

ESXi のアップグレード

Update 3

VMware vSphere 7.0

VMware ESXi 7.0

最新の技術ドキュメントは、VMware の Web サイト (<https://docs.vmware.com/jp/>)

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

VMware株式会社
〒108-0023 東京都港区芝浦 3-1-1
田町ステーションタワー N 18 階
www.vmware.com/jp

Copyright © 2018-2022 VMware, Inc. All rights reserved. 著作権および商標情報。

目次

1	『VMware ESXi のアップグレード』について	5
2	vCenter Server のアップグレード オプション	6
	vSphere のアップグレード プロセスの概要	6
	ESXi ホスト アップグレード プロセスの概要	8
	仮想マシンと VMware Tools のアップグレード	11
3	ESXi ホストのアップグレード	12
	ESXi の要件	12
	ESXi のシステム ストレージの概要	13
	ESXi のハードウェア要件	16
	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	19
	ESXi のパフォーマンスを改善するための推奨事項	20
	ESXi ホストの送受信ファイアウォール ポート	21
	システム ログに必要な空き容量	21
	VMware Host Client のシステム要件	22
	ESXi のパスワードとアカウントのロックアウト	22
	ESXi ホストをアップグレードする前に	25
	サードパーティ製のカスタム VIB があるホストのアップグレード	26
	VMware NSX-T™ Data Center を使用する環境での ESXi ホストのアップグレード	27
	VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするためのカスタム イメージ プロファイルの作成	27
	VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするための新しい ISO イメージの作成	28
	VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするための ESXCLI の使用	29
	ESXi インストーラを起動するためのメディア オプション	30
	ESXi インストーラのダウンロード	36
	ストレージ デバイスの名前と識別子	36
	ホストの対話型アップグレード	38
	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	39
	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	40
	起動オプション	40
	インストール スクリプトおよびアップグレード スクリプトについて	42
	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	54
	スクリプトを使用した、USB フラッシュ ドライブからの ESXi のインストールまたはアップグレード	55
	インストーラのネットワーク ブートによる、スクリプトを使用した ESXi のインストールまたはアップグレード	56
	ネットワーク ブートのインストール プロセスの概要	57

PXE および TFTP を使用した ESXi インストーラの起動	58
iPXE および HTTP を使用した ESXi インストーラの起動	60
ネイティブ UEFI HTTP を使用した ESXi インストーラの起動	63
ネットワーク ブートの背景情報	65
PXELINUX 構成ファイル	66
DHCP の構成例	67
ESXCLI コマンドを使用した、ホストのアップグレード	71
VIB、イメージ プロファイル、およびソフトウェア デポ	71
VIB およびホストの許容レベルについて	72
更新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か、再起動する必要があるかの確認	74
ホストをメンテナンス モードに切り替える	75
個々の VIB によるホストの更新	77
イメージ プロファイルによるホストのアップグレードまたはアップデート	78
Zip ファイルを使用した ESXi ホストのアップデート	81
ホストからの VIB の削除	82
ESXCLI コマンドを使用した、ホストへのサードパーティ製拡張機能の追加	84
ESXCLI インストールまたはアップグレードのドライ ランの実行	84
次回ホストを再起動したあとでアクティブになるインストール済み VIB およびプロファイルの表示	85
ホストのイメージ プロファイルと許容レベルの表示	85
ESXi ホストをアップグレードした後に行う作業	86
ESXi の評価モードとライセンス モードについて	86
アップグレード後の ESXi ホストへのライセンス供与	87
アップグレード後の ESXi ホストでのセキュア ブート検証スクリプトの実行	87
システム ログに必要な空き容量	88
ESXi ホストでの Syslog の構成	89
ESXi Syslog のオプション	90
ESXi ホストのログ フィルタリングの構成	94
4 vSphere Auto Deploy を使用した、ホストの再プロビジョニング	96
vSphere Auto Deploy の概要	96
vSphere Auto Deploy の準備	99
vSphere Auto Deploy 用にシステムを準備	99
vSphere Auto Deploy コマンドレットの使用	102
バルク ライセンス供与の設定	103
ホストの再プロビジョニング	105
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	105
PowerCLI 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	106
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	107
ルールのコンプライアンスのテストおよび修復	109
5 ESXi ホストのトラブルシューティング用のログの収集	111

『VMware ESXi のアップグレード』について

1

『VMware ESXi のアップグレード』では、VMware ESXi™ を最新バージョンにアップグレードする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 のアップグレード』は、ESXi を旧バージョンからアップグレードする必要があるすべてのユーザーを対象にしています。これらのトピックは、Microsoft 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。

vCenter Server のアップグレード オプション

2

vCenter Server 7.0 には、vCenter Server デプロイをアップグレードするためのオプションが数多くあります。vCenter Server のアップグレードを成功させるには、アップグレード オプション、アップグレード プロセスに影響を与える構成の詳細、タスクの順序を理解しておく必要があります。

vSphere の 2 つの主要なコンポーネントは、ESXi™ と VMware vCenter Server™ です。ESXi は、仮想マシンおよび仮想アプライアンスを作成および実行できる仮想プラットフォームです。vCenter Server は、ネットワークに接続された ESXi ホストを統合管理する役割を果たすサービスです。vCenter Server システムを使用して、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 Appliance は、vCenter Server 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です。

組み込みまたは外部の Platform Services Controller を使用する既存の vCenter Server 環境を、vCenter Server Appliance で構成される環境にアップグレードできます。

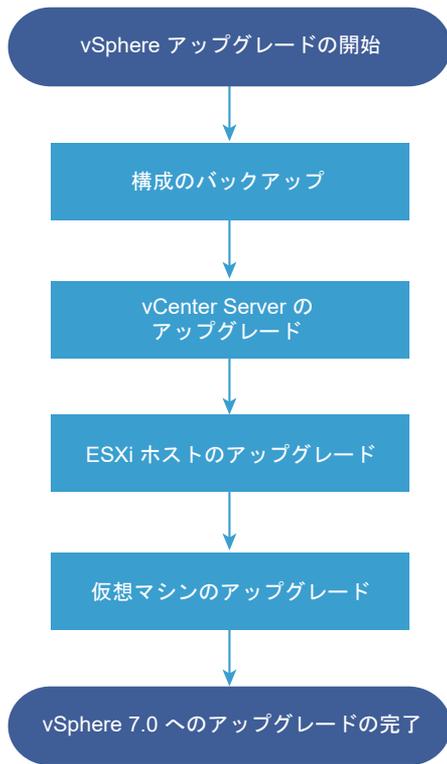
この章には、次のトピックが含まれています。

- [vSphere のアップグレード プロセスの概要](#)

vSphere のアップグレード プロセスの概要

vSphere では、複数のコンポーネントをアップグレードできます。アップグレードに必要な一連のタスクを理解することは、vSphere アップグレードの成功に不可欠です。

図 2-1. vSphere のアップグレード タスクの概要



vSphere のアップグレードには次のタスクが含まれます。

- 1 vSphere リリース ノートを参照します。
- 2 構成をバックアップしたことを確認します。
- 3 vSphere システムに VMware のソリューションまたはプラグインが含まれる場合は、それらがアップグレード後の vCenter Server Appliance のバージョンと互換性があることを確認します。 http://www.vmware.com/resources/compatibility/sim/interop_matrix.php に掲載されている『VMware 製品の相互運用性マトリックス』を参照してください。
- 4 vCenter Server をアップグレードします。
詳細な手順については、『vCenter Server のアップグレード』を参照してください。
- 5 ESXi ホストをアップグレードします。 [ESXi ホスト アップグレード プロセスの概要](#)を参照してください。
- 6 ログ ファイル用に十分なディスク ストレージを確保するために、リモート ログ用に Syslog サーバを設定することを検討します。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 ストレージ容量が限られている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。
- 7 仮想マシンを手動でアップグレードするか、vSphere Lifecycle Manager を使用して組織的にアップグレードします。
[仮想マシンと VMware Tools のアップグレード](#)を参照してください。

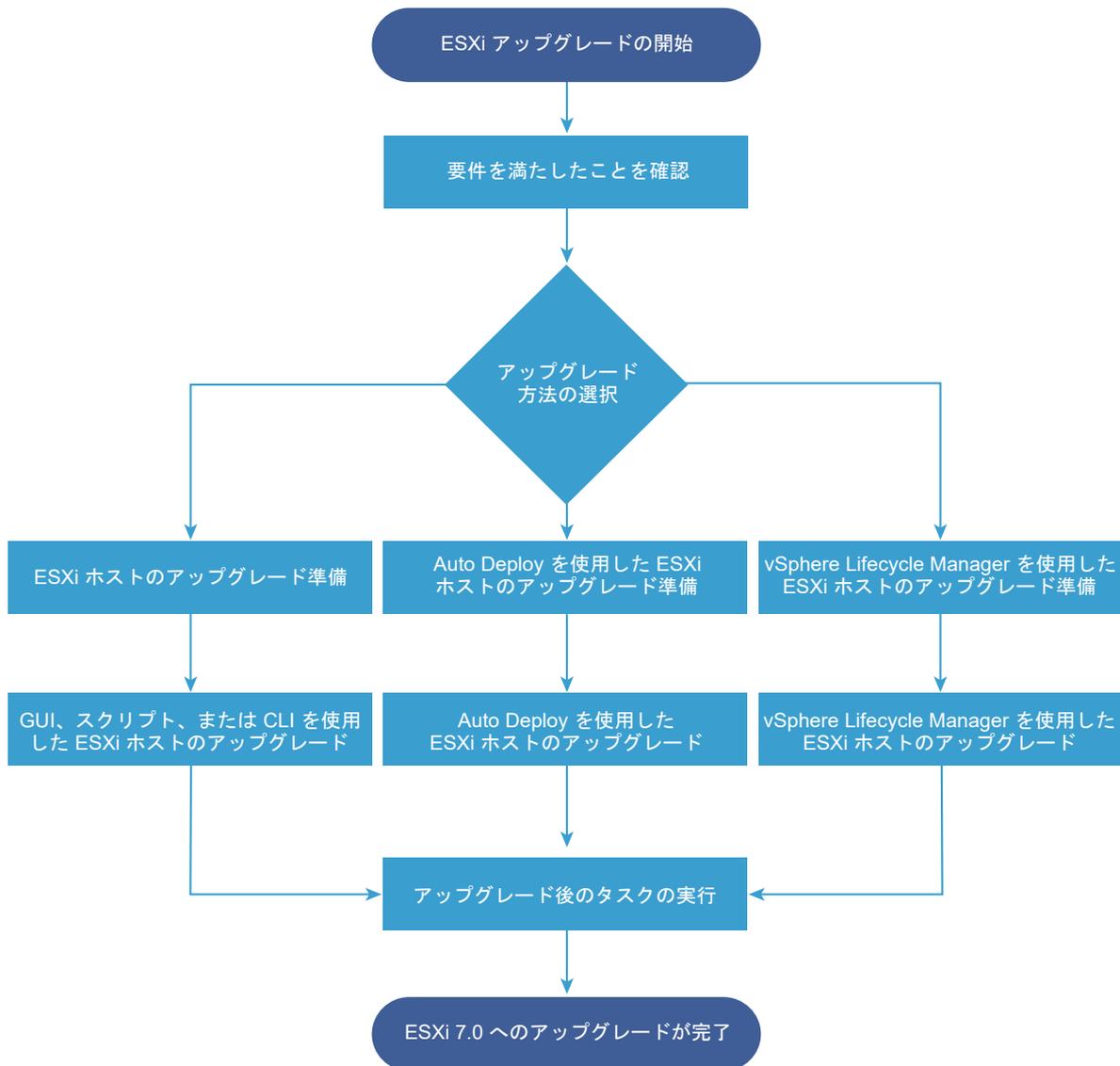
ESXi ホスト アップグレード プロセスの概要

VMware には、バージョン 6.5 とバージョン 6.7 の ESXi ホストを ESXi バージョン 7.0 にアップグレードする方法が複数用意されています。

ESXi7.0 へのアップグレードの詳細とサポート レベルは、アップグレードするホストと使用するアップグレード方法によって異なります。ESXi の現在のバージョンからアップグレード予定バージョンへのアップグレードパスがサポートされていることを確認します。詳細については、http://www.vmware.com/resources/compatibility/sim/interop_matrix.php の「VMware 製品の相互運用性マトリックス」を参照してください。

バージョン 6.5 または 6.7 の ESXi ホストをアップグレードするには、CD、DVD、または USB、スクリプトによるアップグレード、ESXCLI、または vSphere Lifecycle Manager を使用したインタラクティブなアップグレードを利用できます。カスタム VIB を持つ ESXi 6.5 または 6.7 ホストをバージョン 7.0 にアップグレードすると、サポートされるすべてのカスタム VIB が移行されます。詳細については、[サードパーティ製のカスタム VIB があるホストのアップグレード](#)を参照してください。

図 2-2. ESXi ホスト アップグレード プロセスの概要



ESXi をアップグレードする手順の概要を次に示します。

- 1 システムがアップグレード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。
- 2 アップグレードの前に環境を準備します。ESXi ホストをアップグレードする前にを参照してください。
- 3 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。インストーラをネットワーク ブートする場合は、ネットワーク ブート 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。#unique_12 を参照してください。
- 4 ESXi をアップグレードします。3 章 ESXi ホストのアップグレードを参照してください。
- 5 ESXi ホストをアップグレードした後、このホストを vCenter Server に再接続し、ライセンスを適用し直す必要があります。ESXi ホストをアップグレードした後に行う作業を参照してください。

ESXi 7.0 への直接のアップグレードでサポートされる方法は、以下のとおりです。

- CD、DVD、または USB ドライブから、インタラクティブなグラフィカル ユーザー インターフェイス (GUI) インストーラを使用します。
- スクリプトによるアップグレードを実行します。
- ESXCLI を使用します。
- vSphere Auto Deploy を使用します。ESXi ホストが vSphere Auto Deploy を使用してデプロイされる場合は、vSphere Auto Deploy を使用して、7.0 イメージでホストを再プロビジョニングすることができます。
- vSphere Lifecycle Manager を使用する。

グラフィカル ユーザー インターフェイス (GUI) インストーラ

CD、DVD、または USB フラッシュ ドライブにある ESXi インストーラの ISO イメージを使用するかインストーラをネットワーク ブートすることにより、インタラクティブにアップグレードできます。ホストの数が少ないデプロイの場合はこの方法が適しています。インストール中に ESXi インストールを含むターゲット ディスクを選択すると、インストーラによってホストが ESXi バージョン 7.0 にアップグレードされます。インストーラには、一部の既存ホスト設定と構成ファイルを移行して、既存の VMFS データストアを保持するためのオプションも用意されています。[ホストの対話型アップグレード](#)を参照してください。

スクリプトによるアップグレードの実行

スクリプトによるアップグレードを実行するには、CD、DVD、または USB フラッシュ ドライブから ESXi 7.0 インストーラを使用するか、インストーラをネットワーク ブートします。このメソッドは、複数のホストを導入するのに効率的な方法です。詳細については、[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。

ESXCLI

ESXCLI を使用して、ESXi 6.5 ホストまたは ESXi 6.7 ホストを ESXi 7.0 ホストにアップグレードできます。[ESXCLI コマンドを使用した、ホストのアップグレード](#)を参照してください。

vSphere Auto Deploy

ESXi 5.x ホストが vSphere Auto Deploy を使用してデプロイされている場合は、vSphere Auto Deploy を使用して、ホストを再プロビジョニングし、新しいイメージ プロファイルで再起動します。このようなプロファイルには、ESXi アップグレードまたはパッチ、ホスト構成プロファイルが含まれ、さらに VMware のパートナーが提供するサードパーティ製ドライバまたは管理エージェントがオプションで含まれます。vSphere ESXi Image Builder CLI を使用すると、カスタム イメージを作成できます。詳細については、[4 章 vSphere Auto Deploy を使用した、ホストの再プロビジョニング](#)を参照してください。

vSphere Lifecycle Manager

vSphere Lifecycle Manager は、ESXi ホストをインストール、アップグレード、および更新するための vCenter Server サービスです。イメージとベースラインを使用することで、vSphere Lifecycle Manager によって、複数の ESXi ホストのライフサイクル管理をクラスター レベルで一元化および簡素化できます。組織的なインストール、アップグレード、更新の実行の詳細については、『[ホストとクラスターのライフサイクルの管理](#)』ドキュメントを参照してください。

仮想マシンと VMware Tools のアップグレード

ESXi ホストをアップグレードしたら、新機能を利用するためにホスト上の仮想マシンをアップグレードできます。仮想マシンのアップグレードに、次のツールをご利用できます。

vSphere Client

vSphere Client を使用して、手順に従いながら仮想マシンをアップグレードできます。仮想マシンのアップグレードの詳細については、『vSphere の仮想マシン管理』を参照してください。

vSphere Lifecycle Manager

vSphere Lifecycle Manager を使用して、ご利用環境にある仮想マシンのハードウェアおよび仮想マシンの VMware Tools バージョンをアップグレードできます。vSphere Lifecycle Manager はアップグレードプロセスを自動化し、手順が正しい順序で実行されていることを検証します。詳細については、『ホストとクラスターのライフサイクルの管理』を参照してください。

ESXi ホストのアップグレード

3

vCenter Server のアップグレードが完了したら、ESXi ホストをアップグレードします。ESXi 6.5 および 6.7 ホストを直接 ESXi 7.0 にアップグレードできます。

ESXi ホスト アップグレード プロセスの概要に記載のツールと方法を使用して、ホストをアップグレードすることができます。

注意： vCenter Server によって管理されているホストをアップグレードする場合は、ESXi をアップグレードする前に、vCenter Server をアップグレードする必要があります。正しい順序で環境をアップグレードしなかった場合は、データが消失したり、サーバにアクセスできなくなったりする可能性があります。

この章には、次のトピックが含まれています。

- ESXi の要件
- ESXi ホストをアップグレードする前に
- サードパーティ製のカスタム VIB があるホストのアップグレード
- VMware NSX-T™ Data Center を使用する環境での ESXi ホストのアップグレード
- 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 インストーラのダウンロード
- ストレージ デバイスの名前と識別子
- ホストの対話型アップグレード
- 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
- ネットワーク ブートのインストール プロセスの概要
- ESXCLI コマンドを使用した、ホストのアップグレード
- ESXi ホストをアップグレードした後に行う作業

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 7.0 では、デバッグの簡素化とともに、パーティションの柔軟な管理や、大規模なモジュールやサードパーティ製コンポーネントのサポートを可能にするシステム ストレージ レイアウトが導入されています。

ESXi 7.0 のシステム ストレージの変更

ESXi 7.0 より前では、`/scratch` パーティションとオプションの VMFS データストア以外でパーティション サイズが固定されていたため、パーティション番号が静的で、パーティションの管理に制限がありました。ESXi 7.0 では、複数のパーティションが統合されてパーティション数が減少し、パーティションのサイズは大きくなります。また、各パーティションは、使用する起動メディアとその容量に応じて拡張可能です。

ESXi 7.0 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。

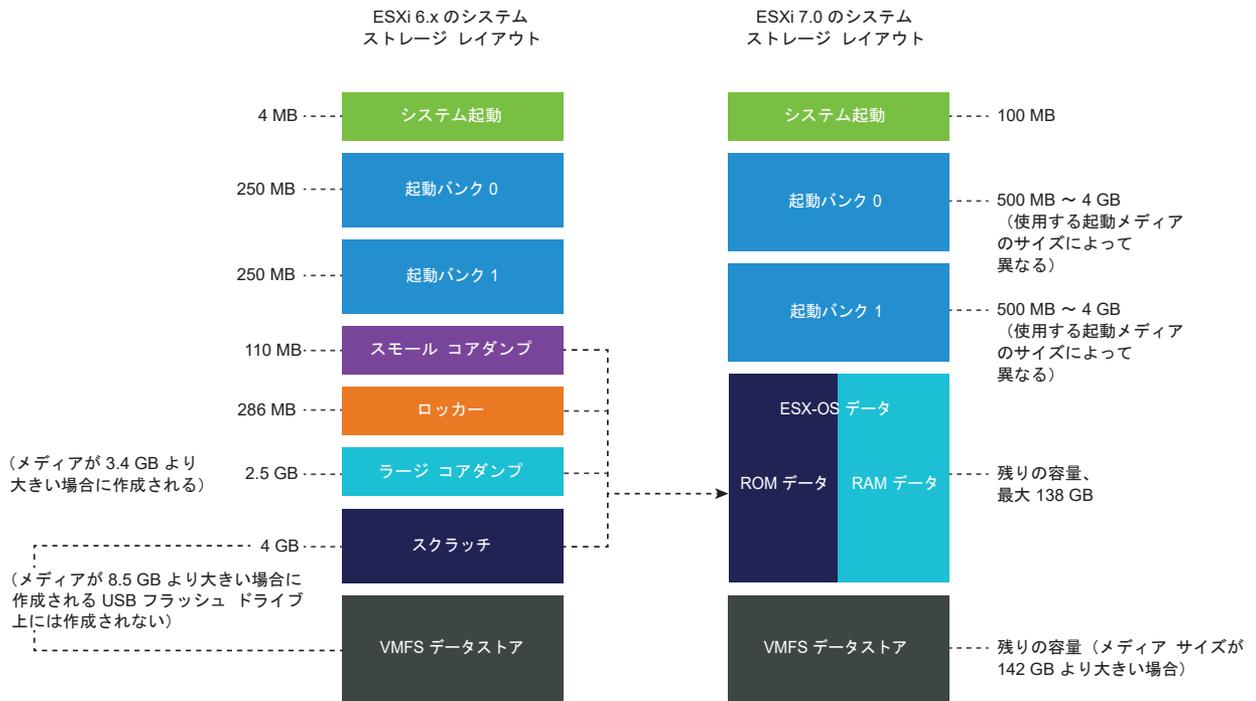
表 3-1. ESXi 7.0 のシステム ストレージのパーティション

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
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	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。 レガシーの <code>/scratch</code> 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。 注意： ESX-OSData パーティションは常に、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 デバイスに作成してください。USB、SD、および非 USB フラッシュ メディア デバイスは、ブートバンク パーティションのみに使用します。	VMFS-L

ESX-OSData ボリュームは、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 3-1. ESXi 7.0 以降の統合システム ストレージ



ESXi 7.0 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 3-2. ESXi 7.0 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	4 ~ 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

vSphere 7.0 Update 1c 以降では、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す。

- 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)

注： GB 単位は 2^{30} バイト、つまり $1,024 \times 1,024 \times 1,024$ バイトの倍数です。

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。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 7.0 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 3-3. ESXi 7.0 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
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

ESXi のハードウェア要件

ホストが、ESXi7.0 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。
- ESXi7.0 では、少なくとも 2 つの CPU コアを持つホストが必要です。
- ESXi7.0 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。
- ESXi7.0 では、BIOS で CPU の NX/XD ビットを有効にする必要があります。
- ESXi7.0 には、少なくとも 4 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 8 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 7.0 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。USB、SD、および非 USB フラッシュ メディア デバイスは、ESXi ブート バンク パーティションのみに使用します。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。ソフトウェア ファイバ チャネル オーバー イーサネット (FCoE) については、[ソフトウェア FCoE を使用した ESXi のインストールと起動](#)を参照してください。

ESXi 起動要件

vSphere 7.0 では、ユニファイド エクステンシブル ファームウェア インターフェイス (UEFI) から ESXi ホストを起動できます。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。

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。

ESXi7.0 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 7.0 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。ESXi7.0 へのアップグレードには、最低 4 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。

ESXi 7.0 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。

注： GB 単位は 2^{30} バイト、つまり $1,024 \times 1,024 \times 1,024$ バイトの倍数です。

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。最高のパフォーマンスを得るには、ESX-OSData ボリュームの /scratch および VMware Tools パーティションを格納する 32 GB 以上の別個のパーシステント ローカル デバイスも用意します。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。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。
- ESXi 7.0 Update 3 以降では、起動デバイスが、HDD、SSD、NVMe デバイスなど、ローカル パーシステント ストレージを持たない USB または SD カードの場合、VMware Tools パーティションが RAM ディスクに自動的に作成されます。詳細については、ナレッジベースの記事 [KB83376](#) を参照してください。
- /scratch パーティションをローカル パーシステント ストレージがない USB または SD カードに割り当てると、警告が表示され、フラッシュ メディア デバイス上のブート バンク パーティション以外のパーティションを作成または構成できません。最適なパフォーマンスを得るには、/scratch パーティションを RAM ディスクに設定します。また、/scratch パーティションを構成した後に、SAN または NFS に移動することもできます。詳細については、ナレッジベースの記事 [KB1033696](#) を参照してください。
- 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。

注意： ローカル ディスクが見つからない場合、または起動メディアが USB または SD デバイスで、パーシステント データ用の耐久性に優れた追加ストレージがない場合、/scratch パーティションは RAM ディスクに配置され、/tmp にリンクされます。また、ESXi 7.0 は劣化モードで動作します。

劣化モードの場合は、次のようなシステム アラートが表示されます：アラート： システム ログおよびデータに使用できるパーシステント ストレージがありません。ESX の動作に使用されるシステム ストレージ容量が制限されるため、再起動すると、ログおよびシステム データは失われます。

ESXi 7.0 が劣化モードで動作している場合は、ログに RAM が使用されることでログの永続性がなくなり、ログにエラーが発生したり、一時的なデータのためにメモリ不足状態になったりすることがあります。ディスク状態の再構築に時間がかかるため、起動時間が長くなる可能性があります。

劣化モードを回避するには、十分なサイズのパーシステント ストレージを使用します。別のディスクまたは LUN を使用するように /scratch を再設定できます。

ESXi7.0 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 プロセスのため、ESXi の以前のバージョンにロールバックすることはできません。バージョン 7.0 へのアップグレード後に以前のバージョンの 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。

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、インストーラは使用可能なローカル ディスクに ESX-OSData 領域を割り当てます。使用可能な容量がない場合、データストアは /scratch に使用されます。ローカル ディスクまたはデータストアが見つからない場合、/scratch は RAM ディスクに置かれます。アップグレード後に、/scratch を再設定して永続的なデータストアを使用するか、システム ストレージ ボリューム用に新しいディスクを追加します。

/scratch パーティションの再設定の詳細については、ドキュメント『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。

ESXi 7.0 にアップグレードした後に、新しいローカル ディスクを追加して、autoPartition=TRUE の設定を有効にすることができます。再起動後、起動ディスクはパーティション分割されます。ESXi システム パーティションのサイズを構成する起動オプションの詳細については、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 ディスクまたはデータストアが見つからない場合、/scratch パーティションは RAM ディスクに置かれます。インストールの後に永続データストアを使用するように、/scratch を再設定します。

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 ボリュームを個別の SAN LUN に設定する必要があります。ただし、/scratch が ESX-OSData を使用しないように設定されている場合は、ホストごとに、/scratch に個別の LUN を割り当てる必要はありません。複数の ESXi ホスト用のスクラッチ領域を単一の LUN に一緒に置くことができます。単一の LUN に割り当てるホスト数は、LUN のサイズと仮想マシンの I/O 動作を考慮して比較検討する必要があります。

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 3-4. 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 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

表 3-4. 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
Dell DRAC 4	1.75	1.6.0_23
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
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 3-5. 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 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 8 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 3-5. 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、ESXi7.0 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 および VMware Host Client では、各サービスのファイアウォール ポートを開閉したり、選択した IP アドレスからのトラフィックを許可したりできます。

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

Auto Deploy を使用して ESXi7.0 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す。

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 ポリユームのスクラッチ 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、ESXi7.0 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 3-6. 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	維持するローテーション数	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

Syslog および Syslog サーバの設定と構成、および vSphere Syslog Collector のインストールについては、『vCenter Server のインストールとセットアップ』ドキュメントを参照してください。

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	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 の詳細オプションを使用して、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 の詳細オプションを使用して、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 の詳細オプションを使用して、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が、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` の詳細オプションを使用します。

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。

レガシー ホストで `/etc/pam.d/passwd` ファイルを変更することは引き続きサポートされますが、今後のリリースで、ファイル変更のサポートは廃止されます。代わりに、`Security.PasswordQualityControl` の詳細オプションを使用します。

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` 詳細オプションを使用します。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』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
```

`Security.PasswordHistory` オプションを 5 に設定し、`Security.PasswordMaxDays` オプションを 90 に設定します。

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設定するには、次の詳細オプションを使用します。

- `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。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 ホストをアップグレードする前に

ESXi ホストを正しくアップグレードするために、変更点を理解し、それに備えてください。

ESXi のアップグレードを正常に実行するには、次のベスト プラクティスに従います。

- 1 ESXi のアップグレード プロセス、そのプロセスが既存のデプロイ環境に与える影響、およびアップグレードに必要な準備事項を理解していることを確認します。
 - vSphere システムに VMware のソリューションまたはプラグインが含まれている場合は、それらのソリューションまたはプラグインに、アップグレード後の vCenter Server のバージョンとの互換性があることを確認します。VMware 製品の相互運用性マトリックス (http://www.vmware.com/resources/compatibility/sim/interop_matrix.php) を参照してください。
 - **ESXi ホスト アップグレード プロセスの概要**を読み、サポートされているアップグレード シナリオ、およびアップグレードの実行時に使用できるオプションやツールについて理解します。
 - インストール上の既知の問題については、VMware vSphere リリース ノートを参照してください。
- 2 アップグレードのためにシステムを準備します。
 - 現在の ESXi バージョンでアップグレードがサポートされていることを確認します。**ESXi ホスト アップグレード プロセスの概要**を参照してください。
 - 使用中のシステム ハードウェアが、ESXi の要件に準拠していることを確認します。**ESXi の要件**および **VMware 互換性ガイド** (<http://www.vmware.com/resources/compatibility/search.php>) を参照してください。システムの互換性、ネットワークおよびホスト バス アダプタ (HBA) カードとの I/O の互換性、ストレージの互換性、およびバックアップ ソフトウェアの互換性を確認します。
 - アップグレード用に、使用可能なディスク容量がホスト上に十分あることを確認します。
 - ホストに SAN が接続されている場合は、アップグレードを続行する前にファイバ チャネル システムを外してください。BIOS では HBA カードを無効にしないでください。
- 3 アップグレードを実行する前に、ホストをバックアップしてください。アップグレードに失敗した場合は、ホストをリストアできます。

- 4 Auto Deploy を使用してホストをプロビジョニングする場合、プロセスを実行するユーザーは、プロビジョニング対象の ESXi ホストに対してローカルの管理者権限を保持している必要があります。インストール プロセスの場合はデフォルトでこれらの権限があり、証明書のプロビジョニングは想定どおりに行われます。ただし、インストーラ以外の手段を使用する場合は、ローカルの管理者権限を保持するユーザーとして実行する必要があります。
- 5 選択したアップグレード オプションによっては、ホスト上のすべての仮想マシンを移行またはパワーオフする必要があります。アップグレード方法の説明を参照してください。
 - CD、DVD、または USB ドライブからのインタラクティブなアップグレードについては、[ホストの対話型アップグレード](#)を参照してください。
 - スクリプトを使用したアップグレードについては、[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。
 - vSphere Auto Deploy については、[4 章 vSphere Auto Deploy を使用した、ホストの再プロビジョニング](#)を参照してください。ESXi 6.5.x または 6.7.x ホストが vSphere Auto Deploy を使用してデプロイされた場合は、vSphere Auto Deploy を使用して、7.0 イメージでホストを再プロビジョニングすることができます。
 - `esxcli` コマンドによる方法については、[ESXCLI コマンドを使用した、ホストのアップグレード](#)を参照してください。
- 6 ESXi ホストのアップグレード後に実行する必要があるタスクを計画します。
 - システムをテストしてアップグレードが正常に完了したことを確認します。
 - ホストのライセンスを適用します。[アップグレード後の ESXi ホストへのライセンス供与](#)を参照してください。
 - 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。vSphere Syslog Collector は、サービスとして vCenter Server 6.0 に含まれており、すべてのホストからのログ収集に使用できます。[システム ログに必要な空き容量](#)を参照してください。Syslog および Syslog サーバの設定と構成、ホスト プロファイル インターフェイスからの Syslog の設定、および vSphere Syslog Collector のインストールについては、ドキュメント 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。
- 7 アップグレードに失敗しても、ホストをバックアップしてある場合はホストをリストアできます。

サードパーティ製のカスタム VIB があるホストのアップグレード

ホストには、サードパーティ製のドライバや管理エージェントなど、カスタム vSphere インストール バンドル (VIB) をインストールできます。ESXi ホストを 7.0 にアップグレードすると、インストーラ ISO に VIB が含まれているかどうかにかかわらず、サポートされているすべてのカスタム VIB が移行されます。

ホストまたはインストーラ ISO イメージに、競合をもたらす、アップグレードを妨げる VIB が含まれる場合、エラーメッセージに競合している VIB が示されます。ホストをアップグレードするには、次のいずれかの操作を実行します。

- 競合する VIB を ESXi ホストから削除して、アップグレードをもう一度試します。esxcli コマンドを使用して、ホストから VIB を削除できます。詳細については、[ホストからの VIB の削除](#) を参照してください。
- vSphere ESXi Image Builder CLI を使用して、競合を解決するカスタム ISO イメージを作成します。vSphere ESXi Image Builder CLI 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。

VMware NSX-T™ Data Center を使用する環境での ESXi ホストのアップグレード

vSphere システムに NSX-T Data Center が含まれている場合、ESXi ホストをアップグレードするには、アップグレードに使用する目的のソフトウェア仕様（ベースライン）に NSX カーネル モジュールが含まれていることを事前に確認する必要があります。

ESXi ホストを 7.0 以降にアップグレードすると、インストーラ ISO に VIB が含まれているかどうかにかかわらず、サポートされているすべてのカスタム VIB が移行されます。ただし、NSX カーネル モジュールはインストーラ ISO イメージに自動的に移行されません。アップグレード操作に進む前に、次のいずれかのアクションを実行する必要があります。

- 新しくアップロードされた NSX カーネル モジュールを使用して拡張機能ベースラインを作成します。詳細については、[ホストとクラスタのライフサイクルの管理](#) を参照してください。
- NSX カーネル モジュールを使用してカスタム イメージ プロファイルを作成します。詳細については、『[VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするためのカスタム イメージ プロファイルの作成](#)』を参照してください。
- PowerCLI を使用して、新しい ISO イメージを作成します。詳細については、『[VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするための新しい ISO イメージの作成](#)』を参照してください。
- ESXCLI を使用します。詳細については、『[VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするための ESXCLI の使用](#)』を参照してください。

VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするためのカスタム イメージ プロファイルの作成

vSphere システムに NSX-T Data Center が含まれている場合、ESXi ホストを以前のバージョンの ESXi から 7.0 以降にアップグレードするには、アップグレードに使用するベースラインに NSX カーネル モジュールが含まれていることを事前に確認する必要があります。このために、ESXi 基本イメージと新しくアップロードされた NSX カーネル モジュールを使用して、カスタム イメージ プロファイルを作成することができます。

前提条件

- [VMware Customer Connect](#) から、環境にデプロイされた NSX-T Data Center のバージョンの NSX Kernel Module for VMware ESXi 7.0 zip ファイルをダウンロードします。たとえば、VMware NSX-T Data Center 3.0.0 の場合は、`nsx-lcp-3.0.0.0.0.15945993-esx70.zip` をダウンロードします。
- vCenter Server システムで Auto Deploy と Image Builder が有効になっていることを確認します。

手順

- 1 vCenter Server 7.0.x システムにログインします。
- 2 [ホーム] - [AutoDeploy] - [ソフトウェア デポ] の順に選択して、ESXi 7.0.x 基本イメージ（まだ使用できない場合）と、NSX カーネル モジュールの 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。
- 3 NSX-T Data Center の NSX カーネル モジュールと ESX 7.0.x の基本イメージを組み合わせたイメージ プロファイルを作成します。詳細な手順については、[イメージ プロファイルの作成](#)を参照してください。
- 4 カスタム 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。
- 5 ISO イメージを vSphere Lifecycle Manager デポにインポートします。

これで、vSphere Lifecycle Manager を使用して、インポートされた ISO イメージに基づいてアップグレード ベースラインを作成できるようになりました。ベースラインを使用する vSphere Lifecycle Manager アップグレード ワークフローの詳細については、『ホストとクラスタのライフサイクルの管理』ガイドを参照してください。

VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするための新しい ISO イメージの作成

vSphere システムに NSX-T Data Center が含まれている場合、ESXi ホストを以前のバージョンの ESXi から 7.0 以降にアップグレードするには、アップグレードに使用するソフトウェア仕様（ベースライン）に NSX カーネル モジュールが含まれていることを事前に確認する必要があります。このために、`New-IsoImage PowerCLI` コマンドレットを使用して新しい ISO イメージを作成し、任意の方法で ESXi アップグレードを実行することができます。

前提条件

- [VMware Customer Connect](#) から、環境にデプロイされた NSX-T Data Center のバージョンの NSX Kernel Module for VMware ESXi 7.0 zip ファイルをダウンロードします。たとえば、VMware NSX-T Data Center 3.0.0 の場合は、`nsx-lcp-3.0.0.0.0.15945993-esx70.zip` をダウンロードします。
- 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 のインストールおよび使用](#)を参照してください。
- 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

- ◆ PowerCLI セッションで `New-IsoImage` コマンドレットを実行し、`Depots`、`Destination`、および `SoftwareSpec` パラメータを渡して ISO イメージを生成します。たとえば、PS

```
C:\Users\Administrator> New-IsoImage -Depots "C:\VMware-ESXi-7.0U1-16850804-depot.zip", "C:\nsx-lcp-3.0.0.0.0.15945993-esx70.zip", -Destination C:\<your new ISO image name>.iso -SoftwareSpec C:\<your file name>.json
```

など。このコマンドでは、ESXi 基本イメージと NSX カーネル zip ファイル、および JSON ファイル内の目的のイメージのソフトウェア仕様を使用して、新しい ISO イメージを作成します。任意の数とソフトウェア デポ（オフラインとオンライン）の組み合わせを使用できます。ESXi 7.0 にアップグレードする場合は、`New-IsoImage` コマンドレットによって、vSphere Lifecycle Manager で必要になる ESXi 7.0.x の他のメタデータが保持されます。

次のステップ

新しい ISO イメージを使用して、任意の方法で ESXi のアップグレードを完了します。vSphere Lifecycle Manager アップグレード ワークフローの詳細については、『[ホストとクラスタのライフサイクルの管理](#)』ガイドを参照してください。

VMware NSX-T Data Center を使用する環境で ESXi ホストをアップグレードするための ESXCLI の使用

vSphere システムに NSX-T Data Center が含まれている場合、ESXi ホストを以前のバージョンの ESXi から 7.0 以降にアップグレードするには、アップグレードに使用するソフトウェア仕様（ベースライン）に NSX カーネル モジュールが含まれていることを事前に確認する必要があります。ESXCLI コマンドを使用して、ESXi ホストをアップグレードし、NSX カーネル モジュールを再インストールできます。

NSX-T Data Center を含む vSphere システムで ESXCLI を使用して ESXi ホストをアップグレードするには、[ESXCLI コマンドを使用した、ホストのアップグレード](#)に記載されている手順を実行する必要があります。

前提条件

- [VMware Customer Connect](#) から、環境にデプロイされた NSX-T Data Center のバージョンの NSX Kernel Module for VMware ESXi 7.0 zip ファイルをダウンロードします。たとえば、VMware NSX-T Data Center 3.0.0 の場合は、`nsx-lcp-3.0.0.0.0.15945993-esx70.zip` をダウンロードします。

手順

- 1 ESXi ホストをメンテナンス モードにします。詳細については、[ホストをメンテナンス モードに切り替える](#)を参照してください。
- 2 URL を指定してアクセスできるソフトウェア デポ、またはオフライン ZIP デポに格納されている ESXi 7.0.x イメージ プロファイルをダウンロードします。

- 3 ESXCLI コマンド `esxcli software profile update --depot <path-to-depot-file> -p ESXi-X.X.X-XXXXXX-standard --allow-downgrades --no-sig-check` を実行します。例: `esxcli software profile update --depot /vmfs/volumes/5e8fd197-68bce4dc-f8f1-005056af93cf/VMware-ESXi-7.0.0-15843807-depot.zip -p ESXi-7.0.0-15843807-standard --allow-downgrades --no-sig-check` 詳細については、[イメージ プロファイルによるホストのアップグレードまたは更新](#)を参照してください。
- 4 ESXCLI コマンド `esxcli software vib install -d <path_to_kernel_module_file> --no-sig-check` を使用して、NSX カーネル モジュールをインストールします。例: `esxcli software vib install -d /tmp/nsx-lcp-3.0.0.0.0.15945993-esx70.zip`
- 5 ESXi ホストを再起動します。
- 6 ESXi ホストのメンテナンス モードを終了します。

ESXi インストーラを起動するためのメディア オプション

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。[#unique_12](#)
- 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。[#unique_34](#) 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。

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
- `isolinux.cfg` ファイルを含む ESXi ISO イメージ、`VMware-VMvisor-Installer-version_number-build_number.x86_64.iso`
- Syslinux 3.86 パッケージ。他のバージョンは、ESXi と互換性がない可能性があります。

手順

- 1 Linux を起動してログインし、`su` または `sudo root` コマンドを使用してスーパーユーザー モードに切り替えます。
- 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 フラッシュ デバイスにパーティション テーブルを作成します。

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

または、`o` と入力して、新しい空の DOS パーティション テーブルを作成します。

- a `d` と入力して、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` と入力して、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` と入力して、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 `a` と入力して、パーティション 1 にアクティブ フラグを設定します。

e 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)
```

f w と入力して、パーティション テーブルを書き込み、プログラムを終了します。

4 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
```

5 USB フラッシュ ドライブに Syslinux ブートローダーをインストールします。

Syslinux 実行ファイルと mbr.bin ファイルの場所は、Syslinux バージョンによって異なる可能性があります。次のコマンドがデフォルトです。

```
/usr/bin/syslinux /dev/sdb1
cat /usr/lib/syslinux/mbr/mbr.bin > /dev/sdb
```

たとえば、syslinux-3.86.zip のコピーをダウンロードし、アーカイブを解凍し、その指示に従って Syslinux ソース コードをコンパイルすることができます。その後、ダウンロードしたディレクトリから Syslinux インストールを実行する場合は、次のコマンドを使用できます。

```
cd ~/Downloads/syslinux-3.86
.mtools/syslinux /dev/sdb1
cat mbr/mbr.bin > /dev/sdb
```

6 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 /usbdisk
mount /dev/sdb1 /usbdisk
```

7 ソース ディレクトリを作成し、ESXi インストーラ ISO イメージをマウントします。

```
mkdir /esxi_cdrom
mount -o loop VMware-VMvisor-Installer-version_number-build_number.x86_64.iso /esxi_cdrom
```

8 ISO イメージの内容を USB フラッシュ ドライブにコピーします。

```
cp -r /esxi_cdrom/* /usbdisk
```

9 ファイル名 isolinux.cfg を syslinux.cfg に変更します。

```
mv /usbdisk/isolinux.cfg /usbdisk/syslinux.cfg
```

10 usbdisk/syslinux.cfg ファイルで、APPEND -c boot.cfg 行を APPEND -c boot.cfg -p 1 に編集し、ブート ローダーが手順 3 で作成した最初のパーティションからファイルを読み取るようにします。パーティション番号は、USB デバイスのレイアウトによって異なる場合があります。

- 11 USB フラッシュ ドライブをアンマウントします。

```
umount /usbdisk
```

- 12 インストーラ ISO イメージをアンマウントします。

```
umount /esxi_cdrom
```

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
```

5 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
```

6 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 スクリプトを開始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。

[インストール スクリプトおよびアップグレード スクリプトについて](#) 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-7.x.x-XXXXXX.x86_64.iso。ここで、7.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

- 1 VMware Web サイトから ESXi ISO イメージをダウンロードします。
- 2 ISO イメージをフォルダにマウントします。

```
mount -o loop VMware-VMvisor-Installer-7.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。

- 3 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
```

- 4 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 (オプション) 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するように、boot.cfg ファイルを変更します。

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
```

UEFI ブートの場合、/efi/boot/ にある boot.cfg ファイルを変更する必要があります。

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。

ESXi インストーラのダウンロード

ESXi のインストーラをダウンロードします。ソフトウェアは、OEM または VMware ダウンロード ポータル (<https://my.vmware.com/>) から入手できます。

前提条件

<https://my.vmware.com/web/vmware/> で VMware Customer Connect アカウントを作成します。

手順

- 1 VMware Customer Connect にログインします。
- 2 [製品とアカウント] - [すべての製品] の順に移動します。
- 3 VMware vSphere を見つけて [製品のダウンロード] をクリックします。
- 4 [バージョンの選択] ドロップダウン メニューから VMware vSphere のバージョンを選択します。
- 5 VMware vSphere Hypervisor (ESXi) のバージョンを選択して、[ダウンロードに移動] をクリックします。
- 6 ESXi ISO イメージをダウンロードします。

ESXi の評価版については、<https://my.vmware.com/en/group/vmware/evalcenter?p=free-esxi7> を参照してください。

ESXi の製品パッチについては、VMware ナレッジベースの記事 [KB1021623](#) を参照するか、<https://my.vmware.com/group/vmware/patch> にアクセスしてください。

- 7 チェックサムを確認します。

ストレージ デバイスの名前と識別子

ESXi 環境では、各ストレージ デバイスは複数の名前で識別されます。

デバイス識別子

ストレージのタイプによって、ESXi ホストは異なるアルゴリズムと規則を使用して、ストレージ デバイスごとに識別子を生成します。

ストレージにより提供される識別子

ESXi ホストは、ターゲット ストレージ デバイスからデバイス名を照会します。返されたメタデータから、ホストはデバイスの一意の識別子を抽出または生成します。識別子は特定のストレージ標準に基づいており、すべてのホストで一貫かつ永続的なもので、次のいずれかの形式をとります。

- naa.xxx
- eui.xxx
- t10.xxx

パスベースの識別子

デバイスが識別子を提供しない場合、ホストは `mpx` を生成します。 *path* の名前。ここで *path* はたとえば `mpx.vmhba1:C0:T1:L3` のようなデバイスの最初のパスを表します。この識別子は、ストレージに提供される識別子と同じように使用できます。

`mpx.path` 識別子は、パス名が一意であることを前提に、ローカル デバイス向けに作成されます。ただし、この識別子は一意でも永続的でもないため、システムを再起動した後に毎回変わる可能性があります。

通常、デバイスへのパスの形式は次の通りです。

`vmhbaAdapter:CChannel:TTarget:LLUN`

- `vmhbaAdapter` はストレージ アダプタの名前です。この名前は、仮想マシンで使用される SCSI コントローラではなく、ホストの物理アダプタを表します。

- `CChannel` はストレージ チャネルの番号です。

ソフトウェア iSCSI アダプタと依存型ハードウェア アダプタは、チャンネル番号を使用して、同じターゲットへの複数のパスを表示します。

- `TTarget` はターゲットの番号です。ターゲットの番号はホストによって決定されますが、ホストに表示されるターゲットのマッピングが変わると、番号も変わることがあります。複数のホストが共有しているターゲットは、同じターゲット番号を持たないことがあります。

- `LLUN` は、ターゲット内の LUN の場所を表す、LUN の番号です。LUN 番号は、ストレージシステムによって提供されます。ターゲットに 1 つの LUN しかない場合、LUN 番号は常にゼロ (0) になります。

たとえば `vmhba1:C0:T3:L1` は、ストレージ アダプタ `vmhba1` とチャンネル 0 を介してアクセスするターゲット 3 上の LUN1 を表します。

レガシー識別子

device-provided identifiers または `mpx.path` 識別子の他に、ESXi は各デバイスの代替のレガシー名も生成します。識別子の形式は次のとおりです。

`vml.number`

レガシー識別子には、デバイスに一意の一連の数字が含まれており、識別子は、SCSI INQUIRY コマンドで取得されたメタデータから部分的に取得することができます。SCSI INQUIRY 識別子を提供しない非ローカル デバイスの場合は `vml.number` 識別子が唯一の使用可能な一意の識別子として使用されます。

例：vSphere CLI でデバイス名を表示

esxcli storage core device list コマンドを使用すると、vSphere CLI にすべてのデバイス名を表示できます。出力例は次のとおりです。

```
# esxcli storage core device list
naa.XXX
    Display Name: DGC Fibre Channel Disk (naa.XXX)
    ...
    Other UIDs: vml.000XXX
mpx.vmhbal:C0:T0:L0
    Display Name: Local VMware Disk (mpx.vmhbal:C0:T0:L0)
    ...
    Other UIDs: vml.0000000000XYZ
```

ホストの対話型アップグレード

CD、DVD、または USB フラッシュ ドライブから ESXi インストーラを起動して、ESXi 6.5 ホストまたは ESXi 6.7 ホストを ESXi 7.0 にアップグレードできます。

アップグレードを行う前に、ネットワーク ストレージの切断を検討してください。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。既存の ESXi 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。

前提条件

- 次のいずれかに ESXi インストーラ ISO があることを確認します。
 - CD または DVD。インストール CD または DVD がない場合は、作成することができます。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： PXE を使用して ESXi インストーラを起動し、対話型インストール、またはスクリプトによるインストールを実行します。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 で行います。
- ESXiEmbedded がホスト上に配置されている必要があります。ESXiInstallable および ESXi Embedded は、同じホスト上に共存できません。
- ESXi ホストのアップグレードを行う場合、ESXi インストーラ ISO に含まれない、サポート対象のカスタム VIB が移行されます。サードパーティ製のカスタム VIB があるホストのアップグレードを参照してください。
- 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。

手順

- 1 ESXi のインストーラ CD または DVD を CD-ROM または 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 を設定します。
- 3 [ディスクの選択] パネルで、ESXi のインストールまたはアップグレードに使用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 によって決まります。ドライブの追加や削除が行われてきたシステムでは、順序が違っている場合があります。

- 4 インストーラが既存の ESXi インストールおよび VMFS データストアを検出できたら、ESXi をアップグレードまたはインストールします。

既存の VMFS データストアを保持できない場合は、ESXi のインストールのみを選択して既存の VMFS データストアを上書きするか、インストールをキャンセルします。既存の VMFS データストアを上書きする場合は、まずそのデータストアをバックアップしてください。
- 5 F11 キーを押して操作を確定し、アップグレードを開始します。
- 6 アップグレードが完了したら、インストール CD または DVD または USB フラッシュ ドライブを取り出します。
- 7 Enter キーを押してホストを再起動します。
- 8 ESXi をアップグレードするときに選択したドライブを最初の起動デバイスとして設定します。

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

スクリプトを使用した無人インストールまたはアップグレードを行って、ESXi ホストを簡単にデプロイできます。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では、複数のホストを効率的にデプロイできます。

インストールまたはアップグレード スクリプトには、ESXi のインストール設定が含まれます。スクリプトは、同様の構成にするホストすべてに適用できます。

スクリプトでインストールまたはアップグレードする場合、サポートされているコマンドを使用してスクリプトを作成する必要があります。スクリプトを編集し、各ホストに固有の設定を変更できます。

インストールまたはアップグレード スクリプトは、次のいずれかの場所に配置できます。

- FTP サーバ
- HTTP/HTTPS サーバ
- NFS サーバ
- USB フラッシュ ドライブ
- CD-ROM ドライブ

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

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 ロードで 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 [#unique_12](#) 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 3-7. 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>	netdevice オプションに似ていますが、syslinux.org サイトで SYSLINUX の IPAPPEND オプションで説明されている、PXELINUX 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 PXELINUX 形式もサポートされています。syslinux.org サイトで、SYSLINUX の中の IPAPPEND オプションを参照してください。
<code>ks=cdrom:/path</code>	<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 CDRROM 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。 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=cdrom:/KS_CUST.CFG</code> 。
<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 http、https、ftp、または nfs 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 vmnicNN 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。

表 3-7. 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>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 <code>device</code> 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。
<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"> ■ <code>min</code> (32 GB、1 台のディスクまたは組み込みのサーバ用) ■ <code>small</code> (64 GB、512 GB 以上の RAM を搭載したサーバ用) ■ <code>default</code> (128 GB) ■ <code>max</code> (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) <p>注： GB 単位は 2^{30} バイト、つまり $1,024 \times 1,024 \times 1,024$ バイトの倍数です。</p>

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。

インストール スクリプトおよびアップグレード スクリプトについて

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` などがあります。

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。

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。インストール スクリプトおよびアップグレード スクリプトについて 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。

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。

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。 <code>--drives=</code> フラグまたは <code>--alldrives</code> 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>	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。 <ol style="list-style-type: none"> ローカルに接続されたストレージ (local) ネットワーク ストレージ (remote) USB ディスク (usb)

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、 `remotesx` 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,...]` 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`)
- 3 USB ディスク (`usb`)

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、 `remotesx` 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** 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。
- novmfsdisk** 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。

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,...]
```

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)
- 3 USB ディスク (usb)

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 がいない場合は、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* には、次のいずれかを指定する必要があります。

- 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。

--nameserver= 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。 --bootproto=static 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。

--nameserver 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: --nameserver="10.126.87.104[,10.126.87.120]"

--netmask= インストールしたシステムのサブネット マスクを 255.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。

--hostname= インストールしたシステムのホスト名を指定します。

--vlanid= vlanid システムの VLAN を指定します。--bootproto=dhcp オプションまたは --bootproto=static 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。

--addvmportgroup=(0|1) 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。

datastore name パーティションのマウント先を指定します。

--ondisk= or --ondrive= 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。

--firstdisk= 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。

disk-type1, 1 ローカルに接続されたストレージ (local)

[disk-type2, ...] 2 ネットワーク ストレージ (remote)

 3 USB ディスク (usb)

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。

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=` 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。

disk-type1,

[*disk-type2*, ...]

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)

3 USB ディスク (usb)

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。

%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す。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 3-8. 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_Blade000000004C5300 01171118101244	SCSI INQUIRY 識別子

表 3-8. ディスク デバイス名 (続き)

フォーマット	例	説明
VML	vml.00025261	レガシー VMkernel 識別子
MPX	mpx.vmhba0:C0:T0:L0	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。

ストレージ デバイスの名前と識別子

ESXi 環境では、各ストレージ デバイスは複数の名前と識別子で識別されます。

デバイス識別子

ストレージのタイプによって、ESXi ホストは異なるアルゴリズムと規則を使用して、ストレージ デバイスごとに識別子を生成します。

ストレージにより提供される識別子

ESXi ホストは、ターゲット ストレージ デバイスからデバイス名を照会します。返されたメタデータから、ホストはデバイスの一意の識別子を抽出または生成します。識別子は特定のストレージ標準に基づいており、すべてのホストで一貫かつ永続的なもので、次のいずれかの形式をとります。

- naa.xxx
- eui.xxx
- t10.xxx

バスベースの識別子

デバイスが識別子を提供しない場合、ホストは mpx を生成します。 *path* の名前。ここで *path* はたとえば mpx.vmhba1:C0:T1:L3 のようなデバイスの最初のパスを表します。この識別子は、ストレージに提供される識別子と同じように使用できます。

mpx.*path* 識別子は、パス名が一意であることを前提に、ローカル デバイス向けに作成されます。ただし、この識別子は一意でも永続的でもないため、システムを再起動した後に毎回変わる可能性があります。

通常、デバイスへのパスの形式は次の通りです。

`vmhbaAdapter:CChannel:TTarget:LLUN`

- *vmhbaAdapter* はストレージ アダプタの名前です。この名前は、仮想マシンで使用される SCSI コントローラではなく、ホストの物理アダプタを表します。
- *CChannel* はストレージ チャンネルの番号です。

ソフトウェア iSCSI アダプタと依存型ハードウェア アダプタは、チャンネル番号を使用して、同じターゲットへの複数のパスを表示します。

- *TTarget* はターゲットの番号です。ターゲットの番号はホストによって決定されますが、ホストに表示されるターゲットのマッピングが変わると、番号も変わることがあります。複数のホストが共有しているターゲットは、同じターゲット番号を持たないことがあります。

- LLUN は、ターゲット内の LUN の場所を表す、LUN の番号です。LUN 番号は、ストレージ システムによって提供されます。ターゲットに 1 つの LUN しかない場合、LUN 番号は常にゼロ (0) になります。

たとえば `vmhba1:C0:T3:L1` は、ストレージ アダプタ `vmhba1` とチャンネル 0 を介してアクセスするターゲット 3 上の LUN1 を表します。

レガシー識別子

device-provided identifiers または `mpx.path` 識別子の他に、ESXi は各デバイスの代替のレガシー名も生成します。識別子の形式は次のとおりです。

`vml.number`

レガシー識別子には、デバイスに一意の一連の数字が含まれており、識別子は、SCSI INQUIRY コマンドで取得されたメタデータから部分的に取得することができます。SCSI INQUIRY 識別子を提供しない非ローカル デバイスの場合は `vml.number` 識別子が唯一の使用可能な一意の識別子として使用されます。

例：vSphere CLI でデバイス名を表示

`esxcli storage core device list` コマンドを使用すると、vSphere CLI にすべてのデバイス名を表示できます。出力例は次のとおりです。

```
# esxcli storage core device list
naa.XXX
    Display Name: DGC Fibre Channel Disk(naa.XXX)
    ...
    Other UIDs: vml.000XXX
mpx.vmhba1:C0:T0:L0
    Display Name: Local VMware Disk (mpx.vmhba1:C0:T0:L0)
    ...
    Other UIDs: vml.0000000000XYZ
```

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するカーネル、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 ロードーを構成します。

表 3-9. boot.cfg に記述されるコマンド

コマンド	説明
title=STRING	ブート ロードーのタイトルを <i>STRING</i> に設定します。
prefix=STRING	(オプション) kernel= コマンドおよび modules= コマンド内の [/] または http:// で始まらないすべての FILEPATH の前に DIRPATH/ を追加します。
kernel=FILEPATH	カーネル パスを FILEPATH に設定します。
kernelopt=STRING	カーネル ブート オプションに STRING を追加します。
modules=FILEPATH1 --- FILEPATH2... --- FILEPATHn	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 #unique_12 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。 [インストール スクリプトおよびアップグレード スクリプトについて](#) 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。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 フラッシュ ドライブからの 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。 [インストール スクリプトおよびアップグレード スクリプトについて](#) 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。

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。

インストーラのネットワーク ブートによる、スクリプトを使用した ESXi のインストールまたはアップグレード

ESXi 7.0 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。

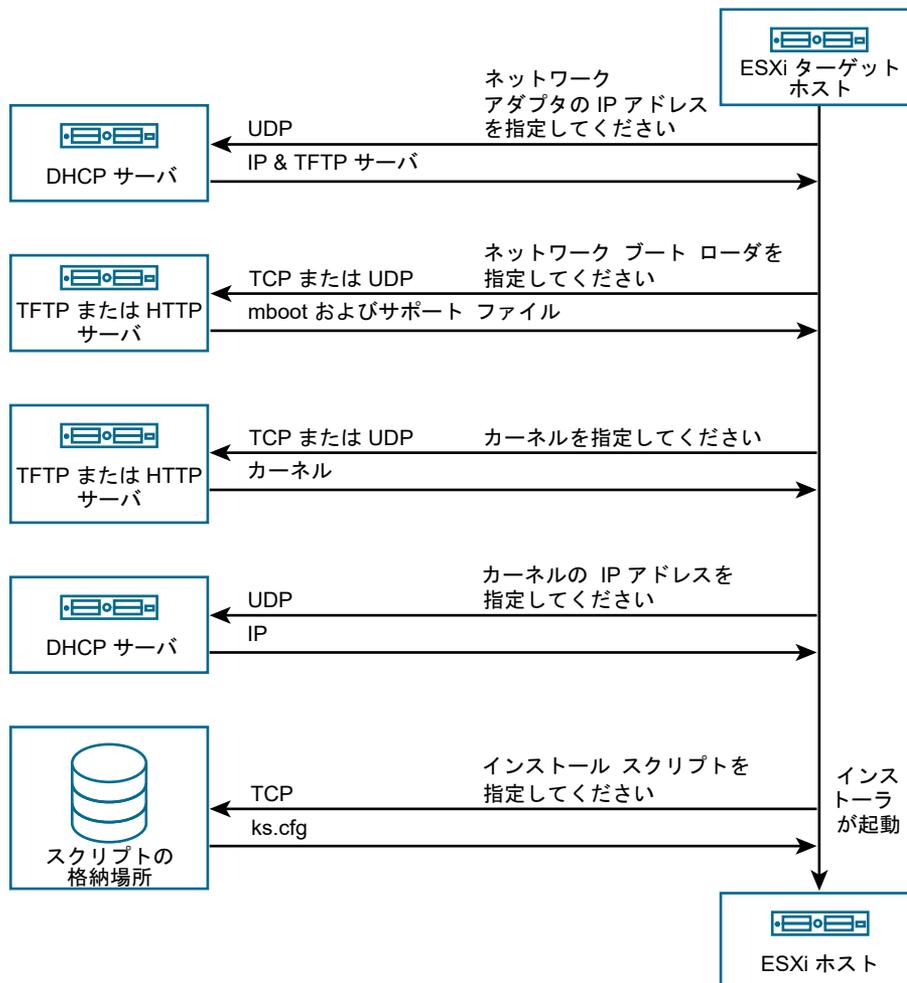
- ネットワーク インフラストラクチャの設定については、[#unique_12](#) を参照してください。
- インストール スクリプトの作成および配置については、[インストール スクリプトおよびアップグレード スクリプトについて](#) を参照してください。
- ESXi のインストーラをネットワーク ブートしインストール スクリプトを使用する具体的な手順については、次のトピックを参照してください。
 - [ネイティブ UEFI HTTP を使用した ESXi インストーラの起動](#)
 - [iPXE および HTTP を使用した ESXi インストーラの起動](#)
 - [PXE 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アップグレードを実行する方法については、[4 章 vSphere Auto Deploy を使用した、ホストの再プロビジョニング](#) を参照してください。

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 3-2. 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。

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。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。

- 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 ローダ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 ローダ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

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-7.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-7.x.x-XXXXXX/mboot.c32
  APPEND -c ESXi-7.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 ファイルをコピーし、ファイル名を mboot.efi に変更します。7.0 Update 3 以降の場合は、efi/boot/crypto64.efi ファイルも /tftpboot フォルダにコピーします。

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。

- 3 DHCP サーバを構成します。
- 4 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-7.x.x-xxxxx のようになります。
- 5 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。

6 boot.cfg ファイルを修正します。

- a 次の行を追加します。

```
prefix=ESXi-7.x.x-xxxxxxx
```

ESXi-7.x.x-xxxxxxx 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。

7 (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
```

8 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	<p>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。</p> <p>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。</p>

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。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。

- 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 ローダ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 ローダ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

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-7.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-7.x.x-XXXXXX/mboot.c32
  APPEND -c ESXi-7.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 default 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-mac_address_of_target_ESXi_host)。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 /tftpboot 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-7.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-7.x.x-XXXXXX
```

http://XXX.XXX.XXX.XXX/ESXi-7.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	/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	<p>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。</p> <p>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。</p>

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-7.x.x-XXXXXX` ディレクトリを作成します。
- 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。
- 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。

```
prefix=http://www.example.com/esxi/ESXi-7.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。

例：`kernelopt=ks=http://www.example.com/esxi_ksFiles/ks.cfg`

- 7 (オプション) ESXi 7.0 Update 2 以降では、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://192.168.30.6/esxi70uc1/efi/boot/bootx64.efi` など。

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	boot.cfg ファイルを mboot.efi と同じディレクトリに追加します。例： http://www.example.com/esxi/boot.cfg
異なるインストーラ	<p>a mboot.efi 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 (<i>mac_address_of_target_ESXi_host</i>) を名前として付けます。01-23-45-67-89-0a-bc のようになります。</p> <p>b カスタム boot.cfg ファイルをディレクトリに追加します。http://www.example.com/esxi/01-23-45-67-89-0a-bc/boot.cfg 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。

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と、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せん。ファームウェアは、UEFI システムに ESXi ブート ローダー 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 ローダ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 ローダ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 ローダーとして 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 ローダ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。

DHCP の構成例

ESXi インストーラをネットワーク ブートするには、DHCP サーバが TFTP または HTTP サーバのアドレスと初期ブート ロードのファイル名を 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 ロード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]:xxxx/snpnonly.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 snpnonly.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";  
}
```

ESXCLI コマンドを使用した、ホストのアップグレード

ESXCLI を使用して、ESXi 6.5 ホストまたは ESXi 6.7 ホストをバージョン 7.0 にアップグレードし、ESXi 6.5、ESXi 6.7、および ESXi 7.0 ホストに対してアップデートまたはパッチ適用を行うことができます。

vSphere 7.0 では、ESXi 7.0 ホストの更新またはパッチ適用に使用できる新しいソフトウェア配布物として、コンポーネント、基本イメージ、アドオンを提供します。ESXi でのコンポーネント、基本イメージ、アドオンの管理については、[ESXCLI Concepts and Examples](#) を参照してください。

ESXCLI コマンドを使用するには、スタンドアロンの ESXCLI をインストールする必要があります。ESXCLI のインストールおよび使用については、次のドキュメントを参照してください。

- [ESXCLI スタート ガイド](#)
- [ESXCLI のリファレンス](#)

注： `esxcli` コマンド実行中に Ctrl+C を押下すると、コマンドライン インターフェイスは、メッセージを表示せずに新しいプロンプトになります。ただし、コマンドは完了するまで実行を続けます。

vSphere Auto Deploy で導入した ESXi ホストでは、tools VIB は、最初の Auto Deploy インストールで使用するベース ブート イメージの一部である必要があります。tools VIB は後で別に追加できません。

VIB、イメージ プロファイル、およびソフトウェア デポ

`esxcli` コマンドを使用して ESXi をアップグレードするには、VIB、イメージ プロファイル、およびソフトウェア デポについて理解している必要があります。

vSphere ドキュメント セット全体のインストール作業およびアップグレード作業の説明で、次の技術用語が使用されます。

VIB

VIB は、ESXi のソフトウェア パッケージです。VMware およびパートナーは、ESXi プラットフォームを拡張するためのソリューション、ドライバ、CIM プロバイダ、およびアプリケーションをパッケージ化します。

VIB は、ソフトウェア デポで使用できます。VIB を使用して、ISO イメージを作成およびカスタマイズするほか、VIB をホストに非同期でインストールすることにより、ESXi ホストをアップグレードできます。

イメージ プロファイル

イメージ プロファイルは VIB で構成され、ESXi イメージを定義します。イメージ プロファイルには常に基本 VIB が含まれますが、ほかの VIB が含まれる場合もあります。イメージ プロファイルは、vSphere ESXi Image Builder を使用して調査および定義します。

ソフトウェア デポ

ソフトウェア デポは、VIB とイメージ プロファイルの集合体です。ソフトウェア デポはファイルとフォルダの階層であり、HTTP URL（オンラインのデポの場合）または ZIP ファイル（オフラインのデポの場合）を介して使用できます。VMware および VMware のパートナーがデポを提供します。大規模な VMware 環境を構築している企業では、内部にデポを作成し、vSphere Auto Deploy で ESXi ホストをプロビジョニングしたり、ESXi インストール用に ISO をエクスポートする場合があります。

VIB およびホストの許容レベルについて

各 VIB は、許容レベルとともにリリースされます。この許容レベルは変更することができません。ホストの許容レベルで、どの VIB をホストにインストールできるかが決まります。

許容レベルは、`esxcli software vib install` コマンドと `esxcli software vib update` コマンドを使用してインストールされた個々の VIB、vSphere Lifecycle Manager を使用してインストールされた VIB、およびイメージ プロファイル内の VIB に適用されます。

ホスト上のすべての 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、ホストの許容レベルが `VMwareAccepted` である場合は、許容レベルが `VMwareCertified` および `VMwareAccepted` の VIB をインストールできますが、許容レベルが `PartnerSupported` または `CommunitySupported` の VIB はインストールできません。許容レベルがホストの許容レベルよりも低くて限定的な VIB をインストールする場合は、vSphere Client を使用するか、`esxcli software acceptance` コマンドを実行して、ホストの設定を変更できます。

ホストにインストールしてイメージ プロファイルと共に使用できる VIB を指定したり、VIB に対して利用できるサポートのレベルを指定したりできるように、ホストの許容レベルを設定するのがベスト プラクティスです。たとえば、テスト環境のホストより限定的な許容レベルを本番環境のホストに設定できます。

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な、詳細なテストが行われます。現在は、I/O Vendor Program (IOVP) プログラム ドライバのみがこのレベルで公開されています。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 は、この許容レベルの VIB に対するサポート コールを、パートナーのサポート組織に送ります。

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

ロジックがこのレベルにあります。VMware は、この許容レベルの VIB に対するサポート コールを、パートナーのサポート組織に送ります。

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。

表 3-10. ホストへのインストールの要件となる VIB 許容レベル

ホストの許容レベル	VMwareCertified VIB	VMwareAccepted VIB	PartnerSupported VIB	CommunitySupported VIB
VMwareCertified	x			
VMwareAccepted	x	x		
PartnerSupported	x	x	x	
CommunitySupported	x	x	x	x

ホストの許容レベルとアップデートの許容レベルのマッチング

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変更することができます。ホスト上のすべての 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。

ここに記載されている手順を使用して、ホストの許容レベル、およびインストールする VIB またはイメージ プロファイルの許容レベルを確認し、必要に応じてホストの許容レベルを変更します。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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 に関する情報をリストします。	<code>esxcli --server=<server_name> software sources vib list --depot=<depot_URL></code>
指定した VIB に関する情報をリストします。	<code>esxcli --server=<server_name> software sources vib list --viburl=<vib_URL></code>

オプション	説明
すべてのイメージ プロファイルに関する情報をリストします。	<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をリストします。	<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取得します。

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

- 3 (オプション) VIB の許容レベルがホストの許容レベルよりも厳格な場合は、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 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: `esxcli software vib` または `esxcli software profile` コマンドの `--force` オプションを使用すると、許容レベルがホストより低い VIB またはイメージ プロファイルを追加することができます。その場合、警告が表示されます。設定の整合性がなくなったため、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。

更新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か、再起動する必要があるかの確認

ライブ インストールでインストール可能な VIB の場合、ホストの再起動は不要ですが、ホストをメンテナンス モードにすることが必要な場合があります。その他の VIB およびプロファイルの場合は、インストールまたは更新後にホストの再起動が必要な場合があります。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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 プロファイルをインストールする場合に、ホストをメンテナンス モードにする必要があるか、またインストールまたは更新後にホストを再起動する必要があるかを確認します。

次のいずれかのコマンドを実行します。

オプション	説明
VIB の確認	<pre>esxcli --server=<server_name> software sources vib get -v <absolute_path_to_vib></pre>
デポ内の VIB の確認	<pre>esxcli --server=<server_name> software sources vib get --depot=<depot_name></pre>
デポ内のイメージ プロファイルの確認	<pre>esxcli --server=<server_name> software sources profile get --depot=<depot_name></pre>

- 2 戻り値を確認します。

VIB メタデータから読み取られる戻り値は、VIB またはイメージ プロファイルをインストールする前にホストがメンテナンス モードになっている必要があるかどうか、および VIB またはプロファイルのインストールでホストを再起動する必要があるかどうかを示します。

注： vSphere Lifecycle Manager は、内部の ESXi ソフトウェア スキャン API を使用して、メンテナンス モードが必要かどうかを判断します。VIB をライブ システムにインストールする際に、Live-Install-Allowed の値が false に設定されている場合、vSphere Lifecycle Manager は、インストール結果に従ってホストを再起動します。VIB をライブ システムから削除する際に、Live-Remove-Allowed の値が false に設定されている場合、vSphere Lifecycle Manager は、その削除結果に従ってホストを再起動します。いずれの場合も、修正が開始されると、vSphere Lifecycle Manager が自動的にホストをメンテナンス モードに移行します。

次のステップ

必要に応じ、ホストをメンテナンス モードにします。[ホストをメンテナンス モードに切り替える](#)を参照してください。ホストの再起動が必要で、そのホストが vSphere HA クラスタに属している場合は、インストールまたは更新の前にホストをクラスタから削除するか、クラスタで HA を無効にします。また、ホストをメンテナンス モードに移行すると、アップグレード中の起動ディスクのアクティビティが最小限に抑えられます。

ホストをメンテナンス モードに切り替える

ライブ インストールを使用するインストールおよびアップデート操作の一部では、ホストをメンテナンス モードにする必要があります。

アップデート操作で再起動が必要な場合は、メンテナンス モードにする必要があります。ただし、アップデートおよびアップグレード操作に esxcli コマンドを使用する場合は、ホストを手動でメンテナンス モードにするだけです。

アップグレード操作でホストをメンテナンス モードにする必要があるかどうかを判断するには、[更新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か、再起動する必要があるかの確認](#)を参照してください。

注： ホストが vSAN クラスターのメンバーであり、そのホスト上の任意の仮想マシン オブジェクトがストレージ ポリシーの「許容する障害の数 =0」設定を使用している場合、そのホストがメンテナンス モードに入るときに異常な遅延が発生する可能性があります。この遅延は、vSAN が、メンテナンス操作の正常な完了のため、このオブジェクトをホストから退避させる必要があるために発生します。

--server=<server_name> を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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 ホストがメンテナンス モードであるかどうかを確認します。

```
esxcli --server=<server_name> system maintenanceMode get
```

- 2 ESXi ホストで実行されている各仮想マシンをパワーオフします。

注： 次のコマンドを実行して、実行中のすべての仮想マシンを一覧表示し、各仮想マシンのワールド ID を取得できます。

```
esxcli --server=<server_name> vm process list
```

オプション	コマンド
ゲスト OS をシャットダウンしてから、仮想マシンをパワーオフします	<code>esxcli --server=<server_name> vm process kill --type soft --world-id <vm_ID></code>
仮想マシンをすぐにパワーオフします	<code>esxcli --server=<server_name> vm process kill --type hard --world-id <vm_ID></code>
強制的にパワーオフ操作を実行します	<code>esxcli --server=<server_name> vm process kill --type force --world-id <vm_ID></code>

または、仮想マシンのパワーオフを避けるために他のホストに移行できます。『vCenter Server およびホストの管理』ドキュメントのトピック「[Migrating Virtual Machines](#)」を参照してください。

- 3 ホストをメンテナンス モードにします。

```
esxcli --server=<server_name> system maintenanceMode set --enable true
```

- 4 ホストがメンテナンス モードであることを確認します。

```
esxcli --server=<server_name> system maintenanceMode get
```

個々の VIB によるホストの更新

URL を指定してアクセスできるソフトウェア デポかオフラインの ZIP デポに格納されている VIB を使用して、ホストを更新できます。

重要： VMware 提供のデポで zip バンドルから ESXi をアップデートする場合、VMware Web サイトからオンラインまたはローカルでダウンロードするかのどちらかで、VMware は、[トピックイメージ プロファイルによるホストのアップグレードまたはアップデート](#) で VMware 提供のデポに対して指定されたアップデート方法のみをサポートします。

注： アップグレード操作では、`esxcli software vib update` コマンドと `esxcli software vib install` コマンドはサポートされていません。[イメージ プロファイルによるホストのアップグレードまたはアップデート](#) を参照してください。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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。
- 更新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か、それとも再起動する必要があるかを確認します。必要に応じ、ホストをメンテナンス モードにします。
[更新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か、再起動する必要があるかの確認](#)を参照してください。[ホストをメンテナンス モードに切り替える](#)を参照してください。
- 更新時に再起動が必要な場合、ホストが vSphere HA クラスタに属しているときは、クラスタからホストを削除するか、クラスタで HA を無効にします。

手順

- 1 ホストにどの VIB がインストールされているか確認します。

```
esxcli --server=<server_name> software vib list
```

2 デポで利用可能な VIB を見つけます。

オプション	説明
URL を指定してアクセスできるデポから	<pre>esxcli --server=<server_name> software sources vib list -- depot=http://<web_server>/<depot_name></pre>
ローカルのデポの ZIP ファイルから	<pre>esxcli --server=<server_name> software sources vib list -- depot=<absolute_path_to_depot_zip_file></pre>

--proxy オプションを使用して、プロキシ サーバを指定することができます。

3 デポにある VIB を含めるか、新しい VIB をインストールするように、既存の VIB を更新します。

オプション	説明
URL を指定してアクセスできるデポから VIB を更新	<pre>esxcli --server=<server_name> software vib update -- depot=http://<web_server>/<depot_name></pre>
ローカルにある ZIP ファイルから VIB を更新	<pre>esxcli --server=<server_name> software vib update -- depot=<absolute_path_to_depot_ZIP_file></pre>
指定したオフラインのデポにある ZIP ファイルから、すべての VIB をインストール (VMware の VIB とパートナー提供の VIB の両方を含む)	<pre>esxcli --server=<server_name> software vib install -- depot <path_to_VMware_vib_ZIP_file>\<VMware_vib_ZIP_file> --depot <path_to_partner_vib_ZIP_file>\<partner_vib_ZIP_file></pre>

update コマンドおよび install コマンドのオプションを使用して、ドライ ラン、特定の VIB の指定、許容レベル検証のバイパスなどを行うことができます。本番システムでは検証をバイパスしないでください。ESXCLI リファレンスを参照してください。

4 VIB が ESXi ホストにインストールされていることを確認します。

```
esxcli --server=<server_name> software vib list
```

イメージ プロファイルによるホストのアップグレードまたはアップデート

URL を指定してアクセスできるソフトウェア デポかオフラインの ZIP デポに格納されているイメージ プロファイルを使用して、ホストをアップグレードまたはアップデートできます。

esxcli software profile update または esxcli software profile install コマンドを使用して、ESXi ホストをアップグレードまたはアップデートできます。

ホストをアップグレードまたはアップデートする場合は、コマンド **esxcli software profile update** または **esxcli software profile install** により、最新バージョン (メジャーまたはマイナー) のすべてのイメージ プロファイルをホストに適用します。この操作と再起動の後、ホストは同じバージョンかそれ以降のバージョンの vCenter Server 環境に参加することができます。

esxcli software profile update コマンドにより、ESXi ホスト イメージのコンテンツ全体を、ISO インストーラを使用してアップグレードする場合と同じレベルに引き上げます。ただし、ISO インストーラは、メモリ不足やサポートされていないデバイスなどの潜在的な問題について、アップグレード前のチェックを実行します。

esxcli のアップグレードでは、ESXi 6.7 Update 1 以降を新しいバージョンにアップグレードするときのみ、このようなチェックが実行されます。

ESXi のアップグレードのプロセスと方法の詳細については、「[ESXi ホスト アップグレード プロセスの概要](#)」を参照してください。

重要： VMware Web サイトからオンラインで実行またはローカルにダウンロードできる、VMware が提供するデポの zip バンドルを使用して ESXi をアップグレードまたはアップデートする場合、VMware はアップデート コマンド `esxcli software profile update --depot=<depot_location> --profile=<profile_name>` のみをサポートします。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

注： `update` および `install` コマンドのオプションを使用すると、ドライ ランの実行や、許容レベルの検証をバイパスできるほか、ハードウェア互換性チェックの警告を無視できるようになります。ハードウェア互換性チェックの警告をバイパスするオプションは、ESXi6.7 Update 1 以降のみで使用可能です。本番稼働システムでは検証をバイパスしないでください。

オプションのヘルプを参照するには、`esxcli software profile install --help` または `esxcli software profile update --help` と入力します。使用可能なコマンドライン オプションの完全なリストについては、[ESXCLI Reference](#) を参照してください。

前提条件

- スタンドアローン 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。
- 更新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か、それとも再起動する必要があるかを確認します。必要に応じ、ホストをメンテナンス モードにします。

更新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か、再起動する必要があるかの確認を参照してください。ホストをメンテナンス モードに切り替えるを参照してください。

重要： ESXCLI を使用してホストをアップデートまたはアップグレードする場合は、アップグレードを開始する前に起動ディスクがアクティブに使用されないようにするために、ホストを手動でメンテナンス モードにします。

- 更新時に再起動が必要な場合、ホストが vSphere HA クラスタに属しているときは、クラスタからホストを削除するか、クラスタで HA を無効にします。

手順

- 1 ホストにどの VIB がインストールされているか確認します。

```
esxcli --server=<server_name> software vib list
```

- 2 デポで提供されているイメージ プロファイルを確認します。

```
esxcli --server=<server_name> software sources profile list --depot=http://<web_server>/<depot_name>
```

--proxy オプションを使用して、プロキシ サーバを指定することができます。

- 3 既存のイメージ プロファイルを更新して、VIB を含めるか、新しい VIB をインストールします。

重要： software profile update コマンドでは、指定したプロファイルの対応する VIB を使用して既存の VIB を更新しますが、ターゲット サーバ上にインストールされているほかの VIB には影響しません。software profile install コマンドでは、デポのイメージ プロファイル内にある VIB をインストールし、ターゲット サーバ上にインストールされているその他すべての VIB を削除します。

オプション	説明
VMware Web サイトからアクセスし、ローカルのデポへダウンロード可能な、VMware が提供するデポの ZIP バンドルを使用して、イメージ プロファイルを更新します。	<pre>esxcli software profile update --depot=<depot_location> --profile=<profile_name></pre> <p>重要： VMware が提供する ZIP バンドルを使用する場合、これが VMware がサポートする唯一のアップデート方法です。</p> <p>VMware 提供の ZIP バンドル名は、次の形式になります。VMware-ESXi-<version_number>-<build_number>-depot.zip</p> <p>VMware 提供の ZIP バンドルのプロファイル名は、次のうちのどちらかの形式になります。。</p> <ul style="list-style-type: none"> ■ ESXi-<version_number>-<build_number>-standard ■ ESXi-<version_number>-<build_number>-notools (VMware Tools を含まない場合)
デポの URL を指定してイメージ プロファイルを更新	<pre>esxcli --server=<server_name> software profile update --depot=http://<web_server>/<depot_name> --profile=<profile_name></pre>
ターゲット サーバ上にローカルに格納されている ZIP ファイルから、イメージ プロファイルを更新	<pre>esxcli --server=<server_name> software profile update --depot=file:///<path_to_profile_ZIP_file>/<profile_ZIP_file> --profile=<profile_name></pre>
データストアにコピーされた、ターゲット サーバ上の ZIP ファイルからイメージ プロファイルを更新	<pre>esxcli --server=<server_name> software profile update --depot=<datastore_name>/<profile_ZIP_file> --profile=<profile_name></pre>
ローカルにコピーされ、ターゲット サーバに適用された ZIP ファイルから、イメージ プロファイルを更新	<pre>esxcli --server=<server_name> software profile update --depot=/<root_dir>/<path_to_profile_ZIP_file>/<profile_ZIP_file> --profile=<profile_name></pre>

オプション	説明
特定のプロファイルの URL を指定して、このプロファイルに新しい VIB をすべてインストール	<pre>esxcli --server=<server_name> software profile install -- depot=http://<web_server>/<depot_name> -- profile=<profile_name></pre>
ターゲット上でローカルに格納されている ZIP ファイルの特定のプロファイル内に、新しい VIB をすべてインストール。	<pre>esxcli --server=<server_name> software profile install -- depot=file:///<path_to_profile_ZIP_file>/ <profile_ZIP_file> --profile=<profile_name></pre>
ターゲット サーバ上の、データストアにコピーされた ZIP ファイルから、すべての新しい VIB をインストール	<pre>esxcli --server=<server_name> software profile install -- depot=<datastore_name>/<profile_ZIP_file> -- profile=<profile_name></pre>
ローカルにコピーされ、ターゲット サーバに適用された ZIP ファイルから、すべての新しい VIB をインストール	<pre>esxcli --server=<server_name> software profile install -- depot=/<root_dir>/<path_to_profile_ZIP_file>/ <profile_ZIP_file> --profile=<profile_name></pre>

4 VIB が ESXi ホストにインストールされていることを確認します。

```
esxcli --server=<server_name> software vib list
```

Zip ファイルを使用した ESXi ホストのアップデート

デポの Zip ファイルをダウンロードして、VIB またはイメージ プロファイルのホストをアップデートできます。

VMware パートナーは、管理エージェントまたは別途リリースされたドライバを提供するために、サードパーティ製 VIB を用意しています。

重要： VMware 提供のデポで zip バンドルから ESXi をアップデートする場合、VMware Web サイトからオンラインまたはローカルでダウンロードするかのどちらかで、VMware は、トピック**イメージ プロファイルによるホストのアップグレードまたはアップデート**で VMware 提供のデポに対して指定されたアップデート方法のみをサポートします。

アップグレード操作では、`esxcli software vib update` コマンドと `esxcli software vib install` コマンドはサポートされていません。イメージ プロファイルによるホストのアップグレードまたはアップデートを参照してください。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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。
- サードパーティ VMware パートナーから、デポ バンドルの ZIP ファイルをダウンロードします。

- 更新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か、それとも再起動する必要があるかを確認します。必要に応じ、ホストをメンテナンス モードにします。

更新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か、再起動する必要があるかの確認を参照してください。ホストをメンテナンス モードに切り替えるを参照してください。

- 更新時に再起動が必要な場合、ホストが vSphere HA クラスタに属しているときは、クラスタからホストを削除するか、クラスタで HA を無効にします。

手順

- ◆ ZIP ファイルをインストールします。

```
esxcli --server=<server_name> software vib update --depot=/<path_to_vib_ZIP>/  
<ZIP_file_name>.zip
```

ホストからの VIB の削除

サードパーティ製の VIB や VMware VIB を ESXi ホストからアンインストールすることができます。

VMware パートナーは、管理エージェントまたは別途リリースされたドライバを提供するために、サードパーティ製 VIB を用意しています。

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。

前提条件

- 削除後に再起動が必要で、ホストが vSphere HA クラスタに属する場合は、そのホストに対して高可用性を無効にします。
- 更新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か、それとも再起動する必要があるかを確認します。必要に応じ、ホストをメンテナンス モードにします。

更新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か、再起動する必要があるかの確認を参照してください。ホストをメンテナンス モードに切り替えるを参照してください。

重要： ESXCLI を使用してホストをアップデートまたはアップグレードする際に、起動ディスクがアクティブに使用されないようにするには、ホストを手動でメンテナンス モードにします。

- 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。

手順

- 1 ESXi ホストで実行されている各仮想マシンをパワーオフします。

注： 次のコマンドを実行して、実行中のすべての仮想マシンを一覧表示し、各仮想マシンのワールド ID を取得できます。

```
esxcli --server=<server_name> vm process list
```

オプション	コマンド
ゲスト OS をシャットダウンしてから、仮想マシンをパワーオフします	<pre>esxcli --server=<server_name> vm process kill --type soft --world-id <vm_ID></pre>
仮想マシンをすぐにパワーオフします	<pre>esxcli --server=<server_name> vm process kill --type hard --world-id <vm_ID></pre>
強制的にパワーオフ操作を実行します	<pre>esxcli --server=<server_name> vm process kill --type force --world-id <vm_ID></pre>

または、仮想マシンのパワーオフを避けるために他のホストに移行できます。『vCenter Server およびホストの管理』ドキュメントのトピック「[Migrating Virtual Machines](#)」を参照してください。

- 2 ホストをメンテナンス モードにします。

```
esxcli --server=<server_name> system maintenanceMode set --enable true
```

- 3 必要に応じ、仮想マシンをシャットダウンするか移行するかします。
- 4 ホストにどの VIB がインストールされているか確認します。

```
esxcli --server=<server_name> software vib list
```

- 5 VIB を削除します。

```
esxcli --server=<server_name> software vib remove --vibName=<name>
```

次のいずれかの形式で、1 つ以上の VIB を指定します。

- <name>
- <name>:<version>
- <vendor>:<name>
- <vendor>:<name>:<version>

たとえば、ベンダーが指定した VIB を削除するコマンドでは、名前とバージョンを次の形式で指定できます。

```
esxcli --server myEsxiHost software vib remove --vibName=PatchVendor:patch42:version3
```

注： remove コマンドでは、ほかにもいくつかのオプションがサポートされます。『ESXCLI のリファレンス』を参照してください。

ESXCLI コマンドを使用した、ホストへのサードパーティ製拡張機能の追加

esxcli software vib コマンドを使用して、VIB パッケージとしてリリースされたサードパーティの拡張機能をシステムに追加できます。このコマンドを使用すると、VIB システムによりファイアウォール ルール セットが更新され、システムの再起動後にホスト デモンが更新されます。

ファイアウォール構成ファイルを使用して、その拡張機能用に有効にするホスト サービスのポート ルールを指定することもできます。『vSphere のセキュリティ』には、ファイアウォール ルール セットを追加、適用、および更新する方法と、esxcli network firewall コマンドのリストが記載されています。

ESXCLI インストールまたはアップグレードのドライ ランの実行

--dry-run 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操作の結果を事前に確認することができます。インストールまたはアップグレード手順のドライ ランでは、変更は一切行われませんが、--dry-run オプションなしでコマンドを実行した場合、実行される VIB レベルでの操作についてレポートされます。

--server=<server_name> を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで esxcli --help を実行します。

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 インストールまたはアップグレードのコマンドを、--dry-run オプションを追加して入力します。

- `esxcli --server=<server_name> software vib install --dry-run`

- `esxcli --server=<server_name> software vib update --dry-run`

- `esxcli --server=<server_name> software profile install --dry-run`

- `esxcli --server=<server_name> software profile update --dry-run`

2 返される出力内容を確認します。

出力では、どの VIB がインストールまたは削除されるか、およびインストールまたはアップグレードで再起動が必要かどうかを示されます。

次回ホストを再起動したあとでアクティブになるインストール済み VIB およびプロファイルの表示

`--rebooting-image` オプションを指定すると、ホストにインストールされ、次回ホストを再起動したあとでアクティブになる、VIB およびプロファイルがリストされます。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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 次のいずれかのコマンドを入力します。

オプション	説明
VIB の場合	<code>esxcli --server=<server_name> software vib list --rebooting-image</code>
プロファイルの場合	<code>esxcli --server=<server_name> software profile get --rebooting-image</code>

2 返される出力内容を確認します。

出力には、次回再起動した後にアクティブになる ESXi イメージの情報が示されます。再起動が保留中のイメージが作成されていない場合は、出力に何も表示されません。

ホストのイメージ プロファイルと許容レベルの表示

`software profile get` コマンドを使用すると、指定したホストに現在インストールされているイメージ プロファイルおよび許容レベルを表示できます。

このコマンドは、プロファイルの変更内容を含め、インストールされたイメージ プロファイル履歴の詳細も表示します。

`--server=<server_name>` を使用してターゲット サーバを指定すると、サーバはユーザー名とパスワードの入力を促します。構成ファイルやセッション ファイルなど、他の接続オプションもサポートされています。接続オプションのリストについては、[ESXCLI スタート ガイド](#)を参照するか、ESXCLI コマンド プロンプトで `esxcli --help` を実行します。

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 次のコマンドを入力します。

```
esxcli --server=<server_name> software profile get
```

- 2 出力内容を確認します。

ESXi ホストをアップグレードした後に行う作業

ホストのアップグレードを完了するには、管理を行う vCenter Server システムにホストが再接続され、必要に応じてホストが再構成されていることを確認します。また、ホストが適切にライセンス供与されていることも確認します。

ESXi ホストをアップグレードした後は、次の操作を実行します。

- アップグレード ログを確認します。ログ ファイルは、vSphere Client を使用してエクスポートできます。
- vCenter Server システムがそのホストを管理している場合、vCenter Server のインベントリでホストを右クリックして [接続] を選択することで、ホストを vCenter Server に再接続する必要があります。
- アップグレードが完了すると、ESXi ホストは評価モードになります。評価期間は 60 日間です。評価期間が終了する前に、vSphere7.0 ライセンスを割り当てる必要があります。既存のライセンスをアップグレードするか、My VMware から新規ライセンスを取得できます。vSphere Client を使用して、環境内のホストのライセンスを構成します。vSphere のライセンス管理の詳細については、『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。
- ホストの sdX デバイスはアップグレード後に番号が変わる可能性があります。必要に応じて、sdX デバイスを参照するスクリプトをアップデートします。
- ホスト上の仮想マシンをアップグレードします。[仮想マシンと VMware Tools のアップグレード](#)を参照してください。
- vSphere Authentication Proxy サービスをセットアップします。以前のバージョンの vSphere Authentication Proxy に vSphere7.0 との互換性はありません。vSphere Authentication Proxy サービスの構成の詳細については、『vSphere セキュリティ』ドキュメントを参照してください。

ESXi 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。

アップグレード後の ESXi ホストへのライセンス供与

ESXi 7.0 にアップグレードした後で、vSphere 7 ライセンスを適用する必要があります。

ESXi ホストを同じ番号で始まるバージョンにアップグレードする場合は、既存のライセンスを新しいライセンスで置き換える必要はありません。たとえば、ESXi 6.5 から 6.7 にホストをアップグレードする場合、ホストで同じライセンスを使用できます。

ESXi ホストを異なる番号で始まるバージョンにアップグレードする場合は、新しいライセンスを適用する必要があります。たとえば、ESXi ホストを 6.x から 7.0 にアップグレードする場合、ホストに vSphere 7 のライセンスを供与する必要があります。

ESXi 6.5 または ESXi 6.7 のホストを ESXi 7.0 ホストにアップグレードすると、正しい vSphere 7 ライセンスを適用するまで、ホストは 60 日間の評価モード期間に入ります。[ESXi の評価モードとライセンス モードについて](#)を参照してください。

My VMware から vSphere 7 ライセンスを取得することができます。vSphere 7 ライセンスを取得したら、vSphere Client のライセンス管理機能を使用して、アップグレードされた ESXi 7.0 のすべてのホストにライセンスを割り当てる必要があります。詳細については、ドキュメント『vCenter Server およびホストの管理』を参照してください。スクリプト化された方法を使用して ESXi 7.0 へのアップグレードを行う場合、キックスタート (ks) ファイルにライセンス キーを設定できます。

アップグレード後の ESXi ホストでのセキュア ブート検証スクリプトの実行

UEFI セキュア ブートをサポートしていない ESXi の以前のバージョンから ESXi ホストをアップグレードした後は、セキュア ブートを有効にできる場合があります。セキュア ブートを有効にできるかどうかは、アップグレードの実行方法と、アップグレードによってすべての既存の VIB が置換されたか、一部の VIB が変更されないまま残されたかによって異なります。アップグレード後に検証スクリプトを実行して、アップグレード後のインストールがセキュア ブートをサポートするかどうかを判断できます。

セキュア ブートを正常に行うためには、インストールされているすべての VIB の署名がシステムで使用できる必要があります。ESXi の以前のバージョンは、VIB のインストール時に署名を保存できません。

- esxcli コマンドを使用してアップグレードすると、古いバージョンの ESXi は新しい VIB のインストールを実行するため、署名が保存されず、セキュア ブートは実行できません。
- ISO を使用してアップグレードすると、新しい VIB は署名を保存できます。これは、ISO を使用した vSphere Lifecycle Manager のアップグレードにもあてはまります。
- 以前の VIB がシステムに残っている場合、それらの VIB の署名は使用できず、セキュア ブートは実行できません。
 - システムがサードパーティ製ドライバを使用しており、VMware のアップグレードにドライバ VIB の新しいバージョンが含まれていない場合、アップグレード後に以前のバージョンの VIB がシステムに残ります。
 - まれに、VMware は特定の VIB の開発を継続せず、古い VIB を置き換える新しい VIB を提供しない場合があります。その際は、アップグレード後に古い VIB がシステムに残ることがあります。

注： UEFI セキュア ブートには、最新のブートローダーも必要です。このスクリプトは、最新のブートローダーをチェックしません。

前提条件

- ハードウェアで UEFI セキュア ブートがサポートされることを確認します。
- すべての VIB が、最低でも許容レベル PartnerSupported で署名されていることを確認します。CommunitySupported レベルの VIB を含めると、セキュア ブートを使用できません。

手順

- 1 ESXi をアップグレードして、次のコマンドを実行します。

```
/usr/lib/vmware/secureboot/bin/secureBoot.py -c
```

- 2 出力を確認します。

Secure boot can be enabled または Secure boot CANNOT be enabled のいずれかが出力されます。

システム ログに必要な空き容量

Auto Deploy を使用して ESXi7.0 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す。

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 ボリュームのスクラッチ 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、ESXi7.0 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 3-11. `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	維持するローテーション数	必要な最小ディスク容量
管理エージェント (<code>hostd</code>)	10 MB	10	100 MB
vCenter Server エージェント (<code>vpxa</code>)	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 (<code>fdm</code>))	5 MB	10	50 MB

Syslog および Syslog サーバの設定と構成、および vSphere Syslog Collector のインストールについては、『vCenter Server のインストールとセットアップ』ドキュメント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。

`esxcli system syslog` コマンドや他の ESXCLI コマンドの使用の詳細については、『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。

手順

- 1 vSphere Client インベントリで、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。

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： `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 3-12. 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 3-12. 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 3-13. 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 3-13. 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>	監査記録 ストレージ ディレクトリの場所を指定します。監査記録 ストレージが有効になっている間は、監査記録 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 ホストで監査記録のストレージを有効にします。監査記録 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 3-13. 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 logfile [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 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 3-13. 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。 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。
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 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す。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。

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 デーモン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 フィルタを作成する場合は、ログ メッセージのログ エントリの最大数を設定します。ログ メッセージは、1つ以上の指定されたシステム コンポーネントにより生成され、指定された語句に一致します。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。

重要： ログ情報の量に制限を設定すると、潜在的なシステム障害を適切にトラブルシューティングする能力が制限されます。最大ログ エントリ数に達した後にログ ローテーションが発生すると、フィルタリングされたメッセージのすべてのインスタンスが失われる可能性があります。

手順

- 1 ESXi シェルに root としてログインします。
- 2 /etc/vmware/logfilters ファイルで、次のエントリを追加して、ログ フィルタを作成します。

```
numLogs | ident | logRegexp
```

説明：

- *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。0 を使用すると、指定したすべてのログ 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。ログ メッセージを生成するシステム コンポーネントについては、syslog 構成ファイルの *idents* パラメータの値を参照してください。これらのファイルは `/etc/vmsyslog.conf.d` ディレクトリ内にあります。複数のシステム コンポーネントにフィルタを適用するには、コンマ区切りのリストを使用します。すべてのシステム コンポーネントにフィルタを適用するには、* を使用します。
- *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、コンテンツでログ メッセージをフィルタリングします。

たとえば、`SOCKET connect failed, error 2: No such file or directory` のような語句でエラー番号がその都度変わるメッセージを対象とする場合、`hostd` コンポーネントからの最大ログ エントリ数の制限を 2 に設定するには、次のエントリを追加します。

```
2 | hostd | SOCKET connect failed, error .*: No such file or directory
```

注： # で始まる行はコメントを示しており、行の残りの部分は無視されます。

- 3 `/etc/vmsyslog.conf` ファイルで、次のエントリを追加して、ログ フィルタリング機能を有効にします。

```
enable_logfilters = true
```

- 4 `esxcli system syslog reload` コマンドを実行して、syslog デーモンを再ロードし、構成の変更を適用します。

vSphere Auto Deploy を使用した、ホストの再プロビジョニング

4

vSphere Auto Deploy を使用してホストをデプロイした場合は、vSphere Auto Deploy を使用して、異なるバージョンの ESXi を含む新しいイメージ プロファイルでホストを再プロビジョニングできます。イメージ プロファイルは、vSphere ESXi Image Builder を使用して作成および管理できます。

注： ESXi 6.0 以降のイメージを使用するようにホストをアップグレードする場合、vSphere Auto Deploy サーバは、VMware 認証局 (VMCA) による署名付きの証明書を使用して ESXi ホストをプロビジョニングします。現在カスタム証明書を使用している場合は、アップグレード後にカスタム証明書を使用するようにホストを設定できます。vSphere のセキュリティ を参照してください。

vSphere Auto Deploy サーバは、対応する vCenter Server システムがアップグレードされると、自動的にアップグレードされます。バージョン 6.0 以降では、vSphere Auto Deploy サーバは常に vCenter Server システムと同じ管理ノード上に置かれます。

この章には、次のトピックが含まれています。

- vSphere Auto Deploy の概要
- vSphere Auto Deploy の準備
- ホストの再プロビジョニング

vSphere Auto Deploy の概要

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 のホスト プロファイル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。

ESXi ホストの状態情報

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。イメージ プロファイルおよびホスト プロファイル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に最初に指定されます。

表 4-1. 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 で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決める構成設定 (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 プロファイルのユーザー インターフェイスを使用して作成されたホスト プロファイル。通常はテンプレート ホストから取得されます。
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

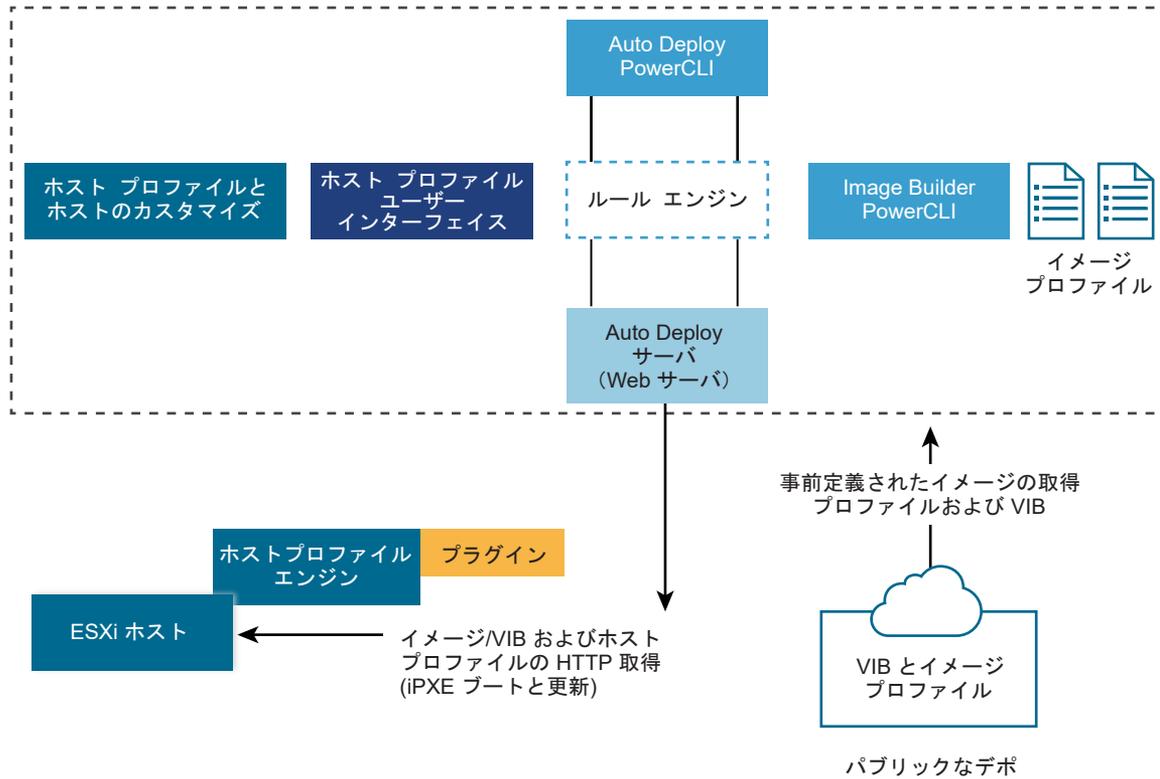
vSphere Auto Deploy のインフラストラクチャは、いくつかのコンポーネントで構成されています。

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 4-1. 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。

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。
- VMware カスタマは、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。

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。

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。

vSphere Auto Deploy の準備

vSphere Auto Deploy の使用を開始する前に、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。

■ 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 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。
『#unique_76』と『#unique_77』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。必要であれば、vSphere Syslog Collector をインストールして使用します。これは vCenter Server サポート ツールであり、システム ログの記録のための統合アーキテクチャを提供し、ネットワーク ログと、複数のホストからのログの結合を有効にします。
- 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。
- 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。
- (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。
- (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-vmware" $licenseKeyEntry.LicenseKey = "XXXXX-XXXXX-XXXXX-XXXXX-XXXXX"
```

- 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 XXXXX-XXXXX-XXXXX-XXXXX-XXXXX 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。

ホストの再プロビジョニング

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 のインストールとセットアップ』の「vSphere Auto Deploy の準備」のトピック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

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。

前提条件

- 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。
- 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、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、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 を使用してホストに対応するルールを変更し、コンプライアンスのテストおよび修復操作を実行することで、PowerCLI セッションで新しいイメージ 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。

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` 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があります。
- 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 のインストールとセットアップ』の「vSphere ESXi Image Builder CLI の使用」のトピック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	<code>Add-EsxSoftwareDepot 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 バ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 <code>Add-EsxSoftwareDepot C:\file_path\my_offline_depot.zip</code> 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_imageprofilemy_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。

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。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。

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

- 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 のインストールとセットアップ』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。

ルールのコンプライアンスのテストおよび修復

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。

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
-----	-----
<i>My Profile 25</i>	<i>MyNewProfile</i>

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。

ESXi ホストのトラブルシューティング用のログの収集

5

ESXi のインストール ログ ファイルまたはアップグレード ログ ファイルを収集できます。インストールまたはアップグレードが失敗した場合、ログ ファイルを確認することで、失敗の原因を特定できる可能性があります。

解決方法

- 1 ESXi Shell で、または SSH を介して、`vm-support` コマンドを入力します。
- 2 `/var/tmp/` ディレクトリに移動します。
- 3 `.tgz` ファイルからログ ファイルを取得します。