



## **Cisco UCS Manager PowerTool ユーザ ガイド、リリース 2.x**

初版：2016年3月1日

最終更新：2019年1月18日

### **シスコシステムズ合同会社**

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター

0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>





## 目次

---

### 第 1 章

#### はじめに 1

Cisco UCS Manager PowerTool 1

管理情報モデル 1

管理対象オブジェクト 2

管理対象オブジェクトの参照 3

管理対象オブジェクトのプロパティ 4

方式 5

PowerTool マッピング 7

システム要件 9

---

### 第 2 章

#### 使用する前に 11

Cisco UCS Manager への接続 11

デフォルト UCS 12

複数の UCS を使用した UCS デフォルトリスト 12

ファイルの資格情報の読み取り/書き込み 13

IPv6 のサポート 13

SSL の処理 14

Cisco UCS Central への登録または登録解除 14

エイリアス 15

---

### 第 3 章

#### 例 17

PowerTool コマンドレットの生成 19

PowerTool DSC 設定の生成 19

UCS Manager GUI アクションからの DSC 設定の生成 20

UCS Manager GUI アクションによるコマンドレットの生成	20
HTML 5 GUI からのコマンドレットの生成	21
UCS Desired State Configuration (DSC)	21
UCS DSC UcsManagedObject リソース	22
UCS DSC UcsSyncMoWithReference リソース	24
UCS DSC UcsSyncFromBackup リソース	26
UCS DSC UcsScript リソース	28
UCS サーバの取得	30
Org	30
障害	30
Get コマンドレット -Hierarchy フラグ	31
Get コマンドレット -LimitScope フラグ	31
トランザクション サポート	31
VLAN の作成と削除	32
MAC プールおよびブロック	33
サーバプール	33
UUID 接尾辞のプールおよびブロック	34
WWNN プールおよびブロック	34
WWNN プールおよびブロック	34
IQN 接尾辞のプールおよびブロック	35
ポート ロール	35
ポート チャネル	36
VLAN の割り当て	37
ブレードの電力および温度の統計情報	37
コンフィギュレーションのバックアップ	37
インポート設定	38
管理対象オブジェクトの同期	38
UCS 管理対象オブジェクト移行の監視	39
テクニカル サポート	39
サービス プロファイル	40
サービス プロファイルのコンポーネント	41

サービス プロファイルのアソシエーション	41
Filters	42
iSCSI ブート	43
vNIC テンプレート	44
vHBA テンプレート	45
ブート ポリシー	45
アダプタ ポリシー	46
BIOS ポリシー	47
ホスト ファームウェア パッケージ	47
IPMI アクセス プロファイル	48
管理ファームウェア パッケージ	48
電力制御ポリシー	49
サーバ プール ポリシー資格情報	49
ダイナミック vNIC 接続ポリシー	50
ネットワーク制御ポリシー	50
権限	50
ユーザ ロール	51
ロケール	51
ユーザ アカウント	52
リモート認証 - RADIUS	52
リモート認証 - TACACS	52
リモート認証 - LDAP	53
RADIUS プロバイダー	53
TACACS プロバイダー	53
LDAP プロバイダー	54
認証ドメイン	54
通信サービス	54
通信サービス - Telnet	55
通信サービス - CIM XML	55
通信サービス - SNMP	55
通信サービス - HTTP	56

通信サービス - HTTPS	56
一般的な管理対象オブジェクトのクエリー	58
一般的な管理オブジェクトのコマンドレット	58
一般的なコマンドレット - XmlTag	59
アップロード ファームウェア	59
XML へのエクスポート	60
XML からのインポート	60
KVM	60
UCS Manager の Java web UI の起動	61
Cisco UCS Manager HTML GUI の起動	62
UCS 統計情報	62
UCS 6324 ファブリック インターコネクットのスケーラビリティ ポートの設定	63
トランザクションの影響	63
コマンドレットのメタ情報	64
Compare-UcsManagedObject - Dn 変換	64
Compare-UcsManagedObject - GetPropertyDiff()	65
コマンドレットの追加 -ModifyPresent フラグ	65
機能カタログの更新	66
サーバの操作	66
32 パラメータ セットの制限	68
Get UcsEquipmentFruVariant	69
Get-UcsEquipmentFruVariantStorage	71
Get-UcsEquipmentManufacturingDef	72
Get-UcsEquipmentManufacturingDefStorage	76
Get-UcsEquipmentPhysicalDef	77
Get-UcsEquipmentPhysicalDefStorage	81
Get-UcsEquipmentPicture	82
Get-UcsEquipmentPictureStorage	84
Get-UcsEquipmentServiceDef	85
Get-UcsEquipmentServiceDefStorage	88
Get-UcsEquipmentSlotArrayRef	90
Get-UcsEquipmentSlotArrayRefStorage	93

Get-UcsFirmwareUpgradeConstraint	94
Get-UcsFirmwareUpgradeConstraintStorage	96

---

**第 4 章****Samples 97**

すべてのグローバルポリシーをフェッチします	97
Cisco UCS コミュニティ	97
関連する Cisco UCS マニュアルとマニュアルフィードバック	97
マニュアルの入手方法およびテクニカルサポート	98





# 第 1 章

## はじめに

---

この章の内容は、次のとおりです。

- [Cisco UCS Manager PowerTool](#) (1 ページ)
- [管理情報モデル](#) (1 ページ)
- [システム要件](#) (9 ページ)

## Cisco UCS Manager PowerTool

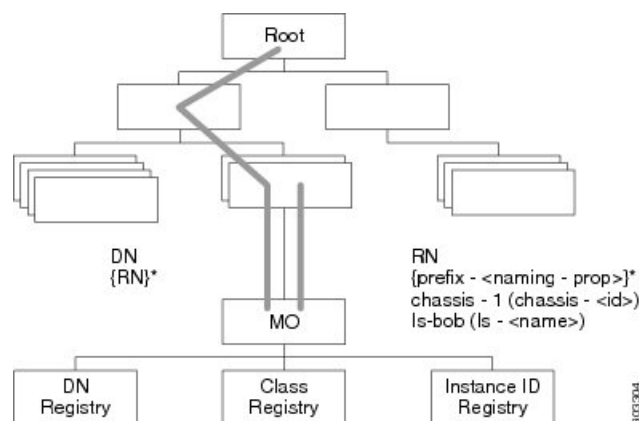
Cisco UCS Manager PowerTool は、サーバ、ネットワーク、ストレージ、ハイパーバイザの管理など、Cisco UCS Manager のあらゆる側面での自動化を支援する PowerShell モジュールです。PowerTool は既存の IT 管理プロセスやツールとの容易な統合を可能にします。

PowerTool コマンドレットは UCS Manager の管理情報ツリー (MIT) で動作し、ツリーの管理対象オブジェクト (MO) の作成、編集、削除を行います。次の章では、Cisco UCS 管理情報モデル (MIM) および PowerTool コマンドレットとの関係について概要を説明します。

UCS PowerTool の設定はについて学習する最も簡単な方法の 1 つは、ConvertTo-UcsCmdlet を使用した GUI で設定作業を行う PowerTool コマンドレットを生成することです。詳細は、を参照してください。

## 管理情報モデル

Cisco UCS ドメインを構成するすべての物理および論理コンポーネントは、管理情報ツリー (MIT) と呼ばれる階層的な管理情報モデル (MIM) で表されます。ツリー内の各ノードは、識別名 (DN) で識別される管理対象オブジェクト (MO) を表します。図は MIM についての説明です。



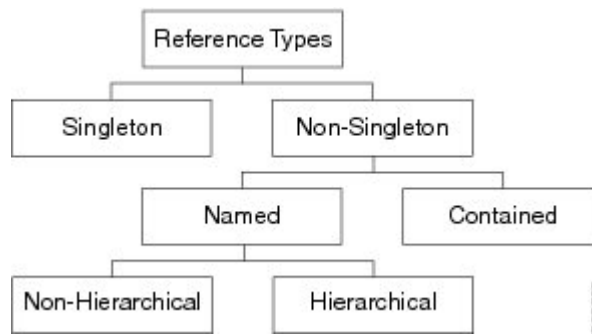
次の図に、シャーシが3つの場合のサンプル（一部）MITを示します。

次の図に、シャーシが3つの場合のサンプル（一部）MITを示します。

ツリー (topRoot)	識別名
- sys	sys
- chassis-1	sys/chassis-1
- chassis-2	sys/chassis-2
- chassis-3	sys/chassis-3
- blade-1	sys/chassis-3/blade-1
- adaptor-1	sys/chassis-3/blade-1/adaptor-1
- blade-2	sys/chassis-3/blade-2
- adaptor-1	sys/chassis-3/blade-2/adaptor-1
- adaptor 2	sys/chassis-3/blade-2/adaptor-2

## 管理対象オブジェクト

管理対象オブジェクト (MO) は、ファブリック インターコネクト、シャーシ、ブレード、ラックマウント サーバなどの Cisco UCS ドメイン リソースを抽象化したものです。管理対象オブジェクトは、Cisco UCS MIT で設定される、または管理される論理または物理エンティティを表します。たとえば、サーバ、シャーシ、I/O カード、およびプロセッサなどの物理エンティティ、リソース プール、ユーザ ロール、サービス プロファイル、およびポリシーなどの論理エンティティが、管理対象オブジェクトとして表されます。



それぞれの管理対象オブジェクトは、識別名（DN）によってツリー内で特定でき、相対名（RN）によって親のコンテキスト内で特定できます。DN は、MIT での MO の位置を識別します。DN は、ルートから開始してその MO 自身までのすべての相対名を連結したものです。Essentially, Dn = [Rn]/[Rn]/[Rn]/.../[Rn].

次の例では、DN はモデル内の adaptor-1 の完全修飾名を示しています。

```
< dn = "sys/chassis-5/blade-2/adaptor-1" />
```

上記の DN は次の RN で構成されます。

```
topSystem MO: rn="sys" equipmentChassis MO: rn="chassis-<id>" computeBlade MO:
rn ="blade-<slotId>" adaptorUnit MO: rn="adaptor-<id>"
```

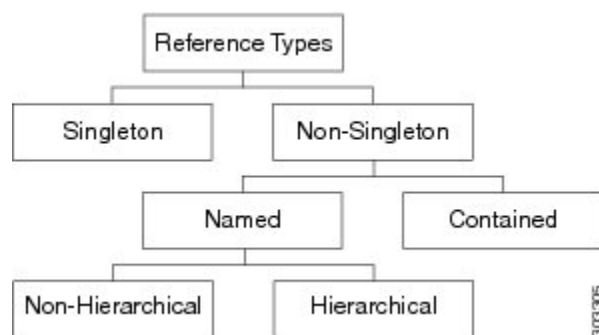
相対名（RN）には、1つ以上の MO のプロパティの値を組み込むことができます。これによって、親のコンテキスト内で同じタイプの複数の MO を区別することができます。前述したような RN の一部を成すプロパティは、Naming プロパティと呼ばれます。

たとえば、複数のブレードの MO が 1つのシャーシの MO の下にあります。ブレード MO には RN の一部としてブレードの ID 情報 (blade- [ID]) が含まれるので、シャーシのコンテキスト内で各ブレード MO を個別に識別できます。

## 管理対象オブジェクトの参照

管理対象オブジェクトの内容は、Cisco UCS の動作中に参照されます。一部の MO は暗黙に追加されるか(ログイン中に PreLoginBanner)または別の MO の展開の一部として参照されます。サービス プロファイル月がテンプレートを参照してくださいまたは VNIC は VLAN MOs の数を示します。

参照は、次のように分類できます。



単一 MO タイプは、MIT 全体で一度だけ出現し、通常、暗黙的に参照されます。

非単一 MO タイプは MIT 内で 1 回以上インスタンス化されることがあります。MO が別の MO を参照するとき、参照は名前で行われることがよくあります。参照された MO のタイプに応じて、解決は階層的になる可能性があります。たとえば、サービス プロファイル テンプレートは Org に定義されているとします。org には sub org が含まれていることがあるため、下位 Org に同じ名前で定義されたサービス プロファイル テンプレートがある場合があります。ここでサービス プロファイル インスタンスがサービス プロファイル テンプレートを（名前で）参照する場合、名前がサービス プロファイル インスタンスの Org からルート Org まで階層的に検索されます。最初に一致したエントリが使用されます。一致するエントリが見つからない場合、「default」という名前が同様の方法で検索され、最初に一致したエントリが使用されます。

参照タイプ	例
単一	ChassisDiscoveryPolicy PreLoginBanner
非単一/名前付き/非階層型	CallHomePolicy
非単一/名前付き/階層型	BiosPolicy BootPolicy
非単一/包含	LsServer (ServiceProfile) の下の BootDefinition VnicEther の下の VnicEtherIf

## 管理対象オブジェクトのプロパティ

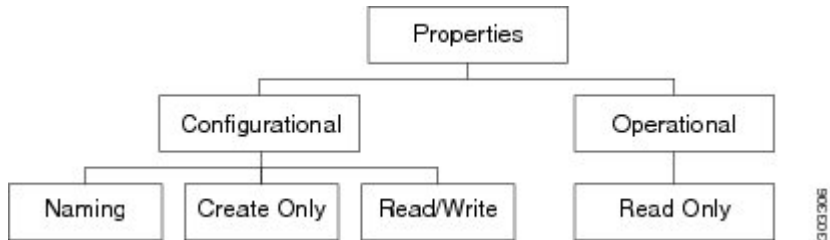
管理対象オブジェクトのプロパティは、コンフィギュレーションまたはオペレーションとして分類されます。

コンフィギュレーション プロパティは次のように分類されます。

- Naming プロパティ：RN の一部になります。MO を作成時に指定でき、これは後で変更できません。

- **Create-Only** プロパティ：MOの作成中にのみ指定でき、後で変更できません。プロパティが指定されていない場合、デフォルト値が使用されます。
- **Read/Write** プロパティ：MOの作成時に指定でき、後で変更もできます。

オペレーションプロパティはMOまたはシステムのステータスを示し、読み取り専用です。



次の表は、さまざまなプロパティタイプの例を示します。

プロパティタイプ	例
Naming	LsServer（サービスプロファイルMO）内の名前
Create-Only	LsServer内のタイプ
読み取り/書き込み	LsServer内の説明
Read-Only	LsServer内の OperState

## 方式

Cisco UCS XML API は、システムの管理および監視に使用されます。以下のサポートするメソッドがあります。

- Authentication
- AaaLogin
- AaaRefresh
- AaaLogout
- Configuration
- ConfigConfMo(s)
- LsClone
- Lsinstantiate\*
- FaultAckFaults
- Query
- ConfigResolveDn(s)

- ConfigResolveClass(es)
- ConfigResolveChildren
- Event Monitor
- EventSubscribe

クラス クエリー メソッド ConfigResolveClass(es) およびConfigResolveChildren は特定のセットの MO と照合し、メソッドごとに返すようにフィルタリングします。

次のフィルタがサポートされています。

- プロパティ フィルタ :

サポートされているフィルタ	定義
allbits	指定したすべての値が複数値プロパティに存在する場合に一致
anybit	指定した値のいずれかが複数値プロパティに存在する場合に一致
bw	プロパティの値が指定した2つの値の間にある場合に一致
eq	プロパティの値が指定した値と同一の場合に一致
ge	プロパティの値が指定した値以上の場合に一致
gt	プロパティの値が指定した値より大きい場合に一致
le	プロパティの値が指定した値以下の場合に一致
lt	プロパティの値が指定した値より小さい場合に一致
ne	プロパティの値が指定した値と等しくない場合に一致
wcard	プロパティの値が指定したパターンに一致した場合に一致

- コンポジット フィルタ (サブ フィルタでの動作)

コンポジット フィルタ	定義
not	サブフィルタの結果を逆にします

コンポジット フィルタ	定義
および	すべてのサブフィルタが <b>true</b> を返した場合に <b>true</b>
または	サブ フィルタのいずれかが <b>true</b> を返した場合に <b>true</b>

## PowerTool マッピング

一部の PowerTool コマンドレットは MO の仕様から生成されます。便利な名詞が使用されます (LsServer の代わりに ServiceProfile など)。コマンドレットの Get、Add、Set、Remove、またはサブセットが、さまざまな MO タイプに対して生成されます。すべてのコマンドレットは、XML 要求および応答を画面にダンプする XML パラメータをサポートします。

**Add コマンドレット** : 指定されたプロパティ値とともに MO ステータス **Created** で ConfigConfMo メソッドを使用します。ModifyPresent パラメータが指定されている場合、**Modified** というステータスが指定されます。**Force** パラメータが指定されている場合、確認を求めるプロンプトは表示されません。

**Get コマンドレット** : ConfigResolveClass メソッドを使用して MO を取得します。プロパティパラメータが指定されている場合、*eq* フィルタの生成に使用されます。複数のプロパティパラメータを指定すると、複数の *eq* フィルタは *and* フィルタと結合されます。

**Set コマンドレット** : 指定されたプロパティ値とともに MO ステータス **Modified** で ConfigConfMo メソッドを使用します。**Force** パラメータが指定されている場合、確認を求めるプロンプトは表示されません。

**Remove コマンドレット** : MO ステータス **Deleted** で ConfigConfMo メソッドを使用します。**Force** パラメータが指定されている場合、確認を求めるプロンプトは表示されません。

次の表は、特定の動詞に対して指定できるプロパティのリストです。

プロパティ	get	Add	Set
Naming	Yes (位置指定)	Yes (位置指定)	No
Create-Only	Yes	Yes	No
Read-Write	Yes	Yes	Yes
Operational / Read-Only	はい	No	×

次の表は、対応するコマンドレットのパイプラインのタイプのリストです。

Verb/タイプ	パイプラインの入力
Get	単一 - 非単一以外 - 親タイプ
Add	単一 - 非単一以外 - 親タイプ

Verb/タイプ	パイプラインの入力
Set	Naming プロパティがある MO – Naming プロパティがない同一タイプの MO – 同一タイプまたは親タイプ
Remove	同一タイプ

次の表は、必要な XML 要求を生成するために呼び出されるメソッドのリストです。

コマンドレット	メソッド
Add-Ucs/ Set-Ucs/	ConfigConfMos
Get-Ucs/	ConfigResolveClass (フィルタ使用)
Get-UcsManagedObject -ClassId	ConfigResolveClass
Get-UcsManagedObject -ClassId -Dnlist	ConfigFindDnsByClassId
Get-UcsManagedObject -Dn	ConfigResolveDns
Connect-Ucs	AaaLogin
Disconnect-Ucs	AaaLogout
Background <a href="#">1</a>	AaaRefresh
Copy-UcsServiceProfile	LsClone
Add-UcsServiceProfileFromTemplate	LsInstantiateTemplate, LsInstantiateNTemplate, LsInstantiateNNamedTemplate
Get-UcsChild	ConfigResolveChildren
Acknowledge-UcsFault	FaultAckFaults
Start-UcsKvmSession	AaaGetNComputeAuthTokenByDn
Watch-Ucs	EventSubscribe (First Watcher)
Clear-UcsStatistics	StatsClearInterval
Get-UcsTransactionImpact	ConfigEstimateImpact

<sup>1</sup> これはコマンドレットではありません。これはバックグラウンドサービスです。

# システム要件

Cisco UCS Manager PowerTool をインストールする前に、システムが次の要件を満たしていることを確認してください。

- Windows PowerShell 3.0 以降をインストール
- .NET Framework バージョン 4.5 以降
- UCS DSC リソース向け Windows PowerShell 4.0 以降

## Cisco UCS Manager

Cisco UCS Manager PowerTool は、次の Cisco UCS Manager リリースと互換性があります。

- リリース 4.0
- リリース 3.2
- リリース 3.1
- リリース 3.0
- リリース 2.5
- リリース 2.2
- リリース 2.1
- リリース 2.0





## 第 2 章

# 使用する前に

---

この章は、次の項で構成されています。

- [Cisco UCS Manager への接続](#) (11 ページ)
- [デフォルト UCS](#) (12 ページ)
- [複数の UCS を使用した UCS デフォルト リスト](#) (12 ページ)
- [ファイルの資格情報の読み取り/書き込み](#) (13 ページ)
- [IPv6 のサポート](#) (13 ページ)
- [SSL の処理](#) (14 ページ)
- [Cisco UCS Central への登録または登録解除](#) (14 ページ)
- [エイリアス](#) (15 ページ)

## Cisco UCS Manager への接続

---

**ステップ 1** デスクトップのショートカットから Cisco UCS Manager PowerTool を起動します。

**ステップ 2** Cisco UCS Manager PowerTool でサポートされるすべてのコマンドレット、機能、およびエイリアスを表示します。

```
Get-Command -Module Cisco.UcsManager
```

```
Get-Command -Module Cisco.UcsManager | group CommandType
```

```
Get-Command -Module Cisco.UcsManager | measure
```

**ステップ 3** Cisco UCS ドメインに接続します。

```
$handle = Connect-Ucs <ip or hostname> -NotDefault
```

---

次のタスク

## デフォルト UCS

ハンドルまたは名前が指定されていない場合、-Ucsパラメータが指定されていなければ、Cisco UCS ドメインハンドルが DefaultUcs ドメイン リストに追加され、パイプラインの最初のコマンドレットがデフォルト Ucs リストで動作します。

**Connect-Ucs**<ip or hostname>

**Get the Default Ucs list**

**Get-UcsPSSession**

**Get UCS consolidated status information**

**Get - UcsChassis**

**Get the set of all chassis objects**

**Get-UcsChassis**

**Get the object pertaining to chassis 1**

**Get-UcsChassis -Id 1**

**Get the set of blades, pertaining to chassis 1**

**Get-UcsChassis -Id 1 | Get-UcsBlade**

**Enable HTTP on the FI**

**Get-UcsHttp | Set-UcsHttp -AdminState enabled**

**Disable HTTP on the FI**

**Get-UcsHttp | Set-UcsHttp -AdminState disabled**

**Disconnect**

**Disconnect - Ucs**

## 複数の UCS を使用した UCS デフォルト リスト

PowerTool コマンドレットは、複数のハンドルを指定することで、複数の Cisco UCS ドメインで使用できます。

**Cisco UCS ドメインに接続します**

```
$handle1 = Connect-Ucs <ip1> -NotDefault
$handle2 = Connect-Ucs <ip2> -NotDefault
Get-UcsStatus -Ucs $handle1,$handle2
Disconnect-Ucs -Ucs $handle1,$handle2
```

デフォルトでは、複数の Cisco UCS ハンドルは DefaultUcs では許可されません。ただし、**Set-UcsPowerToolConfiguration** コマンドレットをしようすることで、オーバーライドできます。

```
Get-UcsPowerToolConfiguration
Set-UcsPowerToolConfiguration -SupportMultipleDefaultUcs $true
Connect-Ucs <ip1>
Connect-Ucs <ip2>
Get-UcsStatus
Disconnect-Ucs
```

同じログイン資格情報を使用して複数の Cisco UCS ドメインに接続します

```
$user = "<username>"
$password = "<password>" |
  ConvertTo-SecureString -AsPlainText -Force
$cred = New-Object System.Management.Automation.PSCredential($user, $password)
$servers = @("<Imc1>", "<Imc2>", "<Imc3>")
Connect-Imc $servers -Credential $cred
```

## ファイルの資格情報の読み取り/書き込み

```
Connect-Ucs <ip1>
Connect-Ucs <ip2>
```

資格情報はファイルに保存できます。保存された資格情報は、指定されたキーで暗号化されます

```
Export-UcsPSSession -LiteralPath C:\work\labs.xml
Disconnect-Ucs
```

ログインはファイルに保存された資格情報から開始できます

```
Connect-Ucs -LiteralPath C:\work\labs.xml
```

ファイルに保存されている資格情報でログイン中にプロキシを指定します

```
$proxy = New-Object System.Net.WebProxy
$proxy.Address = "http://<url>:<port>"
$proxy.UseDefaultCredentials = $false
$proxy.Credentials = New-Object System.Net.NetworkCredential("<user name>", "<password>")
Connect-Ucs -LiteralPath C:\work\lab.xml -Proxy $proxy
```

追加システムにログインし、ファイルに資格情報を追加します

```
Connect-Ucs <ip3>
Export-UcsPSSession -Path C:\work\lab?.xml -Merge
```

## IPv6 のサポート

- IPv6 アドレスを使用して Cisco UCS Manager に接続できます

- scp、ftp、tftp、ntp、dnsなどの外部クライアントアプリケーションや、sshd、httpd、snmpdなどの外部クライアントサービスにIPv6アドレスでアクセスできます。

**Connect-Ucs [2001::0202:\*3F\*:\*E1\*:\*8\*\*9]**

## SSL の処理

ユーザが Cisco UCS サーバに接続した際に有効な証明書が認識されなかった場合は、接続の確立は `InvalidCertificateAction` の設定に依存します。デフォルトでは、無視するよう `InvalidCertificateAction` は設定されています。デフォルトでは、証明書が無効かどうかにかかわらず接続を確立するように PowerTool は設定されています。

**Set-UcsPowerToolConfiguration** コマンドレットを使用してこれを上書きできます。

```
Get-UcsPowerToolConfiguration
Set-UcsPowerToolConfiguration -InvalidCertificateAction Fail
```

名前	説明
<b>Fail</b>	証明書が無効な場合、コマンドレットは接続を確立しません。
<b>Ignore</b>	コマンドレットは、証明書が無効かどうかにかかわらず接続を確立します。
<b>Default</b>	(Windows のデフォルト) 証明書が有効な場合、コマンドレットは接続を確立します。

## Cisco UCS Central への登録または登録解除

Cisco UCS Central を Cisco UCS ドメインの管理に使用する場合は、そのドメインを登録する必要があります。登録する際、Cisco UCS Central や Cisco UCS Manager によって管理されるポリシーのタイプや、バックアップやファームウェアなどのその他の設定を選択する必要があります。

Cisco UCS Central を使用して Cisco UCS ドメインを登録する前に、次を実行してください。

- 
- ステップ 1** Cisco UCS Manager と Cisco UCS Central を確実に同期させるために、双方で NTP サーバおよび正しいタイムゾーンを設定します。Cisco UCS ドメインと Cisco UCS Central の日時が同期していない場合、登録は失敗する可能性があります。
- ステップ 2** Cisco UCS Central のホスト名または IP アドレスを取得します。
- ステップ 3** Cisco UCS Central を導入したときに設定した共有秘密を取得します。

```
$password = "SharedSecret" | ConvertTo-SecureString -AsPlainText -Force
Register-UcsCentral -Name 10.10.10.10 -SharedSecret $password
```

#### ステップ 4 UCS Central からの登録解除

```
Get-UcsCentral | Unregister-UcsCentral
```

## エイリアス

便宜上、いくつかのエイリアスが定義されています。

```
gal | ? {$_.Name -like "*-Ucs*" } | select Name
```

次の表に、エイリアスと対応するコマンドレットを示します。

Alias	コマンドレット
Acknowledge-UcsBlade	Set UcsBlade-Lc rediscover
Acknowledge-UcsChassis	Set UcsChassis-AdminState re-acknowledge
Acknowledge-UcsFault	Confirm-UcsFault
Acknowledge-UcsFex	Set-UcsFex -AdminState re-acknowledge
Acknowledge-UcsRackUnit	Set UcsRackUnit-Lc rediscover
Acknowledge-UcsServerUnit	Set UcsServerUnit-Lc rediscover
Acknowledge-UcsSlot	Set-UcsFabricComputeSlotEp -AdminState reacknowledge
Add-UcsMo	Add-UcsManagedObject
Associate-UcsServiceProfile	Connect-UcsServiceProfile
Compare-UcsMo	Compare-UcsManagedObject
Decommission-UcsBlade	Set-UcsBlade -Lc decommission
Decommission-UcsChassis	Set UcsChassis AdminState decommission
Decommission-UcsFex	Set UcsFex AdminState decommission
Decommission-UcsRackUnit	Set UcsRackUnit Lc decommission
Decommission-UcsServerUnit	Set UcsServerUnit Lc decommission
Disable UcsDiskLocatorLed	Set UcsStorageLocalDisk AdminActionTrigger triggered -AdminAction led-off

Alias	コマンドレット
Disassociate-UcsServiceProfile	Disconnect-UcsServiceProfile
Enable UcsDiskLocatorLed	Set UcsStorageLocalDisk AdminActionTrigger-triggered -AdminAction led-on
Get-UcsCentral	Get UcsPolicyControlEp
Get-UcsMo	Get-UcsManagedObject
Recommission-UcsBlade	Set-UcsFabricComputePhEp -AdminState enabled
Recommission-UcsChassis	Set-UcsFabricSwChPhEp -AdminState enabled
Recommission-UcsFex	Set-UcsFabricSwChPhEp -AdminState enabled
Recommission-UcsRackUnit	Set-UcsFabricComputePhEp -AdminState enabled
Recommission-UcsServerUnit	Set UcsFabricComputeMSlotEp-SlotAdminState reacknowledge
Remove-UcsBlade	Set UcsBlade、 Lc remove
Remove-UcsCartridge	Set-UcsCartridge -Lc remove
Remove-UcsChassis	Set-UcsChassis -AdminState remove
Remove-UcsFex	Set-UcsFex -AdminState remove
Remove-UcsMo	Remove-UcsManagedObject
Remove-UcsRackUnit	Set-UcsRackUnit -Lc remove
Reset-UcsIoModule	Set-UcsIom -AdminPowerState cycle-immediate -AdminState acknowledged -AdminPeerPowerState policy
Reset-UcsPeerIoModule	Set-UcsIom -AdminPowerState policy -AdminState acknowledged -AdminPeerPowerState cycle-immediate
Set-UcsMo	Set-UcsManagedObject
Sync-UcsMo	Sync-UcsManagedObject
Unregister-UcsCentral	Remove-UcsPolicyControlEp



## 第 3 章

### 例

この章は、次の項で構成されています。

- [PowerTool コマンドレットの生成 \(19 ページ\)](#)
- [UCS Desired State Configuration \(DSC\) \(21 ページ\)](#)
- [UCS サーバの取得 \(30 ページ\)](#)
- [Org \(30 ページ\)](#)
- [障害 \(30 ページ\)](#)
- [Get コマンドレット -Hierarchy フラグ \(31 ページ\)](#)
- [Get コマンドレット -LimitScope フラグ \(31 ページ\)](#)
- [トランザクション サポート \(31 ページ\)](#)
- [VLAN の作成と削除 \(32 ページ\)](#)
- [MAC プールおよびブロック \(33 ページ\)](#)
- [サーバ プール \(33 ページ\)](#)
- [UUID 接尾辞のプールおよびブロック \(34 ページ\)](#)
- [WWNN プールおよびブロック \(34 ページ\)](#)
- [WWNN プールおよびブロック \(34 ページ\)](#)
- [IQN 接尾辞のプールおよびブロック \(35 ページ\)](#)
- [ポート ロール \(35 ページ\)](#)
- [ポート チャネル \(36 ページ\)](#)
- [VLAN の割り当て \(37 ページ\)](#)
- [ブレードの電力および温度の統計情報 \(37 ページ\)](#)
- [コンフィギュレーションのバックアップ \(37 ページ\)](#)
- [インポート設定 \(38 ページ\)](#)
- [管理対象オブジェクトの同期 \(38 ページ\)](#)
- [UCS 管理対象オブジェクト移行の監視 \(39 ページ\)](#)
- [テクニカル サポート \(39 ページ\)](#)
- [サービス プロファイル \(40 ページ\)](#)
- [サービス プロファイルのコンポーネント \(41 ページ\)](#)
- [サービス プロファイルのアソシエーション \(41 ページ\)](#)
- [Filters \(42 ページ\)](#)

- iSCSI ブート (43 ページ)
- vNIC テンプレート (44 ページ)
- vHBA テンプレート (45 ページ)
- ブート ポリシー (45 ページ)
- アダプタ ポリシー (46 ページ)
- BIOS ポリシー (47 ページ)
- ホスト ファームウェア パッケージ (47 ページ)
- IPMI アクセス プロファイル (48 ページ)
- 管理ファームウェア パッケージ (48 ページ)
- 電力制御ポリシー (49 ページ)
- サーバ プール ポリシー資格情報 (49 ページ)
- ダイナミック vNIC 接続ポリシー (50 ページ)
- ネットワーク制御ポリシー (50 ページ)
- 権限 (50 ページ)
- ユーザ ロール (51 ページ)
- ロケール (51 ページ)
- ユーザ アカウント (52 ページ)
- リモート認証 - RADIUS (52 ページ)
- リモート認証 - TACACS (52 ページ)
- リモート認証 - LDAP (53 ページ)
- RADIUS プロバイダー (53 ページ)
- TACACS プロバイダー (53 ページ)
- LDAP プロバイダー (54 ページ)
- 認証ドメイン (54 ページ)
- 通信サービス (54 ページ)
- 通信サービス - Telnet (55 ページ)
- 通信サービス - CIM XML (55 ページ)
- 通信サービス - SNMP (55 ページ)
- 通信サービス - HTTP (56 ページ)
- 通信サービス - HTTPS (56 ページ)
- 一般的な管理対象オブジェクトのクエリー (58 ページ)
- 一般的な管理オブジェクトのコマンドレット (58 ページ)
- 一般的なコマンドレット - XmlTag (59 ページ)
- アップロード ファームウェア (59 ページ)
- XML へのエクスポート (60 ページ)
- XML からのインポート (60 ページ)
- KVM (60 ページ)
- UCS Manager の Java web UI の起動 (61 ページ)
- Cisco UCS Manager HTML GUI の起動 (62 ページ)
- UCS 統計情報 (62 ページ)
- UCS 6324 ファブリック インターコネクットのスケラビリティポートの設定 (63 ページ)

- [トランザクションの影響](#) (63 ページ)
- [コマンドレットのメタ情報](#) (64 ページ)
- [Compare-UcsManagedObject - Dn 変換](#) (64 ページ)
- [Compare-UcsManagedObject - GetPropertyDiff\(\)](#) (65 ページ)
- [コマンドレットの追加 -ModifyPresent フラグ](#) (65 ページ)
- [機能カタログの更新](#) (66 ページ)
- [サーバの操作](#) (66 ページ)
- [32 パラメータ セットの制限](#) (68 ページ)

## PowerTool コマンドレットの生成

UCS Manager web UI で指定されたアクションのコマンドレットを、次のコマンドレットを使用して生成します。

```
ConvertTo-UcsCmdlet
```

次のコマンドレットを使用して、xml 要求を取得し、コマンドレットを生成します。

```
ConvertTo-UcsCmdlet -Verbose
```

次のコマンドレットを使用して、指定された web UI ログのアクションのコマンドレットを生成します。

```
ConvertTo-UcsCmdlet -GuiLog -LiteralPath 'C:\Work\centrale_7128.log.1'  
ConvertTo-UcsCmdlet -GuiLog -Path 'C:\Work\centrale_71*.log.*'
```

次のコマンドレットを使用して、指定された xml 要求のコマンドレットを生成します。

```
ConvertTo-UcsCmdlet -Xml -Request '<lsClone dn="org-root/ls-sp1" inTargetOrg="org-root"  
inServerName="sp2" inHierarchical="false"></lsClone>'
```

次のコマンドレットを使用して、ファイルで指定された xml 要求のコマンドレットを生成します。

```
ConvertTo-UcsCmdlet -Xml -LiteralPath 'C:\Work\config.xml'
```

特定の MO のコマンドレットを生成します。

リリース 1.2 (1) から、管理オブジェクトを `ConvertTo-UcsCmdlet` にパイプライン処理し、管理対象オブジェクトの作成に必要なコマンドレットを取得できるようになりました。

```
Get-UcsServiceProfile -Name sp1 | ConvertTo-UcsCmdlet  
Get-UcsServiceProfile -Name sp1 -Hierarchy | ConvertTo-UcsCmdlet
```

## PowerTool DSC 設定の生成

`ConvertTo-UcsDSCConfig` コマンドレットでは、UCS Manager GUI のログを読み取り、対応する DSC 設定ファイルを生成します。この機能は、`ConvertTo-UcsCmdlet` コマンドレットに類似し

ています。生成された DSC 設定ファイルは、カスタム UCS DSC UcsManagedObject および UcsScript リソースを使用します。

UCS GUI で指定されたアクションに対する DSC 設定を生成します。

ConvertTo-UcsDSCConfig



(注) ConvertTo-UcsDSCConfig コマンドレットは、UCS Manager の Java GUI でのみサポートされます。

## UCS Manager GUI アクションからの DSC 設定の生成

ConvertTo-UcsDSCConfig コマンドレットを使用して、UCS Manager GUI で実行されるアクションの DSC 設定を生成できます。

ConvertTo-UcsDSCConfig コマンドレットでは、UCS Manager GUI のログを読み取り、対応する DSC 設定ファイルを生成します。このコマンドレットは、ConvertTo-UcsCmdlet と類似していて、コマンドレットを生成します。生成された DSC 設定ファイルは、UCS DSC モジュールの一部である、UcsManagedObject および UcsScript リソースを使用します。

UCS Manager GUI の起動の詳細については、[UCS Manager GUI アクションによるコマンドレットの生成](#)を参照してください。

CiscoUCS モジュールがロードされている PowerShell ウィンドウから UCS Manager GUI が起動した後で、ConvertTo-UcsCmdlet コマンドレットを実行します。

始める前に

## UCS Manager GUI アクションによるコマンドレットの生成

UCS Manager GUI で実行されたアクションのコマンドレットを、ConvertTo-UcsCmdlet コマンドレットを使用して生成することができます。Cisco UCS Manager GUI では、安全のためにわずかな XML スニペットが考慮されていて、これはログに記録されません。したがって、ConvertTo-UcsCmdlet では、変換を実行するためのログは見つかりません。

GUI でのあらゆるユーザアクションの XML スニペットをログに記録するには、次のいずれかの方法で UCS Manager GUI を起動します。

- Start-UcsGuiSession -LogAllXml コマンドレットを使用する
- 次の手順を実行して手動で UCS Manager GUI を起動する

**ステップ 1** 起動リンクを .jnlp ファイル形式で保存します。例：https://<ip\_or\_hostname>/ucsm/ucsm.jnlp

**ステップ 2** ファイルを右クリックし、メモ帳で開きます。

**ステップ 3** その他のプロパティ定義の後に以下の行を追加します。

- Java のバージョンが Java 7 Update 45 よりも前の場合、`<property name="log.show.encrypted" value="true"/>` を追加します。
- Java 7 Update 45 以降のバージョンの場合、`<property name="jnlp.ucsm.log.show.encrypted" value="true"/>` を追加します

**ステップ 4** ファイルを保存して、閉じます。

**ステップ 5** ファイルを右クリックし、Java™ Web Start Launcher で開きます。

---

UCS Manager GUI が起動したら、Cisco UCS Manager Power Tool モジュールがロードされている PowerShell ウィンドウから `ConvertTo-UcsCmdlet` コマンドレットを実行します。

## HTML 5 GUI からのコマンドレットの生成

HTML GUI では、いずれの操作を行う前に、XML 録画を開始し、操作が終了すると、録画を停止してログをダウンロードできます。

---

**ステップ 1** HTML を使用して、UCS Manager を起動します。

**ステップ 2** ログイングを有効にする方法:

- ショートカット キー `Ctrl + Alt + q` を使用します。
- リンク (録画 XML) は、UI の右上隅に表示されます。リンクをクリックします。

GUI で操作を実行します。

**ステップ 3** ログイングを停止するためには、[ XML 録画の停止 (Stop XML Recording) ] をクリックします。

**ステップ 4** ポップアップには、ログ ファイル名を入力します。

**ステップ 5** OK ボタンをクリックして、ファイルをローカル システムにダウンロードします。

**ステップ 6** UCS Manager PowerTool を起動し、`ConvertTo-UcsCmdlet -xml -LiteralPath 'C:\Work\Ucsm.log'` コマンドレットを実行します。

## UCS Desired State Configuration (DSC)

Desired State Configuration (DSC) は、ローカル マシンやリモート マシンを構成するための新しい方式です。UCS DSC のリソースを使用して、集中型ルート サーバからデータセンターに複数の UCS ドメインを設定できます。PowerTool モジュール

「Cisco.UCS.DesiredStateConfiguration」には、すべてのカスタム UCS DSC リソースが含まれています。

```
Get-Module Cisco.UCS.DesiredStateConfiguration -ListAvailable
Get-DscResource | where{$_ .Module -ilike 'Cisco*' -and $_.Name -ilike 'ucs*'} | Select
Name
```

## UCS DSC UcsManagedObject リソース

UCS DSC モジュールの UcsManagedObject リソースは、DSC フレームワークを使用して複数の UCS Manager で、MO の詳細を指定することにより、UCS Manager MO を設定するためのメカニズムを提供します。

### シンタックス

```
UcsManagedObject [string] #ResourceName
{
  Dn = [string]
  Identifier = [string]
  UcsConnectionString = [string]
  UcsCredentials = [PSCredential]
  [ Action = [string] { Add | Set } ]
  [ ClassId = [string] ]
  [ DependsOn = [string[]] ]
  [ Ensure = [string] { Absent | Present } ]
  [ ModifyPresent = [bool] ]
  [ PropertyMap = [string] ]
  [ WebProxyCredentials = [PSCredential] ]
}
```

プロパティ	説明
Dn	管理対象オブジェクトの Dn を指定します。
ID	DSC リソースの一意の ID を指定します。
UcsConnectionString	UCS Manager の接続文字列を指定します。 <b>シンタックス :</b> Name=<ipAddress> [ `nNoSsl=<bool>][ `nPort=<ushort>] [ `nProxyAddress=<proxyAddress>] [ `nUseProxyDefaultCredentials=<bool>]
UcsCredentials	UCS Manager にアクセスするために必要な資格情報を示します。
アクション	管理対象オブジェクトに対して実行するアクションを指定します。管理対象オブジェクトを追加するには、このプロパティを Add に設定します。既存の管理対象オブジェクトを変更するには、Set に設定します。
ClassId	管理対象オブジェクトのクラス ID を指定します。

プロパティ	説明
DependsOn	<p>当該リソースを設定する前に、別のリソースの設定を実行する必要があることを示します。たとえば、最初に実行するリソース設定スクリプトブロックの ID が <b>ResourceName</b>、そのタイプが <b>ResourceType</b> であるとする、このプロパティを使用するシンタックスは、次のようになります。</p> <p>DependsOn = "[ResourceType]ResourceName"</p>
Ensure	<p>管理対象オブジェクトが存在しているかどうかを示します。このプロパティを <b>Absent</b> に設定します。Absent に設定すると、リソースは対応する MO を削除します。管理対象オブジェクトが存在していることを示すには、<b>Present</b> に設定します。デフォルトは <b>Present</b> に設定されます。</p>
ModifyPresent	<p>管理対象オブジェクトがすでに存在している場合に、<b>Action</b> は <b>Add</b> に設定されることを示します。これらの既存のオブジェクトは変更できません。</p>
PropertyMap	<p>管理対象オブジェクトのプロパティを <b>keyValue</b> ペアとして指定します。</p> <p><b>シンタックス:</b></p> <p>`&lt;key1&gt;=&lt;value1&gt;` `&lt;key2&gt;=&lt;value2&gt;`</p>
WebProxyCredentials	<p>web プロキシの資格情報を示します。</p>

### 例

次の例は、UcsManagedObject リソースを使用して、組織の CiscoTest が Dn org-root に存在することを確認する方法を示しています。

#### Configuration UcsManagedObjectDemo

```
{
param(
[Parameter(Mandatory=$true)]
[PsCredential] $ucsCredential,

[Parameter(Mandatory=$true)]
[string] $ucsConnectionString
)
Node "localhost"
{
UcsManagedObject AddManagedObject
```

```

{
  Ensure = "Present"
  ModifyPresent = $true
  ClassId= "orgOrg"
  Dn = "org-root/org-CiscoTest"
  PropertyMap= "Descr = test org for UCS DSC `nName = CiscoTest"
  UcsCredentials = $ucsCredential
  UcsConnectionString = $ucsConnectionString
  Identifier = "1"
}
}
}
}

```

## UCS DSC UcsSyncMoWithReference リソース

Cisco UCS Manager PowerTool DSC の UcsSyncMoWithReference リソースは、UCS Manager ドメインの参照から設定を同期するメカニズムを提供します。複数の UCS Manager ドメインを指定して、UCS Manager ドメインの参照から同期できます。

### Syntax

```

UcsSyncMoWithReference [string] #ResourceName
{
  Identifier = [string]
  RefUcsConnectionString = [string]
  RefUcsCredentials = [PSCredential]
  UcsConnectionString = [string]
  UcsCredentials = [PSCredential]
  [ ClassId = [string] ]
  [ DeleteNotPresent = [bool] ]
  [ DependsOn = [string[]] ]
  [ Dn = [string] ]
  [ Ensure = [string] { Absent | Present } ]
  [ Hierarchy = [bool] ]
  [ WebProxyCredentials = [PSCredential] ]
}

```

プロパティ	説明
ID	DSC リソースの一意の ID を指定します。
RefUcsConnectionString	UCS Manager の参照用接続文字列を指定します。  Syntax Name=<ipAddress> [ `nNoSsl=<bool>][ `nPort=<ushort>] [ `nProxyAddress=<proxyAddress>] [ `nUseProxyDefaultCredentials=<bool>]
RefUcsCredentials	UCS Manager の参照にアクセスするために必要な資格情報を示します。

プロパティ	説明
UcsConnectionString	同期操作が実行されるターゲットの UCS Manager の接続文字列を指定します。  構文：  Name=<ipAddress> [`\nNoSsl=<bool>][`\nPort=<ushort>] [`\nProxyAddress=<proxyAddress>] [`\nUseProxyDefaultCredentials=<bool>]
UcsCredentials	同期操作が実行されるターゲットの UCS Manager にアクセスするために必要な資格情報を示します。
ClassId	管理対象オブジェクトのクラス ID を指定します。指定されると、「Dn」および「階層」プロパティが無視されます。
DeleteNotPresent	指定されると、参照 UCS の欠落している MO が削除されます。
DependsOn	当該リソースを設定する前に、別のリソースの設定を実行する必要があることを示します。たとえば、最初に実行するリソース設定スクリプトブロックの ID が ResourceName、そのタイプが ResourceType であるとする、このプロパティを使用するシンタックスは、次のようになります。  DependsOn = "[ResourceType]ResourceName"
Dn	管理対象オブジェクトの Dn を指定します。階層プロパティは、Dn プロパティと組み合わせて使用されます。Dn プロパティは、ClassId プロパティが指定されている場合は無視されません。したがって、階層と組み合わせて ClassId または Dn のいずれかをいつでも指定できます。
Ensure	SyncMoWithReference 操作が実行されるかどうかを示します。Present SyncMoWithReference 操作を実行するには、これを Present に設定します。デフォルトは Present です。
階層	指定された Dn のすべての子 MO が同期されているかどうかを示します。Dn が指定したときに動作します。

プロパティ	説明
WebProxyCredentials	web プロキシの資格情報を示します。

### 例

次の例は、UcsSyncMoWithReference リソースを使用して、「システム/ldap-ext」をもつ MO とそのすべての子 MO を同期する方法を示します。

```
Configuration SyncMoWithReferenceResourceDemo
{
  param(
    [Parameter(Mandatory=$true)]
    [PsCredential] $ucsCredential,

    [Parameter(Mandatory=$true)]
    [PsCredential] $RefUcsCredential
  )
  Import-DSCResource -ModuleName Cisco.Ucs.DesiredStateConfiguration
  Node "localhost"
  {
    UcsSyncMoWithReference "sync1"
    {
      UcsCredentials = $ucsCredential
      UcsConnectionString = "Name=10.65.183.5"
      RefUcsCredentials = $RefUcsCredential
      RefUcsConnectionString = "Name=10.105.214.231"
      Ensure="Present"
      Identifier = "1"
      DeleteNotPresent=$true
      Hierarchy=$true
      Dn="sys/ldap-ext"
    }
  }
}
```

## UCS DSC UcsSyncFromBackup リソース

UCS Manager PowerTool DSC で UcsSyncFromBackup リソースは、UCS Manager のバックアップファイルから1つ以上の UCS Manager ドメインへの設定を適用するためのメカニズムを提供します。

### Syntax

```
UcsSyncFromBackup [string] #ResourceName
{
  Identifier = [string]
  LiteralPath = [string]
  UcsConnectionString = [string]
  UcsCredentials = [PSCredential]
  [ DependsOn = [string[]] ]
}
```

```
[ Ensure = [string] { Absent | Present } ]
[ Merge = [bool] ]
[ WebProxyCredentials = [PSCredential] ]
}
```

プロパティ	説明
ID	DSC リソースの一意の ID を指定します。
UcsConnectionString	UCS Manager の接続文字列を指定します。 <b>構文 :</b> Name=<ipAddress> [ `nNoSsl=<bool> ] [ `nPort=<ushort> ] [ `nProxyAddress=<proxyAddress> ] [ `nUseProxyDefaultCredentials=<bool> ]
UcsCredentials	UCS Manager にアクセスするために必要な資格情報を示します。
DependsOn	当該リソースを設定する前に、別のリソースの設定を実行する必要があることを示します。たとえば、最初に実行するリソース設定スクリプトブロックの ID が <b>ResourceName</b> 、そのタイプが <b>ResourceType</b> であるとする、このプロパティを使用するシンタックスは、次のようになります。  DependsOn = "[ResourceType]ResourceName"
Ensure	バックアップから同期が実行されるかどうかを示します。デフォルトは <b>Present</b> です。
Merge	バックアップ設定ファイルの情報が既存の設定情報と比較されているかどうかを示します。重複する場合、Cisco UCS ドメインの情報をバックアップ設定ファイルの情報で上書きします。
WebProxyCredentials	web プロキシの資格情報を示します。

### 例

次の例は、UcsSyncFromBackup リソースを使用して、バックアップ ファイル UcsConfigSystem から同期する方法を示しています。

#### Configuration UcsSyncFromBackupDemo

```
{
param (
[Parameter (Mandatory=$true)]
```

```

[PsCredential] $ucsCredential
)
Import-DSCResource -ModuleName Cisco.Ucs.DesiredStateConfiguration

$user = "ru44\administrator"
$password = "<password>" | ConvertTo-SecureString -AsPlainText -Force
$credential = New-Object System.Management.Automation.PSCredential($user,$password)

Node "localhost"
{
File BackupFileInstance
{
SourcePath = "\\sharedServer\Configs\UcsConfigSystem.xml"
DestinationPath = "C:\BackupFiles\UcsConfigSystem.xml"
Credential = $credential
}

UcsSyncFromBackup SyncFromBackupInstance
{
Ensure = "Present"
LiteralPath = "C:\BackupFiles\UcsConfigSystem.xml"
UcsCredentials = $ucsCredential
UcsConnectionString = "Name=10.65.183.5"
Merge = $true
DependsOn = "[File]BackupFileInstance"
Identifier = "1"
}
}
}

```

## UCS DSC UcsScript リソース

UCS Manager PowerTool DSC で UcsScript リソースは、UCS Manager PowerTool コマンドレットを実行するためのメカニズムを提供します。

### 構文

```

UcsScript [string] #ResourceName
{
Dn = [string]
Identifier = [string]
Script = [string]
UcsConnectionString = [string]
UcsCredentials = [PSCredential]
[ Action = [string] { Add | Set } ]
[ DependsOn = [string[]] ]
[ Ensure = [string] { Absent | Present } ]
[ ModifyPresent = [bool] ]
[ WebProxyCredentials = [PSCredential] ]
}

```

### Properties

プロパティ	説明
Dn	管理対象オブジェクトの Dn を指定します。

プロパティ	説明
ID	DSC リソースの一意的 ID を指定します。
Script	PowerTool コマンドレットのセットを指定します。新しいコマンドレットのプレフィクスとして `n` を使用します。
UcsConnectionString	UCS Manager の接続文字列を指定します。 構文名 = < ipAddress > [' nNoSsl = < bool >] [' nPort = < ushort >] [' nProxyAddress = < proxyAddress >] [' nUseProxyDefaultCredentials = < bool >]
UcsCredentials	UCS Manager にアクセスするために必要な資格情報を示します。
操作	管理対象オブジェクトに対して実行するアクションを指定します。管理対象オブジェクトを追加するには、このプロパティを Add に設定します。既存の管理対象オブジェクトを変更するには Set に設定します。
DependsOn	当該リソースを設定する前に、別のリソースの設定を実行する必要があることを示します。たとえば、最初に実行するリソース設定スクリプトブロックの ID が ResourceName、そのタイプが ResourceType であるとする、このプロパティを使用するシンタックスは、次のようになります。  DependsOn = "[ResourceType]ResourceName"
Ensure	スクリプトが実行されているかどうかを示します。デフォルトは Present です。
ModifyPresent	管理対象オブジェクトがすでに存在して、アクションが Add に設定されている場合、既存のオブジェクトを変更します。
WebProxyCredentials	web プロキシの資格情報を示します。

## 例

```
Configuration UcsScriptResourceDemo
{
    param(
        [Parameter(Mandatory=$true)]
```

```
[PsCredential] $ucsCredential,
[Parameter(Mandatory=$true)]
[string] $ucsConnectionString
)
Import-DSCResource -ModuleName Cisco.Ucs.DesiredStateConfiguration
Node "localhost"
{
  UcsScript ResourceInstance
  {
    Ensure = "Present"
    Dn = "org-root/ls-sp_3"
    Script = "Get-UcsOrg -Level root | Get-UcsServiceProfile -Name 'sp3' -LimitScope |
Rename-UcsServiceProfile -NewName 'sp_3' "
    UcsCredentials = $ucsCredential
    UcsConnectionString = $ucsConnectionString
    Identifier ="2"
  }
}
}
```

## UCS サーバの取得

リリース 1.4.1 からは、新しい `Get-UcsServer` コマンドレットを使用することで、フォーク ファクタにかかわらずすべてのサーバを取得できるようになりました。

## Org

次のコマンドレットを使用して、デフォルト UCS リストに、Cisco UCS ドメイン全体で使用する `Org` のリストを取得します。

```
Get-UcsOrg | select Ucs, Name, Dn
```

次のコマンドレットを使用して、ルートレベルの組織へのハンドルを取得します。

```
Get-UcsOrg -Level root
```

次のコマンドレットを使用して、UCS に 5 組織を追加します。

```
1..5 | % { Add-UcsOrg -Ucs <handle or name> qwerty$_ }
```

## 障害

次のコマンドレットを使用して、障害を取得し、重大度でそれらをグループ化します。

```
Get-UcsFault | group Severity
```

次のコマンドレットを使用して、重大な障害を取得します。

```
Get-UcsFault -Severity critical | select Ucs, Dn, Cause
```

次のコマンドレットを使用して、すべての未承認の障害を承認します。

```
Get-UcsFault | ? {$_.Ack -eq "no" } | Acknowledge-UcsFault
```

## Get コマンドレット -Hierarchy フラグ

次のコマンドレットを使用して、すべての子を含む管理対象オブジェクトを取得します。

```
Get-UcsServiceProfile -Name sp_name -Hierarchy
```

## Get コマンドレット -LimitScope フラグ

次のコマンドレットを使用して、org の子ルートに引き継がずに、ルート レベルでサービスプロファイルを取得します。

```
Get-UcsServiceProfile -Name sp_name -LimitScope
```

次のコマンドレットを使用して、Org Finance の子までたどらずに Org Finance からサービスプロファイルを取得します。

```
Get-UcsOrg -Name Finance | Get-UcsServiceProfile -Name sp_name -LimitScope
```

次のコマンドレットを使用して、LanCloud からの VLAN を取得します。

```
Get-UcsLanCloud | Get-UcsVlan -LimitScope
```

## トランザクション サポート

次のコマンドレットを使用して、トランザクションを開始します。

```
Start-UcsTransaction
```

アクションを実行します。

....

次のコマンドレットを使用して、トランザクションを終了します。

```
Complete-UcsTransaction
```

次のコマンドレットを使用して、トランザクションを元に戻します。

```
Undo-UcsTransaction
```

## VLAN の作成と削除

Cisco UCS ドメインの VLAN は名前で参照され、VLAN 定義は、MIT の下の 4 ノードに作成できます。

ノード	説明
LanCloud	これはグローバル VLAN で、両方の FI に適用できます。
FabricLanCloud	これはファブリック特定の VLAN で、ファブリック A またはファブリック B に適用できます。
ApplianceCloud	これはグローバル VLAN で、両方の FI に適用できます。
FabricApplianceCloud	これはファブリック A またはファブリック B に適用できるファブリック特定の VLAN で、アプライアンス ポートの設定中に使用されます。

次のコマンドレットを使用して、グローバルの LAN クラウドの下で VLAN を作成します。

```
Get-UcsLanCloud | Add-UcsVlan -Name lan_cloud_vlan -Id 500
```

次のコマンドレットを使用して、ファブリック A LAN クラウドの下で VLAN を作成します。

```
Get-UcsFiLanCloud -Id A | Add-UcsVlan -Name fi_a_vlan -Id 500
```

次のコマンドレットを使用して、ファブリック B LAN クラウドの下で VLAN を作成します。

```
Get-UcsFiLanCloud -Id B | Add-UcsVlan -Name fi_b_vlan -Id 500
```

次のコマンドレットを使用して、グローバル アプライアンス クラウドの下で VLAN を作成します。

```
Get-UcsApplianceCloud | Add-UcsVlan -Name appliance_vlan -Id 500
```

次のコマンドレットを使用して、ファブリック A アプライアンス クラウドの下で VLAN を作成します。

```
Get-UcsFabricApplianceCloud -Id A | Add-UcsVlan -Name fi_a_appliance_vlan -Id 500
```

次のコマンドレットを使用して、ファブリック B アプライアンス クラウドの下で VLAN を作成します。

```
Get-UcsFabricApplianceCloud -Id B | Add-UcsVlan -Name fi_b_appliance_vlan -Id 500
```

VLAN のリストを CSV ファイルからインポートし、LAN クラウドで VLAN を作成します（この例では CSV ファイルを作成します）。

次のコマンドレットを使用して、1 つのデバイスで VLAN を作成します。

```
$("Name,Id";foreach ($vlan in 501..550) { "vlan${vlan},${vlan}" }) > C:\work\Demo\vlans.csv
$lc=(Get-UcsLanCloud)
$lc | Get-UcsVlan | select Ucs, Name, Id
Start-UcsTransaction
import-csv C:\work\Demo\vlans.csv | % {$lc | Add-UcsVlan -Name $_.Name -Id $_.Id }
Complete-UcsTransaction
$lc | Get-UcsVlan | select Ucs, Name, Id
```

次のコマンドレットを使用して、追加された VLAN を削除します。

```
$lc | Get-UcsVlan | ? {$_.Id -ge 501 -and $_.Id -le 550 } | Remove-UcsVlan -Force
```

## MAC プールおよびブロック

次のコマンドレットを使用して、MAC プールのデフォルトに MAC ブロックを追加します。

```
Get-UcsMacPool -Ucs <handle or name> default | Add-UcsMacMemberBlock 00:25:B5:00:A0:00
00:25:B5:00:A0:08
```

次のコマンドレットを使用して、デフォルト リストのすべての Cisco UCS ドメインにわたる MAC プール割り当ての任意の競合をチェックします。

```
Get-UcsMacPoolAddr | group Id | where {$_.Count -ne 1 } | select -ExpandProperty Group
| select Ucs, Id, Assigned, AssignedToDn
```

## サーバ プール

次のコマンドレットを使用して、サーバ プールを作成します。

```
$server_pool = Add-UcsServerPool -Name server_pool
```

次のコマンドレットを使用して、サーバ プールにブレード 1/4 を追加します。

```
$server_pool | Add-UcsComputePooledSlot -ChassisId 1 -SlotId 4
```

ラック 1 を次のコマンドレットを使用して、サーバ プールに追加します。

```
$server_pool | Add-UcsComputePooledRackUnit -Id 1
```

次のコマンドレットを使用して、サーバのプールを削除します。

```
$server_pool | Remove-UcsServerPool
```

## UUID 接尾辞のプールおよびブロック

次のコマンドレットを使用して、UUID 接尾辞プールを作成します。

```
$uuid_pool = Add-UcsUuidSuffixPool -Name uuid_pool -Prefix 3864EB9A-89A2-11DF
```

次のコマンドレットを使用して、接尾辞プールに UUID 接尾辞のブロックを追加します。

```
$uuid_pool | Add-UcsUuidSuffixBlock -From 0000-000000000001 -To 0000-00000000002C
```

次のコマンドレットを使用して、UUID 接尾辞プールを削除します。

```
$uuid_pool | Remove-UcsUuidSuffixPool
```

## WWNN プールおよびブロック

次のコマンドレットを使用して、UCS でのすべての WWNN プールを取得します。

```
Get-UcsWwnPool -Purpose node-wnn-assignment
```

次のコマンドレットを使用して、WWNN プールを作成します。

```
$wwnn_pool = Add-UcsWwnPool -Name wwnn_pool -Purpose node-wnn-assignment
```

次のコマンドレットを使用して、WWNN プールに WWN ブロックを追加します。

```
$wwnn_pool | Add-UcsWwnMemberBlock -From 20:00:00:24:B5:00:00:00 -To  
20:00:00:24:B5:00:00:09
```

次のコマンドレットを使用して、WWNN プールに WWNN イニシエータを追加します。

```
$wwnn_pool | Add-UcsWwnInitiator -Id 20:00:00:25:B5:00:00:2C -Name wwnn_initiator
```

次のコマンドレットを使用して、WWNN イニシエータを削除します。

```
$wwnn_pool | Get-UcsWwnInitiator -Id 20:00:00:25:B5:00:00:2C | Remove-UcsWwnInitiator
```

次のコマンドレットを使用して、WWNN プールを削除します。

```
$wwnn_pool | Remove-UcsWwnPool
```

## WWNN プールおよびブロック

次のコマンドレットを使用して、UCS でのすべての WWPN プールを取得します。

```
Get-UcsWwnPool -Purpose port-wnn-assignment
```

次のコマンドレットを使用して、WWPN プールを作成します。

```
$swpn_pool = Add-UcsWwnPool -Name wwpn_pool -Purpose port-wwn-assignment
```

次のコマンドレットを使用して、WWPN プールに WWN ブロックを追加します。

```
$swpn_pool | Add-UcsWwnMemberBlock -From 20:00:00:24:B5:00:00:00 -To  
20:00:00:24:B5:00:00:09
```

次のコマンドレットを使用して、WWPN プールに WWPN イニシエータを追加します。

```
$swpn_pool | Add-UcsWwnInitiator -Id 20:00:00:25:B5:00:00:2D -Name wwpn_initiator
```

次のコマンドレットを使用して、WWPN イニシエータの概要を設定します。

```
$swpn_pool | Get-UcsWwnInitiator -Id 20:00:00:25:B5:00:00:2D | Set-UcsWwnInitiator -Descr  
"WWPN initiator modified"
```

次のコマンドレットを使用して、WWPN プールを削除します。

```
$swpn_pool | Remove-UcsWwnPool
```

## IQN 接尾辞のプールおよびブロック

次のコマンドレットを使用して、UCS の IQN プールを取得します。

```
Get-UcsIqnPoolPool -Name iqnSuffixPool
```

次のコマンドレットを使用して、IQN プールを作成します。

```
$iqn_pool = Get-UcsOrg -Level root | Add-UcsIqnPoolPool -Name iqn_pool -Prefix I
```

次のコマンドレットを使用して、IQN プールブロックを作成します。

```
$iqn_pool_block =  
$iqn_pool | Add-UcsIqnPoolBlock -Suffix B -From 0 -To 10
```

次のコマンドレットを使用して、IQN プールブロックを削除します。

```
$iqn_pool_block | Remove-UcsIqnPoolBlock
```

次のコマンドレットを使用して、IQN プールを削除します。

```
$iqn_pool | Remove-UcsIqnPoolPool
```

## ポート ロール

次のコマンドレットを使用して、ファブリック A のスロット 1 (固定ポートスロット) ポート 19 をサーバポートにします。

```
Get-UcsFabricServerCloud -Id A | Add-UcsServerPort -PortId 19 -SlotId 1
```

次のコマンドレットを使用して、ファブリック A のスロット 1（固定ポートスロット）ポート 19 をサーバポートにしないように設定解除します。

```
Get-UcsFabricServerCloud -Id A | Get-UcsServerPort -PortId 19 -SlotId 1 |
Remove-UcsServerPort -Force
```

次のコマンドレットを使用して、ファブリック A のスロット 1（固定ポートスロット）ポート 15 をアプライアンスポートにします。

```
Get-UcsFabricApplianceCloud -Id A | Add-UcsAppliancePort -PortId 15 -SlotId 1
```

次のコマンドレットを使用して、ファブリック A のスロット 1（固定ポートスロット）ポート 15 をアプライアンスポートにしないように設定解除します。

```
Get-UcsFabricApplianceCloud -Id A | Get-UcsAppliancePort -PortId 15 -SlotId 1 |
Remove-UcsAppliancePort -Force
```

次のコマンドレットを使用して、ファブリック A のスロット 1（固定ポートスロット）ポート 16 をアップリンクポートにします。

```
Get-UcsFiLanCloud -Id A | Add-UcsUplinkPort -PortId 16 -SlotId 1
```

次のコマンドレットを使用して、ファブリック A のスロット 1（固定ポートスロット）ポート 16 をアップリンクポートにしないように設定解除します。

```
Get-UcsFiLanCloud -Id A | Get-UcsUplinkPort -PortId 16 -SlotId 1 | Remove-UcsUplinkPort
-Force
```

## ポートチャネル

次のコマンドレットを使用して、ポート 19 および 20 のあるファブリック A にアプライアンスポートチャネルを作成します。

```
$ap_pc = Get-UcsFabricApplianceCloud -Id A | Add-UcsAppliancePortChannel -PortId 55
$ap_pc | Add-UcsAppliancePortChannelMember -SlotId 1 -PortId 19
$ap_pc | Add-UcsAppliancePortChannelMember -SlotId 1 -PortId 20
```

次のコマンドレットを使用して、アプライアンスの VLAN にポートチャネルを追加します。

```
Get-UcsApplianceCloud | Get-UcsVlan -Name ApplianceVlan | Add-UcsVlanMemberPortChannel
-SwitchId A -PortId $ap_pc.PortId
```

次のコマンドレットを使用して、アプライアンス VLAN からポートチャネルを削除します。

```
Get-UcsApplianceCloud | Get-UcsVlan -Name ApplianceVlan | Get-UcsVlanMemberPortChannel
-SwitchId A -PortId 55 | Remove-UcsVlanMemberPortChannel -Force
```

次のコマンドレットを使用して、アプライアンスポートチャネルを削除します。

```
Get-UcsFabricApplianceCloud -id A | Get-UcsAppliancePortChannel -PortId 55 |
Remove-UcsAppliancePortChannel -Force
```

## VLAN の割り当て

次のコマンドレットを使用して、アプライアンス VLAN にアプライアンスのポート A/1/15 を追加します。

```
Get-UcsApplianceCloud | Get-UcsVlan -name ApplianceVlan | Add-UcsVlanMemberPort -SwitchId A -SlotId 1 -PortId 15
```

次のコマンドレットを使用して、アプライアンス VLAN からアプライアンスのポート A/1/15 を削除します。

```
Get-UcsApplianceCloud | Get-UcsVlan -name ApplianceVlan | Get-UcsVlanMemberPort -SwitchId A -SlotId 1 -PortId 15 | Remove-UcsVlanMemberPort -Force
```

## ブレードの電力および温度の統計情報

次のコマンドレットを使用して、すべてのブレードの電力統計を表示します。

```
Get-UcsBlade | Get-UcsComputeBoard | Get-UcsComputeMbPowerStats | Out-GridView
```

次のコマンドレットを使用して、すべてのブレードの温度の統計情報を表示します。

```
Get-UcsBlade | Get-UcsComputeBoard | Get-UcsComputeMbTempStats | Out-GridView
```

## コンフィギュレーションのバックアップ

UCS の前に保存したバックアップを削除します。

```
Get-UcsMgmtBackup | Remove-UcsMgmtBackup
```

**PathPattern** は自動入力され、コマンドレットを複数の Cisco UCS ドメインで使うことができます。UCS の完全な状態のシステムバックアップを作成し、ダウンロードします。これにより、システム全体のスナップショットを含むバイナリファイルを作成します。このバックアップにより生成されたファイルを使用して、ディザスタリカバリ時にシステムを復元できます。このファイルにより、元のファブリックインターコネクト上で設定を復元または再構築できます。また、別のファブリックインターコネクト上で設定を再現することもできます。このファイルは、インポートには使用できません。

```
Backup-Ucs -Type full-state -PathPattern 'C:\Backups\${ucs}-${yyyy}${MM}${dd}-${HH}${mm}-full-state.tar.gz'
```

UCS の論理バックアップを作成し、ダウンロードします。これにより、サービス プロファイル、VLAN、VSAN、プール、ポリシーなどのすべての論理設定が含まれる XML ファイルが作成されます。このバックアップにより生成されたファイルを使用して、これらの設定を元の

ファブリック インターコネクトまたは別のファブリック インターコネクトにインポートできません。このファイルは、システムの復元には使用できません。

```
Backup-Ucs -Type config-logical -PathPattern
'C:\Backups\${ucs}-${yyyy}${MM}${dd}-${HH}${mm}-config-logical.xml'
```

UCS のシステム バックアップを作成し、ダウンロードします。これにより、ユーザ名、ロール、ロケールなどのすべてのシステム設定を含む XML ファイルが作成されます。このバックアップにより生成されたファイルを使用して、これらの設定を元のファブリック インターコネクトまたは別のファブリック インターコネクトにインポートできます。このファイルは、システムの復元には使用できません。

```
Backup-Ucs -Type config-system -PathPattern
'C:\Backups\${ucs}-${yyyy}${MM}${dd}-${HH}${mm}-config-system.xml'
```

UCS のすべてのコンフィギュレーションのバックアップを作成し、ダウンロードします。これによって、すべてのシステムおよび論理設定を含む XML ファイルが作成されます。このバックアップにより生成されたファイルを使用して、これらの設定を元のファブリック インターコネクトまたは別のファブリック インターコネクトにインポートできます。このファイルは、システムの復元には使用できません。このファイルには、ローカル認証されたユーザのパスワードは含まれません。

```
Backup-Ucs -Type config-all -PathPattern
'C:\Backups\${ucs}-${yyyy}${MM}${dd}-${HH}${mm}-config-all.xml'
```

## インポート設定

インポート機能は、すべてのコンフィギュレーション ファイル、システム コンフィギュレーション ファイル、および論理コンフィギュレーション ファイルで使用できます。インポートは、システムがアップ状態で、稼働中に実行できます。

すべてのコンフィギュレーション xml（すべてのシステムおよび論理設定を含む XML ファイル）をインポートします。現在のコンフィギュレーション情報が、インポートされたコンフィギュレーション ファイルの情報で一度に 1 つのオブジェクトについて置き換えられます。

```
Import-UcsBackup -LiteralPath 'C:\Backups\config-all.xml'
```

すべてのコンフィギュレーション xml をインポートします。インポートされたコンフィギュレーションファイルの情報が既存の設定情報と比較されます。矛盾が存在する場合、インポートされたコンフィギュレーションファイルの情報で Cisci UCS ドメインの情報が上書きされません。

```
Import-UcsBackup -LiteralPath 'C:\Backups\config-all.xml' -Merge
```

## 管理対象オブジェクトの同期

次のコマンドレットを使用して、SYSB に SYSA から MOs の設定を同期します。

```
Sync-UcsManagedobject -Ucs SYSB (Compare-UcsManagedObject (Get-UcsOrg -ucs SYSB)
(Get-UcsOrg -ucs SYSA)) -whatif
Sync-UcsManagedobject -Ucs SYSB (Compare-UcsManagedObject (Get-UcsOrg -ucs SYSB)
(Get-UcsOrg -ucs SYSA)) -Force
```

次のコマンドレットを使用して、一連の MO を SYSA からデフォルト リストのすべてのシステムに同期します。

```
Get-UcsPSSession | % {Sync-UcsManagedobject -Ucs $_ (Compare-UcsManagedObject (Get-UcsOrg
-ucs $_) (Get-UcsOrg -ucs SYSA)) -Force}
```

## UCS 管理対象オブジェクト移行の監視

Cisco UCS ドメインのすべてのイベントを 60 秒間監視します。

```
Watch-Ucs -TimeoutSec 60
```

Cisco UCS ドメインのエラーになったすべての変更を 60 秒間監視します。

```
Watch-Ucs -TimeoutSec 60 -ClassId FaultInst
```

MO の特定のフィールドが成功値になるまで UCS を監視します。

```
Send-UcsFirmware -LiteralPath C:\work\Images\ucs-k9-bundle-b-series.1.4.2b.B.bin |
Watch-Ucs -Property TransferState -SuccessValue downloaded -PollSec 30 -TimeoutSec 600
```

## テクニカル サポート

全体の UCS Manager インスタンスのテクニカル サポート データが指定のファイルに作成およびダウンロードされます。

```
Get-UcsTechSupport -PathPattern 'C:\${ucs}-techsupp-ucsm.tar' -UcsManager -RemoveFromUcs
-TimeoutSec 600
```

UCS Manager 管理サービス (ファブリック インターコネクトを除く) のテクニカル サポート データが指定のファイルに作成およびダウンロードされます。

```
Get-UcsTechSupport -PathPattern 'C:\${ucs}-techsupp-ucsmgmt.tar' -UcsMgmt -RemoveFromUcs
-TimeoutSec 600
```

シャーシ ID 1 および CIMC ID 1 のテクニカル サポート データが指定のファイルに作成およびダウンロードされます。

```
Get-UcsTechSupport -PathPattern 'C:\${ucs}-techsupp-chassis.tar' -RemoveFromUcs -TimeoutSec
600 -ChassisId 1 -CimcId 1
```

シャーシ ID 1 および IOM ID 1 のテクニカル サポート データが指定のファイルに作成およびダウンロードされます。

```
Get-UcsTechSupport -PathPattern 'C:\${ucs}-techsupp-iom.tar' -RemoveFromUcs -TimeoutSec 600 -ChassisId 1 -IomId 1
```

ラック サーバ ID 1 およびラック アダプタ ID 1 のテクニカル サポート データが指定のファイルに作成およびダウンロードされます。

```
Get-UcsTechSupport -PathPattern 'C:\${ucs}-techsupp-rack.tar' -RemoveFromUcs -TimeoutSec 600 -RackServerId 1 -RackAdapterId 1
```

FEXID1 のテクニカル サポート データが指定のファイルに作成およびダウンロードされます。

```
Get-UcsTechSupport -PathPattern 'C:\${ucs}-techsupp-fex.tar' -RemoveFromUcs -TimeoutSec 600 -FexId 1
```

## サービス プロファイル

UCS のすべてのサービス プロファイル インスタンスを取得します。

```
Get-UcsServiceProfile -Type instance
```

UCS のすべてのサービス プロファイルのアップデート テンプレートを取得します。

```
Get-UcsServiceProfile -Type updating-template
```

UCS のサービス プロファイルの初期テンプレートを取得します。

```
Get-UcsServiceProfile -Type initial-template
```

サービス プロファイル テンプレートの `sp_template` から新しいサービス プロファイルの `sp_name` を追加します。

```
Add-UcsServiceProfile -SrcTemplName sp_template -Name sp_name
```

サービス プロファイルを追加します。

```
Add-UcsServiceProfile -Name sp_name -BootPolicyName boot_policy -BiosProfileName bios_policy -HostFwPolicyName 1.4-3i -MgmtFwPolicyName 1.4-3i -MaintPolicyName maint_policy -MgmtAccessPolicyName ipmi_policy -PowerPolicyName power_policy -ScrubPolicyName scrub_policy -SolPolicyName sol_policy -StatsPolicyName stats_policy -AgentPolicyName agent_policy -DynamicConPolicyName vNIC_policy -ExtIPState static -IdentPoolName UUID_pool -LocalDiskPolicyName disk_policy -Uuid "00000000-0000-0000-0000-000000000008" -UsrLbl "serviceprofile"
```

サービス プロファイルの電源状態を取得します。

```
Get-UcsServiceProfile -Name sp_name | Get-UcsServerPower
```

サービス プロファイルをテンプレートにバインドします。

```
Get-UcsServiceProfile -Name sp_name | Set-UcsServiceProfile -SrcTemplName sp_template
```

サービス プロファイルを削除します。

```
Get-UcsServiceProfile -Name sp_name | Remove-UcsServiceProfile
```

## サービス プロファイルのコンポーネント

次のコマンドレットを使用して、サービス プロファイルを作成します。

```
$sp = Add-UcsServiceProfile -Name sp_name
```

次のコマンドレットを使用して、QoS ポリシーの参照付きで vNIC を作成します。

```
$eth0 = $sp | Add-UcsVnic -Name eth0 -QosPolicyName qos_policy
```

次のコマンドレットを使用して、vNIC の VLAN を追加し、それをネイティブ VLAN にします。

```
$eth0 | Add-UcsVnicInterface -Name fi_a_vlan -DefaultNet true
```

次のコマンドレットを使用して、vNIC での VLAN を追加します。

```
$eth0 | Add-UcsVnicInterface -Name fi_b_vlan
```

次のコマンドレットを使用して、テンプレートから vNIC を作成し、インスタンス化します。

```
$sp | Add-UcsVnic -Name eth1 -NwTemplName vnic_template
```

次のコマンドレットを使用して、vHBA を作成します。

```
$fc0 = $sp | Add-UcsVhba -Name fc0 -IdentPoolName wwpn_pool
```

次のコマンドレットを使用して、vHBA の VSAN を追加します。

```
$fc0 | Set-UcsVhbaInterface -Name fi_b_vsan
```

次のコマンドレットを使用して、WWNN プールを追加します。

```
$sp | Add-UcsVnicFcNode -IdentPoolName node_default
```

## サービス プロファイルのアソシエーション

次のコマンドレットを使用して、ブレードをサービス プロファイルに関連付けます。

```
Get-UcsServiceProfile sp_name -LimitScope | Associate-UcsServiceProfile -Blade  
(Get-UcsBlade -Chassis 1 -SlotId 1)
```

次のコマンドレットを使用して、ラックにサービス プロファイルに関連付けます。

```
Get-UcsServiceProfile sp_name -LimitScope | Associate-UcsServiceProfile -RackUnit  
(Get-UcsRackUnit -ServerId 1)
```

次のコマンドレットを使用して、サーバプールにサービス プロファイルに関連付けます。

```
Get-UcsServiceProfile sp_name -LimitScope | Associate-UcsServiceProfile -ServerPoolName
FileServerPool
```

次のコマンドレットを使用して、サーバプール資格ポリシーのあるサーバプールにサービス プロファイルに関連付けます。

```
Get-UcsServiceProfile sp_name -LimitScope | Associate-UcsServiceProfile
-ServerPoolName file_server_pool -ServerPoolQualificationPolicyName file_server_pool
```

次のコマンドレットを使用して、サービス プロファイルの関連付けを解除します。

```
Get-UcsServiceProfile sp_name -LimitScope | Disassociate-UcsServiceProfile
```

次のコマンドレットを使用して、サービス プロファイルのコピーを作成します。

```
Get-UcsServiceProfile -Name sp_name -LimitScope | Copy-UcsServiceProfile -NewName
copy_sp_name
```

次のコマンドレットを使用して、サービス プロファイルの名前を変更します。

```
Get-UcsServiceProfile -Name sp_name | Rename-UcsServiceProfile -NewName rename_sp_name
```

## Filters

次のコマンドレットを使用して、すべてのサービス プロファイルテンプレートを取得します。

```
Get-UcsServiceProfile -Filter 'Type -clike *-template' | select Ucs,Dn,Name
```

次のコマンドレットを使用して、文字列「SIC」が含まれる名前をもつすべてのサービス プロファイルを取得します。

```
Get-UcsServiceProfile -Filter 'Name -cmatch SJC' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、文字列「SIC」で始まる名前をもつすべてのサービス プロファイルを取得します。

```
Get-UcsServiceProfile -Filter 'Name -clike SJC' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、8~50のIDをもつすべてのVLANを取得します。

```
Get-UcsVlan -Filter 'Id -cbw 8,50' | select Ucs,Dn, Name, Id
```

次のコマンドレットを使用して、障害権限があるすべてのロールを取得します。

```
Get-UcsRole -Filter 'Priv -ccontains fault' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、障害または操作権限があるすべてのロールを取得します。

```
Get-UcsRole -Filter 'Priv -canybit fault,operations' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、障害および操作権限があるすべてのロールを取得します。

```
Get-UcsRole -Filter 'Priv -callbits fault,operations' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、45 より高い温度のブレード/ラック ユニットのリストを取得します。

```
Get-UcsProcessorEnvStats -Filter 'Temperature -cgt 45' | Get-UcsParent | Get-UcsParent | Get-UcsParent | select Ucs, Dn
```

次のコマンドレットを使用して、2012/4/18 9:00～2012/4/19 9:30 の間に生成された障害の一覧を取得します。

```
Get-UcsFault -Filter 'Created -cbw "4/18/2012 9:00","4/19/2012 9:30"' | select Ucs, Cause, Dn, Created
```

次のコマンドレットを使用して、文字列「SIC」と等しい名前をもつサービス プロファイルを取得します。

```
Get-UcsServiceProfile -Filter 'Name -ceq SJC' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、文字列「SJC/sjc/SjC」などと等しい名前をもつすべてのサービス プロファイルを取得します。

```
Get-UcsServiceProfile -Filter 'Name -ieq sjc' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、文字列「SJC/sjc/SjC」などで始まる名前をもつすべてのサービス プロファイルを取得します。

```
Get-UcsServiceProfile -Filter 'Name -ilike SJC*' | select Ucs, Dn, Name
```

次のコマンドレットを使用して、文字列「SJC/sjc/SjC」などを除く名前をもつすべてのサービス プロファイルを取得します。

```
Get-UcsServiceProfile -Filter 'Name -ine SJC' | select Ucs, Dn, Name
```

## iSCSI ブート

UCS のトランザクションを開始します。

```
Start-UcsTransaction
```

サービス プロファイルを作成します。

```
$sp = Add-UcsServiceProfile -Type instance -Name iscsiboot
```

スタティック IP **p** (iSCSI ブートに関連していない) を追加します。

```
$staticIp = Add-UcsVnicIpV4StaticAddr -ServiceProfile $sp -Addr 10.65.224.161 -DefGw 10.65.224.1 -Subnet 255.255.255.0
```

必須の vNIC を作成し、VLAN を追加します。

```
$vnic = Add-UcsVnic -ServiceProfile $sp -Name enic1 -SwitchId A -Addr 00:25:B5:07:80:00
$vlan605 = Add-UcsVnicInterface -Vnic $vnic -Name vlan605 -DefaultNet yes
```

iSCSI の vNIC を作成し、上記で作成された vNIC にマッピングします。

iSCSI イニシエータ パラメータの追加- VLAN および IP アドレス。

```
$enic = Add-UcsVnicIScsi -ServiceProfile $sp -Name iscsienic1 -InitiatorName
iqn.1995-05.com.broadcom.iscsiboot -VnicName enic1
$vlan = Add-UcsVnicVlan -VlanName vlan605 -VnicIScsi $enic
$ipv4if = Add-UcsVnicIPv4If -VnicVlan $vlan
$ipv4iscsi = Add-UcsVnicIPv4IscsiAddrp -VnicIPv4If $ipv4if -Addr 10.65.224.157
```

ターゲット パラメータを追加します。

```
$primaryTarget = Add-UcsVnicIScsiStaticTargetIf -VnicVlanp $vlan -IpAddress 10.65.224.16
-Name iqn.1992-08.com.netapp:sn.135037408 -Priority 1
$primaryLun = Add-UcsVnicLunp -VnicIScsiStaticTargetIf $primaryTarget -Id 2
```

特定のブート ポリシーを作成します。

```
$bootPolicy = Add-UcsBootDefinition -ServiceProfile $sp
```

インストールが必要な場合は、LsbootVirtualMedia を作成します。

```
$vmedia = Add-UcsLsbootVirtualMedia -BootDefinition $bootPolicy -Access read-only -Order
1
```

ブートパスに iSCSI enic を追加します。

```
$iscsiBoot = Add-UcsLsbootIScsi -BootDefinition $bootPolicy -Order 2
$iscsiBootImagePath = Add-UcsLsbootIScsiImagePath -LsbootIscsi $iscsiBoot -Type primary
-ISCSTVnicName iscsienic1
```

UCS のトランザクションを完了します。

```
Complete-UcsTransaction | Out-null
```

## vNIC テンプレート

次のコマンドレットを使用して、初期 vNIC テンプレートを作成します。

```
$vnic_init_temp = Add-UcsVnicTemplate -Name vnic_init_temp -TemplType initial-template
-SwitchId A
```

次のコマンドレットを使用して、更新 vNIC テンプレートを作成します。

```
$vnic_update_temp = Add-UcsVnicTemplate -Name vnic_update_temp -TemplType updating-template
-SwitchId B -Target adaptor
```

次のコマンドレットを使用して、初期 vNIC テンプレートに VLAN を追加します。

```
$vnic_init_temp | Add-UcsVnicInterface -Name fi_a_vlan
```

次のコマンドレットを使用して、初期 vNIC テンプレートに VLAN を追加し、それをネイティブ VLAN にします。

```
$vnic_init_temp | Add-UcsVnicInterface -Name lan_cloud_vlan -DefaultNet true
```

次のコマンドレットを使用して、初期 vNIC テンプレートに対して MAC プール、ネットワーク制御ポリシーと QoS ポリシーを設定します。

```
$vnic_init_temp | Set-UcsVnicTemplate -IdentPoolName mac_pool -NwCtrlPolicyName network_policy -QosPolicyName qos_policy
```

次のコマンドレットを使用して、初期 vNIC テンプレートを削除します。

```
$vnic_init_temp | Remove-UcsVnicTemplate
```

## vHBA テンプレート

次のコマンドレットを使用して、初期 vHBA テンプレートを作成します。

```
$vhba_init_temp = Add-UcsVhbaTemplate -Name vhba_init_temp -TemplType initial-template -SwitchId A
```

次のコマンドレットを使用して、更新 vHBA テンプレートを作成します。

```
$vhba_update_temp = Add-UcsVhbaTemplate -Name vhba_update_temp -TemplType updating-template -SwitchId B
```

次のコマンドレットを使用して、更新 vHBA テンプレートに、VSAN を追加します。

```
$vhba_update_temp | Add-UcsVhbaInterface -Name fi_b_vsan
```

次のコマンドレットを使用して、更新 vHBA テンプレートに対して WWN プール、QoS ポリシー、PIN グループ、および統計ポリシーを設定します。

```
$vhba_update_temp | Set-UcsVhbaTemplate -IdentPoolName wwpn_pool -QosPolicyName qos_policy -PinToGroupName san_pin_group -StatsPolicyName threshold_policy
```

次のコマンドレットを使用して、更新 vHBA テンプレートを削除します。

```
$vhba_update_temp | Remove-UcsVhbatemplate
```

## ブート ポリシー

次のコマンドレットを使用して、ブート ポリシーを作成し、ブート順序の変更で Reboot をイネーブルにして、vNIC/vHBA/iSCSI 名にします。

```
$boot_policy = Get-UcsOrg -Name root | Add-UcsBootPolicy -Name boot_policy -EnforceVnicName
yes -RebootOnUpdate yes
```

次のコマンドレットを使用して、フロッピー ディスクを追加します。

```
$boot_policy | Add-UcsLsbootVirtualMedia -Order 3 -Access read-write
```

次のコマンドレットを使用して、CD-ROM を追加します。

```
$boot_policy | Add-UcsLsbootVirtualMedia -Order 2 -Access read-only
```

次のコマンドレットを使用して、ローカル ディスクを追加します。

```
$boot_storage = $boot_policy | Add-UcsLsbootStorage -Order 1 $boot_storage |
Add-UcsLsbootLocalStorage
```

次のコマンドレットを使用して、SAN ブートを追加します。

```
$boot_storage | Add-UcsLsbootSanImage -VnicName fc0 -Type primary |
Add-UcsLsbootSanImagePath -Type primary -Lun 1 -Wwn 20:00:00:25:B5:00:00:00
```

次のコマンドレットを使用して、LAN ブートを追加します。

```
$boot_policy | Add-UcsLsbootLan -Order 4 | Add-UcsLsbootLanImagePath -VnicName eth0 -Type
primary
```

次のコマンドレットを使用して、ブート ポリシーを削除します。

```
$boot_policy | Remove-UcsBootPolicy
```

## アダプタ ポリシー

Receive Checksum Offload を無効にするカスタム イーサネット アダプタ ポリシーを追加します。

```
$eth_adap_policy = Add-UcsEthAdapterPolicy -Name eth_adap_policy -Descr "Custom Adapter
Policy"
$eth_adap_policy | Set-UcsEthAdapterOffloadProfile -TcpRxChecksum disabled
```

FC アダプタ ポリシーを追加します。

```
$fc_adap_policy = Add-UcsFcAdapterPolicy -Name fc_adap_policy -Descr "Fibre Channel
Adapter Policy"
```

ファイバ チャネル アダプタ ポリシーの FCP エラー回復を有効にします。

```
$fc_adap_policy | Set-UcsAdaptorFcErrorRecoveryProfile -FcpErrorRecovery enabled
```

iSCSI アダプタ ポリシーを追加します。

```
$iscsi_adap_policy = Add-UcsIScsiAdapterPolicy -Name iscsi_policy
```

iSCSI アダプタ ポリシーに対して、TCP タイムスタンプ、HBA モード、および対象へのブートをイネーブルにします。

```
$iscsi_adap_policy | Set-UcsIScsiAdapterPolicyProperties -BootToTarget yes -TcpTimeStamp  
yes -HbaMode yes
```

## BIOS ポリシー

次のコマンドレットを使用して、BIOS ポリシーを作成し、BIOS 設定変更でリブートをイネーブルにします。

```
$bios_policy = Add-UcsBiosPolicy -Name bios_policy -RebootOnUpdate yes
```

次のコマンドレットを使用して、USB システムのアイドル時の電力最適化設定を high-performance に変更します。

```
$bios_policy | Set-UcsBiosVfUSBSYSTEMIdlePowerOptimizingSetting -VpUSBIdlePowerOptimizing  
high-performance
```

次のコマンドレットを使用して、仮想化テクノロジーを有効にします。

```
$bios_policy | Set-UcsBiosVfIntelVirtualizationTechnology -VpIntelVirtualizationTechnology  
enabled
```

次のコマンドレットを使用して、BIOS ポリシーのクワイエットブートを有効にします。

```
$bios_policy | Set-UcsBiosVfQuietBoot -VpQuietBoot enabled
```

次のコマンドレットを使用して、最終状態に電力損失の AC を再開します。

```
$bios_policy | Set-UcsBiosVfResumeOnACPowerLoss -VpResumeOnACPowerLoss last-state
```

次のコマンドレットを使用して、ブート オプション再試行を無効にします。

```
$bios_policy | Set-UcsBiosVfBootOptionRetry -VpBootOptionRetry disabled
```

次のコマンドレットを使用して、コンソールリダイレクションを無効にします。

```
$bios_policy | Set-UcsBiosVfConsoleRedirection -VpConsoleRedirection disabled
```

次のコマンドレットを使用して、BIOS ポリシーを削除します。

```
$bios_policy | Remove-UcsBiosPolicy
```

## ホストファームウェアパッケージ

次のコマンドレットを使用して、ホストファームウェアパッケージを作成して、IgnoreCompCheck を No に設定します。

```
$host_firm_pack = Add-UcsFirmwareComputeHostPack -Name host_firm_pack -IgnoreCompCheck  
no
```

次のコマンドレットを使用して、ホストファームウェアパックアイテムを追加します。

```
$host_firm_pack | Add-UcsFirmwarePackItem -Type adaptor -HwModel N20-AC0002 -HwVendor "Cisco Systems Inc" -Version '1.4(1i)'
```

次のコマンドレットを使用して、ホストファームウェアパックアイテムのバージョンを設定します。

```
$host_firm_pack | Get-UcsFirmwarePackItem -HwModel N20-AC0002 | Set-UcsFirmwarePackItem -Version '2.0(1t)'
```

次のコマンドレットを使用して、ホストファームウェアパッケージを削除します。

```
$host_firm_pack | Remove-UcsFirmwareComputeHostPack
```

## IPMI アクセス プロファイル

次のコマンドレットを使用して、IPMI アクセス プロファイルを作成します。

```
$ipmi_profile= Get-UcsOrg -Name root | Add-UcsIpmiAccessProfile -Name ipmi_profile
```

次のコマンドレットを使用して、管理者の役割を持つ IPMI ユーザを追加します。

```
$ipmi_profile | Add-UcsAaaEpUser -Name ipmiUser -Priv admin
```

次のコマンドレットを使用して、IPMI ユーザのロールを変更します。

```
$ipmi_profile | Get-UcsAaaEpUser -Name ipmiUser | Set-UcsAaaEpUser -Priv readonly
```

次のコマンドレットを使用して、IPMI アクセス プロファイルを削除します。

```
$ipmi_profile | Remove-UcsIpmiAccessProfile
```

## 管理ファームウェア パッケージ

次のコマンドレットを使用して、管理ファームウェアパッケージを作成して、IgnoreCompCheck を No に設定します。

```
$mgmt_firm_pack = Add-UcsFirmwareComputeMgmtPack -Name mgmt_firm_pack -IgnoreCompCheck no
```

次のコマンドレットを使用して、管理ファームウェアパックアイテムを追加します。

```
$mgmt_firm_pack | Add-UcsFirmwarePackItem -Type blade-controller -HwModel "N20-B6620-1" -HwVendor "Cisco Systems Inc" -Version '1.4(1i)'
```

次のコマンドレットを使用して、管理ファームウェアパックアイテムのバージョンを設定します。

```
$mgmt_firm_pack | Get-UcsFirmwarePackItem -HwModel N20-B6620-1 | Set-UcsFirmwarePackItem -Version '2.0(1t)'
```

次のコマンドレットを使用して、管理ファームウェア パッケージを削除します。

```
$mgmt_firm_pack | Remove-UcsFirmwareComputeMgmtPack
```

## 電力制御ポリシー

電力制御ポリシーを作成します。優先順位は1～10の段階にランク付けされており、1が最も高い優先順位、10が最も低い優先順位を表します。デフォルトのプライオリティは5です。

```
$power_policy = get-UcsOrg -Level root | Add-UcsPowerPolicy -Name power_policy -Prio 6
```

電力の制限が「非制限」の電力制御ポリシーを作成します。

```
$power_nocap = get-UcsOrg -Name root | Add-UcsPowerPolicy -Name power_nocap -Prio no-cap
```

電力制御ポリシーを削除します。

```
$power_policy | Remove-UcsPowerPolicy
```

## サーバ プール ポリシー 資格情報

次のコマンドレットを使用して、サーバ プール ポリシー 資格情報を作成します。

```
$server_pool_qual = Add-UcsServerPoolQualification -Name server_pool_qual
```

次のコマンドレットを使用して、アダプタ資格情報を作成します。

```
$server_pool_qual | Add-UcsAdaptorQualification
```

次のコマンドレットを使用して、1067Mhzのメモリクロック速度と16このメモリユニットをもつメモリ資格情報ポリシーを作成します。

```
$server_pool_qual | Add-UcsMemoryQualification -Clock 1067 -Units 16
```

次のコマンドレットを使用して、Pentium\_4 プロセッサ アーキテクチャのCPU/コア資格情報ポリシーを作成します。

```
$server_pool_qual | Add-UcsCpuQualification -Arch Pentium_4
```

次のコマンドレットを使用して、ローカルディスクを持たないサーバ (SANのみの構成) のディスクレスストレージ資格情報ポリシーを作成します。

```
$server_pool_qual | Add-UcsStorageQualification -Diskless yes
```

次のコマンドレットを使用して、ラック資格情報を作成します。

```
$server_pool_qual | Add-UcsRackQualification -MaxId 1 -MinId 1[1]
```

次のコマンドレットを使用して、サーバプール ポリシー資格情報を削除します。

```
$server_pool_qual | Remove-UcsServerPoolQualification
```

## ダイナミック vNIC 接続ポリシー

54 のダイナミック vNIC がありフェールオーバー モードの保護がイネーブルのダイナミック vNIC 接続ポリシー dy\_vnic\_conn を作成します。

```
$dy_vnic_conn = Add-UcsDynamicVnicConnPolicy -Name dy_vnic_conn -AdaptorProfileName Linux  
-DynamicEth 54 -Protection protected
```

ダイナミック vNIC 接続ポリシーを削除します。

```
$dy_vnic_conn | Remove-UcsDynamicVnicConnPolicy
```

## ネットワーク制御ポリシー

CDP がイネーブルで、アップリンク接続がファブリック インターコネクで失われるときに vNIC の動作状態を下げるように変更するように設定された VIF を持つネットワーク制御ポリシー network\_policy を作成します。

```
$network_policy = Get-UcsOrg -Level root | Add-UcsNetworkControlPolicy -Name network_policy  
-Cdp enabled -UplinkFailAction link-down
```

次のコマンドレットを使用して、偽の MAC アドレスを許可するネットワーク制御ポリシーの Mac セキュリティを設定します。

```
$network_policy | Set-UcsPortSecurityConfig -Forge allow
```

ネットワーク制御ポリシーの Mac のセキュリティを設定します。これによって、最初のパケットをファブリック インターコネクに送信後は、他のすべてのパケットが同じ MAC アドレスを使用しなければならないか、ファブリック インターコネクによって暗黙的に拒否されます。これは、関連付けられた vNIC のポートセキュリティをイネーブルにします。

```
$network_policy | Set-UcsPortSecurityConfig -Forge deny
```

次のコマンドレットを使用して、ネットワーク制御ポリシーを削除します。

```
$network_policy | Remove-UcsNetworkControlPolicy
```

## 権限

次のコマンドレットを使用して、UCS でのすべての権限を一覧表示します。

```
Get-UcsPrivilege
```

## ユーザ ロール

次のコマンドレットを使用して、**admin** 権限を持つユーザ ロール「**test\_role**」を追加します。

```
Add-UcsRole -Name user_role -Priv admin
```

次のコマンドレットを使用して、ファブリック インターコネクトのインフラストラクチャへの読み取り/書き込みアクセス、ネットワーク セキュリティ動作、およびシステムのその他の部分への読み取りアクセスを許可するためにユーザ ロールの権限を変更します。

```
Get-UcsRole -Name user_role | Set-UcsRole -Priv ls-network
```

次のコマンドレットを使用して、**Set-UcsRole** を使用して複数の権限を設定します。

```
Get-UcsRole -Name multi_priv_role | Set-UcsRole -Priv "ls-network", "ls-qos"
```

次のコマンドレットを使用して、UCS のすべてのユーザ ロールを取得します。

```
Get-UcsRole
```

次のコマンドレットを使用して、名前によりユーザ ロールを取得します。

```
Get-UcsRole -Name multi_priv_role
```

次のコマンドレットを使用して、ユーザ ロールを削除します。

```
Get-UcsRole -Name multi_priv_role | Remove-UcsRole
```

## ロケール

次のコマンドレットを使用して、ロケールを追加します。

```
Add-UcsLocale -Name asia_pacific -Descr "Locale for Asia Pacific users"
```

次のコマンドレットを使用して、すべてのロケールを取得します。

```
Get-UcsLocale
```

次のコマンドレットを使用して、ロケールに組織を追加します。

```
Get-UcsLocale -Name asia_pacific | Add-UcsAaaOrg -Name org_asia_pacific -OrgDn org-root/org-Finance
```

次のコマンドレットを使用して、ロケールを削除します。

```
Get-UcsLocale -Name asia_pacific | Remove-UcsLocale
```

## ユーザアカウント

次のコマンドレットを使用して、ローカルユーザを追加します。

```
$user = Add-UcsLocalUser -Name jdoe -Pwd Passw0rdJdoe
```

次のコマンドレットを使用して、ローカルユーザを編集します。

```
$user | Set-UcsLocalUser -FirstName John -Lastname Doe
```

次のコマンドレットを使用して、ユーザを追加します。

```
$user | Add-UcsUserRole -Name user_role
```

次のコマンドレットを使用して、ローカルユーザを削除します。

```
Get-UcsLocalUser -Name jdoe | Remove-UcsLocalUser
```

## リモート認証 - RADIUS

次のコマンドレットを使用して、RADIUS 認証用のグローバル設定を設定します。

```
Set-UcsRadiusGlobalConfig -Descr "RADIUS authentication configuration" -Timeout 20  
-Retries 3 -Force
```

次のコマンドレットを使用して、サーバキーが「test1234」で最大リトライ回数が2回のRADIUSサーバのインスタンスを作成します。

```
Add-UcsRadiusProvider -Name "192.168.23.84" -Key test1234 -Retries 2
```

次のコマンドレットを使用して、デフォルトの認証としてRADIUSを設定します。

```
Set-UcsDefaultAuth -Realm radius
```

## リモート認証 - TACACS

次のコマンドレットを使用して、TACACS 認証用のグローバル設定を設定します。

```
Set-UcsTacacsGlobalConfig -Descr "TACACS authentication configuration" -Timeout 20  
-Retries 3
```

次のコマンドレットを使用して、TACACS プロバイダーを追加します。

```
Add-UcsTacacsProvider -Name "192.168.23.84" -Key test1234
```

次のコマンドレットを使用して、デフォルトの認証としてTACACSを設定します。

```
Get-UcsNativeAuth | Set-UcsNativeAuth -DefLogin tacacs
```

## リモート認証 - LDAP

次のコマンドレットを使用して、LDAP 認証用のグローバル設定を設定します。

```
Set-UcsLdapGlobalConfig -Descr 'LDAP authentication configuration' -Timeout 20 -Retries 3 -Force
```

次のコマンドレットを使用して、LDAP プロバイダーを追加します。

```
add-UcsLdapProvider -Attribute 'CiscoAVPair' -Basedn 'CN=Users,DC=qasamllab,DC=com' -FilterValue 'cn=$userid' -Key 'Bbv03515' -Name '10.193.23.84' -Rootdn 'CN=Administrator,CN=Users,DC=qasamllab,DC=com'
```

次のコマンドレットを使用して、デフォルトの認証と LDAP を設定します。

```
Get-UcsNativeAuth | Set-UcsNativeAuth -DefLogin ldap
```

## RADIUS プロバイダー

サーバキーが「test1234」で最大リトライ回数が2回の RADIUS サーバのインスタンスを作成します。

```
Add-UcsRadiusProvider -Name "192.168.23.84" -Key test1234 -Retries 2
```

RADIUS プロバイダー グループを追加して、デフォルトのリモート認証方式として設定します。

```
Get-UcsRadiusGlobalConfig | Add-UcsProviderGroup -Name radiusprovidergroup1  
Get-UcsProviderGroup -Name radiusprovidergroup1 | Add-UcsProviderReference -Name "192.168.23.84"  
Get-UcsNativeAuth | Set-UcsNativeAuth -DefLogin radius  
Get-UcsDefaultAuth | Set-UcsDefaultAuth -ProviderGroup radiusprovidergroup1
```

## TACACS プロバイダー

次のコマンドレットを使用して、TACACS プロバイダーを追加します。

```
Add-UcsTacacsProvider -Name "192.168.23.84" -Key test1234
```

次のコマンドレットを使用して、TACACS プロバイダー グループを追加して、デフォルトのリモート認証方式として設定します。

```
Get-UcsTacacsGlobalConfig | Add-UcsProviderGroup -Name tacacsprovidergroup1  
Get-UcsProviderGroup -Name tacacsprovidergroup1 | Add-UcsProviderReference -Name
```

```
"192.168.23.84"
Get-UcsNativeAuth | Set-UcsNativeAuth -DefLogin tacacs
Get-UcsDefaultAuth | Set-UcsDefaultAuth -ProviderGroup tacacsprovidergroup
```

## LDAP プロバイダー

次のコマンドレットを使用して、LDAP プロバイダーを追加します。

```
add-UcsLdapProvider -Attribute 'CiscoAVPair' -Basedn
'CN=Users,DC=qasamllab,DC=com' -FilterValue 'cn=$userid' -Key 'Bbv03515'
-Name '192.168.23.84' -Rootdn 'CN=Administrator,CN=Users,DC=qasamllab,DC=com'
```

次のコマンドレットを使用して、LDAP プロバイダー グループを追加して、デフォルトのリモート認証方式として設定します。

```
Get-UcsLdapGlobalConfig | Add-UcsProviderGroup -Name ldapprovidergroup1
Get-UcsProviderGroup -Name ldapprovidergroup1 | Add-UcsProviderReference -Name
"192.168.23.84"
Get-UcsNativeAuth | Set-UcsNativeAuth -DefLogin ldap
Get-UcsDefaultAuth | Set-UcsDefaultAuth -ProviderGroup ldapprovidergroup1
```

## 認証ドメイン

認証ドメインは、異なる認証方式（ローカル、TACACS+、RADIUS、および LDAP/Active Directory）とプロバイダーのグループへの同時サポートを設定します。

TACACS プロバイダーを持つ TACAS プロバイダー グループを設定します。

```
$tp = Add-UcsTacacsProvider -Name "192.168.23.84" -Key test1234
$tpg = Get-UcsTacacsGlobalConfig | Add-UcsProviderGroup -Name tacacs_pg $tpg |
Add-UcsProviderReference -Name $tp.Name
```

認証ドメインを作成し、TACACS Provider のグループへの参照を追加します。

```
$ad = Add-UcsAuthDomain -Name adtacacs
$ad | Get-UcsAuthDomainDefaultAuth | Set-UcsAuthDomainDefaultAuth -Realm
tacacs-ProviderGroup tacacs_pg
```

これで、ユーザ名「ucs-adtacacs\user」でコンソール、Web UI または XML API からユーザがログインした場合に、上記で作成した TACACS 設定が認証に使用されるようになります。

## 通信サービス

次のコマンドレットを使用して、一度にシステムにアクセスすることを許可された並列 Web セッション（web UI および xml 両方）の最大数を定義する UCS Web セッション制限を取得します。

```
Get-UcsWebSessionLimit
```

次のコマンドレットを使用して、ユーザの Web セッション制限を 30 に、全体のセッション制限を 255 に設定します。

```
Set-UcsWebSessionLimit -SessionsPerUser 30 -TotalSessions 255
```

## 通信サービス - Telnet

UCS telnet の設定を取得します。

```
Get-UcsTelnet
```

Telnet 接続を許可します。

```
Set-UcsTelnet -AdminState enabled -Descr "Telnet configuration for UCS"
```

## 通信サービス - CIM XML

次のコマンドレットを使用して、UCS CIM XML 設定を取得します。

```
Get-UcsCimXml
```

次のコマンドレットを使用して、CIM XML サービスを有効にします。

```
Set-UcsCimXml -AdminState enabled
```

## 通信サービス - SNMP

次のコマンドレットを使用して、UCS の SNMP 設定を取得します。

```
Get-UcsSnmp
```

次のコマンドレットを使用して、コミュニティストリングが「public」、システム担当者が「CiscoSystems」、ホストの場所が「Bangalore」の SNMP をイネーブルにします。

```
Set-UcsSnmp -Descr "SNMP config for UCS" -AdminState enabled -SysContact CiscoSystems -SysLocation Bangalore -Community public
```

次のコマンドレットを使用して、UCS SNMP ユーザを取得します。

```
Get-UcsSnmpUser
```

次のコマンドレットを使用して、UCS SNMP ユーザを追加します。

```
Add-UcsSnmpUser -Name joe -Auth md5 -Privpwd Joe@Cisco -Pwd Joe@Cisco -UseAes true
```

次のコマンドレットを使用して、UCS SNMP ユーザを設定します。

```
Get-UcsSnmpUser -Name joe | Set-UcsSnmpUser -Auth sha -UseAes false
```

次のコマンドレットを使用して、UCS SNMPユーザを削除します。

```
Get-UcsSnmpUser -Name joe | Remove-UcsSnmpUser
```

次のコマンドレットを使用して、UCS SNMP トラップを取得します。

```
Get-UcsSnmpTrap
```

次のコマンドレットを使用して、UCS SNMP トラップを追加します。

```
Add-UcsSnmpTrap -Hostname 168.65.120.32 -Community public -NotificationType traps -Port 162 -V3Privilege noauth -Version v3
```

次のコマンドレットを使用して、UCS の SNMP 設定を設定します。

```
Get-UcsSnmpTrap -Hostname 168.65.120.32 | Set-UcsSnmpTrap -Community public -NotificationType informs -Port 162 -V3Privilege auth -Version v1
```

次のコマンドレットを使用して、UCS の SNMP 設定を削除します。

```
Get-UcsSnmpTrap -Hostname 168.65.120.32 | Remove-UcsSnmpTrap
```

## 通信サービス - HTTP

次のコマンドレットを使用して、UCS http 設定を取得します。

```
Get-UcsHttp
```

次のコマンドレットを使用して、http をイネーブルにし、http から https へのリダイレクションを有効にするための UCS http コンフィギュレーションを設定します。

```
Set-UcsHttp -AdminState enabled -RedirectState enabled
```

## 通信サービス - HTTPS

UCS https コンフィギュレーションを取得します。

```
Get-UcsHttps
```

1024 ビットのキー サイズのキー リングを作成します。

```
Add-UcsKeyring -Name keyring1024 -Modulus mod1024
```

必要なサブジェクト名（マシンのホスト名）を引き渡す証明書要求を作成します。

```
Get-UcsKeyRing -Name keyring2048 | Add-UcsCertRequest -SubjName savbu-pti01
```

生成された証明書要求に対する証明書を取得し、クライアントマシンにインストールしておきます。「certmgr.msc」を実行して証明書を確認します。

トラスト ポイントを追加します。



```
Isx/2dKS4n3YX7x1hYpMubPCCL1fHIPqQwh9ddlHyKFtxqMd6/jQJyhLNOX5yz4h
HpORf14xGGWYsv1Jjqqr2jREbV3kE/uOq0NNi+2efWS0YHq9SESKqulcXgM15LyC
ZKQYolUseboYK90XgLc2yww+75UcgynLZRxgbAPstNeqPTWh12kWogrO4zkpo18Y
Vz2yB2BA6/ugCbtJuIw352HzzHU9FM4Y7R0r9k75CNjA9wScu56hX2rfIFUwnSMT
gWvg -----END CERTIFICATE----- "
```

HTTPS 用のキー リングを設定します。

```
Get-UcsHttps | Set-UcsHttps -KeyRing keyring1024 -AdminState enabled
```

これで、https を介した UCM へのアクセスでは「untrusted connection」メッセージは表示されません。

## 一般的な管理対象オブジェクトのクエリー

次のコマンドレットを使用して、特定の DN の管理対象オブジェクトを取得します。

```
Get-UcsManagedObject -Dn "sys/chassis-1"
```

次のコマンドレットを使用して、特定のクラスのすべての管理対象オブジェクトを取得します。

```
Get-UcsManagedObject -ClassId faultInst
```

次のコマンドレットを使用して、特定のクラスの管理対象オブジェクトの DN を取得します。

```
Get-UcsManagedObject -ClassId faultInst -DnList
```

次のコマンドレットを使用して、すべてのサービスプロファイルの名前を org-root から取得します。

```
Get-UcsOrg -Level root | Get-UcsManagedObject -ClassId lsServer | Select Name
```

次のコマンドレットを使用して、org-root の即時子を取得します。

```
Get-UcsOrg -Level root | Get-UcsChild
```

次のコマンドレットを使用して、管理対象オブジェクトの親を取得します。

```
Get-UcsOrg -Name Finance | Get-UcsParent
```

## 一般的な管理オブジェクトのコマンドレット

親オブジェクトを使用して VLAN を作成します。

```
$propMap = @{Name = "lan_cloud_vlan"; Id = 500} Get-UcsLanCloud | Add-UcsManagedObject
-ClassId FabricVlan -PropertyMap $propMap
```

親オブジェクトを使用して VLAN を作成し、すでに存在している場合は変更します。

```
$propMap = @{Name = "lan_cloud_vlan"; Id = 500}
Add-UcsManagedObject -ClassId FabricVlan -PropertyMap $propMap -Parent (Get-UcsLanCloud)
-ModifyPresent
```

DN を使用して VLAN を作成します。

```
$propMap = @{Dn = "fabric/lan/net-lan_cloud_vlan"; Name = "lan_cloud_vlan"; Id = 500}
Add-UcsManagedObject -ClassId FabricVlan -PropertyMap $propMap
```

管理対象オブジェクトを使用して VLAN を変更します。

```
$vlan = Get-UcsVlan -Name 'lan_cloud_vlan' $propMap = @{DefaultNet = "yes"; Id = 501;
Sharing = "primary"}
Set-UcsManagedObject -PropertyMap $propMap -ManagedObject $vlan
```

DN を使用して VLAN を変更します。

```
$propMap = @{Dn = "fabric/lan/net-lan_cloud_vlan"; DefaultNet = "yes"; Id = 501; Sharing
= "primary"}
Set-UcsManagedObject -PropertyMap $propMap -ClassId FabricVlan
```

管理対象オブジェクトを削除します。

```
Get-UcsOrg -Name Finance | Remove-UcsManagedObject
```

## 一般的なコマンドレット - XmlTag

XmlTag パラメータによって、未知の管理対象オブジェクトを使用することができます。

マルチキャスト ポリシーを作成します。

```
Add-UcsManagedObject -XmlTag fabricMulticastPolicy -PropertyMap
@{Dn="org-root/mc-policy-multicastpolicy";
Name="multicastpolicy"; PolicyOwner="local"; SnoopingState="enabled";
QuerierState="disabled";}
```

マルチキャスト ポリシーを無効にするスヌーピング状態を設定します。

```
Set-UcsManagedObject -XmlTag fabricMulticastPolicy -PropertyMap @{Dn =
"org-root/mc-policy-multicastpolicy"; SnoopingState="disabled";}
```

## アップロード ファームウェア

デフォルトの UCS システムにイメージをアップロードします。

```
Send-UcsFirmware -LiteralPath C:\work\Images\ucs-k9-bundle-b-series.2.0.1q.B.bin
```

## XML へのエクスポート

管理対象オブジェクトのコンフィギュレーションをエクスポートします。

このコマンドレットは管理対象オブジェクトと階層全体のコンフィギュレーションをエクスポートします。

```
Export-UcsXml -Dn org-root/org-Finance -Hierarchy -LiteralPath C:\cmd.xml
```

管理対象オブジェクトの xml をファイルにエクスポートします。

```
Get-UcsServiceProfile -Name sp_name | Export-UcsMoXml | Out-File c:\mo.xml
```

## XML からのインポート

次のコマンドレットを使用して、XML ファイルからコンフィギュレーションをインポートします。

```
Import-UcsXml -LiteralPath C:\cmd.xml
```

次のコマンドレットを使用して、管理対象オブジェクトの xml をインポートし、オブジェクトに変換します。

```
Import-UcsMoXml -LiteralPath c:\mo.xml
```

## KVM

サービス プロファイル、ブレード、およびラック サーバの KVM セッションを開始します。

### シンタックス

```
Start-UcsKvmSession -ServiceProfile <LsServer> [-FrameTitle <string>] [-IPv4Addresses] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Start-UcsKvmSession -Blade <ComputeBlade> [-FrameTitle <string>] [-IPv4Addresses] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Start-UcsKvmSession -RackUnit <ComputeRackUnit> [-FrameTitle <string>] [-IPv4Addresses] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

ブレード、サービスプロファイルまたはラックユニットの KVM セッションを起動するには、次を入力します

```
Get-UcsBlade -SlotId 4 | Start-UcsKvmSession
Get-UcsServiceProfile -Name testSP | Start-UcsKvmSession
Get-UcsRackUnit | Start-UcsKvmSession
```

管理インターフェイスで設定されている複数のインバンドおよびアウトオブバンド IP アドレスがある場合、KVM を起動する IP アドレスを選択するように求められます。

## 例

```
Get-UcsBlade -SlotId 4 | Start-UcsKvmSession

KVM Launch
Choose an IP address:

[1] 1.10.65.183.39 [2] 2.10.65.183.43 [3] 3.2001::105 [4] 4.10.65.183.14 [?] Help
(default is "1"): ?
1 - Ucs:10.65.183.8, IpAddress:10.65.183.39, IpSource:out-band, AccessType:Equipment,
EqDn:sys/chassis-1/blade-4, SpDn:org-root/ls-ssp99
2 - Ucs:10.65.183.8, IpAddress:10.65.183.43, IpSource:in-band, AccessType:Equipment,
EqDn:sys/chassis-1/blade-4, SpDn:org-root/ls-ssp99
3 - Ucs:10.65.183.8, IpAddress:2001::105, IpSource:in-band, AccessType:Equipment,
EqDn:sys/chassis-1/blade-4, SpDn:org-root/ls-ssp99
4 - Ucs:10.65.183.8, IpAddress:10.65.183.14, IpSource:out-band, AccessType:ServiceProfile,
EqDn:sys/chassis-1/blade-4, SpDn:org-root/ls-ssp99
[1] 1.10.65.183.39 [2] 2.10.65.183.43 [3] 3.2001::105 [4] 4.10.65.183.14 [?] Help
(default is "1"):2
```

# UCS Manager の Java web UI の起動

次のコマンドレットを使用して、UCS Manager に接続し、UCS Manager web UI を起動します。

```
Start-UcsGuiSession
```

セキュア ロギングをイネーブルにします。

一部の XML トランザクションは安全なものとして処理されるため、UCS Manager web UI によってログに記録されません。LogAllXml フラグによってセキュア ロギングを可能にします。

```
Start-UcsGuiSession -LogAllXml
```

Get-UcsStatus および Start-UcsGuiSession コマンドレットを使用して UCS Manager web UI を起動します。

```
Get-UcsStatus | Start-UcsGuiSession
```

次のコマンドレットを使用して、UCS Manager に接続せずに、UCS Manager web UI を起動します。

```
Start-UcsGuiSession -Name <IP Address or Hostname of UCSM>
```

資格情報を変数に格納し、コマンドレットに渡します。

```
$password = "<Password>" | ConvertTo-SecureString -AsPlainText -Force
$cred = New-Object System.Management.Automation.PSCredential("UserName", $password)
Start-UcsGuiSession -Name <IP Address or Hostname of UCSM> -Credential $cred
```

## Cisco UCS Manager HTML GUI の起動

新しいスイッチパラメータが、UCS Manager HTML GUI を起動する `Start-UcsGuiSession` に追加します。デフォルトでは、`Start-UcsGuiSession` コマンドレットは、Java GUI を起動します。

Cisco UCS Manager 3.1 (2) のリリースでは、HTML GUI への自動ログインが許可されます。`Start-UcsGuiSession` コマンドレットは自動ログインのために強化されます。また、コンテキストベースのログインをサポートします。たとえば、サービスプロファイル、ポリシー、またはプールなど、特定のエンティティの HTML GUI を起動する場合、必要な MO を `Start-UcsGuiSession` コマンドレットに渡すことができます。これは渡された管理対象オブジェクトのページを自動的に起動します。



- (注) Cisco UCS Manager 3.1 (2) のリリース以降、HTML GUI のトークンを使用して自動署名が有効になります。

### Syntax

次の構文は HTML GUI に自動ログインのためのものです。

```
Start-UcsGuiSession -HTML [-ManagedObject <ManagedObject>] [-Ucs <UcsHandle[]>]
[<CommonParameters>]
```

### 例

```
Get-UcsBlade -SlotId 1 | Start-UcsGuiSession -HTML
```

## UCS 統計情報

次のコマンドレットを使用して、シャーシ ID 1 およびスロット ID 1 の現在の UCS 統計情報を取得します。

```
Get-UcsBlade -ChassisId 1 -SlotId 1 | Get-UcsStatistics -Current
```

次のコマンドレットを使用して、シャーシ ID 1 およびスロット ID 1 の UCS 統計履歴を取得します。

```
Get-UcsBlade -ChassisId 1 -SlotId 1 | Get-UcsStatistics -History
```

次のコマンドレットを使用して、Ucs の統計情報をクリアします。

```
Get-UcsManagedObject -Dn sys/chassis-1/blade-1/board/temp-stats | Clear-UcsStatistics
```

## UCS 6324 ファブリック インターコネクットのスケラビリティ ポートの設定

UCS 6324 ファブリック インターコネクット B のブレイクアウト ポート 1/5/1 をサーバポートとして設定します。

```
$mo = Add-UcsManagedObject -XmlTag fabricSubGroup -PropertyMap
@{dn="fabric/server/sw-B/slot-1-aggr-port-5";aggrPortId="5";slotId="1"}
$mo | Add-UcsManagedObject -XmlTag fabricDceSwSrvEp -PropertyMap
@{rn="slot-1-port-1";portId="1";slotId="1"}
```

UCS 6324 ファブリック インターコネクット B のブレイクアウト ポート 1/5/1 を FCoE ストレージポートとして設定します。

```
$mo = Add-UcsManagedObject -modifyPresent -XmlTag fabricSubGroup -PropertyMap
@{dn="fabric/fc-estc/B/slot-1-aggr-port-5";aggrPortId="5";slotId="1"}
$mo | Add-UcsManagedObject -XmlTag fabricFcoeEstcEp -PropertyMap
@{rn="phys-fcoe-slot-1-port-1";portId="1";slotId="1"}
```

UCS 6324 ファブリック インターコネクット B のブレイクアウト ポート 1/5/1 をアプライアンスポートとして設定します。

```
$mo = Add-UcsManagedObject -modifyPresent -XmlTag fabricSubGroup -PropertyMap
@{dn="fabric/eth-estc/B/slot-1-aggr-port-5";aggrPortId="5";slotId="1"}
$mo | Add-UcsManagedObject -XmlTag fabricEthEstcEp -PropertyMap
@{rn="phys-eth-slot-1-port-1";portId="1";slotId="1"}
```

## トランザクションの影響

Get-UcsTransactionImpact コマンドレットによって、保留状態のトランザクションの影響を推定します。このコマンドレットでは、ConfigEstimateImpact メソッドを使用し、UcsImpact オブジェクトを返します。UCS Manager web UI iによって配信されたメッセージのようなメッセージが UcsImpact オブジェクトの一部として提供されます。

トランザクションを開始します。

```
Start-UcsTransaction
```

サービス プロファイルを作成します。

```
$sp = Add-UcsServiceProfile -Name sp_name
```

vNIC を作成します。

```
$eth0 = $sp | Add-UcsVnic -Name eth0 -IdentPoolName empty_pool
```

vNIC に VLAN を追加し、これをネイティブ VLAN にします。

```
$eth0 | Add-UcsVnicInterface -Name primary -DefaultNet true
```

トランザクションの影響を取得します。

```
Get-UcsTransactionImpact
```

作成されたサービスプロファイルなどのコンフィギュレーションの失敗を示す **UcsImpact** オブジェクトが返されます。

## コマンドレットのメタ情報

コマンドレットにマッピングされているすべての管理オブジェクトに関するメタ情報を取得します。

```
Get-UcsCmdletMeta
```

コマンドレットにマッピングされている **LsServer** に関するメタ情報を取得します。

```
Get-UcsCmdletMeta -ClassId LsServer
```

**ServiceProfile (LsServer)** クラスの階層を表示します。

```
Get-UcsCmdletMeta -Noun UcsServiceProfile -Tree
```

**UcsServiceProfile** 名詞のメタ情報を取得します。

```
Get-UcsCmdletMeta -Noun UcsServiceProfile
```

**LsServer** の管理対象オブジェクト情報を参照します。

```
Get-UcsCmdletMeta -ClassId LsServer | Select -ExpandProperty MoMeta
```

**LsServer** の管理対象オブジェクトのプロパティ情報を参照します。

```
Get-UcsCmdletMeta -ClassId LsServer | Select -ExpandProperty MoMeta | Select -ExpandProperty PropertyMeta
```

## Compare-UcsManagedObject - Dn 変換

Org A の下にサービスプロファイルを作成します。Org A と B はすでに設定されているものとします。

```
$org = Get-UcsOrg -Name A -LimitScope
$destOrg = Get-UcsOrg -Name B -LimitScope
$sp = Add-UcsServiceProfile -Org $org -Name abc
```

変換する必要があるエンティティの DN で変換マップを作成します。

```
$xlateDn = @{ }
$xlateDn['org-root/org-A/ls-abc'] = 'org-root/org-B/ls-xyz'
```

必要な変更を参照する **Compare-UcsMo** で変換マップを結合します。

```
Compare-UcsManagedObject (Get-UcsServiceProfile -Org $destOrg -Name xyz -LimitScope)
(Get-UcsServiceProfile -Org $org -Name abc -LimitScope) -XlateMap $xlateDn
```

必要な変更を参照する **Compare** で変換 **Org** を結合します。

```
Compare-UcsManagedObject (Get-UcsServiceProfile -Org $destOrg -Name xyz -LimitScope)
(Get-UcsServiceProfile -Org $org -Name abc -LimitScope) -XlateOrg org-root/org-B
```

ファイルの名前を変更する際に、**Org A** から **B** へサービス プロファイルを同期します。

```
Sync-UcsManagedObject (Compare-UcsManagedObject (Get-UcsServiceProfile -Org $destOrg
-Name xyz -LimitScope)
(Get-UcsServiceProfile -Org $org -Name abc -LimitScope) -XlateMap $xlateDn) -Force |
select Dn
```

## Compare-UcsManagedObject - GetPropertyDiff()

プロパティ間の相違を確認するために、**Compare-UcsManagedObject** の出力で **GetPropertyDiff()** 関数を使用します。

```
$sp1 = Get-UcsServiceProfile -Dn org-root/ls-abc
$sp2 = $sp1 | Set-UcsServiceProfile -Descr 'GetPropertyDiff Example' -Force
$diff = Compare-UcsManagedObject $sp1 $sp2
```

違いを持つすべてのプロパティを表示します。**\$diff** がオブジェクトの配列の場合、**GetPropertyDiff** は **\$diff [<index>]** で動作します。

```
$diff.GetPropertyDiff()
```

特定のプロパティ **\$diff** 用。

```
GetPropertyDiff('descr')
```

比較に **MO** のすべてのオペレーション プロパティを含みます。

```
Compare-UcsManagedObject $sp1 $sp2 -IncludeOperational
```

## コマンドレットの追加 **-ModifyPresent** フラグ

**ModifyPresent** オプションによって、**MO** がすでにある場合に **Add** コマンドレットがエラーを返す代わりに **MO** を変更できるようにできます。

名前と ID のペアを含む CSV ファイルを作成します。

```
$( "Name,Id"; foreach ($vlan in 501..510) { "vlan${vlan}, ${vlan}" } ) | Out-File
c:\vlans.csv
```

ファイルから名前と VLAN ペアをインポートし、それらの VLAN を作成します。

```
$lc = Get-UcsLanCloud
```

```
Start-UcsTransaction
Import-Csv C:\vlans.csv | % { $lc | Add-UcsVlan -ModifyPresent -Name $_.Name -Id $_.Id
}
Complete-UcsTransaction
```

ID を編集したり、新しい VLAN を追加するには CSV ファイルを編集します。既存の VLAN を再び作成した場合は、上記の同じ `Add-UcsVlan` スニペットを再実行してもエラーになります（変更の有無にかかわらず）。すでに VLAN が存在する場合は、代わりに、`ModifyPresent` オプションを使って `Add-UcsVlan` を起動し、この VLAN を編集することでこのエラーを解決します。

```
$lc = Get-UcsLanCloud
Start-UcsTransaction
Import-Csv C:\vlans.csv | % { $lc | Add-UcsVlan -ModifyPresent -Name $_.Name -Id $_.Id
}
Complete-UcsTransaction
```

## 機能カタログの更新

機能カタログは調整可能なパラメータ、文字列、およびルールのセットです。Cisco UCS では、カタログを使用して、サーバの新しく承認された DIMM やディスク ドライブなどのコンポーネントの表示と設定可能性を更新します。

ローカルのファイルソースから機能カタログを更新する場合は、次のコマンドレットを使用できます。

```
Update-UcsCatalogue -LiteralPath C:\Work\ucs-catalog.2.2.3a.T.bin
```

## サーバの操作

サーバを操作するために追加された新しい簡易コマンドレットを次に示します。

操作の説明	PowerTool リリース 1.4.1 以前の コマンドレット	PowerTool リリース 1.5.1 以降 のコマンドレット
UCS サーバの確認応答	Get-UcsChassis -Id 1   Get-UcsBlade -SlotId 1   Set-UcsBlade -AdminPower "policy" -Lc "rediscover" -PolicyOwner "local"	Get-UcsServer   where { \$_.Dn -eq "sys/chassis-1/blade-1"}   Confirm-UcsServer
UCS サーバの稼働中止	Get-UcsChassis -Id 1   Get-UcsBlade -SlotId 1   Set-UcsBlade -AdminPower "policy" -Lc "decommission" -PolicyOwner "local"	Get-UcsServer   where { \$_.Dn -eq "sys/chassis-1/blade-1"}   Disable-UcsServer

操作の説明	PowerTool リリース 1.4.1 以前の コマンドレット	PowerTool リリース 1.5.1 以降 のコマンドレット
UCS サーバのハードリセット	Get-UcsOrg -Level root   Get-UcsServiceProfile -Name "testSP" -LimitScope   Get-UcsServerPower   Set-UcsServerPower -State "hard-reset-immediate"	Get-UcsServiceProfile -name testSP   Reset-UcsServer
UCS サーバのブート	Get-UcsOrg -Level root   Get-UcsServiceProfile -Name "testSP" -LimitScope   Get-UcsServerPower   Set-UcsServerPower -State "admin-up"	Get-UcsServiceProfile -name testSP   Start-UcsServer
UCS サーバのシャットダウン	Get-UcsOrg -Level root   Get-UcsServiceProfile -Name "testSP" -LimitScope   Get-UcsServerPower   Set-UcsServerPower -State "soft-shut-down"	Get-UcsServiceProfile -name testSP   Stop-UcsServer
UCS サーバの電源再投入	Get-UcsOrg -Level root   Get-UcsServiceProfile -Name "testSP" -LimitScope   Get-UcsServerPower   Set-UcsServerPower -State "cycle-immediate"	Get-UcsServiceProfile -name testSP   Restart-UcsServer
UCS サーバの CMOS のリセッ ト	Get-UcsOrg -Level root   Get-UcsServiceProfile -Name "testSP" -LimitScope   Get-UcsServerPower   Set-UcsServerPower -State "cmos-reset-immediate"	Get-UcsServiceProfile -name testSP   Reset-UcsServerCmos
UCS サーバの BMC のリセッ ト	Get-UcsOrg -Level root   Get-UcsServiceProfile -Name "testSP" -LimitScope   Get-UcsServerPower   Set-UcsServerPower -State "bmc-reset-immediate"	Get-UcsServiceProfile -name testSP   Reset-UcsServerBmc
UCS サーバのロケータ LED の 点灯	Get-UcsChassis -Id 1   Get-UcsBlade -SlotId 1   Get-UcsLocatorLed   Set-UcsLocatorLed -AdminState "on" -BoardType "single" -Id 1	Get-UcsServer   where { \$_.Dn -eq "sys/chassis-1/blade-1" }   Enable-UcsLocatorLed

操作の説明	PowerTool リリース 1.4.1 以前の コマンドレット	PowerTool リリース 1.5.1 以降 のコマンドレット
UCS サーバのロケータ LED の 消灯	Get-UcsChassis -Id 1   Get-UcsBlade -SlotId 1   Get-UcsLocatorLed   Set-UcsLocatorLed -AdminState "off" -BoardType "single" -Id 1	Get-UcsServer   where { \$_.Dn -eq "sys/chassis-1/blade-1" }   Disable-UcsLocatorLed

## 32 パラメータ セットの制限

Microsoft PowerShell のフレームワークに従って、コマンドレットは 32 を超えるパラメータセットを持つことはできません。コマンドレットのパラメータセットの数が複数の 32 を超える場合は、コマンドレットは、パラメータまたはパラメータセットを正しく識別できない場合があります、異常動作の可能性があります。

次の PowerTool コマンドレットは、この制限の影響を受けます。

- Get-UcsEquipmentFruVariant
- Get-UcsEquipmentManufacturingDef
- Get-UcsEquipmentPhysicalDef
- Get-UcsEquipmentPicture
- Get-UcsEquipmentServiceDef
- Get-UcsEquipmentSlotArrayRef
- Get-UcsFirmwareUpgradeConstraint

パラメータセットの論理グループが作成された後に、各コマンドレットは2つのコマンドレットに分割されます。各既存のコマンドレットに対して、ストレージに関連するパラメータセットは元のコマンドレットから削除し、新しいコマンドレットに作成されました。これは、次の追加ストレージ関連 PowerTool コマンドレットを提供します。

- Get-UcsEquipmentFruVariantStorage
- Get-UcsEquipmentManufacturingDefStorage
- Get-UcsEquipmentPhysicalDefStorage
- Get-UcsEquipmentPictureStorage
- Get-UcsEquipmentServiceDefStorage
- Get-UcsEquipmentSlotArrayRefStorage
- Get-UcsFirmwareUpgradeConstraintStorage

ストレージ固有のパラメータセットが、固有のコマンドレットをもつため、後方互換性は、ストレージ固有のパラメータセットの元のコマンドレットの使用中に中断される場合があります。

Cisco は指定されたストレージ関連パラメータセットの新しいストレージコマンドレットを使用することを推奨します。

次のセクションでは、これらのコマンドレットの詳細なシンタックスを示します。

## Get UcsEquipmentFruVariant

このコマンドレットは管理対象オブジェクトの「EquipmentFruVariant」タイプに関する情報を取得するために使用します。この管理対象オブジェクトは FRU バリエントと PID の間のマッピングを確立するために使用します。

```
Get-UcsEquipmentFruVariant [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -AdaptorFruCapProvider <AdaptorFruCapProvider> [-Type
<string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -DiagSrvCapProvider <DiagSrvCapProvider> [-Type <string>]
[-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild | cascade | del
| mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -EquipmentBaseBoardCapProvider
<EquipmentBaseBoardCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -EquipmentBladeBiosCapProvider
<EquipmentBladeBiosCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -EquipmentBladeCapProvider <EquipmentBladeCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -EquipmentCatalogCapProvider <EquipmentCatalogCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -ChassisCapProvider <EquipmentChassisCapProvider> [-Type
<string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -EquipmentDbgPluginCapProvider
<EquipmentDbgPluginCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentFruVariant -FanModuleCapProvider <EquipmentFanModuleCapProvider>
```

```

[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -FexCapProvider <EquipmentFexCapProvider> [-Type <string>]
[-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild | cascade | del
| mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentGemCapProvider <EquipmentGemCapProvider> [-Type
<string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentGraphicsCardCapProvider
<EquipmentGraphicsCardCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentHostIfCapProvider <EquipmentHostIfCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentIOCardCapProvider <EquipmentIOCardCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentIOExpanderCapProvider
<EquipmentIOExpanderCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentMgmtCapProvider <EquipmentMgmtCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentMgmtExtCapProvider <EquipmentMgmtExtCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentProcessorUnitCapProvider
<EquipmentProcessorUnitCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -PsuCapProvider <EquipmentPsuCapProvider> [-Type <string>]
[-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild | cascade | del
| mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -RackUnitCapProvider <EquipmentRackUnitCapProvider> [-Type
<string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentFruVariant -EquipmentSecurityUnitCapProvider
<EquipmentSecurityUnitCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```

```

Get-UcsEquipmentFruVariant -EquipmentServerUnitCapProvider
<EquipmentServerUnitCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariant -EquipmentSwitchCapProvider <EquipmentSwitchCapProvider>
[-Type <string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariant -EquipmentSwitchIOCardCapProvider
<EquipmentSwitchIOCardCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariant -EquipmentTpmCapProvider <EquipmentTpmCapProvider> [-Type
<string>] [-Description <string>] [-Dn <string>] [-Pid <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```

## Get-UcsEquipmentFruVariantStorage

```

Get-UcsEquipmentFruVariantStorage [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-Type <string>] [-Description <string>] [-Dn
<string>] [-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-Type <string>] [-Description <string>] [-Dn <string>]
[-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-Type <string>] [-Description <string>] [-Dn
<string>] [-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentFruVariantStorage -EquipmentStorageSasExpanderCapProvider
<EquipmentStorageSasExpanderCapProvider> [-Type <string>] [-Description <string>] [-Dn
<string>] [-Pid <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

```

## Get-UcsEquipmentManufacturingDef

このコマンドレットは管理対象オブジェクトの「EquipmentManufacturingDef」タイプに関する情報を取得するために使用します。この管理対象オブジェクトはPIDとSKUなどのプロパティの製造関連の保存に使用されます。

```
Get-UcsEquipmentManufacturingDef [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDef -AdaptorFruCapProvider <AdaptorFruCapProvider>
[-Caption <string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn
<string>] [-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no
| true | yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber
<string>] [-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku
<string>] [-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDef -DiagSrvCapProvider <DiagSrvCapProvider> [-Caption
<string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn <string>]
[-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no | true |
yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber <string>]
[-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku <string>]
[-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDef -EquipmentBaseBoardCapProvider
<EquipmentBaseBoardCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDef -EquipmentBladeBiosCapProvider
<EquipmentBladeBiosCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDef -EquipmentBladeCapProvider <EquipmentBladeCapProvider>
[-Caption <string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn
<string>] [-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no
| true | yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber
<string>] [-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku
<string>] [-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDef -EquipmentCatalogCapProvider
```



```

Get-UcsEquipmentManufacturingDef -EquipmentHostIfCapProvider
<EquipmentHostIfCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentIOCardCapProvider
<EquipmentIOCardCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentIOExpanderCapProvider
<EquipmentIOExpanderCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentMgmtCapProvider <EquipmentMgmtCapProvider>
[-Caption <string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn
<string>] [-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no
| true | yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber
<string>] [-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku
<string>] [-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentMgmtExtCapProvider
<EquipmentMgmtExtCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentProcessorUnitCapProvider
<EquipmentProcessorUnitCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -PsuCapProvider <EquipmentPsuCapProvider> [-Caption
<string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn <string>]
[-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no | true |
yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber <string>]

```

```

[-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku <string>]
[-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -RackUnitCapProvider <EquipmentRackUnitCapProvider>
[-Caption <string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn
<string>] [-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no
| true | yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber
<string>] [-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku
<string>] [-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentSecurityUnitCapProvider
<EquipmentSecurityUnitCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentServerUnitCapProvider
<EquipmentServerUnitCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentSwitchCapProvider
<EquipmentSwitchCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentSwitchIOCardCapProvider
<EquipmentSwitchIOCardCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]

Get-UcsEquipmentManufacturingDef -EquipmentTpmCapProvider <EquipmentTpmCapProvider>
[-Caption <string>] [-Clei <string>] [-Descr <string>] [-Description <string>] [-Dn
<string>] [-FruMajorType <string>] [-FruMinorType <string>] [-IsSecFwUpdate {false | no
| true | yes}] [-Name <string>] [-OemName <string>] [-OemPartNumber <string>] [-PartNumber
<string>] [-Pid <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Series <string>] [-Sku
<string>] [-VendorEquipmentType <string>] [-Vid <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```

## Get-UcsEquipmentManufacturingDefStorage

```
Get-UcsEquipmentManufacturingDefStorage [-Caption <string>] [-Clei <string>] [-Descr
<string>] [-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType
<string>] [-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-Caption <string>] [-Clei <string>] [-Descr
<string>] [-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType
<string>] [-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-Caption <string>] [-Clei <string>] [-Descr <string>]
[-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType <string>]
[-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-Caption <string>] [-Clei <string>] [-Descr
<string>] [-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType
```

```
<string> [-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentManufacturingDefStorage -EquipmentStorageSasExpanderCapProvider
<EquipmentStorageSasExpanderCapProvider> [-Caption <string>] [-Clei <string>] [-Descr
<string>] [-Description <string>] [-Dn <string>] [-FruMajorType <string>] [-FruMinorType
<string>] [-IsSecFwUpdate {false | no | true | yes}] [-Name <string>] [-OemName <string>]
[-OemPartNumber <string>] [-PartNumber <string>] [-Pid <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Series <string>] [-Sku <string>] [-VendorEquipmentType <string>] [-Vid
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

## Get-UcsEquipmentPhysicalDef

このコマンドレットは管理対象オブジェクトの「EquipmentPhysicalDef」タイプに関する情報を取得するために使用します。この管理対象オブジェクトは最大温度および削除条件などの物理特性を保存するために使用します。

```
Get-UcsEquipmentPhysicalDef [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDef -AdaptorFruCapProvider <AdaptorFruCapProvider> [-Depth
<string>] [-Descr <string>] [-Dn <string>] [-Height <string>] [-MaximumPower <string>]
[-MaximumTemperature <string>] [-MinimumPower <string>] [-MinimumTemperature <string>]
[-Name <string>] [-NominalPower <string>] [-NominalTemperature <string>]
[-OperatingVoltages <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Weight <string>] [-Width
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDef -DiagSrvCapProvider <DiagSrvCapProvider> [-Depth <string>]
[-Descr <string>] [-Dn <string>] [-Height <string>] [-MaximumPower <string>]
[-MaximumTemperature <string>] [-MinimumPower <string>] [-MinimumTemperature <string>]
[-Name <string>] [-NominalPower <string>] [-NominalTemperature <string>]
[-OperatingVoltages <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy
| policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Weight <string>] [-Width
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDef -EquipmentBaseBoardCapProvider
<EquipmentBaseBoardCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDef -EquipmentBladeBiosCapProvider
<EquipmentBladeBiosCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
```







```
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

## Get-UcsEquipmentPhysicalDefStorage

```
Get-UcsEquipmentPhysicalDefStorage [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-Depth <string>] [-Descr <string>] [-Dn
<string>] [-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>]
[-MinimumPower <string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower
<string>] [-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel
<uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade |
del | mod | none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>]
[-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower
<string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>]
[-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>]
```

```
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPhysicalDefStorage -EquipmentStorageSasExpanderCapProvider <EquipmentStorageSasExpanderCapProvider> [-Depth <string>] [-Descr <string>] [-Dn <string>] [-Height <string>] [-MaximumPower <string>] [-MaximumTemperature <string>] [-MinimumPower <string>] [-MinimumTemperature <string>] [-Name <string>] [-NominalPower <string>] [-NominalTemperature <string>] [-OperatingVoltages <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}] [-Weight <string>] [-Width <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

## Get-UcsEquipmentPicture

このコマンドレットは管理対象オブジェクトの「EquipmentPicture」タイプに関する情報を取得するために使用します。この管理対象オブジェクトを使用して、Cisco UCS Manager でイメージファイルの詳細、タイプ、レイアウトを保存します。

```
Get-UcsEquipmentPicture [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -AdaptorFruCapProvider <AdaptorFruCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -DiagSrvCapProvider <DiagSrvCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentBaseBoardCapProvider <EquipmentBaseBoardCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentBladeBiosCapProvider <EquipmentBladeBiosCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentBladeCapProvider <EquipmentBladeCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentCatalogCapProvider <EquipmentCatalogCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -ChassisCapProvider <EquipmentChassisCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top |
```

```
top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentDbgPluginCapProvider <EquipmentDbgPluginCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -FanModuleCapProvider <EquipmentFanModuleCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -FexCapProvider <EquipmentFexCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentGemCapProvider <EquipmentGemCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentGraphicsCardCapProvider <EquipmentGraphicsCardCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentHostIfCapProvider <EquipmentHostIfCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentIOCardCapProvider <EquipmentIOCardCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentIOExpanderCapProvider <EquipmentIOExpanderCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentMgmtCapProvider <EquipmentMgmtCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentMgmtExtCapProvider <EquipmentMgmtExtCapProvider> [-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentProcessorUnitCapProvider
<EquipmentProcessorUnitCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -PsuCapProvider <EquipmentPsuCapProvider> [-Type {back |
bottom | front | front-bottom-scaled | front-top-scaled | left | right | top | top-scaled
| unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod
| none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -RackUnitCapProvider <EquipmentRackUnitCapProvider> [-Type
{back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top |
top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade |
del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentSecurityUnitCapProvider
<EquipmentSecurityUnitCapProvider> [-Type {back | bottom | front | front-bottom-scaled
| front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentServerUnitCapProvider
<EquipmentServerUnitCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentSwitchCapProvider <EquipmentSwitchCapProvider>
[-Type {back | bottom | front | front-bottom-scaled | front-top-scaled | left | right |
top | top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentSwitchIOCardCapProvider
<EquipmentSwitchIOCardCapProvider> [-Type {back | bottom | front | front-bottom-scaled
| front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPicture -EquipmentTpmCapProvider <EquipmentTpmCapProvider> [-Type
{back | bottom | front | front-bottom-scaled | front-top-scaled | left | right | top |
top-scaled | unknown}] [-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade |
del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

## Get-UcsEquipmentPictureStorage

```
Get-UcsEquipmentPictureStorage [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPictureStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}] [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>]
[-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentPictureStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-Type {back | bottom | front |
```

```

front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}}
[-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}}
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
  [<CommonParameters>]

  Get-UcsEquipmentPictureStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>]
  [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

  Get-UcsEquipmentPictureStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>]
  [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

  Get-UcsEquipmentPictureStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>]
  [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

  Get-UcsEquipmentPictureStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-Type {back | bottom | front | front-bottom-scaled |
front-top-scaled | left | right | top | top-scaled | unknown}} [-Dn <string>] [-FileName
<string>] [-Sacl {addchild | cascade | del | mod | none}} [-Hierarchy] [-Filter <string>]
  [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

  Get-UcsEquipmentPictureStorage -EquipmentStorageSasExpanderCapProvider
<EquipmentStorageSasExpanderCapProvider> [-Type {back | bottom | front |
front-bottom-scaled | front-top-scaled | left | right | top | top-scaled | unknown}}
[-Dn <string>] [-FileName <string>] [-Sacl {addchild | cascade | del | mod | none}}
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
  [<CommonParameters>]

```

## Get-UcsEquipmentServiceDef

このコマンドレットは管理対象オブジェクトの「EquipmentServiceDef」タイプに関する情報を取得するために使用します。この管理対象オブジェクトは削除条件とスロットアレイ記述子名前などのサービスのプロパティを保存するために使用します。

```

Get-UcsEquipmentServiceDef [-CanBeFRUed {false | no | true | yes}} [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}} [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}} [-Sacl {addchild | cascade | del | mod | none}}
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
  [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

  Get-UcsEquipmentServiceDef -AdaptorFruCapProvider <AdaptorFruCapProvider> [-CanBeFRUed
{false | no | true | yes}} [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel
<uint32>] [-PolicyOwner {local | pending-policy | policy}} [-RemovalConditions {Not
Applicable | Removable when off | Removable when on or off | Unknown}} [-Sacl {addchild
| cascade | del | mod | none}} [-ServicePhilosophy <string>] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

  Get-UcsEquipmentServiceDef -DiagSrvCapProvider <DiagSrvCapProvider> [-CanBeFRUed
{false | no | true | yes}} [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel
<uint32>] [-PolicyOwner {local | pending-policy | policy}} [-RemovalConditions {Not
Applicable | Removable when off | Removable when on or off | Unknown}} [-Sacl {addchild
| cascade | del | mod | none}} [-ServicePhilosophy <string>] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]  [<CommonParameters>]

```

```

Get-UcsEquipmentServiceDef -EquipmentBaseBoardCapProvider
<EquipmentBaseBoardCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDef -EquipmentBladeBiosCapProvider
<EquipmentBladeBiosCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDef -EquipmentBladeCapProvider <EquipmentBladeCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentServiceDef -EquipmentCatalogCapProvider <EquipmentCatalogCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentServiceDef -ChassisCapProvider <EquipmentChassisCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentServiceDef -EquipmentDbgPluginCapProvider
<EquipmentDbgPluginCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDef -FanModuleCapProvider <EquipmentFanModuleCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentServiceDef -FexCapProvider <EquipmentFexCapProvider> [-CanBeFRUed
{false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel
<uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not
Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild
| cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter
<string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```

```

    Get-UcsEquipmentServiceDef -EquipmentGemCapProvider <EquipmentGemCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentGraphicsCardCapProvider
<EquipmentGraphicsCardCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr
<string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local
| pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentHostIfCapProvider <EquipmentHostIfCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentIOCardCapProvider <EquipmentIOCardCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentIOExpanderCapProvider
<EquipmentIOExpanderCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentMgmtCapProvider <EquipmentMgmtCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentMgmtExtCapProvider <EquipmentMgmtExtCapProvider>
[-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}]
[-RemovalConditions {Not Applicable | Removable when off | Removable when on or off |
Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentServiceDef -EquipmentProcessorUnitCapProvider
<EquipmentProcessorUnitCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr
<string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local
| pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```

```
Get-UcsEquipmentServiceDef -PscapProvider <EquipmentPscapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentServiceDef -RackUnitCapProvider <EquipmentRackUnitCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentServiceDef -EquipmentSecurityUnitCapProvider <EquipmentSecurityUnitCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentServiceDef -EquipmentServerUnitCapProvider <EquipmentServerUnitCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentServiceDef -EquipmentSwitchCapProvider <EquipmentSwitchCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentServiceDef -EquipmentSwitchIOCardCapProvider <EquipmentSwitchIOCardCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentServiceDef -EquipmentTpmCapProvider <EquipmentTpmCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

## Get-UcsEquipmentServiceDefStorage

```
Get-UcsEquipmentServiceDefStorage [-CanBeFRUed {false | no | true | yes}] [-Descr <string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off | Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}] [-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```

Get-UcsEquipmentServiceDefStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDefStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr
<string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local
| pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDefStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDefStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDefStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr <string>]
[-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDefStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr
<string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local
| pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentServiceDefStorage -EquipmentStorageSasExpanderCapProvider
<EquipmentStorageSasExpanderCapProvider> [-CanBeFRUed {false | no | true | yes}] [-Descr
<string>] [-Dn <string>] [-Name <string>] [-PolicyLevel <uint32>] [-PolicyOwner {local
| pending-policy | policy}] [-RemovalConditions {Not Applicable | Removable when off |
Removable when on or off | Unknown}] [-Sacl {addchild | cascade | del | mod | none}]
[-ServicePhilosophy <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```

## Get-UcsEquipmentSlotArrayRef

コマンドレットは管理対象オブジェクトの「EquipmentSlotArrayRef」タイプに関する情報を取得するために使用します。この管理対象オブジェクトには、FRU からスロット アレイへの参照が含まれています。FRU のスロットのスペンも定義しています。

```
Get-UcsEquipmentSlotArrayRef [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -AdaptorFruCapProvider <AdaptorFruCapProvider> [-Name
<string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>] [-PolicyLevel
<uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade
| del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -DiagSrvCapProvider <DiagSrvCapProvider> [-Name <string>]
[-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -EquipmentBaseBoardCapProvider
<EquipmentBaseBoardCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -EquipmentBladeBiosCapProvider
<EquipmentBladeBiosCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -EquipmentBladeCapProvider <EquipmentBladeCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -EquipmentCatalogCapProvider <EquipmentCatalogCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRef -ChassisCapProvider <EquipmentChassisCapProvider> [-Name
<string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>] [-PolicyLevel
<uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade
| del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn
<string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>]
[-Xml] [<CommonParameters>]
```

```

Get-UcsEquipmentSlotArrayRef -EquipmentDbgPluginCapProvider
<EquipmentDbgPluginCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -FanModuleCapProvider <EquipmentFanModuleCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -FexCapProvider <EquipmentFexCapProvider> [-Name <string>]
[-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod |
none}] [-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -EquipmentGemCapProvider <EquipmentGemCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -EquipmentGraphicsCardCapProvider
<EquipmentGraphicsCardCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -EquipmentHostIfCapProvider <EquipmentHostIfCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -EquipmentIOCardCapProvider <EquipmentIOCardCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -EquipmentIOExpanderCapProvider
<EquipmentIOExpanderCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsEquipmentSlotArrayRef -EquipmentMgmtCapProvider <EquipmentMgmtCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]

```

```

[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentMgmtExtCapProvider <EquipmentMgmtExtCapProvider>
    [-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentProcessorUnitCapProvider
<EquipmentProcessorUnitCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -PsuCapProvider <EquipmentPsuCapProvider> [-Name <string>]
[-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>]
[-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild | cascade | del | mod
| none}] [-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -RackUnitCapProvider <EquipmentRackUnitCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentSecurityUnitCapProvider
<EquipmentSecurityUnitCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentServerUnitCapProvider
<EquipmentServerUnitCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentSwitchCapProvider <EquipmentSwitchCapProvider>
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentSwitchIOCardCapProvider
<EquipmentSwitchIOCardCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

    Get-UcsEquipmentSlotArrayRef -EquipmentTpmCapProvider <EquipmentTpmCapProvider>

```

```
[-Name <string>] [-Descr <string>] [-Dn <string>] [-NumberOfSlotsSpanned <uint16>]
[-PolicyLevel <uint32>] [-PolicyOwner {local | pending-policy | policy}] [-Sacl {addchild
| cascade | del | mod | none}] [-SlotSpanOrientation {inline | transverse | unknown}]
[-TargetDn <string>] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

## Get-UcsEquipmentSlotArrayRefStorage

```
Get-UcsEquipmentSlotArrayRefStorage [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsEquipmentSlotArrayRefStorage -EquipmentStorageSasExpanderCapProvider
<EquipmentStorageSasExpanderCapProvider> [-Name <string>] [-Descr <string>] [-Dn <string>]
[-NumberOfSlotsSpanned <uint16>] [-PolicyLevel <uint32>] [-PolicyOwner {local |
pending-policy | policy}] [-Sacl {addchild | cascade | del | mod | none}]
[-SlotSpanOrientation {inline | transverse | unknown}] [-TargetDn <string>] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

## Get-UcsFirmwareUpgradeConstraint

このコマンドレットは管理対象オブジェクトの「FirmwareUpgradeConstraint」タイプに関する情報を取得するために使用します。この管理対象オブジェクトを使用して、アップグレード操作のファームウェア制約を定義します。

```
Get-UcsFirmwareUpgradeConstraint [-Dn <string>] [-MinVer <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -AdaptorFruCapProvider <AdaptorFruCapProvider> [-Dn
<string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -DiagSrvCapProvider <DiagSrvCapProvider> [-Dn
<string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -EquipmentBaseBoardCapProvider
<EquipmentBaseBoardCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -EquipmentBladeBiosCapProvider
<EquipmentBladeBiosCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -EquipmentBladeCapProvider <EquipmentBladeCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -EquipmentCatalogCapProvider
<EquipmentCatalogCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade
| del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -ChassisCapProvider <EquipmentChassisCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -EquipmentDbgPluginCapProvider
<EquipmentDbgPluginCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]
```

```
Get-UcsFirmwareUpgradeConstraint -FanModuleCapProvider <EquipmentFanModuleCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]
```

```

Get-UcsFirmwareUpgradeConstraint -FexCapProvider <EquipmentFexCapProvider> [-Dn
<string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentGemCapProvider <EquipmentGemCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentGraphicsCardCapProvider
<EquipmentGraphicsCardCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentHostIfCapProvider
<EquipmentHostIfCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade
| del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentIOCardCapProvider
<EquipmentIOCardCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade
| del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentIOExpanderCapProvider
<EquipmentIOExpanderCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentMgmtCapProvider <EquipmentMgmtCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentMgmtExtCapProvider
<EquipmentMgmtExtCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade
| del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentProcessorUnitCapProvider
<EquipmentProcessorUnitCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -PsuCapProvider <EquipmentPsuCapProvider> [-Dn
<string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}] [-Hierarchy]
[-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -RackUnitCapProvider <EquipmentRackUnitCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentSecurityUnitCapProvider
<EquipmentSecurityUnitCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

Get-UcsFirmwareUpgradeConstraint -EquipmentServerUnitCapProvider
<EquipmentServerUnitCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]

```

```

[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraint -EquipmentSwitchCapProvider
<EquipmentSwitchCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade
| del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs
<UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraint -EquipmentSwitchIOCardCapProvider
<EquipmentSwitchIOCardCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraint -EquipmentTpmCapProvider <EquipmentTpmCapProvider>
[-Dn <string>] [-MinVer <string>] [-Sacl {addchild | cascade | del | mod | none}]
[-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>] [-Ucs <UcsHandle[]>] [-Xml]
[<CommonParameters>]

```

## Get-UcsFirmwareUpgradeConstraintStorage

```

Get-UcsFirmwareUpgradeConstraintStorage [-Dn <string>] [-MinVer <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentLocalDiskCapProvider
<EquipmentLocalDiskCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentLocalDiskControllerCapProvider
<EquipmentLocalDiskControllerCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl
{addchild | cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty
<hashtable>] [-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentMemoryUnitCapProvider
<EquipmentMemoryUnitCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentMiniStorageCapProvider
<EquipmentMiniStorageCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentStorageEncCapProvider
<EquipmentStorageEncCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild |
cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentStorageNvmeSwitchCapProvider
<EquipmentStorageNvmeSwitchCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

    Get-UcsFirmwareUpgradeConstraintStorage -EquipmentStorageSasExpanderCapProvider
<EquipmentStorageSasExpanderCapProvider> [-Dn <string>] [-MinVer <string>] [-Sacl {addchild
| cascade | del | mod | none}] [-Hierarchy] [-Filter <string>] [-XtraProperty <hashtable>]
[-Ucs <UcsHandle[]>] [-Xml] [<CommonParameters>]

```



## 第 4 章

### Samples

---

この章は、次の項で構成されています。

- [すべてのグローバルポリシーをフェッチします \(97 ページ\)](#)
- [Cisco UCS コミュニティ \(97 ページ\)](#)
- [関連する Cisco UCS マニュアルとマニュアルフィードバック \(97 ページ\)](#)
- [マニュアルの入手方法およびテクニカルサポート \(98 ページ\)](#)

## すべてのグローバルポリシーをフェッチします

```
$id=(Get-UcsPowerToolConfiguration).InstallDir  
& "$id\Samples\Get-UcsGlobalPolicy.ps1" bgl-abcd18
```

Cisco UCS PowerTool サンプル スクリプトの詳細については、[Cisco コミュニティ](#)を参照してください。

## Cisco UCS コミュニティ

[Cisco UCS コミュニティ](#)はシスコの製品およびテクノロジーについて議論、情報共有、学習するためのプラットフォームです。[Cisco UCS コミュニティ](#) パートナー エコシステムと UCS の統合に関するブログ、ディスカッションフォーラム、およびドキュメントについては、<https://communities.cisco.com/ucsintegrations> にアクセスしてください。

## 関連するCiscoUCSマニュアルとマニュアルフィードバック

詳細については、次のリンクから関連資料を参照できます。

- [Cisco UCS Documentation Roadmap](#)
- [Release Bundle Contents for Cisco UCS Software, Release 2.1](#)

このマニュアルに関する技術的なフィードバック、または誤りや記載漏れに関する報告は、[ucs-docfeedback@external.cisco.com](mailto:ucs-docfeedback@external.cisco.com) に送信してください。ご協力をよろしくお願いいたします。

## マニュアルの入手方法およびテクニカル サポート

マニュアルの入手、サービス要求の提出、および追加情報の収集については、下記 URL の『*What's New in Cisco Product Documentation*』を参照してください。<http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html>

『*What's New in Cisco Product Documentation*』は、シスコの新規および改訂版の技術マニュアルの一覧も示し、RSS フィードとして購読できます。また、リーダー アプリケーションを使用してコンテンツをデスクトップに配信することもできます。RSS フィードは無料のサービスです。

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. シスコの商標の一覧については、[www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks) をご覧ください。掲載されている第三者の商標はそれぞれの権利者の財産です。「パートナー」または「partner」という用語は、シスコと他社との間のパートナーシップ関係を意味するものではありません。(1110R)

このマニュアルで使用している IP アドレスは、実際のアドレスを示すものではありません。マニュアル内の例、コマンド出力、および図は、説明のみを目的として使用されています。説明の中に実際のアドレスが使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。