Cisco Extensible Network Controller 導入ガイド リリース 1.0

初版：2013 年 10 月 07 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

Text Part Number: OL-29728-02-J

【注意】シスコ製品をご使用になる前に、安全上の注意（**www.cisco.com/jp/go/safety_warning/**）をご確認ください。本書は、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。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

目 次

はじめに

ここでは、次の項について説明します。

対象読者

このマニュアルは、Cisco Extensible Network Controller の設定と保守を行う経験豊富なネットワーク管理者を対象としています。

表記法

コマンドの説明には、次のような表記法が使用されます。

表記法	説明
bold	太字の文字は、表示どおりにユーザが入力するコマンドおよびキーワードです。
italic	イタリック体の文字は、ユーザが値を入力する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角カッコで囲んで示しています。
[x | y]	いずれか 1 つを選択できる省略可能なキーワードや引数は、角カッコで囲み、縦棒で区切って示しています。
{x | y}	必ずいずれか1つを選択しなければならない必須キーワードや引数は、波カッコで囲み、縦棒で区切って示しています。

表記法	説明
[x {y | z}]	角カッコまたは波カッコが入れ子になっている箇所は、任意または必須の要素内の任意または必須の選択肢であることを表します。 角カッコ内の波カッコと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。
`variable`	ユーザが値を入力する変数であることを表します。イタリック体が使用できない場合に使用されます。
string	引用符を付けない一組の文字。 string の前後には引用符を使用しません。引用符を使用すると、その引用符も含めて string とみなされます。

例では、次の表記法を使用しています。

表記法	説明
`screen フォント`	スイッチが表示する端末セッションおよび情報は、screen フォントで示しています。
太字の screen フォント	ユーザが入力しなければならない情報は、太字の screen フォントで示しています。
イタリック体の *screen* フォント	ユーザが値を指定する引数は、イタリック体の screen フォントで示しています。
< >	パスワードのように出力されない文字は、山カッコ（<>）で囲んで示しています。
[ ]	システム プロンプトに対するデフォルトの応答は、角カッコで囲んで示しています。
!、#	コードの先頭に感嘆符（!）またはポンド記号（#）がある場合には、コメント行であることを示します。

このマニュアルでは、次の表記法を使用しています。

（注） 「注釈」です。 役立つ情報やこのマニュアルに記載されていない参照資料を紹介しています。

注意 「要注意」の意味です。 機器の損傷またはデータ損失を予防するための注意事項が記述されています。

マニュアルの入手方法およびテクニカル サポート

マニュアルの入手方法、テクニカル サポート、その他の有用な情報について、次の URL で、毎月更新される『*What's New in Cisco Product Documentation*』を参照してください。シスコの新規および改訂版の技術マニュアルの一覧も示されています。

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

『*What's New in Cisco Product Documentation*』は RSS フィードとして購読できます。また、リーダー アプリケーションを使用してコンテンツがデスクトップに直接配信されるように設定することもできます。 RSS フィードは無料のサービスです。シスコは現在、RSS バージョン 2.0 をサポートしています。

第 1 章

Cisco XNC の概要

この章の内容は、次のとおりです。

Cisco Extensible Network Controller について

Cisco Extensible Network Controller（Cisco XNC）は、1 方向（サウスバウンド）のネットワーク要素とサードパーティ アプリケーション（ノースバウンド）の間のインターフェイスとして機能するソフトウェア プラットフォームです。 Cisco XNC は、Java Virtual Machine（JVM）上で実行される JVM ベースのアプリケーションです。 Cisco XNC は、ネットワークをサポートする、可用性、スケーラビリティ、および拡張性が高いアーキテクチャに基づいています。 Cisco XNC は、新しい機能を追加できる Open Services Gateway initiative（OSGi）フレームワークを使用して拡張できるように構築されています。

Cisco XNC は、サウスバンド方向の複数のプロトコル プラグインをサポートできます。 現在のリリースでは、OpenFlow バージョン 1.0 を使用できます。

Cisco XNC は、次の機能を提供します。

- このリリースで使用できる OpenFlow バージョン 1.0 を含むマルチ プロトコル機能。
- ネットワーク トポロジ ディスカバリ、ネットワーク デバイス管理、ルール プログラミングの転送、および詳細なネットワーク統計情報へのアクセスなど、ネットワークの可視性およびプログラム可能性をサポートする機能。
- OpenFlow などのモジュラ サウスバウンド インターフェイスのサポートを有効にするサービス抽象化層（SAL）。
- GUI を使用するか、または Java または Representational State Transfer（REST）Northbound API を使用した一貫した管理アクセス。

- ロール ベースのアクセス コントロール (RBAC) などのセキュリティ機能、および認証、許可、アカウンティング（AAA）機能のために RADIUS または TACACS を使用した外部 Active Directory との統合。
- 分析収集および診断パケット インジェクションなどのトラブルシューティング ツール。
- データ フローがネットワークを通過するパスを管理者がカスタマイズできる Topology Independent Forwarding（TIF）などのシスコの拡張機能。
- フロー スペックを使用してネットワークを論理パーティションに分割できる Network Slicing などのシスコのネットワーク アプリケーション。
- 拡張性と高可用性を実現する高可用性クラスタ。

Cisco XNC のシステム要件

Cisco XNC は JVM で実行されます。 Java ベースのアプリケーションのように、Cisco XNC は Linux ベースの x86 サーバ上で実行できます。 最適な結果を得るためには、次の点を推奨します。

- 2 GHz 以上の 1 個の 6 コア CPU。
- 最低 8 GB のメモリ。
- 最小 40 GB の空きハードディスク領域が、Cisco XNC アプリケーションをインストールするパーティションで使用できる必要があります。
- 次のような、Java をサポートする最新の 64 ビットの Linux ディストリビューション：
  - Ubuntu Linux
  - Red Hat Enterprise（RHEL）Linux
  - Fedora Linux
- Java Virtual Machine 1.7 以降のリリース
- プロファイルの $JAVA_HOME 環境変数が JVM のパスにセットされている。
- バックアップ スクリプトおよび復元スクリプトをサポートする Python 2.7.3。

Cisco XNC でサポートされている Web ブラウザ

Cisco XNC では、次の Web ブラウザがサポートされています。

- Firefox 18.x 以降のバージョン
- Chrome 24.x 以降のバージョン

（注） Javascript 1.5 以降のバージョンをブラウザで有効にする必要があります。

第 2 章

Cisco XNC の導入

この章の内容は、次のとおりです。

Cisco XNC のインストール

Cisco XNC アプリケーションのインストール

ステップ 1 Web ブラウザで、Cisco.com に移動します。

ステップ 2 [Support] で [All Downloads] をクリックします。

ステップ 3 中央のペインで、[Cloud and Systems Management] をクリックします。

ステップ 4 入力を求められたら、Cisco.com のユーザ名およびパスワードを入力して、ログインします。

ステップ 5 右側のペインで、[Network Controllers and Applications] をクリックし、[Cisco Extensible Network Controller (XNC)] をクリックします。

ステップ 6 Cisco XNC アプリケーション バンドルおよび購入した追加のアプリケーションをダウンロードします。

ステップ 7 Linux マシンで、Cisco XNC をインストールするディレクトリを作成します。
たとえば、ホーム ディレクトリに、`CiscoXNC` を作成します。

ステップ 8 作成したディレクトリに Cisco XNC の zip ファイルをコピーします。

ステップ 9 Cisco XNC の zip ファイルを解凍します。
Cisco XNC ソフトウェアが xnc というディレクトリにインストールされます。 ディレクトリには、次の内容が含まれます。

- xncbundle ファイル：Cisco XNC アプリケーション バンドル。
- runxnc.sh ファイル：Linux または UNIX システムで Cisco XNC を起動するためにユーザが使用するファイル。
- version.properties ファイル：Cisco XNC のビルド バージョン。
- adminpasswordreset.sh ファイル：デフォルトの network-admin ユーザ パスワードを工場出荷時のデフォルトにリセットするスクリプト。
- backup.py ファイル：Cisco XNC バックアップ スクリプト。
- configkeystorepwd.sh ファイル：TLS キーストア パスワード設定スクリプト。
- captures ディレクトリ：Cisco XNC で実行された分析の出力ダンプ ファイルが含まれるディレクトリ。
- configuration ディレクトリ：Cisco XNC の基本的な初期設定ファイルが含まれるディレクトリ。
  このディレクトリには、GUI 設定が保存されている startup サブ ディレクトリが含まれます。
- etc ディレクトリ：プロファイル情報が含まれるディレクトリ。
- lib ディレクトリ：Cisco XNC の Java ライブラリが含まれるディレクトリ。
- logs ディレクトリ：Cisco XNC ログが含まれるディレクトリ。

  （注） logs ディレクトリは、Cisco XNC アプリケーションが起動された後に作例されます。

- plugins ディレクトリ：OSGi プラグインが含まれるディレクトリ。
- ObjectStore ディレクトリ：Cisco XNC オブジェクトが含まれるディレクトリ。
- work ディレクトリ：Cisco XNC アプリケーションが起動された後に作成される Web サーバの作業ディレクトリ。

追加の Cisco XNC アプリケーションのインストール

はじめる前に

追加の Cisco XNC アプリケーションを購入し、Cisco.com から .zip ファイルをダウンロードする必要があります。新しいアプリケーションをインストールする前に設定をバックアップすることを推奨します。

ステップ 1 Cisco XNC をインストールしたコマンド ウィンドウを開きます。

ステップ 2 アプリケーション ファイルを解凍し、ソフトウェアをインストールしたときに作成された `xnc/plugins` ディレクトリに .jar ファイルを置きます。

Cisco XNC アプリケーションの起動

ステップ 1 ソフトウェアをインストールしたときに作成された `xnc` ディレクトリに移動します。

ステップ 2 構文 **./runxnc.sh** を使用して Cisco XNC を起動します。
次のいずれかのオプションを選択できます。

オプション	説明
オプションなし	**- start** オプションを指定して Cisco XNC を起動します。
-jmx	Cisco XNC JVM 上で JMX のリモート アクセスを有効にします。これは、パフォーマンスの問題をトラブルシューティングするために使用します。
-jmxport *num*	指定した JVM ポートで JMX リモート アクセスを有効にします。
-debug	Cisco XNC JVM のデバッグを有効にします。
-debugsuspend	デバッガが接続されるまで Cisco XNC の起動を一時停止します。
-debugport *port_number*	指定した JVM ポートでのデバッグを有効にします。
-start	Cisco XNC を起動し、ポート 2400 上のコントローラにセキュア シェル（SSH）アクセスします。 （注） SSH サーバは、ネットワーク管理者ロールを持つ Cisco XNC ユーザがアクセスできます。

オプション	説明
-start *port_num*	Cisco XNC を起動し、指定したポート番号上のコントローラに SSH アクセスします。 （注） SSH サーバは、ネットワーク管理者ロールを持つ Cisco XNC ユーザがアクセスできます。
-stop	Cisco XNC を停止します。
-status	Cisco XNC のステータスを表示します。
-console	OSGi コンソールを使用して Cisco XNC を起動します。
-help	runxnc.sh スクリプトのオプションを表示します。
-tls	Cisco XNC と OpenFlow スイッチ間の TLS セキュア接続を有効にします。 TLS を有効にするには、次のオプションを指定してコントローラを起動します。**./runxnc.sh -tls -tlskeystore** *keystore_file_location* **-tlstruststore** *truststore_file_location*

Cisco XNC が実行されていることの確認

ステップ 1 Cisco XNC をインストールしたコマンド ウィンドウを開きます。

ステップ 2 ソフトウェアをインストールしたときに作成された xnc ディレクトリに移動します。

ステップ 3 次のスクリプトを実行します。**./runxnc.sh -status**
コントローラは次を出力し、コントローラが PID 21680 の Java プロセスを実行していることを示します。

```
Controller with PID:21680 -- Running!
```

次の作業

コントローラにスイッチを接続します。詳細については、適切なコンフィギュレーションガイドを参照してください。

TLS キーストア ファイルと信頼ストア ファイルの使用方法

Cisco XNC と OpenFlow スイッチ間の Transport Layer Security（TLS）接続を有効にするには、TLS キー ストア ファイルと TLS 信頼ストア ファイルが必要です。

- TLS のキーストア ファイルには、Cisco XNC が使用する秘密キーと証明書情報が含まれます。
- TLS 信頼ストア ファイルには、OpenFlow スイッチの証明書に署名するために使用される認証局（CA）の証明書が含まれます。

TLS キーストア ファイルと TLS 信頼ストア ファイルの両方がパスワードで保護されています。

Cisco XNC の実装において TLS 接続を使用する場合は、ネットワーク内の接続はすべて暗号化する必要があり、TLS を有効にして Cisco XNC を実行する必要があります。 TLS キーストア ファイルと TLS 信頼ストア ファイルの両方を用意したら、Cisco XNC がキーストア ファイルのロックを解除できるように、TLS キーストア パスワード設定スクリプトを実行してパスワードを指定できます。

TLS キーストア ファイルの作成

ステップ 1 次のファイルを用意します。

- xnc-private.pem：Cisco XNC 秘密キーを含む .pem ファイル。
- xnc-cert.pem：Cisco XNC 証明書を含む .pem ファイル。

ステップ 2 次のコマンドを実行します。**cat xnc-privkey.pem xnc-cert.pem > xnc.pem**
秘密キーと証明書を含む xnc.pem ファイルが作成されます。

ステップ 3 次のコマンドを実行します。**openssl pkcs12 -export -out xnc.p12 -in xnc.pem**

ステップ 4 プロンプトでパスワードを入力します。

（注） ステップ 4 およびステップ 6 で同じパスワードを使用する必要があります。 このパスワードは、6 文字以上にする必要があります。

xnc.pem ファイルはパスワード保護された .p12 ファイルに変換されます。

ステップ 5 次のコマンドを実行します。**keytool -importkeystore -srckeystore xnc.p12 -srcstoretype pkcs12 -destkeystore tlsKeyStore -deststoretype jks**

ステップ 6 プロンプトでパスワードを入力します。

（注） ステップ 4 およびステップ 6 で同じパスワードを使用する必要があります。 このパスワードは、6 文字以上にする必要があります。

xnc.p12 はパスワード保護された Java キーストア ファイルに変換されます。

TLS 信頼ストア ファイルの作成

ステップ 1 スイッチの CA 証明書を含む cacert.pem というファイルを作成します。

ステップ 2 次のコマンドを実行します。**keytool -import -alias swca1 -file sw-cacert.pem -keystore tlsTrustStore**

ステップ 3 プロンプトでパスワードを入力します。
cacert.pem ファイルがパスワード保護された Java 信頼ストア ファイルに変換されます。

ステップ 4 スイッチがネットワークで複数の CA 証明書を使用する場合、使用する証明書ごとにステップ 1 ～ 3 を繰り返します。

TLS キーストア パスワード設定スクリプトの実行

configkeystorepwd.sh スクリプトを使用すると、Cisco XNC がキーストア ファイルをロック解除して使用できるように TLS キー ストア パスワードを入力できます。

はじめる前に

cURL プログラムがインストールされていることを確認します。

ステップ 1 Cisco XNC が TLS を有効にして実行されていることを確認します。

ステップ 2 Cisco XNC をインストールしたコマンド ウィンドウを開きます。

ステップ 3 ソフトウェアをインストールしたときに作成された `xnc` ディレクトリに移動します。

ステップ 4 次のコマンドを実行します。**./configkeystorepwd.sh**

ステップ 5 プロンプトで、次の情報を入力します。

- Cisco XNC のユーザ名
- Cisco XNC のパスワード
- TLS キーストアのパスワード
- TLS 信頼ストアのパスワード

Cisco XNC GUI へのログイン

HTTP または HTTPS を使用して Cisco XNC GUI にログインできます。

- Cisco XNC GUI のデフォルトの HTTP Web リンクは、`http://Controller_IP:8080` です
- Cisco XNC GUI のデフォルトの HTTPS Web リンクは、`https://Controller_IP:8443` です

（注） HTTPS を使用するには、Web ブラウザに https:// プロトコルを指定します。

ステップ 1 Web ブラウザに、Cisco XNC GUI の Web リンクを入力します。

ステップ 2 起動ページで、次の作業を行います。

a) ユーザ名とパスワードを入力します。
デフォルトのユーザ名とパスワードは admin と admin です。

b) [Log In] をクリックします。

Cisco XNC の設定

ハイ アベイラビリティ クラスタの設定

Cisco XNC は、最大 5 つのコントローラを使用したアクティブ/アクティブ モードのハイ アベイラビリティ クラスタリングをサポートします。 Cisco XNC でハイ アベイラビリティ クラスタリングを使用するには、Cisco XNC のインスタンスごとに config.ini ファイルを編集する必要があります。

はじめる前に

- すべての IP アドレスは、到達可能で、相互に通信できる必要があります。
- クラスタ内のすべてのスイッチは、すべてのコントローラに接続する必要があります。
- すべてのコントローラは、まったく同じ HA クラスタリング設定情報を config.ini ファイルに持つ必要があります。
- すべてのコントローラは、まったく同じ情報を `xnc/configuration/startup` ディレクトリに持つ必要があります。

- クラスタ パスワードを使用する場合、すべてのコントローラはまったく同じパスワードを xncjgroups.xml ファイルに設定する必要があります。ハイ アベイラビリティ クラスタのパスワード保護, （12 ページ）を参照してください。

ステップ 1 クラスタ内のインスタンス上で Cisco XNC が実行されていないことを確認します。

ステップ 2 クラスタ内のインスタンスの 1 つでコマンド ウィンドウを開きます。

ステップ 3 ソフトウェアをインストールしたときに作成された `xnc/configuration` ディレクトリに移動します。

ステップ 4 任意のテキスト エディタで `config.ini` ファイルを開きます。

ステップ 5 次のテキストを探します。

```
# HA Clustering configuration (colon-separated IP addresses of all controllers that are part of
the cluster.)
# supernodes=<ip1>:<ip2>:<ip3>:<ipn>
```

ステップ 6 # supernodes 行のコメントを削除し、<ip1>:<ip2><ip3>:<ipn> をクラスタ内の各 Cisco XNC インスタンスの IP アドレスで置き換えます。 2 ～ 5 の IP アドレスを入力できます。

例：

```
# HA Clustering configuration (colon-separated IP addresses of all controllers that are part of
the cluster.)
supernodes=<10.1.1.1>:<10.2.1.1>:<10.3.1.1>:<10.4.1.1>:<10.5.1.1>
```

ステップ 7 ファイルを保存し、エディタを終了します。

ステップ 8 クラスタ内の Cisco XNC の各インスタンスに対してステップ 3 ～ 7 を繰り返します。

ステップ 9 Cisco XNC を再起動します。

ハイ アベイラビリティ クラスタのパスワード保護

xncjgroups.xml ファイルを使用して HA クラスタをパスワードで保護できます。 このファイルは、Cisco XNC の各インスタンスに対してまったく同じにする必要があります。

ステップ 1 クラスタ内のインスタンス上で Cisco XNC が実行されていないことを確認します。

ステップ 2 クラスタ内のインスタンスの 1 つでコマンド ウィンドウを開きます。

ステップ 3 ソフトウェアをインストールしたときに作成された `xnc/configuration` ディレクトリに移動します。

ステップ 4 任意のテキスト エディタで `xncjgroups.xmli` ファイルを開きます。

ステップ 5 次のテキストを探します。

```
<!-- <AUTH auth_class="org.jgroups.auth.MD5Token" auth_value="ciscoXNC" token_hash="MD5"></AUTH>
-->
```

ステップ 6 AUTH 行からコメントを削除します。

例：
```
<AUTH auth_class="org.jgroups.auth.MD5Token" auth_value="ciscoXNC" token_hash="MD5"></AUTH>
```

ステップ 7 （任意） auth_value 属性のパスワードを変更します。
デフォルトでは、クラスタはパスワード "ciscoXNC" で保護されています。 クラスタ内のすべてのマシン上で同じ変更を行うという条件で、このパスワードをどんな値にでも変更できます。

ステップ 8 ファイルを保存し、エディタを終了します。

ステップ 9 クラスタ内の Cisco XNC の各インスタンスに対してステップ 4 ～ 8 を繰り返します。

ステップ 10 Cisco XNC を再起動します。

Cisco Nexus 3000 シリーズ スイッチのコンフィギュレーション ファイルの編集

次の設定により、Cisco Nexus 3000 シリーズ スイッチに接続する場合に拡張性を向上できます。

ステップ 1 ソフトウェアをインストールしたときに作成された `xnc/configuration` ディレクトリに移動します。

ステップ 2 任意のテキスト エディタで `config.ini` ファイルを開きます。

ステップ 3 次のパラメータを更新します。

名前	デフォルト値	推奨値
of.messageResponseTimer	2000	60000
of.switchLivenessTimeout	60500	120500
of.flowStatsPollInterval	10	240
of.portStatsPollInterval	10	240
of.descStatsPollInterval	60	240
of.barrierMessagePriorCount	100	50
of.discoveryInterval	300	300
of.discoveryTimeoutMultiple	2	2

ステップ 4 ファイルを保存し、エディタを終了します。

ステップ 5 Cisco XNC を再起動します。

バックアップおよび復元スクリプトの実行

バックアップ スクリプトを使用すると、Cisco XNC の設定をバックアップし、後で復元できます。

ステップ 1 Cisco XNC をインストールしたコマンド ウィンドウを開きます。

ステップ 2 ソフトウェアをインストールしたときに作成された xnc ディレクトリに移動します。

ステップ 3 次のコマンドを実行します。**python backup.py**

ステップ 4 プロンプトで、次のいずれかの作業を実行します。

- 最後の設定を保存するには、**backup** を入力します。

  スクリプトは、xnc ディレクトリに次の形式でタイムスタンプが設定された tar ファイルを作成します。xnc-*yy-mm-dd_time*.tar

- 保存したバックアップ ファイルを復元するには、**restore** *filename* と入力します。

  filename はバックアップ tar ファイルです。

  入力を求められたら、[Y] を選択して既存の設定を上書きします。

- プログラムを終了するには、**exit** を入力します。

ステップ 5 設定を復元する場合は、設定を有効にするために Cisco XNC を停止し、再起動します。

パスワード リカバリ スクリプトの実行

パスワード リカバリ スクリプトを使用すると、Cisco XNC ネットワーク管理者ユーザに工場出荷時のデフォルト パスワードが設定されます。

ステップ 1 Cisco XNC をインストールしたコマンド ウィンドウを開きます。

ステップ 2 ソフトウェアをインストールしたときに作成された xnc ディレクトリに移動します。

ステップ 3 次のコマンドを実行します。**./adminpasswordreset.sh**

ステップ 4 プロンプトに対して、[y] を選択してパスワードをリセットします。

Cisco XNC アプリケーションのアンインストール

ステップ 1 Cisco XNC のインストール ディレクトリを作成したディレクトリに移動します。
たとえば、Home/CiscoXNCにコントローラをインストールした場合、Homeディレクトリに移動します。

ステップ 2 CiscoXNC ディレクトリを削除します。