

Cisco Open NX-OSでのサードパーティリポジトリの設定

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[設定](#)

[方式](#)

[コンフィギュレーション](#)

[方法1: 直接接続](#)

[方法2: プロキシ経由で接続する](#)

[確認](#)

[関連リンク](#)

はじめに

このドキュメントでは、Cisco Open NX-OSのサードパーティリポジトリとしてEnterprise Linux(EPEL)用の追加パッケージをセットアップする方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Cisco Open NX-OS
- ドメイン ネーム システム (DNS)

使用するコンポーネント

このドキュメントの情報は、NXOSバージョン10.3(4a)を搭載したCisco Nexus N9K-C9364Cに基づいています。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな (デフォルト) 設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

設定

Cisco NX-OSは、数千もの実稼働環境でCisco Nexusスイッチを稼働させるネットワークオペレーティングシステム(OS)です。これは、Linuxで構築された最初のデータセンターネットワークオペレーティングシステムでした。Cisco NX-OSには常にLinuxが搭載されており、最近ではLinuxの機能の多くがエンドユーザに公開されています。

ユーザは、標準のLinuxサーバ管理ツールとワークフローを使用して、カスタム開発されたLinuxベースのアプリケーションやその他の標準オープンソースプログラムをインストールし、Nexusスイッチ上で「購入後すぐに」機能させることができます。Puppet、Chefなどの一般的なサードパーティ製構成管理エージェントと、ganglia、splunk、collector、nagiosなどのテレメトリアプリケーションをスイッチに統合するのは簡単です。

方式

Nexusスイッチをサードパーティのリポジトリに接続するには、2つの方法があります。

- 直接：任意のレイヤ3インターフェイスを使用してリポジトリに到達できます。
- プロキシ経由：任意のレイヤ3インターフェイスを使用して、プロキシ経由でリポジトリに到達できます。

コンフィギュレーション

これらの設定は、すべての通信に管理VRFを使用してNexus 9000スイッチに実装されます。

方法1：直接接続

ステップ 1：必要な機能を有効にします。

```
Nexus(config)# feature bash
```

ステップ 2：DNSクライアントを設定します。

```
Nexus(config)# ip domain-lookup  
Nexus(config)# vrf context management  
Nexus(config-vrf)# ip name-server <dns server ip>
```

ステップ 3：サードパーティのリポジトリファイルを設定します。ここでは、目的のリポジトリを指定します。

注：

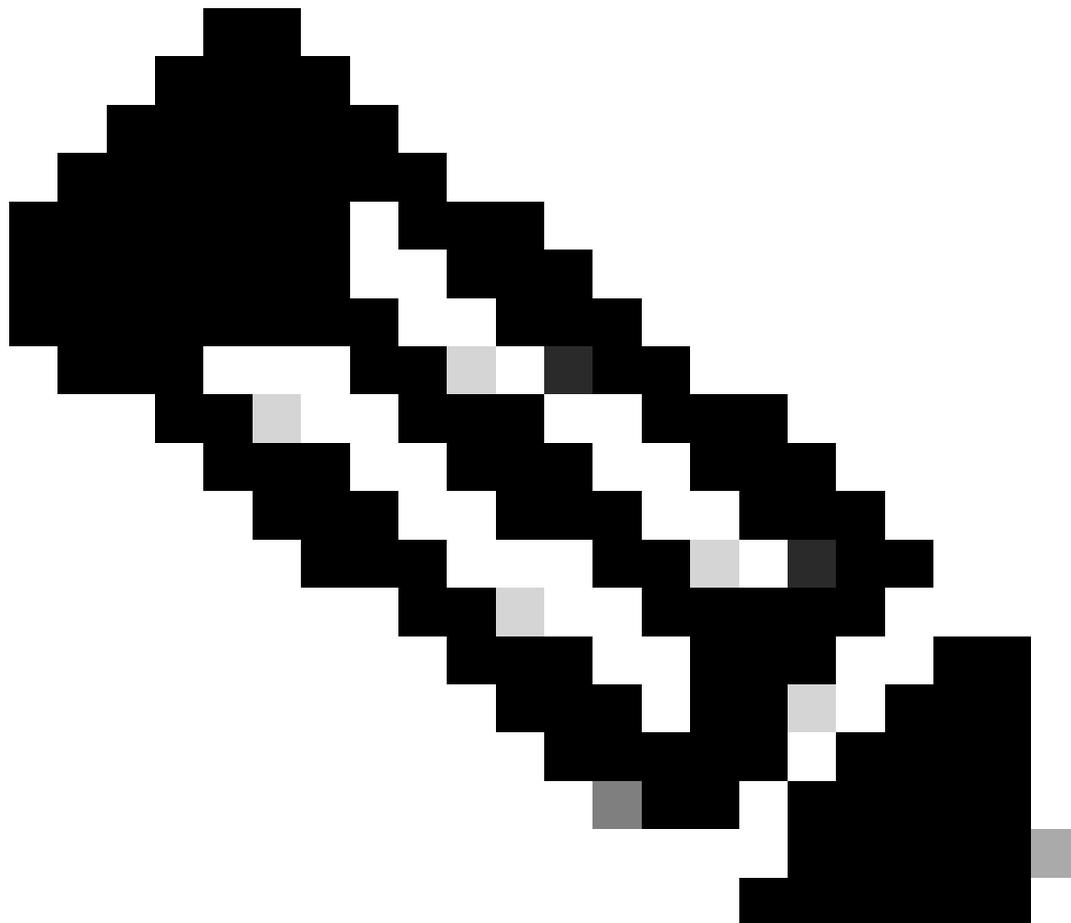
このガイドでは、例としてExtra Packages for Enterprise Linux(EPEL)を使用します

Vimは、ファイルを編集するためにプリインストールされています。

Catコマンドを使用して、コンテンツファイルを表示します。

```
Nexus# run bash
bash-4.3$ sudo su -l
root@cisco#cat /etc/yum/repos.d/thirdparty.repo
[thirdparty]
name=Thirdparty RPM Database
baseurl=https://dl.fedoraproject.org/pub/epel/7/x86_64/
enabled=1
gpgcheck=1
metadata_expire=0
cost=500
sslverify=0
```

ステップ 4 : 適切なGNU Privacy Guard(GPG)キーを対象ディレクトリにダウンロードします。



注 : 管理VRFが指定されています

```
root@cisco#cd /etc/pki/rpm-gpg
root@cisco#ip netns exec management wget https://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-7
```

ステップ 5 : GPGキーをインポートします。

```
root@cisco#rpm --import RPM-GPG-KEY-EPEL-7
```

方法2 : プロキシ経由で接続する

ステップ 1 : 必要な機能を有効にします。

```
Nexus(config)# feature bash
```

ステップ 2 : DNSクライアントを設定します。

```
Nexus(config)# ip domain-lookup  
Nexus(config)# vrf context management  
Nexus(config-vrf)# ip name-server <dns server ip>
```

ステップ 3 : yumのオプションファイルを設定します。ここでは、プロキシをポイントします。

注 :

proxy_usernameオプションとproxy_passwordオプションは、匿名プロキシには必要ありません。

タイムアウト値はデフォルトで60です。別の値が必要な場合はオプションを設定します。

Vimは、ファイルを編集するためにプリインストールされています。

Catコマンドを使用して、コンテンツファイルを表示します。

```
Nexus# run bash
bash-4.3$ sudo su -l
root@cisco#cat /etc/yum/yum.conf
[main]
cachedir=/var/cache/yum
keepcache=1
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
obsoletes=1
pluginpath=/lib/yum-plugins
plugins=1
proxy=http://<proxy ip>:<proxy port>/
proxy_username=cisco
proxy_password=cisco123
timeout=300

# PUT YOUR REPOS HERE OR IN separate files named file.repo
# in /etc/yum/repos.d
```

ステップ4：サードパーティのリポジトリファイルを設定します。ここでは、目的のリポジトリを指定します。



注：

このガイドでは、例としてExtra Packages for Enterprise Linux(EPEL)を使用します。

Vimは、ファイルを編集するためにプリインストールされています。

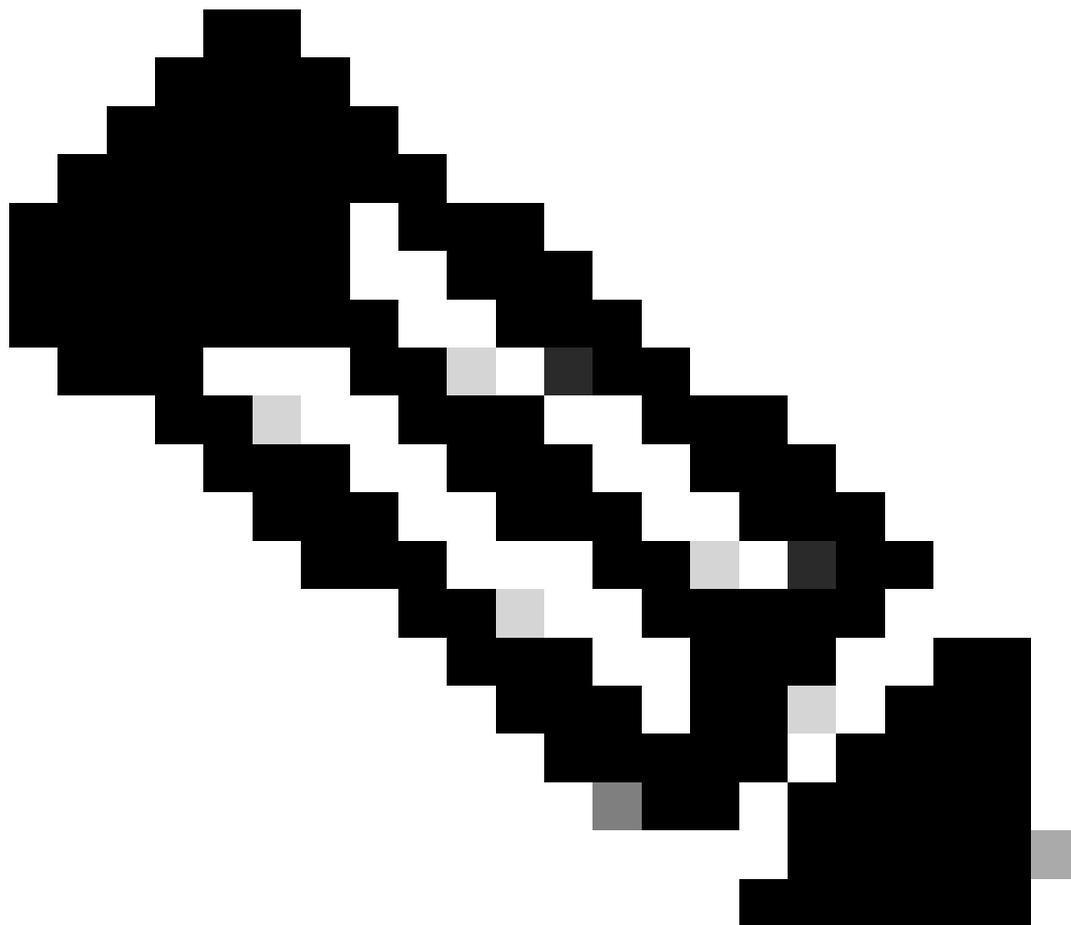
Catコマンドを使用して、コンテンツファイルを表示します。

```
root@cisco#cat /etc/yum/repos.d/thirdparty.repo
[thirdparty]
name=Thirdparty RPM Database
baseurl=https://dl.fedoraproject.org/pub/epel/7/x86_64/
enabled=1
gpgcheck=1
metadata_expire=0
cost=500
sslverify=0
```

ステップ 5 : HTTPS接続のプロキシをセットアップします。

```
root@cisco#export https_proxy=http://<proxy ip>:<proxy port>
```

手順 6 : 適切なGNU Privacy Guard(GPG)キーを対象ディレクトリにダウンロードします。



注 : 管理VRFが指定されています

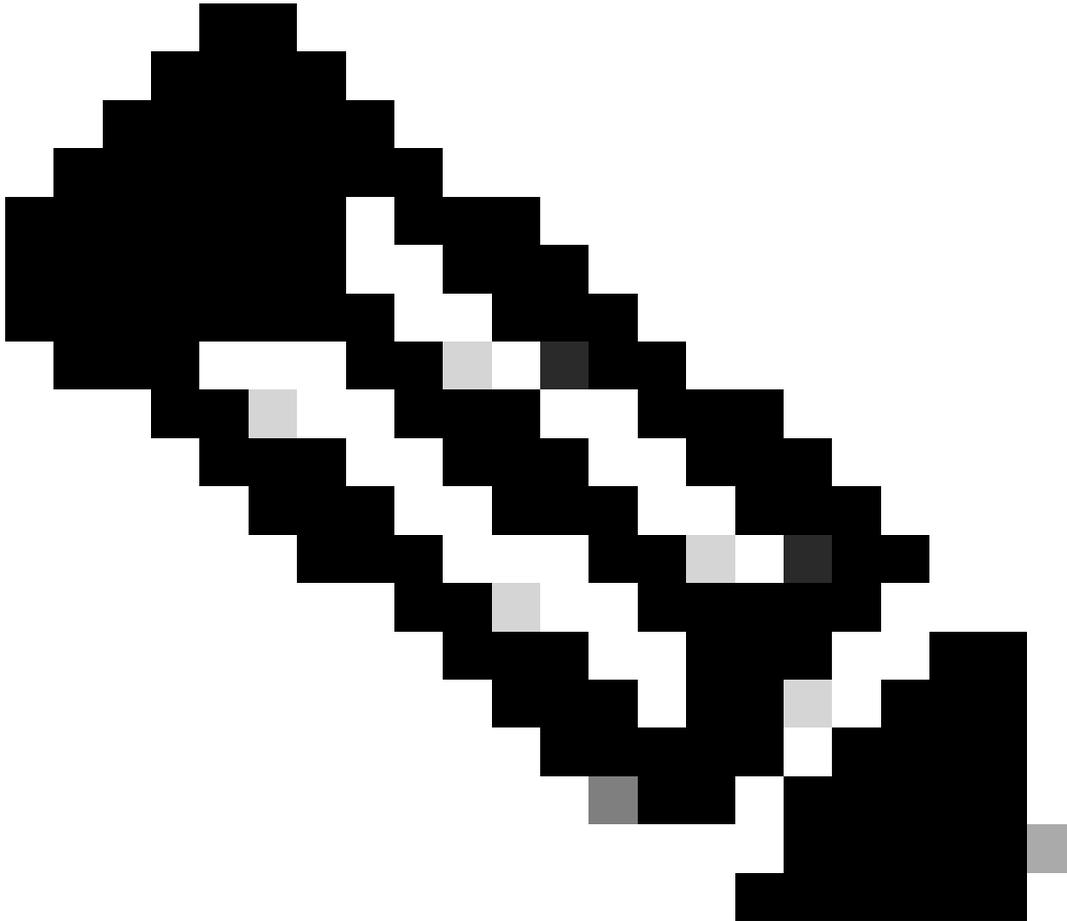
```
root@cisco#cd /etc/pki/rpm-gpg  
root@cisco#ip netns exec management wget https://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-7
```

手順 7 : GPGキーをインポートします。

```
root@cisco#rpm --import RPM-GPG-KEY-EPEL-7
```

確認

このガイドでは、サードパーティのリポジトリが正しく設定されていることを確認するために、SHCパッケージをインストールします。



注：管理VRFは各コマンドで指定されます

手順1:Nexusがパッケージをフェッチできることを確認します。

```
root@cisco#ip netns exec management yum repolist
Loaded plugins: downloadonly, importpubkey, localrpmDB, patchaction, patching,
                : protect-packages
groups-repo      | 1.1 kB    00:00 ...
localdb          | 951 B    00:00 ...
```

```

patching | 951 B 00:00 ...
thirdparty | 4.7 kB 00:00
wrl-repo | 951 B 00:00 ...
repo id      repo name      status
groups-repo Groups-RPM Database 46
localdb     Local RPM Database 0
patching    Patch-RPM Database 0
thirdparty  Thirdparty RPM Database 13,798
wrl-repo    Groups-RPM Database 12
repolist: 13,856

```

ステップ 2 : SHCパッケージのインストール

```

root@cisco#ip netns exec management yum install shc
Loaded plugins: downloadonly, importpubkey, localrpmDB, patchaction, patching,
               : protect-packages
groups-repo | 1.1 kB 00:00 ...
groups-repo/primary | 35 kB 00:00 ...
groups-repo 46/46
localdb | 951 B 00:00 ...
localdb/primary | 197 B 00:00 ...
patching | 951 B 00:00 ...
patching/primary | 197 B 00:00 ...
thirdparty | 4.7 kB 00:00
thirdparty/primary_db | 7.0 MB 00:25
wrl-repo | 951 B 00:00 ...
wrl-repo/primary | 4.2 kB 00:00 ...
wrl-repo 12/12
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package shc.x86_64 0:4.0.3-1.e17 will be installed
--> Finished Dependency Resolution

```

Dependencies Resolved

```

=====
Package      Arch      Version      Repository      Size
=====
Installing:
shc          x86_64    4.0.3-1.e17  thirdparty      38 k

```

Transaction Summary

```

=====
Install      1 Package

```

Total download size: 38 k

Installed size: 90 k

Is this ok [y/N]: y

Downloading Packages:

```

shc-4.0.3-1.e17.x86_64.rpm | 38 kB 00:00

```

Running Transaction Check

Running Transaction Test

Transaction Test Succeeded

Running Transaction

** Found 1 pre-existing rpmdb problem(s), 'yum check' output follows:

busybox-1.23.2-r0.0.x86_64 has missing requires of busybox-syslog

Installing : shc-4.0.3-1.e17.x86_64

1/1

```
Installed:  
shc.x86_64 0:4.0.3-1.e17
```

```
Complete!  
Install operation 21 completed successfully at Tue Apr 9 23:36:10 2024.
```

```
[#####] 100%
```

ステップ 3 : SHCがインストールされていることを確認します。

```
root@cisco#shc  
shc parse(-f): No source file specified
```

```
shc Usage: shc [-e date] [-m addr] [-i iopt] [-x cmdnd] [-l lopt] [-o outfile] [-rvDSUHCABh] -f script
```

関連リンク

[オープンNX-OS](#)

[Cisco Open NX-OSによるプログラマビリティと自動化](#)

[Cisco Nexusスイッチプログラマビリティスクリプトのリポジトリ](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。