

ESXi のインストールとセットアップ

Update 3

VMware vSphere 8.0

VMware ESXi 8.0

VMware by Broadcom の Web サイトで最新の技術ドキュメントを確認できます

<https://docs.vmware.com/jp/>

VMware by Broadcom

3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2018-2024 Broadcom. All Rights Reserved. 「Broadcom」という語表現は、Broadcom Inc. およびその子会社のいずれかまたは両方を指します。詳細については、<https://www.broadcom.com> を参照してください。本書に記載されるすべての商標、製品名、サービス マークおよびロゴは、各社に帰属します。

目次

1	『VMware ESXi のインストールとセットアップ』について	5
2	vSphere のインストールおよびセットアップの概要	6
	vSphere のインストールおよびセットアップ プロセスの概要	6
	ESXi の評価モードとライセンス モードについて	8
3	VMware vSphere Distributed Services Engine [®] について	9
	VMware vSphere Distributed Services Engine を使用した高可用性	11
	VMware vSphere Distributed Services Engine のエラー処理、フェイルオーバー、ロールバック	12
4	ESXi のインストールおよび設定	14
	ESXi の要件	14
	ESXi のシステム ストレージの概要	14
	ESXi のハードウェア要件	19
	リモート管理アプリケーションの使用	22
	ESXi のパフォーマンスを改善するための推奨事項	23
	ESXi ホストの送受信ファイアウォール ポート	24
	システム ログに必要な空き容量	24
	VMware Host Client のシステム要件	25
	ESXi のパスワードとアカウントのロックアウト	25
	ESXi のインストールの準備	28
	ESXi インストーラのダウンロード	28
	ESXi のインストールに必要な情報	28
	ESXi インストーラを起動するためのメディア オプション	29
	vSphere ESXi Image Builder を使用したインストールのカスタマイズ	33
	vSphere ESXi Image Builder の仕組み	34
	ImageProfile、SoftwarePackage、および ImageProfileDiff の各オブジェクトの構造	40
	vSphere ESXi Image Builder の構成	44
	VMware.Image Builder コマンドレットの使用	46
	ESXi イメージ プロファイル タスク	48
	PowerCLI コマンドレットを使用した vSphere ESXi Image Builder のワークフロー	68
	ESXi のインストール	75
	ESXi の対話形式のインストール	75
	スクリプトを使用した ESXi のインストール	79
	ネットワーク デバイスから ESXi ホストを起動する方法	101
	vSphere Auto Deploy を使用した ESXi のインストール	116
	vSphere Auto Deploy のトラブルシューティング	215
	ESXi の設定	223

初期状態の ESXi 構成	223
ダイレクト コンソール ユーザー インターフェイスを使用した、ESXi Shell および SSH アクセスの有効化	227
システム管理者アカウントのパスワードの設定	228
BIOS の起動設定の構成	228
ネットワーク設定の構成	230
管理ネットワークのテスト	234
管理エージェントの再起動	235
管理ネットワークの再起動	235
デバイスとネットワークへの接続のテスト	236
標準スイッチのリストア	236
システム ログの設定	236
ホスト イメージ プロファイル許容レベルの設定	258
ESXi のすべてのカスタム パッケージの削除	258
ESXi 構成ファイルの変更	259
ESXi での非 ASCII 文字のサポートの無効化	259
システム構成のリセット	260
ESXi のインストールおよび設定後	260
ESXi ホストへのライセンス供与	261
ESXi ホストのライセンス キーの記録	261
vSphere Client を使用した ESXi ホストのライセンス キーの表示	262
ダイレクト コンソールを使用した ESXi ライセンス キーへのアクセス	262
システム ログの表示	262
5 ESXi の起動のトラブルシューティング	264
別のホストと起動ディスクを共有する場合の、起動時の予期しないホストの停止	264
UEFI モードでの ESXi のインストール後、ホストの起動に失敗する	265
6 ESXi ホストの廃止	266

『VMware ESXi のインストールとセットアップ』について

1

『VMware ESXi のインストールとセットアップ』では、VMware ESXi™ のインストール方法と設定方法について説明します。

VMware では、多様性の受け入れを尊重しています。お客様、パートナー企業、社内コミュニティとともにこの原則を推進することを目的として、多様性の受け入れに適切でない言葉遣いを削除するため、このガイドを更新しました。

対象読者

『VMware ESXi のインストールとセットアップ』は、ESXi のインストールと設定を行う、経験豊かな管理者の方を対象としています。

ここに記載されている情報は、Windows または Linux のシステム管理者としての経験があり、仮想マシン テクノロジーおよびデータセンターの運用に詳しい方を対象としています。Image Builder と VMware vSphere® Auto Deploy™ の使用方法に関する情報は、Microsoft PowerShell および VMware vSphere® PowerCLI™ の使用経験のある管理者を対象としています。

vSphere のインストールおよびセットアップの概要

2

vSphere 8.0 には、インストールとセットアップのさまざまなオプションがあり、対応する一連のタスクを定義できます。

vSphere の 2 つのコア コンポーネントは ESXi と vCenter Server です。ESXi は、仮想マシンと仮想アプライアンスを作成して実行できる仮想化プラットフォームです。vCenter Server は、ネットワークで接続されている ESXi ホストを統合管理するサービスです。vCenter Server では、複数のホストのリソースをリソース プールにまとめて管理できます。

vCenter Server アプライアンスは、vCenter Server および vCenter Server コンポーネントを実行するために最適化された事前構成済みの仮想マシンとしてデプロイします。vCenter Server アプライアンスは、ESXi ホストまたは vCenter Server インスタンスにデプロイできます。

vCenter Server のインストール プロセスの詳細については、『vCenter Server のインストールとセットアップ』を参照してください。

次のトピックを参照してください。

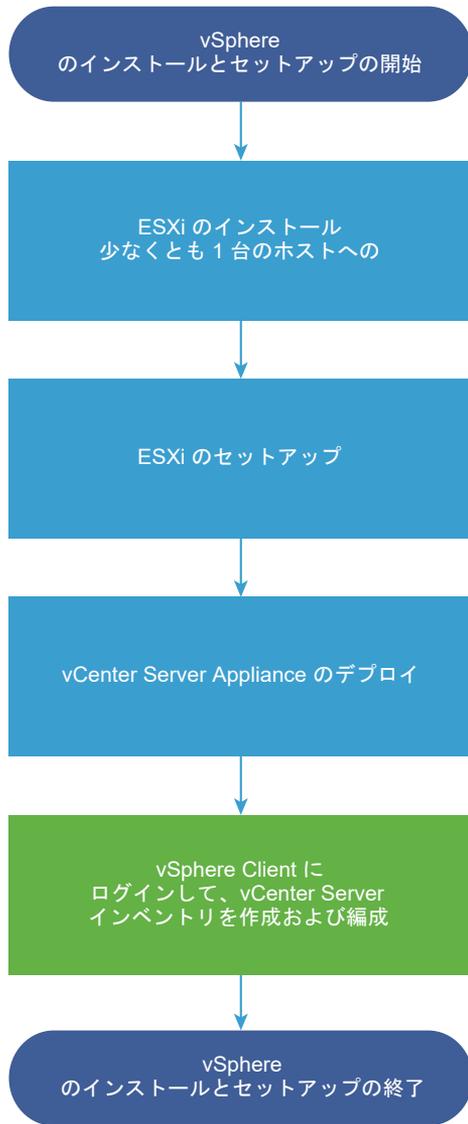
- [vSphere のインストールおよびセットアップ プロセスの概要](#)
- [ESXi の評価モードとライセンス モードについて](#)

vSphere のインストールおよびセットアップ プロセスの概要

vSphere は、インストールおよび設定する複数のコンポーネントを備えた高機能な製品です。vSphere を正しくデプロイするために、必要となる一連のタスクについて理解します。

vSphere のインストールには、次のタスクが含まれます。

図 2-1. vSphere のインストールとセットアップのワークフロー



- 1 vSphere リリース ノートを参照します。
- 2 ESXi をインストールします。
 - a システムがハードウェアの最小要件を満たしていることを確認します。ESXi の要件を参照してください。
 - b ESXi インストール オプションとして、対話型、スクリプト、または vSphere Auto Deploy のいずれかを使用するか決定します。
 - c ESXi インストーラを配置および起動する場所を決定します。ESXi インストーラを起動するためのメディア オプションを参照してください。PXE ブートでインストーラを起動する場合は、ネットワークの PXE インフラストラクチャが正しく設定されていることを確認します。ESXi インストーラのネットワーク ブートを参照してください。
 - d ESXi のインストールに必要な情報を記載したワークシートを作成します。ESXi のインストールに必要な情報を参照してください。

- e ESXi をインストールします。
 - ESXi の対話形式のインストール
 - ESXi のスクリプトによるインストール

注： vSphere Auto Deploy を使用して ESXi ホストをプロビジョニングすることもできます。vSphere Auto Deploy は vCenter Server とともにインストールされます。Auto Deploy を使用して ESXi ホストをプロビジョニングするには、vCenter Server をインストールする必要があります。

- 3 ESXi の起動設定およびネットワーク設定、ダイレクト コンソール、およびその他の設定を行います。ESXi の設定および ESXi のインストールおよび設定後を参照してください。
- 4 ログ ファイル用のディスク ストレージを確保するため、リモート ログ作成用の Syslog サーバの設定を検討します。リモート ホスト上でログ作成を設定することは、ローカル ストレージ容量の少ないホストでは特に重要です。システム ログに必要な空き容量および ESXi ホストでの Syslog の構成を参照してください。
- 5 vCenter Server をインストールします。

詳細については、『vCenter Server インストールおよびセットアップ ガイド』を参照してください。

ESXi の評価モードとライセンス モードについて

評価モードを使用して、vSphere Enterprise Plus ライセンスと同等の一連の機能を確認できます。

評価モードを使用すると、ESXi ホストの機能セット全体を評価検討できます。評価モードでは、vSphere Enterprise Plus のライセンスと同等の機能セットが提供されます。評価モードの有効期限が切れる前に、使用中のすべての機能をサポートするライセンスをホストに割り当てる必要があります。たとえば、評価モードでは、vSphere vMotion テクノロジー、vSphere HA 機能、vSphere DRS 機能などを使用できます。これらの機能を継続して使用する場合は、それらをサポートするライセンスを割り当てる必要があります。

ESXi ホストのインストール可能なバージョンは、必ず評価モードでインストールされます。ESXi Embedded は、ハードウェア ベンダーによって内部ストレージ デバイスにあらかじめインストールされています。これは、評価モードになっているか、あらかじめライセンス付与されています。

評価期間は 60 日間で、ESXi ホストをオンにしたときに始まります。この 60 日の評価期間中はいつでも、ライセンス モードから評価モードに変更できます。評価期間のうち残りの期間は、すでに使用した時間だけ減少します。

たとえば、ESXi ホストを 20 日間評価モードで使用し、その後 vSphere Standard Edition のライセンス キーをそのホストに割り当てたとします。ホストを評価モードに再設定した場合、そのホストの機能セット全体は、残りの評価期間である 40 日間評価検討できます。

ESXi ホストの場合、ライセンスまたは評価期間の有効期限が切れると、ホストが vCenter Server から切断されます。パワーオン状態のすべての仮想マシンの実行は継続されますが、パワーオフ状態の仮想マシンをパワーオンすることはできません。使用中の機能の現在の設定を変更することはできません。ライセンスの有効期限が切れる前に使用していない機能は使用することはできません。

ESXi ホストのライセンス管理については、『vCenter Server およびホストの管理』のドキュメントを参照してください。

VMware vSphere Distributed Services Engine[®] について

3

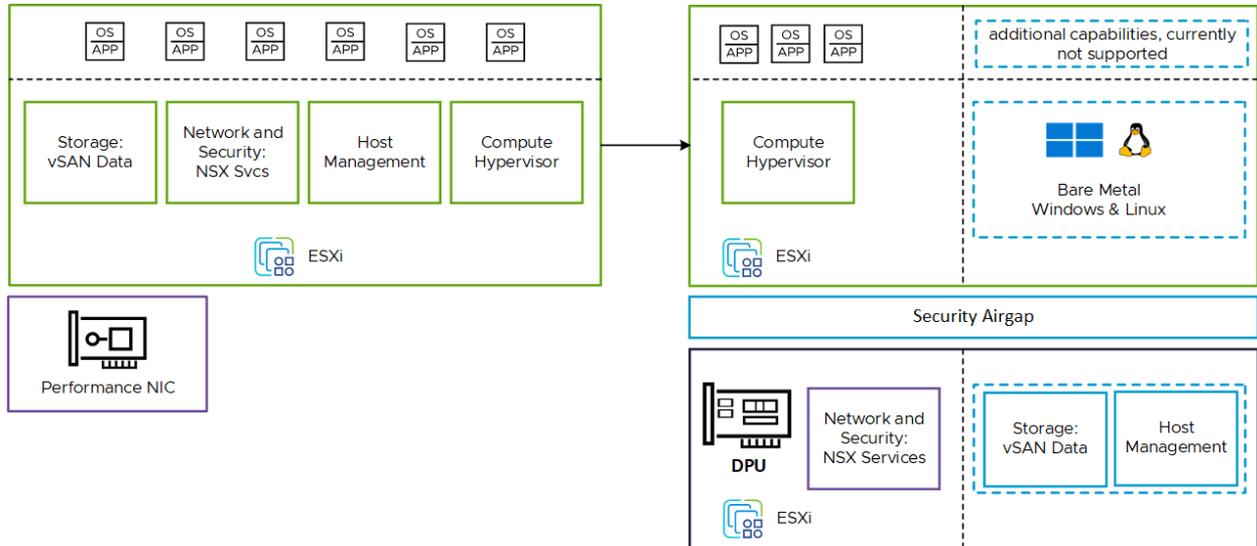
vSphere Distributed Services Engine[®] は、vSphere と VMware Cloud Foundation[®] で DPU を使用できるようにする vSphere の主要機能です。

vSphere 8.0 を使用すると、増え続けるスループットと遅延のニーズを満たすために、ワークロード パフォーマンスを飛躍的に向上させることができます。vSphere Distributed Services Engine を使用すると、インフラストラクチャ サービスは、ESXi ホストで使用可能なさまざまなコンピューティング リソースに分散され、ネットワーク機能は DPU にオフロードされます。そのような機能は、アプリケーションを複数の独立した共同サービスに分割することを模索する、マイクロサービス アーキテクチャ方式を使用して開発される、最新のアプリケーションで十分に機能します。複雑性が高まることにより、CPU が新たに必要になります。たとえば、これらのマイクロサービスに対しストレージ要求を処理したり、ネットワークトラフィックを往復すると、実際のワークロードの CPU サイクルが少なくなります。このコンテキストでは、DPU などの専用のアクセラレータが新しいコンピューティング負荷を引き受け、インフラストラクチャのパフォーマンスと効率を向上させることができます。

vSphere Distributed Services Engine で、DPU はネットワークのパフォーマンスを加速し、データスループットを増やすことができます。同時に、既存の Day-0、Day-1、および Day-2 の vSphere エクスペリエンスは変わらないため、DPU のライフサイクル管理の運用上の負荷はなくなります。vSphere Distributed Services Engine は、NVIDIA および AMD の DPU のほか、Dell、HPE、Lenovo、Fujitsu のサーバ設計によってサポートされています。vSphere Distributed Services Engine は、DPU が事前にインストールされているサーバで使用できます。

vSphere 8.0 以降では、コア CPU で実行されている機能を DPU にオフロードして、ネットワークとセキュリティのパフォーマンスを大幅に向上させることができます。「vSphere アーキテクチャの進化」図に示すように、DPU はストレージ オフロードやベアメタル管理などの追加機能も扱えますが、それらの追加機能は現在サポートされていません。

図 3-1. vSphere アーキテクチャの進化。



vSphere Distributed Services Engine は、VMware vSphere Distributed Switch を DPU、VMware NSX ネットワークおよび Observability に導入することで、DPU にインフラストラクチャ機能をオフロードし、高速化します。これにより、複雑なネットワーク タップがなくともネットワーク インフラストラクチャのポトルネットワークをプロアクティブに監視して識別し、軽減できるようになります。DPU は、インフラストラクチャ機能を拡張するための新しい制御ポイントになり、エージェントレスでワークロード ドメインから分離されたセキュリティ制御を有効にします。

vSphere Distributed Services Engine を使用すると、次の操作を実行できます。

- ESXi イメージを x86 サーバと接続したサポート対象の DPU に同時にインストールして更新し、統合された vSphere ワークフローによる DPU ライフサイクル管理の運用上のオーバーヘッドを軽減します。詳細については、「[vSphere Lifecycle Manager と VMware vSphere Distributed Services Engine の使用](#)」を参照してください。
- DPU ハードウェア アラートのアラームを設定して、使いやすい vCenter Server のインターフェイスからコア、メモリ、およびネットワーク スループットのパフォーマンス メトリックを監視します。新しいツールは不要です。詳細については、[CPU \(DPU\)](#) と [メモリ \(DPU\)](#) を参照してください。
- DPU 上の vSphere Distributed Switch を高速化してネットワーク パフォーマンスを向上させ、使用可能な CPU サイクルを利用して、ESXi ホストあたりのワークロード統合を強化します。詳細については、[ネットワーク オフロード機能について](#) と [vSphere Distributed Switch の作成](#) を参照してください。
- DPU が接続されているホスト上で実行される仮想マシンに対する vSphere DRS および vSphere vMotion のサポートにより、仮想マシンの移植性を犠牲にせずにパススルーの恩恵が受けられます。詳細については、[DPU の同種クラスタ](#) を参照してください。
- ゼロトラスト セキュリティにより、インフラストラクチャのセキュリティを強化します。詳細については、「[vSphere Distributed Services Engine のセキュリティ ベスト プラクティス](#)」を参照してください。

vSphere Distributed Services Engine では、個別の ESXi ライセンスは不要です。他のネットワークから隔離されている内部ネットワークは、DPU を ESXi ホストに接続します。ESXi 8.0 サーバビルドは、x86 と DPU の両方のコンテンツが含まれる統合イメージです。vSphere システムでは、インストールおよびアップグレード中、そしてネットワーク、ストレージ、およびホスト プロファイル ワークフローで、DPU が新しいオブジェクトとして表示されます。

次のトピックを参照してください。

- [VMware vSphere Distributed Services Engine を使用した高可用性](#)
- [VMware vSphere Distributed Services Engine のエラー処理、フェイルオーバー、ロールバック](#)

VMware vSphere Distributed Services Engine を使用した高可用性

ESXi 8.0 Update 3 では、2 つのデータ処理ユニット (DPU) を使用する VMware vSphere Distributed Services Engine インストールを選択して、高可用性を実現できます。

単一の DPU を使用した vSphere システムでは、そのデバイスがネットワーク機能など DPU にオフロードされたワークロードの単一点障害となり、データと生産性に影響を与えることがあります。ESXi 8.0 Update 3 では、ハードウェアの冗長性と回復性を提供する DPU が事前にインストールされているサーバでも、vSphere Distributed Services Engine を使用できます。

2 つの DPU をアクティブ/スタンバイ モードで使用して、高可用性を実現できます。このような構成は、DPU の 1 つが失敗した場合に冗長性を提供します。高可用性構成では、両方の DPU が同じ NSX バッキングされた vSphere Distributed Switch に割り当てられます。たとえば、DPU-1 は vSphere Distributed Switch の vmnic0 に接続され、DPU-2 は同じ vSphere Distributed Switch の vmnic2 および vmnic3 に接続されます。

また、2 つの DPU を独立したデバイスとして使用して、ESXi ホストあたりのオフロード容量を増やすこともできます。各 DPU は個別の vSphere Distributed Switch に接続されており、このような構成では DPU 間のフェイルオーバーはありません。

デュアル DPU システムでは、NVIDIA または Pensando デバイスを使用できます。ESXi 8.0 Update 3 では、デュアル DPU システムが Lenovo サーバ設計でサポートされています。デュアル DPU サーバ上の DPU デバイスは、同じベンダー、同じハードウェアバージョン、および同じファームウェアのすべての面で同一である必要があります。VMware vSphere Distributed Services Engine の現在のベンダーとサーバ設計のリストについては、[VMware 互換性ガイド](#)を参照してください。

2 DPU を使用した VMware vSphere Distributed Services Engine のインストール

vSphere Distributed Services Engine では、個別の ESXi ライセンスは不要です。ESXi 8.0 Update 3 サーバビルドは、x86 と DPU の両方のコンテンツを含む統合イメージであり、x86 と DPU コンテンツを個別にインストールすることはできません。対話型またはスクリプト化された両方の DPU のインストール手順も並行して実行され、単一の DPU システムと比較してパフォーマンスの低下が最小限に抑えられます。

vSphere 8.0 Update 3 では、Dell または Lenovo から 2 つの DPU を使用して事前にインストールされたサーバ構成を取得したり、Dell または Lenovo でサポートされているデュアル DPU サーバ上の単一の DPU システムに 2 つ目の DPU を追加したりできます。

注： いずれの場合も、新しく追加された DPU だけでなく、システムで完全な新規 ESXi 8.0 Update 3 インストールを実行する必要があります。

インストールの詳細については、「[対話形式での ESXi のインストール](#)」および「[ESXi のインストールに使用されるインストール スクリプトとアップグレード スクリプト](#)」を参照してください。

VMware vSphere Distributed Services Engine のエラー処理、フェイルオーバー、ロールバック

VMware vSphere Distributed Services Engine をインストールする前に、エラー処理、フェイルオーバー、ロールバックの各オプションを確認します。

エラー処理

ESXi ホストで x86 と DPU のいずれかのコンテンツのインストールに失敗すると、インストール手順全体が失敗としてマークされます。

DPU は常に同一のソフトウェア状態を維持することが想定されていますが、コンポーネントのインストールやアップグレードなどのライフサイクル処理中に万が一エラーが発生した場合、その処理が特定の DPU では成功し、他の DPU では失敗することがあります。各ライフサイクル処理は DPU ごとの境界内で発生するため、エラーは他の DPU の状態には影響しませんが、それでもインストール全体の結果は失敗としてマークされます。

対話型インストール時に、vSphere Lifecycle Manager ワークフローで ESXCLI を使用すると、処理に失敗した DPU に関する情報が表示されます。

DPU エラーが発生する場合は、インストールが正常に完了した後に、影響を受ける ESXi ホストを再起動することをお勧めします。ホストから DPU に引き続きアクセスできる場合は、一般的なログバンドルの収集で十分にトラブルシューティングを行えます。ホストから DPU にアクセスできない場合は、BMC、iLO、または iDRAC インターフェイスから DPU にログインすると、トラブルシューティングログが得られます。

フェイルオーバー

vSphere 8.0 Update 3 のフェイルオーバー サポート対象は、DPU 内のソフトウェア エラーや、ケーブルの切断などの DPU の物理的切断が原因で動作しなくなった 1 つの DPU に限られます。PCI (Peripheral Component Interconnect) レベルのエラーによるフェイルオーバーはサポートされていません。

ロールバック

ロールバックは、ESXi 起動時のジャンプスタート フェーズの前に障害が発生した場合に、システムを以前の動作状態にリストアするためのベスト エフォート型メカニズムです。起動中にエラーが発生した場合、x86 サーバと接続されているサポート対象 DPU の両方でロールバックが自動的に実行されます。また、手動ロールバックを選択することもできます。その場合は、ブートローダーが起動する前に **Shift + R** を押して、以前の良好な状態に戻します。

ジャンプスタート フェーズの開始後に障害が発生しても、ロールバックは実行されません。

表 3-1. VMware vSphere Distributed Services Engine のインストールに関するロールバック シナリオ

使用例	必要な再起動数
両方の DPU が正常に起動する。ESXi が正常に起動しない。	2
両方の DPU が正常に起動しない。ESXi が正常に起動する。	1
いずれかの DPU が他の DPU よりも前のバージョンで起動し、ESXi が正常に起動する。	2
いずれかの DPU が他の DPU よりも前のバージョンで起動し、ESXi が正常に起動しない。	2

ESXi のインストールおよび設定

4

ESXi を物理ハードウェアにインストールして設定し、仮想マシンのプラットフォームとして動作させることができます。

次のトピックを参照してください。

- [ESXi の要件](#)
- [ESXi のインストールの準備](#)
- [vSphere ESXi Image Builder を使用したインストールのカスタマイズ](#)
- [ESXi のインストール](#)
- [ESXi の設定](#)
- [ESXi のインストールおよび設定後](#)

ESXi の要件

ESXi をインストールするか、このバージョンにアップグレードするには、システムが固有のハードウェアおよびソフトウェア要件を満たしている必要があります。

ESXi のシステム ストレージの概要

ESXi 8.0 には、デバッグを容易にしながら、パーティションとサードパーティ コンポーネントの柔軟な管理を可能にするシステム ストレージ レイアウトがあります。

ESXi のシステム ストレージ

ESXi 8.0 のシステム ストレージ レイアウトは、次の 4 つのパーティションから構成されます。

表 4-1. ESXi のシステム ストレージのパーティション。

パーティション	用途	タイプ
システム起動	ブートローダーと EFI モジュールを格納します。	FAT16
ブートバンク 0	ESXi 起動モジュールを格納するシステム領域です。	FAT16

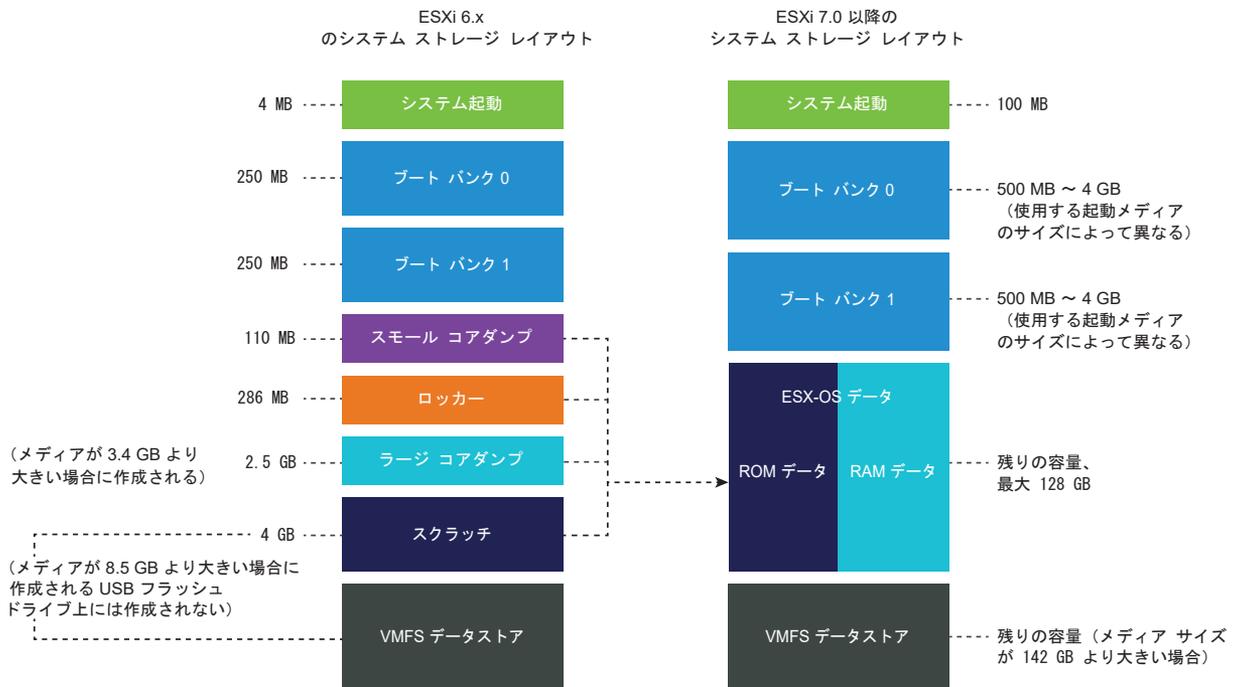
表 4-1. ESXi のシステム ストレージのパーティション。 (続き)

パーティション	用途	タイプ
ブート バンク 1	ESXi 起動モジュールを格納するシステム領域です。	FAT16
ESX-OSData	<p>追加のモジュールを格納する統合された場所として機能します。 起動および仮想マシンには使用されません。</p> <p>レガシーの /scratch パーティション、VMware Tools のロッカー パーティション、およびコア ダンプの宛先を統合します。</p> <p>注意： インストール メディアが USB または SD カード デバイスの場合、ベスト プラクティスは、ESXi ホスト間で共有されていないパーシステント ストレージ デバイス上に ESX-OSData パーティションを作成することです。</p>	VMFS-L

ESX-OSData ポリリュームは、永続データと非永続データの 2 つのデータ カテゴリに大別されます。永続データには、書き込み頻度が低いデータ、たとえば、VMware Tools ISO、構成、コア ダンプなどが含まれます。

非永続データには、書き込み頻度が高いデータ、たとえば、ログ、VMFS グローバル トレース、vSAN エントリ パーシステンス デモン (EPD) データ、vSAN トレース、リアルタイム データベースなどが含まれます。

図 4-1. ESXi 8.0 の統合システム ストレージ



ESXi のシステム ストレージのサイズ

システム起動パーティションを除き、パーティションのサイズは、使用する起動メディアのサイズによって異なります。起動メディアが高耐久性メディアで、その容量が 142 GB を超える場合は、VMFS データストアが自動的に作成され、仮想マシン データが格納されます。

ESXi インストーラで構成された起動メディアの容量と自動サイズ変更を確認するには、vSphere Client を使用して [パーティションの詳細] ビューに移動します。代わりに ESXCLI (`esxcli storage filesystem list` コマンドなど) を使用することもできます。

表 4-2. ESXi のシステム ストレージのサイズ。使用する起動メディアとその容量に応じて異なります。

起動メディアのサイズ	8 ~ 10 GB	10 ~ 32 GB	32 ~ 128 GB	128 GB を超える
システム起動	100 MB	100 MB	100 MB	100 MB
ブートバンク 0	500 MB	1 GB	4 GB	4 GB
ブートバンク 1	500 MB	1 GB	4 GB	4 GB
ESX-OSData	残りの容量	残りの容量	残りの容量	最大 128 GB
VMFS データストア				メディア サイズが 142 GB より大きい場合の残りの容量

ESXi インストーラ起動オプション `systemMediaSize` を使用して、起動メディアのシステム ストレージ パーティションのサイズを制限できます。システムの占有量が小さく、システム ストレージ サイズとして最大値の 128 GB を必要としない場合は、最小で 32 GB に制限できます。`systemMediaSize` パラメータには、次の値を指定できません。

- min (32 GB、1 台のディスクまたは組み込みのサーバ用)
- small (64 GB、512 GB 以上の RAM を搭載したサーバ用)
- default (128 GB)
- max (使用可能なすべての容量を使用、マルチテラバイトのサーバ用)

選択した値は、システムの目的に適合している必要があります。たとえば、1 TB のメモリのあるシステムでは、システム ストレージに 64 GB 以上を使用する必要があります。インストール時に起動オプションを設定するには (例: `systemMediaSize=small`)、[インストール スクリプトまたはアップグレード スクリプトを開始するための起動オプションの入力](#)を参照してください。詳細については、ナレッジベースの記事 [KB81166](#) を参照してください。

ESXi のシステム ストレージ リンク

ESXi パーティションにアクセスする必要がある場合、サブシステムは、次のシンボリック リンクを使用して、これらのパーティションにアクセスします。

表 4-3. ESXi のシステム ストレージのシンボリック リンク。

システム ストレージのボリューム	シンボリック リンク
ブートバンク 0	/bootbank
ブートバンク 1	/altbootbank

表 4-3. ESXi のシステム ストレージのシンボリック リンク。 (続き)

システム ストレージのボリューム	シンボリック リンク
永続データ	/productLocker /locker /var/core /usr/lib/vmware/isoimages /usr/lib/vmware/floppies
非永続データ	/var/run /var/log /var/vmware /var/tmp /scratch

ストレージの動作

ESXi を起動すると、ホストが自動構成段階に入り、この間にシステム ストレージ デバイスがデフォルト設定で構成されます。

ESXi イメージのインストール後に ESXi ホストを再起動するとき、システム ストレージ デバイスがデフォルト設定で構成されます。ESXi 7.0 以降では、`autoPartition` オプションを有効にできます。これにより、レガシーの SD デバイスと USB デバイスを除き、使用可能なすべての空のデバイスが VMFS で自動的にフォーマットされます。デフォルトは `autoPartition=FALSE` です。この場合、サイズが 128 GB を超える起動デバイスのみが VMFS でフォーマットされます。詳細については、VMware ナレッジベースの記事 [KB77009](#) を参照してください。

注意： ESXi は空と思われるディスクを上書きします。ディスクに有効なパーティション テーブルまたはパーティションがない場合、そのディスクは空とみなされます。このようなディスクを利用するソフトウェアを使用している場合、特に従来のパーティション方式に代わり、または従来のパーティション方式に加えて論理ボリューム マネージャ (LVM) を使用している場合は、ESXi によってローカル LVM が再フォーマットされる可能性があります。ESXi を最初にパワーオンする前に、システム データをバックアップします。

ESXi ホストの起動元となるハード ドライブまたは USB デバイスでは、ディスク フォーマット ソフトウェアはハードウェア ベンダーが作成した既存の診断パーティションをそのまま保持します。残りの容量には、ソフトウェアは以下で説明するパーティションを作成します。

ESXi がホスト ドライブ上に作成するパーティション

フレッシュ インストールの場合、システム起動、ブートバンク、ESX-OSData 用に新しいパーティションがいくつか作成されます。ESXi のフレッシュ インストールは、MSDOS ベースのパーティションの代わりに、GUID パーティション テーブル (GPT) を使用します。インストーラは、ディスクのサイズに応じてさまざまなサイズのブートバンクを作成します。スクラッチ パーティションの詳細については、[スクラッチ パーティションについて](#)を参照してください。

インストーラは、インストール ディスクにのみ影響します。インストーラがサーバの他のディスクに影響することはありません。ディスクにインストールすると、インストーラによってディスク全体が上書きされます。インストーラでストレージを自動構成すると、インストーラはハードウェア ベンダーのパーティションを上書きしません。

VMFS データストアを作成するには、ESXi インストーラでインストール ディスクに最低 128 GB の空スペースが必要です。

たとえば、ローカル ストレージではなく共有ストレージ デバイスを使用する場合に、このデフォルト動作をオーバーライドする必要が生じる場合があります。自動のディスク フォーマットを防ぐためには、次の環境下で、ローカル ストレージ デバイスをホストから分離します。

- ホストを最初に起動する前。
- ホストをデフォルト構成にリセットした後で、ホストを起動する前。

自動ディスク フォーマットがすでに実行されている場合に、VMFS フォーマットをオーバーライドするには、データストアを削除することができます。『vCenter Server およびホストの管理』ドキュメントを参照してください。

スクラッチ パーティションについて

ESXi の新規インストールでは、自動構成フェーズ中に、ESX-OSDATA パーティションの一部としてスクラッチパーティションが作成されます。

注： 以前のバージョンから ESXi 7.0 以降にアップグレードされたホストのパーティションは、新規にインストールされた ESXi のパーティションとは大幅に異なります。ESXi 7.0 以降へのアップグレード プロセスでは、起動デバイスの再パーティショニングが行われ、元のコア ダンプ、ロッカー、およびスクラッチ パーティションが ESX-OSData ボリュームに統合されます。

スクラッチ パーティションは、サポート バンドルを作成するときに必要なシステム ログを保存するためのものです。スクラッチ パーティションがない場合、システム ログは ramdisk に保存されます。スクラッチ パーティションは作成されない場合に構成できます。デフォルトの構成をオーバーライドすることもできます。

スクラッチ パーティションは、SAN または NFS がマウントされたりリモート ディレクトリに作成できます。

vSphere Client からのスクラッチ パーティションの設定

スクラッチ パーティションが設定されていない環境では、スクラッチ パーティションの構成が必要になる場合があります（特に、ホストがメモリ不足の場合）。スクラッチ パーティションがない場合、システム ログは ramdisk に保存されます。

前提条件

スクラッチ パーティションに使用するディレクトリは、ホスト上に配置されている必要があります。

手順

- 1 vSphere Client から vCenter Server に接続します。
- 2 インベントリでホストを選択します。
- 3 [設定] タブをクリックします。
- 4 [システム] を選択します。
- 5 [システムの詳細設定] を選択します。

設定 [ScratchConfig.CurrentScratchLocation] にスクラッチ パーティションの現在の場所が表示されません。

- 6 [ScratchConfig.ConfiguredScratchLocation] テキスト ボックスに、このホストに一意のディレクトリパスを入力します。

たとえば、`/vmfs/volumes/DatastoreUUID/DatastoreFolder` のように入力します。

- 7 ホストを再起動して、変更内容を有効にします。

ESXi のハードウェア要件

ホストが、ESXi8.0 でサポートされるハードウェアの最小構成を満たしていることを確認します。

ハードウェアおよびシステム リソース

ESXi をインストールまたはアップグレードするには、ハードウェアとシステムのリソースが次の要件を満たしている必要があります。

- サポートされているサーバ プラットフォーム。サポートされているプラットフォームについては、<http://www.vmware.com/resources/compatibility> の『VMware 互換性ガイド』を参照してください。
- ESXi8.0 では、少なくとも 2 つの CPU コアを持つホストが必要です。
- ESXi8.0 は、64 ビット x86 プロセッサのマルチコアを幅広くサポートしています。サポートされるプロセッサの詳細なリストについては、『VMware 互換性ガイド』(<http://www.vmware.com/resources/compatibility>) を参照してください。
- ESXi8.0 では、BIOS で CPU の NX/XD ビットを有効にする必要があります。
- ESXi8.0 には、少なくとも 8 GB の物理 RAM が必要です。一般的な本番環境で仮想マシンを実行するためには、少なくとも 12 GB の RAM を用意します。
- 64 ビット仮想マシンをサポートするには、ハードウェア仮想化のサポート (Intel VT-x または AMD RVI) を x64 CPU で有効にする必要があります。
- 1 つ以上のギガビットまたはより高速なイーサネット コントローラ。サポートされているネットワーク アダプタ モデルについては、<http://www.vmware.com/resources/compatibility> の『VMware 互換性ガイド』を参照してください。
- ESXi 8.0 には、HDD、SSD、NVMe などの 32 GB 以上のパーシステント ストレージの起動ディスクが必要です。起動デバイスを ESXi ホスト間で共有することはできません。
- 仮想マシン用にパーティショニングされていない容量がある SCSI ディスクまたはローカル (ネットワークに接続されていない) RAID LUN。
- SATA (Serial ATA) の場合、サポート対象の SAS コントローラまたはオンボード SATA コントローラを介して接続されるディスク。SATA ディスクは、ローカルではなくリモートと見なされます。これらのディスクは、リモートと見なされるため、デフォルトではスクラッチ パーティションとして使用されません。

注： ESXi ホストにある仮想マシンに SATA CD-ROM デバイスを接続することはできません。SATA CD-ROM デバイスを使用するには、IDE エミュレーション モードを使用する必要があります。

ストレージ システム

サポートされるストレージ システムについては、<http://www.vmware.com/resources/compatibility> で『VMware 互換性ガイド』を参照してください。ESXi 8.0 以降では、ファイバ チャネル オーバー イーサネット (FCoE) に対してソフトウェア アダプタを使用することはできません。ハードウェア FCoE アダプタのみ使用できます。

ESXi 起動要件

vSphere 8.0 では、レガシー BIOS のサポートは限定的なため、ESXi ホストは Unified Extensible Firmware Interface (UEFI) から起動することが推奨されます。UEFI を使用すると、ハード ドライブ、CD-ROM ドライブ、または USB メディアからシステムを起動できます。vSphere Auto Deploy では、UEFI を使用した ESXi ホストのネットワーク ブートとプロビジョニングがサポートされます。システムにサポートされているデータ処理ユニット (DPU) がある場合、DPU に ESXi をインストールして起動することは UEFI を使用してのみ可能です。サーバ プラットフォームでのレガシー BIOS のサポートを廃止する VMware の計画の詳細については、ナレッジベースの記事 <https://kb.vmware.com/s/article/84233> を参照してください。

ESXi は、2 TB を超える大きさのディスクから起動できます (使用しているシステム ファームウェアおよびアドイン カード上のファームウェアでサポートされている場合)。ベンダーのドキュメントを参照してください。

ESXi8.0 をインストールまたはアップグレードするためのストレージ要件

ESXi 8.0 インストールのパフォーマンスを最適化するには、起動デバイスに 32 GB 以上のパーシステント ストレージ デバイスを使用します。ESXi8.0 へのアップグレードには、最低 8 GB の起動デバイスが必要です。ローカル ディスク、SAN、または iSCSI LUN から起動する場合、ブートパーティション、ブートバンク、および VMFS-L ベースの ESX-OSData ポリリュームを含むシステム ストレージ ポリリュームの作成を可能にするには、32 GB 以上のディスクが必要です。ESX-OSData ポリリュームは、レガシーの /scratch パーティション、VMware Tools のロッカー パーティション、およびコア ダンプ先の役割で使用されます。

注: ESXi 8.0 では、ESX-OSData ポリリュームは統合パーティションと見なされ、/scratch や VMware Tools などの個別のコンポーネントは単一のパーシステント OSDATA パーティションとして統合されます。

ESXi 8.0 インストールのパフォーマンスを最適化するその他のオプションは、次のとおりです。

- ESX-OSData の最適なサポートのための 128 GB 以上のローカル ディスク。このディスクには、ブートパーティション、ESX-OSData ポリリューム、および VMFS データストアが含まれます。
- 128 Terabytes Written (TBW) 以上をサポートするデバイス。
- 100 MB/秒以上のシーケンシャル書き込み速度を提供するデバイス。
- デバイス障害時に回復性を提供するために、RAID 1 ミラー デバイスを使用することを推奨します。

レガシー SD デバイスおよび USB デバイスは、次の制限付きでサポートされます。

- SD および USB デバイスは、ブートバンク パーティションではサポートされます。ESX-OSData パーティションの格納に SD デバイスおよび USB デバイスを使用することは推奨されません。ベスト プラクティスは、ESX-OSData ポリリュームの格納用に 32 GB 以上の独立したパーシステント ローカル デバイスを使用することです。パーシステント ローカル起動デバイスには、産業レベルの M.2 フラッシュ (SLC および MLC)、SAS、SATA、HDD、SSD、または NVMe デバイスを使用できます。パーシステント ローカル デバイスに最適な容量は 128 GB です。

- パーシステント ストレージを指定しないと、「セカンダリ パーシステント デバイスが見つかりません。SD カード/USB のみの構成のサポートは廃止されているため、インストールをパーシステント ストレージに移動してください」というアラームが表示されます。
- SD フラッシュ ストレージ デバイスに ESXi をインストールする特定のサーバ モデルでは、サーバ ベンダー承認の SD フラッシュ デバイスを使用する必要があります。検証済みデバイスのリストについては、partnerweb.vmware.com を参照してください。
- SD カードまたは USB を使用する環境の最新のガイダンスについては、ナレッジベースの記事 [KB85685](#) を参照してください。
- 適切な SD または USB 起動デバイスを選択する手順については、ナレッジベースの記事 [KB82515](#) を参照してください。

7.x より前のバージョンから ESXi 8.0 へのアップグレード プロセスによって起動デバイスの再パーティショニングが行われ、元のコア ダンプ、ロッカー、およびスクラッチ パーティションが ESX-OSData ボリュームに統合されます。

再パーティショニング プロセスで、次のイベントが発生します。

- カスタム コア ダンプの転送先が設定されていない場合、デフォルトのコア ダンプの場所は ESX-OSData ボリューム内のファイルになります。
- 4 GB の VFAT スクラッチ パーティションにログ ファイルを格納するように Syslog サービスが構成されている場合、`var/run/log` のログ ファイルは ESX-OSData ボリュームに移行されます。
- VMware Tools はロッカー パーティションから移行され、パーティションはワイプされます。
- コア ダンプ パーティションはワイプされます。スクラッチ パーティションに格納されているアプリケーションのコア ダンプ ファイルは削除されます。

注： 起動デバイスの再パーティショニング プロセスが理由で、ESXi 8.x から 7.x よりも前のバージョンの ESXi にロールバックすることはできません。バージョン 8.0 へのアップグレード後に 7.x よりも前のバージョンの ESXi を使用するには、アップグレード前に起動デバイスのバックアップを作成し、バックアップから ESXi 起動デバイスをリストアする必要があります。bootbank パーティションに一切の変更が加えられておらず、破損したパーティションが検出されなければ、ESXi 8.x から 7.x へのロールバックは可能です。

USB デバイスまたは SD デバイスを使用してアップグレードを実行する場合、使用可能なパーシステント ディスクまたは SAN LUN に ESX-OSData 領域を割り当てるのがベスト プラクティスです。パーシステント ストレージまたは SAN LUN が使用可能でない場合、ESX-OSData が RAM ディスク上に自動的に作成されます。VMFS は、ESX-OSData パーティションにも使用できます。

アップグレード後、ESX-OSData が RAM ディスクに存在し、その後の起動時に新しいパーシステント デバイスが検出され、このデバイスの設定が `autoPartition=True` の場合、ESX-OSData が新しいパーシステント デバイス上に自動的に作成されます。ESX-OSData はパーシステント ストレージ間を自動的に移動するわけではありませんが、サポートされているストレージ上の ESX-OSData の場所を手動で変更できます。

`/scratch` を再構成するには、[vSphere Client](#) からのスクラッチ パーティションの設定を参照してください。

ESXi システム パーティションのサイズを構成するには、`systemMediaSize` オプションを使用します。詳細については、ナレッジベースの記事 <https://kb.vmware.com/s/article/81166> を参照してください。

Auto Deploy インストールの場合、インストーラはスクラッチ領域を使用可能なローカル ディスクまたはデータストアに割り当てようとします。ローカル ディスクもデータストアも検出されなかった場合、インストールは失敗します。

SAN から起動する環境または Auto Deploy を使用する環境では、ESXi ホストごとに、ESX-OSData ポリリュームを個別の SAN LUN に設定する必要があります。

リモート管理アプリケーションの使用

リモート管理アプリケーションによって、リモートの場所にあるサーバに ESXi をインストールできます。

インストールがサポートされているリモート管理アプリケーションには、HP iLO (Integrated Lights-Out)、DRAC (Dell Remote Access Card)、IBM 管理モジュール (MM)、Fujitsu iRMC、RSA II (Remote Supervisor Adapter II) があります。リモート管理アプリケーションのサポートについては、ベンダーに問い合わせてください。

リモート管理アプリケーションを使用して、リモートからの ESXi のインストールを、対話的に行ったり、スクリプトを使用して行うことができます。

リモート管理アプリケーションを使用して ESXi をインストールする場合、ピーク容量で稼働しているシステムまたはネットワークでは、仮想 CD が破壊されるという問題に直面することがあります。ISO イメージからのリモートインストールが失敗した場合は、物理 DVD メディアからインストールを実行します。

サポートされているリモート管理のサーバ モデルとファームウェア バージョン

リモート管理アプリケーションを使用して、ESXi のインストールまたはアップグレード、またはホストのリモート管理を実行できます。

表 4-4. サポートされているリモート管理のサーバ モデルとファームウェアの最小バージョン

リモート管理サーバ モデル	ファームウェアのバージョン	Java
Dell DRAC 9	6.0.30.00	該当なし
Dell DRAC 7	1.30.30 (ビルド 43)	1.7.0_60-b19
Dell DRAC 6	1.54 (ビルド 15)、1.70 (ビルド 21)	1.6.0_24
Dell DRAC 5	1.0、1.45、1.51	1.6.0_20、1.6.0_203
Dell DRAC 4	1.75	1.6.0_23
Fujitsu iRMC S5	1.10P	1.7.0_60-b19
Fujitsu iRMC S6	1.06S	該当なし
HP iLO	1.81、1.92	1.6.0_22、1.6.0_23
HP iLO 2	1.8、1.81	1.6.0_20、1.6.0_23
HP iLO 3	1.28	1.7.0_60-b19
HP iLO 4	1.13	1.7.0_60-b19

表 4-4. サポートされているリモート管理のサーバモデルとファームウェアの最小バージョン（続き）

リモート管理サーバモデル	ファームウェアのバージョン	Java
HP iLO 5	2.72	該当なし
IBM RSA 2	1.03, 1.2	1.6.0_22

ESXi のパフォーマンスを改善するための推奨事項

パフォーマンスを向上させるには、必要最小限以上の容量を持つ RAM と複数の物理ディスクがある堅牢なシステムで ESXi をインストールまたはアップグレードします。

ESXi システムの要件については、[ESXi のハードウェア要件](#)を参照してください。

表 4-5. パフォーマンス向上に関する推奨事項

システム要素	推奨
RAM	<p>ESXi ホストには、一般的なサーバよりも多くの RAM が必要です。ESXi 8.0 には、少なくとも 8 GB の物理 RAM が必要です。ESXi の機能を十分に活用し、一般的な本番環境で仮想マシンを実行するために、少なくとも 12 GB の RAM を提供します。ESXi ホストに、複数の仮想マシンを同時に実行するのに十分な容量の RAM が必要です。次の例を使用して、ESXi ホストで実行されている仮想マシンが必要とする RAM の計算方法を示します。</p> <p>Red Hat Enterprise Linux または Windows XP がインストールされた 4 台の仮想マシンが動作するには、ベースライン パフォーマンスを保つだけでも最低 3 GB の RAM が必要です。この数値には、仮想マシン用の 1,024 MB、オペレーティング システムごとに最小限必要な 256 MB（ベンダーが推奨する値）が含まれます。</p> <p>これらの 4 台の仮想マシンを 512 MB の RAM で実行するには、ESXi ホストに 4 GB の RAM が必要です（仮想マシン用の 2,048 MB を含む）。</p> <p>これらの計算値には、各仮想マシンに可変オーバーヘッド メモリを使用することで節約できるメモリ容量は含まれません。『vSphere リソース管理』を参照してください。</p>
仮想マシン専用のファスト イーサネット アダプタ	<p>管理ネットワークと仮想マシン ネットワークを異なる物理ネットワーク カードに配置します。Intel PRO 1000 アダプタなど仮想マシン専用のギガビット イーサネット カードを使用すると、高ネットワーク トラフィックでの仮想マシンへのスループットが向上します。</p>
ディスクの場所	<p>仮想マシンで使用されるすべてのデータを仮想マシンに割り当てられている物理ディスク上に配置します。ESXi の起動イメージがあるディスクに仮想マシンを配置しないほうが、パフォーマンスが向上します。物理ディスクは、すべての仮想マシンによって使用されるディスク イメージを保存できる大きさを備えている必要があります。</p>

表 4-5. パフォーマンス向上に関する推奨事項（続き）

システム要素	推奨
VMFS6 パーティショニング	ESXi インストーラでは、最初に検出された空のローカル ディスク上に初期 VMFS ボリュームが作成されます。ディスクを追加するか、元の構成を変更するには、vSphere Client を使用します。この方法では、パーティションの開始セクタが 64K セクタごとの割り当てになるため、ストレージ パフォーマンスが改善されます。 注： SAS のみの環境では、インストーラがディスクをフォーマットしない場合があります。一部の SAS ディスクは、ディスクがローカルかリモートかを識別することができません。インストール後、vSphere Client を使用して VMFS を設定できます。
プロセッサ	プロセッサの処理速度が速いほど、ESXi のパフォーマンスが向上します。特定のワークロードでは、キャッシュが大きい方が ESXi のパフォーマンスが向上します。
ハードウェア互換性	サーバ内で、ESXi ドライバのサポート対象になっているデバイスを使用します。 http://www.vmware.com/resources/compatibility の『ハードウェア互換性ガイド』を参照してください。

ESXi ホストの送受信ファイアウォール ポート

vSphere Client または VMware Host Client を使用して、各サービスのファイアウォール ポートを開閉します。

ESXi には、デフォルトで有効になっているファイアウォールが含まれています。インストール時、ESXi ファイアウォールは、受信トラフィックと送信トラフィックをブロックするように構成されています。ただし、ホストのセキュリティ プロファイルで有効なサービスのトラフィックは除外されます。ESXi ファイアウォールでサポートされているポートとプロトコルのリストについては、<https://ports.vmware.com/>の VMware Ports and Protocols Tool™ を参照してください。

VMware Ports and Protocols Tool では、デフォルトでインストールされているサービスのポート情報が一覧表示されます。他の VIB をホストにインストールすると、追加のサービスおよびファイアウォール ポートが使用可能になる場合があります。この情報は、主に vSphere Client に表示されるサービスに関するものですが、VMware Ports and Protocols Tool にはそれ以外のポートも含まれています。

システム ログに必要な空き容量

hostd、vpxa、および fdm の各ログに対して推奨される最小サイズとローテーション構成を確認してください。

Auto Deploy を使用して ESXi 8.0 ホストをインストールする場合、または VMFS ボリューム上のスクラッチ ディレクトリ内のデフォルトの場所とは別にログ ディレクトリを設定する場合、システム ログ用の十分な空き容量を用意するため、現在のログ サイズとローテーション設定の変更が必要になることがあります。vSphere のすべてのコンポーネントではこのインフラストラクチャを使用しています。このインフラストラクチャでのログ容量のデフォルト値は、使用可能なストレージの量、およびシステム ログの構成内容に応じて変わります。Auto Deploy を使用してデプロイしたホストでは、ログが RAM ディスクに保存されます。つまり、ログに使用できる容量はわずかで

ホストが Auto Deploy を使用してデプロイされている場合は、次のいずれかの方法でログ ストレージを再構成してください。

- ネットワーク経由でリモート コレクタにログをリダイレクトする。
- NAS ストアまたは NFS ストアにログをリダイレクトする。

NAS ストアや NFS ストアなどの非デフォルト ストレージにログをリダイレクトする場合は、ディスクにインストールされたホストのログ サイズとローテーションも再構成できます。

デフォルト構成を使用する ESXi ホストのログ ストレージを再構成する必要はありません。このストレージは、VMFS ポリ्यूムのスクラッチ ディレクトリにログを格納します。これらのホストについては、ESXi 8.0 によって、インストールに最も適した方法でログが構成され、ログ メッセージの保存に十分な空き容量が用意されます。

表 4-6. hostd、vpxa、fdm の各ログに対して推奨される最小サイズとローテーションの構成

ログ	ログ ファイルの最大サイズ	保存するログ ファイルの数	必要な最小ディスク容量
管理エージェント (hostd)	10 MB	10	100 MB
vCenter Server エージェント (vpxa)	5 MB	10	50 MB
vSphere HA エージェント (フォールト ドメイン マネージャ (fdm))	5 MB	10	50 MB

リモート ログ サーバの設定方法の詳細については、[ESXi ホストでの Syslog の構成](#)を参照してください。

オプションで、ログの集約と分析を実行する VMware vCenter Log Insight をインストールできます。

VMware Host Client のシステム要件

使用するブラウザが VMware Host Client をサポートしていることを確認します。

VMware Host Client では、次のゲスト OS および Web ブラウザ バージョンがサポートされます。

サポート対象ブラウザ	Mac OS	Windows 32 ビットおよび 64 ビット	
		Linux	
Google Chrome	89+	89+	75 以降
Mozilla Firefox	80+	80+	60 以降
Microsoft Edge	90+	90+	該当なし
Safari	9.0+	該当なし	該当なし

ESXi のパスワードとアカウントのロックアウト

ESXi ホストに対して、事前に定義された要件を満たすパスワードを使用する必要があります。

Security.PasswordQualityControl の詳細システム設定を使用して、パスワードの文字数や文字の種類の要件

の変更や、パスフレーズの許可ができます。Security.PasswordHistory の詳細システム設定を使用して、ユーザーごとに記憶するパスワードの数を設定することもできます。

注： ESXi パスワードのデフォルト要件は、リリースごとに変更される場合があります。

Security.PasswordQualityControl の詳細システム設定を使用して、デフォルトのパスワード制限を確認および変更できます。

ESXi のパスワード

ESXi では、ダイレクト コンソール ユーザー インターフェイス、ESXi Shell、SSH、または VMware Host Client を使用してアクセスするためのパスワード要件があります。

- パスワードを作成する際、デフォルトでは、小文字、大文字、数字、および特殊文字（アンダースコアやダッシュなど）の 4 種類の文字のうち 3 つ以上を混在させる必要があります。
- デフォルトでは、パスワードの長さは 7 文字以上 40 文字未満です。
- パスワードに、辞書ファイル内の単語または単語の一部を含めることはできません。
- パスワードには、ユーザー名またはユーザー名の一部を含めることはできません。

注： パスワードの先頭に大文字を使用する場合、これは文字の種類に含まれません。パスワードの末尾を数字にする場合、これは文字の種類に含まれません。辞書にある語をパスワードに使用すると、パスワード全体の強度が低下します。

ESXi のパスワードの例

次のようにオプション設定の場合のパスワードの候補です

```
retry=3 min=disabled,disabled,disabled,7,7
```

この設定では、新しいパスワードが十分に強力ではない場合、またはパスワードが 2 回正しく入力されなかった場合、ユーザーは最大 3 回 (retry=3) 入力を要求されます。1 種類または 2 種類の文字が含まれるパスワードと、パスフレーズは許可されません。これは、最初の 3 つのアイテムが無効に設定されているためです。パスワードには 3 種類および 4 種類の文字を使用し、7 文字の長さが必要です。max、passphrase など、その他のオプションの詳細については、pam_passwdqc のメイン ページを参照してください。

この設定では、次のパスワードが許可されます。

- xQaTEhb! : 3 種類の文字を使用した 8 文字のパスワード。
- xQaT3#A : 4 種類の文字を使用した 7 文字のパスワード。

次のパスワード候補は、要件を満たしていません。

- Xqat3hi : 先頭が大文字であるため、有効な文字クラスの数 が 2 に減っています。パスワードには、3 種類以上の文字を使用する必要があります。
- xQaTEh2 : 数字で終わるため、有効な文字クラスの数 が 2 に減っています。パスワードには、3 種類以上の文字を使用する必要があります。

ESXi のパスフレーズ

パスワードの代わりに、パスフレーズを使用することもできます。ただし、パスフレーズはデフォルトで無効になっています。デフォルト設定やその他の設定を変更するには、vSphere Client から `Security.PasswordQualityControl` の詳細システム設定を使用します。

たとえば、このオプションは次のように変更できます。

```
retry=3 min=disabled,disabled,16,7,7
```

この例では、最小で 16 文字を使用し、最小で 3 つの単語を含むパスフレーズを許可しています。

デフォルトのパスワード制限の変更

パスワードまたはパスフレーズのデフォルトの制限を変更するには、ESXi ホストの `Security.PasswordQualityControl` 詳細システム設定を使用します。ESXi の詳細システム設定の変更については、『vCenter Server およびホストの管理』ドキュメントを参照してください。

たとえば、最小 15 文字、最小で 4 つの単語数 (`passphrase=4`) を要求するように変更するには、次のように指定します。

```
retry=3 min=disabled,disabled,15,7,7 passphrase=4
```

詳細については、`pam_passwdqc` の man ページを参照してください。

注： パスワードのオプションは、可能なすべての組み合わせがテストされているわけではありません。デフォルトのパスワード設定を変更した後は、テストを実行します。

この例では、パスワードの複雑性の要件で、大きなパスワードの違い、5 つのパスワードの記憶履歴、および 90 日間のローテーション ポリシーを実施する 4 種類の文字から 8 文字が要求されるように設定します。

```
min=disabled,disabled,disabled,disabled,8 similar=deny
```

ESXi のアカウント ロックアウトの動作

SSH 経由および vSphere Web Services SDK 経由のアクセスで、アカウントのロックがサポートされるようになりました。ダイレクト コンソール インターフェイス (DCUI) と ESXi Shell では、アカウント ロックアウトはサポートされていません。デフォルトでは、アカウントがロックされるまでに、ログイン試行の失敗が最大で 5 回許容されています。デフォルトでは 15 分後に、アカウントのロックが解除されます。

ログイン動作の設定

ESXi ホストのログイン動作を構成するには、次の詳細システム設定を使用します。

- `Security.AccountLockFailures`。ログインが失敗し、ユーザー アカウントがロックされるまでの最大試行回数です。ゼロにすると、アカウントのロックは無効になります。
- `Security.AccountUnlockTime`。ユーザーがロックアウトされる秒数です。
- `Security.PasswordHistory`。ユーザーごとに記憶するパスワードの数。vSphere 8.0 Update 1 以降では、デフォルトは 5 です。ゼロを指定すると、パスワード履歴は無効になります。

ESXi 詳細オプションの設定の詳細については、『vCenter Server およびホストの管理』ドキュメントを参照してください。

ESXi のインストールの準備

ESXi をインストールする前に、環境に適したインストール オプションを特定し、インストール プロセスの準備をします。

ESXi インストーラのダウンロード

ESXi インストーラ ソフトウェアは、OEM または Broadcom サポート ポータルから入手できます。

Broadcom サポート ポータルにご登録ください。詳細については、「[Register for an account on the Broadcom Support Portal and Communities](#)」を参照してください。

製品のダウンロード手順については、「[Download Broadcom products and software](#)」を参照してください。

ESXi のパッチおよび更新用のオフライン バンドル ZIP ファイルのダウンロードについては、「[Downloading Broadcom PTF files and solutions](#)」を参照してください。

詳細については、「[VMware to Broadcom Support Frequently Asked Questions](#)」を参照してください。

ESXi のインストールに必要な情報

対話形式のインストールでは、必要な情報の入力を求めるメッセージが表示されますが、インストール スクリプトでこの情報を入力する必要があります。

対話形式のインストールでは、必要な情報の入力を求めるメッセージがシステムに表示されます。スクリプトによるインストールでは、この情報をインストール スクリプト内に指定する必要があります。今後の使用に備えて、インストール中に使用した値を書き留めてください。ESXi を再インストールし、最初に選択した値を再入力する必要が生じた場合にこのメモが役立ちます。

表 4-7. ESXi のインストールに必要な情報

詳細情報	必須またはオプション	デフォルト	コメント
キーボード レイアウト	必須	米国英語	
VLAN ID	オプション	なし	範囲： 0~4094
IP アドレス	オプション	DHCP	インストール中に DHCP でネットワークを構成できません。インストール後にネットワーク設定を変更できます。
サブネット マスク	オプション	IP アドレスに基づいて計算	
ゲートウェイ	オプション	設定されている IP アドレスとサブネット マスクに基づく	
プライマリ DNS	オプション	設定されている IP アドレスとサブネット マスクに基づく	
セカンダリ DNS	オプション	なし	
ホスト名	固定 IP アドレス 設定の場合は必須	なし	vSphere Client はホスト名または IP アドレスのいずれかを使用して ESXi ホストにアクセスします。

表 4-7. ESXi のインストールに必要な情報 (続き)

詳細情報	必須またはオプション	デフォルト	コメント
インストール場所	必須	なし	コンポーネントを 1 つのディスクにインストールする場合は、5 GB 以上が必要です。
ESXi の既存の設定を移行します。既存の VMFS データストアを維持します。	ESXi がすでにインストールされているドライブ上で、ESXi をインストールする場合に必要です。	なし	すでに、ESXi 5.x をインストールしていると、ESXi インストーラでは、インストール中に VMFS データストアを保持するか上書きするかを選択できます。
root パスワード	必須	なし	root パスワードは、8~40 文字である必要があります。パスワードの詳細については、『vSphere セキュリティ』ドキュメントを参照してください。

ESXi インストーラを起動するためのメディア オプション

ESXi インストーラが、ESXi をインストールするシステムにアクセスできる必要があります。

次の起動メディアが ESXi インストーラ用にサポートされています。

- CD/DVD からの起動。ESXi インストーラ ISO イメージのダウンロードおよび CD or DVD への書き込みを参照してください。
- USB フラッシュドライブからの起動。USB フラッシュドライブのフォーマットによる ESXi インストールまたはアップグレードの起動を参照してください。
- ネットワークからの起動。ESXi インストーラのネットワーク ブートを参照してください。
- リモート管理アプリケーションを使用した、リモートの場所からの起動。リモート管理アプリケーションの使用を参照してください。

ESXi インストーラ ISO イメージのダウンロードおよび CD or DVD への書き込み

ESXi のインストール CD/DVD がない場合は作成できます。

カスタム インストール スクリプトが含まれているインストーラ ISO イメージを作成することもできます。カスタムのインストールまたはアップグレードスクリプトを含む、インストーラ ISO イメージの作成を参照してください。

手順

- 1 ESXi インストーラのダウンロードの手順に従います。
- 2 ISO イメージを CD または DVD に書き込みます。

USB フラッシュドライブのフォーマットによる ESXi インストールまたはアップグレードの起動

USB フラッシュドライブをフォーマットして、ESXi インストールまたはアップグレードを起動できます。

この手順の説明では、USB フラッシュ ドライブが `/dev/sdb` として検出されることを前提としています。

注： インストール スクリプトが含まれている `ks.cfg` ファイルを、インストールまたはアップグレードの起動に使用している同じ USB フラッシュ ドライブ上に置くことはできません。キックスタート ファイルは、BIOS または UEFI ブートに依存しません。

前提条件

- Linux マシンとこれに対するスーパーユーザー アクセス
- Linux マシンが検出することのできる USB フラッシュ ドライブ
- ESXi ISO イメージ `VMware-VMvisor-Installer-version_number-build_number.x86_64.iso` (`isolinux.cfg` ファイルを含む)

手順

- 1 Linux を起動し、ログインし、`su` または `sudo root` コマンドを使用してスーパー ユーザー モードを開始します。
- 2 USB フラッシュ ドライブが `/dev/sdb` として検出されない場合、または USB フラッシュ ドライブの検出方法が不明な場合は、検出方法を指定してください。
 - a USB フラッシュ ドライブを接続します。
 - b コマンドラインで、現在のログ メッセージを表示するコマンドを実行します。

```
tail -f /var/log/messages
```

次のメッセージに似た形式で、USB フラッシュ ドライブを特定するいくつかのメッセージが表示されます。

```
Oct 25 13:25:23 ubuntu kernel: [ 712.447080] sd 3:0:0:0: [sdb] Attached SCSI removable disk
```

この例では、`sdb` によって USB ドライブが識別されます。デバイスが異なるものとして識別された場合は、`sdb` の代わりにその識別を使用します。

- 3 USB ドライブ全体を ISO イメージで上書きします。これにより、USB ドライブのパーティション テーブルと前のコンテンツが上書きされます。

```
dd bs=10M if=VMware-VMvisor-Installer-version_number-build_number.x86_64.iso
of=/dev/sdb
```

- 4 USB ドライブを取り出します。

```
eject /dev/sdb
```

結果

USB フラッシュ ドライブを使用して、ESXi インストーラを起動できます。

USB フラッシュ ドライブを作成して、ESXi インストール スクリプトまたはアップグレード スクリプトを保存する

ESXi のスクリプトによるインストールまたはアップグレードに使用する ESXi のインストール スクリプトまたはアップグレード スクリプトを、USB フラッシュ ドライブに格納することができます。

インストール マシンに複数の USB フラッシュ ドライブがある場合、インストール ソフトウェアは、接続されたすべての USB フラッシュ ドライブでインストールまたはアップグレード スクリプトを検索します。

この手順の説明では、USB フラッシュ ドライブが `/dev/sdb` として検出されることを前提としています。

注： インストールまたはアップグレード スクリプトを含む `ks` ファイルは、インストールまたはアップグレードの起動に使用している同じ USB フラッシュ ドライブ上に格納しないでください。

前提条件

- Linux マシン
- ESXi のインストールまたはアップグレード スクリプト、`ks.cfg` キックスタート ファイル
- USB フラッシュ ドライブ

手順

- 1 インストール スクリプトまたはアップグレード スクリプトにアクセスできる Linux マシンに、USB フラッシュ ドライブを接続します。
- 2 パーティション テーブルを作成します。

```
/sbin/fdisk /dev/sdb
```

- a `d` を入力して、すべてのパーティションが削除されるまで、パーティションを削除します。
- b `n` を入力して、ディスク全体を占めるプライマリ パーティション 1 を作成します。
- c `t` を入力して、FAT32 ファイル システムに適した設定にタイプを設定します (`c` など)。
- d `p` と入力して、パーティション テーブルを出力します。

結果は次のようになります。

```
Disk /dev/sdb: 2004 MB, 2004877312 bytes
255 heads, 63 sectors/track, 243 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            1           243       1951866   c   W95 FAT32 (LBA)
```

- e `w` と入力してパーティション テーブルを書き込み、終了します。

- 3 USB フラッシュ ドライブを FAT32 ファイル システムを指定してフォーマットします。

```
/sbin/mkfs.vfat -F 32 -n USB /dev/sdb1
```

- 4 ターゲット ディレクトリを作成し、USB フラッシュ ドライブをマウントします。

```
mkdir -p /usbdisk
mount /dev/sdb1 /usbdisk
```

- ESXi のインストール スクリプトを USB フラッシュ ドライブにコピーします。

```
cp ks.cfg /usbdisk
```

- USB フラッシュ ドライブをアンマウントします。

```
umount /usbdisk
```

結果

USB フラッシュ ドライブには、ESXi のインストール スクリプトまたはアップグレード スクリプトが格納されています。

次のステップ

ESXi インストーラの起動時に、インストール スクリプトまたはアップグレード スクリプトの場所としてこの USB フラッシュ ドライブの場所を参照します。 [インストール スクリプトまたはアップグレード スクリプトを実行するための起動オプションの入力](#)および [PXELINUX 構成ファイル](#)を参照してください。

カスタムのインストールまたはアップグレードスクリプトを含む、インストーラ ISO イメージの作成

独自のインストールまたはアップグレード スクリプトを使用して、標準の ESXi インストーラ ISO イメージをカスタマイズできます。このカスタマイズを行うと、カスタマイズしたインストーラ ISO イメージを起動するときに、スクリプトを使用して無人インストールまたは無人アップグレードを実行できるようになります。

[スクリプトを使用した ESXi のインストール](#) および [boot.cfg ファイルについて](#) も参照してください。

前提条件

- Linux マシン
- ESXi ISO イメージ VMware-VMvisor-Installer-x.x.x-XXXXXX.x86_64.iso。ここで、x.x.x はインストールする ESXi のバージョンであり、XXXXXX はインストーラ ISO イメージのビルド番号です
- カスタムのインストールまたはアップグレード スクリプト、KS_CUST.CFG キックスタート ファイル

手順

- Broadcom サポート ポータルから ESXi ISO イメージをダウンロードします。
- ISO イメージをフォルダにマウントします。

```
mount -o loop VMware-VMvisor-Installer-x.x.x-XXXXXX.x86_64.iso /
esxi_cdrom_mount
```

XXXXXX には、インストールまたはアップグレード後のバージョンの ESXi ビルド番号を指定します。

- esxi_cdrom の内容を別のフォルダにコピーします。

```
cp -r /esxi_cdrom_mount/* /esxi_cdrom
```

- キックスタート ファイルを /esxi_cdrom にコピーします。

```
cp KS_CUST.CFG /esxi_cdrom
```

- 5 /esxi_cdrom/efi/boot/boot.cfg (UEFI ブートの場合) と /esxi_cdrom/boot.cfg (レガシー BIOS ブートの場合) の両方の boot.cfg ファイルを変更し、kernelopt オプションを使用してインストールまたはアップグレード スクリプトの場所を指定します。

スクリプトのパスを入力するには、大文字を使用する必要があります。例：

```
kernelopt=runweasel ks=cdrom:/KS_CUST.CFG
```

インストールまたはアップグレードは、インストール中またはアップグレード中にキックスタート ファイルを指定する必要なく、完全に自動化されています。

- 6 mkisofs または genisoimage コマンドを使用して、ISO イメージを再作成します。

コマンド	構文
mkisofs	mkisofs -relaxed-filenames -J -R -o custom_esxi.iso -b ISOLINUX.BIN -c BOOT.CAT -no-emul-boot -boot-load-size 4 -boot-info-table -eltorito-alt-boot -eltorito-platform efi -b EFIBOOT.IMG -no-emul-boot /esxi_cdrom
genisoimage	genisoimage -relaxed-filenames -J -R -o custom_esxi.iso -b ISOLINUX.BIN -c BOOT.CAT -no-emul-boot -boot-load-size 4 -boot-info-table -eltorito-alt-boot -e EFIBOOT.IMG -no-emul-boot /esxi_cdrom

この ISO インストーラのイメージは、通常の起動または UEFI セキュア ブートに使用できます。ただし、vSphere Lifecycle Manager はこのような ISO イメージのチェックサムを検証できないため、vSphere Lifecycle Manager ワークフローを使用したアップグレードに使用することはできません。

結果

ISO イメージにカスタムのインストールまたはアップグレード スクリプトが含まれます。

次のステップ

この ISO イメージを使用して ESXi をインストールします。

vSphere ESXi Image Builder を使用したインストールのカスタマイズ

VMware vSphere[®] ESXi[™] Image Builder CLI を使用して、カスタマイズされた一連のアップデート、パッチ、およびドライバを含んだ ESXi インストール イメージを作成できます。

vSphere ESXi Image Builder を vSphere Client または PowerCLI と共に使用して、ESXi アップデートとパッチのカスタム セットを含んだ ESXi インストール イメージを作成できます。また、vSphere の各リリースの間にリリースされた、サードパーティのネットワーク ドライバやストレージ ドライバも含めることができます。

vSphere ESXi Image Builder を使用して作成された ESXi イメージは、次のいずれかの方法でデプロイできます。

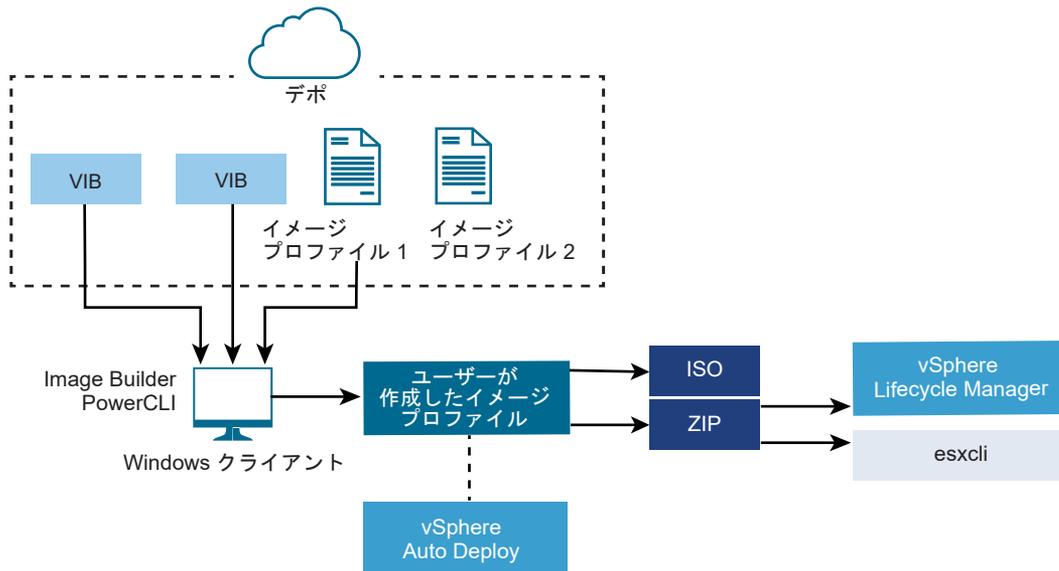
- インストール DVD への書き込み。
- vCenter Server を介して、Auto Deploy 機能を使用する。

vSphere ESXi Image Builder の仕組み

vSphere Auto Deploy で使用 ESXi イメージ プロファイルを作成し、カスタムのサードパーティ製ドライバをイメージ プロファイルに追加してエクスポートするか、アップグレードを実行します。

vSphere ESXi Image Builder を使用すると、vSphere Auto Deploy で使用する ESXi イメージ プロファイルを作成し、既存のイメージ プロファイルにカスタムのサードパーティ製ドライバを追加し、ISO またはバンドルにエクスポートしてアップグレードを実行できます。VMware vSphere ソフトウェアを作成、パッケージ化、配布する方法に関連する基本概念については、[ホストとクラスタのライフサイクルの管理ドキュメントの vSphere Lifecycle Manager](#) が使用できるソフトウェア パッケージ ユニットの参照してください。

図 4-2. Image Builder のアーキテクチャ



vSphere ESXi Image Builder コマンドレットは、さまざまな状況で ESXi ホストにソフトウェアをデプロイするために使用します。

表 4-8. vSphere ESXi Image Builder を使用できるケース

vSphere ESXi Image Builder の使用事例	説明
vSphere Auto Deploy が使用するイメージ プロファイルを作成する	vSphere ESXi Image Builder を使用して、vSphere Auto Deploy がホストのプロビジョニングに使用する VIB を定義する、イメージ プロファイルを作成します。
カスタムのサードパーティ製ドライバを既存のイメージ プロファイルに追加し、ISO またはバンドルにエクスポートする	サードパーティ製ドライバまたはカスタムのエクステンション VIB を ESXi ホストに追加する場合、vSphere ESXi Image Builder を使用して、VMware が提供する基本イメージのクローンを作成し、カスタム VIB を追加して、ISO またはオフライン バンドル ZIP ファイルにエクスポートします。
アップグレードを実行する	カスタム拡張機能またはドライバを含むシステムをアップグレードする場合は、vSphere ESXi Image Builder を使用して、カスタム拡張機能用の vSphere 8.0 互換 VIB を含むカスタム イメージ プロファイルを作成できます。そのカスタム イメージ プロファイルを ISO または ZIP にエクスポートし、vSphere Lifecycle Manager ベースラインを使用してシステムをアップグレードします。

vSphere ESXi Image Builder コマンドレットは、イメージ プロファイルおよび VIB を入力として受け取り、さまざまな出力を生成します。

表 4-9. vSphere ESXi Image Builder コマンドレットの入力および出力

パラメータ	説明
入力	ソフトウェア デボ内に配置されているイメージ プロファイルおよび VIB が、Windows クライアント上で実行中の PowerCLI コマンドレットへの入力として使用されます。
出力	PowerCLI コマンドレット では、ISO イメージまたはオフライン デボの ZIP ファイルにエクスポートできる、カスタム イメージ プロファイルを作成できます。ISO イメージはインストールに使用されます。ZIP のデボは、イメージをアップデートまたはインストールするために、vSphere Lifecycle Manager または <code>esxcli software</code> コマンドで使用できます。イメージ プロファイルは、ESXi ホストのプロビジョニングに使用するソフトウェアをカスタマイズするために、vSphere Auto Deploy のルールで使用されます。

vSphere ESXi Image Builder の詳細については、ビデオ『Image Builder CLI の使用』をご覧ください。



(Image Builder CLI の使用)

イメージ プロファイル

イメージ プロファイルは、ESXi のインストール プロセスまたは更新プロセスで使用される一連の VIB を定義します。イメージ プロファイルは、vSphere Auto Deploy でプロビジョニングされた ESXi ホストに適用されます。イメージ プロファイルの定義および操作は、vSphere ESXi Image Builder を使用して行います。

イメージ プロファイルの要件

カスタムのイメージ プロファイルを新しく作成するか、既存のプロファイルのクローンを作成して、VIB を追加または削除することができます。プロファイルを有効にするため、次の要件を満たす必要があります。

- 各イメージ プロファイルの名前とベンダーの組み合わせは一意である必要があります。
- 各イメージ プロファイルには、許容レベルがあります。vSphere ESXi Image Builder cmdlet を使用してイメージ プロファイルに VIB を追加するときに、Image Builder は、プロファイルに定義されている許容レベルに VIB が一致することを確認します。
- ほかの VIB で必要とされる VIB は削除できません。
- 1つのイメージ プロファイルに、同じ VIB の 2つのバージョンを含めることはできません。VIB の新しいバージョンを追加すると、既存のバージョンの VIB は新しいバージョンで置き換えられます。

イメージ プロファイルの検証

イメージ プロファイルとその VIB を有効にするため、いくつかの条件を満たす必要があります。

- イメージ プロファイルには、少なくとも 1つの基本 VIB と、1つの起動可能なカーネル モジュールを含める必要があります。
- イメージ プロファイル内の VIB にほかの VIB に依存するものがある場合、依存関係がある VIB もイメージ プロファイルに含める必要があります。VIB の作成者は、この情報を SoftwarePackage オブジェクトの Depends プロパティに格納します。
- VIB は互いに競合しないようにします。VIB の作成者は、競合の情報を SoftwarePackage オブジェクトの Conflicts プロパティに格納します。
- 名前が同じでバージョンが異なる 2つの VIB は共存できません。VIB の新しいバージョンを追加すると、既存のバージョンの VIB は新しいバージョンで置き換えられます。
- 許容レベルの検証に関する問題がないことが必要です。

イメージ プロファイルに変更を加える際、vSphere ESXi Image Builder は、変更によってプロファイルが無効にならないことを確認します。

依存関係の検証

VIB を追加または削除する際、vSphere ESXi Image Builder は、パッケージの依存関係が合っていることを確認します。各 SoftwarePackage オブジェクトには、VIB が依存するほかの VIB のリストを指定する Depends プロパティが含まれます。ImageProfile、SoftwarePackage、および ImageProfileDiff の各オブジェクトの構造を参照してください。

許容レベルの検証

vSphere ESXi Image Builder は、イメージ プロファイルが作成または変更されるたびに、許容レベルの検証を行います。vSphere ESXi Image Builder は、プロファイルに許可される最低の許容レベルを基準にして、イメージ プロファイルに含まれる VIB の許容レベルを確認します。VIB の署名が検証されるたびに、VIB の許容レベルも検証されます。

エクスポート中の VIB の検証

イメージ プロファイルを ISO にエクスポートする際、vSphere ESXi Image Builder は次の操作を実行して各 VIB を検証します。

- 各 SoftwarePackage オブジェクトの Conflicts プロパティを確認して、競合がないことを確認します。
- VIB の署名の検証を実行します。署名の検証により、VIB パッケージの不正な変更を防ぎます。署名は、VIB がその著者によって生成されたことを保証する暗号化チェックサムです。署名の検証は、ESXi ホストへの VIB のインストール中や、vSphere Auto Deploy サーバが VIB を使用する際にも行われます。
- VIB がファイル パスの使用ルールに従っていることを確認します。VMware は、VMwareCertified および VMwareAccepted の VIB をテストし、これらの VIB が常にファイル パスの使用ルールに従っていることを保証します。

許容レベルの操作

ホスト、イメージ プロファイル、および個々の VIB には許容レベルがあります。VIB 許容レベルは、VIB がどのようにテストされたのかを表します。各許容レベルの意味、レベルの変更方法、および変更の意味を理解することは、インストールおよび更新作業の重要な手順の 1 つです。

許容レベルは、ホスト、イメージ プロファイル、および個々の VIB に設定されます。ESXi イメージまたはイメージ プロファイルのデフォルトの許容レベルは、PartnerSupported です。

ホスト許容レベル

ホスト許容レベルで、どの VIB をホストにインストールできるかが決まります。ホストの許容レベルは、ESXCLI コマンドを使用して変更できます。デフォルトでは、ESXi ホストには PartnerSupported の許容レベルが設定されていて、PartnerSupported VIB で簡単に更新できるようになっています。

注： VMware は、PartnerSupported 許容レベルのホストをサポートしています。PartnerSupported 許容レベルが設定されている個々の VIB の問題については、パートナーのサポート組織にお問い合わせください。

イメージ プロファイル許容レベル

イメージ プロファイル許容レベルは、イメージ プロファイルで最も低い VIB 許容レベルに設定されます。低い許容レベルの VIB をイメージ プロファイルに追加する場合、Set-ESXImageProfilecmdlet で、イメージ プロファイル許容レベルを変更できます。[イメージ プロファイル許容レベルの設定](#)を参照してください。

vSphere Lifecycle Manager には、実際の許容レベルは表示されません。vSphere ESXi Image Builder cmdlet を使用して、VIB およびイメージ プロファイルの許容レベル情報を取得します。

VIB 許容レベル

VIB の許容レベルは、VIB の作成時に設定されます。VIB の作成者のみが許容レベルを設定できます。

ホストよりも許容レベルが低いイメージ プロファイルまたは VIB でホストをプロビジョニングしようとする、エラーが発生します。ホストの許容レベルを変更して、イメージ プロファイルまたは VIB をインストールします。[ホストの許容レベルの変更](#)を参照してください。ホストの許容レベルを変更すると、そのホストのサポート レベルが変更されます。

ホスト、イメージ プロファイル、または VIB の許容レベルにより、VIB をテストしたユーザーや VIB をサポートするユーザーを判断できます。VMware は、次の許容レベルをサポートしています。

VMwareCertified

VMwareCertified 許容レベルは、最も厳しい要件です。このレベルの VIB では、同じテクノロジーに対して VMware 内部で行われる品質保証テストと完全に同等の詳細なテストが行われます。現在このレベルでは、I/O Vendor Program (IOVP) プログラム ドライバのみが公開されています。この許容レベルの場合は、VMware が VIB に対するサポート コールを受けます。

VMwareAccepted

この許容レベルの VIB では検証テストが行われますが、このテストはソフトウェアのすべての機能を完全にテストするものではありません。テストはパートナーが実行し、VMware がテスト結果を確認します。現在このレベルで公開されている VIB には、CIM プロバイダや PSA プラグインがあります。VMware では、ユーザーがこの許容レベルの VIB に関してサポート コールを行った場合、パートナーのサポート組織に問い合わせるように案内しています。

PartnerSupported

PartnerSupported 許容レベルの VIB は、VMware が信頼するパートナーによって公開されます。そのパートナーがすべてのテストを実行します。VMware はテスト結果を確認しません。このレベルは、パートナーが VMware システム用に採用する、新しいテクノロジー、または主要ではないテクノロジーに使用されます。現在このレベルでは、標準以外のハードウェア ドライバを使用する、Infiniband、ATAoE、SSD などのドライバ VIB テクノロジーが公開されています。VMware では、ユーザーがこの許容レベルの VIB に関してサポート コールを行った場合、パートナーのサポート組織に問い合わせるように案内しています。

CommunitySupported

CommunitySupported 許容レベルは、VMware パートナー プログラムに参加していない個人または企業が作成した VIB に使用されます。このレベルの VIB に対しては VMware が承認したテスト プログラムが実行されておらず、VMware のテクニカル サポートや VMware パートナーによるサポートを受けられません。

ホストの許容レベルの変更

ホストの許容レベルを、インストールする VIB またはイメージ プロファイルの許容レベルと一致するように下げることができます。

ホスト上の各 VIB の許容レベルが、ホストの許容レベル以上である必要があります。たとえば、PartnerSupported 許容レベルの VIB を VMwareAccepted 許容レベルのホストにインストールすることはできません。まず、ホストの許容レベルを下げる必要があります。許容レベルの詳細については、[許容レベルの操作](#)を参照してください。

注意: ホスト許容レベルを CommunitySupported に変更すると、ホストの支援性に影響します。また、ホストのセキュリティに影響する場合があります。

前提条件

ESXCLI をインストールします。ESXCLI スタート ガイドを参照してください。トラブルシューティングするには、ESXi Shell で `esxcli` コマンドを実行します。

手順

- 1 VIB またはイメージ プロファイルの許容レベルを取得します。

オプション	説明
すべての VIB に関する情報の表示	<code>esxcli --server=server_name software sources vib list --depot=depot_URL</code>
指定した VIB に関する情報の表示	<code>esxcli --server=server_name software sources vib list --viburl=vib_URL</code>
すべてのイメージ プロファイルに関する情報の表示	<code>esxcli --server=server_name software sources profile list --depot=depot_URL</code>
指定したイメージ プロファイルに関する情報の表示	<code>esxcli --server=server_name software sources profile get --depot=depot_URL --profile=profile_name</code>

- 2 ホストの許容レベルを表示します。

```
esxcli --server=server_name software acceptance get
```

- 3 ホストの許容レベルを変更します。

```
esxcli --server=server_name software acceptance set --level=acceptance_level
```

acceptance_level の値には、VMwareCertified、VMwareAccepted、PartnerSupported、または CommunitySupported を指定できます。*acceptance_level* の値では、大文字と小文字が区別されます。

注： 追加する VIB またはイメージ プロファイルよりもホストの許容レベルが高い場合、`esxcli software vib` または `esxcli software profile` ネームスペースで、`--force` オプションを使用してコマンドを実行できます。`--force` オプションを使用すると、ホストよりも低い許容レベルの VIB またはイメージ プロファイルが強制されて設定の整合性がなくなるため、警告が表示されます。許容レベルの整合性がないホスト上で VIB のインストール、VIB の削除、およびその他の特定の操作を実行すると、警告が繰り返し表示されます。

イメージ プロファイル許容レベルの設定

VIB をイメージ プロファイルに追加するときに、VIB の許容レベルがイメージ プロファイルの許容レベルよりも低い場合、許容レベルの低いイメージ プロファイルのクローンを作成するか、イメージ プロファイルの許容レベルを変更できます。

VMwareCertified、VMwareAccepted、PartnerSupported、または CommunitySupported をイメージ プロファイルの許容レベルとして指定できます。許容レベルを下げると、その許容レベルでプロビジョニングするイメージ プロファイルとホストのサポート レベルも変更されます。詳細については、[許容レベルの操作](#)を参照してください。

前提条件

PowerCLI およびすべての前提となるソフトウェアをインストールします。vSphere ESXi Image Builder の構成を参照してください。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 イメージ プロファイルの許容レベルを取得します。

```
Get-EsxImageProfile -Name string
```

- 3 イメージ プロファイルの許容レベルを設定します。

```
Set-EsxImageProfile -Name string -AcceptanceLevel level
```

ImageProfile、SoftwarePackage、および ImageProfileDiff の各オブジェクトの構造

ImageProfile、SoftwarePackage、および ImageProfileDiff の各オブジェクトの構造を知ることが、デプロイ プロセスやアップグレード プロセスの管理に役立ちます。

ImageProfile オブジェクトのプロパティ

ImageProfile オブジェクトには、次のプロパティがあります。このオブジェクトには、Get-EsxImageProfile PowerCLI コマンドレットを使用してアクセスできます。

名前	タイプ	説明
AcceptanceLevel	AcceptanceLevel	プロファイルに追加できる VIB を決定します。レベルには、VMwareCertified、VMwareAccepted、PartnerSupported、および CommunitySupported があります。許容レベルの操作を参照してください。
Author	System.String	プロファイルを作成したユーザーを表します。60 文字以下で指定します。
CreationTime	System.DateTime	作成時刻のタイムスタンプを表します。
Description	System.String	プロファイルのフル テキストの説明を表します。長さの制限はありません。
GUID	System.String	グローバルで一意的なイメージ プロファイルの ID を表します。

名前	タイプ	説明
ModifiedTime	System.DateTime	最終更新時刻のタイムスタンプを表します。
Name	System.String	イメージ プロファイルの名前を表します。80 文字以下で指定します。
ReadOnly	System.Boolean	true に設定すると、プロファイルを編集できません。カスタムのイメージ プロファイルを読み取り専用にするには、Set- EsxImageProfile -ReadOnly を使用します。
Rules	ImageProfileRule[]	イメージ プロファイルの OEM ハードウェア要件および制限事項を表します。vSphere Auto Deploy は、イメージ プロファイルをデプロイするときにこのプロパティの値を確認し、一致するハードウェアがある場合は、このプロファイルを実行します。
Vendor	System.String	プロファイルを発行する組織を表します。40 文字以下で指定します。
VibList	SoftwarePackage[]	イメージに含まれる VIB の ID リストを表します。

SoftwarePackage オブジェクトのプロパティ

イメージ プロファイルを準備する際、ソフトウェア パッケージを確認して、どのパッケージを含めるのが適切かを決定できます。SoftwarePackage オブジェクトには、次のプロパティがあります。

名前	タイプ	説明
AcceptanceLevel	AcceptanceLevel	この VIB の許容レベルを表します。
Conflicts	SoftwareConstraint[]	この VIB と同時にインストールできない VIB のリストを表します。各制約では、次の形式が使用されます。 <code>package-name[<< <= > = >> version]</code>
Depends	SoftwareConstraint[]	この VIB と同時にインストールする必要のある VIB のリストを表します。制約の形式は、Conflicts プロパティと同じです。
Description	System.String	VIB の長い説明を表します。
Guid	System.String	VIB の一意の ID を表します。
LiveInstallOk	System.Boolean	この VIB のライブ インストールがサポートされる場合は true です。
LiveRemoveOk	System.Boolean	この VIB のライブの削除がサポートされる場合は true です。
MaintenanceMode	System.Boolean	この VIB をインストールするために、ホストをメンテナンス モードにする必要がある場合は true です。

名前	タイプ	説明
Name	System.String	VIB の名前を表します。通常、実行中の ESXi システムのパッケージを一意に説明します。
Provides	SoftwareProvides	この VIB によって提供される仮想パッケージまたはインターフェイスのリストを表します。 SoftwareProvide オブジェクトのプロパティを参照してください。
ReferenceURLs	SupportReference[]	詳細なサポート情報を含む SupportReference オブジェクトのリストを表します。SupportReference オブジェクトには Title と URL の 2 つのプロパティがあり、どちらもタイプは System.String です。
Replaces	SoftwareConstraint[]	この VIB を置換または廃止する VIB を識別する SoftwareConstraint オブジェクトのリストを表します。VIB は自動的に同じ名前の、バージョンが低い VIB を置換します。
ReleaseDate	System.DateTime	VIB の発行またはリリースの日時を表します。
SourceUrls	System.String[]	この VIB をダウンロードできる、ソース URL のリストを表します。
StatelessReady	System.Boolean	パッケージがホスト プロファイルまたはその他の vSphere Auto Deploy と連携して使用できるようにするための技術をサポートしている場合は True。
Summary	System.String	1 行で表された VIB の概要を表します。
Tags	System.String[]	ベンダーまたは発行元によって定義された、このパッケージ向けの文字列タグの配列を表します。タグは、パッケージの特徴を識別するために使用できます。
Vendor	System.String	VIB のベンダーまたは発行元を表します。
Version	System.String	VIB のバージョンを表します。
VersionObject	Software.Version	VersionObject プロパティのタイプは SoftwareVersion です。 SoftwareVersion クラスは、2 つのバージョンの文字列を比較するために静的な Compare メソッドを実装しています。 SoftwareVersion オブジェクトのプロパティを参照してください。

ImageProfileDiff オブジェクトのプロパティ

Compare-EsxImageProfile cmdlet の実行時に、ユーザーは 2 つのパラメータを渡します（1 つ目は参照プロファイルで、2 つ目は比較プロファイル）。このコマンドレットは、ImageProfileDiff オブジェクトを返します。このオブジェクトには、次のプロパティがあります。

名前	タイプ	説明
CompAcceptanceLevel	System.String	Compare-EsxImageProfile に渡した 2 つ目のプロファイルの許容レベルを表します。
DowngradeFromRef	System.String[]	1 つ目のプロファイルの VIB のダウングレードである、2 つ目のプロファイル内の VIB のリストを表します。
Equal	System.Boolean	2 つのイメージ プロファイルのパッケージと許容レベルが同一の場合は True です。
OnlyInComp	System.String	Compare-EsxImageProfile に渡した 2 つ目のプロファイルにだけ見つかった VIB のリストを表します。
OnlyInRef	System.String[]	Compare-EsxImageProfile に渡した 1 つ目のプロファイルにだけ見つかった VIB のリストを表します。
PackagesEqual	System.Boolean	イメージ プロファイルに、同じ連の VIB パッケージがある場合は True です。
RefAcceptanceLevel	System.String	Compare-EsxImageProfile に渡した 1 つ目のプロファイルの許容レベルを表します。
UpgradeFromRef	System.String[]	1 つ目のプロファイルの VIB のアップグレードである、2 つ目のプロファイル内の VIB のリストを表します。

SoftwareVersion オブジェクトのプロパティ

SoftwareVersion オブジェクトを使用すれば、2 つのバージョンの文字列を比較できます。このオブジェクトには、2 つの文字列を入力として受け入れ、最初のバージョンの文字列が 2 番目のバージョンの文字列より大きい数の場合は 1 を返す Compare の静的なメソッドが含まれています。2 つのバージョンの文字列が等しい場合、Compare は 0 を返します。2 番目のバージョンの文字列が最初の文字列より大きい数の場合、Compare は -1 を返します。このオブジェクトには、次のプロパティがあります。

名前	タイプ	説明
Version	System.String	ハイフンより前のバージョンの部分。この部分はプライマリ バージョンを示しています。
Release	System.String	ハイフンよりあとのバージョンの部分。この部分はリリース バージョンを示しています。

SoftwareConstraint オブジェクトのプロパティ

SoftwareConstraint オブジェクトは、MatchesProvide メソッドを実装しています。このメソッドは、SoftwareProvides オブジェクトまたは SoftwarePackage オブジェクトを入力として受け入れ、制約が SoftwareProvide または SoftwarePackage に一致する場合は True を返し、それ以外の場合は False を返します。

SoftwareConstraint オブジェクトには、次のプロパティがあります。

名前	タイプ	説明
Name	System.String	制約の名前を表します。この名前は対応する SoftwareProvide Name プロパティに一致する必要があります。
Relation	System.String	enum、または <<、<=、>=、>> の比較識別子のいずれか。制約に Relation および Version プロパティがない場合、このプロパティは \$null にできます。
Version	System.String	制約が一致するバージョン。制約に Relation および Version プロパティがない場合、このプロパティは \$null にできます。
VersionObject	SoftwareVersion	SoftwareVersion オブジェクトによって表されるバージョン。

SoftwareProvide オブジェクトのプロパティ

SoftwareProvide オブジェクトには、次のプロパティがあります。

名前	タイプ	説明
Name	System.String	プロバイダの名前を表します。
Version	System.String	プロバイダのバージョンを表します。プロバイダがバージョンを指定していない場合、\$null にできます。
Release	System.String	SoftwareVersion オブジェクトによって表されるプロバイダのバージョンを表します。 SoftwareVersion オブジェクトのプロパティ を参照してください。

vSphere ESXi Image Builder の構成

vSphere ESXi Image Builder コマンドレットを実行する前に、PowerCLI および前提となるすべてのソフトウェアをインストールする必要があります。

前提条件

vSphere ESXi Image Builder コマンドレットを実行する前に、PowerCLI および前提となるすべてのソフトウェアをインストールする必要があります。vSphere ESXi Image Builder スナップインは PowerCLI のインストールに含まれています。PowerCLI コマンドレットを使用して vSphere ESXi Image Builder を管理する場合は、Microsoft .NET Framework 4.5 または 4.5.x および Windows PowerShell 3.0 または 4.0 が Microsoft Windows システムにインストールされていることを確認してください。『PowerCLI ユーザー ガイド』を参照してください。

手順

- 1 ワークステーションで PowerShell を開きます。
- 2 PowerCLI 6.5R1 以降のバージョンの PowerCLI を [PowerCLI のホーム ページ](#) からダウンロードします。

- 3 すべての PowerCLI モジュールをインストールするには、次のコマンドを実行します：Install-Module VMware.PowerCLI -Scope CurrentUser または、モジュール名を使用して Install-Module コマンドレットを実行し、個々の PowerCLI モジュールをインストールすることもできます。信頼されていないリポジトリからモジュールをインストールしていますという警告が表示された場合、[y] を押してから、[Enter] キーを押し、インストールを確認します。

次のコマンドを使用して、PowerCLI モジュールが使用可能であることを確認できます：

```
Get-Module -Name VMware.PowerCLI* -ListAvailable.
```

次のステップ

[VMware.Image Builder コマンドレットの使用](#)を確認します。PowerCLI を初めて使用する場合は、『PowerCLI User's Guide』をお読みください。

vSphere ESXi Image Builder コマンドレットとその他の PowerCLI コマンドレットおよび PowerShell コマンドレットを使用して、イメージ プロファイルおよび VIB を管理します。必要に応じて Get-Help *cmdlet_name* を使用して、コマンドライン ヘルプを表示します。

vSphere ESXi Image Builder の設定

vSphere ESXi Image Builder を vSphere Client と共に使用するには、まず、サービスが有効で、実行されていることを確認する必要があります。

前提条件

- vSphere Auto Deploy リポジトリ用に十分なストレージがあることを確認します。vSphere Auto Deploy サーバは、必要なデータを保存するためにリポジトリを使用します。このデータには、ユーザーが作成するルールおよびルール セットや、ルール内で指定する VIB およびイメージ プロファイルが含まれます。

ベスト プラクティスは、4 つのイメージ プロファイルに十分な容量と、ある程度の余分な空き容量を確保するために、2 GB を割り当てることです。1 つのイメージ プロファイルにつき、約 400 MB が必要です。使用するイメージ プロファイル数を検討することで、vSphere Auto Deploy リポジトリ用にどの程度の容量を予約するかを決定します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。
デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。
- 2 [Auto Deploy] ページで、上部のドロップダウン メニューから vCenter Server を選択します。
- 3 [Image Builder を有効にする] をクリックしてサービスを有効にします。
[ソフトウェア デポ] タブが表示されます。

次のステップ

- [ソフトウェア デポの追加](#).
- [ソフトウェア デポのインポート](#).
- [イメージ プロファイルのクローン作成](#).

- イメージ プロファイルの作成.
- vSphere Auto Deploy 用にシステムを準備.
- [Image Builder サービス] のデフォルト構成プロパティを変更できます。詳細については、『vCenter Server およびホストの管理』で vCenter Server の設定について参照してください。

VMware.Image Builder コマンドレットの使用

VMware.Image Builder コマンドレットを使用すると、すべての PowerCLI 機能を利用できます。

VMware.Image Builder コマンドレットは、Microsoft PowerShell cmdlet として実装され、PowerCLI に含まれます。VMware.Image Builder コマンドレットを使用すると、PowerCLI のすべての機能を利用できます。PowerShell に精通しているユーザーは、ほかの PowerShell コマンドレットと同じように VMware.Image Builder コマンドレットを使用することができます。PowerShell および PowerCLI に慣れていない場合は、次のヒントに従ってください。

コマンドレット、パラメータ、およびパラメータの値は、PowerCLI のシェルに入力できます。

- `Get-Help cmdlet_name` を実行すると、任意のコマンドレットのヘルプを表示することができます。
- PowerShell は大文字と小文字を区別しないことに注意してください。
- コマンドレット名やパラメータ名には、タブ補完を使用します。
- `Format-List` または `Format-Table` (短縮形式は `fl` または `ft`) を使用して、任意の変数およびコマンドレットの出力の形式を整えることができます。`Get-Help Format-List` を参照してください。
- VIB やイメージ プロファイルの検索およびフィルタリングにワイルドカードを使用します。すべてのワイルドカードの式がサポートされています。

名前によるパラメータの指定

パラメータは多くの場合名前で渡すことができ、スペースや特殊文字を含むパラメータの値は二重引用符で囲むことができます。

```
Add-ESXSoftwarePackage -ImageProfile profile42 -SoftwarePackage "partner package 35"
```

オブジェクトによるパラメータの指定

スクリプト処理や自動化を実施する場合は、パラメータをオブジェクトとして渡すことができます。このテクニックは、複数のオブジェクトを返す cmdlet または単一のオブジェクトを返す cmdlet に使用できます。

- 1 複数のオブジェクトを返す cmdlet の出力を変数にバインドします。

```
$profs = Get-ESXImageProfile
```

- 2 そのオブジェクトを入力として必要とする cmdlet を実行する場合は、そのオブジェクトに位置 (0 から始まるリスト) でアクセスします。

```
Add-ESXSoftwarePackage -ImageProfile $profs[4] -SoftwarePackage partner-pkg
```

この例では、指定されたソフトウェア パッケージが、`Get-EsxImageProfile` から返されるリストの 5 番目のイメージ プロファイルに追加されます。

vCenter Server のインストールとセットアップドキュメントのほとんどの例では、パラメータは名前で渡されません。[PowerCLI コマンドレットを使用した vSphere ESXi Image Builder のワークフロー](#)には、パラメータをオブジェクトとして渡す例が含まれています。

VMware.ImageBuilder コマンドレットの概要

VMware PowerCLI の VMware.Image Builder コンポーネントは、ソフトウェア デポの VIB、イメージ プロファイル、およびその他のコンテンツを管理するためのコマンドレットを提供します。

vSphere 7.0 以降では、レガシーの通知やパッチとともに VIB をパッケージ化する新しい方法が導入され、ソフトウェア デポには基本イメージ、ベンダー アドオンとコンポーネント、および VIB とイメージ プロファイルが含まれています。VMware PowerCLI 12.0 以降には、ソフトウェア デポの新しいコンテンツと連携するコマンドレットが提供されています。

VMware.ImageBuilder には、次のコマンドレットが含まれています。

注： VMware.ImageBuilder コマンドレットを実行する場合は、コマンドレットを起動するときにコマンド ラインですべてのパラメータを指定します。対話モードでのパラメータの提供は推奨されません。

詳細な参照情報を確認するには、PowerCLI プロンプトで `Get-Help cmdlet_name` を実行します。

表 4-10. ソフトウェア デポのレガシー コンテンツで使用される VMware.ImageBuilder コマンドレット

コマンドレット	説明
<code>Add-EsxSoftwareDepot</code>	指定された場所にあるソフトウェア デポまたは ZIP ファイルを現在の環境に追加します。デポからメタデータをダウンロードし、VIB の依存関係を分析します。
<code>Remove-EsxSoftwareDepot</code>	指定されたソフトウェア デポとの接続を解除します。
<code>Get-EsxSoftwareDepot</code>	現在の環境に存在するソフトウェア デポのリストを戻します。イメージ プロファイルと VIB を調べて管理する場合は、まず対応するソフトウェア デポを環境に追加する必要があります。
<code>Get-EsxSoftwarePackage</code>	ソフトウェア パッケージ オブジェクト (VIB) のリストを返します。この cmdlet のオプションを使用して、結果をフィルタリングします。
<code>Get-EsxImageProfile</code>	現在追加されているすべてのデポから ImageProfile オブジェクトのアレイを返します。
<code>New-EsxImageProfile</code>	新しいイメージ プロファイルを作成します。多くの場合は、既存のプロファイルのクローンを作成して新しいプロファイルを作成することが推奨されます。 PowerCLI コマンドレットを使用したイメージ プロファイルのクローン作成 を参照してください。
<code>Set-EsxImageProfile</code>	ローカルの ImageProfile オブジェクトを変更して、変更されたプロファイルで検証テストを実行します。この cmdlet は、変更されたオブジェクトを返しますが、維持はしません。
<code>Export-EsxImageProfile</code>	イメージ プロファイルを、ESXi インストール用の ESXi ISO イメージまたは ZIP ファイルとしてエクスポートします。
<code>Compare-EsxImageProfile</code>	2 つのプロファイルの VIB リストと許容レベルが同じかどうかを示す、ImageProfileDiff の構造を返します。 許容レベルの操作 を参照してください。
<code>Remove-EsxImageProfile</code>	ソフトウェア デポからイメージ プロファイルを削除します。

表 4-10. ソフトウェア デポのレガシー コンテンツで使用される VMware.ImageBuilder コマンドレット (続き)

コマンドレット	説明
Add- EsxSoftwarePackage	既存のイメージ プロファイルに 1 つ以上の新しいパッケージ (VIB) を追加します。
Remove- EsxSoftwarePackage	イメージ プロファイルから 1 つ以上のパッケージ (VIB) を削除します。
Set- ESXImageProfileAssoc iation	指定されたイメージ プロファイルを指定された ESXi システムに関連付けます。

表 4-11. ソフトウェア デポの新しいコンテンツで使用される VMware.ImageBuilder コマンドレット

コマンドレット	説明
Get-DepotAddons	ソフトウェア デポのアドオンに関する基本情報を提供するオブジェクトの配列を取得します。
Get-DepotBaseImages	ソフトウェア デポの基本イメージに関する基本情報を提供するオブジェクトの配列を取得します。
Get-DepotComponents	ソフトウェア デポのコンポーネントに関する基本情報を提供するオブジェクトの配列を取得します。
Get-DepotInfo	指定したファイル パスまたは URL アドレスにあるソフトウェア デポに関する基本情報を取得します。
Get-DepotVibs	ソフトウェア デポの VIB に関する基本情報を提供するオブジェクトの配列を取得します。
New-IsoImage	指定したファイル パスにある指定したソフトウェア デポとソフトウェア仕様を使用して ISO イメージを生成します。
New-PxeImage	指定したファイル パスにある指定したソフトウェア デポとソフトウェア仕様を使用して PXE イメージを生成します。

ESXi イメージ プロファイル タスク

VMware PowerCLI の VMware.Image Builder コンポーネントまたは vSphere Client のいずれかを使用して、ソフトウェア デポ、イメージ プロファイル、および VIB を操作します。

ソフトウェア デポの追加

イメージ プロファイルをカスタマイズするには、vSphere ESXi Image Builder インベントリに 1 つ以上のソフトウェア デポを追加します。

前提条件

ソフトウェア デポを使用してイメージ プロファイルをカスタマイズする前に、1 つ以上のソフトウェア デポを vSphere ESXi Image Builder インベントリに追加する必要があります。vSphere Client を使用して、ソフトウェア デポを追加できます。

vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。[vSphere ESXi Image Builder の設定](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っています。

- 2 [ソフトウェア デポ] タブで、[新規] をクリックします。

[ソフトウェア デポの追加] ウィンドウが表示されます。

- 3 作成するデポのタイプを選択します。

オプション	操作
オンライン デポ	<ol style="list-style-type: none"> a インベントリにデポの名前を入力します。 b オンライン デポの URL を入力します。
カスタム デポ	インベントリでデポの名前を入力します。

VMware オンライン ソフトウェア デポは、<https://hostupdate.vmware.com/software/VUM/PRODUCTION/main/vmw-depot-index.xml> にあります。

- 4 [追加] をクリックします。

- 5 (オプション) 選択したデポの内容、およびパッケージの詳細を表示するには、[ソフトウェア パッケージ] タブをクリックします。

- 6 (オプション) [オンライン デポ] を追加してある場合は、以下のことが可能です。

- ◆ [更新の確認] によって最新のデポ パッケージを取得する。
- ◆ [詳細情報] をクリックしてデポの詳細情報を取得する。

結果

ソフトウェア デポがリストに追加されます。

次のステップ

- イメージ プロファイルを新しい vSphere Auto Deploy ルールに関連付けて、ESXi ホストをプロビジョニングすることができます。 [デプロイ ルールの作成またはデプロイ ルールのクローンの作成を参照してください。](#)
- イメージ プロファイルを ESXi ホストに関連付けることができます。 [vSphere Auto Deploy インベントリへのホストの追加を参照してください。](#)
- [ホストのイメージ プロファイルの関連付けの編集。](#)
- カスタム ソフトウェア デポを [削除] します。

ソフトウェア デポのインポート

オフライン デポがローカル ファイル システムに配置されている場合は、ZIP ファイルを vSphere ESXi Image Builder インベントリにインポートします。

前提条件

オフライン デポがローカル ファイル システムに配置されている場合は、vSphere Client を使用して ZIP ファイルを vSphere ESXi Image Builder インベントリにインポートできます。

vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。vSphere ESXi Image Builder の設定を参照してください。

手順

1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っていません。

2 [ソフトウェア デポ] タブで、[インポート] をクリックします。

3 インベントリでソフトウェア デポの名前を入力します。

4 [参照] をクリックし、ローカル システムからインポートするソフトウェア デポを含む ZIP ファイルを選択します。

5 [アップロード] をクリックします。

次のステップ

- イメージ プロファイルを新しい vSphere Auto Deploy ルールに関連付けて、ESXi ホストをプロビジョニングすることができます。デプロイ ルールの作成またはデプロイ ルールのクローンの作成を参照してください。
- イメージ プロファイルを ESXi ホストに関連付けることができます。vSphere Auto Deploy インベントリへのホストの追加を参照してください。
- ホストのイメージ プロファイルの関連付けの編集。

イメージ プロファイルのクローン作成

vSphere Client を使用して、イメージ プロファイルのクローンを作成します。

前提条件

vSphere Client を使用してイメージ プロファイルをクローン作成できます。プロファイルの VIB リストに多少の変更を加える場合や、異なるベンダーの複数のホストに同じ基本プロファイルを使用しながら、ベンダー固有の VIB を追加する場合に、イメージ プロファイルをクローン作成できます。

- vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。vSphere ESXi Image Builder の設定を参照してください。
- ソフトウェア デポを vSphere ESXi Image Builder インベントリに追加またはインポートします。ソフトウェア デポの追加およびソフトウェア デポのインポートを参照してください。
- 少なくとも 1 つのカスタム デポ が vSphere ESXi Image Builder インベントリにあることを確認します。

手順

1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っています。

2 [ソフトウェア デポ] タブで、ドロップダウン メニューを使用して、使用するイメージ プロファイルが含まれているソフトウェア デポを選択します。

3 デポに含まれるイメージ プロファイルのリストからクローン作成するイメージ プロファイルを選択して、[クローン作成] をクリックします。

4 イメージ プロファイル名、ベンダーおよび説明を入力します。

一意のイメージ プロファイル名を入力する必要があります。

5 [ソフトウェア デポ] ドロップダウン メニューから、新しいイメージ プロファイルを追加するカスタム デポを選択します。

6 [次へ] をクリックします。

[ソフトウェア パッケージの選択] ページが表示されます。

7 ドロップダウン メニューから、イメージ プロファイルの許容レベルを選択します。

基本イメージに追加する VIB の許容レベルは、基本イメージの許容レベル以上である必要があります。許容レベルの低い VIB をイメージ プロファイルに追加する場合、イメージ プロファイル許容レベルを低くする必要があります。詳細については、[許容レベルの操作](#)を参照してください。

8 イメージ プロファイルに追加する VIB を選択し、削除するものを選択解除してから、[次へ] をクリックします。

注： イメージ プロファイルには、ブート可能な ESXi イメージが有効な状態で含まれている必要があります。

vSphere ESXi Image Builder が、変更によってイメージ プロファイルが無効にならないことを検証します。VIB には他の VIB に依存するものがありますが、イメージ プロファイルにこれらの VIB を個別に含めると無効になります。VIB を追加または削除する場合、vSphere ESXi Image Builder はパッケージの依存関係が保持されるかどうかを確認します。

9 [設定の確認] ページで、新しいイメージ プロファイルの概要情報を確認して、[終了] をクリックします。

次のステップ

- イメージ プロファイルを新しい vSphere Auto Deploy ルールに関連付けて、ESXi ホストをプロビジョニングすることができます。[デプロイ ルールの作成またはデプロイ ルールのクローンの作成](#)を参照してください。
- イメージ プロファイルを ESXi ホストに関連付けることができます。[vSphere Auto Deploy インベントリへのホストの追加](#)を参照してください。
- [ホストのイメージ プロファイルの関連付けの編集](#)。

PowerCLI コマンドレットを使用したイメージ プロファイルのクローン作成

カスタム イメージ プロファイルを作成する最も簡単な方法は、発行済みのプロファイルのクローンを作成することです。プロファイルのクローン作成は、プロファイルからいくつかの VIB を削除する場合や、異なるベンダーの複

数のホストに同じ基本プロファイルを使用しながら、ベンダー固有の VIB を追加したい場合に特に便利です。VMware パートナーのインストールや、サイズの大きいインストールでは、プロファイルの新規作成を検討する必要があります。

前提条件

- PowerCLI および前提となるすべてのソフトウェアをインストールします。vSphere ESXi Image Builder の構成 を参照してください。
- クローンを作成するイメージ プロファイルを含むソフトウェア デポにアクセスできることを確認します。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 (オプション) Get-EsxImageProfile cmdlet を実行して、クローンを作成するプロファイルの名前を探します。

Get-EsxImageProfile にフィルタリング オプションを使用できます。

- 3 New-EsxImageProfile cmdlet を実行して新しいプロファイルを作成し、-CloneProfile パラメータを使用して、クローンを作成するプロファイルを指定します。

```
New-EsxImageProfile -CloneProfile My_Profile -Name "Test Profile 42"
```

この例では、My_Profile という名前のプロファイルのクローンを作成し、Test Profile 42 という名前を割り当てています。クローンとして作成されたプロファイルには、名前とベンダーの一意の組み合わせを指定する必要があります。

次のステップ

フィルタリングのいくつかの例については、デポの内容の調査を参照してください。

VIB の追加または削除を行って、イメージ プロファイルをカスタマイズします。PowerCLI のコマンドレットを使用したイメージ プロファイルへの VIB の追加を参照してください。

イメージ プロファイルの作成

既存のイメージ プロファイルをクローン作成する代わりに、vSphere Client を使用して新しいイメージ プロファイルを作成できます。

前提条件

インベントリのイメージ プロファイルと著しく異なる場合は、新しいイメージ プロファイルを作成することを検討してください。

- vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。 [vSphere ESXi Image Builder の設定](#)を参照してください。
- ソフトウェア デポを vSphere ESXi Image Builder インベントリに追加またはインポートします。 [ソフトウェア デポの追加およびソフトウェア デポのインポート](#)を参照してください。
- 少なくとも 1 つのカスタム デポ が vSphere ESXi Image Builder インベントリにあることを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っています。

- 2 [ソフトウェア デポ] ドロップダウン メニューから、新しいイメージ プロファイルを追加するカスタム デポを選択します。

- 3 [イメージ プロファイル] タブで、[新しいイメージ プロファイル] をクリックします。

- 4 イメージ プロファイル名、ベンダーおよび説明を入力します。

一意のイメージ プロファイル名を入力する必要があります。

- 5 [次へ] をクリックします。

[ソフトウェア パッケージの選択] ページが表示されます。

- 6 ドロップダウン メニューから、イメージ プロファイルの許容レベルを選択します。

基本イメージに追加する VIB の許容レベルは、基本イメージの許容レベル以上である必要があります。許容レベルの低い VIB をイメージ プロファイルに追加する場合、イメージ プロファイル許容レベルを低くする必要があります。詳細については、[許容レベルの操作](#)を参照してください。

- 7 イメージ プロファイルに追加する VIB を選択し、削除するものを選択解除してから、[次へ] をクリックします。

注： イメージ プロファイルには、ブート可能な ESXi イメージが有効な状態で含まれている必要があります。

vSphere ESXi Image Builder が、変更によってイメージ プロファイルが無効にならないことを検証します。VIB には他の VIB に依存するものがありますが、イメージ プロファイルにこれらの VIB を個別に含めると無効になります。VIB を追加または削除する場合、vSphere ESXi Image Builder はパッケージの依存関係が保持されるかどうかを確認します。

- 8 [設定の確認] ページで、新しいイメージ プロファイルの概要情報を確認して、[終了] をクリックします。

次のステップ

- イメージ プロファイルを新しい vSphere Auto Deploy ルールに関連付けて、ESXi ホストをプロビジョニングすることができます。 [デプロイ ルールの作成](#)または [デプロイ ルールのクローンの作成](#)を参照してください。

- イメージ プロファイルを ESXi ホストに関連付けることができます。vSphere Auto Deploy インベントリへのホストの追加を参照してください。
- ホストのイメージ プロファイルの関連付けの編集。
- イメージ プロファイルの [削除] を選択します。
- 選択したイメージ プロファイルを [ソフトウェア パッケージの表示] で確認します。

PowerCLI コマンドレットを使用したカスタム ESXi ISO イメージの作成

ESXi Image Builder を使用して ESXi イメージ プロファイルをカスタマイズすることはできますが、異なるデポのコンテンツを組み合わせて 1 つの ISO イメージを生成することはできません。VMware PowerCLI 12.0 以降では、複数のソフトウェア デポのコンテンツおよびカスタム ソフトウェア仕様を使用してカスタム ISO イメージを作成できます。

New-IsoImage コマンドレットでは、基本イメージ、アドオン、コンポーネントなど、vSphere Lifecycle Manager が必要とする追加のメタデータが保持されます。この追加のメタデータは、レガシー ESXi Image Builder コマンドレットを使用してエクスポートできる ISO イメージには含まれません。

前提条件

VMware PowerCLI 12.0 以降をインストールします。

使用するソフトウェア仕様を含むソフトウェア デポにアクセスできることを確認します。

手順

- 1 カスタム ISO イメージの作成に使用するソフトウェア仕様に必要な情報を収集します。
 - a `Get-DepotBaseImages` コマンドレットを実行して、必要なパッチまたはアップグレードの基本イメージバージョンを取得します。

```
PS C:\> Get-DepotBaseImages -Depot C:\VMware-ESXi-8.xxx-xxxxxxx-depot.zip
```

コマンド出力は次のとおりです。

```
Version                               Vendor
Release date                          -----
-----
8.0.0-0.0.xxxxxx                       VMware, Inc.
01/01/20xx 00:00:00
```

- b ソフトウェア デポの新しいメタデータで使用されるコマンドレットを使用して、OEM アドオンなどの他のパッケージを取得します。例：

```
PS C:\> Get-DepotAddons -Depot C:\addon-depot.zip
```

コマンド出力は次のとおりです。

Name	Version	ID
Vendor	Release date	

```

----
-----
testaddonv1      1.0.0-1      testaddonv1:1.0.0-1      ESXLifecycle
QE              02/20/20xx 18:28:23

```

Get-DepotComponents コマンドレットを使用して、ソフトウェア デポのすべてのコンポーネントを一覧表示することもできます。

```
PS C:\> Get-DepotComponents -Depot C:\Intel-i40en_1.12.3.0-1OEM.xxxxxxx.zip
```

コマンド出力は次のとおりです。

```

Name                               Version                               Vendor
ID                                  -----
--                                  -----
Intel-i40en                         1.12.3.0-1OEM.xxxxxxx              Intel-
i40en:1.12.3.0-1OEM.xxxxxxx         Intel

```

任意の数とオフラインおよびオンライン ソフトウェア デポの組み合わせを使用できます。

2 ソフトウェア仕様を作成します。例：

```

{
  "base_image": {
    "version": "8.0.0-0.0.xxxxxxx"
  },
  "add_on": {
    "name": "testaddonv1",
    "version": "1.0.0-1"
  },
  "components": {
    "Intel-i40en": "1.12.3.0-1OEM.xxxxxxx"
  }
}

```

ソフトウェア仕様は、ESXi 基本イメージとベンダー アドオンなどの追加パッケージに関する情報が格納されている JSON ファイルです。

3 Depots、SoftwareSpec、および Destination パラメータを指定して New-IsoImage コマンドレットを実行して、カスタム ISO イメージを生成します。例：

```
New-IsoImage -Depots "c:\temp\VMware-ESXi-8.0-xxxxxxx-depot.zip" , "c:\temp\HPE-xxxxxxx-
Jan20xx-Synergy-Addon-depot.zip" -SoftwareSpec "c:\temp\HPE-80xx-custom.JSON" -Destination
"c:\temp\HPE-80xx-custom.iso"
```

デポには、サポートされている ESXi バージョンおよびベンダー アドオンの zip ファイルのパスが含まれています。ターゲットには、カスタム ISO ファイルのパスとファイル名が含まれています。

追加のカーネル オプションの受け渡し、ライブ イメージの作成、既存のファイルの上書き、またはイメージの作成中に使用される個々の VIB の許容レベルの確認を行うことができます。New-IsoImage コマンドレットの詳細については、<https://code.vmware.com/docs/11794/cmdletreference/doc/New-IsoImage.html> を参照してください。

次のステップ

ホストのアップグレード操作に使用するアップグレード ベースラインを作成できるように、新しい ISO イメージを vSphere Lifecycle Manager デポにインポートします。

PowerCLI コマンドレットを使用したカスタム PXE イメージの作成

VMware PowerCLI 12.0 以降では、任意のソフトウェア デポとカスタム ソフトウェア仕様を使用してカスタム PXE イメージを作成できます。

前提条件

VMware PowerCLI 12.0 以降をインストールします。

使用するソフトウェア仕様を含むソフトウェア デポにアクセスできることを確認します。

手順

- 1 カスタム PXE イメージの作成に使用するソフトウェア仕様に必要な情報を収集します。
 - a `Get-DepotBaseImages` コマンドレットを実行して、必要なパッチまたはアップグレードの基本イメージバージョンを取得します。

```
PS C:\> Get-DepotBaseImages -Depot C:\VMware-ESXi-8.xxxx-xxxxx-depot.zip
```

コマンド出力は次のとおりです。

```
Version                               Vendor
Release date                          -----
-----
8.x.x.xxx.xxxxxx                       VMware, Inc.
04/29/20xx 00:00:00
```

- b ソフトウェア デポの新しいメタデータで使用されるコマンドレットを使用して、OEM アドオンなどの他のパッケージを取得します。例：

```
PS C:\> Get-DepotAddons -Depot C:\addon-depot.zip
```

コマンド出力は次のとおりです。

```
Name                               Version          ID
Vendor                             Release date    --
----                             -
-----
testaddonv1                        1.0.0-1        testaddonv1:1.0.0-1  ESXLifecycle
QE                                02/20/20xx 18:28:23
```

`Get-DepotComponents` コマンドレットを使用して、ソフトウェア デポのすべてのコンポーネントを一覧表示することもできます。

```
PS C:\> Get-DepotComponents -Depot C:\Intel-
i40en_1.12.3.0-1OEM.700.1.0.15843807_18058526.zip
```

コマンド出力は次のとおりです。

```

Name                               Version                               Vendor
ID                                  -----                               -----
--                                  -----                               -----
Intel-i40en                         1.12.3.0-1OEM.xxxxxx               Intel-i40en:1.12.3.0-1OEM.xxxxxx
Intel

```

任意の数とオフラインおよびオンライン ソフトウェア デポの組み合わせを使用できます。

2 ソフトウェア仕様を作成します。例：

```

{
  "base_image": {
    "version": "8.0.xxxxxx"
  },
  "add_on": {
    "name": "testaddonv1",
    "version": "1.0.0-1"
  },
  "components": {
    "Intel-i40en": "1.12.3.0-1OEM.xxxxxx"
  }
}

```

ソフトウェア仕様は、ESXi 基本イメージとベンダー アドオンなどの追加パッケージに関する情報が格納されている JSON ファイルです。

3 Depots、SoftwareSpec、および Destination パラメータを指定して New-PxeImage コマンドレットを実行し、カスタム PXE イメージを生成します。例：

```

New-PxeImage -Depots "c:\temp\VMware-ESXi-8.0xxxxxx-xxxxxx-depot.zip" ,
"c:\temp\HPE-8.0xxxxxx-xxx-Synergy-Addon-depot.zip" -SoftwareSpec "c:\temp\HPE-xxx-
custom.JSON" -Destination "C:\pxe-image"

```

デポには、サポートされている ESXi バージョンおよびベンダー アドオンの zip ファイルのパスが含まれています。ターゲットには、カスタム PXE ファイルのパスとファイル名が含まれています。

追加のカーネル オプションの受け渡し、ライブ イメージの作成、既存のファイルの上書き、またはイメージの作成中に使用される個々の VIB の許容レベルの確認を行うことができます。New-PxeImage コマンドレットの詳細については、<https://code.vmware.com/docs/11794/cmdletreference/doc/New-PxeImage.html> を参照してください。

次のステップ

PXE イメージは PXE ブートされた ESXi ホストの修正ワークフローで使用できます。

イメージ プロファイルの編集

vSphere Client を使用してイメージ プロファイルを編集できます。イメージ プロファイルの名前、詳細および VIB リストを変更できます。

前提条件

- vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。vSphere ESXi Image Builder の設定を参照してください。
- ソフトウェア デポを vSphere ESXi Image Builder インベントリに追加またはインポートします。ソフトウェア デポの追加およびソフトウェア デポのインポートを参照してください。
- 少なくとも 1 つのカスタム デポ が vSphere ESXi Image Builder インベントリにあることを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っています。

- 2 [ソフトウェア デポ] タブで、ドロップダウン メニューを使用して、使用するイメージ プロファイルが含まれているソフトウェア デポを選択します。

- 3 [イメージ プロファイル] タブで編集するイメージ プロファイルを選択し、[編集] をクリックします。

[イメージ プロファイルの編集] ウィザードが表示されます。

- 4 (オプション) イメージ プロファイルの名前、ベンダーおよび説明情報を変更します。

- 5 [次へ] をクリックします。

[ソフトウェア パッケージの選択] ページが表示されます。

- 6 ドロップダウン メニューから、イメージ プロファイルの許容レベルを選択します。

基本イメージに追加する VIB の許容レベルは、基本イメージの許容レベル以上である必要があります。許容レベルの低い VIB をイメージ プロファイルに追加する場合、イメージ プロファイル許容レベルを低くする必要があります。詳細については、[許容レベルの操作](#)を参照してください。

- 7 イメージ プロファイルに追加する VIB を選択し、削除するものを選択解除してから、[次へ] をクリックします。

注： イメージ プロファイルには、ブート可能な ESXi イメージが有効な状態で含まれている必要があります。

vSphere ESXi Image Builder が、変更によってイメージ プロファイルが無効にならないことを検証します。VIB には他の VIB に依存するものがありますが、イメージ プロファイルにこれらの VIB を個別に含めると無効になります。VIB を追加または削除する場合、vSphere ESXi Image Builder はパッケージの依存関係が保持されるかどうかを確認します。

- 8 [設定の確認] ページで、編集したイメージ プロファイルの概要を確認して、[終了] をクリックします。

次のステップ

- イメージ プロファイルを新しい vSphere Auto Deploy ルールに関連付けて、ESXi ホストをプロビジョニングすることができます。[デプロイ ルールの作成](#)または[デプロイ ルールのクローンの作成](#)を参照してください。
- イメージ プロファイルを ESXi ホストに関連付けることができます。[vSphere Auto Deploy インベントリへのホストの追加](#)を参照してください。
- [ホストのイメージ プロファイルの関連付けの編集](#)。

PowerCLI のコマンドレットを使用したイメージ プロファイルへの VIB の追加

イメージ プロファイルが読み取り専用で設定されていない場合は、そのイメージ プロファイルに 1 つ以上の VIB を追加できます。新しい VIB がほかの VIB に依存している場合や、同じプロファイル内のその他の VIB と競合している場合には、PowerShell のプロンプトにメッセージが表示され、VIB は追加されません。

VMware または VMware パートナーが提供する VIB をイメージ プロファイルに追加できます。VMware VIB を追加する場合は、vSphere ESXi Image Builder が検証を行います。複数の OEM パートナーの VIB を同時に追加すると、エラーは報告されませんが、結果のイメージ プロファイルが機能しない可能性があります。1 つの OEM ベンダーが提供する VIB のみを一度にインストールします。

許容レベルの問題に関するエラーが表示される場合は、イメージ プロファイルの許容レベルおよびホストの許容レベルを変更します。ホストの許容レベルの変更が的確であるかどうかを十分に検討します。VIB の許容レベルは VIB の作成時に設定され、変更することはできません。

結果のイメージ プロファイルが無効でも、VIB を追加できます。

注： 弊社でサポートできるのは、広範な厳しいテストで安定性と完全な機能性が証明された環境と構成のみです。サポートされている構成のみを使用してください。ホスト許容レベルが低下し、その結果、サポートの対応性も低下した場合はカスタム VIB を使用できます。その場合、変更を追跡すると後でカスタム VIB を削除し、ホスト許容レベルをデフォルト (Partner Supporter) に戻すことができます。 [許容レベルの操作](#) を参照してください。

前提条件

PowerCLI および前提となるすべてのソフトウェアをインストールします。 [vSphere ESXi Image Builder の構成](#) を参照してください。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 Get-EsxImageProfilecmdlet を実行して、現在表示されているすべてのデポ内のすべてのイメージ プロファイルをリスト表示します。

この cmdlet は、すべての使用可能なプロファイルを返します。任意の引数を使用して出力をフィルタリングすると、検索を絞り込むことができます。

- 3 プロファイルのクローンを作成します。

```
New-EsxImageProfile -CloneProfile My_Profile -Name "Test Profile 42" -Vendor "My Vendor"
```

VMware およびそのパートナーから発行されているイメージ プロファイルは、読み取り専用です。変更を加えるには、イメージ プロファイルのクローンを作成する必要があります。vendor パラメータは必須です。

- 4 Add-EsxSoftwarePackagecmdlet を実行し、イメージ プロファイルのいずれかに新しいパッケージを追加します。

```
Add-EsxSoftwarePackage -ImageProfile My_Profile -SoftwarePackage partner-package
```

この cmdlet は、標準の検証テストをイメージ プロファイルに実行します。検証が成功すると、cmdlet は検証および変更済みのイメージ プロファイルを返します。追加する VIB が別の VIB に依存している場合、コマンドレットはその情報を表示し、依存を解決できる VIB を含めます。追加する VIB の許容レベルがイメージ プロファイルの許容レベルより低い場合は、エラーが発生します。

イメージ プロファイルの比較

2 つのイメージ プロファイルを比較するには、vSphere Client を使用します。これにより、両者の VIB リスト、バージョン、または許容レベルが同じかどうかを確認できます。

前提条件

- vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。[vSphere ESXi Image Builder の設定](#)を参照してください。
- ソフトウェア デポを vSphere ESXi Image Builder インベントリに追加またはインポートします。[ソフトウェア デポの追加およびソフトウェア デポのインポート](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っていません。

- 2 [ソフトウェア デポ] タブで、ドロップダウン メニューを使用して、使用するイメージ プロファイルが含まれているソフトウェア デポを選択します。

- 3 [イメージ プロファイル] タブで、イメージ プロファイルを選択し、[比較対象] をクリックします。

[イメージ プロファイルの比較] ウィザードが表示されます。

- 4 [変更] をクリックして 2 つ目のイメージ プロファイルを選択します。

[イメージ プロファイルの選択] ページが表示されます。

- 5 ドロップダウン メニューからソフトウェア デポを選択し、2 つ目のイメージ プロファイルをクリックします。

- 6 [イメージ プロファイルの比較] ページで、[ソフトウェア パッケージ] ドロップダウン メニューから比較オプションを選択します。

リストの左側に、最初に選択したイメージ プロファイルに含まれる VIB の詳細が表示されます。リストの右側には、2 つ目のイメージ プロファイルの情報が表示されます。同じとマークされた VIB は、両方のプロファイルの間で同一です。一方のイメージ プロファイルにのみ存在する VIB は、存在しないイメージ プロファイルの隣になしとマークされます。

PowerCLI コマンドレットを使用したイメージ プロファイルの比較

2 つのイメージ プロファイルを比較するには、Compare-EsxImageProfile コマンドレットを使用します。これにより、たとえば、両者の VIB リストや許容レベルが同じかどうかを確認できます。PowerShell の比較演算子を使用してイメージ プロファイルまたはそのプロパティを比較することもできます。

前提条件

PowerCLI および前提となるすべてのソフトウェアをインストールします。vSphere ESXi Image Builder の構成を参照してください。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	<ol style="list-style-type: none"> a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 (オプション) Get-EsxImageProfile コマンドレットを実行して、使用可能なデポ内のすべてのイメージ プロファイルのリストを表示します。

表示されたリストで、比較するイメージ プロファイルの名前を見つけます。

- 3 イメージ プロファイルを比較するには、比較するイメージ プロファイルを変数に割り当てる必要があります。たとえば、変数 \$imageProfile1 および \$imageProfile2 を作成して、比較するイメージ プロファイルの名前を格納します。

```
$imageProfile1
= Get-EsxImageProfile -Name "ImageProfile1"
$imageProfile2
= Get-EsxImageProfile -Name "ImageProfile2"
```

- 4 Compare-EsxImageProfile コマンドレットまたは -eq 比較演算子 (ブール値を返す) を使用して 2 つのイメージ プロファイルを比較します。

- Compare-EsxImageProfile コマンドレットを使用して、2 つのイメージ プロファイルを比較し、両者の相違点の詳細な説明を取得します。

```
Compare-EsxImageProfile -ReferenceProfile
$imageProfile1 -ComparisonProfile $imageProfile2
```

- `-eq` 比較演算子を使用して、2つのイメージ プロファイルの VIB リストと許容レベルを比較します。

```
if ($imageProfile1 -eq $imageProfile2) {
    Write-host "Successfully verified that both image profiles are equal."
} else {
    Write-host "Failed to verify that the image profiles are equal."
}
```

- `-eq` 比較演算子を使用して、2つのイメージ プロファイルの特定のプロパティを比較します。

```
if ($imageProfile1.vendor -eq $imageProfile2.vendor) {
    Write-host "Successfully verified that both image profiles are equal."
} else {
    Write-host "Failed to verify that the image profiles are equal."
}
```

イメージ プロファイルの別のソフトウェア デポへの移動

vSphere Client を使用してカスタム デポ間でイメージ プロファイルを移動し、ソース デポの構成に影響を与えずに変更できます。

前提条件

- vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。[vSphere ESXi Image Builder の設定](#)を参照してください。
- ソフトウェア デポを vSphere ESXi Image Builder インベントリに追加またはインポートします。[ソフトウェア デポの追加およびソフトウェア デポのインポート](#)を参照してください。
- 少なくとも1つのカスタム デポ が vSphere ESXi Image Builder インベントリにあることを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っています。

- 2 [ソフトウェア デポ] タブで、ドロップダウン メニューを使用して、使用するイメージ プロファイルが含まれているソフトウェア デポを選択します。
- 3 [イメージ プロファイル] タブで、イメージ プロファイルを選択し、[移動先] をクリックします。
- 4 ドロップダウン メニューから、イメージ プロファイルを移動するカスタム デポを選択します。
- 5 [OK] をクリックします。

結果

イメージ プロファイルは新しいソフトウェア デポにあります。

ISO またはオフラインバンドルの ZIP へのイメージ プロファイルのエクスポート

イメージ プロファイルは、vSphere Client を使用して ISO イメージまたは ZIP ファイルにエクスポートできません。

前提条件

イメージ プロファイルは、vSphere Client を使用して ISO イメージまたは ZIP ファイルにエクスポートできます。ISO イメージは ESXi インストーラとして使用するか、vSphere Lifecycle Manager でホストをアップグレードするために使用できます。ZIP ファイルには、メタデータとイメージ プロファイルの VIB が含まれます。ESXi のアップグレードに、またはオフライン デポとして使用できます。

- vSphere ESXi Image Builder サービスが有効で、実行されていることを確認します。[vSphere ESXi Image Builder の設定](#)を参照してください。
- ソフトウェア デポを vSphere ESXi Image Builder インベントリに追加またはインポートします。[ソフトウェア デポの追加およびソフトウェア デポのインポート](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere ESXi Image Builder サービスを利用する権限を持っています。

- 2 [ソフトウェア デポ] タブで、ドロップダウン メニューを使用して、使用するイメージ プロファイルが含まれているソフトウェア デポを選択します。

- 3 [イメージ プロファイル] タブでエクスポートするイメージ プロファイルを選択し、[エクスポート] をクリックします。

[イメージ プロファイルのエクスポート] ウィザードが表示されます。

- 4 エクスポートしたファイルのタイプを選択します。

オプション	説明
ISO	イメージ プロファイルを起動可能な ISO イメージにエクスポートします。CD または DVD に焼くことのできる ISO イメージを作成し、ステートレス ESXi インスタンスの起動に使用する場合は、[ISO にインストーラを含めないでください] チェックボックスを選択します。
ZIP	イメージ プロファイルを ZIP ファイルにエクスポートします。

- 5 (オプション) イメージ プロファイルの許容レベル検証をバイパスする場合は、[許容レベル チェックをスキップします] を選択します。

- 6 [OK] をクリックします。

[ダウンロード] リンクから、選択したイメージ プロファイルの「イメージ プロファイルのダウンロード」列の内容の生成が開始されます。

- 7 イメージが正常に生成されたら、[ダウンロード] をクリックして、エクスポートしたファイルを保存します。

次のステップ

- イメージ プロファイルを新しい vSphere Auto Deploy ルールに関連付けて、ESXi ホストをプロビジョニングすることができます。[デプロイ ルールの作成またはデプロイ ルールのクローンの作成](#)を参照してください。
- イメージ プロファイルを ESXi ホストに関連付けることができます。[vSphere Auto Deploy インベントリへのホストの追加](#)を参照してください。

■ ホストのイメージ プロファイルの関連付けの編集.

PowerCLI コマンドレットを使用した ISO またはオフラインバンドルの ZIP へのイメージ プロファイルのエクスポート

イメージ プロファイルは、コンポーネントのファイルおよびフォルダの ISO イメージまたは ZIP ファイルにエクスポートできます。cmdlet の 1 回の実行で、両方を作成することはできません。ISO イメージは、ESXi インストーラとして使用するか、アップグレード用に vSphere Lifecycle Manager にアップロードすることができます。

前提条件

PowerCLI および前提となるすべてのソフトウェアをインストールします。vSphere ESXi Image Builder の構成を参照してください。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 Export-EsxImageProfile を実行して、イメージ プロファイルをエクスポートします。

エクスポート形式	コマンドレット
ISO イメージ	Export-EsxImageProfile (-ExportToIso パラメータを指定)
オフライン デポの ZIP ファイル	Export-EsxImageProfile (-ExportToBundle パラメータを指定)

結果

ISO イメージの場合、vSphere ESXi Image Builder が VIB の署名を検証し、VIB のバイナリをイメージに追加して、指定された場所にイメージをダウンロードします。ZIP ファイルの場合、vSphere ESXi Image Builder は VIB の署名を検証して、指定された場所に VIB のバイナリをダウンロードします。

例：イメージ プロファイルのエクスポート

次の手順に従って、イメージ プロファイルを ISO イメージにエクスポートします。

- 1 ソフトウェア デポを追加します。

```
Add-EsxSoftwareDepot -DepotUrl url_or_file
```

- 2 使用可能なすべてのイメージ プロファイルを表示して、エクスポートするイメージ プロファイルの名前を探します。

```
Get-EsxImageProfile
```

- 3 イメージ プロファイルをエクスポートします。

```
Export-EsxImageProfile -ImageProfile "myprofile" -ExportToIso -FilePath iso_name
```

次の手順に従って、イメージ プロファイルをコンポーネントのファイルおよびフォルダの ZIP ファイルにエクスポートします。

- 1 ソフトウェア デポを追加します。

```
Add-EsxSoftwareDepot -DepotUrl url_or_file
```

- 2 使用可能なすべてのイメージ プロファイルを表示して、エクスポートするイメージ プロファイルの名前を探します。

```
Get-EsxImageProfile
```

- 3 イメージ プロファイルをエクスポートします。

```
Export-EsxImageProfile -ImageProfile "myprofile" -ExportToBundle -FilePath C:\my_bundle.zip
```

次のステップ

ISO イメージは、ESXi のインストールで使用するか、アップグレードを実行するために vSphere Lifecycle Manager にアップロードします。

ZIP ファイルは、ESXi のインストールをアップグレードするために使用します。

- ZIP ファイルを vSphere Lifecycle Manager にインポートして、バッチ ベースラインとともに使用します。
- ZIP ファイルを ESXi ホストまたはデータストアにダウンロードし、`esxcli software vib` コマンドを実行して ZIP ファイルの VIB をインポートします。

『vSphere のアップグレード』ドキュメントを参照してください。

イメージ プロファイルの再生成

Auto Deploy により、イメージを使用して管理するクラスタにステートフル ESXi ホストを追加する場合は、すべてのホストに同じソフトウェア仕様が設定されていることを確認する必要があります。

Auto Deploy ルールが作成され、ターゲット クラスタがイメージによって管理されている場合、プリブート実行環境 (PXE) イメージがクラスタのイメージ仕様に基づいて作成されます。生成された PXE イメージはキャッシュされ、自動的に更新されません。その結果、vSphere Lifecycle Manager でイメージ仕様を変更する場合は、PXE イメージを手動で更新する必要があります。

イメージによって管理されるクラスタにホストを追加するためのルールを作成する方法については、『vSphere Lifecycle Manager』ドキュメントの「Auto Deploy を使用してイメージによって管理されるクラスタにホストを追加する」を参照してください。

前提条件

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。
デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。
- 2 [デプロイ ルール] タブで、目的のルールをクリックします。
ESXi ホストをイメージによって管理するクラスタに一致させるルールを選択する必要があります。
- 3 ルールが有効な場合は、まずルールを無効にする必要があります。
 - a [ルールの有効化/無効化] タブをクリックします
 - b ダイアログ ボックスで、ルールを選択します。
 - c ダイアログ ボックスで [無効化] を選択し、[OK] をクリックします。
- 4 [イメージ プロファイルの再作成] を選択し、確認ダイアログ ボックスで [再作成] をクリックします。
- 5 (オプション) ルールを再度有効にします。
 - a [ルールの有効化/無効化] タブをクリックします
 - b ダイアログ ボックスで、ルールを選択します。
 - c ダイアログ ボックスで [有効化] を選択し、[OK] をクリックします。

結果

PXE イメージで最新のイメージ仕様が実行されます。

PowerCLI を使用したセッション間でのイメージ プロファイルの保持

イメージ プロファイルをソフトウェア デポの ZIP ファイルにエクスポートして、そのデポを次のセッションで追加することができます。

前提条件

イメージ プロファイルを作成して PowerCLI セッションを終了すると、新しいセッションの開始時にはそのイメージ プロファイルを使用できなくなります。イメージ プロファイルをソフトウェア デポの ZIP ファイルにエクスポートして、そのデポを次のセッションで追加することができます。

PowerCLI および前提となるすべてのソフトウェアをインストールします。[vSphere ESXi Image Builder の構成](#) を参照してください。

手順

- 1 PowerCLI セッションで、たとえば既存のイメージ プロファイルのクローンを作成して VIB を追加することにより、イメージ プロファイルを作成します。
- 2 ExportToBundle パラメータを指定して Export-EsxImageProfile を呼び出すことにより、イメージ プロファイルを ZIP ファイルにエクスポートします。

```
Export-EsxImageProfile -ImageProfile "my_profile" -ExportToBundle -FilePath
"C:\isos\temp-base-plus-vib25.zip"
```

- 3 PowerCLI セッションを終了します。
- 4 新しい PowerCLI セッションを開始するときには、イメージ プロファイルを含むデポを追加してアクセスします。

```
Add-EsxSoftwareDepot "C:\isos\temp-base-plus-vib25.zip"
```

PowerCLI コマンドレットを使用した VIB の比較

PowerShell の比較演算子を使用して、2 つの VIB またはそのプロパティを比較できます。

前提条件

PowerCLI および前提となるすべてのソフトウェアをインストールします。vSphere ESXi Image Builder の構成を参照してください。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	<ol style="list-style-type: none"> a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 (オプション) Get-EsxSoftwarePackage コマンドレットを実行して、使用可能なすべての VIB を表示します。

表示されたリストで、比較する VIB の名前を見つけます。

- 3 VIB を比較するには、比較する VIB を変数に割り当てる必要があります。

たとえば、変数 \$vib1 および \$vib2 を作成して、比較する VIB の名前を格納します。

```
$vib1 = Get-EsxSoftwarePackage -Name "ReferenceVIB"
$vib2 = Get-EsxSoftwarePackage -Name "ComparisonVIB"
```

- 4 比較演算子を使用して、VIB のコンテンツと許容レベル、または個々のプロパティを比較します。

- 2 つの VIB のコンテンツと許容レベルを比較します。

```
if ($vib1 -eq $vib2) {
    Write-host "Successfully verified that both VIBs are equal."
} else {
    Write-host "Failed to verify that the VIBs are equal."
}
```

- `-eq`、`-lt`、`-le`、`-gt`、`-ge` などの比較演算子を使用して、VIB の個々のプロパティを比較します。

```
if ($vib1.VersionObject -lt $vib2.VersionObject) {
    Write-host "Successfully verified that both the VIBs are equal."
} else {
    Write-host "Failed to verify that the VIBs are equal."
}
```

PowerCLI コマンドレットを使用した vSphere ESXi Image Builder のワークフロー

vSphere ESXi Image Builder のワークフローは、コマンドレットの使用例であり、実際のタスクを表すものではありません。

vSphere ESXi Image Builder のワークフローは、コマンドレットの使用例です。ワークフローは、実際のタスクは表しませんが、`cmdlet` のさまざまな使用法を確認する方法を示します。管理者がワークフローを試す場合は、PowerCLI、Microsoft PowerShell、またはその両方の経験が役立ちます。

デポの内容の調査

あらゆる種類のワイルドカードの式を使用して、vSphere ESXi Image Builder コマンドレットでソフトウェア デポと VIB を調べることができます。

ワークフロー自体は、名前別にパラメータを渡します。ただし、変数にアクセスすることにより、パラメータをオブジェクトとして渡すことができます。

フィルタ オプションおよびワイルドカードの式を使用して、デポの内容を調査することができます。

前提条件

PowerCLI および前提となるソフトウェアがインストールされていることを確認します。[vSphere ESXi Image Builder の構成](#) を参照してください。

手順

- 1 PowerCLI セッションで、使用するデポごとに、`Add-EsxSoftwareDepot cmdlet` を実行します。

オプション	操作
リモート デポ	<code>Add-EsxSoftwareDepot -DepotUrl <depot_url></code> を実行します。
ZIP ファイル	<ol style="list-style-type: none"> a ZIP ファイルをローカル ファイル システムにダウンロードします。 b <code>Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip</code> を実行します

この `cmdlet` は、1 つ以上の `SoftwareDepot` オブジェクトを返します。

- 2 イメージ プロファイルを取得します。

ベンダー、名前、および許容レベルでフィルタリングできます。

- `Get-EsxImageProfiles`

セッションに追加したすべてのデポから `ImageProfile` オブジェクトの配列を返します。

```
■ Get-ESXImageProfile -Vendor "C*"
```

名前が C で始まるベンダーによって作成された、すべてのイメージ プロファイルを返します。

3 Get-ESXSoftwarePackage cmdlet を使用することにより、ソフトウェア パッケージを取得します。

ベンダーやバージョンなどでフィルタリングしたり、PowerShell の標準のワイルドカード文字を使用することができます。

```
■ Get-ESXSoftwarePackage -Vendor "V*"
```

名前が V で始まるベンダーから、すべてのソフトウェア パッケージを返します。

```
■ Get-ESXSoftwarePackage -Vendor "V*" -Name "*scsi*"
```

名前が V で始まるベンダーから、名前に文字列 scsi を含むすべてのソフトウェア パッケージを返します。

```
■ Get-ESXSoftwarePackage -Version "2.0*"
```

バージョンの文字列が 2.0 で始まる、すべてのソフトウェア パッケージを返します。

4 -Newest を使用して、最新のパッケージを検索します。

```
■ Get-ESXSoftwarePackage -Vendor "V*" -Newest
```

名前が V で始まるベンダーの最新パッケージを返し、情報を表形式で表示します。

```
■ Get-ESXSoftwarePackage -Vendor "V*" -Newest | format-list
```

ソフトウェア パッケージの要求の出力を PowerShell の format-list cmdlet にリンクするパイプラインを使用して、各ソフトウェア パッケージに関する詳細情報を返します。

5 イメージ プロファイルの VIB のリストを表示します。

```
(Get-ESXImageProfile -Name "Robin's Profile").VibList
```

VibList は ImageProfile オブジェクトのプロパティです。

6 特定の日付の前または後でリリースされたソフトウェア パッケージを、CreatedBefore または CreatedAfter パラメータを使用して取得します。

```
Get-ESXSoftwarePackage -CreatedAfter 7/1/2010
```

例：変数を使用した、デポの内容の調査

このワークフローの例では、パラメータを名前でも代りにオブジェクト (変数内の位置によってアクセスされる) として渡して、デポの内容を調査します。次のコマンドは、PowerCLI プロンプトから順番に実行できます。名前は、インストールに合わせて適切な名前に置き換えます。

```
Get-ESXSoftwarePackage -Vendor "V*"
Get-ESXSoftwarePackage -Vendor "V*" -Name "r*"
```

```
Get-ESxSoftwarePackage -Version "2.0*"
$ip1 = Get-ESxImageProfile -name ESX-5.0.0-123456-full
$ip1.VibList
Get-ESxSoftwarePackage -CreatedAfter 7/1/2010
```

クローン作成によるイメージ プロファイルの作成

vSphere ESXi Image Builder コマンドレットを使用して、使用可能なデポの確認、デポの追加、イメージ プロファイル情報の表示、新しいイメージ プロファイルのクローン作成を行います。

発行済みのプロファイルは、通常読み取り専用であり、変更できません。発行済みのプロファイルが読み取り専用でない場合にも、プロファイルを変更せずに、クローンを作成することがベスト プラクティスです。これは、元のプロファイルを変更すると、元の内容が消えてしまうためです。デポに再接続しない限り、未変更の元のプロファイルに戻すことはできません。

プロファイルのクローン作成のワークフローには、システムの現在の状態の確認、ソフトウェア デポの追加、およびプロファイルのクローン作成が含まれます。

前提条件

vSphere ESXi Image Builder cmdlet を使用すると、使用可能なデポの確認、デポの追加、イメージ プロファイル情報の表示、および使用可能なイメージ プロファイルのクローン作成による新規イメージ プロファイルの作成を行うことができます。

PowerCLI および前提となるソフトウェアがインストールされていることを確認します。[vSphere ESXi Image Builder の構成](#) を参照してください。

手順

- 1 PowerShell のウィンドウで、現在のセッション用に定義されているソフトウェア デポがないか確認します。

```
$DefaultSoftwareDepots
```

PowerShell は、現在定義されているデポを返します。または、PowerShell を起動したばかりの場合は、なにも返しません。

- 2 クローンを作成するプロファイルを含むデポが結果に表示されない場合は、そのデポを現在のセッションに追加します。

オプション	操作
リモート デポ	Add-ESxSoftwareDepot -DepotUrl <i>depot_url</i> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル パスにダウンロードします。 b Add-ESxSoftwareDepot -DepotUrl C:\file_path\offline-bundle.zip を実行します。

PowerShell は、指定されたデポを現在のセッションに追加し、現在のすべてのデポを一覧表示します。

- 3 (オプション) \$DefaultSoftwareDepots 変数を確認します。ここで、新しく追加されたデポが返されます。

- 4 使用できるすべてのイメージ プロファイルを表示します。

```
Get-EsxImageProfile
```

- 5 イメージ プロファイルのクローンを作成するには、イメージ プロファイルの名前、新しいプロファイルの名前、およびベンダーの名前を入力します。

```
$ip = New-EsxImageProfile -CloneProfile base-tbd-v1 -Name "Test Profile 42" -Vendor "Vendor20"
```

- 6 (オプション) 新しく作成されたイメージ プロファイル \$ip を表示します。

PowerShell は、イメージ プロファイルに関する情報を表形式で返します。

Name	Vendor	Last Modified	Acceptance Level
Test Profile 42	Vendor20	9/15/2010 5:45:43...	PartnerSupported

例：変数を使用した、クローン作成によるイメージ プロファイルの作成

このワークフローの例では、このワークフローの手順を繰り返しますが、パラメータを名前で渡す代わりに、変数内の位置に基づいてアクセスされるオブジェクトとして渡します。次の コマンドレット を PowerCLI プロンプトから順番に実行できます。

```
$DefaultSoftwareDepots
Add-EsxSoftwareDepot -DepotUrl depot_url
$DefaultSoftwareDepots
$profs = Get-EsxImageProfile
$profs
$ip = New-EsxImageProfile -CloneProfile $profs[2] -Name "new_profile_name" -Vendor "my_vendor"
$ip
```

イメージ プロファイルの新規作成のワークフロー

既存のイメージ プロファイルのクローンを作成することも、新しいイメージ プロファイルを作成することもできます。このプロファイルでは、依存関係と許容レベルを定義する必要があります。

システムでは、基本イメージに追加する VIB の許容レベルが、基本イメージの許容レベル以上であることが求められます。低い許容レベルの VIB をイメージ プロファイルに追加する場合、イメージ プロファイル許容レベルを低くする必要があります。詳細については、[イメージ プロファイル許容レベルの設定](#)を参照してください。

コマンド ラインでパラメータを指定する代わりに、PowerShell のプロンプト メカニズムを使用して文字列パラメータを指定できます。プロンプトは、オブジェクトなどのその他のパラメータには機能しません。

前提条件

イメージ プロファイルは、ほとんどの場合、既存のプロファイルのクローン作成によって作成します。一部の VMware カスタマやパートナーで、イメージ プロファイルを新しく作成する必要がある場合があります。スクラッチからイメージ プロファイルを新規作成する場合は、依存関係や許容レベルに十分に注意します。

- PowerCLI および前提となるソフトウェアがインストールされています。vSphere ESXi Image Builder の構成を参照してください。
- 基本のイメージと 1 つ以上の VIB を含むデポにアクセスできます。VMware および VMware パートナーは、URL でアクセスできるパブリックなデポを提供します。VMware または VMware パートナーは、ローカル環境に解凍してファイル パスを使用してアクセスできる ZIP ファイルを作成することができます。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-EsxSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-EsxSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-EsxSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1 つ以上の SoftwareDepot オブジェクトを返します。

- 2 Get-EsxImageProfile cmdlet を実行して、現在表示されているすべてのデポ内のすべてのイメージ プロファイルをリスト表示します。任意の引数を使用して出力をフィルタリングすると、検索を絞り込むことができます。

```
Get-EsxSoftwarePackage -CreatedAfter 7/1/2010
```

- 3 新しいプロファイルを作成し、名前とベンダーを割り当てて基本パッケージを追加します。

```
New-EsxImageProfile -NewProfile -Name "Test #2" -vendor "Vendor42" -SoftwarePackage esx-base[0],esx-xlibs[0]
```

この例は、esx-base パッケージを使用します。イメージ プロファイルを新規作成する場合、通常、esx-base パッケージを含めます。スペースを含む名前は、引用符で囲みます。

- 4 新しいパッケージに関する詳細情報については、パイプラインを使用して、新しいイメージ プロファイルを format-list に渡します。

```
(Get-EsxImageProfile -Name "Test #2").VibList | format-list
```

例：変数を使用した、イメージ プロファイルの新規作成

このコマンド シーケンスはこのワークフローの手順を繰り返しますが、パラメータを名前ですべて代わりに、オブジェクト（変数内の位置によってアクセスされる）として渡します。次のコマンドを PowerCLI プロンプトで順番に実行できます。

```
Add-ESXSoftwareDepot depoturl
$pkgs = Get-ESXSoftwarePackage -CreatedAfter 7/1/2010
$ip2 = New-ESXImageProfile -NewProfile -Name "Test #2" -vendor "Vendor42" -SoftwarePackage
$pkgs[0]
$ip2.VibList | format-list
```

イメージ プロファイルの編集のワークフロー

PowerCLI を使用してイメージ プロファイルのクローンを作成して編集することで、カスタム イメージを作成できます。

前提条件

PowerCLI を使用してイメージ プロファイルのクローンを作成して編集することで、カスタム イメージを作成できます。既存のプロファイルでは、1つ以上の VIB を追加したり削除したりすることができます。VIB の追加や削除によって、イメージ プロファイルが正常に機能しない場合は、エラーが発生します。

- PowerCLI および前提となるソフトウェアがインストールされています。[vSphere ESXi Image Builder の構成](#) を参照してください。
- 基本のイメージと 1つ以上の VIB を含むデポにアクセスできます。VMware および VMware パートナーは、URL でアクセスできるパブリックなデポを提供します。VMware または VMware パートナーは、ローカル環境にダウンロードし、ファイル パスを使用してアクセスできる、ZIP ファイルを作成することができます。

手順

- 1 PowerCLI セッションで、使用するデポごとに、Add-ESXSoftwareDepot cmdlet を実行します。

オプション	操作
リモート デポ	Add-ESXSoftwareDepot -DepotUrl <depot_url> を実行します。
ZIP ファイル	<ol style="list-style-type: none"> a ZIP ファイルをローカル ファイル システムにダウンロードします。 b Add-ESXSoftwareDepot -DepotUrl C:\<file_path>\<offline-bundle>.zip を実行します

この cmdlet は、1つ以上の SoftwareDepot オブジェクトを返します。

- 2 パイプラインを使用して、編集するイメージ プロファイルを format-list に渡し、詳細情報を参照します。

この例に示す [イメージ プロファイルの新規作成のワークフロー](#) で作成されたイメージ プロファイルには、基本のイメージのみが格納されます。新しく作成されたイメージ プロファイルは、デポには含まれません。代わりに、名前で、または変数にバインドして、新しいイメージ プロファイルにアクセスします。

```
Get-ESXImageProfile "Test #2" | format-list
```

PowerShell は情報を返します。

```
Name           : Test #2
Vendor         : Vendor42
...
VibList        : {esx-base 5.0.0.-...,}
```

- 3 (オプション) 追加する VIB の許容レベルがイメージ プロファイルのレベルよりも低い場合は、イメージ プロファイルの許容レベルを変更します。

```
Set-ExsImageProfile -ImageProfile "Test #2" -AcceptanceLevel VMwareAccepted
```

PowerShell は、変更されたプロファイルに関する情報を表形式で返します。

Name	Vendor	Last Modified	Acceptance Level
Test #2	Vendor42	9/22/2010 12:05:...	VMwareAccepted

- 4 イメージ プロファイルにソフトウェア パッケージ (VIB) を追加します。パッケージは名前を追加できます。

```
Add-ExsSoftwarePackage -ImageProfile "Test #2"
                        -SoftwarePackage NewPack3
```

PowerShell は、イメージ プロファイルに関する情報を表形式で返します。

Name	Vendor	Last Modified	Acceptance Level
Test #2	Vendor42	9/22/2010 12:05:...	VMwareAccepted

注： ソフトウェア パッケージを追加するときにエラーが発生した場合は、許容レベルで問題が起きることがあります。許容レベルの操作を参照してください。

- 5 イメージ プロファイルを再び表示します。

```
Get-ExsImageProfile "Test #2" | format-list
```

新しいソフトウェア パッケージを含めるように VIB のリストが更新され、その情報が表示されます。

```
Name           : Test #2
Vendor         : Vendor42
...
VibList        : {esx-base 5.0.0.-..., NewPack3}
```

例：変数を使用した、イメージ プロファイルの編集

この コマンドレット シーケンスはこのワークフローの手順を繰り返しますが、パラメータを名前で渡す代わりに、オブジェクト（変数内の位置によってアクセスされる）として渡します。次の コマンドレット を PowerCLI プロンプトから順番に実行できます。

```
Add-ESXSoftwareDepot -DepotUrl depot_url
$ip2 = Get-ESXImageProfile -name "Test #2"
$ip2 | format-list
Set-ESXImageProfile -ImageProfile $ip2 -AcceptanceLevel VMwareAccepted
Add-ESXImageSoftwarePackage -ImageProfile $ip2 -SoftwarePackage NewPack3
$ip2 | format-list
```

ESXi のインストール

スクリプトによるインストール、または vSphere Auto Deploy を使用して、ESXi を対話形式でインストールできます。

ESXi の対話形式のインストール

ホストが 5 台未満の小規模な導入環境では、対話型インストール オプションを使用します。

典型的な対話型インストールでは、ESXi インストーラを起動し、インストーラのプロンプトに回答して、ESXi をローカル ホストのディスクにインストールします。インストーラは、ターゲット ディスクの再フォーマットとパーティショニングを行い、ESXi の起動イメージをインストールします。以前に ESXi をターゲット ディスクにインストールしたことがない場合は、ハードウェア ベンダーのパーティション、オペレーティング システムのパーティション、および関連データを含む、このドライブ上のすべてのデータが上書きされます。

注： データを失わないためには、ESXi をインストールする前にデータを別のマシンに移行してください。

ESXi または ESX の以前のインストール、または VMFS データストアが含まれるディスク上に ESXi をインストールする場合は、インストーラはアップグレードのオプションを提示します。『vSphere アップグレード』のドキュメントを参照してください。

ESXi の対話型インストール

接続した vSphere ESXi CD または DVD、起動可能な USB デバイスからサーバを起動するか、ネットワーク上の場所からサーバを PXE 起動します。インストール ウィザードのプロンプトに従って、ESXi をディスクにインストールします。

対話形式での ESXi のインストール

ESXi の CD/DVD または USB フラッシュ ドライブを使用して、ESXi ソフトウェアを SAS、SATA、SCSI ハード ドライブ、または USB ドライブにインストールします。

前提条件

- 次のいずれかの場所に ESXi のインストーラ ISO が存在する必要があります。
 - CD または DVD。インストール CD/DVD がない場合は作成できます。ESXi インストーラ ISO イメージのダウンロードおよび CD or DVD への書き込みを参照してください。
 - USB フラッシュ ドライブ。USB フラッシュ ドライブのフォーマットによる ESXi インストールまたはアップグレードの起動を参照してください。

注： ESXi インストーラを PXE ブートして、対話形式のインストールまたはスクリプトによるインストールを実行することもできます。ESXi インストーラのネットワーク ブートを参照してください。

- サーバ ハードウェアのクロックが UTC に設定されていることを確認します。この設定はシステム BIOS または UEFI で行います。
- キーボードおよびモニターが、ESXi ソフトウェアをインストールするマシンに接続されていることを確認します。または、リモート管理アプリケーションを使用します。リモート管理アプリケーションの使用を参照してください。
- ネットワーク ストレージを切断することを検討する。ネットワーク ストレージを切断することにより、使用可能なディスク ドライブをインストーラが検索する時間が短縮されます。ネットワーク ストレージを切断すると、切断されたディスク上のファイルはすべて、インストール時に使用できなくなります。

ESX または ESXi の既存のインストールが含まれている LUN は切断しないでください。既存の ESX インストールのサービス コンソールが含まれている VMFS データストアは切断しないでください。これらの操作を行うと、インストールに影響が生じる可能性があります。

- ESXi インストール ウィザードで必要な情報を収集します。ESXi のインストールに必要な情報を参照してください。
- ESXi Embedded がホスト マシンに存在しないことを確認します。ESXi Installable および ESXi Embedded は、同じホスト上に共存できません。

手順

- 1 ESXi のインストーラ CD/DVD を CD/DVD-ROM ドライブに挿入するか、インストーラ USB フラッシュ ドライブを接続して、マシンを再起動します。
- 2 CD-ROM デバイスまたは USB フラッシュ ドライブから起動するように BIOS または UEFI を設定します。

注： システムにデータ処理ユニット (DPU) がある場合、DPU に ESXi をインストールして起動することは UEFI を使用してのみ可能です。

起動順序の変更に関する情報については、ハードウェア ベンダーのドキュメントを参照してください。

- 3 ウェルカム画面で Enter を押し、続行します。

- 4 [次へ] を押してエンド ユーザー使用許諾契約書に同意します。

ESXi 8.0 Update 3 以降では、使用可能なデバイスのスキャンが完了した後、システムに DPU がある場合、それらのデバイスが、それぞれの PCI スロットとともに自動的に一覧表示されます。スロットを選択する必要はなくなりました。DPU デバイスは、同じベンダー、同じハードウェア バージョン、同じファームウェアにする必要があります。

- 5 [ESXi をインストールまたはアップグレードするディスクの選択] 画面で ESXi をインストールするドライブを選択し、[Enter] キーを押します。

選択したディスクの情報を確認するには、[F1] キーを押します。

注： ディスクを選択するときは、リスト上のディスクの順序に注意してください。ディスクの順序は BIOS または UEFI によって決定されるため、順序が正確でない場合があります。頻繁にドライブの追加と削除を行なっているシステムでは、このようなことが発生する可能性があります。

データを含むディスクを選択した場合は、[ディスクの選択内容の確認] 画面が表示されます。

以前にインストールした ESXi/ESX または VMFS データストアが含まれているディスク上にインストールする場合は、インストーラでいくつかのオプションが表示されます。

重要： 既存の ESXi インストールをアップグレードまたは移行する場合は、『ESXi のアップグレード』ドキュメントを参照してください。

vSAN ディスク グループ内のディスクを選択した場合、インストールは結果的に選択したディスクのタイプおよびグループのサイズによって異なります。

- SSD を選択した場合は、同じディスク グループ内の SSD および基盤となるすべての HDD がワイブされます。
- HDD を選択した場合は、ディスク グループのサイズが 2 より大きくなり、選択した HDD のみワイブされます。
- HDD ディスクを選択した場合は、ディスク グループのサイズが 2 以下になり、SSD および選択した HDD がワイブされます。

- 6 ホストのキーボード タイプを選択します。

キーボード タイプは、インストール後にダイレクト コンソールで変更できます。

- 7 ホストの root パスワードを入力します。

パスワードは、インストール後にダイレクト コンソールで変更できます。

- 8 [インストールの確認] 画面では、vSphere システムに DPU がある場合、それぞれが個別の行に一覧表示されます。**F11** を押して、インストールの開始を確認します。

8.0 Update 3 以降では、システムに DPU がある場合に、ESXi と DPU のインストールに関する単一の進行状況バーが表示され、インストーラの実行中のステージを示すラベルが動的に更新されます。

vSAN ディスク グループの管理の詳細については、『vSphere のストレージ』ドキュメントを参照してください。

SD デバイスまたは USB デバイスを選択すると、ESXi-OSData パーティションを保存するパーシステント ディスクを選択するよう求める警告が表示されます。[ESX OSData を保存するディスクを選択] 画面で、使用可能な空き容量が最小 32 GB のパーシステント ストレージ デバイスを選択します。

- 9 インストールが完了したら、インストール CD、DVD、または USB フラッシュ ドライブを取り出します。

8.0 Update 3 以降では、いずれかのターゲットでインストールが成功しなかった場合に、[操作に失敗しました] 画面が表示されます。トラブルシューティングに役立てるため、画面上のエラー ログを記録します。

- 10 [Enter] キーを押してホストを再起動します。

- 11 手順 5 で ESXi をインストールしたドライブを最初の起動デバイスとして設定します。

起動順序の変更については、ハードウェア ベンダーのドキュメントを参照してください。

注： UEFI システムでは、起動デバイスを設定するための追加の手順が必要になる場合があります。UEFI モードでの ESXi のインストール後、ホストの起動に失敗する を参照してください。

結果

インストールが完了したあと、既存の VMFS データを ESXi ホストに移行できます。

各 ESXi イメージから 1 台のマシンを起動できます。1 つの共有 ESXi イメージから複数のデバイスを起動する処理はサポートされていません。

次のステップ

ESXi の基本的な管理内容およびネットワーク構成を設定します。ESXi のインストールおよび設定後を参照してください。

ソフトウェア iSCSI ディスクへの ESXi のインストール

ESXi をソフトウェア iSCSI ディスクにインストールする場合は、ターゲットの iSCSI 修飾名 (IQN) を構成する必要があります。

システムの起動中に、システムは POST (Power-On Self Test) を実施し、システムの BIOS に指定されている順序でアダプタの起動を開始します。起動の順番が iBFT (iSCSI Boot Firmware Table) アダプタの番になると、アダプタはターゲットへの接続を試みますが、ターゲットからは起動しません。前提条件を参照してください。

iSCSI ターゲットへの接続に成功すると、iSCSI 起動ファームウェアは iSCSI の起動の構成を iBFT に保存します。次に起動するアダプタは、ESXi のインストール メディア (マウントされた ISO イメージまたは物理 CD-ROM のいずれか) である必要があります。

前提条件

- ターゲットの IQN が、iBFT BIOS のターゲット パラメータの設定で構成されていることを確認します。この設定は、ネットワーク インターフェイス カード (NIC) のオプションの ROM にあり、iSCSI LUN 用に使用されます。システムのベンダーが提供するドキュメントを参照してください。

- iSCSI ターゲットに起動する iBFT アダプタ オプションを無効にします。この操作は、iSCSI ターゲットではなく、ESXi インストーラが確実に起動するために必要です。システムを起動したら、プロンプトの指示に従って iBFT アダプタにログインし、iSCSI ターゲットに起動するオプションを無効にします。システムと iBFT アダプタのベンダーが提供するドキュメントを参照してください。ESXi のインストールが完了したら、ESXi のインストール先の LUN から起動するオプションを再び有効にできます。

手順

- 1 ESXi のインストール CD/DVD またはマウントされた ISO イメージから、対話形式のインストールを開始します。
- 2 ディスクの選択画面で、iBFT BIOS のターゲット パラメータの設定で指定した iSCSI ターゲットを選択します。

このメニューにターゲットが表示されない場合は、TCP/IP の設定とイニシエータの iSCSI IQN の設定が正しいことを確認します。ネットワークのアクセス制御リスト (ACL) を確認し、ターゲットにアクセスするための適切な権限がアダプタにあることを確認します。
- 3 プロンプトの指示に従って、インストールを完了します。
- 4 ホストを再起動します。
- 5 ホストの BIOS 設定で、iBFT アダプタの BIOS 構成を入力し、iSCSI ターゲットから起動するようにアダプタのパラメータを変更します。

システムのベンダーが提供するドキュメントを参照してください。

次のステップ

iBFT アダプタで、iSCSI ターゲットに起動するためのオプションを再び有効にします。これによって、ESXi をインストールした LUN からシステムが起動します。

スクリプトを使用した ESXi のインストール

インストール/アップグレード スクリプトはサポートされているコマンドを含むテキストファイルで、たとえば `ks.cfg` があります。

スクリプトのコマンド セクションには、ESXi のインストール オプションが含まれます。このセクションは必須で、スクリプトの先頭にある必要があります。

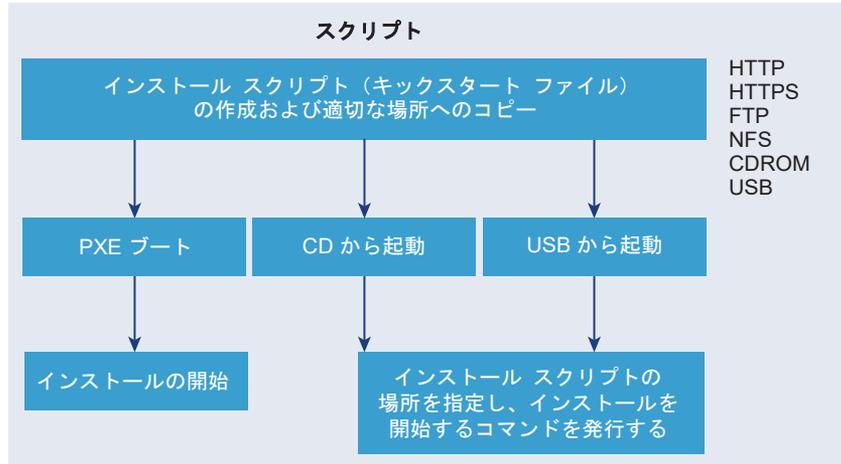
ESXi のスクリプトによるインストール

スクリプトの実行は、無人インストールで複数の ESXi ホストをデプロイするための効率的な方法です。

インストール スクリプトには、ホストの構成設定が含まれます。スクリプトは、同じ設定で複数のホストを構成するために使用できます。「[スクリプトを使用したホストのインストールまたはアップグレード](#)」を参照してください。

インストール スクリプトは、HTTP、HTTPS、FTP、NFS、CD-ROM、または USB を使用してホストからアクセスできる場所に保存されている必要があります。ESXi インストーラを PXE 起動したり、CD/DVD または USB ドライブから起動することが可能です。

図 4-3. スクリプトによるインストール



スクリプトを使用したインストールの方法

複数のマシンに ESXi をインストールする場合は、すべてのマシンに 1 つのスクリプトを使用したり、各マシンに別々のスクリプトを使用したりできます。

たとえば、ディスク名はマシンごとに異なるため、スクリプトで構成が必要な設定の 1 つは、ESXi のインストール先のディスクの選択になります。

表 4-12. スクリプトを使用したインストールの選択肢

オプション	操作
常に、複数のマシン上の最初のディスクにインストールします。	1 つのスクリプトを作成します。
それぞれのマシンで異なるディスクに ESXi をインストールします。	複数のスクリプトを作成します。

インストール先ディスクを指定するために必要なコマンドの詳細については、[インストール スクリプトおよびアップグレード スクリプトのコマンド](#) を参照してください。

インストール スクリプトまたはアップグレード スクリプトを実行するための起動オプションの入力

インストール スクリプトまたはアップグレード スクリプトを起動するには、ESXi インストーラの起動コマンドラインで起動オプションを入力します。

起動時に、キックスタート ファイルへのアクセス オプションの指定が必要な場合があります。起動オプションは、ブート ローダで Shift + O を押すことで入力できます。PXE 起動のインストールの場合は、`boot.cfg` ファイルの `kernelopts` 行を使用してオプションを渡すことができます。[boot.cfg ファイルについて](#) および [ESXi インストーラのネットワーク ブート](#) を参照してください。

インストール スクリプトの場所を指定するには、`ks=filepath` オプションを設定します。ここで、`filepath` は、キックスタート ファイルの場所を示します。このオプションを設定しないと、スクリプトを使用したインストールまたはアップグレードを開始できません。`ks=filepath` を省略すると、テキスト インストーラが実行されます。

サポートされている起動オプションは [起動オプション](#) に記載されています。

手順

- 1 ホストを起動します。
- 2 ESXi インストーラのウィンドウが表示されたら、Shift + O を押して起動オプションを編集します。



- 3 `runweasel` コマンド プロンプトで、
`ks=location of installation script plus boot command-line options` と入力します。

例：起動オプション

次の起動オプションを入力します。

```
ks=http://00.00.00.00/kickstart/ks-osdc-pdp101.cfg nameserver=00.00.0.0 ip=00.00.00.000
netmask=255.255.255.0 gateway=00.00.00.000
```

起動オプション

スクリプトを使用したインストールを実行する際には、起動時にキックスタート ファイルへのアクセス オプションを指定することが必要な場合があります。

サポートされる起動オプション

表 4-13. ESXi インストール用の起動オプション

起動オプション	説明
<code>BOOTIF=hwtype-MAC address</code>	<code>netdevice</code> オプションに似ていますが、 <code>syslinux.org</code> サイトで <code>SYSLINUX</code> の <code>IPAPPEND</code> オプションで説明されている、 <code>PXELINUX</code> 形式に関する点が異なります。
<code>gateway=ip address</code>	このネットワーク ゲートウェイを、インストール スクリプトおよびインストール メディアをダウンロードするために使用する、デフォルトのゲートウェイとして設定します。
<code>ip=ip address</code>	インストール スクリプトおよびインストール メディアのダウンロードに使用する固定 IP アドレスを設定します。注：このオプションでは <code>PXELINUX</code> 形式もサポートされています。 <code>syslinux.org</code> サイトで、 <code>SYSLINUX</code> の中の <code>IPAPPEND</code> オプションを参照してください。

表 4-13. ESXi インストール用の起動オプション (続き)

起動オプション	説明
<code>ks=cddrom:/path</code>	<p><i>path</i>にあるスクリプトによるインストールを実行します。このスクリプトは CD-ROM ドライブの CD にあります。各 CDDROM は、このパスと一致するファイルが見つかるまでマウントされ、チェックされます。</p> <p>重要: カスタム インストールまたはアップグレード スクリプトを使用してインストーラ ISO イメージを作成した場合は、大文字を使用してスクリプトのパスを入力する必要があります。例: <code>ks=cddrom:/KS_CUST.CFG</code>。</p>
<code>ks=file://path</code>	<i>path</i> にあるスクリプトを使用してインストールを実行します。
<code>ks=protocol://serverpath</code>	ネットワーク上の指定された URL に置かれたスクリプトを使用してスクリプトによるインストールを実行します。 <i>protocol</i> には <code>http</code> 、 <code>https</code> 、 <code>ftp</code> 、または <code>nfs</code> を指定できます。NFS プロトコルを使用している例は、 <code>ks=nfs://host/porturl-path</code> です。NFS URL の形式は、RFC 2224 で指定されています。
<code>ks=usb</code>	接続された USB ドライブにあるスクリプトにアクセスして、インストールを実行します。 <code>ks.cfg</code> という名前のファイルが検索されます。このファイルは、ドライブのルート ディレクトリに置かれている必要があります。複数の USB フラッシュ ドライブが接続されている場合は、 <code>ks.cfg</code> ファイルが見つかるまでそれらのドライブが検索されます。サポートされているのは FAT16 および FAT32 のファイル システムだけです。
<code>ks=usb:/path</code>	USB 上にある、指定されたパスにあるスクリプト ファイルを使用して、スクリプトによるインストールを実行します。
<code>ksdevice=device</code>	インストール スクリプトとインストール メディアの検索時にネットワーク アダプタ <i>device</i> の使用を試みます。MAC アドレス (00:50:56:C0:00:01 など) で指定します。この場所は <code>vmnicNN</code> 名にもできます。このサブコマンドを指定せず、ネットワーク経由でファイルを取得する必要がある場合は、最初に検出された接続済みネットワーク アダプタがデフォルトで使用されます。
<code>nameserver=ip address</code>	インストール スクリプトとインストール メディアのダウンロードに使用するドメイン名サーバを指定します。
<code>netdevice=device</code>	インストール スクリプトとインストール メディアの検索時にネットワーク アダプタ <i>device</i> の使用を試みます。MAC アドレス (00:50:56:C0:00:01 など) で指定します。この場所は <code>vmnicNN</code> 名にもできます。このサブコマンドを指定せず、ネットワーク経由でファイルを取得する必要がある場合は、最初に検出された接続済みネットワーク アダプタがデフォルトで使用されます。
<code>netmask=subnet mask</code>	インストール スクリプトおよびインストール メディアをダウンロードするネットワーク インターフェイスのサブネット マスクを指定します。

表 4-13. ESXi インストール用の起動オプション (続き)

起動オプション	説明
<code>vlanid=vlanid</code>	指定した VLAN 上に置くネットワーク カードを構成します。
<code>systemMediaSize=small</code>	<p>起動メディア上のシステム ストレージ パーティションのサイズを制限します。選択した値は、システムの目的に適合している必要があります。次の値の中から選択できます。</p> <ul style="list-style-type: none"> ■ <i>min</i> (32 GB、1 台のディスクまたは組み込みのサーバ用) ■ <i>small</i> (64 GB、512 GB 以上の RAM を搭載したサーバ用) ■ <i>default</i> (128 GB) ■ <i>max</i> (使用可能なすべての容量を使用、マルチテラバイトのサーバ用)

ESXi のインストール後の起動オプションの詳細については、VMware ナレッジベースの記事 [KB77009](#) を参照してください。

ESXi のインストールに使用されるインストール スクリプトとアップグレード スクリプト

デフォルトのスクリプトを使用して、最初に検出されたディスクに ESXi の標準インストールを実行できます。

ESXi のインストール時には、デフォルトのスクリプトを使用すると最初に検出されたディスクに対して標準インストールを実行でき、ブート ロード構成ファイルを使用すると、カーネル、カーネル オプション、および `mboot.c32` または `mboot.efi` ブート ロードがインストールに使用するブート モジュールを指定できます。

デフォルトの `ks.cfg` インストール スクリプトについて

ESXi インストーラには、最初に検出されたディスクへの標準インストールを実行するデフォルトのインストール スクリプトが含まれています。

デフォルトの `ks.cfg` インストール スクリプトは、初期 RAM ディスクの `/etc/vmware/weasel/ks.cfg` にあります。デフォルトの `ks.cfg` ファイルの場所は、`ks=file:///etc/vmware/weasel/ks.cfg` の起動オプションで指定できます。 [インストール スクリプトまたはアップグレード スクリプトを実行するための起動オプションの入力を参照してください。](#)

`ks.cfg` スクリプトを使用して ESXi をインストールする場合、デフォルトの root パスワードは `myp@ssw0rd` です。

インストール メディアにあるデフォルトのスクリプトは変更できません。インストール後、ESXi ホストを管理する vCenter Server に vSphere Client を使用してログインし、デフォルト設定を変更できます。

vSphere 8.0 では、システムにデータ処理ユニット (DPU) がある場合、`ks.cfg` スクリプトを使用して、ESXi を DPU にインストールすることもできます。

デフォルトのスクリプトには、次のコマンドが含まれます。

```
#
# Sample scripted installation file
#

# Accept the VMware End User License Agreement
vmaccepteula
```

```
# Set the root password for the DCUI and Tech Support Mode
rootpw myp@ssw0rd

# Install on the first local disk available on machine
install --firstdisk --overwritevmfs
  In case you system has DPU's, you also specify a PCI slot:
install --firstdisk --overwritevmfs --dpupcislots=<PCIeSlotID>

# Set the network to DHCP on the first network adapter
network --bootproto=dhcp --device=vmnic0

# A sample post-install script
%post --interpreter=python --ignorefailure=true
import time
stampFile = open('/finished.stamp', mode='w')
stampFile.write( time.asctime() )
```

boot.cfg ファイルについて

ブート ロード構成ファイル `boot.cfg` は、`mboot.c32` または `mboot.efi` ブート ロードが ESXi のインストールで使用されるカーネル、カーネル オプション、およびブート モジュールを指定します。

`boot.cfg` ファイルは ESXi のインストーラで提供されます。`boot.cfg` ファイルの `kernelopt` 行を変更して、インストール スクリプトの場所を指定したり、ほかのブート オプションを渡したりすることができます。

`boot.cfg` ファイルの構文は次のとおりです。

```
# boot.cfg -- mboot configuration file
#
# Any line preceded with '#' is a comment.

title=STRING
prefix=DIRPATH
kernel=FILEPATH
kernelopt=STRING
modules=FILEPATH1 --- FILEPATH2... --- FILEPATHn

# Any other line must remain unchanged.
```

`boot.cfg` に記述されるコマンドがブート ロードを構成します。

表 4-14. `boot.cfg` に記述されるコマンド

コマンド	説明
<code>title=STRING</code>	ブート ロードのタイトルを <code>STRING</code> に設定します。
<code>prefix=STRING</code>	(オプション) <code>kernel=</code> コマンドおよび <code>modules=</code> コマンド内の <code>[/]</code> または <code>http://</code> で始まらないすべての <code>FILEPATH</code> の前に <code>DIRPATH/</code> を追加します。
<code>kernel=FILEPATH</code>	カーネル パスを <code>FILEPATH</code> に設定します。

表 4-14. boot.cfg に記述されるコマンド（続き）

コマンド	説明
kernelopt= <i>STRING</i>	カーネル ブート オプションに <i>STRING</i> を追加します。
modules= <i>FILEPATH1</i> --- <i>FILEPATH2</i> ... --- <i>FILEPATHn</i>	読み込むモジュールを 3 つのハイフン (---) で区切ってリストします。

カスタムのインストールまたはアップグレードスクリプトを含む、インストーラ ISO イメージの作成および ESXi インストーラのネットワーク ブートを参照してください。

サポートされている、インストールまたはアップグレード スクリプトの保存場所

スクリプトを使用したインストールおよびアップグレードで ESXi インストーラがインストールまたはアップグレード スクリプト（キックスタート ファイル）にアクセスできる場所は、いくつかあります。

インストールまたはアップグレード スクリプトの保存場所として、次のメディアおよび場所を使用できます。

- CD/DVD。カスタムのインストールまたはアップグレードスクリプトを含む、インストーラ ISO イメージの作成を参照してください。
- USB フラッシュ ドライブ。USB フラッシュ ドライブを作成して、ESXi インストール スクリプトまたはアップグレード スクリプトを保存するを参照してください。
- 次のプロトコルでアクセス可能なネットワークの場所。NFS、HTTP、HTTPS、FTP

インストールまたはアップグレード スクリプトのパス

インストールまたはアップグレード スクリプトのパスを指定できます。

ks=http://XXX.XXX.XXX.XXX/kickstart/KS.CFG は、ESXi のインストール スクリプトのパスです。XXX.XXX.XXX.XXX には、スクリプトが置かれたマシンの IP アドレスを指定します。スクリプトを使用した ESXi のインストールを参照してください。

対話型インストールの場合にインストール スクリプトを起動するには、ks= オプションを手動で入力します。インストール スクリプトまたはアップグレード スクリプトを実行するための起動オプションの入力を参照してください。

インストール スクリプトおよびアップグレード スクリプトのコマンド

デフォルトのインストールまたはアップグレード スクリプトを変更する場合や、独自のスクリプトを作成する場合には、サポートされているコマンドを使用します。インストーラの起動時に起動コマンドで指定するインストール スクリプトでは、サポートされているコマンドを使用します。

ESXi をインストールまたはアップグレードするディスクを判断するには、インストール スクリプトに install または upgrade または installorupgrade のいずれかのコマンドが必要です。install コマンドでは、ほかのパーティションが作成されたあとに使用可能なすべての容量を占有する VMFS データストアを含む、デフォルトのパーティションが作成されます。

vSphere 8.0 では、システムにサポートされているデータ処理ユニット (DPU) がある場合、ホスト上の ESXi とともに、DPU での ESXi のインストール、再インストール、またはアップグレードを必ず検討してください。DPU での ESXi 更新とアップグレードはインタラクティブ方式またはスクリプト方式でサポートされていません。vSphere Lifecycle Manager のみが使用可能です。

注: ESX-OSData パーティションを格納するために SD および USB デバイスを使用することは推奨されません。SD デバイスおよび USB デバイスは、ブート バンク パーティション、boot-bank 0、および boot-bank 1 の作成にのみ使用できます。また、ESX-OSData パーティションをインストールする最小 32 GB のパーティション ディスクを提供できます。このようなディスクを定義するには、install コマンドでパラメータ `systemDisk` を使用します。

accepteula または vmaccepteula (必須)

ESXi の使用許諾契約書に同意します。

clearpart (任意)

ディスク上の既存のパーティションをすべてクリアします。install コマンドを指定する必要があります。既存のスクリプトの clearpart コマンドを編集するときには注意してください。

<code>--drives=</code>	指定したドライブ上のパーティションを削除します。
<code>--alldrives</code>	<code>--drives=</code> の要件を無視し、すべてのドライブのパーティションの削除を可能にします。
<code>--ignoredrives=</code>	指定するドライブ以外のすべてのドライブのパーティションを削除します。--drives= フラグまたは --alldrives フラグを指定していない場合に必要です。
<code>--overwritevmfs</code>	指定したドライブ上の VMFS パーティションの上書きを可能にします。デフォルトでは、VMFS パーティションの上書きは許可されません。
<code>--firstdisk=</code> <code>disk-type1</code> <code>[disk-type2,...]</code>	<p>注: vSphere システムのバージョンが 8.0 Update 3 より前で、DPU がある場合は、次のように PCI スロットを指定します：<code>::install --firstdisk --overwritevmfs --dpupcislots=<PCIeSlotID></code>。バージョン 8.0 Update 3 以降のシステムでは、dpupcislots パラメータは廃止されました。</p>

最初に検出された、対象となるディスクをパーティショニングします。デフォルトでは、対象となるディスクは次の順序で設定されます。

- 1 ローカルに接続されたストレージ (local)
- 2 ネットワーク ストレージ (remote)

引数に添付されたコンマ区切りのリストを使用して、ディスクの順序を変更できます。フィルタ リストを指定すると、デフォルト設定が上書きされます。フィルタ (ESXi がインストールされる最初のディスクの esx、モデルおよびベンダー情報、VMkernel デバイス ドライバの名前など) を組み合わせて、特定のディスクを指定できます。たとえば、モデル名が ST3120814A のディスク、および通常のローカ

ル ディスクではなく mptsas ドライバを使用するディスクを指定する場合、引数は `--firstdisk=ST3120814A,mptsas,local` になります。localesx は ESXi イメージを含むローカル ストレージに対して、remoteesx は ESXi イメージを含むリモート ストレージに対して使用できます。

dryrun (任意)

インストール スクリプトを解析し、確認します。インストールは実行されません。

インストール

これが新規インストールであることを指定します。ESXi をインストールまたはアップグレードするディスクを判断するには、install、upgrade、または installorupgrade コマンドが必要です。

`--disk=` or `--drive=` パーティショニングするディスクを指定します。コマンド `--disk=diskname` では、*diskname* に、ESXi でのディスク名またはディスク ファイルシステムのフルパスを指定できます。例は以下のとおりです。

- ディスク名：`--disk=naa.6d09466044143600247aee55ca2a6405` または
 - デバイス パス：`--disk=/vmfs/devices/disks/mpx.vmhba1:C0:T0:L0`
- 使用できるディスク名形式については、[ディスク デバイス名](#)を参照してください。

`--firstdisk=`
disk-type1,
[disk-type2,...]

注： vSphere システムのバージョンが 8.0 Update 3 より前で、DPU がある場合は、次のように PCI スロットを指定します：`::install --firstdisk --overwritevmfs --dpupcislots=<PCIeSlotID>`。バージョン 8.0 Update 3 以降のシステムでは、`dpupcislots` パラメータは廃止されました。

最初に検出された、対象となるディスクをパーティショニングします。デフォルトでは、対象となるディスクは次の順序で設定されます。

- 1 ローカルに接続されたストレージ (local)
- 2 ネットワーク ストレージ (remote)

引数に添付されたコンマ区切りのリストを使用して、ディスクの順序を変更できます。フィルタ リストを指定すると、デフォルト設定が上書きされます。ESX がインストールされた最初のディスクを指定する `esx`、モデルおよびベンダー情報、VMkernel デバイス ドライバの名前など、フィルタを組み合わせて特定のディスクを指定することができます。たとえば、モデル名が ST3120814A のディスク、および通常のローカル ディスクではなく mptsas ドライバを使用するディスクを指定する場合、引数は `--firstdisk=ST3120814A,mptsas,local` になります。localesx は ESXi イメージを含むローカル ストレージに対して、remoteesx は ESXi イメージを含むリモート ストレージに対して使用できます。

`--ignoressd` パーティショニングの対象からソリッド ステート ディスクを除外します。このオプションは、install コマンドおよび `--firstdisk` オプションとともに使用できます。このオプションは `--firstdisk` オプションよりも優先されます。また、

upgrade コマンドおよび `installorupgrade` コマンドと、`--drive` オプションまたは `--disk` オプションと同時に使用しても無効になります。自動パーティション化の際に SSD のフォーマットを避けるための詳細については、『vSphere のストレージ』ドキュメントを参照してください。

--overwritevsan

vSAN ディスク グループ内のディスク (SSD または HDD (磁気)) に ESXi をインストールする場合は、`--overwritevsan` オプションを使用する必要があります。選択したディスクに vSAN パーティションがない状態でこのオプションを使用した場合、インストールは失敗します。vSAN ディスク グループ内のディスクに ESXi をインストールする場合、結果は選択したディスクによって異なります。

- SSD を選択した場合は、同じディスク グループ内の SSD および基盤となるすべての HDD がワイブされます。
- HDD を選択した場合は、ディスク グループのサイズが 2 より大きくなり、選択した HDD のみワイブされます。
- HDD ディスクを選択した場合は、ディスク グループのサイズが 2 以下になり、SSD および選択した HDD がワイブされます。

vSAN ディスク グループの管理の詳細については、『vSphere のストレージ』ドキュメントを参照してください。

--overwritevmfs

インストール前に、ディスク上の既存の VMFS データストアを上書きするために必要です。

--preservevmfs

インストール時に、ディスク上の既存の VMFS データストアを保持します。

--novmfsondisk

このディスク上に VMFS パーティションが作成されないようにします。ディスク上に VMFS パーティションがある場合は、`--overwritevmfs` と共に使用する必要があります。

--systemdisk

USB デバイスまたは SD デバイスを使用すると、`systemDisk` では ESX-OSData パーティションをインストールするローカル パーシステント ディスクが指定されます。たとえば、

install --firstdisk = usb --systemDisk=<diskID> です。その結果、ブート バンク パーティションは USB デバイ스에配置され、OSData パーティションは `systemDisk` パラメータで指定されたディスクに配置されます。

--repartitionssystemdisk

USB デバイスまたは SD デバイスを使用しており、`systemDisk` パラメータで指定するローカル ディスクが空でない、またはデータストアが含まれている場合、

repartitionSystemDisk を使用して、パーティション ディスクが使用前に再パーティショニングされていることを確認できます。

注： ローカル パーシステント ディスクが使用できない、またはディスク サイズが 32 GB 未満の場合は警告メッセージが表示されますが、インストールは続行されます。

--
forceunsupportedinstall 廃止されている CPU のインストールをブロックします。

installorupgrade

ESXi をインストールまたはアップグレードするディスクを判断するには、install、upgrade、または installorupgrade コマンドが必要です。

--disk= or --drive= パーティショニングするディスクを指定します。コマンド --disk=diskname では、diskname に、ESXi でのディスク名またはディスク ファイルシステムのフルパスを指定できます。例は以下のとおりです。

- ディスク名：--disk=naa.6d09466044143600247aee55ca2a6405 または
- デバイス パス：--disk=/vmfs/devices/disks/mpx.vmhba1:C0:T0:L0

使用できるディスク名形式については、[ディスク デバイス名](#)を参照してください。

--firstdisk=
disk-type1,
[disk-type2,...]

注： vSphere システムのバージョンが 8.0 Update 3 より前で、DPU がある場合は、次のように PCI スロットを指定します： :install --firstdisk --overwritevmfs --dpupcislots=<PCIeSlotID>。バージョン 8.0 Update 3 以降のシステムでは、dpupcislots パラメータは廃止されました。

最初に検出された、対象となるディスクをパーティショニングします。デフォルトでは、対象となるディスクは次の順序で設定されます。

- 1 ローカルに接続されたストレージ (local)
- 2 ネットワーク ストレージ (remote)

引数に添付されたコンマ区切りのリストを使用して、ディスクの順序を変更できます。フィルタ リストを指定すると、デフォルト設定が上書きされます。ESX がインストールされた最初のディスクを指定する esx、モデルおよびベンダー情報、VMkernel デバイス ドライバの名前など、フィルタを組み合わせて特定のディスクを指定することができます。たとえば、モデル名が ST3120814A のディスク、および通常のローカル ディスクではなく mptsas ドライバを使用するディスクを指定する場合、引数は--firstdisk=ST3120814A,mptsas,local になります。localesex は ESXi イメージを含むローカル ストレージに対して、remoteesx は ESXi イメージを含むリモート ストレージに対して使用できます。

--overwritevsan

vSAN ディスク グループ内のディスク (SSD または HDD (磁気)) に ESXi をインストールする場合は、`--overwritevsan` オプションを使用する必要があります。選択したディスクに vSAN パーティションがない状態でこのオプションを使用した場合、インストールは失敗します。vSAN ディスク グループ内のディスクに ESXi をインストールする場合、結果は選択したディスクによって異なります。

- SSD を選択した場合は、同じディスク グループ内の SSD および基盤となるすべての HDD がワイプされます。
- HDD を選択した場合は、ディスク グループのサイズが 2 より大きくなり、選択した HDD のみワイプされます。
- HDD ディスクを選択した場合は、ディスク グループのサイズが 2 以下になり、SSD および選択した HDD がワイプされます。

vSAN ディスク グループの管理の詳細については、『vSphere のストレージ』ドキュメントを参照してください。

--overwritevmfs

ディスク上に VMFS パーティションが存在するが、ESX または ESXi のインストールは存在しない場合に、ESXi をインストールします。このオプションを指定しないと、ディスク上に VMFS パーティションがあっても、ESX または ESXi がない場合は、インストーラの実行に失敗します。

keyboard (任意)

システムのキーボード タイプを設定します。

keyboardType

選択したキーボード タイプのキーボード マップを指定します。*keyboardType* は次のいずれかのタイプである必要があります。

- ベルギー語
- ポルトガル語 (ブラジル)
- クロアチア語
- チェコスロバキア語
- デンマーク語
- エストニア語
- フィンランド語
- フランス語
- ドイツ語
- ギリシャ語
- アイスランド語
- イタリア語
- 日本語

- ラテン アメリカ言語
- ノルウェー語
- ポーランド語
- ポルトガル語
- ロシア語
- スロベニア語
- スペイン語
- スウェーデン語
- フランス語 (スイス)
- ドイツ語 (スイス)
- トルコ語
- ウクライナ語
- 英語 (英国)
- 米国デフォルト
- 米国 Dvorak

serialnum または vmserialnum (任意)

このコマンドは ESXi バージョン 5.1 以降でサポートされています。ライセンスを構成します。指定しなかった場合、ESXi は評価モードでインストールされます。

`--esx=<license-key>` 使用する vSphere のライセンス キーを指定します。形式は 5 文字のグループを連結したものです (XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX)。

network (任意)

システムのネットワーク アドレスを指定します。

`--bootproto=[dhcp|static]` ネットワーク設定を DHCP から取得するか手動で設定するかを指定します。

`--device=` ネットワーク カードの MAC アドレスを指定するか、vmnicNN 形式で vmnic0 のようにデバイス名を指定します。このオプションは、仮想スイッチのアップリンク デバイスを参照します。

`--ip=` インストールするマシンの IP アドレスを xxx.xxx.xxx.xxx 形式で設定します。--bootproto=static オプションを指定する場合に必要です。それ以外の場合は無視されます。

`--gateway=` デフォルトのゲートウェイを IP アドレスで、xxx.xxx.xxx.xxx 形式で指定します。--bootproto=static オプションと同時に使用します。

<code>--nameserver=</code>	プライマリ ネーム サーバを IP アドレスとして指定します。 <code>--bootproto=static</code> オプションと同時に使用します。DNS を使用しない場合はこのオプションは省略します。 <code>--nameserver</code> オプションでは、2 つの IP アドレスを指定することもできます。 例: <code>--nameserver="10.126.87.104[,10.126.87.120]"</code>
<code>--netmask=</code>	インストールしたシステムのサブネット マスクを <code>255.xxx.xxx.xxx</code> 形式で指定します。 <code>--bootproto=static</code> オプションと同時に使用します。
<code>--hostname=</code>	インストールしたシステムのホスト名を指定します。
<code>--vlanid= <i>vlanid</i></code>	システムの VLAN を指定します。 <code>--bootproto=dhcp</code> オプションまたは <code>--bootproto=static</code> オプションと同時に使用します。1 ~ 4096 の整数を指定します。
<code>--addvmportgroup=(0 1)</code>	仮想マシンで使用される仮想マシン ネットワークのポート グループを追加するかどうかを指定します。デフォルト値は 1 です。

paranoid (任意)

警告メッセージでインストールが中断されます。このコマンドを省略した場合、警告メッセージが記録されます。

part または partition (任意)

追加の VMFS データストアをシステムに作成します。作成できるデータストアは、各ディスクにつき 1 つだけです。 `install` コマンドと同じディスクでは使用できません。ディスクごとに 1 つのパーティションしか指定できず、指定するのは VMFS パーティションでなければなりません。

<code>datastore name</code>	パーティションのマウント先を指定します。
<code>--ondisk= or --ondrive=</code>	パーティションを作成するディスクまたはドライブを指定します。
<code>--onfirstdisk=</code> <code>disk-type1,</code> <code>[disk-type2,...]</code>	注: vSphere システムのバージョンが 8.0 Update 3 より前で、DPU がある場合は、次のように PCI スロットを指定します: <code>::install --firstdisk --overwritevmfs --dpupcislots=<PCIeSlotID></code> 。バージョン 8.0 Update 3 以降のシステムでは、 <code>dpupcislots</code> パラメータは廃止されました。

最初に検出された、対象となるディスクをパーティショニングします。デフォルトでは、対象となるディスクは次の順序で設定されます。

- 1 ローカルに接続されたストレージ (local)
- 2 ネットワーク ストレージ (remote)

引数に添付されたコンマ区切りのリストを使用して、ディスクの順序を変更できます。フィルタ リストを指定すると、デフォルト設定が上書きされます。ESX がインストールされた最初のディスクを指定する `esx`、モデルおよびベンダー情報、VMkernel デバイス ドライバの名前など、フィルタを組み合わせて特定のディスクを指定することができます。たとえば、モデル名が ST3120814A のディスク、お

よび通常のローカル ディスクではなく mptsas ドライバを使用するディスクを指定する場合、引数は `--onfirstdisk=ST3120814A,mptsas,local` になります。 `localesx` は ESXi イメージを含むローカル ストレージに対して、 `remoteesx` は ESXi イメージを含むリモート ストレージに対して使用できます。

reboot (任意)

スクリプトを使用したインストールが完了したあと、マシンを再起動します。

`<--noeject>` CD は、インストール後に取り外されません。

rootpw (必須)

システムの root パスワードを設定します。

`--iscrypted` パスワードが暗号化されていることを示します。

`password` パスワード値を指定します。

アップグレード

ESXi をインストールまたはアップグレードするディスクを判断するには、 `install`、 `upgrade`、または `installorupgrade` コマンドが必要です。

`--disk=` or `--drive=` パーティショニングするディスクを指定します。コマンド `--disk=diskname` では、 `diskname` に、ESXi でのディスク名またはディスク ファイルシステムのフルパスを指定できます。例は以下のとおりです。

- ディスク名： `--disk=naa.6d09466044143600247aee55ca2a6405` または
 - デバイス パス： `--disk=/vmfs/devices/disks/mpx.vmhba1:C0:T0:L0`
- 使用できるディスク名形式については、[ディスク デバイス名](#)を参照してください。

`--firstdisk=`
`disk-type1,`
`[disk-type2,...]` 最初に検出された、対象となるディスクをパーティショニングします。デフォルトでは、対象となるディスクは次の順序で設定されます。

- 1 ローカルに接続されたストレージ (local)
- 2 ネットワーク ストレージ (remote)

引数に添付されたコンマ区切りのリストを使用して、ディスクの順序を変更できます。フィルタ リストを指定すると、デフォルト設定が上書きされます。ESX がインストールされた最初のディスクを指定する `esx`、モデルおよびベンダー情報、VMkernel デバイス ドライバの名前など、フィルタを組み合わせて特定のディスクを指定することができます。たとえば、モデル名が ST3120814A のディスク、および通常のローカル ディスクではなく mptsas ドライバを使用するディスクを指定する場合、引数は `--firstdisk=ST3120814A,mptsas,local` になります。 `localesx` は ESXi イメージを含むローカル ストレージに対して、 `remoteesx` は ESXi イメージを含むリモート ストレージに対して使用できます。

%include または include (任意)

解析する別のインストール スクリプトを指定します。このコマンドはマルチライン コマンドと同様に処理されますが、1 つの引数だけを使用します。

filename 例: %include part.cfg

%pre (任意)

キックスタート構成が評価される前に実行するスクリプトを指定します。たとえば、含めるキックスタート ファイル用にファイルを生成する際に使用できます。

--interpreter 使用するインタプリタを指定します。デフォルトは busybox です。

=[python|busybox]

%post (任意)

パッケージのインストールが完了したあとに、指定したスクリプトを実行します。複数の %post セクションを指定した場合は、インストール スクリプト内に出現する順序で実行されます。

--interpreter 使用するインタプリタを指定します。デフォルトは busybox です。

=[python|busybox]

--timeout=secs スクリプト実行時のタイムアウトを指定します。タイムアウトになったときにスクリプトが終了していなかった場合、スクリプトは強制的に停止されます。

--ignorefailure true を指定した場合、%post スクリプトがエラーで停止しても、インストールは正常に終了したと見なされます。

=[true|false]

%firstboot

最初の起動中のみ実行する init スクリプトを作成します。このスクリプトは、次回以降の起動では無効になりません。複数の %firstboot セクションを指定した場合は、キックスタート ファイル内に出現する順序で実行されません。

注: %firstboot スクリプトのセマンティックの確認は、システムが最初に起動するまで実行できません。

%firstboot スクリプトに、インストールが完了したあとでなければ分からない、深刻なエラーが含まれている場合があります。

重要: ESXi ホスト上でセキュア ブートが有効になっていると、%firstboot スクリプトは実行されません。

--interpreter 使用するインタプリタを指定します。デフォルトは busybox です。

=[python|busybox]

注: %firstboot スクリプトのセマンティックの確認は、システムが最初に起動するまで実行できません。スクリプトにエラーが含まれていても、インストールが完了するまで検出されません。

ディスク デバイス名

インストール スクリプトのコマンド `install`、`upgrade`、および `installorupgrade` では、ディスク デバイス名を指定する必要があります。

表 4-15. ディスク デバイス名

フォーマット	例	説明
NAA	naa.6d09466044143600247aee55ca2a6405	SCSI INQUIRY 識別子
EUI	eui.3966623838646463	SCSI INQUIRY 識別子
T10	t10.SanDisk00Cruzer_Blade000000004C530001171118101244	SCSI INQUIRY 識別子
VML	vml.00025261	レガシー VMkernel 識別子
MPX	mpx.vmhba0:CO:TO:LO	バスベースの識別子

ストレージ デバイス名の詳細については、『vSphere Storage』ドキュメントの「ストレージ デバイスの名前と識別子」を参照してください。

スクリプトによるインストール中の外部エントロピー ソースの構成

ESXi 8.0 Update 1 以降でスクリプトによるインストールを行う場合は、キックスタート ファイル内で外部エントロピー ソースを構成できます。

スクリプトによるインストール方法を使用することで、安全性の高い環境において、ハードウェア セキュリティ モジュール (HSM) などの外部エントロピー ソースのエントロピーを使用し、BSI コモン クライテリア、EAL4、NIST FIPS CMVP などの標準に適合するように ESXi を構成することができます。

ESXi 8.0 Update 1 には `entropyd` というエントロピー デーモンが導入されています。このデーモンを使用して vAPI エンドポイントを作成することにより、REST API はエントロピー データのクエリや、ESXi ホストへのエントロピー データの送信を行うことが可能になります。このエントロピー デーモンには、`in-memory-cache size`、`in-storage-cache size`、`in-memory-low watermark`、`entropy-lost-timeout` などのいくつかの構成可能なパラメータがあります。エントロピー ソース (内部および外部) は、エントロピーミキサー モジュールで収集されます。エントロピー デーモンは、ミキサーからカーネルのエントロピー プールにエントロピー ソースを転送します。

外部エントロピー ソースを選択する必要がない場合は、既存のスクリプトの変更は不要です。

外部エントロピー ソースを構成できるのは、新規インストールの場合のみです。エントロピー デーモンは、初回起動時のインストール中に渡されたエントロピー データを使用します。

注： 対話型インストール、クラスタ イメージ、Auto Deploy などの他の方法を使用して、外部エントロピー ソースを構成することはできません。既存の ESXi ホストでこの機能を有効にすることはできません。

キックスタート ファイルを更新する前に、HSM などの外部ソースから合法的なバイナリ エントロピー データを取得し、512 KB ~ 10 MB のサイズのファイル (`entropy_data.dat` など) として保存する必要があります。次に、`entropy_data` ファイルの内容を有効な base64 形式でエンコードして、`entropy_data.b64` などの別のファイルを作成します。

キックスタート ファイルで、entropy コマンドの一部として次の新しいパラメータを指定します。

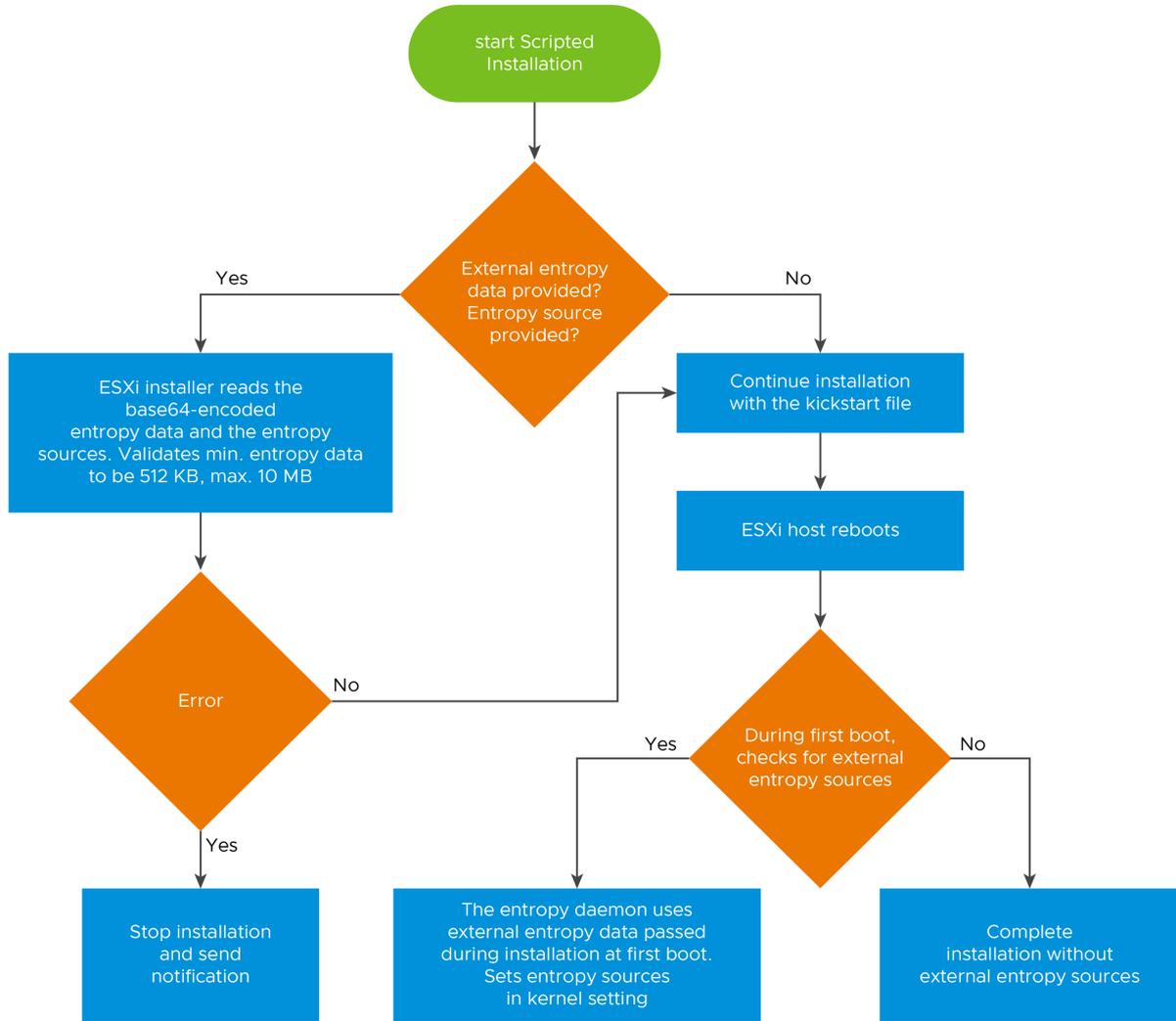
- data : 外部ソースから取得された、base64 でエンコードされたエントロピー データ。
- sources : このビットマスク値 (0 = **default**、1 = **interrupts**、2 = **RDRAND**、4 = **RDSEED**、8 = **entropyd**) は、インストール時にカーネルで設定されます。**RDSEED** がサポートされている場合、デフォルトは FIPS コンプライアンスです。それ以外の場合、デフォルトはすべてのエントロピー ソース (**entropyd** を除く) です。EAL4 に準拠するために必要な値は 8 のみですが、10 進数の範囲 (0-15) 内の他のソースを使用することができます。ソース値は、インストールの間、カーネル設定として保持されます。

注： 外部エントロピー ソースのみを使用して ESXi ホストを構成する場合 (sources 値が **8** に設定されている場合) は、エントロピー API を使用して外部エントロピーをホストに提供し続ける必要があります。ホストで外部エントロピーが枯渇すると、ホストが応答しなくなり、このような状況からホストをリカバリするために強制的な再起動または再インストールが必要になることがあります。

エントロピー パラメータを使用したキックスタートの例：

```
vmaccepteula
rootpw xxxxxxxx
entropy --sources=8 --data=xxxxxxxx/xx/xxxxx/xx/xxxx...
install --firstdisk --overwritevmfs
network --bootproto=dhcp
```

図 4-4. 外部エントロピー ソースを追加するためのスクリプトによるインストールのワークフロー



インストールが完了したら、次の ESXCLI コマンドを使用して ESXi ホストにログインし、シェルからエントロピーデーモンのパラメータをいくつか定義できます。

ESXCLI コマンド

1. コマンドを取得する

##	esxcli system entropyd get	コマンドの説明
1	<no argument>	すべての entropyd パラメータの現在の構成値とデフォルト値を取得します
2	--default-values	デフォルト値を取得します

2. コマンドを設定する

##	esxcli system entropyd set	コマンドの説明
1	--help	[esxcli system entropyd set] コマンドの詳細と引数を出力します
2	--reset=all --reset=memory-cache-size --reset=memory-cache-low-watermark --reset=storage-cache-size --reset=external-entropy-lost-timeout	特定のパラメータまたはすべてのパラメータをデフォルト値にリセットします。
3	--memory-cache-size=<value in KiB>	メモリ キャッシュ サイズを KiB 単位で設定します。
4	--memory-cache-low-watermark=<value in %>	メモリ キャッシュの下位のウォーター マークをパーセンテージで設定します。
5	--storage-cache-size=<value in KiB>	ストレージ キャッシュ サイズを KiB 単位で設定します。
6	--external-entropy-lost-timeout=<value in seconds>	外部エントロピーで失われたタイムアウトを秒単位で設定します。

例 :

```
$ esxcli system entropyd set --external-entropy-lost-timeout=70 --memory-cache-low-watermark=30 --memory-cache-size=612 --storage-cache-size=5096
$ esxcli system entropyd get
External Entropy Lost Timeout Seconds: 70
Memory Cache Low Watermark Percentage: 30
Memory Cache Size Kibibytes: 612
Storage Cache Size Kibibytes: 5096

$ esxcli system entropyd get --default-values
External Entropy Lost Timeout Seconds: 60
Memory Cache Low Watermark Percentage: 20
Memory Cache Size Kibibytes: 512
Storage Cache Size Kibibytes: 4096
```

ESXi インストーラによるインストール中にカーネル設定のエントロピー ソース値は更新され、インストール後も ConfigStore に保持されるようになります。この変更は、ESXi ホストのセキュア ブートには影響しません。

詳細については、『vSphere セキュリティ』および『vSphere Automation SDK プログラミング ガイド』を参照してください。

スクリプトを使用した、CD または DVD からの ESXi のインストールまたはアップグレード

インストールまたはアップグレードのオプションを指定するスクリプトを使用して、CD-ROM または DVD-ROM ドライブから ESXi をインストールまたはアップグレードすることができます。

ホスト起動時に起動オプションを入力して、インストール スクリプトまたはアップグレード スクリプトを起動できます。インストール スクリプトが含まれているインストーラ ISO イメージを作成することもできます。インストーラ ISO イメージを使用すると、インストーラ ISO イメージを起動するときに、スクリプトを使用した無人インストールを実行できます。[カスタムのインストールまたはアップグレードスクリプトを含む、インストーラ ISO イメージの作成](#) を参照してください。

前提条件

スクリプトを使用したインストールまたはアップグレードを実行する前に、次の前提条件を満たしているかどうかを確認します。

- インストールまたはアップグレードを実行するシステムがハードウェア要件を満たしていること。[ESXi のハードウェア要件](#) を参照してください。
- インストール CD または DVD に ESXi インストーラ ISO があること。[ESXi インストーラ ISO イメージのダウンロードおよび CD or DVD への書き込み](#) を参照してください。
- デフォルトのインストール/アップグレード スクリプト (`ks.cfg`) またはカスタムのインストール/アップグレード スクリプトがシステムにアクセスできること。[スクリプトを使用した ESXi のインストール](#) を参照してください。
- スクリプトを使用したインストールまたはアップグレードを実行するための起動コマンドを選択済みであること。[インストール スクリプトまたはアップグレード スクリプトを実行するための起動オプションの入力](#) を参照してください。起動コマンドの完全なリストについては、[起動オプション](#) を参照してください。

手順

- 1 ローカルの CD-ROM または DVD-ROM ドライブから ESXi のインストーラを起動します。
- 2 ESXi インストーラのウィンドウが表示されたら、Shift + O を押して起動オプションを編集します。



- 3 デフォルトのインストール/アップグレード スクリプト、または作成したインストール/アップグレード スクリプト ファイルを呼び出す起動オプションを入力します。

起動オプションの形式は `ks=` です。

- 4 [Enter] を押します。

結果

指定したオプションを使用して、インストール、アップグレード、または移行が実行されます。

スクリプトを使用した、USB スティックからの ESXi のインストールまたはアップグレード

インストールまたはアップグレードのオプションを指定するスクリプトを使用して、USB フラッシュ ドライブから ESXi をインストールまたはアップグレードすることができます。

サポートされている起動オプションは [起動オプション](#) に記載されています。

前提条件

スクリプトを使用したインストールまたはアップグレードを実行する前に、次の前提条件を満たしているかどうかを確認します。

- ESXi のインストールまたはアップグレードを実行するシステムが、インストールまたはアップグレード用のハードウェア要件を満たしていること。ESXi のハードウェア要件を参照してください。
- 起動可能な USB フラッシュ ドライブに ESXi インストーラ ISO があること。USB フラッシュ ドライブのフォーマットによる ESXi インストールまたはアップグレードの起動を参照してください。
- デフォルトのインストール/アップグレード スクリプト (ks.cfg) またはカスタムのインストール/アップグレード スクリプトがシステムにアクセスできること。スクリプトを使用した ESXi のインストールを参照してください。
- スクリプトを使用したインストール、アップグレード、または移行を実行するための起動オプションを選択済みであること。インストール スクリプトまたはアップグレード スクリプトを実行するための起動オプションの入力を参照してください。

手順

- 1 USB フラッシュ ドライブから ESXi のインストーラを起動します。
- 2 ESXi インストーラのウィンドウが表示されたら、Shift + O を押して起動オプションを編集します。



- 3 デフォルトのインストール/アップグレード スクリプト、または作成したインストール/アップグレード スクリプト ファイルを呼び出す起動オプションを入力します。

起動オプションの形式は ks= です。

- 4 [Enter] を押します。

結果

指定したオプションを使用して、インストール、アップグレード、または移行が実行されます。

スクリプトを使用したインストールまたはアップグレードのための ESXi インストーラのネットワーク ブート

ESXi 8.0 には、ネットワーク経由でインストーラを起動するための、およびインストール スクリプトまたはアップグレード スクリプトを使用するためのオプションが数多くあります。

- ネットワーク インフラストラクチャの設定については、[ESXi インストーラのネットワーク ブート](#)を参照してください。
- インストール スクリプトの作成および配置については、[スクリプトを使用した ESXi のインストール](#) を参照してください。
- ESXi のインストーラをネットワーク ブートレインストール スクリプトを使用する具体的な手順については、次のトピックを参照してください。
 - [ネイティブ UEFI HTTP を使用した ESXi インストーラの起動](#)
 - [iPXE および HTTP を使用した ESXi インストーラの起動](#)
 - [PXE および TFTP を使用した ESXi インストーラの起動](#)
- vSphere Auto Deploy を使用して PXE ブートによるスクリプトを使用したインストールを実行する方法については、[vSphere Auto Deploy を使用した ESXi のインストール](#) を参照してください。

ネットワーク デバイスから ESXi ホストを起動する方法

ESXi インストーラのネットワーク ブート

ホストでレガシー BIOS または UEFI が使用されている場合、Preboot Execution Environment (PXE) を利用してネットワーク デバイスから ESXi ホストを起動できます。

または、ESXi ホストでネイティブ UEFI HTTP がサポートされている場合、HTTP (ハイパー テキスト転送プロトコル) を使用してネットワーク デバイスからホストを起動できます。ESXi は、フラッシュ メモリまたはローカル ハード ドライブへのインストールに使用する ISO 形式で提供されます。ネットワーク インターフェイスを介してファイルを抽出し、起動できます。

PXE では、DHCP (Dynamic Host Configuration Protocol) と TFTP (File Transfer Protocol) を使用して、ネットワーク経由でオペレーティング システムを起動します。

PXE ブートを行うには、いくつかのネットワーク インフラストラクチャと、PXE 対応のネットワーク アダプタを搭載したマシンが必要です。ESXi を実行可能なマシンの多くが、PXE ブートが可能なネットワーク アダプタを搭載しています。

ネイティブ UEFI HTTP では、ネットワーク経由での起動に DHCP と HTTP が使用されます。UEFI HTTP ブートを利用するには、ネットワーク インフラストラクチャ、ESXi ホスト上の UEFI ファームウェアの HTTP ブート機能を備えたバージョン、および UEFI ネットワークをサポートするネットワーク アダプタが必要です。

HTTP を使用した起動は、TFTP を使用した起動よりも速度と信頼性が優れています。これは、組み込みのストリーミングや消失したパケットのリカバリなど、HTTP の基盤となる TCP プロトコルの機能によるものです。ESXi ホストがネイティブ UEFI HTTP をサポートしていない場合は、起動プロセスに iPXE HTTP を使用できます。

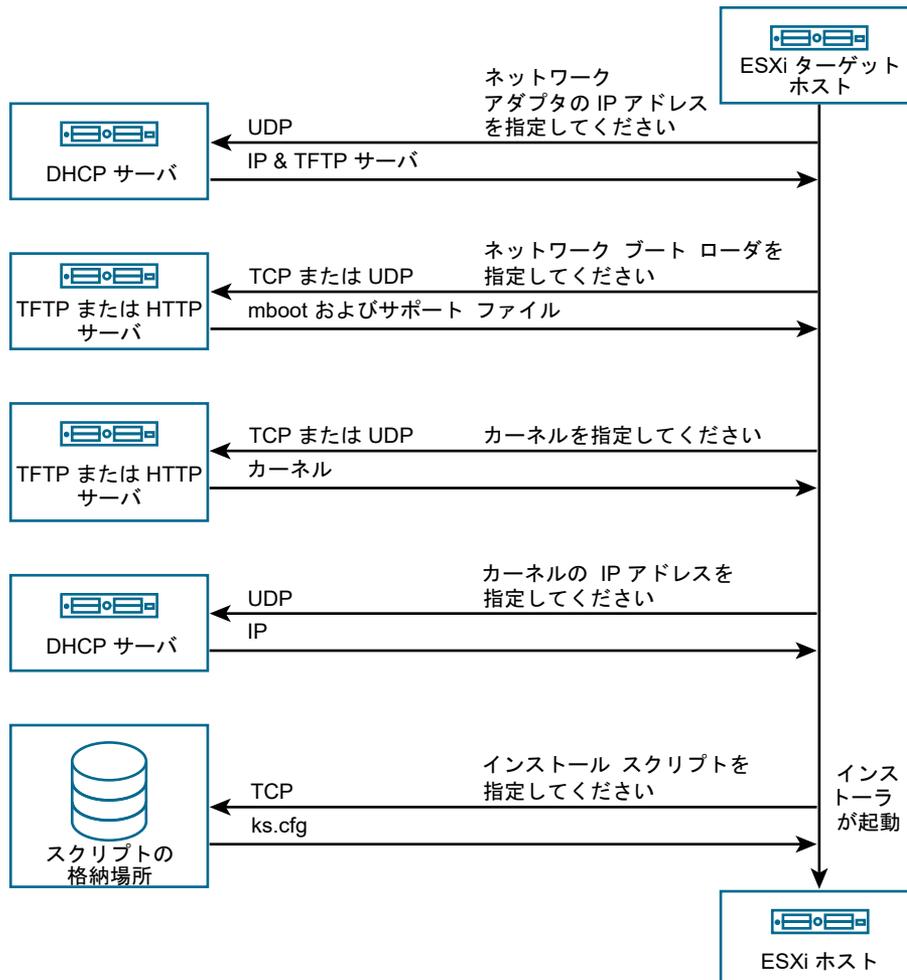
注： レガシー BIOS ファームウェアを使用するネットワーク ブートは IPv4 でのみ実行できます。UEFI BIOS ファームウェアを使用したネットワーク起動は、IPv4 または IPv6 で実行できます。

ネットワーク ブートのインストール プロセスの概要

ESXi ホストをネットワーク インターフェイスから起動できます。ネットワーク ブート プロセスは、ターゲット ホストがレガシー BIOS または UEFI ファームウェアのどちらを使用するか、およびブート プロセスが PXE TFTP、iPXE HTTP、UEFI HTTP のどれを使用するかによって異なります。

ターゲット ホストを起動すると、ターゲット ホストは、環境内の異なるサーバと通信を行い、ネットワーク アダプタ、ブート ローダー、カーネル、カーネルの IP アドレス、そして最後にインストール スクリプトを取得します。すべてのコンポーネントの準備が完了すると、次の図に示すようにインストールが開始します。

図 4-5. PXE ブートのインストール プロセスの概要



ESXi ホストと他のサーバ間のインタラクティブな作業は次のように行われます。

- 1 ユーザーがターゲットの ESXi ホストを起動します。
- 2 ターゲットの ESXi ホストが DHCP リクエストを送信します。
- 3 DHCP サーバは、IP アドレス情報、TFTP または HTTP サーバの場所、および初期ネットワーク ブートローダーのファイル名または URL を返します。
- 4 ESXi ホストが TFTP または HTTP サーバに接続し、DHCP サーバが指定したファイル名または URL を要求します。
- 5 TFTP または HTTP サーバがネットワーク ブート ローダーを送信し、ESXi ホストがそれを実行します。初期ブート ローダーは、サーバから追加のブート ローダー コンポーネントをロードする場合があります。
- 6 ブート ローダーは、TFTP または HTTP サーバ上の構成ファイルを検索し、構成ファイルで指定されたカーネルおよび他の ESXi コンポーネントをダウンロードし、ESXi ホストでカーネルを起動します。
- 7 インストーラは、インタラクティブに実行するか、構成ファイルで指定された kickstart スクリプトを使用して実行します。

ネットワーク ブートの背景情報

ネットワーク ブート プロセスを理解しておく、トラブルシューティング時に役立ちます。

TFTP サーバ

TFTP (Trivial File Transfer Protocol) は FTP サービスに似ており、通常は、システムのネットワーク起動や、ルータなどのネットワーク デバイスへのファームウェアの読み込みにのみ使用します。TFTP は Linux および Windows で使用できます。

- ほとんどの Linux ディストリビューションには、tftp-hpa サーバのコピーが含まれています。サポートされているソリューションが必要な場合は、希望するベンダーからサポートされている TFTP サーバを購入してください。TFTP サーバは、VMware Marketplace にあるアプライアンス パッケージのいずれかで取得することもできます。
- TFTP サーバを Microsoft Windows ホストで実行する場合は、tftpd32 バージョン 2.11 以降を使用してください。 <http://tftpd32.jounin.net/> を参照してください。

SYSLINUX と PXELINUX

レガシーの BIOS 環境で PXE を使用している場合は、さまざまな起動環境について理解する必要があります。

- SYSLINUX は、レガシー BIOS ファームウェアを実行するマシン用のオープンソースの起動環境です。BIOS システムに対する ESXi ブート ローダー `mboot.c32` は、SYSLINUX プラグインとして実行されます。ディスク、ISO イメージ、ネットワークなどの複数の種類のメディアから起動するように SYSLINUX を構成できます。SYSLINUX パッケージは、 <http://www.kernel.org/pub/linux/utils/boot/syslinux/> に格納されています。

- PXELINUX は、PXE 標準に従って TFTP サーバから起動するための SYSLINUX 構成です。PXELINUX を使用して ESXi インストーラを起動する場合、pxelinux.0 バイナリ ファイル、mboot.c32、構成ファイル、カーネル、およびその他のファイルは TFTP で転送されます。

注： VMware では、SYSLINUX バージョン 3.86 で動作する mboot.c32 プラグインを作成し、そのバージョンでの PXE ブートのみをテストしています。他のバージョンは互換性がない場合があります。VMware vSphere Hypervisor 用のオープン ソース公開パッケージには、SYSLINUX バージョン 3.86 のバグ修正が含まれています。

iPXE

iPXE は、HTTP の実装を提供するオープン ソース ソフトウェアです。このソフトウェアを使用して初期起動を実行できます。詳細については、『<https://ipxe.org/>』を参照してください。

VMware には、Auto Deploy の一部として iPXE のビルドが含まれています。このビルドのソース ツリーは VMware vCenter Server 用のオープン ソース公開パッケージで使用できます。

UEFI PXE と UEFI HTTP

ほとんどの UEFI ファームウェアでは、TFTP サーバからの起動を許可する PXE をネイティブでサポートしています。ファームウェアは、UEFI システムに ESXi ブート ロードャー mboot.efi を直接ロードできます。PXELINUX などの追加のソフトウェアは必要ありません。

一部の UEFI ファームウェアはネイティブ UEFI HTTP の起動をサポートしています。この機能は、UEFI 仕様のバージョン 2.5 で導入されています。このファームウェアは、iPXE などの追加ソフトウェアを使用せずに、HTTP サーバから ESXi ブート ロードャーをロードできます。

注： Apple Macintosh 製品は PXE ブートをサポートしていません。代わりに、Apple 固有のプロトコルを介したネットワーク ブートがサポートされます。

ネットワーク ブートの代替アプローチ

ネットワーク ブートの代わりに、異なるホスト上の異なるソフトウェアを起動することができます。例：

- MAC アドレスまたはその他の基準に応じて、異なるホストに異なる初期ブート ロードャーのファイル名を提供するように DHCP サーバを構成する。DHCP サーバのドキュメントを参照してください。
- 初期ブート ロードャーとして iPXE を使用する。iPXE 構成ファイルは、MAC アドレスまたはその他の基準に応じて次のブート ロードャーを選択するように設定します。

PXELINUX 構成ファイル

レガシー BIOS システムで ESXi インストーラを起動するには、PXELINUX 構成ファイルが必要となります。構成ファイルには、ターゲット ESXi ホストの起動時に表示されるメニューが定義されています。

このセクションでは、PXELINUX 構成ファイルに関する全般的な情報を取り上げます。

構文の詳細については、SYSLINUX の Web サイト (<http://www.syslinux.org/>) を参照してください。

必要なファイル

PXE の構成ファイルには、次のファイルへのパスを含める必要があります。

- mboot.c32 はブート ロードャーです。

- `boot.cfg` はブート ロード構成ファイルです。

『[boot.cfg ファイルについて](#)』を参照してください

PXE 構成ファイルのファイル名

PXE 構成ファイルのファイル名には、次のいずれかのオプションを選択します。

- `01-mac_address_of_target_ESXi_host`。例：01-23-45-67-89-0a-bc
- ターゲット ESXi ホストの IP アドレスの 16 進表記。
- `default`

初期起動ファイル `pxelinux.0` は、次の順序で PXE 構成ファイルのロードを試行します。

- 1 ARP タイプ コード（イーサネットは 01）をプリフィックスとする、ターゲット ESXi ホストの MAC アドレスが試されます。
- 2 これに失敗した場合は、ターゲット ESXi システムの IP アドレスの 16 進表記が試されます。
- 3 最後に、`default` というファイルのロードが試されます。

PXE の構成ファイルの場所

ファイルを TFTP サーバの `/tftpboot/pxelinux.cfg/` に保存します。

たとえば、このファイルは TFTP サーバの `/tftpboot/pxelinux.cfg/01-00-21-5a-ce-40-f6` に保存できます。ターゲット ESXi ホストのネットワーク アダプタの MAC アドレスは `00-21-5a-ce-40-f6` です。

PXE および TFTP を使用した ESXi インストーラの起動

TFTP サーバを使用して ESXi インストーラを PXE ブートすることができます。このプロセスは、UEFI を使用するか、レガシー BIOS から起動するかによって多少異なります。

- レガシー BIOS マシンの場合、すべてのターゲット マシンに対して同じ `pxelinux.0` の初期ブート ロードーを使用して複数の異なるバージョンの ESXi インストーラを起動することができますが、PXELINUX 構成ファイルはターゲット マシンの MAC アドレスに応じて異なる場合があります。
- UEFI マシンの場合、すべてのターゲット マシンに対して同じ `mboot.efi` の初期ブート ロードーを使用して複数の異なるバージョンの ESXi インストーラを起動することができますが、`boot.cfg` ファイルはターゲット マシンの MAC アドレスに応じて異なる場合があります。

前提条件

ほとんどの環境には、UEFI ブートをサポートする ESXi ホストと、レガシーの BIOS のみをサポートするホストが含まれています。このトピックでは、両方のタイプのホストにおける PXE ブートの前提条件と手順について説明します。

環境が次の前提条件を満たしていることを確認します。

- VMware Web サイトからダウンロードした ESXi インストーラの ISO イメージ。
- 使用している ESXi のバージョンでサポートされるハードウェア構成を備えたターゲット ホスト。『[VMware の互換性ガイド](#)』を参照してください。
- ターゲット ESXi ホストに、PXE をサポートするネットワーク アダプタ。

- PXE ブート向けに構成できる DHCP サーバ。DHCP の構成例を参照してください。
- TFTP サーバ。
- TFTP トラフィックを許可するネットワーク セキュリティ ポリシー (UDP ポート 69)。
- レガシー BIOS の場合は、IPv4 ネットワークのみを使用できます。UEFI の PXE ブートの場合は、IPv4 または IPv6 ネットワークを使用できます。
- (オプション) インストール スクリプト (キックスタート ファイル)。
- ほとんどの場合、ネイティブ VLAN を使用します。PXE ブートで使用する VLAN ID を指定する場合は、NIC が VLAN ID 仕様をサポートしていることを確認します。

レガシー BIOS システムの場合は、バージョン 3.86 の SYSLINUX パッケージを取得します。詳細については、『[ネットワーク ブートの背景情報](#)』を参照してください。

手順

- 1 ESXi ホストでレガシー BIOS ファームウェアのみが実行されている場合は、PXELINUX を取得して、構成します。
 - a SYSLINUX バージョン 3.86 を取得し、解凍した pxelinux.0 ファイルを TFTP サーバの最上位の /tftpboot ディレクトリにコピーします。
 - b 次のコード モデルを使用して、PXELINUX 構成ファイルを作成します。

ESXi-8.x.x-XXXXXX は、ESXi インストーラ ファイルが含まれる TFTP サブディレクトリの名前です。

```
DEFAULT install
NOHALT 1
LABEL install
  KERNEL ESXi-8.x.x-XXXXXX/mboot.c32
  APPEND -c ESXi-8.x.x-XXXXXX/boot.cfg
  IPAPPEND 2
```

- c PXELINUX ファイルを TFTP サーバの /tftpboot/pxelinux.cfg ディレクトリに保存します。ファイルには、すべてのホストがデフォルトでこのインストーラを起動するかを指定したファイル名を使用します。

オプション	説明
同じインストーラ	すべてのホストがデフォルトでこの ESXi インストーラを起動するには、ファイルに default という名前を付けます。
異なるインストーラ	このファイルを使用して起動するホストを限定するには、ファイル名にインストール先ホスト マシンの MAC アドレス (01-mac_address_of_target_ESXi_host) を使用します。たとえば 01-23-45-67-89-0a-bc のようになります。

- 2 ESXi ホストで UEFI ファームウェアが実行されている場合は、ESXi インストーラ ISO イメージから TFTP サーバ上の /tftpboot フォルダに efi/boot/bootx64.efi ファイルと efi/boot/crypto64.efi ファイルをコピーします。

- 3 efi/boot/bootx64.efi ファイルの名前を mboot.efi に変更します。

注： 通常、mboot.efi の新しいバージョンは、ESXi の前のバージョンを起動できますが、mboot.efi の前のバージョンは、ESXi の新しいバージョンを起動できない場合があります。異なるホストを構成して異なるバージョンの ESXi インストーラを起動する場合は、最新バージョンの mboot.efi を使用します。

- 4 DHCP サーバを構成します。
- 5 TFTP サーバの最上位 /tftpboot ディレクトリの下にサブディレクトリを作成し、格納される ESXi のバージョンの名前を付けます。たとえば /tftpboot/ESXi-8.x.x-xxxxxx のようになります。
- 6 ESXi インストーラ イメージのコンテンツを、新しく作成したディレクトリにコピーします。
- 7 boot.cfg ファイルを修正します。
 - a 次の行を追加します。

```
prefix=ESXi-8.x.x-xxxxxx
```

ESXi-8.x.x-xxxxxx は、TFTP サーバのルート ディレクトリに対して相対的なインストーラ ファイルのパス名です。

- b kernel= と modules= の行のファイル名がスラッシュ (/) で始まる場合は、このスラッシュを削除します。
 - c kernelopt= の行に文字列 cdromBoot が含まれている場合は、その文字列のみを削除します。
- 8 (オプション) スクリプトを使用したインストールの場合は、boot.cfg ファイルの kernel コマンドの後の行に kernelopt オプションを追加して、インストール スクリプトの場所を指定します。

次のコードをモデルとして使用します。XXX.XXX.XXX.XXX には、インストール スクリプトが置かれたサーバの IP アドレスを指定します。esxi_ksFiles は、ks.cfg ファイルが格納されているディレクトリです。

```
kernelopt=ks=http://XXX.XXX.XXX.XXX/esxi_ksFiles/ks.cfg
```

- 9 ESXi ホストで UEFI ファームウェアが実行されている場合は、すべての UEFI ホストで同じインストーラを起動するかどうかを指定します。

オプション	説明
同じインストーラ	/tftpboot/boot.cfg に boot.cfg ファイルをコピーまたはリンクします。
異なるインストーラ	<ol style="list-style-type: none"> a /tftpboot のサブディレクトリを作成します。名前には、たとえば 01-23-45-67-89-0a-bc のようにターゲットのホストマシンの MAC アドレス (01-mac_address_of_target_ESXi_host) を使用します。 b このディレクトリ(たとえば /tftpboot/01-23-45-67-89-0a-bc/boot.cfg) に、ホストの boot.cfg ファイルをコピー (またはリンク) します。

iPXE および HTTP を使用した ESXi インストーラの起動

iPXE を使用して、HTTP サーバから ESXi インストーラを起動できます。

- レガシー BIOS マシンの場合、すべてのターゲット マシンに対して同じ pxelinux.0 の初期ブート ロードーを使用して複数の異なるバージョンの ESXi インストーラを起動することができますが、PXELINUX 構成ファイルはターゲット マシンの MAC アドレスに応じて異なる場合があります。
- UEFI マシンの場合、すべてのターゲット マシンに対して同じ mboot.efi の初期ブート ロードーを使用して複数の異なるバージョンの ESXi インストーラを起動することができますが、boot.cfg ファイルはターゲット マシンの MAC アドレスに応じて異なる場合があります。

前提条件

前提条件と手順は、UEFI ブートまたはレガシー BIOS のサポートのみに依存します。iPXE を使用して、HTTP サーバから ESXi インストーラを起動できます。次のトピックでは、UEFI ブートをサポートする ESXi ホストと、レガシーの BIOS のみをサポートするホストの前提条件と手順について説明します。

使用する環境内に次のコンポーネントがあることを確認します。

- VMware Web サイトからダウンロードした ESXi インストーラの ISO イメージ。
- 使用している ESXi のバージョンでサポートされるハードウェア構成を備えたターゲット ホスト。『VMware の互換性ガイド』を参照してください。
- ターゲット ESXi ホストに、PXE をサポートするネットワーク アダプタ。
- PXE ブート向けに構成できる DHCP サーバ。DHCP の構成例を参照してください。
- TFTP サーバ。
- TFTP トラフィックを許可するネットワーク セキュリティ ポリシー (UDP ポート 69)。
- レガシー BIOS の場合は、IPv4 ネットワークのみを使用できます。UEFI の PXE ブートの場合は、IPv4 または IPv6 ネットワークを使用できます。
- (オプション) インストール スクリプト (キックスタート ファイル)。
- ほとんどの場合、ネイティブ VLAN を使用します。PXE ブートで使用する VLAN ID を指定する場合は、NIC が VLAN ID 仕様をサポートしていることを確認します。

HTTP サーバを使用して PXE ブートを実行するために必要な次の前提条件を、使用中の環境が満たしていることを確認します。

- インストール先の ESXi ホストから HTTP サーバにアクセスできることを確認します。
- ESXi ホストでレガシー BIOS ファームウェアのみが実行されている場合は、SYSLINUX パッケージのバージョン 3.86 を取得します。詳細については、『ネットワーク ブートの背景情報』を参照してください。

手順

- 1 iPXE を取得して構成します。
 - a iPXE ソース コードを取得します。
 - b iPXE のダウンロード ページで、ビルド手順を実行します。ただし、次のいずれかのコマンドを実行してください。
 - ESXi ホストでレガシー BIOS ファームウェアのみが実行されている場合は、`make bin/undionly.kpxe` を実行します。
 - ESXi ホストで UEFI ファームウェアが実行されている場合は、`make bin-x86_64-efi/snponly.efi` を実行します。
 - c `undionly.kpxe` ファイルまたは `snponly.efi` ファイルを TFTP サーバの `/tftpboot` ディレクトリにコピーします。
- 2 ESXi ホストでレガシー BIOS ファームウェアのみが実行されている場合は、PXELINUX を取得して、構成します。
 - a SYSLINUX バージョン 3.86 を取得し、解凍した `pxelinux.0` ファイルを TFTP サーバの `/tftpboot` ディレクトリにコピーします。
 - b 次のコード モデルを使用して、PXELINUX 構成ファイルを作成します。

ESXi-8.x.x-XXXXXX は、ESXi インストーラ ファイルが含まれる TFTP サブディレクトリの名前です。

```
DEFAULT install
NOHALT 1
LABEL install
  KERNEL ESXi-8.x.x-XXXXXX/mboot.c32
  APPEND -c ESXi-8.x.x-XXXXXX/boot.cfg
  IPAPPEND 2
```

- c PXELINUX ファイルを TFTP サーバの `/tftpboot/pxelinux.cfg` ディレクトリに保存します。すべてのホストがこのインストーラをデフォルトで起動するかどうかは、ファイル名によって決まります。

オプション	説明
同じインストーラ	すべてのホストがデフォルトでこの ESXi インストーラを起動するには、ファイルに <code>default</code> という名前を付けます。
異なるインストーラ	特定のホストのみがこのファイルを起動する必要がある場合は、ファイルにターゲット ホスト マシンの MAC アドレスを含む名前を付けます (01- <i>mac_address_of_target_ESXi_host</i>)。たとえば、01-23-45-67-89-0a-bc と指定します。

- 3 ESXi ホストで UEFI ファームウェアが実行されている場合は、ESXi インストーラ ISO イメージから TFTP サーバ上の /tftpbboot フォルダに efi/boot/bootx64.efi ファイルをコピーし、ファイル名を mboot.efi に変更します。

注： 通常、mboot.efi の新しいバージョンは、ESXi の前のバージョンを起動できますが、mboot.efi の前のバージョンは、ESXi の新しいバージョンを起動できない場合があります。異なるホストを構成して異なるバージョンの ESXi インストーラを起動する場合は、最新バージョンの mboot.efi を使用します。

- 4 DHCP サーバを構成します。
- 5 HTTP サーバに保持される ESXi のバージョンと同じ名前のディレクトリを、HTTP サーバに作成します。たとえば、/var/www/html/ESXi-8.x.x-XXXXXX ディレクトリを作成します。
- 6 ESXi インストーラ イメージのコンテンツを、新しく作成したディレクトリにコピーします。
- 7 boot.cfg ファイルを修正します。
 - a 次の行を追加します。

```
prefix=http://XXX.XXX.XXX.XXX/ESXi-8.x.x-XXXXXX
```

http://XXX.XXX.XXX.XXX/ESXi-8.x.x-XXXXXX は HTTP サーバのインストーラ ファイルの場所です。

- b kernel= と modules= の行のファイル名がスラッシュ (/) で始まる場合は、このスラッシュを削除します。
 - c kernelopt= の行に文字列 cdromBoot が含まれている場合は、その文字列のみを削除します。
- 8 (オプション) スクリプトを使用したインストールの場合は、boot.cfg ファイルの kernel コマンドの後の行に kernelopt オプションを追加して、インストール スクリプトの場所を指定します。

次のコードをモデルとして使用します。XXX.XXX.XXX.XXX には、インストール スクリプトが置かれたサーバの IP アドレスを指定します。esxi_ksFiles は、ks.cfg ファイルが格納されているディレクトリです。

```
kernelopt=ks=http://XXX.XXX.XXX.XXX/esxi_ksFiles/ks.cfg
```

- 9 ESXi ホストで UEFI ファームウェアが実行されている場合は、すべての UEFI ホストで同じインストーラを起動するかどうかを指定します。

オプション	説明
同じインストーラ	/tftpbboot/boot.cfg に boot.cfg ファイルをコピーまたはリンクします。
異なるインストーラ	<ol style="list-style-type: none"> a /tftpbboot のサブディレクトリを作成します。名前には、たとえば 01-23-45-67-89-0a-bc のようにターゲットのホストマシンの MAC アドレス (01-mac_address_of_target_ESXi_host) を使用します。 b このディレクトリ(たとえば /tftpbboot/01-23-45-67-89-0a-bc/boot.cfg) に、ホストの boot.cfg ファイルをコピー (またはリンク) します。

ネイティブ UEFI HTTP を使用した ESXi インストーラの起動

ESXi インストーラは、起動プロセスをサポートする追加のソフトウェアを使用せずに、HTTP サーバから直接起動できます。

UEFI HTTP は、ESXi インストーラの複数のバージョンの起動をサポートしています。すべてのターゲットマシンに同じ初期ブートローダー `mboot.efi` を使用します。ただし、ターゲットマシンの MAC アドレスによっては、異なる `boot.cfg` ファイルが使用される場合があります。

注： 起動プロセスで、IPv4 または IPv6 ネットワークを混在させないでください。IPv4 ネットワークまたは IPv6 ネットワークのいずれかを使用します。

前提条件

使用する環境内に次のコンポーネントがあることを確認します。

- HTTP ブート機能をサポートする UEFI ファームウェアを搭載した ESXi ホスト。
- VMware Web サイトからダウンロードした ESXi インストーラの ISO イメージ。
- 使用している ESXi のバージョンでサポートされるハードウェア構成を備えたターゲット ホスト。『VMware の互換性ガイド』を参照してください。
- ターゲット ESXi ホスト上に、UEFI ネットワークをサポート可能なネットワーク アダプタ。
- UEFI HTTP 起動用に構成可能な DHCP サーバ。『DHCP の構成例』を参照してください
- (オプション) インストール スクリプト (キックスタート ファイル)。
- ほとんどの場合、ネイティブ VLAN を使用します。PXE ブートで使用する VLAN ID を指定する場合は、NIC が VLAN ID 仕様をサポートしていることを確認します。

手順

- 1 ESXi インストーラの ISO イメージから HTTP サーバ上のディレクトリに `efi/boot/bootx64.efi` ファイルをコピーし、ファイル名を `mboot.efi` に変更します。たとえば、`http://www.example.com/esxi/mboot.efi` にコピーします。

注： 通常、`mboot.efi` の新しいバージョンは、ESXi の前のバージョンを起動できますが、`mboot.efi` の前のバージョンは、ESXi の新しいバージョンを起動できない場合があります。異なるホストを構成して異なるバージョンの ESXi インストーラを起動する場合は、最新バージョンの `mboot.efi` を使用します。

- 2 DHCP サーバを構成します。
- 3 HTTP サーバに保持される ESXi のバージョンと同じ名前のディレクトリを、HTTP サーバに作成します。たとえば、`http://www.example.com/esxi/ESXi-8.x.x-XXXXXX` などです。
- 4 ESXi インストーラ イメージのコンテンツを、新しく作成したディレクトリにコピーします。
- 5 `boot.cfg` ファイルを修正します。
 - a 新しく作成されたディレクトリの URL を含む次の行を追加します。

```
prefix=http://www.example.com/esxi/ESXi-8.x.x-XXXXXX
```

- b `kernel=` と `modules=` の行のファイル名がスラッシュ (/) で始まる場合は、このスラッシュを削除します。
- c `kernelopt=` の行に文字列 `cdromBoot` が含まれている場合は、その文字列のみを削除します。

- 6 (オプション) スクリプトを使用したインストールの場合は、`boot.cfg` ファイルの `kernel` コマンドの後の行に `kernelopt` オプションを追加して、インストール スクリプトの場所を指定します。

例: `kernelopt=ks=http://www.example.com/esxi_ksFiles/ks.cfg`

- 7 (オプション) 仮想マシンの構成パラメータ `networkBootProtocol` および `networkBootUri` を使用して、仮想マシンの起動元を指定できます。 `networkBootProtocol` を設定すると、起動プロトコルとして IPv4 または IPv6 が指定されます。たとえば、`networkBootProtocol = httpv4` など。 `networkBootUri` を指定すると、ESXi ブートローダー (`bootx64.efi`) の HTTP URL が指定されます。たとえば、`networkBootUri = http://xxx.xxx.xx.x/esxi80uc1/efi/boot/bootx64.efi` です。
- 8 すべての UEFI ホストが同じインストーラを起動するかどうかを指定します。

オプション	説明
同じインストーラ	<code>boot.cfg</code> ファイルを <code>mboot.efi</code> と同じディレクトリに追加します。例: <code>http://www.example.com/esxi/boot.cfg</code>
異なるインストーラ	<p>a <code>mboot.efi</code> ファイルを含むディレクトリのサブディレクトリを作成します。ディレクトリに、ターゲット ホスト マシンの MAC アドレス (<code>01-mac_address_of_target_ESXi_host</code>) を名前として付けます。 <code>01-23-45-67-89-0a-bc</code> のようになります。</p> <p>b カスタム <code>boot.cfg</code> ファイルをディレクトリに追加します。 <code>http://www.example.com/esxi/01-23-45-67-89-0a-bc/boot.cfg</code> のようになります。</p>

両方のインストーラ タイプを使用できます。HTTP サーバ上にカスタム `boot.cfg` ファイルがない ESXi ホストは、デフォルトの `boot.cfg` ファイルから起動します。

DHCP の構成例

DHCP サーバが TFTP または HTTP サーバのアドレスと初期ブートローダーのファイル名を ESXi ホストに送信する必要があります。

ターゲット マシンは初めて起動するときに、ネットワーク全体にパケットをブロードキャストして、起動に必要な情報を要求します。DHCP サーバがこれに応答します。DHCP サーバは、ターゲット マシンの起動が許可されているかどうかと、初期ブート ローダーのバイナリの場所を判断できる必要があります。PXE ブートの場合、場所は TFTP サーバ上のファイルです。UEFI HTTP ブートの場合、場所は URL です。

注意: ネットワークにすでに DHCP サーバがある場合は、2 台目の DHCP サーバを設定しないでください。複数の DHCP サーバが DHCP 要求に応答すると、マシンが不正な、または競合する IP アドレスを取得するか、正しい起動情報を受け取れない可能性があります。DHCP サーバを設定する前に、ネットワーク管理者に確認してください。DHCP の構成についてサポートが必要な場合は、DHCP サーバのベンダーにお問い合わせください。

使用できる DHCP サーバは多数あります。次の例は、ISC DHCP サーバの場合です。Microsoft Windows 用の DHCP バージョンを使用している場合は、DHCP サーバのドキュメントを参照して、`next-server` 引数と `filename` 引数をターゲット マシンに渡す方法を確認してください。

IPv4 で PXE および TFTP を使用した起動の例

この例では、IPv4 アドレスが xxx.xxx.xxx.xxx の TFTP サーバを使用して ESXi を PXE ブートするように ISC DHCP サーバを構成する方法を示しています。

```
#
# ISC DHCP server configuration file snippet.  This is not a complete
# configuration file; see the ISC server documentation for details on
# how to configure the DHCP server.
#
allow booting;
allow bootp;
option client-system-arch code 93 = unsigned integer 16;
class "pxeclients" {
    match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
    next-server xxx.xxx.xxx.xxx;
    if option client-system-arch = 00:07 or option client-system-arch = 00:09 {
        filename = "mboot.efi";
    } else {
        filename = "pxelinux.0";
    }
}
```

マシンが PXE ブートを行うときには、DHCP サーバが IP アドレスと TFTP サーバの pxelinux.0 または mboot.efi バイナリ ファイルの場所を示します。

IPv6 で PXE および TFTP を使用した起動の例

この例では、IPv6 アドレスが xxxx:xxxx:xxxx:xxxx::xxxx の TFTP サーバを使用して ESXi を PXE ブートするように ISC DHCPv6 サーバを構成する方法を示しています。

```
#
# ISC DHCPv6 server configuration file snippet.  This is not a complete
# configuration file; see the ISC server documentation for details on
# how to configure the DHCP server.
#
allow booting;
allow bootp;
option dhcp6.bootfile-url code 59 = string;
option dhcp6.bootfile-url "tftp://[xxxx:xxxx:xxxx:xxxx::xxxx]/mboot.efi";
```

マシンが PXE ブートを行うときには、DHCP サーバが IP アドレスと TFTP サーバの mboot.efi バイナリ ファイルの場所を示します。

IPv4 で iPXE および HTTP を使用した起動の例

この例では、IPv4 アドレスが xxx.xxx.xxx.xxx の TFTP サーバから iPXE をロードすることによって ESXi を起動するように ISC DHCP サーバを構成する方法を示しています。

```
#
# ISC DHCP server configuration file snippet.  This is not a complete
# configuration file; see the ISC server documentation for details on
# how to configure the DHCP server.
#
```

```

allow booting;
allow bootp;
option client-system-arch code 93 = unsigned integer 16;
class "pxeclients" {
    match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
    next-server xxx.xxx.xxx.xxx;
    if option client-system-arch = 00:07 or option client-system-arch = 00:09 {
        if exists user-class and option user-class = "iPXE" {
            # Instruct iPXE to load mboot.efi as secondary bootloader
            filename = "mboot.efi";
        } else {
            # Load the snponly.efi configuration of iPXE as initial bootloader
            filename = "snponly.efi";
        }
    } else {
        if exists user-class and option user-class = "iPXE" {
            # Instruct iPXE to load pxelinux as secondary bootloader
            filename = "pxelinux.0";
        } else {
            # Load the undionly configuration of iPXE as initial bootloader
            filename = "undionly.kpxe";
        }
    }
}

```

マシンが PXE ブートを行うときには、DHCP サーバが IP アドレスと TFTP サーバの undionly.kpxe または snponly.efi バイナリ ファイルの場所を示します。レガシー BIOS の場合は、iPXE が次にロードするファイルを DHCP サーバに要求し、サーバはファイル名として pxelinux.0 を返します。UEFI の場合は、iPXE が DHCP サーバにロードする次のファイルを要求し、この時点でサーバはファイル名として mboot.efi を返します。いずれの場合でも、iPXE は常駐で、システムには HTTP の機能があります。そのため、システムは HTTP サーバから追加のファイルをロードできます。

IPv6 で iPXE および HTTP を使用した起動の例

この例では、IPv6 アドレスが xxxx:xxxx:xxxx:xxxx::xxxx の TFTP サーバから iPXE をロードすることによって ESXi を起動するように ISC DHCPv6 サーバを構成する方法を示しています。

```

#
# ISC DHCPv6 server configuration file snippet.  This is not a complete
# configuration file; see the ISC server documentation for details on
# how to configure the DHCP server.
#
allow booting;
allow bootp;

option dhcp6.bootfile-url code 59 = string;
if exists user-class and option user-class = "iPXE" {
    # Instruct iPXE to load mboot.efi as secondary bootloader
    option dhcp6.bootfile-url "tftp://[xxxx:xxxx:xxxx:xxxx::xxxx]/mboot.efi";
} else {
    # Load the snponly.efi configuration of iPXE as initial bootloader
    option dhcp6.bootfile-url "tftp://[xxxx:xxxx:xxxx:xxxx::xxxx]/snponly.efi";
}

```

マシンが PXE ブートを行うときには、DHCP サーバが IP アドレスと TFTP サーバの `snponly.efi` (iPXE) バイナリ ファイルの場所を示します。iPXE が DHCP サーバにロードする次のファイルを要求し、この時点でサーバはファイル名として `mboot.efi` を返します。iPXE は常駐で、システムには HTTP の機能があります。そのため、システムは HTTP サーバから追加のファイルをロードできます。

IPv4 で UEFI HTTP を使用した起動の例

この例では、Web サーバ `www.example.com` から IPv4 上のネイティブ UEFI HTTP を使用して ESXi を起動するように ISC DHCP サーバを構成する方法を示しています。

```
#
# ISC DHCP server configuration file snippet.  This is not a complete
# configuration file; see the ISC server documentation for details on
# how to configure the DHCP server.
#
allow booting;
allow bootp;
option client-system-arch code 93 = unsigned integer 16;
class "httpclients" {
    match if substring(option vendor-class-identifier, 0, 10) = "HTTPClient";
    option vendor-class-identifier "HTTPClient";

    if option client-system-arch = 00:10 {
        # x86_64 UEFI HTTP client
        filename = http://www.example.com/esxi/mboot.efi;
    }
}
```

IPv6 で UEFI HTTP を使用した起動の例

この例では、Web サーバ `www.example.com` から IPv6 上のネイティブ UEFI HTTP を使用して ESXi を起動するように ISC DHCPv6 サーバを構成する方法を示しています。

```
#
# ISC DHCPv6 server configuration file snippet.  This is not a complete
# configuration file; see the ISC server documentation for details on
# how to configure the DHCP server.
#
allow booting;
allow bootp;

option dhcp6.bootfile-url code 59 = string;
option dhcp6.user-class code 15 = { integer 16, string };
option dhcp6.vendor-class code 16 = { integer 32, integer 16, string };

if option dhcp6.client-arch-type = 00:10 {
    # x86_64 HTTP clients
    option dhcp6.vendor-class 0 10 "HTTPClient";
    option dhcp6.bootfile-url "http://www.example.com/esxi/mboot.efi";
}
```

vSphere Auto Deploy を使用した ESXi のインストール

vSphere Auto Deploy を使用すると、ESXi ソフトウェア搭載の物理ホストを大量にプロビジョニングできます。

Auto Deploy を使用すると、経験の豊富なシステム管理者は、大規模な導入を効率良く管理できます。ホストは、統合 Auto Deploy サーバからネットワーク起動されます。ホストは、必要に応じて、参照ホストのホスト プロファイルを使用して構成されます。ユーザーに入力を求めるように、ホスト プロファイルを設定できます。起動と構成が完了したら、ホストは他の ESXi ホストと同様に、vCenter Server によって管理されます。

Auto Deploy は、ステートレス キャッシュまたはステートフル インストールで使用することもできます。

重要： Auto Deploy には、[vSphere Auto Deploy のセキュリティの考慮事項](#)の説明のとおり、本番ネットワークと管理ネットワークまたはデプロイ ネットワークのセキュアな分離が必要です。分離を使用しない Auto Deploy の使用は安全ではありません。

ステートレス キャッシュ

Auto Deploy は、デフォルトでは、ホスト ディスクの ESXi 構成または状態を保存しません。代わりに、イメージ プロファイルによってそのホストにプロビジョニングされているイメージを定義し、その他のホスト属性はホストのプロファイルによって管理されます。ステートレス キャッシュに Auto Deploy を使用するホストは、引き続き、Auto Deploy サーバと vCenter Server に接続する必要があります。

ステートフル インストール

Auto Deploy によってホストをプロビジョニングし、イメージをディスクに保存するようにホストを設定できます。その後の起動では、ホストがディスクから起動します。

vSphere Auto Deploy について

vSphere Auto Deploy を使用すると、ESXi ソフトウェア搭載の物理ホストを大量にプロビジョニングすることができます。

デプロイするイメージと、そのイメージを使用してプロビジョニングするホストを指定できます。任意で、ホスト、vCenter Server の場所（データセンター、フォルダまたはクラスタ）に適用するホスト プロファイルを指定し、スクリプト バンドルを各ホストに割り当てることができます。

vSphere Auto Deploy の概要

vSphere Auto Deploy は、PXE ブート インフラストラクチャをホスト プロファイル、目的のイメージ、またはクラスタ レベルでの構成とともに 使用して、ESXi ホストをプロビジョニングします。

ESXi ホストの状態情報

注： DPU が構成された ESXi ホストで、vSphere Distributed Services Engine 機能の一部として Auto Deploy を使用することはできません。

vSphere Auto Deploy 用に設定されている物理ホストを起動する場合、vSphere Auto Deploy は、PXE ブートのインフラストラクチャを vSphere ホスト プロファイル、目的のイメージ、またはクラスタ レベルでの構成と組み合わせて使用して、そのホストのプロビジョニングとカスタマイズを行います。ホスト自体に状態は保存されません。代わりに、vSphere Auto Deploy サーバが各ホストの状態情報を管理します。vSphere Auto Deploy は、さまざまな場所でプロビジョニングされる ESXi ホストの情報を保存します。単一のイメージまたはクラスタ レベルでの構成によって管理するイメージ プロファイル、ホスト プロファイル、またはクラスタの場所に関する情報は、マシンをイメージ プロファイルおよびホスト プロファイルにマッピングするルールで最初に指定されます。

表 4-16. vSphere Auto Deploy がデプロイ用に保存する情報

情報のタイプ	説明	情報ソース
イメージの状態	ESXi ホスト上で実行される実行可能ソフトウェア。	vSphere ESXi Image Builder または vSphere Lifecycle Manager イメージを使用して作成されたイメージ プロファイル。
構成の状態	ホストの構成方法を定める構成設定 (仮想スイッチとその設定、ドライバ設定、起動パラメータなど)。	ホスト プロファイル ユーザー インターフェイスを使用して作成されるホスト プロファイル、またはクラスタ レベルですべての ESXi ホスト設定を管理するクラスタをインベントリ ユーザー インターフェイスで設定するときに作成する構成。
動的な状態	生成されたプライベート キーまたはランタイム データベースなどの、実行中のソフトウェアによって生成されるランタイムの状態。	ホストのメモリ、再起動中に失われます。
仮想マシンの状態	ホストに追加された仮想マシンと、仮想マシンの自動起動の情報 (2 回目以降の起動のみ)。	vCenter Server では、vSphere Auto Deploy に送信する仮想マシンの情報を提供できるようにしておく必要があります。
ユーザー入力	システムの起動時にユーザーが入力する IP アドレスなどの、ホスト プロファイルに自動的に含めることができない、ユーザー入力に基づいた状態。	最初の起動時に vCenter Server によって保存されるホストのカスタマイズ情報。 一部の値については、ユーザー入力を求めるホスト プロファイルを作成できます。 vSphere Auto Deploy により、ユーザー入力情報を必要とするホスト プロファイルが適用されると、ホストはメンテナンス モードになります。ホスト プロファイル ユーザー インターフェイスを使用してホスト プロファイルのコンプライアンスを確認し、プロンプトに応答してホストをカスタマイズします。

vSphere Auto Deploy のアーキテクチャ

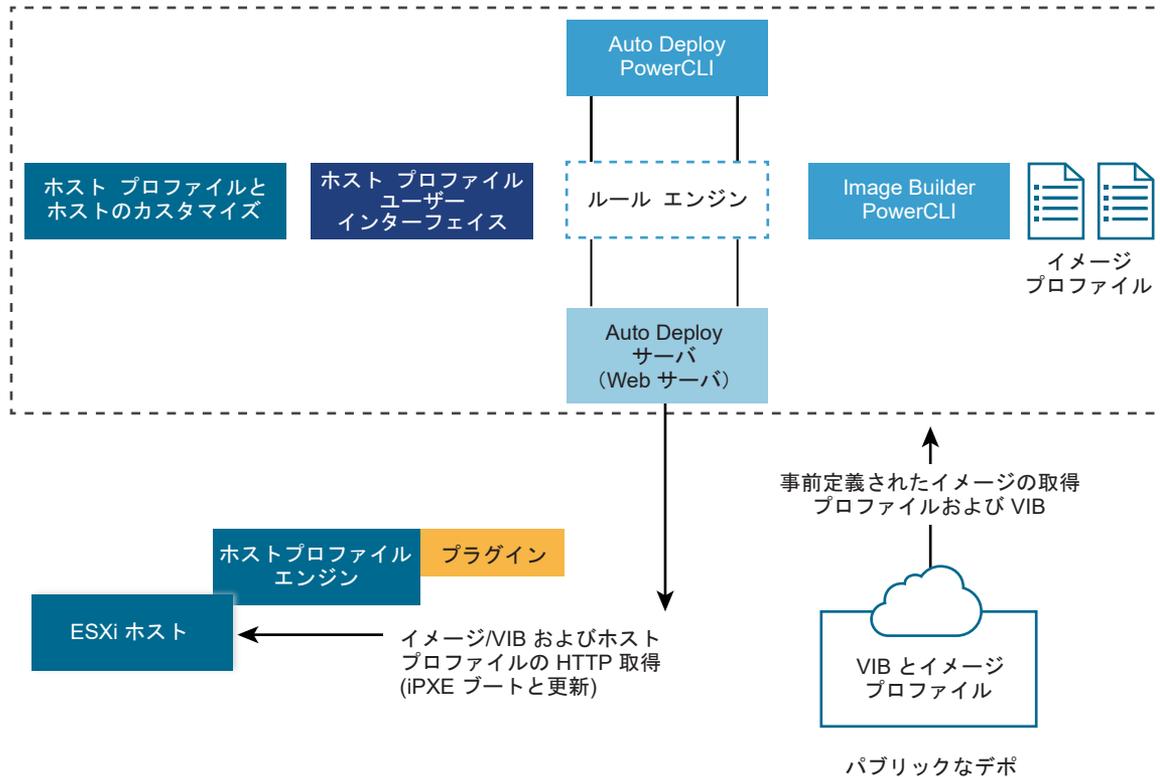
vSphere Auto Deploy のインフラストラクチャは、いくつかのコンポーネントで構成されています。

詳細については、ビデオ「Auto Deploy のアーキテクチャ」を参照してください。



(Auto Deploy のアーキテクチャ)

図 4-6. vSphere Auto Deploy のアーキテクチャ



vSphere Auto Deploy サーバ

イメージ プロファイルとホスト プロファイルを ESXi ホストに提供します。

vSphere Auto Deploy ルール エンジン

提供されるホストと提供するイメージ プロファイルおよびホスト プロファイルとの関係を示す情報を、vSphere Auto Deploy サーバに送信します。システム管理者は、vSphere Auto Deploy を使用して、ホストにイメージ プロファイルとホスト プロファイルを割り当てるルールを定義します。vSphere Auto Deploy ルールおよびルール セットの詳細については、[ルールおよびルール セット](#)を参照してください。

VMware Image Builder とホスト プロファイルを使用して作成するレガシー イメージ プロファイルとは別に、単一の vSphere Lifecycle Manager イメージまたはクラスタ レベルでの構成を使用して、ESXi を展開する vSphere Auto Deploy ルールを作成することもできます。

イメージ プロファイル

ESXi ホストの起動に使用する一連の VIB を定義します。

- VMware および VMware パートナーは、イメージ プロファイルおよび VIB をパブリックなデポに用意しています。vSphere ESXi Image Builder を使用してデポの内容を確認し、vSphere Auto Deploy ルール エンジンを使用してイメージ プロファイルとその割り当て先のホストの関係を指定します。
- vSphere Lifecycle Manager イメージを使用して、ソフトウェアとファームウェアのアップデートをクラスタ内の ESXi ホストに適用します。単一イメージを使用してクラスタ内のすべてのホストを管理することで、クラスタ全体でホスト イメージを同一にすることができます。

- ESXi 8.0 では、クラスターレベルですべての ESXi ホスト設定を管理するクラスターを設定できます。
- VMware カスタムは、デポにあるパブリックなイメージ プロファイルおよび VIB に基づいて、カスタムのイメージ プロファイルを作成し、そのイメージ プロファイルをホストに適用することができます。
[vSphere ESXi Image Builder を使用したインストールのカスタマイズ](#)を参照してください。

ホスト プロファイル

ネットワークやストレージの設定などの、マシンに固有の構成を定義します。ホスト プロファイル ユーザー インターフェイスを使用して、ホスト プロファイルを作成します。リファレンス ホストにホスト プロファイルを作成し、一貫した構成のために、そのホスト プロファイルを環境内のほかのホストに適用することができます。詳細については、『vSphere のホスト プロファイル』ドキュメントまたは [vSphere Auto Deploy のリファレンス ホストの設定](#) セクションを参照してください。

ホストのカスタマイズ

ホストにホスト プロファイルが適用されるときにユーザーが提供する情報を格納します。ホストのカスタマイズには、ユーザーがそのホストに提供した IP アドレスまたはその他の情報が含まれる場合があります。ホストのカスタマイズの詳細については、『vSphere のホスト プロファイル』ドキュメントを参照してください。

以前のリリースの vSphere Auto Deploy では、ホストのカスタマイズは応答ファイルと呼ばれていました。

Auto Deploy の証明書

デフォルトでは、Auto Deploy サーバは VMware Certificate Authority (VMware CA) が署名した証明書を使用して各ホストをプロビジョニングします。詳細については、[ESXi ホストの証明書の管理](#)を参照してください。

また、企業ポリシーでカスタム証明書を使用する必要がある場合は、すべてのホストを VMware CA によって署名されていないカスタム証明書を使用してプロビジョニングするように Auto Deploy サーバを設定できます。

Auto Deploy サーバはサードパーティ認証局の従属認証局になります。カスタム認証局モードでは、ユーザーが証明書を管理する必要があります。vSphere Client からの証明書を更新することはできません。このモードでは、カスタム証明書を使用してプロビジョニングするホストのセットのみを選択することもできません。また、カスタム証明書に手動で署名できるのは、ステートフル ホストに限られます。詳細については、[Auto Deploy でのカスタム証明書の使用](#)を参照してください。

ESXi 8.0 の Auto Deploy には、vSphere の外部で証明書を生成し、vCenter Server の証明書管理から独立できる 3 番目のオプションが用意されています。たとえば、カスタム スクリプトを使用するか、Verisign などのドメイン名レジストリ サービスのプロバイダを使用して、カスタム証明書を生成できます。カスタム証明書は、ESXi ホストのセットにのみ使用できます。カスタム証明書はステートレス ホストにも提供できます。ESXi ホストは、ネットワーク起動に使用される NIC の MAC アドレス、または ESXi ホストの BIOS UUID によって識別されます。VMware Endpoint Certificate Store (VECS) は、PowerCLI を使用してカスタム証明書で更新します。新しい PowerCLI コマンドレットの詳細については、[vSphere Auto Deploy PowerCLI コマンドレットの概要](#)を参照してください。VMware CA はカスタム ESXi 証明書を信頼する必要があるため、カスタム証明書の CA パブリック証明書を VECS の TRUSTED_ROOTS ストアに追加する必要があります。Auto Deploy はカスタム証明書も保存し、ネットワーク起動に使用される NIC の MAC アドレス、または ESXi ホストの BIOS UUID によって起動ホストを認識すると、自動的にカスタム証明書を提供します。カスタム証明書を VECS に追加するときに、Auto Deploy も vCenter Server も停止したり再起動したりする必要はありません。カスタム証明書をアップロードするホストのみを再起動します。詳細については、[Auto Deploy でのカスタム証明書の使用](#)を参照してください。

ルールおよびルール セット

一連のルールを使用することで、vSphere Auto Deploy サーバの動作を指定します。vSphere Auto Deploy ルール エンジンが、一致するホストのパターンをルール セットで確認し、どの項目（イメージ プロファイル、ホスト プロファイル、vCenter Server の場所、またはスクリプト オブジェクト）を使用して各ホストをプロビジョニングするかを決定します。

ルール エンジンには、ホストの属性に基づいて、ソフトウェア設定と構成設定をホストにマッピングします。たとえば、2 つのルールを記述して（それぞれが 1 つのクラスタのネットワーク アドレスに対応）、イメージ プロファイルまたはホスト プロファイルを 2 つのホストのクラスタにデプロイすることができます。

vCenter Server システムに追加されていないホストについては、vSphere Auto Deploy サーバは、ルール エンジンを確認してから、イメージ プロファイル、ホスト プロファイル、およびインベントリの場所の情報をホストに提供します。vCenter Server システムによって管理されているホストについては、vCenter Server がホストのオブジェクト内に格納したイメージ プロファイル、ホスト プロファイル、およびインベントリの場所が使用されます。ルールを変更する場合、PowerCLI セッションで vSphere Client または vSphere Auto Deploy コマンドレットを使用し、ルールのコンプライアンスをテストおよび修復することができます。ホストのルールのコンプライアンスを修復する場合は、ホストのイメージ プロファイルおよびホスト プロファイルの割り当てが更新されます。

ルール エンジンには、ルールおよびルール セットが含まれます。

ルール

ルールを使用して、イメージ プロファイルおよびホスト プロファイルを一連のホストに割り当てたり、ターゲットの vCenter Server システム上のホストの場所（フォルダまたはクラスタ）を指定することができます。ルールでは、起動 MAC アドレス、SMBIOS 情報、BIOS UUID、ベンダー、モデル、または固定 DHCP IP アドレスによってターゲット ホストを特定できます。多くの場合、ルールは複数のホストに適用されます。ルールを作成するには、PowerCLI セッションで vSphere Client または vSphere Auto Deploy コマンドレットを使用します。作成したルールは、ルール セットに追加する必要があります。アクティブなルール セットと作業ルール セットの 2 つのルール セットのみがサポートされます。ルールは両方のセットに属することができますが（デフォルト）、作業ルール セットにだけ属することも可能です。ルール セットに追加したルールは、変更できません。代わりに、ルールをコピーして、そのコピーで項目またはパターンを置き換えます。vSphere Client で vSphere Auto Deploy を管理している場合、ルールが非アクティブな状態であれば編集することができます。

ルールでは次のパラメータを指定することができます。

パラメータ	説明
名前	ルールの名前。-Name パラメータで指定します。
アイテム	1つまたは複数の項目。-Item パラメータで指定します。アイテムには、イメージ プロファイル、ホスト プロファイル、ターゲット ホストの vCenter Server インベントリの場所（データセンター、フォルダ、クラスター）またはカスタム スクリプトがあります。複数の項目を、コンマ区切りで指定できます。
パターン	パターンには、ルールの適用先となる、ホストまたはホストのグループを指定します。
ベンダー	
	マシンのベンダー名。
モデル	
	マシンのモデル名。
シリアル	
	マシンのシリアル番号。
ホスト名	
	マシンのホスト名。
ドメイン	
	ドメイン名。
IPv4	
	マシンの IPv4 アドレス。
ipv6	
	マシンの IPv6 アドレス。
	BIOS ファームウェアを使用する PXE ブートは IPv4 でのみ実行でき、UEFI ファームウェアを使用する PXE ブートは IPv4 または IPv6 のいずれかで実行できます。
MAC	
	起動 NIC の MAC アドレス。
アセット	
	マシンのアセット タグ。
OEM 文字列	
	SMBIOS の OEM 固有の文字列。
	1つの項目または複数の項目をすべてのホストに適用するには、-AllHosts を指定することができます。

アクティブなルール セット

新しく起動したホストが、イメージ プロファイルの要求とともに vSphere Auto Deploy サーバに接続すると、vSphere Auto Deploy サーバはアクティブなルール セットで一致するルールを確認します。一致するルールによってマッピングされたイメージ プロファイル、ホスト プロファイル、vCenter Server インベントリの場所、およびスクリプト オブジェクトが、ホストの起動に使用されます。同じタイプの複数の項目がルールに

よってマッピングされる場合は、vSphere Auto Deploy サーバでは、それらの項目のうち、ルール セット内に最初に出現するものが使用されます。

作業ルール セット

作業ルール セットを使用すると、変更をアクティブにする前にルールへの変更をテストできます。たとえば、vSphere Auto Deploy コマンドレットを使用して、作業ルール セットとのコンプライアンスをテストできます。テストは、vCenter Server システムによって管理されるホストが、作業ルール セットのルールに従っているかを検証します。デフォルトでは、cmdlet は作業ルール セットにルールを追加し、ルールを有効にします。作業ルール セットにのみルールを追加するには、NoActivate パラメータを使用します。

次のワークフローはルールおよびルール セットと一緒に使用します。

- 1 作業ルール セットを変更します。
- 2 ホストに対して作業ルール セットのルールをテストし、すべてが適切に機能することを確認します。
- 3 作業ルール セット内のルールを調整し、再テストします。
- 4 作業ルール セット内のルールを有効にします。

PowerCLI セッションにルールを追加して NoActivate パラメータを指定しない場合、作業ルール セット内に現在あるすべてのルールが有効になります。個々のルールを有効にすることはできません。

PowerCLI コマンドレットによって vSphere Auto Deploy を使用方法の詳細については、PowerCLI コマンドライン ヘルプおよび [PowerCLI コマンドレットを使用した vSphere Auto Deploy の管理](#)を参照してください。vSphere Client によって vSphere Auto Deploy を使用方法の詳細については、[vSphere Client を使用した vSphere Auto Deploy の管理](#)を参照してください。

vSphere Auto Deploy のインストールと構成

vSphere Auto Deploy の使用を開始する前に、いくつかの手順で環境を準備する必要があります。

サーバのセットアップと、ハードウェアの準備から始めます。プロビジョニングするホストの管理に使用する vCenter Server システムで、vSphere Auto Deploy サービスの起動タイプを設定し、PowerCLI をインストールする必要があります。

次に参照するドキュメント

■ [vSphere Auto Deploy のインストール前のチェックリスト](#)

この vSphere Auto Deploy シナリオのタスクを開始する前に、使用環境でハードウェア要件とソフトウェア要件が満たされており、コンポーネントのセットアップに必要な権限を持っていることを確認します。

■ [vSphere Auto Deploy 用にシステムを準備](#)

ESXi ホストを vSphere Auto Deploy で PXE ブートする前に、前提となるソフトウェアをインストールして、vSphere Auto Deploy が通信する DHCP サーバおよび TFTP サーバを設定する必要があります。

■ [vSphere Auto Deploy コマンドレットの使用](#)

vSphere Auto Deploy コマンドレットは、Microsoft PowerShell コマンドレットとして実装され、PowerCLI に含まれます。vSphere Auto Deploy コマンドレットのユーザーは、すべての PowerCLI 機能を利用できます。

■ バルク ライセンス 供与の設定

vSphere Client または ESXi Shell を使用すると、個別のライセンス キーを指定できます。または、PowerCLI コマンドレットを使用して、バルク ライセンス 供与を設定することができます。バルク ライセンス 供与はすべての ESXi ホストで行えますが、vSphere Auto Deploy を使用してプロビジョニングされるホストで特に有用です。

vSphere Auto Deploy のインストール前のチェックリスト

この vSphere Auto Deploy シナリオのタスクを開始する前に、使用環境でハードウェア要件とソフトウェア要件が満たされており、コンポーネントのセットアップに必要な権限を持っていることを確認します。

表 4-17. インストール前チェックリスト

必要なソフトウェアとハードウェア	詳細
vCenter Server	vSphere Auto Deploy サーバは、vCenter Server の一部です。vCenter Server システムで vSphere Auto Deploy サービスを有効にして開始する必要があります。vCenter Server にログインして、設定タスクの多くを実行できます。 vSphere Auto Deploy 用にシステムを準備 を参照してください。
ストレージ	サーバで LUN を検出できるように構成されたサーバおよびストレージ アレイを含む、NFS、iSCSI、またはファイバ チャネルなどの ESXi データストア用のストレージ。 <ul style="list-style-type: none"> ■ NFS または iSCSI 用のターゲット IP アドレスのリスト。 ■ NFS または iSCSI 用のターゲット ボリューム情報のリスト。
ホスト情報 (4 台の ESXi ホストすべて)	NFS または iSCSI 用のターゲット IP アドレスのリスト。 NFS または iSCSI 用のターゲット ボリューム情報のリスト。 <ul style="list-style-type: none"> ■ デフォルトのルート、ネット マスク、およびプライマリ/セカンダリ DNS サーバの IP アドレス。 ■ VMkernel プライマリ管理ネットワーク用の IP アドレスおよびネット マスク。 ■ ストレージ、vSphere FT、または VMware vMotion などの、その他の VMkernel ネットワーク用の IP アドレスおよびネット マスク。 vSphere Auto Deploy では、はデフォルトで既存のパーティション上書きされません。
PowerCLI	[PowerCLI のインストール] を参照してください。
ESXi ソフトウェア デポ	VMware Web サイトの [ダウンロード] ページの ESXi ソフトウェア デポの場所。この場所に格納されているイメージ プロファイルをポイントする URL を使用するか、ローカル デポで使用する ZIP ファイルをダウンロードします。ESXi イメージはダウンロードしないでください。
TFTP サーバ	WinAgents の TFTP サーバなどの、TFTP インストーラ ソフトウェア。
DHCP サーバ	DHCP サーバは、vSphere でサポートされるバージョンの Windows Server に含まれています。
DNS サーバ	動作中の DNS サーバ。エントリは、各ターゲット ホストの正引き (A レコード) および逆引き (PTR レコード) の両方のゾーンに追加する必要があります。

Active Directory サーバ、DNS サーバ、DHCP サーバ、NTP サーバなどの、環境のコア サーバの管理者権限に関する情報も必要です。

セットアップを行うサブネットのブロードキャスト ドメインを完全に管理できる権限が必要です。このサブネットを、ほかの DHCP サーバ、DNS サーバ、または TFTP サーバが使用していないことを確認します。

vSphere Auto Deploy 用にシステムを準備

ESXi ホストを vSphere Auto Deploy で PXE ブートする前に、前提となるソフトウェアをインストールして、vSphere Auto Deploy が通信する DHCP サーバおよび TFTP サーバを設定する必要があります。

PowerCLI コマンドレットで vSphere Auto Deploy を管理する方法については、「vSphere PowerCLI を使用した vSphere Auto Deploy のセットアップとホストのプロビジョニング」を参照してください。

前提条件

- vSphere Auto Deploy でプロビジョニングするホストが、ESXi のハードウェア要件を満たしていることを確認します。[ESXi のハードウェア要件](#)を参照してください。
- ESXi ホストに vCenter Server へのネットワーク接続があることと、すべてのポートの要件が満たされていることを確認します。「vCenter Server のアップグレード」を参照してください。
- 環境内に ファイルを送信する TFTP サーバおよび DHCP サーバが展開されており、Auto Deploy をプロビジョニングする ESXi ホストにネットワーク アドレスが割り当てられていることを確認します。『[TFTP サーバのインストール](#)』と『[vSphere Auto Deploy によるプロビジョニングのための DHCP サーバの準備](#)』を参照してください。
- ESXi ホストが DHCP、TFTP、および vSphere Auto Deploy サーバとネットワーク接続できることを確認します。
- vSphere Auto Deploy 環境で VLAN を使用する場合は、エンド ツー エンドのネットワークを適切に設定する必要があります。ホストの PXE ブート時は、フレームに適切な VLAN ID をタグ付けするようにファームウェア ドライバを設定する必要があります。この設定は、UEFI/BIOS インターフェイスで正しい変更を手動で行う必要があります。また、正しい VLAN ID で ESXi ポート グループを適切に構成する必要があります。環境内で VLAN ID がどのように使用されているかをネットワーク管理者に問い合わせます。
- vSphere Auto Deploy リポジトリ用に十分なストレージがあることを確認します。vSphere Auto Deploy サーバは、必要なデータを保存するためにリポジトリを使用します。このデータには、ユーザーが作成するルールおよびルール セットや、ルール内で指定する VIB およびイメージ プロファイルが含まれます。
ベスト プラクティスは、4 つのイメージ プロファイルに十分な容量と、ある程度の余分な空き容量を確保するために、2 GB を割り当てることです。1 つのイメージ プロファイルにつき、約 400 MB が必要です。使用するイメージ プロファイル数を検討することで、vSphere Auto Deploy リポジトリ用にどの程度の容量を予約するかを決定します。
- DHCP サーバに対する管理者権限を取得します。このサーバは、起動元となるネットワーク セグメントを管理します。環境内の既存の DHCP サーバを使用することも、DHCP サーバをインストールすることもできます。vSphere Auto Deploy の設定では、`gpxelinux.0` ファイル名を、UEFI の場合は `snponly64.efi.vmw-hardwired` に、BIOS の場合は `undionly.kpxe.vmw-hardwired` に置き換えます。DHCP 構成の詳細については、[DHCP の構成例](#)を参照してください。

- 他の PXE ベースのデプロイ方法に関しては、ネットワークをセキュリティ保護します。vSphere Auto Deploy は SSL 経由でデータを転送することで、不正な干渉やアクセスを防ぎます。しかし、PXE ブートの間は、クライアントや vSphere Auto Deploy サーバの整合性は確認されません。
- PowerCLI コマンドレットを使用して vSphere Auto Deploy を管理する場合は、Microsoft .NET Framework 4.5 または 4.5.x および Windows PowerShell 3.0 または 4.0 が Windows マシンにインストールされていることを確認してください。『vSphere PowerCLI User's Guide』を参照してください。
- リモートの Syslog サーバを設定します。Syslog サーバの構成の詳細については、『vCenter Server およびホスト管理』ドキュメントを参照してください。リモートの Syslog サーバを使用するために起動する最初のホストを構成し、そのホストのホスト プロファイルをその他のすべてのデプロイ先ホストに適用します。オプションで、VMware vCenter Log Insight をインストールし、使用します。これにより、VMware 製品および VMware 以外の製品（仮想および物理）のログの集約と分析を行い、ログ イベントを準リアルタイムで検索および分析することができます。
- ESXi Dump Collector をインストールし、すべてのコア ダンプが ESXi Dump Collector に向けられるように最初のホストを設定して、そのホストのホスト プロファイルをその他のすべてのホストに適用します。[ESXCLI を使用した、ESXi Dump Collector の構成](#) を参照してください。
- vSphere Auto Deploy を使用したプロビジョニングを計画するホストでレガシー BIOS が使用されている場合は、vSphere Auto Deploy サーバに IPv4 アドレスがあることを確認します。レガシー BIOS ファームウェアを使用する PXE ブートは IPv4 でのみ実行できます。UEFI ファームウェアを使用する PXE ブートは IPv4 または IPv6 のいずれかで実行できます。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。
デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。
- 2 [Auto Deploy] ページで、上部のドロップダウン メニューから vCenter Server を選択します。
- 3 [Auto Deploy および Image Builder を有効にする] をクリックしてサービスをアクティブ化します。
[Image Builder] サービスがすでにアクティブな場合は、[構成] タブを選択し、[Auto Deploy サービスを有効化] をクリックします。
[ソフトウェア デポ] ページが表示されます。
- 4 TFTP サーバを構成します。
 - a [設定] タブをクリックします。
 - b [TFTP ブート ZIP のダウンロード] をクリックして TFTP 構成ファイルをダウンロードし、そのファイルを TFTP サーバがファイルを格納するディレクトリに解凍します。
 - c (オプション) プロキシ サーバを使用するには、Auto Deploy のランタイム サマリ ペインの [追加] をクリックし、テキスト ボックスにプロキシ サーバの URL を入力します。
リバース プロキシ サーバを使用して、vSphere Auto Deploy サーバに送信される要求の負荷を軽減できます。

- 5 TFTP ZIP ファイルが配置されている TFTP サーバをポイントするように、DHCP サーバを設定します。
 - a TFTP サーバの IP アドレスを DHCP オプション 66 で指定します。このオプションは、next-server と呼ばれます。
 - b 起動ファイル名を DHCP オプション 67 (通常 boot-filename と呼ばれる) で指定します。UEFI の場合は snponly64.efi.vmw-hardwired、BIOS の場合は undionly.kpxe.vmw-hardwired です。
- 6 vSphere Auto Deploy を使用してプロビジョニングする各ホストを、製造元の指示に従って、ネットワークブートまたは PXE ブートに設定します。
- 7 (オプション) サムプリント モードを使用するように環境を設定した場合は、OpenSSL の証明書 rbd-ca.crt と OpenSSL のプライベート キー rbd-ca.key を独自の証明書とキーのファイルに置き換えると、独自の認証局 (CA) を使用できます。
 ファイルは /etc/vmware-rbd/ssl/ にあります。
 vCenter Server では、デフォルトで VMware Certificate Authority (VMCA) が使用されます。

結果

vSphere Auto Deploy 用に設定されている ESXi ホストを起動すると、ホストは DHCP サーバに接続し、その接続は vSphere Auto Deploy サーバに転送されます。vSphere Auto Deploy サーバは、アクティブなルールセットで指定されたイメージ プロファイルを使用してホストをプロビジョニングします。

次のステップ

- [Auto Deploy サービス] のデフォルト構成プロパティを変更できます。詳細については、vCenter Server およびホストの管理ドキュメントで vCenter Server の設定について参照してください。
- [Image Builder サービス] のデフォルト構成プロパティを変更できます。詳細については、『vCenter Server およびホストの管理』で vCenter Server の設定について参照してください。
- イメージ プロファイル、ホスト プロファイル (オプション)、ホストの場所またはスクリプト バンドルをホストに割り当てるルールを定義します。PowerCLI コマンドレットを使用する vSphere Auto Deploy の管理については、[PowerCLI コマンドレットを使用した vSphere Auto Deploy の管理セクション](#)を参照してください。vSphere Client を使用する vSphere Auto Deploy の管理については、[vSphere Client を使用した vSphere Auto Deploy の管理セクション](#)を参照してください。
- (オプション) リファレンス ホストとしてプロビジョニングする最初のホストを構成します。ターゲット ホスト同士で共有するストレージ、ネットワーク、およびその他の設定を使用します。リファレンス ホスト用のホスト プロファイルを作成し、テスト済みのイメージ プロファイルとホスト プロファイルの両方をターゲット ホストに割り当てるルールを記述します。
- (オプション) vSphere Auto Deploy が既存のパーティションを上書きするようにする場合は、自動パーティショニングを行うようにリファレンス ホストを設定し、リファレンス ホストのホスト プロファイルをほかのホストに適用します。[自動パーティショニングのためのリファレンス ホストの構成](#)を参照してください。
- (オプション) ホストに固有の情報を構成する場合は、ユーザー入力を求めるプロンプトを表示するように、リファレンス ホストのホスト プロファイルを設定します。ホストのカスタマイズの詳細については、『vSphere のホスト プロファイル』ドキュメントを参照してください。

vSphere Auto Deploy コマンドレットの使用

vSphere Auto Deploy コマンドレットは、Microsoft PowerShell コマンドレットとして実装され、PowerCLI に含まれます。vSphere Auto Deploy コマンドレットのユーザーは、すべての PowerCLI 機能を利用できます。

PowerShell に精通しているユーザーは、ほかの PowerShell コマンドレットと同じように vSphere Auto Deploy コマンドレットを使用することができます。PowerShell および PowerCLI に慣れていない場合は、次のヒントが役立つ場合があります。

コマンドレット、パラメータ、およびパラメータの値は、PowerCLI のシェルに入力できます。

- `Get-Help cmdlet_name` を実行すると、任意のコマンドレットのヘルプを表示することができます。
- PowerShell は大文字と小文字を区別しないことに注意してください。
- コマンドレット名やパラメータ名には、タブ補完を使用します。
- `Format-List` または `Format-Table` (短縮形式は `fl` または `ft`) を使用して、任意の変数およびコマンドレットの出力の形式を整えることができます。詳細を確認する場合は、`Get-Help Format-List` コマンドレットを実行してください。

名前によるパラメータの指定

パラメータは多くの場合名前で渡すことができ、スペースや特殊文字を含むパラメータの値は二重引用符で囲むことができます。

```
Copy-DeployRule -DeployRule testrule -ReplaceItem MyNewProfile
```

vCenter Server のインストールとセットアップドキュメントのほとんどの例では、パラメータは名前で渡されません。

オブジェクトによるパラメータの指定

スクリプト処理や自動化を実施する場合は、パラメータをオブジェクトとして渡すことができます。パラメータをオブジェクトとして渡すことは、複数のオブジェクトを返す cmdlet や、単一のオブジェクトを返すコマンドレットで便利です。次の例を想定します。

- 1 ホストのルール セットのコンプライアンス情報をカプセル化するオブジェクトを、変数にバインドします。

```
$str = Test-DeployRuleSetCompliance MyEsxi42
```

- 2 オブジェクトの `itemlist` プロパティを表示して、ルール セットの内容と現在ホストが使用している内容の違いを確認します。

```
$str.itemlist
```

- 3 `Repair-DeployRuleSetCompliance` コマンドレットを変数とともに使用して、変更されたルール セットを使用するようにホストを修正します。

```
Repair-DeployRuleSetCompliance $str
```

この例では、次にホストを起動するときにホストが修復されます。

バルク ライセンス供与の設定

vSphere Client または ESXi Shell を使用すると、個別のライセンス キーを指定できます。または、PowerCLI コマンドレットを使用して、バルク ライセンス供与を設定することができます。バルク ライセンス供与はすべての ESXi ホストで行えますが、vSphere Auto Deploy を使用してプロビジョニングされるホストで特に有用です。

vSphere Client を介したライセンス キーの割り当てと PowerCLI コマンドレット機能を使用した別の方法によるライセンスの割り当て

vSphere Client によるライセンス キーの割り当て

ホストを vCenter Server システムに追加するとき、または、ホストが vCenter Server システムによって管理されるときに、ライセンス キーをホストに割り当てることができます。

LicenseDataManager PowerCLI を使用したライセンス キーの割り当て

一連のホストに追加する一連のライセンス キーを指定できます。ライセンス キーは vCenter Server データベースに追加されます。ホストが vCenter Server システムに追加または再接続されるたびに、ホストにライセンス キーが割り当てられます。PowerCLI を介して割り当てられたライセンス キーは、デフォルトのライセンス キーとして扱われます。ライセンスのないホストが追加または再接続されると、このホストにはデフォルトのライセンス キーが割り当てられます。すでにライセンスが供与されたホストの場合は、ホストのライセンス キーが保持されます。

次の例では、データセンターのすべてのホストにライセンスを割り当てています。ホストおよびクラスタにもライセンスを関連付けることができます。

次の例は、PowerShell の変数の使用方法についての知識がある、上級 PowerCLI ユーザー向けです。

前提条件

vSphere Auto Deploy 用にシステムを準備。

手順

- 1 PowerCLI セッションで、使用する vCenter Server システムに接続し、関連付けられたライセンス マネージャを変数にバインドします。

```
Connect-VIServer -Server 192.XXX.X.XX -User username -Password password
$licenseDataManager = Get-LicenseDataManager
```

- 2 バルク ライセンス供与機能を使用する対象となるホストが配置されているデータセンターを取得するコマンドレットを実行します。

```
$hostContainer = Get-Datacenter -Name Datacenter-X
```

また、クラスタを取得するコマンドレット（そのクラスタ内のすべてのホストにバルク ライセンス供与を使用）または、フォルダを取得するコマンドレット（そのフォルダ内のすべてのホストにバルク ライセンス供与を使用）を実行します。

- 3 LicenseData オブジェクトと LicenseKeyEntry オブジェクトを、関連するタイプの ID とライセンス キーで作成します。

```
$licenseData = New-Object VMware.VimAutomation.License.Types.LicenseData
$licenseKeyEntry = New-Object VMware.VimAutomation.License.Types.LicenseKeyEntry
$licenseKeyEntry.TypeId = "vmware-vmphere"
$licenseKeyEntry.LicenseKey = "XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX"
```

- 4 手順 3 で作成した LicenseData オブジェクトの LicenseKeys 属性を LicenseKeyEntry オブジェクトに関連付けます。

```
$licenseData.LicenseKeys += $licenseKeyEntry
```

- 5 データセンターのライセンス データを LicenseData オブジェクトを使用して更新し、ライセンスがホストのコンテナに関連付けられていることを確認します。

```
$licenseDataManager.UpdateAssociatedLicenseData($hostContainer.Uid, $licenseData)
$licenseDataManager.QueryAssociatedLicenseData($hostContainer.Uid)
```

- 6 vSphere Auto Deploy を使用して 1 つ以上のホストをプロビジョニングし、それらのホストを、ライセンス データを割り当てたデータセンターまたはクラスタに割り当てます。
- 7 vSphere Client を使用して、ホストが正しくデフォルトのライセンス XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX に割り当てられていることを確認します。

結果

これで、データセンターに割り当てたすべてのホストに自動的にライセンスが供与されます。

PowerCLI での vSphere Auto Deploy の仕組み

PowerCLI コマンドレットを使用して、ESXi ホストを vCenter Server ターゲット上のさまざまなオブジェクトに関連付けるルールを作成して、vSphere Auto Deploy を管理できます。

PowerCLI コマンドレットを使用して、ホストをイメージ プロファイル、ホスト プロファイル、カスタム スクリプト、および vCenter Server ターゲット上の場所に関連付けるルールを作成して、vSphere Auto Deploy を管理できます。また、ルールのコンプライアンスをテストし、コンプライアンスの問題を修復して、ホストのアップデートを実行できます。

PowerCLI コマンドレットを使用した vSphere Auto Deploy の管理

PowerCLI コマンドレットを使用して、ホストをイメージ プロファイル、ホスト プロファイル、カスタム スクリプト、および vCenter Server ターゲット上の場所に関連付けるルールを作成して、vSphere Auto Deploy を管理できます。また、ルールのコンプライアンスをテストし、コンプライアンスの問題を修復して、ホストのアップデートを実行できます。

PowerCLI を使用した vSphere Auto Deploy プロセスの概要

vSphere Auto Deploy の使用を開始するには、vSphere Auto Deploy のしくみについて学ぶこと、vSphere Auto Deploy サーバをインストールすること、PowerCLI をインストールすること、ホストをプロビジョニングする PowerCLI ルールを記述すること、ユーザーが指定するイメージ プロファイルで起動されるホストをパワーオン

することが必要です。イメージ プロファイル、ホスト プロファイル、および vCenter Server の場所をカスタマイズできます。

初めて vSphere Auto Deploy 環境を設定する際に役立つ順を追った説明については、[vSphere Auto Deploy の vSphere PowerCLI シナリオ](#) を参照してください。

vSphere Auto Deploy を使用して環境内にホストを正しくプロビジョニングするには、次の手順に従います。

- 1 vCenter Server をデプロイします。

vSphere Auto Deploy サーバが含まれています。

- 2 vSphere Auto Deploy サービスの起動タイプを構成します。

[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。

- 3 vSphere Auto Deploy と vSphere ESXi Image Builder コマンドレットを含む PowerCLI をインストールします。

[vSphere ESXi Image Builder の構成](#)、[vSphere Auto Deploy コマンドレットの使用](#)、および [VMware.Image Builder コマンドレットの使用](#) を参照してください。

- 4 ホストにデプロイする VIB を含むイメージ プロファイルを探します。

- 多くの場合、必要なソフトウェアを含むデポを PowerCLI セッションに追加してから、それらのデポのいずれかからイメージ プロファイルを選択します。
- カスタムのイメージ プロファイルを作成する場合は、vSphere ESXi Image Builder コマンドレットを使用して既存のイメージ プロファイルのクローンを作成し、そのクローンにカスタム VIB を追加します。カスタムのイメージ プロファイルを PowerCLI セッションに追加します。

vSphere ESXi Image Builder をカスタマイズ用に使用する必要があるのは、VIB の追加または削除が必要な場合のみです。ほとんどの場合は、VMware がイメージ プロファイルをホストするデポを PowerCLI セッションに URL として追加できます。

- 5 PowerCLI セッションを開始し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

- 6 New-DeployRulePowerCLI コマンドレットを使用して、1つのホスト、パターンによって指定される複数のホスト、またはすべてのホストにイメージ プロファイルを割り当てるルールを記述します。

```
New-DeployRule -Name "testrule" -Item image-profile -AllHosts
```

[イメージ プロファイルのホストへの割り当て](#)を参照してください。

注： vSphere Auto Deploy は、MAC アドレスと IP アドレスのマッピングが DHCP に固定されている (DHCP 予約とも呼ばれる) ホストのプロビジョニング用に最適化されています。固定 IP アドレスを使用する場合は、ホスト プロファイルを設定して、ホストをカスタマイズするためのプロンプトを表示する必要があります。詳細については、『vSphere のホスト プロファイル』ドキュメントを参照してください。

- 7 プロビジョニングするホストをパワーオンします。

- 8 プロビジョニングしたホストを、ホスト プロファイルのリファレンス ホストとして設定します。

リファレンス ホストの Syslog 設定、ファイアウォール設定、ストレージ、ネットワークなどを指定できます。

- 9 プロビジョニングしたホストを、ホスト プロファイルのリファレンス ホストとして設定します。

リファレンス ホストの Syslog 設定、ファイアウォール設定、ストレージ、ネットワークなどを指定できます。
[vSphere Auto Deploy のリファレンス ホストの設定](#)を参照してください。

- 10 リファレンス ホストのホスト プロファイルを作成してエクスポートします。

ホスト プロファイル ドキュメントを参照してください。

- 11 ホスト プロファイルを使用して複数のホストをプロビジョニングするには、Copy-DeployRule コマンドレットを使用して、以前に作成されたルールを編集します。

イメージ プロファイルだけでなく、ホスト プロファイル、vCenter Server の場所、カスタム スクリプト バンドルも割り当てるようにルールを変更します。

```
Copy-DeployRule -DeployRule "testrule" -ReplaceItem
my_host_profile_from_reference_host,my_target_cluster
-ReplacePattern "ipv4=192.XXX.1.10-192.XXX.1.20"
```

ここで *my_host_profile_from_reference_host* はリファレンス ホストのプロファイルの名前、*my_target_cluster* はターゲット クラスタの名前です。

- 12 テストと修復のコンプライアンス処理を実行し、ホストを修正します。

[ルールのコンプライアンスのテストおよび修復](#) を参照してください。

- 13 プロビジョニングしたホストが、次の要件を満たしていることを確認します。

- 各ホストが vCenter Server システムに接続されている。
 - ホストがメンテナンス モードではない。
 - ホストにコンプライアンスのエラーがない。
 - ユーザー入力が必要なホスト プロファイルを持つ各ホストに、最新のカスタマイズ情報が提供されている
- すべてのホストが要件を満たすまで、ホストの関連付けとコンプライアンスの問題を修正して、ホストを再起動します。

起動プロセスの概要、最初の起動と 2 回目以降の起動の違い、およびホストのカスタマイズを使用した場合の概要については、を参照してください。

vSphere Auto Deploy コマンドレットの使用

vSphere Auto Deploy コマンドレットは、Microsoft PowerShell コマンドレットとして実装され、PowerCLI に含まれます。vSphere Auto Deploy コマンドレットのユーザーは、すべての PowerCLI 機能を利用できます。

PowerShell に精通しているユーザーは、ほかの PowerShell コマンドレットと同じように vSphere Auto Deploy コマンドレットを使用することができます。PowerShell および PowerCLI に慣れていない場合は、次のヒントが役立つ場合があります。

コマンドレット、パラメータ、およびパラメータの値は、PowerCLI のシェルに入力できます。

- `Get-Help cmdlet_name` を実行すると、任意のコマンドレットのヘルプを表示することができます。
- PowerShell は大文字と小文字を区別しないことに注意してください。

- コマンドレット名やパラメータ名には、タブ補完を使用します。
- `Format-List` または `Format-Table` (短縮形式は `fl` または `ft`) を使用して、任意の変数およびコマンドレットの出力の形式を整えることができます。詳細を確認する場合は、`Get-Help Format-List` コマンドレットを実行してください。

名前によるパラメータの指定

パラメータは多くの場合名前で渡すことができ、スペースや特殊文字を含むパラメータの値は二重引用符で囲むことができます。

```
Copy-DeployRule -DeployRule testrule -ReplaceItem MyNewProfile
```

vCenter Server のインストールとセットアップドキュメントのほとんどの例では、パラメータは名前で渡されません。

オブジェクトによるパラメータの指定

スクリプト処理や自動化を実施する場合は、パラメータをオブジェクトとして渡すことができます。パラメータをオブジェクトとして渡すことは、複数のオブジェクトを返す `cmdlet` や、単一のオブジェクトを返すコマンドレットで便利です。次の例を想定します。

- 1 ホストのルール セットのコンプライアンス情報をカプセル化するオブジェクトを、変数にバインドします。

```
$str = Test-DeployRuleSetCompliance MyEsxi42
```

- 2 オブジェクトの `itemlist` プロパティを表示して、ルール セットの内容と現在ホストが使用している内容の違いを確認します。

```
$str.itemlist
```

- 3 `Repair-DeployRuleSetCompliance` コマンドレットを変数とともに使用して、変更されたルール セットを使用するようにホストを修正します。

```
Repair-DeployRuleSetCompliance $str
```

この例では、次にホストを起動するときにホストが修復されます。

vSphere Auto Deploy PowerCLI コマンドレットの概要

PowerCLI に含まれる一連の PowerCLI コマンドレットを使用して、イメージ プロファイルおよびホスト プロファイルをホストに割り当てるルールを指定します。

PowerCLI に慣れていない場合は、PowerCLI のマニュアルを読み、[vSphere Auto Deploy コマンドレットの使用](#)を参照してください。PowerShell プロンプトで任意のコマンドのヘルプを表示できます。

- 基本的なヘルプ: `Get-Help cmdlet_name`
- 詳細なヘルプ: `Get-Help cmdlet_name -Detailed`

注: vSphere Auto Deploy コマンドレットを実行する場合は、コマンド ラインですべてのパラメータを指定して実行します。対話モードでのパラメータの提供は推奨されません。

表 4-18. ルール エンジンの PowerCLI コマンドレット

コマンド	説明
Get-DeployCommand	vSphere Auto Deploy コマンドレットのリストを返します。
New-DeployRule	指定された項目およびパターンで新しいルールを作成します。
Set-DeployRule	指定された項目およびパターンで既存のルールを更新します。ルール セットに含まれるルールは更新できません。
Get-DeployRule	指定された名前のルールを取得します。
Copy-DeployRule	既存のルールのクローンを作成して更新します。
Add-DeployRule	作業ルール セットに 1 つ以上のルールを追加し、デフォルトで、アクティブなルール セットにも追加します。作業ルール セットにのみルールを追加するには、NoActivate パラメータを使用します。
Remove-DeployRule	作業ルール セットおよびアクティブなルール セットから、1 つ以上のルールを除外します。ルールを完全に削除するには、-Delete パラメータとともにこのコマンドを実行します。
Set-DeployRuleset	作業ルール セット内のルールのリストを明示的に設定します。
Get-DeployRuleset	現在の作業ルール セットまたは現在アクティブなルール セットを取得します。
Switch-ActiveDeployRuleset	すべての新しい要求がこのルール セットで評価されるようにするために、ルール セットをアクティブ化します。
Get-VMHostMatchingRules	パターンに一致するルールを取得します。たとえば、1 つまたは複数のホストに適用されるすべてのルールを取得できます。デバッグの前に、このコマンドレットを使用します。
Test-DeployRulesetCompliance	指定されたホストに関連付けられた項目が、アクティブなルール セットに準拠しているかどうかを確認します。
Repair-DeployRulesetCompliance	この cmdlet は、Test-DeployRulesetCompliance の出力に基づいて、イメージ プロファイル、ホスト プロファイル、および vCenter Server インベントリ内の各ホストの場所を更新します。この cmdlet は、イメージ プロファイルの適用、ホスト プロファイルの適用、または、vCenter Server システム上の事前に指定されたフォルダまたはクラスターへのホストの移動を行う場合があります。
Apply-EsxImageProfile	指定されたイメージ プロファイルを指定されたホストに関連付けます。
Get-VMHostImageProfile	指定されたホストによって使用されているイメージ プロファイルを取得します。このコマンドレットは、vSphere ESXi Image Builder の Get-EsxImageProfile コマンドレットとは異なります。
Repair-DeployImageCache	このコマンドレットは、vSphere Auto Deploy のイメージ キャッシュが誤って削除された場合にのみ使用します。
Get-VMHostAttributes	vSphere Auto Deploy サーバがルールを評価する際に使用されたホストの属性を取得します。
Get-DeployMachineIdentity	vCenter Server 内の ESXi ホストを物理マシンに論理的にリンクするときに vSphere Auto Deploy が使用する文字列の値を返します。

表 4-18. ルール エンジンの PowerCLI コマンドレット (続き)

コマンド	説明
Set-DeployMachineIdentity	vCenter Server データベース内のホスト オブジェクトを、物理マシンに論理的にリンクします。この cmdlet は、ルールを指定せずにホストを追加する場合に使用します。
Get-DeployOption	vSphere Auto Deploy のグローバル構成オプションを取得します。このコマンドレットは現在、vlan-id オプションをサポートしており、vSphere Auto Deploy でプロビジョニングされたホストの ESXi 管理ネットワークでデフォルトの VLAN ID を指定します。vSphere Auto Deploy は、ホスト プロファイルなしでホストが起動した場合のみ、この値を使用します。
Set-DeployOption	グローバル構成オプションの値を設定します。ESXi 管理ネットワークのデフォルトの VLAN ID の設定では、vlan-id オプションは現在サポートされています。
Add-ProxyServer	プロキシ サーバを vSphere Auto Deploy データベースに追加します。IPv4 アドレスまたは IPv6 アドレスを指定するための -Address パラメータを使用して、コマンドを実行します。アドレスにはポート番号を含めることができます。
List-ProxyServer	現在 vSphere Auto Deploy に登録されているプロキシ サーバを一覧表示します。
Delete-ProxyServer	vSphere Auto Deploy に登録されているプロキシ サーバのリストから 1 台以上のプロキシ サーバを削除します。プロキシ サーバのリストから -id パラメータを使用してコマンドを実行したり、-Address パラメータを使用して、削除するプロキシ サーバの IPv4 アドレスまたは IPv6 アドレスを指定することによってコマンドを実行できます。
Add-ScriptBundle	1 つ以上のスクリプト バンドルを vSphere Auto Deploy サーバに追加します。
Get-ScriptBundle	vSphere Auto Deploy サーバで使用可能なスクリプト バンドルおよびそれらに含まれるスクリプトのリストを取得します。
Remove-ScriptBundle	vSphere Auto Deploy からスクリプト バンドルを削除します。vSphere バージョン 6.7 以降で使用できます。
Get-CustomCertificate	AutoDeploy にアップロードされたカスタム ホスト証明書を取得します。-HostId [MAC_Address BIOS_UUID] パラメータを指定してコマンドを実行する必要があります。カスタム証明書を初めて追加するとき、このコマンドレットによって返されて表示される証明書はありません。
List-CustomCertificates	Auto Deploy によって使用されるすべてのカスタム ホスト証明書に関する情報を取得します。リストには、証明書の名前、Host ID、Associated Host Name の詳細が表示されます。これには、Auto Deploy サーバの vCenter Server の名前が反映されています。

表 4-18. ルール エンジンの PowerCLI コマンドレット (続き)

コマンド	説明
Add-CustomCertificate	カスタム証明書を VMware Endpoint Certificate Store に追加し、ESXi ホストに関連付けます。ホストが再起動すると証明書が有効になります。Get-CustomCertificate コマンドレットを使用して、カスタム ホスト証明書キーを取得できます。-HostId [MAC_Address BIOS_UUID] パラメータを指定してコマンドを実行することにより、-Key [file:///path/to/key.key] と -Cert [file:///path/to/cert.crt] を指定して証明書をホストに関連付けることができます。このコマンドレットを使用するには、vCenter Server のルート フォルダに対する AutoDeploy.Rule.Create 権限が必要です。
Remove-CustomCertificate	一連のカスタム ホスト証明書を Auto Deploy から削除します。証明書エントリがデータベースから削除され、証明書ファイルがファイルストアから削除されます。すでにカスタム証明書を使用して起動しているホストは、新しい証明書を受け取るために再起動する必要があります。-Cert パラメータまたは -HostId パラメータの少なくとも 1 つを指定する必要があります。このコマンドレットを使用するには、vCenter Server のルート フォルダに対する AutoDeploy.Rule.Create 権限が必要です。

イメージ プロファイルのホストへの割り当て

ホストをプロビジョニングする前に、vSphere Auto Deploy を使用してプロビジョニングする各ホストにイメージ プロファイルを割り当てるルールを作成する必要があります。

vSphere Auto Deploy の拡張ルールは、ESXCLI プラグイン パス、jumpstart プラグイン パスなど、事前定義された特定の場所のファイルのみを CommunitySupported レベルの VIB に含めることを強制します。別の場所にある VIB をイメージ プロファイルに追加すると、警告が表示されます。force オプションを使用して警告をオーバーライドできます。

ルールに違反する CommunitySupported レベルの VIB を含むイメージ プロファイルで New-DeployRule コマンドレットを呼び出す場合は、イメージ プロファイルを追加する前に \$DeployNoSignatureCheck = \$true を設定します。これを設定すると、システムは署名の検証を無視し、拡張ルールのチェックを実行しません。

注： CommunitySupported レベルの VIB を含むイメージ プロファイルは、本番システムではサポートされません。

手順

- 1 PowerCLI セッションで Connect-VIServer コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 パブリックなソフトウェア デポの場所を特定するか、vSphere ESXi Image Builder を使用してカスタムのイメージ プロファイルを定義します。

- 3 Add-EsxSoftwareDepot を実行して、イメージ プロファイルが含まれるソフトウェア デポを PowerCLI セッションに追加します。

デポのタイプ	コマンドレット
リモート デポ	Add-EsxSoftwareDepot <i>depot_url</i> を実行します。
ZIP ファイル	a ZIP ファイルをローカル ファイル パスにダウンロードします。 b Add-EsxSoftwareDepot C:\ <i>file_path</i> \my_offline_depot.zip を実行します。

- 4 Get-EsxImageProfile cmdlet を実行して、デポ内の使用するイメージ プロファイルを探します。

デフォルトでは、ESXi デポには、VMware Tools を含んだ 1 つの基本イメージ プロファイル（名前に文字列 standard を含む）と、VMware Tools を含まない 1 つの基本イメージ プロファイルが格納されています。

- 5 IP アドレスの範囲などの特定の属性を持つホストに、イメージ プロファイルを割り当てるためのルールを定義します。

```
New-DeployRule -Name "testrule" -Item "My Profile25" -Pattern "vendor=Acme,Zven",
"ipv4=192.XXX.1.10-192.XXX.1.20"
```

名前にスペースが含まれる場合は二重引用符の使用が必須ですが、それ以外の場合は任意です。項目をすべてのホストに適用するには、パターンの代わりに -AllHosts を指定します。

コマンドレットは testrule という名前のルールを作成します。このルールは、My Profile25 という名前のイメージ プロファイルを、Acme または Zven というベンダーの、指定された範囲内の IP アドレスを持つすべてのホストに割り当てます。

- 6 ルール セットにルールを追加します。

```
Add-DeployRule testrule
```

デフォルトでは、ルールは作業ルール セットとアクティブなルール セットの両方に追加されます。NoActivate パラメータを使用すると、作業ルール セットはアクティブなルール セットになりません。

結果

ホストが iPXE から起動する場合、ホストはマシンの属性をコンソールにレポートします。デプロイのルールを記述する際には、同じ形式の属性を使用します。

```
*****
* Booting through VMware AutoDeploy...
*
* Machine attributes:
* . asset=No Asset Tag
* . domain=vmware.com
* . hostname=myhost.mycompany.com
* . ipv4=XX.XX.XXX.XXX
* . mac=XX:XX:XX:XX:XX:XX
* . model=MyVendorModel
* . oemstring=Product ID: XXXXXX-XXX
```

```
* . serial=XX XX XX XX XX XX...
* . uuid=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX
* . vendor=MyVendor
*****
```

次のステップ

- vSphere Auto Deploy ですでにプロビジョニングされたホストについては、コンプライアンス テストおよび修復を実施して、新しいイメージ プロファイルでそれらのホストがプロビジョニングされるようにします。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストを起動し、新しいイメージ プロファイルを使用してそれらのホストがプロビジョニングされるようにします。

ルールの記述とホストへのホスト プロファイルの割り当て

vSphere Auto Deploy は、1 台以上の ESXi ホストにホスト プロファイルを割り当てることができます。

多くの場合、ホスト プロファイルを明示的に指定する代わりに、ホストをクラスタに割り当てます。ホストは、割り当てられたクラスタのホスト プロファイルを使用します。

前提条件

ホスト プロファイルには、ストレージ構成、ネットワーク構成、またはホストのその他の特性に関する情報が含まれる場合があります。クラスタにホストを追加する場合は、そのクラスタのホスト プロファイルが使用されます。

- PowerCLI および前提となるすべてのソフトウェアをインストールします。詳細については、[vCenter Server Installation and Setup](#) を参照してください。
- 使用するホスト プロファイルをエクスポートします。

手順

- 1 PowerCLI セッションで `Connect-VIServer` コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 vSphere Client を使用して、使用する設定内容をホストに設定し、そのホストでホスト プロファイルを作成します。
- 3 `Get-VMhostProfile` PowerCLI コマンドレットを実行して、ホスト プロファイルの作成元となる ESXi ホストを渡すことで、ホスト プロファイルの名前を探します。
- 4 PowerCLI プロンプトで、特定の属性 (IP アドレスの範囲など) を備えたホストをホスト プロファイルに割り当てるルールを定義します。

```
New-DeployRule -Name "testrule2" -Item my_host_profile -Pattern "vendor=Acme,Zven",
"ipv4=192.XXX.1.10-192.XXX.1.20"
```

指定された属性を持つすべてのホストに、指定された項目が割り当てられます。この例は、`testrule2` という名前のルールを指定します。このルールは、指定された範囲内の IP アドレスを持ち、メーカーが Acme または Zven のすべてのホストに、指定されたホスト プロファイル `my_host_profile` を割り当てます。

5 ルール セットにルールを追加します。

```
Add-DeployRule testrule2
```

デフォルトでは、作業ルール セットがアクティブなルール セットになります。また、ルール セットへのすべての変更は、ルールを追加した時にアクティブになります。NoActivate パラメータを使用すると、作業ルール セットはアクティブなルール セットになりません。

次のステップ

- vSphere Auto Deploy によってプロビジョニング済みのホストを新しいホスト プロファイルに割り当てるときには、それらのホスト上でコンプライアンス テストおよび修復処理を実行します。詳細については、[ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストをパワーオンし、ホスト プロファイルを使用してそれらのホストをプロビジョニングします。

ルールの記述とフォルダまたはクラスタへのホストの割り当て

vSphere Auto Deploy は、ホストをフォルダまたはクラスタに割り当てることができます。vSphere Auto Deploy は、ホストの起動時にそのホストを vCenter Server 上の指定された場所に追加します。クラスタに割り当てられたホストは、クラスタのホスト プロファイルを継承します。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#) を参照してください。
- 選択したフォルダがデータセンター内またはクラスタ内に存在することを確認します。ホストはスタンドアロンの最上位のフォルダに割り当ててはできません。

手順

- 1 PowerCLI セッションで `Connect-VIServer` コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 IP アドレスの範囲などの特定の属性を持つホストを、フォルダまたはクラスタに割り当てるときのルールを定義します。

```
New-DeployRule -Name testrule3 -Item "my folder" -Pattern "vendor=Acme,Zven",
"ipv4=192.XXX.1.10-192.XXX.1.20"
```

この例では、フォルダを名前です。これ以外に、Get-Folder、Get-Cluster、または Get-Datacenter コマンドレットで取得するフォルダ、クラスタ、またはデータセンター オブジェクトを渡すこともできます。

3 ルール セットにルールを追加します。

```
Add-DeployRule testrule3
```

デフォルトでは、作業ルール セットがアクティブなルール セットになります。また、ルール セットへのすべての変更は、ルールを追加した時にアクティブになります。NoActivate パラメータを使用すると、作業ルール セットはアクティブなルール セットになりません。

次のステップ

- vSphere Auto Deploy によってすでにプロビジョニングされているホストを新しいフォルダまたはクラスタの場所に割り当てるには、コンプライアンスのテストと修復を実行します。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストを起動し、指定された vCenter Server の場所にそれらのホストを追加します。

カスタム スクリプトの実行によるステートレス システムの構成

vSphere Auto Deploy を使用して、カスタム スクリプトを vSphere Auto Deploy ルールに関連付けることで、1 台以上のホストを構成できます。

スクリプトは、ホストの最初の ESXi の起動ワークフロー後に、アルファベット順に実行されます。

前提条件

- vSphere Auto Deploy ルールに関連付けるスクリプト バンドルの形式が .tgz で、最大サイズが 10 MB、Python または BusyBox ash スクリプト言語で記述されていることを確認します。

手順

- 1 PowerCLI セッションで Connect-VIServer コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 Add-ScriptBundle コマンドレットを実行して、必要なスクリプトを含むスクリプト バンドルを vSphere Auto Deploy インベントリに追加します。

```
Add-ScriptBundle c:/temp/MyScriptBundle.tgz
```

.tgz 拡張子を持たないスクリプト バンドルの名前は、スクリプト バンドル項目の名前識別子またはオブジェクトです。Add-ScriptBundle コマンドレットと -Update パラメータを使用して、既存のスクリプト バンドルを更新できます。

- 3 (オプション) Get-ScriptBundle コマンドレットを実行して、スクリプト バンドルが vSphere Auto Deploy インベントリに追加されていることを確認します。
- 4 IP アドレスの範囲などの特定の属性を持つホストを、スクリプト バンドルに割り当てるためのルールを定義します。

```
New-DeployRule -Name "testrule4" -Item "MyScriptBundle" -Pattern "vendor=Acme,Zven",
"ipv4=192.XXX.1.10-192.XXX.1.20"
```

名前にスペースが含まれる場合は二重引用符の使用が必須ですが、それ以外の場合は任意です。項目をすべてのホストに適用するには、パターンの代わりに `-AllHosts` を指定します。

`testrule4` という名前のルールを作成します。このルールは、My Script Bundle という名前のスクリプト バンドルを、Acme または Zven というベンダーの、指定された範囲内の IP アドレスを持つすべてのホストに割り当てます。スクリプト バンドルの名前識別子、または Get-ScriptBundle コマンドレットによって返されたオブジェクトを使用して、ルールに関連付けるスクリプト バンドルを識別できます。

- 5 ルール セットにルールを追加します。

```
Add-DeployRule testrule4
```

デフォルトでは、ルールは作業ルール セットとアクティブなルール セットの両方に追加されます。NoActivate パラメータを使用すると、作業ルール セットはアクティブなルール セットになりません。

次のステップ

- vSphere Auto Deploy ですでにプロビジョニングされたホストについては、コンプライアンス テストおよび修復を実施して、新しいスクリプトでそれらのホストがプロビジョニングされるようにします。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストを起動し、新しいスクリプトを使用してそれらのホストがプロビジョニングされるようにします。

ルールのコンプライアンスのテストおよび修復

vSphere Auto Deploy ルール セットの変更は自動的に更新されないため、新しいルールまたは変更されたルールのコンプライアンスと修復を適宜テストします。

前提条件

vSphere Auto Deploy のルール セットにルールを追加したり、1 つ以上のルールを変更したりする場合、ホストは自動的にアップデートされません。ユーザーがルールのコンプライアンスのテストと修復を行った場合にのみ、vSphere Auto Deploy は新しいルールを適用します。

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#) を参照してください。
- インフラストラクチャに vSphere Auto Deploy を使用してプロビジョニングされた ESXi ホストが 1 つ以上含まれており、PowerCLI をインストールしたホストがそれらの ESXi ホストにアクセスできることを確認します。

手順

- 1 PowerCLI セッションで `Connect-VIServer` コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 PowerCLI を使用して、現在使用可能な vSphere Auto Deploy ルールを確認します。

```
Get-DeployRule
```

システムは、該当するルールと、それに関連する項目およびパターンを返します。

- 3 使用可能なルールの 1 つを変更します。

たとえば、イメージ プロファイルおよびルールの名前を変更することができます。

```
Copy-DeployRule -DeployRule testrule -ReplaceItem MyNewProfile
```

すでにアクティブなルール セットに追加されているルールは編集できません。代わりに、ルールをコピーして、変更する項目またはパターンを置き換えることはできます。

- 4 ルール セットのコンプライアンスをテストするホストにアクセスできることを確認します。

```
Get-VMHost -Name MyEsxi42
```

- 5 そのホストのルール セットのコンプライアンスをテストし、後で使用するために、戻り値を変数にバインドするコマンドレットを実行します。

```
$str = Test-DeployRuleSetCompliance MyEsxi42
```

- 6 ルール セットの内容と、ホストの構成の間の差異を確認します。

```
$str.itemlist
```

新しいルール セットのコンプライアンスをテストするホストが、アクティブなルール セットに準拠している場合は、現在のアイテムと予測されるアイテムの表が返されます。

CurrentItem	ExpectedItem
-----	-----
My Profile 25	MyNewProfile

- 7 ホストの次の起動時に、変更されたルール セットを使用するように、ホストを修復します。

```
Repair-DeployRuleSetCompliance $str
```

次のステップ

変更したルールでインベントリの場所が指定されている場合、コンプライアンスの修復時に変更内容が即座に反映されます。それ以外のすべての変更内容は、ホストを再起動することで、vSphere Auto Deploy によって新しいルールが適用され、ルール セットとホスト間のコンプライアンスが達成されます。

vSphere Auto Deploy へのキャッシング プロキシ サーバ のアドレスの登録

多数のステートレス ホストを同時に起動すると、vSphere Auto Deploy サーバに大きな負荷がかかります。vSphere Auto Deploy サーバと、vSphere Auto Deploy に登録する 1 つ以上のプロキシ サーバ間で要求のロード バランシングを行うことができます。

前提条件

手順

- 1 PowerCLI セッションで `Connect-VIServer` コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

このコマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 `Add-ProxyServer` コマンドレットを実行して、vSphere Auto Deploy にキャッシング プロキシ サーバのアドレスを登録します。

```
Add-ProxyServer -Address 'https://proxy_server_ip_address:port_number'
```

複数のプロキシ サーバを登録する場合は、コマンドレットを複数回実行します。アドレスにはポート番号を含めることができます。

- 3 (オプション) `List-ProxyServer` コマンドレットを実行して、vSphere Auto Deploy にキャッシング プロキシ サーバが登録されていることを確認します。

vSphere Client を使用した vSphere Auto Deploy タスク

vSphere Client を使用した vSphere Auto Deploy の管理

vSphere Client を使用して、ESXi ホストを vSphere Auto Deploy インベントリに追加し、ルールと ホストの関連付けを作成、監視および管理することができます。



(vSphere Client での Auto Deploy の機能拡張)

vSphere Client を使用した vSphere Auto Deploy プロセスの概要

vSphere Auto Deploy の使用を開始するには、vSphere Auto Deploy のしくみについて学ぶこと、vSphere Auto Deploy と vSphere ESXi Image Builder の vCenter Server サービスを開始すること、ホストをプロビジョニングするデプロイ ルールを作成すること、ユーザーが指定するイメージ プロファイルで起動されるホストをパワーオンすることが必要です。

vSphere Auto Deploy を使用する環境でホストをプロビジョニングするワークフローには、次のタスクが含まれます。

- 1 vCenter Server をデプロイします。

vSphere Auto Deploy サーバが含まれています。

- 2 vSphere Auto Deploy と vSphere ESXi Image Builder サービスの起動タイプを構成します。

vSphere Auto Deploy 用にシステムを準備および『vSphere ESXi Image Builder の設定』を参照してください。

- 3 ソフトウェア デポを vSphere Auto Deploy インベントリに追加またはインポートします。

ソフトウェア デポの追加またはソフトウェア デポのインポートを参照してください。

- 4 (オプション) カスタム イメージ プロファイルを作成する場合は、vSphere Client を使用してイメージ プロファイルのクローンを作成するか、イメージ プロファイルを作成します。

イメージ プロファイルのクローン作成またはイメージ プロファイルの作成を参照してください。

- 5 1つのホスト、パターンによって指定される複数のホスト、またはすべてのホストにイメージ プロファイルを割り当てるデプロイ ルールを作成します。

デプロイ ルールの作成を参照してください。

注： vSphere Auto Deploy は、MAC アドレスと IP アドレスのマッピングが DHCP に固定されている (DHCP 予約とも呼ばれる) ホストのプロビジョニング用に最適化されています。固定 IP アドレスを使用する場合は、ホスト プロファイルを設定して、ホストをカスタマイズするためのプロンプトを表示する必要があります。詳細については、『vSphere のホスト プロファイル』ドキュメントを参照してください。

- 6 プロビジョニングするホストをパワーオンします。

- 7 プロビジョニングしたホストを、ホスト プロファイルのリファレンス ホストとして設定します。

リファレンス ホストの Syslog 設定、ファイアウォール設定、ストレージ、ネットワークなどを指定できます。

- 8 リファレンス ホストからホスト プロファイルを抽出します。

ホスト プロファイル ドキュメントを参照してください。

- 9 ホスト プロファイルを使用して複数のホストをプロビジョニングするには、vSphere Client を使用して以前に作成されたルールのクローンを作成するか、編集します。

デプロイ ルールのクローンの作成またはデプロイ ルールの編集を参照してください。

- 10 新しいルールをアクティベートし、古いルールのアクティベーションを解除します。

デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替えを参照してください。

- 11 ホストの関連付けを修正し、新しいルールをホストに適用します。

非遵守ホストの修正を参照してください。

- 12 プロビジョニングしたホストが、次の要件を満たしていることを確認します。

- 各ホストが vCenter Server システムに接続されている。

- ホストがメンテナンス モードではない。
 - ホストにコンプライアンスのエラーがない。
 - ユーザー入力が必要なホスト プロファイルを持つ各ホストに、最新のカスタマイズ情報が提供されている
- すべてのホストが要件を満たすまで、ホストの関連付けとコンプライアンスの問題を修正して、ホストを再起動します。

起動プロセスの概要、最初の起動と 2 回目以降の起動の違い、およびホストのカスタマイズを使用した場合の概要については、[を参照してください](#)。

デプロイ ルールの作成

vSphere Auto Deploy を使用して ESXi ホストをプロビジョニングする前に、ホストの場所、イメージ、およびホスト プロファイルをホストに割り当てるルールを作成する必要があります。

前提条件

ESXi ホストが複数の vSphere Auto Deploy ルール基準と一致する場合は、ルール順序が考慮されます。

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- ルールにイメージ プロファイルを含める場合は、必要なソフトウェア デポがインベントリに追加されていることを確認します。[ソフトウェア デポの追加](#)または[ソフトウェア デポのインポート](#)を参照してください。

手順

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ ルール] タブで [新規デプロイ ルール] をクリックします。

[新規デプロイ ルール] ウィザードが起動します。

- 3 ウィザードの [名前とホスト] のページで、新しいルールの名前を入力します。

- 4 ルールをインベントリ内のすべてのホストに適用するか、特定のパターンに一致するホストにのみ適用するかを選択します。

1 つ以上のパターンを選択することができます。

たとえば、ルールを vCenter Single Sign-On ドメイン内の特定のホスト名を持つホスト、または特定の IPv4 範囲に一致するホストにのみ適用することができます。

- 5 ウィザードの [設定] ページで、オプションでルールにアイテムを含めることができます。

有効にしたアイテムごとに、ウィザードに新しいページが追加されます。

オプション	操作
ホストの場所	ルールの条件に一致するホストを特定の場所に追加します。
イメージ プロファイル	ルール条件に一致するホストにイメージ プロファイルを割り当てます。
ホスト プロファイル	ルール条件に一致するホストにホスト プロファイルを割り当てます。
スクリプト バンドル	ルール条件に一致するホストにスクリプト バンドルを割り当てます。

- 6 (オプション) ウィザードの [ホストの場所の選択] ページで、ルールに一致するホストの場所として、データセンター、フォルダー、またはクラスタを選択します。
- 7 (オプション) ウィザードの [イメージ プロファイルの選択] ページで、ドロップダウン メニューを使用してソフトウェア デポを選択し、リストからイメージ プロファイルを選択します。
イメージ プロファイルの許容レベル検証をしない場合は、[イメージ プロファイルの署名チェックをスキップ] チェック ボックスを選択します。
- 8 (オプション) ウィザードの [ホスト プロファイルの選択] ページで、リストからホスト プロファイルを選択します。
- 9 (オプション) ウィザードの [スクリプト バンドルの選択] ページで、リストからスクリプト バンドルを選択します。
- 10 [設定の確認] ページで、新しいルールの概要情報を確認します。

結果

[デプロイのルール] タブには新規作成されたルールがリスト表示されます。

次のステップ

- vSphere Auto Deploy ルールを有効にします。[デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え](#)を参照してください。
- vSphere Auto Deploy ルールを編集します。[イメージ プロファイルの編集](#)を参照してください。
- vSphere Auto Deploy ルールのクローンを作成します。[デプロイ ルールのクローンの作成](#)を参照してください。
- ホストの場所、イメージ プロファイル、ホスト プロファイル、および追加されたスクリプト バンドルを確認します。[ホストの関連付けの表示](#)を参照してください。
- 準拠しないホストを修正します。[非遵守ホストの修正](#)を参照してください。
- ホストのイメージ プロファイルの関連付けを変更します。[ホストのイメージ プロファイルの関連付けの編集](#)を参照してください。

デプロイ ルールのクローンの作成

vSphere Auto Deploy ルールをテンプレートとして使用し、ルールを新規作成せずにルールの一部のみを修正することができます。

[デプロイ ルールのクローン作成] ウィザードを使用して、既存の vSphere Auto Deploy ルールのクローンを作成できます。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- vSphere Auto Deploy のルールを作成します。[デプロイ ルールの作成](#)を参照してください。
- ルールにイメージ プロファイルを含める場合は、必要なソフトウェア デポがインベントリに追加されていることを確認します。[ソフトウェア デポの追加またはソフトウェア デポのインポート](#)を参照してください。

手順

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ ルール] タブで、リストからルールを選択します。

- 3 [クローン作成] をクリックします。

[デプロイ ルールのクローン作成] ウィザードが起動します。

- 4 ウィザードの [名前とホスト] のページで、新しいルールの名前を入力します。

- 5 ルールをインベントリ内のすべてのホストに適用するか、特定のパターンに一致するホストにのみ適用するかを選択します。

1 つ以上のパターンを選択することができます。

たとえば、ルールを vCenter Single Sign-On ドメイン内の特定のホスト名を持つホスト、または特定の IPv4 範囲に一致するホストにのみ適用することができます。

- 6 ウィザードの [設定] ページで、オプションでルールにアイテムを含めることができます。

有効にしたアイテムごとに、ウィザードに新しいページが追加されます。

オプション	操作
ホストの場所	ルールの条件に一致するホストを特定の場所に追加します。
イメージ プロファイル	ルール条件に一致するホストにイメージ プロファイルを割り当てます。
ホスト プロファイル	ルール条件に一致するホストにホスト プロファイルを割り当てます。
スクリプト バンドル	ルール条件に一致するホストにスクリプト バンドルを割り当てます。

7 ウィザードの [ホストの場所の選択] ページで、ルールに一致するホストの場所を選択します。

オプション	操作
クローンが作成されたルールで使用されているホストの場所を維持する場合	[同じホストの場所] チェック ボックスをオンにします。
選択したホストの新しい配置場所を選択する場合	<ol style="list-style-type: none"> [ホストの場所を参照] チェック ボックスを選択します。 データセンター、フォルダまたはクラスタをホストの配置場所を選択します。 [次へ] をクリックします。

8 ウィザードの [イメージ プロファイルの選択] ページで、イメージ プロファイルを選択します。

オプション	操作
イメージ プロファイルを変更しない場合	[同じイメージ プロファイル] チェック ボックスをオンにします。
選択したホストに新しいイメージ プロファイルを割り当てる場合	<ol style="list-style-type: none"> [イメージ プロファイルを参照] チェックボックスをオンにします。 ドロップダウン メニューからソフトウェア デポを選択します。 リストからイメージ プロファイルを選択します。 (オプション) イメージ プロファイルの許容レベル検証をしない場合は、[イメージ プロファイルの署名チェックをスキップ] チェック ボックスを選択します。

9 ウィザードの [ホスト プロファイルの選択] ページで、ホスト プロファイルを選択します。

オプション	操作
クローンが作成されたルールで使用されるホスト プロファイルを維持する場合	[同じホスト プロファイル] チェック ボックスを選択します。
選択したホストに新しいホスト プロファイルを割り当てる場合	<ol style="list-style-type: none"> [ホスト プロファイルを参照] チェック ボックスを選択します。 リストからホスト プロファイルを選択し、[次へ] をクリックします。

10 ウィザードの [スクリプト バンドルの選択] ページで、リストからスクリプト バンドルを選択します。

11 [設定の確認] ページで、新しいルールの概要情報を確認します。

次のステップ

- vSphere Auto Deploy ルールを有効にします。 [デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え](#) を参照してください。
- vSphere Auto Deploy ルールを編集します。 [イメージ プロファイルの編集](#) を参照してください。

デプロイ ルールの編集

非アクティブな Auto Deploy ルールの名前、ルールが一致するホスト、ホストの場所、イメージ プロファイル、およびホスト プロファイルを編集できます。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- vSphere Auto Deploy のルールを作成します。[デプロイ ルールの作成](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイのルール] タブで、インベントリ内のルールのリストからルールを選択し、[編集] をクリックします。

デプロイのルールを編集するダイアログ ボックスが表示されます。

- 3 (オプション) ウィザードの [名前とホスト] の画面で、ルールの新しい名前を入力します。

- 4 ルールをインベントリ内のすべてのホストに適用するか、特定のパターンに一致するホストにのみ適用するかを選択します。

1 つ以上のパターンを選択することができます。

たとえば、ルールを vCenter Single Sign-On ドメイン内の特定のホスト名を持つホスト、または特定の IPv4 範囲に一致するホストにのみ適用することができます。

- 5 ウィザードの [設定] ページで、オプションでルールにアイテムを含めることができます。

有効にしたアイテムごとに、ウィザードに新しいページが追加されます。

オプション	操作
ホストの場所	ルールの条件に一致するホストを特定の場所に追加します。
イメージ プロファイル	ルール条件に一致するホストにイメージ プロファイルを割り当てます。
ホスト プロファイル	ルール条件に一致するホストにホスト プロファイルを割り当てます。
スクリプト バンドル	ルール条件に一致するホストにスクリプト バンドルを割り当てます。

- 6 ウィザードの [ホストの場所の選択] ページで、ルールに一致するホストの場所を選択します。

オプション	操作
クローンが作成されたルールで使用されているホストの場所を維持する場合	[同じホストの場所] チェック ボックスをオンにします。
選択したホストの新しい配置場所を選択する場合	<ol style="list-style-type: none"> 1 [ホストの場所を参照] チェック ボックスを選択します。 2 データセンター、フォルダまたはクラスタをホストの配置場所を選択します。 3 [次へ] をクリックします。

7 ウィザードの [イメージ プロファイルの選択] ページで、イメージ プロファイルを選択します。

オプション	操作
イメージ プロファイルを変更しない場合	[同じイメージ プロファイル] チェック ボックスをオンにします。
選択したホストに新しいイメージ プロファイルを割り当てる場合	<ol style="list-style-type: none"> 1 [イメージ プロファイルを参照] チェックボックスをオンにします。 2 ドロップダウン メニューからソフトウェア デボを選択します。 3 リストからイメージ プロファイルを選択します。 4 (オプション) イメージ プロファイルの許容レベル検証をしない場合は、[イメージ プロファイルの署名チェックをスキップ] チェック ボックスを選択します。

8 ウィザードの [ホスト プロファイルの選択] ページで、ホスト プロファイルを選択します。

オプション	操作
クローンが作成されたルールで使用されるホスト プロファイルを維持する場合	[同じホスト プロファイル] チェック ボックスを選択します。
選択したホストに新しいホスト プロファイルを割り当てる場合	<ol style="list-style-type: none"> 1 [ホスト プロファイルを参照] チェック ボックスを選択します。 2 リストからホスト プロファイルを選択し、[次へ] をクリックします。

9 ウィザードの [スクリプト バンドルの選択] ページで、リストからスクリプト バンドルを選択します。

10 [設定の確認] ページで、新しいルールの概要情報を確認します。

次のステップ

- vSphere Auto Deploy ルールを有効にします。[デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え](#)を参照してください。
- vSphere Auto Deploy ルールのクローンを作成します。[デプロイ ルールのクローンの作成](#)を参照してください。

デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え

vSphere Auto Deploy ルールを作成した直後、ルールはアクティベーションが解除された状態です。ルールを有効にするには、アクティベーションを実行する必要があります。

ウィザードの [アクティベートして並べ替え] ページの上のリストは、アクティブなルール セットのルールを表示します。下のリストは非アクティブなルールを表示します。

前提条件

[アクティベートして並べ替え] ウィザードを使用して、ルールのアクティベーション、アクティベーションの解除、および並べ替えを実行することができます。

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- vSphere Auto Deploy のルールを作成します。[デプロイ ルールの作成](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ ルール] タブで、[ルールのアクティベート/アクティベート解除] をクリックします。

[アクティベートして並べ替え] ウィザードが表示されます。

- 3 (オプション) アクティブなルールをアクティベート解除するには、アクティブなルール リストからルールを選択し、[アクティベート解除] ボタンをクリックします。

- 4 有効にするルールを無効なルールのリストから選択し、[有効化] ボタンをクリックします。

- 5 (オプション) 有効なルール リストのルールの順番を変更するには、移動するルールを選択して、有効なルール リストの上部にある [上へ移動] または [下へ移動] をクリックします。

ルールは優先度の高い順にリストされます。たとえば、2 つ以上のルールが同じホストに適用され、それぞれのルールでホストをプロビジョニングする際に使用するホストの場所、イメージ プロファイル、およびホスト プロファイルが異なる場合は、リスト内の最上位のルールが適用されます。

- 6 (オプション) 無効なルールを有効にする前にテストするには、[有効にする前にルールをテスト] をクリックします。

- a リストからホストを選択し、[コンプライアンスの確認] をクリックすると、ホストの現在のステータスと、ルールを有効にした後に予想される変化が表示されます。

ホストがルールに準拠している場合は、ルールを有効にした後にホストを修正する必要はありません。

- b (オプション) 選択したホストをルールの有効化後に修正するには、切り替えボタンを有効にするか、[ルールの有効化後にすべてのホストの関連付けを修正] チェック ボックスをオンにしてすべてのホストを修正します。

- 7 有効なルールのリストを確認し、[OK] をクリックします。

結果

[デプロイ ルール] タブで、ルールは [ステータス] 列にアクティブとしてリストされます。

次のステップ

- ホストの場所、イメージ プロファイル、ホスト プロファイル、および追加されたスクリプト バンドルを確認します。[ホストの関連付けの表示](#)を参照してください。
- 準拠しないホストを修正します。[非遵守ホストの修正](#)を参照してください。

ホストの関連付けの表示

vSphere Auto Deploy インベントリ内の一部の ESXi ホストは、アクティブなデプロイ ルールに準拠していない可能性があるため、準拠しているかどうかを確認する必要があります。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。

- vSphere Auto Deploy のルールを作成します。[デプロイ ルールの作成](#)を参照してください。
- vSphere Auto Deploy のルールを有効にします。[デプロイ ルールのアクティベーション](#)、[アクティベーション解除](#)、および[並べ替え](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 ホスト関連付けコンプライアンスを確認します。

[ホスト関連付けコンプライアンスの確認] ウィンドウに、ホストのステータスと、ホストがアクティブなルールセットに準拠しているかどうかが表示されます。現在割り当てられているホストの場所、イメージ プロファイル、ホスト プロファイル、スクリプト バンドルと、ホストの修正後に有効になる関連付けを確認できます。

オプション	手順
単一ホストのホスト関連付けコンプライアンスを確認する場合	<ol style="list-style-type: none"> 1 [デプロイ済みのホスト] タブで、ESXi ホストを選択してください。 2 [ホスト関連付けコンプライアンスの確認] をクリックします。 3 ホストの関連付けが現在アクティブなルール セットを遵守していることを確認します。 4 (オプション) ホストを修正する場合は、[修正] をクリックします。 5 [ホスト関連付けコンプライアンスの確認] ウィンドウを閉じます。
複数ホストのホスト関連付けコンプライアンスを確認する場合	<ol style="list-style-type: none"> 1 [デプロイ済みのホスト] タブで、複数の ESXi ホストを選択します。 2 [ホスト関連付けコンプライアンスの確認] をクリックします。 3 すべての選択したホストのコンプライアンスを確認することを確定します。 4 左ペインでホストのコンプライアンス ステータスを確認します。 5 (オプション) コンプライアンス ステータスの詳細を表示するホストを選択します。 6 (オプション) ホストを選択し、[修正] をクリックします。 7 (オプション) すべてのホストを修正するには、[ルールのアクティベーション後にすべてのホストの関連付けを修正します] チェック ボックスを選択します。 8 [ホスト関連付けコンプライアンスの確認] ウィンドウを[閉じます]。

次のステップ

- 準拠しないホストを修正します。[非遵守ホストの修正](#)を参照してください。
- ホストのイメージ プロファイルの関連付けを編集します。[ホストのイメージ プロファイルの関連付けの編集](#)を参照してください。
- vSphere Auto Deploy ルールを編集します。[イメージ プロファイルの編集](#)を参照してください。

ホストのイメージ プロファイルの関連付けの編集

ホストが vSphere Auto Deploy ルールに関連付けられていない場合は、単一ホストのイメージ プロファイルの関連付けを編集できます。

前提条件

または、ルールを編集して、複数のホストのイメージ プロファイルの関連付けを変更しないようにすることもできます。

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- vSphere Auto Deploy のルールを作成します。[デプロイ ルールの作成](#)を参照してください。
- vSphere Auto Deploy のルールを有効にします。[デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え](#)を参照してください。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ済みのホスト] タブで、ESXi ホストを選択してください。

- 3 [イメージ プロファイルの関連付けの編集] をクリックします。

[イメージ プロファイルの関連付けの編集] ダイアログ ボックスが開きます。

- 4 ホストのイメージ プロファイルの関連付けを編集します。

オプション	操作
イメージ プロファイルを変更しない場合	[同じイメージ プロファイル] チェック ボックスをオンにします。
選択したホストに新しいイメージ プロファイルを割り当てる場合	<ol style="list-style-type: none"> 1 [イメージ プロファイルを参照] チェックボックスをオンにします。 2 ドロップダウン メニューからソフトウェア デボを選択します。 3 リストからイメージ プロファイルを選択します。 4 (オプション) イメージ プロファイルの許容レベル検証をしない場合は、[イメージ プロファイルの署名チェックをスキップ] チェック ボックスを選択します。

- 5 [OK] をクリックします。

結果

ページの更新後に、新しいイメージ プロファイルが [関連付けられたイメージ プロファイル] 列にリストされます。

次のステップ

- ホストの場所、イメージ プロファイル、ホスト プロファイル、および追加されたスクリプト バンドルを確認します。[ホストの関連付けの表示](#)を参照してください。
- ホストがルールに関連付けられていて、ルール内で定義したイメージ プロファイルに戻す場合は、ホストを修正します。[非遵守ホストの修正](#)を参照してください。

非遵守ホストの修正

ESXi ホストの関連付けの修正は、vSphere Auto Deploy のアクティブなルール セットにルールを追加するとき、または 1 つ以上のルールに変更を加えるときに行います。

前提条件

vSphere Auto Deploy のアクティブなルール セットにルールを追加したり、1 つ以上のルールに変更を加えたりしても、ホストは自動的にアップデートされません。ホストの関連付けを修正し、新しいルールをホストに適用する必要があります。

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- vSphere Auto Deploy のルールを作成します。[デプロイ ルールの作成](#)を参照してください。
- vSphere Auto Deploy のルールを有効にします。[デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え](#)を参照してください。
- ホストを修正した結果、場所が変更になった場合、ホストは必ずメンテナンス モードになります。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ済みのホスト] タブで、単一または複数の ESXi ホストを選択します。

- 3 [ホストの関連付けの修正] をクリックします。

編集されたイメージ プロファイルの関連付けがあるホストを修正すると、ホストは、一致するルールで定義された設定に戻ります。

修正プロセスの進行状況は、[最近のタスク] ペインで監視できます。

次のステップ

- ホストの場所、イメージ プロファイル、ホスト プロファイル、および追加されたスクリプト バンドルを確認します。[ホストの関連付けの表示](#)を参照してください。
- ホストのイメージ プロファイルの関連付けを変更します。[ホストのイメージ プロファイルの関連付けの編集](#)を参照してください。

vSphere Auto Deploy インベントリへのホストの追加

vSphere Auto Deploy ルールに一致しないホストを表示し、ホストを手動で vSphere Auto Deploy インベントリに追加することができます。

デプロイしたホストの現在の vSphere Auto Deploy インベントリにホストを追加するには、新しいルールを作成するか既存のルールを編集し、そのルールに vSphere Auto Deploy でデプロイされていないホストを含め、それを特定のホストの場所、イメージ プロファイル、ホスト プロファイル、およびスクリプト バンドルに関連付けます。または、ホストにホストの場所、イメージ プロファイル、ホスト プロファイル、およびスクリプト バンドルを割り当てることで、ホストを手動でインベントリに追加することもできます。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、[vSphere Auto Deploy 用にシステムを準備](#)を参照してください。
- イメージ プロファイルをホストに割り当てるには、必要なソフトウェア デポをインベントリに追加します。ソフトウェア デポの追加またはソフトウェア デポのインポートを参照してください。

手順

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [検出されたホスト] タブで、ホストの場所、イメージ プロファイル、およびホスト プロファイルを関連付けてプロビジョニングする 1 台以上のホストを選択します。

- 3 [インベントリへの追加] を選択します。

あるいは、[見つかったホスト] タブで [削除] をクリックして、選択されているホストを解除します。

インベントリへの追加ウィザードが起動します。

- 4 ウィザードの [ホストの場所の選択] ページで、ルールに一致するホストの場所として、データセンター、フォルダー、またはクラスタを選択します。

- 5 ウィザードの [イメージ プロファイルの選択] ページで、ドロップダウン メニューを使用してソフトウェア デポを選択し、リストからイメージ プロファイルを選択します。

イメージ プロファイルの許容レベル検証をしない場合は、[イメージ プロファイルの署名チェックをスキップ] チェック ボックスを選択します。

- 6 ウィザードの [ホスト プロファイルの選択] ページで、リストからホスト プロファイルを選択します。

- 7 ウィザードの [ホスト プロファイルの選択] ページで、[フィルタ] を使用して、ホスト プロファイルのリストを検索するか、[ホスト プロファイルを含めない] チェック ボックスを有効にして、ホスト プロファイルを追加せずに続行します。

- 8 ウィザードの [スクリプト バンドルの選択] ページで、リストからスクリプト バンドルを選択します。

- 9 [設定の確認] ページで、選択したホストの関連付けを確認します。

次のステップ

- vSphere Auto Deploy ルールを編集します。[イメージ プロファイルの編集](#)を参照してください。
- vSphere Auto Deploy ルールのクローンを作成します。[デプロイ ルールのクローンの作成](#)を参照してください。
- ホストの場所、イメージ プロファイル、ホスト プロファイル、および追加されたスクリプト バンドルを確認します。[ホストの関連付けの表示](#)を参照してください。
- 準拠しないホストを修正します。[非遵守ホストの修正](#)を参照してください。

単一イメージを使用するクラスタへのホストの追加

Auto Deploy で、単一イメージで管理するクラスタを、新しく追加された ESXi ホストのホストの場所として割り当てるルールを作成します。

ホストのターゲットの場所をイメージで管理されているクラスタにする Auto Deploy ルールを作成することで、ステートフル ESXi ホストをクラスタに移行できます。ホスト識別メカニズムに基づいて、ルールによってターゲットクラスタにホストが追加されます。

このような Auto Deploy ルールには、イメージ プロファイルまたはホスト プロファイルを含めることはできません。これは、ターゲット クラスタのホストのイメージ仕様と構成が自動的に作成されるためです。

前提条件

単一イメージで管理されているクラスタに ESXi ホストを追加するには、新しく追加されるホストの場所としてこのようなクラスタを割り当てるルールを Auto Deploy に作成します。各ホストは同じイメージを継承し、これによりハードウェアの互換性チェック、クラスタ全体の修正、アップグレードの簡素化が可能になります。

- システムの準備および Auto Deploy サーバのインストール詳細については、『VMware ESXi のインストールとセットアップ』ドキュメントの「vSphere Auto Deploy 用にシステムを準備」を参照してください。
- 各 ESXi ホストがバージョン 8.0 以降であることを確認します。
- クラスタ内のすべてのホストがステートフルで、物理ストレージに接続されていることを確認します。
- vSphere Lifecycle Manager と統合されていないソリューションがクラスタに対して有効になっていないことを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ ルール] タブで [新規デプロイ ルール] をクリックします。

[新規デプロイ ルール] ウィザードが起動します。

- 3 ウィザードの [名前とホスト] のページで、新しいルールの名前を入力します。

- 4 ルールをインベントリ内のすべてのホストに適用するか、特定のパターンに一致するホストにのみ適用するかを選択します。

1 つ以上のパターンを選択することができます。

たとえば、ルールを vCenter Single Sign-On ドメイン内の特定のホスト名を持つホスト、または特定の IPv4 範囲に一致するホストにのみ適用することができます。

- 5 ウィザードの [構成] 画面で [ホストの場所] チェック ボックスを選択し、[次へ] をクリックします。

必要に応じて、[スクリプト バンドル] チェック ボックスを選択して、ルール条件に一致する ESXi ホストにスクリプト バンドルを含めることができます。

注： ターゲット クラスタに添付されているイメージ プロファイルとホスト プロファイルが使用されます。

- 6 ウィザードの [ホストの登録場所の選択] 画面で、単一のイメージを使用するクラスタを選択します。

- 7 ウィザードの [スクリプト バンドルの選択] ページで、リストからスクリプト バンドルを選択します。
- 8 [設定の確認] ページで、新しいルールの概要情報を確認します。
- 9 [終了] をクリックします。

結果

[デプロイのルール] タブには新規作成されたルールがリスト表示されます。

次のステップ

- vSphere Auto Deploy のルールを有効にします。デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替えを参照してください。
- ステートレス キャッシュおよびステートフル インストールの詳細については、ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用を参照してください。

クラスタ レベルで ESXi 構成を管理するクラスタへのホストの追加

Auto Deploy で、クラスタ レベルで ESXi 構成を管理するクラスタに、新しく追加されたホストを割り当てるルールを作成します。

ホストのターゲットの場所をクラスタ レベルで ESXi 構成を管理するクラスタにする Auto Deploy ルールを作成することにより、ホスト プロファイルを使用する必要も、手動で構成する必要もなくなります。同時に、ホストごとにカスタム設定を定義したり、ホストのグループに対してクラスタレベルの設定をオーバーライドしたりする柔軟性は保持されます。構成を使用してクラスタ レベルで管理するクラスタを設定する方法、および VMware vSphere 構成プロファイルの詳細については、「vSphere Configuration Profiles を使用したクラスタ レベルでのホスト構成の管理」を参照してください。

注： クラスタ レベルで ESXi 構成を管理するようにクラスタを設定すると、ホスト プロファイルまたは単一イメージを使用する状態にロールバックすることはできません。また、これらのクラスタではクイックスタート オプションを使用できません。ただし、単一イメージを使用して管理するクラスタから、クラスタ レベルの構成で管理するクラスタには、[構成] - [目的の状態] - [ホストの設定] の順に選択し、[ホスト設定のセットアップ] オプションを選択して切り替えることができます。

前提条件

クラスタ レベルで ESXi 構成を管理するクラスタに ESXi ホストを追加するには、新しく追加したホストのための場所としてこのクラスタを割り当てるというルールを、Auto Deploy 内に作成します。ホストは同じ設定を継承するため、手動で構成する必要はありません。

- システムの準備および Auto Deploy サーバのインストール詳細については、『vSphere Auto Deploy のインストールと構成』を参照してください。
- 各 ESXi ホストがバージョン 8.0 以降であることを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイ ルール] タブで [新規デプロイ ルール] をクリックします。
[新規デプロイ ルール] ウィザードが起動します。
- 3 ウィザードの [名前とホスト] のページで、新しいルールの名前を入力します。
- 4 ルールをインベントリ内のすべてのホストに適用するか、特定のパターンに一致するホストにのみ適用するかを選択します。

1 つ以上のパターンを選択することができます。
たとえば、ルールを vCenter Single Sign-On ドメイン内の特定のホスト名を持つホスト、または特定の IPv4 範囲に一致するホストにのみ適用することができます。
- 5 ウィザードの [構成] 画面で [ホストの場所] チェック ボックスを選択し、[次へ] をクリックします。

[イメージ]、[ホストの設定] および [スクリプト バンドル] のチェック ボックスは無視できます。イメージ仕様とターゲット クラスタ内のホストの構成、およびインストール後の設定があれば自動的に作成されます。いずれかのチェック ボックスを選択しても、ルールによって無視されます。
- 6 ウィザードの [ホストの登録場所の選択] 画面で、クラスタ レベルで ESXi 構成を管理するクラスタを選択します。
- 7 [設定の確認] ページで、新しいルールの概要情報を確認します。
- 8 [終了] をクリックします。

結果

[デプロイのルール] タブには新規作成されたルールがリスト表示されます。

次のステップ

- vSphere Auto Deploy のルールを有効にします。デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替えを参照してください。
- ステートレス キャッシュおよびステートフル インストールの詳細については、ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用を参照してください。

スクリプト バンドルの操作

デプロイ後の追加のホスト構成にカスタム スクリプトを追加できます。スクリプトは、ESXi ホストを Auto Deploy を使用してプロビジョニングした後に実行されます。たとえば、カスタムの ESXi ファイアウォール ルールや、[ホスト プロファイル] では使用できないその他の構成を作成できます。

vSphere 6.7 Update 1 以降では、vSphere Client を使用してカスタム スクリプトを追加または削除できます。スクリプト バンドルには複数のスクリプトを含めることができ、拡張子が .tgz の単一の圧縮ファイルとして配布する必要があります。vCenter Server にアップロードした後、スクリプト バンドルを Auto Deploy ルールに含めることができます。

前提条件

- ESXi Shell でスクリプトを実行できることを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [スクリプト バンドル] タブを選択します。

- 3 [アップロード] をクリックします。

- 4 スクリプト バンドル ファイルを参照し、[アップロード] を選択します。

スクリプトは、[スクリプト バンドル] リストにあります。

- 5 (オプション) スクリプト バンドルを選択し、[削除] をクリックして選択を確定します。

スクリプト バンドルがリストから削除されます。

次のステップ

- vSphere Auto Deploy ルールを有効にします。[デプロイ ルールのアクティベーション、アクティベーション解除、および並べ替え](#)を参照してください。
- vSphere Auto Deploy ルールを編集します。[イメージ プロファイルの編集](#)を参照してください。
- vSphere Auto Deploy ルールのクローンを作成します。[デプロイ ルールのクローンの作成](#)を参照してください。
- ホストの場所、イメージ プロファイル、ホスト プロファイル、および追加されたスクリプト バンドルを確認します。[ホストの関連付けの表示](#)を参照してください。
- 準拠しないホストを修正します。[非遵守ホストの修正](#)を参照してください。
- ホストのイメージ プロファイルの関連付けを変更します。[ホストのイメージ プロファイルの関連付けの編集](#)を参照してください。

vSphere Auto Deploy ログのダウンロード

vSphere Client からの vSphere Auto Deploy ログの情報を使用して、vSphere Auto Deploy で発生する問題を解決できます。

前提条件

vSphere Client を使用して、vSphere Auto Deploy が登録されている vCenter Server インスタンスにログインします。

手順

- 1 [ホーム] - [管理] に移動し、[デプロイ] - [システム設定] を選択します。

- 2 サポート バンドルを取得するノードを 1 台クリックします。サポート バンドルにはサービス ログが保持されています。

- 3 [サポート バンドルのエクスポート] をクリックします。

- 4 [VirtualAppliance] - [Auto Deploy] を選択します。

- 5 [サポート バンドルのエクスポート] ボタンをクリックしてログ ファイルをダウンロードします。

vSphere Auto Deploy サービスの開始、停止、または再起動

vCenter Server 管理インターフェイスで、Auto Deploy サービスを開始、停止、または再起動できます。

vCenter Server でサービスを開始、停止、および再起動するには、vCenter Server 管理インターフェイスを使用します。

前提条件

vCenter Server 管理インターフェイスへの root アクセス権があることを確認します。

手順

1 vCenter Server 管理インターフェイス (<https://IP-address-or-FQDN:5480>) にログインします。

2 [サービス] をクリックします。

[サービス] ペインに、インストールされているすべてのサービスの表が表示されます。これらのサービスを名前、開始タイプ、健全性、および状態で並べ替えることができます。

3 [Auto Deploy] サービスを選択し、アクションを選択します。

使用可能なアクションは、Auto Deploy サービスがすでに実行されているかどうかによって異なります。

- サービスを再起動するには、[再起動] をクリックします。

サービスを再起動するには確認が必要です。また、Auto Deploy 機能が一時的に使用できなくなる可能性があります。

- サービスを開始するには、[開始] をクリックします。
- サービスを停止するには、[停止] をクリックします。

サービスを停止するには確認が必要です。

vSphere Auto Deploy を使用した ESXi ホストのプロビジョニング

vSphere Auto Deploy を使用して、ESXi ソフトウェアを使用して数百台の物理ホストをプロビジョニングまたは再プロビジョニングします。

vSphere Auto Deploy を使用して、ESXi ソフトウェアを備えた数百台の物理ホストの初めてのプロビジョニング（初回起動）、ホストの再起動、または異なるイメージ プロファイル、ホスト プロファイル、カスタム スクリプト、フォルダ、またはクラスタの場所を使用した再プロビジョニングを実行できます。VMware Tools バイナリを含まないイメージ プロファイルを使用してホストをプロビジョニングすることもできます。

vSphere Auto Deploy を使用した ESXi システムのプロビジョニング

vSphere Auto Deploy では、ESXi ソフトウェアを備えた数百台の物理ホストをプロビジョニングして、初回起動、再起動、または再プロビジョニングを実行できます。

異なるイメージ プロファイル、ホスト プロファイル、カスタム スクリプト、または、フォルダやクラスタの場所を使用して、これまで ESXi ソフトウェアを実行していなかったホストのプロビジョニング（最初の起動）、ホストの再起動、またはホストの再プロビジョニングを行うことができます。vSphere Auto Deploy プロセスは、ホストの状態や加える変更の内容によって異なります。

vSphere Auto Deploy の起動プロセス

vSphere Auto Deploy を使用してプロビジョニングまたは再プロビジョニングを行うホストを起動するときに、vSphere Auto Deploy インフラストラクチャがそのイメージ プロファイルを提供します。また、オプションでホスト プロファイル、vCenter Server の場所およびそのホストのスクリプト バンドルを提供します。

vSphere Auto Deploy を使用したプロビジョニングを実施したことがないホスト（最初の起動）と、vSphere Auto Deploy を使用したプロビジョニングを実施済みで vCenter Server システムに追加されたホスト（2 回目以降の起動）では、起動プロセスが異なります。

最初の起動の前提条件

最初の起動プロセスの前に、システムを設定する必要があります。設定には次のタスクが含まれます。詳細は、[vSphere Auto Deploy のインストールと構成](#) に説明があります。

- DHCP サーバを設定します。このサーバから、各ホストの起動時に IP アドレスが割り当てられ、iPXE のブート ローダのダウンロード元となる TFTP サーバにホストがポイントされます。
- vSphere Auto Deploy を使用したプロビジョニングを計画するホストでレガシー BIOS が使用されている場合は、vSphere Auto Deploy サーバに IPv4 アドレスがあることを確認します。レガシー BIOS ファームウェアを使用する PXE ブートは IPv4 でのみ実行できます。UEFI ファームウェアを使用する PXE ブートは IPv4 または IPv6 のいずれかで実行できます。
- 次のいずれかの方法で、使用するイメージ プロファイルを識別します。
 - パブリックなデポにある ESXi イメージ プロファイルを選択します。
 - （オプション）vSphere ESXi Image Builder を使用してカスタムのイメージ プロファイルを作成し、vSphere Auto Deploy サーバがアクセスできるデポ内に、作成したイメージ プロファイルを配置します。イメージ プロファイルには、基本の ESXi VIB を含める必要があります。
- （オプション）環境内にリファレンス ホストがある場合は、リファレンス ホストのホスト プロファイルをエクスポートし、そのホスト プロファイルを 1 つ以上のホストに適用するルールを定義します。[vSphere Auto Deploy のリファレンス ホストの設定](#) を参照してください。
- ホストのデプロイのためのルールを指定し、そのルールをアクティブなルール セットに追加します。

最初の起動の概要

vSphere Auto Deploy を使用したプロビジョニングを実施したことがないホストが起動するとき（最初の起動）、ホストはいくつかの vSphere Auto Deploy コンポーネントと対話します。

- 1 管理者がホストを起動すると、ホストは PXE ブート シーケンスを開始します。

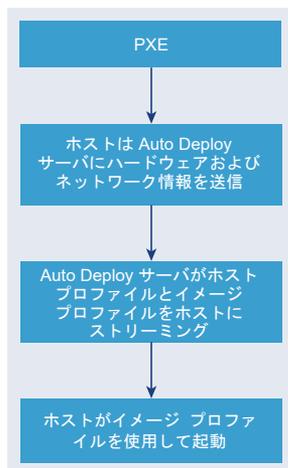
DHCP サーバがホストに IP アドレスを割り当て、TFTP サーバに接続するようにホストに指示します。
- 2 ホストは TFTP サーバに接続し、iPXE ファイル（実行可能なブート ローダ）と iPXE の構成ファイルをダウンロードします。
- 3 iPXE が実行を開始します。

構成ファイルは、vSphere Auto Deploy サーバに HTTP の起動を要求することをホストに指示します。HTTP の要求には、ハードウェアとネットワークの情報が含まれます。
- 4 応答として、vSphere Auto Deploy サーバは次のタスクを実行します。
 - a ホストに関する情報をルール エンジンに照会します。

- b イメージ プロファイル、ホスト プロファイル (オプション)、および vCenter Server の場所に関する情報 (オプション) で指定されたコンポーネントをストリーミングします。
- 5 イメージ プロファイルを使用してホストが起動します。
vSphere Auto Deploy サーバがホスト プロファイルを提供した場合は、そのホスト プロファイルがホストに適用されます。
 - 6 vSphere Auto Deploy は、自身が登録されている vCenter Server システムにホストを追加します。
 - a vCenter Server システム上のターゲット フォルダまたはクラスタがルールに指定されている場合は、そのフォルダまたはクラスタにホストが配置されます。ターゲット フォルダは、データセンターの下に置かれている必要があります。
 - b vCenter Server インベントリの場所を指定するルールがない場合は、vSphere Auto Deploy は vSphere Client ユーザー インターフェイス内に最初に表示されるデータセンターにホストを追加します。
 - 7 (オプション) 固定 IP アドレスなどの特定の情報を指定することがホスト プロファイルでユーザーに要求される場合、vCenter Server システムにホストが追加されるときに、ホストはメンテナンス モードになります。
ホスト プロファイルを再適用し、ホストのカスタマイズ内容を更新して、ホストのメンテナンス モードを終了する必要があります。ホストのカスタマイズ内容を更新する際は、質問が表示されたら回答します。
 - 8 ホストが DRS クラスタの一部である場合は、ホストが vCenter Server システムに正しく追加されたあとで、ほかのホストの仮想マシンがこのホストに移行される場合があります。

ホストのプロビジョニング (最初の起動) を参照してください。

図 4-7. vSphere Auto Deploy のインストール、最初の起動



2 回目以降の起動 (更新なし)

vSphere Auto Deploy でプロビジョニングされ、vCenter Server によって管理されているホストでは、2 回目以降の起動は完全に自動化できます。

- 1 管理者がホストを再起動します。
- 2 ホストが起動すると、イメージ プロファイルおよびホスト プロファイルを使用して、vSphere Auto Deploy がホストをプロビジョニングします。

3 ホストの設定に基づいて、仮想マシンが起動またはホストに移行されます。

- スタンドアロン ホスト。ホスト上で定義されている自動開始のルールに従って、仮想マシンがパワーオンされます。
- DRS クラスタ ホスト。ほかのホストに正しく移行された仮想マシンは、移行先に残ります。十分なリソースがないホストの仮想マシンは、再起動されたホストに登録されます。

vCenter Server が使用可能でない場合は、ホストは vSphere Auto Deploy サーバに接続して、イメージ プロファイルによりプロビジョニングされます。ホストは、vCenter Server に vSphere Auto Deploy が再接続するまで、vSphere Auto Deploy サーバにアクセスし続けます。

vCenter Server が使用できない場合は、vSphere Auto Deploy は vSphere Distributed Switch を設定できないため、ホストが vSphere HA クラスタに参加している場合のみ、仮想マシンがホストに割り当てられます。ホストが vCenter Server に再接続されてホスト プロファイルが適用されるまでは、スイッチを作成できません。ホストがメンテナンス モードであるため、仮想マシンは開始できません。 [シンプルな再起動操作によるホストの再プロビジョニング](#)を参照してください。

ホストがユーザー入力を要求するように設定されると、それらのホストはメンテナンス モードになります。

[vSphere Client でのホストのカスタマイズの更新](#)を参照してください。

2 回目以降の起動（更新あり）

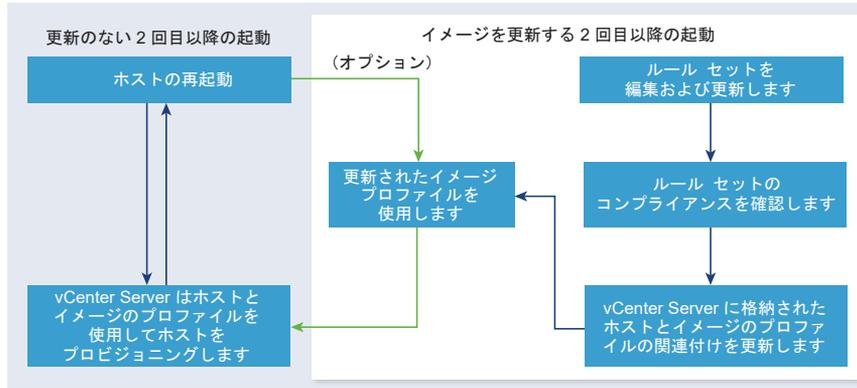
イメージ プロファイル、ホスト プロファイル、vCenter Server の場所、またはホストのスクリプト バンドルを変更することができます。このプロセスには、ルールの変更と、ホストのルール コンプライアンスのテストおよび修復が含まれます。

- 1 管理者は、Copy-DeployRule PowerCLI コマンドレットを使用して1つ以上のルールをコピーして編集し、ルール セットを更新します。例については、 [PowerCLI を使用した vSphere Auto Deploy プロセスの概要](#)を参照してください。
- 2 管理者は、Test-DeployRulesetCompliance コマンドレットを実行して、各ホストが現在のルール セットが指定する情報を使用しているかどうかを確認します。
- 3 ホストは、コンプライアンス情報をカプセル化した PowerCLI オブジェクトを返します。
- 4 管理者は、Repair-DeployRulesetCompliance コマンドレットを実行して、vCenter Server システムが各ホスト用に格納しているイメージ プロファイル、ホスト プロファイル、または vCenter Server の場所を更新します。
- 5 ホストが再起動するときに、そのホストは更新されたイメージ プロファイル、ホスト プロファイル、vCenter Server の場所、またはホストのスクリプト バンドルを使用します。

ユーザーの入力を要求するようにホスト プロファイルが設定されている場合は、ホストはメンテナンス モードになります。 [vSphere Client でのホストのカスタマイズの更新](#) の手順に従います。

[ルールのコンプライアンスのテストおよび修復](#)を参照してください。

図 4-8. vSphere Auto Deploy のインストール、2 回目以降の起動



Distributed Switch を持つシステムのプロビジョニング

vSphere Auto Deploy のリファレンス ホストのホスト プロファイルを、Distributed Switch を使用して構成することができます。

Distributed Switch の構成時に、再起動後にホストに接続する際に必要なネットワーク パラメータと一致するように、起動構成パラメータのポリシーが自動的に設定されます。

vSphere Auto Deploy がホスト プロファイルを使用して ESXi ホストのプロビジョニングを行う場合、ホストでは 2 つの手順からなるプロセスが実施されます。

- 1 ホストは、起動構成パラメータ フィールドで指定されたプロパティを使用して、標準の仮想スイッチを作成します。
- 2 ホストは VMkernel NIC を作成します。VMkernel NIC は、vSphere Auto Deploy および vCenter Server システムへの接続をホストに許可します。

ホストが vCenter Server に追加されると、vCenter Server は標準スイッチを削除し、ホストに Distributed Switch を適用し直します。

注： Distributed Switch に問題が発生することを避けるために、起動構成パラメータは変更しないでください。

ホストのプロビジョニング（最初の起動）

vSphere Auto Deploy を使用したプロビジョニングを実施したことがないホスト（最初の起動）のプロビジョニングは、それ以降の起動でのプロセスとは異なります。ホストを準備し、その他すべての前提条件を満たしてから、ホストをプロビジョニングする必要があります。任意で vSphere Client または PowerCLI コマンドレットを使用して、vSphere ESXi Image Builder でカスタム イメージ プロファイルを定義できます。

前提条件

- ホストが ESXi ホストのハードウェア要件を満たしていることを確認します。
ESXi のハードウェア要件を参照してください。
- vSphere Auto Deploy 用にシステムを準備します（vSphere Auto Deploy のインストールと構成を参照）。

- ホストにイメージ プロファイルを割り当てるルールを記述し、任意で、ホストにホスト プロファイルと vCenter Server の場所を割り当てます。 [PowerCLI コマンドレットを使用した vSphere Auto Deploy の管理](#)または [vSphere Client を使用した vSphere Auto Deploy の管理](#)を参照してください。

vSphere Auto Deploy サービスが有効になり、DHCP の設定が完了し、プロビジョニング対象のホストのルールをアクティブなルール セットに含めると、設定が完了します。

手順

- 1 ホストを起動します。

ホストは DHCP サーバに接続し、サーバがポイントする場所から iPXE をダウンロードします。次に、ルール エンジンで指定されたイメージを使用して、vSphere Auto Deploy サーバがホストをプロビジョニングします。vSphere Auto Deploy サーバは、ルール セットで指定されている場合は、ホスト プロファイルをホストに適用する場合があります。最後に、vSphere Auto Deploy は、ルール セットで指定されている vCenter Server システムにホストを追加します。

- 2 (オプション) vSphere Auto Deploy が、IP アドレスなどのユーザー入力を要求するホスト プロファイルを適用する場合、ホストはメンテナンス モードになります。vSphere Client でホスト プロファイルを再び適用して、プロンプトが表示されたらユーザー入力を行います。

結果

最初の起動プロセス後は、ホストは実行され、vCenter Server システムによって管理されます。vCenter Server はホストのイメージ プロファイル、ホスト プロファイル、および場所の情報を保存します。

これで、必要に応じてホストを再起動できます。再起動するたびに、ホストは vCenter Server システムによって再びプロビジョニングされます。

次のステップ

必要に応じてホストを再プロビジョニングします。 [ホストの再プロビジョニング](#)を参照してください。

イメージ プロファイル、ホスト プロファイル、カスタム スクリプト、またはホストの場所を変更する場合は、vSphere Client を使用してルールを更新し、それらをアクティベートするか、PowerCLI セッションでコンプライアンスのテストおよび修復の操作を実行します。 [ルールおよびルール セット](#)または [ルールのコンプライアンスのテストおよび修復](#) を参照してください。

ホストの再プロビジョニング

vSphere Auto Deploy により、異なるイメージ プロファイルまたは異なるホスト プロファイルを使用して ESXi ホストを再プロビジョニングします。

vSphere Auto Deploy では、複数の再プロビジョニング オプションがサポートされます。単純な再起動や、異なるイメージ プロファイルまたは異なるホスト プロファイルを使用した再プロビジョニングを実行できます。

vSphere Auto Deploy を使用して最初に起動するときには、環境を設定し、ルール セットにルールを追加する必要があります。 [vSphere Auto Deploy のインストールと構成](#)を参照してください。

次の再プロビジョニング操作が可能です。

- 単純な再起動。
- 起動操作中にユーザーが質問に対して行った回答の対象となるホストの再起動。

- 異なるイメージ プロファイルを使用した再プロビジョニング。
- 異なるホスト プロファイルを使用した再プロビジョニング。

シンプルな再起動操作によるホストの再プロビジョニング

イメージ プロファイル、ホスト プロファイル、カスタム スクリプト、および最初の起動時に割り当てられた vCenter Server の場所を使用して、ESXi ホストを再プロビジョニングできます。

前提条件

vSphere Auto Deploy でプロビジョニングされているホストの単純な再起動では、すべての前提条件が満たされていることのみが必要です。このプロセスでは、以前に割り当てられたイメージ プロファイル、ホスト プロファイル、および vCenter Server の場所が使用されます。

- 最初の起動操作中に実行した設定が適切であることを確認します。 [ホストのプロビジョニング（最初の起動）](#) を参照してください。
- すべての関連付けられた項目などが使用可能であることを確認します。項目には、イメージ プロファイル、ホスト プロファイル、カスタム スクリプトや vCenter Server インベントリの場所があります。
- ホストに、以前の起動操作中の識別情報（アセット タグ、IP アドレス）があることを確認します。

手順

- 1 ホストをメンテナンス モードにします。

ホスト タイプ	操作
DRS クラスタの一部であるホスト	VMware DRS では、ホストがメンテナンス モードになると、仮想マシンが適切なホストに移行されます。
DRS クラスタの一部ではないホスト	すべての仮想マシンを別のホストに移行し、各ホストをメンテナンス モードにする必要があります。

- 2 ホストを再起動します。

結果

ホストがシャットダウンします。ホストが再起動するときに、そのホストは vSphere Auto Deploy サーバが提供するイメージ プロファイルを使用します。vSphere Auto Deploy サーバでは、vCenter Server システムに保存されているホスト プロファイルも適用します。

PowerCLI を使用したホストの再プロビジョニング

vSphere Auto Deploy では、PowerCLI セッションで新しいイメージ プロファイルを使用してホストを再プロビジョニングできます。

ホストの再プロビジョニングにはいくつかのオプションがあります。

- 使用する VIB でライブ アップデートがサポートされる場合は、`esxcli software vib update` コマンドを使用できます。その場合は、新しい VIB を含むイメージ プロファイルを使用するように、ルール セットを更新する必要もあります。

- テスト時に、Apply-EsxImageProfile コマンドレットを使用して個々のホストにイメージ プロファイルを適用し、ホストを再起動して変更を有効にすることができます。Apply-EsxImageProfile コマンドレットは、ホストとイメージ プロファイルの関連付けを更新しますが、ホストに VIB をインストールすることはありません。
- その他すべての場合は、この手順を実行します。

前提条件

- ホストの再プロビジョニングに使用するイメージ プロファイルが使用できることを確認します。PowerCLI セッションで vSphere ESXi Image Builder を使用します。 [vSphere ESXi Image Builder を使用したインストールのカスタマイズ](#)を参照してください。
- 最初の起動操作中に実行した設定が適切であることを確認します。

手順

- 1 PowerShell プロンプトで Connect-VIServer PowerCLI コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_or_ipv6_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の警告が表示されないことを確認します。開発環境では、この警告は無視できます。

- 2 使用するイメージ プロファイルが格納されているパブリックなソフトウェア デポを特定するか、vSphere ESXi Image Builder を使用してカスタム イメージ プロファイルを定義します。
- 3 Add-EsxSoftwareDepot を実行して、イメージ プロファイルが含まれるソフトウェア デポを PowerCLI セッションに追加します。

デポのタイプ	コマンドレット
リモート デポ	Add-EsxSoftwareDepot <i>depot_url</i> を実行します。
ZIP ファイル	<ol style="list-style-type: none"> a ZIP ファイルをローカル ファイル パスにダウンロードするか、PowerCLI マシンにローカルなマウント ポイントを作成します。 b Add-EsxSoftwareDepot C:\<i>file_path</i>\<i>my_offline_depot.zip</i> を実行します。

- 4 Get-EsxImageProfile を実行してイメージ プロファイルのリストを表示し、使用するプロファイルを決定します。
- 5 Copy-DeployRule を実行し、ReplaceItem パラメータを指定して、イメージ プロファイルをホストに割り当てるルールを変更します。

次のコマンドレットは、*my_new_imageprofile* プロファイルを使用して、ルールがホストに割り当てる現在のイメージ プロファイルを置き換えます。コマンドレットが完了すると、*myrule* で新しいイメージ プロファイルがホストに割り当てられます。古いバージョンの *myrule* は、名前が変更され、非表示になります。

```
Copy-DeployRule myrule -ReplaceItem my_new_imageprofile
```

6 イメージをデプロイする各ホストのルール コンプライアンスをテストします。

- a ルール セットのコンプライアンスをテストするホストにアクセスできることを確認します。

```
Get-VMHost -Name ESXi_hostname
```

- b そのホストのルール セットのコンプライアンスをテストし、後で使用するために、戻り値を変数にバインドするコマンドレットを実行します。

```
$str = Test-DeployRuleSetCompliance ESXi_hostname
```

- c ルール セットの内容と、ホストの構成の間の差異を確認します。

```
$str.itemlist
```

システムは、新しいルール セットのコンプライアンスをテストするホストがアクティブなルール セットを遵守している場合、現在の項目と想定される項目のテーブルを返します。

CurrentItem	ExpectedItem
-----	-----
my_old_imageprofile	my_new_imageprofile

- d ホストの次の起動時に、変更されたルール セットを使用するように、ホストを修復します。

```
Repair-DeployRuleSetCompliance $str
```

7 新しいイメージ プロファイルを使用して再プロビジョニングするホストを再起動します。

vSphere Client を使用した、新しいイメージ プロファイルを使用するホストの再プロビジョニング

vSphere Auto Deploy を使用してホストが対応するルールを変更し、そのルールをアクティベートすることで、vSphere Client で新しいイメージ プロファイルを使用するホストを再プロビジョニングすることができます。

前提条件

- ホストの再プロビジョニングに使用するイメージ プロファイルが使用できることを確認します。[イメージ プロファイルの作成](#)を参照してください。
- 最初の起動操作中に実行した設定が適切であることを確認します。

手順

- 1 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

- 2 [デプロイのルール] タブで、インベントリ内のルールのリストからルールを選択し、[編集] をクリックします。

デプロイのルールを編集するダイアログ ボックスが表示されます。

- 3 (オプション) ウィザードの [名前とホスト] のページで、新しいルールの名前を入力します。

- 4 (オプション) ルールをインベントリ内のすべてのホストに適用するか、特定のパターンに一致するホストにのみ適用するかを選択します。

1つ以上のパターンを選択することができます。

たとえば、ルールを vCenter Single Sign-On ドメイン内の特定のホスト名を持つホスト、または特定の IPv4 範囲に一致するホストにのみ適用することができます。

- 5 ウィザードの [設定] ページで、オプションでルールにアイテムを含めることができます。

有効にしたアイテムごとに、ウィザードに新しいページが追加されます。

オプション	操作
ホストの場所	ルールの条件に一致するホストを特定の場所に追加します。
イメージ プロファイル	ルール条件に一致するホストにイメージ プロファイルを割り当てます。
ホスト プロファイル	ルール条件に一致するホストにホスト プロファイルを割り当てます。
スクリプト バンドル	ルール条件に一致するホストにスクリプト バンドルを割り当てます。

- 6 [次へ] をクリックして、ホストの場所の選択をスキップします。

- 7 ウィザードの [イメージ プロファイルの選択] ページで、イメージ プロファイルをルール条件に一致するホストに割り当てます。

オプション	操作
イメージ プロファイルを変更しない場合	[同じイメージ プロファイル] チェック ボックスをオンにします。
選択したホストに新しいイメージ プロファイルを割り当てる場合	<ol style="list-style-type: none"> [イメージ プロファイルを参照] チェックボックスをオンにします。 ドロップダウン メニューからソフトウェア デポを選択します。 リストからイメージ プロファイルを選択します。 (オプション) イメージ プロファイルの許容レベル検証をしない場合は、[イメージ プロファイルの署名チェックをスキップ] チェック ボックスを選択します。

- 8 [次へ] をクリックして、ホスト プロファイルの選択をスキップします。

- 9 [設定の確認] ページで、新しいイメージ プロファイルの概要情報を確認して、[終了] をクリックします。

- 10 [ルールのアクティベーション/アクティベーションの解除] をクリックします。

- 11 有効にするルールを無効なルールのリストから選択し、[有効化] ボタンをクリックします。

- 12 (オプション) 有効なルール リストのルールの順番を変更するには、移動するルールを選択して、有効なルール リストの上部にある [上へ移動] または [下へ移動] をクリックします。

ルールは優先度の高い順にリストされます。たとえば、2つ以上のルールが同じホストに適用され、それぞれのルールでホストをプロビジョニングする際に使用するホストの場所、イメージ プロファイル、およびホスト プロファイルが異なる場合は、リスト内の最上位のルールが適用されます。

13 (オプション) 無効なルールを有効にする前にテストするには、[有効にする前にルールをテスト] をクリックします。

- a リストからホストを選択し、[コンプライアンスの確認] をクリックすると、ホストの現在のステータスと、ルールを有効にした後に予想される変化が表示されます。

ホストがルールに準拠している場合は、ルールを有効にした後にホストを修正する必要はありません。

- b (オプション) 選択したホストをルールの有効化後に修正するには、切り替えボタンを有効にするか、[ルールの有効化後にすべてのホストの関連付けを修正] チェック ボックスをオンにしてすべてのホストを修正します。

14 有効なルールのリストを確認し、[OK] をクリックします。

15 新しいイメージ プロファイルを使用して再プロビジョニングするホストを再起動します。

vSphere Client でのホストのカスタマイズの更新

ホストの前の起動でユーザー入力が求められた場合、ユーザーが入力した回答は vCenter Server に保存されています。そのユーザーに新しい情報の入力を求める場合は、ホストを修正する必要があります。

前提条件

ユーザー入力を求めるホスト プロファイルをホストに添付します。

手順

1 すべての仮想マシンを別のホストに移行し、ホストをメンテナンス モードにします。

ホスト タイプ	操作
DRS クラスタの一部であるホスト	VMware DRS では、ホストがメンテナンス モードになると、仮想マシンが適切なホストに移行されます。
DRS クラスタの一部ではないホスト	すべての仮想マシンを別のホストに移行し、各ホストをメンテナンス モードにする必要があります。

2 [ホーム] - [Auto Deploy] の順に移動します。

デフォルトでは、管理者ロールのみが vSphere Auto Deploy サービスを利用する権限を持っています。

3 [デプロイ済みのホスト] タブで、ESXi ホストを選択してください。

4 [ホストの関連付けの修正] をクリックします。

修正プロセスの進行状況は、[最近のタスク] ペインで監視できます。

5 プロンプトが表示されたら、ユーザー入力を行います。

6 ホストがメンテナンス モードを終了するように指示します。

結果

ホストのカスタマイズ内容が保存され、次回ホストを起動したときに有効になります。

VMware Tools なしのイメージ プロファイルを使用した ESXi ホストのプロビジョニング

VMware Tools バイナリを含まないイメージ プロファイルを使用して ESXi ホストをプロビジョニングすることもできます。

標準イメージを使用するとネットワーク ブートに時間がかかりすぎる場合や、ホストの容量を節約する必要がある場合は、VMware Tools を含まないイメージ プロファイルを使用し、共有ストレージに VMware Tools のバイナリを保存することができます。

前提条件

vSphere Auto Deploy で ESXi ホストをプロビジョニングする場合は、VMware Tools バイナリを含まないイメージ プロファイルを使用したホストのプロビジョニングを選択できます。このイメージ プロファイルは小さく、メモリ オーバーヘッドも少ないため、PXE ブート環境で迅速に起動します。

VMware のダウンロード サイトから、xxxxx-no-tools イメージ プロファイルをダウンロードします。

手順

- 1 vSphere Auto Deploy でプロビジョニングされていない ESXi ホストを起動します。
- 2 ESXi ホストから共有ストレージに /productLocker ディレクトリをコピーします。

SSH クライアントを使用して ESXi ホストに接続できます。ナレッジベースの記事 [KB 1019852](#) を参照してください。
- 3 新しい /productLocker ディレクトリの場所を参照するように *UserVars.ProductLockerLocation* 変数を変更します。
 - a vSphere Client でリファレンス ホストを選択し、[構成] タブをクリックします。
 - b [システム] メニューの [システムの詳細設定] をクリックします。
 - c [編集] をクリックします。
 - d **uservars** の設定をフィルタリングして、[UserVars.ProductLockerLocation] を選択します。
 - e 現在の値をクリックし、共有ストレージを参照するように場所を編集します。
- 4 リファレンス ホストからホスト プロファイルを作成します。
- 5 リファレンス ホストの xxxxx-no-tools イメージ プロファイルおよびホスト プロファイルを他のすべてのホストに割り当てる vSphere Auto Deploy ルールを作成します。
- 6 このルールを割り当てたターゲット ホストを起動すると、リファレンス ホストからプロダクト ロッカーの場所が取得されます。

ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用

ステートレス キャッシュを使用すると、ESXi ホストのイメージをキャッシュできます。ステートフル インストールでは、ネットワーク経由でホストをインストールできます。

vSphere Auto Deploy のステートレス キャッシュ機能を使用すると、ホストのイメージをキャッシュできます。vSphere Auto Deploy のステートフル インストール機能を使用すると、ネットワーク経由でホストをインストールできます。初回ネットワーク ブート後、これらのホストは、他の ESXi ホストと同じように起動します。ステートレス キャッシュ ソリューションは、主に複数のホストを同時に起動する状況を対象としています。ローカルにイメージをキャッシュすることで、何百ものホストが同時に vSphere Auto Deploy サーバに接続される場合に発生するボトルネックを回避できます。起動操作が完了すると、ホストが vSphere Auto Deploy に接続されて設定が完了します。

ステートフル インストール機能では、PXE ブート インフラストラクチャを設定しなくても、イメージ プロファイルを使用してホストをネットワーク経由でプロビジョニングできます。

次に参照するドキュメント

- [ステートレス キャッシュおよびステートフル インストールの概要](#)

システム キャッシュ構成のホスト プロファイルを使用すると、vSphere Auto Deploy のステートレス キャッシュおよびステートフル インストールでホストをプロビジョニングできます。

- [ステートレス キャッシュおよびステートフル インストールについて](#)

ステートレス キャッシュまたはステートフル インストールで vSphere Auto Deploy を使用する場合、ホスト プロファイルを設定して適用し、起動順序を設定する必要があります。

ステートレス キャッシュおよびステートフル インストールの概要

システム キャッシュ構成のホスト プロファイルを使用すると、vSphere Auto Deploy のステートレス キャッシュおよびステートフル インストールでホストをプロビジョニングできます。

ステートレス キャッシュおよびステートフル インストールの例

vSphere Auto Deploy でプロビジョニングされたホストがイメージをキャッシュする（ステートレス キャッシュ）

ステートレス キャッシュ用にホスト プロファイルを設定し、適用します。イメージは、ローカル ディスク、リモート ディスク、または USB ドライブでキャッシュできます。このホストを引き続き vSphere Auto Deploy でプロビジョニングします。数百のホストから同時にアクセスが試行されるなどの理由で vSphere Auto Deploy サーバが使用できなくなると、ホストがキャッシュから起動されます。起動操作後、ホストは構成を完了させるために vSphere Auto Deploy サーバへのアクセスを試行します。

vSphere Auto Deploy でプロビジョニングされたホストがステートフル ホストになる

ステートフル インストール用にホスト プロファイルを設定し、適用します。vSphere Auto Deploy でホストをプロビジョニングすると、ローカル ディスク、リモート ディスク、または USB ドライブにイメージがインストールされます。以降の起動は、ディスクから行います。ホストは vSphere Auto Deploy を使用しなくなります。

準備

ステートレス キャッシュまたはステートフル インストールを正常に使用するために、システムの構成方法を決め、起動順序を設定します。

表 4-19. ステートレス キャッシュまたはステートフル インストールの準備

要件または決定事項	説明
VMFS パーティションの上書きの決定	対話形式のインストーラを使用して ESXi をインストールする場合、既存の VMFS データストアを上書きするかどうかを求めるプロンプトが表示されます。システム キャッシュ構成のホスト プロファイルには、既存の VMFS パーティションを上書きするためのオプションがあります。 USB ドライブを使用するようにホスト プロファイルを設定した場合は、このオプションは表示されません。
可用性の高い環境が必要かどうかの決定	ステートレス キャッシュで vSphere Auto Deploy を使用する場合、可用性の高い vSphere Auto Deploy 環境を設定することで、新しくプロビジョニングされたホストで仮想マシンを移行したり、vCenter Server システムが一時的に使用できなくなっても環境で vNetwork Distributed Switch をサポートしたりできます。
起動順序の設定	ホストに対して指定する起動順序は、使用する機能によって異なります。 <ul style="list-style-type: none"> ■ ステートレス キャッシュで vSphere Auto Deploy を設定するには、最初にネットワークから起動し、次にディスクから起動するようにホストを構成します。vSphere Auto Deploy サーバを使用できない場合は、キャッシュを使用してホストが起動します。 ■ 起動可能なディスクが現在ないホストでステートフル インストール用に vSphere Auto Deploy を設定するには、最初にディスクから起動し、次にネットワークから起動するようにホストを構成します。 <p>注： 起動可能なイメージがディスクに現在ある場合は、PXE ブートを 1 回だけ行うようにホストを構成し、vSphere Auto Deploy でホストをプロビジョニングして、ステートフル インストールを指定するホスト プロファイルを使用します。</p>

ステートレス キャッシュと接続の喪失

vSphere Auto Deploy サーバ、vCenter Server システム、またはその両方への接続が仮想マシンを実行する ESXi ホストで失われると、次回にホストを再起動したときに制限が加えられることがあります。

- vCenter Server は使用できても vSphere Auto Deploy サーバが使用できない場合は、ホストが vCenter Server システムに自動的に接続されません。ホストを vCenter Server に手動で接続するか、vSphere Auto Deploy サーバが再び使用可能になるまで待ちます。
- vCenter Server と vSphere Auto Deploy のどちらも使用できない場合は、VMware Host Client を使用して各 ESXi ホストに接続し、各ホストに仮想マシンを追加できます。
- vCenter Server を使用できない場合は、vSphere DRS が動作しません。vSphere Auto Deploy サーバは、ホストを vCenter Server に追加できません。VMware Host Client を使用して各 ESXi ホストに接続し、各ホストに仮想マシンを割り当てることができます。
- 接続が失われている間に設定に変更を加えると、vSphere Auto Deploy サーバへの接続がリストアされた時点で変更が失われます。

ステートレス キャッシュおよびステートフル インストールについて

ステートレス キャッシュまたはステートフル インストールで vSphere Auto Deploy を使用する場合、ホスト プロファイルを設定して適用し、起動順序を設定する必要があります。

ホストへのキャッシュを有効にするホスト プロファイルを適用すると、指定したディスクが vSphere Auto Deploy によってパーティショニングされます。その後の処理は、ホスト プロファイルの設定内容およびホスト上の起動順序の設定内容によって異なります。

- システム キャッシュ構成のホスト プロファイルで [ホスト上でステートレス キャッシュを有効化] を選択した場合は、ホスト プロファイルを適用すると vSphere Auto Deploy がイメージをキャッシュします。再起動は必要ありません。その後再起動した場合、ホストは引き続き vSphere Auto Deploy インフラストラクチャを使用してイメージを取得します。vSphere Auto Deploy サーバが使用できない場合、ホストはキャッシュされたイメージを使用します。
- システム キャッシュ構成のホスト プロファイルで [ホスト上でステートフル インストールを有効化] を選択した場合は、vSphere Auto Deploy がイメージをインストールします。再起動すると、ホストは、まず、vSphere Auto Deploy を使用して起動し、インストールを完了します。インストーラでプロビジョニングされたホストと同じように、ホストがディスクから起動した後に、自動的に再起動されます。以後、vSphere Auto Deploy はホストのプロビジョニングを行いません。

vSphere Client からのホスト プロファイルを適用するか、PowerCLI セッションで vSphere Auto Deploy ルールを記述することができます。

ステートフル キャッシュまたはステートフル インストール用に vSphere Client を使用して vSphere Auto Deploy を設定する

リファレンス ホストにホスト プロファイルを作成し、そのホスト プロファイルを追加ホストまたは vCenter Server のフォルダやクラスタに適用できます。次のワークフローを使用します。

- 1 vSphere Auto Deploy でホストをプロビジョニングし、そのホストの [システム イメージ キャッシュ構成] ホスト プロファイルを編集します。
- 2 1つ以上のターゲット ホストをメンテナンス モードにし、各ホストにホスト プロファイルを適用してホストのメンテナンス モードを解除します。
- 3 その後の処理は選択したホスト プロファイルによって異なります。
 - ホスト プロファイルでステートレス キャッシュを有効にした場合、イメージがディスクにキャッシュされます。再起動は必要ありません。
 - ホスト プロファイルでステートレス インストールを有効にした場合、イメージがインストールされます。再起動時に、ホストはインストールされたイメージを使用します。

ステートフル キャッシュまたはステートフル インストール用に PowerCLI を使用して vSphere Auto Deploy を設定する

リファレンス ホストにホスト プロファイルを作成し、そのホスト プロファイルを PowerCLI セッションで他のターゲット ホストに適用する vSphere Auto Deploy ルールを記述できます。次のワークフローを使用します。

- 1 vSphere Auto Deploy でリファレンス ホストをプロビジョニングし、キャッシュの形式を有効にするホスト プロファイルを作成します。
- 2 vSphere Auto Deploy で追加ホストをプロビジョニングし、リファレンス ホストのホスト プロファイルをこれらのホストに適用するルールを記述します。

- 3 vSphere Auto Deploy は、イメージ プロファイルまたはルールに関連付けられたスクリプト バンドルを使用して、各ホストをプロビジョニングします。ホスト プロファイルを適用した場合の厳密な影響は、選択したホスト プロファイルによって異なります。
- ステートフル インストールの場合、vSphere Auto Deploy は次のように処理を進めます。
 - 最初の起動中に、vSphere Auto Deploy はホストにイメージをインストールします。
 - その後の起動では、ホストがディスクから起動します。ホストは、vSphere Auto Deploy サーバと接続する必要はありません。
 - ステートフル キャッシュの場合、vSphere Auto Deploy は次のように処理を進めます。
 - 最初の起動中に、vSphere Auto Deploy はホストをプロビジョニングし、イメージをキャッシュします。
 - その後の起動では、vSphere Auto Deploy はホストをプロビジョニングします。vSphere Auto Deploy が使用できない場合、キャッシュされたイメージからホストが起動されますが、ホストが vSphere Auto Deploy サーバに接続できる場合にのみセットアップを実行できます。

ステートレス キャッシュを使用するホスト プロファイルの設定

vSphere Auto Deploy Server が使用できない場合、ホストはキャッシュ内のイメージを使用します。ステートレス キャッシュを使用するには、ホスト プロファイルを構成する必要があります。

前提条件

ステートレス キャッシュを使用するようにホストが設定されている場合、vSphere Auto Deploy Server が使用できないと、ホストはキャッシュ内のイメージを使用します。ステートレス キャッシュを使用するには、ホスト プロファイルを設定する必要があります。このホスト プロファイルは、ステートレス キャッシュ用に設定されたほかのホストに適用できます。

- キャッシュで使用するディスクを決定し、キャッシュ処理で既存の VMFS パーティションを上書きするかどうかを指定します。
- 本番環境では、高可用性環境に含めることにより、vCenter Server システムと vSphere Auto Deploy サーバを保護します。管理クラスタに vCenter Server を含めると、Distributed Switch および仮想マシンの移行を確実に移行できるようになります。可能であれば、インフラストラクチャのその他の要素も保護してください。高可用性 vSphere Auto Deploy インフラストラクチャのセットアップを参照してください。
- vSphere Auto Deploy 向けの環境を設定します。vSphere Auto Deploy のインストールと構成を参照してください。
- 最低でも 4GB の空き容量があるディスクが用意されていることを確認します。ディスクがまだパーティショニングされていない場合は、ホスト プロファイルを適用するときにパーティショニングが実行されます。
- 最初にネットワーク起動を試み、ネットワーク起動に失敗した場合はディスクから起動するようにホストを設定します。ハードウェア ベンダーのマニュアルを参照してください。
- ホスト プロファイルを作成します。ホスト プロファイル ドキュメントを参照してください。

手順

- 1 [ホーム] - [ポリシーおよびプロファイル] - [ホスト プロファイル] の順に移動します。

- 2 設定するホスト プロファイルをクリックし、[管理] タブを選択します。
- 3 [プロファイルの編集] をクリックします。
- 4 ウィザードの [ホストプロファイルの編集] ページで、[詳細な構成設定] - [システム イメージのキャッシュ構成] - [システム イメージのキャッシュ構成] の順に選択します。
- 5 [システム イメージのキャッシュ プロファイル設定] ドロップダウン メニューで、ポリシーのオプションを選択します。

オプション	説明
ホスト上でステートレス キャッシュを有効化	ディスクにイメージをキャッシュします。
ホスト上で USB ディスクへのステートレス キャッシュを有効化	ホストに接続された USB ディスクにイメージをキャッシュします。

6 選択したポリシー オプションに応じて、次の操作を行う必要があります。

a [ホスト上でステートレス キャッシュを有効化] を選択した場合。

1 使用するディスクに関する情報を指定します。

オプション	説明
[最初のディスクの引数]	<p>システム イメージ インストール ディスクを構成する場合は、複数のオプションを使用して、ESXi のインストール先および起動元となるデバイスを定義できます。インストール用のディスクを定義するには、次の引数を使用します。</p> <ul style="list-style-type: none"> ■ localesx - ESXi の有効なインストールを含む最初に検出されたディスク ■ local - 起動後に ESXi によって最初に検出されたローカル ディスク ■ Remotesx - ESXi の有効なインストールを含む最初に検出されたリモート ディスク ■ sortedremotesx - 検出された最も小さい LUN ID でソートされた、ESXi の有効なインストールを含む最初のリモート ディスク ■ remote - 起動後に ESXi によって最初に検出されたリモート ディスク ■ sortedremote - 起動後に ESXi によって検出された最も小さい LUN ID でソートされた最初のリモート ディスク ■ device model ■ device vendor ■ vmkernel device driver name <p>デバイス モデルとベンダーの引数の値を取得するには、コンソールで ESXi ホストに <code>esxcli storage core device list</code> コマンドを実行して、ルートとしてログインします。 <code>esxcli storage core adapter list</code> コマンドを実行して、<code>vmkernel</code> デバイス ドライバ名の引数を取得します。次に、起動デバイスが接続されているストレージアダプタを特定する必要があります。</p> <p>デフォルトでは、システムは既存の ESXi を置き換えてローカル ディスクに書き込もうとします。</p> <p>[最初のディスクの引数] フィールドで、使用するディスクのコンマ区切りリストを任意の順序で指定できます。ディスクは複数指定できます。ESX がインストールされている最初のディスクを示す localesx、またはモデルおよびベンダー情報を使用するか、<code>vmkernel</code> デバイス ドライバの名前を指定します。たとえば、モデル名が <code>ST3120814A</code> のディスク、<code>mptsas</code> ドライバを使用するディスク、ローカル ディスクの順にシステムで検索するには、このフィールドの値として ST3120814A,mptsas,local を指定します。</p> <p>ホスト プロファイルの最初のディスク設定では、キャッシュに使用するディスクを決める際の検索順序を指定します。検索順序は、値のコンマ区切りのリストとして指定します。デフォルト設定 localesx,local では、vSphere Auto Deploy が最初に既存のローカル キャッシュ ディスクを検索するように指定されています。キャッシュ ディスクは、既存の ESXi ソフトウェア イメージが保存されたディスクとして識別されます。vSphere Auto Deploy が既存のキャッシュ ディスクを見つけられない場合は、使用可能なローカル ディスク デバイスが検索されます。vSphere Auto Deploy は、既存の VMFS パーティションがない最初の空ディスクを使用して、使用可能なディスクを検索します。</p> <p>最初のディスク引数は、検索順序の指定にのみ使用できます。ディスクを明示的に指定することはできません。たとえば、SAN に特定の LUN を指定することはできません。</p>
[選択したディスクですべての VMFS ボリュームを上書きするには、チェックボックスを選択]	<p>このチェック ボックスをオンにすると、システムは、容量が不十分でイメージ、イメージ プロファイル、およびホスト プロファイルを保存できない場合、既存の VMFS ボリュームを上書きします。</p>
[ホストに接続されたすべての SSD デバイ	<p>このチェック ボックスをオンにすると、システムは既存の SSD デバイスを無視し、デバ</p>

オプション	説明
スを無視するには、チェックボックスを選択]	イス上のイメージ プロファイルとホスト プロファイルを保存しません。

- 1 [システム ディスク構成] ドロップダウン メニューで、[ユーザーがポリシー オプションを明確に選択することが必要] を選択します。
 - b [ホスト上で USB ディスクへのステートレス キャッシュを有効化] を選択した場合。
 - 1 [システム ディスク構成] ドロップダウン メニューで、[ホストのカスタマイズでユーザーによって指定されたシステム ディスク] を選択します。
 - 2 [ホストおよびクラスタ] で、ホストを右クリックし、[ホスト プロファイル] - [ホストのカスタマイズの編集] の順に選択します。[システム ディスク] プロパティの [値] フィールドに、パーシステント ストレージを持つディスクを定義します。
- 7 [保存] をクリックしてホスト プロファイルの設定を完了します。

次のステップ

vSphere Client のホスト プロファイル機能を使用して、個別のホストにホスト プロファイルを適用します。ホスト プロファイル ドキュメントを参照してください。もしくは、vSphere Client または PowerCLI を使用してホスト プロファイルをホストに割り当てるルールを作成します。ルールの記述とホストへのホスト プロファイルの割り当てを参照してください。

- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、ルールの記述とホストへのホスト プロファイルの割り当てを参照してください。
- vSphere Auto Deploy がすでにプロビジョニングされているホストの場合は、テストを実行し、PowerCLI セッションでコンプライアンスを修正します。ルールのコンプライアンスのテストおよび修復を参照してください。
- プロビジョニングされていないホストをパワーオンし、新しいホスト プロファイルを使用して、それらのホストがプロビジョニングされるようにします。

ステートフル インストールを有効にするホスト プロファイルの設定

vSphere Auto Deploy でプロビジョニングされた ESXi ホストをディスクから起動するように設定するには、ホスト プロファイルを構成する必要があります。

単一のホストにホスト プロファイルを設定できます。また、リファレンス ホストでホスト プロファイルを作成して、そのホスト プロファイルをほかのホストに適用することもできます。

前提条件

- イメージを保存するディスクを決定し、新しいイメージで既存の VMFS パーティションを上書きするかどうかを指定します。
- vSphere Auto Deploy 向けの環境を設定します。vSphere Auto Deploy のインストールと構成を参照してください。

- 最低でも 4GB の空き容量があるディスクが用意されていることを確認します。ディスクがまだパーティショニングされていない場合は、ホスト プロファイルを適用するときにパーティショニングが実行されます。
- ディスクから起動するようにホストを設定します。ハードウェア ベンダーのマニュアルを参照してください。
- ホスト プロファイルを作成します。ホスト プロファイル ドキュメントを参照してください。

手順

- 1 [ホーム] - [ポリシーおよびプロファイル] - [ホスト プロファイル] の順に移動します。
- 2 設定するホスト プロファイルをクリックし、[管理] タブを選択します。
- 3 [プロファイルの編集] をクリックします。
- 4 ウィザードの [ホストプロファイルの編集] ページで、[詳細な構成設定] - [システム イメージのキャッシュ構成] - [システム イメージのキャッシュ構成] の順に選択します。
- 5 [システム イメージのキャッシュ プロファイル設定] ドロップダウン メニューで、ポリシーのオプションを選択します。

オプション	説明
ホスト上でステートフル インストールを有効化	ディスクにイメージをキャッシュします。
ホスト上の USB ディスクへのステートフル インストールを有効化	ホストに接続された USB ディスクにイメージをキャッシュします。

6 選択したポリシー オプションに応じて、次の操作を行う必要があります。

a [ホスト上でステートフル インストールを有効化] を選択した場合。

1 使用するディスクに関する情報を指定します。

オプション	説明
[最初のディスクの引数]	<p>システム イメージ インストール ディスクを構成する場合は、複数のオプションを使用して、ESXi のインストール先および起動元となるデバイスを定義できます。インストール用のディスクを定義するには、次の引数を使用します。</p> <ul style="list-style-type: none"> ■ localesx - ESXi の有効なインストールを含む最初に検出されたディスク ■ local - 起動後に ESXi によって最初に検出されたローカル ディスク ■ Remotesx - ESXi の有効なインストールを含む最初に検出されたリモート ディスク ■ sortedremotesx - 検出された最も小さい LUN ID でソートされた、ESXi の有効なインストールを含む最初のリモート ディスク ■ remote - 起動後に ESXi によって最初に検出されたリモート ディスク ■ sortedremote - 起動後に ESXi によって検出された最も小さい LUN ID でソートされた最初のリモート ディスク ■ device model ■ device vendor ■ vmkernel device driver name <p>デバイス モデルとベンダーの引数の値を取得するには、コンソールで ESXi ホストに <code>esxcli storage core device list</code> コマンドを実行して、ルートとしてログインします。 <code>esxcli storage core adapter list</code> コマンドを実行して、vmkernel デバイス ドライバ名の引数を取得します。次に、起動デバイスが接続されているストレージアダプタを特定する必要があります。</p> <p>デフォルトでは、システムは既存の ESXi を置き換えてローカル ディスクに書き込もうとします。</p> <p>[最初のディスクの引数] フィールドで、使用するディスクのコンマ区切りリストを任意の順序で指定できます。ディスクは複数指定できます。ESX がインストールされている最初のディスクを示す localesx、またはモデルおよびベンダー情報を使用するか、vmkernel デバイス ドライバの名前を指定します。たとえば、モデル名が ST3120814A のディスク、mptsas ドライバを使用するディスク、ローカル ディスクの順にシステムで検索するには、このフィールドの値として ST3120814A,mptsas,local を指定します。</p> <p>ホスト プロファイルの最初のディスク設定では、キャッシュに使用するディスクを決める際の検索順序を指定します。検索順序は、値のコンマ区切りのリストとして指定します。デフォルト設定 localesx,local では、vSphere Auto Deploy が最初に既存のローカル キャッシュ ディスクを検索するように指定されています。キャッシュ ディスクは、既存の ESXi ソフトウェア イメージが保存されたディスクとして識別されます。vSphere Auto Deploy が既存のキャッシュ ディスクを見つけれない場合は、使用可能なローカル ディスク デバイスが検索されます。vSphere Auto Deploy は、既存の VMFS パーティションがない最初の空ディスクを使用して、使用可能なディスクを検索します。</p> <p>最初のディスク引数は、検索順序の指定にのみ使用できます。ディスクを明示的に指定することはできません。たとえば、SAN に特定の LUN を指定することはできません。</p>
[選択したディスクですべての VMFS ボリュームを上書きするには、チェックボックスを選択]	<p>このチェック ボックスをオンにすると、システムは、容量が不十分でイメージ、イメージ プロファイル、およびホスト プロファイルを保存できない場合、既存の VMFS ボリュームを上書きします。</p>
[ホストに接続されたすべての SSD デバイ	<p>このチェック ボックスをオンにすると、システムは既存の SSD デバイスを無視し、デバ</p>

オプション	説明
ス时无視するには、チェックボックスを選択]	イス上のイメージ プロファイルとホスト プロファイルを保存しません。

- 1 [システム ディスク構成] ドロップダウン メニューで、[ユーザーがポリシー オプションを明確に選択することが必要] を選択します。
 - b [ホスト上で USB ディスクへのステートフル インストールを有効化] を選択した場合。
 - 1 [システム ディスク構成] ドロップダウン メニューで、[ホストのカスタマイズでユーザーによって指定されたシステム ディスク] を選択します。
 - 2 [ホストおよびクラスタ] で、ホストを右クリックし、[ホスト プロファイル] - [ホストのカスタマイズの編集] の順に選択します。[システム ディスク] プロパティの [値] フィールドに、パーシステント ストレージを持つディスクを定義します。
- 7 [保存] をクリックしてホスト プロファイルの設定を完了します。

次のステップ

vSphere Client のホスト プロファイル機能を使用して、個別のホストにホスト プロファイルを適用します。ホスト プロファイル ドキュメントを参照してください。もしくは、vSphere Client または PowerCLI を使用してホスト プロファイルをホストに割り当てるルールを作成します。ルールの記述とホストへのホスト プロファイルの割り当てを参照してください。

- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、ルールの記述とホストへのホスト プロファイルの割り当てを参照してください。
- vSphere Auto Deploy がすでにプロビジョニングされているホストの場合は、テストを実行し、PowerCLI セッションでコンプライアンスを修正します。ルールのコンプライアンスのテストおよび修復を参照してください。
- プロビジョニングされていないホストをパワーオンし、新しいホスト プロファイルを使用して、それらのホストがプロビジョニングされるようにします。

vSphere Auto Deploy のリファレンス ホストの設定

ESXi ホストに状態が保存されていない環境では、複数のホストを同じ構成で設定する際に、リファレンス ホストが役立ちます。

ログ、コアダンプ、およびその他の必要な設定を指定してリファレンス ホストを構成し、ホスト プロファイルを保存し、ほかのホストへのホスト プロファイルに必要な応じて適用するルールを記述します。リファレンス ホストのストレージ、ネットワーク、およびセキュリティの設定を構成し、Syslog や NTP などのサービスを設定できます。

リファレンス ホストの設定について

適切に設計されたリファレンス ホストは、Syslog、NTP などのすべてのサービスに接続します。リファレンス ホストの設定には、セキュリティ、ストレージ、ネットワーク、および ESXi Dump Collector も含まれる場合があります。このようなホストの設定は、ホスト プロファイルを使用して他のホストに適用できます。

リファレンス ホストのセットアップ方法は環境に依存しますが、次ようなのカスタマイズが可能です。

NTP サーバのセットアップ

大規模環境でログ情報を収集するときは、ログの作成回数が適切に調整されていることを確認します。すべてのホストが共有できる、環境内の NTP サーバを使用するようにリファレンス ホストを設定します。NTP サーバは `esxcli system ntp set` コマンドを実行して指定できます。ホストの NTP サービスは `esxcli system ntp set` コマンドまたは vSphere Client を使用して開始および停止することができます。

Syslog サーバの設定

すべての ESXi ホストは、VMkernel およびその他のシステム コンポーネントからのメッセージをファイルに記録する、Syslog サービス (`vm syslogd`) を実行しています。 `esxcli system syslog` コマンドを実行するか、vSphere Client を使用して、ログ ホストの指定や、ログの場所、ローテーション、サイズなどの他の属性を管理できます。vSphere Auto Deploy でプロビジョニングされる、ローカル ストレージを持たないホストにとっては、リモート ホストにログ機能を設定することが特に重要となります。

必要に応じて、ログの集約と分析を行う VMware vCenter Log Insight をインストールできます。

コア ダンプの設定

共有の SAN LUN にコア ダンプを送るようにリファレンス ホストを設定するか、環境内に ESXi Dump Collector をインストールして、ESXi Dump Collector を使用するようにリファレンス ホストを構成することができます。 [ESXCLI を使用した、ESXi Dump Collector の構成](#) を参照してください。vCenter Server のインストール メディアを使用して ESXi Dump Collector をインストールするか、vCenter Server に含まれている ESXi Dump Collector を使用します。設定が完了すると、システムに重大なエラーが発生した場合に、VMkernel メモリは指定されたネットワーク サーバに送られます。

セキュリティの設定

ほとんどの環境では、vSphere Auto Deploy でプロビジョニングするすべてのホストで、セキュリティ設定を同じにする必要があります。たとえば、特定のサービスから ESXi システムへのアクセスを許可するようにファイアウォールを設定したり、vSphere Client または ESXCLI コマンドを使用して、リファレンス ホストのセキュリティ構成、ユーザー構成、およびユーザー グループ構成を設定することができます。セキュリティの設定には、すべてのホスト向けの共有ユーザー アクセス設定も含まれます。Active Directory を使用するようにリファレンス ホストを設定することで、統合されたユーザー アクセスを実現できます。『vSphere Security』ドキュメントを参照してください。

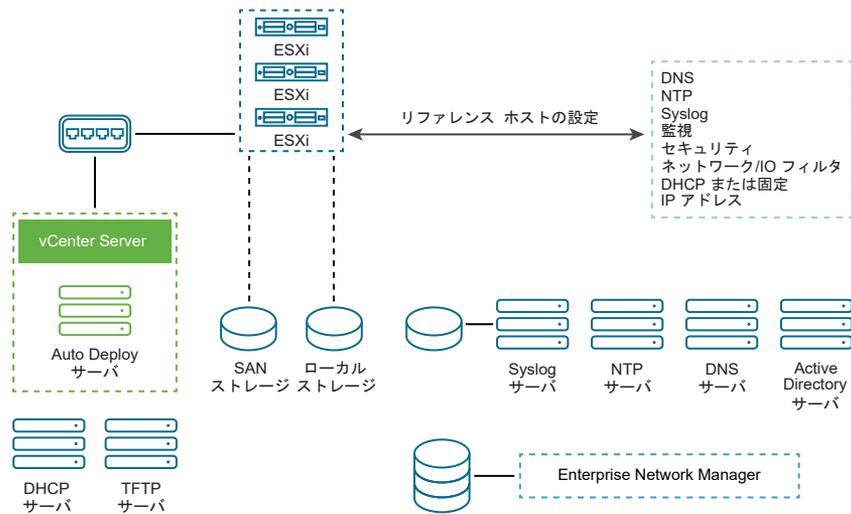
注： ホスト プロファイルを使用して Active Directory を設定する場合は、パスワードは保護されません。Active Directory パスワードの漏洩を防ぐように Active Directory を設定するには、vSphere 認証サービスを使用します。

ネットワークおよびストレージの設定

vSphere Auto Deploy でプロビジョニングするホストで使用するために、一連のネットワーク リソースとストレージ リソースを予約する場合、これらのリソースを使用するようにリファレンス ホストを設定することができます。

大規模なデプロイでは、環境内で実行されているさまざまな監視サービスからあらゆる情報を収集する Enterprise Network Manager が、リファレンス ホストの設定でサポートされます。

図 4-9. vSphere Auto Deploy のリファレンス ホストのセットアップ



vSphere Auto Deploy リファレンス ホストの構成オプションでは、セットアップの方法について説明していません。

リファレンス ホストのセットアップについては、ビデオ『Auto Deploy のリファレンス ホスト』をご覧ください。



(vSphere Auto Deploy のリファレンス ホスト)

vSphere Auto Deploy リファレンス ホストの構成オプション

vSphere Client、ESXCLI、またはホスト プロファイルを使用してリファレンス ホストを構成できます。

リファレンス ホストを設定するために、最も適した方法を使用することができます。

vSphere Client

vSphere Client は、ネットワーク、ストレージ、セキュリティ、およびその他の ESXi ホストのほとんどの要素の設定をサポートします。環境を設定し、vSphere Auto Deploy で使用するためのホスト プロファイルを作成できます。

ESXCLI

ESXCLI を使用して、ホストのさまざまな設定を行うことができます。ESXCLI は、vSphere 環境でさまざまなサービスを構成するのに適しています。コマンドには、NTP サーバを設定する `esxcli system ntp`、Syslog サーバを設定する `esxcli system syslog`、ルートを追加し、デフォルト ルートを設定する `esxcli network route`、および ESXi Dump Collector を構成する `esxcli system coredump` が含まれます。

ホスト プロファイル機能

ベスト プラクティスは、vSphere Client または ESXCLI でホストを設定し、そのホストでホスト プロファイルを作成することです。代わりに、vSphere Client でホスト プロファイル機能を使用してホスト プロファイルを保存することもできます。

vSphere Auto Deploy は、ホスト プロファイルのすべての共通設定を、すべてのターゲット ホストに適用します。ユーザー入力を求めるようにホスト プロファイルを設定する場合、そのホスト プロファイルを使用してプロビジョニングされるすべてのホストはメンテナンス モードになります。ホスト固有の情報についてプロンプトが表示されるためには、ホスト プロファイルを再適用するか、ホストのカスタマイズ内容をリセットする必要があります。

ESXi Dump Collector の構成

コア ダンプを保存するように ESXi Dump Collector を構成するには、ESXCLI コマンドを使用するか、リファレンス ホストを構成します。

vSphere Auto Deploy でプロビジョニングされるホストには、コア ダンプを保存するためのローカル ディスクがありません。ESXi Dump Collector でコア ダンプを保存する方法として、ESXCLI コマンドを使用するように、またはリファレンス ホストが vSphere Client のホスト プロファイル機能を使用して ESXi Dump Collector を使用するように構成することができます。

ESXCLI を使用した、ESXi Dump Collector の構成

vSphere Auto Deploy でプロビジョニングされるホストには、コア ダンプを保存するためのローカル ディスクがありません。ESXCLI コマンドを使用して ESXi Dump Collector を構成し、デバッグ中に使用できるようにコア ダンプをネットワーク サーバに保持することができます。

コア ダンプは、ホストで障害が発生した場合の、動作中のメモリの状態です。デフォルトでは、コア ダンプはローカル ディスクに保存されます。ESXi Dump Collector は、vSphere Auto Deploy に特に有用ですが、すべての ESXi ホストでサポートされます。ESXi Dump Collector は、コア ダンプのローカル ディスクへの送信などその他のカスタマイズをサポートし、vCenter Server 管理ノードに含まれています。

注： ESXi Dump Collector は、NSX-T N-VDS スイッチで実行されている VMkernel インターフェイスでは構成できません。

IPv6 を使用し、ESXi ホストと ESXi Dump Collector の両方を同じローカル リンクで使用する場合は、その両方で、ローカル リンク スコープの IPv6 アドレスまたはグローバル スコープの IPv6 アドレスのどちらかを使用できます。

IPv6 を使用し、ESXi と ESXi Dump Collector が異なるホストに存在する場合は、その両方でグローバル スコープの IPv6 アドレスが必要になります。トラフィックは、デフォルトの IPv6 ゲートウェイを介して経路選択されず。

前提条件

ESXi Dump Collector を使用するようにホストを構成するには、ESXCLI をインストールします。トラブルシューティングが必要な場合は、代わりに ESXi Shell 内の ESXCLI を使用できます。

手順

- 1 ローカルの ESXi Shell で `esxcli system coredump` を実行するか、または ESXCLI を使用することにより、ESXi Dump Collector を使用するように ESXi システムを設定します。

```
esxcli system coredump network set --interface-name vmk0 --server-ip 10xx.xx.xx.xx --server-port 6500
```

コア ダンプの送信先サーバの VMkernel の NIC、IP アドレス、およびポート（オプション）を指定する必要があります。IPv4 アドレスまたは IPv6 アドレスを使用できます。vSphere Standard スイッチを使用している仮想マシンで実行されている ESXi システムを構成する場合は、無差別モードの VMkernel ポートを選択する必要があります。

2 ESXi Dump Collector を有効にします。

```
esxcli system coredump network set --enable true
```

3（オプション） ESXi Dump Collector が正しく構成されていることを確認します。

```
esxcli system coredump network check
```

結果

ESXi Dump Collector を設定したホストは、指定された VMkernel の NIC とオプションのポートを使用することによって、指定されたサーバにコア ダンプを送信するように構成されます。

次のステップ

- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、[ルールの記述とホストへのホスト プロファイルの割り当て](#)を参照してください。
- vSphere Auto Deploy がすでにプロビジョニングされているホストの場合は、テストを実行し、PowerCLI セッションでコンプライアンスを修正します。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストをパワーオンし、新しいホスト プロファイルを使用して、それらのホストがプロビジョニングされるようにします。

vSphere Client のホスト プロファイル機能による ESXi Dump Collector の構成

vSphere Auto Deploy でプロビジョニングされるホストには、コア ダンプを保存するためのローカル ディスクがありません。vSphere Client のホスト プロファイル機能を使用してリファレンス ホストを構成し、ESXi Dump Collector を使用することができます。

esxcli system coredump コマンドを使用して、ESXi Dump Collector を使用するホストを設定し、そのホストのホスト プロファイルを保存することがベスト プラクティスです。詳細については、[ESXCLI を使用した、ESXi Dump Collector の構成](#) を参照してください。

前提条件

- コアダンプ ポリシーを構成するホスト プロファイルを作成したことを確認します。ホスト プロファイルの作成方法の詳細については、『vSphere のホスト プロファイル』ドキュメントを参照してください。
- vSphere Auto Deploy を使用してプロビジョニングされる複数のホストからのコア ダンプ用に、少なくとも 1 つのパーティションで十分なストレージ機能があることを確認します。

手順

- 1 [ホーム] - [ポリシーおよびプロファイル] - [ホスト プロファイル] の順に移動します。

- 2 設定するホスト プロファイルをクリックし、[管理] タブを選択します。
- 3 [プロファイルの編集] をクリックします。
- 4 [ネットワーク設定] - [ネットワーク コアダンプの設定] の順に選択します。
- 5 [有効化] チェック ボックスを選択します。
- 6 使用するホスト NIC、ネットワーク コアダンプ サーバ IP アドレス、ネットワーク コアダンプ サーバ ポートを指定します。
- 7 [保存] をクリックしてホスト プロファイルの設定を完了します。

次のステップ

- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、[ルールの記述とホストへのホスト プロファイルの割り当て](#)を参照してください。
- vSphere Auto Deploy がすでにプロビジョニングされているホストの場合は、テストを実行し、PowerCLI セッションでコンプライアンスを修正します。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストをパワーオンし、新しいホスト プロファイルを使用して、それらのホストがプロビジョニングされるようにします。

vSphere Client でのホスト プロファイルによる Syslog の構成

ホスト プロファイルを適用して、リモート Syslog サーバを指定します。

vSphere Client または `esxcli system syslog` コマンドを使用して、リファレンス ホスト上に Syslog サーバを設定し、ホスト プロファイルを保存することがベスト プラクティスです。vSphere Client のホスト プロファイル機能から Syslog を設定することもできます。

前提条件

vSphere Auto Deploy でプロビジョニングされるホストには、通常、システム ログを保存するための十分なローカルストレージがありません。これらのホストには、リファレンス ホストを設定し、ホスト プロファイルを保存し、そのホスト プロファイルを必要に応じてほかのホストに適用することで、リモートの Syslog サーバを指定できます。

- リモートの Syslog ホストを使用する場合は、ホスト プロファイルをカスタマイズする前に対象のホストを設定します。
- vSphere Client および vCenter Server システムにアクセスできることを確認します。

手順

- 1 [ホーム] - [ポリシーおよびプロファイル] - [ホスト プロファイル] の順に移動します。
- 2 (オプション) 環境内にリファレンス ホストがない場合、[ホスト プロファイルの抽出] をクリックしてホスト プロファイルを作成します。
- 3 設定するホスト プロファイルをクリックし、[管理] タブを選択します。

- 4 [プロファイルの編集] をクリックします。
- 5 [詳細設定] - [詳細オプション] - [詳細設定オプション] の順に選択します。
特定のサブプロファイルを選択し、Syslog 設定を編集することができます。
- 6 (オプション) 詳細設定オプションを作成するには、次の手順に従います。
 - a [サブプロファイルの追加] アイコンをクリックします。
 - b [詳細オプション] ドロップダウン リストから、[固定オプションを構成] を選択します。
 - c オプションの名前として Syslog.global.loghost を指定し、オプションの値としてホストを指定します。
- 7 [保存] をクリックしてホスト プロファイルの設定を完了します。

次のステップ

- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、[ルールの記述とホストへのホスト プロファイルの割り当て](#)を参照してください。
- vSphere Auto Deploy がすでにプロビジョニングされているホストの場合は、テストを実行し、PowerCLI セッションでコンプライアンスを修正します。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストをパワーオンし、新しいホスト プロファイルを使用して、それらのホストがプロビジョニングされるようにします。

vSphere Client でのリファレンス ホストの NTP クライアントの有効化

大規模な環境でログ情報を収集するときは、ログの回数が調整されていることを確認する必要があります。

リファレンス ホストを設定すると、環境内での NTP サーバの使用、ホスト プロファイルの抽出、vSphere Auto Deploy ルールの作成が可能になります。

手順

- 1 [ホーム] - [ホストおよびクラスタ] の順に移動し、リファレンス ホストとして使用する ESXi ホストを選択します。
- 2 [構成] タブを選択します。
- 3 [システム] の下で、[時間の構成] を選択して [編集] をクリックします。
- 4 [Network Time Protocol を使用 (NTP クライアントを有効にする)] ラジオ ボタンを選択します。
このオプションはホストの日時を NTP サーバの日時と同期します。ホストの NTP サービスは、NTP サーバから時刻と日付を定期的に取得します。
- 5 [NTP サービス起動ポリシー] ドロップダウン メニューから、[ホストと連動して起動および停止] を選択します。
- 6 [NTP サーバ] テキスト ボックスに、使用する NTP サーバの IP アドレスまたはホスト名を入力します。
- 7 [OK] をクリックします。

次のステップ

- リファレンス ホストからホスト プロファイルを抽出します。ホスト プロファイル ドキュメントを参照してください。
- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、[ルールの記述とホストへのホスト プロファイルの割り当て](#)を参照してください。

vSphere Client での vSphere Auto Deploy ホスト用ネットワークの設定

vSphere Auto Deploy のリファレンス ホストのネットワークを設定し、そのホストのホスト プロファイルをそのホストすべてのホストに適用して、完全に機能するネットワーク環境を確保します。

前提条件

リファレンス ホストとして使用するホストを、vSphere Auto Deploy で ESXi イメージを使用してプロビジョニングします。

手順

- 1 [ホーム] - [ホストおよびクラスター] の順に移動し、リファレンス ホストとして使用する ESXi ホストを選択します。
- 2 [設定] タブをクリックし、[ネットワーク] に移動します。
- 3 ネットワークの設定を実行します。
vSphere Distributed Switch ではなく仮想スイッチを使用している場合は、vSwitch0 にほかの VMkernel NIC を追加しないでください。
- 4 リファレンス ホストを構成したら、システムを再起動して vmkO が管理ネットワークに接続されていることを確認します。
- 5 リファレンス ホストにホスト プロファイルが存在しない場合は、ホスト プロファイルを作成します。

次のステップ

- リファレンス ホストで指定した設定でプロビジョニングするすべてのホストに、ホスト プロファイルを適用するルールを作成します。PowerCLI セッションでルールを作成する方法については、[ルールの記述とホストへのホスト プロファイルの割り当て](#)を参照してください。
- vSphere Auto Deploy がすでにプロビジョニングされているホストの場合は、テストを実行し、PowerCLI セッションでコンプライアンスを修正します。 [ルールのコンプライアンスのテストおよび修復](#) を参照してください。
- プロビジョニングされていないホストをパワーオンし、新しいホスト プロファイルを使用して、それらのホストがプロビジョニングされるようにします。

自動パーティショニングのためのリファレンス ホストの構成

vSphere Auto Deploy でプロビジョニングするすべてのホストに対して自動パーティショニングを行うようにリファレンス ホストを作成できます。

注意： 自動パーティショニングのデフォルトの動作を変更すると、パーティションの内容に関係なく既存のパーティションが vSphere Auto Deploy によって上書きされます。このオプションをオンにする場合は、誤ってデータが失われないことを確認してください。

自動パーティション化の際にローカル SSD がパーティション化されないようにするには、リファレンス ホストでパラメータ **skipPartitioningSsds=TRUE** を設定する必要があります。

自動パーティション化の際に SSD のフォーマットを避けるための詳細については、『vSphere のストレージ』ドキュメントを参照してください。

前提条件

デフォルトでは、ホストでパーティションが使用できる場合にのみ、vSphere Auto Deploy はホストをプロビジョニングします。自動パーティション オプションによりホストのローカルストレージに VMFS データストアが作成されます。vSphere Auto Deploy でプロビジョニングするすべてのホストに対して自動パーティショニングを行うようにリファレンス ホストを設定できます。

- リファレンス ホストとして使用するホストを、vSphere Auto Deploy で ESXi イメージを使用してプロビジョニングします。
- vCenter Server システムに接続できる vSphere Client にアクセスできることを確認します。

手順

- 1 [ホーム] - [ホストおよびクラスタ] の順に移動し、リファレンス ホストとして使用する ESXi ホストを選択します。
- 2 [構成] タブを選択します。
- 3 [システム] で、[システムの詳細設定] の順に選択し、[編集] をクリックします。
- 4 VMkernel.Boot.autoPartition キーを検索し、値を [true] に設定します。
- 5 (オプション) ローカル SSD がパーティション化されないようにするには、VMkernel.Boot.skipPartitioningSsds キーを検索して、値を [true] に設定します。
- 6 [OK] をクリックします。
- 7 リファレンス ホストにホスト プロファイルが存在しない場合は、ホスト プロファイルを作成します。

結果

自動パーティショニングは、ホストが起動すると実行されます。

次のステップ

- vSphere Auto Deploy を使用して、リファレンス ホストのホスト プロファイルを起動時にすべてのホストにすぐに適用するルールを作成します。vSphere Client を使用してルールを作成する方法については、[デプロイルールの作成](#)を参照してください。PowerCLI セッションでルールを作成する方法については、[ルールの記述とホストへのホスト プロファイルの割り当て](#)を参照してください。

ステートレス ホストからステートフル ホストへの変換

ステートレス ESXi に物理ストレージを追加し、ステートフル ESXi ホストに変換して、イメージで管理するクラスタにホストを追加することができます。

イメージで管理されているクラスタは、プリブート実行環境 (PXE) ブートを使用するステートレス ESXi ホストをサポートしていないため、ESXi ホストには物理ストレージが接続されている必要があります。

起動プロセス中にイメージ コンポーネントを ESXi ホストの物理ディスクに確実にインストールするために、Auto Deploy によって、ホスト プロファイルが PXE ブート イメージの一部であることが確認されます。ホスト プロファイルには、「ホスト上でステートフル インストールを有効化」または「ホスト上の USB ディスクへのステートフル インストールを有効化」に設定された「システム イメージ キャッシュ プロファイル設定」ポリシーが含まれている必要があります。添付されたホスト プロファイルにこのポリシーが含まれていない場合、またはポリシーの設定が異なる場合は、ステートフル インストールをサポートするようにポリシーが自動的に設定されます。ホスト プロファイルがない場合は、新しいホスト プロファイルがクラスタに添付されます。このプロファイルには、ステートフル インストールの設定を反映した「システム イメージ キャッシュ プロファイル設定」ポリシーのみが含まれます。

ディスクレス ESXi ホストの変換

イメージによって管理されるクラスタにディスクレス ESXi ホストを追加するには、物理ストレージを追加し、ステートフル ホストに変換して、ESXi 8.0 イメージをインストールします。

ステートレス キャッシュおよびステートフル インストールの詳細については、[ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用](#)を参照してください。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、『VMware ESXi のインストールとセットアップ』ドキュメントの「vSphere Auto Deploy 用にシステムを準備」を参照してください。
- ESXi ホストはイメージ プロファイルが含まれている Auto Deploy ルールから起動していること。
- Auto Deploy サービスが実行されていること。
- 空の USB スティックまたはその他のサポートされているストレージがあること。

手順

- 1 ホーム メニューで [ポリシーおよびプロファイル] をクリックし、[ホスト プロファイル] を選択します。
- 2 実行中の ESXi ホストからホスト プロファイルを抽出するか、指定された構成およびホストの場所を持つ既存のホスト プロファイルを複製します。

既存のホスト プロファイルは、ホストまたはクラスタに添付できます。

- 3 新しいホスト プロファイルを右クリックし、[ホスト プロファイルの編集] を選択して、[詳細設定] - [システム イメージ キャッシュ設定] - [システム イメージ キャッシュ設定] の順に移動します。

- 4 [システム イメージのキャッシュ プロファイル設定] ドロップダウン メニューで、ポリシーのオプションを選択します。

オプション	説明
ホスト上でステートレス キャッシュを有効化	ディスクにイメージをキャッシュします。
ホスト上で USB ディスクへのステートレス キャッシュを有効化	ホストに接続された USB ディスクにイメージをキャッシュします。

- 5 選択したポリシー オプションに応じて、次の操作を行う必要があります。
- a [ホスト上でステートレス キャッシュを有効化] を選択した場合。
 - 1 最初のディスクに対する引数を入力し、必要に応じてチェック ボックスを選択します。
 - 2 [システム ディスク構成] ドロップダウン メニューで、[ユーザーがポリシー オプションを明確に選択することが必要] を選択します。
 - b [ホスト上で USB ディスクへのステートレス キャッシュを有効化] を選択した場合。
 - 1 [システム ディスク構成] ドロップダウン メニューで、[ホストのカスタマイズでユーザーによって指定されたシステム ディスク] を選択します。
 - 2 [ホストおよびクラスタ] で、ホストを右クリックし、[ホスト プロファイル] - [ホストのカスタマイズの編集] の順に選択します。[システム ディスク] プロパティの [値] フィールドに、パーシステント ストレージを持つディスクを定義します。
- 6 [保存] をクリックしてホスト プロファイルの設定を完了します。
- 7 既存のホスト プロファイルを複製していて、ホスト プロファイルがクラスタに添付されている場合は、新しいホスト プロファイルをクラスタに添付します。
- 8 ホーム メニューから [Auto Deploy] をクリックします。
- 9 イメージ プロファイルが含まれている Auto Deploy ルールを無効にして、[編集] をクリックします。デプロイのルールを編集するダイアログ ボックスが表示されます。
- 10 ウィザードの [イメージ プロファイルの選択] 画面で、ESXi 8.0 イメージ プロファイルを選択します。
- 11 ウィザードの [ホスト プロファイルの選択] 画面で、新しいホスト プロファイルを選択します。
- 12 ルールを有効にして、順序付けされたリスト内の初期位置に、そのルールを移動します。
- 13 [デプロイ済みのホスト] タブで、単一または複数の ESXi ホストを選択します。
- 14 ESXi ホストについて、[ホストの関連付けの修正] をクリックします。
修正プロセスの進行状況は、[最近のタスク] ペインで監視できます。
- 15 修正した ESXi ホストをシャットダウンします。
- 16 ESXi ホストがパワーオフされたら、選択されているストレージをローカル起動ディスクとして取り付けます。

- 17 各 ESXi ホストをパワーオンし、BIOS/UEFI セットアップに入り、新しく追加されたストレージから最初に起動し、次にネットワークから起動するよう起動順序を変更します。

新しく追加されたストレージは空であるため、各 ESXi ホストはネットワークから起動し、ストレージ上で以前に指定した ESXi 8.0 イメージをインストールします。インストールが完了すると、各 ESXi ホストが再起動し、新しく追加されたストレージから起動します。

結果

ESXi ホストは、デフォルトで新しいストレージから起動し、ESXi 8.0 イメージが標準 DVD からインストールされたように動作します。

次のステップ

1 つのイメージで管理されているクラスタに ESXi ホストを追加します。詳細については、『vCenter Server およびホストの管理』ドキュメントの「ホストの追加」を参照してください。

ステートレス キャッシュを有効にしたステートレス ESXi ホストの変換

ステートレス ESXi ホストを、イメージで管理されているクラスタに追加するには、標準の 8.0 イメージをインストールして、ホストをステートフル ホストに変換します。

ステートレス キャッシュおよびステートフル インストールの詳細については、[ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用](#)を参照してください。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、『VMware ESXi のインストールとセットアップ』ドキュメントの「vSphere Auto Deploy 用にシステムを準備」を参照してください。
- ESXi ホストはイメージ プロファイルが含まれている Auto Deploy ルールから起動していること。
- Auto Deploy サービスが実行されていること。

手順

- 1 ホーム メニューで [ポリシーおよびプロファイル] をクリックし、[ホスト プロファイル] を選択します。
- 2 実行中の ESXi ホストからホスト プロファイルを抽出するか、指定された構成およびホストの場所を持つ既存のホスト プロファイルを複製します。

既存のホスト プロファイルは、ホストまたはクラスタに添付できます。
- 3 新しいホスト プロファイルを右クリックし、[ホスト プロファイルの編集] を選択して、[詳細設定] - [システム イメージ キャッシュ設定] - [システム イメージ キャッシュ設定] の順に移動します。
- 4 ドロップダウン メニューから [ホスト上でステートフル インストールを有効化] を選択し、[最初のディスクの引数] テキスト ボックスに local と入力し、[選択したディスクですべての VMFS ボリュームを上書きするには、チェックボックスを選択] チェック ボックスを選択します。
- 5 既存のホスト プロファイルを複製していて、ホスト プロファイルがクラスタに添付されている場合は、新しいホスト プロファイルをクラスタに添付します。
- 6 ホーム メニューから [Auto Deploy] をクリックします。

- 7 イメージ プロファイルが含まれている Auto Deploy ルールを無効にして、[編集] をクリックします。
デプロイのルールを編集するダイアログ ボックスが表示されます。
- 8 ウィザードの [イメージ プロファイルの選択] 画面で、ESXi 8.0 イメージ プロファイルを選択します。
- 9 ウィザードの [ホスト プロファイルの選択] 画面で、新しいホスト プロファイルを選択します。
- 10 ルールを有効にして、順序付けされたリスト内の初期位置に、そのルールを移動します。
- 11 [デプロイ済みのホスト] タブで、単一または複数の ESXi ホストを選択します。
- 12 ESXi ホストについて、[ホストの関連付けの修正] をクリックします。
修正プロセスの進行状況は、[最近のタスク] ペインで監視できます。

13 ESXi ホストを再起動します。

ESXi ホストにレガシー BIOS が搭載されている場合は、BIOS セットアップの起動順序を変更して、最初にローカル ストレージから起動できます。UEFI ベースの ESXi ホストの場合は、再起動中に起動順序が自動的に変更されます。

各 ESXi ホストはネットワークから起動し、以前キャッシュに使用したストレージ上で指定した ESXi 8.0 イメージをインストールします。インストールが完了すると、各 ESXi ホストが再起動し、ローカル ストレージから起動します。

結果

ESXi ホストは、デフォルトで新しいストレージから起動し、ESXi 8.0 イメージが標準 DVD からインストールされたように動作します。

次のステップ

1 つのイメージで管理されているクラスタに ESXi ホストを追加します。詳細については、『vCenter Server およびホストの管理』ドキュメントの「ホストの追加」を参照してください。

ローカル ディスク上の単一 VMFS パーティションを使用したステートレス ESXi ホストの変換

ステートレス ESXi ホストを、イメージで管理されているクラスタに追加するには、VMFS パーティションの再パーティショニングと標準の ESXi 8.0 イメージのインストールを行い、最初にホストをステートフル ホストに変換する必要があります。

ステートレス キャッシュおよびステートフル インストールの詳細については、[ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用](#)を参照してください。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、『VMware ESXi のインストールとセットアップ』ドキュメントの「vSphere Auto Deploy 用にシステムを準備」を参照してください。
- ESXi ホストはイメージ プロファイルが含まれている Auto Deploy ルールから起動していること。
- Auto Deploy サービスが実行されていること。

手順

- 1 ホーム メニューで [ポリシーおよびプロファイル] をクリックし、[ホスト プロファイル] を選択します。
- 2 実行中の ESXi ホストからホスト プロファイルを抽出するか、指定された構成およびホストの場所を持つ既存のホスト プロファイルを複製します。
既存のホスト プロファイルは、ホストまたはクラスタに添付できます。
- 3 新しいホスト プロファイルを右クリックし、[ホスト プロファイルの編集] を選択して、[詳細設定] - [システム イメージ キャッシュ設定] - [システム イメージ キャッシュ設定] の順に移動します。
- 4 ドロップダウン メニューから [ホスト上でステートフル インストールを有効化] を選択し、[最初のディスクの引数] テキスト ボックスに `localesx` と入力します。
- 5 既存のホスト プロファイルを複製していて、ホスト プロファイルがクラスタに添付されている場合は、新しいホスト プロファイルをクラスタに添付します。
- 6 ホーム メニューから [Auto Deploy] をクリックします。
- 7 イメージ プロファイルが含まれている Auto Deploy ルールを無効にして、[編集] をクリックします。
デプロイのルールを編集するダイアログ ボックスが表示されます。
- 8 ウィザードの [イメージ プロファイルの選択] 画面で、ESXi 8.0 イメージ プロファイルを選択します。
- 9 ウィザードの [ホスト プロファイルの選択] 画面で、新しいホスト プロファイルを選択します。
- 10 ルールを有効にして、順序付けされたリスト内の初期位置に、そのルールを移動します。
- 11 [デプロイ済みのホスト] タブで、単一または複数の ESXi ホストを選択します。
- 12 ESXi ホストについて、[ホストの関連付けの修正] をクリックします。
修正プロセスの進行状況は、[最近のタスク] ペインで監視できます。
- 13 ESXi ホストを再起動します。

ESXi ホストにレガシー BIOS が搭載されている場合は、BIOS セットアップの起動順序を変更して、最初にローカル ストレージから起動できます。UEFI ベースの ESXi ホストの場合は、再起動中に起動順序が自動的に変更されます。

各 ESXi ホストはネットワークから起動し、VMFS パーティションを標準の ESXi インストール パーティションに再パーティショニングして、以前にパーティションで指定した ESXi 8.0 イメージをインストールします。インストールが完了すると、各 ESXi ホストが再起動し、標準 ESXi パーティションから起動します。

結果

ESXi ホストはデフォルトで、新しいパーティションから起動し、ESXi 8.0 イメージが標準 DVD からインストールされたように動作します。

次のステップ

1つのイメージで管理されているクラスタに ESXi ホストを追加します。詳細については、『vCenter Server およびホストの管理』ドキュメントの「ホストの追加」を参照してください。

リモート ディスク上の単一 VMFS パーティションを使用したステートレス ESXi ホストの変換

ステートレス ESXi ホストを、イメージで管理されているクラスタに追加するには、リモートの VMFS パーティションの再パーティショニングと標準の ESXi 8.0 イメージのインストールを行い、最初にホストをステートフル ホストに変換する必要があります。たとえば、ESXi ホストは、ファイバ チャネル SAN または iSCSI SAN から起動できます。

ステートレス キャッシュおよびステートフル インストールの詳細については、[ステートレス キャッシュおよびステートフル インストールの vSphere Auto Deploy の使用](#)を参照してください。

リモート ディスクからの起動の詳細については、『vSphere ストレージ』ドキュメントの「iSCSI SAN からの起動」を参照してください。

前提条件

- システムの準備および Auto Deploy サーバのインストール詳細については、『VMware ESXi のインストールとセットアップ』ドキュメントの「vSphere Auto Deploy 用にシステムを準備」を参照してください。
- ESXi ホストはイメージ プロファイルが含まれている Auto Deploy ルールから起動していること。
- Auto Deploy サービスが実行されていること。

手順

- 1 ホーム メニューで [ポリシーおよびプロファイル] をクリックし、[ホスト プロファイル] を選択します。
- 2 実行中の ESXi ホストからホスト プロファイルを抽出するか、指定された構成およびホストの場所を持つ既存のホスト プロファイルを複製します。

既存のホスト プロファイルは、ホストまたはクラスタに添付できます。
- 3 新しいホスト プロファイルを右クリックし、[ホスト プロファイルの編集] を選択して、[詳細設定] - [システム イメージ キャッシュ設定] - [システム イメージ キャッシュ設定] の順に移動します。
- 4 ドロップダウン メニューから [ホスト上でステートフル インストールを有効化] を選択し、[最初のディスクの引数] テキスト ボックスに `remoteesx, remote` または `sortedremoteesx, sortedremote` と入力します。
- 5 既存のホスト プロファイルを複製していて、ホスト プロファイルがクラスタに添付されている場合は、新しいホスト プロファイルをクラスタに添付します。
- 6 ホーム メニューから [Auto Deploy] をクリックします。
- 7 イメージ プロファイルが含まれている Auto Deploy ルールを無効にして、[編集] をクリックします。

デプロイのルールを編集するダイアログ ボックスが表示されます。
- 8 ウィザードの [イメージ プロファイルの選択] 画面で、ESXi 8.0 イメージ プロファイルを選択します。
- 9 ウィザードの [ホスト プロファイルの選択] 画面で、新しいホスト プロファイルを選択します。
- 10 ルールを有効にして、順序付けされたリスト内の初期位置に、そのルールを移動します。
- 11 [デプロイ済みのホスト] タブで、単一または複数の ESXi ホストを選択します。

12 ESXi ホストについて、[ホストの関連付けの修正] をクリックします。

修正プロセスの進行状況は、[最近のタスク] ペインで監視できます。

13 ESXi ホストを再起動します。

各 ESXi ホストはネットワークから起動し、VMFS パーティションを標準の ESXi インストール パーティションに再パーティショニングして、以前にパーティションで指定した ESXi 8.0 イメージをインストールします。インストールが完了すると、各 ESXi ホストが再起動し、リモートの標準 ESXi パーティションから起動します。

結果

ESXi ホストはデフォルトで、新しいパーティションから起動し、ESXi 8.0 イメージが標準 DVD からインストールされたように動作します。

次のステップ

1 つのイメージで管理されているクラスタに ESXi ホストを追加します。詳細については、『vCenter Server およびホストの管理』ドキュメントの「ホストの追加」を参照してください。

vSphere Auto Deploy のベスト プラクティス

大規模な本番環境で、またはステートレス キャッシュを使用する場合、可用性の高い vSphere Auto Deploy インフラストラクチャを設定します。

vSphere Auto Deploy のインストール時や、vSphere Auto Deploy をその他の vSphere コンポーネントと連携して使用する際には、ベスト プラクティスに従います。大規模な本番環境で、またはステートレス キャッシュを使用する場合、可用性の高い vSphere Auto Deploy インフラストラクチャを設定します。PXE ブート環境のすべてのセキュリティ ガイドラインに従い、この章の推奨事項について考慮してください。

vSphere Auto Deploy のベスト プラクティス

vSphere Auto Deploy のベスト プラクティスに従い、ネットワークを設定したり、vSphere HA を構成したり、vSphere Auto Deploy を使用する環境を最適化したりできます。

追加のベスト プラクティスの情報は、VMware のナレッジ ベースを参照してください。

vSphere Auto Deploy および vSphere HA のベスト プラクティス

ベスト プラクティスに従うと、vSphere Auto Deploy でプロビジョニングされるホスト上で実行されている仮想マシンの可用性を向上することができます。

一部の環境では、vSphere Auto Deploy でプロビジョニングされるホストを Distributed Switch で構成したり、ホスト上で実行している仮想マシンを Auto Start Manager で構成します。そのような環境では、vCenter Server システムの可用性と vSphere Auto Deploy サーバの可用性が一致するように、vCenter Server システムをデプロイします。これを実現する手法はいくつかあります。

- vCenter Server をデプロイします。vSphere Auto Deploy サーバが含まれています。
- vSphere HA が有効なクラスタ内の vCenter Server を実行し、vSphere HA の再起動の優先順位を高くするようにその仮想マシンを構成します。vSphere Auto Deploy で管理されていないクラスタ内のホストを 2

台以上含め、ルールを使用して、これらのホストに vCenter Server 仮想マシンを固定します (vSphere HA DRS を必要とする仮想マシンをホストに固定するルール)。クラスタ内で DRS を使用しない場合は、ルールを設定してから DRS を無効にすることができます。vSphere Auto Deploy によって管理されていないホストの数が多く、ホストのエラーに対する復元力が強化されます。

注： このアプローチは、Auto Start Manager を使用する場合には適しません。Auto Start Manager は、vSphere HA が有効なクラスタではサポートされません。

vSphere Auto Deploy ネットワークのベスト プラクティス

vSphere Auto Deploy のネットワーク設定のベスト プラクティスに従うことで、ネットワーク設定の問題を防ぐことができます。

vSphere Auto Deploy および IPv6

vSphere Auto Deploy は iPXE インフラストラクチャを利用するため、vSphere Auto Deploy によるプロビジョニングを計画しているホストにレガシー BIOS がある場合、vSphere Auto Deploy サーバでは IPv4 アドレスを使用する必要があります。レガシー BIOS ファームウェアを使用する PXE ブートは IPv4 のみ実行できます。UEFI ファームウェアを使用する PXE ブートは IPv4 または IPv6 のいずれかで実行できます。

IP アドレスの割り当て

アドレスの割り当てでは、DHCP 予約を使用します。固定 IP アドレスはホストのカスタマイズ メカニズムでサポートされていますが、ホストごとに入力することは推奨されません。

VLAN の考慮事項

vSphere Auto Deploy は、VLAN を使用しない環境で使用してください。

VLAN を使用する環境で vSphere Auto Deploy を使用する場合は、プロビジョニング対象のホストが DHCP サーバにアクセスできることを確認します。ホストがどのように VLAN に割り当てられるかは、サイトの設定に依存します。VLAN ID は、スイッチまたはルーターによって割り当てられる場合や、ホストの BIOS で設定されるか、またはホスト プロファイル経由で設定される場合があります。ネットワーク管理者に問い合わせ、DHCP サーバへの到達をホストに許可する手順を確認します。

vSphere Auto Deploy および VMware Tools のベスト プラクティス

vSphere Auto Deploy を装備したホストをプロビジョニングする際に、VMware Tools を含むイメージ プロファイルを選択することも、VMware Tools を含まないイメージ プロファイルと関連付けられた小さいイメージを選択することもできます。

VMware ダウンロード サイトから 2 種類のイメージ プロファイルをダウンロードできます。

- `xxxxx-standard` : 仮想マシン内で実行するゲスト OS で必要な VMware Tools バイナリを含むイメージ プロファイル。通常、このイメージには `esxi-version-xxxxx-standard` という名前が付けられます。
- `xxxxx-no-tools` : VMware Tools バイナリを含まないイメージ プロファイル。通常、このイメージ プロファイルは小さく、メモリ オーバーヘッドも少ないため、PXE ブート環境で迅速に起動します。通常、このイメージには `esxi-version-xxxxx-no-tools` という名前が付けられます。

いずれかのイメージ プロファイルを使用して ESXi をデプロイできます。

- ネットワーク ブート時間が長くて問題がなく、使用している環境に十分なメモリとストレージ オーバーヘッドがある場合は、VMware Tools を含むイメージを使用します。
- 標準イメージを使用するとネットワーク ブートに時間がかかりすぎる場合や、ホストのストレージ容量を節約する必要がある場合は、VMware Tools を含まないイメージ プロファイルを使用し、共有ストレージに VMware Tools のバイナリを保存することができます。 [VMware Tools なしのイメージ プロファイルを使用した ESXi ホストのプロビジョニング](#) を参照してください。

vSphere Auto Deploy の負荷管理のベスト プラクティス

多数のホストを同時に起動すると、vSphere Auto Deploy サーバに大きな負荷がかかります。vSphere Auto Deploy は根本的には Web サーバであるため、Web サーバの既存のスケーリング技術を負荷の分散に役立てることができます。たとえば、キャッシュとして作用する 1 つ以上のリバース プロキシ サーバを vSphere Auto Deploy と連携させることができます。リバース プロキシは、ESXi の起動イメージの大半を占める静的ファイル进行处理します。静的な内容をキャッシュし、すべての要求を vSphere Auto Deploy サーバに渡すようにリバース プロキシを構成します。詳細については、ビデオ「[vSphere Auto Deploy のスケーラビリティを強化するリバース Web プロキシ サーバの使用](#)」を参照してください。



([vSphere Auto Deploy のスケーラビリティを強化するリバース Web プロキシ サーバの使用](#))

複数の TFTP サーバを使用してさまざまなプロキシ サーバを指定します。リバース プロキシ サーバごとに TFTP サーバを使用します。その後、異なるホストを異なる TFTP サーバに送信するように DHCP サーバを設定します。

ホストを起動すると、DHCP サーバはそれらのホストを異なる TFTP サーバにリダイレクトします。各 TFTP サーバは、vSphere Auto Deploy サーバまたはリバース プロキシ サーバのいずれかの異なるサーバにホストをリダイレクトするため、vSphere Auto Deploy サーバの負荷が大幅に軽減されます。

大規模な停電が発生した後は、クラスタ単位でホストを起動します。複数のクラスタを同時にオンラインにすると、vSphere Auto Deploy サーバで CPU のボトルネックが発生する可能性があります。すべてのホストが遅れて起動する場合があります。リバース プロキシを設定している場合、ボトルネックは比較的軽度です。

vSphere Auto Deploy のログ機能およびトラブルシューティングのベスト プラクティス

vSphere Auto Deploy で発生する問題を解決するには、vSphere Client から vSphere Auto Deploy のログ情報を使用し、ログ情報とコア ダンプをリモート ホストに送信するように環境を設定します。

vSphere Auto Deploy ログのダウンロード

vSphere Client の [vSphere Auto Deploy] ページに移動して、vSphere Auto Deploy のログをダウンロードします。 [vSphere Auto Deploy ログのダウンロード](#) を参照してください。

Syslog の設定

リモートの Syslog サーバを設定します。Syslog サーバの構成の詳細については、『vCenter Server およびホスト管理』ドキュメントを参照してください。リモートの Syslog サーバを使用するために起動する最初のホストを構成し、そのホストのホスト プロファイルをその他のすべてのデプロイ先ホストに適用します。オプションで、vSphere Syslog Collector をインストールして使用します。これは、vCenter Server のサポート ツ

ールで、システム ログの統合アーキテクチャを提供し、ネットワーク ログを有効にし、複数のホストからのログを組み合わせることができます。

ESXi Dump Collector の設定

vSphere Auto Deploy でプロビジョニングされるホストには、コア ダンプを保存するためのローカル ディスクがありません。ESXi Dump Collector をインストールし、すべてのコア ダンプが ESXi Dump Collector に向けられるように最初のホストを設定して、そのホストからのホスト プロファイルをその他すべてのホストに適用します。ESXCLI を使用した、ESXi Dump Collector の構成 を参照してください。

本番環境での vSphere Auto Deploy の使用

事前検証 (POC) 設定から本番環境に移行する場合は、環境の回復力を高めるように注意します。

- vSphere Auto Deploy サーバを保護します。vSphere Auto Deploy および vSphere HA のベスト プラクティスを参照してください。
- DHCP サーバと TFTP サーバを含む、環境内のその他すべてのサーバを保護します。
- vSphere Auto Deploy のセキュリティの考慮事項で説明されているセキュリティ ガイドラインを含む、VMware のセキュリティ ガイドラインに従います。

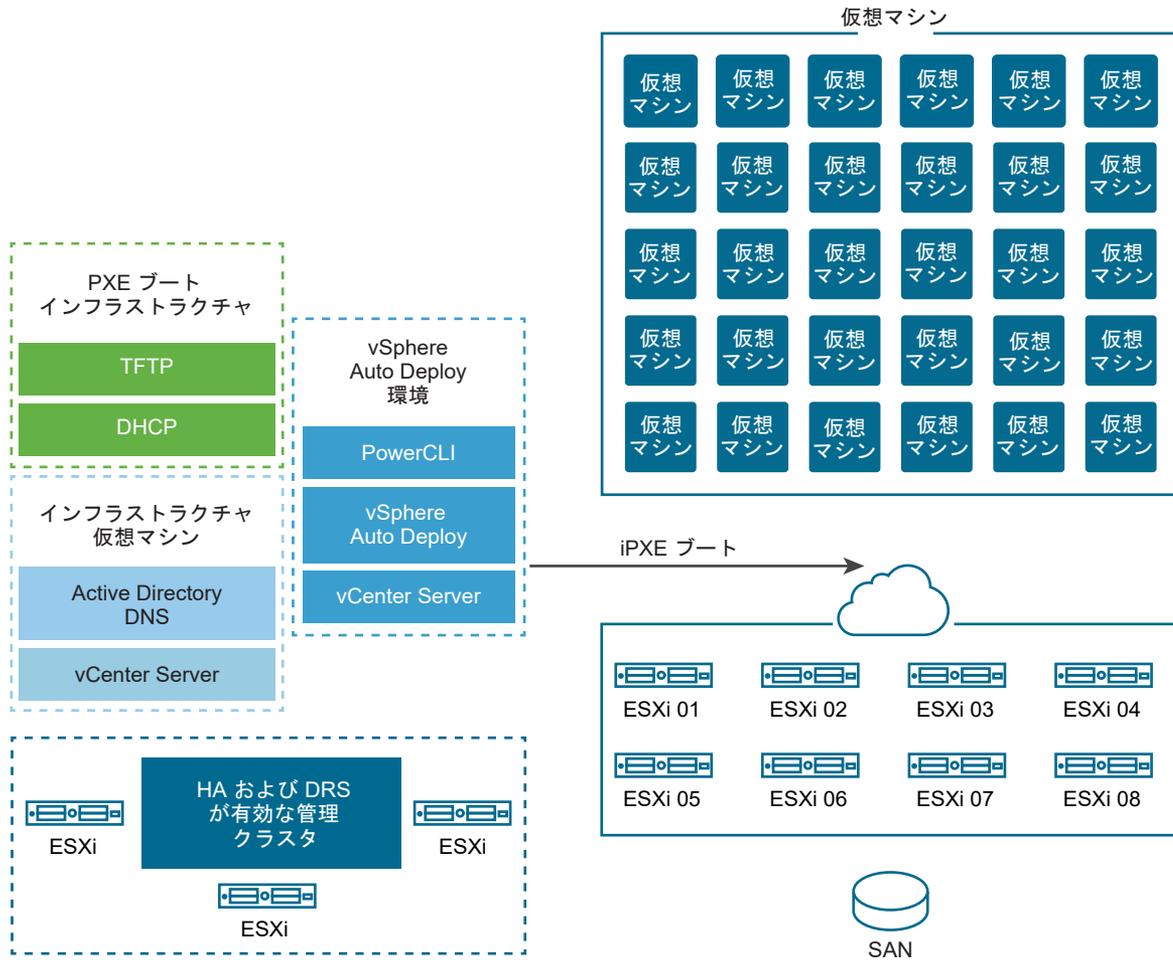
高可用性 vSphere Auto Deploy インフラストラクチャのセットアップ

高可用性 vSphere Auto Deploy インフラストラクチャは、データ損失を防ぎ、ステートレス キャッシュで vSphere Auto Deploy を使用するための前提条件です。



(高可用性 vSphere Auto Deploy インフラストラクチャ)

図 4-10. 高可用性 vSphere Auto Deploy インフラストラクチャ



前提条件

本番状況では多くの場合、高可用性 vSphere Auto Deploy インフラストラクチャでデータの損失を防ぐ必要があります。また、このようなインフラストラクチャは、ステートレス キャッシュで vSphere Auto Deploy を使用するうえでの必須事項となります。

管理クラスタでは、3つのホストに ESXi をインストールします。vSphere Auto Deploy を使用して管理クラスタのホストをプロビジョニングしないでください。

高可用性 vSphere Auto Deploy インフラストラクチャの実装の詳細については、ビデオ「高可用性 vSphere Auto Deploy インフラストラクチャ」をご覧ください。

手順

- 1 管理クラスタで vSphere HA および vSphere DRS を有効にします。

2 管理クラスタで次の仮想マシンを設定します。

インフラストラクチャ コンポーネント	説明
PXE ブート インフラストラクチャ	TFTP サーバおよび DHCP サーバ。
インフラストラクチャ仮想マシン	Active Directory、DNS、vCenter Server。
vSphere Auto Deploy 環境	PowerCLI、vSphere Auto Deploy サーバー、vCenter Server。本番システムの 1 台の仮想マシンまたは 3 台の別個の仮想マシンでこの環境を設定します。

インフラストラクチャ仮想マシンの vCenter Server は、vSphere Auto Deploy 環境の vCenter Server とは異なります。

3 vSphere Auto Deploy を設定して、必要に応じてほかのホストをプロビジョニングします。

管理クラスタにあるコンポーネントは vSphere HA により保護されているため、高可用性がサポートされません。

vSphere Auto Deploy のセキュリティの考慮事項

vSphere Auto Deploy を使用する場合は、使用環境を保護するために、ネットワーク セキュリティ、起動イメージ セキュリティ、およびパスワードの漏洩の可能性に注意してください。

ネットワーク セキュリティ

PXE ベースのほかのデプロイの場合と同様に、ネットワークをセキュリティ保護します。vSphere Auto Deploy は SSL 経由でデータを転送することで、不正な干渉やアクセスを防ぎます。しかし、PXE ブートの間は、クライアントや Auto Deploy サーバの整合性は確認されません。

Auto Deploy が使用されているネットワークを完全に隔離すると、Auto Deploy のセキュリティ リスクを大幅に低減することができます。

起動イメージおよびホスト プロファイルのセキュリティ

vSphere Auto Deploy サーバがマシンにダウンロードする起動イメージには、次のコンポーネントが含まれる場合があります。

- イメージ プロファイルから構成される VIB パッケージは、起動イメージに必ず含まれます。
- ホスト プロファイルまたはホストのカスタマイズを使用してホストをプロビジョニングするように Auto Deploy ルールが設定されている場合は、ホスト プロファイルとホストのカスタマイズが起動イメージに含まれます。
 - ホスト プロファイルおよびホストのカスタマイズに含まれる、管理者 (root) パスワードおよびユーザー パスワードは、SHA-512 でハッシュ化されます。
 - プロファイルに関連するその他すべてのパスワードは、暗号化されていません。ホスト プロファイルを使用して Active Directory を設定する場合は、パスワードは保護されません。

Active Directory パスワードの漏洩を防ぐために、vSphere Authentication Proxy を使用します。ホスト プロファイルを使用して Active Directory を設定すると、パスワードは保護されません。

- ホストの SSL のパブリック キーおよびプライベート キーと証明書が、起動イメージに含まれます。

デバイス エイリアス設定

デバイス エイリアスはデバイス名とも呼ばれる、I/O サブシステムの I/O アダプタに関連付けられた短い名前です。

たとえば、ネットワーク アップリンクには、vmnic0、vmnic1 などのエイリアスが設定されています。ストレージ サブシステム内の SCSI アダプタ オブジェクト、およびグラフィック デバイス オブジェクトにもエイリアスが設定されています。I/O サブシステム内の複数の I/O アダプタを、1つのハードウェア デバイスで表示することができます。I/O アダプタは、基盤となる物理デバイスとは異なるタイプにすることができます。たとえば、FCoE デバイスは、NIC ハードウェアを使用するストレージ I/O アダプタです。ソフトウェア iSCSI は、IP レイヤーのネットワーク スタックを使用するストレージ アダプタです。したがって、ESXi ネイティブ ドライバ モデルのエイリアスは正式には PCI NIC や PCI HBA などの物理デバイスを表すものではなく、I/O アダプタのみを表します。

デバイス エイリアスの割り当て

ステートレス ESXi デプロイ モデルとは、ESXi ホストがハード ディスクにインストールされておらず、通常は PXE を使用して起動されるモデルのことです。ステートフル ESXi デプロイ モデルとは、ESXi ホストがローカル ハード ディスクにインストールされているモデルのことです。デバイス エイリアスの割り当ては、ステートレス ESXi の起動中、またはステートフル ESXi の新規インストール中に実行されます。ESXi ホストは、基盤となるハードウェアの列挙順に基づいて、I/O アダプタにエイリアスを割り当てます。ESXi ホストは、最初にエイリアスをオンボード デバイスに割り当てた後、スロットの順序に基づいてアドイン カードに割り当てます。ESXi ホストは、存在しないデバイスや、サポート対象のドライバがインストールされていないデバイスにはエイリアスを割り当てることができません。

マザーボードに組み込まれている NIC を使用するアップリンクは、PCI アドイン カードのアップリンクよりも値が小さい vmnicN エイリアスを受け取ります。NIC ドライバで複数のアップリンクが登録されることがあります。いずれかのアップリンクが列挙可能なハードウェア デバイスに対応していない場合は、アップリンクがシステムに登録された後に、ESXi ホストによって次に使用可能なエイリアスがアップリンクに割り当てられます。

デバイス エイリアス設定の永続性

ESXi ホストからエイリアスが割り当てられた後は、エイリアスの設定が維持されます。ESXi ホストは、ESXi バージョンのアップデートや、スロットからのデバイスの追加や削除などのハードウェアの変更に関係なく、各デバイスのエイリアスを保持を試行します。

エイリアス構成の永続性は、デプロイ モデルによって異なります。

- ステートフル システムでは、エイリアス構成がホスト上でローカルに維持されます。
- ステートレス システムでは、ユーザーがホスト プロファイルを使用してステートレス システムを管理しない場合、エイリアスの構成はホスト上でローカルに保存されません。
- ホスト プロファイルを使用して管理されるステートフル システムとステートレス システムでは、エイリアス構成がホスト プロファイルに保持されます。ステートフル ホストにホスト プロファイルを適用すると、ローカルに保持されているエイリアス構成がホスト プロファイルによってオーバーライドされます。

デバイス エイリアス構成の変更

エイリアス構成の永続性は、デバイスのバス アドレスに基づいて決まります。デバイスのバス アドレスが変更されると、保持されているエイリアス構成が適用できなくなり、デバイスに割り当てられたエイリアスが変更されることがあります。

デバイス エイリアス構成は、次の場合に変更されることがあります。

- ドライバをアップグレードすると、ドライバをアップグレードする前の I/O アダプタの表示方法と異なる方法で、I/O アダプタがシステムに対して列挙または表示されることがあります。
- スタックをアップグレードすると、マルチモジュール ドライバの設定の一部が変更されたり、複数モジュールのドライバをサポートする ESXi I/O スタックが変更されたりすることがあります。
- BIOS またはデバイス ファームウェアをアップグレードすると、ポートまたはスロットの情報が不完全になることがあります。
- デバイスのスロット位置が変更されます。

注： システムからデバイスを削除すると、そのデバイスの I/O アダプタのエイリアス構成が削除されます。後で同じデバイスをシステムに再び追加すると、そのデバイスの I/O アダプタが以前のエイリアスを受信しなくなることがあります。

ESXi クラスタのデバイス エイリアスの構成

初期のエイリアス構成は、同一システムのクラスタ全体で同じになります。ただし、同種であるとみなされるクラスタであっても、ハードウェアまたはファームウェアに若干の違いがあると、ホスト間のエイリアスの構成に違いが生じる可能性があります。

ドライバをバインドするときの処理順序が異なるために、エイリアスの構成に違いが生じる場合もあります。たとえば、1つの NIC ドライバによって、同じ PCI デバイスの 2 つのポートに対してアップリンク 1 とアップリンク 2 という 2 つのアップリンクが登録されるとします。この場合、システムは一方のポートに対してハードウェアを列挙できません。アップリンクの登録順序のタイミングが変わると、ESXi ホストがアップリンクにエイリアスを割り当てる方法に違いが生じることがあります。たとえば、アップリンク 1 には、ある ESXi ホストによってハードウェアベースのエイリアスが割り当てられ、アップリンク 2 には、別の ESXi ホストによってハードウェアベースのエイリアスが割り当てられることがあります。

同種のホスト間でエイリアスの構成を一致させるには、ホスト プロファイルを使用します。デバイス エイリアス構成ホスト プロファイルは、エイリアス構成内のデバイスを ESXi ホスト デバイスにマッピングすることによって、ESXi ホストにエイリアス構成を適用します。マッピング操作は、初期エイリアス割り当ての基準として使用されるハードウェア情報ソースに基づきます。エイリアスの割り当てに使用される情報源については、ナレッジベースの記事 [KB 2091560](#) を参照してください。

デバイスがホスト プロファイルに含まれていても、ホスト上にない場合などでは、エラーにもデバイス エイリアス構成ホスト プロファイルのフラグが設定されます。

異種クラスタでは、ホスト間にデフォルトで共通のエイリアス設定がありません。デバイス間の差異が理由で、1つのホスト プロファイルを整然と適用することはできません。

ESXi Shell コマンドを使用したデバイス エイリアス情報の表示

実行中の ESXi システムでは、ESXi Shell でコマンドを実行して I/O アダプタ エイリアスに関する情報を表示できます。

ESXi Shell コマンドを使用したデバイス エイリアス情報の表示

コマンド	説明
<code>device alias list</code>	現在のすべての I/O アダプタ エイリアスが一覧表示されます。
<code>device alias get -n<alias></code>	I/O アダプタ エイリアスのマッピング先の物理デバイスが表示されません。
<code>network nic list</code>	エイリアスと、ネットワーク デバイスに関する一般的な情報が一覧表示されます。
<code>storage core adapter list</code>	すべてのストレージ I/O アダプタが一覧表示されます。

注： ESXCLI コマンドが、サポートされているコマンドです。エイリアス情報の表示に代替ソースを使用することは推奨されません。

ホスト プロファイルを使用した ESXi ホストでのデバイス エイリアスの変更

ホストに添付されているホスト プロファイルのデバイス エイリアス セクションを編集して、ESXi ホストのデバイス エイリアスを変更できます。

BIOS またはデバイス ファームウェアのアップデートを適用した場合などに、デバイス エイリアスを変更される可能性があります。デバイス エイリアス構成の変更の詳細については、[デバイス エイリアス設定](#)を参照してください。

通常、PCI ハードウェアをベースとする I/O アダプタには、論理エイリアス エントリと PCI エイリアス エントリがあります。両方のエイリアスの値は同じにする必要があります。通常、一部の I/O アダプタには論理エイリアスのみがあります。通常、I/O アダプタがない PCI ハードウェア デバイスには、PCI エイリアス エントリのみがあります。PCI エイリアス エントリは、別のデバイスにエイリアスが必要な場合にのみ変更します。

注： 2 つの異なる I/O アダプタに同じエイリアスを設定しないでください。ただし、PCI ハードウェアをベースとし、論理エイリアス エントリと PCI エイリアス エントリを含む I/O アダプタは除きます。

手順

ホスト プロファイルのエクスポート、抽出、および編集の詳細については、『vSphere ホスト プロファイル』ドキュメントを参照してください。

- 1 ESXi ホストに添付されている現在のホスト プロファイルをエクスポートします。
- 2 ESXi ホストから新しいホスト プロファイルを抽出しますが、ホスト プロファイルは適用しません。
- 3 現在のホスト プロファイルのデバイス エイリアス セクションからすべてのエントリを削除します。
- 4 新しいホスト プロファイルのデバイス エイリアス セクションから現在のホスト プロファイルの同じセクションに、すべてのエントリを転送します。

エントリ内のエイリアスは、現在のホスト プロファイルに追加する前に変更できます。

- 5 ESXi ホストに現在のホスト プロファイルを再適用します。

ESXCLI コマンドを使用したステートフル ESXi ホストでのデバイス エイリアスの変更

ホスト プロファイルが添付されていないステートフル ESXi ホストのデバイス エイリアスを変更するには、ホスト上で ESXCLI コマンドを実行します。

デバイスのスロット位置が変更された場合などに、デバイスエイリアスが変更される可能性があります。デバイスエイリアス構成の変更の詳細については、[デバイスエイリアス設定](#)を参照してください。

注： 次の ESXCLI コマンドは ESXi バージョン 6.7 以前と互換性がない可能性があります。

手順

- 1 デバイス アドレスに対するエイリアスの現在の割り当てを一覧表示するには、ESXi Shell で `localcli --plugin-dir /usr/lib/vmware/esxcli/int/ deviceInternal alias list` を実行します。
 PCI ハードウェアをベースとする I/O アダプタの場合、通常は、論理エイリアス エントリと PCI エイリアス エントリが表示されます。
 通常、一部の I/O アダプタには論理エイリアス エントリのみが表示されます。
 PCI ハードウェア デバイスに I/O アダプタがない場合は、通常、PCI エイリアス エントリのみが表示されません。
- 2 エイリアスを変更するには、ALIAS、PCI_ADDRESS、および LOGICAL_ADDRESS プレースホルダの値を実際の値に置き換えて、いずれかのサブステップを実行します。
 - a I/O アダプタに論理エイリアスと PCI エイリアスがある場合は、`localcli --plugin-dir /usr/lib/vmware/esxcli/int/ deviceInternal alias store --bus-type logical --alias ALIAS --bus-address LOGICAL_ADDRESS` および `localcli --plugin-dir /usr/lib/vmware/esxcli/int/ deviceInternal alias store --bus-type pci --alias ALIAS --bus-address PCI_ADDRESS` を実行します。
 論理エイリアスと PCI エイリアスの値は同じにする必要があります。
 - b I/O アダプタに論理エイリアス エントリのみがある場合は、`localcli --plugin-dir /usr/lib/vmware/esxcli/int/ deviceInternal alias store --bus-type logical --alias ALIAS --bus-address LOGICAL_ADDRESS` を実行します。
 - c PCI ハードウェア デバイスに PCI エイリアス エントリのみがある場合は、`localcli --plugin-dir /usr/lib/vmware/esxcli/int/ deviceInternal alias store --bus-type pci --alias ALIAS --bus-address PCI_ADDRESS` を実行します。
 PCI エイリアスは、別のデバイスにエイリアスが必要な場合にのみ変更します。

注： 2 つの異なる I/O アダプタに同じエイリアスを設定しないでください。ただし、PCI ハードウェアをベースとし、論理エイリアス エントリと PCI エイリアス エントリを含む I/O アダプタは除きます。

- 3 システムを再起動します。

vSphere Auto Deploy の vSphere PowerCLI シナリオ

PowerCLI を使用して、vSphere Auto Deploy 作業環境をセットアップおよび構成するシナリオを示します。

このシナリオでは、4 台のホストを含む vSphere Auto Deploy の作業環境をセットアップおよび構成します。ルールを作成し、ホストうちの 2 台はイメージ プロファイルを使用してプロビジョニングし、他の 2 台は同じイメージ プロファイルとユーザーの入力を要求するように設定されているホスト プロファイルを使用してプロビジョニングします。このシナリオでは、本番環境向けの基本的な情報について説明します。タスクの説明では、物理ホストと環境内のその他の要素間に VLAN タギングが行われていない、フラットなネットワークの使用を想定しています。

このシナリオでタスクを実行するには、次の予備知識と権限が必要です。

- vSphere (vCenter Server および ESXi) の使用経験。
- Microsoft PowerShell および PowerCLI の基礎知識。
- Windows システムおよび vCenter Server システムに対する管理者権限。

このシナリオで説明する順番で、タスクを実行します。一部の手順は異なる順序で実行できますが、ここで使用される順序はいくつかのコンポーネントの繰り返し操作を制限します。vSphere Auto Deploy を構成するためのインストール前チェックリストとその他の前提条件の詳細については、[vSphere Auto Deploy のインストールと構成](#)を参照してください。

図 4-11. vSphere Auto Deploy の設定とホスト プロビジョニングのワークフロー



vSphere Auto Deploy は iPXE インフラストラクチャを利用します。レガシーの BIOS ファームウェアを使用する PXE ブートは IPv4 でのみ実行できます。vSphere Auto Deploy を使用してプロビジョニングするホストでレガシーの BIOS が使用されている場合は、vSphere Auto Deploy サーバに IPv4 アドレスが必要です。UEFI ファームウェアを使用する PXE ブートは IPv4 または IPv6 のいずれかで実行できます。

手順

1 TFTP サーバのインストール

vSphere Auto Deploy インフラストラクチャを設定するには、環境に TFTP サーバを設置する必要があります。vSphere Auto Deploy は、プロビジョニング対象のホストに起動イメージを送るために、TFTP サーバを利用します。

2 PowerCLI のインストール

PowerCLI コマンドレットで作成するルールで vSphere Auto Deploy を管理するには、PowerCLI をインストールする必要があります。

3 vSphere Auto Deploy のターゲット ホストの準備

vSphere Auto Deploy によるプロビジョニングでターゲット ホストを準備する場合、4 台のホストの BIOS 設定を行い、プライマリ ネットワーク デバイスの MAC アドレスを確認しておく必要があります。

4 vSphere Auto Deploy によるプロビジョニングのための DHCP サーバの準備

vSphere Auto Deploy のターゲット ホストを準備するには、このシナリオに沿って DHCP サーバをセットアップし、各ターゲット ホストに iPXE バイナリを提供できるようにします。

5 vSphere Client での vSphere Auto Deploy および TFTP 環境の構成

DHCP サーバを準備したら、vSphere Auto Deploy vCenter Server サービスを開始して TFTP サーバを構成する必要があります。TFTP ZIP ファイルを vSphere Auto Deploy サーバからダウンロードする必要があります。カスタマイズされた FTP サーバ経由で、vSphere Auto Deploy によって起動イメージが提供されます。

6 ESXi ソフトウェア デポの準備とルールの作成

vSphere Auto Deploy インフラストラクチャを構成したら、ESXi ソフトウェア デポを追加して、イメージ プロファイルを指定し、ルールを作成してそれを有効なルール セットに追加する必要があります。

7 vSphere Auto Deploy を使用した、最初のホストのプロビジョニング

ルールを作成して有効なルール セットに追加したら、最初のホストをプロビジョニングし、その vCenter Server の場所を確認して、セットアップしたイメージのプロビジョニングの検証を行います。

8 リファレンス ホストからのホスト プロファイルの抽出と設定

最初のホストをプロビジョニングしたら、同じ設定を他のターゲット ホストに適用するためのホスト プロファイルを抽出して設定することができます。固定 IP アドレスなどのホストごとに異なる設定は、ホストをカスタマイズすることで管理します。

9 特定の IP アドレス範囲でホストをプロビジョニングするルールの作成

リファレンス ホストからホスト プロファイルを作成したら、以前に確認したイメージ プロファイルおよび抽出したホスト プロファイルを特定の IP アドレス範囲にあるターゲット ホストに適用するルールを作成することができます。

10 ホストのプロビジョニングとホストのカスタマイズ設定

イメージ プロファイルおよびホスト プロファイルを使用してホストをプロビジョニングするルールを設定したら、特定のターゲット ホストをプロビジョニングできます。ホスト プロファイルの項目のいずれかが、ユーザー入力を求めるように設定されている場合、ホストはメンテナンス モードになります。情報の入力を求めるには、ホスト プロファイルを適用するか、ホストのコンプライアンスを確認します。システムにより、ホストにホストのカスタマイズが関連付けられます。

TFTP サーバのインストール

vSphere Auto Deploy インフラストラクチャを設定するには、環境に TFTP サーバを設置する必要があります。vSphere Auto Deploy は、プロビジョニング対象のホストに起動イメージを送るために、TFTP サーバを利用します。

このタスクでは、TFTP サーバのインストールのみを実施します。後の手順で構成ファイルをサーバにダウンロードします。vSphere Client での vSphere Auto Deploy および TFTP 環境の構成を参照してください。

手順

- 1 vCenter Server へのネットワーク アクセスが可能な場所に優先 TFTP サーバをダウンロードして、インストールします。
- 2 TFTP ルート ディレクトリ (D:\TFTP_Root\ など) を構成します。

次のステップ

PowerCLI をインストールし、PowerCLI コマンドレットで vSphere Auto Deploy を管理します。

PowerCLI のインストール

PowerCLI コマンドレットで作成するルールで vSphere Auto Deploy を管理するには、PowerCLI をインストールする必要があります。

手順

- ◆ 『VMware PowerCLI User's Guide』を使用して PowerShell の基本、PowerCLI の概念、PowerCLI のインストールおよび構成方法を確認します。

次のステップ

ターゲット ホストの設定を行い、vSphere Auto Deploy によるプロビジョニングを準備します。

vSphere Auto Deploy のターゲット ホストの準備

vSphere Auto Deploy によるプロビジョニングでターゲット ホストを準備する場合、4 台のホストの BIOS 設定を行い、プライマリ ネットワーク デバイスの MAC アドレスを確認しておく必要があります。

前提条件

vSphere Auto Deploy でプロビジョニングするホストは、ESXi の要件を満たす必要があります。

ESXi のハードウェア要件を参照してください。

手順

- 1 4 台の物理ホストの BIOS 設定をそれぞれ変更し、ホストが強制的にプライマリ ネットワーク デバイスから起動するようにします。
- 2 プライマリ ネットワーク デバイスの MAC アドレスを再度確認してください。

次のステップ

各ターゲット ホストに iPXE バイナリを提供するように DHCP サーバを設定します。

vSphere Auto Deploy によるプロビジョニングのための DHCP サーバの準備

vSphere Auto Deploy のターゲット ホストを準備するには、このシナリオに沿って DHCP サーバをセットアップし、各ターゲット ホストに iPXE バイナリを提供できるようにします。

このシナリオの環境では、DNS と DHCP を設定し、Active Directory を使用します。DHCP サーバは、vSphere でサポートされるバージョンの Windows Server に含まれています。

手順

- 1 管理者権限を使用して DHCP サーバにログインします
- 2 IP アドレス範囲用に DHCP スコープを作成します。
 - a [スタート > 設定 > コントロール パネル > 管理ツール] の順にクリックして、[DHCP] をクリックします。
 - b [DHCP] - [*hostname*] - [IPv4] の順に移動します。
 - c [IPv4] を右クリックし、[新しいスコープ] を選択します。
 - d ようこそ画面で、[次へ] をクリックし、スコープの名前と説明を指定します。
 - e IP アドレスの範囲を指定して、[次へ] をクリックします。
 - f DHCP オプションの構成ページが表示されるまで [次へ] をクリックし、[後でオプションを構成する] を選択します。
- 3 各ターゲットの ESXi ホストの DHCP 予約を作成します。
 - a DHCP のウィンドウで、[DHCP] - [*hostname*] - [IPv4] - [Autodeploy スコープ] - [予約] の順に移動します。
 - b [予約] を右クリックして、[新しい予約] を選択します。

- c 新しい予約ウィンドウで、ホストのいずれかの名前、IP アドレス、および MAC アドレスを指定します。MAC アドレスにはコロン (:) を含めないでください。



- d その他のホストそれぞれにつき、このプロセスを繰り返します。

4 ホストを TFTP サーバへポイントするように、DHCP サーバを設定します。

- a DHCP のウィンドウで、[DHCP] - [*hostname*] - [IPv4] - [Autodeploy スコープ] - [スコープ オプション] の順に移動します。
- b [スコープ オプション] を右クリックして、[オプションの構成] を選択します。
- c スコープ オプションのウィンドウで [全般] タブをクリックします。
- d [066 ブート サーバ ホスト名] をクリックし、インストールした TFTP サーバのアドレスを、[使用可能なオプション] の下に表示される文字列の値フィールドに入力します。



- e [067 ブートファイル名] をクリックして、[undionly.kpxe.vmw-hardwired] と入力します。ESXi ホストの起動に、iPXE バイナリ undionly.kpxe.vmw-hardwired が使用されます。
- f [適用] > [OK] の順にクリックしてウィンドウを閉じます。

5 DHCP ウィンドウで [DHCP] - [*hostname*] - [IPv4] - [スコープ] - [有効化] を右クリックし、[有効化] をクリックします。

- 6 DHCP および DNS を Active Directory に使用している場合は、DHCP サーバからログアウトしません。使用していない場合はログアウトします。

次のステップ

vSphere Auto Deploy の vCenter Server サービスを開始し、TFTP サーバを構成します。

vSphere Client での vSphere Auto Deploy および TFTP 環境の構成

DHCP サーバを準備したら、vSphere Auto Deploy vCenter Server サービスを開始して TFTP サーバを構成する必要があります。TFTP ZIP ファイルを vSphere Auto Deploy サーバからダウンロードする必要があります。カスタマイズされた FTP サーバ経由で、vSphere Auto Deploy によって起動イメージが提供されます。

手順

- 1 vSphere Auto Deploy サーバを管理する vCenter Server システムに接続するには vSphere Client を使用します。
- 2 vSphere Auto Deploy サービスを開始します。
 - a [ホーム] - [Auto Deploy] の順に移動します。
 - b [Auto Deploy] ページで、上部のドロップダウン メニューから vCenter Server を選択します。
 - c [Auto Deploy および Image Builder を有効にする] をクリックしてサービスをアクティブ化します。
[Image Builder] サービスがすでにアクティブな場合は、[構成] タブを選択し、[Auto Deploy サービスを有効化] をクリックします。
- 3 Auto Deploy インベントリで [設定] タブをクリックします。
- 4 [TFTP Zip ファイルのダウンロード] をクリックします。
- 5 deploy-tftp.zip ファイルを、TFTP サーバのインストール時に作成した TFTP_Root ディレクトリに保存してファイルを解凍します。

次のステップ

インベントリにソフトウェア デポを追加し、デポのイメージ プロファイルを使用してホストのプロビジョニングのためのルールを作成します。

ESXi ソフトウェア デポの準備とルールの作成

vSphere Auto Deploy インフラストラクチャを構成したら、ESXi ソフトウェア デポを追加して、イメージ プロファイルを指定し、ルールを作成してそれを有効なルール セットに追加する必要があります。

vSphere Auto Deploy は、ESXi インストール プロセスで使用する VIB セットを定義するイメージ プロファイルをホストにプロビジョニングします。イメージ プロファイルは、ソフトウェア デポに格納されます。ホストのプロビジョニングを開始する前に、適切なイメージ プロファイルが利用できることを確認します。PowerCLI セッションにソフトウェア デポを追加する場合は、そのセッションでのみ利用可能です。ソフトウェア デポを複数のセッションで継続して使用することはできません。

このタスクの手順には、PowerCLI コマンドレットの実行が含まれます。PowerCLI セッションで実行できる vSphere Auto Deploy コマンドレットの詳細については、[vSphere Auto Deploy PowerCLI コマンドレットの概要](#)を参照してください。

前提条件

PowerCLI を実行するシステムから、プロビジョニングする ESXi ホストにアクセスできることを確認します。

手順

- 1 Windows システムのコンソールに管理者権限で直接ログインするか、RDP を使用してログインします。

この作業では、vCenter Server システムが実行中のシステム上に、PowerCLI がインストールされていることを想定しています。

- 2 PowerCLI セッションで `Connect-VIServer` コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_address
```

このコマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の問題が発生していないことを確認します。開発環境では、この警告は無視できます。

- 3 vCenter Server 認証情報を入力します。
- 4 `Add-EsxSoftwareDepot` を実行して PowerCLI セッションにオンライン デポを追加します。

```
Add-EsxSoftwareDepot https://hostupdate.vmware.com/software/VUM/PRODUCTION/main/vmw-depot-index.xml
```

新しく PowerCLI セッションを開始するたびに、ソフトウェア デポを追加する必要があります。

- 5 `Get-EsxImageProfile` コマンドレットでデポの内容を確認して、ソフトウェア デポが正しく追加されたかどうか検証します。

コマンドレットはデポに含まれるすべてのイメージ プロファイルの情報を返します。

- 6 `New-DeployRule` コマンドレットを実行して新しいルールを作成します。

```
New-DeployRule -Name "InitialBootRule" -Item ESXi-6.0.0-2494585-standard -AllHosts
```

コマンドレットは、指定されたイメージ プロファイルをインベントリのすべてのホストに割り当てるルールを作成します。

- 7 有効なルール セットに新しいルールを追加して、そのルールを vSphere Auto Deploy サーバで使用できるようにします。

```
Add-DeployRule -DeployRule "InitialBootRule"
```

次のステップ

最初のホストに vSphere Auto Deploy をプロビジョニングし、そのイメージ プロビジョニングを検証します。

vSphere Auto Deploy を使用した、最初のホストのプロビジョニング

ルールを作成して有効なルール セットに追加したら、最初のホストをプロビジョニングし、その vCenter Server の場所を確認して、セットアップしたイメージのプロビジョニングの検証を行います。

手順

- 1 最初の ESXi ターゲット ホストとして使用する物理ホストへのコンソール セッションを開き、ホストを起動して、正常に iPXE ブートされたことを示すメッセージを確認します。

起動プロセスで、DHCP がホストに IP アドレスを割り当てます。この IP アドレスは、事前に DNS サーバで指定した名前です。ホストは vSphere Auto Deploy サーバに接続し、iPXE の tramp ファイルに示されている HTTP URL から ESXi のバイナリをダウンロードします。この tramp ファイルは、事前に TFTP_Root ディレクトリにダウンロードしたものです。vSphere Auto Deploy の各インスタンスは、TFTP サーバ用に一連のカスタム ファイルを作成します。

- 2 vSphere Auto Deploy サーバを管理する vCenter Server システムに接続するには vSphere Client を使用します。
- 3 vSphere Client のホーム ページで、[ホストおよびクラスタ] をクリックします。
- 4 新しくプロビジョニングされたホストが、データセンター レベルの vCenter Server インベントリ内にあることを確認します。

デフォルトでは、vSphere Auto Deploy は、起動プロセスの完了時にホストをデータセンターに追加します。

次のステップ

ホストからホスト プロファイルを抽出し、ユーザー入力を要求するように設定します。

リファレンス ホストからのホスト プロファイルの抽出と設定

最初のホストをプロビジョニングしたら、同じ設定を他のターゲット ホストに適用するためのホスト プロファイルを抽出して設定することができます。固定 IP アドレスなどのホストごとに異なる設定は、ホストをカスタマイズすることで管理します。

vSphere Auto Deploy を使用することで、各ホストに同じホスト プロファイルをプロビジョニングできます。さらに vSphere Auto Deploy では、ホストのカスタマイズ機能を使用して、それぞれのホストに異なる情報を指定できます。たとえば、vMotion 用またはストレージ用に VMkernel ポートを設定する場合、ホストのカスタマイズ機能によって、ポートに固定 IP アドレスを指定できます。

手順

- 1 vSphere Auto Deploy サーバを管理する vCenter Server システムに接続するには vSphere Client を使用します。
- 2 [ポリシーおよびプロファイル] をクリックし、[ホスト プロファイル] を選択します。
- 3 [ホスト プロファイルの抽出] をクリックします。
- 4 ウィザードの [ホストの選択] 画面で、以前に構成したリファレンス ホストを選択し、[次へ] をクリックします。
- 5 ウィザードの [名前と説明] ページで、新しいプロファイルの名前と説明を入力し、[終了] をクリックします。
- 6 編集するホスト プロファイルを選択し、[設定] タブをクリックします。
- 7 [プロファイルの編集] をクリックします。
- 8 [セキュリティおよびサービス] - [セキュリティの設定] - [セキュリティ] - [ユーザー設定] - [root] の順に選択します。

- 9 [パスワード] ドロップダウン メニューから、[ユーザー入力のパスワード設定] を選択します。
- 10 [保存] をクリックしてホスト プロファイルを設定します。

次のステップ

ホスト プロファイルを他の ESXi ホストに適用する vSphere Auto Deploy ルールを作成します。

特定の IP アドレス範囲でホストをプロビジョニングするルールの作成

リファレンス ホストからホスト プロファイルを作成したら、以前に確認したイメージ プロファイルおよび抽出したホスト プロファイルを特定の IP アドレス範囲にあるターゲット ホストに適用するルールを作成することができます。

手順

- 1 Windows システムのコンソールに、管理者権限で直接ログインするか、RDP を使用してログインします。
- 2 PowerCLI セッションで `Connect-VIServer` コマンドレットを実行し、vSphere Auto Deploy が登録される vCenter Server システムに接続します。

```
Connect-VIServer ipv4_address
```

この コマンドレットはサーバ証明書の警告を返す場合があります。本番環境では、サーバ証明書の問題が発生していないことを確認します。開発環境では、この警告は無視できます。

- 3 `Add-EsxSoftwareDepot` を実行して PowerCLI セッションにオンライン デポを追加します。

```
Add-EsxSoftwareDepot https://hostupdate.vmware.com/software/VUM/PRODUCTION/main/vmw-depot-index.xml
```

新しく PowerCLI セッションを開始するたびに、ソフトウェア デポを追加する必要があります。

- 4 (必須) `Get-DeployRuleset` コマンドレットを実行して、有効なルール セット内のルールを表示します。
- 5 事前選択したイメージ プロファイルとリファレンス ホストから作成したホスト プロファイルを使用して、特定の IP アドレス範囲内の一連のホストをプロビジョニングするよう vSphere Auto Deploy に指示するルールを作成します。

```
New-DeployRule -name "Production01Rule" -item "image_profile",ESXiGold -Pattern "ipv4=IP_range"
```

- 6 有効なルール セットに新しいルールを追加します。

```
Add-DeployRule -DeployRule "Production01Rule"
```

- 7 `Get-DeployRuleset` コマンドを実行して、有効なルール セットを確認します。

次の例のような情報が PowerCLI に表示されます。

```
Name:                Production01Rule
PatternList:         {ipv4=address_range}
ItemList:            {ESXi-version-XXXXXX-standard, Compute01, ESXiGold}
```

次のステップ

ホストをプロビジョニングし、ホストのカスタマイズを行います。

ホストのプロビジョニングとホストのカスタマイズ設定

イメージ プロファイルおよびホスト プロファイルを使用してホストをプロビジョニングするルールを設定したら、特定のターゲット ホストをプロビジョニングできます。ホスト プロファイルの項目のいずれかが、ユーザー入力を求めるように設定されている場合、ホストはメンテナンス モードになります。情報の入力を求めるには、ホスト プロファイルを適用するか、ホストのコンプライアンスを確認します。システムにより、ホストにホストのカスタマイズが関連付けられます。

手順

- 1 プロビジョニングする残りのホストを起動します。

vSphere Auto Deploy はホストを起動し、ホスト プロファイルを適用して、ホストを vCenter Server インベントリに追加します。リファレンス ホストのホスト プロファイル、各ホストでユーザー入力を求めるように設定しているため、ホストはメンテナンス モードのままになります。

- 2 vSphere Auto Deploy サーバを管理する vCenter Server システムに接続するには vSphere Client を使用します。
- 3 [ポリシーおよびプロファイル] をクリックし、[ホスト プロファイル] を選択します。
- 4 Auto Deploy に対して新規作成されたホスト プロファイルを右クリックし、[ホストのカスタマイズの編集] をクリックします。
- 5 ホストを選択、必要なホストのカスタマイズ情報を入力し、[完了] をクリックします。
あるいは、[ホストのカスタマイズのインポート] ファイルを使用することもできます。
- 6 各ホストにホスト プロファイルを適用し、ホストをメンテナンス モードから復帰させます。または、各ホストを再起動します。

再起動が完了すると、すべてのホストが、指定したイメージを使用して実行され、リファレンス ホストのプロファイル設定を使用します。クラスタに、すべてのホストがコンプライアンスに完全に準拠していることが示されます。

結果

これで、リファレンス ホストのプロファイルの共有情報と、ホストのカスタマイズ メカニズムによるホスト固有の情報を使用して、すべてのホストが構成されました。次回ホストを起動する時に、ホストはメンテナンス モードから復帰し、ホスト固有の情報を含む完全なホスト プロファイル情報を取得して、完全に設定された状態で起動します。

vSphere Auto Deploy のトラブルシューティング

vSphere Auto Deploy のトラブルシューティングに関するトピックでは、vSphere Auto Deploy を使用したホストのプロビジョニングが期待どおりに動作しない場合について説明します。

vSphere Auto Deploy ルールの完了に時間がかかる

Auto Deploy ルールの実行後、完了までに時間がかかり、タスクの進行状況が表示されません。

問題

Auto Deploy では、クラスタが使用するイメージの新しいバージョンが使用可能かどうか、またはファイルのキャッシュの有効期間がいつ切れるかを自動的に検出し、ソフトウェア デポの最新バージョンからファイルを再ダウンロードしてキャッシュを更新します。その結果、Auto Deploy ルールの実装に遅延が発生することがあります。

原因

Auto Deploy によってキャッシュされたファイルのキャッシュの有効期間が切れると、Auto Deploy はソフトウェア デポからキャッシュを自動的に更新します。通常、デフォルトのソフトウェア デポは 300 MB を超えているため、ネットワークによってはダウンロードに時間がかかることがあります。

また、単一のイメージを使用して管理するクラスタを、クラスタ レベルの構成を使用して管理するクラスタに変換するときに、Auto Deploy ルールのデプロイに遅延が発生する場合があります。Auto Deploy が構成ファイルをキャッシュしている間にホストを起動しようとする、Auto Deploy がキャッシュを作成する必要があるために遅延が発生することがあります。

また、vSphere インフラストラクチャの一般的な問題により、Auto Deploy ルールが適切なタイミングで実行されなくなる場合があります。

解決方法

- ◆ 接続の問題が破損かに関係なく、Auto Deploy はアクティブなセッションを維持し、成功するまでホストを起動し続けます。

起動時の vSphere Auto Deploy TFTP タイムアウト エラー

vSphere Auto Deploy を使用してプロビジョニングされているホストが起動する際に、TFTP タイムアウト エラー メッセージが表示されます。メッセージのテキストは BIOS によって異なります。

問題

vSphere Auto Deploy を使用してプロビジョニングされているホストが起動する際に、TFTP タイムアウト エラー メッセージが表示されます。メッセージのテキストは BIOS によって異なります。

原因

TFTP サーバがダウンしているまたはアクセス不可能です。

解決方法

- TFTP サービスが実行されていて、起動を試みているホストによってアクセスできることを確認してください。
- 表示されたエラーの詳細を診断ログで確認するには、TFTP サービスのドキュメントを参照してください。

vSphere Auto Deploy ホストが間違っただ構成で起動する

ルールで指定されたものとは異なる ESXi イメージ、ホスト プロファイル、フォルダの場所でホストが起動していません。

問題

ルールが指定するイメージ プロファイルまたは構成とは異なる ESXi イメージ プロファイルまたは構成でホストが起動しています。たとえば、異なるイメージ プロファイルを割り当てるようルールを変更したが、ホストは変更前のイメージ プロファイルを使用しています。

原因

ホストが vCenter Server システムに追加された後、起動構成は vCenter Server システムによって決定されます。vCenter Server システムは、イメージ プロファイル、ホスト プロファイル、またはフォルダの場所をホストに関連付けます。

解決方法

- ◆ Test-DeployRuleSetCompliance および Repair-DeployRuleSetCompliance vSphere PowerCLI コマンドレットを使用して、ルールを再評価し、正しいイメージ プロファイル、ホスト プロファイル、またはフォルダの場所をホストに関連付けます。

ホストが vSphere Auto Deploy サーバにリダイレクトされない

起動中に、vSphere Auto Deploy でのプロビジョニングの対象となるホストが、iPXE をロードします。ホストが vSphere Auto Deploy サーバにリダイレクトされません。

問題

起動中に、vSphere Auto Deploy でのプロビジョニングの対象となるホストが、iPXE をロードします。ホストが vSphere Auto Deploy サーバにリダイレクトされません。

原因

TFTP ZIP ファイル内に含まれる tramp ファイルに、vSphere Auto Deploy サーバの誤った IP アドレスが含まれています。

解決方法

- ◆ 『vSphere のインストールとセットアップ』ドキュメントで説明されているとおりに、tramp ファイルで、vSphere Auto Deploy サーバの IP アドレスを修正します。

イメージ プロファイルを vSphere Auto Deploy ホストに割り当てる際のパッケージ警告メッセージ

vSphere Auto Deploy が準備できていないイメージ プロファイルを割り当てる vSphere PowerCLI コマンドレットを実行すると、警告メッセージが表示されます。

問題

1 台以上のホストにイメージ プロファイルを割り当てるルールを作成または変更すると、次のエラーが発生します。

```
Warning: Image Profile <name-here> contains one or more software packages that are not stateless-ready. You may experience problems when using this profile with Auto Deploy.
```

原因

イメージ プロファイル内の各 VIB には、vSphere Auto Deploy で使用されることを示す `stateless-ready` フラグがあります。vSphere Auto Deploy ルールの作成に、このフラグが `FALSE` に設定された VIB を 1 つ以上含むイメージ プロファイルを使用すると、エラーが発生します。

注： ステートレスに対応していない VIB が含まれる vSphere Auto Deploy を使用してプロビジョニングされたホストは、問題なく使用できます。ただし、ステートレス対応でない VIB が含まれるイメージ プロファイルを使用して起動する場合は、フレッシュ インストールと同様に扱われます。vSphere Auto Deploy を使用してプロビジョニングされたホストの再起動で使用できるはずの構成データが、ホストを起動するたびにすべて失われます。

解決方法

- 1 vSphere PowerCLI セッションで vSphere ESXi Image Builder コマンドレットを使用して、イメージ プロファイル内の VIB を表示します。
- 2 ステートレスに対応していない VIB を削除します。
- 3 vSphere Auto Deploy コマンドレットを再実行します。

組み込み型 USB フラッシュ ドライブを備えた vSphere Auto Deploy ホストがコアダンプをローカル ディスクに送信しない

vSphere Auto Deploy ホストが組み込み型の USB フラッシュ ドライブを備えており、エラーが発生してコアダンプが出力された場合に、コアダンプが失われます。ネットワーク化されたホスト上でコアダンプを格納するには、ESXi Dump Collector を使用するようにシステムを設定します。

問題

vSphere Auto Deploy ホストが組み込み型の USB フラッシュを備えており、そこでエラーが発生してコアダンプが出力された場合に、コアダンプがローカル ディスクに送信されません。

解決方法

- 1 任意のシステムに ESXi Dump Collector をインストールします。
ESXi Dump Collector は vCenter Server インストーラに含まれています。
- 2 ESXCLI を用いて、ホストが ESXi Dump Collector を使用するように構成します。

```
esxcli conn_options system coredump network set IP-addr,port
esxcli system coredump network set -e true
```

- 3 ESXCLI を使用して、ローカル コアダンプのパーティションを無効にします。

```
esxcli conn_options system coredump partition set -e false
```

vSphere Auto Deploy ホストが 5 分後に再起動する

vSphere Auto Deploy ホストが起動し iPXE 情報が表示されても、5 分後に再起動します。

問題

vSphere Auto Deploy でプロビジョニングされたホストが iPXE から起動し、コンソールに iPXE 情報が表示されます。しかし、5 分後に、ホストは次のメッセージをコンソールに表示し、再起動します。

```
This host is attempting to network-boot using VMware
AutoDeploy. However, there is no ESXi image associated with this host.
Details: No rules containing an Image Profile match this
host. You can create a rule with the New-DeployRule PowerCLI cmdlet
and add it to the rule set with Add-DeployRule or Set-DeployRuleSet.
The rule should have a pattern that matches one or more of the attributes
listed below.
```

また、ホストは次の詳細を表示することもあります。

```
Details: This host has been added to VC, but no Image Profile
is associated with it. You can use Apply-ESXImageProfile in the
PowerCLI to associate an Image Profile with this host.
Alternatively, you can reevaluate the rules for this host with the
Test-DeployRuleSetCompliance and Repair-DeployRuleSetCompliance cmdlets.
```

コンソールは、ベンダー、シリアル番号、IP アドレスなどを含むホストのマシン属性を表示します。

原因

このホストに現在関連付けられているイメージ プロファイルはありません。

解決方法

Apply-ESXImageProfile コマンドレットを実行するか、次のルールを作成して、ホストにイメージ プロファイルを割り当てることができます。

- 1 New-DeployRule コマンドレットを実行し、イメージ プロファイルを持つホストに一致するパターンを含むルールを作成します。
- 2 Add-DeployRule コマンドレットを実行し、ルールセットにルールを追加します。
- 3 Test-DeployRuleSetCompliance コマンドレットを実行し、そのコマンドレットの出力を Repair-DeployRuleSetCompliance コマンドレットの入力に使用します。

vSphere Auto Deploy ホストが TFTP サーバに接続できない

vSphere Auto Deploy でプロビジョニングしたホストが TFTP サーバに接続できません。

問題

vSphere Auto Deploy でプロビジョニングしたホストを起動しようとする、ネットワーク ブートが実行され、DHCP サーバによって DHCP アドレスが割り当てられますが、TFTP サーバに接続できません。

原因

TFTP サーバが実行を停止したか、ファイアウォールで TFTP ポートがブロックされた可能性があります。

解決方法

- WinAgents TFTP サーバがインストールされている場合は、WinAgents TFTP 管理コンソールを開き、サービスが実行されていることを確認します。サービスが実行されている場合は、Windows ファイアウォールの着信ルールで TFTP ポートがブロックされていないことを確認します。一時的にファイアウォールを無効にし、問題の原因がファイアウォールかどうかを確認します。
- それ以外のすべての TFTP サーバでのデバッグ手順の詳細については、サーバのドキュメントを参照してください。

vSphere Auto Deploy ホストが ESXi イメージを vSphere Auto Deploy サーバから取得できない

vSphere Auto Deploy でプロビジョニングされたホストが iPXE 起動画面で停止します。

問題

vSphere Auto Deploy でプロビジョニングされたホストを起動しようとする、iPXE 起動画面で起動プロセスが停止し、ホストが vSphere Auto Deploy サーバから ESXi イメージを取得しようとしていることを示すステータス メッセージが表示されます。

原因

vSphere Auto Deploy サービスが停止しているか、vSphere Auto Deploy サーバにアクセスできない可能性があります。

解決方法

- 1 vSphere Auto Deploy サーバをインストールしたシステムにログインします。
- 2 vSphere Auto Deploy サーバが実行中であることを確認します。
 - a [スタート] - [設定] - [コントロール パネル] - [管理ツール] をクリックします。
 - b [サービス] をダブルクリックし、サービス管理パネルを開きます。
 - c サービス フィールドで VMware vSphere Auto Deploy Waiter サービスを探し、サービスが実行されていない場合は再び開始します。
- 3 Web ブラウザを開き、次の URL を入力して、vSphere Auto Deploy サーバがアクセス可能であるかどうかを確認します。

`https://Auto_Deploy_Server_IP_Address:Auto_Deploy_Server_Port/vmw/rdb`

注： このアドレスは、サーバがアクセス可能かどうかを確認する目的でのみ使用してください。

- 4 サーバにアクセスできない場合は、ファイアウォールに問題がある可能性があります。
 - a vSphere Auto Deploy サーバ ポート用に TCP 着信許容ルールを設定します。
ポートは、インストール時に別のポートを指定しない限り 6501 です。
 - b 最後の手段として、一時的にファイアウォールを無効にし、トラフィックがファイアウォールによってブロックされているかどうかを確認してから再び有効にします。本番環境ではファイアウォールを無効にしないでください。

ファイアウォールを無効にするには、**netsh firewall set opmode disable** を実行します。ファイアウォールを有効にするには、**netsh firewall set opmode enable** を実行します。

vSphere Auto Deploy ホストが DHCP によって割り当てられたアドレスを取得しない

vSphere Auto Deploy を使用してプロビジョニングしたホストが、DHCP アドレスを取得できません。

問題

vSphere Auto Deploy でプロビジョニングしたホストを起動しようとする、ネットワーク ブートは実行されませんが、DHCP アドレスが割り当てられません。vSphere Auto Deploy サーバでは、イメージ プロファイルを使用するホストをプロビジョニングできません。

原因

DHCP サービスまたはファイアウォールの設定に問題がある可能性があります。

解決方法

- 1 DHCP サーバ サービスが、ホストをプロビジョニングするための DHCP サーバが設定されている Windows システムで実行されていることを確認してください。
 - a [スタート] - [設定] - [コントロール パネル] - [管理ツール] をクリックします。
 - b [サービス] をダブルクリックし、サービス管理パネルを開きます。
 - c サービス フィールドで DHCP サーバ サービスを探し、それが実行されていない場合は再び開始します。
- 2 DHCP サーバが実行中の場合は、ターゲット ホストに対して構成した DHCP 範囲と DHCP 予約を再確認します。

DHCP 範囲と DHCP 予約が正しく構成されている場合、問題の原因として最も有力なのはファイアウォール関連です。
- 3 一時的な回避策として、ファイアウォールを無効にし、問題が解決されるかどうかを確認します。
 - a [スタート] - [プログラム] - [アクセサリ] - [コマンド プロンプト] をクリックして、コマンド プロンプトを開きます。
 - b 次のコマンドを入力して、ファイアウォールを一時的に無効にします。本番環境ではファイアウォールを無効にしないでください。

netsh firewall set opmode disable

- c vSphere Auto Deploy を使用してホストをプロビジョニングします。
- d 次のコマンドを入力して、ファイアウォールを再び有効にします。

```
netsh firewall set opmode enable
```

- 4 ターゲット ホストへの DHCP ネットワーク トラフィックを許可するルールを設定します。

詳細については、DHCP および DHCP サーバを実行する Windows システムについてのファイアウォールに関するドキュメントを参照してください。

vSphere Auto Deploy ホストがネットワーク ブートしない

vSphere Auto Deploy を使用してプロビジョニングしたホストが起動しますが、ネットワーク ブートしません。

問題

vSphere Auto Deploy でプロビジョニングしたホストを起動しようとしても、ネットワーク ブートのプロセスが開始しません。

原因

ネットワーク ブートがホストで有効になっていません。

解決方法

- 1 ホストを再起動し、画面の説明に従って BIOS 構成にアクセスします。
- 2 BIOS 構成の起動デバイス構成で、ネットワーク ブートを有効にします。

vSphere Auto Deploy サーバのデータベースの破損からのリカバリ

場合によっては、vSphere Auto Deploy データベースで問題が発生することがあります。最も効率的なリカバリオプションは、既存のデータベース ファイルを最新のバックアップに置き換えることです。

問題

vSphere Auto Deploy を使用して環境内の ESXi ホストをプロビジョニングすると、vSphere Auto Deploy データベースで問題が発生することがあります。

重要： この問題はまれに発生します。現在のデータベース ファイルを置き換える前に、その他すべての vSphere Auto Deploy トラブルシューティング方法に従ってください。選択したバックアップ以降に作成したルールまたはアソシエーションは失われます。

原因

この問題は、vSphere Auto Deploy でプロビジョニングされたホストでのみ発生します。

解決方法

- 1 vSphere Auto Deploy サーバ サービスを停止します。
- 2 vSphere Client の [vSphere Auto Deploy] ページに移動して、vSphere Auto Deploy のログを検索します。

- 3 ログで次のメッセージがないかを確認します。

```
DatabaseError: database disk image is malformed.
```

このメッセージが表示された場合は、既存のデータベースを最新のバックアップに置き換えます。

- 4 vSphere Auto Deploy のデータ ディレクトリ (`/var/lib/rbd`) に移動します。

ディレクトリには `db` という名前のファイルと、`db-yyy-mm-dd` という名前のバックアップ ファイルが含まれています。

- 5 現在の `db` ファイルの名前を変更します。

VMware サポートに問い合わせたときに、そのファイルが要求される可能性があります。

- 6 `db` に最新のバックアップの名前を変更します。

- 7 vSphere Auto Deploy サーバ サービスを再起動します。

- 8 まだメッセージがログに表示される場合は、この手順を繰り返して、vSphere Auto Deploy がデータベース エラーを発生させることなく機能するまで次に最新のバックアップを使用します。

ESXi の設定

各トピックでは、ダイレクト コンソール ユーザー インターフェイスの使用、および ESXi のデフォルトの構成について説明します。

初期状態の ESXi 構成

ESXi の自動構成

ESXi ホストの初回起動時や、構成のデフォルト値の再設定後に、ホストは自動構成段階に入ります。この段階で、システムのネットワーク デバイスやストレージ デバイスがデフォルト設定で構成されます。

デフォルトでは、DHCP (Dynamic Host Configuration Protocol) によって IP が構成され、仮想マシンをディスクに格納できるよう、すべての空の仮想内蔵ディスクが VMFS (仮想マシン ファイル システム) でフォーマットされます。

次に参照するドキュメント

ESXi のリモート管理

ESXi ホストは、VMware Host Client、vSphere Client、および vCenter Server を使用して管理することができます。

vCenter Server および vCenter Server コンポーネントのダウンロードとインストールの手順については、『vCenter Server のインストールとセットアップ』を参照してください。VMware Host Client のインストールの詳細については、『vSphere 単一ホスト管理』を参照してください。

ESXi のダイレクト コンソール インターフェイスについて

ダイレクト コンソール インターフェイスは、初期の ESXi 構成とトラブルシューティングに使用します。

ダイレクト コンソールを使用するには、キーボードとモニタをホストに接続します。ホストの自動構成段階が完了すると、モニタにダイレクト コンソールが表示されます。デフォルトのネットワーク構成を確認し、ネットワーク環境と互換性がない設定を変更できます。

ダイレクト コンソールでは、キーを使用して次の処理を行えます。

- ホストの構成
- 管理用アクセスの設定
- トラブルシューティング

また、vSphere Client を使用して vCenter Server からホストを管理することもできます。

表 4-20. ダイレクト コンソールでの操作

操作	キー
構成の表示と変更	F2
ユーザー インターフェイスのハイコントラスト モードへの変更	F4 キー
ホストのシャットダウンまたは再起動	F12 キー
VMkernel ログの表示	Alt + F12
シェル コンソールへの切り替え	Alt + F1
ダイレクト コンソール ユーザー インターフェイスへの切り替え	Alt + F2
フィールド間での選択の移動	矢印キー
メニュー項目の選択	Enter
値の切り替え	スペースバー
構成をデフォルトにリセットするなどの、注意を払う必要があるコマンドの確認	F11 キー
保存して終了	Enter
保存せずに終了	Esc
システム ログを終了	q

ダイレクト コンソール用のキーボード レイアウトの構成

ダイレクト コンソールで使用するキーボードの配列を構成できます。

手順

- 1 ダイレクト コンソールから、[キーボードの構成] を選択して [Enter] を押します。
- 2 使用する配列を選択します。
- 3 スペースバーを押して、オンやオフの選択を切り替えます。

- 4 [Enter] を押します。

ダイレクト コンソールのセキュリティ バナーの作成

セキュリティ バナーは、ダイレクト コンソールの[ようこそ]画面に表示されるメッセージです。

手順

- 1 vSphere Client から vCenter Server に接続します。
- 2 インベントリでホストを選択します。
- 3 [設定] タブをクリックします。
- 4 [システム] の下で [システムの詳細設定] を選択します。
- 5 Annotations.WelcomeMessage を選択します。
- 6 [編集] アイコンをクリックします。
- 7 セキュリティ メッセージを入力します。

結果

このメッセージがダイレクト コンソールの[ようこそ]画面に表示されます。

ダイレクト コンソールのシリアル ポートへのリダイレクト

ESXi ホストをシリアル コンソールからリモートで管理するために、ダイレクト コンソールをシリアル ポートにリダイレクトすることができます。

vSphere は、シリアル ポートを介してダイレクト コンソールを表示するため、VT100 のターミナル タイプと PuTTY のターミナル エミュレータをサポートします。

ダイレクト コンソールは、いくつかの方法でシリアル ポートにリダイレクトできます。

次に参照するドキュメント

起動オプションの手動設定によるダイレクト コンソールのシリアル ポートへのリダイレクト

起動オプションを設定してダイレクト コンソールをシリアル ポートにリダイレクトする場合、その変更は次回の起動では維持されません。

前提条件

シリアル ポートがシリアル ログ機能およびデバッグに使用されていないことを確認します。

手順

- 1 ホストを起動します。
- 2 VMware ハイパーバイザーのロード中ウィンドウが表示されたら、Shift + O を押して起動オプションを編集します。

- 3 次の起動オプションを入力して、com1 の logPort および gdbPort を無効にし、tty2Port を com1 に設定します。

```
"gdbPort=none logPort=none tty2Port=com1";
```

代わりに com2 を使用するには、com1 を com2 に置き換えます。

結果

ホストを再起動するまで、ダイレクト コンソールはシリアル ポートにリダイレクトされます。次回以降の起動でもダイレクト コンソールをリダイレクトするには、[vSphere Client からの、ダイレクト コンソールのシリアルポートへのリダイレクト](#) を参照してください。

vSphere Client からの、ダイレクト コンソールのシリアルポートへのリダイレクト

com1 または com2 のいずれかのシリアルポートにダイレクト コンソールをリダイレクトすることにより、シリアルポートに接続されたコンソールから、リモートで ESXi ホストを管理できます。vSphere Client を使用してダイレクト コンソールをシリアルポートにリダイレクトする場合、設定する起動オプションは以降の再起動後も維持されます。

前提条件

- vSphere Client からホストにアクセスできることを確認します。
- シリアルポートが、シリアルログおよびデバッグ、または ESX Shell (tty1Port) 用に使用されていないことを確認します。

手順

- 1 vSphere Client から vCenter Server に接続します。
- 2 インベントリでホストを選択します。
- 3 [設定] タブをクリックします。
- 4 [システム] の下で [システムの詳細設定] を選択します。
- 5 [VMkernel.Boot.logPort] および [VMkernel.Boot.gdbPort] のフィールドが、ダイレクト コンソールのリダイレクト先にする COM ポートを使用するように設定されていないことを確認します。
- 6 ダイレクト コンソールをリダイレクトするシリアルポート **com1** または **com2** に、[VMkernel.Boot.tty2Port] を設定します。
- 7 ホストを再起動します。

結果

これで、シリアルポートに接続されたコンソールから、リモートで ESXi ホストを管理できるようになりました。

Auto Deploy でデプロイされたホストのシリアルポートへの、ダイレクト コンソールのリダイレクト

ダイレクト コンソールをシリアルポートにリダイレクトしたあとで、その設定を、Auto Deploy でホストを再プロビジョニングするときも保持されるホスト プロファイルに組み込むことができます。

前提条件

シリアル ポートがシリアル ログ機能およびデバッグに使用されていないことが必要です。

手順

- 1 から vCenter Server に接続します。
- 2 インベントリでホストを選択します。
- 3 [システム] の下で [システムの詳細設定] を選択します。
- 4 [VMkernel.Boot.logPort] および [VMkernel.Boot.gdbPort] のフィールドが、ダイレクト コンソールのリダイレクト先にする COM ポートを使用するように設定されていないことを確認します。
- 5 ダイレクト コンソールをリダイレクトするシリアル ポート **com1** または **com2** に、[VMkernel.Boot.tty2Port] を設定します。
- 6 [OK] をクリックします。
- 7 ホスト プロファイルを保存し、そのプロファイルにホストを添付します。『vSphere のホスト プロファイル』ドキュメントを参照してください。

結果

ダイレクト コンソールをシリアル ポートにリダイレクトする設定は、vCenter Server で保存され、Auto Deploy でホストを再プロビジョニングするときも保持されます。

ダイレクト コンソール ユーザー インターフェイスを使用した、ESXi Shell および SSH アクセスの有効化

ダイレクト コンソール ユーザー インターフェイスを使用して、ESXi Shell を有効にします。

手順

- 1 ダイレクト コンソール ユーザー インターフェイスで、F2 を押してシステムのカスタマイズ メニューにアクセスします。
- 2 [トラブルシューティング オプション] を選択し、Enter キーを押します。
- 3 [トラブルシューティング モード オプション] メニューから、有効にするサービスを選択します。
 - ESXi Shell の有効化
 - SSH の有効化
- 4 Enter キーを押してサービスを有効にします。
- 5 (オプション) ESXi Shell のタイムアウトを設定します。

デフォルトでは、ESXi Shell のタイムアウトは 0 (無効) となっています。

可用性タイムアウト設定は、ESXi Shell を有効にしてから何分後までにログインする必要があるかを示します。タイムアウト期間後にログインしていない場合は、シェルは無効になります。

注： タイムアウト期間が経過したときにログイン済みの場合は、セッションが維持されます。ただし、ESXi Shell は無効にされて、他のユーザーがログインできないようにします。

a トラブルシューティング モード オプション メニューから、[ESXi Shell および SSH のタイムアウトの変更] を選択し、Enter を押します。

b 可用性タイムアウト時間を分単位で入力します。

可用性タイムアウトは、ESXi Shell を有効にしてから何分後までにログインする必要があるかを示します。

c Enter を押します。

d アイドル タイムアウトを入力します。

アイドル タイムアウト設定は、ユーザーが対話形式のアイドル セッションからログアウトされるまでの許容分数を示します。アイドル タイムアウトの変更は、ユーザーが次に ESXi Shell にログインする際に適用されるため、既存のセッションは影響を受けません。

6 ダイレクト コンソール ユーザー インターフェイスのメイン メニューに戻るまで、Esc を押します。

システム管理者アカウントのパスワードの設定

ダイレクト コンソールを使用して、システム管理者アカウント (root) のパスワードを設定できます。

ESXi ホストの管理者ユーザー名は root です。デフォルトでは、システム管理者のパスワードは設定されていません。

手順

1 ダイレクト コンソールから、[パスワードの構成] を選択します。

2 (オプション) パスワードがすでに設定されている場合は、[古いパスワード] 行にそのパスワードを入力し、[Enter] を押します。

3 [新しいパスワード] 行に新しいパスワードを入力し、[Enter] を押します。

4 新しいパスワードを再入力し、[Enter] を押します。

BIOS の起動設定の構成

サーバに複数のドライブがあると、BIOS 設定を構成する必要がある場合があります。

BIOS の起動構成によって、サーバの起動方法が決定されます。一般に、CD-ROM デバイスが最初にリストされません。

注： ESXi Embedded を使用している場合、BIOS の起動構成によって、サーバが ESXi ブート デバイスで起動するか別のブート デバイスで起動するかが決まります。一般的に、ESXi のホスト マシンの BIOS の起動設定では、USB フラッシュ デバイスが最初にリストされます。

ESXi を UEFI モードでインストールまたはアップグレードする場合、インストーラは VMware ESXi という名前の UEFI 起動オプションを作成し、それをデフォルトの起動オプションにするため、起動順序を変更する必要はありません。

起動時に BIOS で起動順序を構成したり、起動デバイスの選択メニューから起動デバイスを選択したりすることで、起動設定を変更できます。BIOS で起動順序を変更すると、新しい設定は、次回以降のすべての再起動に影響します。起動デバイスの選択メニューから起動デバイスを選択すると、その選択は現在の起動のみに影響します。

一部のサーバには起動デバイスの選択メニューがありません。この場合、1 回の起動時に限定して変更する場合でも BIOS で起動順序を変更する必要があります。その後、次の起動時に起動順序を元に戻します。

ESXi 用の BIOS の起動設定の変更

サーバの起動時にデフォルトで ESXi が起動されるようにするには、ESXi 用に BIOS の起動設定を構成します。

ESXi Installable および ESXi Embedded は、同じホスト上に共存できません。

手順

- 1 ESXi ホストの起動中に、ホストの BIOS セットアップに切り替えるために必要なキーを押します。

サーバのハードウェアに応じて、このキーはファンクション キーまたは Delete キーになる場合があります。BIOS セットアップを開くオプションは、サーバによって異なる場合があります。

- 2 BIOS の起動設定を選択します。

オプション	説明
インストール可能なバージョンの ESXi を使用している場合	ESXi ソフトウェアをインストールしたディスクを選択し、リストの先頭に移動します。ホストは ESXi で起動します。
ESXi Embedded を使用している場合	USB フラッシュ デバイスを選択し、リストの先頭に移動します。ホストは ESXi モードで起動します。

仮想メディア用の起動設定の構成

ESXi の設定にリモート管理ソフトウェアを使用している場合、仮想メディアの起動設定をする必要がある場合があります。

仮想メディアは、ネットワーク上のあらゆる場所にある可能性のあるターゲット サーバに、リモートのストレージ メディア（CD-ROM、USB 大容量記憶装置、ISO イメージ、およびフロッピー ディスクなど）を接続する方法です。ターゲット サーバはリモート メディアに接続し、サーバの USB ポートに物理的に接続されているかのように、読み取りや書き込みを行うことができます。

前提条件

ESXi Installable および ESXi Embedded は、同じホスト上に共存できません。

手順

- 1 メディアを仮想デバイスに接続します。

たとえば Dell サーバを使用している場合、DRAC (Dell Remote Access Controller) または類似のリモート管理インターフェイスにログインして、物理フロッピーまたは CD-ROM ドライブを選択するか、フロッピーイメージまたは CD-ROM イメージへのパスを指定します。

- 2 サーバを再起動します。
- 3 サーバの起動中に、デバイス選択メニューに入ります。

サーバのハードウェアに応じて、このキーはファンクション キーまたは Delete キーになる場合があります。

- 4 表示される指示に従って仮想デバイスを選択します。

結果

サーバは構成されたデバイスで一度起動し、次の起動時にはデフォルトの起動順序に戻ります。ESXi を UEFI モードでインストールまたはアップグレードする場合、システムはデフォルトの起動順序が VMware ESXi に設定されているため、起動順序を変更する必要はありません。

ネットワーク設定の構成

ESXi では、管理ネットワーク用に IP アドレスが 1 つ必要です。基本的なネットワーク設定を構成するには、vSphere Client またはダイレクト コンソールを使用します。

DHCP サーバによって割り当てられる IP アドレスがニーズを満たす場合は、vSphere Client を使用します。

次の場合に、ネットワーク構成にダイレクト コンソールを使用します。

- DHCP サーバによって割り当てられる IP アドレスがニーズを満たしていない。
- DHCP サーバによって割り当てられる IP アドレスの使用が許可されていない。
- ESXi に IP アドレスがない。この状況は、自動設定段階で DHCP の設定が失敗した場合に発生する可能性があります。
- 自動構成段階中に、誤ったネットワーク アダプタが選択された。

ESXCLI コマンドを使用してネットワーク設定を構成します。 [esxcli network Commands](#) を参照してください。

ESXi ホストへのネットワーク アクセス

デフォルトの動作は、DHCP を使用した、ESXi 管理ネットワークの構成です。インストールの完了後に、デフォルトの動作をオーバーライドし、管理ネットワーク用の固定 IP 設定を使用できます。

表 4-21. ESXi がサポートするネットワーク構成シナリオ

使用例	アプローチ方法
DHCP 構成の IP 設定を受け入れる。	ESXi ダイレクト コンソールで、ESXi 管理インターフェイスに DHCP 経由で割り当てられた IP アドレスを確認できます。この IP アドレスは、vSphere Client からホストに接続して、管理 IP アドレスの変更を含む設定のカスタマイズを行うために使用できます。
次のいずれかに当てはまる場合 <ul style="list-style-type: none"> ■ DHCP サーバがない。 ■ ESXi ホストが DHCP サーバに接続されていない。 ■ 接続している DHCP サーバが正常に機能していない。 	自動構成段階中に、ソフトウェアによってサブネット 169.254.x.x/16 内のリンク ローカル IP アドレスが割り当てられます。割り当てられた IP アドレスがダイレクト コンソールに表示されます。 リンク ローカル IP アドレスは、ダイレクト コンソールを使用して固定 IP アドレスを構成すると、オーバーライドできます。
ESXi ホストは作動中の DHCP サーバに接続されているが、DHCP 構成の IP アドレスを使用したくない。	自動構成段階中に、ソフトウェアによって DHCP 構成の IP アドレスが割り当てられます。 最初の接続は、DHCP 構成の IP アドレスを使用して確立できます。そのあとで、固定 IP アドレスを構成できます。 ESXi ホストに物理的に接続している場合、ダイレクト コンソールを使用して固定 IP アドレスを構成し、DHCP 構成の IP アドレスをオーバーライドできます。
導入しているセキュリティ ポリシーによって、ネットワーク上で構成されていないホストの起動が許可されない。	ネットワークに接続されていないホストのネットワーク設定の構成の設定手順を実行してください。

ESXi のネットワーク セキュリティに関する推奨事項

ESXi 環境の保護には、ネットワーク トラフィックの隔離が不可欠です。それぞれのネットワークで、さまざまなアクセスおよび隔離レベルが必要です。

ESXi ホストは、複数のネットワークを使用します。各ネットワークに適切なセキュリティ対策を使用し、特定のアプリケーションと機能のトラフィックを隔離します。たとえば、仮想マシンが配置されたネットワーク上を VMware vSphere® vMotion® トラフィックが通過しないようにします。隔離するとスヌーピングされません。パフォーマンス上の理由から、別個のネットワークを使用することも推奨されます。

- vSphere vMotion、VMware vSphere Fault Tolerance、VMware vSAN、およびストレージなどの機能には、vSphere インフラストラクチャ ネットワークを使用します。それぞれの機能用にネットワークを分離します。多くの場合、単一の物理サーバ ラックの外部にこれらのネットワークをルーティングさせる必要はありません。
- 管理ネットワークは、クライアントのトラフィック、コマンドライン インターフェイス (CLI) または API トラフィック、およびサードパーティ製のソフトウェア トラフィックを他のトラフィックから隔離します。通常、管理ネットワークには、システム管理者、ネットワーク管理者およびセキュリティ管理者のみがアクセスできます。管理ネットワークへのアクセスを保護するには、Bastion ホストまたは仮想プライベート ネットワーク (VPN) を使用します。このネットワーク内のアクセスを厳密に管理します。
- 仮想マシンのトラフィックは、1つ以上または多数のネットワークを通過できます。仮想ネットワーク コントローラでファイアウォール ルールを設定した仮想ファイアウォール ソリューションを使用すると、仮想マシンの隔離を強化できます。vSphere 環境内のホスト間で仮想マシンを移行すると、これらの設定も仮想マシンとともに移行されます。

管理ネットワーク用ネットワーク アダプタの選択

ESXi ホストとほかの外部管理ソフトウェア間のトラフィックは、ホストのイーサネット ネットワーク アダプタを通して送受信されます。ダイレクト コンソールを使用して、管理ネットワークで使用されるネットワーク アダプタを選択します。

外部管理ソフトウェアの例には、vCenter Server および SNMP クライアントが含まれます。ホストのネットワーク アダプタは `vmnicN` と名前が付けられます。この N は、ネットワーク アダプタを識別する一意の番号です (`vmnic0`、`vmnic1` など)。

自動構成段階中に、ESXi ホストは管理トラフィック用に `vmnic0` を選択します。ホストの管理トラフィックを送受信するネットワーク アダプタを手動で選択して、デフォルトの選択をオーバーライドできます。管理トラフィック用にギガビット イーサネット ネットワーク アダプタを使用することもできます。可用性の確保に役立つ別の方法として、複数のネットワーク アダプタを選択できます。複数のネットワーク アダプタを使用すると、ロード バランシング機能とフェイルオーバー機能が有効になります。

手順

- 1 ダイレクト コンソールから、[管理ネットワークの構成] を選択して Enter キーを押します。
- 2 [ネットワーク アダプタ] を選択して Enter キーを押します。
- 3 ネットワーク アダプタを選択して Enter キーを押します。

結果

ネットワークが機能し始めたら、vSphere Client を使用して vCenter Server を介して ESXi ホストに接続できます。

VLAN ID の設定

ESXi ホストの仮想 LAN (VLAN) ID 番号を設定できます。

手順

- 1 ダイレクト コンソールから、[管理ネットワークの構成] を選択して [Enter] を押します。
- 2 [VLAN] を選択して [Enter] を押します。
- 3 1 ~ 4094 の範囲内で VLAN ID 番号を入力します。

ESXi の IP 設定の構成

デフォルトで、DHCP は IP アドレス、サブネット マスク、およびデフォルト ゲートウェイを設定します。

今後の参考のために、IP アドレスを書き留めておいてください。

DHCP が機能するためには、ネットワーク環境に DHCP サーバを置く必要があります。DHCP を使用できない場合、ホストはサブネット 169.254.x.x/16 内のリンク ローカル IP アドレスを割り当てます。割り当てられた IP アドレスは、ダイレクト コンソールに表示されます。ホストにモニタを物理的に接続していない場合は、リモート管理アプリケーションを使用してダイレクト コンソールにアクセスできます。[リモート管理アプリケーションの使用](#) を参照してください。

ダイレクト コンソールにアクセスすると、固定ネットワーク アドレスを任意で構成できます。デフォルトのサブネット マスクは 255.255.0.0 です。

ダイレクト コンソールからの IP 設定の構成

ホストに物理的に接続しているか、ダイレクト コンソールにリモート アクセスしている場合、ダイレクト コンソールを使用して、IP アドレス、サブネット マスク、およびデフォルト ゲートウェイを構成できます。

手順

- 1 [管理ネットワークの構成] を選択して [Enter] を押します。
- 2 [IP 構成] を選択して [Enter] を押します。
- 3 [固定 IP アドレスおよびネットワーク構成を設定] を選択します。
- 4 IP アドレス、サブネット マスク、およびデフォルト ゲートウェイを入力し、Enter を押します。

vSphere Client からの IP 設定の構成

ホストに物理的にアクセスできない場合には、vSphere Client を使用して固定 IP 設定を構成できます。

手順

- 1 vSphere Client から vCenter Server にログインします。
- 2 インベントリでホストを選択します。
- 3 [設定] タブで、[ネットワーク] を展開します。
- 4 [VMkernel アダプタ] を選択します。
- 5 [vmk0 管理ネットワーク] を選択し、編集アイコンをクリックします。
- 6 [IPv4 設定] を選択します。
- 7 [固定 IPv4 設定を使用します] を選択します。
- 8 固定 IPv4 アドレス設定を入力または変更します。
- 9 (オプション) 固定 IPv6 アドレスを設定します。
 - a [IPv6 設定] を選択します。
 - b [固定 IPv6 アドレス] を選択します。
 - c 追加アイコンをクリックします。
 - d IPv6 アドレスを入力し、[OK] をクリックします。
- 10 [OK] をクリックします。

ESXi の DNS の構成

ESXi ホストの DNS の構成は、手動または自動のいずれかを選択できます。

デフォルトは自動です。自動 DNS が機能するためには、ネットワーク環境に DHCP サーバと DNS サーバを置く必要があります。

自動 DNS を使用できない、または自動 DNS が望ましくないネットワーク環境では、ホスト名、プライマリ ネーム サーバ、セカンダリ ネーム サーバ、および DNS サフィックスなどの固定 DNS 情報を構成できます。

ダイレクト コンソールからの DNS 設定の構成

ホストに物理的に接続している場合、またはダイレクト コンソールにリモート アクセスしている場合、ダイレクト コンソールを使用して DNS 情報を構成できます。

手順

- 1 [管理ネットワークの構成] を選択して [Enter] を押します。
- 2 [DNS 構成] を選択して [Enter] を押します。
- 3 [次の DNS サーバ アドレスとホスト名を使用] を選択します。
- 4 プライマリ サーバ、代替サーバ (任意)、およびホスト名を入力します。

DNS サフィックスの構成

ホストに物理的に接続している場合、ダイレクト コンソールを使用して DNS 情報を構成できます。デフォルトでは、DHCP によって DNS サフィックスが取得されます。

手順

- 1 ダイレクト コンソールから、[管理ネットワークの構成] を選択します。
- 2 [カスタム DNS サフィックス] を選択して [Enter] を押します。
- 3 新しい DNS サフィックスを入力します。

ネットワークに接続されていないホストのネットワーク設定の構成

一部の高セキュリティ環境では、ネットワーク上の未構成のホストの起動は許可されません。ホストをネットワークに接続する前にホストを構成できます。

前提条件

ホストにネットワーク ケーブルが接続されていないことを確認します。

手順

- 1 ホストを起動します。
- 2 ダイレクト コンソール ユーザー インターフェイスを使用して、管理者アカウント (root) のパスワードを構成します。
- 3 ダイレクト コンソール ユーザー インターフェイスを使用して、固定 IP アドレスを構成します。
- 4 ホストにネットワーク ケーブルを接続します。
- 5 (オプション) vSphere Client を使用して、vCenter Server システムに接続します。
- 6 (オプション) ホストを vCenter Server インベントリに追加します。

管理ネットワークのテスト

ダイレクト コンソールを使用して、簡単なネットワーク接続テストを実行できます。

ダイレクト コンソールは次のテストを実行します。

- デフォルト ゲートウェイの接続の確認
- プライマリ DNS ネームサーバの接続の確認
- セカンダリ DNS ネームサーバの接続の確認
- 構成されているホスト名の解決

手順

- 1 ダイレクト コンソールから、[管理ネットワークのテスト] を選択して [Enter] を押します。
- 2 [Enter] を押してテストを開始します。

管理エージェントの再起動

管理エージェントは VMware コンポーネントと同期し、vSphere Client および vCenter Server を使用して ESXi ホストにアクセスできるようになります。

vSphere Client と vCenter Server は、vSphere ソフトウェアとともにインストールされます。リモート アクセスが中断された場合、管理エージェントを再起動する必要がある場合があります。管理エージェントを再起動すると、ESXi ホストの /etc/init.d にインストールされ実行されているすべての管理エージェントおよびサービスが再起動します。通常、このエージェントには、hostd、ntpd、sfcbd、slpd、wsman、vobd などが含まれています。ソフトウェアは、フォールト ドメイン マネージャ (FDM) も再起動します (インストールされている場合)。

管理エージェントを再起動すると、ユーザーは vSphere Client および vCenter Server を使用してこのホストにアクセスできなくなります。

手順

- 1 ダイレクト コンソールから、[トラブルシューティング オプション] を選択して Enter キーを押します。
- 2 [管理エージェントの再起動] を選択して Enter キーを押します。
- 3 F11 を押して再起動します。

結果

ESXi ホストが、管理エージェントおよびサービスを再起動します。

管理ネットワークの再起動

管理ネットワーク インターフェイスの再起動には、ネットワークのリストアまたは DHCP リースの更新が必要になる場合があります。

管理ネットワークの再起動によって、ネットワークの機能が短時間停止し、実行中の仮想マシンに一時的に影響を与える場合があります。

DHCP リースの更新によって新しいネットワーク ID (IP アドレスやホスト名) が提供された場合、リモート管理ソフトウェアは切断されます。

手順

- 1 ダイレクト コンソールから、[管理ネットワークの再起動] を選択して [Enter] を押します。
- 2 [F11] を押して再起動します。

デバイスとネットワークへの接続のテスト

ダイレクト コンソールを使用して、いくつかの単純なネットワーク接続テストを実行し、他のデバイスとネットワークを指定できます。

手順

- 1 ダイレクト コンソールから、[管理ネットワークのテスト] を選択して Enter キーを押します。
- 2 アドレスを入力して接続を確認するか、ほかの DNS ホスト名を入力して特定します。
- 3 Enter キーを押してテストを開始します。

標準スイッチのリストア

vSphere Distributed Switch は、関連するすべてのホストにおいて単一の仮想スイッチとして機能します。

仮想マシンは、複数のホスト間を移動するときに、一貫したネットワーク構成を維持できます。既存の標準スイッチまたは仮想アダプタを Distributed Switch に移行し、Distributed Switch が不要になったり、機能を停止した場合に、標準スイッチをリストアして、ホストへのアクセスを維持できます。標準スイッチをリストアすると、新しい仮想アダプタが作成され、現在 Distributed Switch に接続されている管理ネットワーク アップリンクが新しい仮想スイッチに移行されます。

次の理由によって、標準スイッチのリストアが必要になる場合があります。

- 分散スイッチが不要である、または機能していない。
- vCenter Server への接続をリカバリするために分散スイッチの修復が必要で、ホストはアクセス可能なままにする必要がある。
- vCenter Server でホストを管理しない。ホストが vCenter Server に接続していない場合、ホストではほとんどの分散スイッチの機能が使用できません。

前提条件

管理ネットワークが分散スイッチに接続されていることを確認します。

手順

- 1 ダイレクト コンソールから、[標準スイッチをリストア] を選択して Enter キーを押します。
ホストが標準スイッチにある場合、この選択肢は淡色表示され、選択できません。
- 2 F11 キーを押して確認します。

システム ログの設定

ESXi ホストは、システム コンポーネントからのメッセージをログ ファイルに書き込み、Syslog コレクタにメッセージを転送できる Syslog サービス (vmsyslogd) を実行します。

ログのサイズと場所を構成できます。また、ログ フィルタを作成して適用することで、ESXi ホストのログ ポリシーを変更できます。

リモート ホストを Syslog 用に構成する場合は、ログ メッセージの転送を許可するために、ESXi ホスト ファイアウォールの特定のポートを開くことも必要です。詳細については、「[Opening the firewall for syslog emission to remote hosts](#)」を参照してください。

ESXi ホストでの Syslog の構成

vSphere Client、VMware Host Client、または `esxcli system syslog` コマンドを使用して syslog サービスを構成できます。

Syslog サービスは、環境内で予防的なアクションを実行するのに役立つ分析用のログ メッセージを受信、分類、および保存します。

vSphere Client を使用した ESXi Syslog の設定

vSphere Client コマンドを使用して、Syslog サービスをグローバルに構成し、さまざまな詳細設定を編集できます。

手順

- 1 vSphere Client インベントリで、ESXi ホストに移動して参照します。
- 2 [構成] をクリックします。
- 3 [システム] メニューの [システムの詳細設定] をクリックします。
- 4 [編集] をクリックします。
- 5 **syslog** でフィルタリングします。
- 6 ログ作成をグローバルに設定し、さまざまな詳細設定を構成するには、[ESXi Syslog のオプション](#)を参照してください。
- 7 (オプション) 任意のログで、デフォルトのログ サイズとログ ローテーションを上書きします。
 - a カスタマイズするログの名前をクリックします。
 - b ローテーション数とログ サイズを入力します。
- 8 [OK] をクリックします。

結果

Syslog オプションの変更が有効になります。

注： vSphere Client または VMware Host Client を使用して定義された Syslog パラメータの設定は、すぐに有効になります。ただし、ESXCLI を使用して定義されたほとんどの設定を有効にするには、追加のコマンドを実行する必要があります。詳細については、[ESXi Syslog のオプション](#)を参照してください。

VMware Host Client を使用した ESXi Syslog の設定

VMware Host Client を使用して、ESXi ホストの Syslog サービス パラメータを構成および編集できます。

手順

- 1 VMware Host Client の [ホスト] で、[管理 > システム > 詳細設定] の順にクリックします。
- 2 [検索] パネルで、定義する Syslog 設定を入力します。ESXi Syslog のオプションを参照してください。
- 3 設定を選択し、[オプションの編集]をクリックします。
- 4 ESXi Syslog のオプションのパラメータの表の説明に従って値を設定します。
- 5 [保存] をクリックします。

ESXCLI を使用した ESXi Syslog の設定

ESXCLI コマンド (`esxcli system syslog config set <syslog option>`) を使用して、ESXi ホストで Syslog サービスを構成できます。

前提条件

`esxcli system syslog` コマンドおよびその他の ESXCLI コマンドの使用方法については、『ESXCLI スタート ガイド』を参照してください。各リモート ホストの仕様で指定されたポート向けに ESXi のファイアウォールを開く方法の詳細については、『ESXi ファイアウォールの構成』を参照してください。

注： ESXCLI を使用するには、ESXi で SSH ログインを開く必要がありますが、これはセキュリティ上のリスクであり、推奨されません。ESXCLI を使用する場合は、各パラメータを設定した後に `esxcli system syslog reload` コマンドを使用して、有効になっていることを確認してください。

手順

- ◆ ESXCLI コマンド `esxcli system syslog config set <syslog option>` を使用して、有効にする Syslog オプションを設定します。たとえば、`Syslog.global.logHost` オプションを設定するには、`esxcli system syslog config set --loghost=<str>` コマンドを使用します。

`Syslog.global.logHost` を設定した後、ESXi ホストは Syslog コレクタとの接続を開いて維持し、メッセージ転送がただちに開始されます。ESXi は Syslog メッセージを生成するときに、ESXi ホスト上の適切なログ ファイルにメッセージを書き込み、構成されたすべての Syslog コレクタに転送します。

ESXi ホストでの Syslog の微調整

適切な Syslog 設定を使用することで、環境のプロアクティブな監視を実現し、ダウンタイムを短縮し、サーバに対して予防的なアクションを実行できます。

Syslog の設定時には、ログ ファイルの保持、Syslog の転送、転送の長さ、エラー処理、および Syslog メッセージを安全に転送するための SSL 証明書の設定に影響するいくつかのパラメータを考慮する必要があります。次に、Syslog パラメータを微調整するための推奨事項を示します。使用可能なすべてのパラメータの説明は、『ESXi Syslog のオプション』で確認できます。

ログ ファイルの保持を指定する方法

デフォルトでは、ログ ファイルは構成済みのサイズを越えて拡張できません。ログ ファイルが構成されたサイズに達すると、ログは新しいログ ファイルに切り替わり、最も古いログ ファイルが削除されます。

注： ベスト プラクティスは、このローテーションとサイズ設定のバランスを調整することです。ローテーションの設定値を引き上げると、Syslog ファイルは十分な頻度で生成されるようになり、他のログ ファイルが破損または破壊される可能性を抑えることができます。サイズの設定値を引き上げると、別のログ ファイルに切り替えるまでの時間が短縮されます。サイズの最適な設定値は 1,024 KiB の倍数です。

Syslog.global.defaultSize の設定を使用して、ログ ファイルの最大サイズを KiB 単位で指定し、Syslog.global.defaultRotate の設定を使用して、新しいログ ファイルにローテーションする前に保持する古いログ ファイルの最大数を設定します。特定のプログラムに関連付けられているログ ファイル保持パラメータを変更するには、Syslog.loggers.<progName>.rotate と Syslog.loggers.<progName>.size settings を使用します。ここで、<progName> はパラメータを調整するプログラムの名前です。

仮想マシンのログ ファイルに影響する設定の管理

vmx ファイルまたは /etc/vmware/config ファイルのいずれかで、仮想マシンのログ ファイルである vmware.log に影響するいくつかの設定を構成できます。vmx ファイルを編集するには、仮想マシンをパワーオフする必要があり、編集はその仮想マシンでのみ有効になります。/etc/vmware/config ファイルを使用する場合は、プリフィックス「vmx」を設定に追加する必要があります（例：vmx.log.keepOld = "20"）。編集は、ESXi ホスト上のすべての仮想マシンに影響します。

表 4-22. vmware.log ファイルの構成可能な設定

パラメータ	説明	例	メモ
logging	すべての仮想マシンのログ作成を無効にします。	デフォルト値は logging = "TRUE" です。 仮想マシンのログ作成を無効にするには、次の設定にします。 logging = "FALSE"	この設定は使用しないでください。仮想マシンのログ作成を無効にすると、仮想マシンの問題についてサポートを受けることが非常に困難または不可能になるためです。何らかの理由でこの設定を使用する必要がある場合は、仮想マシンの vmx ファイル内のみ配置できます。
log.throttleBytesPerSec	どのような場合にログ ファイルを調整するかを制御します。ログ ファイルの調整は、vmware.log への書き込みが長時間にわたって指定されたレートを超えると実行されます。このような状況が発生するのは、仮想マシンを制御する VMX プロセス内のコードにより、ログ メッセージが過剰に作成された場合です。この設定のデフォルト値は 1 KB/秒です。 ログの調整が実行された場合は、vmware.log ファイル内に <<< Log Throttled >>> というメッセージが含まれます。	log.throttleBytesPerSec = "1500" ログの調整を無効にするには、次の設定を使用します。 log.throttleBytesPerSec = "0xFFFFFFFF"	ログ ファイルの調整により、影響を受ける仮想マシンの問題を診断するために必要な情報が見つかりにくくなる場合があります。ログの調整を無効にする必要がある場合は、影響を受ける仮想マシンの vmx ファイル内に例の行を配置します。デバッグセッションの終了後に、その行を削除します。

表 4-22. vmware.log ファイルの構成可能な設定 (続き)

パラメータ	説明	例	メモ
log.keepOld	保持する古い vmware.log ファイルの数を制御します。	log.keepOld = "20"	この設定の値をデフォルト値 (10) 未満にしないでください。仮想マシンを頻繁に変更または移動する場合は、この設定を 20 以上に引き上げることを検討してください。
log.rotateSize	vmware.log ファイルの最大サイズをバイト単位で制御します。	log.rotateSize = "2500000" vmware.log ファイルの最大サイズの制限を無効にするには、次の設定を使用します。 log.rotateSize = "0"	この設定の値を 100,000 未満にすると、重要なログメッセージが失われ、仮想マシンのパフォーマンスに影響する可能性があります。ESXi 7.x 以前では、この設定のデフォルト値によって vmware.log ファイルのサイズが制限されることはありません。ESXi 8.x 以降では、この設定のデフォルト値は 2,048,000 です。
log.fileName	仮想マシンのログ ファイルの名前と場所を制御します。	log.fileName = "myVMLog" この設定により、仮想マシンのログ ファイルの名前が vmware.log から myVMlog に変更されます。 log.fileName = "/vmfs/volumes/vol1/myVM/myVM.log" この設定により、仮想マシンのログ ファイルが、ファイル名に myVM を使用して、別の VMFS ボリューム (vol1) 上のディレクトリに転送されます。	ホスト サポート バンドルの収集によってログ ファイルが確実に収集されるようにするため、ログ ファイルを仮想マシンのディレクトリ外には配置しないでください。仮想マシンの問題をデバッグするうえで、このログ ファイルが重要になる可能性があります。

表 4-22. vmware.log ファイルの構成可能な設定 (続き)

パラメータ	説明	例	メモ
log.fileLevel	<p>メッセージが vmware.log に書き込まれる最低レベルを制御します。すべてのログメッセージにはレベルが関連付けられています。指定された設定より下のレベルのメッセージは、仮想マシンのログファイルには追加されません。仮想マシンのメッセージ ログレベル (制限が最大のものから最小のものまで) は以下のとおりです。</p> <ul style="list-style-type: none"> ■ エラー ■ 警告 ■ 注意 ■ 情報 (デフォルト) ■ 最詳細 ■ デバッグ ■ debug1 ■ debug2 ■ debug3 ■ debug4 ■ debug5 ■ debug6 ■ debug7 ■ debug8 ■ debug9 ■ debug10 	log.fileLevel = "debug1"	仮想マシンの問題をデバッグするために必要なメッセージが除外されるのを防ぐため、「情報」よりも制限が厳しいレベルには設定しないでください。認定済みのサポートから要求された場合にのみ、レベルを「情報」未満に下げます。デバッグの終了後に、この設定を「情報」に戻します。
log.filter.minLogLevel.<groupName>	特殊なデバッグメッセージの出力を制御します。	log.filter.minLogLevel.debug.isklib = "debug5"	認定済みのサポートから要求された場合にのみ、この設定を使用します。サポートから1つ以上の<groupName>パラメータが指定されます。デバッグの終了後に、この設定を削除します。
log.syslogID	Syslog などの ESXi ホストのシステム ロガーに対する仮想マシンのログメッセージの送信を有効にします。	log.syslogID = "vmx"	この設定の値として "vmx" を使用すると、ESXi Syslog デモンの vmmsyslogd から、これらのメッセージを別のログファイルに送信できます。
log.syslogLevel	Syslog などの ESXi ホストのシステム ロガーにメッセージを出力する最低レベルを制御します。	log.syslogLevel = "debug"	この設定のレベルと機能は、log.fileLevel 設定のものと同じです。

リモート ホストへのメッセージ転送を指定する方法

必要に応じて、Syslog メッセージを 1 つ以上のリモート ホストに送信して Syslog メッセージを収集するように ESXi を構成できます。これらのリモート ホストは Syslog コレクタと呼ばれ、VMware vRealize Log Insight (旧称は vCenter Log Insight) などが該当します。

注： ベスト プラクティスは、Syslog メッセージを 1 台以上の Syslog コレクタに送信するように各 ESXi ホストを構成することです。これにより、致命的なシステム イベントが発生した場合にメッセージを確実に保存し、Syslog メッセージをリアルタイムの分類や分析 (タイプ、時間範囲、マシン別など) やアーカイブ メッセージなど、さまざまな方法で処理できます。

Syslog.global.logHost 設定を使用して、リモート ホストの仕様を定義します。複数のリモート ホスト仕様はカンマ (,) で区切ります。Syslog.global.logHost を設定した後、ESXi ホストは Syslog コレクタとの接続を開いて維持し、メッセージ転送がただちに開始されます。ESXi は Syslog メッセージを生成するときに、ESXi ホスト上の適切なログ ファイルにメッセージを書き込み、構成されたすべての Syslog コレクタに転送します。

Syslog メッセージに加えて、監査メッセージをセキュリティ上の目的で Syslog コレクタに送信することもできます。監査レコードは、ESXi ホスト上のセキュリティ関連のアクティビティを追跡します。監査レコードの詳細については、「[監査レコード](#)」を参照してください。

注： 監査レコードを設定する必要があるかどうか、およびその方法については、会社のセキュリティ対策チームに確認してください。認証された構成では、通常、監査レコードを有効にする必要があります。

次に、Syslog.global.logHost リモート ホスト仕様の構文を示します。

```
protocol://target[:port]
```

パラメータ	説明	メモ
protocol	ネットワーク プロトコルを指定します。有効な値は、udp、tcp、および ssl です。	ssl プロトコルでは、Syslog メッセージの送信を暗号化することが指定されています。tcp および udp プロトコルでは、転送は暗号化されません。 注： システムで Syslog メッセージまたは監査メッセージのキャプチャが重要である場合は、udp プロトコルを使用しないでください。ESXi 外部のネットワーク インフラストラクチャでは、UDP メッセージがドロップされる可能性があります。
target	リモート ホストを指定します。IPv4 または Ipv6 アドレスのいずれか、またはホスト名を使用できます。	IPv6 アドレスを使用する場合は、[xxx] のように角括弧で囲む必要があります。xxx は IPv6 アドレスです。
port	(オプション) 使用するリモート ホスト ポートを指定します。UDP または TCP を使用する場合、デフォルト ポートは 1514 です。SSL を使用する場合、デフォルト ポートは 514 です。514 または 1514 以外のポートを使用する場合は、ESXi ファイアウォールを調整してポートを開く必要があります。	各リモート ホストの仕様で指定されたポート向けに ESXi のファイアウォールを開く方法の詳細については、「 ESXi ファイアウォールの構成 」を参照してください。

リモートマシン仕様のサンプル:

Syslog.global.logHost 文字列の例	メモ
tcp://10.176.130.7:12345	TCP/IP およびポート 12345 を使用して、Syslog メッセージを 10.176.130.7 に転送します。
tcp://[2001:db8:85a3:8d3:1319:8a2e:370:7348]	ポート 1514 を使用して、Syslog メッセージを IPv6 アドレスに転送します。
tcp://[2001:db8:85a3:8d3:1319:8a2e:370:7348]:5432	ポート 54321 を使用して、Syslog メッセージを IPv6 アドレスに転送します。
udp://company.com	UDP およびポート 514 を使用して、Syslog メッセージを company.com に転送します。
udp://company.com,tcp://10.20.30.40:1050	Syslog メッセージを 2 台のリモート ホストに転送します。最初のリモート ホストは、UDP を使用して、ポート 514 による company.com との通信を行います。2 番目のリモート ホストは TCP を使用して、ポート 1050 による IPV4 アドレス 10.20.30.40 との通信を行います。
ssl://company.com	SSL (TLS) およびポート 514 を使用して、Syslog メッセージを company.com に転送します。

メッセージ転送の最大長

UDP を使用する場合は Syslog メッセージ転送の最大長は、IPv4 で 480 バイト、IPv6 で 1,180 バイトです。

TCP または SSL の場合、Syslog メッセージ転送のデフォルトの最大長は 1 キビバイト (KiB) です。この長さは、Syslog.global.remoteHost.maxMsgLen パラメータを使用して増やすことができます。最大値は 16 KiB です。16 KiB を超えるメッセージは切り詰められます。

注： 最大転送長を増やす必要がある場合は、具体的に必要な分の長さを増やすことがベスト プラクティスになります。

Syslog メッセージの最大長を増やすと、ESXi 外部のネットワークおよび Syslog インフラストラクチャが 1 KiB を超えるメッセージを処理できない場合、問題が発生する可能性があります。

注： UDP はパケット長に制約があり、外部のネットワーク インフラストラクチャによってメッセージがドロップされる可能性があるため、Syslog メッセージの送信に UDP を使用しないことがベスト プラクティスです。

リモート ホストへの SSL 転送を構成する際の証明書に関する考慮事項

SSL を使用して Syslog メッセージをリモート ホストに転送するように ESXi を構成する場合は、各リモート ホストの SSL 証明書を ESXi ホストの CA ストアに追加する必要があります。詳細については、「[ESXi ホストの証明書管理](#)」および「[ESXCLI を使用した CA 証明書の管理](#)」を参照してください。

注： SSL とプライベート キーを使用して Syslog メッセージを安全に受信できるようにコレクタを構成する方法については、Syslog コレクタのドキュメントを参照してください。

その他の SSL 転送パラメータ

セキュリティ認証の要件に準拠した ESXi システムでは、X509 CRL チェックの有効化が必要になる場合があります。詳細設定の `Syslog.global.certificate.strictX509Compliance` と `Syslog.global.certificate.checkCRL` を有効にするには、デフォルト値の `false` を `true` に変更します。実装上の制限により、`Syslog.global.certificate.checkSSLCerts` の設定を使用して CRL チェックを有効にしている場合は、証明書チェーン内のすべての証明書で CRL リンクを提供する必要があります。デフォルトでは、この設定は有効になっています。この設定を `false` に変更することで、SSL 証明書のチェックを無効にできますが、この方法はベスト プラクティスではありません。リモート ホストとの通信のトラブルシューティングを行う場合は、SSL 証明書チェックの無効化が必要になる場合がありますが、オフにする期間は制限するようにしてください。

Syslog デーモンのエラーとステータスの情報を確認できる場所

ESXi の Syslog デーモンは、ステータスとエラー情報を、ドロップされたメッセージも含めてログ ファイル `/var/run/log/vmsyslogd.log` に保存します。監査レコードの転送が有効になっている場合、Syslog デーモンは、デーモンの開始、停止、エラー状態など、その操作に関連する監査レコードも出力します。これにより、Syslog デーモンが適切に実行されていることを確認できます。

デフォルトの Syslog ログ ファイル ストレージ領域を変更する方法

デフォルトの Syslog ログ ファイル ストレージ領域は、各 ESXi ホストのローカルの `/var/run/log` です。デフォルトの Syslog ログ ファイル ストレージ領域は、Syslog 構成変数 `Syslog.global.logDir` を使用することにより、パーシステント ストレージ上であれば任意の場所に変更できます。`Syslog.global.logDir` が、Syslog ログ ファイルを保存するために複数の ESXi ホストで共有されているパーシステント ストアに構成されている場合は、ログの混在を防ぐために `Syslog.global.logDirUnique` 設定を `true` に変更します。`Syslog.global.logDirUnique` を設定すると、各 ESXi マシンは `Syslog.global.logDir` パスに追加された一意の名前を取得し、ログ ファイルを他のホストから区別するようになります。

リモート ホストおよびメッセージ ドロップ用の Syslog メッセージ キュー

Syslog の排出量が開始されると、ESXi の再起動と障害、または停止する Syslog 再構成を除いて停止することはありません。

ESXi は、Syslog メッセージと監査レコードをリモート ホストに転送するために、容量が制限されたメモリベースのキュー メカニズムを使用します。このメカニズムにより、`vmsyslogd` サービスは大量の短いメッセージを処理でき、ネットワーク接続の問題をデータのドロップなしで迅速に解決できます。

接続の問題が長引くか、受信データのフローがキュー メカニズムで許容できる制限を継続的に超えると、`vmsyslogd` サービスは一部の Syslog メッセージをドロップします。

このようなドロップを最小限に抑えるには、`vmsyslogd` ログ ファイルを使用可能な最高速のストレージに配置し、エンドツーエンドのバンド幅が 1 GigE 以上のネットワーク上に Syslog コレクタを構成します。

アプリケーションからログが過剰に生成される場合は、分析と修正のためにサポート サービス リクエストを発行します。

メッセージのドロップに関する統計情報は、Syslog デーモン ログ ファイルに記録されます。

ドロップされたメッセージは、`/var/run/log/vmsyslogd-dropped.log` に記録されます。このログ ファイルには、プログラム固有の保持パラメータと同様に、固有の保持設定があります。ドロップされたメッセージのログ ファイルの保持パラメータは、`Syslog.global.droppedMsgs.fileRotate` と `Syslog.global.droppedMsgs.fileSize` です。

ESXi Syslog のオプション

一連の Syslog オプションを使用して、ESXi Syslog ファイルと転送の動作を定義することができます。

ESXi 7.0 Update 1 以降では、`Syslog.global.logHost` などの基本設定とは別に、カスタマイズおよび NIAP コンプライアンス用の一連の詳細オプションを使用できます。

注： 監査レコード パラメータまたは `Syslog.global.logDir` パラメータのいずれかを設定する前には、常にパーシステント ストレージを構成してください。

注： `Syslog.global.auditRecord` 以降のすべての監査レコード設定がすぐに有効になります。ただし、ESXCLI を使用して定義された他の設定については、`esxcli system syslog reload` コマンドを実行して変更を有効にするようにしてください。

表 4-23. レガシー Syslog オプション

オプション	ESXCLI コマンド	説明
<code>Syslog.global.logHost</code>	<code>esxcli system syslog config set --loghost=<str></code>	リモート ホストのカンマ区切りリストとメッセージ転送の仕様を定義します。 <code>loghost=<str></code> フィールドが空白の場合、ログは転送されません。Syslog メッセージを受信するリモート ホストの数にハード制限はありませんが、リモート ホストの数は 5 台以下にすることを推奨します。リモート ホストの仕様の形式は、 <code>protocol://hostname ipv4 ['ipv6'][:port]</code> です。プロトコルは、TCP、UDP、または SSL のいずれかにする必要があります。ポートの値には、1 ~ 65,535 の任意の 10 進数を指定できます。ポートが指定されていない場合、SSL と TCP では 1514 が使用されます。UDP では 514 が使用されます。たとえば、 <code>ssl://hostName1:1514</code> です。
<code>Syslog.global.defaultRotate</code>	<code>esxcli system syslog config set --default-rotate=<long></code>	古いログ ファイルの最大保持数。この数字はグローバルに、また個別のサブロガーについて設定できます (<code>Syslog.global.defaultSize</code> を参照してください)。
<code>Syslog.global.defaultSize</code>	<code>esxcli system syslog config set --default-size=<long></code>	ログ ファイルのデフォルト サイズ (KiB 単位)。ファイルがデフォルト サイズに達すると、Syslog サービスによって新しいファイルが作成されます。この数字はグローバルに、また個別のサブロガーについて設定できます。

表 4-23. レガシー Syslog オプション (続き)

オプション	ESXCLI コマンド	説明
Syslog.global.logDir	esxcli system syslog config set --logdir=<str>	ログが配置されたディレクトリ。ディレクトリは、マウントされた NFS または VMFS ポリ्यूームに配置できます。リブート後も変わらないのは、ローカル ファイル システムの /scratch ディレクトリのみです。ディレクトリを [datastorename] path_to_file と指定します。ここで、パスはデータストアをバックアップするポリ्यूームのルートへの相対パスです。例えば、パスの [storage1] /systemlogs はパスの /vmfs/volumes/storage1/systemlogs にマッピングします。
Syslog.global.logDirUnique	esxcli system syslog config set --logdir-unique=<bool>	Syslog.global.logDir の値に連結する ESXi ホスト名を指定します。複数の ESXi ホストのログが共有ファイル システムに記録される場合は、この設定を有効にすることが重要です。このオプションを選択すると、ESXi ホストの名前を持つサブディレクトリを [Syslog.global.LogDir] で指定されるディレクトリの下に作成します。同一の NFS ディレクトリが複数の ESXi ホストによって使用される場合、固有のディレクトリを作成しておくくと便利です。
Syslog.global.certificate.checkSSLCerts	esxcli system syslog config set --check-ssl-certs=<bool>	リモート ホストへのメッセージの転送時に SSL 証明書の確認を実施します。

表 4-24. ESXi 7.0 Update 1 以降で使用可能な Syslog オプション

オプション	ESXCLI コマンド	説明
Syslog.global.auditRecord.storageCapacity	esxcli system auditrecords local set --size=<long>	ESXi ホストにある監査レコード ストレージ ディレクトリの容量を MiB 単位で指定します。監査レコード ストレージの容量を減らすことはできません。監査レコード ストレージが有効になる前または後に、ストレージの容量を増やすことが実行できます (Syslog.global.auditRecord.storageEnable 参照)。
Syslog.global.auditRecord.remoteEnable	esxcli system auditrecords remote enable	リモート ホストへの監査レコードの転送を有効にします。リモート ホストは、Syslog.global.logHost パラメータを使用して指定します。

表 4-24. ESXi 7.0 Update 1 以降で使用可能な Syslog オプション (続き)

オプション	ESXCLI コマンド	説明
Syslog.global.auditRecord.storageDirectory	esxcli system auditrecords local set --directory=<dir>	監査レコード ストレージ ディレクトリを作成します。指定がなければ /scratch/auditLog がデフォルトの場所として設定されます。監査レコード ストレージ ディレクトリは手動で作成しないでください。また、監査レコード ストレージが有効になっている間は、監査レコード ストレージ ディレクトリを変更できません (Syslog.global.auditRecord.storageEnable を参照)。
Syslog.global.auditRecord.storageEnable	esxcli system auditrecords local enable	ESXi ホストで監査レコードのストレージを有効にします。監査レコード ストレージ ディレクトリが存在しない場合は、Syslog.global.auditRecord.storageCapacity で指定された容量で作成されます。
Syslog.global.certificate.checkCRL	esxcli system syslog config set --crl-check=<bool>	SSL 証明書チェーン内のすべての証明書の失効ステータスの確認を有効にします。 X.509 CRL の検証を有効にします。この CRL 検証は、業界の規則に従ってデフォルトでチェックされることはありません。NIAP で検証された構成の場合は、CRL チェックが必要です。実装上の制限により、CRL チェックが有効になっている場合は、証明書チェーン内のすべての証明書が CRL リンクを提供する必要があります。 CRL チェックを使用する環境を適切に構成するのは困難なため、証明に関連しないインストール環境の場合は、crl-check オプションを有効にしないでください。
Syslog.global.certificate.strictX509Compliance	esxcli system syslog config set --x509-strict=<bool>	X.509 への厳密なコンプライアンスを有効にします。検証中に CA ルート証明書に対して追加の妥当性チェックを実行します。これらのチェックは一般に実行されません。CA ルートは本来信頼されるものであり、構成に誤りのある既存の CA ルートとの互換性が失われる可能性があるためです。NIAP で検証された構成の場合は、CA ルートも検証に合格する必要があります。 CRL チェックを使用する環境を適切に構成するのは困難なため、証明に関連しないインストール環境の場合は、x509-strict オプションを有効にしないでください。
Syslog.global.droppedMsgs.fileRotate	esxcli system syslog config set --drop-log-rotate=<long>	ドロップされた古いメッセージ ログ ファイルの保持数を指定します。
Syslog.global.droppedMsgs.fileSize	esxcli system syslog config set --drop-log-size=<long>	ドロップされた各メッセージ ログ ファイルが新しいログ ファイルに切り替わるサイズ (KiB 単位) を指定します。

表 4-24. ESXi 7.0 Update 1 以降で使用可能な Syslog オプション (続き)

オプション	ESXCLI コマンド	説明
Syslog.global.logCheckSSLCerts	esxcli system syslog config set --check-ssl-certs=<bool>	リモート ホストへのメッセージの転送時に SSL 証明書の確認を実施します。 注: 廃止されました。ESXi 7.0 Update 1 以降では Syslog.global.certificate.checkSSLCerts を使用します。
Syslog.global.logFilters	esxcli system syslog config logfilter [add remove set] ...	1つ以上のログ フィルタリング仕様を指定します。各ログ フィルタは、二重の縦棒「 」で区切る必要があります。ログ フィルタの形式は、numLogs ident logRegexp です。numLogs では、指定したログ メッセージの最大ログ エントリ数を設定します。この数に達すると、指定したログ メッセージがフィルタリングされて無視されます。ident では、1つ以上のシステム コンポーネントを指定し、これらのコンポーネントで生成されるログ メッセージにフィルタを適用します。logRegexp では、Python 正規表現構文を使用して大文字と小文字を区別する語句を指定し、ログ メッセージを内容でフィルタリングします。
Syslog.global.logFiltersEnable		ログ フィルタの使用を有効にします。
Syslog.global.logLevel	esxcli system syslog config set --log-level=<str>	ログ フィルタリング レベルを指定します。このパラメータを変更する必要があるのは、Syslog デーモンの問題をトラブルシューティングする場合のみです。debug 値は最も詳細なレベル、info 値はデフォルトの詳細レベル、warning 値は警告またはエラーの場合のみ、error 値はエラーの場合のみ使用できます。
Syslog.global.msgQueueDropMark	esxcli system syslog config --queue-drop-mark=<long>	メッセージのドロップが開始されるメッセージ キューの容量の割合を指定します。
Syslog.global.remoteHost.connectRetryDelay	esxcli system syslog config set --default-timeout=<long>	接続の試行が失敗してからリモート ホストへの接続を再試行するまでの遅延時間を秒単位で指定します。

表 4-24. ESXi 7.0 Update 1 以降で使用可能な Syslog オプション (続き)

オプション	ESXCLI コマンド	説明
Syslog.global.remoteHost.maxMsgLen	esxcli system syslog config set --remote-host-max-msg-len=<long>	TCP および SSL プロトコルの場合、このパラメータは、切り詰める前の Syslog 転送の最大長をバイト単位で指定します。リモート ホスト メッセージのデフォルトの最大長は 1 KiB です。メッセージの最大長を最大 16 KiB まで増やすことができます。ただし、この値を 1 KiB より大きい値にしても、長い転送が切り詰められないで Syslog コレクタに到着するとは限りません。たとえば、メッセージを発行する Syslog インフラストラクチャが ESXi の外部にある場合などです。 この設定は、UDP プロトコルには影響しません。RFC 5426 は、UDP プロトコルの最大メッセージ転送長を、IPV4 の場合は 480 バイト、IPV6 の場合は 1,180 バイトに設定します。この制限のため、また、UDP パケットはネットワーク インフラストラクチャによって予期せずドロップされる可能性があるため、重要な Syslog メッセージの転送に UDP を使用することは推奨されません。
Syslog.global.vsanBacking	esxcli system syslog config set --vsan-backing=<bool>	ログ ファイルと監査レコード ストレージ ディレクトリを vSAN クラスタに配置できるようにします。ただし、このパラメータを有効にすると、ESXi ホストが応答しなくなる可能性があります。

ESXi Syslog メッセージのプロトコル、フォーマット、フレーミング

ESXi 8.0 以降の Syslog サービスでは 3 つのパラメータを使用してメッセージと監査レコード（プロトコル、フォーマット、フレーミング）を定義します。

サポートされているプロトコルは、UDP、TCP、および TLS (SSL) です。Syslog メッセージのフォーマットは、RFC 3164 または RFC 5424 のいずれかで定義されます。フレーミングは、メッセージをカプセル化する方法を指定します。カプセル化されたメッセージのフレーミングは透過的 (octet_counting と呼ばれます) と定義され、メッセージがカプセル化されていない場合は非透過的と定義されます。透過的なフレーミングでは、メッセージに埋め込まれた改行によって Syslog コレクタで混乱が発生することがありません。UDP プロトコルを使用して送信された Syslog メッセージは、透過的にフレーミングされたと見なされます。Syslog コレクタは、これを把握して送信内容を単一のメッセージとして受け入れます。

RFC 3164 では Syslog メッセージの最大合計長が 1,024 バイトに設定されていますが、RFC 5424 ではこの最大長が 2,048 バイトに拡張されています。

ESXi では、リモート ホスト メッセージのデフォルトの最大長は 1 KiB です。メッセージの最大長を最大 16 KiB まで増やすことができます。ただし、この値を 1 KiB より大きい値にしても、長い転送が切り詰められないで Syslog コレクタに到着するとは限りません。たとえば、メッセージを発行する Syslog インフラストラクチャが ESXi の外部にある場合などです。

vmsyslogd が送信する Syslog メッセージは、RFC 5424 に準拠したフォーマットのプロパティ リストである構造化データと、自由形式の非構造化データで構成されます。

メッセージが最大長よりも長い場合、ESXi 8.0 は可能なかぎり多くの構造化データを保持しながら、メッセージを削減します。

メッセージが削減されると、既存の構造化データに 3 つのパラメータが追加されるか、構造化データが msgModified、remoteHostMaxMsgLen、originalLen の各パラメータを含むように作成されます。

msgModified パラメータは、削減がメッセージにどのように影響するかを示します（構造化データのみ、非構造化データのみ、またはその両方）。

remoteHostMaxMsgLen パラメータは、ESXi が処理できる最大メッセージ長を指定します。

originalLen パラメータは、削減される前のメッセージの長さを指定します。

ESXi Syslog メッセージのプロトコル、フォーマット、フレーミングでサポートされるオプション。

フォーマット	フレーミング	UDP	TCP	SSL	コメント
未指定	未指定	サポート RFC 5426	サポート	サポート	メッセージのフォーマットは RFC 3164 に準拠しており、タイムスタンプのみが RFC 3339 形式です。 構造化データが各メッセージの先頭に付加されます。 フレーミングのデフォルトは、TCP または SSL (TLS) では非透過的であり、構造化データに改行を埋め込むとメッセージが破損する可能性があります。 UDP では、パケットはフレーミングされます。
未指定	Non_transparent	禁止	サポート	サポート	メッセージのフォーマットは RFC 3164 に準拠しており、タイムスタンプのみが RFC 3339 形式です。 構造化データが各メッセージの先頭に付加されます。 フレーミングのデフォルトは、TCP または SSL (TLS) では非透過的であり、構造化データに改行を埋め込むとメッセージが破損する可能性があります。
未指定	Octet_counting	禁止	サポート RFC 6587	サポート RFC 6587	メッセージのフォーマットは RFC 3164 に準拠しており、タイムスタンプのみが RFC 3339 形式です。 構造化データが各メッセージの先頭に付加されます。

フォーマット	フレーミング	UDP	TCP	SSL	コメント
RFC 5424	未指定	サポート RFC 5426	サポート RFC 5425	サポート RFC 5424	メッセージのフォーマットは RFC 5424 に準拠しています。 フレーミングのデフォルトは、TCP または SSL (TLS) ではオクテットカウントです。UDP では、フレーミングが明示的に指定されない場合があります。
RFC 5424	Non_transparent	禁止	サポート対象外	サポート対象外	構造化データへの改行の組み込みは、メッセージの破損が発生する可能性があるためサポートされません。
RFC 5424	Octet_counting	禁止	サポート RFC 5425	サポート RFC 5425	メッセージのフォーマットは RFC 5424 に準拠しています。
RFC 3164	未指定	サポート RFC 5426	サポート	サポート	メッセージのフォーマットは RFC 3164 に準拠しており、タイムスタンプのみが RFC 3339 形式です。 構造化データが各メッセージの先頭に付加されます。 フレーミングのデフォルトは、TCP または SSL (TLS) では非透過的であり、構造化データに改行を埋め込むとメッセージが破損する可能性があります。 UDP では、パケットはフレーミングされます。
RFC 3164	Non_transparent	禁止	サポート	サポート	メッセージのフォーマットは RFC 3164 に準拠しており、タイムスタンプのみが RFC 3339 形式です。 構造化データが各メッセージの先頭に付加されます。 フレーミングのデフォルトは、TCP または SSL (TLS) では非透過的であり、構造化データに改行を埋め込むとメッセージが破損する可能性があります。
RFC 3164	Octet_counting	禁止	サポート RFC 6587	サポート RFC 6587	メッセージのフォーマットは RFC 3164 に準拠しており、タイムスタンプのみが RFC 3339 形式です。 構造化データが各メッセージの先頭に付加されます。

ログ ファイルの形式

ESXi 8.0 以降では、ログ ファイルの形式が標準化され、Augmented Backus-Naur Form (ABNF) で表されず。

ESXi 8.0 でのログ ファイルへの書き込みは、VMX などの単一サービスから直接、またはサービスからのログが Syslog に送信されるときに間接的に行われます。たとえば、VMX は、各仮想マシンの `vmware.log` ファイルに常時ログ メッセージを書き込みます。システム リソースを節約するために、VMX は Syslog にログ メッセージを送信しません。一方、`vmsyslogd` によって生成されるログ ファイルの一部には、複数のプログラムからのメッセージが表示されます。これは、ESXi Syslog デーモンがすべてのログ ファイルを作成、管理しており、ログ ファイルには複数のサービスからのメッセージが書き込まれるためです。

直接ログ メッセージの形式：

パラメータ	値
LOG-MSG	HEADER SP MSG
HEADER	TIMESTAMP SP SEVERITY SP THREAD-NAME SP OPID
TIMESTAMP	FULL-DATE T FULL-TIME (RFC 5424 に準拠、UTC/GMT 形式の要件に従い、ミリ秒または可能であればそれ以上の精度)
FULL-DATE	DATE-FULLYEAR - DATE-MONTH - DATE-MDAY
DATE-FULLYEAR	4DIGIT
DATE-MONTH	2DIGIT ; 01-12
DATE-MDAY	2DIGIT ; 01-28, 01-29, 01-30, 01-31 (月/年に応じて)
FULL-TIME	TIME-HOUR : TIME-MINUTE : TIME-SECOND[TIME-SECFRAC] Z
TIME-HOUR	2DIGIT ; 00-23
TIME-MINUTE	2DIGIT ; 00-59
TIME-SECOND	2DIGIT ; 00-59
TIME-SECFRAC	!*6DIGIT
SEVERITY	SEVERITY-STRING SEVERITY-VALUE [LINE-MARKER]
SEVERITY-STRING	Em/Al/Cr/Er/Wa/No/In/Db (RFC 5424 で指定されている 8 つの重要度レベルは、次の略語で示されます。 <ul style="list-style-type: none"> ■ Em - 緊急 ■ Al - アラート ■ Cr - 重大 ■ Er - エラー ■ Wa - 警告 ■ No - 注意 ■ In - 情報 ■ Db - デバッグ
SEVERITY-VALUE	(*DIGIT) (SEVERITY-VALUE は、SEVERITY-STRING に関連付けられるオプションの数値です。これにより、情報を損なうことなく、ロガーがサポートするレベルを 8 つの必須文字列に短縮できます (例：レベル 5 のデバッグを表す Db(5) など。)
LINE-MARKER	+ (複数行の送信で生成される後続のすべての行には、LINE-MARKER が追加されます。これによって複数行の送信を識別し、ログ インジェクション セキュリティ攻撃を防止します。)
NILVALUE	- (単スレッド プログラムにはスレッド名が含まれていない場合があり、NILVALUE を使用できます。)

THREAD-NAME	NILVALUE / 1*32PRINTUSASCII (単一のプログラムがファイルを書き込むとき、コンポーネント (APP-NAME) は暗黙で指定され、コンポーネント フィールドは不要になり、スレッド名のみが必要です。)
OPID	NILVALUE / 1*128UTF-8-STRING
STRUCTURED-DATA	1*SD-ELEMENT
SD-ELEMENT	[SD-ID *(SP SD-PARAM)]
SD-PARAM	PARAM-NAME %d34 PARAM-VALUE %d34
SD-ID	SD-NAME
PARAM-NAME	SD-NAME
PARAM-VALUE	UTF-8-STRING; ", '\', ']' の文字はエスケープする必要があります。
SD-NAME	1*32PRINTUSASCII ; ", SP, ']', %d34 () を除く
MSG	[STRUCTURED-DATA SP] UTF-8-STRING

vmsyslogd サービスが管理するログ ファイルの形式。

パラメータ	値
LOG-MSG	HEADER SP MSG
HEADER	TIMESTAMP SP SEVERITY SP APP-NAME [PROC-IDENTIFIER] :
APP-NAME	1*32PRINTUSASCII
PROC-IDENTIFIER	[*DIGITS] ; APP-NAME に関連付けられた PID
TIMESTAMP	FULL-DATE T FULL TIME (ミリ秒または可能であればそれ以上の精度)
FULL-DATE	DATE-FULLYEAR - DATE-MONTH - DATE-MDAY
DATE-FULLYEAR	4DIGIT
DATE-MONTH	2DIGIT ; 01-12
DATE-MDAY	2DIGIT ; 01-28, 01-29, 01-30, 01-31 (月/年に応じて)
FULL-TIME	TIME-HOUR : TIME-MINUTE : TIME-SECOND[TIME-SECFRAC] Z
TIME-HOUR	2DIGIT ; 00-23
TIME-MINUTE	2DIGIT ; 00-59
TIME-SECOND	2DIGIT ; 00-59
TIME-SECFRAC	!.1*6DIGIT

SEVERITY-STRING	Em/Al/Cr/Er/Wa/No/In/Db (RFC 5424 で指定されている 8 つの重要度レベルは、次の略語で示されます。 <ul style="list-style-type: none"> ■ Em - 緊急 ■ Al - アラート ■ Cr - 重大 ■ Er - エラー ■ Wa - 警告 ■ No - 注意 ■ In - 情報 ■ Db - デバッグ
-----------------	---

SEVERITY	SEVERITY-STRING PRI-STRING [LINE-MARKER]
PRIVAL	1*3DIGIT ; range 0 ..191 (MSG PRI; 機能と重要度の値の OR 演算の結果が含まれます)
PRI-STRING	(PRIVAL) PRIVAL には、メッセージ PRI のビットが含まれています。これにより、メッセージの機能と重要度ビット自体を確認できます。 _
LINE-MARKER	+ (複数行の送信で生成される後続のすべての行には、LINE-MARKER が追加されます。これによって複数行の送信を識別し、ログ インジェクション セキュリティ攻撃を防止します。)
STRUCTURED-DATA	1*SD-ELEMENT
SD-ELEMENT	[SD-ID *(SP SD-PARAM)]
SD-PARAM	PARAM-NAME %d34 PARAM-VALUE %d34
SD-ID	SD-NAME
PARAM-NAME	SD-NAME
PARAM-VALUE	UTF-8-STRING; 文字 ", '\, ']' はエスケープする必要があります。
SD-NAME	1*32PRINTUSASCII ; ", SP, ']', %d34 () を除く
MSG	[STRUCTURED-DATA SP] UTF-8-STRING

監査レコード

ESXi 監査レコードは、機能コード 13 とともに RFC 3164 と 5424 の両方の形式に準拠しており、構造化データセクションに表示されます。監査レコードには、イベントベースのトレーサビリティ情報が使用可能な場合は、そのデータも表示されます。監査レコードは、通常のログ ファイルではなく特殊な形式で保存されます。viewAudit プログラムと仮想インフラストラクチャ管理機能 FetchAuditRecords を使用すると、ローカルで監査レコードにアクセスできます。監査レコード ストレージ ファイルを直接読み取り、使用、または編集しないでください。ローカルに格納された監査レコードは、HOSTNAME と MSGID が常に NILVALUE である RFC 5424 転送形式に準拠しています。

ESXi Syslog メッセージ転送形式

ESXi 8.0 では、Syslog メッセージが RFC 3164 または RFC 5424 に準拠してフォーマットされます。

RFC 3164 および RFC 5424 の転送形式の定義は、Augmented Backus-Naur Form (ABNF) です。

RFC 3164 転送メッセージ形式

パラメータ	値
SYSLOG-MSG	HEADER SP MSG
HEADER	PRI TIMESTAMP SP HOSTNAME SP APP-NAME [PROC-IDENTIFIER] ":"
PRI	"<" PRIVAL ">"
PRIVAL	1*3DIGIT ; range 0 ..191 (MSG PRI; 機能と重要度の値の OR 演算の結果が含まれます)
APP-NAME	1*32PRINTUSASCII
HOSTNAME	1*255PRINTUSASCII
PROC-IDENTIFIER	"[" *DIGITS "]" ; APP-NAME と関連付けられている PID
TIMESTAMP	FULL-DATE "T" UTC-TIME (TIME-OFFSET は設定しないでください)
FULL-DATE	DATE-FULLYEAR "-" DATE-MONTH "-" DATE-MDAY
DATE-FULLYEAR	4DIGIT
DATE-MONTH	2DIGIT ; 01-12
DATE-MDAY	2DIGIT ; 01-28, 01-29, 01-30, 01-31 (月/年に応じて)
UTC-TIME	TIME-HOUR ":" TIME-MINUTE ":" TIME-SECOND [TIME-SECFRAC] "Z"
TIME-HOUR	2DIGIT ; 00-23
TIME-MINUTE	2DIGIT ; 00-59
TIME-SECOND	2DIGIT ; 00-59
TIME-SECFRAC	"."1*6DIGIT
STRUCTURED-DATA	1*SD-ELEMENT
SD-ELEMENT	"[" SD-ID *(SP SD-PARAM) "]"
SD-PARAM	PARAM-NAME "%d34 PARAM-VALUE %d34
SD-ID	SD-NAME
PARAM-NAME	SD-NAME
PARAM-VALUE	UTF-8-STRING ; '"', '\', および ']' の文字はエスケープする必要があります。
SD-NAME	1*32PRINTUSASCII ; ('、SP、']、%d34 (" を除く
MSG	[STRUCTURED-DATA SP] UTF-8-STRING

RFC 5424 転送メッセージ形式

パラメータ	値
SYSLOG-MSG	HEADER SP STRUCTURED-DATA [SP MSG]

HEADER	PRI VERSION SP TIMESTAMP SP HOSTNAME SP APP-NAME SP PROCID SP MSGID
PRI	"<" PRIVAL ">"
PRIVAL	1*3DIGIT ; range 0 ..191; には機能および重要度データが含まれます
VERSION	NONZERO-DIGIT 0*2DIGIT
HOSTNAME	1*255PRINTUSASCII
APP-NAME	1*48PRINTUSASCII
NILVALUE	'.'
PROCID	NILVALUE *DIGITS ; APP-NAME と関連付けられている PID
MSGID	NILVALUE
TIMESTAMP	FULL-DATE "T" UTC-TIME
FULL-DATE	DATE-FULLYEAR "-" DATE-MONTH "-" DATE-MDAY
DATE-FULLYEAR	4DIGIT
DATE-MONTH	2DIGIT ; 01-12
DATE-MDAY	2DIGIT ; 01-28, 01-29, 01-30, 01-31 (月/年に応じて)
UTC-TIME	TIME-HOUR ":" TIME-MINUTE ":" TIME-SECOND [TIME-SECFRAC] "Z"
TIME-HOUR	2DIGIT ; 00-23
TIME-MINUTE	2DIGIT ; 00-59
TIME-SECOND	2DIGIT ; 00-59
TIME-SECFRAC	"."1*6DIGIT
STRUCTURED-DATA	NILVALUE / 1*SD-ELEMENT
SD-ELEMENT	"[" SD-ID *(SP SD-PARAM) "]"
SD-PARAM	PARAM-NAME " %d34 PARAM-VALUE %d34
SD-ID	SD-NAME
PARAM-NAME	SD-NAME
PARAM-VALUE	UTF-8-STRING ; 文字 '"', '\', および ']' はエスケープする必要があります。
SD-NAME	1*32PRINTUSASCII ; ('', SP, ']', %d34 (')) を除く
MSG	MSG-UTF8
MSG-UTF8	BOM UTF-8-STRING
BOM	%xEF.BB.BF

ESXi ホストのログ フィルタリングの構成

ログ フィルタリング機能を使用すると、ESXi ホストで実行されている syslog サービスのログ ポリシーを変更できません。

vSphere 7.0 Update 2 以降では、ESXCLI を使用してログ フィルタを追加し、ログ フィルタリングを有効にすることができます。ログ フィルタは、一度設定すると削除しないかぎり、ESXi を再起動しても保持されます。

ログ フィルタは、記録先（ログ ディレクトリまたはリモート Syslog サーバ）に関係なく、ESXi ホストの vmsyslogd サービスによって処理されるすべてのログ イベントに影響します。

ログ フィルタリング機能を有効にし、syslog デーモンを再ロードして、ESXi ホストのログ フィルタを有効化する必要があります。

ログ フィルタを構成する ESXCLI コマンドは、次の形式に従います：`esxcli system syslog config logfilter {cmd} [cmd options]`。

たとえば、使用可能なログ フィルタのリストを取得するには、次のコマンドを実行します：`[root@xxx-xx-dhcp-xx-xx:~] esxcli system syslog config logfilter list`。

ログ フィルタリングを有効または無効にするには、`set` コマンドを使用します：`[root@xxx-xx-dhcp-xx-xx:~] esxcli system syslog config logfilter set`。

ログ フィルタを追加するには `add` コマンド、ログ フィルタを削除するには `remove` コマンドを使用します。

ログ フィルタリングが有効になっているかどうかを確認するには、`get` コマンドを使用します。

ログ フィルタは 3 つのコンポーネントによって指定され、次の構文に従います：`numLogs | ident | logRegexp`。

パラメータ	説明
numLogs	フィルタリングを開始する前に、許可される logRegexp Python 正規表現の一致の数を指定します。
ident	ident 文字列は、アプリケーションが Syslog 機能に対して自身を識別する方法を示します。logRegexp フィルタは、同じアプリケーションに関連付けられている必要があります。アプリケーションの ident 文字列は、 <code>/var/run/log</code> のログ ファイルを調べることによって確認できます。各ログ ファイルの 3 番目のフィールドは、ident 文字列で始まり、[で終わります。
logRegexp	除外するメッセージを識別する Python 正規表現。

たとえば、「mark」という単語を含む `hostd` デーモンからの 10 回目より後のメッセージをすべて除外するには、次のコマンドを使用します：`esxcli system syslog config logfilter add --filter="10|Hostd|mark"`。

ログ フィルタを削除するには、次のコマンドを使用します：`esxcli system syslog config logfilter remove --filter="10|Hostd|mark"`。

詳細については、「[ESXi Syslog のオプション](#)」を参照してください。

前提条件

ログ フィルタを作成すれば、ESXi ログの重複エントリ数を削減したり、特定のログ イベントをすべて拒否リストに登録したりできます。

ESXCLI をインストールします。ESXCLI スタート ガイドを参照してください。トラブルシューティングするには、ESXi Shell で `esxcli` コマンドを実行します。

ホスト イメージ プロファイル許容レベルの設定

ホスト イメージ プロファイル許容レベルは、どの vSphere インストール バンドル (VIB) がインストール用に許容されるかを決定します。

VIB 許容レベルとホスト イメージ プロファイル許容レベルに基づき、インストールできるかどうかについて VIB 署名がチェックおよび許容されます。署名ステータスに応じて、VIB には許容レベルのタグが付けられます。

[許容レベルの操作](#)を参照してください。

前提条件

必要な権限：ホスト.構成.セキュリティ プロファイル および ホスト.構成.ファイアウォール

手順

- 1 vSphere Client から vCenter Server に接続します。
- 2 インベントリでホストを選択します。
- 3 [設定] タブをクリックします。
- 4 [システム] で、[セキュリティ プロファイル] 選択します。
- 5 [ホスト イメージ プロファイル許容レベル] までスクロールダウンし、[編集] をクリックします。
- 6 許容レベルを選択して、[OK] をクリックします。

表 4-25. ホスト イメージ プロファイル許容レベル

ホスト イメージ プロファイル許容レベル	VIB の許容されるレベル
保証された VMware	保証された VMware
承認された VMware	VMware 認定済み、VMware 承認済み
サポートされているパートナー	VMware 認定済み、VMware 承認済み、パートナーによるサポート
サポートされているコミュニティ	VMware 認定済み、VMware 承認済み、パートナーによるサポート、コミュニティによるサポート

ESXi のすべてのカスタム パッケージの削除

追加したカスタム パッケージは必要に応じて削除できます。

前提条件

カスタム パッケージを削除する前に、実行中の仮想マシンをシャットダウンするか、ESXi ホストから移行する必要があります。

手順

- 1 ESXi ホストを再起動します。
- 2 ダイレクト コンソールで [カスタム拡張機能の削除] を選択し、[F11] を押して確定します。
- 3 ホストを再起動します。

結果

カスタム パッケージがすべて削除されます。

ESXi 構成ファイルの変更

/bin/configstorecli ツールを使用して、ESXi 構成ストア (ConfigStore) 内の ESXi 構成ファイルを変更できます。

ESXi 7.0 Update 1 で導入された ESXi Shell ツール configstorecli の目的は、さまざまな方法や多様な構成ファイルを使用せずに、ESXi ホストのすべての構成を一元的に管理することです。

ConfigStore の管理方法の詳細については、VMware ナレッジベースの記事 [KB82227](#) および [KB93720](#) を参照してください。

ESXi での非 ASCII 文字のサポートの無効化

ESXi のバージョンに応じた 2 つの方法を使用して、仮想マシンのファイル名とディレクトリ名での非 ASCII 文字のサポートを無効にできます。

デフォルトでは、ESXi は、仮想マシンのファイル名およびディレクトリ名での ASCII 以外の文字の使用をサポートします。

ESXi 7.0 Update 2 以降で非 ASCII 文字のサポートを無効にする必要がある場合は、VMware ナレッジベースの記事「[KB82227](#)」および「[KB93720](#)」を参照してください。

ESXi 7.0 Update 2 よりも前のバージョンでは、次の手順で /etc/vmware/hostd/config.xml ファイルを変更することで、このサポートを無効にできます。

手順

- 1 テキスト エディタを使用して、ESXi ホストの /etc/vmware/hostd/config.xml ファイルを開きます。
- 2 <config></config> タグの間に、次のコードを追加します。

```
<g11nSupport>false</g11nSupport>
```
- 3 ファイルを保存して閉じます。

4 ホストを再起動します。

このサポートを無効にした後でも、仮想マシン名には ASCII 以外の文字を入力することができます。vSphere ユーザー インターフェイスに、ASCII 以外の文字の仮想マシン名が表示されますが、ESXi が実際のファイルおよびディレクトリ名を ASCII 文字列に変換します。

システム構成のリセット

ESXi ホストで発生した問題の原因が特定できない場合は、システム構成をリセットできます。

システム構成に変更を加えると、ネットワークおよびデバイスへの接続の問題を含むさまざまな問題につながる場合があります。システム構成をリセットすると、そのような問題を解決できる可能性があります。システム構成をリセットしても問題が解決されない場合は、初期設定以降に加えられた変更を問題の原因から除外することができます。

構成をリセットすると、ソフトウェアは構成に加えたすべての変更をオーバーライドし、管理者アカウント (root) のパスワードを削除したあと、ホストを再起動します。IP アドレスの設定やライセンス構成など、ハードウェア ベンダーによる構成の変更も削除される場合があります。

構成をリセットしても、その ESXi ホストの仮想マシンは削除されません。構成をデフォルトにリセットしたあと仮想マシンを認識できなくなりますが、ストレージを再構成し、仮想マシンを再登録することで再び表示されるようになります。

注意： 構成をデフォルトにリセットすると、ホストにアクセスしているユーザーの接続は失われます。

前提条件

構成をリセットする前に、ESXi 構成をリストアする必要が生じた場合に備えて構成をバックアップします。

手順

- 1 Get-VMHostFirmware PowerCLI コマンドレットを使用して、構成をバックアップします。
- 2 ダイレクト コンソールから、[システム構成をリセット] を選択して Enter キーを押します。
- 3 F11 キーを押して確認します。

結果

システムは、すべての設定をデフォルト値にリセットしたあと再起動します。

ESXi のインストールおよび設定後

ESXi をインストールして設定した後は、さまざまなインターフェイスでホストを管理し、ホストにライセンスを付与して構成をバックアップすることができます。

ESXi をインストールして設定した後は、vSphere Client と vCenter Server を使用してホストを管理し、ホストにライセンスを付与して ESXi 構成をバックアップすることができます。VMware Host Client を使用して ESXi ホストに直接接続し、管理することができます。VMware Host Client のインストールおよび使用については、vSphere 単一ホスト管理を参照してください。

注： vCenter Server インスタンスに接続されていない、TMP が有効なスタンドアロン ESXi ホストをインストールする場合は、ESXi 構成のリカバリ キーのバックアップを作成します。リカバリ キーを取得するには、ESXi ホストで `esxcli system settings encryption recovery list` コマンドを実行して、キーをメモします。ホスト構成の暗号化に関連する問題が原因で、ホストの起動が完了しない可能性があります。このような場合は、リカバリ キーを使用して構成のリカバリ プロセスを実行し、ホスト構成をリストアできます。

ベスト プラクティスと VMware の推奨事項については、[セキュアな ESXi 構成のベストプラクティス](#)を参照してください。

ESXi ホストへのライセンス供与

インストールされた ESXi には、60 日間の評価期間があり、その期間中に、vSphere Enterprise Plus ライセンスで提供されるすべての vSphere 機能を評価検討できます。評価期間が終了する前に、適切なライセンスをホストに割り当てる必要があります。

ESXi ホストは、CPU 単位のキャパシティを持つ vSphere ライセンスでライセンス付与されます。ホストに適切にライセンス供与するには、ホスト内のすべての CPU に対応できる十分な CPU キャパシティを持つ vSphere ライセンスをホストに割り当てる必要があります。ライセンスは、ホストで使用するすべての機能をサポートする必要があります。たとえば、ホストが vSphere Distributed Switch に接続されている場合は、vSphere Distributed Switch 機能を備えたライセンスを割り当てる必要があります。

次のいずれかの方法を使用して、ESXi ホストにライセンスを供与できます。

- vSphere Client のライセンス管理機能を使用して、複数のホストに一度にライセンスを供与します。ホストは vCenter Server システムに接続されている必要があります。詳細については、『vCenter Server およびホストの管理』を参照してください。
- PowerCLI コマンドを使用して、バルク ライセンス供与を設定します。バルク ライセンス供与はすべての ESXi ホストで行えますが、Auto Deploy を使用してプロビジョニングされるホストで特に有用です。 [バルクライセンス供与の設定](#) を参照してください。
- 個々の ESXi ホストにライセンスを供与するには、VMware Host Client との直接接続を使用します。ESXi ホストにライセンス キーを割り当てる方法については、vSphere 単一ホスト管理を参照してください。

ESXi ホストのライセンス キーの記録

ホストがアクセス不能になったり起動できなくなったりした場合、そのライセンス キーの記録が必要になります。ライセンス キーを書き留めて、サーバにテープで貼るか、安全な場所にライセンス キーを置いておくことができます。ライセンス キーには、ダイレクト コンソール ユーザー インターフェイスまたは vSphere Client からアクセスできます。

vSphere Client を使用した ESXi ホストのライセンス キーの表示

vSphere Client を使用して、vCenter Server システムに接続されているホストのライセンス キーを表示できません。

手順

- 1 vSphere Client で、[管理] を選択します。
- 2 [ライセンス] で [ライセンス] を選択します。
- 3 [資産] タブで、[ホスト] を選択します。
- 4 [ライセンス] 列で、ライセンスをクリックします。

結果

ライセンス使用量やライセンス キーなどのライセンスに関する情報が表示されます。

ダイレクト コンソールを使用した ESXi ライセンス キーへのアクセス

ホストに物理的アクセスしている場合、またはダイレクト コンソールにリモート アクセスしている場合、ダイレクト コンソールを使用して ESXi のライセンス キーにアクセスできます。

手順

- ◆ ダイレクト コンソールから、[サポート情報の表示] を選択します。
ライセンス キーが、XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX の形式で表示され、そのラベルはライセンス シリアル番号です。

注： また、物理マシンのシリアル番号がシリアル番号のラベルで表示されます。ライセンス キーと物理マシンのシリアル番号を間違えないように気をつけてください。

システム ログの表示

システム ログは、システムの操作イベントに関する詳細情報を提供します。

手順

- 1 ダイレクト コンソールから、[システム ログの表示] を選択します。
- 2 対応する番号キーを押して、ログを表示します。
ホストを vCenter Server に追加した場合、vCenter Server Agent (vpxa) ログが表示されます。
- 3 [Enter] またはスペースバーを押してメッセージをスクロールします。

- 4 正規表現での検索を実行します。
 - a スラッシュ キー (/) を押します。
 - b 検索するテキストを入力します。
 - c [Enter] を押します。

検索されたテキストが、スクリーンにハイライト表示されます。

- 5 [q] を押して、ダイレクト コンソールに戻ります。

次のステップ

ESXi ホストでの [Syslog の構成](#) も参照してください。

ESXi の起動のトラブルシューティング

5

ESXi の起動のトラブルシューティング トピックでは、ESXi の起動時に発生する可能性のある問題に対する解決策を示します。

次のトピックを参照してください。

- 別のホストと起動ディスクを共有する場合の、起動時の予期しないホストの停止
- UEFI モードでの ESXi のインストール後、ホストの起動に失敗する

別のホストと起動ディスクを共有する場合の、起動時の予期しないホストの停止

物理または仮想のいずれかの複数のホストが同じ共有の物理ディスクまたは LUN から起動する場合、これらのホストは同一のスクラッチ パーティションを使用できません。

問題

起動ディスクを別のホストと共有している場合に、ホストが起動時に停止します。

原因

複数の ESXi ホストは、同じ物理ディスクまたは LUN を共有できます。これらのホストのうち 2 つで、同じスクラッチ パーティションが構成されている場合、いずれかのホストが起動に失敗する可能性があります。

解決方法

- 1 順番に起動するようにホストを設定してから、ホストを起動します。

このように設定すると、いずれかのホストのスクラッチ パーティションを変更できるため、ホストを起動することができます。

- 2 vSphere Client から vCenter Server に接続します。
- 3 インベントリでホストを選択します。
- 4 [設定] タブをクリックします。
- 5 [システム] の下で [システムの詳細設定] を選択します。
- 6 [ScratchConfig] を選択します。

[ScratchConfig.CurrentScratchLocation] テキスト ボックスにスクラッチ パーティションの現在の場所が表示されます。

- 7 [ScratchConfig.ConfiguredScratchLocation] テキスト ボックスに、このホストに一意のディレクトリ パスを入力します。

たとえば、`/vmfs/volumes/DatastoreUUID/DatastoreFolder` のように入力します。

- 8 ホストを再起動して、変更内容を有効にします。

UEFI モードでの ESXi のインストール後、ホストの起動に失敗する

UEFI モードのホスト マシンに ESXi をインストールした後で、マシンの起動に失敗する場合があります。

問題

ESXi をインストールまたはアップグレードする際に、インストーラは VMware ESXi という名前の UEFI ブート オプションを作成し、それをデフォルトの起動オプションに設定します。ESXi インストール後に再起動すると、再起動に失敗することがあります。この問題が発生すると、次のようなエラー メッセージが表示されます：起動デバイスが利用できません。

原因

- インストーラによって UEFI ブート オプションが作成されると、ホストのマザーボード上の NVRAM への書き込み中に問題が発生します。
- ホスト ファームウェアは、最初の起動オプションとして UEFI ブート オプションを設定する試みを認識しないか、起動順序をオーバーライドします。
- 起動ディスクには、MBR または MSDOS パーティション テーブルがあります。技術的な制限のため、UEFI ブート オプションは GPT (GUID パーティション テーブル) に対してのみ作成されます。

注： UEFI ファームウェアは、FAT ベースの EFI システム パーティションから起動イメージをディスクにロードしようとします。EFI システム パーティションからの起動は、ディスクが GPT を使用して配置されている場合のみ機能します。起動ディスクに MBR または MSDOS パーティション テーブルがある場合、UEFI の起動は失敗します。MBR の起動エントリを追加できません。ディスクは ESXi によって完全に使用されている場合は GPT に変換できず、レガシー BIOS モードで起動する必要があります。

解決方法

- 1 エラー メッセージが画面に表示されている間に、起動オプション メニューを開きます。システムによっては、起動オプション メニューを開くのに、キーボード ショートカット、BIOS メニュー、または BMC、iLO、または iDRAC インターフェイスを使用することができます。
- 2 起動オプション VMware ESXi があるかどうかを確認し、そのオプションで起動を試してください。起動に成功した場合は、起動順序を変更し、最初の起動オプションとして VMware ESXi を設定します。
- 3 問題が解決されない場合は、[起動オプションの追加] に似た名前のオプションを選択します。
オプションの表現と場所はシステムによって異なる場合があります。
- 4 ESXi をインストールしたディスク上の `\EFI\BOOT\BOOTx64.EFI` ファイルを選択します。
- 5 追加したオプションからホストが起動するように、起動の順番を変更します。

ESXi ホストの廃止

6

サーバを ESXi ホストにしない場合は、ESXi ホスト マシンを廃止します。

手順

- 1 内部ディスクの VMFS データストアを削除し、仮想マシンをこれ以上内部ディスクに格納しないように設定します。
- 2 BIOS の起動設定を変更し、ホストが ESXi で起動しないようにします。
ESXi を UEFI モードでインストールした場合は、起動オプション VMware ESXi、または手動で作成した他の起動オプションを削除します。
- 3 その場所に、別のオペレーティング システムをインストールします。