではホストサーバー）は、次の条件で再起動します。

- ・ `reboot` オプションを選択するか、そのオプションをコマンドライン引数内で使用する場合。
- ・ インストール用に選択されたすべてのコンポーネントが正常にインストールされた場合。
- ・ インストールされたコンポーネントのうち少なくとも1つで、インストールを完了するために再起動が必要な場合。

### 再起動メッセージ

#### 構文

`--reboot_message "reboot message"` (Linux)

`/reboot_message "reboot message"` (Windows)

#### 説明

再起動するサーバーに接続されているリモートコンソールに、指定した再起動メッセージを表示します。このパラメーターは `reboot` オプションと組み合わせて使用してください。そうでない場合は無視されます。

---

**注記:** SLES12 OS イメージには `cli` コマンドとしてパラメーターを実行できないという制約があるため、`reboot_message cli` パラメーターは SLES12 OS では機能しません。

---

### 再起動遅延

#### 構文

`--reboot_delay timeout_in_secs` (Linux)

`/reboot_delay timeout_in_secs` (Windows)

#### 説明

`timeout_in_seconds` 変数で指定された時間だけ、サーバーの再起動を遅らせます。

デフォルトは 60 秒です。

Linux システムでは、再起動遅延の時間は秒単位から分単位に変換され、1 分未満（59 秒以下）の値は切り上げられます。

#### オプション

15~3600 までの値を指定できます。

#### 要件

この引数は `reboot` オプションと組み合わせて使用してください。そうでない場合は無視されます。

## 常に再起動

### 構文

```
--reboot_always (Linux)  
/reboot_always (Windows)
```

### 説明

SUM は、このコマンドラインオプションが渡されると常にサーバーを再起動します。

## 実行パラメーターのテスト

### 構文

```
--dryrun (Linux)  
/dryrun (Windows)
```

### 説明

この属性をその他の属性とともに使用し、実際の展開を除いてコマンドのすべてのプロセスを SUM で実行します。このパラメーターを使用して、リモートノードの接続と構成を確認し、SUM が何を更新するかを確認します。

### 要件

この引数は、テスト実行用にインストールをシミュレーションします。SUM は、どの更新も展開しません。

## ノードパラメーター

以下の属性は、ノードのアップデートオプションを定義します。属性を指定するときに、ターゲットノードを指定してから属性を指定します。ターゲットノードの前に属性を指定しないでください。

以下に例を示します。

有効例：

```
smartupdate /s /target 10.0.1.2 /user root /password root
```

無効例：

```
smartupdate /s /user root /password root /target 10.0.1.2
```

グローバル属性を、属性を指定しないノードに適用できます。属性を適用する 1 つのターゲットノードとともに、コマンドラインの最後にグローバル属性を記述します。以下に例を示します。

```
smartupdate /s /target 10.0.1.3 /target 10.0.1.4 /target 10.0.1.5 /user root /  
password root
```

以下の属性は、グローバルに使用できます。

username/user、password/passwd、use\_location、current\_credential、su\_username、  
su\_password、use\_sshkey、vcenter\_host、vcenter\_username、および vcenter\_password。

## ユーザー名

### 構文

```
--user <username>または--username <username> (Linux)  
/user <username>または/username <username> (Windows)
```

### 説明

ユーザー ID を使用してリモートノードにログインする場合は、この引数を使用します。

### 要件



ユーザー名が root または管理者グループに属していることを確認してください。

SUM では、Windows ドメイン（たとえば、domain1\userid1）がサポートされています。

Windows ドメインの一部である Linux ノードを管理している場合は、次の形式を使用します。

```
<domain>\<username>
```

## ユーザーパスワード

### 構文

```
--password <password> (Linux)
```

```
/password <password> (Windows)
```

smartupdate は、passwd と pwd もサポートします (Windows)

### 説明

user パラメーターで指定したユーザー ID のパスワード。システムはリモートノードへのログインにこのパスワードを使用します。

## iLO ユーザー名

### 構文

```
--ilo_username <username> (Linux)
```

```
/ilo_username <username> (Windows)
```

### 説明

サーバーの iLO が高セキュリティモードで構成されている場合、ホスト OS から iLO に接続するには、このパラメーターを使用します。iLO 管理者の認証情報を入力します。

このパラメーターが有効なのは、Windows、Linux、および不明のノードタイプに対してのみです。

### 要件

iLO 管理者アカウントに **iLO 設定の構成**の許可が付与されていることを確認してください。

## iLO パスワード

### 構文

```
--ilo_password <password> (Linux)
```

```
/ilo_password <password> (Windows)
```

### 説明

ilo\_username パラメーターで指定された iLO ユーザー名に関連付けられたパスワード。

## SSH キーファイルの使用

### 構文

```
--use_sshkey (Linux)
```

```
/use_sshkey (Windows)
```

### 説明

SSH キーを使用して、Linux ノードに接続します。SUM は、他のタイプのノードについては、このパラメーターを無視します。

このパラメーターを指定する場合、SUM はプライベートキーファイルの場所を必要とします。キーファイル (PEM 形式) は、ユーザーが渡すことができるオプションのパラメーターです。

## プライベートキーファイルの場所

### 構文

```
privatekeyfile  
smartupdate /target 10.0.1.2 /user root /use_sshkey privatekeyfile=c:/  
<keyfile_directory/ssh.pem> /r /s
```

### 説明

専用の SSH キーファイルが生成されている場合、SUM によって、パスフレーズが必要であることが自動的に検出され、その入力求められます。

### 要件

`use_sshkey` パラメーターとともに使用します。

## SSH パスフレーズ

### 構文

```
passphrase
```

### 説明

このオプションを使用して、パラメーター `use_sshkey` とともに使用するパスフレーズと、オプションの `privatekeyfile` を指定します。

パスフレーズがキーファイルによって要求されるときに、パラメーターとして指定されていない場合、SUM はパスフレーズの入力を求めます。

```
smartupdate /target 10.0.1.2 /user root use_sshkey privatekeyfile=c:/  
<keyfile_directory/ssh.pem> passphrase=12345 /r /s
```

### 要件

`use_sshkey` パラメーターとともに使用します。

## スーパーユーザー名

### 構文

```
--su_username (Linux)  
/su_username (Windows)
```

### 説明

`username` および `passwd` で指定された認証情報にコンポーネントを更新するための root 権限がない場合、リモートノードで root (スーパーユーザー) ユーザー名を使用してセッションを開始し、コンポーネントのインベントリと更新を実行します。

ローカルホストでは、このコマンドを使用しないでください。

## スーパーユーザーのパスワード

### 構文

```
--su_password (Linux)  
/su_password (Windows)
```

### 説明

`username` および `passwd` で指定された認証情報にコンポーネントを更新するための root 権限がない場合、リモートノードで root (スーパーユーザー) ユーザー名を使用してセッションを開始し、コンポーネントのインベントリと更新を実行します。

ローカルホストでは、このコマンドを使用しないでください。su を使用してローカルホスト上で SUM を起動するには、次の例を使用します。

```
su root  
smartupdate --s --f --use_location /mnt/spp
```

## sudo 認証情報の使用

### 構文

```
--use_sudo (Linux)  
/use_sudo (Windows)
```

### 説明

リモートノードの username と password が sudo ユーザーの認証情報であることを指定します。

### 要件

username および passwd と併用すると、両方のパラメーターが sudo ユーザー認証情報であることを示します。

## ターゲットアドレスの定義

### 構文

```
--target "netAddress" (Linux)  
/target "netAddress" (Windows)
```

### 説明

リモートサーバー、リモートの iLO NIC ポート、c-Class BladeSystem 用の Virtual Connect Ethernet もしくはファイバーチャネルモジュール、または BladeSystem OA となりうるリモートホストの IP アドレスまたは DNS 名です。

2 つの OA がエンクロージャー内にあるときは、アクティブな OA を引数として使用します。IP アドレスを指定する場合は、IPv4 または IPv6 形式を使用します。

VC ノードを追加する場合は、プライマリ Enet モジュールの IP アドレスを使用します。

## ターゲットタイプの定義

### 構文

```
--targettype "type" (Linux)  
/targettype "type" (Windows)
```

### 説明

スクリプト展開のインベントリ時間を短縮します。これはオプションのコマンドライン引数です。

### 設定可能な値

Windows

Linux

iLO

fc\_switch

oa

virtual\_connect - プライマリ VC Enet モジュールの IP アドレスを使用します。

sas\_switch

VMware

sdx

sdflex

targettype は、target パラメーターと一緒に使用してください。targettype と target の順番は入れ替え可能です。ノード名にスペースが含まれる場合は、引用符 “ ” で名前を囲んでください。

## ユーザーの現在の認証情報

### 構文

/current\_credential (Windows のみ)

### 説明

ローカルホストの認証情報を使用してノードにアクセスでき、ノードごとに明示的にユーザー名とパスワードを入力する必要がなくなります。処理しているノードで現在の認証情報が有効であることが前提です (Windows ノードにのみ適用)。

## OA ユーザー名

### 構文

--oa\_username (Linux)

/oa\_username (Windows)

### 説明

target コマンドを使用して指定した VC ターゲットの OA ユーザー名認証情報を指定します。コマンドラインパラメーターを使用すると、1 組の OA 認証情報しか指定できません。target パラメーターを使用して複数の VC ノードを追加するには、指定のそれぞれの VC で OA 認証情報が同一であることを確認してください。VC に関連付けられた OA ネットワークアドレスを提供する必要はありません。SUM は、指定した VC ノードから OA ネットワークアドレスを取得します。

さまざまな認証情報を持つ OA で VC を更新するには、対応する oa\_username および oa\_password を使用します。

## OA ユーザーのパスワード

### 構文

--oa\_password (Linux)

/oa\_password (Windows)

### 説明

target コマンドを使用して指定した VC に関連する OA のパスワード認証情報を提供します。コマンドラインパラメーターには、ただ 1 組の OA 認証情報しか指定できません。指定した VC に関連する OA の認証情報が同じ場合のみ、“node”パラメーターでコマンドラインパラメーターに複数の VC ノードを追加します。VC に関連付けられた OA ネットワークアドレスを提供する必要はありません。SUM は、指定した VC ノードから OA ネットワークアドレスを取り出します。

さまざまな認証情報を持つ OA で VC を更新するには、対応する oa\_username および oa\_password を使用します。

## FIPS モードでの VC の有効化

### 構文

--encryption\_key (Linux)

/encryption\_key (Windows)

### 説明

このパラメーターを使用して、FIPS 対応 VC の暗号化キーを指定します。

### 指定可能な値

最小 8 文字の文字列。

## VC-Enet モジュールの有効化順序

### 構文

--ethactorder (Linux)  
/ethactorder (Windows)

### 説明

VC-Enet モジュールの有効化順序を設定するには、このコマンドを使用します。

### 設定可能な値

ODDEVEN (デフォルト)  
PARALLEL  
SERIAL  
MANUAL

## VC-FC 有効化順序

### 構文

--fcactorder (Linux)  
/fcactorder (Windows)

### 説明

VC-FC の有効化順序を設定するには、このコマンドを使用します。

### 設定可能な値

ODDEVEN  
PARALLEL  
SERIAL (デフォルト)  
MANUAL

## VC-Enet の有効化と reboot オプション

### 構文

--ethactdelay (Linux)  
/ethactdelay (Windows)

### 説明

VC-Enet モジュールの有効化または再起動の間の時間を分単位で設定するには、このコマンドを使用します。

### 設定可能な値

最大で 60 分。デフォルトは 0 分。

## VC-FC の有効化と reboot オプション

### 構文

--fcactdelay (Linux)  
/fcactdelay (Windows)

#### 説明

VC-FC モジュールの有効化または再起動の間に待機する時間を指定するには、このコマンドを使用します。

#### 設定可能な値

最大で 60 分。デフォルトは 0 分。

## VCenter ホスト

#### 構文

--vcenter\_host (Linux)  
/vcenter\_host (Windows)

#### 説明

VMware vCenter の IP アドレスを入力します。

ロックダウンモードになっている VMware vCenter ノードからチケットを生成するには、VMware ESXi ノードでこのコマンドを使用します。このコマンドは、コマンドライン内のすべてのターゲットに適用されます。VMware 以外のノードの場合、このコマンドは無視されます。

## VCenter ユーザー名

#### 構文

--vcenter\_username (Linux)  
/vcenter\_username (Windows)

#### 説明

ユーザー名を入力し、VMware vCenter にログインします。詳しくは、vcenter\_host を参照してください。

## VCenter のパスワード

#### 構文

--vcenter\_password (Linux)  
/vcenter\_password (Windows)

#### 説明

パスワードを入力し、VMware vCenter にログインします。詳しくは、vcenter\_host を参照してください。

## iLO 5 のパラメーター

### インストールセットの保存

#### 構文

--save\_install\_set (Linux)  
/save\_install\_set (Windows)

#### 説明

このパラメーターは、iLO 5 レポジトリにインストールセットを保存します。

### インストールセットの名前

#### 構文

```
--install_set_name (Linux)
/install_set_name (Windows)
```

#### 説明

このパラメーターは、iLO 5 レポジトリに指定された名前を使用してインストールセットを保存します。

#### インストールセットの説明

##### 構文

```
--install_set_description (Linux)
/install_set_description (Windows)
```

#### 説明

このパラメーターは、iLO 5 レポジトリに指定された説明を使用してインストールセットを保存します。

#### 欠落しているコンポーネント署名のスキップ

##### 構文

```
--skip_missing_compsig (Linux)
/skip_missing_compsig (Windows)
```

#### 説明

ノード全体の展開を停止する代わりに、コンポーネントの署名が欠落しているコンポーネントをスキップします。

#### iLO レポジトリの手動管理

##### 構文

```
--manually_manage_ilo_repository (Linux)
/manually_manage_ilo_repository (Windows)
```

#### 説明

自動的に管理する代わりに、手動で iLO レポジトリを管理します。

---

#### 注記:

デフォルトは自動です。つまり、SUM でアップロードされた新しいコンポーネントを収容する領域が iLO NAND がない場合、SUM は（コンポーネントの合計サイズに基づいて）最小サイズのインストールセットを削除し、新しいコンポーネントおよびインストールセット用の領域を作成します。

---

#### 保存された iLO の使用

##### 構文

```
--use_ilo_saved (Linux)
/use_ilo_saved (Windows)
```

#### 説明

ベースラインとして iLO レポジトリに保存されたインストールセットを使用します。SUM は、このインストールセットを iLO NAND で呼び出し、展開時に結果のタスクキューを処理します。

#### iLO のスキップ

##### 構文

```
--skip_ilo (Linux)
```

/skip\_ilo (Windows)

### 説明

このコマンドは、インストールセットから iLO Smart コンポーネントを削除するよう SUM に指示します (EFM のみ)。

## ログファイルのパラメーター

以下の属性は、SUM がログファイルを作成する方法を決定します。

### ログパラメーター

#### ログディレクトリ

##### 構文

--logdir "path" (Linux)

/logdir "path" (Windows)

##### 説明

SUM からの出力をデフォルトディレクトリ以外のディレクトリにリダイレクトします。

Windows システムで実行されている SUM の場合、デフォルト位置は%SYSTEMDRIVE%\CPQSYSTEM\sum\log\

SUM は、コンポーネントを処理するときに%SYSTEMDRIVE%\CPQSYSTEM\ディレクトリを作成します。SUM は、リダイレクトされたディレクトリに smartupdate\_log.txt、smartupdate\_detail\_log.txt、および smartupdate\_InstallDetails.txt ファイルのみを書き込みます。SUM は、その他のログをデフォルトディレクトリに書き込みます。

Linux システムで実行されている SUM の場合、デフォルト位置は/var/log/sum/<netAddress>で、リダイレクト先は<path>/log/sum/<netAddress>です。

#### デバッグログディレクトリ

##### 構文

--debuglogdir [directory\_path] (Linux)

/debuglogdir [directory\_path] (Windows)

##### 説明

このパラメーターは、デバッグログファイルが保存される SUM のディレクトリを割り当てます。

#### 終了時にクリーンアップ

##### 構文

--cleanup\_onexit (Linux)

/cleanup\_onexit (Windows)

##### 説明

このパラメーターは、ターゲットおよびホストから以下のものを削除します。

- ・ Linux フォルダ /usr/lib/i386-linux-gnu/\* または /usr/lib/x86\_64-linux-gnu にコピーされたファームウェア RPM。
- ・ 抽出されたファームウェア RPM ディレクトリとフォルダーの内容。



- ・ ベースラインインベントリの実行中に抽出されたコンポーネント。
- ・ SUM が読み取り専用の場所から起動される場合、このパラメーターは「localsum」ディレクトリを削除します。

---

**注記:** このパラメーターでは、デバッグログファイルは削除されません。

---

### 終了時にすべてのログをクリーンアップ

#### 構文

```
--cleanupall_onexit (Linux)
/cleanupall_onexit (Windows)
```

#### 説明

このパラメーターは、ターゲットおよびホストからすべてのファイル（デバッグファイルを含む）を削除します。このパラメーターでは、ユーザーのログは削除されません。

## 冗長レベルのログ記録

#### 構文

```
--v[erbose]または--veryv[erbose] (Linux)
/v[erbose]または--veryv[erbose] (Windows)
```

#### 説明

SUM 実行ログファイル `sum_execution_log_*.log` の冗長レベルを設定します。ログファイルに維持される詳細レベルを上げることができます。デフォルトは通常の冗長レベルです。

## レポートパラメーター

---

**注記:** 統合レポートとインストール済みレポートは、展開の完了後にのみ生成できます。

---

SUM はコマンドライン引数を使用して、指定されたシステムまたはリポジトリ位置に関するレポートを生成します。コマンドラインにその他の場所を指定しない場合は、SUM でローカルホストとデフォルトのリポジトリ位置（SUM が起動されたディレクトリ）が使用されます。適切な認証情報を提供する場合は、ノードを指定します。他のコマンドラインパラメーターを使用してリポジトリを指定し、レポートを生成します。特定のコマンドについては、[SUM CLI パラメーター](#)を参照してください。

SUM は、JavaScript 対応 Web ブラウザーで表示できる XML または HTML ファイルとしてレポートを生成するか、CSV 形式をサポートする任意のアプリケーションで開くことができる CSV 形式のレポートを生成します。

HTML 形式のレポートは、`SUM_<タイプ>_Report_<日付>_<時刻>` という名前のディレクトリに生成されます。

デフォルトの場所は、SUM が起動された現在のワーキングディレクトリです。この場所が書き込み禁止になっている場合、SUM はレポートを SUM ログファイルと同じディレクトリに保存します。SUM がレポートを保存できる別のディレクトリを指定するには、`reportdir` パラメーターを使用します。

SUM レポートファイルは次のデフォルトのディレクトリにあります。

**Windows :** `C:\cpqsystem\sum\log`

**Linux :** `/var/log/sum`

SUM がレポートを生成するときに、SUM GUI は表示されません。SUM に、生成されたレポートのファイルの場所が表示されます。

## ノード概要レポートの生成

### 構文

```
--report (Linux)
/report (Windows)
```

### 説明

ノードの概要、およびレポジトリ内のコンポーネントがノードに与える影響について説明した展開プレビューレポートを生成します。たとえば、各コンポーネントがノードに適用されるかどうかなどです。レポートは HTML、XML、および CSV 形式で生成され、SUM\_Deploy\_preview\_Report\_<日付>\_<時刻>.html、SUM\_Deploy\_preview\_Report\_<日付>\_<時刻>.xml、および SUM\_Deploy\_preview\_Report\_<日付>\_<時刻>.csv というファイル名が付けられます。

SUM は、インベントリを実行し、レポートを生成してから終了します。どの更新も展開しません。

## インベントリレポジトリレポートの生成

### 構文

```
--inventory_report (Linux)
/inventory_report (Windows)
```

### 説明

指定されたレポジトリ内のコンポーネントをリスト表示するレポートを生成します。レポートは HTML、XML、および CSV 形式で生成され、SUM\_Inventory\_Report\_<日付>\_<時刻>.html、SUM\_Inventory\_Report\_<日付>\_<時刻>.xml、および SUM\_Inventory\_Report\_<日付>\_<時刻>.csv というファイル名が付けられます。

SUM は、インベントリを実行し、レポートを生成してから終了します。どの更新も展開しません。

## インストール済みファームウェアレポートの生成

### 構文

```
--firmware_report (Linux)
/firmware_report (Windows)
```

### 説明

インストール済みファームウェアとソフトウェアのリスト、およびノードの詳細を示すレポートを生成します。レポートは HTML、XML、および CSV 形式で生成され、SUM\_Firmware\_Report\_<日付>\_<時刻>.html、SUM\_Firmware\_Report\_<日付>\_<時刻>.xml、および SUM\_Firmware\_Report\_<日付>\_<時刻>.csv というファイル名が付けられます。

SUM は、インベントリを実行し、レポートを生成してから終了します。どの更新も展開しません。

## 失敗した依存関係のレポートの生成

### 構文

```
--dependency_report (Linux)
/dependency_report (Windows)
```

### 説明

すべてのノードの失敗した依存関係をリスト表示するレポートを生成します。レポートは HTML、XML、および CSV 形式で生成され、SUM\_FailedDependency\_Report\_<日付>\_<時刻>.html、SUM\_FailedDependency\_Report\_<日付>\_<時刻>.xml、および SUM\_FailedDependency\_Report\_<日付>\_<時刻>.csv というファイル名が付けられます。

SUM は、インベントリを実行し、レポートを生成してから終了します。どの更新も展開しません。

## インストール済みアップデートレポートの生成

### 構文

```
--installed_report (Linux)
/installed_report (Windows)
```

### 説明

SUM のこのセッション中に、すべてのノードにインストールされているすべてのファームウェア、ソフトウェア、およびドライババージョンをリスト表示するレポートが生成されます。レポートは HTML、XML、および CSV 形式で生成され、SUM\_Installed\_Report\_<日付>\_<時刻>.html、SUM\_Installed\_Report\_<日付>\_<時刻>.xml、および SUM\_Installed\_Report\_<日付>\_<時刻>.csv というファイル名が付けられます。

SUM は、ノードのインベントリを作成し、アップデートを展開してから、レポートを生成します。

## 結合レポートの生成

### 構文

```
--combined_report (Linux)
/combined_report (Windows)
```

### 説明

このレポートは、1つのファイルにすべてのレポートタイプを生成します。レポートは HTML、XML、および CSV 形式で生成され、SUM\_Combined\_Report\_<日付>\_<時刻>.html、SUM\_Combined\_Report\_<日付>\_<時刻>.xml、および SUM\_Combined\_Report\_<日付>\_<時刻>.csv というファイル名が付けられます。

SUM は、ノードのインベントリを作成し、アップデートを展開してから、レポートを生成します。

## レポートディレクトリ

### 構文

```
--reportdir (Linux)
/reportdir (Windows)
```

### 説明

SUM のレポートの保存先ディレクトリを指定するには、すべてのレポートコマンドでこのパラメーターを使用します。例: smartupdate --report --reportdir /etc/user/reports

## 入力ファイルのパラメーター

複数のノードか個々のノード、またはノードグループのアップデートのスクリプトを作成するには、次のパラメーターを使用します。

## 入力ファイルの使用

### 構文

```
--inputfile "filename" (Linux)
/inputfile "filename" (Windows)
```

### 説明

このパラメーターがファイル名で使用されている場合、SUM は展開を実行するために、ファイルの内容を使用してターゲットおよびベースラインのリストを生成します。詳しくは、[入力ファイルのセクション](#)を参照してください。

## 入力ファイルの削除

### 構文

```
--deleteinputfile (Linux)
/deleteinputfile (Windows)
```

### 説明

SUM に、入力ファイルを読み取り後に削除するように指示します。

## Superdome X サーバーのコマンド

以下の属性は、Superdome X サーバーに適用されます。

## ファームウェアのアップデートタイプ

### 構文

```
--update_type <type> (Linux)
/update_type <type> (Windows)
```

### 説明

どの Superdome X ファームウェアを更新するかを決定します。

### オプション

ALL (デフォルト)  
COMPLEX  
PARTITIONS

### 例

```
update_type PARTITIONS
```

## 特定デバイスのアップデート

### 構文

```
--device_list <device[,device]...> (Linux)
/device_list <device[,device]...> (Windows)
```

### 説明

update\_type 引数が PARTITIONS の場合は、この引数を使います。この引数は、全部のパーティションや未割り当てのブレードを更新したくない場合に、更新するパーティションのサブセットや未割り当てのブレードを指定します。

### オプション

npar (number)  
blade (enc) / (bay)

パラメーターの間にスペースを入れしないでください。

### 例

```
device_list npar1,npar2,blade1/1,blade1/2
```

## パーティション reboot オプション

### 構文

```
--reboot_list <npar[,npar]...> (Linux)  
/reboot_list <npar[,npar]...> (Windows)
```

### 説明

update\_type が ALL または PARTITIONS の場合は、この引数を使います。この引数は、パーティションファームウェアが更新された後に再起動する特定のパーティションを指定します。

### オプション

npar 値は npar (number) です。パラメーターの間にスペースを入れしないでください。

### 例

```
reboot_list npar1,npar2
```

## リターンコード

SUM は Linux と Windows smart コンポーネントのリターンコードを、拡張されたリターンコードマッピングに統合しました。これらのリターンコードは、コンポーネントインストールのステータスを決定します。また、スクリプトのリターンコードを使用して、スクリプトの実行を制御したり、必要な分岐を決定したりできます。

Linux では、負のリターンコードは、256 から値を引くことにより、正のリターンコードに再計算されます。

### SUCCESS\_NO\_REBOOT

**Linux :** 0

**Windows :** 0

**テキスト :** The installation was successful. (インストールに成功しました。)

### SUCCESS\_REBOOT

**Linux :** 1

**Windows :** 1

**テキスト :** The installation was successful, but a reboot is required. (インストールに成功しましたが、再起動が必要です。)

### SUCCESS\_NOT\_REQUIRED

**Linux :** 3

**Windows :** 3

**テキスト :** The component was current or not required. (このコンポーネントは最新であるか、または不要です。)

### FAILURE\_GENERAL

**Linux :** 255

**Windows :** -1

**テキスト :** A general failure occurred. (一般障害が発生しました。) For details, see the error log. (詳しくは、エラーログを参照してください。)

### FAILURE\_BAD\_PARM

**Linux :** 254

**Windows :** -2

テキスト : A bad input parameter was encountered. (不正な入力パラメーターが検出されました。)

#### FAILURE\_COMPONENT\_FAILED

Linux : 253

Windows : -3

テキスト : The installation of the component failed or was blocked by a failed dependency. (コンポーネントのインストールが失敗したか、失敗した依存関係によってブロックされました。)

#### FAILURE\_COMMAND\_FAILED

Linux : 252

Windows : -4

テキスト : The CLI command execution failed. (CLI コマンドの実行に失敗しました。)

## Windows Smart コンポーネントのリターンコード

0

Smart コンポーネントはインストールされませんでした。詳しくは、ログファイルを参照してください。

1

Smart コンポーネントのインストールに成功しました。

2

Smart コンポーネントのインストールは成功しましたが、システムを再起動する必要があります。

3

必要なハードウェアが存在しないか、ソフトウェアが最新のものか、利用可能でないか、インストールするものがないためインストールは試行されませんでした。

## Linux Smart コンポーネントのリターンコードとファームウェア RPM のリターンコード

0

Smart コンポーネントのインストールに成功しました。

1

Smart コンポーネントのインストールは成功しましたが、システムを再起動する必要があります。

2

必要なハードウェアが存在しないか、ソフトウェアが最新のものか、インストールするものがないためインストールは試みられませんでした。

3

必要なハードウェアが存在しないか、ソフトウェアが最新のものか、利用可能でないか、インストールするものがないためインストールは試行されませんでした。

4

このリターンコードは、OA を始めとするネットワークベースの装置などのリモートノードにコンポーネントをインストールする場合に表示され、ノードが見つからないことを示します。

5

インストールが開始する前に、ユーザーによってインストールが取り消されました。

6

依存関係が満たされていないかまたはインストールツールに問題があるため、インストーラーを実行できません。

7

インストールツールに問題はありますが、実際のインストール操作が失敗しました。

## Linux ソフトウェア RPM のリターンコード

0

Linux RPM のインストールに成功しました。

1

Linux RPM のインストールに失敗しました。

## VMware ESXi Smart コンポーネントのリターンコード

0

提供できるコンポーネントのインストールは成功しました。リブートは不要です。

1

提供できるコンポーネントのインストールは成功しました。インストールしたコンポーネントを有効にするために、再起動が必要です。

2

インストール予定のバージョンと既存のバージョンが一致するため、インストールは試みられませんでした。

3

次のいずれかの理由により、インストールは試みられませんでした。

- ・ インストール予定のバージョンが、インストールされているバージョンより古い
- ・ サポートされているハードウェアが存在しないか、有効になっていないか、またはインストールを試みることができない状態にある
- ・ Smart コンポーネントが環境をサポートしない

4

このリターンコードは、OA を始めとするネットワークベースの装置などのリモートノードにコンポーネントをインストールする場合に表示され、ノードが見つからないことを示します。

5

インストールが開始する前に、ユーザーによってインストールが取り消されました。

6

依存関係が満たされていないかまたはインストールツールに問題があるため、インストーラーを実行できません。

7

インストールツールには問題はありませんが、実際のインストール操作が失敗しました。



# 入力ファイルのパラメーターを使用した CLI

## 入力ファイルにおけるコンポーネント固有の構成

入力ファイルのコンポーネントの短い名前を使用して、コンポーネント構成を定義します。コンポーネントのファイル名は、リリースで変更される可能性があります。ファイル名を表示するには、SPP のリリースノートを参照してください。

ベースライン内のコンポーネントで使用可能な構成パラメーターのリストを取得するには、次の対話式 CLI コマンドを使用します。

- ・ Linux : `smartupdate configure --show`
- ・ Windows : `smartupdate configure /show`

---

**注記:** 対話式 CLI はコンポーネントの短い名前をサポートしていません。構成パラメーターを表示するときはコンポーネントの正式名を使用してください。

対話型 CLI の使用方法について詳しくは、Smart Update Manager ユーザーガイドを参照してください。

---

### **hpsmh-windows-x64**

HP System Management Homepage for Windows x64

### **hpinsightmgmtwbemprovider-windows-x64**

HP Insight Management WBEM Provider for Windows Server x64 Edition

### **hpinsightmgmtagent-windows-x64**

HP Insight Management Agents for Windows Server x64 Editions

### **hpmouse-linux**

HP iLO High-Performance Mouse for Linux

### **hpsmh-linux-x64**

HP System Management Homepage for Linux (AMD64/EM64T)

### **hpsnmpagent-rhel6-x64**

HP SNMP Agents for Red Hat Enterprise Linux 6 (AMD64/EM64T)

### **hpsnmpagent-sles11-x64**

HP SNMP Agents for SUSE Linux Enterprise Server 11 (AMD64/EM64T)

### **hpsnmpagent-rhel5-x64**

HP SNMP Agents for Red Hat Enterprise Linux 5 (AMD64/EM64T)

### **hpsnmpagent-rhel7 - x64**

HP SNMP Agents for Red Hat Enterprise Linux 7 (AMD64/EM64T)

### **hpsnmpagent-sles12 - x64**

HP SNMP Agents for SUSE Linux Enterprise Server 12 (AMD64/EM64T)

### **hpqlogic-smartsan-windows-x64**

HPE QLogic SmartSAN Enablement Kit for Windows 64 bit Operating Systems

**hpemulex-smartsan-windows-x64**

HPE Emulex SmartSAN Enablement Kit for Windows 64 bit Operating Systems

**hpemulex-smartsan-linux**

Emulex SmartSAN Enablement Kit for Linux

**hpqlogic-smartsan-linux**

QLogic SmartSAN Enablement Kit for Linux

## 入力ファイルでのコンポーネントの構成

```
[COMPONENT_CONFIG=<component short name>]
[<PARAM_NAME1>]
Value1-line1
Value1-line2
[<PARAM_NAME1>]
Value1-line1
Value1-line2
[END_COMPONENT_CONFIG]
```

## 入力ファイルのエンコードについて

入力ファイルに UTF-8 形式を使用します。UTF-8 では、ダブルバイト文字を使用できます。

## エラーのレポート

入力ファイルでエラーが発生すると、SUM は、-2 という値（不良パラメーター）を返して終了します。エラーの発生場所や性質は、ログファイル `smartupdate_execution_log_<date>_<time>.raw` で調べます。

## 入力ファイルのパラメーター

入力ファイルを使用する場合は、ターゲットノードおよび使用するベースラインを含める必要があります。ベースラインが含まれているディレクトリで SUM を実行する場合、SUM はデフォルトでそのディレクトリにあるベースラインを使用します。

以下の属性はセッションに関連しており、1 回のみ使用できます。

### **BUNDLESLIST**

設定可能な値：

<bundlefilename>

バンドルファイル名

指定されたベースラインの場所に複数のバンドルファイルがある場合に、更新対象として考慮されるバンドル xml ファイルのリスト。

### **CLEANUPALLONEXIT**

設定可能な値：

YES、デフォルト = NO

このパラメーターは、すべてのファイル（デバッグファイルを含む）を削除します。このパラメーターでは、ユーザーのログは削除されません。

#### CLEANUPONEXIT

設定可能な値：

YES、デフォルト = NO

このパラメーターは、以下のものを削除します。

- ・ Linux フォルダ `/usr/lib/i386-linux-gnu/*` または `/usr/lib/x86_64-linux-gnu` にコピーされたファームウェア RPM。
- ・ 抽出されたファームウェア RPM ディレクトリとフォルダーの内容。
- ・ ベースラインインベントリの実行中に抽出されたコンポーネント。
- ・ SUM が読み取り専用の場所から起動される場合、このパラメーターは「localsum」ディレクトリを削除します。

---

**注記:** このパラメーターでは、デバッグログファイルは削除されません。

---

#### COMBINED\_REPORT

設定可能な値：

YES、NO

このレポートは、1つのファイルにすべてのレポートタイプを生成します。

SUM は、インベントリを実行し、更新を展開し、レポートを生成してから終了します。

SUM がレポートを保存する場所については、レポートパラメーターを参照してください。

#### COMPONENTSLIST

設定可能な値：

ファイル拡張子（.exe、.rpm、または.scexe）の付いたコンポーネント名  
更新するコンポーネントのリストを制限します。

#### DEBUGLOGDIR

設定可能な値：

<directorypath>

デバッグファイルを保存する場所を SUM に指示します。

#### DELETEINPUTFILE

設定可能な値：

YES、デフォルト = NO

SUM に、入力ファイルを読み取り後に削除するように指示します。

#### DEPENDENCY\_REPORT

設定可能な値：

YES、NO

すべてのノードの失敗した依存関係をリスト表示するレポートを生成します。

SUM は、インベントリを実行し、レポートを生成してから終了します。更新を展開しません。

SUM がレポートを保存する場所については、レポートパラメーターを参照してください。

#### **DEVICELIST**

設定可能な値：

以下に例を示します。DEVICELIST=npa1,blade1/1

パラメーターの間にスペースを入れないでください。

Superdome X サーバーの場合のみ、UPDATETYPE を PARTITIONS に設定するときこの引数を使用します。この引数は、全部のパーティションを更新したくない場合に、更新するパーティションやブレードを指定します。有効なデバイスは、npa (number) と blade (enc) / (bay) です。

#### **DOWNGRADE**

設定可能な値：

YES、NO

現在インストールされているバージョンよりも低いバージョンを利用できる、インストール対象のコンポーネントを選択します。これは、アップグレードとは別の機能です。rewrite と組み合わせることができます。

#### **DRYRUN**

設定可能な値：

YES、NO

テスト実行用にインストールをシミュレーションします。何もインストールされません。

#### **FIRMWARE\_REPORT**

設定可能な値：

YES、NO

インストール済みファームウェアとインストール済みソフトウェアのリスト、およびノードの詳細を示すレポートを生成します。

SUM は、インベントリを実行し、レポートを生成してから終了します。更新を展開しません。

SUM がレポートを保存する場所については、レポートパラメーターを参照してください。

#### **FORCEALL**

設定可能な値：

YES、NO

現在インストールされ、選択されているソフトウェアコンポーネント、ファームウェアコンポーネント、およびバンドルを再書き込みまたはダウングレードします。

#### **FORCEBUNDLE**

設定可能な値：

YES、デフォルト = NO

選択されているバンドルに含まれ、現在インストールされているコンポーネントを再書き込みまたはダウングレードします。

#### **FORCEROM**

設定可能な値：

YES、NO

現在インストールされ、選択されているファームウェアコンポーネントを再書き込みまたはダウングレードします。

## FORCESOFTWARE

設定可能な値 :

YES、NO

現在インストールされ、選択されているソフトウェアコンポーネントを再書き込みまたはダウングレードします。

## HOST

設定可能な値 :

IP アドレス、DNS 名

リモートサーバー、リモートの iLO NIC ポート、c-Class BladeSystem 用の Virtual Connect Ethernet もしくはファイバーチャネルモジュール、または BladeSystem OA の IP アドレスまたは DNS 名です。

エンクロージャーに 2 つの OA がある場合、このパラメーターはアクティブな OA です。IP アドレスを指定する場合は、IPv4 または IPv6 形式を使用します。

## IGNOREERRORS

このパラメーターにより、インストールが継続され、エラーが無視されます。

設定可能な値 :

All - エラーが発生したリモートノードを無視し、他のノードへの展開を続行します。

ServerNotFound - ファームウェアまたはソフトウェアを複数のリモートホストに同時に展開する場合に、アクティブではない、または利用できないリモートホストをバイパスします。

BadPassword - 提供された認証応報が誤っていると報告されたリモートノードをバイパスし、他のノードの処理を続行します。

FailedDependencies - 失敗した依存状態のあるすべてのノードを無視し、準備ができていないノードによってプロセスが進められます。

ILOCACENABLED - iLO で CAC モードが有効になっているノードをバイパスし、他のノードで続行します。iLO 管理者認証情報は必要ありません。

iLOHighSecurityMode - iLO で高セキュリティモードが有効になっているノードをバイパスし、他のノードで続行します。iLO 管理者認証情報は必要ありません。

CHIFSignFailure - CHIF ドライバーコンポーネントの署名検証に失敗したときにノードをバイパスし、他のノードで続行します。

CHIFInstallFailure - CHIF ドライバーのインストールに失敗したときにノードをバイパスし、他のノードで続行します。

## IGNORETPM

設定可能な値 :

YES、NO

TPM を有効にした場合は、警告メッセージを無視し、コンポーネントのインストールを続行します。TPM について詳しくは、Smart Update Manager ユーザーガイドを参照してください。

## IGNOREWARNINGS

設定可能な値 :

YES、NO

SUM がノードに関する警告を受けた後でも、インストールの続行が許可されます。一部の警告は、次のとおりです。

- ・ Serviceguard クラスターのアクティブメンバー
- ・ TPM の警告
- ・ Superdome X 固有の警告
- ・ 保留中 iLO インストールキューの警告
- ・ iLO タスクキューの例外
- ・ iLO 5 および iSUT の警告

#### IMPORT\_CONFIGURATION=<path>

このパラメーターは、指定したベースライン内のコンポーネントのコンポーネント構成設定を含むディレクトリを示します。ベースライン内のコンポーネント構成は、インポートしたコンポーネント構成によって上書きされます。

#### INSTALLED\_REPORT

設定可能な値：

YES、NO

すべてのノードにインストールされているすべてのファームウェア、ソフトウェア、およびドライババージョンをリスト表示するレポートが生成されます。

SUM は、インベントリを実行し、更新を展開し、レポートを生成してから終了します。

SUM がレポートを保存する場所については、レポートパラメーターを参照してください。

#### INVENTORY\_REPORT

設定可能な値：

YES、NO

指定されたレポジトリ内のコンポーネントをリスト表示するレポートを生成します。

SUM は、インベントリを実行し、レポートを生成してから終了します。更新を展開しません。

SUM がレポートを保存する場所については、レポートパラメーターを参照してください。

#### LOGFILENAME = "path"

設定可能な値：

ログファイル名

SUM または BladeSystem c-Class OA フラッシュユーティリティからの出力をデフォルトディレクトリ以外のディレクトリにリダイレクトします。

Windows コンポーネントの場合、デフォルト位置は%SYSTEMDRIVE%\CPQSYSTEM\sum\log\

SUM は、コンポーネントを処理するときにディレクトリ%SYSTEMDRIVE%\CPQSYSTEM\を作成します。SUM は、リダイレクトされたディレクトリにファイル smartupdate\_log.txt、smartupdate\_detail\_log.txt、および smartupdate\_InstallDetails.txt のみを書き込みます。SUM は、その他のログをすべてデフォルトディレクトリに書き込みます。

Linux コンポーネントの場合は、デフォルト位置は/var/log/sum/<netAddress>で、リダイレクト先は<path>/log/sum/<netAddress>です。

#### NOMGMT

設定可能な値：

YES

SNMP および WBEM Provider を使用するコンポーネントを、コンポーネントの選択画面でオプションのアップデートとして指定します。

サイレントモードでは、SUM は、AMS、SNMP、または WBEM Provider を更新しません。

#### **OAPWD**

ユーザーは、各 HOST 変数の前に OAPWD 変数を複数回定義することができます。

VC に関連する OA のパスワード認証情報を提供します。[TARGETS] セクションの HOST 変数の前に、これらの変数の値を定義します。このパラメーターは VC ターゲットに対してのみ適用されます。

OAPASSWORD を使用することもできます。

#### **OAUID**

ユーザーは、各 HOST 変数の前に OAUID 変数を複数回定義することができます。

VC に関連する OA のユーザー名認証情報を提供します。[TARGETS] セクションの HOST 変数の前に、これらの変数の値を定義します。このパラメーターは VC ファームウェアに対してのみ適用されます。

OAUSER または OAUSERNAME を使用することもできます。

#### **ONFAILEDDEPENDENCY**

失敗した依存状態がコンポーネントに発生した場合のプロセスの進め方を SUM に指示します。

設定可能な値：

OmitHost (デフォルト) - ホストは障害状態に設定され、SUM はインストールを試行しません。

OmitComponent: 影響を受けるコンポーネントの選択が解除され、失敗した依存状態が発生していない更新によってプロセスが進められます。

Force: 失敗した依存状態を含むアップデートも含めすべてのアップデートのインストールが試みられません。

#### **OPTIONS**

設定可能な値：

1 つ以上の CLI スイッチ

入力ファイル内にある、構成設定を上書きする SUM CLI オプションを指定します。カンマを使用してパラメーターを区切ります。

LDU で以前にサポートされた LSPOPTIONS パラメーターに代わるパラメーターです。

#### **REBOOTALLOWED**

設定可能な値：

YES、NO

必要に応じて再起動できます。

#### **REBOOTALWAYS**

設定可能な値：

YES、NO

常にノードを再起動します。

#### **REBOOTDELAY**

設定可能な値：

秒数

再起動する前の待機時間です。

## REBOOTLIST

設定可能な値 :

以下に例を示します。REBOOTLIST=npa1,npa2

パラメーターの間にスペースを入れしないでください。

Superdome X サーバーの場合にのみ、この引数を以下の引数とともに使用します。UPDATETYPE = ALL または UPDATETYPE = PARTITIONS パーティションファームウェアの更新後に再起動するパーティションを指定します。有効な npar 値は npar (number) です。

## REBOOTMESSAGE

設定可能な値 :

任意の文字列 (256 文字以内)

再起動の前に表示するメッセージを作成します。

## REPORT

設定可能な値 :

YES、NO

ノードの概要のレポートリストとレポジトリ内のコンポーネントがノードに与える影響 (たとえば、各コンポーネントがノードに適用されるかどうか) に関する説明を生成します。

SUM は、インベントリを実行し、レポートを生成してから終了します。更新を展開しません。

SUM がレポートを保存する場所については、レポートパラメーターを参照してください。

## REPORTDIR

SUM のレポートの保存先ディレクトリを指定するには、すべてのレポートコマンドでこのパラメーターを使用します。

## REWRITE

設定可能な値 :

YES、デフォルト = NO

インストールされているバージョンと同じバージョンを利用できる、インストール対象のコンポーネントを選択します。これは、アップグレードとは別の機能です。downgrade と組み合わせることができます。

## ROMONLY

設定可能な値 :

YES、NO

インストール対象とみなされるコンポーネントをファームウェアコンポーネントのみに制限します。softwareonly と一緒に使用することはできません。softwareonly および romonly を使用しない場合、すべてのコンポーネントがインストール対象となる可能性があると考えられます。

## SCHEDULEDEPLOYFROM

設定可能な値 :

以下に例を示します。SCHEDULEDEPLOYFROM = MMDDYYYY HH:MM

現在の展開にスケジュール開始時刻を追加します。

## SCHEDULEDEPLOYTO

設定可能な値 :

以下に例を示します。SCHEDULEDEPLOYTO = MMDDYYYY HH:MM

現在の展開にスケジュール終了時刻を追加します。



## SILENT

設定可能な値 :

YES

GUI や対話式 CLI の出力を伴わないサイレントインストールが行われます。データはすべてログファイルに書き込まれます。生成されるプロンプトではすべてデフォルトオプションが使用され、ユーザーの入力なしでインストールが続行されます。

---

**注記:** NO の値を入力すると、入力ファイルが正しく機能しません。SUM は GUI モードを起動しようとしています。

---

## SKIPTARGET

設定可能な値 :

YES (デフォルト)、NO

デフォルト設定では、リモートノードで実行中のリモートセッションを SUM が認識した場合、リモートノードをスキップします。

リモートノードで既存の SUM セッションが進行しているときの動作を定義します。

既存の SUM セッションがある場合にホストをスキップするには、このパラメーターを使用します。NO は、進行中のセッションを無効化し、リモートホストのインストールフレームワークを再初期化します。

## SOFTWAREONLY

設定可能な値 :

YES、NO

インストール対象とみなされるコンポーネントをソフトウェアコンポーネントのみに制限します。`romonly` と一緒に使用することはできません。`softwareonly` または `romonly` を使用しない場合、すべてのコンポーネントがインストール対象となる可能性があるかとみなされます。

## TPMBYPASS

TPM 警告をバイパスするには、このパラメーターを使用します。

## UPDATEEXISTINGRECOVERYSET

設定可能な値 :

YES、NO

iLO リポジトリ内の既存のリカバリセットを、iLO、BIOS、CPLD、IE、および ME (インテルプロセッサの場合、IE と ME) の新しいファームウェアバージョンを使用して、現在の展開からアップデートします。Gen10 サーバーのみ。

## UPDATETYPE

設定可能な値 :

ALL、COMPLEX、PARTITIONS

この引数は、どの Superdome X ファームウェアを更新するかを決定します。

## USE\_SSHKEY

設定可能な値 :

YES、NO

SSH PEM 形式のキーファイルを Linux ノードに使用できます。ノード属性としてではなく、セッション属性として使用する必要があります。

## USEAMS

設定可能な値 :

YES

AMS コンポーネントをインストールするように SUM を指定します。

このオプションは、Gen8 以降のサーバーに適用されます。ProLiant G7 以前のサーバーにこのパラメーターを設定すると、このパラメーターは SUM で無視されます。

デフォルトでは、これらのコンポーネントは Gen8 以降のサーバーのみにインストールされます。このパラメーターは、Integrity サーバーには適用されません。

## USECURRENTCREDENTIAL

設定可能な値 :

YES、NO

ローカルホストの認証情報を使用してノードにアクセスでき、ノードごとに明示的にユーザー名とパスワードを入力する必要がなくなります。

ただし、アクセス対象のノードで現在の認証情報が有効であることが前提です (Windows にのみ適用)。

## USESNMP

設定可能な値 :

YES

SNMP コンポーネントをインストールするように SUM を指定します。

これらのコンポーネントはデフォルトではオプションで、このパラメーターを使用しない場合はインストールされません。このパラメーターは、Integrity サーバーには適用されません。

Gen9 のみ : SUM は、SNMP エージェントがインストールされるときに SMH テンプレートをインストールします。

---

**注記:** Gen10 サーバーでは、SNMP エージェントがサポートされていません。

---

## USEWMI

設定可能な値 :

YES (Windows のみ)

WBEM コンポーネントをインストールするように SUM を指定します。

これらのコンポーネントはデフォルトではオプションで、このパラメーターを使用しない場合はインストールされません。

---

**注記:** Gen10 サーバーでは、WBEM エージェントがサポートされていません。

---

## VERBOSE

設定可能な値 :

YES、NO

このパラメーターは、展開時に SUM が提供する詳細情報の量を決定します。

以下の属性はノードおよびベースラインに関連します。これらの属性をノードまたはベースラインごとに使用します。

## ETHACTDELAY

VC-Enet モジュールの有効化または再起動の間の時間を分単位で設定するには、このコマンドを使用します。

設定可能な値 :

0 (デフォルト) ~60 分。

#### **ETHACTORDER**

VC-Enet モジュールの有効化順序を決めるには、このコマンドを使用します。

設定可能な値 :

ODDEVEN (デフォルト)

PARALLEL

SERIAL

MANUAL

#### **FCACTDELAY**

VC-FC モジュールの有効化または再起動の間に待機する時間を指定するには、このコマンドを使用します。

設定可能な値 :

0 (デフォルト) ~60 分。

#### **FCACTORDER**

VC-FC の有効化順序を指定するには、このコマンドを使用します。

設定可能な値 :

ODDEVEN (デフォルト)

PARALLEL

SERIAL

MANUAL

#### **ILO\_PASSWORD**

設定可能な値 :

<ilo password>

ILO\_USERNAME パラメーターに関連付けられている iLO 管理者のパスワード。

#### **ILO\_USERNAME**

設定可能な値 :

<ilo username>

iLO 管理者の認証情報を入力します。サーバーの iLO が高セキュリティモードで構成されている場合、ホスト OS から iLO に接続するには、このパラメーターを使用します。

#### **INSTALLSETDESCRIPTION**

設定可能な値 :

<string>

インストールセットの説明を設定します。

Gen10 サーバーのみ。

#### **INSTALLSETNAME**

設定可能な値 :

<string>

iLO に保存する必要があるインストールセットの名前を設定します。

Gen10 サーバーのみ。

#### **MANUALLYMANAGEILOREPOSITORY**

設定可能な値 :

YES、NO

YES に設定すると、SUM は新しいインストールセット用の領域を作成するために、iLO レポジトリ上のインストールセットを自動的に削除できません。

Gen10 サーバーのみ。

デフォルトは自動です。つまり、SUM でアップロードされた新しいコンポーネントを収容する領域が iLO NAND がない場合、SUM は (コンポーネントの合計サイズに基づいて) 最小サイズのインストールセットを削除し、新しいコンポーネントおよびインストールセット用の領域を作成します。

#### **PASSPHRASE**

設定可能な値 :

<passphrase=12345>

SSHkey ファイルのパスフレーズ。USE\_SSHKEY および PRIVATEKEYFILE パラメーターとともに使用します。

#### **PRIVATEKEYFILE**

設定可能な値 :

<c:/keyfile\_directory/ssh.pem>

プライベートキーファイルへのフルパスを指定します。

USE\_SSHKEY パラメーターとともに使用します。

#### **PWD**

設定可能な値 :

<password>

UID に指定されたユーザー ID のパスワードを使用します。

ノードにログインするためのパスワードを指定します。

#### **SAVEINSTALLSET**

設定可能な値 : YES、NO

iLO レポジトリにインストールセットを保存します。

Gen10 サーバーのみ。

#### **SKIP\_PREREQS**

設定可能な値 :

YES、NO (Windows のみ)

パラメーター YES を指定すると、前提条件のセルフインベントリコンポーネント (CHIF ドライバーなど) がインストールされません。

#### **SKIPMISSINGCOMPSIG**

設定可能な値 :

YES、NO

YES は、コンポーネントの署名ファイルが欠落しているコンポーネントをスキップするように SUM に指示します。

Gen10 サーバーのみ。

#### SOURCEPATH

設定可能な値：

Directory path、UNC location

単一のローカルのベースラインパスまたは UNC ファイル共有を指定します。この操作により、ローカルまたはデフォルトベースラインの代わりに、指定されたパスからインベントリが作成されます。

入力ファイルモードを使用した SUM CLI では、http ベースラインがサポートされていません。

#### STAGEONLY

設定可能な値：

YES、デフォルト = NO

インストールセットを作成して、適用可能なすべてのコンポーネントを iLO レポジトリにアップロードします。展開を実行するために作成されたインストールセットは開始されません。作成されたインストールセットを使用して、後でシステムを更新できます。

#### SUPASSWORD

設定可能な値：

<superuserpassword>

スーパーユーザーのパスワードを指定するには、この引数を使用します。

Linux ノードにログインしたら、通常の認証情報を使用して、これらの認証情報を使用するセッションを昇格できます。更新を実行するには、SUM に管理者レベルのアクセス権が必要です。

#### SUUSERNAME

設定可能な値：

<superusername>

スーパーユーザーのユーザー名を指定するには、この引数を使用します。

---

**注記:** <superuserpassword>および<superusername>を使用して、スーパーユーザーのユーザー名とパスワードを指定します。Linux ノードにログインしたら、通常の認証情報を使用して、これらの認証情報を使用するセッションを昇格できます。更新を実行するには、SUM に管理者レベルのアクセス権が必要です。

---

#### TARGETTYPE

ノードのタイプ (--targettype Linux (Linux)、/targettype (Windows)) を指定します。これにより、インベントリプロセスを短縮できます。

有効なノードのタイプは次のとおりです。

Windows

Linux

fc\_switch

OA

sdx

iLO

virtual\_connect - プライマリ VC Enet モジュールの IP アドレスを使用します。

`sas_switch`

`VMware`

コマンドでグループを指定すると、SUM は、そのグループ内のすべてのノードが同じノードタイプであると見なします。

#### **UID**

設定可能な値 :

`<username>`

ノードにログインするためのユーザー ID を指定します。

SUM では、Windows ドメイン (たとえば、`domain1/userid1`) がサポートされています。

#### **UNC\_PASSWORD**

設定可能な値 :

`<password>`

UNC の場所にアクセスするためのパスワードを指定します。

#### **UNC\_USERNAME**

設定可能な値 :

`<username>`

SOURCEPATH で指定した UNC の場所のユーザー名認証情報を指定します。

#### **USEILOSAVED**

設定可能な値 :

`<string>`

iLO レポジトリに保存した名前が指定されたインストールセットがノードのベースラインとして使用されます。

Gen10 サーバーのみ。

#### **USESUDO**

設定可能な値 :

YES、NO

`sudo` コマンドを使用できます。

リモートノードの `username` と `password` が `sudo` ユーザーの認証情報であることを指定します。入力ファイルで `username` および `password` とともに `USESUDO` を指定すると、`username` および `password` は `sudo` の認証情報であるとみなされます。

#### **VCENTER\_HOST**

VMware vCenter の IP アドレスを入力します。

ロックダウンモードになっている VMware vCenter サーバーからチケットを生成するには、VMware ESXi ノードでこのコマンドを使用します。

#### **VCENTER\_PASSWORD**

パスワードを入力し、VMware vCenter にログインします。

詳しくは、`VCENTER_HOST` を参照してください。

#### **VCENTER\_USERNAME**

ユーザー名を入力し、VMware vCenter にログインします。

詳しくは、VCENTER\_HOST を参照してください。

## SDR からの SUM およびコンポーネントのダウンロード

Linux システムを使用している場合は、SUM を RPM として SDR からダウンロードできます。一般的な YUM コマンドを使用して、SUM およびコンポーネントを SDR からダウンロードし、インストールすることができます。RPM として SUM をダウンロードし、インストールする方法については、次の URL にある Linux best practices: Using Service Pack for ProLiant (SPP) and Software Delivery Repository (SDR) を参照してください。

<https://www.hpe.com/info/spp/documentation>

SDR の使用については、次を参照してください。

<https://www.hpe.com/servers/sdr>

# 対話式 CLI モードでのコマンドの実行

## ノードの展開

以下の手順は、ノードを展開するための基本的な手順にリンクしています。

### 手順

1. ノードおよびベースラインを追加します。
2. インベントリを実行します。
3. アップデートを展開します。
4. レポートを生成します。

## activate

### 説明

特定のデバイスに基づいて、以前にステージングまたはインストールしたアップデート（再起動、またはその他の処理のため）をアクティブにします。

### 構文

```
smartupdate activate --nodes <nodelist> --cancelpending
```

### パラメーター

**--nodes <nodelist>**

ノードのアクティブ化操作を実行するノードのリスト（完全なホスト名または IP アドレス）。指定しない場合、コマンドはセッションのすべてのノードに適用されます。

### 制限

activate コマンドは、サーバーノードにのみ適用されます。

### Windows での使用法

```
C:\> smartupdate activate /nodes 10.0.0.1 10.0.0.2
```

### Linux での使用法

```
$ ./smartupdate activate --nodes 10.0.0.1 10.0.0.2
```

### SUCCESS\_NO\_REBOOT

**Windows : 0**

**Linux : 0**

**テキスト :** The command was successful. (コマンドが成功しました。)

### SUCCESS\_REBOOT

**Windows : 1**

**Linux : 1**

**テキスト :** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)



#### **SUCCESS\_NOT\_REQUIRED**

**Windows** : 3

**Linux** : 3

**テキスト** : The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows** : -1

**Linux** : 255

**テキスト** : A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows** : -2

**Linux** : 254

**テキスト** : A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows** : -3

**Linux** : 253

**テキスト** : The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows** : -4

**Linux** : 252

**テキスト** : The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## **add**

### **説明**

ベースラインまたはノードを現在のセッションの現在の作業セットに追加します。

### **構文**

```
smartupdate add --nodes
```

```
smartupdate add --baselines
```

### **パラメーター**

/nodes コマンドの直後に指定され、ノードの前に使用されているパラメーターは、グローバルパラメーターと見なされます。これらのパラメーターは、コマンドに指定されているすべてのノードに適用されます。以下に例を示します。

```
smartupdate add /nodes user=user1 password=password1 type=linux 192.168.1.2  
192.168.1.10
```

user、password、および type パラメーターは、両方のノードに適用されます。

指定したノードの後で使用されているパラメーターは、そのノードのみに適用されます。

```
smartupdate add /nodes 192.168.1.15 user=user2 password=password2 type=linux
192.168.1.17 user=user3 password=password3 type=windows
```

user2、password2、および linux パラメーターは、ノード 192.168.1.15 に適用されます。user3、password3、および windows パラメーターは、192.168.1.17 に適用されます。

#### **--nodes <nodeslist>**

追加操作を実行するオブジェクト。リストの項目はスペースで区切ってください。ユーザーとパスワードはすべてのノードでグローバルに指定できます。

```
domain_name: localhost
```

```
ip: 127.0.0.1
```

```
ipv6: ::1
```

ノードまたはベースラインのいずれかを追加します。同じコマンドで両方を追加しないでください。

VC ノードを追加する場合は、プライマリ Enet モジュールの IP アドレスを使用します。FC モジュールを含めて、インストールされているすべての VC モジュールが、プライマリ Enet モジュールを通して更新されます。

#### **type = <node type>**

このパラメーターはノードタイプを定義します。これを、ローカルとグローバルのパラメーターとして使用します。SUM は、次のノードタイプをサポートします。

- ・ windows
- ・ linux
- ・ ilo
- ・ virtual\_connect

プライマリ Enet モジュールの IP アドレスを使用します。VC ノードには、oa\_username および oa\_password が必要です。VC ノードを追加する場合は、oa\_username と oa\_password の属性を組み込みます。

- ・ oa
- ・ vmware
- ・ sas\_switch
- ・ fc\_switch
- ・ sdx
- ・ unknown
- ・ sdflex

#### **skip\_prereqs = <true|false>**

デフォルトでは、Windows ノードに CHIF ドライバーがインストールされます。CHIF ドライバーのインストールをスキップするには、true オプションを使用します。

#### **user=<user\_name>**

```
例 : user=john_doe@example.com
```

これは、ノードのユーザー名認証情報を提供します。これをコマンドラインで入力しない場合、SUM はユーザー名の入力を求めます。

これを、ローカルとグローバルのパラメーターとして使用します。

ノードが Linux ノードであり、Windows ドメインの一部である場合、ユーザー名に次の形式を使用します。

<domain>\<username>

**password=<password>**

例 : password=userpassword

このユーザーパスワードは、ノードの認証に使用されます。これをコマンドラインで入力しない場合、SUM はパスワードの入力を求めます。

これを、ローカルとグローバルのパラメーターとして使用します。

SUM は、ユーザー認証情報について、Windows ドメインアカウントをサポートします。

**ilo\_username=<ilo username>**

サーバーの iLO が高セキュリティモードで構成されている場合、ホスト OS から iLO に接続するには、このパラメーターを使用します。iLO 管理者の認証情報を入力する必要があります。このパラメーターが有効なのは、Windows、Linux、および不明のノードタイプに対してのみです。iLO 管理者アカウントに **iLO 設定の構成**の許可が付与されていることを確認してください。

**ilo\_password=<ilo password>**

ilo\_username パラメーターに関連付けられているユーザーのパスワード。

**su\_user=<super\_userID>**

例 : user=su\_john\_doe@example.com

スーパーユーザーのユーザー名。

**su\_password=<super\_user\_password>**

例 : password=su\_userpassword

スーパーユーザーのパスワード。

**use\_sudo**

有効なオプション : TRUE、FALSE

認証情報を sudo 認証情報として使用します。

指定される可能性があるオプション :

- ・ true
- ・ false

**vcenter\_host**

VMware がロックダウンモードになっている場合の VMware vCenter のアドレス。vcenter\_username および vcenter\_password とともにこれらのパラメーターを使用して、ロックダウンモードの VMware ホストとの接続を確立します。

**vcenter\_username**

VMware ホストとの接続を確立するためのユーザーアカウント。

**vcenter\_password**

VMware ホストとの接続を確立するためのアカウントのパスワード。

**use\_sshkey**

指定される可能性があるオプション :

- ・ true
- ・ false

Linux ノードのみ。このコマンドでは `privatekeyfile` と `passphrase` (オプション) を使用します。

**privatekeyfile=<PEM\_file\_absolute\_path>**

プライベートキーの絶対アドレスを入力します。プライベートキーは、PEM 形式でなければなりません。

これを、ローカルとグローバルのパラメーターとして使用します。

**passphrase=<passphrase>**

(オプション) プライベートキーのパスフレーズを入力します。

これを、ローカルとグローバルのパラメーターとして使用します。

**--baselines <directory\_path>**

このパラメーターを使用してディレクトリを指定することで、ローカルベースラインを追加します。

HTTP ベースラインを追加する場合は、`save_location` パラメーターを使用します。

**save\_location=<working\_location>**

ダウンロードする HTTP ベースラインを保存するディレクトリ。

**type=<baseline\_type>**

このパラメーターはベースラインタイプを定義します。SUM は次のベースラインタイプをサポートしません。

http

unc (Windows のみ)

---

**注記:** ローカルホストシステムで UNC ディレクトリが開いていないことを確認します。

---

**unc\_username=<unc\_username>**

UNC ディレクトリパスのユーザー名を入力します。Windows に対してのみ有効です。

**unc\_password=<unc\_password>**

UNC ディレクトリパスのパスワードを入力します。Windows に対してのみ有効です。

**bundle=<bundle\_name>**

追加するバンドルファイル名を入力します。バンドルファイル名のリストを表示するには、コマンド

`getbaselines /list` を使用します。例: `bundle=spp.2014.02.0.B`

**filter\_os=<os\_filter>**

(HTTP ベースラインのみ) この属性を使用して、SUM がダウンロードするコンポーネントを、オペレーティングシステムを基準にフィルタリングできます。有効なオプションは、次のとおりです。

windows

linux

vmware

## 制限

- ・ 1つのコマンドで、ベースラインとノードを同時に追加しないでください。2つの別々のコマンドを使用します。
- ・ SUM の対話式 CLI モードはノードまたはベースラインを自動的に追加しません。対話式 CLI モードでは、関連ノードが自動的に追加されません。GUI モードを使用して、関連するデバイスを追加します。

## 使用方法

```
smartupdate add [--nodes n1 [type=<node_type>] [user=<username>]
[password=<userpassword>] [su_user=<su_user>] [su_password=<suuser_password>]
[use_sudo=true/false]
--baselines b1 [save_location=<directory>] [type=<baseline_type>]
```

## Windows での使用法

### Windows のサンプルのコマンドライン

```
C:\>smartupdate add /nodes 10.0.1.15 type=windows
C:\>smartupdate add /baselines type=http save_location=c:\temp\spp
bundle=http://10.1.2.3/spp/hp/swpackages/bp002524.xml filter_os=linux
C:\>smartupdate add /nodes type=vc 10.0.1.16 user=Administrator
password=12iso*help oa_username=Administrator oa_password=12iso*help
C:\>smartupdate add /nodes 10.0.1.4 type=vmware vcenter_host=10.0.1.20
vcenter_username=Administrator vcenter_password=password
C:\>smartupdate add /nodes 10.1.2.3 type=virtual_connect
oa_username=Administrator oa_password=mypassword1
```

## Linux での使用法

```
$ ./smartupdate add --nodes 10.0.1.16 type=linux
$ ./smartupdate add --baselines "/SPP/" "/tmp/myrepos/"
$ ./smartupdate add --baselines type=http save_location=c:\temp\spp
bundle=http://10.1.2.3/spp/hpe/swpackages/bp002524.xml filter_os=linux
$ ./smartupdate add --nodes type=vc 10.0.1.18 user=Administrator
password=12iso*help oa_username=Administrator oa_password=12iso*help
$ ./smartupdate add --nodes 10.0.1.4 type=vmware vcenter_host=10.0.1.20
vcenter_username=Administrator vcenter_password=password
$ ./smartupdate add --nodes 10.1.2.3 type=virtual_connect
oa_username=Administrator oa_password=mypassword1
```

**注記:** ノードタイプが Linux の場合は、su\_user および su\_password、または use\_sudo を使用できます。以下に例を示します。

```
smartupdate add /nodes 192.168.1.1 type=Linux user=user password=password
su_user=user1 su_password=password1
smartupdate add /nodes 192.168.1.1 type=Linux user=user password=password
use_sudo=true
```

# configure

## 説明

ベースライン内のコンポーネント構成設定のエクスポート、インポート、表示ができます。

## 構文

```
smartupdate configure --export baseline=<baselinename> dir=<directoryname>
[component=<componentname> includeCredentials=true|false]
```

```
smartupdate configure --import baseline=<baselinename> dir=<directoryname>
[component=<componentname>]
```

```
smartupdate configure --show baseline=<baselinename> [component=<componentname>]
```

## Parameters/Options/Flags/Strings/...

### /export

このパラメーターは、指定されたディレクトリにコンポーネント構成をエクスポートします。

### /import

このパラメーターは、指定されたディレクトリからコンポーネント構成をインポートします。

### /show

このパラメーターは、ベースライン内のコンポーネントのコンポーネント構成オプションを表示します。

### baseline=<baselinename>

インポート、エクスポート、または表示するコンポーネントが含まれるベースラインを指定するには、このパラメーターを指定します。

### dir=<directoryname>

コンポーネント構成設定のエクスポート先またはインポート元のディレクトリを指定するには、このパラメーターを使用します。

### component=<componentname>

これはオプションのパラメーターです。これには、インポート、エクスポート、または表示するコンポーネントの名前を指定します。コンポーネントを指定しない場合、SUM はベースライン内のすべてのコンポーネントをエクスポート、インポート、または表示します。

### includeCredentials=<true|false>

これはオプションのパラメーターです。true を選択した場合、SUM は構成設定をエクスポートするときにユーザー認証情報を含めます。このパラメーターのデフォルトは true です。

## Windows での使用法

```
C:\>smartupdate configure /import baseline=C:\spp.2012.01 dir=C:\Components
```

この例では、C:\spp.2012.01 ベースラインの C:\Components ディレクトリの下にあるすべてのコンポーネント構成が SUM にロードされます。

```
C:\>smartupdate configure /import baseline=C:\spp.2012.01 dir=C:\Components
\cp022423_conf component=cp022423.exe
```

この例では、SUM は、ベースライン C:\spp.2012.01 のコンポーネント cp022423.exe の構成のみをディレクトリ C:\Components にインポートします。

```
C:\>smartupdate configure /show baseline=C:\spp.2012.01
```

この例では、SUM は、ベースライン C:\spp.2012.01 から、すべてのコンポーネント構成を表示します。

## Linux での使用法

```
$ ./smartupdate configure --export baseline=/root/SPP.2012.01 dir=/root/
Components
```

この例では、SUM は、ベースライン /root/SPP.2012.01 から、すべてのコンポーネント構成をディレクトリ /root/Components にエクスポートします。

```
$ ./smartupdate configure --export baseline=/root/SPP.2012.01 dir=/root/  
Components component=hp-snmp-agents-10.10-2657.7.rhel6.i686.rpm  
includeCredentials=false
```

この例では、SUM は、ベースライン `/root/SPP.2012.01` にあるコンポーネント `hp-snmp-agents-10.10-2657.7.rhel6.i686.rpm` の構成のみをディレクトリ `/root/Components` にエクスポートします。この例では、SUM は、認証情報をエクスポートしません。

```
$ ./smartupdate configure --show baseline=/root/SPP.2012.01 component=hp-snmp-  
agents-10.10-2657.7.rhel6.i686.rpm
```

この例では、SUM は、ベースライン `/root/SPP.2012.01` から、コンポーネント `hp-snmp-agents-10.10-2657.7.rhel6.i686.rpm` の構成のみを表示します。

## delete

### 説明

選択されたベースラインまたはノード、および選択されていないベースラインまたはノード（あるいは両方）を現在のセッションから削除できます。iLO レポジトリからインストールセットを削除できます。ノード上で機能を実行しているベースラインまたはノードを削除することはできません。

### 構文

```
smartupdate delete --nodes [nodelist]  
smartupdate delete --baselines [baselinelist]
```

### Parameters/Options/Flags/Strings/...

#### --nodes [nodelist]

削除操作を実行するオブジェクト。リストはスペースで区切ります。ノードリストを指定しない場合、すべてのノードが削除されます。

#### --baselines [baselinelist]

削除操作を実行するオブジェクト。リストはスペースで区切ります。ベースラインリストを指定しない場合、すべてのベースラインが削除されます。

```
--baselines <install_set_name> node=<node IP>
```

iLO レポジトリに保存されているインストールセットを削除します。インストールセット名と iLO 5（または Windows または Linux を実行する Gen10 サーバー）の IP アドレスを提供します。SUM は、ロックされているインストールセットを削除できません。

### 制限

1つのコマンドで、ベースラインとノードを同時に削除しないでください。

### Windows での使用法

```
C:\> smartupdate delete /baselines "spp.2012.01"  
C:\> smartupdate delete /nodes 10.0.0.1
```

### Linux での使用法

```
$ ./smartupdate delete --baselines "spp.2012.01"  
$ ./smartupdate delete --nodes 10.0.0.1
```

## deploy

現在のセッションの現在の作業セットのノードに展開を実行できます。ノードを指定しない場合、デフォルトでは現在選択されているリストのすべてが対象になります。

## 構文

```
smartupdate deploy [--nodes [nodelist]]
```

### パラメーター

#### --nodes [nodelist]

展開操作を実行するノードのリスト。各ノードは次の3つの方法で指定できます。

```
domain_name: localhost
```

```
ip: 127.0.0.1
```

```
ipv6: ::1
```

ノードリストを指定しない場合、セッションのすべてのノードが展開されます。

#### --cleanup\_onexit

以下のファイルを削除します。

- ・ Linux ディレクトリ /usr/lib/i386-linux-gnu または /usr/lib/x86\_64-gnu にコピーされたファームウェア RPM。
- ・ 抽出されたファームウェア RPM ディレクトリとフォルダーの内容。
- ・ ベースラインの抽出されていないコンポーネント。
- ・ SUM が読み取り専用の場所から起動される場合、このパラメーターは localsum ディレクトリを削除します。

#### --cleanupall\_onexit

以下のファイルを削除します。

- ・ Linux ディレクトリ /usr/lib/i386-linux-gnu または /usr/lib/x86\_64-gnu にコピーされたファームウェア RPM。
- ・ 抽出されたファームウェア RPM ディレクトリとフォルダーの内容。
- ・ ベースラインの抽出されていないコンポーネント。
- ・ SUM が読み取り専用の場所から起動される場合、このパラメーターは localsum ディレクトリを削除します。
- ・ デバッグログ。

## 制限

ノードへの更新が存在しない場合、SUM がノードに更新を展開したことを示すことがありますが、何も展開されていません。

### Windows での使用法

```
C:\> smartupdate deploy /nodes 10.7.0.0
```

### Linux での使用法

```
$ ./smartupdate deploy --nodes 10.7.0.0
```

#### SUCCESS\_NO\_REBOOT

```
Windows : 0
```

```
Linux : 0
```

```
テキスト : The command was successful. (コマンドが成功しました。)
```



#### **SUCCESS\_REBOOT**

**Windows :** 1

**Linux :** 1

**テキスト :** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### **SUCCESS\_NOT\_REQUIRED**

**Windows :** 3

**Linux :** 3

**テキスト :** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows :** -1

**Linux :** 255

**テキスト :** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows :** -2

**Linux :** 254

**テキスト :** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows :** -3

**Linux :** 253

**テキスト :** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows :** -4

**Linux :** 252

**テキスト :** The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## **findavailablenodes**

ネットワークで使用可能なノードを検索できます。使用可能にするには、ノードにアクセスできるようにし、別の管理システムがそのノードを排他的に制御できないようにする必要があります。このパラメーターは、指定した範囲の IP アドレスを検索し、オプションのポートを受け入れます。

### **構文**

```
smartupdate findavailablenodes --type ip|port [ipAddrLow ipAddrHi] [port]
```

### **説明**

**Parameters/Options/Flags/Strings/...**

### **--type ip|port**

type オプションを渡す場合、値は次の2つのタイプのいずれかに一致している必要があります。

1. ip - SUM が検索する IP アドレス範囲を入力します。オプション: ipAddrLow; ipAddrHi
2. port - SUM が検索できる IP アドレス範囲とともに、ポート番号を入力します。オプション: ipAddrLow; ipAddrHi; portNum

### **Windows での使用法**

```
C:\>smartupdate findavailablenodes /type ip 10.7.85.0 10.7.85.128
```

### **Linux での使用法**

```
$ ./smartupdate findavailablenodes --type port 10.7.85.0 10.7.85.128 445
```

## **generatereports**

SUM セッションのレポートを生成できます。このレポートには、インベントリ、ファームウェア、失敗した依存関係、展開プレビュー、インストールされているコンポーネント、および統合レポートが含まれます。出力はデフォルトで `attributes values` になり、レポートのタイプは `all available` になります。

### **構文**

```
smartupdate generatereports [--type inventory | firmware | installables | faileddep | installed | combined] [--output html | xml | csv] [--nodes [nodelist]] [--baselines [baselinelist]]
```

### **Parameters/Options/Flags/Strings/...**

#### **--type**

type パラメーターを指定する場合は、次のいずれかである必要があります。

- ・ Inventory
- ・ Firmware
- ・ Installables
- ・ Faileddep
- ・ Installed
- ・ Combined

タイプを指定しない場合は、使用可能なすべてのタイプが生成されます。

#### **--output**

- ・ xml
- ・ csv
- ・ html

出力タイプを指定しない場合は、すべての形式が生成されます。

#### **--nodes [nodelist]**

SUM は、リスト内のすべてのノードの詳細情報を含む単一のレポートを生成します。ノードリストを指定しない場合、現在のセッションのすべてのノードが選択されます。

**--baselines [baselinelist]**

レポートに含めるベースラインのリスト。ベースラインリストを指定しない場合、現在のセッションのすべてのベースラインが選択されます。

### Windows での使用法

```
C:\>smartupdate generatereports /type inventory /nodes localhost
```

### Linux での使用法

```
$ ./smartupdate generatereports --type faileddep --nodes 10.7.85.0 10.7.85.128
```

#### SUCCESS\_NO\_REBOOT

**Windows : 0**

**Linux : 0**

**テキスト :** The command was successful. (コマンドが成功しました。)

#### SUCCESS\_REBOOT

**Windows : 1**

**Linux : 1**

**テキスト :** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### SUCCESS\_NOT\_REQUIRED

**Windows : 3**

**Linux : 3**

**テキスト :** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### FAILURE\_GENERAL

**Windows : -1**

**Linux : 255**

**テキスト :** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### FAILURE\_BAD\_PARM

**Windows : -2**

**Linux : 254**

**テキスト :** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### FAILURE\_COMPONENT\_FAILED

**Windows : -3**

**Linux : 253**

**テキスト :** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### FAILURE\_COMMAND\_FAILED

**Windows : -4**

**Linux : 252**

**テキスト:** The command failed. (コマンドが失敗しました。) For details, see the logs.  
(詳細は、ログを参照してください。)

## getattributes

セッションまたは選択したノードの属性のいずれかを表示できます。属性は SetAttributes コマンドを使用して変更できます。

### 構文

```
smartupdate getattributes [--nodes [nodelist]] [--session]
```

### Parameters/Options/Flags/Strings/...

**--nodes** [<nodelist>]

属性を取得するノードのリスト。ノードとセッション属性の両方が指定されていない場合、このコマンドはすべてのノードおよびセッションの属性を返します。

**--session**

SUM セッションで使用可能な属性を一覧表示します。

### Windows での使用法

```
C:\> smartupdate getattributes /nodes localhost
```

```
C:\> smartupdate getattributes /nodes 10.0.1.2
```

### Linux での使用法

```
$ ./smartupdate getattributes --nodes localhost
```

```
$ ./smartupdate getattributes --nodes 10.0.1.4
```

## getbaselines

現在の SUM セッションのベースラインのリストを表示できます。

### 構文

```
smartupdate getbaselines [<baselinelist --details>] [nodeip]
```

### パラメーター

**baselinelist** [--details]

この操作の実行に使用するベースラインの場所。詳細なベースライン情報（コンポーネントとそれらのバージョン値のリストを含む）を取得するには、--details を指定します。

**nodeip** <ipaddress>

ノード上のインストールセットをリストします。(Gen10 iLO 5 ノードのみ)

### Windows での使用法

```
C:\> smartupdate getbaselines c:\spp\baseline1 /details
```

### Linux での使用法

```
$ ./smartupdate getbaselines
```

## getcomponentlogs

ノードのコンポーネントログを表示できます。

## 構文

```
smartupdate getcomponentlogs [--nodes [<nodelist>]] [--component  
<componentname>]
```

## パラメーター

**--nodes** [<nodelist>]

この要求の対象となるノード。ノードリストを指定しない場合、セッションのすべてのノードが使用されます。

**--component** <componentname>

SUM がログを表示するコンポーネントの名前。このパラメーターを使用する場合は、コンポーネント名を入力する必要があります。コンポーネントを指定しない場合、SUM はすべてのコンポーネントのログを生成します。

## Windows での使用法

```
C:\> smartupdate getcomponentlogs /nodes localhost
```

## Linux での使用法

```
$ ./smartupdate getcomponentlogs --nodes 10.0.1.1 --component cp020024.exe
```

### SUCCESS\_NO\_REBOOT

**Windows** : 0

**Linux** : 0

**テキスト**: The command was successful. (コマンドが成功しました。)

### SUCCESS\_REBOOT

**Windows** : 1

**Linux** : 1

**テキスト**: The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

### SUCCESS\_NOT\_REQUIRED

**Windows** : 3

**Linux** : 3

**テキスト**: The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

### FAILURE\_GENERAL

**Windows** : -1

**Linux** : 255

**テキスト**: A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

### FAILURE\_BAD\_PARM

**Windows** : -2

**Linux** : 254

**テキスト**: A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### FAILURE\_COMPONENT\_FAILED

**Windows** : -3

**Linux** : 253

**テキスト** : The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### FAILURE\_COMMAND\_FAILED

**Windows** : -4

**Linux** : 252

**テキスト** : The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## getcurrentlyinstalledversions

現在の SUM セッション内でのノードインベントリ後に、ノードに現在インストールされているコンポーネントのバージョンを取得できます。

### 構文

```
smartupdate getcurrentlyinstalledversions [--nodes [<nodelist>]]
```

### パラメーター

**--nodes** [<nodelist>]

インストールされているバージョンを取得するノードのリスト。ノードリストを指定しない場合、セッションのすべてのノードが使用されます。

### Windows での使用法

```
C:\> smartupdate getcurrentlyinstalledversions /nodes localhost
```

### Linux での使用法

```
$ ./smartupdate getcurrentlyinstalledversions --nodes localhost
```

#### SUCCESS\_NO\_REBOOT

**Windows** : 0

**Linux** : 0

**テキスト** : The command was successful. (コマンドが成功しました。)

#### SUCCESS\_REBOOT

**Windows** : 1

**Linux** : 1

**テキスト** : The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### SUCCESS\_NOT\_REQUIRED

**Windows** : 3

**Linux** : 3

**テキスト** : The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows** : -1

**Linux** : 255

**テキスト** : A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows** : -2

**Linux** : 254

**テキスト** : A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows** : -3

**Linux** : 253

**テキスト** : The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows** : -4

**Linux** : 252

**テキスト** : The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## **getenginestatus**

SUM セッションのステータス情報を取得できます。

### **構文**

```
smartupdate getenginestatus
```

### **Windows での使用法**

```
C:/>smartupdate getenginestatus
```

### **Linux での使用法**

```
$ ./smartupdate getenginestatus
```

### **出力例**

```
No SUM engine found running.
```

### **出力例**

```
Process name:
```

```
Copyright:
```

```
Version:
```

```
Active Sessions:
```

```
Engine status:
```

```
Total number of sessions:
```

#### **SUCCESS\_NO\_REBOOT**

**Windows** : 0

**Linux** : 0

**テキスト** : The command was successful. (コマンドが成功しました。)

#### **SUCCESS\_REBOOT**

**Windows** : 1

**Linux** : 1

**テキスト** : The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### **SUCCESS\_NOT\_REQUIRED**

**Windows** : 3

**Linux** : 3

**テキスト** : The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows** : -1

**Linux** : 255

**テキスト** : A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows** : -2

**Linux** : 254

**テキスト** : A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows** : -3

**Linux** : 253

**テキスト** : The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows** : -4

**Linux** : 252

**テキスト** : The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## getlogs

すべてのログを収集して圧縮ファイルに格納する、SUM の GatherLogs 関数を起動します。デフォルトでは、SUM はログファイルを次のいずれかの場所に保存します。

- ・ Windows



- C:\cpqsystem\log
  - C:\cpqsystem\sum\log
  - C:\Windows\Temp\SUM
  - %localappdata%\sum
- ・ Linux
    - /var/log/sum
    - /var/cpq
    - /var/tmp/SUM

## オプション

### **debuglogdir <directorypath>**

デフォルトディレクトリ以外のディレクトリにあるログファイルを検索するように SUM に指示します。

### **logdir <directorypath>**

デフォルトディレクトリ以外のディレクトリにあるログファイルを検索するように SUM に指示します。

## 構文

```
smartupdate getlogs
```

### Windows での使用法

```
C:\> smartupdate getlogs
```

### Linux での使用法

```
$ ./smartupdate getlogs
```

### **SUCCESS\_NO\_REBOOT**

**Windows :** 0

**Linux :** 0

**テキスト:** The command was successful. (コマンドが成功しました。)

### **SUCCESS\_REBOOT**

**Windows :** 1

**Linux :** 1

**テキスト:** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

### **SUCCESS\_NOT\_REQUIRED**

**Windows :** 3

**Linux :** 3

**テキスト:** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

### **FAILURE\_GENERAL**

**Windows :** -1

**Linux :** 255

**テキスト:** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows :** -2

**Linux :** 254

**テキスト:** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows :** -3

**Linux :** 253

**テキスト:** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows :** -4

**Linux :** 252

**テキスト:** The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## Getneededupdates

ノードインベントリ後にコマンドの一部として渡される現在の選択または項目のいずれかの状態に基づいて、必要なアップデートの詳細を表示できます。ノードを指定しない場合、コマンドはセッションで現在選択されているノードを使用します。

### 構文

```
smartupdate getneededupdates [--nodes [<nodelist>]]
```

### 説明

#### パラメーター

**--nodes** [<nodelist>]

必要なアップデート操作のチェックを実行するノードのリスト。ノードリストを指定しない場合、セッションのすべてのノードが使用されます。

### Windows での使用法

```
C:\> smartupdate getneededupdates /nodes localhost
```

### Linux での使用法

```
$ ./smartupdate getneededupdates --nodes localhost
```

#### **SUCCESS\_NO\_REBOOT**

**Windows :** 0

**Linux :** 0

**テキスト:** The command was successful. (コマンドが成功しました。)

#### **SUCCESS\_REBOOT**

**Windows :** 1

**Linux : 1**

**テキスト:** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

**SUCCESS\_NOT\_REQUIRED**

**Windows : 3**

**Linux : 3**

**テキスト:** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

**FAILURE\_GENERAL**

**Windows : -1**

**Linux : 255**

**テキスト:** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

**FAILURE\_BAD\_PARM**

**Windows : -2**

**Linux : 254**

**テキスト:** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

**FAILURE\_COMPONENT\_FAILED**

**Windows : -3**

**Linux : 253**

**テキスト:** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

**FAILURE\_COMMAND\_FAILED**

**Windows : -4**

**Linux : 252**

**テキスト:** The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## getnodes

現在の SUM セッションのノードのリストを表示できます。Gen10 および iLO 5 ノードでは、iLO レポジトリのインストールセット、各インストールセットのコンポーネント、および保留中の iLO タスクキューアイテムもリストされます。

### 構文

```
smartupdate getnodes [<nodelist> --details]
```

### パラメーター

--<nodelist>

操作を実行するノード。

## --details

[--details]を指定すると、詳細情報を取得できます。

## 制限

### Windows での使用法

```
C:\>smartupdate getnodes 10.0.1.1 /details
```

### Linux での使用法

```
$ ./smartupdate getnodes 10.0.1.1 --details
```

#### SUCCESS\_NO\_REBOOT

**Windows : 0**

**Linux : 0**

**テキスト :** The command was successful. (コマンドが成功しました。)

#### SUCCESS\_REBOOT

**Windows : 1**

**Linux : 1**

**テキスト :** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### SUCCESS\_NOT\_REQUIRED

**Windows : 3**

**Linux : 3**

**テキスト :** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### FAILURE\_GENERAL

**Windows : -1**

**Linux : 255**

**テキスト :** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### FAILURE\_BAD\_PARM

**Windows : -2**

**Linux : 254**

**テキスト :** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### FAILURE\_COMPONENT\_FAILED

**Windows : -3**

**Linux : 253**

**テキスト :** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### FAILURE\_COMMAND\_FAILED

**Windows : -4**

**Linux : 252**

**テキスト:** The command failed. (コマンドが失敗しました。) For details, see the logs.  
(詳細は、ログを参照してください。)

## inventory

ノードをスキャンして、ファームウェア、ドライバー、およびハードウェアコンポーネントの現在インストールされているバージョンを特定できます。

### 構文

```
smartupdate inventory [--nodes <nodelist>] [--baselines <baselinelist>]
```

### パラメーター

#### --nodes [<nodelist>]

inventory コマンドを実行するノードのリスト。各ノードは次の方法で指定できます。

- ・ localhost
- ・ 127.0.0.1
- ・ ::1

#### --baselines [<baselinelist or install\_set\_name>]

SPP、コンポーネント、または iLO に保存されたインストールセットのパスを指定します。ベースラインの場所にスペースや特殊文字が含まれる場合は、エスケープ文字 (Linux) または二重引用符 (Windows) を使用します。

iLO レポジトリに保存されたインストールセットを使用するには、ノードの IP アドレスまたは DNS 名とベースラインインストールセットの名前を含めます。

ベースラインにノードリストを含める場合、SUM は、指定されたベースラインを持つノードでインベントリを実行します。ノードリストを含めない場合、SUM はベースラインでのみインベントリを実行します。

### 制限

インベントリを実行する前に、setattributes コマンドを使用して管理オプション (use\_ams、use\_wmi、use\_snmp、no\_mgmt) を設定します。

暗号化キーを使用して VC ノード上でインベントリが正常に完了したら、setattributes コマンドを使用して、encryption\_key パラメーターの詳細を割り当てます。

### Windows での使用法

```
C:\>smartupdate inventory /nodes 10.0.1.1 /baselines C:\Users\Administrator\Desktop\blds\ilo\26.33
```

```
C:\>smartupdate inventory /nodes 10.0.1.1 /baseline <install_set_name>
```

### Linux での使用法

```
$ ./smartupdate inventory --nodes 10.0.1.1 --baselines /root/Desktop/rono/blds/ilo
```

### SUCCESS\_NO\_REBOOT

**Windows :** 0

**Linux :** 0

**テキスト:** The command was successful. (コマンドが成功しました。)

#### **SUCCESS\_REBOOT**

**Windows** : 1

**Linux** : 1

**テキスト** : The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### **SUCCESS\_NOT\_REQUIRED**

**Windows** : 3

**Linux** : 3

**テキスト** : The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows** : -1

**Linux** : 255

**テキスト** : A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows** : -2

**Linux** : 254

**テキスト** : A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows** : -3

**Linux** : 253

**テキスト** : The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows** : -4

**Linux** : 252

**テキスト** : The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## login

現在のセッションに使用する認証情報を設定できます。

### 構文

認証情報を使用して初めて認証する :

```
smartupdate login [--username <username> --password <password>] [--port <port>]
[--ssl_port <sslport>] [--ftp_port <ftpport>] [--open_firewall]
```

現在の認証情報で認証する :

```
smartupdate login
```

## パラメーター

### --username

ユーザー名を追加するためのオプションパラメーター。

### --password

ユーザーパスワードを追加するためのオプションパラメーター。

### --port

SUM が使用するポートを定義するためのオプションパラメーター。

指定できる値は、次のとおりです：

80、1024～65535

### --ssl\_port

SUM が使用する SSL ポートを定義するためのオプションパラメーター。

指定できる値は、次のとおりです：

443、1024～65535

### --ftp\_port

SUM が使用する FTP ポートを定義するためのオプションパラメーター。FC スイッチをアップデートする場合は、このパラメーターを指定します。

指定できる値は、次のとおりです：

disabled、21、1024～65535

### --debuglogdir <filepath>

ログファイルを保存する場所を SUM に指示します。ディレクトリパスを入力します。

### --open\_firewall

SUM は、すべてのファイアウォールインスタンスを開くことができるわけではありません。SUM がファイアウォールを開くことができない場合は、そのファイアウォールを手動で開きます。

## Windows での使用法

```
C:\> smartupdate login /username rono /password pwd123 /ssl_port 9091
```

## Linux での使用法

```
$ ./smartupdate login --username rono --password pwd123 --ssl_port 9091
```

### SUCCESS\_NO\_REBOOT

**Windows :** 0

**Linux :** 0

**テキスト :** The command was successful. (コマンドが成功しました。)

### SUCCESS\_REBOOT

**Windows :** 1

**Linux :** 1

**テキスト :** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

### SUCCESS\_NOT\_REQUIRED

**Windows :** 3

**Linux :** 3

**テキスト:** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows :** -1

**Linux :** 255

**テキスト:** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows :** -2

**Linux :** 254

**テキスト:** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows :** -3

**Linux :** 253

**テキスト:** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

#### **FAILURE\_COMMAND\_FAILED**

**Windows :** -4

**Linux :** 252

**テキスト:** The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## setattributes

SUM ノードまたはセッションの属性を変更できます。権限が適切でない場合、要求されたセットにパラメーターが含まれていない場合、または値が使用できない場合、setattributes コマンドは無効なパラメーターであるというメッセージを返すことがあります。

---

**注記:** setattributes オプションとその値は、等号 (=) で区切ります。オプションと値の組み合わせでは、等号の前後にスペースを追加しないでください。

---

変更できる属性のリストを表示するには、getattributes コマンドを使用します。使用可能な属性は、各セッションに追加されたノードおよびベースラインに基づいて変わります。以下に例を示します。

```
smartupdate /getattributes /session (Windows)
```

```
smartupdate /getattributes /nodes (Windows)
```

### 構文

```
smartupdate setattributes --nodes [<nodelist>] attribute1=value1  
attribute2=value2
```

```
smartupdate setattributes --session attribute1=value1
```

対話式 CLI のセッション属性を割り当てます。

```
smartupdate setattributes --session <attribute>=<value> (Linux)
```



smartupdate setattributes /session <attribute>=<value> (Windows)

## パラメーター

**--nodes** [<nodelist>] attribute=value

ノード、およびノードに適用する属性を指定します。

**--session** attribute=value

適用するセッション属性を指定します。セッション属性は全体に適用されます。

## セッションパラメーターのオプション

**auto\_save\_session**

[true|false]

**auto\_save\_session\_time**

<integer> - 時間（分単位） デフォルトは 15 分です。

**engine\_mode**

[debug|release]

**software**

[true|false] - この値には、このセッションで追加されたすべてのノードを対象に、オプションを定義します。

**firmware**

[true|false] - この値には、このセッションで追加されたすべてのノードを対象に、オプションを定義します。

**rewrite**

[true|false] - この値には、このセッションで追加されたすべてのノードを対象に、オプションを定義します。

**downgrade**

[true|false] - この値には、このセッションで追加されたすべてのノードを対象に、オプションを定義します。

**language**

[en|ja|cn]

**verbosity**

[normal|verbose|veryverbose]

- ・ normal - 展開について SUM が提供するデフォルトの量の情報。
- ・ verbose - 情報を使用可能な場合、展開に関してより詳細な情報を提供します。
- ・ veryverbose - 情報を使用可能な場合、展開に関してさらに詳細な情報を提供します。

**verbose**

[true|false]

smartupdate setattributes /nodes verbose=true

展開時に SUM が提供する詳細情報の量を決定します。

## **report\_dir**

[directory\_path] - SUM がレポートを保存するディレクトリのパス。以下に例を示します。c:\SUM\_reports\.

## **open\_firewall**

[true|false]

外部通信に使用する HTTP および HTTPS ポートを開くことを SUM に許可するオプションパラメーター。リモートノード機能およびリモートブラウザアクセスのためにポートを開くときにも使用します。

## **ノードパラメーターのオプション**

### **Use\_vCenter**

VMware ノード用のオプション :

[true|false]

この属性を true に設定する場合は、以下の設定を入力します。

- ・ vcenter\_host - VMware vCenter ノードの IP アドレス。
- ・ vcenter\_username - VMware vCenter ノードのユーザー名。
- ・ vcenter\_password - VMware vCenter ノードに接続するユーザーのパスワード。

これらのパラメーターは、セッション属性として設定することもできます。

### **encryption\_key**

オプション :

最小 8 文字の文字列。

FIPS モードを有効にしている VC ノードに必要なパラメーター。SUM は、暗号化キーを使用して、アップデートプロセス中にディスクに保存された構成を暗号化します。この値は独立しており、VC のパスワードに一致させる必要はありません。

VC インベントリが正常に完了した後にこのパラメーターを割り当てます。

### **ethactorder**

設定可能な値 :

ODDEVEN/PARALLEL/SERIAL/MANUAL。デフォルト値は ODDEVEN。

VC-Enet モジュールの有効化順序を決めるには、このコマンドを使用します。

### **ilo\_username**

設定可能な値 :

<ilo username>

iLO 5 が高セキュリティモードのいずれかで動作している Windows および Linux Gen10 のターゲットノードに対して管理者権限を持つ iLO ユーザー名。

### **ilo\_password**

設定可能な値 :

<ilo password>

iLO 管理者ユーザーのパスワード。

### **fcactorder**

設定可能な値 :

ODDEVEN/PARALLEL/SERIAL/MANUAL。デフォルト値は SERIAL。

VC ターゲットに対する VC - FC の有効化順序を決めるには、このコマンドを使用します。

#### **ethactdelay**

有効なエントリー :

0~60 分。デフォルトは 0 分です。

VC ターゲットに対する VC-Enet モジュールの有効化または再起動の間の時間を分単位で決めるには、このコマンドを使用します。

#### **fcactdelay**

有効なエントリー :

0~60 分。デフォルトは 0 分です。

VC-FC モジュールの有効化または再起動の間の時間を分単位で決めるには、このコマンドを使用します。

#### **use\_sshkey**

(Linux ノードのみ)

指定される可能性があるオプション :

- ・ true
- ・ false

その他の属性 :

- ・ privatekeyfile
  - プライベートキーのアドレスを入力します。プライベートキーは、PEM 形式でなければなりません。
- ・ passphrase
  - (オプション) プライベートキーのパスフレーズを入力します。

これらのパラメーターは、セッション属性として設定することもできます。

#### **no\_mgmt**

Windows および Linux のターゲット

指定される可能性があるオプション :

- ・ true
- ・ false

展開では、AMS、SNMP、または WBEM Provider は選択されません。

インストールセットが更新されるように、ノードインベントリの前にこの属性を設定する必要があります。ノードインベントリの後に設定しても、作成されるインストールセットには反映されません。

#### **use\_snmp**

Gen8/9 Windows および Linux のターゲットの場合。

Windows : 他のコンポーネントとともに、展開のための HPE Insight マネジメントエージェント (SNMP エージェント) を選択します。

Linux : 他のコンポーネントとともに、sm テンプレートおよび snmp エージェント rpm コンポーネントを選択します。

指定される可能性があるオプション :

- ・ true
- ・ false

SUM は SNMP コンポーネントを展開します。

インストールセットが更新されるように、ノードインベントリの前にこの属性を設定する必要があります。ノードインベントリの後に設定しても、作成されるインストールセットには反映されません。

#### **use\_wmi**

(Windows のみ)

指定される可能性があるオプション :

- ・ true
- ・ false

SUM は、WBEM Provider コンポーネントを展開します。(Gen8/9 のみ)

インストールセットが更新されるように、ノードインベントリの前にこの属性を設定する必要があります。ノードインベントリの後に設定しても、作成されるインストールセットには反映されません。

#### **use\_ams**

(Gen9 以降)

指定される可能性があるオプション :

- ・ true
- ・ false

SUM は Windows システムと Linux システムの両方に AMS コンポーネントを展開します。

インストールセットが更新されるように、ノードインベントリの前にこの属性を設定する必要があります。ノードインベントリの後に設定しても、作成されるインストールセットには反映されません。

#### **skip\_prereqs = <true|false>**

デフォルトでは、Windows ノードに CHIF ドライバーがインストールされます。true オプションを使用すると、CHIF ドライバーなど、必須のセルフインベントリコンポーネントのインストールをスキップします。

#### **ignore\_tpm**

指定される可能性があるオプション :

[true|false]

このパラメーターは、TPM または TM モジュールがサーバーにインストールされている場合に、TPM 警告メッセージを無視するように SUM に指示します。

このパラメーターは、サーバーおよび iLO ノードでサポートされます。

#### **on\_failed\_dependency**

[omitthost|omitcomponent|force]

- ・ `omitost - SUM` は、失敗した依存関係を検出した場合、ホストをスキップします。(デフォルト)
- ・ `omitcomponent - SUM` は、失敗した依存関係を含むコンポーネントをスキップします。ノード上の他のコンポーネントは展開されます。
- ・ `force - SUM` は、失敗した依存関係を含むコンポーネントを強制的に展開します。

#### **ignore\_warnings**

指定される可能性があるオプション :

`[true|false]`

このパラメーターは、何かが発生した場合に警告を無視するよう SUM に指示します。エラーは、次のとおりです。

- ・ 保留中の iLO 5 タスクキューの警告。
- ・ 例外タスクキューの警告。
- ・ iLO 5 または iSUT の警告。
- ・ TPM/TM の警告。
- ・ ディスク容量の不足。

#### **action**

有効なオプション :

`ifneeded、always、never`

このノードの再起動オプションを決定します。

#### **delay**

有効なエンタリー :

60~3600 秒

指定された時間 (秒) だけ、サーバーの再起動を遅らせます。

指定された値が 60 未満の場合のデフォルトは 60、3600 を超える値の場合は 3600 です。

#### **message**

有効なエンタリー :

`<Reboot message>`

再起動するサーバーに接続されているリモートコンソールに、指定した再起動メッセージを表示します。

#### **iLO 5 のパラメーター**

##### **save\_install\_set <true/false>**

True : iLO レポジトリにインストールセットを保存します。

False : (デフォルト) iLO レポジトリにインストールセットを保存しません。

##### **install\_set\_name <SPP\_baseline>**

iLO レポジトリに保存するインストールセットの名前を指定します。SPP ベースラインにのみ有効です。

##### **install\_set\_description <description>**

インストールセットを割り当てる説明を保存します。

**manually\_manage\_iLO\_repository = <true>**

True : 新しいインストールセット用のスペースを作るためにインストールセットが自動的に削除されないようにします。

False (デフォルト)

**update\_existing\_recoveryset<true/false>**

True : iLO リポジトリ内の既存のリカバリセットを、iLO、BIOS、CPLD、IE、および ME (インテルプロセッサの場合にのみ、IE と ME) の新しいファームウェアバージョンを使用して、現在の展開からアップデートします。

False (デフォルト) iLO リポジトリ内の既存のリカバリセットをアップデートしません。

## Superdome X パラメーターのオプション

**skip\_missing\_compsig**

設定可能な値 : [TRUE|FALSE]

コンポーネント署名ファイルがないコンポーネントの展開をスキップします。デフォルト値は false です。

**update\_type**

[ALL/COMPLEX/PARTITIONS] - この値は、Superdome X ノードの何をアップデートするのかを SUM に通知します。

**device\_list**

[devicelist] - この値は、アップデートの対象にする Superdome X ノードを SUM に通知します。

**reboot\_list**

[rebootlist] - この値は、展開後に再起動する Superdome X ノードを SUM に通知します。

## パラメーターオプションのダウングレードと再書き込み

GUI のダウングレードと再書き込みの組み合わせ、および対応する対話式 CLI パラメーターについて以下で説明します。

### ソフトウェアのみ

```
smartupdate setattributes /nodes localhost rewrite=false downgrade=false  
firmware=false software=true
```

### ファームウェアのみ

```
smartupdate setattributes /nodes localhost rewrite=false downgrade=false  
firmware=true software=false
```

### ファームウェアとソフトウェア

```
smartupdate setattributes /nodes localhost rewrite=false downgrade=false  
firmware=true software=true
```

### ソフトウェアのダウングレード

```
smartupdate setattributes /nodes localhost rewrite=false downgrade=true  
firmware=false software=true
```

### ファームウェアのダウングレード

```
smartupdate setattributes /nodes localhost rewrite=false downgrade=true  
firmware=true software=false
```

## 両方のダウングレード

```
smartupdate setattributes /nodes localhost rewrite=false downgrade=true  
firmware=true software=true
```

## ソフトウェアの再書き込み

```
smartupdate setattributes /nodes localhost rewrite=true downgrade=false  
firmware=false software=true
```

## ファームウェアの再書き込み

```
smartupdate setattributes /nodes localhost rewrite=true downgrade=false  
firmware=true software=false
```

## 両方の再書き込み

```
smartupdate setattributes /nodes localhost rewrite=true downgrade=false  
firmware=true software=true
```

## ソフトウェアの再書き込みとダウングレード

```
smartupdate setattributes /nodes localhost rewrite=true downgrade=true  
firmware=false software=true
```

## ファームウェアの再書き込みとダウングレード

```
smartupdate setattributes /nodes localhost rewrite=true downgrade=true  
firmware=true software=false
```

## 両方の再書き込みとダウングレード

```
smartupdate setattributes /nodes localhost rewrite=true downgrade=true  
firmware=true software=true
```

## Windows での使用法

```
C:\> smartupdate setattributes /nodes 10.0.1.5 firmware=true software=true  
rewrite=true downgrade=true
```

```
C:\> smartupdate setattributes /nodes 10.0.1.5 update_type=all
```

```
C:\> smartupdate setattributes /session open_firewall=yes
```

```
C:\> smartupdate setattributes /nodes 10.0.0.4 use_vcenter=true  
vcenter_host=10.0.0.1 vcenter_username=test vcenter_password=password
```

## Linux での使用法

```
$ ./smartupdate setattributes --nodes 10.0.1.5 firmware=true software=true  
rewrite=true downgrade=true
```

```
$ ./smartupdate setattributes --nodes 10.0.1.5 update_type=all
```

```
$ ./smartupdate setattributes -session open_firewall=yes
```

```
$ ./smartupdate setattributes --nodes 10.0.0.5 use_vcenter=true  
vcenter_host=10.0.0.1 vcenter_username=test vcenter_password=password
```

## SUCCESS\_NO\_REBOOT

**Windows : 0**

**Linux : 0**

**テキスト :** The command was successful. (コマンドが成功しました。)

## SUCCESS\_REBOOT

**Windows : 1**

**Linux : 1**

**テキスト:** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

**SUCCESS\_NOT\_REQUIRED**

**Windows : 3**

**Linux : 3**

**テキスト:** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

**FAILURE\_GENERAL**

**Windows : -1**

**Linux : 255**

**テキスト:** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

**FAILURE\_BAD\_PARM**

**Windows : -2**

**Linux : 254**

**テキスト:** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

**FAILURE\_COMPONENT\_FAILED**

**Windows : -3**

**Linux : 253**

**テキスト:** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

**FAILURE\_COMMAND\_FAILED**

**Windows : -4**

**Linux : 252**

**テキスト:** The command failed. (コマンドが失敗しました。) For details, see the logs. (詳細は、ログを参照してください。)

## shutdownengine

SUM エンジンを終了できます。他のセッションがアクティブの場合、`--force` オプションを使用しないと失敗します。

**△ 注意:** `--force` を使用すると、情報が失われる場合があります。

### 構文

```
smartupdate shutdownengine [--force]
```

### パラメーター

`--force`

他のセッションがアクティブの場合でも強制的にシャットダウンします。



#### **--cleanup\_onexit**

SUM ファイルをリモートノードから削除します（ユーザーおよびデバッグログファイルを除く）。

#### **--cleanupall\_onexit**

SUM ファイルを削除します（インストール済みのファームウェア RPM を含む）。ユーザーログファイルは削除されません。

#### **Windows での使用法**

```
C:/>smartupdate shutdownengine /force
```

#### **Linux での使用法**

```
$ ./smartupdate shutdownengine --force
```

#### **SUCCESS\_NO\_REBOOT**

**Windows :** 0

**Linux :** 0

**テキスト :** The command was successful. (コマンドが成功しました。)

#### **SUCCESS\_REBOOT**

**Windows :** 1

**Linux :** 1

**テキスト :** The command was successful but a reboot is required. (コマンドが成功しましたが、再起動が必要です。)

#### **SUCCESS\_NOT\_REQUIRED**

**Windows :** 3

**Linux :** 3

**テキスト :** The command completed successfully, but was not required or everything was already current. (コマンドは正常に完了しましたが、コマンドが不要であったか、すべてが最新です)

#### **FAILURE\_GENERAL**

**Windows :** -1

**Linux :** 255

**テキスト :** A general failure occurred. (一般障害が発生しました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_BAD\_PARM**

**Windows :** -2

**Linux :** 254

**テキスト :** A bad input parameter was encountered. (不正な入力パラメーターが検出されました。) For details, see the logs. (詳細は、ログを参照してください。)

#### **FAILURE\_COMPONENT\_FAILED**

**Windows :** -3

**Linux :** 253

**テキスト :** The installation of the component failed. (コンポーネントのインストールに失敗しました。)

FAILURE\_COMMAND\_FAILED

Windows : -4

Linux : 252

テキスト: The command failed. (コマンドが失敗しました。) For details, see the logs.  
(詳細は、ログを参照してください。)

## コンポーネントのステージング

### 構文

```
--stage_only (Linux)
```

```
/stage_only (Windows)
```

### 説明

新しいインストールセットを作成して、適用可能なすべてのコンポーネントを iLO レポジトリにアップロードできます。ステージング後にノードの展開を選択し、新しいインストールセットを呼び出して展開を完了することができます。

### 例

```
smartupdate --silent <other_params> --save_install_set --install_set_name Test1  
--stage_only (Linux)
```

```
/smartupdate /silent <other_params> /save_install_set /install_set_name Test1 /  
stage_only (Windows)
```

## stageonly

新しいインストールセットを作成して、適用可能なすべてのコンポーネントを iLO レポジトリにアップロードします。

### 構文

```
--stage_only (Linux)
```

```
/stage_only (Windows)
```

### 設定可能な値

YES、NO (デフォルト)

## validate baselines

### 説明

ベースラインの追加後、バンドル/コンポーネントのメタデータのインベントリ作成について、問題がないことを確認します。

### 構文

```
smartupdate validate --baselines [<baselinelocation>]
```

### パラメーター

#### baselinelocation

この操作の実行に使用するベースラインの場所。

### Windows での使用法

```
C:\>smartupdate validate /baselines c:\spp\baseline1
```

### **Linux での使用法**

```
$ ./smartupdate validate --baselines /root/spp/baseline1
```

# Web サイトおよびサポート

## Web サイト

Hewlett Packard Enterprise Information Library	<a href="http://www.hpe.com/info/EIL">http://www.hpe.com/info/EIL</a>
Smart Update Manager	<a href="http://www.hpe.com/servers/sum">http://www.hpe.com/servers/sum</a>
Smart Update Manager Information Library	<a href="http://www.hpe.com/info/sum-docs">http://www.hpe.com/info/sum-docs</a>
Smart Update ツール	<a href="http://www.hpe.com/servers/sut">http://www.hpe.com/servers/sut</a>
Smart Update ツールの Information Library	<a href="http://www.hpe.com/info/isut-docs">http://www.hpe.com/info/isut-docs</a>
Service Pack for ProLiant	<a href="http://www.hpe.com/jp/spp">http://www.hpe.com/jp/spp</a>
Service Pack for ProLiant のドキュメント	<a href="http://www.hpe.com/info/spp/documentation">http://www.hpe.com/info/spp/documentation</a>
Service Pack for ProLiant のダウンロード	<a href="http://www.hpe.com/jp/servers/spp_dl">http://www.hpe.com/jp/servers/spp_dl</a>
Service Pack for ProLiant カスタムダウンロード	<a href="http://www.hpe.com/servers/spp/custom">http://www.hpe.com/servers/spp/custom</a>
HPE SDR サイト	<a href="http://downloads.linux.hpe.com">http://downloads.linux.hpe.com</a>

上記以外の Web サイトについては、[サポートと他のリソース](#)を参照してください。

## サポートと他のリソース

### Hewlett Packard Enterprise サポートへのアクセス

- ・ ライブアシスタンスについては、Contact Hewlett Packard Enterprise Worldwide の Web サイトにアクセスします。

<https://www.hpe.com/info/assistance>

- ・ ドキュメントとサポートサービスにアクセスするには、Hewlett Packard Enterprise サポートセンターの Web サイトにアクセスします。

<https://www.hpe.com/support/hpesc>

#### ご用意いただく情報

- ・ テクニカルサポートの登録番号（該当する場合）
- ・ 製品名、モデルまたはバージョン、シリアル番号
- ・ オペレーティングシステム名およびバージョン
- ・ ファームウェアバージョン

- ・ エラーメッセージ
- ・ 製品固有のレポートおよびログ
- ・ アドオン製品またはコンポーネント
- ・ 他社製品またはコンポーネント

## アップデートへのアクセス

- ・ 一部のソフトウェア製品では、その製品のインターフェイスを介してソフトウェアアップデートにアクセスするためのメカニズムが提供されます。ご使用の製品のドキュメントで、ソフトウェアの推奨されるソフトウェアアップデート方法を確認してください。
- ・ 製品のアップデートをダウンロードするには、以下のいずれかにアクセスします。

### Hewlett Packard Enterprise サポートセンター

<https://www.hpe.com/support/hpesc>

### Hewlett Packard Enterprise サポートセンター：ソフトウェアのダウンロード

<https://www.hpe.com/support/downloads>

### Software Depot

<https://www.hpe.com/support/softwaredepot>

- ・ eNewsletters およびアラートをサブスクライブするには、以下にアクセスします。

<https://www.hpe.com/support/e-updates-ja>

- ・ お客様の資格を表示、アップデート、または契約や保証をお客様のプロファイルにリンクするには、Hewlett Packard Enterprise サポートセンターの **More Information on Access to Support Materials** ページに移動します。

<https://www.hpe.com/support/AccessToSupportMaterials>

- 
- ❗ **重要:** 一部のアップデートにアクセスするには、Hewlett Packard Enterprise サポートセンターからアクセスするときに製品資格が必要になる場合があります。関連する資格を使って HPE パスポートをセットアップしておく必要があります。
- 

## リモートサポート（HPE 通報サービス）

リモートサポートは、保証またはサポート契約の一部としてサポートデバイスでご利用いただけます。リモートサポートは、インテリジェントなイベント診断を提供し、ハードウェアイベントを Hewlett Packard Enterprise に安全な方法で自動通知します。これにより、ご使用の製品のサービスレベルに基づいて、迅速かつ正確な解決が行われます。ご使用のデバイスをリモートサポートに登録することを強くおすすめします。

ご使用の製品にリモートサポートの追加詳細情報が含まれる場合は、検索を使用してその情報を見つけてください。

### リモートサポートおよびプロアクティブケア情報

#### HPE 通報サービス

<http://www.hpe.com/jp/hpalert>

#### HPE プロアクティブケアサービス

<http://www.hpe.com/services/proactivecare>

#### HPE データセンターケアサービス

<http://www.hpe.com/services/datacentercare>

HPE プロアクティブケアサービス：サポートされている製品のリスト

<http://www.hpe.com/services/proactivecaresupportedproducts>

HPE プロアクティブケアアドバンストサービス：サポートされている製品のリスト

<http://www.hpe.com/services/proactivecareadvancedsupportedproducts>

## 保証情報

ご使用の製品の保証情報を確認するには、以下のリンクを参照してください。

HPE ProLiant と IA-32 サーバーおよびオプション

<https://www.hpe.com/support/ProLiantServers-Warranties>

HPE Enterprise および Cloudline サーバー

<https://www.hpe.com/support/EnterpriseServers-Warranties>

HPE ストレージ製品

<https://www.hpe.com/support/Storage-Warranties>

HPE ネットワーク製品

<https://www.hpe.com/support/Networking-Warranties>

## 規定に関する情報

安全、環境、および規定に関する情報については、Hewlett Packard Enterprise サポートセンターからサーバー、ストレージ、電源、ネットワーク、およびラック製品の安全と準拠に関する情報を参照してください。

<https://www.hpe.com/support/Safety-Compliance-EnterpriseProducts>

### 規定に関する追加情報

Hewlett Packard Enterprise は、REACH（欧州議会と欧州理事会の規則 EC No 1907/2006）のような法的な要求事項に準拠する必要に応じて、弊社製品の含有化学物質に関する情報をお客様に提供することに全力で取り組んでいます。この製品の含有化学物質情報レポートは、次を参照してください。

<https://www.hpe.com/info/reach>

RoHS、REACH を含む Hewlett Packard Enterprise 製品の環境と安全に関する情報と準拠のデータについては、次を参照してください。

<https://www.hpe.com/info/ecodata>

社内プログラム、製品のリサイクル、エネルギー効率などの Hewlett Packard Enterprise の環境に関する情報については、次を参照してください。

<https://www.hpe.com/info/environment>

## ドキュメントに関するご意見、ご指摘

Hewlett Packard Enterprise では、お客様により良いドキュメントを提供するように努めています。ドキュメントを改善するために役立てさせていただきたくしますので、何らかの誤り、提案、コメントなどがございましたら、ドキュメントフィードバック担当 ([docsfeedback@hpe.com](mailto:docsfeedback@hpe.com)) へお寄せください。このメールには、ドキュメントのタイトル、部品番号、版数、およびドキュメントの表紙に記載されている刊行日をご記載ください。オンラインヘルプの内容に関するフィードバックの場合は、製品名、製品のバージョン、ヘルプの版数、およびご利用規約ページに記載されている刊行日もお知らせください。