



Cisco Elastic Services Controller のアップグレード

Cisco Elastic Service Controller は、次の 2 種類のアップグレードをサポートしています。

- **Backup and Restore アップグレード** : このアップグレードプロセスでは、ESC キープアライブデーモンの停止 (ESC HA の場合)、データベースのバックアップ、ESC インスタンスの停止と名前変更 (または削除)、ESC インスタンスの再インストール、データベースの復元が行われます。ESC 5.8 アップグレードでサポートされる ESC バージョンについては、次の表を参照してください。
- **インサービスアップグレード** : ESC は、最小限のダウンタイムで、アクティブ/スタンバイ高可用性ノードのインサービスアップグレードをサポートします。

ESC インスタンスは、スタンドアロンインスタンスとして、または高可用性ペアとしてアップグレードできます。アップグレード手順は、スタンドアロンの場合と高可用性ペアの場合で異なります。

この章では、ESC スタンドアロンインスタンスと ESC 高可用性インスタンスをアップグレードする方法について、それぞれの手順を示します。ESC インスタンスのアップグレードを決定する前に、次の手順を確認する必要があります。[インストールのシナリオ](#) を参照し、インストールシナリオの詳細を確認してください。

- ESC は、先行する 2 つのマイナーリリースからの直接アップグレードのみをサポートします。たとえば ESC 5.8 は、ESC 5.7 と ESC 5.6 からの直接アップグレードをサポートしません。直接アップグレードでサポートされているバージョンよりも古いリリースの場合は、段階的なアップグレードを実行する必要があります。

(注) ESC メジャーリリースの例 : ESC 3.x、ESC 4.x、ESC 5.x

ESC マイナーリリースの例 : ESC 5.1.x、ESC 5.4.x

ESC メンテナンスリリースの例 : ESC 5.4.0.x、5.4.1.x

ESC パッチリリースの例 : ESC 5.4.0.88、ESC 5.4.0.89、ESC 5.4.0.100

- ESC をアップグレードするには、ESC のインストールプロセスに精通している必要があります。

- OpenStack については、OpenStack のインストール手順を参照してください。第 4 章「OpenStack への Cisco Elastic Services Controller のインストール」を参照してください。
- VMware については、VMware のインストール手順を参照してください。第 7 章「Cisco Elastic Services Controller の VMware vCenter へのインストール」を参照してください。
- ESC HA については、ESC HA のインストール手順を参照してください。第 5 章「OpenStack での高可用性の設定」と第 8 章「VMware での高可用性の設定」を参照してください。

表 1: ESC 5.8 へのアップグレードでサポートされる ESC バージョン

仮想インフラストラクチャ マネージャ	Backup and Restore アップグレードでサポートされるバージョン	インサービスアップグレードでサポートされるバージョン
OpenStack	5.7、5.6	5.7、5.6
VMware	5.7、5.6	5.7、5.6
CSP	5.7、5.6	5.7、5.6

特記事項

- 新しい ESC バージョンにアップグレードした後も、ESC サービスは、以前のリリースで展開されたすべての VNF のライフサイクルを管理します。既存の VNF に新しい機能（新しいデータモデルを使用）を適用するには、既存の VNF を展開解除して再展開する必要があります。
- アップグレードは、アクティブ/アクティブ HA のみでサポートされます。



(注) ETSI 展開では、ESC のアップグレードで VIP を変更しないでください。

ETSI の REST スキーマを変更した場合、http から https への変更中、ESC は、既存の展開に対する ESC コアからのリカバリ通知の送信を停止します。

- [スタンドアロン ESC インスタンスのアップグレード \(3 ページ\)](#)
- [ESC HA アクティブ/スタンバイインスタンスのアップグレード \(4 ページ\)](#)
- [OpenStack での ESC HA アクティブ/スタンバイノードのインサービスアップグレード \(7 ページ\)](#)
- [カーネルベースの仮想マシン \(KVM\) ESC HA アクティブ/スタンバイノードのインサービスアップグレード \(11 ページ\)](#)
- [VMware での ESC HA アクティブ/スタンバイノードのインサービスアップグレード \(14 ページ\)](#)

- [CSP での ESC HA アクティブ/スタンバイノードのインサービスアップグレード](#) (17 ページ)

スタンドアロン ESC インスタンスのアップグレード

スタンドアロン ESC インスタンスをアップグレードするには、次のタスクを実行します。

1. ESC データベースをバックアップします。詳細については、「[ESC スタンドアロンインスタンスのデータベースのバックアップ](#)」を参照してください。



(注) 展開で使用されるカスタムスクリプトをバックアップし、データベース復元の前にそれらを復元するには、バックアップにもスクリプトを再インストールする必要があります。

2. ESC インスタンスを再展開します。詳細については、以下の「[アップグレードを目的とした ESC の展開](#)」のセクションを参照してください。
3. 新しい ESC インスタンスで ESC データベースを復元します。詳細については、以下の「[ESC データベースの復元](#)」の項を参照してください。

アップグレードを目的とした ESC の展開

古い ESC VM をバックアップしてシャットダウンした後、新規またはアップグレード済み（新しい ESC パッケージに基づく）の ESC VM をインストールする必要があります。ESC インストールのすべてのパラメータは、古い ESC VM 展開と同じである必要があります。

- OpenStack の場合、新しいイメージ名で Glance コマンドを使用して新しい ESC qcow2 イメージを登録した後、新しい `bootvm.py` スクリプトと新しいイメージ名を使用して ESC VM をインストールする必要があります。



(注) OpenStack で、古い ESC VM がフローティング IP で割り当てられていた場合は、インストール後に新しい ESC VM を同じフローティング IP に関連付ける必要があります。

- VMWare の場合、新しい ESC OVA ファイルを使用して ESC VM をインストールする必要があります。他のすべての設定とプロパティ値は、古い VM と同じである必要があります。

ESC データベースの復元

次の手順を使用して、新しい ESC インスタンスで ESC データベースを復元します。

手順

ステップ 1 SSH を使用して新しい ESC インスタンスに接続します。

```
$ ssh USERNAME@NEW_ESC_IP
```

ステップ 2 ESC サービスを停止します。

```
$ sudo escadm stop
```

ステップ 3 ESC サービスのステータスをチェックして、すべてのサービスが停止していることを確認します。

```
$ sudo escadm status
```

ステップ 4 データベースファイルを復元します。

```
$ scp://<username>:<password>@<standby_ip>:<filename>
$ sudo escadm restore --file /path/where/file/scp-ed/to/db.tar.bz2
```

ステップ 5 ESC サービスを開始します。

```
$ sudo escadm start
```

ESC サービスが開始されると、スタンドアロン ESC のアップグレードが完了します。新しい ESC VM で `$ sudo escadm status` を実行することにより、新しい ESC サービスの正常性を確認できます。

ステップ 6 Openstack で、データベースを正常に復元した後、古い ESC インスタンスを削除します。

```
$ nova delete OLD_ESC_ID
```

特記事項 :

新しい ESC バージョンにアップグレードした後も、ESC サービスは、古いバージョンによって展開されたすべての VNF のライフサイクル管理を継続します。ただし、古いバージョンの ESC によって展開された VNF への新しい機能（新しいデータモデルを使用）の適用は保証されません。新しい ESC バージョンの新しい機能を既存の VNF に適用する場合は、既存の VNF を展開解除して再展開する必要があります。

ESC HA アクティブ/スタンバイインスタンスのアップグレード

ESCHA アクティブ/スタンバイノードをアップグレードするには、次のタスクを実行します。

1. 古い ESC HA アクティブ/スタンバイ アクティブ インスタンスからデータベースをバックアップします。詳細については、「[ESC HA Active/Standby Instances からのデータベースのバックアップ](#)」を参照してください。



- (注) 展開で使用されているカスタムスクリプトをバックアップし、データベースを復元する前にそれらを復元するには、バックアップにもスクリプトをコピーする必要があります。
2. 新しい ESC バージョンに基づいて、新しい ESC HA アクティブ/スタンバイノードを展開します。詳細については、次の「アップグレードを目的とした ESC の展開」のセクションを参照してください。
 3. アクティブ ESC インスタンス（スタンバイ ESC インスタンスはアクティブ ESC インスタンスと同期）のデータベースを復元します。詳細については、次の「新しいアクティブおよびスタンバイインスタンスでの ESC データベースの復元」のセクションを参照してください。

アップグレードを目的とした ESC HA アクティブ/スタンバイノードの展開

2 つの古い ESC VM をバックアップしてシャットダウンした後、新しい ESC パッケージに基づいて新しい ESC VM をインストールします。

- OpenStack の場合、新しいイメージ名と Glance コマンドを使用して新しい ESC qcow2 イメージを登録した後、新しい bootvm.py スクリプトと新しいイメージ名を使用して ESC VM をインストールする必要があります。他のすべての bootvm.py 引数は、古い VM のセットアップに使用したものと同じである必要があります。
- VMWare の場合、HA のアクティブ/スタンバイペアを VMware で起動するには 2 つの手順があります。1) 2 つのスタンドアロンインスタンスを設定 2) HA アクティブ/スタンバイ情報を使用して各インスタンスを再設定他のすべての設定とプロパティ値は、古い VM と同じにする必要があります。
- VIP がノースバウンドアクセスに使用されている場合は、古い HA アクティブ/スタンバイペアを再設定するために使用したものと同じ VIP を新しい展開で保持してください。

新しいアクティブおよびスタンバイ ESC インスタンスでの ESC データベースの復元

手順

スタンバイ ESC インスタンスをシャットダウンします。

ステップ 1 SSH を使用してスタンバイ ESC インスタンスに接続します。

```
$ ssh USERNAME@ESC_STANDBY_IP
```

ステップ 2 ESC インスタンスがスタンバイになっていることを確認し、スタンバイ ESC HA アクティブ/スタンバイインスタンス名をメモします。

```
$ sudo escadm status
```

「STANDBY」という出力値が表示される場合、そのノードはスタンバイ ESC ノードです。

(注) ダイナミック マッピング ファイル (`dynamic_mapping.xml`) が ESC サービスで使用されている場合、ダイナミック マッピング ファイルをスタンバイ ESC VM に復元する必要があります。スタンバイ ESC ノードの電源をオフにする前に、バックアップ ダイナミック マッピング ファイル (`dynamic_mapping.xml`) をパス `/opt/cisco/esc/esc-dynamic-mapping/` にコピーする必要があります。

ステップ 3 スタンバイ ESC インスタンスは、OpenStack Kg/Horizon を介して、Nova コマンドを使用してシャットダウンします。VMware vSphere を基盤とした ESC VM インスタンスの場合は、VMware クライアントダッシュボードを使用してアクティブインスタンスをシャットダウンします。OpenStack でのスタンバイ ESC インスタンスのシャットダウンの例を次に示します。

```
$ nova stop NEW_ESC_STANDBY_ID
```

新しいアクティブ ESC インスタンスを復元します。

ステップ 4 SSH を使用してアクティブな ESC インスタンスに接続します。

```
$ ssh USERNAME@ESC_ACTIVE_IP
```

ステップ 5 ESC インスタンスがアクティブであることを確認します。

```
$ sudo escadm status
```

出力値に「ACTIVE」と表示される場合、ノードはアクティブ ESC ノードです。

ステップ 6 アクティブノードの ESC サービスを停止し、ステータスを確認して、サービスが停止していることを確認します。

```
$ sudo escadm stop
$ sudo escadm status
```

ステップ 7 データベースファイルを復元します。

```
$ sudo escadm restore --file /tmp/db.tar.bz2
$ scp://<username>:<password>@<standby_ip>:<filename>
```

(注) ダイナミック マッピング ファイル (`dynamic_mapping.xml`) が ESC サービスで使用されている場合、ダイナミック マッピング ファイルを ESC VM に復元する必要があります。ESC ノードを起動する前に、バックアップ ダイナミック マッピング ファイル (`dynamic_mapping.xml`) をパス `/opt/cisco/esc-dynamic-mapping/` にコピーする必要があります。

ステップ 8 VM を再起動して、完全な ESC サービスを再起動します。

```
$ sudo escadm restart
```

ステップ 9 `$ Sudo escadm status` を使用して、ESC サービスのステータスを確認します。

ステップ 10 スタンバイ ESC ノードを起動します。

OpenStack Nova/Horizon または VMware クライアントを使用して、スタンバイ ESC ノードの電源をオンにします。スタンバイノードを起動した後、ESCHA アクティブ/スタンバイアップグレードのプロセスを完了する必要があります。

ステップ 11 古い HA アクティブ/スタンバイインスタンスを OpenStack Nova/Horizon または VMware クライアントから削除します。OpenStack で VM を削除する例を次に示します。

```
$ nova delete OLD_ESC_ACTIVE_RENAMED OLD_ESC_STANDBY_RENAMED
```

VNF モニタリングルールのアップグレード

VNF モニタリングルールをアップグレードするには、`dynamic_mappings.xml` ファイルをバックアップしてから、アップグレードされた ESC VM にファイルを復元する必要があります。詳細については、「バックアップと復元の手順」を参照してください。HA アクティブ/スタンバイインスタンスのアップグレードについては、「[ESCHA アクティブ/スタンバイインスタンスのアップグレード](#)」を参照してください。スタンドアロンインスタンスのアップグレードについては、「[スタンドアロン ESC インスタンスのアップグレード](#)」を参照してください。

OpenStack での ESC HA アクティブ/スタンバイノードのインサースビスアップグレード

手順

ステップ 1 ESC データベースとログファイルをバックアップします。

- アクティブノードから ESC データベースのバックアップを実行します。データベースのバックアップの詳細については、「[ESCHA アクティブ/スタンバイインスタンスからのデータベースのバックアップ](#)」を参照してください。
- アクティブ VM とスタンバイ VM の両方からすべてのログを収集してバックアップします。ログをバックアップするには、次のコマンドを実行します。

```
# sudo escadm log collect
```

(注) タイムスタンプファイルは、`/var/tmp/esc_log-<timestamp>.tar.bz2` に生成されます。

- ESC VM からデータベースのバックアップファイルとログファイル (`/tmp/esc_log-.tar.bz2` に生成されています) * をコピーします。

ステップ 2 新しいバージョンの ESC イメージを使用してスタンバイインスタンスを再展開し、データが同期されるまで待機します。

- Horizon Web UI または nova CLI を使用してスタンバイインスタンスを削除します。OpenStack コントローラで、nova クライアント経由で次のコマンドを実行します。

```
nova delete <secondary_vm_name>
```

- b) 新しい ESC イメージを OpenStack Glance に登録して、再展開の使用状況を確認します。

```
glance image-create --name <image_name> --disk-format qcow2 --container-format bare
--file <esc_qcow2_file>
```

- c) 新しいイメージバージョンに基づいて、スタンバイ ESC VM インスタンスを再展開します。新しい ESC パッケージ (bootvm.py および新しく登録したイメージ) を使用して、新規のスタンバイインスタンスを再インストールします。他のすべてのインストールパラメータは、以前の ESC VM 展開と同じである必要があります。たとえば、ホスト名、IP アドレス、gateway_ip、ha_node_list、kad_vip、kad_vif には同じ値を使用する必要があります。アップグレードされたバージョンの新規 ESC インスタンスが稼働すると、そのインスタンスはスタンバイ状態になります。
- d) 新規インスタンスにログインして次のコマンドを実行し、新しい ECS ノードの同期状態を確認します。

```
$ drbdadm status
```

drbdadm ステータスが出力されるまで待機し、両方のノードの出力が次のように「UpToDate」になっているかを確認します。これは、新規 ESC インスタンスで、アクティブインスタンスからのデータ同期が完了していることを示します。

アクティブ/スタンバイの例

```
esc role:Secondary
disk:UpToDate
101.1.0.119:7789 role:Primary
peer-disk:UpToDate
```

アクティブ ESC の例

```
esc role:Primary
disk:UpToDate
101.1.0.120:7789 role:Secondary
peer-disk:UpToDatee
```

ステップ 3 スタンバイインスタンスで keepalived サービスを停止し、アクティブインスタンスの電源をオフにします。次に、スタンバイ keepalived サービスを開始します。

- a) keepalived サービスを停止するには、次のコマンドを使用します。

```
escadm keepalived stop
```

- b) アクティブインスタンスにログインして、ESC アクティブノードをメンテナンスモードに設定します。

```
$ sudo escadm op_mode set --mode=maintenance
```

次のステップに進む前に、実行中のトランザクションがないことを確認してください。実行中のトランザクションがないことを確認するには、次のコマンドを実行します。

```
For ESC 2.3:
$ sudo escadm ip_trans
```


ESC 2.3 より前のバージョンの場合は、escmanager ログ (/var/log/esc/escmanager.log) に新しいトランザクションがないことを確認します。

- c) アップグレードされたスタンバイインスタンスにログインし、ESC サービスをシャットダウンします。

```
$ sudo escadm stop
```

- d) OpenStack Nova クライアント/Horizon を使用してアクティブインスタンスの電源をオフにし、オフになっていることを確認します。OpenStack コントローラで次を実行します。

```
$ nova stop <active_vm_name>
$ nova list | grep <active_vm_name>
```

- e) 以前アップグレードされたスタンバイインスタンス (停止状態) にログインし、ESC サービスを再起動します。スタンバイ ESC インスタンスはアクティブインスタンスのロールを引き継ぎ (スイッチオーバーがトリガーされます)、新しいバージョンでサービスの提供を開始します。

```
$ sudo escadm start
```

- ステップ 4** 新しいアクティブインスタンスの ESC バージョンを確認し、バージョンが正しくアップグレードされているかを確認します。

```
$ sudo escadm status (check ha status)
```

Expected output:

```
0 ESC status=0 ESC HA Active Healthy
```

```
$ esc_version (check esc version)
```

```
version : 3.x.x
```

```
release : xxx
```

- ステップ 5** 新しい ESC イメージを使用して、旧アクティブインスタンスを再展開します。

古いアクティブインスタンスを削除し、新しい ESC パッケージ (bootvm.py および新しく登録したイメージ) を使用して再展開します。

- a) 新たに展開したインスタンスにログインして、HA ステータスを確認します。新しいインスタンスはスタンバイ状態である必要があります。

```
$ sudo escadm status --v
```

- b) 次のコマンドを実行して、新しい ESC スタンバイノードの同期状態を確認します。

```
$ drbdadm status
```

drbdadm ステータスの出力に「UpToDate」と表示されるまで待機します。

- c) 新しい ESC スタンバイノードの場合、正常性チェックに合格し、ESC バージョンが正しくアップグレードされていることを確認します。

```
$ sudo escadm status (check ha status)
```

```

Expected output:
0 ESC status=0 ESC HA Standby Healthy
$ esc_version (check esc version)version : 2.x.x
release : xxx
$ health.sh
Expected output:
ESC HEALTH PASSED

```

ステップ 6 最初にアップグレードしたアクティブインスタンスに戻り、正常性と `keepalived` の状態をチェックします。

```
$ drbdadm status
```

```

Expected output:
1:esc/0 Connected Primary/Secondary UpToDate/UpToDate /opt/cisco/esc/esc_database ext4
2.9G 52M 2.7G 2%

```

```
$ sudo escadm status (check ha status)
```

```

Expected output:
0 ESC status=0 ESC HA Active Healthy

```

```
$ esc_version (check esc version) Expected output:
```

```

version : 2.x.x
release : xxx

```

```
$ health.sh (check esc health)
```

```

Expected output:
ESC HEALTH PASSED

```

(注) クイックロールバック：アップグレードに失敗した場合は、アップグレードされたインスタンスをシャットダウンし、古いアクティブインスタンスを起動して、クイックロールバックを実行します。

インサービスアップグレードのロールバック手順

1. ESC VM からデータベースとログのバックアップファイルを任意の場所にコピーします。
2. 残りの ESC インスタンスを削除したら、古いバージョンの `qcow2` イメージを使用して ESC HA アクティブ/スタンバイ VM を再展開します。
3. データベースを復元します。「HA アクティブ/スタンバイデータベース復元のための Backup and Restore を使用した ESC HA アクティブ/スタンバイインスタンスのアップグレード」の手順に従います。
4. データベースの復元後、ESC サービスを古いバージョンに戻す必要があります。

カーネルベースの仮想マシン (KVM) ESC HA アクティブ/スタンバイノードのインサービスアップグレード

手順

ステップ 1 ESC データベースとログファイルをバックアップします。

- アクティブノードから ESC データベースのバックアップを実行します。データベースのバックアップの詳細については、「[ESC HA アクティブ/スタンバイインスタンスからのデータベースのバックアップ](#)」を参照してください。
- アクティブ VM とスタンバイ VM の両方からすべてのログを収集してバックアップします。ログをバックアップするには、次のコマンドを実行します。

```
$ sudo escadm log collect
```

(注) タイムスタンプログファイルは、`/var/tmp/esc_log-<timestamp>.tar.bz2` に生成されます。

- ESC VM からデータベースのバックアップファイルとログファイル (`/tmp/esc_log-.tar.bz2` に生成) * をコピーします。

ステップ 2 スタンバイ ESC インスタンスを再展開します。スタンバイインスタンスに新しい ESC イメージを登録します。

- `lib vert Virsh` コマンドを使用して、スタンバイインスタンスを削除します。KVM ホストで、次のコマンドを実行します。

```
$ virsh destroy the <secondary_vm_name>
$ virsh undefine --remove-all-storage <secondary_vm_name>
```

- 再展開の使用のために、新しい ESC イメージを Kvm ホストにコピーします。

```
sshpass -p "host Password" scp /scratch/BUILD-2_x_x_x/BUILD-2_x_x_x/ESC-2_x_x_x.qcow2
root@HOSTIP:
```

- 新しいイメージバージョンに基づいて、スタンバイ ESC VM インスタンスを再展開します。新しい ESC パッケージ (`bootvm.py` および新しく登録したイメージ) を使用して、新規のスタンバイインスタンスを再インストールします。他のすべてのインストールパラメータは、以前の ESC VM 展開と同じである必要があります。たとえば、ホスト名、IP アドレス、`gateway_ip`、`ha_node_list`、`kad_vip`、`kad_vif` には同じ値を使用する必要があります。アップグレードされたバージョンの新規 ESC インスタンスが稼働すると、そのインスタンスはスタンバイ状態になります。
- 新規インスタンスにログインして次のコマンドを実行し、新しい ECS ノードの同期状態を確認します。

```
$ drbdadm status
```

drbdadm ステータスが出力されるまで待機し、両方のノードが次の出力のように「UpToDate」になっていることを確認します。これは、新規 ESC インスタンスで、アクティブインスタンスからのデータ同期が完了していることを示します。

```
esc/0 Connected Secondary/Primary UpToDate/UpToDate
```

ステップ 3 スタンバイインスタンスで `keepalived` サービスを停止し、アクティブインスタンスの電源をオフにします。次に、スタンバイ `keepalived` サービスを開始します。

- a) アクティブインスタンスにログインして、ESC アクティブノードをメンテナンスモードに設定します。

```
$ sudo escadm op_mode set --mode=maintenance
```

次のステップに進む前に、実行中のトランザクションがないことを確認してください。実行中のトランザクションがないことを確認するには、次のコマンドを実行します。

```
$ sudo escadm ip_trans
```

escmanager ログ (`/var/log/esc/escmanager.log`) に新しいトランザクションがないことを確認します。

- b) アップグレードされたスタンバイインスタンスにログインして、`keepalived` サービスをシャットダウンします。

```
$ sudo escadm stop
```

- c) アクティブインスタンスの電源をオフにして、完全にオフになっていることを確認します。KVM ESC コントローラで、次を実行します。

```
$ virsh destroy <active_vm_name>
$ virsh list --all
```

- d) 以前にアップグレードされたスタンバイインスタンス (停止状態) にログインし、ESC サービスを起動します。スタンバイ ESC インスタンスはアクティブインスタンスのロールを引き継ぎ (スイッチオーバーがトリガーされます)、新しいバージョンでサービスの提供を開始します。

```
$ sudo escadm restart
```

ステップ 4 新しいアクティブインスタンスの ESC バージョンを確認し、バージョンが正しくアップグレードされているかを確認します。

```
$ sudo escadm status (check ha status)
```

```
Expected output:
0 ESC status=0 ESC HA Active Healthy
```

```
$ esc_version (check esc version)
version : 4.1.x
release : xxx
```

```
$ health.sh (check esc health)
```

```
Expected output:
ESC HEALTH PASSED
```

ステップ 5 新しい ESC イメージを使用して、旧アクティブインスタンスを再展開します。

古いアクティブリインスタンスを削除し、新しい ESC パッケージ (bootvm.py および新しく登録したイメージ) を使用して再展開します。他のすべてのインストールパラメータは、古い ESC VM の展開と同じにする必要があります。たとえば、ホスト名、IP アドレス、gateway_ip、ha_node_list、kad_vip、kad_vif は同じ値にする必要があります。

- a) 新たに展開したインスタンスにログインして、HA ステータスを確認します。新しいインスタンスはスタンバイ状態である必要があります。

```
$ sudo escadm status
```

- b) 次のコマンドを実行して、新しい ESC スタンバイノードの同期状態を確認します。

```
$ drbdadm status
```

drbdadm ステータスの出力に「UpToDate」と表示されるまで待機します。

- c) 新しい ESC スタンバイノードの場合、正常性チェックに合格し、ESC バージョンが正しくアップグレードされていることを確認します。

```
$ sudo escadm status (check ha status)
Expected output:
0 ESC status=0 ESC HA Active Healthy
$ esc_version (check esc version)version : 4.1.x
release : xxx
$ health.sh
Expected output:
ESC HEALTH PASSED
```

ステップ 6 最初にアップグレードしたアクティブインスタンスに戻り、正常性と keepalived の状態をチェックします。

```
$ drbdadm status
Expected output:
1:esc/0 Connected Primary/Secondary UpToDate/UpToDate /opt/cisco/esc/esc_database ext4
2.9G 52M 2.7G 2%

$ sudo escadm status (check ha status)
Expected output:
0 ESC status=0 ESC HA Active Healthy

$ esc_version (check esc version) Expected output:
version : 2.x.x
release : xxx

$ health.sh (check esc health)
Expected output:
ESC HEALTH PASSED
```

- (注) クイックロールバック：アップグレードに失敗した場合は、アップグレードされたインスタンスをシャットダウンし、古いアクティブインスタンスを起動して、クイックロールバックを実行します。

インサービスアップグレードのロールバック手順

1. 残りの ESC インスタンスを削除したら、古いバージョンの qcow2 イメージを使用して ESC HA アクティブ/スタンバイ VM を再展開します。
2. データベースを復元します。「HA アクティブ/スタンバイデータベース復元のための Backup and Restore を使用した ESC HA アクティブ/スタンバイインスタンスのアップグレード」の手順に従います。
3. データベースの復元後、ESC サービスを古いバージョンに戻す必要があります。

VMware での ESC HA アクティブ/スタンバイノードのインサービスアップグレード

手順

ステップ 1 ESC データベースとログファイルをバックアップします。

- a) アクティブノードから ESC データベースのバックアップを実行します。データベースのバックアップの詳細については、「[ESC HA アクティブ/スタンバイインスタンスからのデータベースのバックアップ](#)」を参照してください。
- b) アクティブ VM とスタンバイ VM の両方からすべてのログを収集してバックアップします。ログをバックアップするには、次のコマンドを実行します。

```
# sudo escadm log collect
```

- (注) タイムスタンプログファイルは、`/var/tmp/esc_log-<timestamp>.tar.bz2` に生成されます。

- c) ESC VM からデータベースのバックアップファイルとログファイル (`/tmp/esc_log-.tar.bz2` に生成) * をコピーします。

ステップ 2 スタンバイ ESC インスタンスを再展開します。スタンバイインスタンスに新しい ESC イメージを登録し、データが同期されるまで待機します。

- a) スタンバイインスタンスを削除します。スタンバイ ESC インスタンスを削除するには、まず vSphere クライアントからインスタンスの「電源をオフ」にし、次に **[ディスクから削除 (Delete from Disk)]** オプションを使用する必要があります。VMware vSphere クライアントで、**[ホーム (Home)] > [インベントリ (Inventory)] > [VM とテンプレート (VMs and Templates)]** を選択します。左側のパネルからプライマリインスタンス名を右クリックして、**[電源 (Power)] > [電源オフ (Power Off)]** を選択します。次に、スタンバイインスタ

ンスを削除するには、[ホーム (Home)]>[インベントリ (Inventory)]>[VM とテンプレート (VMs and Templates)]を選択します。左側のパネルからインスタンス名を右クリックして、[ディスクから削除 (Delete from Disk)]を選択します。

- b) 新しいイメージバージョンに基づいて、スタンバイ ESC VM インスタンスを再展開します。新しい ESC パッケージ (bootvm.py および新しく登録したイメージ) を使用して、新規のスタンバイインスタンスを再インストールします。アップグレードされたバージョンの新規 ESC インスタンスが稼働すると、そのインスタンスはスタンバイ状態になります。
- c) 新規インスタンスにログインして次のコマンドを実行し、新しい ECS ノードの同期状態を確認します。

```
$ drbdadm status
```

drbdadm ステータスが出力されるまで待機し、両方のノードの出力が次のように

「UpToDate」になっているかを確認します。これは、新規 ESC インスタンスで、アクティブインスタンスからのデータ同期が完了していることを示します。

```
esc/0 Connected Secondary/Primary UpToDate/UpToDate
```

ステップ 3 スタンバイインスタンスで keepalived サービスを停止し、アクティブインスタンスの電源をオフにします。次に、スタンバイ keepalived サービスを開始します。

- a) アクティブインスタンスにログインして、ESC アクティブノードをメンテナンスモードに設定します。

```
$ sudo escadm op_mode set --mode=maintenance
```

次のステップに進む前に、実行中のトランザクションがないことを確認してください。実行中のトランザクションがないことを確認するには、次のコマンドを実行します。

```
For ESC 2.3:  
$ sudo escadm ip_trans
```

ESC 2.3 より前のバージョンの場合は、escmanager ログ (/var/log/esc/escmanager.log) に新しいトランザクションがないことを確認します。

- b) アップグレードされたスタンバイインスタンスにログインして、keepalived サービスをシャットダウンします。

```
$ sudo escadm stop
```

- c) アクティブインスタンスの電源をオフにして、アクティブインスタンスの電源がオフになっていることを確認します。VMware vSphere クライアントで、[ホーム (Home)]>[インベントリ (Inventory)]>[VM とテンプレート (VMs and Templates)]を選択します。左側のパネルからプライマリインスタンス名を右クリックして、[電源 (Power)]>[電源オフ (Power Off)]を選択します。
- d) 以前アップグレードされたスタンバイインスタンス (停止状態) にログインし、keepalived サービスを起動します。スタンバイ ESC インスタンスはアクティブインスタンスのロールを引き継ぎ (スイッチオーバーがトリガーされます) 、新しいバージョンでサービスの提供を開始します。

```
$ sudo escadm start
```

ステップ 4 新しいアクティブインスタンスの ESC バージョンを確認し、バージョンが正しくアップグレードされているかを確認します。

```
$ sudo escadm status --v(check ha status)

Expected output:
0 ESC status=0 ESC HA Active Healthy
vimmanager (pgid 41908) is running
monitor (pgid 42067) is running
mona (pgid 42130) is running
drbd (pgid 38488) is active
snmp (pgid 2121) is running
etsi (pgid 43247) is running
pgsql (pgid 42566) is running
keepalived (pgid 40281) is running
portal (pgid 43307) is running
confd (pgid 25644) is running
filesystem (pgid 0) is running
escmanager (pgid 42775) is running

$ esc_version (check esc version)
version : 3.x.x
release : xxx

$ health.sh (check esc health)

Expected output:
ESC HEALTH PASSED
```

ステップ 5 新しい ESC イメージを使用して、旧アクティブインスタンスを再展開します。

古いアクティブインスタンスを削除し、新しい ESC パッケージ (bootvm.py および新しく登録したイメージ) を使用して再展開します。他のすべてのインストールパラメータは、古い ESC VM の展開と同じにする必要があります。たとえば、ホスト名、IP アドレス、gateway_ip、ha_node_list、kad_vip、kad_vif は同じ値にする必要があります。削除する場合は、[ホーム (Home)] > [インベントリ (Inventory)] > [VM とテンプレート (VMs and Templates)] を選択します。左側のパネルからインスタンス名を右クリックして、[ディスクから削除 (Delete from Disk)] を選択します。

- 新たに展開したインスタンスにログインして、HA ステータスを確認します。新しいインスタンスはスタンバイ状態である必要があります。

```
$ sudo escadm status
```

- 次のコマンドを実行して、新しい ESC スタンバイノードの同期状態を確認します。

```
$ drbdadm status
```

drbdadm ステータスの出力に「UpToDate」と表示されるまで待機します。

- 新しい ESC スタンバイノードの場合、正常性チェックに合格し、ESC バージョンが正しくアップグレードされていることを確認します。

```
$ sudo escadm status (check ha status)
Expected output:
0 ESC status=0 ESC HA Active Healthy
$ esc_version (check esc version)version : 3.x.x
```



```
release : xxx
$ health.sh
Expected output:
ESC HEALTH PASSED
```

ステップ 6 最初にアップグレードしたアクティブインスタンスに戻り、正常性と `keepalived` の状態をチェックします。

```
$ drbdadm status
Expected output:
1:esc/0 Connected Primary/Secondary UpToDate/UpToDate /opt/cisco/esc/esc_database ext4
2.9G 52M 2.7G 2%

$ sudo escadm status (check ha status)
Expected output:
0 ESC status=0 ESC HA Active Healthy

$ esc_version (check esc version) Expected output:
version : 3.x.x
release : xxx

$ health.sh (check esc health)
Expected output:
ESC HEALTH PASSED
```

(注) クイックロールバック : アップグレードに失敗した場合は、アップグレードされたインスタンスをシャットダウンし、古いアクティブインスタンスを起動して、クイックロールバックを実行します。

インサービスアップグレードのロールバック手順

1. ESC VM からデータベースとログのバックアップファイルを任意の場所にコピーします。
2. 残りの ESC インスタンスを削除したら、古いバージョンの `qcow2` イメージを使用して ESC HA アクティブ/スタンバイ VM を再展開します。
3. データベースを復元します。「HA アクティブ/スタンバイデータベース復元のための Backup and Restore を使用した ESC HA アクティブ/スタンバイインスタンスのアップグレード」の手順に従います。
4. データベースの復元後、ESC サービスを古いバージョンに戻す必要があります。

CSP での ESC HA アクティブ/スタンバイノードのインサービスアップグレード

CSP で ESC HA アクティブ/スタンバイノードのインサービスアップグレードを実行するには、次の手順を実行します。

始める前に

次のコマンドを使用して、アップグレードの前に ESC HA ノードが適切に稼働していることを確認します。

```
# escadm status
```

1つのノードがアクティブ状態で、もう1つのノードがスタンバイ状態である必要があります。次のコマンドを使用して、アクティブノードを確認します。

```
# health.sh
```

正常性チェックに合格すると、次が表示されます。

```
ESC HEALTH PASSED
```

手順

ステップ1 スタンバイインスタンスのシャットダウン

VMの電源をオフにする前に、スタンバイ ESC VM をアップグレードします。アップグレードするには、次の手順に従います。

1. 次のコマンドを使用して、スタンバイ ESC VM からログを収集し、別のマシンにコピーします。

```
# collect_esc_log.sh
# scp /tmp/LOG_PACKAGE_NAME <username>@<backup_vm_ip>:<filepath>
```

2. CSP を使用して、スタンバイ ESC の電源をオフにします。

ステップ2 新しい ESC パッケージを使用して、スタンバイノードを再展開します。

スタンバイ ESC VM の電源をオフにした後、アップグレードするために、新しい ESC パッケージを使用して新しい ESC VM をインストールします。以前の ESC VM とは異なる ESC パッケージを使用する場合を除き、ESC インストールで使用するその他すべてのパラメータは、以前の ESC VM 展開と同じにする必要があります。

ESC ノードがスタンバイ状態になっていることを確認します。次のコマンドを使用して、新しい ESC スタンバイノードの同期状態を確認します。

```
# drbdadm status
```

次の出力が表示されるまで待機します。これは、新しい ESC VM がアクティブノードからのデータ同期を完了しており、最新の状態であることを示しています。

```
[admin@esc-xyx-upgradetest1-4-5-0-105 ~]$ drbdadm status
esc role:Secondary
disk:UpToDate
172.20.117.55:7789 role:Primary
peer-disk:UpToDate
```

ダイナミック マッピング ファイル (`dynamic_mapping xm`) が ESC サービスによって使用されている場合は、ESC VM に復元する必要があります。バックアップファイルを `/opt/cisco/esc-dynamic-mapping/` パスにコピーします。

ステップ3 アクティブノードを停止し、スイッチオーバーをトリガーします。

CSPを使用してアクティブインスタンスの電源をオフにします。その後、HAスイッチオーバーが自動的にトリガーされ、スタンバイインスタンスが新しいバージョンのESCサービスを引き継ぎます。新しいESCがアクティブになった後、その新しいESCアクティブノードが正常性チェックに合格したかを確認します。

ステップ4 古いアクティブを置き換えるための新しいESCノードの展開

新しいESCパッケージを使用して以前のアクティブインスタンスをアップグレードするため、新しいESC VMをインストールします。古いバージョンのESC VMの電源をオフにする前に、次のコマンドを使用して、ESC VMからログを収集して別のマシンにコピーします。

```
# collect_esc_log.sh
# scp /tmp/LOG_PACKAGE_NAME <username>@<backup_vm_ip>:<filepath>
```

(注) ダイナミック マッピング ファイルがESCサービスによって使用されている場合は、ダイナミック マッピング ファイルをESCログと同じタイミングでバックアップする必要があります。ダイナミック マッピング ファイルのデフォルトパスは `/opt/cisco/esc/esc-dynamic-mapping/dynamic_mappings.xml` です。

CSPを介して古いバージョンのESC VMの電源をオフにします。

ESCノードがスタンバイ状態になっていることを確認します。次のコマンドを使用して、新しいESCスタンバイノードの同期状態を確認します。

```
# drbdadm status
```

次の出力が表示されるまで待機します。これは、新しいESC VMがアクティブノードからのデータ同期を完了しており、最新の状態であることを示しています。

```
[admin@esc-xyz-upgradetesthal-4-5-0-105 ~]$ drbdadm status
esc role:Secondary
disk:UpToDate
172.20.117.55:7789 role:Primary
peer-disk:UpToDate
```

ダイナミック マッピング ファイル (`dynamic_mapping xm`) がESCサービスによって使用されている場合は、ESC VMに復元する必要があります。バックアップファイルを `/opt/cisco/esc-dynamic-mapping/` パスにコピーします。

(注) ダイナミック マッピング ファイルがESCサービスによって使用されている場合は、ダイナミック マッピング ファイルをESC VMに復元する必要があります。バックアップダイナミックマッピングファイルを、ダイナミックマッピングファイルのデフォルトパス `/opt/cisco/esc/esc-dynamic-mapping/dynamic_mappings.xml` にコピーします。

インサービスアップグレードが正常に完了すると、古いESCインスタンスを削除できます。

- (注) 新しい ESC バージョンにアップグレードした後も、ESC サービスは、古いバージョンによって展開されたすべての VNF のライフサイクル管理を継続します。新しい ESC バージョンの新機能を既存の VNF に適用する場合は、それらの VNF の展開を解除し、新たに展開を実行します。
-

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。