HP Network Node Manager i Software

ソフトウェアバージョン: 10.10
Windows®およびLinux®オペレーティングシステム

デプロイメントリファレンス

ドキュメントのリリース日 : 2015年11月
ソフトウェアのリリース日 : 2015年11月

ご注意

保証

HP製品とサービスに関する単独の保証は、かかる製品とサービスに付属する保証ステートメントに明示的に定められています。ここに記載された情報は追加の保証をなすものではありません。HPではここに記載されている技術的、または編集上の不正確さや脱漏については責任を負いません。

ここに記載されている情報は予告なく変更されることがあります。

制限付き権利

機密コンピューターソフトウェア所有、使用、またはコピーに必要なHP提供の有効ライセンス。FAR 12.211および12.212に準拠し、商用コンピューターソフトウェア、コンピューターソフトウェアドキュメント、および商用アイテムの技術データは、ベンダーの標準商用ライセンスの下、米国政府にライセンスされています。

Oracleテクノロジの制限された権限に関する通知

DOD FAR Supplementによって届けられたプログラムは、「商業用コンピューターソフトウェア」であり、ドキュメントを含むプログラムの使用、複製、開示についてはOracleの適切なライセンス契約に基づくライセンス制限に拠る必要があります。それ以外の場合は、連邦調達規則に従って供給されたプログラムは、「制限されたコンピューターソフトウェア」であり、関連文書を含むプログラムの使用、複製、および公開は、FAR 52.227-19、『商用コンピューターソフトウェア - 制限された権限』(1987年6月) に記載されている制限に従うものとします。Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Oracleライセンスの全文は、NNMiの製品DVDにあるlicense-agreementsのディレクトリを参照してください。

著作権

商標について

Adobe ® はAdobe Systems Incorporatedの登録商標です。

Appleは、米国およびその他の国で登録されたApple Computer, Incの商標です。

AMDは、Advanced Micro Devices, Inc.の商標です。

Google™ は、Google Inc.の登録商標です。

Intel ® 、Intel ® Itanium ® 、Intel ® Xeon ® 、およびItanium ® は、米国およびその他の国におけるIntel Coporationの商標です。

Linux ® は、Linus Torvalds氏の米国およびその他の国における登録商標です。

Internet Explorer、Lync、Microsoft、Windows、およびWindows Serverは、米国およびその他の国における Microsoft Corporationの登録商標または商標です。

OracleおよびJavaはOracleおよびその関連会社の登録商標です。

Red Hat ® Enterprise Linux Certifiedは、米国およびその他の国におけるRed Hat, Incの登録商標です。

sFlowは、InMon Corp.の登録商標です。

UNIX ® はThe Open Groupの登録商標です。

謝辞

この製品には、Apache Software Foundationで開発されたソフトウェアが含まれています。(http://www.apache.org/)

この製品には、Visigoth Software Society (http://www.visigoths.org/) によって開発されたソフトウェアが含まれています。

マニュアル更新

このドキュメントのタイトルページには、次の識別情報が含まれています。

- ソフトウェアーバージョン番号。ソフトウェアーのバージョンを示します。
- ドキュメントリリース日。ドキュメントが更新されるたびに変更されます。
- ソフトウェアリリース日。ソフトウェアのこのバージョンのリリース日を示します。

最近の更新を確認するか、ドキュメントの最新版を使用していることを確認するには、次のサイトを参照してください。**https://softwaresupport.hp.com**

このサイトでは、HPパスポートに登録してサインインする必要があります。HPパスポートIDに登録するには、次のURLにアクセスしてください。**https://hpp12.passport.hp.com/hppcf/createuser.do**

または、[HPソフトウェアサポート] ページ上部にある [**登録**] リンクをクリックしてください。

適切な製品サポートサービスの契約をしている場合は、更新版または新版を受信することもできます。詳細については、HPの営業担当者に問い合わせてください。

サポート

HPソフトウェアーサポートオンラインWebサイトへのアクセス:**https://softwaresupport.hp.com**

このWebサイトでは、製品、サービス、およびHPソフトウェアーが提供するサポートに関する詳細と連絡先の情報を提供します。

HPソフトウェアーオンラインサポートでは、お客様ご自身で問題を解決できるケーパビリティを提供しています。すばやく効率的な方法で、お客様のビジネス管理に必要な対話型テクニカルサポートツールにアクセスできます。サポートの大切なお客様として、サポートWebサイトで次の操作が可能です。

- 興味のあるナレッジドキュメントの検索
- サポート事例と改善要求の送信と追跡
- ソフトウェアーパッチのダウンロード
- サポート契約の管理
- HPサポート契約の検索
- 利用可能なサービスに関する情報のレビュー
- 他のソフトウェアーユーザーとの情報交換
- ソフトウェアートレーニングの調査と登録

ほとんどのサポートエリアでは、HPパスポートのユーザーとして登録してサインインする必要があります。また、多くのエリアではサポート契約も必要です。HPパスポートIDに登録するには、次のURLにアクセスしてください。

https://hpp12.passport.hp.com/hppcf/createuser.do

アクセスレベルの詳細については、次のURLにアクセスしてください。

https://softwaresupport.hp.com/web/softwaresupport/access-levels

HP Software Solutions Now (英語) はHPSWのソリューションと統合に関するポータルWebサイトです。このサイトでは、お客様のビジネスニーズを満たすHP製品ソリューションを検索したり、HP製品間の統合に関する

http://h20230.www2.hp.com/sc/solutions/index.jsp です。

目次

第1章: このガイドについて

この章には、以下のトピックがあります。

ガイドの説明

このガイドには、NNMi PremiumやNNMi Ultimateなど、HP Network Node Manager i Softwareを配備するための情報およびベストプラクティスが記載されています。対象読者は、熟練したシステム管理者、ネットワークエンジニアー、または大規模システムのネットワークデプロイメントおよび管理に経験のあるHPサポートエンジニアーです。

このガイドでは、制限のある環境 (またはテスト環境) にNNMiをインストール済みであること、クイックスタート設定ウィザードを使用したコミュニティ文字列の設定、ネットワークノードの制限範囲の検出設定、初期管理者アカウントの作成のような、設定作業の開始に慣れていることを想定しています。これらの作業の詳細については、『HP Network Node Manager i Softwareインタラクティブインストールガイド』(「使用可能な製品ドキュメント」) を参照してください。

新しい情報が入手可能になると、製品リリースの間に、HPはこのガイドを更新します。ドキュメントの更新バージョン取得の詳細については、「使用可能な製品ドキュメント」を参照してください。

このドキュメントで使用するパス表記

このドキュメントでは、主に以下の2つのNNMi環境変数を使用して、ファイルやディレクトリの場所を参照します。以下に示す変数はデフォルト値です。実際の値は、NNMiのインストール時に行った選択内容によって異なります。

- Windows Serverの場合:
  - `%NnmInstallDir%:<drive>\Program Files (x86)\HP\HP BTO Software`
  - `%NnmDataDir%:<drive>\ProgramData\HP\HP BTO Software`

  Windowsシステムでは、以下の点に注意してください。
  - NNMiのインストールプロセスによってこれらのシステム環境変数が作成されるため、すべてのユーザーがいつでも使用できます。
  - パス名にスペースが含まれる場合は必ず引用符を使用します (例:"%NnmInstallDir%\bin\ovstatus" -c)。
- Linuxの場合:
  - `$NnmInstallDir:/opt/OV`
  - `$NnmDataDir:/var/opt/OV`

> **注:** Linuxシステムでは、これらの環境変数を使用する場合は手動で作成する必要があります。

また、このドキュメントには、NNMi管理サーバーでユーザーログオン設定を行うときに使用するNNMi環境変数も一部掲載されています。これらの変数の形式は`NNM_*`です。NNMi環境変数の詳細リストについては、「他の使用可能な環境変数」(504ページ) を参照してください。

改訂履歴

次の表に、このドキュメントの新規リリースごとの主要な変更をリストします。

ドキュメントリリース日	主要な変更の説明
2014年5月 (10.00)	初期リリース。
2014年12月 (10.01)	「インシデントの設定」の章に「**ソート順序に使用するロケールの設定**」を追加しました。 「NNMiセキュリティ」の章に「**NNMiでSSLv3暗号を有効化または無効化する設定**」を追加しました。 「詳細設定」の章の情報「**NNMiでの証明書の使用**」を更新しました。

ドキュメントリリース日	主要な変更の説明
2015年11月 (10.10)	「NNMi通信」の章に「**仮想環境における通信の設定**」を追加しました。 「NNMiとLDAPによるディレクトリサービスの統合」の章から「**ディレクトリサービスのアクセス設定を変更し、NNMiのセキュリティモデルをサポートする**」を削除しました。

NNMiの詳細

NNMi製品の完全な情報を入手するには、このガイドと他のNNMiドキュメントを一緒に使用してください。次の表に、現在までのすべてのNNMiドキュメントを示します。ガイドとホワイトペーパーの両方を含みます。

注: 情報はすべて**http://h20230.www2.hp.com/selfsolve/manuals**からダウンロードできます。詳細については、「使用可能な製品ドキュメント」を参照してください。

目的	詳しい情報の参照先
このバージョンのNNMiで入手可能な文章の一覧を表示する。	「NNMiドキュメント一覧」をダウンロードします。このファイルを使用して、このバージョンのNNMiのNNMiドキュメントセットにある追加や改訂を調べることができます。リンクをクリックして、HPマニュアルWebサイト上のドキュメントにアクセスします。
NNMi、NNMi Advanced、NNMi Premium、またはNNMi Ultimateをインストールする (初回)。	『HP Network Node Manager i Softwareインタラクティブインストールガイド』をダウンロードします。このガイドには、製品をインストールおよびアンインストールする基本手順、およびNNMiクイックスタート設定ウィザードを使用して初期設定を行う方法が記載してあります。
ネットワーク導入の計画 (システム要件へのリンクを含む)。	このガイドの「準備」(27ページ)を参照してください。
製品環境向けにNNMiを設定する。	このガイドの「設定」(30ページ)を参照してください。
VMwareハイパーバイザーベースの仮想ネットワークについてNNMi設定の考慮事項を確認する。	管理者用のヘルプの「VMwareハイパーバイザーベースの仮想ネットワークの検出とモニタリング (NNMi Advanced)」と「VMware ハイパーバイザーベースの仮想ネットワークの管理 (NNMi Advanced)」を参照してください。
NNMiの高度設定を行う。	このガイドの「詳細設定」(312ページ)を参照してくだ

目的	詳しい情報の参照先
	さい。
NNMiの設定を維持管理する。	このガイドの「NNMiのメンテナンス」(243ページ) を参照してください。
Network Node Manager i Softwareの前バージョンからNNMiにアップグレードする。	HPマニュアルWebサイトにあるHP Network Node Manager i Softwareインタラクティブインストールガイドを参照してください。
NNMi環境変数、ポート、メッセージのリファレンスを参照する。	このガイドの「追加情報」(499ページ) を参照してください。
特定のトピックに関する詳細情報を取得する。	サンプルドキュメントやホワイトペーパーからダウンロードします。入手可能なホワイトペーパーの一覧は、「NNMiドキュメント一覧」を参照してください。
NNMiヘルプを印刷する。	ヘルプコンテンツのPDFをダウンロードします。入手可能なヘルプPDFの一覧は、「NNMiドキュメント一覧」を参照してください。
HP NNM iSPI NET (NNM iSPI NET) 診断サーバーをインストールし、NNM iSPI NETの機能について学ぶ。	Network Node Manager SPI for NET製品カテゴリから、Windowsオペレーティングシステム用の『HP NNM iSPI Network Engineering Toolset計画とインストールガイド』(HP NNM iSPI Network Engineering Toolset Planning and Installation Guide) をダウンロードします。 **注:** NNM iSPI NET診断サーバーにはNNM iSPI NETライセンスまたはNNMi Ultimateライセンスが必要です。このサーバーのインストール方法および設定方法については、『HP NNM iSPIネットワークエンジニアリングツールセットソフトウェア計画とインストールガイド』(HP NNM iSPI Network Engineering Toolset Software Planning and Installation Guide)を参照してください。
NNMi開発者ツールキット (SDK) のドキュメントを入手する。	「NNMiのライセンス」(312ページ)を参照して、SDKの関連情報、SDKライセンスの取得およびインストール、SDKのドキュメントおよびサンプルを確認してください。

第2章: 準備

このセクションでは以下の章について説明します。

ハードウェアとソフトウェアの要件

この章には、以下のトピックがあります。

サポートされるハードウェアとソフトウェア

NNMiをインストールする前に、以下の表で説明するNNMiのハードウェアとソフトウェアの要件に関する情報を読んでください。

注: ここに示すすべてのドキュメントの最新版は、次のサイトから取得できます。

http://h20230.www2.hp.com/selfsolve/manuals

ソフトウェアおよびハードウェアのプレインストールのチェックリスト

チェック欄 (はい/いいえ)	確認していただくドキュメント
	HP Network Node Manager i Softwareインタラクティブインストールガイド • **ファイル名** = nnmi_interactive_installation_ja.zipまたはnnmi_interactive_installation_ja.jar • **指示ファイル名**:nnmi_interactive_installation_en_README.txt

ソフトウェアおよびハードウェアのプレインストールのチェックリスト (続き)

チェック欄 (はい/いいえ)	確認していただくドキュメント
	NNMiリリースノート • **ファイル名** = release_notes_nnmi_en.pdf • **NNMiコンソール** = **[ヘルプ] > [NNMi ドキュメントライセンス] > [リリースノート]**
	NNMi対応マトリックス • **ファイル名** = support_matrix_nnmi_en.pdf • **NNMiコンソール** =リリースノートからリンクしている

注: 新しい情報が入手可能になると、HPは『NNMi対応マトリックス』を更新します。NNMiを配備する前に、以下のWebサイトで、お持ちのバージョンのソフトウェアに関する最新のNNMi対応マトリックスをチェックしてください。

http://www.hp.com/go/hpsoftwaresupport/support_matrices

(このWebサイトにアクセスするには、HP PassportのIDが必要です。)

注: NNMスマートプラグイン (NNM iSPIs) をインストールする場合は、NNMi導入時に、これらの製品のシステム要件を組み入れてください。

必要なパッチの確認

NNMiをインストールする前に、必要なオペレーティングシステム更新の有無をNNMiリリースノートで確認してください。

システム設定 (Linux)

NNMi管理サーバーにNNMiのマニュアルページを表示できない場合は、MANPATH変数に/opt/OV/manの場所が含まれていることを確認します。含まれていない場合は、/opt/OV/manの場所をMANPATH変数に追加します。

NNMiおよびNNM iSPIのインストール

いずれかのHP NNM iSPIsをNNMiとともに使用する場合、HP NNM iSPIsをインストールする前に、NNMiをインストールする必要があります。

NNMiの共存

NNMiを他のHP製品と併用する場合は、以下の点に注意してください。

- (HP Operations Manager (HPOM) と通信するために) NNMi管理サーバーにHP Operationsエージェントをインストールする場合は、HP Operationsエージェントをインストールする前にNNMiをインストールします。

注: Network Performance Server (NPS) もインストールしている場合、NPSはNNMiの後、Operationsエージェントの前にインストールする必要があります。

- RHEL7.xシステム上のHP Business Service Management Connector (BSMC) バージョン10.00のみ: NNMi管理サーバーにBSMCをインストールする場合は、NNMiをインストールする前にBSMCをインストールしてください。

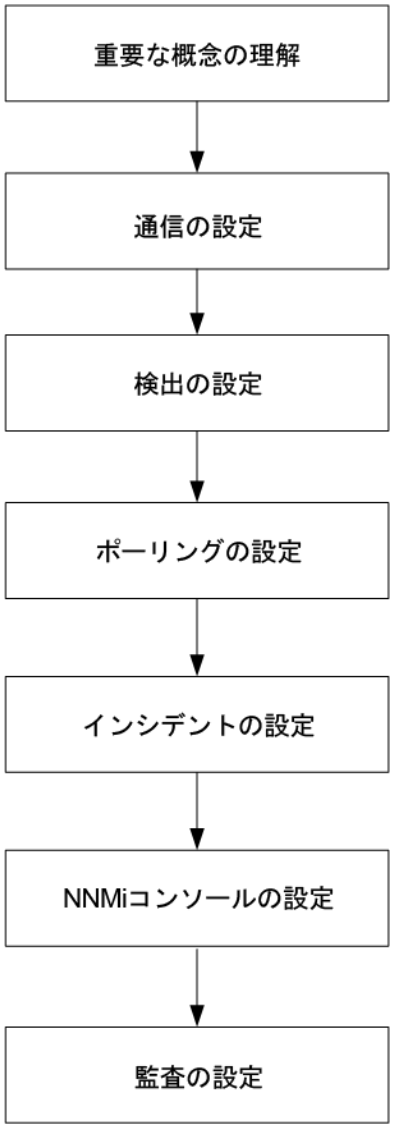
NNM i Smart Plug-Inのバージョン要件

NNMiと各NNM i Smart Plug-Inは、同等のバージョンである必要があります。たとえば、NNM iSPI Performance for Metricsバージョン10.10は、NNMi 10.10でのみサポートされています。

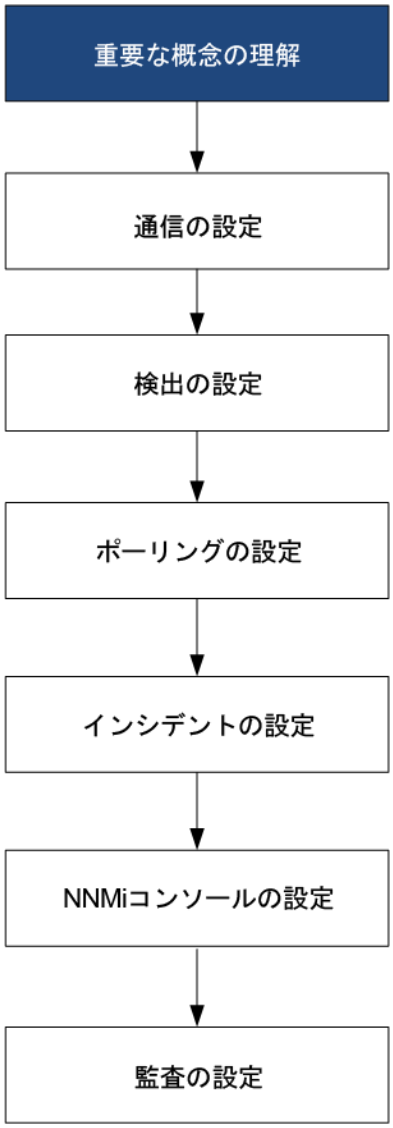
NNMi PremiumとNNMi Ultimateに含まれているiSPIのリストについては、http://h20230.www2.hp.com/selfsolve/manualsで入手できる『NNMiリリースノート』を参照してください。

第3章: 設定

このセクションでは以下の章について説明します。


重要な概念の理解
通信の設定
検出の設定
ポーリングの設定
インシデントの設定
NNMiコンソールの設定
監査の設定

設定の一般概念



この章では概念の概論を説明しています。詳細については、このガイドの後のほうで説明しています。この章では、すべてのHP Network Node Manager i Software (NNMi) 設定領域に適用されるベストプラクティスについても記載しています。

この章には、以下のトピックがあります。

- 「ノードグループおよびインタフェースグループ」(35ページ)
- 「ノードインタフェースおよびアドレス階層」(40ページ)
- 「NNMi設定およびデータベースのリセット」(40ページ)

タスクフローモデル

このガイドの設定の各章では、以下のタスクフローに役立つ情報を記載しています。

1. **概念**—設定領域の概略を理解できます。このガイドの情報は、NNMiヘルプの情報を補足しています。
2. **計画**—設定にどのように取り組むかを決定します。これは、会社のネットワーク管理のマニュアル化を開始または更新するよい機会です。
3. **設定**—NNMiコンソール、設定ファイル、コマンドラインインタフェースの組み合わせを使用して、設定をNNMiに入力します。具体的な手順については、NNMiヘルプを参照してください。

   注意: コマンドラインインタフェース(PSQLコマンドなど)や外部ユーティリティ使用して、組み込みデータベースの設定を作成、修正、または変更することはできません。これを行おうとすると、データベースに取り返しのつかない損傷を与える可能性があります。
4. **評価**—NNMiコンソールで、設定結果を確認します。設定を最適なものにするために、必要に応じて調節します。
5. **調整**—省略可能。設定を調整して、NNMiのパフォーマンスを向上します。

ベストプラクティス:既存の設定を保存する

大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。設定を変更した結果が気に入らなくても、保存した設定に簡単に戻すことができます。

`nnmconfigexport.ovpl`コマンドを使用して、現在の設定を保存します。保存した設定を復元するには、nnmconfigimport.ovplコマンドを使用します。

これらのコマンドの使用方法の詳細については、該当するリファレンスページ、またはLinuxのマニュアルページを参照してください。

ヒント: `nnmconfigexport.ovpl`コマンドではSNMPv3資格情報は保持されません。詳細については、nnmconfigexport.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

『HP Network Node Manager i Softwareステップバイステップガイド(NNMiインポートおよびエクスポートツールの使用に関するホワイトペーパー)』(HP Network Node Manager i Software Step-by-Step Guide to Using NNMi Import and Export Tools White Paper)も参照してください。

ベストプラクティス:作成者属性を使用する

多くのNNMi設定フォームには、**作成者**属性が含まれています。

これらのフォーム上で設定を作成や変更する際は、作成者の組織がわかる値に[**作成者**]属性を設定してください。NNMi設定をエクスポートするときに、作成者値を指定して作成者の組織がカスタマイズした項目のみを引き出すことができます。

NNMiをアップグレードする際、作成者値がHPではない設定は上書きされません。

ユーザーインタフェースモデル

一部のNNMiコンソールフォームでは、データベースの更新にトランザクションアプローチが使用されます。NNMiコンソールのフォームで行った変更は、フォームを保存して閉じる操作がNNMiコンソールまで行われないと有効になりません。保存されていない変更(フォーム上または含まれるフォーム上)が含まれるフォームを閉じると、NNMiによって保存されていない変更があるため、終了を取り消すよう求める警告が表示されます。

注: [**検出シード**]フォームは、トランザクションアプローチの例外です。このフォームは便宜上[**検出の設定**]フォーム上にありますが、他の検出設定からは切り離されています。このため、[**検出の設定**]フォームを保存して閉じ、自動検出ルールを実装した後で、これらの自動検出ルールに検出シードを設定する必要があります。

順序

いくつかのNNMiコンソール設定フォームには、設定を適用する優先順位を設定する**順序**属性が含まれています。ある設定領域で、NNMiは設定内容に対して各項目を、順序番号が最も小さい(低い)ものから大きいものへの順に、NNMiが一致を見つけるまで評価し続けます。一致が見つかった時点で、NNMiは一致する設定の情報を使用し、これ以上一致を探すのをやめます。(通信設定は例外です。NNMiは、通信設定を完了するためにその他のレベルで情報の検索を続行します。)

順序属性は、NNMiの設定で重要な役割を果たします。予想外の検出結果やステータス結果に遭遇した場合は、その領域の設定の順序を確認してください。

順序はローカルコンテンツ内で適用されます。[メニュー]および[メニュー項目]テーブルには、ローカルコンテキストであるため同じ順序番号の複数のオブジェクトが含まれます。

順序番号は次の箇所でも使用されますが、その意味は異なります。

- [**メニュー**]および[**メニュー項目**]フォームの順序で、関連メニューのローカルコンテキスト内の項目の順序が設定されます。
- [**ノードグループマップの設定**]フォームのトポロジマップ順序で、[**トポロジマップ**]ワークスペースの項目の順序が設定されます。

順序属性が指定の設定領域にどのように影響するかの情報については、その領域のNNMiヘルプを参照してください。

注: 各設定領域で、小さい順序番号は最も限定的な設定に適用し、大きな順序番号は限定度の最も低い設定に適用します。

注: 各設定領域で、すべての順序番号を一意にしてください。初期設定時は、通常の間隔の順序番号を使用して、将来設定を変更できるような柔軟性を確保しておいてください。たとえば、1番目から3番目の設定には100、200、300の順序番号を付けます。

ノードグループおよびインタフェースグループ

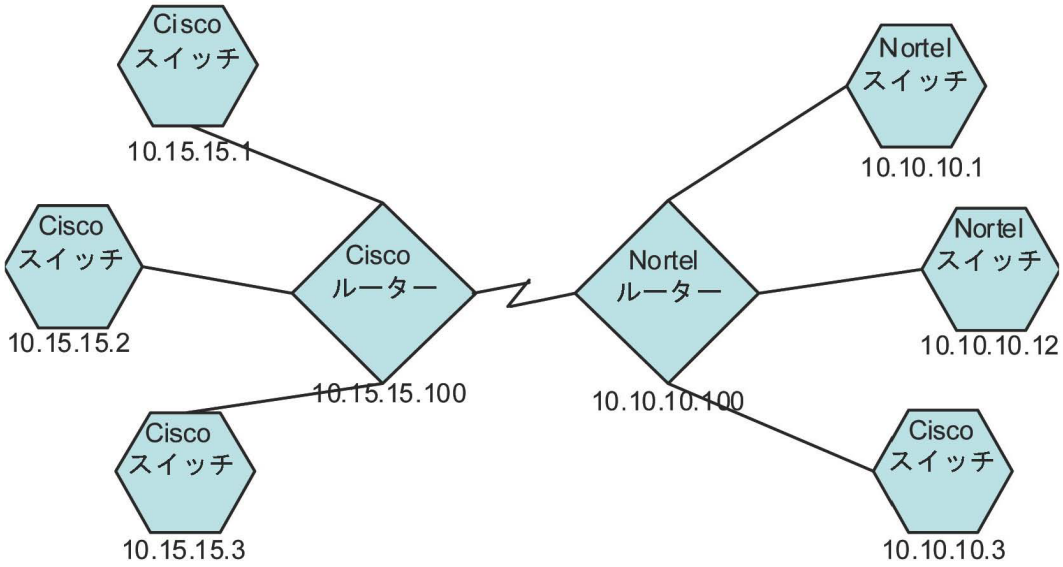
NNMiの基本的なフィルタリング手法では、ノードまたはインタフェースをグループ化してから、設定をグループに適用または可視化がグループ別にフィルタリングされます。

- ノードグループは、以下のいずれかまたはすべての目的に使用できます。
  - 監視設定
  - インシデント負荷量のフィルタリング
  - テーブルフィルタリング
  - マップビューのカスタマイズ
  - グローバルネットワーク管理機能のリージョナルマネージャーからグローバルマネージャーに渡されたノードのフィルタリング
- インタフェースグループは、以下のいずれかまたはすべての目的に使用できます。
  - 検出からのインタフェース除外
  - 監視設定
  - インシデント負荷量のフィルタリング
  - テーブルフィルタリング

グループの重複

グループ定義をどのように使用するかにかかわらず、最初のステップでは、どのノードまたはインタフェースをグループのメンバーにするかを定義します。さまざまな目的でグループが作成されるため、各々の対象が複数のグループに含まれる可能性があります。以下の例を考えてみます。

ノードグループの重複



- 監視を目的とした場合、ベンダーや場所を問わずすべてのスイッチに3分間のポーリング間隔を設定するのがよいでしょう。この場合は、デバイスカテゴリフィルターを使用します。
- 保守を目的とした場合はすべてのCiscoスイッチを1つのグループにし、IOSアップグレードでこのグループをまとめて [サービス停止中] にできるようにするのがよいでしょう。この場合は、ベンダーフィルターを使用します。
- 可視化の場合は、10.10.*.*サイト上のすべてのデバイスを、ステータスを反映したコンテナーにグループ化するのがよいでしょう。この場合は、IPアドレスフィルターを使用します。

IPアドレスが10.10.10.3のCiscoスイッチはこの3つのグループすべてに適しています。

設定や表示に便利なようにグループセットを豊富にするのもよいですが、使用されることのない必要以上のエントリを一覧に詰め込みすぎることのないよう、バランスをとってください。

ノードグループのメンバーシップ

NNMiは、検出した各ノードを、設定された各ノードグループと比較することにより、ノードグループのメンバーシップを判断します。

- [**追加のノード**] タブで指定したすべてのノードは、ノードグループのメンバーです。

  注意: NNMi管理サーバーのリソースを過度に消費するため、[**追加のノード**] タブを使用してノードグループにノードを追加することはほとんどありません。

- [**子ノードグループ**] タブで指定した少なくとも1つのノードグループのメンバーになっているすべてのノードは、そのノードグループのメンバーです。
- [**デバイスフィルター**] タブの1つ以上のエントリ (存在する場合)、および [**追加のフィルター**] タブで指定したフィルターに一致するすべてのノードは、そのノードグループのメンバーです。

階層/コンテインメント

単純で再利用可能な原子グループを作成し、これらを監視や可視化のために階層的に組み合わせることができます。階層的なノードのコンテナーを使用することにより、障害時にオブジェクトの場所やタイプに関する手がかりが得られるので、マップビューが大きく向上します。NNMiにより、グループの定義とそのドリルダウン順序の徹底管理が可能になります。

単純で再利用可能な原子グループを最初に作成し、その後にこれらを増築するときの子グループとして指定します。また、最初に一番大きな親グループを指定し、それから子グループを作成していくこともできます。

たとえば、ネットワークがCiscoスイッチ、Ciscoルーター、Nortelスイッチ、Nortelルーターで構成されているとします。Ciscoデバイスの親グループとすべてのスイッチの親グループを作成できます。親を作成してその子を指定するときに階層が指定されるので、Ciscoスイッチのようなそれぞれの子グループには複数の親ができる可能性があります。

階層は、以下の状況で使用すると効果的です。

- 監視 ニーズが類似したノードのタイプ
- ノードの地理的な配置
- まとめて [サービス停止中] にするノードのタイプ
- オペレーターの職務別のノードのグループ

マップビューおよびテーブルビューでグループを使用すると、伝達された (設定可能な) グループのステータスが表示されます。

注: グループ定義を使用して監視設定を指定する際に階層は設定の順序を示すのではないことを留意してください。小さい順序番号の設定は、ノードに適用されます。順序番号を注意深く増分することで、設定の継承概念を真似ることができます。

設定インタフェースでは、循環階層の定義が自動的に防御されます。

デバイスフィルター

検出中、NNMiは直接情報をSNMPクエリーで収集し、そこから他の情報を、デバイスのプロファイルルを通じて導き出します。(詳細については、「NNMiはデバイスのプロファイルルから属性を導き出す」(68ページ) を参照してください。)システムオブジェクトIDを収集することにより、NNMiは正しいデバイスのプロファイルルを通じて索引化して、次の情報を導き出します。

- ベンダー
- デバイスカテゴリ
- カテゴリ内のデバイスファミリ

導出されたこれらの値は、デバイスのプロファイルルそのものとともに、フィルターとして使用できます。

たとえば、特定のベンダー製のすべての対象物を、デバイスタイプやファミリに関係なくグループ化できます。また、ある種類のデバイス (たとえばルーター) をすべて、ベンダーを問わずにまとめることができます。

追加フィルター

追加のフィルターエディターを使用すると、以下のようなフィールドに一致するカスタム論理を作成できます。

- hostname (ホスト名)
- mgmtIPAddress (管理アドレス)
- hostedIPAddress (アドレス)
- sysName (システム名)
- sysLocation (システムのロケーション)
- sysContact (システムの連絡先)
- capability (機能の一意キー)
- customAttrName (カスタム属性名)
- customAttrValue (カスタム属性値)

フィルターには、AND、OR、NOT、EXISTS、NOT EXISTS、およびグループ化 (括弧) 操作を含めることができます。詳細については、NNMiヘルプの「ノードグループの追加のフィルターを指定する」を参照してください。

機能は、本来はNNMiと統合される他のプログラムを目的としていました。たとえば、ルーター冗長性とコンポーネント稼働状態は、機能 (フィールド) をNNMiデータベースに追加します。これらの機能は、すでに検出されてデバイスからノード詳細を調べることにより、見ることができます。

iSPIによりカスタム属性を追加したり、独自のカスタム属性を作成できます。Web Services SDKを購入していない方は、各ノードのフィールドに手動で値を入れる必要があります。たとえば資産番号やシリアル番号は属性となりえますが、機能ではありません。

追加ノード

ノードグループに対してノードを限定するには、[**追加フィルター**] を使用することをお勧めします。フィルターを使用して制限することが困難である重要なデバイスがネットワークに含まれている場合、それらのデバイスをホスト名ごとに1つのグループに追加します。ホスト名ごとにノードをノードグループに追加するのは、他に手段がない場合のみにしてください。

注意: NNMi管理サーバーのリソースを過度に消費するため、[**追加のノード**] タブを使用してノードグループにノードを追加することはほとんどありません。

ノードグループのステータス

そのように設定すると、以下のいずれかのアルゴリズムを使用してNNMiによってノードグループのステータスが決定されます。

- ノードグループの任意のノードの最も深刻なステータスと一致するようにノードグループを設定します。このアプローチを使用するには、[**ステータスの設定**] フォームの [**ほとんどの重大なステータスを伝達**] チェックボックスを選択します。
- 各ターゲットステータスに設定されたしきい値を使用してノードグループのステータスを設定します。たとえば、警戒域のターゲットステータスのデフォルトしきい値は20%です。NNMiでは、ノードグループ内のノードの20% (または、それ以上) が警戒域ステータスになると、ノードグループのステータスが警戒域に設定されます。このアプローチを使用するには、[**ステータスの設定**] フォームの [**ほとんどの重大なステータスを伝達**] チェックボックスをオフにします。ターゲットしきい値のパーセントしきい値は、このフォームの [**ノードグループのステータス設定**] タブで変更できます。

大きなノードグループのステータス計算には大量のリソースが必要になるため、新規インストール時にはノードグループのステータス計算はNNMiのデフォルトでオフに設定されます。ステータスの計算は、各ノードグループの [**ノードグループ**] フォームの [**ステータスの計算**] チェックボックスで有効にすることができます。

インタフェースグループ

インタフェースグループは、ノード内のインタフェースを、IFType別に、またはifAlias、ifDesc、ifName、ifIndex、IPアドレスなど他の属性別にフィルタリングします。インタフェースグループは階層もコンテインメントも継承しませんが、インタフェースをホスト管理しているノードのノードグループに基づいてメンバーシップをさらに限定することができます。

インタフェースグループを、ノードグループと同様のカスタム機能および属性でフィルタリングできます。

インタフェースグループの制限は、タブ内およびタブ間でまとめてANDを適用します。

注: インタフェースグループのインタフェースは、以下の条件での検出中に必ずしも最初から除外されるわけではありません。

- インタフェースグループは、インタフェースグループ定義で1つ以上のインタフェース機能をフィルタリングして作成されます。
- インタフェースグループは、[**除外対象インタフェース**] 検出の設定オプションで指定されます。

インタフェース機能はインタフェースグループのインタフェースに適用された後に、再検出中に除外フィルターが再適用されると除外されます。

NNMiで提供されるインタフェース機能と [**除外対象インタフェース**] 検出の設定オプションの詳細については、『NNMi管理者用のオンラインヘルプ』を参照してください。

ノードインタフェースおよびアドレス階層

NNMiは監視設定を、以下の方式で割り当てます。

1. **インタフェース設定**—NNMiは、各ノードのインタフェースおよびIPアドレスが、最初に一致する**インタフェース設定**定義に基づいてモニタリングされます。最初に一致するのは、順序番号が最も小さい**インタフェース設定**定義です。
2. **ノード設定**—NNMiによって、各ノードと前回一致しなかった各インタフェースまたはIPアドレスが、最初に一致する**ノード設定**定義に基づいてモニタリングされます。最初に一致するのは、順序番号が最も小さい**ノードの設定**定義です。

   注: 子ノードグループは、順序階層に含まれます。親ノードグループの順序番号のほうが小さい場合 (たとえば、親=10、子=20)、親ノードグループに指定された監視設定は子ノードグループ内のノードにも適用されます。親ノードグループ 監視設定を上書きするには、子ノードグループの順序番号を親よりも小さな番号に設定します (たとえば、親=20、子=10)。

3. **デフォルト設定** — 手順1または手順2のノード、インタフェース、IPアドレスに一致が見つからない場合、NNMiではデフォルトのモニタリング設定が適用されます。

NNMi設定およびデータベースのリセット

検出を完全に再スタートしてNNMi設定のすべてのやり直したい場合、またはNNMiデータベースが破損した場合は、NNMi設定およびデータベースをリセットできます。このプロセスにより、NNMi設定、トポロジ、およびインシデントのすべてが削除されます。

この手順で説明しているコマンドの詳細については、該当する参照ページかLinuxのマニュアルページを参照してください。

以下の手順を実行します。

1. NNMiサービスを、次のコマンドを使用して停止します。

   ```
   ovstop -c
   ```

2. 省略可能。この手順によってデータベースが削除されるため、実行する前に次のコマンドで既存のデータベースをバックアップするとよいでしょう。

   nnmbackup.ovpl -type offline -target <backup_directory>

3. 省略可能。現在のNNMi設定を保持する場合は、`nnmconfigexport.ovpl`コマンドを使用してNNMi設定をXMLファイルに出力します。

   ヒント: `nnmconfigexport.ovpl`コマンドではSNMPv3資格情報は保持されません。詳細については、nnmconfigexport.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

4. **省略可能。**nnmtrimincidents.ovplコマンドを使用して、NNMiインシデントをアーカイブします。インシデントは、nnmtrimincidents.ovplリファレンスページ、またはLinuxのマニュアルページの説明に従って、CSV形式でアーカイブされます。
5. NNMiデータベースを削除して再作成します。
   - 組み込みデータベースの場合は、以下のコマンドを実行します。

     nnmresetembdb.ovpl -nostart

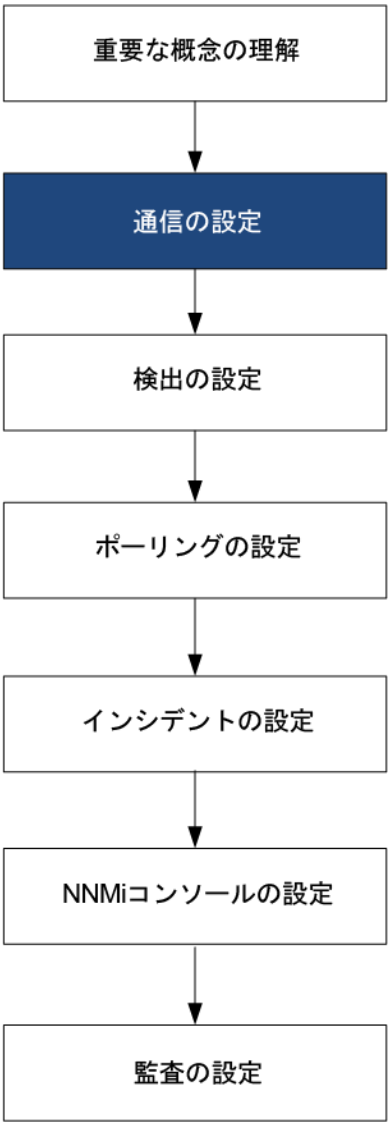
   - Oracleデータベースの場合は、Oracleデータベース管理者にNNMiデータベースの削除と再作成を依頼してください。データベースインスタンス名は、削除せずに保持してください。
6. iSPIまたはNNMiと統合されるスタンドアロン製品をインストールした場合は、これらの製品をリセットして古いトポロジ識別名を削除します。具体的な手順については、製品のマニュアルを参照してください。
7. NNMiサービスを、次のコマンドを使用して開始します。

   ovstart -c

   これでNNMiはデフォルト設定のみとなり、本製品を新しいシステムにインストールしたのと同じ状態です。
8. NNMiの設定を開始します。以下のいずれかを行います。
   - 「クイックスタート設定ウィザード」を使用します。
   - NNMiコンソールの [**設定**] ワークスペースに情報を入力します。
   - nnmconfigimport.ovplコマンドを使用して、手順3で保存したNNMi設定の一部またはすべてをインポートしてください。

ヒント: nnmconfigimport.ovplコマンドを使用して大量の設定をインポートする場合 (9,500個のノードグループや10,000個のインシデントの設定など)、-timeoutオプションを使用して、インポートトランザクションのタイムアウトをデフォルト値の60分 (3600秒) よりも長くなるように調整することを検討してください。詳細については、nnmconfigimport.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

NNMi通信



HP Network Node Manager i Software (NNMi) は、簡易ネットワーク管理プロトコル (SNMP) とインターネット制御メッセージプロトコル (ICMP ping) を使用してデバイスを検出し、デバイスのステータスと稼働状態をモニタリングします。

注: (SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコルを使用できます。たとえば、VMware環境用のSOAPプロトコルなどです。

各自の環境で実行可能な通信を確立するには、ネットワークのさまざまなデバイスとエリアについて、アクセス資格認定、適切なタイムアウト、再試行値すべてでNNMiを設定します。ネットワークのいくつかのエリアでプロトコルを無効にし、トラフィックを削減またはファイアウォールを順守できます。

設定する通信の値はNNMiの検出および状態ポーリングの基礎を形成します。NNMiは、検出またはポーリングのクエリーを作成するときに、各デバイスに該当する値を適用します。このように、ネットワークのいくつかの領域とのSNMP通信を無効にするようNNMiを設定すると、NNMi検出とNNMi状態ポーリングはどちらも、SNMP要求をその領域には送信できません。

注意: デバイスがSNMP v1またはSNMP v2Cを使用する場合は、以下の点に注意してください。

- SNMP v1とSNMP v2Cは、その情報パケットをクリアテキストで送信します。
- 環境をセキュリティで保護するには、SNMPトラップのフローやデバイスからの情報の収集に、SNMP v3を使用するかまたはファイアウォール制御などの保護を追加してください。

この章には、以下のトピックがあります。

通信の概念

NNMiは、SNMPとICMPを主に要求と応答の方式で使います。ICMP Ping要求への応答で、アドレスの応答性を確認します。他の管理プロトコル (特定のMIBオブジェクトに対するSNMP要求など) への応答で、ノードに関するより総合的な情報を取得します。

注: (SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコルを使用できます。たとえば、VMware環境用のSOAPプロトコルなどです。

以下の概念がNNMi通信設定に適用されます。

通信の設定レベル

NNMi通信設定には、以下のレベルがあります。

- 特定のノード
- 領域
- グローバルなデフォルト

各レベルで、アクセスの資格情報、タイムアウトと再試行の値、管理プロトコルの有効化 (ICMPやSNMPなど)、管理プロトコルのアクセス設定 (SNMPなど) を指定できます。あるレベルで設定をブランクにしておくと、NNMiは次のレベルのデフォルトを適用します。

指定ノードと通信するとき、NNMiは設定を以下のように適用します。

1. ノードが**特定のノード**の設定と一致する場合、NNMiはその設定に含まれている通信の値をすべて利用します。
2. どの設定もまだ定義されていない場合、NNMiはノードがいずれかの**領域**に属するか判断します。領域は重なる可能性があるため、NNMiでは順序番号が最小のものと一致する領域が使用されます。NNMiは、その領域に対して指定された値を、該当する特定のノードの空白の値 (ある場合) に使用します。追加領域の設定は考慮されません。
3. まだ定義されない設定がある場合、NNMiは**グローバルなデフォルト**設定を使用して、残りの空白の設定に取り込みます。

特定のデバイスとの管理プロトコル通信に使用される値は、必要な設定がすべて決まるまで、累積的に構築されます。

ネットワーク待ち時間とタイムアウト

通常のネットワーク遅延は、NNMi管理サーバーがICMPクエリーへの応答を得るための待ち時間に影響を与えます。一般に、ネットワークのエリアが異なれば、応答が返る時間も異なります。たとえば、NNMi管理サーバーが置かれているローカルネットワークからは、ほぼ即時の応答が返り、ダイヤルアップワイドエリアリンク経由でアクセスする遠隔地にあるデバイスからの応答は、通常、はるかに長く時間がかかります。さらに、負荷が大きいデバイスは処理量が多いためICMPクエリーにただちに応答できません。タイムアウトと再試行の設定を決定するときには、こうした遅延に関する事項を考慮してください。

ネットワーク領域と特定のデバイスの両方について、固有のタイムアウトと再試行の設定を行うことができます。設定により、応答がない場合に要求を破棄するまでの、NNMiの応答待ち時間、NNMiがデータを要求する回数が決まります。

要求を再試行するたびに、NNMiは設定したタイムアウト値をそれまでのタイムアウト値に加算します。そのため、再試行するごとに停止時間が長くなります。たとえば、NNMiの設定を5秒でタイムアウト、再試行は3回とすると、NNMiは最初の要求への応答を5秒待ち、2回目の要求への応答は10秒待ち、3回目の要求の応答は15秒待ってから次のポーリングサイクルに移ります。

SNMPアクセス制御

管理対象デバイス上のSNMPエージェントとの通信には、アクセス制御資格情報が必要です。

- SNMPv1とSNMPv2c

  各NNMi要求内のコミュニティ文字列は、応答するSNMPエージェントで設定されているコミュニティ文字列と一致する必要があります。通信はすべて、クリアテキスト (暗号化なし) でネットワークを通過します。

- SNMPv3

  SNMPエージェントとの通信は、ユーザーベースのセキュリティモデル (USM) に従います。各SNMPエージェントには、設定済みのユーザー名とそれに関連する認証要件のリストがあります (認証プロファイル)。すべての通信のフォーマットは、設定によって制御されます。NNMi SNMP要求は、有効なユーザーを指定し、そのユーザーに対して設定されている認証とプライバシの制御に従う必要があります。

  - 認証プロトコルは、メッセージダイジェストアルゴリズム5 (MD5) またはセキュアーハッシュアルゴリズム (SHA) のいずれか選択した方を使って、ハッシュベースのメッセージ認証コード (HMAC) を使用します。
  - プライバシプロトコルは、暗号化を使用しないか、またはデータ暗号化標準 - 暗号ブロック連鎖 (DES-CBC) 対称暗号化プロトコルを使用します。

注: DES-CBCは弱い暗号と考えられています。そのため、DES-CBCを使用する場合は、より強い暗号を選択することをお勧めします。暗号の選択を変更するには:

1. NNMiコンソールから、[**設定**] ワークスペースをクリックします。
2. [**インシデント**] フォルダーを展開します。
3. [**トラップサーバー**] フォルダーを展開します。
4. [**トラップ転送の設定**] をクリックします。
5. [**プライバシプロトコル**] リストで、より強い暗号を選択します。

注: NNMiが管理するノードでSNMPv3通信を設定する場合は、DES-CBCを使用しないでください。

NNMiは、(IPアドレスフィルターやホスト名フィルター経由で定義された) ネットワークの領域のマルチSNMPアクセス制御資格情報の仕様をサポートします。NNMiは、設定したすべての値を、所定のSNMPセキュリティレベルで並行して試し、その領域内のデバイスと通信しようとします。NNMiがその領域で使用する最小限のSNMPセキュリティレベルを指定できます。NNMiは、各ノードから返される最初の値 (デバイスのSNMPエージェントからの応答) を検出と監視の目的で使用します。

「高可用性 (HA) 環境でのSNMPアクセス制御」(45ページ)も参照してください。

高可用性 (HA) 環境でのSNMPアクセス制御

NNMiを高可用性 (HA) 環境で設定すると、SNMPのソースアドレスが物理クラスターノードのアドレスに設定されます。SNMPのソースアドレスをNNM_INTERFACEに設定する (仮想IPアドレスに設定される) には、`ov.conf`ファイルを編集して、`IGNORE_NNM_IF_FOR_SNMP`の値を`OFF`に設定する必要があります(デフォルトでは、`ON`に設定されています)。

HA環境でSNMPのソースアドレスをNNM_INTERFACEに設定するには、以下の手順を実行します。

1. クラスターの両方のノードで、以下のファイルを編集します。

   Windowsの場合:%NnmDataDir%\shared\nnm\conf\ov.conf

   Linuxの場合:$NnmDataDir/shared/nnm/conf/ov.conf
2. IGNORE_NNM_IF_FOR_SNMPの値をOFFに設定します。(デフォルトでは、ONに設定されています)。

   IGNORE_NNM_IF_FOR_SNMP=OFF
3. NNMi管理サーバーを停止して再起動します。

   注: ovstopおよびovstartコマンドを実行する前に、ノードをメンテナンスモードにします。

   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

SNMPバージョンの優先

SNMPプロトコルはバージョン1から バージョン2(c) へと長年をかけて発展したもので、現在はバージョン3です。この間、とりわけセキュリティ機能は強化されてきました。NNMiは、各自のネットワーク環境でどのバージョンでも処理できますし、全バージョンの混合したものも処理できます。

NNMiが特定のノードについて受信する最初のSNMP応答によって、そのノードとの通信にNNMiが使用する通信の資格情報とSNMPバージョンが決まります。

注: ノードのSNMPバージョンにより、NNMiでのノードからのトラップの受け入れが、以下のように異なります。

- NNMiがSNMPv3を使用して受信トラップのソースノードやソースオブジェクトを検出すると、NNMiは、受信するSNMPv1、SNMPv2c、およびSNMPv3のトラップを受け入れます。
- NNMiがSNMPv1またはSNMPv2cを使用して受信トラップのソースノードやソースオブジェクトを検出すると、NNMiは受信するSNMPv3トラップを廃棄します。このトラップを受信する必要がある場合は、「モニタリング対象外のノードのSNMPv3トラップを認証するようにNNMiを設定する」(Configuring NNMi to Authenticate SNMPv3 Traps for Nodes Not Being Monitored)の手順に従います。

SNMPバージョンと、ネットワークの各領域で受け入れられる最小レベルのセキュリティ設定を指定します。[SNMP最小セキュリティレベル] フィールドのオプションは、以下のとおりです。

- [**コミュニティのみ (SNMPv1)**] —NNMiは、コミュニティ文字列、タイムアウトおよび再試行用に設定した値でSNMPv1を使って更新を試みます。NNMiは、SNMPv2cやSNMPv3の設定は試みません。
- [**コミュニティのみ (SNMPv1またはv2c)**] — NNMiは、コミュニティ文字列、タイムアウトおよび再試行用に設定した値でSNMPv2cを使って更新を試みます。SNMPv2を使ったコミュニティ文字列への応答がない場合は、NNMiはコミュニティ文字列、タイムアウト、および再試行用に設定した値でSNMPv1を使って通信を試みます。NNMiは、SNMPv3の設定は試みません。

- [**コミュニティ**]—NNMiは、コミュニティ文字列、タイムアウトおよび再試行用に設定した値でSNMPv2cを使って更新を試みます。SNMPv2を使ったコミュニティ文字列への応答がない場合は、NNMiはコミュニティ文字列、タイムアウト、および再試行用に設定した値でSNMPv1を使って通信を試みます。機能するものがない場合、NNMiはSNMPv3を試みます。
- [**認証なし、プライバシなし**]—認証もプライバシもないユーザーについて、NNMiはタイムアウトと再試行用に設定した値でSNMPv3を使って通信を試みます。機能するものがない場合、必要に応じて、NNMiは認証はあるがプライバシがないユーザー、次に、認証とプライバシがあるユーザーを試みます。
- [**認証、プライバシなし**]—認証はあるがプライバシはないユーザーについて、NNMiはタイムアウトと再試行用に設定した値でSNMPv3を使って通信を試みます。機能するものがない場合、NNMiは認証とプライバシのあるユーザーを試みます。
- [**認証、プライバシ**]—認証もプライバシもあるユーザーについて、NNMiはタイムアウトと再試行用に設定した値でSNMPv3を使って通信を試みます。

管理アドレスの優先

ノードの**管理アドレス**とは、NNMiがノードのSNMPエージェントと通信する場合に使用するアドレスです。ノードの管理アドレスを指定するか (特定ノードの設定で)、または、ノードに関連するIPアドレスの中からNNMiがアドレスを選択するようにできます。検出設定で検出から特定のアドレスを除外することにより、この動作を微調整できます。NNMiが管理アドレスを決定する方法については、NNMiヘルプの「[ノード] フォーム」を参照してください。

注: ハイパーバイザーNNMiを検出するには、管理アドレスではなくノード名が必要です。

NNMiは、デバイスの検出と監視を継続的に行います。最初のNNMi検出サイクルの後、以前検出したSNMPエージェントが応答しない場合 (たとえば、デバイスのSNMPエージェントを再設定した場合など) は、[**SNMPアドレス再検出を有効にする**] フィールドの設定によりNNMiの動作が制御されます。

- [**SNMPアドレス再検出を有効にする**] チェックボックスがオンになっている場合、NNMiは機能するアドレスの検索で設定した値を再試行します。
- [**SNMPアドレス再検出を有効にする**] チェックボックスがオフになっている場合、NNMiはデバイスが「停止中」であると報告し、そのデバイスについて別の通信設定を試みません。

ヒント: [**SNMPアドレス再検出を有効にする**] チェックボックスは、通信設定のすべてのレベルで利用可能です。

ヒント: 自動検出ルール設定フィールドの [**SNMPデバイスの検出**] と [**非SNMPデバイス**] は、NNMiのSNMP使用方法に影響します。詳細については、NNMiヘルプにある「自動検出ルールの基本設定を設定する」を参照してください。

SNMPv3トラップと通知

デバイスと通信するためにNNMiでSNMPv3を使用する場合、検出プロセスを使用して、デバイスのエンジンID、ブートカウント、エンジン時間が識別されます。NNMiは、ユーザーおよびプロトコルに関する設定済みの詳細とこの情報を併用して、デバイスへのメッセージ送信を開始します。

デバイスからNNMiにトラップを送信する場合、トラップは単一パケットのトランザクションであり必要な情報を取得する手段がないため、デバイスにNNMi情報が存在しないことがあります。したがって、デバイス自体のエンジンID、ブートカウント、エンジン時間が、ユーザー名およびプロトコルの詳細とともにトラップで使用されます。デバイスの詳詳細については、NNMiでデバイス用に設定された内容と同じである必要があります。NNMiでは、デバイスごとに複数のSNMPv3ユーザーを設定できません。

通知は確認済みのパケットであるため、NNMiからデバイスに対して行うSNMP要求に似ています。ただし、この場合は、最初のパケットを開始するデバイス、および確認に応答するNNMiは対象外となります。このため、NNMiのエンジンID、ブートカウント、エンジン時間を取得するために、デバイスからNNMiに対して検出が実行されます。デバイスで使用されるユーザー名およびプロトコルの設定は、NNMiトラップ転送の設定 (つまり、NNMiのSNMPv3エージェント設定) の内容と一致する必要があります。

ポーリングプロトコル

ネットワークの一部でNNMiがSNMPまたはICMP用を使用できないようにすることができます (たとえば、インフラストラクチャー内のファイアウォールがICMPまたはSNMPトラフィックを禁止する場合など)。

特定のネットワークエリアでデバイスへのICMPトラフィックを無効にすると、NNMiで以下のような結果になります。

- 自動検出ルールpingスィープ機能 (任意で追加可能) は、ネットワークの領域内で追加ノードを見つけられません。すべてのノードが、シードされるか、または隣接ARPキャッシュ、Cisco Discovery Protocol (CDP)、またはExtreme Discovery Protocol (EDP) など、MIBオブジェクト要求への応答を通して使用できる必要があります。広域ネットワークデバイスは、すべてシードしないと失われる可能性があります。
- State Pollerは、SNMP要求に応答するように設定されていないデバイスは監視できません。(ただし、デバイスがSNMPに応答すると、StatePollerはICMPを使用しません。)
- オペレーターはトラブルシューティングの間は、**[アクション] > [Ping]** を使ってデバイス到達可能性をチェックできません。

特定のネットワークエリアでデバイスへのSNMPトラフィックを無効にすると、NNMiで以下のような結果になります。

- 検出では、存在しないデバイスの情報は収集できません。すべてのデバイスでNo SNMPデバイスのプロファイルルを受信します。
- 検出では、クエリーによって追加の隣接デバイスを見つけることができません。デバイスはすべて直接にシードされる必要があります。

- 検出では、データベースから接続情報を収集できないため、デバイスはNNMiマップには未接続として示されます。
- No SNMPデバイスのプロファイルルを持つデバイスについては、StatePollerはICMP (Ping) のみを使用するデバイスの監視のデフォルトが優先されます。
- State Pollerは、コンポーネントの稼働状態やパフォーマンスデータをデバイスから収集できません。
- Causal Engineは、デバイスに接触して近隣分析を実行し、インシデントの根本分析を見つけることはできません。

通信設定およびnnmsnmp*.ovplコマンド

`nnmsnmp*.ovpl`コマンドは、NNMiデータベースで指定されていないデバイス通信設定の値を検索します。この方法ではovjbossプロセスが動作している必要があります。ovjbossが動作していない場合、`nnmsnmp*.ovpl`コマンドは次のように動作します。

- SNMPv1エージェントとSNMPv2cエージェントの場合、コマンドは未指定通信設定にデフォルト値を使用します。
- SNMPv3エージェントの場合は、ユーザーとパスワードを指定すると、コマンドは未指定通信設定にデフォルト値を使用します。ユーザーとパスワードを指定しないと、コマンドはエラーになります。

通信の計画作成

以下の領域で決定します。

- 「デフォルトの通信設定」(49ページ)
- 「通信設定領域」(50ページ)
- 「特定のノードの設定」(51ページ)
- 「再試行とタイムアウトの値」(51ページ)
- 「アクティブなプロトコル」(51ページ)
- 「複数のコミュニティ文字列または認証プロファイル」(52ページ)

デフォルトの通信設定

NNMiは、該当する領域や特定のノードで指定しなかった設定をデフォルト値を使用して完成させるため、大半のネットワークで妥当なものになるようデフォルトを設定します。

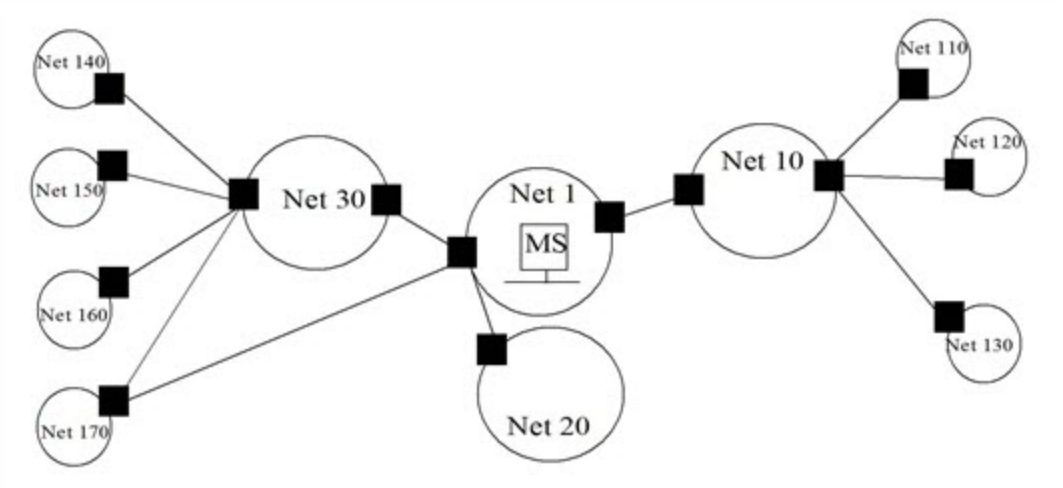
- NNMiが試す必要のある一般に使われるコミュニティ文字列がありますか?
- ネットワークではどのようなタイムアウトと再試行のデフォルト値が合理的でしょうか?

通信設定領域

領域とは、ネットワーク内で同じ通信設定を適用するのが妥当なエリアのことです。たとえば、NNMi管理サーバーの近くにあるローカルネットワークからは、通常はすぐに応答が戻ってきます。複数ホップ離れたネットワークエリアなら応答にもっと時間がかかるのが普通です。

ネットワークのサブネットやエリアを個別に設定する必要はありません。ラグタイムが近い複数のエリアを1つの領域にまとめることができます。以下のネットワークマップについて考えてみてください。

通信領域のネットワーク例



タイムアウトと再試行を考慮した場合、以下のように領域を設定することができます。

- 領域A - Net 1
- 領域B - Net 10、Net 20、およびNet 30を含める
- 領域C - さらに遠くにある外部のネットワーク

NNMi管理サーバーから1ホップまたは2ホップのどちらのパスを優先するようトラフィック管理構成が設定されているかどうかに従って、Net 170をグループにまとめる最良の方法を決定します。

また、類似したアクセス資格認定を使用するデバイスをグループにまとめる場合にも領域を使用します。ネットワークのすべてのルーターで同じコミュニティ文字列 (または可能なコミュニティ文字列の一部) が使用されていて、命名規約 (`rtrnnn.yourdomain.com`など) でルーターを識別できる場合は、全ルーターを1つの領域に設定すれば、すべてのルーターが同じように処理されます。ワイルドカードを使ってデバイスをグループにまとめられない場合は、各デバイスを特定のノードとして設定できます。

同じタイムアウト/再試行の値とアクセス資格証明設定を1つの領域のすべてのノードに適用できるように、領域設定を計画してください。

領域定義は重複することがあり、1つのデバイスが複数の領域の定義にあてはまることもあります。NNMiは、順序番号が最も小さい (かつ、他に一致する領域がない) 領域から設定を適用します。

特定のノードの設定

固有の通信設定要件を持つデバイスの場合、特定ノードの設定を使用して、そのノードの通信設定を指定します。特定ノードの設定の使用例として、以下の例があります。

- SNMPv2c/SNMPv3 GetBulk要求に適切に応答しないノード
- 他の類似ノードと名前のパターンが一致しないノード

注: 特定のデバイスのSNMP通信を有効または無効にできます。NNMiヘルプの「特定ノードの設定フォーム」を参照してください。

再試行とタイムアウトの値

タイムアウトの時間を長く、再試行の回数を多く設定すると、ビジー状態にあるか、または離れたところにあるデバイスからより多くの応答を集められます。このように応答率が高まると、偽のダウンメッセージを除外できます。しかし、実際にダウンしているデバイスに注意が必要なことを知るのに時間がかかるようにもなります。ネットワークの各領域のバランスを見出すことは重要であり、このために各自の環境で値のテストと調整の期間が必要になる可能性があります。

各ホップの現在のラグタイムに関するヒントを得るには、以下の手順を実行します。

- Windowsの場合:それぞれのネットワークエリア内のデバイスに対して`tracert`を実行する。
- Linuxの場合:それぞれのネットワークエリア内のデバイスに対して`traceroute`を実行する。

アクティブなプロトコル

通信の設定とモニタリングの設定を使用して、ネットワーク内でデバイスと通信を行うときにNNMiが生成するトラフィックの種類を制御することができます。インフラストラクチャーのファイアウォールでICMPまたはSNMPのトラフィックが許可されていない場合は通信の設定を使用します。デバイスに関するデータの特定のサブセットが必要ない場合は、モニタリングの設定を使用してプロトコルの使用を微調整します。通信またはモニタリングの設定のどちらかによってデバイスのプロトコルが無効にされると、NNMiはその種類のトラフィックをデバイスに送信しません。

注: SNMP通信を無効にすると、ネットワークのNNMiのステータスと稼働状態のモニタリングがかなり危険な状態になります。

各領域または特定のデバイスはICMPトラフィックを受信するはずであるかに注意してください。

アクセス資格認定を与えないデバイスとのSNMP通信を明示的に無効にする必要はありません。デフォルトで、NNMiはこれらのデバイスを`No SNMP`デバイスのプロファイルルに割り当て、ICMPのみを使ってデバイスを監視します。

(SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコル (たとえば、VMware環境用のSOAPプロトコル) を使用できます。

「ネットワーク設定プロトコル (NETCONF) を使用したデバイス対応」(55ページ)も参照してください。

複数のコミュニティ文字列または認証プロファイル

ネットワークの各エリアで試みるコミュニティ文字列と認証プロファイルの計画を作成します。デフォルト設定と領域設定については、並行して試みる複数のコミュニティ文字列と認証プロファイルを設定できます。

注: 有望なコミュニティ文字列を試す間に、NNMiクエリーにより、デバイスで資格認定不合格が生成されることがあります。NNMiが初期検出を完了する間に、資格認定不合格は安全に無視できる可能性があることを業務部に知らせてください。代わりに、領域 (と試行する関連コミュニティ文字列と認証プロトコル) が可能な限り厳しく設定して、資格認定不合格の数を最小にすることもできます。

環境でSNMPv1またはv2とSNMPv3が使用されている場合は、各領域で受け入れられる最低のセキュリティレベルを決定してください。

SNMPv1とSNMPv2のコミュニティ文字列

SNMPv1またはv2cアクセスが可能な領域では、領域内で使用されるコミュニティ文字列と特定のデバイスで必要とされるコミュニティ文字列を集めます。

SNMPv3の認証プロファイル

SNMPv3アクセスが可能なデバイスを含む領域では、受け入れられる最小限のデフォルト認証プロファイル、各領域に適した認証プロファイル、および特定のデバイスで使用される固有の認証資格証明 (ある場合) を決定します。ネットワーク内で使用中の認証プロトコルとプライバシプロトコルも判断します。

SNMPv3通信の場合、NNMiでは以下の認証プロトコルがサポートされます。

- HMAC-MD5-96
- HMAC-SHA-1

SNMPv3通信の場合、NNMiでは以下のプライバシプロトコルがサポートされます。

- DES-CBC
- TripleDES
- AES-128
- AES-192
- AES-256

固有ノードまたは領域設定ごとに、1つの認証プロトコルおよび1つのプライバシプロトコルを指定できますが、指定しないこともできます。

注: TripleDES、AES-192、AES-256のプライバシプロトコルを使用するには、Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Filesライブラリが必要です。このライブラリはNNMiインストールプロセスの一部として自動的にインストールされます。ライブラリを誤っ

て削除してしまった場合は、「設定問題に関するトラブルシューティング」(548ページ)の手順に従って復元できます。

通信の設定

このセクションを読んだ後、特定の手順については、NNMiヘルプの「通信プロトコルを設定する」を参照してください。

注: 大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。

通信の以下のエリアの設定

- デフォルト設定
- 領域定義とその設定
- 特定のノードの設定

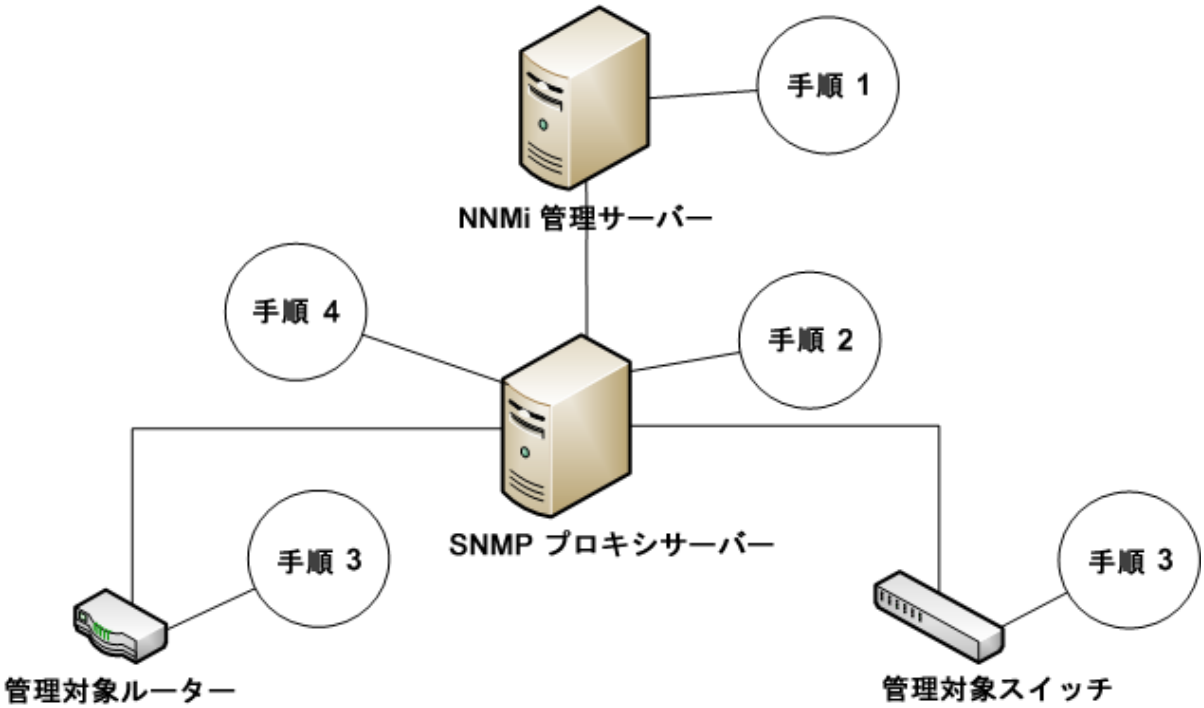
特定のノードについて、NNMiコンソールまたは構成ファイルによって、ノードの設定を入力できます。

注: 定義した領域の順序番号をダブルチェックします。ノードが複数の領域を認証する場合、NNMiはそのノードの順序番号の最も小さい領域の設定を適用します。

SNMPプロキシの設定

一部のネットワークでは、ネットワークデバイスとの通信にSNMPプロキシエージェントを使用します。次の図に、NNMiコンソールから [**設定**] > [**通信の設定**] を使用して [SNMPプロキシアドレス] と [SNMPプロキシポート] を設定した場合に、NNMiが使用するSNMP通信手順を示します。NNMiは、SecurityPackAgentAddressOid OID (.1.3.6.1.4.1.99.12.45.1.1) の使用をサポートするSNMPプロキシサーバーに対応しています。

プロキシサーバーの使用



1. NNMi管理サーバーがSNMPプロキシアドレスとSNMPプロキシポートにSNMP要求を送信し、管理対象ルーターと管理対象スイッチから情報を取得します。NNMi管理サーバーが特殊なプロキシvarbindであるSecurityPackAgentAddressOid (.1.3.6.1.4.1.99.12.45.1.1) で管理対象ルーターとスイッチのリモートアドレスおよびポートをエンコードし、このvarbindをSNMP要求に追加します。
2. SNMPプロキシサーバーがこの特殊なプロキシvarbindを読み取り、SNMP要求の送信先を判別して、NNMi管理サーバーによって要求された情報を取得するために管理対象ルーターとスイッチにSNMP要求を送信します。
3. 管理対象スイッチとルーターがSNMPプロキシサーバーに応答し (SNMPプロキシアドレスとSNMPプロキシポートを使用)、要求された情報を返します。
4. SNMPプロキシサーバーがNNMi管理サーバーに応答します (設定されたSNMPポートを使用)。

プロキシサーバーを使用するように設定されている場合、NNMiは以下のOIDを使用してSNMP応答を処理します。

- SecurityPackAgentAddressOid .1.3.6.1.4.1.99.12.45.1.1 (SNMP Research NetDiscover SECURITY-PACK-MIB)
- SecurityPackNotificationAddressOid .1.3.6.1.4.1.99.12.45.2.1 (SNMP Research NetDiscover SECURITY-PACK-MIB)
- ProxyOid .1.3.6.1.4.1.11.2.17.5.1.0 (HP)
- TrapForwardingAddressTypeOid .1.3.6.1.4.1.11.2.17.2.19.1.1.2.0 (HP)
- TrapForwardingAddressOid .1.3.6.1.4.1.11.2.17.2.19.1.1.3.0 (HP)

- Rfc3584TrapAddressOid .1.3.6.1.6.3.18.1.3.0 (RFC 3584)
- Rfc3584TrapCommunityOid .1.3.6.1.6.3.18.1.4.0 (RFC 3584)

SNMPプロキシサーバーでNNMiを使用する場合、プロキシベンダーに連絡してこのリスト内のOIDをサポートしているかどうかを確認してください。

ネットワーク設定プロトコル(NETCONF)を使用したデバイス対応

NNMiは、対応デバイスから管理情報を収集する方法として、簡易ネットワーク管理プロトコル(SNMP)に主に依存します。ただし、NNMiは、SNMPを使用しても必要な管理情報を得られない特定のベンダーのデバイスにネットワーク設定プロトコル(NETCONF)を使用することもあります。

現在、NNMiはジュニパーネットワークスのQFabricシステムのサポートのみにNETCONFを使用します。更新については、『HP Network Node Manager i Softwareデバイス対応マトリックス』を参照してください。

以下のセクションでは、NETCONFの簡単な紹介と管理対象デバイスおよびNNMiの両方に必要な設定についての情報について説明します。

「ネットワーク設定プロトコル(NETCONF)」(55ページ)

「ネットワーク設定プロトコル(NETCONF)の操作」(56ページ)

「管理対象デバイスのネットワーク設定プロトコル(NETCONF)の有効化および設定」(56ページ)

「NNMiのネットワーク設定プロトコル(NETCONF)デバイス資格情報の設定」(57ページ)

ネットワーク設定プロトコル(NETCONF)

SNMPなどのネットワーク設定プロトコル(NETCONF)は、ネットワーク管理のためのInternet Engineering Task Force (IETF)の規格です。NETCONFは、IETF Request for Comments (RFC) 4741および4742 (バージョン1)で定義され、その後、RFC 6241および6242 (バージョン1.1)で更新されます。

SNMPが通常、モニタリング、ポーリング、障害の通知に使用されるのに対して、NETCONFはデバイス設定メカニズムとしての使用を主な目的としています。両方のプロトコルが、NNMiに役立つ管理情報をレポートします。

NNMiは、NETCONFを使用して検出または再検出中にデバイスについての情報を収集します(つまり、読み取り専用の情報)。NNMiは、デバイス設定の変更やステータスまたはパフォーマンスメトリックスのモニタリングではNETCONFを使用しません。

NETCONFはXML形式のコマンドレスポンス型プロトコルで、主にSecure Shell (SSH)トランスポートを介して実行されます。NETCONFプロトコルは、XML形式のコマンドである点と、デバイスを使用したユーザーとのインタラクション向けではなく管理アプリケーション向けに結果が設計されている点を除いて、いくつかの点で従来のデバイスコンソールのコマンドラインインタフェース(CLI)に似ています。

NETCONFは比較的新しい管理プロトコルです。そのため、使用できるデバイスベンダーは、SNMPと比較すると限定的です。

ベンダーがNNMiで管理されているデバイスにNETCONFを実装する場合、以下の点に注意してください。

- NETCONFのコマンドは概してベンダー固有であることが多く、SNMPの多くの標準のベンダー固有のMIBほどは知られていません。その結果、NNMiがNETCONFを活用できる範囲はきわめて限定的です。
- 特定のベンダーがデバイスにNETCONFを実装し、NNMiで必要な管理情報をレポートする場合、NNMiでそのデバイス固有のNETCONFに対応する必要があります。詳細については、「管理対象デバイスのネットワーク設定プロトコル (NETCONF) の有効化および設定」(56ページ)および「NNMiのネットワーク設定プロトコル (NETCONF) デバイス資格情報の設定」(57ページ)を参照してください。

ネットワーク設定プロトコル (NETCONF) の操作

NNMiと管理対象デバイス間のNETCONF通信の詳細をNNMiユーザーが意識することはありません。ただし、トラブルシューティングに以下の概要が役立つことがあります。

- NETCONFクライアント (NNMiなどの管理アプリケーション) は、管理対象デバイスでNETCONFサーバー (サブシステム) とのSSH接続を確立します。有効なSSHのユーザー名およびパスワードの資格情報は、クライアントで識別され、デバイスで認証される必要があります。
- クライアントアプリケーションおよびデバイスの <hello> メッセージ形式による交換機能。
- クライアントはリモートプロシージャコール (RPC) メッセージ形式でデバイスへの要求を開始します。これには、標準の <get> や <get-config> の操作、さらにデバイスに定義されているベンダー固有の操作が含まれます。
- デバイスは、RPC応答メッセージ形式の操作の結果に応答します。
- クライアントアプリケーションは、要求の送信および応答の処理が完了すると、<close-session> RPCメッセージをデバイスに送信します。
- デバイスは <ok> RPC応答メッセージで受諾します。
- 最後に、両側でSSH接続が終了されます。

管理対象デバイスのネットワーク設定プロトコル (NETCONF) の有効化および設定

NNMiを管理対象デバイスと通信可能にするには、事前に管理対象デバイスでNETCONFを明示的に有効化および設定することが必要な場合があります。特定の指示については、ベンダーが提供するデバイス設定ドキュメントを参照してください。たとえば、ジュニパーネットワークスのQFabricシステムについては、『Juniper Networks' NETCONF XML Management Protocol Guide』の「Establishing a NETCONF Session」を参照してください。

一般的に、管理対象デバイスは以下の前提条件を満たす必要があります。

- デフォルトのNETCONF TCPポート830、または標準SSH TCPポート22でNETCONFを有効化します。
- NETCONF通信へのアクセス用に、デバイスにSSHユーザー名とパスワードの資格情報を設定します。NNMiは、読み取り専用アクセス権のみを必要とします。

NNMiのNETCONFを使用する対応デバイスの現在のリストと追加のベンダー固有の前提条件およびリファレンスについては、『HP Network Node Manager i Software (NNMi) デバイス対応マトリックス』を参照してください (「既知の制限」セクション)。

NNMiのネットワーク設定プロトコル (NETCONF) デバイス資格情報の設定

NNMiがNETCONFを使用してデバイスと通信できるようにするには、管理対象デバイスの設定に一致するようにNNMiにNETCONF SSH資格情報を設定する必要があります。

注: 適切なNETCONF資格情報がデバイスに設定されていない場合、NNMiの検出が続行されます (SNMPのみを使用)。ただし、そのデバイスのNNMiにレポートされる管理情報が不完全になる可能性があります。

NNMiコンソールを使用して、デバイスに対して、該当する [特定ノードの設定]、[領域設定]、または [デフォルト設定] の [**通信の設定**]、[**デバイス資格証明**] タブで、NETCONFデバイス資格情報の設定を行います。

注: 各管理対象デバイスには、SSHユーザーおよびパスワードを1つのみ設定できます。これは、そのデバイスに対する正規のSSHセッションおよびNETCONFセッションに対して、同じ資格情報の組み合わせが使用されることを意味します。

いったん設定されると、NNMiは、指定されたデバイス (ノード) に対して次の検出サイクルの間に新しい資格情報を使用します。

NNMiの [**通信の設定**] フォームの編集方法の詳細な手順については、NNMiの『管理者用のヘルプ』を参照してください。

仮想環境における通信の設定

このセクションでは、サポートされている仮想環境とNNMiが通信できるようにする設定情報について説明します。

ハイパーバイザー上にホストされた仮想マシンを監視するための前提条件

NNMiでは以下の操作がサポートされます。

- サポート対象ハイパーバイザーの検出と監視。

  ハイパーバイザーのノードフォームでは、各仮想マシンは [**ホスト対象ノード**] タブに表示されます。
- 各仮想マシン (ルーター、スイッチ、ノードなど) の検出と監視。

  仮想マシンのノードフォームでは、[**ホスト元ノード**] 属性にハイパーバイザーの名前が表示されます。

次の表に、ハイパーバイザーでホストされているハイパーバイザーと仮想マシンを検出するための前提条件を示します。

ハイパーバイザーとそのVMを監視するための前提条件

検出対象	前提条件	詳細情報
ハイパーバイザー	ハイパーバイザーはSNMP通信をサポートする必要があり、SNMPを使用してNNMiからアクセスできる必要があります。	該当しない
	NNMiは関連するSNMPエージェントと通信するように設定する必要があります (IPアドレスとコミュニティ文字列またはSNMPv3認証)。	NNMiユーザーインタフェースを使用して設定するには、「管理者用のヘルプ」の「通信プロトコルを設定する」に記載されているデフォルト、領域、または特定ノードについてのSNMPの設定方法を参照してください。 CLIを使用して設定するには、nnmcommunication.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。
	NNMiは、HTTPSを使用してハイパーバイザーと通信するように設定する必要があります。 **注:** VMwareのみ。VMwareのデフォルト証明書 (localhost.localdomain) を、ESXiサーバーのホスト名を使用して生成された証明書と置き換える必要があります。詳細については、VMWareのドキュメントを参照してください。ESX5.1およびESX5.5サーバーで実行する手順の例については、「VMwareデフォルト証明書の置換」(59ページ)を参照してください。	CLIを使用して設定するには、「ハイパーバイザーとの通信にHTTPSを使用するようにNNMiを設定する」(60ページ)を参照してください。 NNMiユーザーインタフェースを使用して設定するには、「管理者用のヘルプ」の「通信プロトコルを設定する」に記載されているデフォルト、領域、または特定ノードについての信頼された証明書の設定方法を参照してください。
ハイパーバイザー上の仮想マシン	ハイパーバイザーのWebサービスで認証を行うには、ハイパーバイザーについて記載されたSNMP要件の他にハイパーバイザーデバイスの資格	NNMiユーザーインタフェースを使用して設定するには、「管理者用のヘルプ」の「通信プロトコルを設定する」に記載されて

ハイパーバイザーとそのVMを監視するための前提条件 (続き)

検出対象	前提条件	詳細情報
	証明もNNMiに設定する必要があります。	いるデフォルト、領域、または特定ノードについての資格証明の設定方法を参照してください。 CLIを使用して設定するには、nnmcommunication.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

VMwareデフォルト証明書の置換

注: 自己署名またはCA署名証明書は、完全修飾ドメイン名をESXiサーバーのホスト名として使用して生成する必要があります。

デフォルトでは、VMware証明書はlocalhost.localdomainをESXiサーバーのホスト名として使用します。

VMwareのデフォルト証明書を、ESXiサーバーのホスト名を使用して生成された証明書と置き換えるには、ESXiサーバーでこれらの手順例を実行します。

注: この例では、ESX5.1およびEXS5.5サーバーで実行する手順について説明します。最新情報については、VMwareのデフォルト証明書の置き換え方法を説明しているVMwareドキュメントを参照してください。

1. /etc/hostsファイルに、ホストを解決するための以下のフォーマットがあることを確認します。

   ```
   #/etc/hosts
   127.0.0.1 localhost.localdomain localhost
   ::1 localhost.localdomain localhost
   16.78.xx.xxx hostname.usa.hp.com hostname
   ```
2. ESXiサーバーでSSHが有効になっていることを確認します。
3. 管理者権限のあるユーザーとしてESXiシェルにログインします。
4. 以下のディレクトリに移動します。

   ```
   /etc/vmware/ssl
   ```
5. 以下のコマンドを使用し、名前を変更して既存の証明書をすべてバックアップします。

   ```
   mv rui.crt orig.rui.crt
   mv rui.key orig.rui.key
   ```

6. 新しい証明書を生成するには、次のコマンドを実行します

   /sbin/generate-certificates
7. ホストを再起動します。
8. 次の手順によってホストで新しい証明書が正常に生成されたことを確認します。
   a. 次のコマンドを使用して証明書を表示します。

      ls -la
   b. 元のファイルが使用できる場合は、新しい証明書ファイルのタイムスタンプを orig.rui.crtおよびorig.rui.keyと比較します。

ハイパーバイザーとの通信にHTTPSを使用するようにNNMiを設定する

注: ハイパーバイザーとの通信にHTTPを使用する必要がある場合は、「ハイパーバイザーとの通信でHTTPを有効にする」(62ページ)も参照してください。

ハイパーバイザー上でホストされている仮想マシン (VMWare ESXiなど) をHTTPSプロトコルを使用してNNMiが監視できるようにするには、以下のいずれかの方法でハイパーバイザーの信頼された証明書をNNMiにアップロードする必要があります。

- NNMiユーザーインタフェースを使用して信頼された証明書をアップロードする。
- コマンドラインインタフェース (CLI) を使用して信頼された証明書をアップロードする。

注: 信頼された証明書は、HTTPSプロトコルを使用してハイパーバイザーとの信頼性のある接続を確立するためにNNMiが使用するSSL証明書の1つです。デフォルトレベルと領域レベルでは、これは同じCAによって発行された証明書を使用するハイパーバイザーを信頼するためにNNMiが使用するCA証明書を指します。ノードレベルでは、これはFQDNをサブジェクト名として使用して生成された、ハイパーバイザーのSSL証明書 (自己署名またはCA署名) のことです。

このセクションでは、CLIを使用して証明書をアップロードする方法を説明します。NNMiユーザーインタフェースを使用してアップロードする方法については、「管理者用のヘルプ」の「通信プロトコルを設定する」を参照してください。

信頼された証明書をNNMiにアップロードするには、以下の手順を実行します。

1. ハイパーバイザーの信頼された証明書を取得し、NNMi管理サーバー上の一時的な場所にこれをコピーします。

   注: VMwareのみ。VMwareのデフォルト証明書 (localhost.localdomain) を、ESXiサーバーのホスト名を使用して生成された証明書と置き換える必要があります。詳細については、VMWareのドキュメントを参照してください。ESX5.1およびESX5.5サーバーで実行する手順の例については、「VMwareデフォルト証明書の置換 」(59ページ)を参照してください。
2. 証明書がサポートされている形式であることを確認します。サポートされている信頼された証明書ファイルの拡張子は、.pem、.crt、.cer、および.derです。

3. 該当するコマンドを実行し、必要なレベルで証明書をアップロードします。次の表から、要件に合うコマンドを選択してください。

レベル	目的	コマンド
デフォルト (グローバル)	同じCAによって署名された証明書をハイパーバイザー全体で使用する組織が、信頼された証明書をデフォルトレベルでアップロードするために使用します。	`nnmcommunication.ovpl addCertificate -default -cert <fully qualified path to the certificate file>`
領域	同じCAによって署名された証明書を特定の領域のハイパーバイザーで使用する組織が、その領域の信頼された証明書をアップロードするために使用します。	`nnmcommunication.ovpl addCertificate -region <region name or UUID> -cert <fully qualified path to the certificate file>`
ノード	特定のハイパーバイザーで使用するSSL証明書 (CA署名または自己署名サーバー証明書) をアップロードするために使用します。 **注:** 自己署名またはCA署名証明書は、完全修飾ドメイン名 (FQDN) をサブジェクト名として使用して生成する必要があります。	`nnmcommunication.ovpl addCertificate -nodeSetting <node name or UUID> -cert <fully qualified path to the certificate file>`

コマンド例:

- **デフォルト**: `nnmcommunication.ovpl addCertificate -default -cert /tmp/new.pem`
- **領域**: `nnmcommunication.ovpl addCertificate -region region1 -cert /tmp/region1.der`
- **ノード**: `nnmcommunication.ovpl addCertificate -nodeSetting node1 -cert /tmp/node1.crt`

4. コマンドが正常に実行されると、コマンド出力に、アップロードされた証明書についての情報が表示されます。証明書の情報を確認します。

ヒント:

- アップロードした証明書は、`listCertificates`コマンドを使用して表示でき、`removeCertificate`コマンドを使用して削除できます。詳細については、`nnmcommunication.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

- ハイパーバイザーが検出された後、Webエージェント上でupdateWebagentSettingsコマンドを使用して証明書を直接アップロード、置き換え、または削除できます。詳細については、nnmcommunication.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

ハイパーバイザーとの通信でHTTPを有効にする

デフォルトでは、NNMiはHTTPSプロトコルを使用してハイパーバイザーと通信します。

HTTPを使用する必要がある場合は、以下の手順でserver.propertiesファイルに必要なプロパティを追加します。

1. server.propertiesファイルに移動します。

   Windowsの場合:

   %NnmDataDir%\nmsas\NNM\server.properties

   Linuxの場合:

   $NnmDataDir/nmsas/NNM/server.properties
2. 以下の行を追加します。

   #VMware vSphere APIなどのSOAPエージェントとの通信でhttpを使用するかどうかを決定します。

   #このプロパティはデモ環境またはテスト環境のみで有効にすること、およびHPPTSは本番環境の場合に

   #設定することをお勧めします。

   nms.comm.soap.targetconfig.HTTP_ENABLED=true
3. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで**ovstop**コマンドを実行します。

   NNMi管理サーバーで**ovstart**コマンドを実行します。

ハイパーバイザーとの通信でHTTPを無効にするには、次の手順を実行します。

1. server.propertiesファイルに移動します。

   Windowsの場合:

   %NnmDataDir%\nmsas\NNM\server.properties

   Linuxの場合:

   $NnmDataDir/nmsas/NNM/server.properties
2. HTTP_ENABLEDプロパティ値をfalseに変更します。

   nms.comm.soap.targetconfig.HTTP_ENABLED=false
3. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで**ovstop**コマンドを実行します。

   NNMi管理サーバーで**ovstart**コマンドを実行します。

注: 「ハイパーバイザーとの通信にHTTPSを使用するようにNNMiを設定する」(60ページ)の手順を実行します。

通信の評価

このセクションでは、通信設定の進行と成功を評価する方法をリストします。これらの作業のほとんどを完了できるのは、検出が完了した後です。

以下について考えます。

- 「すべてのノードがSNMP用に設定されましたか?」(63ページ)
- 「デバイスについてSNMPアクセスは現在利用できますか?」(63ページ)
- 「SNMPデバイスの管理IPアドレスは正しいですか?」(63ページ)
- 「NNMiは正しい通信設定を使っていますか?」(64ページ)
- 「State Poller設定は通信設定と一致していますか?」(64ページ)

すべてのノードがSNMP用に設定されましたか?

1. **[ノード]** インベントリビューを開きます。
2. **[デバイスのプロファイル]** 列を、文字列「SNMPなし」が含まれるようにフィルタリングします。
   - 管理するデバイスごとに、特定ノードの通信設定を行います。その代わりに、領域を拡張して、ノードを組み入れ、アクセス資格認定を更新することもできます。
   - 通信設定が正しい場合は、デバイスのSNMPエージェントが実行中であり、適切に設定されていることを確認します (ACLを含みます)。

デバイスについてSNMPアクセスは現在利用できますか?

1. インベントリビューでノードを選択します。
2. **[アクション] > [ステータスのポーリング]** または **[アクション] > [設定のポーリング]** を選択します。

   結果にSNMPの値が表示された場合、通信は動作中です。

コマンドラインから`nnmsnmpwalk.ovpl`コマンドで通信をテストすることもできます。詳細については、nnmsnmpwalk.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

SNMPデバイスの管理IPアドレスは正しいですか?

デバイスに対してNNMiが選択した管理アドレスを判定するには、以下の手順を実行します。

1. インベントリビューでノードを選択します。
2. **[アクション] > [通信の設定]** を選択します。

3. [**通信の設定**] フォームで、[アクティブなSNMPエージェント設定] リストにあるSNMPエージェントの管理アドレスが正しいことを確認します。

NNMiは正しい通信設定を使っていますか?

SNMPコミュニティ文字列が欠落しているか、または正しくない場合は、検出が不完全になる可能性があり、検出パフォーマンスに悪影響を及ぼす可能性もあります。

デバイスの通信設定を確認するには、`nnmcommunication.ovpl`コマンドを使用するか、または以下の手順を実行します。

1. インベントリビューでノードを選択します。
2. **[アクション] > [通信の設定]** を選択します。
3. [**通信の設定**] フォームで、SNMP設定テーブルにリストされた値が、NNMiでこのノードに使用する設定であることを確認します。

   通信設定が正しくない場合、問題解決の手始めとして、SNMP設定テーブル内のソース情報を使用します。領域や特定ノードの設定や順序番号を変更する必要がでてくる場合もあります。

   > **注:** VMware通信の場合、[Webエージェント] フォームでアクティブ設定を確認するか、またはnnmcommunication.ovpl listWebAgentSettingsコマンドを使用します。

   詳細については、管理者用のNNMiヘルプを参照してください。

State Poller設定は通信設定と一致していますか?

通信設定によってネットワークの領域へのプロトコルトラフィックが許可される場合でも、その種類のトラフィックは監視設定で無効にされることがあります。設定が上書きされるかどうかを知る手順は次のとおりです。

1. インベントリビューでノードを選択します。
2. **[アクション] > [モニタリングの設定]** を選択します。

監視設定または通信設定のどちらかによってデバイスへのある種類のトラフィックが無効にされる場合、そのトラフィックはNNMiから送信されません。

通信の調整

認証不合格の削減

検出の間にNNMiがあまりにも多くの認証トラップを生成している場合は、NNMiが試行するアクセス資格認定の、より小さいグループで小さい領域または特定のノードを設定します。

タイムアウトと再試行の調整

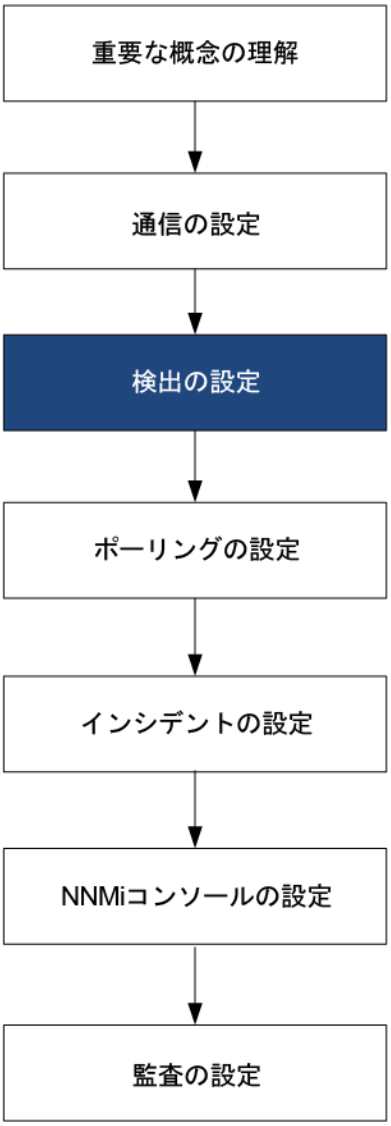
NNMiが検出中にSNMPを使ってデバイスに接触を試みるとき、通信設定はNNMiが必要なデバイス情報を収集できるかどうかを調べます。通信設定に正しいSNMPコミュニティ文字列が含まれていない場

合、またはNNMiが非SNMPデバイスを検出している場合、NNMiはSNMPタイムアウトと再試行用に設定済みの構成を使います。この場合、タイムアウトの値が大きいか、または再試行の回数が多いと、検出の全般的パフォーマンスに悪影響が及ぶ可能性があります。SNMP/ICMP要求に低速で応答することが分かっているデバイスがネットワークにある場合は、[**通信の設定**] フォームの [**領域**] タブまたは [**特定ノードの設定**] タブを使用してこれらのデバイスについてのみタイムアウト値と再試行値を微調整することを検討してください。

デフォルトコミュニティ文字列の削減

デフォルトコミュニティ文字列が多数あると、検出パフォーマンスに悪影響が及ぶことがあります。多数のデフォルトコミュニティ文字列を入力する代わりに、[**通信の設定**] フォームの [**領域**] タブまたは [**特定ノードの設定**] タブを使って、ネットワークの特定エリアのコミュニティ文字列設定を微調整します。

NNMi検出



ネットワーク管理で最も重要な作業の1つは、常に最新のネットワークトポロジを把握しておくことです。HP Network Node Manager i Software (NNMi) 検出により、トポロジインベントリにネットワーク内のノードに関する情報が挿入されます。NNMiでは、継続的なスパイラル検出によってこのトポロジ情報が維持され、根本原因解析ツールとトラブルシューティングツールで、インシデントに関する正確な情報を把握できるようになります。

この章では、NNMi検出を設定するために役立つ情報を記載しています。検出がどのようにして行われるのかと検出の設定方法については、NNMiヘルプの「ネットワークの検出」を参照してください。

この章には、以下のトピックがあります。

検出の概念

ルーターとスイッチのみを検出するNNMiのデフォルト動作により、ネットワーク管理を最も重要なデバイスに集中させることができます。つまり、最初にネットワークの基幹をターゲットにします。一般に、末端ノード (たとえばパソコンやプリンター) を管理対象にするのは、それらを重大リソースと見なすのでない限り避けるべきでしょう。たとえば、データベースやアプリケーションサーバーがクリティカルなリソースとして考えられます。

NNMiで検出するデバイスを管理してNNMiトポロジに加えるには、いくつかの方法があります。ネットワークをどのように構成するかやNNMiで何を管理するかによって、検出構成を非常に単純にしたり、極めて複雑にしたり、その間の適当なレベルにできます。

注: NNMiはデフォルトの検出を何も実行しません。各種のデバイスがNNMiトポロジに存在する前に、検出を設定する必要があります。

検出された各ノード (物理または仮想ホスト) は、NNMiがそのノードを積極的に管理しているかどうかに関係なく、ライセンスの限度までカウントします。所有しているNNMiライセンスの内容は、検出方法にも影響を及ぼします。

ライセンス情報を追跡する際には、以下の点に注意してください。

- **消費量**:NNMiは、NNMiのライセンス容量限界までノードを検出および管理します (切り上げ)。
  - VMware環境:デバイスプロファイルがvnwareVMの各デバイスは、1/10のノードと同等です。
  - 他のすべてのデバイスは1つの検出されたノードと同等です。

ライセンス限度の詳細については、NNMi管理者用のヘルプの「NNMiライセンスを追跡する」を参照してください。

- 検出されたノードの数がライセンスされた容量限界に到達または超えた場合、次のいずれかが行われないかぎり、新しいノードは検出されません。
  - ライセンス拡張をインストールする。
  - 設定を確認し、NNMi検出をネットワーク環境内の重要なノードのみに限定する。次にノードを削除し、NNMiの再検出でノードの管理対象インベントリをリセットする。

注: 多数のノードを検出する設定については、NNMiヘルプを参照してください。

ステータス 監視の考慮事項も、選択肢に影響を及ぼします。State Pollerは、デフォルトではNNMiが検出したデバイスに接続したインタフェースしか監視しません。ネットワークのいくつかの領域ではこのデフォルト設定を変更できるため、職責の範囲を超えたデバイスの検出が可能になります。(StatePollerの詳細については、「NNMi状態ポーリング」(87ページ) を参照してください。)

NNMiには、次の2つの基本的な検出設定モデルがあります。

- **リストベース検出**—NNMiに、リストのシードによってどのデバイスをデータベースに追加し、監視するかを明示的に指定します。
- **ルールベース検出**— NNMiにネットワークのどの領域とデバイスタイプをデータベースに追加するかを伝え、NNMiに各領域の開始アドレスを指定して、NNMiに定義されたデバイスを検出させます。

リストベース検出とルールベース検出を自由に組み合わせて、NNMiの検出対象を設定できます。初回の検出によってこれらのデバイスがNNMiトポロジに追加され、スパイラル検出ではネットワークが日常的に再検出されるため、トポロジは常に最新の状態が維持されます。

注: NNMiでは、テナントを使用して重複アドレスドメインを含むネットワークに対応します。重複アドレスドメインは、ネットワーク管理ドメインの静的ネットワークアドレス変換 (NAT)、動的ネットワークアドレス変換 (NAT)、またはポートアドレス変換 (PAT) 領域内に存在する可能性があります。そのようなネットワークの場合、重複アドレスドメインを異なるテナントに配置します (これはシード済み検出を使用して行います)。詳細については、NNMiヘルプを参照してください。

注: NNMiを使用してVMwareハイパーバイザーベースの仮想ネットワークを管理する場合は、管理者用のヘルプの「仮想環境内のテナント」のヘルプトピックを参照してください。

ヒント: マルチテナントを設定する場合は、ネットワーク検出を開始する前に、テナントを設定してください。

NNMiはデバイスのプロファイルルから属性を導き出す

NNMiはデバイスを検出する際に、SNMPを使用していくつかの属性を直接収集します。重要な属性の1つはMIB IIシステムオブジェクトID (`sysObjectID`) です。システムオブジェクトIDから、NNMiはベンダー、デバイスカテゴリ、デバイスファミリなどの追加属性を導き出します。

検出中、NNMiはMIB IIシステムの性能を収集して、データベースのトポロジ部分に格納します。システム性能は、**ノード**フォームに表示されます。ただし、これらの性能はNNMiの他の部分 (つまり、監視設定) では使用されません。NNMiでは、デバイスカテゴリ (システムオブジェクトIDのデバイスのプロファイルルにより) を使用して、デバイスをノードグループに分類します。ノードビュー表では、**「デバイスカテゴリ」**列に各ノードのデバイスカテゴリが明示されます。

注: (SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコル (たとえば、VMware環境用のSOAPプロトコル) を使用できます。

NNMiには、リリース時に入手できた数千のシステムオブジェクトIDのデバイスのプロファイルルが付属しています。ご使用の環境内にしかないデバイス用にデバイスのプロファイルルをカスタム設定して、これらのデバイスをカテゴリ、ベンダーなどに対応付けることができます。

検出の計画

以下の領域で決定します。

基本的な検出方法を選択する

完全なリストベース検出を行うのか、完全なルールベース検出を行うのか、それともこの2つの方法を組み合わせて使用するのかを決定します。

リストに基づいた検出

リストベース検出では、NNMiで検出する各ノードを (検出シードとして) 明確に指定します。

注: NNMiでは、テナントを使用して重複アドレスドメインを含むネットワークに対応します。重複アドレスドメインは、ネットワーク管理ドメインの静的ネットワークアドレス変換 (NAT)、動的ネットワークアドレス変換 (NAT)、またはポートアドレス変換 (PAT) 領域内に存在する可能性があります。そのようなネットワークの場合、重複アドレスドメインを異なるテナントに配置します (これはシード済み検出を使用して行います)。詳細については、NNMiヘルプを参照してください。

注: NNMiを使用してVMwareハイパーバイザーベースの仮想ネットワークを管理する場合は、管理者用のヘルプの「仮想環境内のテナント」のヘルプトピックを参照してください。

ヒント: マルチテナントを設定する場合は、リストベース検出を使用して検出することをお勧めします。

リストベース検出のみを使用することの利点を以下に示します。

- NNMiの管理対象を厳密に管理できます。
- 検出時にデフォルト以外のテナントの仕様をサポートします。
- 設定が最も簡単です。
- 固定的なネットワークに適しています。
- NNMiを初めて使用する場合に適した方法です。自動検出ルールを、徐々に追加していくことができます。

リストベース検出のみを使用することのデメリットを以下に示します。

- NNMiは、ネットワークに新規ノードが追加されても検出しません。
- 検出対象とするノードのリストを指定しなければなりません。

ルールベースの検出

ルールベース検出では、NNMiが検出してNNMiトポロジに入れるネットワークの領域を定義するために1つ以上の自動検出ルールを作成します。各々のルールに対して、1つ以上の検出シードを (シードを明確に指定するかpingスィープを有効にすることにより) 指定する必要があります。それによりNNMiがネットワークを自動的に検出します。

ルールベース検出を使用することの利点を以下に示します。

- 大規模なネットワークに適しています。NNMiは大量の数のデバイスを、最低限の設定項目に基づいて検出できます。
- 頻繁に変わるネットワークに適しています。ネットワークに追加した新しいデバイスは、管理者が介在しなくても検出されます (各デバイスは自動検出ルールの適用範囲内であることが前提)。
- 新規デバイスをタイミングよく管理するためのサービス内容合意書や、許可されていない新規デバイスがあれば注意を与えるためのセキュリティガイドラインを順守するために、新しいデバイスがネットワークに追加されると検出されます。

ルールベース検出を使用することのデメリットを以下に示します。

- すぐにライセンス限度に達してしまいます。
- ネットワークの構造によっては、自動検出ルールの調整が複雑になることがあります。
- 自動検出ルールが非常に広範囲で、管理しようしている数よりも多くのデバイスをNNMiが検出する場合は、不要なデバイスをNNMiトポロジから削除できます。ノードの削除には時間がかかる可能性があります。
- すべての非シードノードは、検出時にデフォルトのテナントを受信します。NNMiマルチテナント方式を使用する場合は、検出後にテナント割り当てを更新する必要があります。

自動検出ルール

自動検出ルールを設定するときは、以下の内容を指定します。

- 自動検出ルールの順序
- 検出から除外するデバイス

- Pingスィープを使用するかどうか
- 該当するものがある場合、使用する検出シード

自動検出ルールの順序

自動検出ルールの**順序**属性の値は、検出範囲に次のように影響します。

- IPアドレス範囲

  デバイスが2つの自動検出ルールに該当すると、順序番号が小さい方の自動検出ルールの設定が適用されます。たとえばある自動検出ルールによりIPアドレスの一式が除外されると、それより大きな順序番号の自動検出ルールはこれらのノードを処理せず、そのアドレス範囲内のノードは、検出シードとしてリストされない限り検出されません。
- システムオブジェクトIDの範囲
  - 自動検出ルールにIPアドレス範囲が含まれていない場合は、システムオブジェクトIDの設定が、それより大きな順序番号のすべての自動検出ルールに適用されます。
  - 自動検出ルールにIPアドレス範囲が含まれている場合、システムオブジェクトID範囲は自動検出ルール内でのみ適用されます。

デバイスを検出から除外

- 特定のオブジェクトタイプが検出されないようにするには、検出したくないシステムオブジェクトIDを無視する自動検出ルールを、順序番号を小さくして作成します。このルールにIPアドレス範囲を含めないでください。この自動検出ルールに小さい順序番号を付けることで、このルールに一致するオブジェクトを検出プロセスはすぐにとばします。
- IPアドレス範囲またはシステムオブジェクトID範囲の**ルールにより無視された**設定は、その自動検出ルールのみに影響します。無視される範囲内に含まれるデバイスは、別の自動検出ルールに含めることが可能です。

注: 一部のネットワークでは、Hot Standby Router Protocol (HSRP) やVirtual Router Redundancy Protocol (VRRP) などのルーティングプロトコルを使用して、ルーターに冗長性を持たせています。HSRPを使用するときのように、ルーターがルーター冗長グループ (RRG) で設定されている場合、RRGで設定されているルーターは保護されたIPアドレス (1つがアクティブで、1つがスタンバイ) を共有します。NNMiは、同じ保護されたIPアドレスを使用して設定された複数のRRGの検出および管理をサポートしません。各RRGには固有の保護されたIPアドレスが必要です。

Pingスィープ

pingスィープを使用して、設定した自動検出ルールのIPアドレス範囲内のデバイスを検索することができます。初期検出では、すべてのルールでpingスィープを有効にするとよいでしょう。そうすることで十分な情報がNNMi検出に提供されるので、検出シードを設定する必要がなくなります。

注: pingスィープは、16ビット以下のサブネット (たとえば`10.10.*.*`) で機能します。

pingスィープは特に、ISPネットワークのように制御が不要なWAN全体でのデバイスの検出で便利です。

注: ファイアウォールはpingスィープをネットワークに対する攻撃としてみなすことがよくあり、その場合、ファイアウォールはpingスィープを発信したデバイスからのすべてのトラフィックをブロックすることがあります。

ヒント:pingスィープは、小さな検出範囲にのみ有効にしてください。

自動検出ルールの検出シード

自動検出ルールごとに少なくとも1つの検出シードを指定してください。検出シードを指定するためのオプションを以下に示します。

- [**設定**] ワークスペースの [**検出**] にある [**シード**] をクリックして [**検出シード**] フォームのシードを入力します。
- `nnmloadseeds.ovpl`コマンドを使用して、シードファイルから情報をロードします。
- 少なくとも初回の検出で、pingスィープをルールに対して有効にします。
- SNMPトラップをNNMi管理サーバーに送信するようにデバイスを設定します。

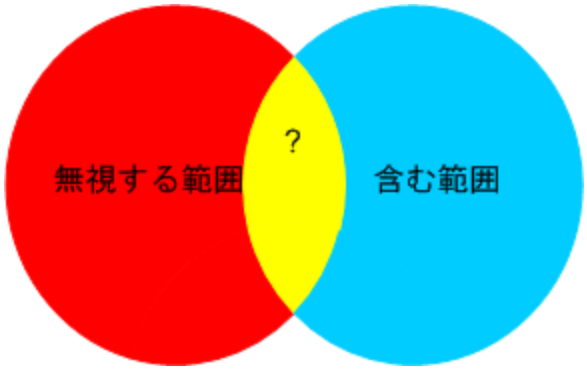
自動検出ルールのベストプラクティス

- NNMiはすべての検出対象デバイスを自動的に管理するため、管理するネットワークの範囲に厳密に一致するIPアドレス範囲を使用してください。
  - 複数のIPアドレス範囲を1つの自動検出ルール内で使用して、検出を限定することができます。
  - 自動検出ルールに大きなIPアドレス範囲を追加した後に、そのルール内の検出からいくつかのIPアドレスを除外することができます。
- システムオブジェクトID範囲の指定は接頭部分であり、絶対値ではありません。たとえば、範囲1.3.6.1.4.1.11は1.3.6.1.4.1.11.*と同じです。

検出ルールの重複

以下の図に、重複する2つの検出範囲を示します。左側の円は、NNMi検出で無視されるIPアドレス範囲またはシステムオブジェクトID範囲を表しています。右側の円は、NNMi検出で検出されて含まれるIPアドレス範囲またはシステムオブジェクトID範囲を表しています。重複している領域は、これらの自動検出ルールの順序に応じて検出に含まれるか無視されます。

重複している検出範囲



デバイスタイプ検出を制限する

ネットワーク内のプリンター以外のすべてのHPデバイスを検出するには、HPエンタープライズシステムオブジェクトID (1.3.6.1.4.1.11) を含む範囲を持つ1つの自動検出ルールを作成します。この自動検出ルールで、HPプリンター (1.3.6.1.4.1.11.2.3 9) のシステムオブジェクトIDを無視する2番目の範囲を作成します。IPアドレス範囲を未設定のままにしてください。

ノード名の解決

デフォルトでは、NNMiはノードを次の順序で識別しようとします。

1. 短いDNS名
2. 短いsysName
3. IPアドレス

> **注:** ノードのホスト名を変更した場合、NNMiデータで名前変更が反映されるまでに時間がかかります。これは、パフォーマンスを向上させるために、NNMiがDNS名をキャッシュするためです。

以下のシナリオでは、ノード名解決のデフォルト順序を変更したほうがよい場合を説明しています。

- 組織がDNS設定の更新を外部者にまかせている場合、ネットワークに新しいデバイスが追加されるごとにそのsysNameを定義するポリシーを設定できます。この場合、sysNameの選択をノード名解決の最初の選択肢として設定して、新しいデバイスがネットワークに導入されるとすぐにNNMiが検出できるようにします。(sysNameを、そのデバイスを使用している間は維持します。)
- 組織が管理対象デバイスのsysNameを設定も維持もしない場合、sysNameをノード名解決の3番目の選択肢として選択します。

> **ヒント:** DNS完全名またはDNS短縮名を基本的な命名方法として使用している場合、NNMi管理サーバーからすべての管理対象デバイスへの順方向と逆方向のDNS解決があることを確認してください。

注: DNS完全名が命名方法の場合、トポロジマップ上のラベルを長くできます。

ヒント: NNMiでは最小のループバックアドレスをCiscoデバイスの管理アドレスとして選択されるため、各Ciscoデバイスの最小のループバックアドレス上にDNS解決を配置してください。

サブネット接続ルール

リストベース検出のみ

リストベース検出では、NNMiはサブネット接続ルールを使用してWAN上の接続を検出します。NNMiは予測される接続の各末端で検出したデバイスのサブネットメンバーシップを評価し (IPアドレスとサブネット接頭部を調べて)、サブネット接続ルールで一致があるか調べます。

ルールベース検出のみ

自動検出ルールが有効でNNMiが/28と/31の間のサブネット接頭部が設定されたデバイスを見つけると、

1. NNMiは適用可能なサブネット接続ルールについて調べます。
2. 一致が見つかると、NNMiはサブネット内の有効な各アドレスをヒントとして使用して、そのアドレスでの検出を試みます。

ヒント: ヒント: デフォルトの接続ルールを使用してください。問題がある場合のみそれらを変更してください。

検出シード

検出シードとして使用するデバイスをリストします。

ヒント: 優先管理IPアドレスを選択するNNMiのルールの1つによって、最初に検出したIPアドレスを管理アドレスとして使用することが指定されます。優先IPアドレスをシードアドレスとして設定することにより、NNMiに影響を与えることができます。

ヒント: Ciscoデバイスの場合、ループバックアドレスを検出シードとして使用してください。ループバックアドレスが、デバイス上の他のアドレスより確実に到達可能であるためです。DNSが、デバイスホスト名からループバックアドレスを解明するように正しく設定されていることを確認します。

リストベース検出のみ

リストベース検出の場合、NNMiの管理対象にするすべてのデバイスをリストします。このリストを、資産管理ソフトウェアから、または他のツールからエクスポートすることが可能です。

NNMiはこのリストにデバイスを自動的に追加することがないため、責任を負っているデバイスだけがリストに追加含まれるようにするか、監視/ステータス計算に影響を及ぼすデバイスだけがリストに含まれるようにしてください。

ルールベース検出のみ

ルールベース検出の場合、検出シードは省略可能です。

- pingスィープが自動検出ルールに対して有効の場合、そのルールのシードを指定する必要はありません。
- pingスィープが無効な各自動検出ルールで、ルールごとに少なくとも1つのシードを確認してください。ルールにIPアドレス範囲が複数含まれる場合、ルーターはWANリンク全体のARPエントリを維持しないため、それぞれのルーティング可能範囲でシードが必要になります。

ヒント: ルールベース検出を最も完璧なものにするためには、スイッチではなくルーターを検出シードとして使用してください。一般にルーターはスイッチより大きなARPキャッシュを持っているためです。検出したいネットワークにコアルーターが接続されていれば、検出シードとしては最適な選択肢になります。

再検出の間隔

NNMiは、データベース内の各デバイスの設定情報を、設定された再検出間隔に従って再チェックします。さらに、NNMiは自動検出ルールの対象となる各ルーターからARPキャッシュを収集して、ネットワーク上に新しいノードがあるか調べます。

デバイスの通信関連の設定に、インタフェースの番号変更のような変更があると、NNMiは自動的に、そのデバイスとその隣接デバイスに関するデータを更新します。

次のような変更では自動再検出は行われません。デバイスは設定された再検出間隔に基づいて更新されます。

- ノード内の変更 (たとえば、ファームウェアアップグレードまたは接点システム)。
- ネットワークに追加された新しいノード。

ネットワーク内の変更のレベルに合った再検出間隔を選択します。非常に動的なネットワークでは、最低24時間の間隔を使用するとよいでしょう。これより安定したネットワークでは、その期間を広げることができます。

オブジェクトを検出しない

NNMiでは、NNMiが特定のオブジェクトを無視するように設定する3つの方法があります。

- **[通信の設定]** フォームで、ICMP通信またはSNMP通信あるいはその両方を、グローバルレベル、通信領域レベル、または特定のホスト名またはIPアドレスといった異なるレベルでオフにできます。これらのプロトコルのいずれかまたは両方を無効にした場合の影響の詳細については、「ポーリングプロトコル」(48ページ)を参照してください。
- **[検出の設定]** フォームで、NNMiに特定のIPアドレスやSNMPシステムオブジェクトIDからヒントを収集しないように指示する自動検出ルールを設定できます。この基準に一致するノードはマップとデータベース上で存在し続けますが、スパイラル検出はこれらのIPアドレスまたはオブジェクトタイプを超える隣接デバイスまで行われません。

- **[検出の設定]** フォームで、特定のIPアドレス範囲または特定のIPアドレス、あるいはその両方をデータベースから除外するようNNMiに指示する自動検出ルールを設定できます。スパイラル検出では、あらゆるノードのアドレスリストでこれらのアドレスを表示したり、デバイス間に接続を確立するときこれらのアドレスを使用することがないので、NNMiがこれらのアドレスの使用状況を監視することはありません。
- **[検出の設定]** フォームの **[除外対象IPアドレス]** タブで、除外対象IPアドレスフィルターを設定して、IPアドレス範囲を検出から除外することができます。

  ノードが検出された後にそのノードのすべてのIPアドレスが [除外対象IPアドレス] リストに入力された場合、NNMiはそのノードを削除しません。また、NNMi管理者がNNMiデータベースからそのノードを意図的に削除しない限り、NNMiがノードの履歴全体を削除することはありません。

  注: IPアドレス範囲を除外する場合、ネットワーク管理ドメインの静的ネットワークアドレス変換 (NAT)、動的ネットワークアドレス変換 (NAT)、またはポートアドレス変換 (PAT) 領域内の重複アドレスも除外されます。

  NNMiでは、テナントを使用して重複アドレスドメインを含むネットワークに対応します。そのようなネットワークの場合、重複アドレスドメインを異なるテナントに配置します (これはシード済み検出を使用して行います)。詳細については、NNMiヘルプを参照してください。
- **[検出の設定]** フォームの **[除外対象インタフェース]** タブで、インタフェースグループを選択して、特定のタイプのインタフェースを検出プロセスから除外することができます。詳細については、NNMiヘルプを参照してください。

インタフェースの検出範囲

NNMiにより、フィルターを定義して検出されるインタフェース範囲を指定できます。これは、ノードが大きく、インタフェースのサブセットのみを検出する場合に特に便利です。[除外対象インタフェース] オプションを使用する場合は、デバイスから情報を取得した後でインタフェースがフィルタリングされますが、検出するインタフェース範囲を指定する場合は、NNMiから範囲外のインタフェースに関する情報は要求されません。そのため、範囲ベースの検出では、大きいデバイスの検出パフォーマンスを向上させることができます (特にそのようなデバイスのすべてのインタフェースを管理しない場合)。

[検出の設定] フォームの **[含まれるインタフェース範囲]** タブで定義する含まれるインタフェース範囲のフィルターでは、システムオブジェクトIDプレフィック値およびifIndex値を使用してインタフェース範囲を定義します。詳細については、NNMiヘルプを参照してください。

NNMiによる仮想IPアドレスの監視

NNMiは、仮想IPアドレスを共有するクラスター化されたサーバーなどのデバイスを検出および監視します。クラスターが新しいアクティブノードにフェイルオーバーすると、NNMiはその仮想IPアドレスを新しいアクティブノードに関連付けます。フェイルオーバーしてからNNMiが変更を検出するまでにしばらく時間がかかるため、この関連付けはすぐには行われません。

特定の状況に合わせてNNMiを設定するため、いくつかのアクションを実行できます。

NNMiで仮想IPアドレスを監視する場合は、以下のオプションのいずれか1つだけを使用してください。

- オプション1:このオプションの場合、NNMiはN+1個の非SNMPデバイスを管理します。ここでNは、非仮想IPアドレスによって検出されたクラスターに属するメンバーの数です。NNMiは、さらにもう1つの (+1) 非SNMPノードを検出し、仮想IPアドレスを使用して設定します。

  NNMiが仮想IPアドレスを検出する動作を停止しないでください。この方法を使用することによりNNMiは、仮想IPアドレスと、この仮想IPアドレスを使用するように設定されたデバイスのネットワークインタフェースカード (NIC) に関連付けられている物理IPアドレスを検出します。NNMiは、各デバイスを別々の非SNMPノードとして検出および監視します。
- オプション2:デバイスの物理IPアドレスをクラスター化されたサーバーの優先される管理アドレスとして使用するようにNNMiを設定します。この方法の詳細については、NNMiのヘルプの「特定ノードの設定フォーム (通信設定)」のトピックを参照してください。

注: NNMiは、アクティブノードから新しいアクティブノードへの仮想IPアドレスの移行をすぐには認識しない場合があります。NNMiは、クラスター内の現在のアクティブノードとは別のノードを使用して仮想IPアドレスのステータスを表示することがあります。

NNMiで仮想IPアドレスを監視しない場合は、NNMiコンソールを使用して以下の手順を実行します。

1. [**設定**] ワークスペースの [**検出の設定**] をクリックします。
2. [**除外対象IPアドレス**] タブをクリックします。
3. 仮想IPアドレスまたはアドレス範囲を、検出対象から除外するアドレスの一覧に追加します。
4. 変更を保存します。

SNMPトラップからの検出ヒントの使用

NNMiは受信したSNMPトラップのソースIPアドレスをNNMi自動検出ルールに対するヒントとして処理するようになりました。

SNMPトラップインシデントの詳細については、NNMiの『管理者用のヘルプ』を参照してください。

検出の設定

このセクションでは、設定のヒントを一覧にし、いくつかの設定例について説明します。このセクションの情報を読んだ後で、特定の手順のNNMiヘルプの「検出の設定」を参照してください。

注: NNMiは、[**検出シード**] フォームを [**保存して閉じる**] とすぐにシードから検出を開始するため、シードを設定する前に以下のことを必ず行ってください。

- すべての通信設定を完了する。
- すべての自動検出ルール (ある場合) を完了する。
- サブネット接続ルールを設定する。

- 名前解決設定を設定する。
- NNMiコンソールまでさかのぼってすべての設定フォームの [**保存して閉じる**] を行う。

ヒント:大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。

自動検出ルールを設定する場合のヒント

新しい自動検出ルールを定義するときは、それぞれの設定を慎重に確認してください。新しいルールでは、自動検出がデフォルトで有効になっており、IPアドレス範囲はデフォルトで含まれており、システムオブジェクトID範囲はデフォルトで無視されます。

シードを設定する場合のヒント

シードを設定するときは、以下のベストプラクティスを検討してください。

- 検出対象ノードがリストされたファイルがすでにある場合は、この情報をシードファイルとして書式設定し、`nnmloadseeds.ovpl`コマンドを使用してそのノードリストをNNMiにインポートします。
- シードファイルで、管理アドレスとしてNNMiが選択するIPアドレスに影響を与える手段としてIPアドレスを指定します。(ホスト名を使用すると、DNSはIPアドレスを各ノードに提供します。)
- シードファイルのエントリとして適切な書式を以下に示します。

  IP_address1 # node name

  IP_address2, <tenant_UUID_or_tenant_name> # node name

  以下の書式は、NNMiと人間の両方が容易に理解できます。
- 保守目的のため、使用するシードファイルは1つだけにすることをお勧めします。ノードを必要に応じて追加して、`nnmloadseeds.ovpl`コマンドを再度実行します。NNMiは新しいノードを検出しますが、既存のノードは再判定しません。

注: シードファイルをロードできない場合、`nmsproc` (644権限) でファイルを読み取れるようにします。

- ノードをシードファイルから削除しても、NNMiトポロジからは削除されません。ノードは直接NNMiコンソールで削除してください。
- ノードをマップやインベントリビューから削除しても、シードは削除されません。
- NNMiでノードを再検出する場合は、そのノードをマップまたはインベントリビューから削除し、NNMiコンソールの [**設定**] ワークスペースの [**検出**] エリアにある [**シード**] フォームから削除してから、そのノードをNNMiコンソールで再入力するか、`nnmloadseeds.ovpl`コマンドを実行します。

ルールベース検出のみ

- 検出ルールを、そのルールのシードを指定する前に、完全に設定します。つまり、[**検出の設定**] フォームで [**保存して閉じる**] をクリックします。([**検出シード**] フォームは、データベースモデルの [**検出の設定**] フォームに含まれていない個別のフォームです。結果として、[**検出シード**] フォームについての情報を保存すると、NNMiによってシード設定はただちに更新されます。)

リンクアグリゲーションの検出

注: リンクアグリゲーションには、NNMi AdvancedライセンスまたはNNMi Premiumライセンスが必要です。

リンクアグリゲーション (LAG) プロトコルによって、ネットワーク管理者はアグリゲータインタフェースとしてスイッチでインタフェースのセットを設定できます。この設定により、帯域幅、データ通信速度、冗長性の向上と並行して、複数のインタフェースを使用して別のデバイスにアグリゲータレイヤー2接続を作成します。

詳細については、NNMiヘルプで**リンクアグリゲーション**を検索してください。

サーバーからスイッチへのリンクアグリゲーション (S2SLA) の検出

注: リンクアグリゲーションには、NNMi AdvancedライセンスまたはNNMi Premiumライセンスが必要です。

ネットワーク管理者は、信頼性の向上およびサーバーとスイッチ間のリソースのさらなる活用を頻繁に求められます。多くのネットワーク管理者が、ネットワーク機器プロバイダーでは広範な使用法があるために、Link Aggregation Configuration Protocol (LACP) の使用を選択します。LACPは、ITエンジニアがサーバーからスイッチへの設定の両側でポートを結合した後に、自動的にネゴシエーションされます。

ネットワーク管理者は多くの場合、信頼性および必要なサーバーとスイッチ間のリソースの使用を実現するために、2種類のうちいずれかのスイッチからサーバーへの接続を使用することを選択します。

- オプション1:サーバーの2つ以上のポートを結合し、スイッチにある同じ番号のポートに接続します。サーバーまたはスイッチのポートに障害が発生すると、バックアップポートがアクティブ化されます。
- オプション2:サーバーとスイッチの両方を結合し、集約してすべてのポートの集約合計帯域幅を提供します。

NNMiは、サーバーからスイッチへのリンクアグリゲーション (S2SLA) の検出機能を提供し、スイッチからサーバーへの接続の管理を容易にします。NNMiがノードのS2SLA情報を適切に検出できるか確認するには、以下のタスクを実行してください。

- デフォルトで、LinuxはSNMPエージェントパッケージ、Net-SNMPをインストールしません。Net-SNMPがNNMi管理サーバーにない場合、インストールする必要があります。
- Linux上で結合しているインタフェースは、集約されたインタフェースの1つのMACアドレスを前提

とすることがありますが、必須ではありません。結合されたインタフェースは、どのサーバーのインタフェースにも属さないMACアドレスを持つことができます。

ヒント: 集約でのすべてのインタフェースで同じMACアドレスが使用されます。SNMPインタフェーステーブルを確認して、アグリゲータインタフェースおよび集約されたインタフェースに同じMACアドレスを返します。共有MACアドレスは送信パケットで使用されます。アクセススイッチのFDBテーブルは、スイッチの集約されたインタフェースを介して伝えられると、このMACアドレスを示します。

元のMACアドレスを表示するには、以下のコマンドを実行します。

```
cat /proc/net/bonding/bond0
```

検出の評価

このセクションでは、検出の進行状況と成功したかどうかを判定する方法を記載しています。

初期検出の進行状況をたどる

NNMi検出は、動的かつ継続的です。完了することはないため、「検出完了」のメッセージが表示されることはありません。初回の検出と接続には、多少の時間がかかります。初期検出の進行状況を測定する方法を以下に示します。

- [**システム情報**] ウィンドウの [**データベース**] タブで、ノードカウントが予想レベルに達して一定になるのを監視します。このウィンドウは自動的に更新されません。初期検出時に、[**システム情報**] ウィンドウを複数回開きます。
- [**設定**] ワークスペースの [**検出**] で、[**シード**] ページを確認します。このページを、すべてのシードに「ノードが作成されました」結果が表示されるまで更新してください。「ノードが作成されました」結果は、デバイスがトポロジデータベースに追加されたことを示します。この結果は、NNMiがデバイスからすべての情報を収集してデバイスの接続を処理したことを示すものではありません。
- 代表ノードの [**ノード**] フォームを開きます。[**検出状態**] フィールド ([**全般**] タブにあります) が `Discovery Completed`に移行するときには、NNMiはノードの基本特性、ノードのARPキャッシュ、隣接検出プロトコル (該当する場合) の収集を済ませています。この状態は、NNMiがデバイスの接続解析を完了したことを示すものではありません。
- [**ノード**] インベントリビューで、ネットワークのさまざまな領域のキーデバイスが存在していることを確認します。
- 代表ノードの [**レイヤー2近隣接続ビュー**] を開き、その領域の接続解析が完了したかどうかを確認します。
- [**レイヤー2接続**] および [**VLAN**] インベントリビューを調べて、レイヤー2処理の進行状況を測定します。

すべてのシードが検出されたか?

1. **[設定]** ワークスペースの **[検出]** で、**[シード]** をクリックします。
2. **[シード]** ページで、ノードのリストを **[検出シードの結果]** 列でソートします。ノードがエラー状態の場合は、以下について検討してください。
   - **ノードに到達できなかったかDNS名またはIPアドレスが解決されなかったために検出が失敗した** — これらのタイプの失敗に対しては、ノードへのネットワーク接続を確認して、DNS名解決が正しいかどうかを調べてください。DNS問題に対処するには、IPアドレスを使用してノードをシードするか、ホスト名を`hostnolookup.conf`ファイルに加えます。ホスト名に解決されるべきではないIPアドレスが原因で発生する問題に対処するには、該当するIPアドレスを`ipnolookup.conf`ファイルに含めます。詳細については、`hostnolookup.conf`および`ipnolookup.conf`のリファレンスページ、またはLinuxのマニュアルページを参照してください。
   - **ライセンスノード数超過**—この状況は、すでに検出されたデバイス数がライセンス限度に達したときに発生します。検出したノードをいくつか削除するか、ノードパックライセンスを追加購入します。

     ライセンス情報を追跡する際には、以下の点に注意してください。
     - **消費量**:NNMiは、NNMiのライセンス容量限界までノードを検出および管理します (切り上げ)。
       - VMware環境:デバイスプロファイルが`vnwareVM`の各デバイスは、1/10のノードと同等です。
       - 他のすべてのデバイスは1つの検出されたノードと同等です。

     ライセンス限度の詳細については、NNMi管理者用のヘルプの「NNMiライセンスを追跡する」を参照してください。
   - **ノードが検出されたがSNMP応答がない**—SNMP通信の問題は、シードされたデバイスだけでなく自動検出によって検出されたデバイスにも発生します。詳細については、「通信の評価」(63ページ)を参照してください。

すべてのノードには有効なデバイスのプロファイルルがあるか?

1. **[ノード]** インベントリビューを開きます。
2. **[デバイスのプロファイル]** 列を、「デバイスのプロファイルなし」文字列が含まれるようにフィルタリングします。
3. ノードが検出されてもデバイスのプロファイルがない場合は、**[設定] > [デバイスのプロファイル]** で新規デバイスのプロファイルを追加してから、ノード上で設定ポーリングを実行してそのデータを更新します。

すべてのノードが正しく検出されたか?

検出の問題を回避するには、管理ドメイン内の他のドメインには表示されない固有のIPアドレスを使用するノードのみをNNMiで管理するようにします。たとえば、ノードが突然消えたり、データベース内の別のノードとマージされたりし、そのノードがルーター冗長グループ (RRG) の一部になっている場合には、特別な要件があります。RRGに参加しているルーターを管理するには、ルーターの管理アドレスとして固有のIPアドレス (保護されたアドレス以外) を使用する必要があり、そのアドレスでSNMPを有効にする必要があります。

注: NNMiは、保護されたIPアドレスを管理アドレスとして使用しようとすると、ルーターを適切に管理できません。

[**ノード**] インベントリビューでデータを調べます。管理アドレスがないノードがある場合は、これらのノードの通信設定を「すべてのノードがSNMP用に設定されましたか?」(63ページ)の説明に従って確認します。

予想したノードが [**ノード**] インベントリビューにない場合は、以下について確認します。

- 見つからなかったノードごとに、検出プロトコル (たとえばCDP) が正しく設定されていることを確認します。
- 見つからないノードがWAN上にある場合、そのノードを含む自動検出ルールのpingスィープを有効にします。

自動検出ルール

リストベース検出のみ。

予期しない検出結果に遭遇した場合は、自動検出ルールを再検討します。

NNMi検出でアドレスヒントが見つかる場合は、最初の一致ルールを使用してノードを作成するかどうかを判定しています。一致するルールがない場合、NNMi検出はヒントを廃棄します。自動検出ルールの順序番号によって、自動検出ルール設定が適用される順序が決まります。

それぞれの自動検出ルールで、以下の設定を確認してください。

- [**含まれているノードの検出**] を有効にし、自動検出がルールに実行されるようにする必要があります。
- 以下の設定が、検出したいノードのタイプに対して正しいかどうかを確認します。
  - **SNMPデバイスの検出**
  - **非SNMPデバイスの検出**

  デフォルトではルーターとスイッチのみが検出されて、SNMP以外のノードは検出されないことを忘れないでください。ご使用の環境を考慮せずにこれらの設定を有効にすると、NNMiが予期した以上のノードを検出してしまう可能性があります。

IPアドレス範囲

検出ヒントのIPアドレスは、IPアドレス範囲リスト内の [**ルールに含める**] エントリに一致する必要があります。含まれるIPアドレス範囲が自動検出ルールの中にない場合、すべてのアドレスヒントが一致とみなされます。(この場合は、「自動検出ルールを設定する場合のヒント」(78ページ)を参照してください。)さらに、ヒントは「**ルールにより無視された**」とマークされたエントリと一致してはなりません。すべてのチェックが正常に一致すると、このルールの設定がヒントの処理に使用されます。

- 予想したデバイスのいくつかが検出されない場合、設定したIP範囲を確認してそのデバイスのIPアドレスが範囲の中に含まれていて小さい順序番号のルールで無視されないようにしてください。
- 必要以上のデバイスが検出されている場合は、含む範囲を変更するか、検出したくないデバイスのIPアドレスの無視される範囲を追加してください。また、[**SNMPデバイスの検出**] も有効かどうかを確認します。

システムオブジェクトIDの範囲

検出ヒントのシステムオブジェクトID (OID) は、システムオブジェクトID範囲リストの中の [**ルールに含める**] エントリと一致する必要があります。含まれるシステムオブジェクトID範囲が自動検出ルールの中にない場合、すべてのオブジェクトIDが一致とみなされます。さらに、OIDは**「ルールにより無視された」**とマークされたエントリと一致してはなりません。すべてのチェックが正常に一致すると、このルールの設定がヒントの処理に使用されます。

- システムオブジェクトID範囲を使用して、自動検出を拡大してデフォルトのルーターおよびスイッチ以外も含めるか、特定のルーターおよびスイッチを除外します。
- 各ノードは、検出されてトポロジデータベースに追加される前に指定されたIPアドレス範囲とシステムオブジェクトID範囲の両方と一致する必要があります。

すべての接続とVLANは正しいか?

NNMiはレイヤー2接続とVLANを、デバイスがトポロジに追加された後の別個のステップとして作成します。NNMiに接続とVLANを評価する前の初期検出として十分な時間を考慮してください。

レイヤー2接続の評価

レイヤー2の接続を評価するには、対象とする各ネットワーク領域のノードグループを作成し、続いてそのノードグループのトポロジマップを表示します。([**ノードグループ**] インベントリで、ノードグループを選択して、**[アクション] > [ノードグループマップ]** をクリックします。)このマップで他のノードに接続していないノードを探します。

VLANを評価するには、[**VLAN**] インベントリビューから、各々の [**VLAN**] フォームを開いて、そのVLANのポートのリストを調べます。

NNMi検出と重複MACアドレス

検出ではMACアドレスが考慮され、以下の利点があります。

- IPアドレスを変更するDHCPまたはほかのノードのサポートが改善される。
- 重複するIPアドレスを使用して設定されたノードの識別が改善される。
- ホストされるIPアドレスをレポートしないデバイスのサポートが改善される。

NNMiは、検出の実行中、ネットワークデバイス間の通信パスを判断するため、ネットワーク内のEthernetスイッチから転送データベース (FDB) テーブルを読み取ります。NNMiは、これらのFDBテーブルで、検出されたノードに関する情報を検索します。NNMi管理サーバーは、重複するメディアアクセス制御 (MAC) アドレスへのFDB参照を検出すると、以下の処理を行います。

- 検出された2つ以上のノード (同一テナント内のノード、またはデフォルトテナントのノードとそれ以外のテナントのノード) に同じメディアアクセス制御 (MAC) アドレスに関連付けられたインタフェースが含まれる場合、NNMiは、FDBにあるそれらの重複MACアドレスについてレポートされている通信パスを無視します。これにより、それらの重複MACアドレスを含むネットワーク領域のNNMiマップで、接続が失われる場合があります。

  NNMi AdvancedまたはNNMi Premium - グローバルネットワーク管理機能:2つのNNMi管理サーバーが、同じメディアアクセス制御 (MAC) アドレスに関連付けられている1つのインタフェースを含むノードを検出すると、リージョナルNNMi管理サーバーのマップで認識される接続がグローバルNNMi管理サーバーのマップでは失われる可能性があります。
- 1つのノードに同じMACアドレスを持つ複数のインタフェースが含まれる場合、NNMiは、それらのインタフェースについてのすべての通信パス情報を収集し、NNMiマップにその情報を表示します。

データベース (FDB) 情報を転送すると、以下の場合にNNMiが誤ったL2接続を確立する可能性があります。

- FDBがキャッシュとして設定されており、使用されていないデータが含まれている。
- それぞれ異なる (場合によっては競合する) FDBデータを生成するさまざまなベンダーのハードウェアがネットワーク環境に含まれている。

省略可能: NNMi管理者は、特定のノードグループでこのFDBデータを無視するように検出を設定できます。

デバイスを再検出する

1. デバイスの削除を確認するには、デバイスの設定ポーリングを実行します。
2. デバイスを削除します。

そのデバイスがシードの場合、シードを削除し、それからシードを再度追加します。

検出の調整

標準的な検出が行われるようにするためには、検出設定を調整して重大なデバイスと重要なデバイスのみが検出されるようにしてください。

- IPアドレス範囲またはシステムオブジェクトID、あるいはその両方でフィルタリングします。
- 非SNMPデバイスとSNMPデバイス (スイッチでもルーターでもないデバイス) の検出を制限します。

コマンドラインでNNMiデータベースから1つ以上のノードを削除するには、`nnmnodedelete.ovpl`コマンドを使用します。このコマンドにより、NNMiデータベースからノードが削除されますが、シード定義は削除されません。

コマンドラインでNNMiデータベースから1つ以上のシード定義を削除するには、`nnmseeddelete.ovpl`コマンドを使用します。

特別な検出状況は、検出プロトコルコレクションまたはVLANのインデックス付けを無効にすることによって修復できます。詳細については、「特定ノードの検出プロトコルの使用を抑える」(292ページ) または「大規模スイッチのVLANインデックス付けの使用を抑制する」(294ページ)を参照してください。

ログファイルの検出

どの検出クラスに問題があるかを確認するには、`nnm.log`ファイル内で、文字列`com.hp.ov.nms.disco`で始まるクラスの**Exception**というキーワードを含むメッセージを探します。

ログファイルの詳細については、「NNMiロギング」(301ページ)を参照してください。

無番号インタフェース

NNMiでは、グローバルネットワーク管理 (GNM) 環境のものも含め、無番号インタフェースおよび関連するレイヤー2接続を検出し、モニタリングすることができます。

GNM 環境で無番号インタフェースのレイヤー2接続を有効にする場合は、リージョナルマネージャーとグローバルマネージャーの両方で有効にする必要があります。

NNMiの [**設定**] > [**検出**] ワークスペースを使用して、無番号インタフェースのレイヤー2接続を設定 (有効化または無効化) できます。詳細については、管理者用のNNMiヘルプを参照してください。

必要に応じて、`nnmunnumberedcfg.ovpl`コマンドを使用して、無番号インタフェースの接続を設定します。詳細については、`nnmunnumberedcfg.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

> **注:** ノードグループは、リージョナルマネージャーとグローバルマネージャー間で複製されません。

リージョナルマネージャーとグローバルマネージャー間で設定を複製するには、`nnmunnumberedcfg.ovpl`コマンドを使用します。この機能を使用すると、リージョナルマネー

ジャーとグローバルマネージャーで異なるノードグループを定義できます。たとえば、すべてのルーターをグローバルレベルで定義し、ルーターのサブセットのみを各リージョナルマネージャーで定義できます。

グローバルマネージャーは、リージョナルマネージャーと異なる設定にすることを推奨します。たとえば、グローバルマネージャーからノードを直接管理しない限り、データはリージョナルマネージャーでのみ収集されるため、グローバルマネージャーでサブセット (省略可能) を設定する必要はありません。

非応答オブジェクトの削除の制御

オブジェクトが応答しなくなってからの待機日数を指定して、以下の非応答オブジェクトの削除を制御できます。

- 非応答ノード
- 停止している接続

非応答オブジェクトの削除を制御するには、以下の手順を実行します。

1. [**設定**] ワークスペースで、[**検出の設定**] をクリックします。
2. [**非応答オブジェクト制御の削除**] 領域で、該当のオブジェクトを削除するまでにシステムが待機する日数を入力します。ゼロ (0) の値は、ノードが削除されないことを示します。

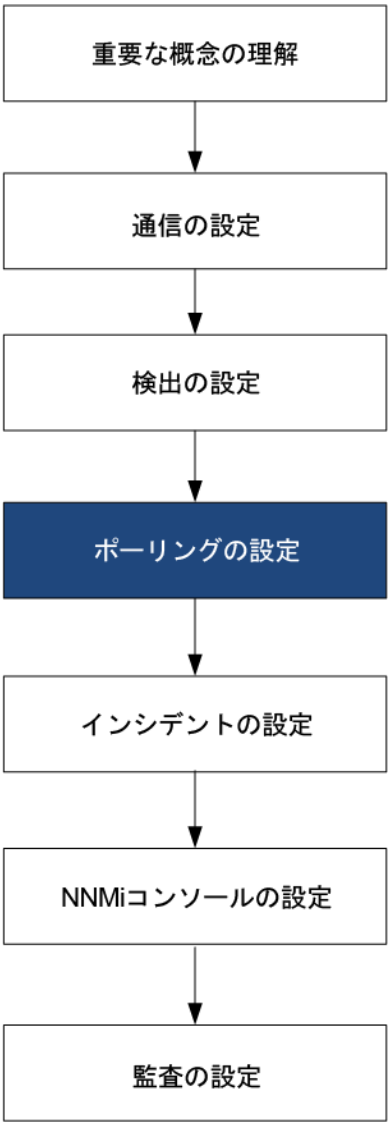
指定した待機期間が経過すると、非応答オブジェクトがデータベースから削除されます。

注: [非応答ノードの削除] が有効な場合、NNMiは以下の状況下にある仮想マシンノードを削除しません。

- VMがSNMPエージェントをサポートしていない
- VMware ToolsがインストールされていないためにVMにIPアドレスがない
- VMのIPアドレス障害モニタリングが設定されていない

詳細については、管理者用のヘルプの「非応答ノードを削除するかどうかの設定」のヘルプトピックを参照してください。

NNMi状態ポーリング



この章では、HP Network Node Manager i Software (NNMi) StatePollerサービスを設定し、ネットワーク監視を拡張および微調整するのに役立つ情報を示します。この章は、NNMiヘルプの情報を補充するものです。監視動作方法の紹介、および監視設定方法の詳細については、NNMiヘルプの「ネットワークの稼働状態をモニタリングする」を参照してください。

この章には、以下のトピックがあります。

状態ポーリングの概念

このセクションでは、State Pollerがポーリンググループの評価に使う順序など、ネットワーク 監視の簡単な概要を示します。このセクションを読んだ後、さらに詳細な情報については「状態ポーリングの計画を作成」(89ページ)に進んでください。

ネットワーク検出と同じように、ネットワークでクリティカルであるか、または最も重要なデバイスのネットワーク 監視に関心を集中する必要があります。NNMiでは、トポロジデータベースでのみデバイスをポーリングできます。NNMiがどのネットワークデバイスを監視するか、使用するポーリングの種類、およびポーリングする間隔を制御できます。

[モニタリングの設定] フォームのインタフェースとノードの設定を使って、デバイスのステータスのポーリングを高度化し、さまざまなクラス、インタフェースの種類、およびノードの種類についてポーリングの種類と間隔を設定することができます。

State Pollerのデータ収集がICMP (ping) 応答を基礎にするように、またはSNMPデータを基礎にするように設定できます。NNMiは、ユーザーが有効にするデータ収集の種類から、実際のMIBオブジェクトへの内部的なマップを自動処理し、設定を大幅に簡単にします。

注: (SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコル (たとえば、VMware環境用のSOAPプロトコル) を使用できます。

ポーリング設定の計画を作成するときは、State Pollerサービス用にインタフェースグループとノードグループをセットアップする方法を注意深く考える必要があります。グループという概念が初めての場合は、その概要について「ノードグループおよびインタフェースグループ」(35ページ)と「ノードインタフェースおよびアドレス階層」(40ページ)を参照してください。

評価の順序

インタフェースまたはノードは複数のグループに属することがあるので、State Pollerは、明確に定義された評価順序で、設定されたポーリング間隔およびポーリング種類を適用します。検出されたトポロジ内の各オブジェクトについて:

1. オブジェクトがインタフェースの場合、State Pollerは基準を満たすインタフェースグループを探します。グループは最も小さい順序番号から最も大きい順序番号へという順序で評価されます。最初に一致するグループが使われ、その時点で評価は停止します。
2. オブジェクトを把握したインタフェースグループがない場合、グループは最も小さい順序番号から最も大きい順序番号への順序で評価されます。最初に一致するグループが使われ、その時点で評価は停止します。含まれているインタフェースのうち、独自の特性に関してインタフェースグループの基準を満たしていないものは、ホストであるノードからポーリング設定を継承します。
3. 検出されたものの、ノードまたはインタフェースの設定定義に含まれないデバイスは、グローバルな監視設定 ([**モニタリングの設定**] フォームの [**デフォルト設定**] タブ) によって監視動作が確定されます。

状態ポーリングの計画を作成

このセクションでは、ポーリング設定チェックリストなど、State Poller設定の計画作成について説明します。監視の計画作成に便利な詳細情報によって、ポーリンググループの作成法が決まり、ポーリングプロセスの間にどの種類のデータを取得する必要があるかが決まります。

ポーリングチェックリスト

次のチェックリストを使って、State Poller設定の計画を作成できます。

- NNMiの監視対象は何ですか?
- オブジェクトの種類、場所、相対的重要性、その他の基準に基づいて、監視対象は論理的にどのように分類できますか?
- NNMiは、各グループをどのくらいの頻度で監視する必要がありますか?
- 監視されるアイテムの情報を取得するために、何のデータを収集する必要がありますか?以下のものが含まれることがあります。
  - ICMP (ping) 応答
  - SNMP障害データ
  - 1つ以上のNNM Performance iSPIに対応するライセンスが1つある場合は、SNMPパフォーマンスデータ
  - 追加のSNMPコンポーネント稼働状態データ

    注: (SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコル (たとえば、VMware環境用のSOAPプロトコル) を使用できます。
- ネットワークデバイスからNNMiにどのSNMPトラップを送信するのですか?

ポーリング設定の例

ポーリング設定プロセスの理解を深めるために、次の例について考えます。ネットワークにProximiTの最新のプロキシサーバーが含まれていると仮定します。これらのデバイスに到達できることを確認する必要がありますが、プロキシサーバーのSNMP監視は要求しません。

1. NNMiで何を監視できますか?

   監視できるのは検出されたもののみであるため、自動検出ルールを設定して、NNMiのデータベースに自分のProximiTプロキシサーバーがあることを確認します。検出の設定の詳細については、「NNMi検出」(66ページ)を参照してください。
2. 監視対象は論理的にどのように分類できますか?

   複数のProximiTプロキシサーバーを1つのグループにまとめ、同じ監視設定を適用することは理にかなっています。デバイスのインタフェース (SNMP) 監視を行っているのですから、インタフェースグループは必要ありません。

このノードグループを使って、ビューをフィルターし、プロキシサーバーのステータスをグループとしてチェックし、グループをサービス停止中にしてファームウェアを更新することもできます。

3. NNMiは、各グループをどのくらいの頻度で監視する必要がありますか?

   サービスレベル契約条項で、プロキシサーバーについて5分間のポーリング間隔で十分です。

4. どのデータを収集する必要があるでしょうか?

   監視設定が他のグループと異なるのは次の点です。ProximiT proxyサーバーの例として、ICMP障害の監視を有効にし、SNMP障害およびポーリングの監視を無効にします。グループについてのSNMP障害モニタリングがない場合、コンポーネント稼働状態モニタリングは適用されません。

5. ネットワークデバイスからNNMiにどのSNMPトラップを送信するのですか?

   次のポーリング間隔を待機せずにトラップが受信される場合、NNMiは一部のSNMPトラップを使用してデバイスをポーリングします。

これらの設定選択肢に関する計画作成情報の詳細については、以下のトピックを参照してください。

- 「NNMiで何を監視できますか?」(90ページ)
- 「グループの計画作成」(93ページ)
- 「ポーリング間隔の計画作成」(96ページ)
- 「どのデータを収集するかの決定」(96ページ)
- 「NNMiにどのSNMPトラップを送信するかの決定」(97ページ)

NNMiで何を監視できますか?

State Pollerサービスは、検出された各インタフェース、アドレス、および管理ドメインでアクティブに監視されるように指定されているSNMPエージェントを監視します。State Pollerサービスは、カード、シャーシ、ノードセンサー、物理センサー、ルーター冗長性グループなどを監視するようにも設定できます。

注: ほとんどの場合、インタフェースに接続されたポーリングによってのみ、十分に正確な根本原因分析ができます。監視対象インタフェースのセットを拡張すると、ポーリングのパフォーマンスに影響が及ぶ可能性があります。

NNMiがハイパーバイザーネットワーク環境を監視している場合は、以下のものを含むオブジェクトもさらに監視されます。

- ハイパーバイザー
- ハイパーバイザーでホストされている仮想マシン (VM)
- 仮想スイッチ
- アップリンク (インタフェースオブジェクトとして表される)

ヒント: 仮想マシンにVMware Toolsがインストールされていることを確認し、NNMiによって提供されている仮想マシンノードグループを使用して、仮想マシンに関連付けられているIPアドレスの障害ポーリングを有効にしてください。基盤となる仮想マシンが削除された場合やNNMiが管

理できないハイパーバイザーに移動された場合にも、NNMiがすべてのVMノードを特定できるようにするには、この方法を実践することをお勧めします。障害ポーリングを有効にする方法の詳細については、管理者用のNNMiヘルプの「監視のデフォルト設定」を参照してください。

ヒント: 仮想マシン (VM) に関連付けられているIPアドレスに対して障害ポーリングを有効にするには、NNMiが提供している**仮想マシン**ノードグループを使用してください。基盤となる仮想マシンが削除された場合やNNMiが管理できないハイパーバイザーに移動された場合にも、NNMiがすべてのVMノードを特定できるようにするには、この方法を実践することをお勧めします。詳細については、管理者用のNNMiヘルプの「監視のデフォルト設定」および「非応答ノードを削除するかどうかの設定」を参照してください。

モニタリングの詳細については、NNMiヘルプ**を参照してください。**

「モニタリングの拡張」(92ページ)も参照してください。

監視の停止

NNMi管理モードを使用して、デバイスまたはインタフェースを [管理対象外] または [サービス停止中] に設定できます。[管理対象外] は恒久的な状況と見なされます。オブジェクトのステータスを知る心配をする必要はありません。[サービス停止中] は一時的な状況と見なされます。1つ以上のオブジェクトがオフラインになり、停止中のインシデントが過剰になります。

すべてのグループ設定全体のオーバーレイとして、管理モードを考えてください。グループ、ポーリング間隔、種類に関係なく、オブジェクトのステータスが [管理対象外] または [サービス停止中] に設定されている場合、State Pollerはそのオブジェクトと通信しません。

ヒント:検出を行い、データベースに配置することを選択したデバイスやインタフェース (またはその両方) の中には、ポーリングの必要がないものもあります。[管理対象外] に恒久的に設定するオブジェクトに注意してください。1つ以上のノードグループを作成し、管理モードを簡単に設定することもできます。

監視されないノードへのインタフェース

直接管理していないデバイスに接続されているインタフェースのステータスを知る必要があることがあります。たとえば、アプリケーションまたはインターネットサーバーへの接続が確立されているかどうか知る必要があるものの、そのサーバーのメンテナンスは担当していないことがあります。検出ルールにそのサーバーを組み入れていないと、NNMiはそのサーバーに面するインタフェースを未接続と見なします。

監視されていないノードに接続する重要なインタフェースのステータスを監視する方法には次の2つがあります。

- 監視されていないノードの検出。

  監視されていないノードをNNMiトポロジに追加するとき、NNMiは、トポロジの残りの部分にノードを接続しているインタフェースを接続済みと見なします。この場合、NNMiは、監視設定に従ってこれらのインタフェースをポーリングできます。NNMiはノードを管理対象として検出します。NNMiに監視させたくない、管理されていないノード。

  注: 検出された各ノードは、NNMiが積極的にそのノードを管理しているかどうかに無関係に、ライセンスの最大数まで数えられます。

- 未接続インタフェースのポーリング

  未検出ノードの接続を備えたネットワークデバイスを含むノードグループを作成できます。次に、ノードグループの未接続インタフェースのポーリングを有効にします。

  NNMiは、多数のインタフェースのあるデバイスに大量のトラフィックを追加できる、ノードグループのデバイス上のインタフェースをすべてポーリングします。

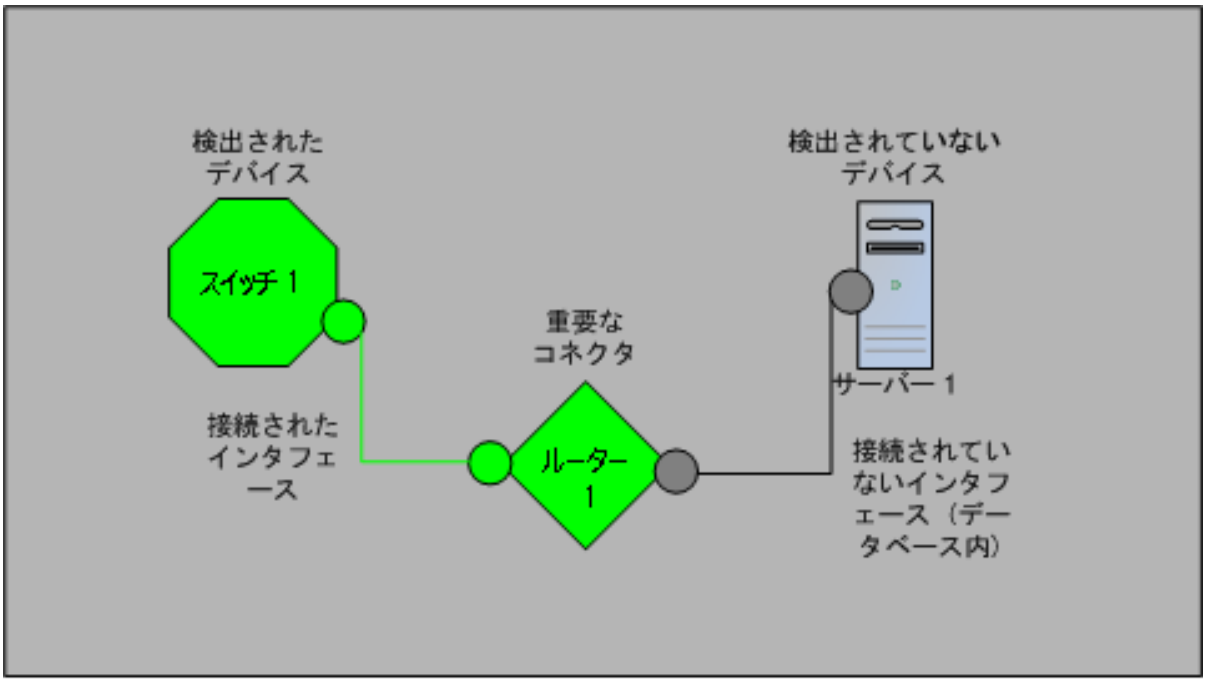
モニタリングの拡張

監視を拡張して、以下が含まれるようにできます。

- 未接続インタフェース。デフォルトでは、NNMiがモニタリングする未接続インタフェースは、IPアドレスがあり、かつ、**ルーター**ノードグループに含まれるもののみです。

  注: NNMiは、以下の図に示すように、NNMiが検出した別のデバイスに接続されていないインタフェースとして未接続インタフェースを定義します。

  未接続インタフェース例



- ルーターインタフェースのように、IPアドレスのあるインタフェース。
- SNMPをサポートしないデバイス用のICMPポーリング。デフォルトで、ICMPポーリングは、**非SNMPデバイス**ノードグループについて有効です。

グループの計画作成

ノードグループとインタフェースグループをセットアップしてから、監視を設定する必要があります。したがって、ノードグループとインタフェースグループを設定するときはポーリング要求について考慮する必要があります。重要なデバイスを頻繁に監視できるようにノードグループとインタフェースグループを設定するのが理想的です。クリティカルでないデバイスのチェックをあまり頻繁でないようにできます (そもそもチェックを行う場合です)。

> **ヒント**:ネットワーク監視を行うノードおよびインタフェースグループのセットを1つ設定します。マップにより、ネットワーク可視化用に異なるノードグループのセットを設定します。

これらのグループは、**[設定] > [ノードグループ]** または **[設定] > [インタフェースグループ]** ワークスペースを使用して定義します。デフォルトでは、インシデント、ノード、インタフェース、およびアドレスビューのフィルターに使用されるものと同じグループになります。モニタリング設定用にノードフィルターまたはインタフェースフィルターの別個のセットを作成するには、ノードグループまたはインタフェースグループを開き、[**ノードグループ**] フォームまたは [**インタフェースグループ**] フォームで [**ビューフィルターリストに追加**] チェックボックスをオンにします。[**保存して閉じる**] をクリックします。

[**モニタリングの設定**] フォームの [**ノードの設定**] タブと [**インタフェースの設定**] タブにあるノードグループまたはインタフェースグループのレベルで、ポーリングの種類とポーリングの間隔を設定します。

類似のポーリングのニーズごとに、インタフェースやデバイス (またはその両方) をグループにまとめる基準を決定します。計画作成に際して考慮すべきいくつかの要因は次のとおりです。

- ネットワークのどのエリアにこれらのデバイスがありますか?タイミング制限があるか?
- デバイスの種類ごとに収集したポーリング間隔またはデータを差別化しますか?インタフェースの種類ごとにか?
- NNMiには使用できる事前設定されたグループがあるか?

> **ヒント**:同時にサービス停止中になりそうなオブジェクトのグループ定義を、場所ごとであれ、他の何らかの基準ごとであれ、作成することができます。たとえば、IOSアップグレードを適用しながら、すべてのCiscoルーターを [サービス停止中] モードにできます。

インタフェースグループ

基準に基づいて、どのインタフェースグループを作成するか決定します。インタフェースグループが最初に評価されることを覚えておいてください (「状態ポーリングの概念」(88ページ)を参照)。イン

タフェースグループはノードグループメンバーシップを参照できるので、計画を実現するインタフェースグループの前に、ノードグループの設定を完了できます。

事前設定されたインタフェースグループ

NNMiには、使用できるようにすでに設定済みの便利なインタフェースグループがいくつかあります。たとえば、次のとおりです。

- ISDN接続に関連付けられたIFTypeのある全インタフェース
- 音声接続用のインタフェース
- ポイントツーポイント通信用のインタフェース
- ソフトウェアループバックインタフェース
- VLANインタフェース
- リンク集合プロトコルに関与するインタフェース

HPは、時間をかけてさらに多くのデフォルトのグループを追加し、設定作業を簡単にしていきます。既存のグループを使用または変更するか、または自分専用のグループを作成できます。

インタフェースグループには、ホストであるノードとIFTypeのノードグループメンバーシップ、またはインタフェース用のほかの属性の2つの種類の修飾子があります。これらは次のように組み合わせできます。

- ノードグループ内のノードの全インタフェースをIFTypeと無関係にグループにまとめます。IFTypeまたは属性 (名前、エイリアス、説明、速度、インデックス、アドレス、またはその他のIFType属性など) は選択しません。
- 特定のIFTypeまたは属性のセットのインタフェースは、それらインタフェースが存在するノードに無関係にすべてグループにまとめられます。
- 特定のノードグループに存在する特定のIFTypeまたは属性のインタフェースのみがグループにまとめられます。

ノードグループ

インタフェースグループの計画を作成してから、ノードグループの計画を作成します。監視用に作成された全ノードグループがフィルタービューに意味があるとは限らないので、ノードグループは独立に設定できます。

事前設定されたノードグループ

HPは、ノードグループのデフォルト集合を用意して、設定作業を簡単にしています。これらの基礎になっているのは、検出プロセスの間にシステムオブジェクトIDから導出されたデバイスカテゴリです。デフォルトのノードグループには以下が含まれます。

- ルーター
- ネットワーキングインフラストラクチャーデバイス (スイッチ、ルーターなど)
- Microsoft Windowsシステム
- SNMPコミュニティ文字列を持っていないデバイス

- 重要ノード。Causal Engineによって内部的に使用されており、コネクター障害の危険にさらされているデバイスの特殊処理を提供します。詳細については、NNMiヘルプの「定義済みビューフィルターとして使用されるノードグループ」を参照してください。
- 仮想マシン

HPは、時間をかけてさらに多くのデフォルトのグループを追加し、設定作業を簡単にしていきます。既存のグループを使用または変更するか、または自分専用のグループを作成できます。

次のノード属性を使用して、関連するノードの定義に条件を付けることができます。

- ノード上のIPアドレス
- ホスト名ワイルドカード規約
- デバイスのプロファイルル派生物。たとえば、カテゴリ、ベンダー、ファミリ
- MIB II sysName、sysContact、sysLocation

ヒント: 簡単で再使用可能な極小のグループを作成し、監視または視覚化のためにこれらを結合して階層クラスターにすることができます。グループ定義は重なることがあります。たとえば、「すべてのルーター」と「IPアドレスの末尾が100のすべてのシステム」です。ノードは複数のグループに属することができると考えられます。

バランスを取るためには、使われない余分なエントリのリストで負担を大きくしないように、設定および表示用に豊富なグループのセットを作成します。

デバイスのプロファイルルとの相互作用

各デバイスが検出されると、NNMiはシステムオブジェクトIDを使用して、使用可能なデバイスのプロファイルルのリストにインデックスを作成します。デバイスプロパティは、ベンダー、製品、ファミリ、デバイスカテゴリなど、デバイスの追加属性を導出するために使用されます。

ノードグループを設定するとき、これら導出された属性を使用して、監視設定に適用するデバイスをカテゴリにまとめられます。たとえば、ベンダーを問わず、ネットワーク全体のすべてのスイッチを特定のポーリング間隔でポーリングすることもできます。デバイスカテゴリ「スイッチ」を自分のノードグループの定義特性として使えます。システムオブジェクトIDがカテゴリ「スイッチ」にマップされる、検出されたデバイスはすべて、ノードグループについての設定を受け取ります。

ヒント: NNMiがハイパーバイザーネットワーク環境を管理している場合は、仮想マシン (VM) だけが含まれるノードグループを作成できます。これらのノードは、`vmwareVM`デバイスプロファイルを使用して識別できます。このノードグループを使用すると、ハイパーバイザーでホストされなくなったVMがないかをチェックすることもできます。このノードグループを選択した後、`Hosted On = null`でフィルターし、これらのVMを特定します。このノードグループを使用して、VMに関連付けられているIPアドレスの障害ポーリングを有効にすることもできます。これは、関連付けられたハイパーバイザーが削除されている場合でもVMを継続的に監視できるようにするベストプラクティスでもあります。

ポーリング間隔の計画作成

オブジェクトグループごとに、NNMiがデータを収集するのに使うポーリング間隔を選択します。サービスレベル契約条項に最も適切に一致するように、間隔は1分間と短くすることもできますし、数日間と長くすることもできます。

ヒント:間隔が短いと、可能な限り迅速にネットワーク問題を認識するのに役立ちます。しかし、あまりに短い間隔であまりに多くのオブジェクトをポーリングすると、State Pollerにバックログを発生させる可能性があります。各自の環境について、リソース利用と間隔の間で最良のバランスを見つけてください。

注: Causal Engineは24時間ごとに各ノードのステータスのポーリングを実行し、必要に応じてステータス、結果、およびインシデント情報を更新します。ステータスのポーリングは、デバイスに設定されたポーリング間隔のタイミングには影響しません。

どのデータを収集するかの決定

State Pollerサービスは、ポーリングを使って、ネットワークで監視されているデバイスに関する状態情報を収集します。ポーリングはICMPやSNMP (またはその両方) を使用して実行できます。

ICMP (ping)

ICMPアドレス監視は、ping要求を使って、管理対象の各IPアドレスの使用可能性を確認します。

SNMPポーリング

SNMP監視は、監視されている各SNMPエージェントがSNMPクエリーに応答していることを確認します。

- State Pollerは、間隔ごとに1つのクエリーで、監視されている各オブジェクトから設定済みSNMP情報を収集するよう、高度に最適化されています。設定の変更を保存すると、State Pollerは、各オブジェクトのグループメンバーシップを再計算し、収集する設定済み間隔とデータセットに再適用します。
- SNMP監視は、監視されているすべてのインタフェースとコンポーネントにSNMPクエリーを発行し、MIB IIインタフェーステーブル、HostResources MIB、およびベンダー特有のMIBから現在の値を要求します。障害監視に使われる値もあります。NNM iSPI Performance for Metricsをインストールしてある場合は、パフォーマンス測定に使われる値もあります。

Webポーリング

(SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコルを使用できます。たとえば、VMware環境用のSOAPプロトコルなどです。

SNMPコンポーネント稼働状態データ

コンポーネントヘルス監視をグローバルなレベルで有効または無効にできます。障害に関するコンポーネント稼働監視は、デバイスの障害ポーリング間隔設定に従います。

ポーリングごとに追加データを収集しても、ポーリングを実行する時刻への影響はありません。しかし、各オブジェクトについて保存された追加データによって、State Poller用にメモリ要求が増加する可能性があります。

注: パフォーマンス監視設定はNNM iSPI Performance for Metricsでのみ使用されます。パフォーマンスに関するコンポーネント稼働監視は、デバイスのパフォーマンスポーリング間隔設定に従います。

ヒント: 監視設定変更をバッチ処理すると、State Pollerの進行中の操作が混乱することは少なくなります。

NNMiにどのSNMPトラップを送信するかの決定

NNMiは、SNMPトラップを受信したとき、次のポーリング間隔を待つのではなく、デバイスのポーリングに以下のSNMPトラップを使用します。

- CempMemBufferNotify
- CiscoColdStart
- CiscoEnvMonFanNotification
- CiscoEnvMonFanStatusChangeNotif
- CiscoEnvMonRedundantSupplyNotification
- CiscoEnvMonSuppStatusChangeNotif
- CiscoEnvMonTemperatureNotification
- CiscoEnvMonTempStatusChangeNotif
- CiscoEnvMonVoltageNotification
- CiscoEnvMonVoltStatusChangeNotif
- CiscoFRUInserted
- CiscoFRURemoved
- CiscoLinkDown
- CiscoLinkUp
- CiscoModuleDown
- CiscoModuleUp
- CiscoModuleStatusChange
- CiscoRFProgressionNotif
- CiscoRFSwactNotif
- CiscoWarmStart
- HSRPStateChange
- IetfVrrpStateChange
- Rc2kTemperature

- RcAggLinkDown
- RcAggLinkUp
- RcChasFanDown
- RcChasFanUp
- RcChasPowerSupplyDown
- RcChasPowerSupplyUp
- Rcn2kTemperature
- RcnAggLinkDown
- RcnAggLinkUp
- RcnChasFanDown
- RcnChasFanUp
- RcnChasPowerSupplyDown
- RcnChasPowerSupplyUp
- RcnSmltIstLinkDown
- RcnSmltIstLinkUp
- RcSmltIstLinkUp
- RcVrrpStateChange
- SNMPColdStart
- SNMPLinkDown
- SNMPLinkUp
- SNMPWarmStart

トラップを受信したときにNNMiにデバイスをポーリングさせるには、これらのトラップをNNMiに送信するようにネットワークデバイスを設定します。

ヒント: SNMPトラップインシデント設定の詳細については、NNMiコンソールから、[設定] ワークスペースに移動し、[**インシデント**] > [**SNMPトラップの設定**] の順に選択します。

「SNMPトラップからの検出ヒントの使用」(77ページ)も参照してください。

状態ポーリングの設定

このセクションでは、設定のヒントを示し、設定例をいくつか挙げます。このセクションを読んだ後、特定の手順については、NNMiヘルプの「モニタリング動作の設定」を参照してください。

注: 大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。

インタフェースグループとノードグループの設定

[設定] ワークスペースでインタフェースグループとノードグループを作成します。詳細については、NNMiヘルプの「ノードまたはインタフェースのグループを作成する」を参照してください。

例

たとえば、ProximiTプロキシサーバー用にノードグループを設定する方法は次のとおりです。

1. **[設定] > [ノードグループ]** を開き、✱ **[新規作成]** をクリックします。
2. グループ**Proxy Servers**という名前を挙げ、[**ビューフィルターリストに追加**] をオンにします。
3. [**追加のフィルター**] タブで、**hostname**属性を選択し、=(等しい)演算子を選択します。
4. 値は、`prox*.example.com`のようにワイルドカードを入力します。

   ProximiTデバイスについてDevice Profile (デバイスのプロファイルル) とCategory (カテゴリ) を設定してある場合は、[**デバイスフィルター**] タブを使って [**デバイスカテゴリ**] セレクターにアクセスし、作成したProxy Serverカテゴリをグループの基礎にすることができます。
5. グループ定義で [**保存して閉じる**] をクリックします。

注: ノードグループを設定してから、インタフェースグループ設定でノードグループを参照する必要があります。

インタフェースのモニタリングの設定

State Pollerは、ノードグループの前に、インタフェースグループメンバーシップを分析します。作成した各インタフェースグループ、および使用する既存のインタフェースグループごとに、[**モニタリングの設定**] ダイアログと [**インタフェースの設定**] タブを開き、State Pollerがそのグループを処理する方法に関する指示のカスタムセットを作成します。指示には以下のものが含まれます。

- 障害モニタリングの有効化または無効化
- 障害ポーリング間隔の設定
- NNM iSPI Performance for Metricsがある場合、パフォーマンスポーリングの有効化または無効化
- NNM iSPI Performance for Metricsがある場合、パフォーマンスポーリング間隔の設定
- NNM iSPI Performance for Metricsがある場合、パフォーマンス管理しきい値の設定
- NNMiがグループ内の未接続インタフェース (またはIPアドレスをホストしている未接続インタフェース) を監視するかどうかの選択

インタフェースグループごとに異なる設定ができます。State Pollerは、小さい順序番号から大きい順序番号へとリストを評価することを覚えておいてください。

ヒント: 複数のグループにあてはまるオブジェクトは最も順序番号の小さいグループから設定を適用されることを頭に入れておきつつ、順序番号をダブルチェックします。

ノードのモニタリングの設定

あるオブジェクトが設定済みのインタフェースグループにあてはまらない場合、State Pollerはノードグループ内のメンバーシップについて、そのオブジェクトを評価します。最も小さい順序番号から最も高い順序番号へと、設定は最初の合致するノードグループに適用されます。

ノードグループごとに、[**モニタリングの設定**] フォームを開いてから [**ノードの設定**] タブを開きます。State Pollerがグループを処理する方法に関する指示のカスタムセットを作成します。指示には以下のものを入れられます。

- 障害モニタリングの有効化または無効化
- 障害ポーリング間隔の設定
- NNM iSPI Performance for Metricsがある場合、パフォーマンスポーリングの有効化または無効化
- NNM iSPI Performance for Metricsがある場合、パフォーマンスポーリング間隔の設定
- NNM iSPI Performance for Metricsがある場合、パフォーマンス管理しきい値の設定
- NNMiがグループ内の未接続インタフェース (またはIPアドレスをホストしている未接続インタフェース) を監視するかどうかの選択

ノードグループごとに異なる設定ができます。

ヒント: 複数のグループにあてはまるオブジェクトは最も順序番号の小さいグループから設定を適用されることを頭に入れておきつつ、順序番号をダブルチェックします。

デフォルト設定の確認

State Pollerは、定義済みのインタフェース設定またはノードの設定に合致しないオブジェクトについて [**デフォルト設定**] タブの設定を適用します。このタブの設定を検討し、デフォルトレベルで自分の環境に合致することを確認します。たとえば、デフォルト設定としてすべての未接続インタフェースをポーリングすることはほとんどありません。

注: 変更を実現するためには、コンソールに戻り、すべての [**設定の監視**] ダイアログボックスを必ず [**保存して閉じる**] ようにしてください。

状態ポーリングの評価

このセクションでは、監視設定の進行と成功を評価する方法をリストします。

ネットワークモニタリングの設定を確認します。

NNMiが指定のノードまたはインタフェースの監視に使う設定を決定すると、ステータスのポーリングをいつでも開始できます。

ネットワークモニタリングの設定を確認するには、以下のチェックを使用します。

インタフェースまたはノードは正しいグループのメンバーでしょうか?

あるグループにどのインタフェースまたはノードが属するか確認するには、**[設定]** ワークスペースで次の1つを選択します。

- ノードグループ
- インタフェースグループ

ヘルプの指示に従って、グループのメンバーを表示します。オブジェクトは複数のグループのメンバーになれること、他のグループの順序番号の方が小さい可能性があることを念頭に置いてください。

その代わりに、オブジェクト (インタフェースまたはノード) を開き [**ノードグループ**] タブまたは [**インタフェースグループ**] タブをクリックして、オブジェクトが属するグループの完全なリストを表示することもできます。このリストは、グループ名ごとにアルファベット順であって、どの設定が適用されるかを決定する順序番号を反映していません。

オブジェクトがグループのメンバーでない場合は次のとおりです。

1. インベトリビューのデバイスのプロファイルルを取得します。
2. **[設定] > [デバイスのプロファイル]** 下にあるデバイスのプロファイルに関する属性マップを確認します。
3. ノードグループ定義の属性要件を確認します。

不一致がある場合は、[デバイスのプロファイル] に由来するカテゴリを調整して、その種類のデバイスが自分のノードグループにあてはまるようにできます。**[アクション] > [設定のポーリング]** を実行して、ノードがあてはまるようにノードの属性を更新する必要がある場合もあります。

どの設定が適用されていますか?

特定のノード、インタフェース、またはアドレスに有効な監視設定をチェックするには、該当するインベントリビュー内でそのオブジェクトを選択し、**[アクション] > [モニタリングの設定]** を選択します。NNMiに現在の監視設定が表示されます。

[**有効化された障害ポーリング**] と [**障害ポーリング間隔**] の値を調査します。これらの値が予想どおりでない場合は、[**ノードグループ**] または [**インタフェースグループ**] の値を見て、どの順序付けられたグループ一致が適用されるか調べます。

オブジェクト用にトラフィックが無効にされていないことを確認するために、オブジェクトの [**アクション**] > [**通信の設定**] をチェックする必要がある場合もあります。

どのデータが収集されていますか?

特定のデバイスのステータスのポーリングを開始し、予想された種類のポーリング (SNMP、ICMP) がそのデバイスについて実行されていることを確認できます。

注: (SNMPエージェントに加えて) Webエージェントが設定されている場合、NNMiは追加のプロトコルを使用できます。たとえば、VMware環境用のSOAPプロトコルなどです。

ノードを選択し、**[アクション] > [ポーリング] > [ステータスのポーリング]** をクリックします。

NNMiはデバイスのリアルタイムのステータスチェックを実行します。実行中のポーリングの種類と結果は出力に表示されます。

ポーリングの種類が予想したものでない場合は、ノードの監視設定、および監視設定のそれぞれのグローバル、インタフェース、またはノードに関する設定をチェックします。

ステータスのポーリングのパフォーマンスの評価

自分の環境のステータスのポーリングのパフォーマンスを評価するには、State Poller稼働状態チェックの情報を使って、State Pollerサービスの動作を数値で表し、評価します。

State Poller稼働状態情報は、Status Pollerがポーリング要求に応じることができるかどうかを示します。

State Pollerは最新の状態に付いていっていますか?

[システム情報] ウィンドウの **[StatePoller]** タブで、以下の表にの説明に従ってStatePollerサービスの現在の稼働状態統計をいつでもチェックできます。

StatePollerの稼働状態情報

情報	説明
ステータス	State Pollerサービスの全般的なステータス
ポーリングカウンター	• 要求された収集 • 完了した収集 • 進行中の収集 • 収集要求の遅延
最後の1分にスキップを実行する時刻	設定済みのポーリング間隔内で完了しなかった、定期的にスケジュールされたポーリングの数。値がゼロでない場合は、ポーリングエンジンが最新の状態に付いていっていないか、またはターゲットが応答より速くポーリングされています。 • 監視の要点:この値が増加し続ける場合は、ターゲットとの通信に問題があるか、またはNNMiの負荷が過剰です。

StatePollerの稼働状態情報 (続き)

情報	説明

	• 実行すべきアクション:nnm?.0.logファイルで文字列 com.hp.ov.nms.statepollerで始まるクラスのメッセージを探して、スキップされたポーリングのターゲットを特定します。
	  • スキップされたポーリングのターゲットが同じ場合、設定を変更してこれらのターゲットのポーリング頻度を低くするか、タイムアウトを増やします。
	  • スキップされたポーリングのターゲットが異なる場合、NNMiのシステムパフォーマンス (特にovjbossの使用可能メモリ) を確認します。

過去1分以内の古い収集	古い収集というのは、少なくとも10分間、ポーリングエンジンから応答を受信していない収集のことです。稼働状態が良好なシステムでは古い収集はありません。
	• 監視の要点:この値が一定して増加する場合は、ポーリングエンジンに問題があります。
	• 実行すべきアクション:nnm?.0.logファイルで文字列 com.hp.ov.nms.statepollerで始まるクラスのメッセージを探して、古い収集のターゲットを特定します。
	  • 古い収集のターゲットが1つの場合、この問題を解決できるまでターゲットを管理から除外します。
	  • 古い収集のターゲットが異なる場合、NNMiシステムとNNMiデータベースのパフォーマンスを確認します。NNMiを停止して再起動します。

ポーリング結果のキューの長さ	• 監視の要点:この値はほとんどの時間0に近いはずです。 • 実行すべきアクション:キューのサイズがきわめて大きい場合、ovjbossはメモリが不足するという問題が発生することがあります。
状態マッパーキュー期間	• 監視の要点:この値はほとんどの時間0に近いはずです。 • 実行すべきアクション:このキュー期間がきわめて大きい場合は、NNMiシステムとNNMiデータベースのパフォーマンスをチェックします。
状態アップデータキュー期間	• 監視の要点:この値はほとんどの時間0に近いはずです。 • 実行すべきアクション:このキューのサイズがきわめて大きい場合は、NNMiシステムとNNMiデータベースのパフォーマンスをチェックします。
状態アップデータ例外	監視の要点:この値は0になるはずです。

状態ポーリングの調整

状態ポーリングのパフォーマンスは次の重要な変数の影響を受けます。

- ポーリングされるデバイス/インタフェースの数
- 設定されるポーリングの種類
- 各デバイスのポーリングの頻度

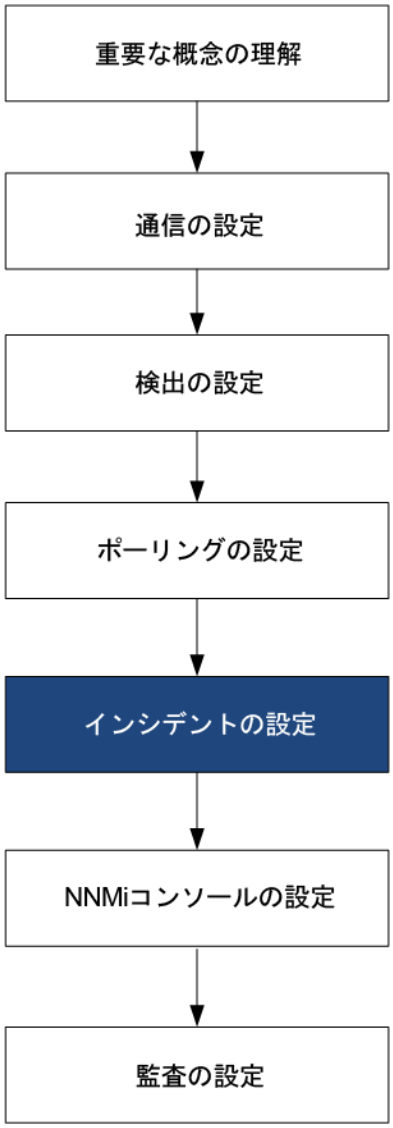
これらの変数は、ネットワーク管理のニーズによって促進されます。ステータスのポーリングについてパフォーマンス上の問題がある場合は、次の設定を考慮してください。

- 個別のノードのポーリング設定はノードグループとインタフェースグループ内のメンバーシップによって制御されるので、類似のポーリング要求のあるノードまたはインタフェースがグループに含まれていることを確認します。
- 未接続インタフェースまたはIPアドレスをホストするインタフェースをポーリングしている場合は、設定をチェックして、必要なインタフェースのみをポーリングしていることを確認します。[**ノードの設定**] フォームまたは [**インタフェースの設定**] フォーム ([**モニタリングの設定**] フォームでグローバルにではなく) でこれらのポーリングを有効にし、最も特定な制御を維持し、ポーリングするインタフェースの最も小さいサブセットを選択します。
- 未接続インタフェースのポーリングでは、未接続のすべてのインタフェースが監視されることを覚えておいてください。IPアドレスのある未接続のインタフェースのみを監視するには、IPアドレスをホストするインタフェースのポーリングを有効にします。

監視設定とは無関係に、ステータスのポーリングは、ネットワーク応答性に左右され、全般的なシステムパフォーマンスの影響を受ける可能性があります。デフォルトのポーリング間隔のあるステータスのポーリングは多くのネットワーク負荷をかけませんが、サーバーとポーリングされているデバイスの間のネットワークリンクのパフォーマンスが低い場合、ステータスのポーリングのパフォーマンスも低くなる可能性があります。タイムアウトを大きく、再試行の数を小さく設定すると、ネットワーク負荷を低減できますが、これらの設定変更でできるのはそれだけです。タイミングの良いポーリングを行うには、適切なネットワークパフォーマンスと十分なシステムリソース (CPU、メモリ) が必要です。

コンポーネント稼働状態監視を有効または無効にしても、ポーリングのタイミングには影響がありません。スケジュールされた時刻に、追加のMIBオブジェクトが収集されるだけです。ただし、コンポーネントヘルス監視を無効にすると、State Pollerが使用するメモリの量が減少する可能性があります。

NNMiインシデント



HP Network Node Manager i Software (NNMi) には、NNMiコンソールに作業可能インシデント数を提供する受信SNMPトラップをフィルタリングする多数のデフォルトインシデントと相関処理が用意されています。この章では、NNMiインシデントを設定することでネットワーク管理を微調整するのに役立つ情報を説明します。この章は、NNMiヘルプの情報を補充するものです。NNMiインシデントの概要およびインシデント設定方法の詳細については、NNMiヘルプの [インシデントを設定する] を参照してください。

この章には、以下のトピックがあります。

- 「インシデント設定のバッチロード」(119ページ)
- 「インシデントの評価」(121ページ)
- 「インシデントの調整」(121ページ)

インシデントの概念

NNMiでは、以下のソースからネットワークステータス情報が収集されます。

- NNMi Causal Engineではネットワークの稼動状態が分析され、継続的に各デバイスの稼働状態ステータス値が提供されます。Causal Engineでは、可能な場合は常にネットワーク障害の根本原因も広範囲に評価され、決定されます。
- ネットワークデバイスからのSNMPトラップ。NNMiのCausal Engineは、分析中にトラップを症状に関する情報として使用します。
- HP ArcSight Logger統合からのsyslogメッセージ。

NNMiは、この情報をネットワーク管理に有用な情報を提供するこのネットワークステータス情報に変換します。NNMiには、ネットワークオペレーターが考慮する必要があるインシデント数を減らす多くのデフォルトインシデント相関処理が用意されています。デフォルトのインシデント相関処理をカスタマイズして、環境のネットワーク管理要件に一致する新規インシデント相関処理を作成することができます。

NNMiコンソールのインシデント設定によって、NNMiが作成できるインシデントタイプが定義されます。インシデント設定が、受信したSNMPトラップsyslogメッセージと一致しない場合、その情報は廃棄されます。ソースオブジェクトの管理モードが、NNMiデータベースで [管理対象外] または [サービス停止中] に設定されている場合、またはデバイスの障害ポーリングがモニタリングされていない場合、NNMiでは常に受信トラップが廃棄されます。

ヒント: `nnmtrapconfig.ovpl -dumpBlockList`は、インシデント設定がないか、または無効なためインシデントパイプラインに渡されなかったSNMPトラップなど、現在のインシデント設定に関する情報を出力します。

さらに、NNMiではNNMiトポロジにないネットワークデバイスからのSNMPトラップは廃棄されます。このデフォルト動作の変更の詳細については、NNMiヘルプの「未解決の受信トラップを処理する」を参照してください。

詳細については、以下を参照してください。

- NNMiヘルプの「イベントパイプラインについて」
- NNMiヘルプの「NNMiのCausal Engineとインシデント」
- 『HP Network Node Manager i-series Software因果関係分析ホワイトペーパー』は以下のURLから入手できます。

  http://h20230.www2.hp.com/selfsolve/manuals

インシデントライフサイクル

以下の表に、インシデントのライフサイクルの段階を示します。

NNMiインシデントライフサイクル

ライフサイクル状態	説明	状態設定者	インシデント使用者
なし	NNMiイベントパイプラインはすべてのソースから入力を受領し、必要に応じてインシデントを作成します。	該当なし	• NNMi
抑止済み	インシデントは保管場所にあり、別のインシデントとの相関処理待ちです。インシデントビューアーのインシデントを減らすために、この待機期間があります。 ダンプニング周期はインシデントタイプによって異なります。詳細については、「インシデントの抑制、強化、およびダンプニング」(113ページ)を参照してください。	NNMi	• NNMi
登録済み	インシデントは、インシデントビューで見ることができます。 インシデントは任意の設定済み宛先へ転送されます (近隣またはグローバルマネージャー)。	NNMi ユーザーはインシデントビューでこの状態を設定することもできます。	• ユーザー • ライフサイクル移行アクション • インシデントを転送する統合
進行中	インシデントは問題を調査するいずれかのユーザーに割り当てられています。 ネットワーク管理者によってこの状態の特定の意味が定義されます。	ユーザー	• ユーザー • ライフサイクル移行アクション • インシデントを転送する統合
完了	インシデントによって指定された問題の統合は完了し、ソリューションが配置されています。 インシデントが識別する問題 ネットワーク管理者によってこの状態の特定の意味が定義されます。	ユーザー	• ユーザー • ライフサイクル移行アクション • インシデントを転送する統合
解決済み	このインシデントによってレポートされた問題	ユーザーまたはNNMi	• ユーザー

NNMiインシデントライフサイクル(続き)

ライフサイクル状態	説明	状態設定者	インシデント使用者
	が解決したことをNNMiが確認したことを示します。たとえば、デバイスからインタフェースを取り外すと、そのインタフェースに関するインシデントはすべて自動的に「解決済み」になります。		• ライフサイクル移行アクション • インシデントを転送する統合

トラップおよびインシデント転送

以下の表は、トラップおよびインシデントをNNMi管理サーバーから別の宛先へ転送する方法を要約したものです。テーブルの補足テキストによって、NNMiのSNMPトラップ転送メカニズムとNNMiのノースバウンドインタフェースSNMPトラップ転送メカニズムが比較できます。

トラップおよびNNMiインシデント転送でサポートされている方法

	NNMiトラップ転送	NNMiNorthboundインタフェーストラップ転送	グローバルネットワーク管理のトラップ転送
転送対象	• ネットワークデバイスからのSNMPトラップ • HP ArcSight Loggerからのsyslogメッセージ	• ネットワークデバイスからのSNMPトラップ • NNMi管理イベント • HP ArcSight Loggerからのsyslogメッセージ	• ネットワークデバイスからのSNMPトラップ • HP ArcSight Loggerからのsyslogメッセージ
転送フォーマット	受信したままのSNMPv1、v2c、またはv3トラップ (SNMPv3トラップはSNMPv2cトラップへ変換可能)	NNMiインシデントから作成されたSNMPv2cトラップ	NNMiインシデント
追加情報	ほとんどの場合、NNMiはvarbindを追加して元のソースオブジェクトを識別します。 NNMiがSNMPv1トラップを変更することはありません。	NNMiはvarbindを追加して元のソースオブジェクトを識別します。	リージョナルマネージャープロセスによってインシデントに追加された情報はすべて、転送済みインシデントに保持されます。
設定先	[設定] ワークスペースの	[統合モジュールの設定]	[SNMPトラップの設定]

トラップおよびNNMiインシデント転送でサポートされている方法 (続き)

	NNMiトラップ転送	**NNMiNorthboundインタフェーストラップ転送**	**グローバルネットワーク管理のトラップ転送**
	[トラップ転送の設定]	ワークスペースの **[HPOM]**、**[Northboundインタフェース]**、または **[Netcool]**	フォームまたはsyslog設定の **[グローバルマネージャーへの転送]** タブ
注		NNMiには、NNMi Northboundインタフェース上にいくつかの統合が構築されています。 『HP Network Node Manager i Software—IBM Tivoli Netcool/ OMNIbus統合ガイド』および『HP Network Node Manager i Software—HP Operations Manager統合ガイド』も参照してください。 .	グローバルマネージャーのインシデントビューに表示されるリモートインシデントを転送します。転送済みインシデントはグローバルマネージャー上での相関処理に参加します。
詳細情報	NNMiヘルプにトラップ転送を設定する	NNMiデプロイメントリファレンスの「NNMi Northboundインタフェース」の章を参照してください。	• NNMiヘルプのSNMPトラップインシデントのグローバルマネージャー設定への転送設定

比較:サードパーティSNMPトラップを別のアプリケーションに転送する

NNMiが管理デバイスから受信するSNMPトラップを別のアプリケーションに転送する場合は、以下のいずれかの方法を使用します。

- NNMi SNMPトラップ転送メカニズムを使用。NNMi SNMPトラップ転送の設定方法の詳細については、NNMiヘルプの「トラップ転送設定」を参照してください。
- NNMiノースバウンドインタフェースSNMPトラップ転送メカニズムを使用。受信したSNMPトラップを転送するNNMi Northboundインタフェースの設定の詳細については、『NNMi統合リファレンス』の「NNMi Northboundインタフェース」の章を参照してください。

受信側アプリケーションがトラップを識別する方法は、SNMPトラップ転送メカニズムでは以下のように異なります。

- Windows (すべて) およびLinux (元のトラップ転送なし)

  この説明は、デフォルトおよびSNMPv3からSNMPv2cへの変換転送オプションに該当します。

  Windows NNMi管理サーバー上のNNMi SNMPトラップ転送メカニズムにより、トラップ転送先へ転送する前に各SNMPトラップが収集されます。トラップはNNMi管理サーバーからのものと考えられます。(この情報は、[**トラップ転送先**] フォームで元のトラップ転送オプションが選択されていないLinux NNMi管理サーバーにも適用されます。)

  受信側アプリケーションのトラップ送信デバイスとイベント間の関連付けを正しくするため、これらのトラップのルールを収集したvarbindに対してカスタマイズする必要があります。originIPAddress (.1.3.6.1.4.1.11.2.17.2.19.1.1.3) varbindからoriginIPAddressの値は汎用タイプInetAddressのバイト文字列で、originIPAddressType (.1.3.6.1.4.1.11.2.17.2.19.1.1.2) varbindの値によって決まるInetAddressIPv4またはInetAddressIPv6です。ルールによってoriginIPAddressType varbindを読み取って、originIPAddress varbindのインターネットアドレスタイプ (ipv4(1)、ipv6(2)) の値を決定する必要があります。ルールによってoriginIPAddressの値を表示文字列に変換する必要もあります。

  NNMiが転送されたトラップに追加するvarbindの詳細については、NNMiヘルプ、RFC 2851、および以下のファイルの「NNMiが提供するトラップvarbind」を参照してください。

  - Windowsの場合: `%NNM_SNMP_MIBS\Vendor\Hewlett-Packard\hp-nnmi.mib`
  - Linuxの場合: `$NNM_SNMP_MIBS/Vendor/Hewlett-Packard/hp-nnmi.mib`

- 元のトラップ転送が搭載されたLinux

  Linux NNMi管理サーバーにより、NNMiが受信するものと同じ形式でトラップを転送できます。各トラップは管理対象デバイスがトラップ転送先に直接送信したように表示されるため、受信側アプリケーションに設定された既存のトラップ処理は変更なしで動作する必要があります。

  詳細については、NNMiヘルプの 「トラップ転送先フォーム」の元のトラップ転送オプションを参照してください。

- NNMiノースバウンドインタフェース (全オペレーティングシステム)

  NNMi Northboundインタフェースは各SNMPトラップを強化してから、トラップ転送先に転送します。トラップはNNMi管理サーバーからのものと考えられます。受信側アプリケーションのトラップ送信デバイスとイベント間の関連付けを正しくするため、これらのトラップのルールを収集したvarbindに対してカスタマイズする必要があります。IncidentNodeHostname (1.3.6.1.4.1.11.2.17.19.2.2.21) およびIncidentNodeMgmtAddr (1.3.6.1.4.1.11.2.17.19.2.2.24) varbindによって元のソースオブジェクトが識別されます。

MIB

NNMiでは、以下の管理情報ベース (MIB) ファイルをNNMiデータベースにロードする必要があります。

- Custom Poller機能、折れ線グラフ、またはその両方のMIB式で使用するすべてのMIB変数
- NNMiが稼働状態をモニタリングするセンサー (ファン、または電源など)

- (NNM iSPI Performance for Metrics) しきい値監視で使用するすべてのMIB変数

NNMiでは、以下の管理情報ベース (MIB) ファイル、またはMIBファイルで定義されているトラップをNNMiデータベースにロードする必要があります。

- ノースバウンド宛先に転送するすべてのSNMPトラップ
- (NNM iSPI NET) トラップ分析レポートからアクセスするすべてのMIB変数

ヒント: NNMiには、現在サポートされていないMIBがリストされたREADME.txtファイルがあります。README.txtファイルは以下のディレクトリに保存されています。

- **Windowsの場合:** %NnmInstallDir%\misc\nnm\snmp-mibs
- **Linuxの場合:** $NnmInstallDir/misc/nnm/snmp-mibs

カスタムインシデント属性

NNMiでは、カスタムインシデント属性 (CIA) を使用して、インシデントに追加情報が追加されます。

- SNMPトラップインシデントの場合、NNMiでは元のトラップvarbindはインシデントのCIAとして格納されます。
- 管理イベントインシデントの場合、NNMiでは関連情報 (com.hp.ov.nms.apa.symptomなど) はインシデントのCIAとして追加されます。

インシデントCIAを使用すると、インシデントライフサイクル移行アクション、抑制、重複解除、強化などの範囲を絞り込むことができます。CIAを使用して、インシデントビューまたはフォームのアプリケーションメニュー項目の信頼性を絞り込むこともできます。

指定のインシデントにNNMiがどのCIAを追加するかを決定するには、インシデントビューのサンプルインシデントを開き、[カスタム属性] タブの情報を確認します。

解決済み管理イベントインシデントに追加されるCIA

管理イベントインシデントの原因となった状態が該当しなくなったとNNMi Causal Engineが判断すると、NNMiはそのインシデントのライフサイクル状態を [解決済み] に設定し、以下のテーブルにリストされているCIAをインシデントに追加します。NNMiコンソールユーザーは、[**インシデント**] フォームの [**相関処理の注**] フィールドでこの情報を確認できます。ライフサイクル移行アクションでは、CIAの値が直接使用されることがあります。

解決済みインシデントのカスタムインシデント属性

名前	説明
cia.reasonClosed	NNMiがインシデントをキャンセルしたか解決済みにした理由。この理由は、NodeUpやInterfaceUpなど、結果の名前にもなります。 このフィールドが設定されていない場合は、NNMiコンソールユーザーがインシデントを解決済みにしたということになります。

解決済みインシデントのカスタムインシデント属性 (続き)

名前	説明
	cia.reasonClosed CIAのNNMiの期待値を判断するには、NNMiヘルプの「NNMiによるインシデントの解決方法」を参照してください。
cia.incidentDurationMs	機能停止の時間 (ミリ秒単位)。ステータスが停止中になってから動作中に戻るまで、NNMiが測定します。この値は、cia.timeIncidentDetectedMsとcia.timeIncidentResolvedMsのCIAの差です。停止中インシデントと動作中インシデントのタイムスタンプを比較するより正確な測定値です。
cia.timeIncidentDetectedMs	NNMi Causal Engineが最初に問題を検出したときのタイムスタンプ (ミリ秒単位)。
cia.timeIncidentResolvedMs	問題が解決したことをNNMi Causal Engineが検出したときのタイムスタンプ (ミリ秒単位)。

NNMiは、多くの一次的根本原因インシデントと二次的根本原因インシデントに、前述の表にリストされたCIAを追加します。たとえばNodeDownインシデントには、InterfaceDownインシデントとAddressDownインシデントが二次的根本原因として含まれることがあります。NNMiがNodeDownインシデントを解決済みにすると、NNMiは二次的インシデントも解決済みにして、それぞれのインシデントのコンテキストの値を含むCIAを二次的インシデントに追加します。

NNMiは、以下のデフォルト管理イベントインシデントタイプに、前述の表にリストされたCIAを追加しません。

- NNMiコンソールユーザーが手動で解決済みにしたインシデント
- NNMiデータベースから削除されたオブジェクトに応答してNNMiが解決済みにしたインシデント
- IslandGroupDownインシデント
- NnmClusterFailover、NnmClusterLostStandby、NnmClusterStartup、NnmClusterTransferの各インシデント
- 以下のファミリのインシデント
  - 相関処理
  - ライセンス
  - NNMiヘルス
  - トラップ分析

インシデント数の削減

NNMiには、ネットワークオペレーターがNNMiコンソールで見るインシデント数を削減する以下のカスタマイズ可能相関処理が用意されています。

- Pairwise相関処理—あるインシデントが別のインシデントによってキャンセルされます。
- 重複解除相関処理—指定した時間ウィンドウ内に複数のインシデントのコピーを受信すると、重複解除インシデントの重複が相関処理されます。新たに受信した各重複インシデントの時間ウィンドが再開始されます。このように、NNMiでは相関処理時間ウィンドウの全期間中、重複を受信しなくなるまで重複インシデントが相関処理されます。
- レート相関処理—指定時間帯内にインシデントに関する指定コピー数を受信すると、レートインシデントの重複が相関処理されます。時間ウィンドウの残り時間にかかわらず、指定数のインシデントを受信するとNNMiによってレートインシデントが生成されます。

インシデントの抑制、強化、およびダンプニング

NNMiには、インシデントからほとんどの値を取得する便利な機能セットが用意されています。各インシデントタイプに対して、以下のインシデント設定オプションでインシデントが関連する場合を具体的に指定することができます。

- 抑制—インシデントが抑制設定に一致すると、そのインシデントはNNMiコンソールインシデントビューに表示されません。インシデントの抑制は、あるノード (ルーター、スイッチなど) にとっては重要であるが、他にとっては重要ではないインシデント (SNMPLinkDownトラップなど) の場合に便利です。
- 強化—インシデントが強化設定に一致すると、インシデントのコンテンツに応じて、NNMiによって1つ以上のインシデント値 (重大度、メッセージなど) が変更されます。インシデントの強化は、トラップvarbind (負荷量) に識別情報を継承するトラップ処理 (RMONFallingAlarmなど) の場合に便利です。
- ダンプニング—インシデントがダンプニング設定に一致すると、ダンプニング周期中、NNMiによってそのインシデントのアクティビティが遅延されます。インシデントのダンプニングには、NNMi Causal Engineがインシデントの根本原因分析を実行する時間があり、NNMiコンソール内のインシデント数を減らし、より意味のあるインシデントにする上で便利です。

NNMiには、各インシデントタイプに抑制、強化、ダンプニングに対する以下の設定レベルが用意されています。

- インタフェースグループ設定 - ソースオブジェクトがNNMiインタフェースグループのメンバーである場合のインシデント動作が指定されます。各インタフェースグループに異なる動作を指定できます。
- ノードグループ設定 - ソースオブジェクトがNNMiノードグループのメンバーである場合のインシデントの動作が指定されます。各ノードグループに異なる動作を指定できます。
- デフォルト設定—デフォルトのインシデント動作が指定されます。

NNMiでは、各インシデントの設定領域 (抑制、強化、ダンプニング) に対して、以下の手順を使用して特定のインシデントの動作が決定されます。

1. インタフェースグループ設定のチェック:
    - ソースオブジェクトが任意のインタフェースグループ設定に一致する場合は、一致内で最下位順序番号で定義された動作を実行し、一致検索を停止します。

- ソースオブジェクトがどのインタフェースグループ設定とも一致しない場合は、手順2に進みます。

2. ノードグループ設定のチェック:
   - ソースオブジェクトが任意のノードグループ設定に一致する場合は、一致内で最下位順序番号で定義された動作を実行し、一致検索を停止します。
   - ソースオブジェクトがどのノードグループ設定とも一致しない場合は、手順3に進みます。
3. デフォルト設定で定義された動作を実行します (ある場合)。

ライフサイクル移行アクション

ライフサイクル移行アクションは管理者が提供するコマンドであり、インシデントのライフサイクル状態が変化してアクション設定と一致したときに実行されます。インシデントのアクション設定は、1つのインシデントタイプの1つのライフサイクル状態に固有です。このインシデントタイプが特定のライフサイクル状態に移行すると、アクション設定により、実行するコマンドが特定されます。コマンドには引数を含めることができ、これによってインシデント情報がアクションコードに渡されます。

アクションコードは、NNMi管理サーバーで正しく実行されるJythonファイル、スクリプト、実行可能ファイルのいずれかにすることができます。アクションコードは1つのインシデントタイプに固有のものにしたり、多くのインシデントタイプを処理するようにしたりできます。たとえば、ConnectionDown、NodeDown、NodeOrConnectionDownのいずれかのインシデントをNNMiが作成したときにネットワークオペレーターを呼び出すアクションコードを作成できます。それぞれのインシデントタイプの [登録済み] ライフサイクル状態に1つのインシデントアクションというように、3つのインシデントアクションを設定できます。

同じように、アクションコードを1つのライフサイクル状態の変化に固有にしたり、複数のライフサイクル状態の変化に対応させたりすることができます。たとえば、NNMiがInterfaceDownインシデントを作成したときにトラブルチケットを生成し、InterfaceDownインシデントがキャンセルされたときにトラブルチケットを解決済みにするアクションコードを作成できます。[登録済み] 状態に1つ、[解決済み] 状態に1つというように、InterfaceDownインシデントに2つのインシデントアクションを設定できます。

それぞれのアクション設定には、CIAに基づいて負荷量フィルターを組み込んで、アクションが実行されるときを制限できます。さらにフィルタリングするには、インシデントの強化を使用してCIAをインシデントに追加できます。NNMiはインシデントソースからその属性の値を判別します。たとえば一部のノードにカスタム属性を追加した場合は、この情報をインシデントにCIAとして追加し、インシデントアクションの負荷量フィルターをこの属性値に基づくようにすることができます。

インシデントの計画

以下の領域で決定します。

NNMiが処理するデバイストラップ

ネットワークに関連するデバイストラップを識別し、各トラップのインシデント設定を計画します。NNMiでは、MIBをNNMiにロードしないでトラップを処理できます。MIBにTRAP-TYPEまたはNOTIFICATION-TYPEマクロが含まれる場合は、MIBで定義されたトラップにスケルトンインシデント設定を作成できます。

NNMiトポロジにないデバイスからのトラップを表示するかどうかを決定します。

NNMiで表示するインシデント

インシデントのデフォルトセットで開始することをお勧めします。インシデント設定は徐々に拡大および削減できます。

重複解除、レート設定、ペア相関処理によって削減できるインシデントを計画します。

詳細については、管理者用のNNMiヘルプを参照してください。

インシデントに対するNNMiの対応方法

インシデントが発生した場合のNNMiのアクション(ネットワークオペレーターへの電子メール送信など)各アクションを実行するライフサイクルの状態

詳細については、管理者用のNNMiヘルプを参照してください。

NNMiによる別のイベントレシーバーへのトラップ転送の可否

環境にサードパーティのトラップ統合が含まれる場合は、NNMi SNMPトラップ転送メカニズムをNNMiノースバウンドインタフェースSNMPトラップ転送メカニズムと一緒に使用するかどうかを決定します。

NNMiノースバウンドインタフェースSNMPトラップ転送メカニズムを選択する場合は、NNMiがイベントレシーバーに転送するすべてのトラップのMIBをロードします。

インシデントの設定

このセクションでは、設定のヒントを一覧にし、いくつかの設定例について説明します。このセクションの情報を読んだ後で、具体的な手順のNNMiヘルプの「インシデントを設定する」を参照して

ください。

注: 大きな設定変更を行う前には、既存の設定のコピーを保存しておくことをお勧めします。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。

- 計画したインシデントタイプを設定します。可能な場合は、MIBで定義したトラップのスケルトンインシデント設定から開始します。
- トラップ転送に必要なMIBをすべてロードします。
- NNMi管理サーバーにトラップを送信するデバイスが設定されていることを確認します。

インシデントの抑制、強化、およびダンプニングの設定

インシデントの抑制、強化、ダンプニングを設定するときは、以下に注意してください。

- 各インタフェースグループ、ノードグループ、またはデフォルト設定に対して、設定を適用できる場合にさらに絞り込むための負荷量フィルターを指定できます。
- インシデント設定フォームの [**インタフェースの設定**] タブにインタフェースグループ設定を設定します。
- インシデント設定フォームの [**ノードの設定**] タブにノードグループ設定を設定します。
- インシデント設定フォームの [**抑制**]、[**強化**]、および [**ダンプニング**] タブにデフォルト設定を設定します。

ライフサイクル移行アクションの設定

ライフサイクル移行アクションを設定するときは、以下に注意してください。

- デフォルトでは、NNMiは以下の場所でアクションを実行します。
  - Windowsの場合: `%NnmDataDir%\shared\nnm\actions`
  - Linuxの場合: `$NnmDataDir/shared/nnm/actions`

  アクションがこの場所にない場合は、[**ライフサイクルの移行アクション**] フォームの [**コマンド**] フィールドでアクションの絶対パスを指定します。

  注: Jythonファイルは`actions`ディレクトリに配置する必要があります。
- アクション設定を変更するたびに、NNMiによってactionsディレクトリでJythonファイルが再読み取りされてNNMiにロードされます。
- アクションは、グループとしてインシデントタイプに対して有効になります。
- アクションに渡すことができるNNMi情報については、NNMiヘルプの「インシデントアクションを設定するための有効なパラメーター」を参照してください。

トラップログの設定

NNMiでは、すべての着信SNMPトラップをログファイル (テキストファイルまたはCSVファイル) に記録できます。トラップは以下の場所に記録されます。

- Windowsの場合: %NnmDataDir%\nnm\log
- Linuxの場合: $NnmDataDir/nnm/log

トラップログファイルは、nnmtrapconfig.ovplスクリプトを使用して設定します。以下の形式を選択できます。

- CSV (デフォルト): トラップはCSV形式で記録されます (trap.csv)。
- TXT: トラップはTXT形式で記録されます (trap.log)。
- BOTH: トラップはCSVとTXTの両方の形式で記録されます (2つのログファイル)。
- OFF: トラップは記録されません。

たとえば、BOTHモードでトラップを記録するように指定する場合は、以下のコマンドを使用します。

nnmtrapconfig.ovpl -setProp trapLoggingMode BOTH -persist

-persist引数を使用することで、トラップサービスの再起動後もすべてのトラップサーバープロパティがそのまま有効になります。-persist引数を使用しない場合、すべてのトラップサーバープロパティはサービスが停止されるまでの間のみ有効です。

トラップはロールファイルに書き込まれます。ログファイルのサイズが定義された上限 (nnmtrapconfig.ovplスクリプトを使用して定義) に達すると、ファイル名がtrap.<format>.oldに変更されます。既存のファイルは置き換えられます。

詳細については、nnmtrapconfig.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。NNMiヘルプの「トラップログ記録を設定する」も参照してください。

インシデントログの設定

受信インシデント情報がincident.logファイルに書き込まれるように、インシデントログを設定できます。この機能は、インシデント履歴を追跡およびアーカイブする場合に役立ちます。

インシデントログを設定して有効にするには、[**設定**] ワークスペースの [**インシデントの設定**] エリアにある [**インシデントログの設定**] タブに移動して設定します。詳細については、NNMiヘルプを参照してください。

トラップサーバープロパティの設定

トラップサーバープロパティ (nnmtrapserver.properties) を設定するには、nnmtrapconfig.ovplスクリプトを使用します。

注: nnmtrapserver.propertiesファイルが存在するファイルディレクトリは編集しないでください。nnmtrapconfig.ovplスクリプトを使用してこのファイルを変更してください。

以下の表に、トラップサーバープロパティのデフォルト値を示します。

トラップサーバープロパティとそのデフォルト値

トラップサーバープロパティ	デフォルト値
com.hp.ov.nms.trapd.udpPort	162
com.hp.ov.nms.trapd.rmiPort	1097
com.hp.ov.nms.trapd.trapInterface	すべてのインタフェース
com.hp.ov.nms.trapd.recvSocketBufSize	2048キロバイト
com.hp.ov.nms.trapd.pipeline.qSize	50000トラップ
com.hp.ov.nms.trapd.connectToWinSNMP	false
com.hp.ov.nms.trapd.blocking	true
com.hp.ov.nms.trapd.blockTrapRate	50トラップ/秒
com.hp.nms.trapd.unblockTrapRate	50トラップ/秒
com.hp.ov.nms.trapd.overallBlockTrapRate	150トラップ/秒
com.hp.nms.trapd.overallUnblockTrapRate	150トラップ/秒
com.hp.ov.nms.trapd.analysis.minTrapCount	100トラップ
com.hp.ov.nms.trapd.analysis.numSources	10ソース
com.hp.ov.nms.trapd.analysis.windowSize	300秒 (5分)
com.hp.nms.trapd.updateSourcesPeriod	30秒
com.hp.nms.trapd.notifySourcesPeriod	300秒
com.hp.ov.nms.trapd.hosted.object.trapstorm.enabled	false
com.hp.ov.nms.trapd.hosted.object.trapstorm.threshold	10トラップ/秒
com.hp.ov.nms.trapd.database.fileSize	100メガバイト
com.hp.ov.nms.trapd.database.fileCount	5ファイル
com.hp.ov.nms.trapd.database.qSize	300000トラップ
com.hp.ov.nms.trapd.discohint.cacheSize	5000エントリ
com.hp.ov.nms.trapd.discohint.cacheEntryTimeout	3600ミリ秒

詳細については、`nnmtrapconfig.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

インシデントを割り当てるときのユーザー名のソート順序に使用されるロケールの設定

NNMi管理者は、インシデントを割り当てるときにユーザー名のソート順序を決定するために使用される、NNMi管理サーバーの言語ロケールを指定できます。

注: 設定したソート順序ロケールは、[**インシデントの割り当て**]ダイアログにのみ適用されます。

アルファベット順を決定するときに、NNMiはユーザーの実際のログイン名ではなく表示名を使用し、大文字を小文字と分けてソートすることはありません。

注: ソート順序を決定する際にNNMiが使用するのは、`sortLocale`に設定されているロケールだけです。`forceClientLocale`プロパティに設定されているブラウザーロケールがソート順序に影響を与えることはありません。詳細については、「ブラウザーのロケール設定の上書き」(261ページ)を参照してください。

注: 高可用性 (HA) 下でファイル変更を行う場合、更新する必要があるserver.propertiesファイルの場所は、`<Shared_Disk>/NNM/dataDir/nmsas/NNM/server.properties`です。

インシデントを割り当てるときに表示されるユーザー名のソート順序に使用する言語ロケールを設定するには、以下の方法で`server.properties`ファイルを編集します。

1. 以下のファイルを開きます。
   - Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`
   - Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`
2. `server.properties`ファイルの次の行をコメント解除します。

   ```
   #nmsas.server.sortLocale = en_US
   ```
3. デフォルトの値を、NNMi管理サーバーの正しいロケールに変更します。たとえば、ロケールをロシア語に変更するには、次のエントリを使用します。

   ```
   nmsas.server.sortLocale = ru_RU
   ```
4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

インシデント設定のバッチロード

`nnmincidentcfgdump.ovpl`および`nnmincidentcfgload.ovpl`の2つのスクリプトをインシデント設定のバッチロードと併用できます。

nnmincidentcfgdump.ovplによるインシデント設定ファイルの生成

NNMi nnmincidentcfgdump.ovplスクリプトでは、インシデント設定を作成または更新し、その後nnmincidentcfgload.ovplスクリプトを使用してNNMiデータベースにロードできます。ファイルは非XML形式で生成されます。

以下のディレクトリにある形式の説明を使用して、ファイルを編集できます。

Windowsの場合: %NnmInstallDir%/examples/nnm/incidentcfg

Linuxの場合: /opt/OV/examples/nnm/incidentcfg

インシデント設定のファイルを生成するには、以下の構文の例を使用します。

```
nnmincidentcfgdump.ovpl -dump <file_name> -u <NNMiadminUsername>
-p <NNMiadminPassword>
```

詳細については、nnmincidentcfgdump.ovplリファレンスページ、またはLinuxのマニュアルページを参照してください。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。HA設定を使用したNNMiでは、変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。

nnmincidentcfgload.ovplによるインシデント設定のロード

NNMi nnmincidentcfgload.ovplスクリプトでは、フォーマットされた設定ファイルからNNMiデータベースにインシデント設定をロードできます。

ヒント: nnmincidentcfgdump.ovplスクリプトを使用して、既存のインシデント設定の設定ファイルを非XML形式で作成します。その後必要に応じて、NNMiデータベースにロードする前にこのファイルを編集できます。

必要な形式については、以下のディレクトリを参照してください。

Windowsの場合: %NnmInstallDir%\examples\nnm\incidentcfg

Linuxの場合: /opt/OV/examples/nnm/incidentcfg

インシデント設定ファイルをNNMiデータベースにロードする前に検証するには、以下の構文の例を使用します。

```
nnmincidentcfgload.ovpl -validate <file_name> -u <NNMiadminUsername>
-p <NNMiadminPassword>
```

インシデント設定をロードするには、以下の構文の例を使用します。

```
nnmincidentcfgload.ovpl -load <file_name> -u <NNMiadminUsername>
-p <NNMiadminPassword>
```

以下の点に注意してください。

- NNMiは、名前またはその他のキー識別子が一致するすべての設定を更新します。

注意: NNMiは、これらの設定に関連付けられたコード値 (インシデントファミリーなど) の上書きも行います。

- NNMiは、NNMiデータベースに存在しないキー識別子のすべてのインシデント設定を追加します。
- NNMiは、エクスポートされたファイル内で一致しないキー識別子の既存のインシデント設定は変更しません。
- NNMiは、設定ファイルで提供されていない場合は一意のオブジェクトID (UUID) を解決します。
- NNMiがUUIDを解決できない場合は、UUIDが作成されます。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。HA設定を使用したNNMiでは、変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。

詳細については、`nnmincidentcfgload.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

インシデントの評価

このセクションでは、インシデント設定を評価する方法を説明します。

- NNMiがネットワークのすべての管理対象デバイスからトラップを受信したことを確認します。

  NNMiがトラップを受信していない場合は、NNMi管理サーバーでファイアウォールの設定を確認します。

  注: 一部のウイルス対策ソフトウェアにはファイアウォールが組み込まれており、システムのファイアウォールとは別に設定されています。

- 最も重要なトラップがインシデントに変換されることを確認します。
- 正しいライフサイクルの状態移行でインシデントアクションが実行されてることを確認します。
- NNMiがインシデントを期待どおり処理していることを確認します。

  [アクション] > [インシデントの設定レポート] メニューには、既存のインシデントをそのインシデントタイプの現在の設定に対してテストする複数のオプションがあります。これらのメニュー項目のいずれかを使用しても、現在NNMiコンソールにあるインシデントは変更されません。

インシデントの調整

NNMiコンソールインシデントビューのインシデント数を削減します。以下のメソッドのいずれかを使用します。

- NNMiコンソールでは必要のないインシデントタイプのインシデント設定を無効にします。
- [管理対象外] または [サービス停止中] をモニタリングする必要がないネットワークオブジェクトの管理モードを設定します。NNMiでは、これらのノードとそのインタフェースからのほとんどの受信トラップは廃棄されます。
- NNMiでネットワークオブジェクトが監視されないように設定します。NNMiでは、監視されないソースオブジェクトからのほとんどの受信トラップは廃棄されます。
- 受信インシデントの追加条件または関係を識別します。これらの条件または関係が発生すると、NNMiでは受信管理イベントやSNMPトラップの条件またはパターンを識別して、関連するインシデントどうしを相関関係の子として入れ子にすることで、インシデントのフローが変更されます。

未定義トラップのインシデントの有効化および設定

NNMiは、デフォルトで未定義トラップをサイレントにドロップします。NNMi 9.01以降、NNMiは、ドロップされる可能性がある未定義SNMPトラップを特定できるようになります。

注: NNM iSPI NETまたはNNMi PremiumがNNMi管理サーバーでライセンス供与されている場合は、`Total Traps Received (by OID)` レポートを使用して、ドロップされたSNMPトラップを調べます。詳細については、NNMiヘルプの「トラップ情報を分析する(NNM iSPI NET)」を参照してください。

NNM iSPI NETまたはNNMi PremiumがNNMi管理サーバーでライセンス供与されておらず、インシデントとして欠落したトラップを確認する場合は、未定義SNMPトラップインシデントを以下のように設定します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\nms-jboss.properties`
   - Linuxの場合: `$NNM_PROPS/nms-jboss.properties`
2. ファイルから、以下の行のようなセクションを特定します。

   #!com.hp.nnm.events.allowUndefinedTraps=false

   この行を以下のように変更します。

   com.hp.nnm.events.allowUndefinedTraps=true
3. 省略可能。`nms-jboss.properties`ファイルで説明されている値を使用し、インシデントの重大度を指定します。ファイルから、以下の行のようなセクションを特定します。

   #!com.hp.nnm.events.undefinedTrapsSeverity=NORMAL

   この行を以下のように変更し、定義した重大度の値を`YourSpecifiedSeverity`の代わりに使用します。

   com.hp.nnm.events.undefinedTrapsSeverity=YourSpecifiedSeverity
4. 省略可能。`nms-jboss.properties`ファイルで説明されている値を使用し、インシデントの特性

を指定します。ファイルから、以下のようなセクションを特定します。

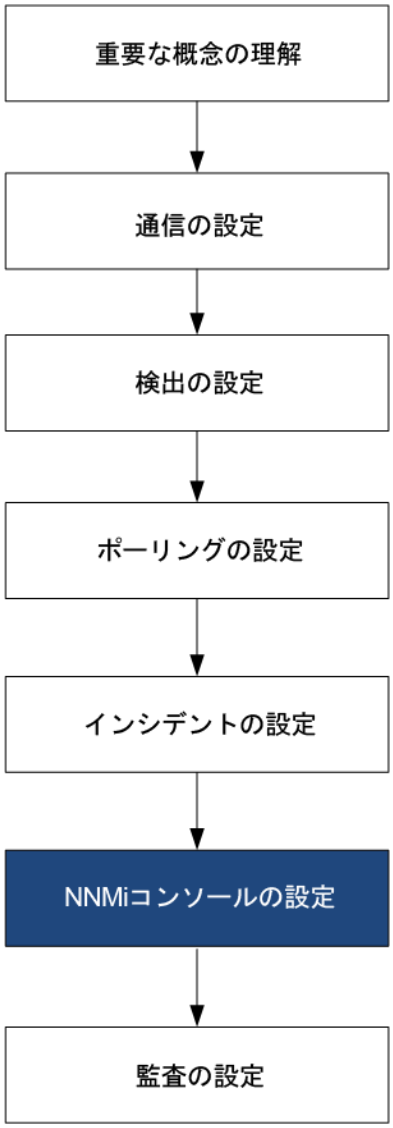
#!com.hp.nnm.events.undefinedTrapsNature=INFO

この行を以下のように変更し、定義した特性の値をYourSpecifiedNatureの代わりに使用します。

com.hp.nnm.events.undefinedTrapsNature=YourSpecifiedNature

5. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。
6. 未定義トラップのリストを検討し、制御するトラップ用に新しいインシデント設定を作成します。NNMiで新しいインシデントを表示する場合はそれを有効にして、NNMiで新しいインシデントを無視する場合はそれを無効にします。詳細については、NNMiヘルプの「SNMPトラップインシデントを設定する」を参照してください。

NNMiコンソール



この章の情報を読み、NNMiコンソールを使用してNNMiの機能を設定する具体的な方法について理解してください。

この章には、以下のトピックがあります。

- 「デバイスのプロファイルルアイコンのカスタマイズ」(130ページ)
- 「テーブルビューのリフレッシュレートの設定」(131ページ)

ネットワークの概要マップに表示されるノードの最大数の削減

[**ネットワークの概要**] マップには、レイヤー3ネットワークで最も高度に接続された250までのノードを含むマップが表示されます。このマップに含まれるノード数が多すぎると、ノードを移動するときのマップの反応が遅くなったり、複雑すぎて実際の表示に適さなくなったりする可能性があります。

[**ネットワークの概要**] マップに表示する最大ノード数を増減させることが可能です。これを行うには、[ユーザーインタフェースの設定] フォームの [**デフォルトのマップ設定**] タブにある [表示するノードの最大数] 属性を編集します。

[**ネットワークの概要**] マップに表示する最大ノード数の増減は、次に示す例の手順を実行しても行うことができます。

たとえば、[**ネットワークの概要**] マップに表示されるノードの最大数を250から100に変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\nms-ui.properties`
   - Linuxの場合: `$NNM_PROPS/nms-ui.properties`
2. 以下の行に似たテキストを特定します。

   #!com.hp.nnm.ui.networkOverviewMaxNodes = 250

   行を以下のように変更します。

   com.hp.nnm.ui.networkOverviewMaxNodes = 100

   > **注:** 行の始めにある**#!**文字を必ず削除してください。
3. 変更を保存します。

ノードグループマップの表示ノード数の削減

数百単位のノードを含むようにノードグループマップを設定すると、ノードグループを表示するマップには、予期される詳細なノードアイコンではなく、多くの小さいノードアイコンが表示されます。より詳細なマップを表示するには、ズーム機能を使用する必要があります。

> **注:** ズーム機能を使用すると、マップを表示するときのNNMiコンソールのパフォーマンスが低下する可能性があります。

表示されるノードまたは表示されるエンドポイント、あるいはその両方の数を制限するには、以下の手順を実行します。

1. NNMiコンソールで、[**設定**] をクリックします。
2. [**ユーザーインタフェースの設定**] をクリックします。
3. [**デフォルトのマップ設定**] タブを選択します。
4. [**表示するノードの最大数**] フィールドに表示された値を変更します。
5. [**表示するエンドポイントの最大数**] フィールドに表示された値を変更します。
6. [**保存して閉じる**] をクリックします。

詳細については、NNMiヘルプの「デフォルトマップ設定を定義する」を参照してください。

[分析] ペインのゲージの設定

[分析] ペインの [ゲージ] タブには、State Pollerとカスタムポーラーの SNMPデータを示すために、リアルタイムのSNMPゲージが表示されます。これらのゲージには、ノード、インタフェース、カスタムノード収集のデータや、CPU、メモリ、バッファー、バックプレーンタイプのノードコンポーネントのデータが表示されます。

以下のプロパティファイルを編集してゲージを設定できます。

- Windowsの場合: `%NNM_PROPS%\nms-ui.properties`
- Linuxの場合: `$NNM_PROPS/nms-ui.properties`

設定する各プロパティで、行の始めにコメント文字 (`#!`) が存在する場合は削除します。

注: 後続の項で説明するプロパティはすべてのノードに適用されます (個別のノードグループにプロパティを適用することはできません)。

ヒント: 変更を行う前に`nms-ui.properties`ファイルのバックアップコピーを作成します。バックアップコピーは、編集するプロパティファイルが格納されているディレクトリに配置しないでください。

詳細については、`nms-ui.properties`ファイル内のコメントも参照してください。

表示されるゲージ数の制限

以下の行を編集して目的の値を入力し、表示するゲージの最大数を設定します。

com.hp.nnm.ui.maxGaugePerAnalysisPanel =

ヒント: ゲージ数が多いほど、分析ペインの表示時のパフォーマンスに影響します。ゲージ数が少ないほどゲージのサイズが大きくなります。

[分析] ペインにあるゲージの更新間隔の設定

以下のプロパティ値を編集して、[分析] ペインに表示されるゲージの更新間隔 (秒) を設定します。

com.hp.nnm.ui.analysisGaugeRefreshSecs =

ヒント: 値を「0」に設定すると、ゲージが更新されなくなります。更新間隔を10秒より速くすると、一部のSNMPエージェントでは短時間で値がキャッシュされ、結果が同じになります。

ゲージの非表示

以下の行を編集し、非表示にするゲージのリストを入力して、(すべてのゲージビューの) 表示しないゲージを定義します。

com.hp.nnm.ui.analysisGaugeNoDisplayKeyPatterns =

以下の点に注意してください。

- 関連するすべての行からコメント文字を削除してください。
- ゲージのリスト内にコメントを含めることはできません。
- ゲージのリスト内に空白行を含めないようにします。
  空白行がある場所でエントリが終了します。
- コメント内の設定がこのプロパティのデフォルト設定
  この設定を拡張または修正する場合、これらの設定を含める必要があります。含めないと、予期しない数のゲージが表示されます。

表示されるノードゲージの順序の制御

ノードゲージが表示される順序を制御するには、以下の行を編集します。

com.hp.nnm.ui.analysisGaugeNodeComponentKeys =

以下の点に注意してください。

- このプロパティ設定では、ワイルドカードはサポートされていません。
- リストにコメントまたは空白行が含まれていないことを確認してください。
- このプロパティのデフォルト設定がコメントとして表示されます。この設定を拡張または修正する場合、これらの設定を含める必要があります。含めないと、設定した順序で表示されません。

表示されるインタフェースゲージの順序の制御

インタフェースゲージが表示される順序を制御するには、以下の行を編集します。

com.hp.nnm.ui.analysisGaugeInterfaceKeys =

このプロパティ設定では、ワイルドカードはサポートされていません。リストにコメントまたは空白行が含まれていないことを確認してください。

コメント内の設定がこのプロパティのデフォルト設定です。この設定を拡張または修正する場合、これらの設定を含める必要があります。含めないと、意図した順序で表示されません。

表示されるカスタムポーラーゲージの順序の制御

カスタムポーラーゲージが表示される順序を制御するには、以下の行を編集します。

com.hp.ov.nnm.ui.analysisGaugeCustomPolledInstanceKeys =

注: この属性にデフォルト設定はありません。

ゲージプロパティの適用方法の理解

ゲージプロパティは以下の順序で適用されます。

1. すべてのゲージのリストがState Pollerから取得されます。
2. `analysisGaugeNoDisplayKeyPatterns`が最初に適用されて、指定のゲージがリストから削除されます。
3. `analysisGaugeNodeComponentKeys`、`analysisGaugeInterfaceKeys`、または`analysisGaugeCustomPolledInstanceKeys`が必要に応じて適用され、表示されるゲージのリストの順序が決まります。
4. 最後に、`maxGaugePerAnalysisPanel`が適用されて、表示されるリストが切り捨てられます。

ゲージに関する問題のトラブルシューティング

このセクションでは、ゲージに関する以下の問題のトラブルシューティングについて説明します。

- 「表示されるゲージが多すぎる」(128ページ)

表示されるゲージが多すぎる

ゲージが多すぎる場合は、以下のいずれかを実行します。

- `maxGaugePerAnalysisPanel`プロパティを使用して、表示されるゲージ数を制限します。
  詳細については、「表示されるゲージ数の制限」(126ページ)を参照してください。
- `analysisGaugeNoDisplayKeyPatterns`プロパティを使用して、不要なゲージを削除します。
  詳細については、「ゲージの非表示」(127ページ)を参照してください。

マップラベルのスケールサイズと境界の設定

NNMi管理者は、nms-ui.propertiesファイルを使用してマップビューに以下の調整を加えることができます。

- マップとしてのノードラベルおよびポートラベルのスケール値は、ズーム機能によってサイズ変更される。
- マップ上におけるノードまたはポートとそれらのラベル間のサイズ差を決定するために使用できる最大相対スケール係数。
- ノードとポートのラベルが黒い枠で囲まれるかどうか。

> **注:** デフォルトでは、ラベルが重なるときに読みやすいように、ノードとポートのラベルは黒い枠で囲まれます。

次の表に変更するプロパティを示します。

> **ヒント:** 各スケール調整プロパティ値は、NNMiで使用される実際のスケール係数を掛けたものです。たとえば、labelScaleAdjust値を0.50に変更すると、マップ上に表示されるラベルはその通常のサイズの半分になります。

nms-ui.propertiesファイルで変更するプロパティ

プロパティ	デフォルト値	説明
!com.hp.nnm.ui.labelScaleAdjust	1.0	ノードとポートのマップラベルのスケールサイズを調整します。
!com.hp.nnm.ui.maxLabelScaleAdjust	1.0	ノードまたはポートおよびそれらのラベル間のサイズ差を決定するために使用できる最大相対スケール係数を調整します。
!com.hp.nnm.ui.omitLabelRectangle	true	ノードラベルとポートラベルを囲むために黒い枠を使用するかどうかを決定します。 **注:** 枠を表示しない場合、この値をfalseに設定します。

> **注:** 変更を適用するには、マップビューを開き直すか、または変更します。

Loom図およびWheel図の自動折りたたみしきい値の設定

NNMi管理者は、Loom図とWheel図が相当複雑になったときに読みやすくするために、これらの図が初期動作として自動的にノードの折りたたみ (インタフェースの非表示) とスイッチの折りたたみ (ポートの非表示) を行うポイントを設定できます。この設定は、nms-ui.propertiesファイルの以下のプロパティを調整して行います。

LoomおよびWheelの自動折りたたみしきい値

プロパティ	説明
com.hp.nnm.ui.wheelAutoCollapseThreshold	このプロパティは、Wheel図の自動的な折りたたみが開始されるまでに境界線の周囲に必要なラベル数を指定するために使用します。
com.hp.nnm.ui.loomAutoCollapseThreshold	このプロパティは、Loom図の自動的な折りたたみが開始されるまでに図全体で必要なラベル数を指定するために使用します。

自動折りたたみしきい値を設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS\nms-ui.properties
   - Linuxの場合: $NNM_PROPS/nms-ui.properties
2. 必要に応じ、必要なプロパティをコメント解除します。詳細については、nms-ui.propertiesファイル内のコメントを参照してください。
3. 必要に応じてしきい値を更新し、変更を保存します。
4. 変更を適用するには、NNMiコンソールで図を開き直します。

デバイスのプロファイルルアイコンのカスタマイズ

NNMiでは、デバイスのプロファイルルまたは特定のノードに関連付けられているアイコンをカスタマイズできます。これらのアイコンはテーブルビューやメニュー項目に表示されます。また、NNMiトポロジマップの前景イメージとしても表示されます。

nnmicons.ovplコマンドを使用して1つ以上のアイコンをカスタマイズできます。詳細については、nnmicons.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

NNMiの『管理者用のヘルプ』も参照してください。

テーブルビューのリフレッシュレートの設定

NNMiでは、NNMi管理者はNNMiコンソールのテーブルビューのデフォルトのリフレッシュレートを上書きできます。

注: 最小の推奨リフレッシュレートは30秒です。リフレッシュレートを30秒未満に設定すると、パフォーマンスを低下させる可能性があります。

NNMiテーブルビューのデフォルトのリフレッシュレートを上書きするには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: `%NMS-PROPS%\nms-ui.properties`

   Linuxの場合: `$NNM_PROPS/nms-ui.properties`
2. 変更対象のリフレッシュレートを持つビューの`viewInfoId` URLパラメーターを決定します。
   a. 変更対象のリフレッシュレートを持つビューを開きます。
   b. [**新しいウィンドウでビューを表示**] をクリックします。
   c. `viewInfoId URL`パラメーターをメモします。たとえば、**viewInfoId=allIncidentsTableView**です。
3. 以下のフォーマットを使用すると、`nms-ui.properties`にビューとそのリフレッシュレートを秒で指定する行が追加されます。

   com.hp.ov.nms.ui.refreshViewSecs.VIEWKEYWORD = SECS

   以下の点に注意してください。

   - **VIEWKEYWORD**は、ビューのviewInfoId URLパラメーターです。
   - **SECS**は、秒数で表したリフレッシュレートです。
   - コマンドラインの最後に余分なスペースがないことを確認してください。

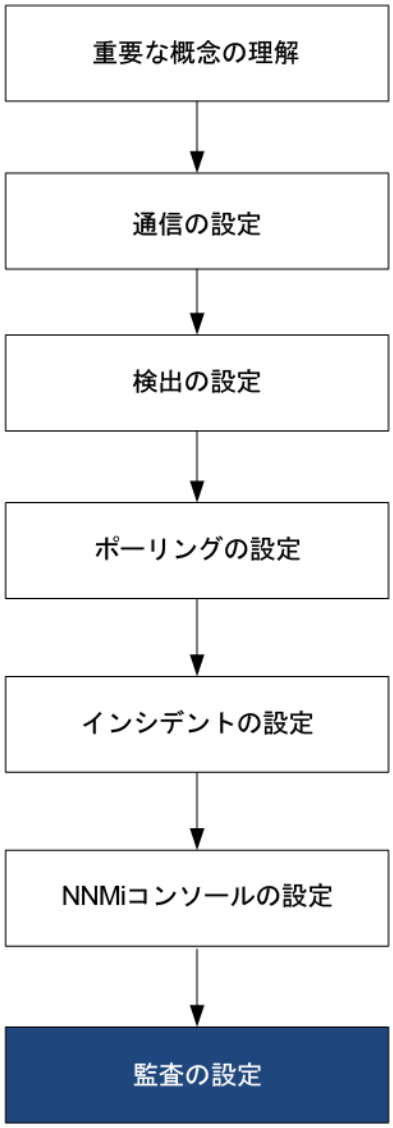
   たとえば、[**すべてのインシデント**] ビューのリフレッシュレートを120秒に変更するには、nms-ui.propertiesに以下の行を追加します。

   com.hp.ov.nms.ui.refreshViewSecs.allIncidentsTableView = 120
4. 変更を保存します。

新しいリフレッシュレートを表示するには、別のビューを開いてから、設定したばかりのリフレッシュレートを持つビューに戻ります。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMi監査



デフォルトでは、NNMiはNNMiデータベースに変更を加えることになるユーザーアクションを監査します。こうしたユーザーアクションには以下のユーザーアクションが含まれますが、その他にも含まれるものがあります。

- NNMiトポロジオブジェクトへの変更 (たとえば、ノード、ノードグループ、インタフェース、インタフェースグループ)。たとえば、ノードグループまたはインタフェースグループの作成または削除、ノードグループまたはインタフェースグループのフィルターまたはメンバーシップの変更などです。
- インシデントライフサイクル情報への変更。たとえば、インシデントの所有者または状態の変更などです。

- ユーザーおよびアクセス情報への変更。たとえば、パスワードの変更、ユーザーアカウントまたはユーザーグループの追加または削除、テナントの作成などです。
- NNMiコンソールの [**設定**] ワークスペースまたはコマンドラインツールを使用して行われた設定変更。たとえば、SNMP設定、検出設定、モニタリング設定への変更などです。
- NNMiコンソールの [**アクション**] メニューからのユーザーアクション。たとえば、設定のポーリングとステータスのポーリングなどです。

監査ログに書き込まれる情報のタイプの例については、「NNMi監査ログファイルについて」(137ページ) を参照してください。

注: デフォルトでは、以下のアクションまたは変更は、監査ログに含まれません。

- **system**ユーザーによって実行されるアクション
- NNMiによって自動的に実行されるアクションは監査ログに含まれません。このデフォルト動作を変更するには、「NNMi監査ログファイルに含まれるアクションの設定」(135ページ)を参照してください。

以下の点に注意してください。

- NNMi監査は、デフォルトで有効になっています。
- 監査情報は、1日に1つのログファイルに書き込まれます。
- 監査ログファイルは、以下のディレクトリに存在します。

  ヒント: NNMi管理者は、NNMiコンソールの **[ツール] > [NNMi監査ログ]** メニューオプションから最新の監査ログを表示することもできます。

  Windowsの場合:%NnmDataDir%\nmsas\NNM\log\audit-<date>.log

  Linuxの場合:$NnmDataDir/nmsas/NNM/log/audit-<date>.log

- **ログエントリの例**:

  2014-04-30T01:20:25.301 joe.operator 10.12.203.55 MODEL abb44ddb-ae52-40d9-855f-f6ab0ab899e1 **UPDATE Node** 151434 172.20.12.7 **managementMode** MANAGED NOTMANAGED

  監査ログの各レコードには、以下のような情報が含まれます。

  - タイムスタンプ
  - ユーザー名
  - (該当する場合リモートホストの) リモートアドレス
  - レコードタイプ (変更のタイプを示すカテゴリ)
  - トランザクションID (該当する場合)
  - 操作/アクション (該当する場合、実行されるアクション)

- ターゲットオブジェクトのタイプ (該当する場合、変更されたオブジェクト)
- オブジェクトまたはアクションに使用できる追加メタデータ (該当する場合):
  ターゲットオブジェクトID
  ターゲットオブジェクト名
  フィールド名
  フィールドの前の値
  フィールドの新しい値

> **注:** パスワードの値は、「パスワード**********」のようにアスタリスクで表示されます。

ログファイルエントリの例については、「NNMi監査ログファイルについて」(137ページ)を参照してください。

- NNMiでは、各監査ログファイルが14日間保持されます。

NNMi管理者は、以下を設定できます。

- 「監査の無効化」(134ページ)
- 「NNMi監査ログの保持日数の指定」(135ページ)
- 「NNMi監査ログファイルに含まれるアクションの設定」(135ページ)

監査の無効化

NNMi監査は、デフォルトで有効になっています。

NNMi監査を無効にするには、以下の手順を実行します。

1. 以下の設定ファイルを開きます。

   Windows

   %NnmDataDir%\nmsas\NNM\conf\nms-audit-config.xml

   Linux

   $NnmDataDir/nmsas/NNM/conf/nms-audit-config.xml
2. 以下を含むテキストブロックを探します。

   <enabled>true</enabled>
3. `true`を`false`に変更します。

   <enabled>false</enabled>
4. 変更を保存します。
5. NNMi管理サーバーを再起動します。
   NNMi管理サーバーで`ovstop`コマンドを実行します。
   NNMi管理サーバーで`ovstart`コマンドを実行します。

NNMi監査ログの保持日数の指定

デフォルトでは、アーカイブされた各監査ログファイル (1日に1つ) がNNMiで14日間保持されます。

アーカイブされた監査ログファイルのNNMiでの保持日数を変更するには、以下の手順を実行します。

注: この数値は、現在の日付の監査ログファイルには影響しません。

1. 以下の設定ファイルを開きます。

   Windows

   %NnmDataDir%\nmsas\NNM\conf\nms-audit-config.xml

   Linux

   $NnmDataDir/nmsas/NNM/conf/nms-audit-config.xml

2. 以下を含むテキストブロックを探します。

   <retain>14</retain>

3. 各監査ログファイルのNNMiでの保持日数を含めるように行を変更します。たとえば、日数を1週間に変更するには、以下のように入力します。

   <retain>7</retain>

   これに対し、NNMiで以下のものが保持されます。

   - 現在の監査ログ
   - 1日に1つの監査ログをあと7日間

4. 変更を保存します。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

NNMi監査ログファイルに含まれるアクションの設定

デフォルトでは、NNMiはNNMiデータベースに変更を加えることになるユーザーアクションを監査します。こうしたユーザーアクションには以下のユーザーアクションが含まれますが、その他にも含まれるものがあります。

- NNMiトポロジオブジェクトへの変更 (たとえば、ノード、ノードグループ、インタフェース、インタフェースグループ)。たとえば、ノードグループまたはインタフェースグループの作成または削除、ノードグループまたはインタフェースグループのフィルターまたはメンバーシップの変更な

どです。

- インシデントライフサイクル情報への変更。たとえば、インシデントの所有者または状態の変更などです。
- ユーザーおよびアクセス情報への変更。たとえば、パスワードの変更、ユーザーアカウントまたはユーザーグループの追加または削除、テナントの作成などです。
- NNMiコンソールの [**設定**] ワークスペースまたはコマンドラインツールを使用して行われた設定変更。たとえば、SNMP設定、検出設定、モニタリング設定への変更などです。
- NNMiコンソールの [**アクション**] メニューからのユーザーアクション。たとえば、設定のポーリングとステータスのポーリングなどです。

監査ログに書き込まれる情報のタイプの例については、「NNMi監査ログファイルについて」(137 ページ) を参照してください。

NNMi監査ログファイルを調べた後で、特定のアクション、エンティティ、またはフィールドの監査を含めるまたは除外する場合があります。例については手順3を参照してください。

ヒント: 各監査ログメッセージには、<entity_name>の直前に<action_name>があります。フィールド名は<entity_name>の後に表示されます。以下にメッセージ例を示します。アクション (UPDATE)、エンティティ (Node)、およびフィールド名 (managementMode) はボールドで示しています。

2014-04-30T01:20:25.301 joe.operator 10.12.203.55 MODEL abb44ddb-ae52-40d9-855f-f6ab0ab899e1 UPDATE **Node** 151434 172.20.12.7 **managementMode** MANAGED NOTMANAGED

NNMi監査ログに含める情報を変更するには、以下の手順を実行します。

1. 以下の設定ファイルを開きます。

   Windows

   %NnmDataDir%\nmsas\NNM\conf\nms-audit-config.xml

   Linux

   $NnmDataDir/nmsas/NNM/conf/nms-audit-config.xml

2. 以下を含むテキストブロックを探します。

   <rules>

   <!-- define custom audit rules here.Any rules here will override system defaults -->

   </rules>

3. ルールを以下のように変更します。
   - 監査ログの1つのメッセージを除外するには、以下の構文を使用します。

     <exclude entity="<entity_name>" field="<field_name>" action="<action_name>"/>

     以下の例では、この監査ログメッセージ例を除外します。

     2014-04-30T01:20:25.301 joe.operator 10.12.203.55 MODEL abb44ddb-ae52-40d9-855f-f6ab0ab899e1 **UPDATE Node** 151434 172.20.12.7 **managementMode** MANAGED NOTMANAGED

<exclude entity="Node" field="managementMode" action="UPDATE" />

- エンティティに対するすべてのアクションを監査ログから除外するには、以下の構文を使用します。

  <exclude entity="<entity_name>" />

  以下の例では、ノードに対するすべての更新操作を監査ログから除外します。

  <exclude entity="Node" />

  - エンティティに対して指定されたアクションを除外するには、以下の構文を使用します。

  <exclude entity="<entity_name>" action="<action_name>" />

  以下の例では、ノードに対するすべての更新操作を監査ログから除外します。

  <exclude entity="Node" action="UPDATE" />

  以下の例では、ノードに対するすべての削除操作を監査ログから除外します。

  <exclude entity="Node" action="DELETE" />

- 任意のオブジェクトの指定されたフィールドに対するすべてのアクションを監査ログから除外するには、以下の構文を使用します。

  <exclude field="<field_name>" />

  以下の例では、任意のオブジェクトのmanagementModeフィールドに対するすべての更新を監査ログから除外します。

  <exclude field="managementMode" action="UPDATE" />

4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

NNMi監査ログファイルについて

このセクションでは、監査ログファイルに含まれる情報タイプの例を示します。

- ノードのセキュリティグループの変更後に生成される監査ログエントリの例

  以下は、**mimcisco3**という名前のノードのセキュリティグループが**デフォルトのセキュリティグループ**から**testgrp**に変更された場合に生成されるログエントリの例です。

  2014-04-15T01:56:54.979 admin "" MODEL 5fd8ed33-e671-494e-ab25-06d293347c4f **UPDATE Node 50281 mimcisco3** securityGroup "138/**Default Security Group**" 56651/**testgrp**

- ユーザーアカウントが作成された場合に生成される監査ログエントリの例

  以下は、ユーザーop1にアカウントを作成した場合に生成されるログエントリの例です。

  2014-04-15T01:55:48.574 admin "" MODEL 4654e06c-5c1f-4955-bf82-e317dcbf38f3 CREATE Account 56647 **op1** alg "" SHA-256

2014-04-15T01:55:48.574 admin "" MODEL 4654e06c-5c1f-4955-bf82-e317dcbf38f3 CREATE Account 56647 **op1** external "" false

2014-04-15T01:55:48.574 admin "" MODEL 4654e06c-5c1f-4955-bf82-e317dcbf38f3 CREATE Account 56647 **op1** name "" op1

2014-04-15T01:55:48.574 admin "" MODEL 4654e06c-5c1f-4955-bf82-e317dcbf38f3 CREATE Account 56647 **op1** password "" ********

- ユーザーアカウントがユーザーグループに割り当てられた場合に生成される監査ログエントリの例

  以下は、ユーザー**op1**が**NNMi レベル1オペレーター**ユーザーグループに割り当てられた場合に生成されるログエントリの例です。

  2014-04-15T01:55:48.597 admin "" MODEL 4654e06c-5c1f-4955-bf82-e317dcbf38f3 CREATE UserGroupMember 56650 5486f4cf-a3e0-4f24-abd6-28f5169f9f92 account "" 56647/**op1**

  2014-04-15T01:55:48.597 admin "" MODEL 4654e06c-5c1f-4955-bf82-e317dcbf38f3 CREATE UserGroupMember 56650 5486f4cf-a3e0-4f24-abd6-28f5169f9f92 **userGroup** "" 141/**level1**

- ユーザーアカウントのパスワードが変更された場合に生成される監査ログエントリの例

  以下は、ユーザーアカウント**op2**のパスワードが変更された場合に生成されるログエントリの例です。

  注: 1番目のユーザー名は、変更を加えるユーザーの名前です。2番目のユーザー名は、パスワードが変更されたアカウント名です。

  2014-04-15T02:04:39.121 **admin** "" MODEL 0ae97c60-3035-46e0-a20c-20b6da04615f UPDATE Account 56645 **op2 password** ******** ********

第4章: 復元

HP Network Node Manager i Software (NNMi) では、ハードウェア障害の場合にNNMiデータを保護するため、次の2つの方法がサポートされます。

- NNMiのアプリケーションフェイルオーバーでは、組み込みNNMiデータベースのトランザクションログのコピーが同一設定システムで維持され、ディザスターリカバリが提供されます(NNMiでOracleデータベースが使用されている場合は、2つのシステムが同一のデータベースに別々の時間に接続されます)。
- 高可用性 (HA) クラスターでNNMiを実行すると、組み込みNNMiデータベースと設定ファイルが共有ディスクに保持され、NNMi管理サーバーがほぼ100パーセント利用されます(NNMiでOracleデータベースが使用されている場合は、共有ディスクにNNMi設定ファイルが含まれ、2つのシステムが同一のデータベースに別々の時間に接続されます)。

両方の手法では、現在のNNMi管理サーバーで障害が発生すると、第2システムが自動的にNNMi管理サーバーになります。

以下の表では、NNMiデータ復元の2つの方法のさまざまな側面を比較しています。

注: NNMi (単品)、およびNNMiに同梱されているNNMi Advanced機能とNNM iSPI NET機能を購入した場合、アプリケーションフェイルオーバーおよび高可用性環境で使用するためのライセンスには2つのタイプがあります。

- アプリケーションフェイルオーバー
  - 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスをプライマリサーバーのIPアドレスと関連付けます。
  - 非商用 - このライセンスは、アプリケーションフェイルオーバー環境で使用するために個別に購入されます。このライセンスをセカンダリ (スタンバイ) サーバーのIPアドレスと関連付けます。

  高可用性 (HA)
  - 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスを物理クラスターノードのいずれかのIPアドレスに関連付けます。
  - 非商用 - このライセンスは、高可用性環境で使用するために個別に購入されます。このライセンスをNNMi HAリソースグループの仮想IPアドレスに関連付けます。
- 指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、アップリケーションフェイルオーバーまたは高可用性で使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。必ず以下

を要求します。
- 高可用性:NNMi HAリソースグループの仮想IPアドレス用のライセンスキーを取得します。このライセンスキーは、最初はプライマリサーバーで使用され、必要に応じてセカンダリサーバーで使用されます。
- アプリケーションフェイルオーバー:プライマリサーバーの物理IPアドレスに1つと、スタンバイサーバーの物理IPアドレスに1つの、2つのライセンスキーを取得します。

注:同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。

- 以下で入手できる各NNM iSPIのドキュメントも参照してください。http://h20230.www2.hp.com/selfsolve/manuals。

NNMiデータ復元の比較

比較項目	NNMiのアプリケーションフェイルオーバー	HAクラスターで動作するNNMi
必要なソフトウェア製品	NNMiまたはNNMi Advanced	• NNMiまたはNNMi Advanced • 個別に購入するHA製品
フェイルオーバーにかかる時間	通常の状態では、インストール済みのNNM iSPIsの数に応じて5-30分間。	通常の状態では、インストール済みのNNM iSPIsの数に応じて5-30分間。
フェイルオーバーの透過性	部分的。NNMi管理サーバーのIPアドレスは、スタンバイサーバーだったものの物理アドレスに変わります。ユーザーは新しいIPアドレスでNNMiコンソールに接続する必要があります。一部のアプリケーションはNNMi管理サーバーの動作に従いますが、大部分のアプリケーション (NNM iSPIsなど) は従いません。	完全。すべての接続ではHAクラスターの仮想IPアドレスが使用され、これはフェイルオーバー時にも変わりません。
アクティブサーバーとスタンバイサーバーの相対的な近接性	LANまたはWAN	LANまたはWAN (一部のHA製品のみ)
インストールするライセンス	• 最初のアクティブなサーバーにはライセンスキー。 • 最初のスタンバイサーバーにはライセンスキー。	最初のアクティブなサーバーには、共有ディスクで管理されるライセンスキー。
NNM iSPIsのサポート	さまざまなサポートがあります。各NNM iSPIのマニュアルを参照してください。	

NNMiデータ復元の比較 (続き)

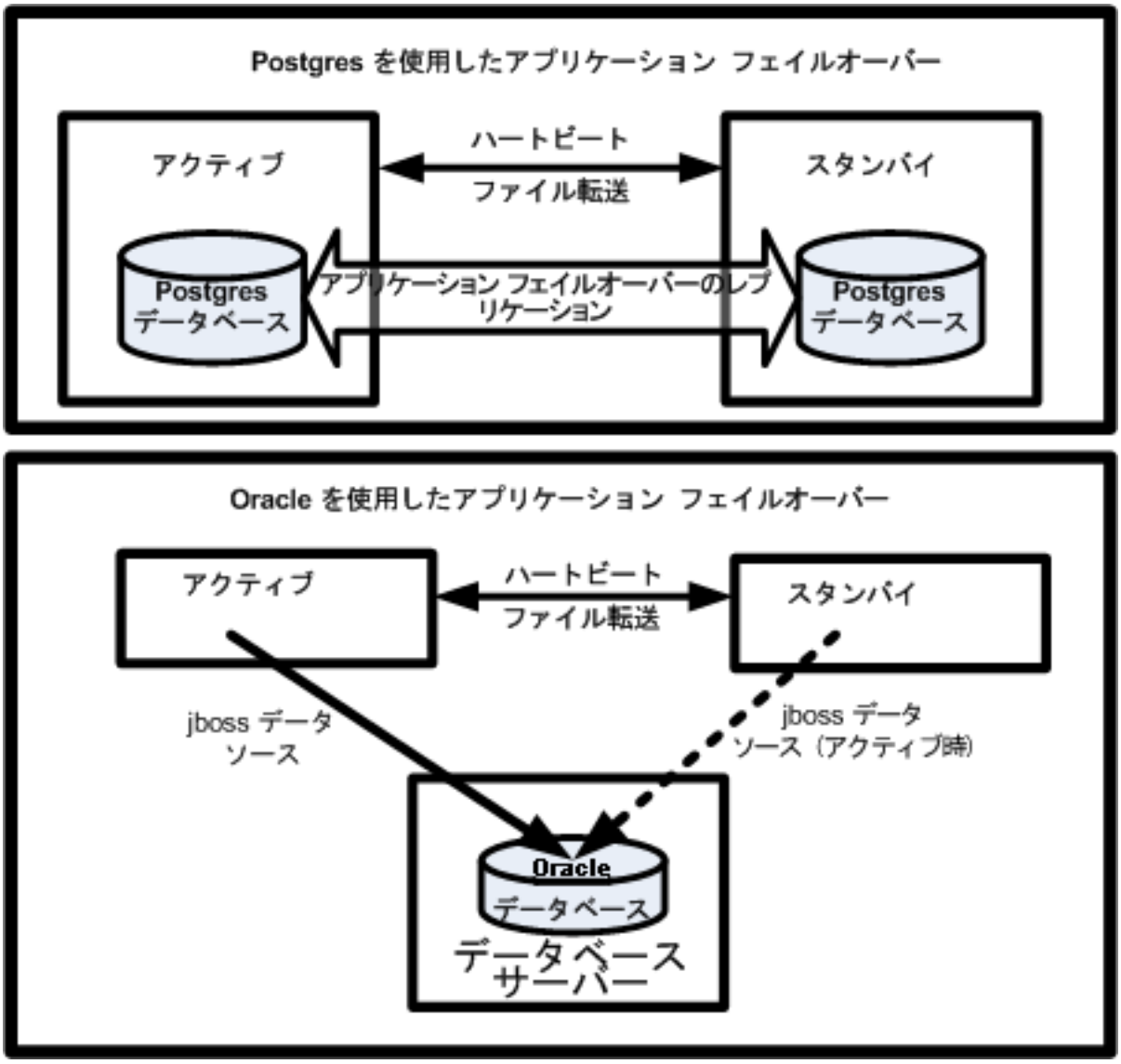
比較項目	NNMiのアプリケーションフェイルオーバー	HAクラスターで動作するNNMi
グローバルネットワーク管理とのインタラクション	• アプリケーションフェイルオーバーまたはHA用に各グローバルマネージャーを設定可能。 • アプリケーションフェイルオーバーまたはHA用に各リージョナルマネージャーを設定可能。 • それぞれの設定には、2つの物理または仮想システムが必要です。[a] • グローバルマネージャーまたはリージョナルマネージャーがフェイルオーバーすると、NNMiは、グローバルマネージャーとリージョナルマネージャー間の接続を再確立します。	
NNMiのメンテナンス	パッチまたはアップグレードを適用する前に、NNMiのアプリケーションフェイルオーバークラスターを停止する必要があります。	HAを設定解除しないで、NNMiにパッチおよびアップグレードを適用できます。

このセクションでは以下の章について説明します。

- 「アプリケーションフェイルオーバー構成のNNMiの設定」(142ページ)
- 「高可用性クラスターにNNMiを設定する」(172ページ)

[a]HAの仮想マシンサポートは、HAソフトウェアベンダーによる仮想システムのサポートに依存します。

アプリケーションフェイルオーバー構成のNNMiの設定



重要なネットワーク機器の障害発生を知らせ、その障害の根本原因を示すHP Network Node Manager i Software (NNMi) は、多くのITプロフェッショナルから信頼を寄せられています。NNMi管理サーバーに障害が発生した場合でも、引き続きNNMiがネットワーク機器の障害発生を知らせてくれる必要があります。このニーズを満たすのが**NNMi**の**アプリケーションフェイルオーバー**で、NNMiプロセスのアプリケーションコントロールをアクティブなNNMi管理サーバーからスタンバイNNMi管理サーバーに引き渡すことで、NNMiの機能は中断なく提供されます。

この章には、以下のトピックがあります。

アプリケーションフェイルオーバーの概要

アプリケーションフェイルオーバー機能は、組み込みデータベースまたはOracleデータベースを使用してNNMiをインストールすることで利用できるようになります。システムにアプリケーションフェイルオーバー機能を設定すると、NNMiはNNMi管理サーバーの障害を検出した場合に、セカンダリサーバーにNNMiの機能を引き渡します。

NNMiのアプリケーションフェイルオーバー設定では、以下の用語と定義を使用しています。

- **アクティブ**: NNMiプロセスを実行中のサーバー。
- **スタンバイ**:フェイルオーバーのイベントを待機しているNNMiクラスター内のシステム。このシステムはNNMiプロセスを実行していません。
- **クラスターメンバー**: クラスターに接続するためにJGroups技術を使用しているシステムで実行中のJavaプロセス。1つのシステムに複数のメンバーを登録できます。
- **Postgres**:トポロジ、インシデント、設定情報などの情報を保存するためにNNMiが使用する組み込みデータベース。
- **Cluster Manager**: アプリケーションフェイルオーバー機能におけるサーバーのモニタリングと管理に使用される`nnmcluster`プロセスおよびツール。

アプリケーションフェイルオーバーの要件

アプリケーションフェイルオーバー機能を導入するには、NNMiを2つのサーバーにインストールします。この章では、この2つのNNMi管理サーバーを**アクティブ**サーバーと**スタンバイ**サーバーとして説明します。通常の運用では、アクティブサーバーのみがNNMiサービスを実行します。

アクティブおよびスタンバイNNMi管理サーバーは、各NNMi管理サーバーのハートビートを監視するクラスターの一部です。 アクティブサーバーに障害が発生し、そのハートビートが消失すると、スタンバイサーバーがアクティブサーバーになります。

アプリケーションフェイルオーバーが正しく機能するには、NNMi管理サーバーが以下の要件を満たしている必要があります。

- 両方のNNMi管理サーバーが同じ種類のオペレーティングシステムを実行している必要があります。たとえば、アクティブなサーバーがLinuxオペレーティングシステムを実行している場合、スタンバイサーバーもLinuxオペレーティングシステムを実行している必要があります。
- 両方のNNMi管理サーバーは同じバージョンのNNMiを実行している必要があります。たとえば、アクティブサーバーでNNMi 10.01を実行している場合、スタンバイサーバーでも同一のNNMiバージョンであるNNMi 10.01がインストールされている必要があります。NNMiパッチレベルについても、同一レベルのパッチが両サーバーに適用されている必要があります。
- 両方のNNMi管理サーバーのシステムパスワードが同一である必要があります。
- WindowsオペレーティングシステムのNNMiインストールでは、`%NnmDataDir%`および`%NnmInstallDir%`のシステム変数を両方のサーバーで同一の値に設定している必要があります。
- 両方のNNMi管理サーバーは同じデータベースを実行している必要があります。たとえば、両方のNNMi管理サーバーでOracleを実行しているか、両方のNNMi管理サーバーで組み込みデータベースを実行している必要があります。アプリケーションフェイルオーバー機能を使用する場合、種類の異なるデータベースを組み合わせて使用することはできません。
- 両方のNNMi管理サーバーのライセンス属性が同一である必要があります。たとえば、ノードカウントおよびライセンス取得済みの機能が同一である必要があります。
- NNMiが初回検出の高度なステージに入るまで、アプリケーションフェイルオーバーを有効にしないでください。詳細については、「検出の評価」(80ページ)を参照してください。

アプリケーションフェイルオーバーが正しく機能するには、アクティブサーバーとスタンバイサーバーは相互のネットワークアクセスに制限のないことが必要です。この条件を満たしたら、「アプリケーションフェイルオーバー用のNNMiのセットアップ」(145ページ)に示した手順を実行してください。詳細については、「NNMiおよびNNM iSPIのデフォルトポート」(506ページ)を参照してください。

注: NNMi (単品)、およびNNMiに同梱されているNNMi Advanced機能とNNM iSPI NET機能を購入した場合、アプリケーションフェイルオーバー環境で使用するためのライセンスには2つのタイプがあります。

- 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスをプライマリサーバーのIPアドレスと関連付けます。
- 非商用 - このライセンスは、アプリケーションフェイルオーバー環境で使用するために個別に購入されます。このライセンスをセカンダリ (スタンバイ) サーバーのIPアドレスと関連付けます。

注:同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。

指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、アップリケーションフェイルオーバーで使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。プライマリサーバーの物理IPア

ドレスに1つと、スタンバイサーバーの物理IPアドレスに1つの、2つのライセンスキーを取得します。

以下で入手できる各NNM iSPIのドキュメントも参照してください。
http://h20230.www2.hp.com/selfsolve/manuals。

注: ファイルをロックしたり、ネットワークのアクセスを制限したりするソフトウェアが原因で、NNMiの通信の問題が発生する場合があります。こうしたアプリケーションで、NNMiが使用するファイルとポートを無視するように設定します。

注: NNMiのインストールまたはアップグレード時に、NNMiインストールによってNNMiクラスター通信用のネットワークインタフェースが選択されます。通常、選択されたネットワークインタフェースは、システムの最初に非ループバックインタフェースになります。NNMiクラスターが設定された場合、選択されたインタフェースがその設定で使用されます。インタフェースを調整する必要がある場合は、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NnmDataDir%\conf\nnm\props\nms-cluster-local.properties`
   - Linuxの場合: `$NnmDataDir/conf/nnm/props/nms-cluster-local.properties`
2. 目的のインタフェースを指し示すように`com.hp.ov.nms.cluster.interface`パラメーターを調整します。

アプリケーションフェイルオーバー用のNNMiのセットアップ

注: NNMi (単品)、およびNNMiに同梱されているNNMi Advanced機能とNNM iSPI NET機能を購入した場合、アプリケーションフェイルオーバー環境で使用するためのライセンスには2つのタイプがあります。

- 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスをプライマリサーバーのIPアドレスと関連付けます。
- 非商用 - このライセンスは、アプリケーションフェイルオーバー環境で使用するために個別に購入されます。このライセンスをセカンダリ (スタンバイ) サーバーのIPアドレスと関連付けます。

指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、アップリケーションフェイルオーバーで使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。プライマリサーバーの物理IPア

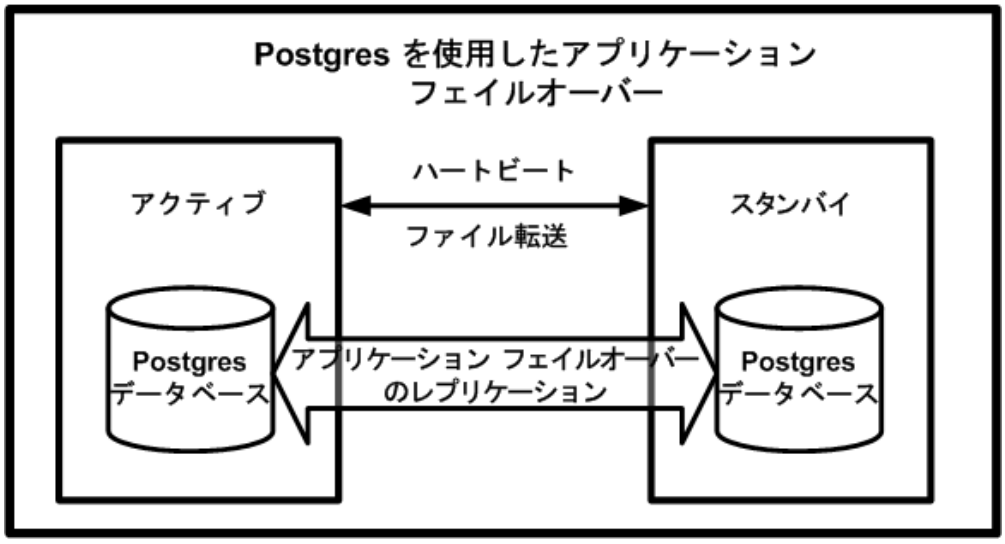
ドレスに1つと、スタンバイサーバーの物理IPアドレスに1つの、2つのライセンスキーを必ず取得します。

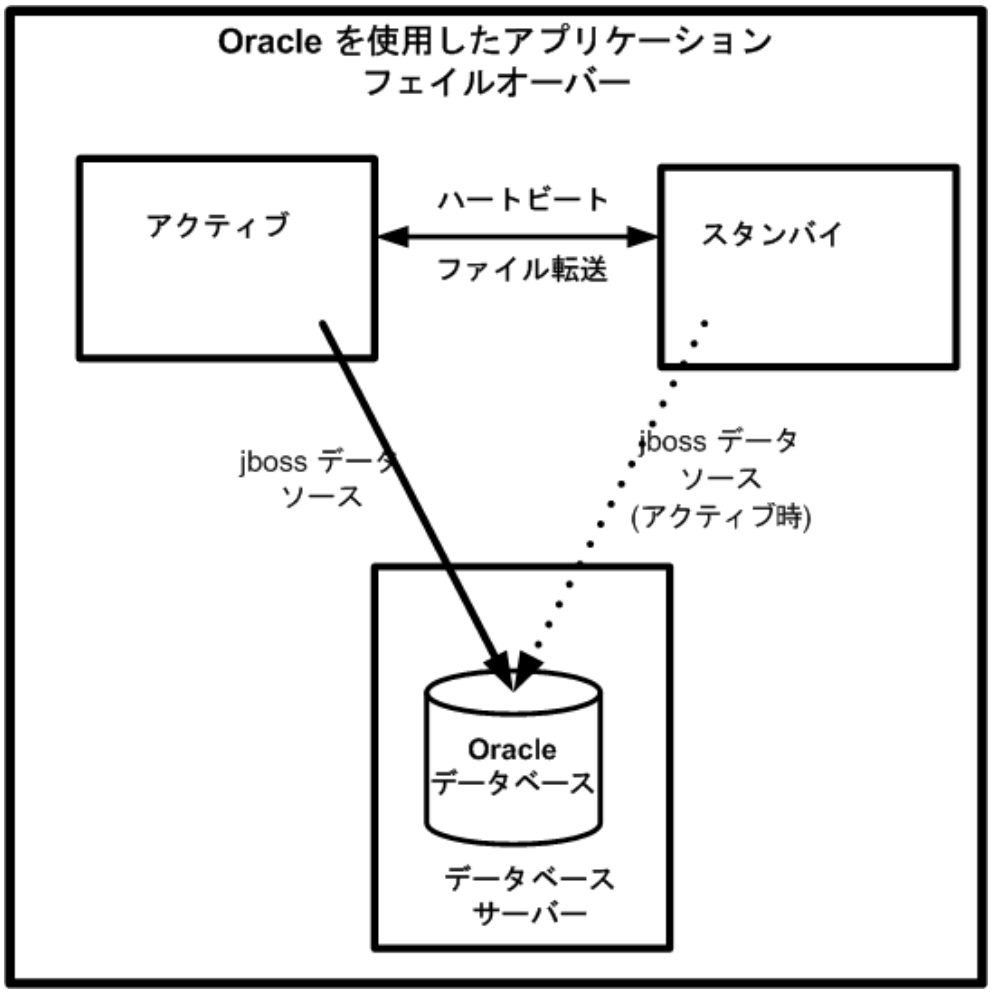
警告:同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。

以下で入手できる各NNM iSPIのドキュメントも参照してください。
http://h20230.www2.hp.com/selfsolve/manuals。

1. 以下の図で示すように、HP Network Node Manager i Softwareインタラクティブインストールガイドの説明に従ってアクティブなサーバー (サーバーX) とスタンバイサーバー (サーバーY) にNNMiをインストールします。

   NNMiでのアプリケーションフェイルオーバーのセットアップ





2. 「NNMiのライセンス」(312ページ)の説明に従って、サーバーXの各ライセンスに対し、サーバーYで使用するのに必要なライセンスを取得し、サーバーYにインストールします。
3. 各サーバーでovstopコマンドを実行してNNMiをシャットダウンします。

   注: Oracleデータベースでアプリケーションフェイルオーバーを使用している場合は、スタンバイサーバーのNNMiプロセスはすでに停止しています。

4. Oracleデータベースでアプリケーションフェイルオーバーを使用している場合、「アプリケーションフェイルオーバー構成のNNMiの手動設定」(499ページ)の設定手順を実行します。

NNMiクラスターセットアップウィザードを使用したクラスターの設定 (組み込みデータベースユーザーのみ)

NNMiクラスターセットアップウィザードは、アプリケーションフェイルオーバーで使用するNNMi内のクラスターの設定プロセスを自動化します。ウィザードでは、以下の操作ができます。

- クラスターノードの指定および検証を行う
- クラスターのプロパティおよびポートを定義する
- 両方のノードのnnm.keystoreおよびnnm.truststoreファイルの内容をマージして、1つのnnm.keystoreおよびnnm.truststoreファイルにする

1. サポートされるWebブラウザーに以下を入力して、クラスターセットアップウィザードを起動します。

   http://<NNMiserver>:<port>/cluster

   - <NNMiserver>は、NNMiホストの値です。
   - <port>は、NNMiポートの値です。

2. システムの [**ユーザー名**] と [**パスワード**] を入力して [**ログイン**] ボタンをクリックし、NNMiにサインインします。
3. [**ローカルホスト名**] と [**リモートクラスターノード**] の値を入力してクラスターノードを定義し、[**次へ**] をクリックします。
4. [通信結果] ページで、通信の検証結果を確認します。エラーが発生した場合は [**前へ**] をクリックして問題を修正します。エラーが発生しなかった場合は [**次へ**] をクリックします。

   緑のステータスメッセージは、リモートクラスターノードに正常に接続されたことを示します。
5. [クラスタープロパティを定義] ページで、[**クラスター名**] を入力して [**バックアップ周期 (時間)**] を定義します。次に自動フェイルオーバーを有効にするかどうかを指定します。[**次へ**] をクリックします。
6. [クラスターポートを定義] ページで、[**開始クラスターポート**] と [**ファイル転送ポート**] の値を入力します。

   注: NNMiクラスターでは、[**開始クラスターポート**] で始まる4個の連続したポートが使用されます。
7. [**次へ**] をクリックします。
8. 入力した情報の概要を確認します。戻って設定情報を変更する場合は [**前へ**] をクリックします。変更しない場合は [**コミット**] をクリックしてクラスター設定を保存します。

   最後の概要は、情報が設定ファイルに正常に書き込まれたことを示します。
9. 両方のノードでovstopコマンドを実行して、両方のノードのNNMiをただちに停止します。
10. 両方のノードでnnmclusterコマンドを実行して、2つのノードをクラスター構成にできることを確認します。ノードをクラスター構成にできない場合は、「アプリケーションフェイルオーバー構成のNNMiの手動設定」(499ページ) を参照してください。
11. nnmclusterコマンドを使用して、アクティブにするノード上のNNMiを起動します。NNMiがACTIVEをレポートするまで待機します (「アプリケーションフェイルオーバー構成のNNMiの手動設定」(499ページ) を参照)。
12. ovstartコマンドを使用して、スタンバイノードを起動します。

クラスター通信の設定 (省略可能)

インストール時に、NNMiはシステム上のすべてのネットワークインタフェースカード (NIC) に対してクエリーを実行し、クラスター通信に使用するNICを特定します (使用可能な最初のNICが選択されます)。システムに複数のNICが存在する場合、以下の手順を実行して、nnmcluster操作に使用するNICを選択できます。

1. `nnmcluster -interfaces`を実行して、使用可能なすべてのインタフェースをリスト表示します。詳細については、nnmclusterのリファレンスページ、またはUNIXのマンページを参照してください。
2. 以下のファイルを編集します。
   - Windowsの場合:

     %NnmDataDir%\conf\nnm\props\nms-cluster-local.properties

   - Linuxの場合:

     $NnmDataDir/conf/nnm/props/nms-cluster-local.properties

3. 以下のような内容のテキストが含まれる行を見つけます。

   com.hp.ov.nms.cluster.interface =<value>

4. 必要に応じて値を変更します。

   注: インタフェース値は有効なインタフェースに関係している必要があります。そうでない場合、クラスターを起動できないことがあります。

5. `nms-cluster-local.properties`ファイルを保存します。

   注: `com.hp.ov.nms.cluster.interface` パラメーターにより、NNMi管理者はnnmcluster通信に使用する通信インタフェースを選択できます。このインタフェースは、埋め込みデータベースまたはSecure Sockets Layer通信に使用するインタフェースではありません。

   注: アプリケーションフェイルオーバーが特定のインタフェースによって遵守されるように通信を設定するには、ホスト名を使用する場合とは異なり、`com.hp.ov.nms.cluster.member.hostnames`パラメーターのIPアドレスを使用します。以下のファイルで`com.hp.ov.nms.cluster.member.hostnames`パラメーターを設定します。

   Windowsの場合:

   %NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties

   Linuxの場合:

   $NnmDataDir/shared/nnm/conf/props/nms-cluster.properties

アプリケーションフェイルオーバー機能の使用

両方のNNMi管理サーバーでクラスターマネージャーを実行した後で (アクティブノードとスタンバイノード)、クラスターマネージャーを使用してクラスターのステータスを表示できます。クラスターマネージャーには3つのモードがあります。

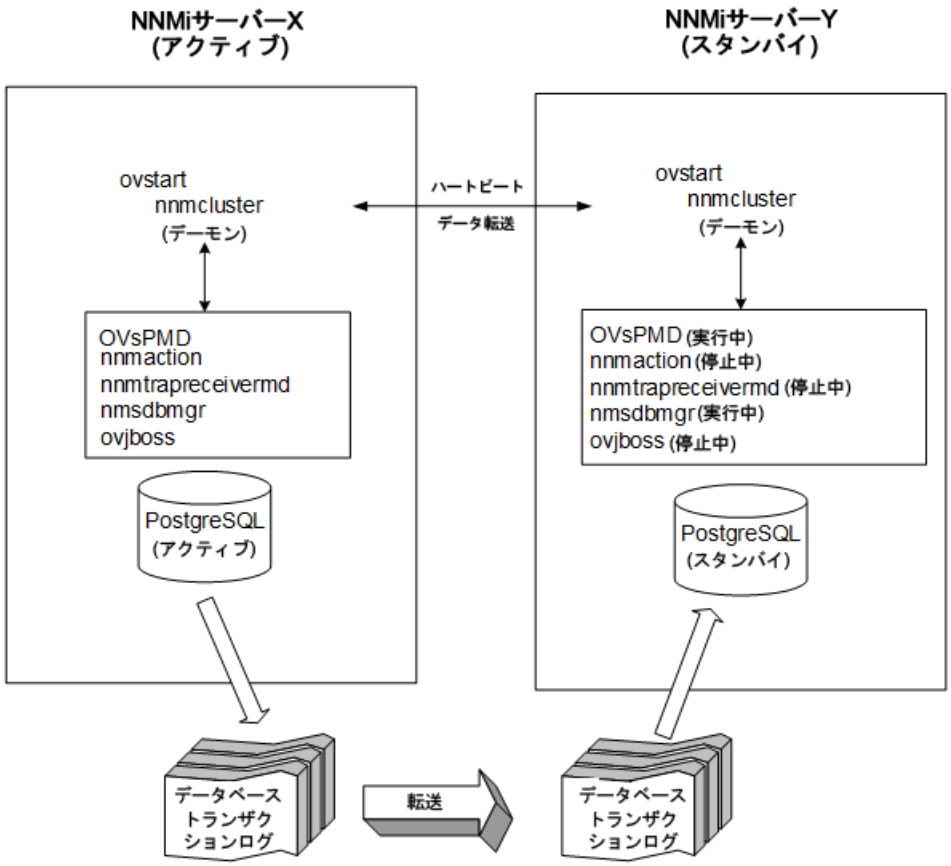
- **デーモンモード**:クラスターマネージャーのプロセスはバックグラウンドで実行し、**ovstop**および**ovstart**コマンドを使用してNNMiサービスを開始および停止します。
- **インタラクティブモード**:クラスターマネージャーは、NNMi管理者がクラスターの属性を表示および変更できるインタラクティブセッションを実行します。たとえば、NNMi管理者はこのセッションを使用して、アプリケーションフェイルオーバー機能を有効または無効にしたり、デーモンプロセスをシャットダウンしたりできます。
- **コマンドラインモード**: NNMi管理者は、コマンドプロンプトでクラスターの属性を表示および変更します。

詳細については、nnmclusterのリファレンスページ、またはLinuxのマンページを参照してください。

組み込みデータベースを使用したアプリケーションフェイルオーバーの動作

以下の図に、組み込みデータベースを使用した2つのNNMi管理サーバーのアプリケーションフェイルオーバー設定を示します。この章の以降のセクションについて、この図を参照してください。

アプリケーションフェイルオーバーの設定 (組み込みデータベース)



注: スタンバイサーバーをクラスターから削除し、そのサーバーをスタンドアロンサーバーとして実行してからクラスターに戻すと、データベースエラーが発生する場合があります。この場合、nnmcluster dbsyncコマンドをコマンドラインから実行します。`nnmcluster -dbsync`

NNMiには、アプリケーションフェイルオーバー内にストリーミングレプリケーション機能が含まれており、スタンバイサーバーとアクティブなサーバーが同期した状態のまま、データベーストランザクションがアクティブなサーバーからスタンバイサーバーに送信されます。これにより、(以前のバージョンのNNMiのように) フェイルオーバーでデータベーストランザクションログをスタンバイサーバーにインポートする必要がなくなり、スタンバイサーバーがアクティブサーバーを引き継ぐのに要する時間が大幅に短縮されます。この機能には、データベースバックアップファイルが必要な場合にのみノード間で送信されるという利点もあり、データベーストランザクションファイルの通常の転送で、大きなデータベースバックアップファイルを送信する頻度が少なくなります。

注: アクティブノードとスタンバイノードの両方で、ファイアウォールが有効になっている場合、組み込みデータベースに使用しているポート (デフォルトではポート5432) が開いていることを確認します。このポートは以下のファイルで設定されます。

Windowsの場合: `%NNM_CONF%\nnm\props\nms-local.properties`

Linuxの場合: `$NNM_CONF/nnm/props/nms-local.properties`

アクティブノードとスタンバイノードの両方を開始すると、スタンバイノードはアクティブノードを検知してアクティブノードにデータベースのバックアップをリクエストしますが、NNMiサービスは開始しません。このデータベースのバックアップは1つのJava-ZIPファイルとして保存されます。すでにスタンバイノードに以前のクラスター接続から得たZIPファイルがあり、NNMiが、そのファイルとアクティブサーバーの同期が確認された場合は、ファイルは再送されません。

アクティブノードとスタンバイノードの両方が実行している間、アクティブノードは定期的にデータベースのトランザクションログをスタンバイノードに送信します。`nms-cluster.properties`ファイルの`com.hp.ov.nms.cluster.timeout.archive`パラメーターの値を変更すると、このデータの転送頻度を変更できます。これらのトランザクションログはスタンバイノードに蓄積されるため、スタンバイからアクティブになったときにすぐに利用することができます。

スタンバイノードがアクティブノードからデータベースの完全バックアップを受信すると、その情報を組み込みデータベースに取り込みます。また、`recovery.conf`ファイルを作成して、受信したすべてのトランザクションログを取り込んでからでないと他のサービスがデータベースを使用できないことを組み込みデータベースに知らせます。

何らかの理由でアクティブノードが利用できなくなると、スタンバイノードはNNMiサービスを開始する**ovstart**コマンドを実行してアクティブになります。スタンバイNNMi管理サーバーは、残りのNNMiサービスを開始する前に、トランザクションログをインポートします。

アクティブNNMiシステムに障害が発生すると、スタンバイシステムは、ディスカバリとポーリングアクティビティを開始します。このトランジションによって、障害が発生したシステムの診断と修理を行う間、NNMiはネットワークを監視およびポーリングし続けます。

注:

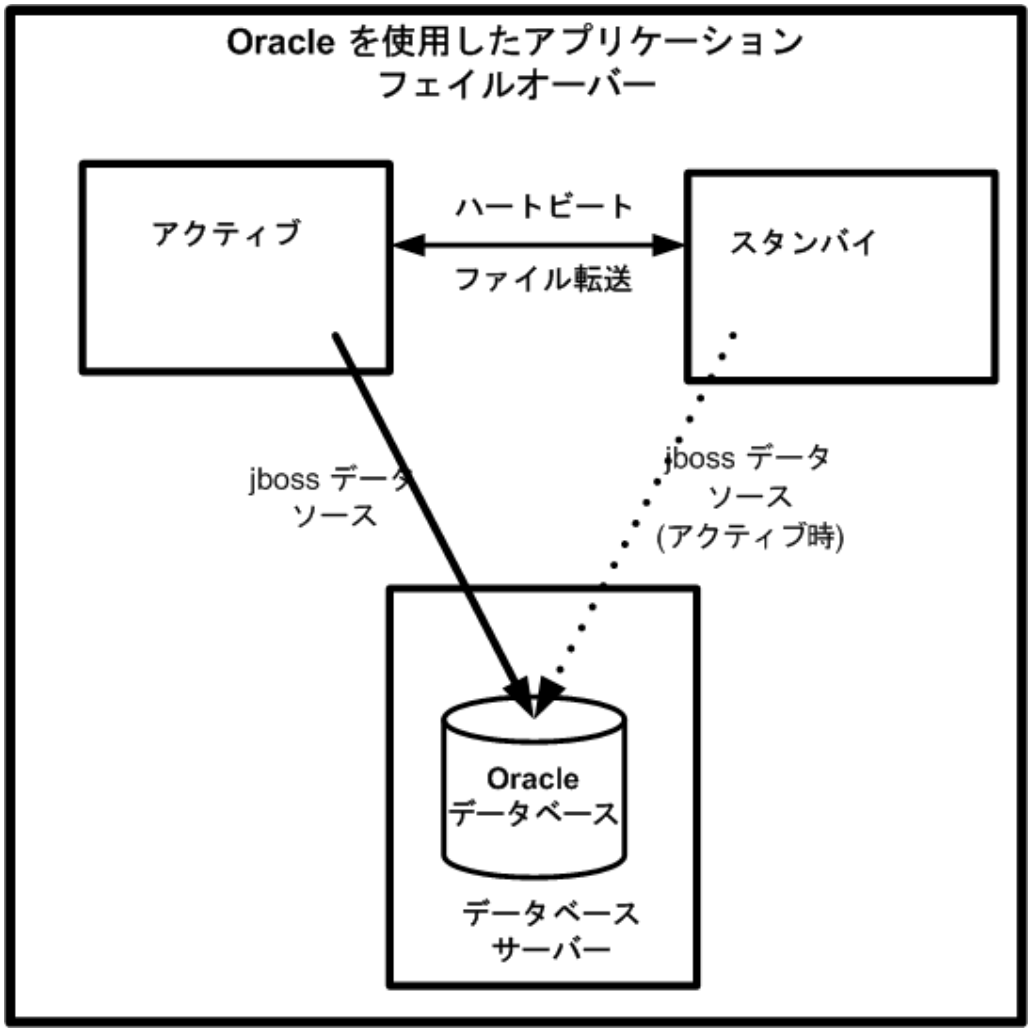
- NNMiでは、アプリケーションフェイルオーバーの後にトポロジ、状態、およびステータスが自動的に再同期されます。
- 再同期中にNNMiを停止しないでください。

  再同期を確実に完了するには、アプリケーションフェイルオーバーの後でNNMiを数時間実行し続けます。実際の所要時間は、ノード数、状態変化の量、および再同期中に受信されたトラップデータによって異なります。
- 再同期が完了する前にNNMiを停止する必要がある場合は、再同期をもう一度実行して完了する必要があります。
- 管理サーバー全体の再同期を手動で実行するには、`nnmnoderediscover.ovpl -all -fullsync`を実行します。

Oracleデータベースを使用したアプリケーションフェイルオーバーの動作

以下の図に、Oracleデータベースを使用した2つのNNMi管理サーバーのアプリケーションフェイルオーバー設定を示します。この章の以降のセクションについて、この図を参照してください。

アプリケーションフェイルオーバーの設定 (Oracleデータベース)



何らかの理由でアクティブノードが利用できなくなると、スタンバイノードはNNMiサービスを開始するovstartコマンドを実行してアクティブになります。

アクティブNNMiシステムに障害が発生すると、スタンバイシステムは、ディスカバリとポーリングアクティビティを開始します。このトランジションによって、障害が発生したシステムの診断と修理を行う間、NNMiはネットワークを監視およびポーリングし続けます。

注:

- NNMiではアプリケーションフェイルオーバー後に再同期が行われるためステータスおよびインシデントの更新が遅延する可能性がある。
- この再同期中に以下のメッセージが表示されても問題はありません。

  Causal Engineのキューサイズが大きいため、ステータスおよびインシデントの更新が遅延しています。これは、アップグレード、アプリケーションフェイルオーバー、バックアップの

復元の後に再同期が行われる場合、または手動で再同期を実行する場合に発生する可能性があります。

- この再同期中にNNMiを停止しないでください。再同期を確実に行うには、アプリケーションフェイルオーバー後に数時間NNMiが実行されている必要があります。

アプリケーションフェイルオーバーの例

アクティブなNNMi管理サーバーがハートビートを送信しなくなり、フェイルオーバーが発生してしまう原因はいくつかあります。

- シナリオ1:アクティブNNMi管理サーバーに障害が発生した。
- シナリオ2:システム管理者がアクティブなNNMi管理サーバーをシャットダウンまたはリブートした。
- シナリオ3: NNMi管理者がクラスターをシャットダウンした。
- シナリオ4:アクティブNNMi管理サーバーとスタンバイの間のネットワーク接続に障害が発生した。

  例4では、両方のNNMi管理サーバーがアクティブな状態で稼働します。ネットワークデバイスが復旧すると、2つのNNMi管理サーバーは自動的にネゴシエーションしてアクティブノードとして稼働するサーバーを決定します。

その他のovstartおよびovstopオプション

アプリケーションフェイルオーバーが設定されたNNMi管理サーバーで**ovstop**コマンドおよび**ovstart**コマンドを使用した場合、NNMiは以下のコマンドを実行します。

- ovstart:**nnmcluster -daemon**
- ovstop:**nnmcluster -disable -shutdown**

注: ovstopコマンドを実行すると、NNMiはスタンバイノードにフェイルオーバーしません。**ovstop**コマンドは、メンテナンスによる一時的な停止をサポートするように設計されています。フェイルオーバーを手動で行うには、**ovstop**コマンドに**-failover**オプションを使用します。詳細については、ovstopのリファレンスページ、またはLinuxのマンページを参照してください。

ovstopコマンドに使用する以下のオプションは、アプリケーションフェイルオーバークラスターに構成されたNNMi管理サーバーで使用します。

- **ovstop -failover**:ローカルのデーモンモードのクラスタープロセスを停止し、スタンバイNNMi管理サーバーに強制的にフェイルオーバーします。以前にフェイルオーバーモードが無効にされている場合は、このコマンドで有効になります。このコマンドは次のものと同等です。**nnmcluster -enable -shutdown**

- **ovstop -nofailover**: フェイルオーバーモードを無効にし、ローカルのデーモンモードのクラスタープロセスを停止します。フェイルオーバーは行われません。このコマンドは次のものと同等です。**nnmcluster -disable -shutdown**
- **ovstop -cluster**: アクティブノードとスタンバイノードを停止し、これらをクラスターから削除します。このコマンドは次のものと同等です。**nnmcluster -halt**

注: Linuxオペレーティングシステムを実行しているNNMi管理サーバーで**shutdown**コマンドを実行すると、**ovstop**コマンドが自動的に実行され、アプリケーションフェイルオーバーが無効になります。これが最適な設定ではない場合もあります。メンテナンス中にアプリケーションフェイルオーバーを制御するには、shutdownコマンドを実行する前に、**nnmcluster -acquire**コマンドと**nnmcluster -relinquish**コマンドを使用してアクティブノードとスタンバイノードを目的の動作に設定します。詳細については、nnmclusterのリファレンスページ、またはLinuxのマンページを参照してください。

アプリケーションフェイルオーバーのインシデント

nnmcluster**プロセスまたは****nnmcluster**コマンドを使用するユーザーが、ノードをアクティブとして開始すると、NNMiではそのたびに以下のいずれかのインシデントが生成されます。

- NnmClusterStartup: NNMiクラスターは、アクティブノードがない状態で開始されました。したがって、このノードはアクティブ状態で起動されました。このインシデントの重大度は「正常域」です。
- NnmClusterFailover: NNMiクラスターでアクティブノードの障害が検出されました。そのため、スタンバイノードがアクティブノードになり、そのノードでNNMiサービスが開始されました。このインシデントの重大度は「重要警戒域」です。

フェイルオーバー後、元の設定に戻る

アクティブノードで障害が発生し、スタンバイノードがアクティブノードとして機能している場合、以前のアクティブノードで問題を解決した後で、元の設定に戻すことができます。

以下の手順を実行します。

1. 以前のアクティブノードで問題を解決します。
2. 目的のアクティブノードで以下のコマンドを実行し、元の設定に戻ります。

   nnmcluster -acquire

詳細については、nnmclusterのリファレンスページ、またはLinuxのマンページを参照してください。

NNM iSPIsおよびアプリケーションフェイルオーバー

NNMiと一緒にSmart Plug-in (iSPI) を導入する場合、以下の要件を満たすとiSPI用のアプリケーションフェイルオーバー機能を使用できます。

- NNM iSPIはNNMi管理サーバーで動作する。
- 組み込みデータベースのみ。NNM iSPIは、NNMiと同じ組み込みデータベースインスタンスを使用する。
- Oracleデータベースのみ。NNM iSPIは、NNMiが使っているものから、一意のOracleデータベースのインスタンスを使用する必要があります。

NNM iSPI Performance for MetricsおよびNNMi SPI Performance for Trafficには、この説明は該当しません。NNMiアプリケーションフェイルオーバー機能を設定する場合は、これらのiSPIを専用サーバーにインストールする必要があります。この場合、iSPIは、フェイルオーバーが発生すると、新しいNNMi管理サーバーに自動的に接続します。NNMiアプリケーションフェイルオーバー設定の一環として、クラスターの各NNMi管理サーバーに、NNM iSPI Performance for MetricsまたはNNMi SPI Performance for Traffic用のイネーブルメントスクリプトを実行します。

詳細については、NNM iSPI Performance for Metrics、NNMi SPI Performance for QA、またはNNMi SPI Performance for Trafficヘルプの「アプリケーションフェイルオーバーのサポート」を参照してください。

NNM iSPIのインストールに関する情報

アプリケーションフェイルオーバークラスターのすでに一部であるNNMi管理サーバーにNNM iSPIをインストールするには、以下の手順を実行します。

1. 万一に備えて、以降の操作を行う前に、アクティブおよびスタンバイNNMi管理サーバーの両方で、`nnmconfigexport.ovpl`スクリプトを実行します。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。
2. 万一に備えて、以降の操作を行う前に、アクティブおよびスタンバイNNMi管理サーバーのNNMiデータをバックアップします。詳細については、「バックアップ領域」(245ページ)を参照してください。
3. 組み込みデータベースのみ:万一に備えて、アクティブNNMi管理サーバーで`nnmcluster -dbsync`コマンドを実行し、コマンドが完了するまで待ちます。
4. スタンバイNNMi管理サーバーで、以下のコマンドを実行します。

   ```
   nnmcluster -shutdown
   ```

5. スタンバイNNMi管理サーバーの以下のファイルを編集します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
   - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
6. `com.hp.ov.nms.cluster.name`オプションをコメントアウトし、ファイルを保存します。

7. 以下のトリガーファイルを作成します。このファイルは、Postgresにスタンバイモードでの実行を中止し、完全に実行するように指示します。

   Windowsの場合: `%NnmDataDir%\tmp\postgresTriggerFile`

   Linuxの場合: `%NnmDataDir%/tmp/postgresTriggerFile`
8. スタンバイNNMi管理サーバーで**ovstart**コマンドを実行します。すると、スタンドアロン (クラスターに属しない) 状態のNNMiサービスが表示されます。
9. 『iSPIインストールガイド』の説明に従って、スタンバイNNMi管理サーバーにNNM iSPIをインストールします。
10. アクティブNNMi管理サーバーで**nnmcluster -halt**コマンドを実行します。
11. アクティブNNMi管理サーバーの以下のファイルを編集します。
    - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
    - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
12. `com.hp.ov.nms.cluster.name` オプションをコメントアウトし、ファイルを保存します。
13. アクティブなNNMi管理サーバーで**ovstart**コマンドを実行します。すると、スタンドアロン (クラスターに属しない) 状態のNNMiサービスが表示されます。
14. 『iSPIインストールガイド』の説明に従って、アクティブNNMi管理サーバーにNNM iSPIをインストールします。
15. アクティブおよびスタンバイNNMi管理サーバーの**両方**で、以下のファイルを編集します。
    - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
    - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
16. `com.hp.ov.nms.cluster.name`オプションをコメント解除し、各ファイルを保存します。
17. アクティブなNNMi管理サーバーで**ovstart**コマンドを実行します。
18. アクティブNNMi管理サーバーがクラスターの最初のアクティブノードになるまで数分待ちます。アクティブNNMi管理サーバーで**nnmcluster -display**コマンドを実行し、表示された結果で、`ACTIVE_NNM_STARTING`または`ACTIVE_`SomeOtherStateの「`ACTIVE`」という語を検索します。アクティブNNMi管理サーバーがアクティブノードであることを確認するまで手順20に進まないでください。
19. アクティブノードで、以下のコマンドを実行します。

    nnmcluster -dbsync
20. スタンバイNNMi管理サーバーで**ovstart**コマンドを実行します。

統合アプリケーション

HPソフトウェア製品または第三者の製品がNNMiに統合された場合、統合に対するNNMiアプリケーションフェイルオーバー機能の影響は、製品がNNMiと通信する方法によって異なります。詳細については、適切な統合ドキュメントを参照してください。

統合製品の設定にNNMi管理サーバーに関する情報が必要な場合は、以下の情報が適用されます。

- 将来的に必要であれば、統合する製品の設定でNNMi管理サーバーの情報を更新できます。詳細については、適切な統合ドキュメントを参照してください。
- 機能停止が一時的なものである場合、サーバーXが復旧した後に統合する製品の使用を再開始できます。サーバーXのサービスを復旧するには、以下の手順を実行します。

1. サーバーXで以下のコマンドを実行します。

   ```
   nnmcluster -daemon
   ```

   サーバーXがスタンバイ状態でクラスターに参加します。
2. サーバーXで以下のコマンドを実行します。

   ```
   nnmcluster -acquire
   ```

   サーバーXはアクティブ状態になります。

元のサーバーXがより長期に渡って機能停止となる可能性がある場合は、統合する製品内で、NNMi管理サーバーのIPアドレスを更新できます。[IPアドレス] フィールドの変更方法については、統合する製品のドキュメントを参照してください。

アプリケーションフェイルオーバーの無効化

以下の情報は、アプリケーションフェイルオーバーを完全に無効にする方法を説明しています。アプリケーションフェイルオーバークラスターに構成された、アクティブおよびスタンバイNNMi管理サーバーでのアクションを含め、以下の指示に従ってください。

注: NNMi (単品)、およびNNMiに同梱されているNNMi Advanced機能とNNM iSPI NET機能を購入した場合、アプリケーションフェイルオーバーで使用するためのライセンスには2つのタイプがあります。

- 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスをプライマリサーバーのIPアドレスと関連付けます。
- 非商用 - このライセンスは、アプリケーションフェイルオーバー環境で使用するために個別に購入されます。このライセンスをセカンダリ (スタンバイ) サーバーのIPアドレスと関連付けます。

指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、アップリケーションフェイルオーバーで使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。プライマリサーバーの物理IPアドレスに1つと、スタンバイサーバーの物理IPアドレスに1つの、2つのライセンスキーを取得します。

注:同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。

以下で入手できる各NNM iSPIのドキュメントも参照してください。
http://h20230.www2.hp.com/selfsolve/manuals。

1. アクティブNNMi管理サーバーで**nnmcluster -enable**コマンドを実行します。
2. アクティブNNMi管理サーバーで**nnmcluster -shutdown**コマンドを実行します。
3. 既存のスタンバイNNMi管理サーバーが新しくアクティブNNMi管理サーバーになるまで数分待ちます。
4. 新しいアクティブ (以前のスタンバイ) NNMi管理サーバーで**nnmcluster -display**コマンドを実行します。
5. 表示された結果で、ACTIVE_NNM_RUNNINGステータスを検索します。ACTIVE_NNM_RUNNINGステータスを確認できるまで、手順4を繰り返します。
6. 新しいアクティブ (以前のスタンバイ) NNMi管理サーバーで**nnmcluster -shutdown**コマンドを実行します。
7. DAEMONプロセスがなくなるまで、新しいアクティブ (以前のスタンバイ) で**nnmcluster -display**コマンドを繰り返し実行します。
8. クラスターに構成されている両方のNNMi管理サーバーで、以下のファイルを編集します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
   - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
9. 両方のNNMi管理サーバーのcom.hp.ov.nms.cluster.nameオプションをコメントアウトし、各ファイルを保存します。
10. 両方のNNMi管理サーバーの以下のファイルを編集します。
    - Windowsの場合: `%NnmDataDir%\shared\nnm\databases\Postgres\postgresql.conf`
    - Linuxの場合: `$NnmDataDir/shared/nnm/databases/Postgres/postgresql.conf`
11. 以下の行を削除します。これらは、アプリケーションフェイルオーバーにより自動的に追加されたものです。これらの行の例を以下に示します。サーバーによって、表示がやや異なります。

    ```
    # The following lines were added by the NNM cluster.
    archive_command = ...
    archive_timeout = 900
    max_wal_senders = 4
    archive_mode = 'on'
    wal_level = 'hot_standby'
    hot_standby = 'on'
    wal_keep_segments = 500
    listen_addresses = 'localhost,16.78.61.68'
    ```

    必ず変更を保存してください。
12. Windows NNMi管理サーバーの場合、Services(Local) コンソールに移動し、各サーバーで以下

の手順を実行します。

a. HP NNM Cluster Managerの [スタートアップの種類] を [無効] に設定します。

b. HP Openview Process Managerの [スタートアップの種類] を [自動] に設定します。

13. 以下のトリガーファイルを作成します。このファイルは、Postgresにスタンバイモードでの実行を中止し、完全に実行するように指示します。

Windowsの場合: %NnmDataDir%\tmp\postgresTriggerFile

Linuxの場合: $NnmDataDir/tmp/postgresTriggerFile

14. 以前のアクティブNNMi管理サーバーのみにovstartコマンドを実行します。アプリケーションフェイルオーバー構成では、このサーバーは恒久NNMiライセンスを取得しているNNMi管理サーバーです。

15. 以前のスタンバイサーバーで非商用ライセンスを使用している場合は、そのNNMi管理サーバーでovstartコマンドを実行しないでください。アプリケーションフェイルオーバー構成では、このサーバーは、非商用ライセンスを取得しているNNMi管理サーバーです。このNNMi管理サーバーをスタンドアロンサーバーとして実行するには、恒久ライセンスを購入し、インストールする必要があります。詳細については、「NNMiのライセンス」(312ページ)を参照してください。

16. 両方のNNMi管理サーバーが正常に開始したら、スタンバイおよびアクティブNNMi管理サーバーから以下のディレクトリを削除します。

- Windowsの場合: %NnmDataDir%\shared\nnm\databases\Postgres_standby
- Linuxの場合: $NnmDataDir/shared/nnm/databases/Postgres_standby

> **注:** このディレクトリはデフォルトのディレクトリで、nms-cluster.propertiesファイルにあるcom.hp.ov.nms.cluster.archivedirパラメーターの値です。この手順では、この値が変更されていないことを前提としています。nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.archivedirパラメーターの値を変更した場合は、変更後の新しい値に相当するディレクトリを削除します。

17. スタンバイおよびアクティブNNMi管理サーバーから以下のディレクトリを削除します。

- Windowsの場合: %NnmDataDir%\shared\nnm\databases\Postgres.OLD
- Linuxの場合: $NnmDataDir/shared/nnm/databases/Postgres.OLD

管理タスクおよびアプリケーションフェイルオーバー

以下は、NNMi管理サーバーへのパッチ適用や再起動などの管理タスクを行うときに、アプリケーションフェイルオーバーを効果的に管理する方法を説明しています。

NNMiフェイルオーバー環境の復元

異なるサーバーセット上のNNMiフェイルオーバー環境を復元するには、NNMiアクティブおよびスタンバイシステム両方のバックアップを取得し、必要なサーバー上でそれらを復元するとともに、所定のプロパティファイルでホスト名を変更する必要があります。

NNMiフェイルオーバー環境を復元するには、以下の手順を実行します。

1. ソースフェイルオーバー環境内のアクティブシステムとスタンバイシステムのすべてのNNMiデータの完全なオフラインバックアップを取得します。詳細については、「NNMiデータのバックアップ」(244ページ)を参照してください。
2. バックアップファイルを、それぞれの送り先であるアクティブシステムとスタンバイシステムにコピーします。
3. バックアップデータの場合と同じバージョンおよびパッチレベルのNNMiをインストールします。
4. アクティブシステムとスタンバイシステムの両方でNNMiデータを復元します。
   - **組み込みデータベース:** `nnmrestore.ovpl`コマンドを使用し、完全復元を実行します。詳細については、「バックアップと復元の方針」(250ページ)を参照してください。
   - **Oracleデータベース**:システムファイルのみを復元するには、次のような復元コマンドを使用します。詳細については、「ファイルシステムのファイルのみの復元」(251ページ)を参照してください。

     ```
     nnmrestore.ovpl -partial -source nnmi_backups\offline\<newest_backup>
     ```
5. アクティブおよびスタンバイNNMi管理サーバーの両方で、以下の手順を実行します。
   a. アクティブおよびスタンバイNNMiサーバーの両方のホスト名を確認します。
   b. 以下のファイルを開きます。
      - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
      - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
   c. アクティブノードおよびスタンバイノードのホスト名を`com.hp.ov.nms.cluster.member.hostnames`パラメーターに追加します。

      ```
      com.hp.ov.nms.cluster.member.hostnames = fqdn_for_active, fqdn_for_standby
      ```
6. セキュア通信用のSSL証明書を使用するようにNNMiフェイルオーバー環境を設定します。詳細については、「証明書の管理」(316ページ)を参照してください。

アプリケーションフェイルオーバーおよびNNMiパッチ

両方のNNMi管理サーバーで同じバージョンとパッチレベルのNNMiを実行している必要があります。アクティブおよびスタンバイのNNMi管理サーバーにパッチを追加するには、以下のいずれかの方法を使用します。

- 「アプリケーションフェイルオーバー用にパッチを適用する (アクティブとスタンバイの両方をシャットダウン)」(162ページ)

  ネットワーク監視が中断されても問題にならない場合は、この手順を使用してください。
- 「アプリケーションフェイルオーバー用にパッチを適用する (1つのアクティブNNMi管理サーバーを保持)」(164ページ)

  ネットワーク監視の中断を回避する必要がある場合は、この手順を使用してください。

アプリケーションフェイルオーバー用にパッチを適用する (アクティブとスタンバイの両方をシャットダウン)

この手順を実行すると、パッチプロセス中の一定期間、両方のNNMi管理サーバーが非アクティブになります。アプリケーションフェイルオーバーを設定しているNNMi管理サーバーにパッチを適用するには、以下の手順を実行します。

1. 万一に備えて、以降の操作を行う前に、アクティブおよびスタンバイNNMi管理サーバーの両方で、**nnmconfigexport.ovpl**スクリプトを実行します。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。
2. 万一に備えて、以降の操作を行う前に、アクティブおよびスタンバイNNMi管理サーバーのNNMiデータをバックアップします。詳細については、「バックアップ領域」(245ページ)を参照してください。
3. nms-cluster.propertiesファイルにcom.hp.ov.nms.cluster.nameプロパティ値があります。この値はパッチインストールの後で必要になります。このファイルは以下の場所にあります。
   - Windowsの場合: `%nnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
   - Linuxの場合: `$nnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
4. 万一に備えて、アクティブなNNMi管理サーバーで、以下の手順を実行します。
   a. **nnmcluster**コマンドを実行します。
   b. 組み込みデータベースのみ: NNMiに入力を求められたら、「**dbsync**」と入力し、[Enter]キーを押します。表示される情報に以下のメッセージが含まれていることを確認します。

      ACTIVE_DB_BACKUP:アクティブNNMi管理サーバーが新しいバックアップを実行しています。

      ACTIVE_NNM_RUNNING:アクティブNNMi管理サーバーが、前のメッセージによって示されたバックアップを完了しました。

      STANDBY_READY:スタンバイNNMi管理サーバーの前のステータスを示します。

      STANDBY_RECV_DBZIP:スタンバイNNMi管理サーバーは、アクティブNNMi管理サーバーから新しいバックアップを取得しています。

      STANDBY_READY:スタンバイNNMi管理サーバーは、アクティブNNMi管理サーバーで障害が発生した場合に実行できる準備が整っています。
5. アクティブなNNMi管理サーバーで**nnmcluster -halt**コマンドを実行します。アクティブおよびスタンバイNNMi管理サーバーのすべてのnnmclusterプロセスをシャットダウンします。
6. 両方のサーバーでnnmclusterノードが実行していないことを確認するには、アクティブおよびスタンバイNNMi管理サーバーの両方で以下の手順を実行します。

a. **nnmcluster**コマンドを実行します。
b. (SELF) とマークされているもの以外にnnmclusterノードが存在しないことを確認します。
c. **exit**または**quit**を実行して、手順aで開始したインタラクティブnnmclusterプロセスを停止します。

7. アクティブNNMi管理サーバーで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメーターをコメントアウトします。
   a. 以下のファイルを編集します。
      - Windowsの場合: %NNM_SHARED_CONF%\props\nms-cluster.properties
      - Linuxの場合: $NNM_SHARED_CONF/props/nms-cluster.properties
   b. com.hp.ov.nms.cluster.nameパラメーターをコメントアウトします。
   c. 変更を保存します。
8. パッチとともに提供された指示に従って、アクティブなNNMi管理サーバーにNNMiパッチを適用します。
9. アクティブNNMi管理サーバーで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメーターをコメント解除します。

   注: パッチのインストール時に、com.hp.ov.nms.cluster.nameプロパティ値がNNMiデフォルト値に置き換わります。com.hp.ov.nms.cluster.nameパラメーターが含まれる行をコメント解除した後、com.hp.ov.nms.cluster.nameプロパティ値を、パッチのインストール前に設定した値に置き換える必要もあります。

   a. 以下のファイルを編集します。
      - Windowsの場合: %NNM_SHARED_CONF%\props\nms-cluster.properties
      - Linuxの場合: $NNM_SHARED_CONF/props/nms-cluster.properties
   b. アクティブなNNMi管理サーバーで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメータをコメント解除します。
   c. com.hp.ov.nms.cluster.nameプロパティのデフォルト値を、パッチのインストール前にnms-cluster.propertiesで設定した名前に置き換えます。
   d. 変更を保存します。
10. アクティブなNNMi管理サーバーで**ovstart**コマンドを実行します。
11. NNMiコンソールの **[ヘルプ] > [システム情報]** ウィンドウにある **[製品]** タブで情報を表示し、アクティブなNNMi管理サーバーにパッチが正しくインストールされたことを確認します。
12. **nnmcluster -dbsync**コマンドを実行して新規バックアップを作成します。
13. 手順aから手順cに示されているように、スタンバイで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメーターをコメントアウトします。
14. NNMiパッチをスタンバイNNMi管理サーバーに適用します。
15. 手順aから手順dに示されているように、スタンバイNNMi管理サーバーで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメーターをコメント解除します。

16. スタンバイNNMi管理サーバーで**ovstart**コマンドを実行します。
17. NNMi SPI Performance for QA、NNM iSPI Performance for Metrics、またはNNMi SPI Performance for Trafficをインストールし、アプリケーションフェイルオーバー機能を使用しており、さらに上記のパッチプロセスを完了した場合は、アクティブおよびスタンバイNNMi管理サーバーの各NNM iSPIにNNM iSPIイネーブルメントスクリプトを実行します。

アプリケーションフェイルオーバー用にパッチを適用する(1つのアクティブNNMi管理サーバーを保持)

この手順を実行すると、パッチプロセスの間、1つのNNMi管理サーバーが常にアクティブになります。

注: このプロセスでは、ネットワークが継続的に監視されますが、NNMiでパッチプロセス中に生じたトランザクションログは失われます。

アプリケーションフェイルオーバーを設定しているNNMi管理サーバーにNNMiパッチを適用するには、以下の手順を実行します。

1. 万一に備えて、以降の操作を行う前に、アクティブおよびスタンバイNNMi管理サーバーの両方で、**nnmconfigexport.ovpl**スクリプトを実行します。詳細については、「ベストプラクティス:既存の設定を保存する」(33ページ)を参照してください。
2. 万一に備えて、以降の操作を行う前に、アクティブおよびスタンバイNNMi管理サーバーのNNMiデータをバックアップします。詳細については、「バックアップ領域」(245ページ)を参照してください。
3. nms-cluster.propertiesファイルにcom.hp.ov.nms.cluster.nameプロパティ値があります。この値はパッチインストールの後で必要になります。このファイルは以下の場所にあります。

   Windowsの場合: %nnmDataDir%\shared\nnm\conf\props\nms-cluster.properties

   Linuxの場合: $nnmDataDir/shared/nnm/conf/props/nms-cluster.properties
4. ノードのいずれかで**nnmcluster**コマンドを実行します。
5. 前の手順で2つのデータベースの同期に使用したNNMi管理サーバーで**dbsync**を入力します。

   注: **dbsync**オプションは、組み込みデータベースを使用するNNMi管理サーバーで機能します。Oracleデータベースを使用するように設定されたNNMi管理サーバーで、**dbsync**オプションを使用しないでください。
6. アクティブなNNMi管理サーバーがACTIVE_NNM_RUNNINGに戻り、スタンバイNNMi管理サーバーがSTANDBY_READYに戻るまで待機してから、次に進んでください。
7. **nnmcluster**を終了または中断させます。
8. 以下のコマンドをスタンバイNNMi管理サーバーで実行して、スタンバイNNMi管理サーバーのクラスターを停止します。

   nnmcluster -shutdown
9. 以下のプロセスとサービスが終了しているのを確認してから、次に進みます。

- postgres
- ovjboss

10. nnmclusterプロセスが終了しているのを確認してから、次に進みます。nnmclusterプロセスが終了していない場合、他に方法がなければ、nnmclusterプロセスを手動で強制終了します。
11. スタンバイNNMi管理サーバーで、以下のファイルを編集します。

    Windowsの場合: %nnmDataDir%\shared\nnm\conf\props\nms-cluster.properties

    Linuxの場合: $nnmDataDir/shared/nnm/conf/props/nms-cluster.properties
12. 行の先頭に#を入れてクラスター名をコメントアウトし、変更を保存します。

    #com.hp.ov.nms.cluster.name = NNMicluster
13. スタンバイNNMi管理サーバーにNNMiパッチをインストールします。
14. この時点で、スタンバイNNMi管理サーバーはパッチが適用済みで停止中、アクティブなNNMi管理サーバーはパッチが未適用で実行中です。アクティブなNNMi管理サーバーを停止し、ただちにスタンバイNNMi管理サーバーをオンラインに戻してネットワークを監視させます。
15. アクティブなNNMi管理サーバーで以下のコマンドを実行して、アクティブなNNMi管理サーバーのクラスターをシャットダウンします。

    nnmcluster -halt
16. nnmclusterプロセスの終了を確認します。数分以内に終了しない場合は、nnmclusterプロセスを手動で終了してください。
17. スタンバイNNMi管理サーバーで、nms-cluster.propertiesファイルからクラスター名をコメント解除します。

    注: パッチのインストール時に、com.hp.ov.nms.cluster.nameプロパティ値がNNMiデフォルト値に置き換わります。com.hp.ov.nms.cluster.nameパラメーターが含まれる行をコメント解除した後、com.hp.ov.nms.cluster.nameプロパティ値を、パッチのインストール前に設定した値に置き換える必要もあります。

    a. 以下のファイルを編集します。
       - Windowsの場合: %NNM_SHARED_CONF%\props\nms-cluster.properties
       - Linuxの場合: $NNM_SHARED_CONF/props/nms-cluster.properties
    b. アクティブなNNMi管理サーバーで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメータをコメント解除します。
    c. com.hp.ov.nms.cluster.nameプロパティのデフォルト値を、パッチのインストール前にnms-cluster.propertiesで設定した名前に置き換えます。
    d. 変更を保存します。
18. 以下のコマンドをスタンバイNNMi管理サーバーで実行して、スタンバイNNMi管理サーバーのクラスターを起動します。

    nnmcluster -daemon
19. アクティブなNNMi管理サーバーにNNMiパッチをインストールします。

20. この時点で、以前のアクティブなNNMi管理サーバーはパッチが適用済みですが、オフラインです。以下の手順を実行して、(スタンバイNNMi管理サーバーとして) クラスターに復帰させます。
    a. アクティブなNNMi管理サーバーで、nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.nameパラメータをコメント解除します。
    b. com.hp.ov.nms.cluster.nameプロパティのデフォルト値を、パッチのインストール前にnms-cluster.propertiesで設定した名前に置き換えます。
    c. 以下のコマンドを使用して、アクティブなNNMi管理サーバーを起動します。
    nnmcluster -daemon
21. 進行状況を監視するには、アクティブとスタンバイの両方のNNMi管理サーバーで以下のコマンドを実行します。

    nnmcluster

    以前のアクティブNNMi管理サーバーが、以前のスタンバイNNMi管理サーバーからデータベースの取得を完了するまで待機します。
22. 以前のアクティブなNNMi管理サーバーにSTANDBY_READYが表示されたら、以前のアクティブなNNMi管理サーバーで以下のコマンドを実行します。
    nnmcluster -acquire
23. NNMi SPI Performance for QA、NNM iSPI Performance for Metrics、またはNNMi SPI Performance for Trafficをインストールし、アプリケーションフェイルオーバー機能を使用しており、さらに上記のパッチプロセスを完了した場合は、アクティブおよびスタンバイNNMi管理サーバーの各NNM iSPIにNNM iSPIイネーブルメントスクリプトを実行します。

アプリケーションフェイルオーバーおよびNNMi管理サーバーの再起動

スタンバイNNMi管理サーバーは、いつでも再起動でき、再起動に関する特別な指示はありません。スタンバイとアクティブの両方のNNMi管理サーバーを再起動する場合は、アクティブNNMi管理サーバーを先に再起動してください。

アクティブまたはスタンバイNNMi管理サーバーを再起動するには、以下の手順を実行します。

1. NNMi管理サーバーで**nnmcluster -disable**コマンドを実行し、アプリケーションフェイルオーバー機能を無効にします。
2. NNMi管理サーバーを再起動します。
    a. NNMi管理サーバーで**ovstop**コマンドを実行します。
    b. NNMi管理サーバーで**ovstart**コマンドを実行します。
3. NNMi管理サーバーで**nnmcluster -enable**コマンドを実行し、アプリケーションフェイルオーバー機能を有効にします。

> **注:** NNMiのTrapReceiverプロセス、およびそのフェイルオーバーとの関連に関する重要情報については、「NNMi NmsTrapReceiverプロセス」(274ページ) を参照してください。

通信障害後のアプリケーションフェイルオーバーの制御

2つのクラスターノード間の通信障害が解決すると、通信障害の前に最も長時間実行していた (以前にアクティブであった) NNMi管理サーバーがアクティブなサーバーとして指定されます。

アプリケーションフェイルオーバーおよび以前のデータベースバックアップから復旧 (組み込みデータベースのみ)

アクティブおよびスタンバイNNMi管理サーバーがアプリケーションフェイルオーバー構成の場合に、元のバックアップからNNMiデータベースを復旧するには、以下の手順を実行します。

1. アクティブNNMi管理サーバーで**nnmcluster -halt**コマンドを実行します。
2. アクティブおよびスタンバイNNMi管理サーバーの以下のディレクトリを削除または移動します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\databases\Postgres_standby`
   - Linuxの場合: `$NnmDataDir/shared/nnm/databases/Postgres_standby`
3. アクティブNNMi管理サーバーでデータベースを復元します。
   a. 以下のファイルのクラスター名をコメントアウトして変更します。
      - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
      - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props\nms-cluster.properties`
   b. 通常どおり、データベースを復旧します。「NNMiデータの復元」(247ページ)を参照してください。
   c. アクティブNNMi管理サーバーで**ovstop**コマンドを実行します。
   d. 以下のファイルでクラスター名をコメント解除して変更します。
      - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties`
      - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nms-cluster.properties`
4. アクティブなNNMi管理サーバーで**ovstart**コマンドを実行します。
5. アクティブNNMi管理サーバーが新しいバックアップを生成するまで待ちます。この手順が完了したことを確認するには、**nnmcluster -display**コマンドを実行し、`ACTIVE_NNM_RUNNING`メッセージを検索します。
6. スタンバイNNMi管理サーバーで**ovstart**コマンドを実行します。スタンバイNNMi管理サーバーは新しいバックアップをコピーして抽出します。この手順が完了したことを確認するには、**nnmcluster -display**コマンドを実行し、`STANDBY_READY`メッセージを検索します。

ネットワークレイテンシ/帯域に関する考慮

NNMiアプリケーションフェイルオーバーは、クラスターのノード間で継続的なハートビート信号を交換することによって機能します。これには、NNMi組み込みデータベース、データベーストランザクションログ、その他のNNMi設定ファイルなどのデータファイルの交換に使用されるネットワークチャネルが使用されます。HPは、WAN (広域ネットワーク) にNNMiアプリケーションフェイルオーバーを導入する場合、パフォーマンスが高く、レイテンシが低い接続を使用することをお勧めします。

NNMi組み込みデータベースは必ず圧縮されていますが、非常に容量が大きくなり、1GB以上に増大することがあります。また、NNMiは、ビルトインバックアップインターバル (設定パラメーター、デフォルトは6時間) の間に膨大な数のトランザクションログを生成します。各トランザクションログのサイズは数メガバイトから、最大16MBになることもあります。(これらのファイルは圧縮されています)。以下は、HPのテスト環境から収集されたデータの例です。

Number of nodes managed:15,000

Number of interfaces:100,000

Time to complete spiral discovery of all expected nodes:12 hours

Size of database:850MB (compressed)

During initial discovery:~10 transaction logs per minute (peak of ~15/min)

---

10 TxLogs/minute X 12 hours = 7200 TxLogs @ ~10MB = ~72GB

これでは、ネットワークで送信するにはデータ量が多すぎます。2つのノード間のネットワークがNNMiアプリケーションフェイルオーバーの帯域幅の要求に応じられない場合、スタンバイノードへのデータベースファイルの送信に遅延が発生してしまいます。このため、アクティブサーバーに障害が発生した場合、潜在的なデータ喪失の可能性が高くなります。

同様に、2つのノード間のネットワークのレイテンシが高いか信頼性が低い場合、ノード間で偽のハートビート喪失となります。たとえば、ハートビート信号がただちに応答しない場合に、スタンバイノードは、アクティブノードに障害が発生したと判断します。ハートビート喪失の検出に関与する要素にはいくつかあります。NNMiは、ネットワークがアプリケーションフェイルオーバーのデータ転送の要求に応答できる限り、偽のフェイルオーバー通知を回避します。

マルチサブネットNNMiアプリケーションフェイルオーバーに関するHPの検証では、アクティブサーバーおよびスタンバイサーバーは、それぞれ米国のコロラド州とヒューストンにあります。許容できる帯域幅とレイテンシにより、偽のフェイルオーバーは発生しませんでした。

アプリケーションフェイルオーバーとNNMi組み込みデータベース

アプリケーションフェイルオーバーは、NNMi 10.01の組み込みデータベースとOracleデータベースの両方で動作します。ところがOracleではデータベースがNNMi管理サーバーとは別のサーバーに存在し

ます。Oracleデータベースと連動するようにNNMiを設定すると、データベースのレプリケーションは行われません。このため、Oracleデータベースを使用すると、アプリケーションフェイルオーバーのネットワーク要求が減少します。Oracleでアプリケーションフェイルオーバーを使用しているとき、組み込みデータベースのアプリケーションフェイルオーバーを使用しているときと比較すると、ネットワークではネットワーク要求の1%未満しか使用されません。このセクションでは、組み込みデータベースを使用するアプリケーションフェイルオーバーに関連するNNMiトラフィック情報について説明します。

アプリケーションフェイルオーバーに組み込みデータベースを使用するようにNNMiを設定すると、NNMiは以下のように動作します。

1. アクティブノードがデータベースバックアップを実行し、1つのZIPファイルにデータを保存します。
2. NNMiは、ネットワークを通してこのZIPファイルをスタンバイノードに送信します。
3. スタンバイノードはZIPファイルを展開し、組み込みデータベースを設定して最初の起動でトランザクションログをインポートします。
4. アクティブノードの組み込みデータベースは、データベースアクティビティにより、トランザクションログを生成します。
5. アプリケーションフェイルオーバーでは、トランザクションログがネットワークを通してスタンバイノードに送信され、ディスクに蓄積されます。
6. スタンバイノードがアクティブになると、NNMiが起動して、データベースがネットワークを通してすべてのトランザクションログをインポートします。これにかかる時間は、ファイル数、およびそのファイルに保存されている情報の複雑さによって決まります (サイズが同程度でも、一部のファイルのインポートには別のファイルより時間がかかります)。
7. スタンバイノードがすべてのトランザクションログをインポートすると、データベースが使用可能になり、スタンバイノードは残りのNNMiプロセスを開始します。
8. 元のスタンバイノードがアクティブになり、手順1から処理がやり直されます。

アプリケーションフェイルオーバー環境でのネットワークトラフィック

アプリケーションフェイルオーバー環境では、NNMiはアクティブノードからスタンバイノードにネットワークを介して多くの項目を転送します。

- データベースアクティビティ: 1つのZIPファイルとしてのデータベースバックアップ。
- トランザクションログ。
- それぞれのアプリケーションフェイルオーバーノードが、他方のノードが動作していることを確認するための定期的なハートビート。
- ファイルがアクティブノードのものと同期していることをスタンバイノードが確認できるようにするファイル比較リスト。
- パラメーターの変更 (フェイルオーバーやその他の有効/無効) およびクラスターでのノードの追加や除外などの、その他のイベント。

最初の2つの項目により、アプリケーションフェイルオーバーで使用されるネットワークトラフィックの99%が生成されます。このセクションでは、この2つの項目について詳しく説明します。

データベースアクティビティ: NNMiはすべてのデータベースアクティビティのトランザクションログを生成します。データベースアクティビティには、NNMiのすべてが含まれます。このアクティビティには以下のデータベースアクティビティが含まれますが、その他にも含まれるものがあります。

- 新しいノードを検出する。
- ノード、インタフェース、VLAN、その他の管理対象オブジェクトに関する属性を検出する。
- 状態ポーリングとステータス変更。
- インシデント、イベント、根本原因分析。
- NNMiコンソールでのオペレーターのアクション。

データベースアクティビティを制御することはできません。たとえば、ネットワークが停止すると、NNMiは多くのインシデントとイベントを生成します。このインシデントとイベントにより、ネットワーク上のデバイスの状態ポーリングが開始され、NNMiでデバイスのステータスが更新されます。停止が復旧されると、ノード開始インシデントによってステータスがさらに変化します。このすべてのアクティビティにより、データベースのエントリが更新されます。

組み込みデータベース自体はデータベースアクティビティによって拡大しますが、時間の経過とともに拡大は穏やかになり、環境でのサイズは安定します。

データベーストランザクションログ: 組み込みデータベースは、空の16MBのファイルを作成してからデータベーストランザクション情報をそのファイルに書き込むことで動作します。NNMiは、15分が経過した時点か、16 MBのデータがファイルに書き込まれた時点のいずれかの早い時点でこのファイルを閉じて、アプリケーションフェイルオーバーで使用可能にします。つまり、完全にアイドル状態のデータベースにより、15分ごとに1つのトランザクションログファイルが生成されますが、このファイルは本質的に空です。アプリケーションフェイルオーバーでは、すべてのトランザクションログが圧縮され、空の16 MBのファイルは1 MB未満に圧縮されます。満杯の16 MBのファイルは約8 MBに圧縮されます。データベースアクティビティが多い期間は、それぞれのファイルがすぐに満杯になるため、アプリケーションフェイルオーバーによって短時間により多くのトランザクションログが生成されます。

アプリケーションフェイルオーバーのトラフィックテスト

以下のテストでは、1分ごとにおよそ2個のトランザクションログファイルが生成され、1つのファイルの平均ファイルサイズは7 MBになります。これは、それぞれのフェイルオーバーイベントで追加される5000個のノードの検出に関連するデータベースアクティビティによるものです。このテストケースのデータベースは、最終的に約1.1 GBで安定し (バックアップZIPファイルのサイズで測定)、ノードは31,000個、インタフェースは960,000個になります。

テストモード:最初の4時間でテスト担当者が5,000個のノードをNNMiにシードして、検出が安定するまで待機しました。4時間後、テスト担当者がフェイルオーバーを誘発しました (スタンバイノードがアクティブになり、以前のアクティブノードがスタンバイになりました)。テスト担当者はフェイルオーバー直後に約5,000個のノードをさらに追加し、また4時間待機してNNMiの検出プロセスを安定させてから、別のフェイルオーバーを誘発しました (以前のアクティブノードに戻りました)。テスト担当者は、フェイルオーバー間の時間を、4時間、6時間、2時間というよう変更して、このサイクル

を数回繰り返しました。テスト担当者は、それぞれのフェイルオーバーイベント後に、以下の項目を測定します。

- ノードが初めてアクティブになったときに作成されるデータベースバックアップZIPファイルのサイズ。
- トランザクションログ: ファイル総数、およびディスク容量の使用量。
- フェイルオーバーを誘発する直前のNNMiデータベースのノードとインタフェースの数。
- フェイルオーバーが完了するまでの時間。アクティブノードで`ovstop`コマンドを最初に実行してから、スタンバイノードが完全にアクティブになってNNMiが動作するまでの時間。

以下の表に、結果をまとめます。

アプリケーションフェイルオーバーのテスト結果

時間	DB.zip サイズ (MB)	トランザクションログの数	の数 (GB)	ノード	インタフェース	フェイルオーバーの時間 (分)
4	6.5	50	.3	5,000	15,000	5
8	34	500	2.5	12,000	222,000	10
12	243	500	2.5	17,000	370,000	25
16	400	500	3.5	21,500	477,000	23
20	498	500	3.5	25,500	588,000	32
26	618	1100	7.5	30,600	776,000	30
28	840	400	2.2	30,600	791,000	31
30	887	500	2.5	30,700	800,000	16

所見: NNMiがアクティブノードからスタンバイノードにファイルを転送する場合、転送は4時間ごとに平均で約5GB、連続スループットは約350KB/s (1秒あたりのキロバイト数) または2.8MB/s (1秒あたりのメガビット数) になっています。

注: このデータには、ハートビート、ファイル整合性チェック、その他のアプリケーションフェイルオーバー通信など、その他のアプリケーションフェイルオーバートラフィックは含まれていません。このデータでは、パケットヘッダーなどのネットワークI/Oのオーバーヘッドも除外されています。このデータには、ネットワークで移動する各ファイルの内容の実ネットワークペイロードのみが含まれます。

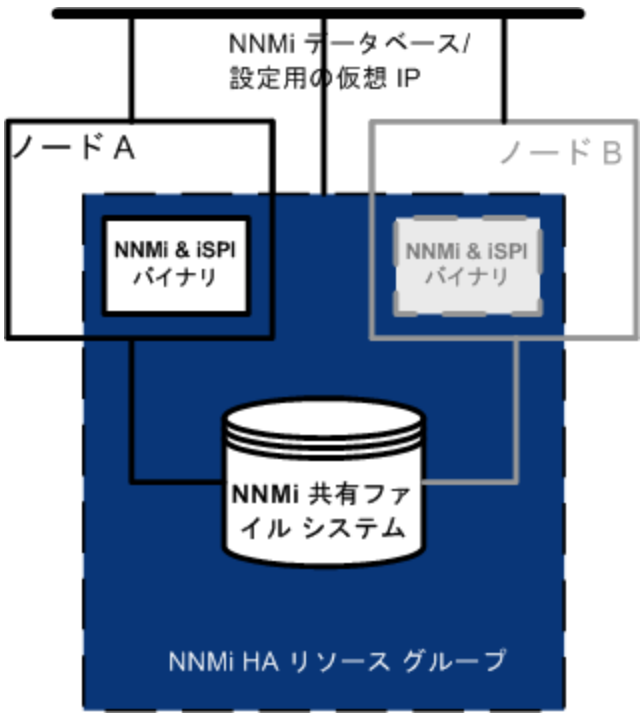
注: NNMiのアプリケーションフェイルオーバー環境で生成されるトラフィックは非常に爆発的です。アプリケーションフェイルオーバーでは、5分ごとにアクティブノードで新しいトランザクションログが識別され、スタンバイノードに送信されます。ネットワークの速度により、スタンバイノードではすべての新しいファイルが短時間で受信され、この5分間隔の残りの間、ネット

ワークは比較的にアイドル状態となることが多くなります。

アクティブノードとスタンバイノードがロールを切り替えるたびに(スタンバイノードがアクティブになり、アクティブノードがスタンバイになる)、新しいアクティブノードは完全なデータベースバックアップを生成し、ネットワークを介して新しいスタンバイノードに送信します。このデータベースバックアップも定期的に発生し、デフォルトで24時間ごとにバックアップされます。NNMiは、新しいバックアップを生成するたびに、このバックアップをスタンバイノードに送信します。この新しいバックアップがスタンバイノードで使用可能になると、その24時間にNNMiが生成したすべてのトランザクションログがデータベースに反映されて、フェイルオーバー時にインポートする必要がなくなるため、フェイルオーバー時間が短縮されます。

前述の情報により、組み込みデータベースを使用してアプリケーションフェイルオーバーでNNMiを使用するとき、フェイルオーバー後にネットワークがどのようなパフォーマンスになるかを理解できます。

高可用性クラスターにNNMiを設定する



高可用性(HA)とは、構成された動作中のハードウェアおよびソフトウェアの一部に障害が発生しても中断されないサービスを提供するシステムです。HAクラスターは、フェイルオーバー発生時の機能とデータの継続性を保証するために、協調して動作するハードウェアとソフトウェアのグループ化を定義します。

NNMiでは、別途購入が必要なHA製品を使って構成されるHAクラスター内でNNMiを実行する設定をサポートするようになりました。ほとんどのNNM Smart Plug-ins (iSPI) も、NNM iSPI NET診断サーバーを除いて、HAで実行できるようになります。

注: NNM iSPI NET診断サーバーはNNM iSPI NETおよびNNMi Ultimateと一緒にインストールできます。

注: 高可用性クラスターでNNMiを設定する場合、この章で述べている標準的な設定手順に従うことが重要です。非標準的な設定はサポートされていません。

この章では、HA環境で実行するようにNNMiを設定するためのテンプレートについて説明します。この章では、HA製品の詳細な設定手順については説明しません。NNMiに用意されているHA設定コマンドは、サポートされるHA製品用のコマンドに関するラッパーとなります。

注: NNMi HAコマンドを使用して、NNMi用にHAを適切に設定します。

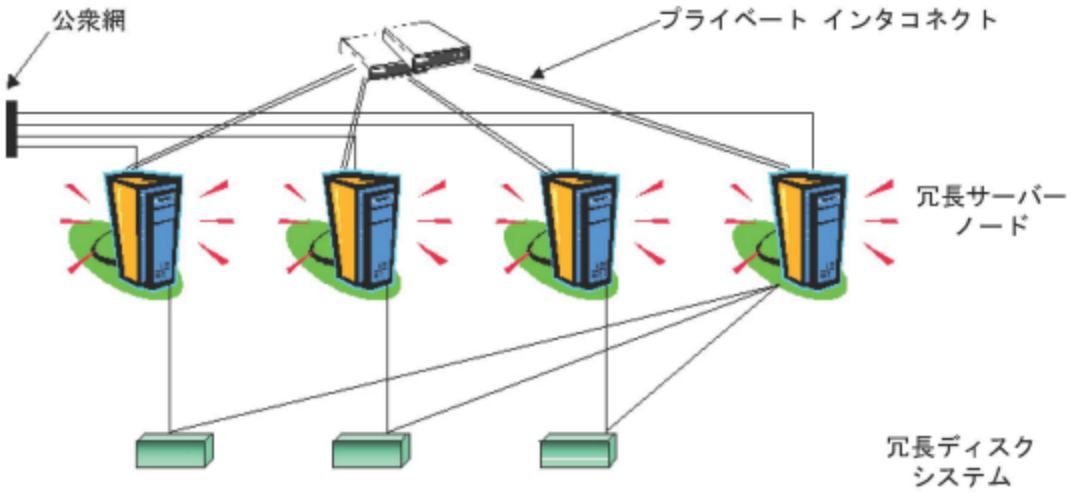
ヒント: NNMi管理サーバーにいずれかのNNM iSPIsをインストールする場合は、そのNNM iSPIsのマニュアルも参照してください。

この章には、以下のトピックがあります。

高可用性の概念

クラスターアーキテクチャーには、クラスター内の複数のノードのプロセスとリソース用の、単一のグローバルに首尾一貫した管理ビューが備わっています。以下の図に、クラスターアーキテクチャーの例を示します。

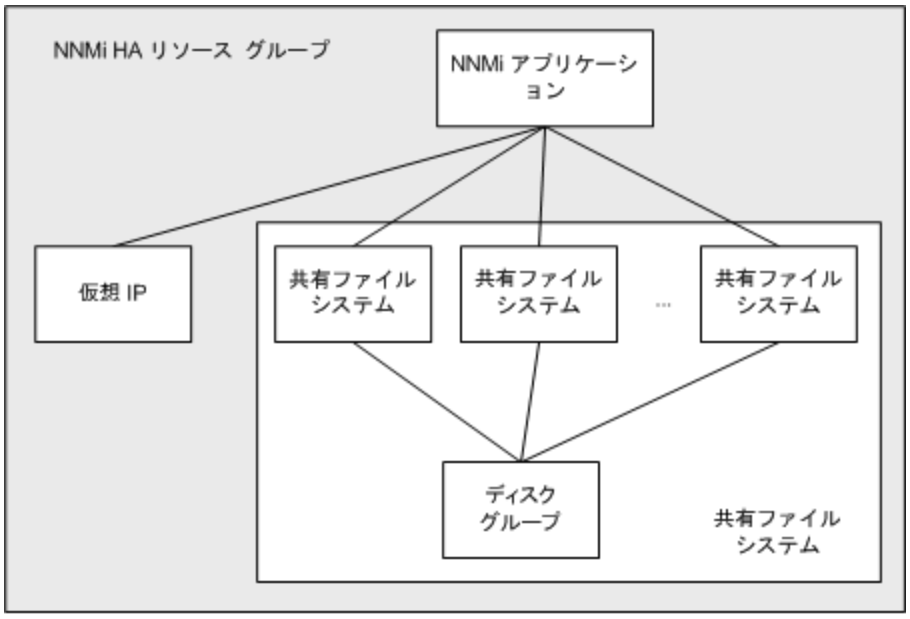
高可用性クラスターのアーキテクチャー



クラスター内の各ノードは、1つ以上のパブリックネットワークと1つのプライベートインタコネクト (クラスターノード間のデータ伝送用の通信チャネル) に接続されます。

Veritas Cluster Server、Microsoftフェイルオーバークラスタリング、Microsoft Cluster Serviceなどの最新のクラスター環境では、アプリケーションがリソースの複合体として表現され、単純な操作でアプリケーションをクラスター環境で実行することができます。リソースは、クラスター環境で動作するアプリケーションを表す、**HAリソースグループ**に構成されます。以下の図に、高可用性 (HA) リソースグループの例を示します。

典型的なHAリソースグループのレイアウト



このマニュアルでは、各種のクラスター環境内のリソースの集合を指すために、HAリソースグループという用語を使います。各HA製品では、HAリソースグループに対して、異なる名前が使われています。以下の表に、このドキュメントのHAリソースグループに相当する、サポート対象のHA製品で

使用されている用語をリストします。(各HA製品のサポート対象バージョンについては、NNMi対応マトリックスを参照してください)。

サポート対象のHA製品でHAリソースグループに相当する名前

HA製品	略語	HAリソースグループで対応する用語
Windows Serverフェイルオーバークラスタリング	WSFC	リソースグループ
Veritas Cluster Server	VCS	サービスグループ
Red Hat Cluster Suite	RHCS	サービス

高可用性の用語集

以下の表に、一般的な高可用性 (HA) の用語の定義をリストします。

一般的なHA用語

用語	説明
HAリソースグループ	クラスター環境内で (HA製品下で) 動作するアプリケーションです。HAリソースグループは、同時に、クラスター内のアプリケーションを表すクラスターオブジェクトでもあります。
ボリュームグループ	1つの大規模ストレージエリアを形成するよう設定された1つ以上のディスクドライブです。
論理ボリューム	ボリュームグループ内で、個別のファイルシステムまたはデバイススワップ空間として使われる任意のサイズの領域です。
プライマリクラスターノード	ソフトウェア製品が最初にインストールされるシステムであり、かつ、HAが最初に設定されるシステムです。 初期セットアップでは、共有ディスクはプライマリクラスターノードにマウントされます。 プライマリクラスターノードは、通常、最初のアクティブなクラスターノードになりますが、HAの設定完了後には、プライマリとしての役割を解除できます。HA設定を変更すると、他のノードをプライマリクラスターノードにできます。
セカンダリクラスターノード	プライマリクラスターノードでのHA設定の完了後に、HA設定に追加される任意のシステムです。
アクティブなクラスターノード	現在HAリソースグループを実行中のシステムです。
パッシブなクラスター	HA用に設定されているが、現在HAリソースグループを実行していないシ

一般的なHA用語 (続き)

用語	説明
ノード	ステムです。アクティブなクラスターノードで障害が発生すると、HAリソースグループはパッシブなクラスターノードの中で利用可能なノードにフェイルオーバーし、そのノードがアクティブなクラスターノードになります。

NNMi高可用性クラスターのシナリオ

注: NNMiでは、アプリケーションが複数のクラスターノードで実行できるクラスターをサポートしています。詳細については、nms-haのマンページおよびnnmdatareplicator.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

NNMi高可用性 (HA) 設定では、NNMiが各システムにインストールされ、HAリソースグループの一部になります。NNMiデータベースは独立したディスクにインストールされ、各システムで動作中のNNMiプログラムからアクセスされます。(任意の時点で共有ディスクにアクセスできるのは、アクティブなクラスターノードである1つのシステムだけです。)

このアプローチは、組み込み型のデータベースと他社製データベースソリューションの場合に有効です。

注: NNMiデータベースのバックアップスクリプトと復元スクリプトを実行できるのは、アクティブなクラスターノードだけです。

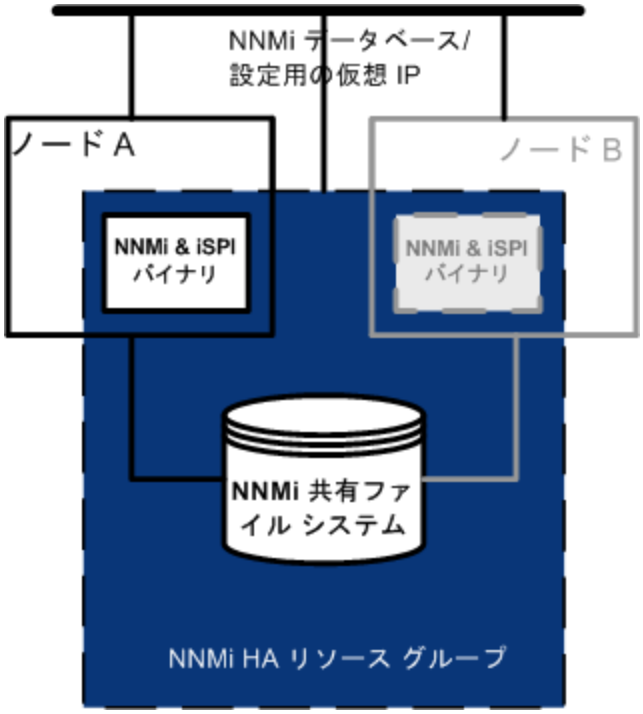
NNMiのみのシナリオ

NNMi HAクラスターのシナリオを以下に図示します。この図では、NNMi HAリソースグループは、NNMi HAクラスターと同義語です。

ノードAとノードBはどちらも、すべてのソフトウェアがインストールされたNNMi管理サーバーであり、そのシステムで実行するNNMiプログラムとNNM iSPIsがすべて含まれています。アクティブなクラスターノードが、共有ディスクのランタイムデータにアクセスします。他の製品は、HAリソースグループの仮想IPアドレスを使ってNNMiに接続します。

クラスターに3つ以上のNNMiノードがある場合は、追加ノードに以下の図のノードBと同様の設定を行います。

NNMi HAクラスター用の基本的なシナリオ

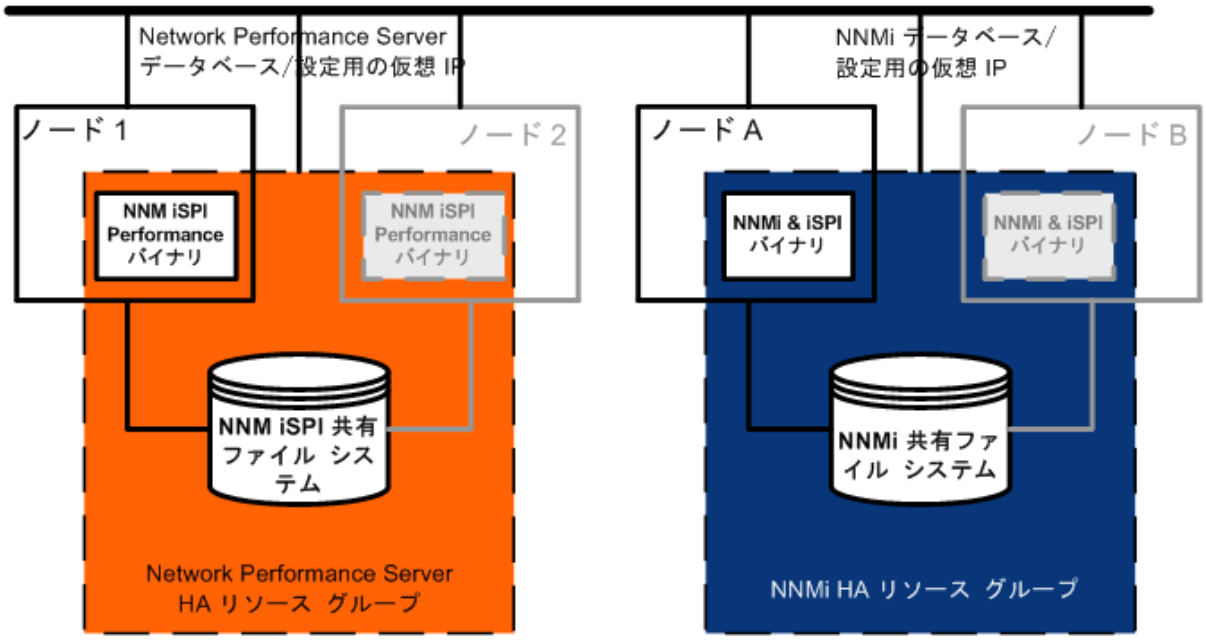


このシナリオの実装方法については、「高可用性用のNNMiの設定」と「高可用性用のNNM iSPIの設定」を参照してください。

スタンドアロンサーバーシナリオでの**NNMiおよびNNM Performance iSPI**

いずれかのNNM Performance iSPI製品をスタンドアロンサーバーで実行する場合は、以下の図に示すように、NNMi HAクラスター内で別個のHAリソースグループとして実行されるようこのNNM iSPIsを設定できます。NNMi HAリソースグループは、NNMiのみのシナリオで説明したものと同じです。

スタンドアロンサーバーでNNMiとNNM Performance iSPIを実行する場合のHA



このシナリオの実装方法については、「高可用性用のNNMiの設定」と「高可用性用のNNM iSPIの設定」を参照してください。

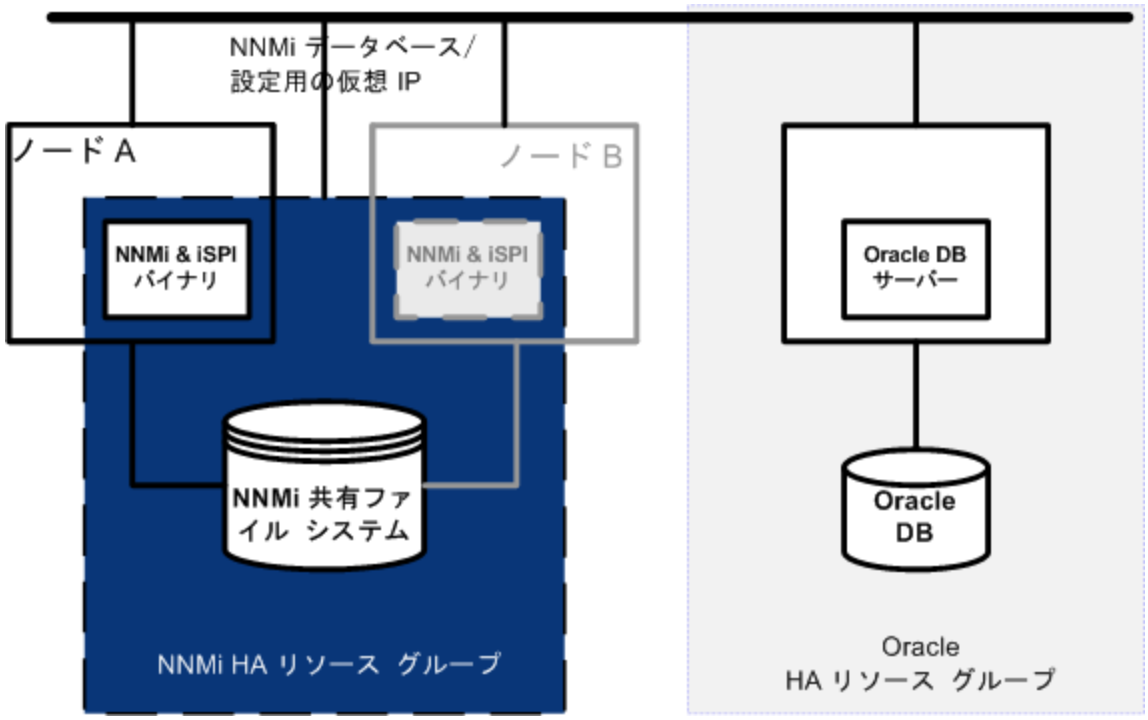
- スタンドアロンサーバーで実行されるNNM Performance iSPIのその他の選択肢は以下のとおりです。
- NNM Performance iSPIをHAを設定していない単一システムで実行します。このアプローチは、NNM iSPIsを評価する場合、あるいは、パフォーマンスデータが必ずしも必要ではない環境で使用します。
- NNM Performance iSPIをNNMi用とは異なるHAクラスター下で実行するように設定します。この場合は、NNM Performance iSPIのNNMiへの依存関係を手動で管理する必要があります。

NNMiでOracleデータベースを使う場合のシナリオ

NNMi実装でOracleをメインNNMiデータベースとして使用する場合、Oracleデータベースは、パフォーマンス上の理由から以下の図のように独立したサーバーにインストールする必要があります。そのため、NNMi HAクラスターでは、次の2つのHAリソースグループを設定する必要があります。

- NNMi HAリソースグループは、NNMiノードと、Oracleデータベースに格納されないNNMiデータ用の共有ディスクで構成します。
- Oracle HAリソースグループは、Oracleデータベースサーバーとデータベースディスクで構成します。

Oracleデータベースを使っているNNMi用のHA

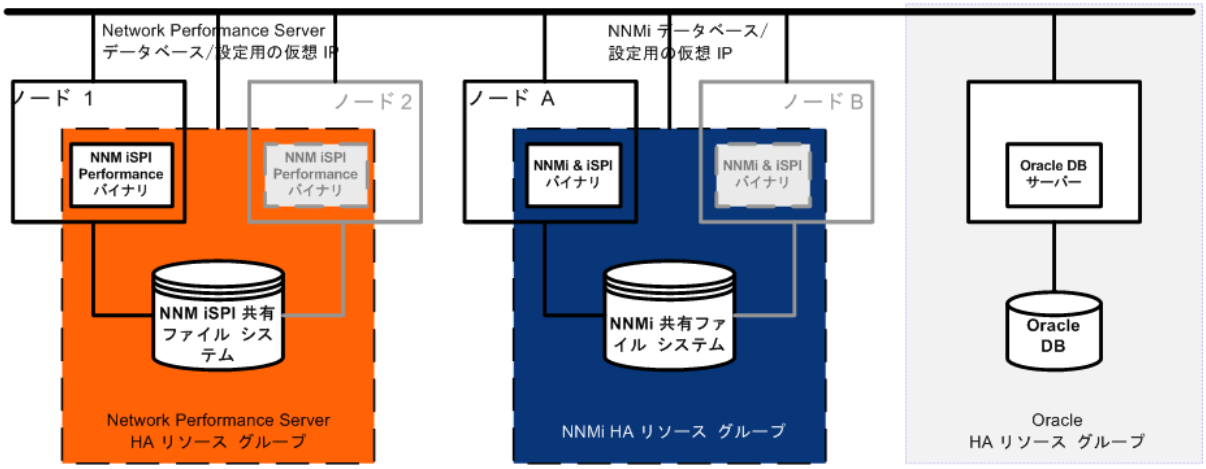


このシナリオの実装方法については、「Oracle環境での高可用性用のNNMiの設定」(195ページ)と「高可用性用のNNM iSPIsの設定」(193ページ)を参照してください。

NNMiでOracleデータベースを使用し、NNM Performance iSPI をスタンドアロンサーバーで実行する場合のシナリオ

NNMi実装でOracleをメインNNMiデータベースとして使用し、いずれかのNNM Performance iSPI製品をスタンドアロンサーバーで実行する場合は、以下の図に示すように、NNMi HAクラスター内に3つのHAリソースグループを設定できます。

NNMiでOracleデータベースを使用し、NNM Performance iSPIをスタンドアロンサーバーで実行する場合のHA



このシナリオの実装方法については、「Oracle環境での高可用性用のNNMiの設定」(195ページ) と「高可用性用のNNM iSPIsの設定」(193ページ)を参照してください。

マニュアルページ

NNMiには、NNMi高可用性設定に役立つ以下のマニュアルページがあります。

- nnm-ha
- nnmhaconfigure.ovpl
- nnmhaunconfigure.ovpl
- nnmhadisk.ovpl
- nnmhaclusterinfo.ovpl
- nnmhastartrg.ovpl
- nnmhastoprg.ovpl

Windowsオペレーティングシステムでは、これらのマニュアルページはテキストファイルで提供されます。

高可用性用NNMiを設定するための前提条件の検証

高可用性 (HA) 用NNMiを正常に設定できるかどうかは、以下のいくつかの要因に依存します。

- 適切なハードウェア
- HA製品についての理解
- 系統的な設定方法

HA用NNMiの設定を開始する前に、以下の準備手順を実行してください。

1. NNMi対応マトリックスの情報を調べて、使用するHA製品がNNMiでサポートされているかを確認します。
2. HA製品のマニュアルを読み、その製品の機能に精通してから設計上の決定を行います。

   ヒント: HA製品のマニュアルは頻繁に変更されます。必ず最新版のマニュアルを入手してください。

3. NNMi HAクラスターのノードとして含める各システムが以下の要件を満たすことを確認します。
   - HA製品のマニュアルに記載されているすべての要件に適合する。
   - 少なくとも2つのネットワークインタフェースカード (NICカード) が組み込まれている。

     注: HA製品、オペレーティングシステム、およびNICカードのマニュアルで調べて、これらの製品を一緒に使用できるかどうか確認してください。

- HAリソースグループの仮想IPアドレスの使用をサポートする。このIPアドレスは、NNMiライセンスで使用されるIPアドレスです。

  注: WSFCでは複数の仮想IPアドレスが必要です。1つはHAクラスター用、もう1つは各HAリソースグループ用です。この場合、NNMi HAリソースグループの仮想IPアドレスは、NNMiライセンスで使用されるIPアドレスです。

- 共有ディスクまたはディスクアレイの使用をサポートする

  注: HA製品、オペレーティングシステム、およびディスク製造業者のマニュアルで調べて、関連するSCSIカードを含め、これらの製品を一緒に使用できるかどうか確認してください。

- 「NNMi対応マトリックス」記載されているNNMiのすべての要件に適合する。

4. NNMi HAクラスターでいずれかのNNM iSPIsを実行する場合は、HA設定の追加の前提条件について、該当するNNM iSPIのマニュアルをお読みください。
5. 以下の仮想IPアドレスとホスト名を割り当てます。
   - HAクラスターに1つの仮想IPアドレス (WSFCのみ)
   - 設定する各HAリソースグループに1つの仮想IPアドレス
6. 任意のシステムから、`nslookup`コマンドを使用して、手順5で割り当てたすべてのIPアドレスとホスト名に対してDNSが正しく応答することを確認します。
7. 各システムのオペレーティングシステムが、HA製品とNNMiに適切なバージョンとパッチレベルになっていることを確認します。
8. 必要な場合は、HA製品をインストールします。
9. 「高可用性環境での手動による共有ディスクの準備」(198ページ)の説明に従って、共有ディスクを準備します。
10. HA製品用のコマンドを使用して、HAクラスターを設定 (必要な場合) およびテストします。

    HAクラスターには、アプリケーションハートビートのチェックやフェイルオーバー起動などの機能が用意されています。HAクラスター設定には、少なくとも、以下の項目を含める必要があります。

    - (Linuxのみ) ssh、remsh、または両方
    - (Windowsのみ) DNSで解決可能な、HAクラスター用の仮想IPアドレス
    - DNSで解決可能な、HAクラスター用の仮想ホスト名
    - NNMi固有の一意のリソースグループ。

      注: NNMiでは、必要なすべてのリソースがNNMi HAリソースグループに含まれているが期待されます。不足がある場合は、HA製品の機能を使用して、NNMi HAリソースグループ

とその他のHAリソースグループの間の依存関係を管理してください。たとえば、Oracleが別個のHAリソースグループ内で実行されている場合は、HA製品がNNMi HAリソースグループを起動する前にOracle HAリソースグループが完全に起動されるようにHA製品を設定します。

- WSFCの場合:Failover Cluster Management for Windows Serverのクラスター作成ウィザードを使用します。
- VCSの場合:必要ありません。製品のインストールによりHAクラスターが作成されました。
- RHCSの場合:RHCSのドキュメントの説明に従って、サービス(cman、rgmanager)を追加します。

NNMi HAリソースグループに入れるリソースのテストの詳細については、「HAリソーステスト」(214ページ)を参照してください。

高可用性の設定

このセクションでは、NNMi用の新規高可用性(HA)設定の設定手順を説明します。内容は以下のとおりです。

注: HAを設定するときは、以下の一般的なガイドラインを検討してください。

- RHCSの設定では、HAクラスターの各ノード上で、HAクラスターデーモンとすべてのアプリケーションを完全に再起動する必要があります。これを考慮して、設定作業を計画してください。
- NNMiリソースグループを変更するためにRHCS luci Webインタフェースを使用しないでください。NNMiリソースグループに変更を加えると、luci WebインタフェースはNNMiリソースグループのグローバル変数を/etc/cluster/cluster.confから削除します。NNMiリソースグループのグローバル変数は、NNMi HAを適切に機能させるために必要です。
- デフォルトでは、HA環境で、SNMPのソースアドレスが物理クラスターノードのアドレスに設定されます。SNMPのソースアドレスをNNM_INTERFACEに設定する(仮想IPアドレスに設定される)には、ov.confファイルを編集して、IGNORE_NNM_IF_FOR_SNMPの値をOFFに設定する必要があります(デフォルトでは、ONに設定されています)。
- 高可用性(HA)でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更を加えることでNNMi管理サーバーを停止して再起動する必要がある場合は、`ovstop`および`ovstart` コマンドを実行する前に、ノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

高可用性用のNNMi証明書の設定

NNMiのインストールプロセスでは、NNMiコンソールとNNMiデータベースの間でセキュアー通信が行われるよう、自己署名証明書を設定します。NNMi高可用性 (HA) を正しく設定するプロセスでは、プライマリクラスターノードとセカンダリクラスターノードの間で自己署名証明書を共有します。HA下で実行されるNNMiでデフォルトの証明書を使用するために、追加の手順を実行する必要はありません。

NNMiの通信で別の自己署名証明書または認証機関 (CA) 署名の証明書を使用する場合は、追加の手順を実行する必要があります。新しい証明書を入手してから、「高可用性環境での証明書の使用」(327ページ) に従って手順を実行します。この手順は、HA用NNMiを設定する前または後に実行できます。

高可用性用のNNMiの設定

高可用性 (HA) 用にNNMiを設定する場合の主な作業は、以下の2つです。

1. NNMiデータファイルを共有ディスクにコピーします。
   - 「プライマリクラスターノードでのNNMiの設定」(188ページ)の手順1 ~ 手順9の説明に従って、プライマリノードでこのタスクを実行します。
2. HA下でNNMiを実行するように設定します。
   - 「プライマリクラスターノードでのNNMiの設定」(188ページ)の手順10 ~ 手順15の説明に従って、プライマリノードでこのタスクを実行します。
   - また、「セカンダリクラスターノードでのNNMiの設定」(191ページ)の説明に従って、セカンダリノードでもこの作業を行います。

1つのHAクラスターノードを、プライマリNNMi管理サーバーとして割り当てます。これが大部分の時間にアクティブとなるノードです。プライマリノードを設定します。次にHAクラスター内の残りのすべてのノードをセカンダリノードとして設定します。

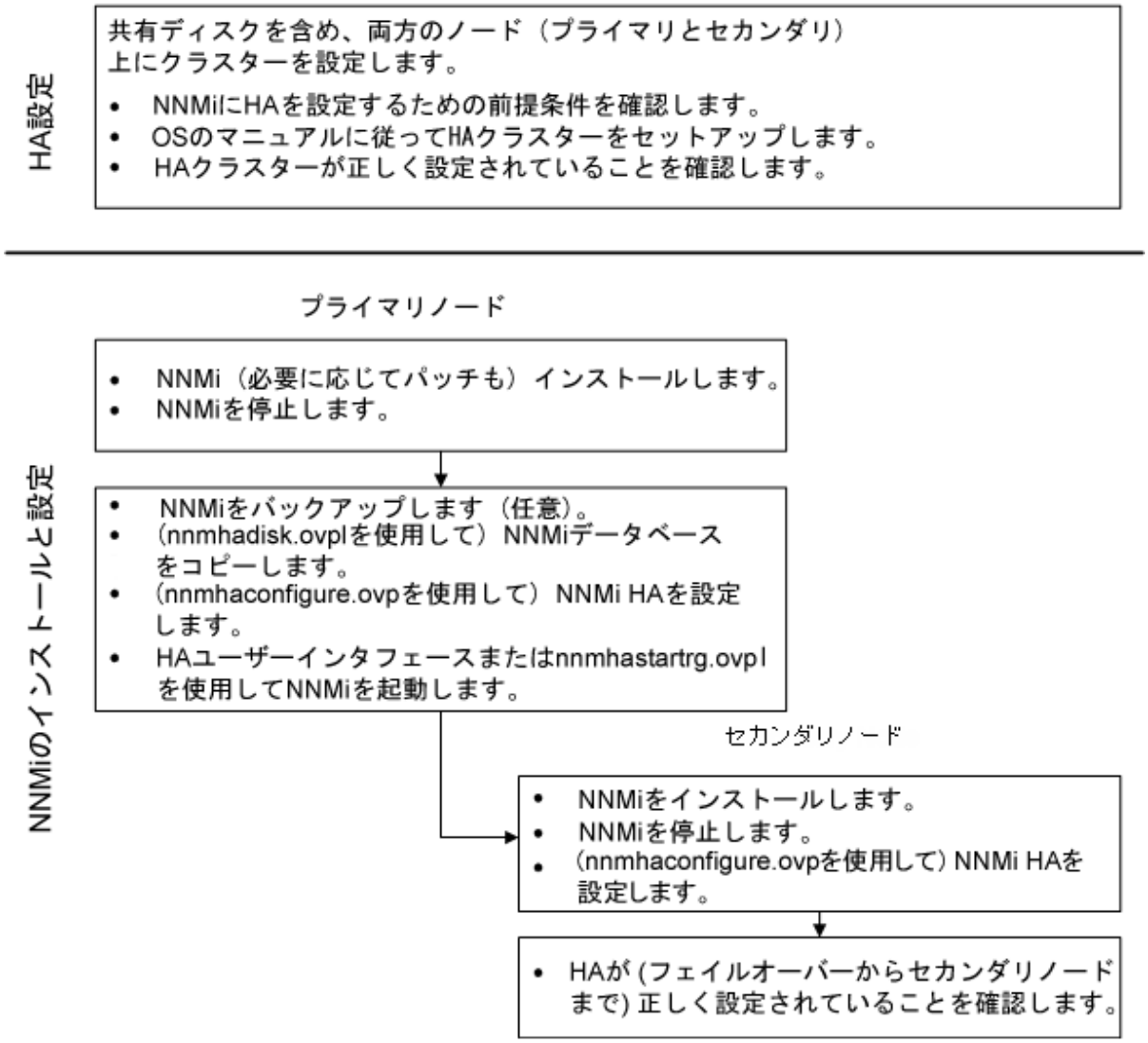
注意: HA用のNNMiの設定は、複数のクラスターノードで同時には行えません。1つのクラスターノードでHA設定プロセスが完了した後、次のクラスターノードでのHA設定プロセスを開始するというように、クラスター環境内のすべてのノードでHA用にNNMiを設定するまで、この作業を繰り返します。

注:

- フェイルオーバー中にはNNMiコンソールは応答しません。フェイルオーバーが完了してから、NNMiユーザーは、ログオンしてNNMiコンソールのセッションを続行する必要があります。
- NNMiのTrapReceiverプロセス、およびそのフェイルオーバーとの関連に関する重要情報については、「NNMi NmsTrapReceiverプロセス」(274ページ) を参照してください。

以下の図に、NNMi HA設定プロセスを示します。

NNMi HA設定ワークフロー



注: HA設定時にエラーが発生した場合は、以下の手順を実行します。

1. `nnmhaunconfigure.ovpl`コマンドを実行して、HA環境からNNMi設定を解除します。
2. エラーメッセージが示す状態を修正します。
3. `nnmhaconfigure.ovpl` コマンドを実行して、HA環境にNNMiを再設定します。

   (RHCSのみ) `nnmhaconfigure.ovpl`コマンドと`nnmhaunconfigure.ovpl`コマンドが正しく機能するためには`<failoverdomains/>`タグが`/etc/cluster/cluster.conf`ファイルに存在している必要があります。

   `<failoverdomains/>`タグはリソースマネージャーセクション内に埋め込まれています。たとえば以下のようになります。

   ...

```
...
<rm>
 <failoverdomains/>
</rm>
```

nnmhaconfigure.ovplコマンドには、以下の構造例を使用して、NNMiリソースグループを作成するために<failoverdomains/>タグが必要です。

```
...
<rm>
  <failoverdomains>
   <failoverdomain name="<rg-name>-dom" nofailback="0"
ordered="0" restricted="1">
      <failoverdomainnode name="<node1>" priority="1"/>
      <failoverdomainnode name="<node2>" priority="1"/>        </failoverdomain>
  </failoverdomains>
  <service autostart="1" domain="<rg-name>-dom"
exclusive="0" name="nnmha" recovery="relocate">
    <ip address="<addr>" monitor_link="1">
      <fs device="<nnmhalvol>" force_fsck="1"
force_unmount="1" fsid="" fstype="ext3"
mountpoint="<nnm-hamount>" name="nnmha-mount"
options="" self_fence="0">
       <NNMscript GLOBAL_VARIABLES="NNM_INTERFACE=
<virtual hostname>;HA_LOCALE=en_US.UTF-8;
HA_MOUNT_POINT=/<nnm-hamount>"
file="/var/opt/OV/hacluster/<rg-name>/nnmharhcs"
name="nnmha-APP"/>
       </fs>
    </ip>
  </service>
</rm>
```

nnmhaunconfigure.ovplコマンドでも、ノードのfailoverdomainエントリを削除するために上記の構造が必要です。

詳細については、nnmhaunconfigure.ovplおよびnnmhaconfigure.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

NNMi高可用性設定情報

高可用性 (HA) 設定スクリプトは、NNMi HAリソースグループに関する情報を収集します。NNMi HAを設定する前に、以下の表にリストされた情報を用意してください。この情報は、使用するオペレーティングシステムまたはHAソフトウェアに応じて、対話形式でHAスクリプト (nnmhaconfigure.ovpl) を実行するために必要です。

NNMi HAプライマリノードの設定情報

HA設定項目	説明
HAリソースグループ	NNMiを含むHAクラスターのリソースグループの名前です。この名前はNNMiに対して一意であり、現在使用されていない名前にする必要があります。有効な名前の情報については、HAシステムプロバイダーの参考資料を参照してください。 HAリソースグループ名の入力時に、NNMiはLinuxおよびWindowsシステムの以下のリソースを生成します。 <resource group name>-IP <resource group name>-Mount <resource group name>-App また、Windowsシステムでは、仮想ホスト名を入力すると次のリソースを生成します。 <virtual hostname>
仮想ホストの短い名前	仮想ホストの短い名前です。このホスト名は、HAリソースグループの仮想IPアドレスにマッピングする必要があります。nslookupコマンドで、仮想ホストの短い名前と仮想IPアドレスを解決できる必要があります。 **注:** NNMiが仮想ホストの短い名前とIPアドレスを解決できない場合は、HA設定スクリプトのためにシステムが不安定な状態になる可能性があります。したがって、NNMi HAの設定中にDNSが利用できない場合に備えて、予備のネーミングストラテジ (たとえば、Windowsオペレーティングシステムの場合は、%SystemRoot%\system32\drivers\etc\hostsファイルに、UNIXオペレーティングシステムの場合は、/etc/hostsファイルに、それぞれ情報を記述する) を用意しておくことをお勧めします。
仮想ホストのネットマスク	仮想ホストIPアドレスで使われるサブネットマスクです。これは、

NNMi HAプライマリノードの設定情報 (続き)

HA設定項目	説明
	IPv4アドレスであることが必要です。
仮想ホストのネットワークインタフェース	仮想ホストIPアドレスが使われるネットワークインタフェースです。例: • Windowsの場合:ローカルエリア接続 • Linuxの場合: eth0
共有ファイルシステムのタイプ	HAリソースグループで使われる共有ディスクの設定タイプです。使用できる値は次のとおりです。 • disk: 共有ディスクは、標準のファイルシステムタイプを使う、物理的に接続されたディスクです。HA設定スクリプトは、共有ディスクを設定できます。詳細については、この表のファイルシステムタイプの欄を参照してください。 • none: 共有ディスクには、diskオプションで説明している設定以外のNFS構成などを使います。HA設定スクリプトを実行すると、「高可用性環境での手動による共有ディスクの準備」(198ページ)の説明に従って、共有ディスクが設定されます。
ファイルシステムタイプ	(Linuxのみ) 共有ディスクのファイルシステムタイプです (共有ファイルシステムのタイプがdiskの場合)。HA設定スクリプトは、ディスクの検証方法を調べるために、この値をHA製品に渡します。 以下の共有ディスクフォーマットはテスト済みです。 • Windowsの場合:基本型 (「Windows Serverでの共有ディスク設定についての注記」(201ページ)を参照); SAN • Linuxの場合: VCSおよびRHCSにはext2、ext3、およびvxfs **注:** HA製品はほかのファイルシステムタイプをサポートしています。テストされていない共有ディスクフォーマットを使用する場合は、HA下で実行するようNNMiを設定する前にディスクを準備し、次にNNMi HA設定スクリプトを実行する間に共有ファイルシステムタイプとしてnoneと指定します。
ディスク情報 (使用するオペレーティングシステムに応じて、ディスクグループ、ボリュームグループ、論理ボリュームのいずれか、またはすべて)	NNMi共有ファイルシステムのディスク情報と関連付けられた名前です。 **注:** vxfsやlvmなどのUNIXプラットフォームのディスクを作成および接続する場合、ディスクグループ、ボリュームグループ、論理ボリュームなどの異なる項目を作成します。これらの項目の名前は、作成時にシステム管理者が割り当てます。NNMiには命名規約はありません。システム管理者に連絡して、会社の命名規約情

NNMi HAプライマリノードの設定情報 (続き)

HA設定項目	説明
	報を確認してください。
マウントポイント	NNMiの共有ディスクをマウントするディレクトリの場所です。このマウントポイントは、すべてのシステムで同じである必要があります。(つまり、各ノードでは、マウントポイントに同じ名前を使う必要があります。)次に例を示します。 • Windowsの場合: `S:\` **注:** ドライブ名は完全に指定してください。`S`および`S:` は受け入れられないフォーマットであり、共有ディスクにアクセスできません。 • Linuxの場合: `/nnmmount`

プライマリクラスターノードでのNNMiの設定

プライマリクラスターノードで以下の手順を実行します。

注: メインのNNMiデータベースとしてOracleを使用する場合は、まず「Oracle環境での高可用性用のNNMiの設定」(195ページ)を参照してください。

1. 「高可用性用NNMiを設定するための前提条件の検証」(180ページ)の作業を完了していない場合は、完了させます。
2. NNMiがインストールされていない場合は、NNMiを (最新の統合パッチも含めて) インストールしてから、正しく動作することを確認します。
3. このNNMi管理サーバー上でいずれかのNNM iSPIsを実行する場合は、この手順を進める前に「高可用性用のNNM iSPIsの設定」(193ページ) を参照してください。
4. `nnmbackup.ovpl`コマンド、または別のデータベースコマンドを使用して、NNMiデータをすべてバックアップします。次に例を示します。

   ```
   nnmbackup.ovpl -type offline -scope all -target nnmi_backups
   ```

   このコマンドの詳細については、「NNMiのバックアップおよび復元ツール」(243ページ)を参照してください。
5. NNMi HAリソースグループ用に、少なくとも1つの共有ディスクを含む、ディスクデバイスグループ (および論理ボリューム) を定義します。次に例を示します。
   - WSFCの場合: WSFCの場合: ディスクの管理を使用してディスクのマウントポイントを設定し、ディスクをフォーマットします。
   - VCSの場合:

vxdiskadm、vxassist、およびmkfsなどのVSFコマンドを使用して、ディスクを追加および初期化し、領域ごとにディスクを割り当て、論理ボリュームを作成します。

- RHCSの場合:

  pvcreate、vgcreate、およびlvcreateなどのLVMコマンドを使ってディスクの初期化、ボリュームグループの作成、および論理ボリュームの作成を行います。

注: NNMiが正しく開始および停止するために、NNMiでは、/etc/cluster/cluster.conf ファイルで指定されているクラスターノード名が完全修飾名であるようにRHCSクラスターを構成する必要があります。

Linuxオペレーティングシステムの参考Webサイトは、以下のとおりです。
http://www.unixguide.net/unixguide.shtml

6. ディレクトリのマウントポイント (たとえば、S:\または/nnmmount) を作成し、共有ディスクをマウントします。
   - Windowsの場合:Windowsの場合: Windows Explorerとディスクの管理ツールを使用してドライブ名を割り当てます。

     注意: ディスクの管理ツールを使用して、共有ディスクで [**オンライン**] と表示されるようにします。[**予約済み**] と表示される場合、これはWSFCが共有ディスクを制御することを示しています。WSFCユーザーインタフェースから [**削除**] アクションを使用して、共有ディスクをWSFCコントロールから削除します。また、ディスクの管理ツールを使用して、[**予約済み**] フラグが [**オンライン**] に変更されることも確認します。

   - Linuxの場合:
     - mkdirコマンドおよびmountコマンドを使用します。
     - 共有ディスクのディレクトリマウントポイントが、ユーザーはroot、グループはsysで作成され、権限には555が設定されていることを確認します。次に例を示します。

       ls -l /nnmmount

       注意: 設定後、HA製品はディスクのマウントを管理します。このマウントポイントを使用して、ファイルシステムテーブルを更新しないでください。

7. NNMiを停止します。

   ovstop -c

   注: このHAリソースグループに含めるノードにNNMiがすでにインストールされている場合は、このとき、そのノードでovstop -cも実行します。

8. NNMiデータベースを共有ディスクにコピーします。

- Windowsの場合:

  %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM -to <HA_mount_point>

- Linuxの場合:

  $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM -to <HA_mount_point>

注: データベースの破壊を避けるために、この (-toオプションを指定した) コマンドは1回しか実行できません。代替方法については、「すべてのクラスターノードを設定解除した後の高可用性用NNMiの再有効化」(215ページ) を参照してください。

9. (Linuxのみ) 共有ディスクをマウント解除し、ディスクグループを非アクティブ化します。

   umount <HA_mount_point>

   vgchange -a n <disk_group>

10. NNMiが実行中でないことを確認します。

    ovstop -c

11. (RHCSのみ) 以下の手順を実行し、必要なNNMスクリプトリソースを /usr/share/cluster/cluster.rngファイルに追加します。

    a. cluster.rngファイルのコピーを保存します。

    b. /usr/share/cluster/cluster.rngファイルを以下のように編集します。

       i. <define name="CHILDREN">を見つけます。

       ii. 前の手順で見つかったステートメントの先頭に /opt/OV/misc/nnm/ha/NNMscript.rngファイルの内容を埋め込みます。

       たとえば、<define name="CHILDREN">の上の1行に移動し、以下のように入力します。

       :r /opt/OV/misc/nnm/ha/NNMscript.rng

       iii. CHILDREN XMLブロックに、以下でボールドになっているテキストを追加します。

       ```
       <define name=" CHILDREN" >
        <zeroOrMore>
         <choice>
            ...
            <ref name=" SCRIPT" />
            <ref name=" NNMSCRIPT" />
            <ref name=" NETFS" />
       ```

       iv. cluster.rngファイルを保存します。

    c. /opt/OV/misc/nnm/ha/NNMscript.shファイルを/usr/share/clusterにコピーし、root:root所有権で555の権限があることを確認します。

d. ccsdサービスを再起動するかリブートします。
e. 前の手順でシステムをリブートした場合、クラスター設定を続行する前に、NNMiを停止します。

```
ovstop -c
```

f. NNMiが実行中でないことを確認します。

```
ovstatus -c
```

12. NNMi HAリソースグループを設定します。
- Windowsの場合:

```
%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM
```

- Linuxの場合:

```
$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM
```

13. (Linuxの場合のみ) NNMiは、デフォルトで、nnmhaconfigure.ovplコマンドを実行したユーザーのロケールで起動します。NNMiのロケールを変更するには、以下のコマンドを実行します。

```
$NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -set HA_LOCALE <locale>
```

14. 手順12で共有ファイルシステムタイプとして指定した値を判別します。
- タイプdiskを指定した場合は、nnmhaconfigure.ovplコマンドによって、共有ディスクが設定されています。手順15に進みます。
- タイプnoneを指定した場合は、「高可用性環境での手動による共有ディスクの準備」(198ページ)の説明に従って共有ディスクを準備し、手順15に進みます。

15. NNMi HAリソースグループを起動します。
- Windowsの場合:

```
%NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM <resource_group>
```

- Linuxの場合:

```
$NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM <resource_group>
```

NNMiを正常に起動できなかった場合は、「HA設定のトラブルシューティング」(212ページ)を参照してください。

注意: これで、NNMiがHA下で動作するようになりました。通常の操作では、ovstartコマンドやovstopコマンドは使用しないでください。これらのコマンドを使うのは、HAのメンテナンスが目的で、使うことが指示された場合だけです。

セカンダリクラスターノードでのNNMiの設定

セカンダリクラスターノードでは1つのノードごとに順番に以下の手順を実行します。

1. 「プライマリクラスターノードでのNNMiの設定」(188ページ)の作業を完了していない場合は、完了させます。
2. 「高可用性用NNMiを設定するための前提条件の検証」(180ページ)の作業を完了していない場合は、完了させます。
3. NNMiがインストールされていない場合は、NNMiを (最新の統合パッチも含めて) インストールしてから、正しく動作することを確認します。
4. 「プライマリクラスターノードでのNNMiの設定」(188ページ)の手順3でインストールしたNNM iSPIsをインストールします。
5. NNMiを停止します。

   ```
   ovstop -c
   ```
6. 共有ディスクのマウントポイントを作成します (たとえば、`S:\`または`/nnmmount`)。

   注: このマウントポイントでは、手順「プライマリクラスターノードでのNNMiの設定」(188ページ)の手順6で作成したマウントポイントと同じ名前を使用する必要があります。

7. (RHCSのみ) 以下の手順を実行し、必要なNNMスクリプトリソースを`/usr/share/cluster/cluster.rng`ファイルに追加します。
   a. `cluster.rng`ファイルのコピーを保存します。
   b. `/usr/share/cluster/cluster.rng`ファイルを以下のように編集します。
      i. `<define name="CHILDREN">`を見つけます。
      ii. 前の手順で見つかったステートメントの先頭に`/opt/OV/misc/nnm/ha/NNMscript.rng`ファイルの内容を埋め込みます。

      たとえば、`<define name="CHILDREN">`の上の1行に移動し、以下のように入力します。

      ```
      :r /opt/OV/misc/nnm/ha/NNMscript.rng
      ```

      iii. CHILDREN XMLブロックに、以下でボールドになっているテキストを追加します。

      ```
      <define name=" CHILDREN" >
       <zeroOrMore>
        <choice>
           ...
           <ref name=" SCRIPT" />
           <ref name=" NNMSCRIPT" />
           <ref name=" NETFS" />
      ```

      iv. `cluster.rng`ファイルを保存します。
8. (RHCSのみ) NNMiカスタムスクリプトを所定の場所にコピーし、HAクラスターデーモンを再起動します。

a. /opt/OV/misc/nnm/ha/NNMscript.shファイルを、以下の場所にコピーします。

/usr/share/cluster/NNMscript.sh

b. /sbin/ccsdプロセスを停止して、再起動します。

9. NNMi HAリソースグループを設定します。
   - Windowsの場合:**%NnmInstallDir%\misc\nnm\ha\nnmhaconfigure.ovpl NNM**
   - Linuxの場合:**$NnmInstallDir/misc/nnm/ha/nnmhaconfigure.ovpl NNM**

   コマンドの要求に応じて、HAリソースグループ名を指定します。
10. 設定が正常に行われたことを確認します。
    - Windowsの場合:

      %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -group <resource_group> -nodes

    - Linuxの場合:

      $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -group <resource_group> -nodes

    このコマンドの出力には、指定したHAリソースグループ用に設定されたノードがすべてリストされます。
11. 必要に応じて、プライマリノードのNNMi HAリソースグループをオフラインにし、セカンダリノードのNNMi HAリソースグループをオンラインにすることで、設定をテストします。

高可用性用のNNM iSPIsの設定

NNMi管理サーバー上でいずれかのNNM iSPIsを実行する場合は、NNMiをHA下で実行する設定を行う前に、このセクションをお読みください。

NNM iSPI Performance for Metrics、 NNMi SPI Performance for QA、 およびNNMi SPI Performance for Traffic

NNM iSPI Performance for MetricsはNNMi管理サーバーまたはスタンドアロンサーバーにインストールできます。

NNMi SPI Performance for Trafficには2つの異なるコンポーネント (Traffic MasterとTraffic Leaf) があり、これらのコンポーネントはNNMi管理サーバーまたはスタンドアロンサーバーにインストールすることも、両方を組み合わせる (1つのコンポーネントをNNMi管理サーバーにインストールしてもう一方をリモートサーバーにインストールする) こともできます。

注:

- NNM iSPI (またはコンポーネント) をNNMi管理サーバーに配置する場合は、HA下でNNMiを実行するように設定する前に、この製品をインストールします。

- NNM iSPI (またはコンポーネント) をスタンドアロンサーバー上に配置する場合は、この製品をインストールする前に、HA下で実行するようNNMiを設定します。NNM iSPIのインストールプロセス中に、NNMi HAリソースグループ仮想ホスト名をNNMi管理サーバー名として指定します。

NNM iSPIのインストールの詳細については、適切なNNM iSPIインストールガイドを参照してください。

NNMi SPI Performance for QA、 NNM iSPI for MPLS、 NNM iSPI for IP Multicast、 およびNNM iSPI for IP Telephony

NNMi SPI Performance for QA、 NNM iSPI for MPLS、 NNM iSPI for IP Multicast、 およびNNM iSPI for IP Telephonyは、 NNMi管理サーバーにのみインストールできます。

HA下で実行するようにNNM iSPIsを設定する場合の詳細については、該当するNNM iSPIのマニュアルを参照してください。

HA下で実行中のNNMi SPIネットワークエンジニアリングツールセットソフトウェアとNNMi

NNMi SPIネットワークエンジニアリングツールセットソフトウェア SNMPトラップ分析とMicrosoft Visioエクスポート機能は、NNMi PremiumまたはNNMi Ultimate製品と一緒に自動的にインストールされます。これらのツールをHA下で実行するには、これ以上の作業は不要です。

NNM iSPI NET診断サーバーは、NNMi HAリソースグループに含めることはできません。このコンポーネントは、NNMi管理サーバーにインストールしないでください。NNM iSPI NET診断サーバーをNNMi HAリソースグループ外のシステム上で実行するには、以下の手順を実行します。

注: NNM iSPI NET診断サーバーにはNNM iSPI NETライセンスまたはNNMi Ultimateライセンスが必要です。このサーバーのインストール方法および設定方法については、『HP NNM iSPIネットワークエンジニアリングツールセットソフトウェア計画とインストールガイド』(HP NNM iSPI Network Engineering Toolset Software Planning and Installation Guide)を参照してください。

1. NNMi HAリソースグループを完全に設定します。
2. NNM iSPI NET診断サーバーをNNMi HAリソースグループ外のシステムに、インストールします。NNM iSPI NET診断サーバーのインストールプロセス中に、NNMi HAリソースグループ仮想ホスト名をNNMサーバーホスト名として指定します。

   詳細は、『NNMi SPIネットワークエンジニアリングツールセットソフトウェア計画とインストールガイド』(NNM iSPI Network Engineering Toolset Software Planning and Installation Guide)を参照してください。

NNM iSPI NET診断サーバーがすでにHA下で実行されるNNMi管理サーバーにインストールされている場合、HA下で実行するようNNMiを設定する前にNNM iSPI NET診断サーバーをアンインストールします。

注意: NNM iSPI NET診断サーバーをアンインストールすると、既存のレポートがすべて削除され

ます。

注: ここで説明するように既存のレポートを保存することもできますが、次の手順はテストされていません。

1. MySQL Workbenchを使って、既存の`nnminet`データベースのバックアップを行う。

   MySQL Workbenchは、dev.mysql.comのダウンロードエリアから入手できます。
2. NNM iSPI NET診断サーバーをアンインストールします。
3. HA下でNNMiを実行するように設定します。
4. 別のシステムにNNM iSPI NET診断サーバーをインストールします。
5. フローを実行する前に、MySQL Workbenchを使って、新しいインストール先にある`nnminet`データベースを復旧します。

Oracle環境での高可用性用のNNMiの設定

このセクションでは、Oracleデータベースを使用しているNNMiを高可用性 (HA) 下で実行するための設定作業の概要を説明します。

注: Oracleの設定方法は多数あり、Oracleのリリースによっても異なります。OracleをHA下で実行するための設定方法とOracle HAリソースグループでのNNMiの依存関係の作成方法については、HA製品マニュアルを参照してください。OracleのWebサイト (www.oracle.com) でも、HA製品用のOracle設定方法が紹介されています。

高可用性環境でのNNMiのOracleへの依存

OracleとNNMiの両方を高可用性 (HA) 下で実行する場合は、NNMi HAリソースグループに、Oracleデータベースに格納されていないNNMiデータ用の共有ディスクを含める必要があります。

また、以下の情報を考慮してください。

- HA製品が依存関係をサポートする場合、各製品を別々のHAリソースグループ内で実行されるように設定するのが推奨される方法です。Oracle HAリソースグループは、NNMi HAリソースグループを起動する前に、完全に起動している必要があります。両方のHAリソースグループが同じHAクラスターに含まれている場合は、クラスター設定を変更してリソースグループの起動順序を設定します。それぞれのHAリソースグループが異なるHAクラスターに含まれている場合は、Oracle HAリソースグループに対するNNMiHAリソースグループの依存関係が満たされているかを確認します。
- HA製品が依存関係をサポートしない場合は、OracleシステムとNNMiシステムをNNMi HAリソースグループに含めてください。

Oracle環境での高可用性用のNNMiの設定

1. Oracleを高可用性 (HA) 下で実行することを予定している場合は、最初に、以下の手順を実行します。

2. NNMi用のに空のOracleデータベースインスタンスを作成します。
3. プライマリNNMiノードに、(最新の統合パッチも含めて) NNMiをインストールします。インストールの間に、以下の手順を実行します。
   a. [**Oracle**] データベースタイプを選択してから、[**プライマリサーバーのインストール**] を選択します。
   b. Oracle HAリソースグループ用の仮想IPアドレスまたは仮想ホスト名を指定します。
4. プライマリNNMiノードで、「プライマリクラスターノードでのNNMiの設定」(188ページ) の説明に従って、NNMiをHA下で実行できるように設定します。
5. Oracle HAリソースグループでのNNMiの依存関係を設定します。

   具体的な手順については、HA製品のマニュアルを参照してください。
6. セカンダリNNMiノードに、(最新の統合パッチも含めて) NNMiをインストールします。インストールの間に、以下の手順を実行します。
   - [**Oracle**] データベースタイプを選択してから、[**セカンダリサーバーのインストール**] を選択します。
   - Oracle HAリソースグループ用の仮想IPアドレスまたは仮想ホスト名を指定します。
7. セカンダリNNMiノードで、「セカンダリクラスターノードでのNNMiの設定」(191ページ) の説明に従って、NNMiをHA下で実行するように設定します。
8. 各追加のセカンダリNNMiノードで、手順6と手順7を繰り返します。

高可用性環境での共有NNMiデータ

高可用性 (HA) 下で実行するNNMi実装では、HAクラスター内のすべてのNNMiノード間でファイルを共有するために、独立したディスクを使用する必要があります。

注: Oracleをプライマリデータベースとして使っているNNMiの実装でも、共有データ用に独立したディスクを使う必要があります。

高可用性環境でのNNMiの共有ディスク上のデータ

このセクションでは、NNMiを高可用性 (HA) 下で実行する場合に、共有ディスクで管理されるNNMiのデータファイルをリストします。

ファイルの場所は、次のように、共有ディスク内の場所にマッピングされます。

- Windowsの場合:
  - `%NnmInstallDir%`は、`%HA_MOUNT_POINT%\NNM\installDir`にマッピングされます。
  - `%NnmDataDir%`は、`%HA_MOUNT_POINT%\NNM\dataDir`にマッピングされます。
- Linuxの場合:

- $NnmInstallDirは、$HA_MOUNT_POINT/NNM/installDirにマッピングされます。
- $NnmDataDirは、$HA_MOUNT_POINT/NNM/dataDirにマッピングされます。

共有ディスクに移動されるディレクトリは、以下のとおりです。

- Windowsの場合:
  - %NnmDataDir%\shared\nnm\databases\Postgres
    組み込みデータベース。Oracleデータベースを使用する場合は存在しません。
  - %NnmDataDir%\log\nnm
    NNMiのログディレクトリ。
  - %NnmDataDir%\nmsas\NNM\log
    NNMiの監査ログディレクトリ。
  - %NnmDataDir%\nmsas\NNM\conf
    監査ログファイルを設定するためのNNMiのディレクトリ。
  - %NnmDataDir%\nmsas\NNM\data
    ovjbossで使われるトランザクションストアー。
- Linuxの場合:
  - $NnmDataDir/shared/nnm/databases/Postgres
    組み込みデータベース。Oracleデータベースを使用する場合は存在しません。
  - $NnmDataDir/log/nnm
    NNMiのログディレクトリ。
  - %NnmDataDir/nmsas/NNM/log
    NNMiの監査ログディレクトリ。
  - %NnmDataDir/nmsas/NNM/conf
    監査ログファイルを設定するためのNNMiのディレクトリ。
  - $NnmDataDir/nmsas/NNM/data
    ovjbossで使われるトランザクションストアー。

これらのファイルは、nnmhadisk.ovplコマンドによって、共有ディスク間でコピーされます。この章の手順に従って、このコマンドを実行します。コマンド構文の概要については、nnm-haマンページを参照してください。

高可用性環境での設定ファイルの複製

NNMi高可用性 (HA) の実装では、ファイルレプリケーションを使用してHAクラスター内のすべてのNNMiノードのNNMi設定ファイルのコピーを管理します。

デフォルトでは、NNMiはファイルレプリケーションを管理し、フェイルオーバープロセス中に、アクティブノードからパッシブノードにNNMi設定ファイルをコピーします。nnmdatareplicator.conf ファイルには、データレプリケーションに含めるNNMiのフォルダーとファイルを指定します。

データレプリケーションの無効化

データレプリケーションは、以下の方法で無効にできます。

1. 以下のファイルを編集します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf\ov.conf
   - Linuxの場合: $NnmDataDir/shared/nnm/conf/ov.conf
2. 以下の行を含めます。

   DISABLE_REPLICATION=DoNotReplicate
3. 変更を保存します。

   注: アクティブノードでファイル (設定ファイルなど) を変更すると、これらのファイルはフェイルオーバーで自動的にスタンバイノードに複製されます。
4. NNMi管理サーバーを再起動します。

   注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

高可用性環境での手動による共有ディスクの準備

共有ディスクがHPでサポートされるフォーマットである場合は、高可用性 (HA) 設定スクリプトによって共有ディスクが準備されるため、以下の手順はスキップしてください。サポートされるディスクフォーマットの詳細については、「NNMi高可用性設定情報」(186ページ) を参照してください。

共有ディスクで、HA製品によってサポートされているディスクフォーマットなど、未検証の設定が使用されている場合は、共有ディスクを手動で準備する必要があります。HAの設定作業時に、ファイルシステムタイプの値としてnoneと入力してから、共有ディスクとNNMi HAリソースグループでの共有ディスクの使用を設定します。

ヒント: ディスクの設定は、NNMi HAリソースグループを設定する前、または後に実行できます。

共有ディスクを手動で準備するには、以下の手順を実行します。

1. 「SANまたは物理的に接続されたディスクの設定」(199ページ)の説明に従って、共有ディスクを設定します。
2. 以下の両方の手順を実行して、ディスクを認識するようにNNMi HAリソースグループを設定します。
   - 「ov.confファイルへの高可用性変数の設定」(199ページ)
   - 「NNMi HAリソースグループへの共有ディスクの移動」(200ページ)

SANまたは物理的に接続されたディスクの設定

ディスクを、vxfsまたはext3ファイルシステムに接続およびフォーマットします。SANまたは物理的に接続されたディスクを設定するには、以下の手順を実行します。

1. 共有ディスクがシステムブート時にマウントされるように設定されていないことを確認します。

   リソースグループには、共有ディスクをマウントする役割があります。
2. 以下のように、デバイスを接続します。
   - SANディスクの場合、SANデバイスをネットワークに追加します。

     SANディスクの論理ボリュームは、排他モードが使用できる場合には、排他モードである必要があります。
   - 物理的に接続されたディスクの場合、Yケーブルを使用してディスクを接続します。
3. オペレーティングシステムエントリを、すべてのクラスターノード(ディスクグループ、論理ボリューム、ボリュームグループ、およびディスク)に追加します。
   - SANディスクの場合、エントリはSANを参照します。
   - 物理的に接続されたディスクの場合、エントリはディスクハードウェアを参照します。
4. サポートされているディスクフォーマットを使用してディスクをフォーマットします。詳細については、「NNMi高可用性設定情報」(186ページ)を参照してください。
5. SANがマウントされていることを確認します。

   ヒント: Linuxシステムの参考Webサイトは、以下のとおりです。
   http://www.unixguide.net/unixguide.shtml
6. ディスクをマウント解除してデポートします。
7. 設定をテストするには、ディスクをリソースグループに追加し、フェイルオーバーを開始します。

ov.confファイルへの高可用性変数の設定

NNMi高可用性(HA)リソースグループは、以下の変数を使用して共有ディスクにアクセスします。

- HA_POSTGRES_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/databases/Postgres
- HA_EVENTDB_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/eventdb
- HA_NNM_LOG_DIR=<HA_mount_point>/NNM/dataDir/log
- HA_JBOSS_DATA_DIR=<HA_mount_point>/NNM/dataDir/nmsas/NNM/data
- HA_MOUNT_POINT=<HA_mount_point>
- HA_CUSTOMPOLLER_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/databases/custompoller

ヒント: NNMi HAリソースグループでNNM iSPIsを実行する場合は、それらのNNM iSPIsごとにov.conf変数も設定します。詳細については、該当するNNM iSPIのマニュアルを参照してください。

ov.confファイルで共有ディスクにアクセスするための製品の変数を設定するには、前述の各変数に対して、以下のコマンドを実行します。

- Windowsの場合:

  %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl – config NNM – set <variable> <value>

- Linuxの場合:

  $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl – config NNM – set <variable> <value>

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMi HAリソースグループへの共有ディスクの移動

製品マニュアルに従ってディスク設定ファイルを変更し、共有ディスクをNNMi HAリソースグループに移動します。次に例を示します。

ヒント: このプロセスを使用して、NICカードやバックアップディスクなどの他のリソースをNNMi HAリソースグループに追加することもできます。

- WSFCの場合: フェイルオーバー管理を使用して、リソースをリソースグループに追加します。
- VCSの場合: ディスクエントリを追加し、/opt/VRTSvcs/bin/haresコマンドを使ってHA設定ファイルにリンクします。次に例を示します。
- RHCSの場合:

  /etc/cluster/cluster.conf

Windows Serverでの共有ディスク設定についての注記

注: Microsoft Knowledge Baseの文書237853によれば、Windows Serverのクラスタリングではダイナミックディスクはサポートされていません。

正しくディスクを設定するには、以下のWebサイトの情報を参照してください。

- http://support.microsoft.com/kb/237853
- http://www.petri.co.il/difference_between_basic_and_dynamic_disks_in_windows_xp_2000_2003.htm

高可用性クラスターでのNNMiのライセンス

高可用性 (HA) クラスター内のNNMiを実行するには、NNMiに以下の2つのライセンスが必要です。

- 物理クラスターノードのいずれかのIPアドレスにロックされた商用ライセンス
- NNMi HAリソースグループの仮想IPアドレスにロックされた非商用ライセンス

NNMiライセンスキーは、共有ディスクで管理されます。このため、各NNMi HAリソースグループで、別個にライセンス契約された各製品に必要なのは非商用ライセンスキーのみです。

HAクラスターでNNMiのライセンスを設定する場合、アクティブノードのライセンスファイルにある新しい情報で共有ディスクの`licenses.txt`ファイルを更新する必要があります。HAクラスターでNNMiのライセンスを正しく設定するには、以下の手順を実行します。

- 指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、高可用性で使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。NNMi HAリソースグループの仮想IPアドレス用のライセンスキーを取得します。このライセンスキーは、最初はプライマリサーバーで使用され、必要に応じてセカンダリサーバーで使用されます。

注意: 同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。

HAクラスターでNNMiのライセンスを正しく設定するには、アクティブなNNMiクラスターノードで以下の手順を実行します。

1. 「NNMiのライセンス」(312ページ)の説明に従って、注文した製品ごとに恒久ライセンスキーを入手してインストールします。NNMi管理サーバーのIPドレスを入力するよう求められたら、NNMi HAリソースグループの仮想IPアドレスを入力します。
2. アクティブノードの`licenses.txt`ファイルにある新しい情報で、共有ディスクの`LicFile.txt`ファイルを更新します。以下のいずれかを行います。
   - `licenses.txt`ファイルが共有ディスクの`NNM`ディレクトリにある場合は、アクティブノードの`LicFile.txt`内の新しいライセンスキーを共有ディスクの`licenses.txt`に追加します。

- licenses.txtファイルが共有ディスクにない場合は、アクティブノードから共有ディスクのNNMディレクトリ内のlicenses.txtに、LicFile.txtをコピーします。

アクティブノードでは、LicFile.txtファイルが以下の場所にあります。

- Windowsの場合: %NnmDataDir%\shared\nnm\conf\licensing\LicFile.txt
- Linuxの場合: $NnmiDataDir/shared/nnm/conf/licensing/LicFile.txt

共有ディスクでは、licenses.txtファイルの場所は、たとえば以下のとおりです。

- Windowsの場合: S:\NNM\licenses.txt
- Linuxの場合: /nnmount/NNM/licenses.txt

高可用性設定のメンテナンス

このセクションでは、以下の高可用性設定メンテナンスタスクを実行する方法について説明します。

「メンテナンスモード」(202ページ)

「HAクラスター内のNNMiのメンテナンス」(203ページ)

「NNMi HAクラスター内のアドオンNNM iSPIsのメンテナンス」(207ページ)

メンテナンスモード

NNMiのパッチまたは更新プログラムを新しいバージョンのNNMiに適用する必要がある場合は、NNMi HAリソースグループをメンテナンスモードにし、処理中のフェイルオーバーを回避します。NNMi HAリソースグループがメンテナンスモードにある場合、ユーザー (またはインストールスクリプト) は必要に応じて、プライマリ (アクティブ) クラスターノード上で ovstopコマンドやovstartコマンドを実行できます。

注意: ovstartコマンドやovstopコマンドは、セカンダリ (バックアップ) クラスターノードでは絶対に実行しないでください。

HAリソースグループをメンテナンスモードにする

HAリソースグループをメンテナンスモードにすると、HAリソースグループの監視が無効になります。HAリソースグループがメンテナンスモードになっていると、そのHAリソースグループの製品の停止や起動を行ってもフェイルオーバーは行われません。

HAリソースグループをメンテナンスモードにするには、アクティブノードで以下のファイルを作成します。

- Windowsの場合: %NnmDataDir%\hacluster\<resource_group>\maintenance
- Linuxの場合: $NnmDataDir/hacluster/<resource_group>/maintenance

注: maintenanceファイルの内容は以下のとおりです。

- HAリソースグループの監視を無効にするには、maintenanceファイルを作成します。このファイルは空にすることもできますし、キーワードNORESTARTを含めることもできます。
- 設定手順を行っている間にNNMiが開始しないようにするには、maintenanceファイルの1番目の行に以下の1語のみを記載してください。
  NORESTART

HAリソースグループのメンテナンスモードを解除する

HAリソースグループのメンテナンスモードを解除すると、HAリソースグループの監視が再び有効になります。HAリソースグループの製品を停止すると、HAリソースグループはパッシブなクラスターノードへフェイルオーバーします。

HAリソースグループのメンテナンスモードを解除するには、以下の手順を実行します。

1. NNMiが正しく実行していることを確認します。

   ```
   ovstatus -c
   ```

   すべてのNNMiサービスで、[実行中] 状態が表示されます。
2. メンテナンスが開始される前にアクティブクラスターノードであったノードから、maintenanceファイルを削除します。このファイルについては、「HAリソースグループをメンテナンスモードにする」(202ページ) を参照してください。

HAクラスター内のNNMiのメンテナンス

このセクションでは、高可用性 (HA)クラスターでNNMiを維持するために必要な場合がある以下のタスクを実行する方法について説明します。

「NNMiの起動と停止」(203ページ)

「クラスター環境でNNMiのホスト名やIPアドレスを変更する」(204ページ)

「フェイルオーバーを行わせないようにNNMiを停止する」(207ページ)

「メンテナンス後にNNMiを再起動する」(207ページ)

NNMiの起動と停止

注: NNMiを高可用性 (HA) 下で実行している場合は、HAのメンテナンスが目的とする指示がない限り、ovstartコマンドやovstopコマンドは使用しないでください。

通常のオペレーションでは、NNMiに用意されているHAコマンドまたはHA製品の適切なコマンドを使用して、HAリソースグループの起動や停止を行います。

クラスター環境でNNMiのホスト名やIPアドレスを変更する

クラスター環境内のノードは、複数のIPアドレスやホスト名を持つことができます。ノードが別のサブネットのメンバーになった場合は、IPアドレスを変更する必要があります。それにより、IPアドレスや完全修飾ドメイン名が変更されます。

たとえば、Linuxシステムでは、IPアドレスと関連ホスト名は通常以下のいずれかの方法を使用して設定されています。

- /etc/hosts
- ドメインネームサービス (DNS)
- ネットワーク情報サービス (NIS)

NNMiは、管理対象ノードが参照できるように、NNMiデータベース内に管理サーバーのホスト名とIPアドレスを格納します。

ネームサーバーがない環境からネームサーバー (すなわち、DNSやBIND) がある環境に移行した場合は、ネームサーバーが新しいIPアドレスを解決することを確認してください。

ホスト名は、IPネットワーク内で管理対象ノードを特定するために使われます。ノードには複数のIPアドレスが設定されていることがありますが、ホスト名は特定のノードを指定するために使われます。システムのホスト名は、`hostname`コマンドを使ったときに返される文字列です。

NNMi HAリソースグループの仮想ホスト名またはIPアドレスを変更する場合は、アクティブノードのライセンスファイルにある新しい情報で、共有ディスクの`licenses.txt`ファイルを更新する必要があります。HA設定を正しく更新するには、以下の手順を実行します。

NNMi HAリソースグループの仮想ホスト名またはIPアドレスを変更するには、アクティブなNNMiクラスターノードで以下の手順を実行します。

注: NNMi (単品)、およびNNMiに同梱されているNNMi Advanced機能とNNM iSPI NET機能を購入した場合、アプリケーションフェイルオーバーおよび高可用性環境で使用するためのライセンスには2つのタイプがあります。

- アプリケーションフェイルオーバー
  - 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスをプライマリサーバーのIPアドレスと関連付けます。
  - 非商用 - このライセンスは、アプリケーションフェイルオーバー環境で使用するために個別に購入されます。このライセンスをセカンダリ (スタンバイ) サーバーのIPアドレスと関連付けます。

  高可用性 (HA)
  - 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスを物理クラスターノードのいずれかのIPアドレスに関連付けます。

- 非商用 - このライセンスは、高可用性環境で使用するために個別に購入されます。このライセンスをNNMi HAリソースグループの仮想IPアドレスに関連付けます。

- 指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、アップリケーションフェイルオーバーまたは高可用性で使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。必ず以下を要求します。
  - 高可用性:NNMi HAリソースグループの仮想IPアドレス用のライセンスキーを取得します。このライセンスキーは、最初はプライマリサーバーで使用され、必要に応じてセカンダリサーバーで使用されます。
  - アプリケーションフェイルオーバー:プライマリサーバーの物理IPアドレスに1つと、スタンバイサーバーの物理IPアドレスに1つの、2つのライセンスキーを取得します。

  注:同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。
- 以下で入手できる各NNM iSPIのドキュメントも参照してください。
  http://h20230.www2.hp.com/selfsolve/manuals。

1. NNMi HAリソースグループの以前の仮想IPアドレスのライセンスキーを、NNMi HAリソースグループの新しい仮想IPアドレスに変換します。

   注意: この時点で、新しいライセンスキーをインストールしないでください。

2. 「HAリソースグループをメンテナンスモードにする」(202ページ)の説明に従って、NNMi HAリソースグループをメンテナンスモードにします。
3. NNMi HAリソースグループを停止します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM <resource_group>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM <resource_group>

4. NNMi HAリソースグループのIPアドレスまたはノード名を変更します。
   a. ov.confファイルのNNM_INTERFACEエントリを編集して、新しいホスト名またはIPアドレスに変更します。
   b. ovspmd.authファイル内の旧ホスト名を含む行を編集して、新しいホスト名を含むようにします。

   ov.confファイルとovspmd.authファイルは、以下の場所にあります。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf
   - Linuxの場合: $NnmDataDir/shared/nnm/conf

5. NNMi HAリソースグループのノード名を変更した場合、nnmsetofficialfqdn.ovplコマンドを

使用して、NNMi HAリソースグループの新しい完全修飾ドメイン名を使用するように、NNMiを設定します。例:

nnmsetofficialfqdn.ovpl newnnmi.servers.example.com

詳細については、nnmsetofficialfqdn.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

6. 新しいIPアドレスを使うように、クラスター設定を変更します。
   - WSFCの場合:

     Failover Cluster Managementで、<resource_group>を開きます。

     <resource_group>-ipをダブルクリックして、[**パラメーター**]を選択し、新しいIPアドレスを入力します。

   - VCSの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhargconfigure.ovpl NNM <resource_group> -set_value <resource_group>-ip Address <new_IP_address>

   - RHCSの場合:

     アクティブなHAクラスターノードで、/etc/cluster/cluster.confファイルを編集して、ip address="<old_IP_address>"をip address="<new_IP_address>"に置き換えます。次に、ccs_tool update /etc/cluster/cluster.confを実行して、残りのシステムをすべて更新します。

7. 「NNMiのライセンス」(312ページ)の説明に従って、NNMi HAリソースグループの新しい仮想IPアドレスのライセンスキーをインストールします。
8. アクティブノードのlicenses.txtファイルにある新しい情報で、共有ディスクのLicFile.txtファイルを更新します。以下のいずれかを行います。
   - licenses.txtファイルが共有ディスクのNNMディレクトリにある場合は、アクティブノードのLicFile.txt内の新しいライセンスキーを共有ディスクのlicenses.txtに追加します。
   - licenses.txtファイルが共有ディスクにない場合は、アクティブノードから共有ディスクのNNMディレクトリ内のlicenses.txtに、LicFile.txtをコピーします。

   アクティブノードでは、LicFile.txtファイルが以下の場所にあります。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf\licensing\LicFile.txt
   - Linuxの場合: $NnmDataDir/shared/nnm/conf/licensing/LicFile.txt

   共有ディスクでは、licenses.txtファイルの場所は、たとえば以下のとおりです。
   - Windowsの場合: S:\NNM\licenses.txt
   - Linuxの場合: /nnmount/NNM/licenses.txt

9. NNMi HAリソースグループを起動します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM <resource_group>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM <resource_group>

10. NNMiを正常に起動できたことを確認します。

    ```
    ovstatus -c
    ```

    すべてのNNMiサービスで、[実行中] 状態が表示されます。
11. 「HAリソースグループのメンテナンスモードを解除する」(203ページ)の説明に従って、NNMi HAリソースグループのメンテナンスモードを解除します。

フェイルオーバーを行わせないようにNNMiを停止する

NNMiのメンテナンスを行う必要がある場合は、アクティブクラスターノードのNNMiを、パッシブノードへフェイルオーバーさせないように停止できます。

アクティブクラスターノードで以下の手順を実行します。

1. 「HAリソースグループをメンテナンスモードにする」(202ページ)の説明に従って、NNMi HAリソースグループをメンテナンスモードにします。
2. NNMiを停止します。

   ```
   ovstop -c
   ```

メンテナンス後にNNMiを再起動する

フェイルオーバーしないようにNNMiを停止した場合は、以下の手順を実行して、NNMiとHA監視を再起動します。

1. NNMiを起動します。

   ```
   ovstart -c
   ```

2. NNMiを正常に起動できたことを確認します。

   ```
   ovstatus -c
   ```

   すべてのNNMiサービスで、[実行中] 状態が表示されます。
3. 「HAリソースグループのメンテナンスモードを解除する」(203ページ)の説明に従って、NNMi HAリソースグループのメンテナンスモードを解除します。

NNMi HAクラスター内のアドオンNNM iSPIsのメンテナンス

NNM iSPIsは、NNMiに密接にリンクしています。アドオンNNM iSPIsをNNMi HAクラスター内のノードにインストールする場合は、NNMi HAクラスターのメンテナンス手順を使います。

HAクラスター内のNNMiの設定解除

NNMiノードを高可用性 (HA) クラスターから削除する手順には、NNMiのインスタンスのHA設定を解除する手順も含まれます。設定を解除すると、NNMiのインスタンスをスタンドアロン管理サーバーとして実行できます。また、そのノードからNNMiをアンインストールできます。

注: NNMiをアンインストールする前に、最新のパッチから開始して、NNMiパッチをすべて逆順で削除します。パッチの削除プロセスは、NNMi管理サーバーで実行しているオペレーティングシステムによって異なります。インストールおよび削除手順については、パッチのマニュアルを参照してください。

高可用性用のNNMiの設定を維持するには、HAクラスターに、NNMiを実行中の1つのノードと、少なくとも、1つのパッシブNNMiノードが必要です。HAクラスターからNNMiを完全に削除するには、クラスター内のすべてのノードでHA機能の設定を解除します。

HAクラスター環境からNNMiを完全に設定解除するには、以下の手順を実行します。

1. HAクラスター内のアクティブなノードを特定します。スタンバイで、以下のコマンドを実行します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -group <resource_group> -activeNode

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -group <resource_group> -activeNode

2. 各パッシブノードで、HAクラスターから任意のアドオンNNM iSPIsの設定を解除します。

   詳細については、各NNM iSPIのマニュアルを参照してください。
3. HAクラスター内の任意のノードで、すべてのパッシブノード上のアドオンNNM iSPIsがHAクラスターから設定解除されていることを確認します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -config NNM -get NNM_ADD_ON_PRODUCTS

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -get NNM_ADD_ON_PRODUCTS

   コマンドの出力には、アドオンiSPIの設定が`<iSPI_PM_Name>[hostname_list]`のフォーマットでリストされます。例:

   PerfSPIHA[hostname1, hostname2]

   このとき、アクティブノードのホスト名のみが出力に表示されます。パッシブノードのホスト名が出力に表示される場合は、このコマンドの出力にアクティブノードのホスト名のみが表示されるようになるまで、手順2を繰り返します。

4. アクティブノードで、HAクラスターからアドオンNNM iSPIsの設定を解除します。

   詳細については、各NNM iSPIのドキュメントを参照してください。HAクラスター内の任意のノードで、すべてのノード上のアドオンNNM iSPIsがHAクラスターから設定解除されていることを確認します。

   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -config NNM -get NNM_ADD_ON_PRODUCTS

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -config NNM -get NNM_ADD_ON_PRODUCTS

   ホスト名が出力に表示される場合は、このコマンドの出力にiSPIが設定されていないことが示されるまで、手順6を繰り返します。
5. 各パッシブノードで、HAクラスターからNNMiの設定を解除します。

   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM <resource_group>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM <resource_group>

     このコマンドにより、共有ディスクへのアクセスが削除されますが、ディスクグループやボリュームグループは設定解除されません。

6. 各パッシブノードで、NNMi HAリソースグループ固有のファイルを安全に保持できるように別の場所に移動します。

   %NnmDataDir%\hacluster\<resource_group>\フォルダー。

   ヒント: NNMi HAリソースグループを再設定する予定がない場合は、これらのファイルのコピーを保存する必要はありません。

7. アクティブノードで、NNMi HAリソースグループを停止します。

   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhastoprg.ovpl NNM <resource_group>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhastoprg.ovpl NNM <resource_group>

   このコマンドでは、共有ディスクへのアクセス権は削除しません。また、ディスクグループやボリュームグループの設定も解除しません。
8. アクティブノードで、HAクラスターからNNMiを設定解除します。

- Windowsの場合:

  %NnmInstallDir%\misc\nnm\ha\nnmhaunconfigure.ovpl NNM <resource_group>

- Linuxの場合:

  $NnmInstallDir/misc/nnm/ha/nnmhaunconfigure.ovpl NNM <resource_group>

このコマンドにより、共有ディスクへのアクセスが削除されますが、ディスクグループやボリュームグループは設定解除されません。

9. アクティブノードで、NNMi HAリソースグループ固有のファイルを安全に保持できるように別の場所に移動します。

   %NnmDataDir%\hacluster\<resource_group>\フォルダー

   ヒント: NNMi HAリソースグループを再設定する予定がない場合は、これらのファイルのコピーを保存する必要はありません。

10. 共有ディスクをマウント解除します。
    - 将来、NNMi HAクラスターを再設定する必要がある場合は、ディスクを現在の状態のまま保持しておきます。
    - 共有ディスクを別の目的で使用する場合は、保存するデータをすべてコピーして (「既存データベースを使用したHA外でのNNMi実行」(210ページ) の説明を参照) から、HA製品のコマンドを使用し、ディスクグループとボリュームグループの設定を解除します。

既存データベースを使用したHA外でのNNMi実行

NNMiをHAの外部の任意のノードで既存のデータベースを使って実行する場合は、以下の手順を実行します。

1. アクティブノードで (存在する場合)、NNMiが実行中ではないことを確認します。

   ovstop

   あるいは、タスクマネージャー (Windows) またはpsコマンド (Linux) を使用して、ovspmdプロセスのステータスをチェックします。

2. 現在のノード (HAの外部でNNMiの実行を予定しているノード) で、NNMiが実行中ではないことを確認します。

   ovstop

   注意: データの破壊を避けるために、NNMiのインスタンスが動作中ではないことや、共有ディスクにアクセス中ではないことを確認します。

3. (Linuxのみ) ディスクグループをアクティブ化します。たとえば、以下を実行します。

   vgchange -a e <disk_group>

4. 適切なオペレーティングシステムのコマンドを使って、共有ディスクをマウントします。例:

- Windowsの場合:[サーバー マネージャ] > [ディスクの管理] を使用します。
- Linuxの場合: `mount /dev/vgnnm/lvnnm /nnmmount`

5. NNMiのファイルを共有ディスクからローカルディスクにコピーします。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM -from <HA_mount_point>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM -from <HA_mount_point>

6. 適切なオペレーティングシステムのコマンドを使って、共有ディスクのマウントを解除します。例:
   - Windowsの場合:Windowsエクスプローラーを使用します。
   - Linuxの場合: `umount /nnmmount`

7. (Linuxのみ) ディスクグループを非アクティブ化します。たとえば、以下を実行します。

   vgchange -a n <disk_group>

8. 「NNMiのライセンス」(312ページ)の説明に従って、このNNMi管理サーバーの物理IPアドレスの商用恒久ライセンスキーを取得し、インストールします。
9. NNMiを起動します。

   ovstart -c

   従来、NNMi HAリソースグループで使われていたデータベースのコピーを使って、NNMiが起動されます。このNNMi管理サーバーから管理対象としないノードのNNMi設定を手動で削除します。

HA下のNNMiのパッチ

パッチをNNMiに適用するには、高可用性 (HA) メンテナンスモードで作業します。以下の手順を実行します。

1. HAクラスター内のアクティブなノードを特定します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhaclusterinfo.ovpl -group <resource_group> -activeNode

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhaclusterinfo.ovpl -group <resource_group> -activeNode

2. 「HAリソースグループをメンテナンスモードにする」(202ページ)の説明に従って、各パッシブノードで、NNMi HAリソースグループをメンテナンスモードにします。

   `NORESTART`キーワードを含めます。

3. 各パッシブノードで、該当するパッチを適用します。

   注意: ovstartコマンドや ovstopコマンドは、セカンダリ (バックアップ) クラスターノードでは絶対に実行しないでください。

4. 「HAリソースグループのメンテナンスモードを解除する」(203ページ)の説明に従って、すべてのパッシブノードで、NNMi HAリソースグループをメンテナンスモードから解除します。
5. パッシブノードにフェイルオーバーします。
6. この手順を開始したときにアクティブだったノードに移動し(ステップ1)、以下の手順を実行します。
   a. 「HAリソースグループをメンテナンスモードにする」(202ページ)の説明に従って、そのノードのNNMi HAリソースグループをメンテナンスモードにします。

      NORESTARTキーワードを含めます。

   b. ノードで、該当するパッチを適用します。

      注意: ovstartコマンドや ovstopコマンドは、セカンダリ (バックアップ) クラスターノードでは絶対に実行しないでください。

   c. ノードで、「HAリソースグループのメンテナンスモードを解除する」(203ページ)の説明に従って、NNMi HAリソースグループをメンテナンスモードから解除します。

HA設定のトラブルシューティング

このセクションでは、以下のトピックについて説明します。

- 「一般的な高可用性設定の誤り」(212ページ)
- 「RHCS 6での設定の問題」(213ページ)
- 「HAリソーステスト」(214ページ)
- 「一般的なHAのトラブルシューティング」(220ページ)
- 「NNMi固有の高可用性のトラブルシューティング」(215ページ)
- 「NNM iSPI固有の高可用性のトラブルシューティング」(222ページ)

一般的な高可用性設定の誤り

高可用性 (HA) 設定における一般的な誤りの一部を以下に示します。

- 正しくないディスク設定
  - VCSの場合: リソースをプローブできない場合は、設定に何らかの間違いがあります。ディスクをプローブできない場合、オペレーティングシステムはディスクにアクセスできなくなる可能性があります。

  - 手動でディスク設定をテストし、設定が適切であることをHAのマニュアルの記載内容と照合して確認してください。

- ディスクが使用中で、HAリソースグループで起動できない。

  HAリソースグループを起動する前に、ディスクがアクティブでないことを必ず確認してください。
- WSFC: ネットワーク設定が正しくない

  ネットワークトラフィックが複数のNICカード上を流れる場合は、NNMi ovjbossプロセスなどのネットワーク帯域幅を大量に消費するプログラムをアクティブ化するとRDPセッションが失敗します。
- 一部のHA製品がブート時に自動的に再起動しない。

  ブートアップ時の自動再起動の設定方法については、HA製品のマニュアルを確認してください。
- NFSまたは他のアクセスがOSに直接追加される (リソースグループ設定でこの動作を管理している必要があります)。
- フェイルオーバーの間、またはHAリソースグループをオフラインにする間に、共有ディスクのマウントポイントに存在している。

  HAは、共有ディスクのマウント解除を阻止するプロセスをすべて抹消します。
- HAクラスターの仮想IPアドレスをHAリソースの仮想IPアドレスとして再使用している (一方のシステムで有効で、他方では無効)
- タイムアウトが短すぎる。製品に不具合があると、HA製品はHAリソースをタイムアウトさせ、フェイルオーバーが実行されます。

  WSFC: Failover Cluster Managementで、[**リソースが開始するまでの待機時間**] の設定値を確認します。NNMiでは、この値は15分に設定されます。この値を増やすことができます。
- メンテナンスモードを使用していない

  メンテナンスモードは、HAの障害をデバッグするために作成されました。リソースグループをシステムでオンラインにしようとして、その後すぐにフェイルオーバーする場合、メンテナンスモードを使用してリソースグループのオンラインを維持し、障害のある部分を見つけます。
- クラスターログを再確認していない (クラスターログで多くの一般的な間違いを確認できます)。

RHCS 6での設定の問題

ricciサービスがダウンしていたり、意図的に無効化されている場合、HA環境の2つのシステム間で`/etc/cluster/cluster.conf`ファイルのバージョンが異なる可能性があります。そのため、`cluster.conf`ファイルを定期的に監視して、ファイルのバージョンが同期されていることを確認します。

`cluster.conf`ファイルのバージョンが同期されていない場合は、次のいずれかを実行しようとする場合に問題が発生する可能性があります。

- 変更を`cluster.conf`に適用する
- リソースグループの設定を解除する

- クラスターを起動する
- `clustat`コマンドを使用する

HAリソーステスト

このセクションでは、NNMi HAリソースグループに入れるリソースのテストを行うための一般的な方法を説明します。このテストによって、ハードウェア設定の問題が特定されます。高可用性 (HA) の下で実行するようにNNMiを設定する前に、このテストを実行することをお勧めします。好ましい結果を出した設定値を記録しておき、NNMi HAリソースグループの完全な設定を行うときに、それらの値を使用します。

ここに記載されているコマンドについての詳細については、HA製品の最新マニュアルを参照してください。

HAリソースをテストするには、以下の手順を実行します。

1. 必要に応じて、HAクラスターを起動します。
2. (Windowsのみ) HAクラスターに、以下の仮想IPアドレスが定義されていることを確認します。
   - HAクラスターの仮想IPアドレス
   - 各HAリソースグループの仮想IPアドレス

   これらの各IPアドレスは、別の場所で使用しないでください。
3. HAリソースグループをHAクラスターに追加します。

   このHAリソースグループには、`test`など、非商用名を使用してください。
4. HAリソースグループへの接続をテストします。
   a. 仮想IPアドレスと、リソースグループに対応する仮想ホスト名を、リソースとしてHAリソースグループに追加します。

      後でNNMi HAリソースグループに関連付ける値を使用します。
   b. アクティブクラスターノードからパッシブクラスターノードにフェイルオーバーし、HAクラスターが正常にフェイルオーバーすることを確認します。
   c. 新しいアクティブクラスターノードから新しいパッシブクラスターノードにフェイルオーバーし、フェイルバックを確認します。
   d. リソースグループが正しくフェイルオーバーしない場合、アクティブノードにログオンして、IPアドレスが正しく設定され、アクセス可能であることを確認します。また、ファイアウォールによってIP address.vがブロックされていないかも確認します。
5. 「SANまたは物理的に接続されたディスクの設定」(199ページ)の説明に従って、共有ディスクを設定します。
6. 共有ディスクへの接続をテストします。
   a. 「NNMi HAリソースグループへの共有ディスクの移動」(200ページ)の説明に従って、共有ディスクをリソースとしてHAリソースグループに追加します。

b. アクティブクラスターノードからパッシブクラスターノードにフェイルオーバーし、HAクラスターが正常にフェイルオーバーすることを確認します。
c. 新しいアクティブクラスターノードから新しいパッシブクラスターノードにフェイルオーバーし、フェイルバックを確認します。
d. リソースグループが正しくフェイルオーバーしない場合、アクティブノードにログオンして、ディスクがマウントされ、使用可能であることを確認します。

7. 共有ディスクの設定に使用したコマンドおよび入力値の記録を取っておきます。NNMi HAリソースグループを設定するときに、この情報が必要になる場合があります。
8. 各ノードからリソースグループを削除します。
   a. IPアドレスエントリを削除します。
   b. リソースグループをオフラインに設定して、ノードからリソースグループを削除します。

   この時点で、NNMiに付属しているツールを使用して、HA下で実行するようにNNMiを設定することができます。

NNMi固有の高可用性のトラブルシューティング

このセクションの内容が適用されるのは、NNMiのみの高可用性 (HA) 設定です。以下の内容が含まれます。

すべてのクラスターノードを設定解除した後の高可用性用NNMiの再有効化

すべてのNNMi高可用性 (HA) クラスターノードの設定を解除した場合は、NNMiの共有ディスクのマウントポイントへのリンクが`ov.conf`ファイルから削除されます。

共有ディスク内のデータを上書きすることなく、マウントポイントへのリンクを作成しなおすには、プライマリノードで以下の手順を実行します。

1. NNMiが実行中であれば、停止します。

   ```
   ovstop -c
   ```

2. 共有ディスクへのリンクを削除します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM -setmount <HA_mount_point>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM -setmount <HA_mount_point>

3. ov.confファイルのHAマウントポイント関連のエントリを確認します。
   ov.confファイルの場所は、「NNMi高可用性設定ファイル」(223ページ)を参照してください。

NNMiを高可用性下で正常に起動できない

NNMiが正しく起動しない場合、問題が仮想IPアドレスまたはディスクに関するハードウェアの問題であるのか、ある種のアプリケーション障害の問題であるのかをデバッグする必要があります。このデバッグプロセスの間、NORESTARTキーワードを設定しないで、システムをメンテナンスモードにします。

1. HAクラスターのアクティブノードで、以下のメンテナンスファイルを作成して、HAリソースグループの監視を無効にします。
   - Windowsの場合: %NnmDataDir%\hacluster\<resource_group>\maintenance
   - Linuxの場合: $NnmDataDir/hacluster/<resource_group>/maintenance
2. NNMiを起動します。

   ovstart
3. NNMiを正常に起動できたことを確認します。

   ovstatus -c

   すべてのNNMiサービスで、[実行中] 状態が表示されます。このように表示されない場合、正しく開始していないプロセスをトラブルシューティングします。
4. トラブルシューティングが完了したら、メンテナンスファイルを削除します。
   - Windowsの場合: %NnmDataDir%\hacluster\<resource_group>\maintenance
   - Linuxの場合: $NnmDataDir/hacluster/<resource_group>/maintenance

NNMiデータへの変更がフェイルオーバーの後に表示されない

NNMiの設定で、NNMiを実行中のシステム以外のシステムを指しています。この問題を解決するには、ov.confファイルに以下の項目に対応した適切なエントリがあるか確認します。

- NNM_INTERFACE=<virtual_hostname>
- HA_RESOURCE_GROUP=<resource_group>
- HA_MOUNT_POINT=<HA_mount_point>
- NNM_HA_CONFIGURED=YES

- HA_POSTGRES_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/databases/Postgres
- HA_EVENTDB_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/eventdb
- HA_CUSTOMPOLLER_DIR=<HA_mount_point>/NNM/dataDir/shared/nnm/databases/custompoller
- HA_NNM_LOG_DIR=<HA_mount_point>/NNM/dataDir/log
- HA_JBOSS_DATA_DIR=<HA_mount_point>/NNM/dataDir/nmsas/NNM/data
- HA_LOCALE=C

ov.confファイルの場所は、「NNMi高可用性設定ファイル」(223ページ)を参照してください。

高可用性の設定後、nmsdbmgrを起動できない

この状況は、通常、nnmhaconfigure.ovplコマンドを実行したが、-toオプションを指定してnnmhadisk.ovplコマンドを実行せずにNNMiを起動した場合に発生します。この状況では、ov.confファイルのHA_POSTGRES_DIRエントリは、共有ディスクの組み込みデータベースの場所を指していますが、この場所はNNMiからはアクセスできません。

この問題を解決するには、以下の手順を実行します。

1. 高可用性 (HA) クラスターのアクティブノードで、以下のメンテナンスファイルを作成して、HAリソースグループのモニタリングを無効にします。
   - Windowsの場合: %NnmDataDir%\hacluster\<resource_group>\maintenance
   - Linuxの場合: $NnmDataDir/hacluster/<resource_group>/maintenance
2. NNMiデータベースを共有ディスクにコピーします。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM -to <HA_mount_point>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM -to <HA_mount_point>

   注意: データベースの破壊を避けるために、この (-toオプションを指定した) コマンドは1回しか実行できません。代替方法については、「すべてのクラスターノードを設定解除した後の高可用性用NNMiの再有効化」(215ページ) を参照してください。

   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM <resource_group>

   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM <resource_group>

3. NNMiを起動します。

   ```
   ovstart
   ```

4. NNMiを正常に起動できたことを確認します。

   ```
   ovstatus -c
   ```

   すべてのNNMiサービスで、[実行中] 状態が表示されます。

5. トラブルシューティングが完了したら、メンテナンスファイルを削除します。
   - Windowsの場合: `%NnmDataDir%\hacluster\<resource_group>\maintenance`
   - Linuxの場合: `$NnmDataDir/hacluster/<resource_group>/maintenance`

NNMiが1つの高可用性クラスターノードでのみ正常に実行される (Windows)

Windowsオペレーティングシステムには、2つの異なる仮想IPアドレス (高可用性 (HA) クラスター用に1つと、HAリソースグループ用に1つ) が必要です。

HAクラスターの仮想IPアドレスとNNMi HAリソースグループの仮想IPアドレスが同じ場合、NNMiは、HAクラスターのIPアドレスと関連付けられているノードでのみ正常に実行されます。

この問題を修正するには、HAクラスターの仮想IPアドレスをネットワークで一意の値に変更します。

ディスクフェイルオーバーが行われない

この状況は、オペレーティングシステムが共有ディスクをサポートしていない場合に発生します。HA製品、オペレーティングシステム、ディスクのメーカーのマニュアルで調べて、これらの製品を混在させて使用できるか確認してください。

ディスク障害が発生すると、NNMiはフェイルオーバーでは起動しません。`nmsdbmgr`が失敗する理由の多くは、`HA_POSTGRES_DIR`ディレクトリが存在しないことにあります。共有ディスクがマウント済みであり、該当するファイルにアクセスできる状態になっていることを確認してください。

共有ディスクにアクセスできない (Windows)

`nnmhaclusterinfo.ovpl -config NNM -get HA_MOUNT_POINT`コマンドを実行しても何も戻されません。

共有ディスクのマウントポイントのドライブは、HA設定時に完全に指定する必要があります (たとえば、`S:\`)。

この問題を修正するには、HAクラスターの各ノードで`nnmhaconfigure.ovpl`コマンドを実行します。共有ディスクのマウントポイントのドライブを完全に指定します。

共有ディスクに最新データが含まれない

ディスクタイプについてのnnmhaconfigure.ovplコマンドの質問にテキストnoneで応答すると、ov.confファイルでディスク関連の変数を設定するコードがバイパスされます。この状況を修正するには、「高可用性環境での手動による共有ディスクの準備」(198ページ)の手順に従います。

フェイルオーバー後にセカンダリノードが共有ディスクファイルを見つけられない

この状況は、通常、共有ディスクがマウントされていないときに、-toオプションを付けたnnmhadisk.ovplコマンドを実行した場合に発生します。この場合には、データファイルはローカルディスクにコピーされ、共有ディスクには格納されません。

この問題を解決するには、以下の手順を実行します。

1. 高可用性 (HA) クラスターのアクティブノードで、以下のメンテナンスファイルを作成して、HAリソースグループのモニタリングを無効にします。
   - Windowsの場合: %NnmDataDir%\hacluster\<resource_group>\maintenance
   - Linuxの場合: $NnmDataDir/hacluster/<resource_group>/maintenance
2. アクティブノードにログオンして、ディスクがマウントされ、使用可能であることを確認します。
3. NNMiを停止します。

   ovstop
4. NNMiデータベースを共有ディスクにコピーします。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhadisk.ovpl NNM -to <HA_mount_point>
   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhadisk.ovpl NNM -to <HA_mount_point>

   注意: データベースの破壊を避けるために、この (-toオプションを指定した) コマンドは1回しか実行できません。代替方法については、「すべてのクラスターノードを設定解除した後の高可用性用NNMiの再有効化」(215ページ)を参照してください。
5. NNMi HAリソースグループを起動します。
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM <resource_group>
   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM <resource_group>

6. NNMiを起動します。

   ```
   ovstart
   ```

7. NNMiを正常に起動できたことを確認します。

   ```
   ovstatus -c
   ```

   すべてのNNMiサービスで、[実行中] 状態が表示されます。

8. トラブルシューティングが完了したら、メンテナンスファイルを削除します。
   - Windowsの場合: `%NnmDataDir%\hacluster\<resource_group>\maintenance`
   - Linuxの場合: `$NnmDataDir/hacluster/<resource_group>/maintenance`

一般的なHAのトラブルシューティング

このセクションのトピックは、NNMiおよびNNM iSPIのHA設定に適用されます。以下の内容が含まれます。

- 「エラー:引数の数が正しくない」(220ページ)
- 「リソースをホストするサブシステムプロセスが予期せず停止する (Windows Server)」(220ページ)
- 「アクティブなクラスターノードのログファイルが更新されない」(221ページ)
- 「NNMi HAリソースグループを特定のクラスターノードで起動できない」(221ページ)

エラー:引数の数が正しくない

Perlモジュール製品の名前は、大部分のNNMi高可用性 (HA) 設定コマンドで必須パラメーターになりました。

- NNMiでは、値として`NNM`を使用します。
- NNM iSPIで使用する値を調べるには、そのNNM iSPIのマニュアルを参照してください。

リソースをホストするサブシステムプロセスが予期せず停止する (Windows Server)

Windows Serverオペレーティングシステムを実行しているコンピューターで高可用性 (HA) クラスターリソースを起動すると、リソースをホストするサブシステム (`Rhs.exe`) プロセスが予期せずに停止します。

この既知の問題の詳細については、MicrosoftサポートWebサイトの記事「Windows Serverでは、クラスターリソースを起動すると、リソースをホストするサブシステム (Rhs.exe) プロセスが予期せず停止します」(http://support.microsoft.com/kb/978527) を参照してください。

ヒント: NNMiリソースを実行するときは、必ず、リソースグループに固有の別個のリソースモニター (`rhs.exe`) で実行してください。

製品の起動タイムアウト (Windows WSCS 2008)

NNMi 10.01へのアップグレード後、フェイルオーバークラスターマネージャーのアプリケーションリソース (<resource>-app) が「Pending」から「Failed」に変わった場合は、タイムアウトの問題である可能性があります。この場合は、以下の手順を実行します。

1. cluster log /genコマンドを使用して、cluster.logファイルを生成します。
2. 以下のディレクトリにあるログを開きます。

   C:\Windows\cluster\reports\cluster.log

3. cluster.logファイルで以下のようなエラーが表示される場合は、DeadlockTimeoutに問題があります。

   ERR [RHS] Resource <resource-name>-APP handling deadlock.Cleaning current operation.

   DeadlockTimeout はエージェントがブロックされた可能性がある場合フェイルオーバーの合計時間です。PendingTimeoutは、オンライン操作またはオフライン操作のいずれかを表します。DeadlockTimeoutのデフォルト値は45分 (2,700,000ミリ秒)、PendingTimeoutのデフォルト値は30分 (1,8000,000ミリ秒) です。

   DeadlockTimeoutとPendingTimeoutの値は変更できます。たとえば、75分のDeadlockTimeoutおよび60分のPendingTimeoutを設定するには、以下のコマンドを実行できます。

   cluster res "<resource group>-APP" /prop DeadlockTimeout=4500000

   cluster res "<resource group>-APP" /prop PendingTimeout=3600000

   詳細については、高可用性ベンダーのドキュメントを参照してください。

アクティブなクラスターノードのログファイルが更新されない

これは正常です。ログファイルは、共有ディスクにリダイレクトされているため、このような状況になります。

NNMiの場合は、ov.confファイル内のHA_NNM_LOG_DIRで指定された場所にあるログファイルを調べてください。

NNMi HAリソースグループを特定のクラスターノードで起動できない

nnmhastartrg.ovplコマンドまたはnnmhastartrg.ovplコマンドで、NNMi HAリソースグループを正常に起動、停止、または切り替えできない場合は、以下の事柄を調べてください。

- MSFCの場合:
  - Failover Cluster Managementで、NNMi HAリソースグループと基盤リソースの状態を調べてください。
  - イベントビューアーのログにエラーが記録されていないか調べてください。
- VCSの場合:

- **/opt/VRTSvcs/bin/hares -state**を実行してリソース状態を確認します。
- 障害が発生しているリソースでは、障害が発生しているリソース用の /var/VRTSvcs/log/<resource>.logファイルを調べます。リソースは、IP*.log、Mount*.log、Volume*.logなどのエージェントタイプで指定します。

問題の原因を特定できない場合は、HA製品のコマンドを使用してNNMi HAリソースグループを手動で起動できます。

1. 共有ディスクをマウントします。
2. 仮想ホストをネットワークインタフェースに割り当てます。
   - MSFの場合:
     - Failover Cluster Managementを起動します。
     - リソースグループを展開します。
     - **<resource_group>-ip**を右クリックして、[**Bring Online**] をクリックします。
   - VCSの場合:**/opt/VRTSvcs/bin/hares -online <resource_group>-ip -sys <local_hostname>**
   - RHCS:/usr/sbin/cmmodnetを実行して、IPアドレスを追加します。
3. NNMi HAリソースグループを起動します。例:
   - Windowsの場合:

     %NnmInstallDir%\misc\nnm\ha\nnmhastartrg.ovpl NNM -start <resource_group>
   - Linuxの場合:

     $NnmInstallDir/misc/nnm/ha/nnmhastartrg.ovpl NNM -start <resource_group>

リターンコード0は、NNMiを正常に起動できたことを意味します。

リターンコード1は、NNMiを正常に起動できなかったことを意味します。

NNM iSPI固有の高可用性のトラブルシューティング

高可用性下で実行中のNNM iSPIのトラブルシューティングについては、そのNNM iSPIのドキュメントを参照してください。

高可用性設定リファレンス

このセクションでは、以下の高可用性設定項目に関するリファレンス情報を記載します。

「NNMi高可用性設定ファイル」(223ページ)

NNMi高可用性設定ファイル

以下の表に、NNMi高可用性 (HA) 設定ファイルをリストします。これらのファイルは、NNMi管理サーバー上のNNMiとアドオンNNM iSPIsに適用されます。これらのファイルは、以下の場所にインストールされます。

- Windowsの場合: `%NnmDataDir%\shared\nnm\conf`
- Linuxの場合: `$NnmDataDir/shared/nnm/conf`

NNMi HA設定ファイル

ファイル名	説明
ov.conf	このファイルは、NNMi HA実装の状態を示し、`nnmhaclusterinfo.ovpl` コマンドによって更新されます。NNMiの各プロセスは、このファイルを読み取って、HA設定を確認します。
nnmdatareplicator.conf	このファイルは、`nnmdatareplicator.ovpl`コマンドで、アクティブノードからパッシブノードへのデータレプリケーションに含むNNMiのフォルダーとファイルを調べるために使われます。NNMi設定のレプリケーション用に異なる手段を実装する場合は、含めるデータのリストは、このファイルを参照してください。 詳細については、このファイルのコメントを参照してください。

NNMiに付属しているHA設定スクリプト

以下の表に、NNMiに付属しているHA設定スクリプトを示します。NNMi HA設定スクリプトに示したNNMi付属のスクリプトは、カスタマーPerlモジュールを持つすべての製品にHAを設定する場合に使用できる便利なスクリプトです。必要に応じて、HA製品に付属しているコマンドを使って、NNMi用にHAを設定できます。

NNMi管理サーバーでは、NNMiに付属しているHA設定スクリプトは、以下の場所にインストールされます。

- Windowsの場合: `%NnmInstallDir%\misc\nnm\ha`
- Linuxの場合: `$NnmInstallDir/misc/nnm/ha`

NNMi HA設定スクリプト

スクリプト名	説明
nnmhaconfigure.ovpl	NNMiまたはNNM iSPIをHAクラスター用に設定します。 このスクリプトは、HAクラスター内のすべてのノードで実行してくだ

NNMi HA設定スクリプト (続き)

スクリプト名	説明
	さい。
nnmhaunconfigure.ovpl	HAクラスターのNNMiまたはNNM iSPIの設定を解除します。 必要に応じて、HAクラスター内の1つ以上のノードでこのスクリプトを実行します。
nnmhaclusterinfo.ovpl	NNMiに関するクラスター情報を取得します。 このスクリプトは、必要に応じて、HAクラスター内の任意のノードで実行します。
nnmhadisk.ovpl	データファイルを、NNMiおよびNNM iSPIと共有ディスクの間でコピーします。 HAの設定時には、このスクリプトはプライマリノードで実行します。 それ以外の場合は、この章の手順に従って、このスクリプトを実行します。
nnmhastartrg.ovpl	HAクラスターでNNMi HAリソースグループを起動します。 HAの設定時には、このスクリプトはプライマリノードで実行します。
nnmhastoprg.ovpl	HAクラスターでNNMi HAリソースグループを停止します。 HAの設定解除時には、このスクリプトはプライマリノードで実行します。

以下の表に示すNNMi付属のスクリプトは、NNMi HA設定スクリプトに示すスクリプトで使用します。以下の表に示したスクリプトは直接実行しないでください。

NNMi HAサポートスクリプト

スクリプト名	説明
nnmdatareplicator.ovpl	nnmdatareplicator.conf設定ファイルを調べて、リモートシステムに送信するファイルの変更やコピーを確認します。
nnmharg.ovpl	HAクラスターのNNMiを起動/停止/監視します。 VCS設定では、VCSの起動、停止、および監視のスクリプトで使用します。(nnmhargconfigure.ovplで、この使用法を設定します。) また、トレースを有効/無効にするために、nnmhastartrg.ovplでも使われます。
nnmhargconfigure.ovpl	HAのリソースとリソースグループを設定します。 nnmhaconfigure.ovplとnnmhaunconfigure.ovplで使われます。

NNMi HAサポートスクリプト (続き)

スクリプト名	説明
nnmhastart.ovpl	HAクラスターでNNMiを起動します。nnmharg.ovplで使われます。
nnmhastop.ovpl	HAクラスターのNNMiを停止します。nnmharg.ovplで使われます。
nnmhamonitor.ovpl	HAクラスターのNNMiプロセスを監視します。nnmharg.ovplで使われます。
nnmhamscs.vbs	MSFC HAクラスターで、NNMiプロセスを起動、停止、および監視するスクリプトを作成するためのテンプレートです。生成されるスクリプトはMSFCによって使用され、以下の場所に保存されます。%NnmDataDir%\hacluster\<resource_group>\hamscs.vbs

NNMi高可用性設定ログファイル

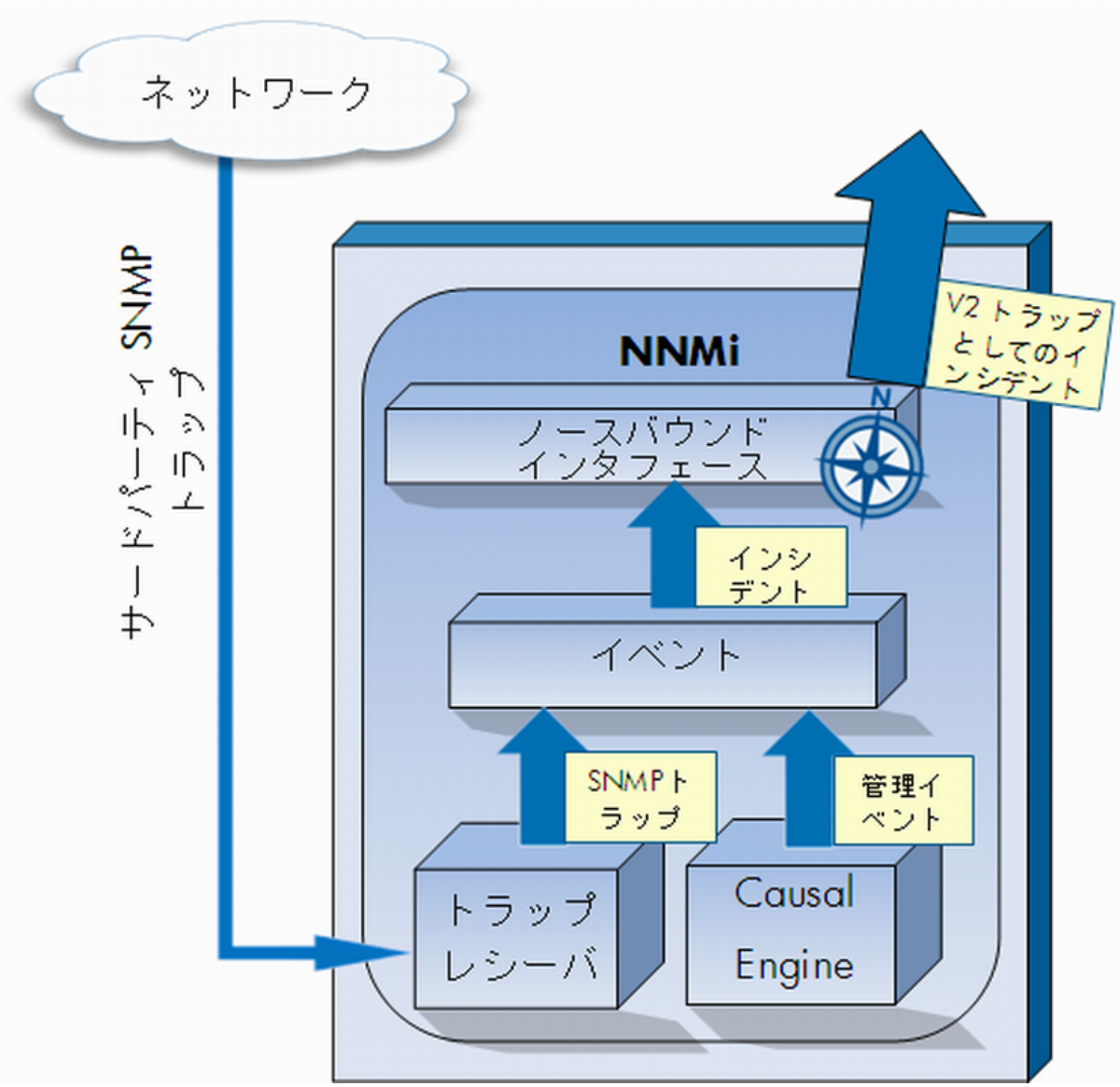
以下のログファイルは、NNMi管理サーバー上のNNMiとアドオンNNM iSPIs用のHA設定に適用されます。

- Windows設定:
  - %NnmDataDir%\tmp\HA_nnmhaserver.log
  - %NnmDataDir%\log\haconfigure.log
- Linux設定:
  - $NnmDataDir/tmp/HA_nnmhaserver.log
  - $NnmDataDir/log/haconfigure.log
- Windows実行時:
  - イベントビューアーのログ
  - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\ovspmd.log
  - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\public\postgres.log
  - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\public\nmsdbmgr.log
  - %HA_MOUNT_POINT%\NNM\dataDir\log\nnm\nnm.log
  - %SystemRoot%\Cluster\cluster.log
    これは、リソースとリソースグループの追加/削除、ほかの設定上の問題点、起動/停止上の問題点を含む、クラスター実行時の問題点に関するログファイルです。
- Linuxの場合:

- /var/adm/syslog/syslog.log
- $HA_MOUNT_POINT/NNM/dataDir/log/nnm/ovspmd.log
- $HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/postgres.log
- $HA_MOUNT_POINT/NNM/dataDir/log/nnm/public/nmsdbmgr.log
- $HA_MOUNT_POINT/NNM/dataDir/log/nnm/nnm.log

ヒント: HAベンダーログを参照する必要がある場合もあります。たとえば、Veritasはログファイルを/var/VRTSvcs/logフォルダーに保存します。RHCSはログメッセージをsyslogに記録します。

NNMi Northboundインタフェース



HP Network Node Manager i Software (NNMi) には、NNMi Northboundインタフェースが用意されており、SNMPv2cトラップを受信できるアプリケーションにNNMiインシデントを転送することができます。各NNMi管理サーバーに、別々に設定された複数のNNMi Northboundインタフェースを実装できます。

NNMiには、NNMi Northboundインタフェースを使用して以下の製品との統合をサポートする機能も組み込まれています。

- HP Business Service Management (BSM) プラットフォームのOperations Management機能。
- HP Operations Manager (HPOM) アクティブメッセージブラウザー。

- IBM Tivoli Netcool/OMNIbus。
- HP ArcSight Logger。

異なるNorthboundアプリケーションと統合するには、この章の指示に従ってください。

この章には、以下のトピックがあります。

NNMi Northboundインタフェース

NNMi Northboundインタフェースは、NNMi管理イベントをSNMPv2cトラップとしてNorthboundアプリケーションに転送します。Northboundアプリケーションは、NNMiトラップをフィルタリング、処理、および表示します。Northboundアプリケーションには、NNMiトラップのコンテキストでNNMiコンソールにアクセスするツールも用意されています。

NNMi Northboundインタフェースは、インシデントライフサイクルの状態変更通知、インシデント相関処理通知、およびインシデント削除通知をNorthboundアプリケーションに送信できます。このように、NorthboundアプリケーションはNNMiの因果関係分析の結果を複製することができます。

NNMi Northboundインタフェースは、NNMiが受信するSNMPトラップをNorthboundアプリケーションに転送することもできます。

値

NNMiノースバウンドインタフェースにより、サードパーティまたはカスタムイベント統合アプリケーションでイベント統合を実行することができます。NNMi Northboundインタフェースは、その他のアプリケーションとNNMiの統合に使用できる情報でイベントを強化します。

サポートされるバージョン

この章の情報は、NNMiバージョン9.00以降に適用されます。

サポートされているハードウェアプラットフォームおよびオペレーティングシステムの最新情報については、『NNMi対応マトリックス』を参照してください。

用語

この章では、以下の用語を使用します。

- Northboundアプリケーション—SNMPv2cトラップを受信および処理できる任意のアプリケーション。
- トラップ受信コンポーネント—SNMPトラップを受信する、ノースバウンドアプリケーションの一部分。
  - 一部のアプリケーションには、SNMPトラップを受信して処理用に別のコンポーネントに転送する、個別にインストール可能なコンポーネントが含まれます。
  - そのようなコンポーネントがないNorthboundアプリケーションの場合、「トラップ受信コンポーネント」は「Northboundアプリケーション」と同義語です。
- NNMi Northboundインタフェース — NNMiインシデントをSNMPv2cトラップとしてNorthboundアプリケーションに転送するNNMiの機能。
- Northbound転送先 — Northboundアプリケーションのトラップ受信コンポーネントへの接続を定義し、NNMiがそのNorthboundアプリケーションに送信するトラップのタイプを指定するNNMiNorthboundインタフェースの設定の1つ。

ドキュメント

この章では、NNMiインシデントを任意のNorthboundアプリケーションに転送するようにNNMiを設定する方法を説明します。特定のNorthboundアプリケーションの詳細については、そのアプリケーションのマニュアルを参照してください。

NNMi Northboundインタフェースの有効化

注意: NNMiは、UDPを使用してSNMPトラップで送信される情報の量を制限しません。トラップデータのサイズが大きくて処理不能なネットワークハードウェアが伝送経路上にあったり、ネットワークトラフィックの量が多かったりすると、トラップが失われることがあります。そのため、Northboundアプリケーションのトラップ受信コンポーネントを NNMi管理サーバーにインストールすることをお勧めします。Northboundアプリケーションは、信頼性のある情報を転送する役割を担います。

NNMiノースバウンドインタフェースを有効にするには、以下の手順を実行します。

1. 必要に応じて、NNMiトラップ定義を認識できるようにNorthboundアプリケーションを設定します。
2. NNMi管理サーバーで、NNMiインシデント転送を設定します。
   a. NNMiコンソールで、[**HP NNMi – Northbound Interfaceデスティネーション**] フォーム ([**統合モジュールの設定**] > [**Northboundインタフェース**]) を開き、[**新規作成**] をクリックします。

(使用可能な転送先を選択してある場合、[**リセット**] をクリックして、[**新規作成**] ボタンを使用可能にしてください。)

b. [**有効にする**] チェックボックスをオンにし、フォームの残りのフィールドを入力可能にします。

c. Northboundアプリケーションへの接続情報を入力します。

これらのフィールドの詳細については、「Northboundアプリケーションの接続パラメーター」(237ページ)を参照してください。

d. 送信オプションおよびNorthboundアプリケーションに送信する内容に対するインシデントフィルターを指定します。

これらのフィールドの詳細については、「NNMi Northboundインタフェース統合の内容」(239ページ)を参照してください。

e. フォームの下部にある [**送信**] をクリックします。

新しいウィンドウが開き、ステータスメッセージが表示されます。設定に問題があることを示すメッセージが表示されたら、[**戻る**] をクリックして、エラーメッセージを参考に値を調整してください。

3. 省略可能。NorthboundアプリケーションからNNMiビューにアクセスするためのURLを作成し、NNMiとのコンテキストインタラクションを作成します。

詳細については、NNMiコンソールで、**[ヘルプ] > [NNMi ドキュメントライブラリ] > [NNMiを別の場所でURLと統合]** をクリックしてください。

NNMiノースバウンドインタフェースの使用法

NNMi Northboundインタフェースを有効にすると、Northbound転送先によってNNMiがNorthboundアプリケーションに送信する情報が決まります。Northboundアプリケーションを設定して、転送されるトラップがネットワーク環境に応じて表示および解釈されるようにします。NNMiがNorthboundアプリケーションに送信するトラップの内容および形式の詳細については、`hp-nnmi-nbi.mib`および`hp-nnmi-registration.mib`ファイルを参照してください。

NNMiは、各管理イベント、SNMPトラップ、または通知トラップのコピーを1つだけNorthbound転送先に送信します。NNMiはトラップをキューに入れません。NNMiがトラップを転送するときにNorthboundアプリケーションのトラップ受信コンポーネントに接続できないと、トラップは失われます。

このセクションでは、統合で送信可能なトラップのタイプを説明します。コンテンツ設定の設定詳細については、「NNMi Northboundインタフェース統合の内容」(239ページ) を参照してください。

インシデント転送

管理イベント

Northbound転送先に管理イベントが含まれる場合、そのインシデントのライフサイクル状態が [登録済み] に変更されると、NNMiは各管理イベントのインシデントをNorthboundアプリケーションに転送します。

転送される管理イベントのOIDは、NNMiコンソールの [**管理イベントの設定**] フォームに表示されるSNMPオブジェクトIDです。NNMiは、OIDが1.3.6.1.4.1.11.2.17.19.2.0.9999のすべてのカスタム管理イベントを転送します。

サードパーティSNMPトラップ

Northbound転送先にサードパーティのSNMPトラップが含まれる場合、関連インシデントのライフサイクル状態が [登録済み] に変更されると、NNMiはSNMPv1、v2c、またはv3形式の各受信トラップをNorthboundアプリケーションに転送します。NNMiは、(MIBで定義される) 元のトラップvarbindの順序を維持し、メッセージペイロードにNNMi固有のvarbindを追加します。元のトラップに含まれていない定義済みvarbindがある場合、NNMiは、その欠落しているvarbindの部分にNULL値をパディングします。MIBをNNMiにロードしていない場合、NNMi固有のvarbindのみがトラップに追加され、次にこのトラップが転送されます。

サードパーティのSNMPトラップの場合は、以下の点に注意してください。

- NNMiはSNMPトラップインシデントからのトラップを再構成するため、転送されるトラップの形式は、NNMiが受信した元のトラップの形式に関係なく、SNMPv2cとなります。
- 転送されるSNMPトラップは、NNMi管理サーバーをソースオブジェクトとして示します。元のソースオブジェクトを判断するには、(n + 21) 番目のvarbindの値IncidentNodeHostname (1.3.6.1.4.1.11.2.17.19.2.2.21) と、(n + 24) 番目のvarbindの値IncidentNodeMgmtAddr (1.3.6.1.4.1.11.2.17.19.2.2.24) を調べてください。nはMIBでトラップに定義されているvarbindの数です。

NNMiが管理するデバイスのいずれかがNorthboundアプリケーションにトラップを送信する場合、Northboundアプリケーションで重複デバイストラップを管理する必要があります。

トラップ転送メカニズムの比較については、『NNMiデプロイメントリファレンス』の「トラップおよびインシデント転送」を参照してください。

インシデントライフサイクル状態変化通知

このセクションの情報は、[**HP NNMi – Northbound Interfaceデスティネーション**] ページの [**送信オプション**] で行った選択によって異なります。

エンハンスド解決済みしたトラップ

Northbound転送先にエンハンスド解決済み通知が含まれる場合、NNMiのインシデントのライフサイクル状態が [解決済み] に変化したときに、NNMiはEventLifecycleStateClosed (1.3.6.1.4.1.11.2.17.19.2.0.1000) トラップをNorthboundアプリケーションに送信します。EventLifecycleStateClosedトラップは、元のインシデントのデータの多くを含んでいます。前のライフサイクル状態の値は含んでいません。EventLifecycleStateClosedトラップは、6番目のvarbindであるIncidentUuid (1.3.6.1.4.1.11.2.17.19.2.2.6) で元のインシデントを識別します。

状態変化トラップ

Northbound転送先にライフサイクル状態変更通知が含まれる場合、NNMiのインシデントのライフサイクル状態が [進行中]、[完了]、または [解決済み] に変化したときに、NNMiは LifecycleStateChangeEvent (1.3.6.1.4.1.11.2.17.19.2.0.1001) トラップをNorthboundアプリケーションに送信します。Northboundアプリケーションは、LifecycleStateChangeEventと元のインシデントを関連付けできます。

LifecycleStateChangeEventトラップは、以下のvarbindで元のインシデントとライフサイクル状態の変化を識別します。

- IncidentUuid、6番目のvarbind
  (1.3.6.1.4.1.11.2.17.19.2.2.6)

  この値は、管理イベントの6番目のvarbindの値、またはサードパーティSNMPトラップvarbindの (n+6) 番目のvarbindの値と一致します。
- IncidentLifecycleStatePreviousValue、7番目のvarbind
  (1.3.6.1.4.1.11.2.17.19.2.2.200)
- IncidentLifecycleStateCurrentValue、8番目のvarbind
  (1.3.6.1.4.1.11.2.17.19.2.2.201)

以下の表は、ライフサイクル状態に使用できる整数値を示したものです。

名前	整数値
登録済み	1
進行中	2
完了	3
解決済み	4
抑止済み	5

インシデント相関処理通知

Northbound転送先にインシデント相関処理通知が含まれる場合、NNMiの因果関係分析でインシデントが相関処理されると、NNMiはインシデント相関処理トラップをNorthboundアプリケーションに送信します。Northboundアプリケーションはトラップ内の情報を使用して相関変更を複製することができます。

単一相関トラップ

単一相関トラップオプションの場合、この統合では、以下の相関トラップを送信します。

- EventDedupCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1100)
- EventImpactCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1101)
- EventPairwiseCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1102)
- EventRateCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1103)

- EventApaCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1104)
- EventCustomCorrelation (1.3.6.1.4.1.11.2.17.19.2.0.1105)

各トラップは、以下のvarbindにおいて、1つの親子インシデント相関関係を示します。

- IncidentCorrelationIndicatorParentUuid、6番目のvarbind (1.3.6.1.4.1.11.2.17.19.2.2.6)
- IncidentCorrelationIndicatorChildUuid、7番目のvarbind (1.3.6.1.4.1.11.2.17.19.2.2.300)

グループ相関トラップ

グループ相関トラップオプションの場合、この統合では、以下の相関トラップを送信します。

- EventDedupCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2100)
- EventImpactCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2101)
- EventPairwiseCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2102)
- EventRateCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2103)
- EventApaCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2104)
- EventCustomCorrelationGroup (1.3.6.1.4.1.11.2.17.19.2.0.2105)

各トラップは、以下のvarbindでにおいて、親子インシデント相関関係を示します。

- IncidentCorrelationIndicatorParentUuid、6番目のvarbind (1.3.6.1.4.1.11.2.17.19.2.2.6)
- IncidentCorrelationIndicatorChildCount、7番目のvarbind (1.3.6.1.4.1.11.2.17.19.2.2.301)
- IncidentCorrelationIndicatorChildUuidCsv、8番目のvarbind (1.3.6.1.4.1.11.2.17.19.2.2.302)

  この値は子インシデントUUIDのカンマ区切りリストです。

インシデント削除通知

Northbound転送先にインシデント削除通知が含まれる場合、インシデントがNNMiで削除されると、NNMiはEventDeleted (1.3.6.1.4.1.11.2.17.19.2.0.3000) トラップをNorthboundアプリケーションに送信します。EventDeletedトラップは、6番目のvarbindであるIncidentUuid (1.3.6.1.4.1.11.2.17.19.2.2.6) で元のインシデントを識別します。

イベント転送フィルター

Northbound転送先にインシデントフィルターが含まれる場合、選択した設定オプションに応じて、フィルターのオブジェクトID (OID) には、以下のイベントタイプが包含または除外されます。

- NNMi管理イベントインシデント
- サードパーティSNMPトラップ
- EventLifecycleStateClosedトラップ

- LifecycleStateChangeEventトラップ
- EventDeletedトラップ
- 相関関係通知トラップ

  以下の注は、相関関係通知トラップに適用されます。

  - インシデントフィルターが相関処理に親インシデントを転送しない場合、NNMiは相関関係通知トラップをNorthboundアプリケーションに送信しません。
  - インシデントフィルターが相関処理に子インシデントを転送しない場合、転送される相関関係通知トラップにその子インシデントのUUIDは含まれません。(相関関係通知トラップに子インシデントUUIDが含まれない場合、NNMiはそのトラップをNorthboundアプリケーションに送信しません。)
  - DuplicateCorrelation管理イベントは、EventDedupCorrelationまたはEventDedupCorrelationGroup相関関係通知トラップとは無関係に転送されます。同様に、RateCorrelation管理イベントはEventRateCorrelationまたはEventRateCorrelationGroup相関関係通知トラップとは無関係に転送されます。インシデントフィルターがこれらの相関関係通知トラップのいずれかを転送しない場合でも、NNMiにより関連管理イベントが転送される場合があります。

NNMiノースバウンドインタフェースの変更

NNMiノースバウンドインタフェースの設定パラメーターを変更するには、以下の手順を実行します。

1. NNMiコンソールで、**[HP NNMi – Northbound Interfaceデスティネーション]** フォーム (**[統合モジュールの設定] > [Northboundインタフェース]**) を開きます。
2. 転送先を選択し、**[編集]** をクリックします。
3. 該当するように値を変更します。

   このフォームのフィールドの詳細については、「[NNMi Northbound Interfaceデスティネーション] フォームのリファレンス」(237ページ)を参照してください。
4. フォームの上端の **[有効にする]** チェックボックスがオンであることを確認し、フォームの下端の **[送信]** をクリックします。

   変更はただちに有効になります。

NNMiノースバウンドインタフェースの無効化

Northbound転送先が無効な間は、SNMPトラップはキューイングされません。

Northboundアプリケーションへの NNMiの転送を中止するには、以下の手順を実行します。

1. NNMiコンソールで、**[HP NNMi – Northbound Interfaceデスティネーション]** フォーム (**[統合モジュールの設定] > [Northboundインタフェース]**) を開きます。

2. 転送先を選択し、[**編集**] をクリックします。

   または、[**削除**] をクリックして、選択した転送先の設定をすべて削除します。
3. フォームの上端の [**有効にする**] チェックボックスをオフにし、フォームの下端の [**送信**] をクリックします。

   変更はただちに有効になります。

NNMiノースバウンドインタフェースのトラブルシューティング

NNMiノースバウンドインタフェースが正常に機能しない場合は、以下の手順を実行して問題を解決してください。

1. トラップ転送先ポートがファイアウォールによってブロックされていないことを確認します。

   NNMi管理サーバーが、ホストとポートによってNorthboundアプリケーションを直接処理できることを確認します。
2. 統合が正常に実行されていることを確認します。
   a. NNMiコンソールで、**[HP NNMi – Northbound Interfaceデスティネーション]** フォーム (**[統合モジュールの設定] > [Northboundインタフェース]**) を開きます。
   b. 転送先を選択し、[**編集**] をクリックします。
   c. [**有効にする**] オプションが選択されていることを確認します。
3. Northbound転送先に管理イベントが含まれる場合は、この機能を確認します。
   a. NNMiコンソールの [**解決済みの重要なインシデント**] ビューで、任意のインシデントを開きます。
   b. インシデントライフサイクル状態を [**登録済み**] に設定して、 [**保存**] をクリックします。
   c. インシデントライフサイクル状態を [**解決済み**] に設定して、 [**保存して閉じる**] をクリックします。
   d. 30秒後、Northboundアプリケーションがこのインシデントの EventLifecycleStateClosed トラップ (または LifecyleStateChangeEvent トラップ) を受信したかどうかを確認します。
      - Northboundアプリケーションがトラップを受信した場合は、手順4を続行します。
      - Northboundアプリケーションがトラップを受信しなかった場合は、異なるNorthboundアプリケーションに接続する新規Northbound転送先を設定した後で、手順aからこのテストを繰り返します。

        再テストに合格した場合、問題は最初のNorthboundアプリケーションにあります。アプリケーションのドキュメントでトラブルシューティング情報を参照してください。

        再テストに不合格になった場合は、HPサポートにご連絡ください。
4. Northbound転送先にSNMPトラップが含まれる場合は、この機能を確認します。
   a. NNMi管理サーバーで以下のコマンドを入力することにより、NNMiトポロジ内のノードに対するSNMPトラップを生成します。

**nnmsnmpnotify.ovpl -u username -p password – a \
discovered_node NNMi_node linkDown**

`discovered_node`はNNMiトポロジのノードのホスト名またはIPアドレス、`NNMi_node`はNNMi管理サーバーのホスト名またはIPアドレスです。

b. 30秒後に、Northboundアプリケーションが転送されたトラップを受信したがどうかを確認します。
   - Northboundアプリケーションがトラップを受信した場合、NNMi Northboundインタフェースは正常に機能しています。
   - Northboundアプリケーションがトラップを受信しなかった場合は、異なるNorthboundアプリケーションに接続する新規Northbound転送先を設定した後で、手順aからこのテストを繰り返します。

     再テストに合格した場合、問題は最初のNorthboundアプリケーションにあります。アプリケーションのドキュメントでトラブルシューティング情報を参照してください。

     再テストに不合格になった場合は、HPサポートにご連絡ください。

アプリケーションフェイルオーバーとNNMiNorthboundインタフェース

NNMi管理サーバーがNNMiアプリケーションフェイルオーバーに関係することになる場合、ここでの情報は、Northboundレシーバーにトラップを送信するNNMi Northboundアプリケーションを実装するすべての統合に適用されます。

NNMiがNorthboundアプリケーションに送信するトラップには、NmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2) のNNMi URLが含まれます。アプリケーションフェイルオーバー前に受信したトラップは、現在のスタンバイNNMi管理サーバーを参照します。URLがスタンバイNNMi管理サーバーを指す場合、そのURL値を使用するすべてのアクション (たとえば、NNMiコンソールの起動) は失敗します。

ローカルNorthboundアプリケーション

Northboundアプリケーションのトラップ受信コンポーネントがNNMi管理サーバー上にある場合は、以下の考慮事項がNNMi Northboundインタフェースの設定に適用されます。

- Northboundアプリケーションのトラップ受信コンポーネントは、アクティブおよびスタンバイNNMi管理サーバーに同じようにインストールおよび設定する必要があります。 両方のNNMi管理サーバーの同じポートでSNMPトラップ受信を設定します。
- プライマリNNMi管理サーバーでのみ、NNMiノースバウンドインタフェースを設定します。

  [HP NNMi – Northbound Interfaceデスティネーション] フォームの **[ホスト]** 識別で、**[NNMi FQDN]** または **[ループバックを使用]** オプションを選択します。

NNMiノースバウンドインタフェースは、起動時に、現在のNNMi管理サーバーの正しい名前またはIPアドレスを判断します。このように、Northboundインタフェースは、トラップをアクティブなNNMi管理サーバー上のNorthboundアプリケーションのトラップ受信コンポーネントに送信します。

リモートNorthboundアプリケーション

Northboundアプリケーションのトラップ受信コンポーネントがNNMi管理サーバー上にない場合は、NNMi Northboundインタフェースをプライマリ NNMi管理サーバーにのみ設定します。[**HP NNMi – Northbound Interfaceデスティネーション**] フォームの [**ホスト**] 識別で、[**その他**] オプションを選択します。

[NNMi Northbound Interfaceデスティネーション] フォームのリファレンス

[**HP NNMi – Northbound Interfaceデスティネーション**] フォームには、NNMiとNorthboundアプリケーション間の通信設定パラメーターがあります。このフォームは、[**統合モジュールの設定**] ワークスペースから使用できます。([**HP NNMi – Northbound Interfaceデスティネーション**] フォームで、[**新規作成**] をクリックするか、または転送先を選択して、[**編集**] をクリックします)。

注: AdministratorロールのNNMiユーザーのみが [**HP NNMi – Northbound Interfaceデスティネーション**] フォームにアクセスできます。

[**HP NNMi – Northbound Northbound Interfaceデスティネーション**] フォームには、以下の領域の情報が表示されます。

- 「Northboundアプリケーションの接続パラメーター」(237ページ)
- 「NNMi Northboundインタフェース統合の内容」(239ページ)
- 「NNMi Northboundインタフェース転送先のステータス情報」(241ページ)

統合設定に変更を適用するには、[**HP NNMi – Northbound Interfaceデスティネーション**] フォームの値を更新し、[**送信**] をクリックします。

Northboundアプリケーションの接続パラメーター

以下の表は、Northboundアプリケーションへの接続設定用パラメーターをリストしたものです。

Northboundアプリケーションの接続情報

フィールド	説明
ホスト	Northboundアプリケーションのトラップ受信コンポーネントを含むサーバーの完全修飾ドメイン名 (推奨) またはIPアドレス。 統合では、以下のサーバーの識別方法がサポートされています。

Northboundアプリケーションの接続情報 (続き)

フィールド	説明
	• **NNMi FQDN** NNMiがNNMi管理サーバー上のNorthboundアプリケーションへの接続を管理し、[**ホスト**] フィールドが読み取り専用になります。 これが、NNMi管理サーバー上でのNorthboundアプリケーションの推奨設定です。 • **ユーザーループバック** NNMiがNNMi管理サーバー上のNorthboundアプリケーションへの接続を管理し、[**ホスト**] フィールドが読み取り専用になります。 • **その他** Northboundアプリケーションサーバーを識別するホスト名またはIPアドレスを、[**ホスト**] フィールドに入力します。 NNMiは、[**ホスト**] フィールドのホスト名またはIPアドレスがループバックアダプターとして設定されていないことを確認します。 これがデフォルト設定です。 **注:** NNMi管理サーバーがNNMiアプリケーションフェイルオーバーに参加する場合にアプリケーションフェイルオーバーが統合に与える影響については、「アプリケーションフェイルオーバーとNNMiNorthboundインタフェース」(236ページ) を参照してください。
ポート	NorthboundアプリケーションがSNMPトラップを受信するUDPポート。 Northboundアプリケーション固有のポート番号を入力します。 **注:** Northboundアプリケーションのトラップ受信コンポーネントがNNMi管理サーバー上にある場合、このポート番号は、NNMiコンソールの [**通信の設定**] フォームの [**SNMPポート**] フィールドで設定した、NNMiがSNMPトラップを受信するために使用するポートと別にする必要があります。
コミュニティ文字列	トラップを受信するNorthboundアプリケーションの読み取り専用コミュニティ文字列。 Northboundアプリケーション設定で、受信したSNMPトラップにコミュニティ文字列が必要な場合は、その値を入力します。 Northboundアプリケーション設定で、特定のコミュニティ文字列が不要な場合は、デフォルト値のpublicを使用します。

NNMi Northboundインタフェース統合の内容

Northboundインタフェースの内容設定情報に、NNMi Northboundインタフェースが Northboundアプリケーションに送信する内容を設定するためのパラメーターをリストします。

NNMi Northboundインタフェースの内容設定情報

フィールド	説明
インシデント	インシデント転送の指定。 • **管理** NNMiは、NNMiが生成した管理イベントのみをNorthboundアプリケーションに転送します。 • **サードパーティSNMPトラップ** NNMiは、NNMiが管理対象デバイスから受信するSNMPトラップのみをNorthboundアプリケーションに転送します。 • **Syslog** NNMiは、NNMiが管理対象デバイスから受信するArcSight Syslogメッセージのみを NorthBound統合モジュールを使用してNorthboundアプリケーションに転送します。 NNMiは、Northbound転送先を有効にするとただちにインシデントの転送を開始します。 詳細については、「インシデント転送」(230ページ)を参照してください。
ライフサイクル状態の変化	インシデント変更通知の仕様。 • **解決済みに変化** NNMiは、ライフサイクル状態が [解決済み] に変化したインシデントごとに、インシデント解決済みトラップをNorthboundアプリケーションに送信します。 これがデフォルト設定です。 • **状態が変化した** NNMiは、ライフサイクル状態が [進行中]、[完了]、または [解決済み] に変化したインシデントごとに、インシデントのライフサイクル状態変化トラップをNorthboundアプリケーションに送信します。 • **両方** NNMiは、ライフサイクル状態が [解決済み] に変化したインシデントごとに、インシデント解決済みトラップをNorthboundアプリケーションに送信します。また、この統合では、ライフサイクル状態が [進行中]、[完了]、または [解決済み] に変化したインシデントごとに、インシデントのライフサイクル状態変化トラップをNorthboundアプリケーションに送信します。

NNMi Northboundインタフェースの内容設定情報 (続き)

フィールド	説明
	注: この場合、インシデントが [解決済み] ライフサイクル状態に変化するたびに、インシデント解決済みトラップとインシデントライフサイクル状態変更トラップの2つの通知トラップが統合によって送信されます。 詳細については、「インシデントライフサイクル状態変化通知」(231ページ)を参照してください。
相関処理	インシデント相関処理通知の仕様。 • **なし** NNMiは、NNMi因果関係分析によるインシデント相関処理結果をNorthboundアプリケーションに通知しません。 これがデフォルト設定です。 • **単一** NNMiは、NNMi因果関係分析で判明した親子インシデント相関関係ごとにトラップを1つ送信します。 • **グループ** NNMiは、親インシデントに相関するすべての子インシデントをリストした相関処理ごとに、トラップを1つ送信します。 詳細については、「インシデント相関処理通知」(232ページ)を参照してください。
削除	インシデント削除の仕様。このセクションは、[**インシデント**] フィールドでの選択項目に対して、削除トラップをNorthboundアプリケーションに送信するかどうかを設定します。 • **送信しない** NNMiは、インシデントがNNMiで削除されてもNorthboundアプリケーションに通知しません。 これがデフォルト設定です。 • **送信** NNMiは、NNMiで削除されるインシデントごとに、削除トラップをNorthboundアプリケーションに送信します。 詳細については、「インシデント削除通知」(233ページ)を参照してください。
NNMiコンソールアクセス	NorthboundアプリケーションからNNMiコンソールを参照するURLの接続プロトコル仕様。NNMiがNorthboundアプリケーションに送信するトラップのNmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2) には、NNMi URLが含まれます。

NNMi Northboundインタフェースの内容設定情報 (続き)

フィールド	説明
	設定ページのデフォルトは、NNMi設定と一致する設定になります。 NNMiコンソールがHTTPとHTTPS両方の接続を承認するよう設定されている場合、NNMi URLでHTTP接続プロトコルの指定を変更できます。たとえば、Northboundアプリケーションのすべてのユーザーがイントラネット上にある場合は、NorthboundアプリケーションからNNMiコンソールへのアクセスをHTTP経由に設定できます。NorthboundアプリケーションからNNMiコンソールに接続するプロトコルを変更する場合は、必要に応じて、[**HTTP**] オプションまたは [**HTTPS**] オプションを選択します。
インシデントフィルター	Northboundアプリケーションに送信されたイベントをフィルターするために統合で使用されるオブジェクトID (OID) のリスト。各フィルターエントリは、有効な数値OID (たとえば、.1.3.6.1.6.3.1.1.5.4.1.3.6.1.4.1.9) またはOIDプレフィックス (たとえば、.1.3.6.1.6.3.1.1.5.*) にすることができます。 次のいずれかのオプションを選択します。 • **なし** NNMiはすべてのイベントをNorthboundアプリケーションに送信します。これがデフォルト設定です。 • **含む** NNMiは、フィルターで識別されたOIDと一致する特定のイベントのみを送信します。 • **除外する** NNMiは、フィルターで識別されたOIDと一致する特定のイベントを除くすべてのイベントを送信します。 インシデントフィルターを指定します。 • フィルターエントリを追加するには、下側のテキストボックスにテキストを入力してから、[**追加**] をクリックします。 • フィルターエントリを削除するには、上側のボックスのリストからエントリを選択して、[**削除**] をクリックします。 詳細については、「イベント転送フィルター」(233ページ)を参照してください。

NNMi Northboundインタフェース転送先のステータス情報

以下の表に、Northbound転送先の読み取り専用ステータス情報を示します。この情報は、統合が現在機能しているか確認する場合に役立ちます。

NNMi Northboundインタフェース転送先のステータス情報

フィールド	説明
トラップ先IPアドレス	転送先ホスト名の解決先となるIPアドレス。 この値は、このノースバウンド転送先に固有です。
アップタイム (秒)	Northboundコンポーネントが最後に起動されてからの時間 (秒)。NNMiがNorthboundアプリケーションに送信するトラップのsysUptimeフィールド (1.3.6.1.2.1.1.3.0) にはこの値が含まれます。 この値は、NNMi Northboundインタフェースを使用するすべての統合に対して同じです。最新の値を表示するには、リフレッシュするか、フォームを閉じて再び開いてください。
NNMi URL	NNMiコンソールに接続するためのURL。NNMiがNorthboundアプリケーションに送信するトラップのNmsUrl varbind (1.3.6.1.4.1.11.2.17.19.2.2.2) にはこの値が含まれます。 この値は、このノースバウンド転送先に固有です。

NNMi Northboundインタフェースで使用されるMIB情報

特定のMIBをNNMiにロードし、NNMi Northbound統合によって送信されるインシデント通知で使用される管理情報を表示するには、以下の手順を実行します。

1. コマンドプロンプトで、**`nnmloadmib.ovpl -load hp-nnmi.mib`**コマンドを実行して`hp-nnmi.mib`ファイルをロードします。
2. コマンドプロンプトで、**`nnmloadmib.ovpl -load p-nnmi-registration.mib`**コマンドを実行して`hp-nnmi-registration.mib`ファイルをロードします。
3. コマンドプロンプトで、**`nnmloadmib.ovpl -load hp-nnmi-nbi.mib`**コマンドを実行して`hp-nnmi-nbi.mib`ファイルをロードします。
4. 省略可能な追加の手順:コマンドプロンプトで、**`nnmloadmib.ovpl -load hp-nnmi-ispi-perf-nbi.mib`**コマンドを実行して`hp-nnmi-ispi-perf-nbi.mib`ファイルをロードします。
5. NNMiコンソールから、[**設定**] ワークスペースを開きます。
6. [**MIB**] -> [**ロード済みMIB**] をクリックします。
7. ロードした各MIBをダブルクリックし、[**MIB変数**] をクリックしてMIB情報を表示します。

第5章: NNMiのメンテナンス

このセクションでは以下の章について説明します。

NNMiのバックアップおよび復元ツール

どのようなビジネスでも、中断することなく業務を確実に継続して行うには、バックアップおよび復元に関して優れた方針を持つことが重要です。HP Network Node Manager i Software (NNMi) は、ネットワークを運用する上で重要な資産であり、定期的にバックアップする必要があります。

NNMiインストールに関連した重要データは、以下の2種類です。

- ファイルシステム内のファイル
- リレーショナルデータベース (組み込みまたは外部) のデータ

この章では、重要なNNMiファイルおよびデータをバックアップおよび復元するためにNNMiで装備しているツールについて説明しています。

この章には、以下のトピックがあります。

バックアップコマンドと復元コマンド

NNMiには、NNMiデータをバックアップおよび復元するために以下のスクリプトがあります。

- `nnmbackup.ovpl` - 必要なすべてのファイルシステムデータ (設定情報を含む) とNNMi組み込みデータベースに保管されたデータをバックアップします。
- `nnmrestore.ovpl` - `nnmbackup.ovpl`スクリプトを使用して作成されたバックアップを復元します。

- nnmbackupembdb.ovpl - NNMi組み込みデータベース (ファイルシステムデータではない) の完全バックアップを、NNMiの稼働中に作成します。
- nnmrestoreembdb.ovpl - nnmbackupembdb.ovplスクリプトを使用して作成されたバックアップを復元します。
- nnmresetembdb.ovpl - NNMi組み込みデータベーステーブルをドロップします。ovstartコマンドを実行してテーブルを再作成します。

コマンド構文については、該当するリファレンスページ、またはLinuxのマニュアルページを参照してください。

NNMiデータのバックアップ

NNMiバックアップコマンド (nnmbackup.ovpl) は、主要なNNMiファイルシステムデータ、およびNNMi Postgresデータベースのテーブルの一部またはすべてを、指定されたターゲットディレクトリにコピーします。

各バックアップ操作では、ターゲットディレクトリ内のnnm-bak-<TIMESTAMP>という名前の親ディレクトリにファイルを保存します。-noTimestampオプションを指定すると、ディスク容量を節約できます。 -noTimestampオプションを使用すると、親ディレクトリは単にnnm-bakという名前になります。-noTimestampオプションを使用した以前のバックアップの後でバックアップを実行すると、以前のバックアップはnnm-bak.previousに名前変更され、それによってロールバックアップが作成されます。バックアップデータが失われないように、この名前変更は、2回目のバックアップが完了した後で実行されます。

NNMiバックアップコマンドにより、バックアップデータのtarアーカイブを作成したり、独自のツールを使用してバックアップファイルを圧縮したりできます。これで、適切なツールを使用してバックアップのコピーを保存できます。

ヒント: NNMi実装でOracleをメインNNMiデータベースとして使用する場合は、NNMiファイルシステムデータでのみNNMiバックアップコマンドと復元コマンドを使用できます。外部データベースの保守は、既存のデータベースバックアップおよび復元手順の一環として扱う必要があります。

バックアップデータと復元データには、ご使用のネットワーク環境にインストールされているNNM iSPIsすべてのデータが含まれていることも、含まれていないこともあります。詳細については、各NNM iSPIに付属のドキュメントで確認してください。

注意: ファイルをロックするソフトウェア (たとえば、ウイルス対策ソフトウェアやシステムバックアップソフトウェア) は、すべてNNMiデータベースへのNNMiのアクセスを妨害する可能性があります。これにより、ウイルス対策アプリケーションなど、他のプロセスで使用されているファイルに対する読み取りまたは書き込みができなくなるような問題が生じる可能性があります。NNMi Postgresデータベースの場合は、NNMiデータベースディレクトリ (Windowsの%NNM_DB%、Linuxの$NNM_DB) を除外するようにアプリケーションを設定してください。NNMiデータベースを定期的にバックアップするには、nnmbackup.ovplを使用します。

詳細については、nnmbackup.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

バックアップタイプ

NNMiのバックアップコマンドでは、2種類のバックアップがサポートされます。

- オンラインバックアップはNNMiの稼働中に行われます。NNMiでは、バックアップされたデータ内でデータベーステーブルが確実に同期されます。オンラインバックアップ中でも、オペレーターは制約を受けることなくNNMiコンソールを使用することができ、他のプロセスはNNMiデータベースとやりとりできます。オンラインバックアップを実行することにより、「バックアップ領域」(245ページ)の説明に従って、機能に応じてNNMiのデータすべてまたはデータの一部のみをバックアップできます。組み込みNNMiデータベースの場合は、nmsdbmgrサービスが実行されている必要があります。外部データベースの場合、このバックアップにはNNMiファイルシステムデータが含まれます。外部データベースをバックアップするために、NNMiプロセスが実行されている必要はありません。
- オフラインバックアップは、NNMiが完全に停止している間に行われます。オフラインバックアップでは、バックアップ領域がファイルシステムのファイルにのみ適用されます。オフラインバックアップには、バックアップ領域に関係なく、必ずNNMiデータベースの全体が含まれます。組み込みNNMiデータベースの場合、このバックアップではPostgresデータベースのファイルがコピーされます。外部データベースの場合、このバックアップにはNNMiファイルシステムデータのみが含まれます。

バックアップ領域

NNMiバックアップコマンドでは、NNMiのバックアップ量を定義する領域をいくつか指定できます。

設定領域

設定領域 (-scope config) は、大まかにはNNMiコンソールの [**設定**] ワークスペース内の情報と一致します。

設定領域には以下のデータが含まれます。

- オンラインバックアップの場合は、NNMi設定情報を保存している組み込みデータベーステーブルのみ。
- オフラインバックアップの場合は、組み込みデータベース全体。
- 全バックアップの場合は、設定領域ファイルとディレクトリのリストに示すファイルシステム内のNNMi設定情報。

トポロジ領域

トポロジ領域 (-scope topology) は、大まかにはNNMiコンソールの [**インベントリ**] ワークスペース内の情報と一致します。ネットワークトポロジが依存している設定はそのトポロジの検出に使用されているため、トポロジ領域には設定領域が含まれます。

トポロジ領域には以下のデータが含まれます。

- オンラインバックアップの場合は、NNMi設定情報とネットワークトポロジ情報を保存している組み込みデータベーステーブルのみ。
- オフラインバックアップの場合は、組み込みデータベース全体。
- 全バックアップの場合は、以下の表にリストされたファイルシステム内のNNMi設定情報。現在、トポロジ領域に関連付けられているファイルシステムのファイルはありません。

イベント領域

イベント領域 (`-scope event`) は、大まかにはNNMiコンソールの [**インシデントの参照**] ワークスペース内の情報と一致します。イベントはこれらのイベントに関連したネットワークトポロジに依存しているため、イベント領域には設定領域とトポロジ領域が含まれます。

イベント領域には以下のデータが含まれます。

- オンラインバックアップの場合は、NNMi設定情報、ネットワークトポロジ情報、およびイベント情報を保存している組み込みデータベーステーブルのみ。
- オフラインバックアップの場合は、組み込みデータベース全体。
- 全バックアップの場合は、以下の表にリストされたファイルシステム内のNNMi設定情報と、イベント領域ファイルとディレクトリにリストされたNNMiイベント情報。

全領域

完全バックアップ (`-scope all`) には、NNMiのすべての重要ファイルと組み込みデータベース全体が含まれます。

設定領域ファイルとディレクトリ

ディレクトリまたはファイル名	説明
%NnmInstallDir%/conf (Windowsのみ)	設定情報
%NnmInstallDir%\misc\nms\lic $NnmInstallDir/misc/nms/lic	その他のライセンス情報
%NnmInstallDir%\nmsas\server\nms\conf $NnmInstallDir/nmsas/server/nms/conf	jbossの設定
%NnmDataDir%\conf $NnmDataDir/conf	他のHP製品が共有する設定
%NnmDataDir%\conf\nnm\props $NnmDataDir/conf/nnm/props	ローカルNNMi設定のプロパティファイル
%NnmDataDir%\shared\nnm\conf\licensing\LicFile.txt $NnmDataDir/shared/nnm/conf/licensing/LicFile.txt	ライセンス情報
%NnmDataDir%\NNMVersionInfo $NnmDataDir/NNMVersionInfo	NNMiバージョン情報ファイル

設定領域ファイルとディレクトリ(続き)

ディレクトリまたはファイル名	説明
%NnmDataDir%\shared\nnm\user-snmp-mibs $NnmDataDir/shared/nnm/user-snmp-mibs	共有されるユーザー追加のSNMP MIB情報
%NnmDataDir%\shared\nnm\actions $NnmDataDir/shared/nnm/actions	共有されるライフサイクルの移行アクション
%NnmDataDir%\shared\nnm\certificates $NnmDataDir/shared/nnm/certificates	共有NNMiSSL証明書
%NnmDataDir%\shared\nnm\conf $NnmDataDir/shared/nnm/conf	共有NNMi設定情報
%NnmDataDir%\shared\nnm\conf\licensing $NnmDataDir/shared/nnm/conf/licensing	共有NNMiライセンス設定情報
%NnmDataDir%\shared\nnm\lrf $NnmDataDir/shared/nnm/lrf	共有されるNNMiコンポーネント登録ファイル
%NnmDataDir%\shared\nnm\conf\props $NnmDataDir/shared/nnm/conf/props	共有されるNNMi設定のプロパティファイル
%NnmDataDir%\shared\nnm\www\htdocs\images $NnmDataDir/shared/nnm/www/htdocs/images	共有されるNNMiノードグループマップの背景イメージ

このコンテキストで、共有ディレクトリのファイルは、NNMiアプリケーションフェイルオーバーまたは高可用性環境の別のNNMi管理サーバーと共有されるファイルです。

イベント領域ファイルとディレクトリ

ディレクトリまたはファイル名	説明
$NnmDataDir/log/nnm/signin.0.0.log	NNMiコンソールサインインログ

NNMiデータの復元

NNMi復元スクリプト(`nnmrestore.ovpl`)は、バックアップデータをNNMi管理サーバーに配置します。バックアップの種類と領域により、NNMiで復元可能なバックアップデータが決まります。

注: `nnmrestore.ovpl`スクリプトを使用してデータベースレコードを2番目のNNMi管理サーバーに配置する場合は、どちらのNNMi管理サーバーも同じタイプのオペレーティングシステム、NNMiバージョン、およびパッチレベルである必要があります。

あるNNMi管理サーバーから2番目のNNMi管理サーバーにバックアップデータを配置すると、これらの両方のサーバーに同じデータベースUUIDが存在することになります。2番目のNNMi管理サーバーでNNMiを復元したら、元のNNMi管理サーバーからNNMiをアンインストールします。

NNMiをアンインストールする前に、最新のパッチから開始して、NNMiパッチをすべて逆順で削除します。パッチの削除プロセスは、NNMi管理サーバーで実行しているオペレーティングシステムによって異なります。インストールおよび削除手順については、パッチのマニュアルを参照してください。

- オンラインバックアップを復元するため、NNMiは、ファイルシステムデータを正しい場所にコピーし、バックアップのデータベーステーブルの内容を上書きします。上書きするのは、バックアップの復元以後に削除されたオブジェクトと、バックアップの削除以後に作成されたオブジェクトです。また、バックアップの実行後に変更されたすべてのオブジェクトは、バックアップ時の状態に戻されます。組み込みNNMiデータベースの場合は、`nmsdbmgr`サービスが実行されている必要があります。外部データベースの場合、復元にはNNMiファイルシステムデータのみが含まれ、実行中のNNMiプロセスが存在しないようにする必要があります。
- オフラインバックアップを復元するため、NNMiは、ファイルシステム内のPostgresファイルを上書きし、データベースファイルをバックアップデータで完全に置き換えます。外部データベースの場合、このバックアップにはNNMiファイルシステムデータのみが含まれます。

`-force`オプションを指定すると、`nnmrestore.ovpl`コマンドはすべてのNNMiプロセスを停止し、`nmsdbmgr`サービスを開始し (NNMi組み込みデータベースのオンラインバックアップからの復元の場合)、データを復元し、その後すべてのNNMiプロセスを再開始します。

指定されたソースがtarファイルの場合は、NNMi復元コマンドにより、現在の作業ディレクトリの一時フォルダーにtarファイルが抽出されます。この場合、現在の作業ディレクトリに十分な記憶領域があるため一時フォルダーを使用できることを確認するか、復元コマンドを実行する前にアーカイブを抽出してください。

注: NNMiのあるバージョンから次のバージョンへデータベースのスキーマが変わる恐れがあるため、データバックアップをNNMiの異なるバージョン間で共有することはできません。

注: NNMiでは、バックアップからの復元の後にトポロジ、状態、およびステータスが自動的に再同期されます。

再同期中にNNMiを停止しないでください。再同期を確実に完了するには、バックアップからの復元の後でNNMiを数時間実行し続けます。実際の所要時間は、ノード数、状態変化の量、および再同期中に受信されたトラップデータによって異なります。

再同期が完了する前にNNMiを停止する必要がある場合は、再同期をもう一度実行して完了する必要があります。

管理サーバー全体の再同期を手動で実行するには、`nnmnoderediscover.ovpl -all -fullsync`を実行します。

同じシステムでの復元

1つのシステムでバックアップコマンドと復元コマンドを使用することにより、データを復旧できます。バックアップの実行時から復元の実行時までの間に、以下の項目が変更されていないようにする必要があります。

- NNMiのバージョン (パッチを含む)
- オペレーティングシステムタイプ
- キャラクターセット (言語)
- ホスト名
- ドメイン

異なるシステムでの復元

バックアップコマンドと復元コマンドを使用して、NNMi管理サーバーから他の管理サーバーへデータを転送することができます。異なるシステムでの復元の用途には、システム障害からの復旧や、オペレーティングシステムのアップグレード時のNNMiの異なるシステムへの転送などがあります。

注: NNMi UUIDがデータベースの復元中にターゲットシステムにコピーされるため、ソースとターゲットの両システムがNNMiの同じインスタンスを実行している可能性があります。ソースシステムからNNMiをアンインストールしてください。

NNMiをアンインストールする前に、最新のパッチから開始して、NNMiパッチをすべて逆順で削除します。パッチの削除プロセスは、NNMi管理サーバーで実行しているオペレーティングシステムによって異なります。インストールおよび削除手順については、パッチのマニュアルを参照してください。

ヒント: グローバルネットワーク管理を導入する間など、同様の設定で機能するNNMi管理サーバーを複数作成するには、`nnmconfigexport.ovpl`および`nnmconfigimport.ovpl`コマンドを使用します。

異なるシステムの復元では、両方のシステムで以下の項目を同じにする必要があります。

- NNMiのバージョン (パッチを含む)
- OSのタイプとバージョン
- キャラクターセット (言語)

以下の項目は、2つのシステム間で異なっていても構いません。

- ホスト名
- ドメイン

異なるシステムでの復元の場合、`nnmrestore.ovpl`コマンドはライセンス情報を新規システムにコピーしません。新しいNNMi管理サーバーの新規ライセンスを取得して適用してください。詳細については、「NNMiのライセンス」(312ページ)を参照してください。

バックアップと復元の方針

このセクションでは、バックアップおよび復元に関する以下の方針について説明します。

すべてのデータを定期的にバックアップする

ディザスターリカバリ計画には、すべてのNNMiデータの完全バックアップを定期的に実行するスケジュールを含めてください。このバックアップを作成するためにNNMiを停止する必要はありません。バックアップをスクリプトに組み込む場合は、-forceオプションを使用して、バックアップが開始される前にNNMiが正しい状態になるようにしてください。次に例を示します。

```
nnmbackup.ovpl -force -type online -scope all -archive
  -target nnmi_backups\periodic
```

ハードウェアに障害が発生したためにNNMiデータを復旧する必要が生じた場合は、以下の手順を実行します。

1. ハードウェアを再構成するか、新規ハードウェアを取得します。
2. バックアップデータの場合と同じバージョンおよびパッチレベルのNNMiをインストールします。
3. NNMiデータを復元します。
   - リカバリNNMi管理サーバーが「同じシステムでの復元」(249ページ)の一覧にある要件を満たす場合は、以下の例のようなコマンドを実行します。

     ```
     nnmrestore.ovpl -force -lic
     -source nnmi_backups\periodic\newest_backup
     ```

   - リカバリNNMi管理サーバーが同じシステムでの復元を行うのに適格ではなくても、「異なるシステムでの復元」(249ページ)の一覧にある要件を満たす場合は、以下の例に似たコマンドを実行します。

     ```
     nnmrestore.ovpl -force
       -source nnmi_backups\periodic\newest_backup
     ```

     必要に応じてライセンスを更新します。

設定変更前のデータのバックアップ

設定変更を開始する前に、領域を限定したバックアップ(「バックアップ領域」(245ページ) の説明に従って) を必要に応じて実施してください。このようにすると、設定を変更しても期待した効果が見られない場合、周知の作動設定に戻すことが可能になります。例:

```
nnmbackup.ovpl -type online -scope config
-target nnmi_backups\config
```

このバックアップを同じNNMi管理サーバーに復元するには、すべてのNNMiプロセスを停止してから、以下の例のようなコマンドを実行します。

```
nnmrestore.ovpl -force -source nnmi_backups\config\newest_backup
```

NNMiまたはオペレーティングシステムのアップグレード前のバックアップ

大規模なシステム変更 (NNMiまたはオペレーティングシステムのアップグレードを含む) を行う前に、すべてのNNMiデータの完全バックアップを実行します。バックアップの実行後NNMiデータベースに対する変更が何も行われないようにするため、すべてのNNMiプロセスを停止し、オフラインバックアップを作成してください。例:

```
nnmbackup.ovpl -type offline -scope all
-target nnmi_backups\offline
```

システムの変更後にNNMiが正常に実行されなくなった場合は、変更をロールバックするか、または異なるNNMi管理サーバーを設定し、「異なるシステムでの復元」(249ページ) にリストされた要件が確実に満たされるようにしてください。その後、以下の例に似たコマンドを実行します。

```
nnmrestore.ovpl -lic -source nnmi_backups\offline\newest_backup
```

ファイルシステムのファイルのみの復元

データベーステーブルに影響を与えることなくNNMiファイルを上書きするには、以下の例に似たコマンドを実行します。

```
nnmrestore.ovpl -partial
-source nnmi_backups\offline\newest_backup
```

このコマンドは、NNMi実装のメインNNMiデータベースとしてOracleを使用する場合に役立ちます。

組み込みデータベースのみをバックアップおよび復元する

NNMiでは、`nnmbackupembdb.ovpl`コマンドと`nnmrestoreembdb.ovpl`コマンドにより、NNMi組み込みデータベースのみをバックアップおよび復元します。この機能は、NNMiの設定においてデータのスナップショットを作成する場合に便利です。`nnmbackupembdb.ovpl`コマンドと`nnmrestoreembdb.ovpl`コマンドは、オンラインバックアップのみを実行します。最低でも、`nmsdbmgr`サービスが実行されている必要があります。

詳細については、`nnmbackup.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

各バックアップ操作では、ターゲットディレクトリ内のnnm-bak-<TIMESTAMP>という名前の親ディレクトリにファイルを保存します。-noTimestampオプションを指定すると、ディスク容量を節約できます。 -noTimestampオプションを使用すると、親ディレクトリは単にnnm-bakという名前になります。-noTimestampオプションを使用した以前のバックアップの後でバックアップを実行すると、以前のバックアップはnnm-bak.previousに名前変更され、それによってロールバックアップが作成されます。バックアップデータが失われないように、この名前変更は、2回目のバックアップが完了した後で実行されます。

注: nnmresetembdb.ovplコマンドは、組み込みデータベースにデータを復元する前に実行してください。このコマンドによりデータベースにエラーが含まれないようになるため、データベース制約違反が発生する可能性がなくなります。組み込みデータベースリセットコマンドの実行については、nnmresetembdb.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

高可用性 (HA) 環境におけるバックアップおよび復元ツールの使用

このセクションでは、高可用性環境でバックアップおよび復元ツールを使用する場合に役立つヒントについて説明します。

HA環境でのバックアップのベストプラクティス

HA環境でNNMiバックアップツールを使用するときは、以下のベストプラクティスを検討してください。

- アクティブ (プライマリ) システムを使用してバックアップを実行する(設定ファイルが古かったり、共有ディスク情報が含まれていなかったりするため (バックアップノードは共有ディスクにアクセスできないため)、バックアップ (セカンダリ) ノードのバックアップはお勧めできません)。
- 共有ディスクはアクティブノードに接続する。cronジョブを使用している場合、共有ディスクがマウントされていることを確認します。
- システムをメンテナンスモードにする (フェイルオーバーをトリガーしないように)。
- アクティブノードでのみnnmbackup.ovplスクリプトを使用してオンラインバックアップを実行する。
- 定期的にオフラインバックアップを実行する。

詳細については、nnmbackup.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

HA環境での復元のベストプラクティス

HA環境でNNMi復元ツールを使用するときは、以下のベストプラクティスを検討してください。

- 共有ディスクがマウントされていることを確認する。
- システムがメンテナンスモードになっていることを確認する。
- `nnmrestore.ovpl`スクリプトを使用して復元を実行する。

詳細については、nnmrestore.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

HA環境でNNMiを使用する方法の詳細については、「高可用性クラスターにNNMiを設定する」(172ページ)を参照してください。

NNMiの保守

NNMi管理サーバーが機能するようになったら、複数のNNMi機能を最適化するためにメンテナンス作業を実施することができます。

この章には、以下のトピックがあります。

- 「管理上停止中のインタフェースのIPアドレスに対するモニタリングの抑制」(294ページ)
- 「大規模スイッチのVLANインデックス付けの使用を抑制する」(294ページ)
- 「計画停止」(296ページ)
- 「センサーステータスの設定」(296ページ)
- 「インタフェースの入力速度と出力速度のインポート」(301ページ)

NNMiフォルダーのアクセス制御リストの管理

「アクションサーバー名のパラメーターの設定」(259ページ)に示されているように、HP NNM Action Serverを実行するユーザー名の変更が必要な状況が発生する場合があります。ユーザー名の権限を変更せずにアクションサーバーを実行するユーザー名を変更すると、HP NNM Action Serverが起動しなくなり、インシデントアクションの実行中にNNMiがメッセージを記録しなくなる可能性があります。このセクションでは、この発生を防ぐ方法について説明します。

NNMi (Everest) には、以下のディレクトリを変更する権限が含まれています。

- /var/opt/OV/log/nnm/public
- /var/opt/OV/shared/perfSpi

NNMi Everestの/var/opt/OV/log/nnm/publicフォルダーに対する既定の権限は755ですが、NNMiはACLを使用して、データベースユーザー (nmsdbmgr) およびnnmactionユーザー (bin) のアクセス権を調整します。NNMi Everestのポストインストール (インストールまたはアップグレードスクリプトの一部) 中に、インストールスクリプトによって/var/opt/OV/log/nnm/publicフォルダーの権限が変更され、ACLが追加されます。

インストールスクリプトが予期しないエラーによって/var/opt/OV/log/nnm/publicフォルダーにACLを設定できない場合、スクリプトは/var/opt/OV/log/nnm/publicフォルダーをワールド (その他のユーザー) により書き込み可能にし、NNMiインストールは正常に完了します。NNMiインストールの成功後、/var/opt/OV/log/nnm/publicフォルダーへのワールドによる書き込み権限を制限するには、NNMi管理サーバーのオペレーティングシステムにACLを設定するためのシステム管理者マニュアルを参照してください。

/var/opt/OV/log/nnm/publicフォルダーのユーザーアクセスを調整するには、Linux ACL (アクセス制御リスト) を使用します。ACLの設定は、owner/group/otherの権限を拡張するのに役立ちます。ACLは、Linuxのすべてのプラットフォーム (RedHatおよびSuSE) でサポートされています。

たとえば、以下のコマンドの実行後、USER変数で示されたユーザーは**/var/opt/OV/log/nnm/public**フォルダーへの書き込み権限を取得します。以下のコマンドを実行しないと、**/var/opt/OV/log/nnm/public**フォルダーの権限は755で、root以外のユーザーはディレクトリ内のファイルに書き込めません。

```
setfacl -m user:<USER>:rwx /var/opt/OV/log/nnm/public
```

setfaclコマンドの使用方法については、Linuxのマニュアルページを参照してください。

ノードグループの設定

NNMiには、ノードグループの設定を自動化できるコマンドラインツールが用意されています。nnmnodegroup.ovpl コマンドでは、ノードグループを作成、表示、変更、および削除できます。

詳細については、nnmnodegroup.ovpl のリファレンスページ、またはLinuxのマニュアルページを参照してください。

ノードグループマップ設定の構成

ノードグループマップの設定は、NNMiコンソールだけでなく、nnmnodegroupmapsettings.ovplコマンドラインツールを使用して行うこともできます。nnmnodegroupmapsettings.ovplツールでは、ノードグループマップの設定を作成、変更、および削除できます。このツールを使用して、TXT、XML、またはCSV形式で現在のノードグループマップの設定を表示することもできます。

ヒント: NNMiを現在実行しているWebブラウザーをリフレッシュすると、ノードグループマップの設定に加えた変更がただちに反映されます。

詳細については、nnmnodegroupmapsettings.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

通信設定の構成

nnmcommunication.ovplコマンドラインツールを使用して、NNMi通信設定を行うことができます。nnmcommunication.ovplツールでは、通信設定を作成、表示、変更、削除できます。このツールでは、テキストテーブル、テキストリスト、またはXML形式でリストを生成できます。

管理者は、nnmcommunication.ovpl ツールを使用して、管理アドレスやコミュニティ文字列などのフィールドのSNMPエージェント設定をロックして直接管理することで、通常の設定をバイパスすることもできます。

nnmcommunication.ovplツールは、SNMPプロキシポートまたはSNMPプロキシアドレスのロード、追加、削除をサポートしていません。プロキシ設定は廃止され、今後のリリースで削除されます。

詳細については、nnmcommunication.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

カスタムポーラー収集エクスポートの管理

NNMiカスタムポーラー機能では、SNMP MIB式を使用してNNMiがポーリングする必要のある追加情報を指定することによって、積極的にネットワーク管理を行えます。

カスタムポーラー収集は、収集 (ポーリング) する情報および収集したデータのNNMiによる処理方法を定義します。詳細については、NNMiヘルプの「カスタムポーラー収集を作成する」および「カス

タムポーリングを設定する」を参照してください。『HP Network Node Manager i Softwareステップバイステップガイド (カスタムポーラーに関するホワイトペーパー)』も参照してください。

カスタムポーラー機能を使用する場合でも、処理が終わったファイルをエクスポートディレクトリから削除するのはユーザーの責任です。

注: 長期の保存にエクスポートファイルを使用しないでください。設定された最大ディスク容量を超えると、NNMiによって古いファイルが削除され、新しいファイルが作成されます。これらのファイルを処理して別の場所に保存していないと、ファイルは失われます。

カスタムポーラー収集のエクスポートディレクトリの変更

NNMiは、ユーザーがエクスポートした収集データを以下のディレクトリに書き込みます。

- Windowsの場合: `%NNmDataDir%\shared\nnm\databases\custompoller\export`
- Linuxの場合: `$NnmDataDir/shared/nnm/databases/custompoller/export`

NNMiがカスタムポーラーファイルを書き込むディレクトリを変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\nms-custompoller.properties`
   - Linuxの場合: `$NNM_PROPS/nms-custompoller.properties`
2. `exportdir`エントリを特定します。このエントリは以下の行のように記述されています。

   #!com.hp.nnm.custompoller.exportdir=<base directory to export custom poller metrics>

   NNMiがカスタムポーラー収集情報を`C:\CustomPoller`ディレクトリに書き込むように設定するには、以下のように行を変更します。

   com.hp.nnm.custompoller.exportdir=C:\CustomPoller
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

カスタムポーラー収集のエクスポートに使用する最大ディスク容量の変更

`collection_name.csv`ファイルにデータをエクスポートするときにNNMiが使用する最大ディスク容量を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS%\nms-custompoller.properties
   - Linuxの場合: $NNM_PROPS/nms-custompoller.properties
2. maxdiskspaceエントリを特定します。このエントリは以下の行のように記述されています。

   #!com.hp.nnm.custompoller.maxdiskspace=1000

   各collection_name.csvファイルに最大2,000MB (2GB) のストレージ容量を確保するようにNNMiを設定するには、行を以下のように変更します。

   com.hp.nnm.custompoller.maxdiskspace=2000
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

カスタムポーラーメトリックスの累積周期の変更

NNMiは、データをファイルに書き込む前に、カスタムポーラー収集メトリックスを累積する期間を分単位で設定します。

カスタムポーラーメトリックスの累積周期を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS%\nms-custompoller.properties
   - Linuxの場合: $NNM_PROPS/nms-custompoller.properties
2. 以下のような行を探します。

   #!com.hp.nnm.custompoller.accumulationinterval=5

   デフォルト値である5分間ではなく10分間、メトリックスを収集するようにNNMiを設定するには、その行を以下のように変更します。

   com.hp.nnm.custompoller.accumulationinterval=10
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

インシデントアクションの管理

アクションは、インシデントライフサイクルの任意の時点で自動的に実行されるように設定できます。たとえば、設定しているタイプのインシデントが生成されるときにあるアクションが発生するよ

うに設定するとします。詳細については、NNMiヘルプの「インシデントのアクションを設定する」を参照してください。

アクションのパラメーターを調整するには、次の項に示す手順に従ってください。

注: 望まない結果 (予期せぬメモリ使用量の増大、イベントアクション処理時間の延長など) を避けるには、イベントアクション処理のデフォルトのプロパティ値を変更しないことをお勧めします。

同時アクション数の設定

NNMiが実行できる同時アクション数を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\shared\nnmaction.properties`
   - Linuxの場合: `$NNM_PROPS/shared/nnmaction.properties`
2. 以下のような行を探します。

   #!com.hp.ov.nms.events.action.numProcess=10

   デフォルト値ではなく、20個の同時アクションを実行できるようにNNMiを設定するには、その行を以下のように変更します。

   com.hp.ov.nms.events.action.numProcess=20

   注: 行の始めにある**#!**文字を必ず削除してください。
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

Jythonアクションのスレッド数の設定

jythonスクリプトを実行するためにアクションサーバーが使用するスレッド数を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\shared\nnmaction.properties`
   - Linuxの場合: `$NNM_PROPS/shared/nnmaction.properties`
2. 以下のような行を探します。

   #!com.hp.ov.nms.events.action.numJythonThreads=10

デフォルトのスレッド数ではなく、20個のスレッドでjythonスクリプトを実行できるようにNNMiを設定するには、その行を以下のように変更します。

com.hp.ov.nms.events.action.numJythonThreads=20

注: 行の始めにある**#!**文字を必ず削除してください。

3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

アクションサーバー名のパラメーターの設定

WindowsオペレーティングシステムでNNMi管理サーバーを実行している場合、HP NNM Action ServerはLocal Systemアカウントの**Windows**サービスとして実行されます。つまり、アクションサーバーでアクションを実行するには、Local Systemアカウントを使用する必要があります。

Windows NNMi管理サーバーでHP NNM Action Server Windowsサービスを実行するユーザー名を変更するには、HP NNM Action Serverサービスの**LogOn**プロパティを変更します。

LinuxオペレーティングシステムでNNMi管理サーバーを実行している場合、アクションサーバーはbinユーザー名で実行されます。これらのオペレーティングシステムでアクションサーバーを実行するユーザー名を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   $NNM_PROPS/nnmaction.properties

2. 以下のような行を探します。

   #!com.hp.ov.nms.events.action.userName=bin

   デフォルト値ではなく、rootがアクションサーバーを実行するようにNNMiを設定するには、その行を以下のように変更します。

   com.hp.ov.nms.events.action.userName=root

   注: 行の始めにある**#!**文字を必ず削除してください。

3. 変更を保存します。
4. アクションサーバーを再起動します。
   a. NNMi管理サーバーで**ovstop nnmaction**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart nnmaction**コマンドを実行します。

アクションサーバーのキューサイズを変更する

トラップストームへの応答など、高実行率でLongアクションコマンド文字列を使用するアクションの場合、アクションサーバーは多くのメモリを使用する可能性があります。アクションサーバーのパ

フォーマンスを上げるために、HPではアクションサーバーで利用可能なメモリサイズが制限されています。

これらの制限を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - %NNM_PROPS%\shared\nnmaction.properties
   - $NNM_PROPS/shared/nnmaction.properties
2. 以下のような2行を探します。
   - com.hp.ov.nms.events.action.jvmargs.minMemsize=-Xms6m
   - com.hp.ov.nms.events.action.jvmargs.maxMemsize=-Xmx30m
3. 上記のパラメーターでは、最小メモリサイズが6MBに、最大が30MBに設定されていることがわかります。これらのパラメーターをニーズに合わせて調整します。
4. 変更を保存します。
5. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

インシデントアクションログ

アクションを実行すると、関連付けられたインシデントアクションログファイルに出力が記録されます。選択したインシデントのログの内容を表示するには、**[ツール] > [インシデントアクションログ]** メニューオプションを使用します。次の表にログに含まれる項目を示します。

インシデントアクションログ項目

項目	説明
コマンド	インシデントの発生時に実行するスクリプト
インシデント名	インシデント設定で定義されたインシデント名
インシデントUUID	インシデントのUUID ([**登録**] タブ)
コマンドタイプ	コマンドのタイプ ([**Jython**] または [**ScriptOrExecutable**])
ライフサイクル状態	インシデントのライフサイクル状態 ([**登録済み**]、[**進行中**]、[**完了**]、または[**解決済み**])
終了コード	コマンドのリターンコード (エラーコードと同様)
標準出力	アクションの標準出力

インシデントアクションログ項目 (続き)

項目	説明
標準エラー	標準エラー出力
実行ステータス	アクションごとに判別されるステータス

server.propertiesファイルの設定の上書き

注: システムには2つのserver.propertiesファイルがある場合があります。

以下のファイルは製品のインストーラーによって作成され、アプリケーションインスタンス用にアプリケーションサーバーをカスタマイズするプロパティが含まれています。このファイルはユーザーによる変更は不可能で、コードメンテナンス (アップグレードおよびパッチ) で置き換えられます。

Windowsの場合: `%NnmDataDir%\NNM\server\server.properties`

Linuxの場合: `$NnmDataDir/NNM/server/server.properties`

以下のファイルは、ユーザーによって独自の環境用にアプリケーションを設定するために使用され、製品によってアップグレードまたはパッチで変更されることはありません。このファイルは、その他のファイルで設定された値を上書きします。そのため、すべてのカスタマイズはこのファイルで実行されます。

Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

Linuxの場合 : `$NnmDataDir/nmsas/NNM/server.properties`

このセクションでは、`nmsas/NNM/server.properties`ファイルの以下の設定の上書き方法について説明します。

「ブラウザーのロケール設定の上書き」(261ページ)

「SNMP Setオブジェクトアクセス権限の設定」(263ページ)

「リモートアクセスには暗号化を必須とするようにNNMiを設定する」(264ページ)

ブラウザーのロケール設定の上書き

以下のserver.propertiesファイルを使用して、ブラウザーのロケール値に関係なく、指定されたロケール値をすべてのNNMiクライアントに強制的に適用できます。

Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`

server.propertiesファイルを使用してこの値が設定されている場合、ブラウザーのロケール値は無視されます。

ブラウザーのロケール設定を上書きするには、以下の手順を実行します。

1. server.propertiesファイルを開きます。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`
2. `nmsas.server.forceClientLocale`に移動します。
3. nmsas.server.forceClientLocaleを以下のいずれかに設定します。

   `nmsas.server.forceClientLocale`= <two-letter ISO Language code>

   たとえば、ISO言語コードのみを使用してロケールを英語に設定するには、以下のように入力します。

   nmsas.server.forceClientLocale = en

   `nmsas.server.forceClientLocale`= <two-letter ISO Language code>_<two-letter ISO country code>

   たとえば、ISO言語コードと国コードを使用してロケールを英語に設定するには、以下のように入力します。

   nmsas.server.forceClientLocale = en_US
4. NNMi ovjbossサービスを再起動します。

   NNMi管理サーバーで`ovstop ovjboss`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

注: `server.properties`ファイルへの変更は、ovjbossの起動時にのみ読み取られます。

詳細については、`server.properties`ファイル内のコメントを参照してください。

インシデントを割り当てるときのユーザー名のソート順序に使用されるロケールの設定

NNMi管理者は、インシデントを割り当てるときにユーザー名のソート順序を決定するために使用される、NNMi管理サーバーの言語ロケールを指定できます。

注: 設定したソート順序ロケールは、[**インシデントの割り当て**] ダイアログにのみ適用されます。

アルファベット順を決定するときに、NNMiはユーザーの実際のログイン名ではなく表示名を使用し、大文字を小文字と分けてソートすることはありません。

注: ソート順序を決定する際にNNMiが使用するのは、`sortLocale`に設定されているロケールだけです。`forceClientLocale`プロパティに設定されているブラウザーロケールがソート順序に

影響を与えることはありません。詳細については、「ブラウザーのロケール設定の上書き」(261 ページ)を参照してください。

注: 高可用性 (HA) 下でファイル変更を行う場合、更新する必要があるserver.propertiesファイルの場所は、`<Shared_Disk>/NNM/dataDir/nmsas/NNM/server.properties`です。

インシデントを割り当てるときに表示されるユーザー名のソート順序に使用する言語ロケールを設定するには、以下の方法で`server.properties`ファイルを編集します。

1. 以下のファイルを開きます。
   - Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`
   - Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`
2. `server.properties`ファイルの次の行をコメント解除します。

   ```
   #nmsas.server.sortLocale = en_US
   ```
3. デフォルトの値を、NNMi管理サーバーの正しいロケールに変更します。たとえば、ロケールをロシア語に変更するには、次のエントリを使用します。

   ```
   nmsas.server.sortLocale = ru_RU
   ```
4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

SNMP Setオブジェクトアクセス権限の設定

以下のファイルを使用して、ユーザーがアクセスできるノードでSNMP Set機能を使用するために必要なオブジェクトアクセス権限を設定できます。

Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`

SNMP Set機能の詳細については、NNMiの「オペレーター用のヘルプ」を参照してください。オブジェクトアクセス権限の詳細については、NNMiの「管理者用のヘルプ」を参照してください。

SNMP Set機能に対するオブジェクトアクセス権限を設定するには、以下の手順を実行します。

1. `server.properties`ファイルを開きます。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`
2. 以下の行を追加します。

   `permission.override.com.hp.nnm.SNMP_SET=`<object access role>

   <object access role>で有効な値は以下のとおりです。

   com.hp.nnm.ADMIN

com.hp.nnm.LEVEL2

com.hp.nnm.LEVEL1

com.hp.nnm.GUEST

たとえば、[**オブジェクト管理者**] および [**オブジェクトオペレーターレベル2**] オブジェクトアクセス権限でSNMP Set機能を使用できるようにするには、以下のように入力します。

permission.override.com.hp.nnm.SNMP_SET=com.hp.nnm.ADMIN,com.hp.nnm.LEVEL2

3. アクセスを有効にする各オブジェクトアクセス権限を含めます。
4. NNMi ovjbossサービスを再起動します。

   NNMi管理サーバーで`ovstop ovjboss`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

注: `server.properties`ファイルへの変更は、ovjbossの起動時にのみ読み込まれます。

リモートアクセスには暗号化を必須とするようにNNMiを設定する

管理者は、ネットワークからNNMiへのHTTPやその他の非暗号化アクセスを無効にできます。

注: 暗号化リモートアクセスのみを許可するようにNNMiを設定する前に、グローバルネットワーク管理、NNM iSPIs、およびその他の統合がSSLをサポートしていることを確認します。暗号化リモートアクセスのみを許可するようにNNMiを設定する前に、これらをSSL用に設定します。

ネットワークからNNMiへのHTTPやその他の非暗号化アクセスを無効にするには、server.propertiesファイルを以下のように編集します。

1. 以下のファイルを編集します (ファイルが存在しない場合は作成が必要な場合があります)。
   - Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`
   - Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`
2. `server.properties`ファイルに以下の4行を追加します。

   nmsas.server.net.bind.address = 127.0.0.1

   nmsas.server.net.bind.address.ssl = 0.0.0.0

   nmsas.server.net.hostname = localhost

   nmsas.server.net.hostname.ssl = ${com.hp.ov.nms.fqdn}
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

上記の変更によって、NNMiはリモートシステムからのHTTP要求を「待機」しなくなりますが、ローカルホストアクセスによるHTTP要求はそのままサポートされます。

SNMPトラップの管理

このセクションでは、以下のタスクの実行方法について説明します。

- 「hosted-object-trapstorm.confファイルによるトラップストームのブロック」(265ページ)
- 「SNMPv1またはSNMPv2cを使用して管理されているノードまたは監視対象外のノードのSNMPv3トラップを認証するためのNNMiの設定」(266ページ)
- 「Causal Engineがトラップを受け入れる期間の設定」(268ページ)
- 「最も古いSNMPトラップインシデントの自動トリム機能の設定」(269ページ)
- 「プロキシSNMPゲートウェイによって送信されたトラップから元のトラップアドレスを判別するためのNNMiの設定」(273ページ)

hosted-object-trapstorm.confファイルによるトラップストームのブロック

NNMiには、ホスト元デバイス (インタフェースを含む) からのトラップストームをブロックする方法があります。

1. nnmtrapconfig.ovplスクリプトを実行します。nnmtrapconfig.ovplのリファレンスページ、またはLinuxのマニュアルページの説明に従って -hostedOnTrapstormおよび-hostedOnThresholdに適切な値を指定し、トラップサービスを設定します。プロパティの変更を反映させるようにトラップサーバーを再設定するには、-setPropパラメーターを使用します。
2. 必要に応じて既定の設定を変更するには、以下のファイルを編集します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf\hosted-object-trapstorm.conf
   - Linuxの場合: $NnmDataDir/shared/nnm/conf/hosted-object-trapstorm.conf

   hosted-object-trapstorm.confのリファレンスページ、またはLinuxのマニュアルページで示された形式に従って変更します。
3. hosted-object-trapstorm.confファイルを変更した場合、nnmtrapconfig.ovpl -stopに続いてnnmtrapconfig.ovpl -startを実行することでトラップサービスを再起動する必要があります。詳細については、nnmtrapconfig.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

   注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモー

ドにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

SNMPv1またはSNMPv2cを使用して管理されているノードまたは監視対象外のノードのSNMPv3トラップを認証するためのNNMiの設定

NNMiが以下のいずれかの条件を満たしているノードからSNMPv3トラップを受信している場合、このセクションの手順を実行します。

- デバイスがSNMPv2またはSNMPv1を使用して管理されている
- デバイスがNNMiによって検出されていない

これらのデバイスのSNMPv3エンジンIDをSNMPv3キャッシュに追加するようにNNMiを設定できます。このようにNNMiを設定することで、NNMiはこれらのSNMPv3トラップを認証して保存できます。

NMPv1またはSNMPv2cを使用して管理されているノードまたは検出されていないノードのSNMPv3トラップを受信して保存するようにNNMiを設定するには、以下の手順を実行します。

1. NNMiコンソールで、**[設定] > [通信設定]** に移動します。各受信トラップにトラップの認証に使用するための対応する設定が適用されるように、[領域] または [特定ノードの設定] レベルのデフォルトのエントリを設定します。詳細については、NNMiヘルプの「デフォルトSNMPv3の設定」を参照してください。

   ヒント: SNMPv3ノードの含まれるアドレス範囲の領域を使用するか、それぞれに対して [特定ノードの設定] を設定することをお勧めします。

2. NNMiコンソールで、**[設定] > [インシデント] > [インシデントの設定]** に移動します。
3. **[未解決のSNMPトラップおよびSyslogメッセージを破棄する]** を選択解除します。

   [未解決のSNMPトラップおよびSyslogメッセージを破棄する] の選択解除後、NNMiは管理していないノードから送信されたトラップを保持します。
4. NNMi管理サーバーで`ovstop`コマンドを実行します。
5. 以下のファイルを編集します。

   Windowsの場合: `%NNM_PROPS%\nms-communication.properties`

   Linuxの場合: `$NNM_PROPS/nms-communication.properties`
6. ファイルの最下部に以下の行を追加します。

   com.hp.nnm.snmp.engineid.file=<ファイルへのパス>file.txt

   <ファイルへのパス>file.txtエントリは、デバイスを含むファイルの完全なパスとファイル名です。

これらの設定の変更によって、NNMiはNNMiプロセスが再起動されるたびにこのファイルからのエントリをSNMPv3キャッシュに読み込みます。

注: Linux NNMi管理サーバーでは、ファイルパスは/var/opt/OV/etcなどの通常の形式です。

Windows NNMi管理サーバーでは、ドライブを無視し、区切り文字のスラッシュを使用します。たとえば、`C:/temp/file.txt`などのファイルは`/temp/file.txt`と指定します。

7. 変更を保存します。
8. <ファイルへのパス>file.txtファイルを編集します。
   a. デバイスのIPアドレス、ポート、およびエンジンIDの各項目をカンマで区切って追加します。
   b. 個別の行にデバイスごとに1つのエントリを追加します。

      エンジンIDは一連の16進数バイトです。NNMi は大文字と小文字を区別せず、スペースを認識します。

      以下の例を使用してエントリを作成します。

      16.1.2.3,161,80 00 00 09 30 00 00 1f e9 a3 33 01

      16.1.2.4,161,80 00 00 11 03 00 00 2d 51 99 30 00

      1050:0000:0000:0000:0005:0600:300c:326b, 161, 800000090300001f9ea33000

      ff06::c3,161,80 00 00 09 03 00 00 1f 9A A3 30 00
   a. NNMi管理サーバーで`ovstart`コマンドを実行し、NNMiを起動して<ファイルへのパス>file.txtファイルを読み込みます。
   b. Boot.logファイルで、NNMiがファイルを読み込んでいることを確認します。

      このファイルに、ファイルが読み込まれたことを示す以下のようなログメッセージが含まれていることを確認します。

      2012-10-17 14:44:44.876 WARNING [SnmpV3EngineIdCachePopulator] V3

      2012-10-17 14:45:08.017 INFO [SnmpV3EngineIdCachePopulator] Successfully loaded 3 V3

      Engine IDs from file /temp/patch2/v3hosts.txt

      ノードの有効な設定へのマッピングエラーが発生した場合は、以下のようなメッセージが含まれています。

      2012-10-17 14:45:03.485 WARNING [SnmpV3EngineIdCachePopulator] V3

      Engine IDs: Could not resolve SNMPv3 configuration for 16.1.2.6

      上記のようなメッセージが含まれている場合は、このノードの **[設定] > [通信の設定]** 設定を調整します。

注: <path to file>file.txtファイルだけでなくキャッシュからもエントリを削除する必要がある場合、<path to file>file.txtからエントリを削除してから、NNMiを再起動することが最良の方法です。

1. NNMi管理サーバーでovstopコマンドを実行します。
2. NNMi管理サーバーでovstartコマンドを実行します。

Causal Engineがトラップを受け入れる期間の設定

広範囲のネットワークが一定の予測可能な時間に利用できなくなる場合、NNMiではCausal Engineへのトラップの配信を阻止することでCausal Engineの分析負荷を抑制できます。トラップの配信を阻止するには、NNMi管理者として、NNMi Causal Engineがイベントシステムからのトラップの受け入れを停止する期間を設定します。

注: この機能は、NNMiコンソールに配信されるトラップには影響しません。

Causal Engineに配信されるトラップは、StatePollerをトリガーし、StatePollerのポーリングポリシーによって指示されたスケジュールより早くノードをポーリングする場合に使用されます。トラップの配信を阻止する場合、NNMiはStatePollerから更新情報を取得する前に、スケジュールされたポーリング間隔まで待機する必要があります。あらゆる場合において、NNMi Causal EngineはNNMi StatePollerからのステートフローを使用して、トラップがあるかないかにかかわらず同じ結論に達します。

Causal Engineがトラップの受け入れを停止する期間を設定するには、以下の手順を実行します。

1. 以下のファイルを作成します。

   Windowsの場合: `%NNM_PROPS%\shared\nms-apa.properties`

   Linuxの場合: `$NNM_PROPS/shared/nms-apa.properties`
2. ファイルに以下の内容を追加します。

   PROPERTY NAME: com.hp.ov.nms.apa.trapGateSchedule

   以下の例をガイドラインとして使用します。

   以下の例では、トラップは深夜に流れ、午前8:30に阻止され、午前10:00に再度流れてから、午後4:30に再度阻止されます。

   com.hp.ov.nms.apa.trapGateSchedule = ENABLE_APA_TRAPS 08:30 10:00 16:30

   以下の例では、トラップは深夜に阻止され、午前8:30に再度流れ、午前10:00に阻止されてから、午後4:30に再度流れます。

   com.hp.ov.nms.apa.trapGateSchedule = DISABLE_APA_TRAPS 08:30 10:00 16:30
3. 変更を保存します。
4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで ovstopコマンドを実行します。
   b. NNMi management serverでovstartコマンドを実行します。

最も古いSNMPトラップインシデントの自動トリム機能の設定

NNMiが常に高いパフォーマンスを発揮するように、NNMiはデータベース内に一定数のSNMPトラップを保存した後に着信SNMPトラップ (syslogメッセージを含む) をドロップします。最も古いSNMPトラップインシデントの自動トリム機能を使用して、NNMiデータベース内に保存するSNMPトラップ数を制御し、重要な着信SNMPトラップを保持できます。

注: NNMiは根本原因ではないSNMPトラップインシデントのみをトリムします。

最も古いSNMPトラップインシデントの自動トリム機能は、デフォルトでは無効になっています。最も古いSNMPトラップインシデントの自動トリム機能を有効にすると、NNMiはNNMiデータベースから最も古いSNMPトラップインシデントを削除します。

ヒント: SNMPトラップインシデントをNNMiデータベースから手動でトリムするには、`nnmtrimincidents.ovpl`スクリプトを使用します。詳細については、nnmtrimincidents.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

最も古いSNMPトラップインシデントの自動トリム機能の有効化 (インシデントアーカイブなし)

最も古いSNMPトラップインシデントの自動トリム機能を使用して、NNMiデータベース内のSNMPトラップインシデント数が60,000個を超えた場合は30,000個のSNMPトラップインシデント (syslogメッセージを含む) をトリムするとします。この例では、NNMiでSNMPトラップインシデントをトリムする前にアーカイブしません。以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS\nms-jboss.properties`
   - Linuxの場合: `$NNM_PROPS/nms-jboss.properties`
2. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.snmpTrapAutoTrimStartPercentage=50
3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.snmpTrapAutoTrimStartPercentage=60
4. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.snmpTrapAutoTrimPercentageToDelete=25
5. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.snmpTrapAutoTrimPercentageToDelete=50
6. 以下の行を含むテキストブロックを探します。

#!com.hp.nnm.events.snmpTrapAutoTrimSetting=Disabled

7. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.snmpTrapsAutoTrimSetting=TrimOnly

8. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

com.hp.nnm.events.snmpTrapMaxStoreLimitのデフォルト値は100,000です。この設定で以下の数式を使用することで、NNMiはNNMiデータベースに60,000個のSNMPトラップインシデント (syslogメッセージを含む) を保存した後に、NNMiデータベースから30,000個のSNMPトラップインシデントをトリムします。

com.hp.nnm.events.snmpTrapAutoTrimStartPercentage X

com.hp.nnm.events.snmpTrapMaxStoreLimit X

com.hp.nnm.events.snmpTrapAutoTrimPercentageToDelete

最も古いSNMPトラップインシデントの自動トリム機能の有効化 (インシデントアーカイブ有効)

最も古いSNMPトラップインシデントの自動トリム機能を使用して、NNMiデータベース内のSNMPトラップインシデント数が80,000個を超えた場合は60,000個のSNMPトラップインシデント (syslogメッセージを含む) をトリムするとします。この例では、NNMiでSNMPトラップインシデントをトリムする前にアーカイブします。以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS\nms-jboss.properties
   - Linuxの場合: $NNM_PROPS/nms-jboss.properties
2. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.snmpTrapAutoTrimStartPercentage=50

3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.snmpTrapAutoTrimStartPercentage=80

4. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.snmpTrapAutoTrimPercentageToDelete=25

5. この行をコメント解除し、以下のように編集します。

com.hp.nnm.events.snmpTrapAutoTrimPercentageToDelete=75

6. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.snmpTrapAutoTrimSetting=Disabled

7. この行を以下のように編集します。

   com.hp.nnm.events.snmpTrapsAutoTrimSetting=TrimAndArchive

8. NNMiを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

`com.hp.nnm.events.snmpTrapMaxStoreLimit`のデフォルト値は100,000です。この設定で以下の数式を使用することで、NNMiはNNMiデータベースに80,000個のSNMPトラップインシデント (syslogメッセージを含む) を保存した後にアーカイブし、NNMiデータベースから60,000個のSNMPトラップインシデントをトリムします。

com.hp.nnm.events.snmpTrapAutoTrimStartPercentage X

com.hp.nnm.events.snmpTrapMaxStoreLimit X

com.hp.nnm.events.snmpTrapAutoTrimPercentageToDelete

> **ヒント:** デフォルトのアーカイブファイルパスのカスタマイズ方法を含む、トラップインシデントアーカイブファイルの詳細については、`nnmtrimincidents.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

保存するSNMPトラップインシデント数の削減

NNMiで長期間SNMPトラップインシデントを保持する必要がない場合、NNMiデータベースに保存するSNMPトラップインシデント数を削減できます。

> **注:** NNMiは、データベース内のSNMPトラップインシデント数が100,000個に達すると、SNMPトラップ (syslogメッセージを含む) のドロップを開始します。この制限値をより高く設定するとNNMiのパフォーマンスが低下するため、制限値を高くすることはできません。

保存するSNMPトラップインシデント (syslogメッセージを含む) の最大数を50,000 SNMPトラップインシデントに削減するとします。これを行うには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS\nms-jboss.properties`
   - Linuxの場合: `$NNM_PROPS/nms-jboss.properties`
2. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.snmpTrapMaxStoreLimit=100000

3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.snmpTrapMaxStoreLimit=50000

4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

最も古いSNMPトラップインシデントの自動トリム機能の 監視

最も古いSNMPトラップインシデントの自動トリム機能の稼動状態を確認するには、**NNMiコンソール**から **[ヘルプ] > [システム情報] > [ヘルス]** をクリックします。NNMiは、最も古いSNMPトラップインシデントの自動トリム機能に関する以下のアラームも生成します。

- NNMiは、保存されたSNMPトラップインシデント (syslogメッセージを含む) の数が**com.hp.nnm.events.snmpTrapMaxStoreLimit**値の100%に達したときに危険域アラームを生成します。
- NNMiは、保存されたSNMPトラップインシデント (syslogメッセージを含む) の数が**com.hp.nnm.events.snmpTrapMaxStoreLimit**値の95%に達したときにsnmpTrapLimitMajorAlarmアラームを生成します。
- NNMiは、保存されたSNMPトラップインシデント (syslogメッセージを含む) の数が**com.hp.nnm.events.snmpTrapMaxStoreLimit**値の90%に達したときにsnmpTrapLimitWarningAlarmアラームを生成します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

最も古いSNMPトラップインシデントの自動トリム機能の無効化

最も古いSNMPトラップインシデントの自動トリム機能を無効にするには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS\nms-jboss.properties
   - Linuxの場合: $NNM_PROPS/nms-jboss.properties
2. 以下を含むテキストブロックを探します。

com.hp.nnm.events.snmpTrapAutoTrimSetting

3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.snmpTrapAutoTrimSetting=Disabled

4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

プロキシSNMPゲートウェイによって送信されたトラップから元のトラップアドレスを判別するためのNNMiの設定

NNMiのデフォルト設定を使用している場合、プロキシSNMPゲートウェイによって送信されたトラップには元のトラップアドレスが表示されない可能性があります。管理者は、元のトラップアドレスを判別するようにNNMiを設定できます。

以下の点に注意してください。

- NNMiにはカスタムインシデント属性cia.originaladdressが含まれます。NNMiはcom.hp.nnm.trapd.useUdpHeaderIpAddressプロパティと併せてcia.originaladdress属性の意味を判別します。
- com.hp.nnm.trapd.useUdpHeaderIpAddress パラメーターの値はデフォルトでfalseであるため、NNMiは通常cia.originaladdress属性を無視します。
- com.hp.nnm.trapd.useUdpHeaderIpAddress値をtrueに設定すると、 cia.originaladdress属性によってSNMPエージェントアドレスの値が提供されます。

NNMiでソースとしてUDPヘッダーアドレスを使用する一方で、管理対象デバイスの実際のSNMPアドレスへのアクセスが必要な場合、com.hp.nnm.trapd.useUdpHeaderIpAddress値をtrueに設定すると便利です。

注: com.hp.nnm.trapd.useUdpHeaderIpAddress属性がfalse (デフォルト設定) の場合、cia.originaladdressとcia.addressの両方の属性には同じ値が含まれます。

cia.originaladdressの値を使用して元のトラップアドレスを判別するようにNNMiを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NNM_PROPS%\nms-jboss.properties

   Linuxの場合: $NNM_PROPS/nms-jboss.properties

2. 以下の行を含むテキストブロックを探します。

#!com.hp.nnm.trapd.useUdpHeaderIpAddress=false

3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.trapd.useUdpHeaderIpAddress=true

4. 変更を保存します。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMiはcia.originaladdressの値を使用して元のトラップアドレスを判別します。

トラップアドレスの順序

NNMiは、ソースアドレスを以下のように分析します。

- com.hp.nnm.trapd.useUdpHeaderIpAddressプロパティがtrueに設定されたSNMPv1およびSNMPv2cトラップは、以下のアドレス順序を使用する。

  rfc3584TrapAddress (.1.3.6.1.6.3.18.1.3.0)

  nnmTrapForwardingAddress (.1.3.6.1.4.1.11.2.17.2.19.1.1.3.0)

  securityPackNotificationAddress (.1.3.6.1.4.1.99.12.45.2.1.0)

  proxyOid (.1.3.6.1.4.1.11.2.17.5.1.0)

  IPヘッダーのソースアドレス
- com.hp.nnm.trapd.useUdpHeaderIpAddress プロパティがfalseに設定されたSNMPv1トラップは、以下のアドレス順序を使用する。

  rfc3584TrapAddress (.1.3.6.1.6.3.18.1.3.0)

  nnmTrapForwardingAddress (.1.3.6.1.4.1.11.2.17.2.19.1.1.3.0)

  securityPackNotificationAddress (.1.3.6.1.4.1.99.12.45.2.1.0)

  proxyOid (.1.3.6.1.4.1.11.2.17.5.1.0)

  v1トラップのagent-addrフィールド

  IPヘッダーのソースアドレス

NNMi NmsTrapReceiverプロセス

NNMiには、フェイルオーバー時にSNMPトラップの損失を最小限に抑えるのに役立つスタンドアロンNmsTrapReceiverプロセスが備えられています。NmsTrapReceiverは、アクティブノードとスタンバイノードの両方で実行されます。

NmsTrapReceiverの設定

NNMiには、ユーザーが構成できる以下の設定があります。

- trapReceiverReplay

  trapReceiverReplay設定は、スタンバイノードがアクティブノードになった場合、フェイルオーバー後の起動時にトラップの再実行に使用される時間差です(デフォルトの時間は10秒)。

  注: trapReceiverReplay設定は、アプリケーションフェイルオーバー環境および高可用性 (HA) 環境にのみ適用されます。

- trapReceiverJmsTTL

  trapReceiverJmsTTLオプションは、TrapReceiverでトラップをキャッシュする最大時間を設定します。デフォルト設定は5分です。jbossのダウン時間がこの時間を超えると、データが失われます。

  ヒント: この設定を行う前に、フェイルオーバーの所要時間を計ってベンチマークを判断してから、trapReceiverJmsTTLをその時間の2倍に設定します。

このような設定の変更方法については、`nnmtrapconfig.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

注: 正しく動作するには、アクティブノードとスタンバイノードの間でクロックが同期されていることが重要です。同期されていないと、トラップの大量の重複または損失が生じる可能性があります。

詳細については、`nnmtrapconfig.ovpl`のリファレンスページ、またLinuxのマニュアルページを参照してください。

注: 高可用性でTrapReceiverに変更を加える場合は、クラスターの両方のノードに変更を加える必要があります。その後、TrapReceiverプロセスを停止して、再起動する必要があります (「NmsTrapReceiverプロセスの開始と停止」(275ページ)を参照)。

NmsTrapReceiverセキュリティ

NNMiには、NmsTrapReceiverのパスワードを変更できるnnmchangetrappw.ovplコマンドがあります。

注: 高可用性環境におけるアクティブなNNMi管理サーバーでパスワードを変更する場合は、スタンバイNNMi管理サーバーでNmsTrapReceiverを停止および再起動することをお勧めします。

詳細については、`nnmchangetrappw.ovpl` のリファレンスページ、またはLinuxのマニュアルページを参照してください。

NmsTrapReceiverプロセスの開始と停止

NmsTrapReceiverプロセスは、オペレーティングシステム (Linuxの場合: init.d nettrap、Windowsの場合: HP NNM NmsTrapReceiverサービス) によって自動的に開始されます。 また、ovstartで

NmsTrapReceiverプロセスが実行されていないことが検出された場合も、ovstartによって開始されます。

NmsTrapReceiverを手動で開始または停止する必要がある場合は、オペレーティングシステムのサービスを使用します。

注: ovstartおよびovstopコマンドは、リモートトラップサーバーではなく、トラップ処理のjbossパイプラインを開始および停止するのみです。

nnmtrapd.confファイルおよびtrapFilter.confファイルによるインシデントのブロック

NNMi管理サーバーに流れるインシデントの数が一定のレートに達して、新しく到着するインシデントをNNMiがブロックする場合、以下の点に注意してください。

- NNMiは`TrapStorm`インシデントを生成し、インシデントがブロックされていることを示します。
- NNMiは主要なヘルスメッセージも生成し、インシデントレートが高くてインシデントがブロックされていることを示すことがあります。

インシデント数を削減するには、以下のいずれかの方法を使用します。

- `nnmtrapd.conf`ファイルを使用し、インシデントがNNMiに入るのをブロックしてインシデントトラフィックの削減を試みます。

  注: `nnmtrapd.conf`ファイルによる方法を使用すると、NNMiは引き続きこれらのインシデントを使用してトラップレートを計算し、トラップバイナリストアに書き込みます。`nnmtrapd.conf`ファイルによる方法を使用しても、インシデントがデータベースで作成されたり保存されたりすることを停止することしかできません。

  詳細については、nnmtrapd.confのリファレンスページ、またはLinuxのマンページを参照してください。

- `trapFilter.conf`ファイルを使用し、NNMiイベントパイプラインで早期にインシデントをブロックして、このインシデントがトラップレート計算で分析されること、またはNNMiトラップバイナリストアに保存されることを回避します。

  ヒント: デバイスのIPアドレスまたはOIDを`trapFilter.conf`ファイルに追加すると、この大量のインシデントをブロックして、インシデントのボリュームの問題を回避できます。

  詳細については、trapFilter.confのリファレンスページ、またはLinuxのマンページを参照してください。

以前サポートされていたvarbind順序を保持するためのNNMiの設定

すべてのSNMPv2トラップには、1番目と2番目のvarbindとしてsysUptime.0 OIDとsnmpTrapOID.0 OIDが含まれています。

注: SNMPv2トラップ定義にトラップパラメーターとしてsysUptime.0またはsnmpOID.0が含まれている場合、varbindリストの1番目と2番目以外の位置に追加varbindとしてこれらがNNMiに表示される可能性があります。

NNMi 9.21 (パッチ1) より前は、NNMiはsysUpTime.0 OIDとsnmpTrapOID.0 OIDのすべてのインスタンスをvarbindリストから削除していました。

NNMi 9.21 (パッチ1) 以降では、NNMiはこれらのOIDがトラップ定義に含まれていて、受信したトラップのvarbindリストの1番目と2番目以外の位置にある場合、OIDを保持します。この変更によって、トラップパラメーターとしてsysUpTime.0 OIDまたはsnmpTrapOID.0 OIDが含まれるトラップのvarbind順序が変わります。

以下の例では、1番目のボールドのvarbindにsnmpTrapOID.0の値が含まれ、2番目のボールドのvarbindにsysUpTime.0の値が含まれています。この例に示されているように、これらのvarbindはvarbindリストの1番目と2番目以外の位置に追加varbindとして表示されます。

//0: SNMP MESSAGE (0x30): 115 bytes

//2: INTEGER VERSION (0x2) 1 bytes: 1 (SNMPv2C)

//5: OCTET-STR COMMUNITY (0x4) 6 bytes: "public"

//13: V2-TRAP-PDU (0xa7): 102 bytes

//15: INTEGER REQUEST-ID (0x2) 2 bytes: 18079

//19: INTEGER ERROR-STATUS (0x2) 1 bytes: noError(0)

//22: INTEGER ERROR-INDEX (0x2) 1 bytes: 0

//25: SEQUENCE VARBIND-LIST (0x30): 90 bytes

//27: SEQUENCE VARBIND (0x30): 13 bytes

//29: **OBJ-ID (0x6) 8 bytes: .1.3.6.1.2.1.1.3.0**

//39: TIMETICKS (0x43) 1 bytes: 9

//42: SEQUENCE VARBIND (0x30): 32 bytes

//**44: OBJ-ID (0x6) 10 bytes:.1.3.6.1.6.3.1.1.4.1.0**

//56: OBJ-ID (0x6) 18 bytes: .1.3.6.1.6.3.1.1.5.3.1.3.6.1.4.1.9.1.14

//76: SEQUENCE VARBIND (0x30): 14 bytes

//78: OBJ-ID (0x6) 9 bytes: .1.3.6.1.2.1.2.2.1.1

//89: INTEGER (0x2) 1 bytes: 92

//92: SEQUENCE VARBIND (0x30): 23 bytes

//94: OBJ-ID (0x6) 10 bytes: .1.3.6.1.6.3.1.1.4.3.0

//106: OBJ-ID (0x6) 9 bytes: .1.3.6.1.4.1.11.2.3.14

ヒント: NNMiで`sysUpTime.0` OIDと`snmpTrapOID.0` OIDのすべてのインスタンスをvarbindリストから削除する場合にのみ、`com.hp.nnm.events.preserveOldVarbindListOrder`プロパティをtrueに設定します。

元のNNMiの動作を保持するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: `%NNM_PROPS%\nms-jboss.properties`

   Linuxの場合: `$NNM_PROPS/nms-jboss.properties`
2. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.events.preserveOldvarbindListOrder=false
3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.events.preserveOldvarbindListOrder=true
4. 変更を保存します。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

ICMPエコー要求パケットのデータペイロードサイズの設定

ネットワークレイテンシの1つの定義は、ICMPパケットがターゲットデバイスへのラウンドトリップを完了して戻ってくるまでの時間です。低レイテンシの測定値は、ネットワークがより効率的であることを意味します。

ネットワークレイテンシをテストする1つの一般的な方法は、NNMiによって管理される管理アドレスのICMPポーリング間隔およびICMPエコー要求パケットのデータペイロードサイズを調整することです。パケットが大きい場合は小さい場合に比べてネットワークレイテンシが長くなることを考慮し、NNMiでは異なるパケットサイズを使用してネットワークレイテンシの測定をテストできます。

ノードグループ内のノードまたはインタフェースグループ内のインタフェースに属するIPアドレスに対して、NNMiがICMPエコー要求パケットで送信するデータペイロードサイズを設定できます。たとえば、ネットワークレイテンシを比較するため、管理アドレスのポーリング時間を調整しながら、ノードグループまたはインタフェースグループに送信されるICMPエコー要求パケットのサイズを変更できます。

ノードグループ内のノードおよびインタフェースグループ内のインタフェースに属するアドレスに対して異なるペイロードサイズを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NNM_PROPS%\nms-mon-config.properties

   Linuxの場合: $NNM_PROPS/nms-mon-config.properties
2. 以下を含むテキストブロックを探します。

   #!com.hp.nnm.icmp.payload.sizeInBytes=4096
3. この行を以下のようにコメント解除し、4096の値を必要なペイロード値に変更します。

   com.hp.nnm.icmp.payload.sizeInBytes=4096

   sizeInBytesパラメーターに使用できる最小値は12バイトで最大値は65492バイトです。

注: データペイロードサイズを設定するには、少なくとも1つのグループのプロパティを定義する必要があります。以下の手順の説明に従って、グループのプロパティ定義も行わない場合、NNMiはcom.hp.nnm.icmp.payload.sizeInBytesプロパティを無視します。

1. 以下を含むテキストブロックを探します。

   #!com.hp.nnm.icmp.nodegroup.name=My Node Group
2. この行を以下のようにコメント解除し、私のノードグループ設定をNNMiモニタリングの設定で参照するノードグループに変更します。

   com.hp.nnm.icmp.nodegroup.name=My Node Group

   注: 指定するノードグループ名は、NNMiモニタリングの設定で参照されるノードグループにする必要があります。
3. 以下を含むテキストブロックを探します。

   #!com.hp.nnm.icmp.ifacegroup.name=My Interface Group
4. この行を以下のようにコメント解除し、私のインタフェースグループ設定をNNMiモニタリングの設定で参照するインタフェースグループに変更します。

   com.hp.nnm.icmp.ifacegroup.name=My Interface Group

   注: 指定するインタフェースグループ名は、NNMiモニタリングの設定で参照されるインタフェースグループにする必要があります。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで ovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要が

あります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMiでデバイスのホスト名を判別する方法の設定

NNMi 9.0より前のバージョンでは、NNMiはループバックインタフェースで利用可能なすべてのIPアドレスを調べ、検出されたデバイスの有効なホスト名を検索します。NNMi 9.0以降では、NNMiは(デフォルト設定として)管理IPアドレスを使用し、検出されたデバイスのホスト名を判別します。

HostNameMatchManagementIPプロパティをfalseに変更することで、検出されたデバイスの有効なホスト名の検索にNNMi 9.0より前の方法を使用するようにNNMiを設定できます。

ヒント: 通常、このプロパティの値はtrue(デフォルト値)のままにします。HostNameMatchManagementIPプロパティの詳細については、nms-disco.propertiesファイルを参照してください。

HostNameMatchManagementIPプロパティをfalseに変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NNM_PROPS%\nms-disco.properties

   Linuxの場合: $NNM_PROPS/nms-disco.properties
2. 以下のプロパティを含むテキストブロックを探します。

   HostNameMatchManagementIP=true
3. プロパティ値を以下のように変更します。

   HostNameMatchManagementIP=false
4. 作業内容を保存します。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

NNMiはループバックインタフェースで利用可能なすべてのIPアドレスを調べ、検出されたデバイスの有効なホスト名を検索します。

注: 高可用性(HA)でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMiの文字セットエンコードの設定

NNMi管理サーバーに設定したロケールに応じて、NNMiでSNMP OCTETSTRINGデータの解釈に使用するソースエンコードの設定が必要な場合があります。これを行うには、nms-jboss.propertiesファイルを以下のように編集します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS%\nms-jboss.properties
   - Linuxの場合: $NNM_PROPS/nms-jboss.properties
2. 以下の行を含むテキストブロックを探します。

   #!com.hp.nnm.sourceEncoding=UTF-8
3. この行をコメント解除し、以下のように編集します。

   com.hp.nnm.sourceEncoding=UTF-8
4. nms-jboss.propertiesファイルの指示と例に従って、手順3に示すUTF-8プロパティ値を変更します。
5. 変更を保存します。
6. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーで ovstartコマンドを実行します。

   注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMiがNNM iSPIライセンス要求を待機する時間の設定

NNMiコンソールの応答が遅かったり応答しないことがあり、1つ以上のNNM iSPIsがインストールされている場合、NNMiがNNM iSPIライセンス要求からの応答を待機する時間の調整が必要な場合があります。

NNMiがNNM iSPIライセンス要求からの応答を待機するデフォルトの時間は20秒です。

このデフォルト値を変更するには、以下の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: %NNM_PROPS%\nms-jboss.properties

Linuxの場合: `$NNM_PROPS/nms-jboss.properties`

2. 以下を含むテキストブロックを探します。

   `#!com.hp.ov.nms.licensing.EXTENSION_WAIT_TIMEOUT=20`

3. この行をコメント解除し、以下のように変更します。

   `com.hp.ov.nms.licensing.EXTENSION_WAIT_TIMEOUT=<time in seconds>`

   たとえば、応答時間を25秒に変更するには、以下のように入力します。

   `com.hp.ov.nms.licensing.EXTENSION_WAIT_TIMEOUT=25`

   ヒント: このパラメーターを最適な値に調整するには、数回のテストが必要な場合があります。遅いサーバーで実行中の使用率が極めて高いNNM iSPIなど、応答が遅いNNM iSPIsでは、パラメーターを高い値に調整します。

4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

ユーザーインタフェースプロパティの管理

このセクションでは、`ui.properties`ファイルの以下のユーザーインタフェースプロパティの設定方法について説明します。

「SNMP MIB変数名を表示するためのNNMiゲージタイトルの変更」(282ページ)

「MIBブラウザーパラメーターの変更」(283ページ)

「レベル2オペレーターによるノードおよびインシデントの削除の有効化」(284ページ)

「レベル2オペレーターによるノードグループマップの編集の有効化」(285ページ)

「レベル1オペレーターによる [ステータスのポーリング] と [設定のポーリング] の実行の有効化」(286ページ)

SNMP MIB変数名を表示するためのNNMiゲージタイトルの変更

NNMi分析ペインの [**ノードセンサーゲージ**] タブと [**物理センサーゲージ**] タブには、MIB OIDがポーリングされるときにNNMiコンポーネント名を表示するゲージが含まれています。これにより、コンポーネントに属するゲージを判別できます。ノードセンサー名は、NNMiでノードに多数のゲージが表示される場合にゲージを区別するのに便利です。たとえば、ノードに多数のCPUが含まれる場合、NNMiにはCPUごとに異なる名前が表示されます。

この機能を無効にすると、NNMiにはすべてのCPUに同じSNMP MIB変数名が表示されます。

NNMiノードセンサー名ではなくSNMP MIB変数名としてゲージタイトルを表示するようにこのプロパティを変更する場合は、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS\nms-ui.properties`
   - Linuxの場合: `$NNM_PROPS/nms-ui.properties`
2. 以下の行を含むテキストブロックを探します。

   com.hp.nnm.ui.analysisGaugeTitleIsNodeComponentName = true
3. この行を以下のように編集します。

   com.hp.nnm.ui.analysisGaugeTitleIsNodeComponentName = false
4. 変更を保存します。
5. NNMiを再起動します。
   a. NNMi管理サーバーで**ovstart**コマンドを実行します。
   b. NNMi管理サーバーで**ovstop**コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

MIBブラウザーパラメーターの変更

NNMi MIBブラウザー ([**アクション**] > [**MIB情報**] > [**MIBを参照**] メニュー) を使用して、ノードの情報を取得し、SNMPコミュニティ文字列 (省略可能) をそのノードに指定する場合は、NNMi MIBブラウザーは、MIBブラウザーSNMP通信用のnms-ui.propertiesファイルにあるMIBブラウザーパラメーターを使用します。

注: MIBブラウザーを使用するときにコミュニティ文字列を指定しない場合は、NNMiではノードで確立されている [**通信の設定**] 設定 (ある場合) を使用します。これらの設定は、[**設定**] ワークスペースの [**通信の設定**] ビューを使用してNNMiコンソールで設定されます。詳細については、NNMiヘルプの「通信プロトコルを設定する」を参照してください。

nms-ui.propertiesファイルのMIBブラウザーパラメーターを変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS\nms-ui.properties`
   - Linuxの場合: `$NNM_PROPS/nms-ui.properties`
2. 以下の行を含むテキストブロックを探します。

```
# MIB Browser Parameters
```

3. 次のテキストを含む行を検索し、`# MIB Browser Parameters`の下にあるMIBブラウザーパラメーターを探します。

   ```
   mibbrowser
   ```

4. nms-ui.propertiesファイル内の手順に従って、MIBブラウザーパラメーターを変更します。
5. 変更を保存します。
6. NNMiを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

レベル2オペレーターによるノードおよびインシデントの削除の有効化

デフォルトでは、NNMiはNNMi管理者に対してNNMiでのノードまたはインシデントの作成、編集、削除を許可します。NNMiオペレーターレベル2 (L2) ユーザーグループに割り当てられたアカウントに対しても、ノードまたはインシデントの削除を許可するように設定できます。この設定は以下のいずれかの方法で実行することができます。

- (推奨) 必要なノードまたはインシデントを削除するためL2ユーザーの必要な権限を引き上げる。この設定は、NNMi Webコンソールを使用して行うことができます。詳細については、NNMi管理者ヘルプを参照してください。
- L2ユーザーが全体的にノードまたはインシデントを削除できるようにNNMiを設定する。この設定は、一定のNNMiプロパティファイルを変更してデフォルト権限を上書きすることによって行うことができます。

  注意: 上書きによる方法は、全体的に許可する場合だけに使用してください。一度許可すると、NNMi WebコンソールでL2ユーザーアクセス権限を制御できなくなります。

L2ユーザーがノード、ノードに関連するインシデント、またはこの両方を編集または削除できるようにするには、以下の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: `%NNM_PROPS%\nms-topology.properties`

   Linuxの場合: `$NNM_PROPS/nms-topology.properties`

2. 必要に応じて次の行を追加します。

- L2ユーザーがノードを削除できるようにするには、次の行を追加します。

  ```
  permission.override.com.hp.nnm.DELETE_
  OBJECT=com.hp.nnm.ADMIN,com.hp.nnm.LEVEL2
  ```

- L2ユーザーがインシデントを削除できるようにするには、次の行を追加します。

  ```
  permission.override.com.hp.nnm.incident.DELETE=com.hp.nnm.ADMIN,com.hp.nnm.LEVEL2
  ```

3. ファイルを保存します。
4. NNMiを再起動します。
   - NNMi管理サーバーでovstopコマンドを実行します。
   - NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

レベル2オペレーターによるノードグループマップの編集の有効化

デフォルトでは、NNMiはNNMi管理者にノードグループの作成、変更、および削除によるマップの編集を許可します。NNMiオペレーターレベル2ユーザーグループに割り当てられたアカウントに対しても、この編集を許可するように設定できます。

NNMiオペレーターレベル2ユーザーグループに割り当てられたユーザーアカウントに、アクセス権があるノードでのノードグループの作成、変更、および削除を許可するようにNNMiを変更する必要がある場合は、以下の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: %NNM_PROPS%\nms-ui.properties

   Linuxの場合: $NNM_PROPS/nms-ui.properties
2. 以下のテキストブロックを探し、コメント解除します。

   #!com.hp.nnm.ui.level2MapEditing = true
3. 変更を保存します。
4. NNMiを再起動します。
   a. NNMi管理サーバーでovstartコマンドを実行します。
   b. NNMi管理サーバーでovstopコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必

要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

手順1 ~ 4を完了したら、NNMiコンソールは以下のように変更されます。

- **[インベントリ] > [ノードグループ]** メニューに、NNMiオペレーターレベル2の [作成および削除] ツールバーアイコンが表示される。
- **[インベントリ] > [ノードグループ]** メニューに、NNMiオペレーターレベル2の **[アクション] > [削除]** メニュー項目が含まれる。
- [**すべてのノードグループ**] フォルダーが [**トポロジマップ**] ワークスペースに表示される。詳細については、NNMiオンラインヘルプの「ワークスペースについて」を参照してください。
- ノードグループマップの場合、NNMiコンソールに [**レイアウトの保存**] ツールバーボタンおよび **[ファイル] > [レイアウトの保存]** メニュー項目が含まれる。
- [**レイアウトの保存**] アクションメニューの動作は、ノードグループマップ設定がそのノードグループマップに対して設定されているかどうかによって異なる。ノードグループマップに対するノードグループマップ設定が存在しない場合は作成する必要があります。

NNMiオペレーターレベル2ユーザーにノードグループマップ設定の作成権限を付与するようにNNMiを設定することもできます。

1. NNMiコンソールから、**[トポロジマップ] > [ノードグループの概要]** を開きます。
2. 関心のある [ノード グループ] アイコンをダブルクリックします。

   NNMiは、選択したノードグループに関連付けられたノードグループマップを開きます。
3. 以下の手順を実行して、変更するノードグループマップの設定を開きます。

   [ファイル] > [ノードグループマップの設定を開く] を選択します。
4. [**レイアウトの保存のための最小NNMiロール**] を [オペレーターレベル2] に設定します。
5. 変更を保存します。

これでNNMiオペレーターレベル2がノードグループマップビューからノードグループマップ設定を作成、編集、および削除できるようになります。

レベル1オペレーターによる [ステータスのポーリング] と [設定のポーリング] の実行の有効化

NNMiは、NNMiオペレーターレベル2ユーザーグループに割り当てられたユーザーアカウントに対して、アクセス権があるノードでの [ステータスのポーリング] と [設定のポーリング] の実行を許可します。それぞれのnms-topology.propertiesファイルでオブジェクトアクセス権限レベルを変更するだけでなく、NNMiコンソールで [メニュー項目] 設定も変更する必要があります。

NNMiがNNMiオペレーターレベル1ユーザーグループに割り当てられたユーザーアカウントに [ステータスのポーリング] メニュー項目の表示を許可するように [メニュー項目] 設定を変更するには、以下の手順を実行します。

1. **[設定] > [ユーザーインタフェース] > [メニュー項目] > [ステータスのポーリング]** フォームを開きます。
2. [**メニュー項目**] タブから、[**ステータスのポーリング**] メニュー項目ラベルまでスクロールします。
3. [**メニュー項目コンテキスト**] タブから、変更する各 [**必要なNNMiロール**] と [**オブジェクトのタイプ**] 項目のエントリを開きます。
4. レベル1オペレーターにステータスのポーリングを許可する各オブジェクトタイプに対して、[**必要なNNMiロール**] の値を [オペレーターレベル1] に変更します。

   この手順によって、NNMiオペレーターレベル1ユーザーグループに割り当てられたユーザーアカウントで、指定されたオブジェクトタイプに対するステータスのポーリングアクションを表示できます。

NNMiオペレーターレベル1ユーザーグループに割り当てられたユーザーアカウントに [設定のポーリング] メニュー項目の表示を許可するようにNNMiを変更するには、以下の手順を実行します。

1. **[設定] > [ユーザーインタフェース] > [メニュー項目] > [設定のポーリング]** フォームを開きます。
2. [**メニュー項目コンテキスト**] タブから、変更する各 [**必要なNNMiロール**] と [**オブジェクトのタイプ**] 項目のエントリを開きます。
3. レベル1オペレーターに設定のポーリングを許可する各オブジェクトタイプに対して、[**必要なNNMiロール**] の値を [オペレーターレベル1] に変更します。

   この手順によって、NNMiオペレーターレベル1ユーザーグループに割り当てられたユーザーアカウントで、指定されたオブジェクトタイプに対する設定のポーリングアクションを表示できます。

注: NNMiオペレーターレベル1ユーザーグループに割り当てられたユーザーアカウントがNNMiコンソールからステータスのポーリングと設定のポーリングの両方のコマンドを実行できるようにするには、`nms-topology.properties`ファイルを編集する必要があります。これらの手順を実行しない場合、NNMiの [アクション] メニューに [ステータスのポーリング] と [設定のポーリング] オプションは表示されますが、ユーザーがステータスのポーリングコマンドまたは設定のポーリングコマンドを実行しようとするとエラーメッセージが表示されます。

ステータスのポーリングと設定のポーリングに必要なアクセスレベル (必要なオブジェクトアクセス権限レベル) を変更するには、以下の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: `%NNM_PROPS%\nms-topology.properties`

   Linuxの場合: `$NNM_PROPS/nms-topology.properties`
2. ファイルの最下部までスクロールし、ステータスのポーリングを変更するために以下の行を追加します。

   ```
   permission.override.com.hp.nnm.STATUS_
   POLL=com.hp.nnm.ADMIN,com.hp.nnm.LEVEL2,com.hp.nnm.LEVEL1
   ```
3. 設定のポーリングを変更するために以下の行を追加します。

permission.override.com.hp.nnm.CONFIG_
POLL=com.hp.nnm.ADMIN,com.hp.nnm.LEVEL2,com.hp.nnm.LEVEL1

4. 変更を保存します。
5. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

同時SNMP要求の変更

NNMiでは、1つのノードに対して同時SNMP要求が3個に制限されています。これにより、ノードのSNMPエージェントが応答をドロップするリスクが減ります。

この値をより高く調整し、検出速度を高めることができます。ただしこの値を高く設定しすぎると、応答がドロップされるリスクが増して、検出精度が落ちます。

この制限を変更する必要がある場合は、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: %NNM_PROPS%\nms-communication.properties
   - UNIXの場合: $NNM_PROPS/nms-communication.properties
2. 1つのノードに対する同時SNMP要求の現在の数を増やすには、以下の手順を実行します。
   a. 以下のような行を探します。

      #!com.hp.ov.nms.comm.snmp.maxConcurrentRequests=3

   b. プロパティをコメント解除します。

      com.hp.ov.nms.comm.snmp.maxConcurrentRequests=3

      注: プロパティをコメント解除するには、行の先頭から#!文字を削除します。

   c. 1つのノードに対する同時SNMP要求の目的の数に、既存の値を変更します。
   d. 変更を保存します。
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

組み込みデータベースポートの変更

組み込みデータベースに異なるポートを使用するようにNNMiを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_CONF%\nnm\props\nms-local.properties`
   - Linuxの場合: `$NNM_CONF/nnm/props/nms-local.properties`
2. 以下のような行を探します。

   `#!com.hp.ov.nms.postgres.port=5432`
3. このプロパティのコメントを解除します。

   `com.hp.ov.nms.postgres.port=5432`

   > **ヒント:** プロパティをコメント解除するには、行の先頭から`#!`文字を削除します。
4. 既存の値を新しいポート番号に変更します。
5. 変更を保存します。
6. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

NNMi正規化プロパティの変更

NNMiでは、ホスト名とノード名の両方が大文字と小文字を区別して保存されます。NNMiコンソールのすべての検索、ソート、およびフィルターの結果も大文字と小文字を区別して返されます。使用するDNSサーバーが、すべて大文字、すべて小文字、大文字と小文字の混合などのように大文字と小文字を区別してさまざまなノード名とホスト名を返す場合、最良の結果が得られない場合があります。

ユーザーの特定のニーズに合うように、NNMiの正規化プロパティを変更できます。NNMiの初期検出シードを行う前に、これらの変更を行うことをお勧めします。HPは、デプロイ中の初期検出を実行する前に、本項の設定を調整することをお勧めします。

初期検出を実行してから正規化プロパティの変更を行う場合は、完全な検出を開始する**nnmnoderediscover.ovpl -all**スクリプトを実行できます。詳細については、nnmnoderediscover.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

以下のプロパティを変更できます。

- 検出されるノード名をUPPERCASE、LOWERCASE、またはOFFに正規化します。
- 検出されるホスト名をUPPERCASE、LOWERCASE、またはOFFに正規化します。

正規化プロパティを変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\nms-topology.properties`
   - Linuxの場合: `$NNM_PROPS/nms-topology.properties`
2. 検出される名称を正規化するようにNNMiを設定するには、以下のような行を探します。

   #!com.hp.ov.nms.topo.NAME_NORMALIZATION=OFF

   a. プロパティをコメント解除します。

      com.hp.ov.nms.topo.NAME_NORMALIZATION=OFF

      注: プロパティをコメント解除するには、行の先頭から#!文字を削除します。

   b. `OFF`を`LOWERCASE`または`UPPERCASE`に変更します。
   c. 変更を保存します。
3. 検出されるホスト名を正規化するようにNNMiを設定するには、以下のような行を探します。

   #!com.hp.ov.nms.topo.NAME_NORMALIZATION=OFF

   a. プロパティをコメント解除します。

      com.hp.ov.nms.topo.HOSTNAME_NORMALIZATION=OFF

   b. `OFF`を`LOWERCASE`または`UPPERCASE`に変更します。
   c. 変更を保存します。
4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

初期検出後の正規化プロパティの変更

初期検出を実行した後に正規化プロパティを変更すると、NNMiは、次回検出までプロパティ変更との食い違いが続きます。これを解消するには、NNMi正規化プロパティを変更した後に、`nnmnoderediscover.ovpl -all`スクリプトを実行して完全検出を開始します。

同時SNMP要求の変更

NNMiでは、1つのノードに対して同時SNMP要求が3個に制限されています。これにより、ノードのSNMPエージェントが応答をドロップするリスクが減ります。

この値をより高く調整し、検出速度を高めることができます。ただしこの値を高く設定しすぎると、応答がドロップされるリスクが増して、検出精度が落ちます。

同時SNMP要求の制限を変更する場合は、以下の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: `%NNM_PROPS%\nms-communication.properties`

   Linuxの場合: `$NNM_PROPS/nms-communication.properties`
2. 以下のような行を探します。

   #!com.hp.ov.nms.comm.snmp.maxConcurrentRequests=3
3. このプロパティのコメントを解除します。

   com.hp.ov.nms.comm.snmp.maxConcurrentRequests=3

   ヒント: プロパティをコメント解除するには、行の先頭から#!文字を削除します。
4. 1つのノードに対する同時SNMP要求の目的の数に、既存の値を変更します。
5. 変更を保存します。
6. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで `ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMi自己監視

NNMiでは、メモリー、CPU、ディスクリソースなどの自己監視チェックが実行されます。NNMi管理サーバーのリソースが少なくなる、または重大な状態が検出されると、NNMiによってインシデントが生成されます。

NNMiの稼働状態情報を表示するには、以下のいずれかの方法を使用します。

- NNMiコンソールで、[**ヘルプ**] > [**システム情報**] を選択してから、[**ヘルス**] タブをクリックします。
- 自己監視の詳細レポートについては、[**ヘルプ**] > **[NNMiシステム情報]** > **[ヘルス]** を選択してから、[**詳細ヘルスレポートの表示 (サポート)**] をクリックします。
- `nnmhealth.ovpl`スクリプトを実行します。

NNMiが自己監視稼働状態の例外を検出すると、NNMiによりNNMiコンソールの下部とフォームの上部にステータスメッセージが表示されます。

この警告メッセージを無効にするには、以下の手順を実行します。

1. 以下のファイルを開きます。
   - Windowsの場合: `%NNM_PROPS\nms-ui.properties`
   - Linuxの場合: `$NNM_PROPS/nms-ui.properties`
2. 以下の行を含むテキストブロックを探します。

   #!com.hp.nms.ui.health.disablewarning=false
3. この行をコメント解除し、以下のように編集します。

   com.hp.nms.ui.health.disablewarning==true
4. 変更を保存します。
5. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

特定ノードの検出プロトコルの使用を抑える

NNMiでは複数のプロトコルを使用し、ネットワークデバイス間のレイヤー2接続を検出します。定義されている検出プロトコルは多数あります。たとえばLink Layer Discovery Protocol (LLDP) は業界標準プロトコルですが、Ciscoデバイス用のCisco Discovery Protocol (CDP) のように、ベンダー固有のプロトコルも多数あります。

指定したデバイスの検出プロトコル収集を抑制するようにNNMiを設定できます。検出プロトコル収集を抑制することで解決できる、特別な状況があります。

以下に例を挙げます。

- Enterasysデバイス:SNMPを使用してEnterasys Discovery Protocol (EnDP) およびLLDPのテーブルから一部のEnterasysデバイスに関する情報を収集すると、NNMiでメモリが不足するという問題が発生することがあります。このようなデバイスでEnDPおよびLLDPの処理をスキップするようにNNMiを設定すると、これを防止できます。これを実行するには、「検出プロトコル収集の使用の抑制」(293ページ)に示すように、デバイスの管理アドレスを`disco.SkipXdpProcessing`ファイルに追加します。

  注: 一部のEnterasysデバイスの新バージョンのオペレーティングシステムでは、**`set snmp timefilter break`**コマンドがサポートされます。このようなEnterasysデバイスでは、**`set snmp timefilter break`**コマンドを実行します。このコマンドを使用してデバイスを設定した場合、このデバイスを`disco.SkipXdpProcessing`ファイルにリストする必要はありませ

ん。

- Nortelデバイス:多くのNortelデバイスではSynOptics Network Management Protocol (SONMP) を使用し、レイヤー2レイアウトおよび接続を検出します。一部のデバイスでは複数のインタフェースで同一MACアドレスを使用するため、このプロトコルで適切に動作しません。相互接続した2つのNortelデバイスがインタフェースの誤ったセット間でレイヤー2接続を示し、接続が接続ソースSONMPを示す場合、この問題が発生することがあります。
  この例では、誤った接続に関与しているデバイスのレイヤー2接続の検出に対して、SONMPプロトコルを使用しないようにNNMiを設定することを推奨します。これを実行するには、「検出プロトコル収集の使用の抑制」(293ページ)で示すように、2つのデバイスの管理アドレスを`disco.SkipXdpProcessing`ファイルに追加します。

検出プロトコル収集の使用の抑制

この収集を抑制する必要がある場合は、以下の手順を実行します。

1. 以下のファイルを作成します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\disco\disco.SkipXdpProcessing`
   - Linuxの場合: `$NnmDataDir/shared/nnm/conf/disco/disco.SkipXdpProcessing`

   `disco.SkipXdpProcessing`ファイルでは、大文字と小文字が区別されます。
2. プロトコル収集を抑制するすべてのデバイスで、デバイスのIPアドレスを`disco.SkipXdpProcessing`ファイルに追加します。disco.SkipXdpProcessingのリファレンスページ、またはLinuxのマンページの指示に従ってください。
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

注: 1つまたは複数のノードの検出プロトコル処理を抑制すると、管理対象ネットワークのレイヤー2レイアウトの精度が多少落ちることがあります。HPはこの精度低下の責任を負いません。

注: `ovjboss`サービスは起動時に`disco.SkipXdpProcessing`ファイルを読み取ります。NNMiの起動後に変更を行った場合は、この手順で示すようにNNMiを再起動してください。

注: Enterasysデバイスで**`setsnmp timefilter break`**コマンドを実行した場合は、デバイスのアドレスを`disco.SkipXdpProcessing`ファイルから削除し、この手順で示すようにNNMiを再起動します。NNMiは、検出プロトコルを使用したとき、より正確なレイヤー2マップを表示します。

詳細については、disco.SkipXdpProcessingのリファレンスページ、またはLinuxのマンページを参照してください。

管理上停止中のインタフェースのIPアドレスに対するモニタリングの抑制

通常、NNMiユーザーは複数のインタフェース (1つは管理上で動作状態にある、アドレスがICMP要求に応答するインタフェース、もう1つは管理上で停止状態にある、ICMP要求に応答しないインタフェース) を同じIPアドレスで設定します。このような場合、管理上で停止状態にあるインタフェースとそのIPアドレスがノードステータスに影響しないようにする必要があります。

デフォルトでは、NNMiは管理上で停止状態にあるインタフェースのIPアドレスに対するモニタリングを抑制し、ノードステータスが変更されることを防ぎます。

管理上で停止状態にあるインタフェースのIPアドレスに対するモニタリングを実行するかどうかを設定するには、以下の手順を実行します。

1. 以下の場所にあるnms-disco.propertiesファイルを開きます。

   Windowsの場合: %NnmDataDir%\shared\nnm\conf\props\nms-disco.properties

   Linuxの場合: $NnmDataDir/shared/nnm/conf/props/nms-disco.properties
2. ファイルから、以下のようなセクションを特定します。

   #!com.hp.ov.nms.disco.suppressMonitoringOfAddressOnAdminDownInterface=true
3. プロパティを以下のように設定できます。

   管理上で停止状態にあるインタフェースのIPアドレスに対するモニタリングを抑制するには、この行をコメント解除し、プロパティをtrue (デフォルト設定) に設定します。この行は以下のようになります。

   com.hp.ov.nms.disco.suppressMonitoringOfAddressOnAdminDownInterface=true

   管理上で停止状態にあるインタフェースのIPアドレスをNNMiでモニタリングするには、この行をコメント解除し、プロパティの値を以下のように編集します。

   com.hp.ov.nms.disco.suppressMonitoringOfAddressOnAdminDownInterface=false
4. nms-disco.propertiesファイルへの変更を保存します。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

大規模スイッチのVLANインデックス付けの使用を抑制する

NNMiが管理対象ネットワークのスイッチデバイス間でレイヤー2接続を認識する方法の1つは、dot1dTpFdbTable (FDB) をスイッチから取得することです。ただしCiscoスイッチの場合、NNMiは

VLAN-indexing方法を使用してFDB全体を取得する必要があります。各デバイスで設定されているVLANの数が多い場合、VLAN-indexingによるFDBの取得の完了には数時間かかることがあります。

Ciscoスイッチは、多くの場合、Cisco Discovery Protocol (CDP) を使用するように設定されています。CDPは、レイヤー2接続を認識するための優れた方法であるとみなされています。ネットワークのコアに配置されている大規模スイッチには、多くのVLANが含まれていることがあります。このスイッチには一般的に、直接接続されているエンドノードがありません。管理するスイッチに直接接続されているエンドノードがない場合は、この大規模スイッチでFDBの収集を抑制するとよいでしょう。NNMiは、CDPから収集したデータを使用してレイヤー2検出を完了します。この大規模スイッチは、VLAN-indexingの抑制の主な候補となります。多くのエンドノードが接続している、ネットワークのエッジにある小規模スイッチ (アクセススイッチと呼ばれる) では、VLAN-indexingを抑制しないでください。

VLAN-indexingを抑制するようにNNMiを設定できます。これを実行するには、「VLANインデックス付けの使用を抑制する」(295ページ) で示すように、NNMi管理者が大規模スイッチの管理アドレスまたはアドレス範囲を作成してdisco.NoVLANIndexingファイルに追加する必要があります。ovjbossサービスは起動時にdisco.NoVLANIndexingファイルを読み取ります。ovjbossサービスの起動後、NNMi管理者がdisco.NoVLANIndexingファイルを変更した場合、その変更内容は、ovjbossサービスを次回起動するまで有効になりません。デフォルトでは、disco.NoVLANIndexingファイルは存在しません。disco.NoVLANIndexingが存在しない場合、この機能は無効になり、NNMiはVLAN-indexingを使用して、すべてのデバイスでFDBテーブル全体を収集しようとします。

VLANインデックス付けの使用を抑制する

このvlan-indexingを無効にする必要がある場合は、以下の手順を実行します。

注: 1つまたは複数のノードのvlan-indexingを抑制すると、管理対象ネットワークのレイヤー2レイアウトの精度が多少落ちることがあります。HPはこの精度低下の責任を負いません。

1. 以下のファイルを作成します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf\disco\disco.NoVLANIndexing
   - Linuxの場合: $NnmDataDir/shared/nnm/conf/disco/disco.NoVLANIndexing

   disco.NoVLANIndexingファイルでは、大文字と小文字が区別されます。
2. vlan-indexingを無効にするすべてのデバイスのIPアドレスまたはアドレス範囲をdisco.NoVLANIndexingファイルに追加します。disco.NoVLANIndexingリファレンスページ、またはUNIXのマンページの指示に従ってください。
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

   注: ovjbossサービスは起動時にdisco.NoVLANIndexingファイルを読み取ります。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

詳細については、disco.Disco.NoVLANIndexingのリファレンスページ、またはLinuxのマンページを参照してください。

計画停止

NNMiでは、 nnmscheduledoutage.ovpl コマンドを使用して任意のノードセットの停止をスケジュールできます。たとえば、ルーターセットの毎週のメンテナンスで停止をスケジュールしたり、特定のノードの電源を交換したりできます。

詳細については、nnmscheduledoutage.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

ヒント: NNMiを使用した停止のスケジュールの詳細については、NNMiヘルプを参照してください。

センサーステータスの設定

NNMiには、ステータス判別用に監視できる以下の物理センサーとノードセンサーが含まれています。

物理センサーとノードセンサー

物理センサー	デフォルトで物理コンポーネントにステータスを伝達	ノードセンサー	デフォルトでノードにステータスを伝達
FAN	はい	CPU	いいえ
POWER_SUPPLY	はい	MEMORY	はい
TEMPERATURE	いいえ	BUFFERS	いいえ
VOLTAGE	いいえ	DISK_SPACE	いいえ
BACK_PLANE	はい		

注: デフォルトでは、FAN、POWER_SUPPLY、BACK_PLANE、およびMEMORYがステータスを物理コンポーネントレベルに伝達します。たとえば、ファンが赤色のステータスインジケーターを示している場合、対応する物理コンポーネント (シャーシ) は黄色のステータスインジケーターを受

け取ります。この場合、シャーシのステータスを表示しているユーザーには、そのシャーシのコンポーネントに何らかの障害があることが警告されます。

物理センサーステータスの設定

以下のセクションの手順を実行して、物理センサーでステータスを物理コンポーネント (シャーシなど) レベルに伝達するかどうかを設定できます。

物理コンポーネントへの物理センサーステータスの伝達

1. 以下のディレクトリにnnm-apa.propertiesという名前の新しいプロパティファイルを作成します (このファイルが存在しない場合)。

   Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props`

   Linuxの場合: `$NnmDataDir/shared/nnm/conf/props`
2. テキストエディターを使用して、プロパティファイル内に以下のテキストを挿入します。

   com.hp.ov.nms.apa.PhysicalSensorPropagateToPhysicalComponentStatus_<Type>=true

   <Type>は物理センサーです。詳細については、「センサーステータスの設定」(296ページ)を参照してください。
3. プロパティファイルを保存します。
4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーで ovstart コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

物理コンポーネントに伝達しない物理センサーステータスの設定

1. 以下のディレクトリにnnm-apa.propertiesという名前の新しいプロパティファイルを作成します (このファイルが存在しない場合)。

   Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props`

   Linuxの場合: `$NnmDataDir/shared/nnm/conf/props`
2. テキストエディターを使用して、プロパティファイル内に以下のテキストを挿入します。

   com.hp.ov.nms.apa.PhysicalSensorNoPropagateToPhysicalComponentStatus_<Type>=true

   <Type>は物理センサーです。詳細については、「センサーステータスの設定」(296ページ)を参照してください。

3. プロパティファイルを保存します。
4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

物理センサーステータス値の上書き

デフォルトでは、3つのセンサーのステータス値 ([なし]、[注意域]、および [利用不可])は、Causal Engineによって [正常域] ステータスにマッピングされます。デフォルトのステータスマッピングは、[なし]、[注意域]、[利用不可] を [危険域] にマッピングするように上書きできます。

物理センサーのステータス値を上書きするには、以下の手順を実行します。

1. 以下のディレクトリにnnm-apa.propertiesという名前の新しいプロパティファイルを作成します (このファイルが存在しない場合)。

   Windowsの場合: %NnmDataDir%\shared\nnm\conf\props

   Linuxの場合: $NnmDataDir/shared/nnm/conf/props
2. テキストエディターを使用して、プロパティファイル内に必要に応じて以下の行の1つ、2つ、または3つすべてを挿入します。

   com.hp.ov.nms.apa.PhysicalSensorValueReMappedToDown_None=true

   com.hp.ov.nms.apa.PhysicalSensorValueReMappedToDown_Warning=true

   com.hp.ov.nms.apa.PhysicalSensorValueReMappedToDown_Unavailable= true
3. プロパティファイルを保存します。
4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

注: [利用不可] の状態を [未ポーリング] 状態にマッピングできます ([利用不可] は測定機能が利用できないことを指すため)。この状態は、多くの場合コンポーネントの機能不全ではなくセンサーの機能不全で発生します。[利用不可] を [未ポーリング] にマッピングするには、手順2で以下のテキストを使用する以外は上記と同じ手順を実行します。

com.hp.ov.nms.apa.PhysicalSensorValueReMappedToUnpolled_Unavailable= true

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要が

あります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

ノードセンサーステータスの設定

以下のセクションの手順を実行して、ノードセンサーでステータスをノードレベルに伝達するかどうかを設定できます。

ノードへのノードセンサーステータスの伝達

1. 以下のディレクトリに`nnm-apa.properties`という名前の新しいプロパティファイルを作成します (このファイルが存在しない場合)。

   Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props`

   Linuxの場合: `$NnmDataDir/shared/nnm/conf/props`
2. テキストエディターを使用して、プロパティファイル内に以下のテキストを挿入します。

   com.hp.ov.nms.apa.NodeSensorPropagateToNodeStatus_<Type>=true

   <Type>はノードセンサーです。詳細については、「センサーステータスの設定」(296ページ)を参照してください。
3. プロパティファイルを保存します。
4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

ステータスをノードに伝達しないようにするためのノードセンサーの設定

1. 以下のディレクトリに`nnm-apa.properties`という名前の新しいプロパティファイルを作成します (このファイルが存在しない場合)。

   Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props`

   Linuxの場合: `$NnmDataDir/shared/nnm/conf/props`
2. テキストエディターを使用して、プロパティファイル内に以下のテキストを挿入します。

   com.hp.ov.nms.apa.NodeSensorNoPropagateToNodeStatus_<Type>=true

   <Type>はノードセンサーです。詳細については、「センサーステータスの設定」(296ページ)を参照してください。
3. プロパティファイルを保存します。

4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

ノードコンポーネントのステータス値の上書き

デフォルトでは、3つのノードコンポーネントのステータス値 ([なし]、[注意域]、および [利用不可]) は、Causal Engineによって正常域ステータスにマッピングされます。デフォルトのステータスマッピングは、[なし]、[注意域]、[利用不可] を [危険域] にマッピングするように上書きできます。

ノードコンポーネントのステータス値を上書きするには、以下の手順を実行します。

1. 以下のディレクトリにnnm-apa.propertiesという名前の新しいプロパティファイルを作成します (このファイルが存在しない場合)。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf\props
   - Linuxの場合: $NnmDataDir/shared/nnm/conf/props
2. テキストエディターを使用して、プロパティファイル内に必要に応じて以下の行の1つ、2つ、または3つすべてを挿入します。

   com.hp.ov.nms.apa.NodeComponentValueReMappedToDown_None: true

   com.hp.ov.nms.apa.NodeComponentValueReMappedToDown_Warning: true

   com.hp.ov.nms.apa.NodeComponentValueReMappedToDown_Unavailable: true

   注: [利用不可] の状態を [未ポーリング] 状態にマッピングできます ([利用不可] は測定機能が利用できないことを指すため)。この状態は、多くの場合コンポーネントの機能不全ではなくセンサーの機能不全で発生します。[利用不可] を [未ポーリング] にマッピングするには、以下のテキストを使用します。

   com.hp.ov.nms.apa.NodeComponentValueReMappedToUnpolled_Unavailable: true
3. プロパティファイルを保存します。
4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

   注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモー

ドにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

インタフェースの入力速度と出力速度のインポート

NNMiでは、nnmsetiospeed.ovplコマンドを使用してインタフェースの入力速度と出力速度をインポートできます。このコマンドでは、指定されたノードのインタフェースセットまたはすべてのインタフェースの入力速度と出力速度を指定できます。また、カンマ区切り値 (CSV) ファイルを使用してインポート条件も指定できます。インポートされた値は、NNMiコンソールの [インタフェース] フォームに表示されます。

詳細については、nnmsetiospeed.ovpl のリファレンスページ、またはLinuxのマニュアルページを参照してください。

NNMiロギング

このセクションでは、NNMiログファイル形式およびログファイルのプロパティの変更方法について説明します。

- 「NNMiログファイル」(301ページ)
- 「ロギングファイルのプロパティの変更」(302ページ)

NNMiログファイル

HP Network Node Manager i Software (NNMi) のパフォーマンスを調べる、またはNNMiのプロセスとサービスがどのように動作しているかを観察するには、プロセスとサービスアクティビティの履歴を表示するログファイルを確認できます。これらのファイルは、以下の場所で入手できます。

- Windowsの場合: %NnmDataDir%\log\nnm\
- Linuxの場合: $NnmDataDir/log/nnm

NNMiでは、name.logという形式のファイル名でログファイルが保存されます。アーカイブされたログファイルには、name.log.%gという形式で番号が追加されます。

- nameは、ログファイルのベース名です。
- %gは、アーカイブされたログファイルのアーカイブ番号です。最も高いアーカイブ番号は最も古いファイルを示します。

ログファイルは、そのサイズが設定した制限を超えたときにアーカイブされる可能性があります。ログファイルのサイズが設定した制限を超えると、最後のアクティブなログファイルがアーカイブされます。たとえば、nnm.logファイルをnnm.log.1ファイルとしてアーカイブした後に、NNMiは新しいnnm.logファイルへの記録を開始します。

NNMiでは、以下のロギングレベルでメッセージが記録されます。

- SEVERE: NNMiの異常な動作に関するイベント。
- WARNING:潜在的な問題を示すイベント、およびSEVEREロギングレベルに含まれるすべてのメッセージ。
- INFO: NNMiコンソール (またはそれと同等のもの) 書き込まれるメッセージ、およびWARNINGロギングレベルに含まれるすべてのメッセージ。

ロギングファイルのプロパティの変更

NNMiには、NNMiロギングを変更できるいくつかの機能があります。このセクションでは、これらの機能の調整方法について説明します。

監査ログファイルの変更の詳細については、「NNMi監査」(132ページ)も参照してください。

ロギングのサインインおよびサインアウト

NNMi 10.01は、各ユーザーによるNNMiコンソールへのサインインまたはサインアウトのログを生成するように設定されていません。サインインおよびサインアウトアクティビティを記録するようにNNMiを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\conf\props\nnm-logging.properties`
   - Linuxの場合: `$NnmDataDir/shared/nnm/conf/props/nnm-logging.properties`
2. 以下の行を含むテキストブロックを探します。

   ```
   com.hp.ov.nnm.log.signin.level = OFF
   ```
3. この行を以下のように変更します。

   ```
   com.hp.ov.nnm.log.signin.level = INFO
   ```
4. 変更を保存します。
5. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**を実行します。
   b. NNMi管理サーバーで**ovstart**を実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

管理サーバーの変更

他のシステムでHP Network Node Manager i Software設定を複製できます。たとえば、テスト環境から生産環境に移動したり、NNMi管理サーバーのハードウェアを変更したりできます。

NNMi設定に影響を及ぼさずに、NNMi管理サーバーのIPアドレスを変更できます。

この章には、以下のトピックがあります。

NNMi設定移動の準備のベストプラクティス

以下のベストプラクティスは、NNMi設定の異なるシステムへの移動に適用されます。

- ノードグループ設定が管理対象ノードの識別にホスト名を使っている場合、製品およびテストのNNMi管理サーバーは同じDNSサーバーを使う必要があります。製造システムとテストシステムが異なるDNSサーバーを使っている場合、管理対象ノードの解決済みの名前を変更すると、2つのNNMi管理サーバーの間でポーリング設定が異なる結果になることがあります。
- 設定エクスポートを1人の作成者に制限できます。自分のグループまたは会社に一意の新しい作成者値を作成します。以下のアイテムを作成または変更するときに、この作成者の値を指定します。
  - デバイスプロファイル
  - インシデントの設定
  - URLアクション
- Smart Plug-ins (iSPI) をインストールする場合は、NNM iSPIの適切なドキュメントを参照してください。すべてのNNM iSPIのドキュメントは、http://support.openview.hp.com/selfsolve/manualsにあるHP Software製品マニュアルのWebサイトで利用できます。

NNMi設定および組み込みデータベースの移動

NNMiの設定と組み込みデータベースを、たとえばテストシステムから本稼働システムなどへ移動するには、ソース (テスト) システム上のすべてのNNMiデータをバックアップしてから、バックアップをターゲット (本稼働) システムに復元します。

バックアップの実行後NNMiデータベースに対する変更が何も行われないようにするため、すべてのNNMiプロセスを停止し、オフラインバックアップを作成してください。例:

```
nnmbackup.ovpl -type offline -scope all -target nnmi_backups\offline
```

「異なるシステムでの復元」(249ページ)にリストされた要件が新規システム上で満たされることを確認してから、以下の例のようなコマンドを実行します。

```
nnmrestore.ovpl -source nnmi_backups\offline\newest_backup
```

注意: NNMiは同じSSL証明書を使用して、データベース (組み込みまたは外部) へのアクセス、およびNNMiコンソールへのHTTPSアクセスをサポートします。データベースへアクセスするための証明書は、ソースシステム上でNNMiプロセスを最初に開始したときに作成されました。この証明書はバックアップおよび復元データに含まれています。この証明書がないと、NNMiはターゲットシステムからデータベースにアクセスできません。

ただし、NNMiコンソールへのHTTPSアクセスの場合は、SSL証明書をターゲットシステムに生成する必要があります。jbossの現在の実装は証明書のマージをサポートしていません。そのため、別のシステムからのデータを復元して設定したシステム上のNNMiコンソールに対するHTTPSアクセスを、NNMiはサポートしません。ターゲットシステムがNNMiコンソールへのHTTPSアクセスをサポートする必要がある場合は、「NNMi設定の移動」(304ページ)の手順を実行してから、ターゲットシステム上で新たにデータ収集を開始します。

NNMi設定の移動

`nnmconfigexport.ovpl`コマンドを使用して、NNMi設定をXMLファイルに出力します。次に、`nnmconfigimport.ovpl`コマンドを使用して、XMLファイルから新しいシステムのNNMiにこの設定をインポートします。

注意: nnmconfigimport.ovplスクリプトを使用してファイルをインポートする前に、nnmconfigexport.ovplスクリプトでエクスポートしたファイルを編集しないでください。

これらのコマンドの詳細については、該当するリファレンスページ、またはLinuxのマニュアルページを参照してください。

ヒント: `nnmconfigexport.ovpl`コマンドではSNMPv3資格情報は保持されません。詳細については、`nnmconfigexport.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

注: NNMi設定のみを移動できます。HPは、あるNNMi管理サーバーから異なるNNMi管理サーバー

へのトポロジまたはインシデントデータの移動をサポートしません。また、NNM iSPI Performance for Metrics用に収集されたパフォーマンスデータのようなiSPIデータの移動もサポートしません。

NNMi公開キー証明書の復元

注意: NNMi管理サーバーがNNMiアプリケーションフェイルオーバーに関与、または高可用性 (HA) クラスターのメンバーの場合は、サポート担当者に問い合わせてください。

nnm.keystoreファイルにはNNMiが暗号化に使用する公開キー証明書が格納されます。NNMiのインストールプロセスでnnm.keystoreファイルが作成され、このファイルの証明書がNNMiデータベースのnms_sec_keyレコード (PostgresまたはOracle) にリンクされます。

NNMiが後でアンインストールされ、再インストールする前にNNMiのOracleユーザーおよびデータベーステーブルが削除 (Oracleユーザーのカスケード削除) されていない場合、nms_sec_keyエントリは新規に作成されるnnm.keystoreファイルに対して有効ではありません。

NNMi公開キー証明書を復元するには、以下のタスクを実行します。

「タスク1:KeyManagerサービスのステータスの確認」(305ページ)

「タスク2:現在のnnm.keystoreファイルをバックアップする」(305ページ)

「タスク3:元のnnm.keystoreファイルを検索する」(306ページ)

「タスク4:可能な場合、元のnnm.keystoreファイルをリストアーする」(307ページ)

タスク1:KeyManagerサービスのステータスの確認

1. 以下のコマンドを実行します。

   ovstatus -v ovjboss

2. コマンド出力で、KeyManagerサービスが実行中でないことを確認します。これは、通常、nnm.keystoreファイルが破損している、または存在しないことを示します。

   注: ovstatus出力でKeyManagerサービスが開始されていることが示される場合は、サポート担当者に問い合わせてください。

タスク2:現在のnnm.keystoreファイルをバックアップする

1. NNMiトラストストアーが格納されているディレクトリに変更します。

   Windowsの場合: %NnmDataDir%\shared\nnm\certificates

   Linuxの場合: $NnmDataDir/shared/nnm/certificates

2. バックアップ用に、以下のファイルのコピーを保存します。

nnm.keystore

nnm.truststore

タスク3:元のnnm.keystoreファイルを検索する

1. NNMiデータベースのセキュリティキーのフィンガープリントを特定します。
   - 組み込みPostgresデータベースの場合は、以下を入力します。
     - Windowsの場合:

       %NnmInstallDir%\nonOV\Postgres\bin\psql -U postgres

       -d nnm -c "<database_command>"

     - Linuxの場合:

       $NnmInstallDir/nonOV/Postgres/bin/psql -U postgres

       -d nnm -c "<database_command>"

     <database_command>を以下のSQLコマンド文字列に置き換えます。

     select fingerprint from nms_sec_key;

   - Oracleデータベースの場合は、Oracleデータベース管理者に<database_command>を (組み込みデータベース用にこの手順で先に説明したとおりに) 適切なOracle管理ツールで実行するよう依頼します。

   コマンド結果は単一データベース行にする必要があります。正しいnnm.keystoreファイルには、このフィンガープリントも含まれます。
2. テストするバックアップnnm.keystoreファイルを確認します。

   このファイルは、元のインストールディレクトリのNNMi管理サーバーのバックアップ内に置くことができます。
3. バックアップnnm.keystoreファイルのフィンガープリントをテストします。
   a. NNMi証明書が含まれるディレクトリを変更します。

      Windowsの場合: %NnmDataDir%\shared\nnm\certificates

      Linuxの場合: $NnmDataDir/shared/nnm/certificates

   b. キーストアの内容を確認します。
      - Windowsの場合:

        %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool -list

        -keystore nnm.keystore

      - Linuxの場合:

        $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -list

        -keystore nnm.keystore

キーストアーのパスワードの入力を求められたら、「nnmkeypass」と入 力します。

キーストアの出力形式は以下のとおりです。

Keystore type: jks

Keystore provider:SUN

Your keystore contains 1 entry

selfsigned, Oct 28, 2008, keyEntry,

Certificate fingerprint (MD5):29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02

c. このnnm.keystoreファイルのMD5フィンガープリントの値とNNMiデータベース(このタスクの手順1) のフィンガープリントを比較します。
   - フィンガープリントが完全一致する場合は、このNNMiデータベースの正しいnnm.keystoreファイルを検索したことになります。「タスク4:可能な場合、元のnnm.keystoreファイルをリストアーする」(307ページ)に進みます。
   - フィンガープリントが完全一致しない場合は、異なるnnm.keystoreファイルでこのタスクを実行します。

注: 上記の手順で元のnnm.keystoreファイルを検索できない場合は、サポート担当者に問い合わせてください。「タスク4:可能な場合、元のnnm.keystoreファイルをリストアーする」(307ページ)には進まないでください。

タスク4:可能な場合、元のnnm.keystoreファイルをリストアーする

正しいnnm.keystoreファイルを検索できた場合は、以下の手順を実行してそのファイルを復元します。

1. NNMi管理サーバーを停止します。

   NNMi管理サーバーでovstopコマンドを実行します。
2. 既存のファイルの上部にあるnnm.keystoreファイルを以下の場所にコピーします。

   Windowsの場合: %NnmDataDir%\shared\nnm\certificates

   Linuxの場合: $NnmDataDir/shared/nnm/certificates
3. NNMi管理サーバーを起動します。

   NNMi管理サーバーでovstartコマンドを実行します。
4. 以下のコマンドを実行します。

   ovstatus -v ovjboss
5. コマンド出力で、KeyManagerサービスが起動されていることを確認します。

NNMiが正しく動作していることを確認したら、「タスク2:現在のnnm.keystoreファイルをバックアップする」(305ページ) で作成したnnm.keystoreファイルのバックアップコピーを削除できます。

スタンドアロンNNMi管理サーバーのIPアドレスの変更

NNMi管理サーバーのIPアドレスを変更するには、以下の手順を実行します。

1. http://www.webware.hp.comに移動します。
2. ログインし、プロンプトに従って新規IPアドレスのライセンスキーを取得します。
3. 新しいライセンスキーをlicense.txtという名前のテキストファイルにコピーします。
4. コマンドプロンプトで、以下のコマンドを入力します。

   nnmlicense.ovpl NNM -f license.text -nosync

   ovstop
5. NNMi管理サーバーを新しいIPアドレスで設定します。
6. NNMi管理サーバーの新しいIPアドレスを認識するようにDNSサーバーを設定します。
7. NNMi管理サーバーを再起動します。
8. コマンドプロンプトで、以下のコマンドを入力します。

   nnmlicense.ovpl NNM -g
9. [**Autopass:ライセンス管理**] ダイアログボックスで、[**ライセンスキーの削除**] をクリックします。
10. 古いIPアドレスに関連付けられた、削除するライセンスキーを選択します。
11. [**ライセンスを恒久的に削除**] を選択します。
12. [**削除**] をクリックしてから、ダイアログボックスを閉じます。

NNMi管理サーバーのホスト名またはドメイン名の変更

注: NNMi管理サーバーがNNMiアプリケーションフェイルオーバーに関与、または高可用性 (HA) クラスターのメンバーの場合は、サポート担当者に問い合わせてください。

NNMi管理サーバーのホスト名、ドメイン名、または両方を変更するには、nnmsetofficialfqdn.ovplコマンドを使用して、NNMi管理サーバーの新しい完全修飾ドメイン名 (FQDN) を使用するようにNNMiを設定します。例:

nnmsetofficialfqdn.ovpl newnnmi.servers.example.com

詳細については、nnmsetofficialfqdn.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

注: FQDNは、ドメイン名と組み合わされたホスト名です。このいずれかを変更すると、NNMi管

理サーバーのFQDNを変更することににになります。SSL証明書は、常にFQDNにリンクされます。証明書の共通名 (CN) フィールドは、サーバーのFQDNと一致する必要があります。このため、FQDNを変更する場合は、一致するCNを持つ新しいSSL証明書が必要になります。`nnmsetofficialfqdn.ovpl`コマンドは、NNMi管理サーバーのFQDNを更新し、新しいFQDNと一致する新しい自己署名証明書も作成します。ただし、CA証明書を使用している場合は、新しいCA証明書を生成する必要があります。詳細については、「CA署名証明書の生成」(320ページ)を参照してください。

(FQDNを変更したかどうかに関係なく) NNMi管理サーバーのIPアドレスを変更する場合は、新しいライセンスを取得する必要があります。詳細については、「スタンドアロンNNMi管理サーバーのIPアドレスの変更」(308ページ)を参照してください。

Oracleデータベースインスタンス接続情報の変更

NNMiが一度に接続できるOracleデータベースインスタンスは1つです。この接続を設定できます。

以下のような場合にOracleデータベースインスタンス接続情報を変更します。

- Oracleデータベースのサーバー名を変更する必要がある。
- データベースへ接続するポートが別のプロセスと競合している、または企業ポリシーでデフォルト以外のポートを使用する必要がある。
- データベースインスタンス名を変更する必要がある (たとえば、企業ポリシーに準拠するため)。
- Oracleデータベースサーバーのハードウェアを変更する必要がある。

NNMiで使用するOracleデータベースインスタンスを変更するには、以下のタスクを実行します。

「タスク1:Oracleデータベースインスタンスの更新」(309ページ)

「タスク2:NNMi設定の更新」(310ページ)

タスク1:Oracleデータベースインスタンスの更新

1. NNMi管理サーバーを停止します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。
2. データベースを移動、Oracleデータベースサーバー名を変更、またはその他の必要な変更を行ってOracleデータベースを準備します。
3. ターゲットのOracleデータベースインスタンスが、以下の前提条件を満たしていることを確認します。
   - データベースインスタンスが存在している。
   - データベースインスタンスに正しいNNMiデータが入力されている。
   - Oracleツールを使用して、NNMiデータを作業用データベースインスタンスからターゲットの

データベースインスタンスにコピーする。

- データベースインスタンスを実行中である。

タスク2:NNMi設定の更新

1. データベース接続設定ファイルのバックアップ

   ディレクトリを次のように変更します。

   Windowsの場合: `%NnmInstallDir%\nonOV\jboss\nms\server\nms\`

   Linuxの場合: `$NnmInstallDir/nonOV/jboss/nms/server/nms/`

   `nms`ディレクトリ内に、`deploy.save`というディレクトリを作成します。

   `nms-ds.xml`ファイルを配備ディレクトリから`deploy.save`ディレクトリにコピーします。

   注意: `ovjboss`プロセスは起動時に配備ディレクトリ階層内のすべてのファイルを読み取ります。そのため、この例で`deploy.save`ディレクトリを使用する場合と同様に、デプロイされているファイルは`deploy`ディレクトリ階層外の場所にバックアップコピーを保存します。

2. データベース接続設定ファイルの編集

   `deploy`ディレクトリに移動します。

   任意のテキストエディターで、`nms-ds.xml`ファイルを開きます。

   connection-urlエントリを検索します。

   例:

   ```
   <connection-url>jdbc:oracle:thin:@ohost:1521:nnmidb1</connection-url>
   ```

   このエントリの最後の3つのパラメーターが重要です。これらの形式は`oracle_hostname:database_port:database_instance_name`です。

   connection-urlエントリの4番目、5番目、または6番目のパラメーターの1つ以上を変更します。

   例:

   異なるOracleデータベースサーバーを指すには、`ohost`を別のホスト名に変更します。

   別のポートのOracleデータベースサーバーに接続するには、`1521`を別のポート番号に変更します。

   別のOracleデータベースインスタンスに接続するには、`nnmidb1`を別のデータベースインスタンス名に変更します。

   注: このデータベースインスタンスはすでに存在している必要があります。

   `nms-ds.xml`ファイルを保存します。

3. NNMi管理サーバーを起動します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

NNMiがOracleデータベースインスタンスへの接続に使用するパスワードの変更

NNMiデータベースインスタンスへの接続に異なるパスワードを使用するようにOracle設定を変更するには、以下の手順を実行してNNMi設定を更新します。

1. NNMi管理サーバーを停止します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。
2. `nnmchangedbpw.ovpl`コマンドを実行し、プロンプトに従います。
3. NNMi管理サーバーを起動します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

詳細については、`nnmchangedbpw.ovpl`のリファレンスページ、またLinuxのマニュアルページを参照してください。

第6章: 詳細設定

このセクションでは以下の章について説明します。

NNMiのライセンス

恒久ライセンスキーをインストールしていない場合、NNMi製品には、NNMiのインストール後60日間有効な一時試用ライセンスキーが含まれています。この一時試用ライセンスキーを使用すると、NNMi Ultimate機能を使用できるようになります。できるだけ早く、恒久ライセンスキーを入手し、インストールしてください。

注: NNMi (単品)、およびNNMiに同梱されているNNMi Advanced機能とNNM iSPI NET機能を購入した場合、アプリケーションフェイルオーバーおよび高可用性環境で使用するためのライセンスには2つのタイプがあります。

- アプリケーションフェイルオーバー
  - 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセンスです。このライセンスをプライマリサーバーのIPアドレスと関連付けます。
  - 非商用 - このライセンスは、アプリケーションフェイルオーバー環境で使用するために個別に購入されます。このライセンスをセカンダリ (スタンバイ) サーバーのIPアドレスと関連付けます。

  高可用性 (HA)
  - 商用 - これは、アプリケーションフェイルオーバーまたは高可用性環境があるかないかに関係なく、NNMi、NNMi Advanced、またはNNM iSPI NETのために購入される主要なライセ

ンスです。このライセンスを物理クラスターノードのいずれかのIPアドレスに関連付けます。

- 非商用 - このライセンスは、高可用性環境で使用するために個別に購入されます。このライセンスをNNMi HAリソースグループの仮想IPアドレスに関連付けます。

- 指示されたように非商用ライセンスを使用する代わりにNNMi PremiumまたはNNMi Ultimateを購入した場合、アップリケーションフェイルオーバーまたは高可用性で使用するには、HPパスワード配信センターから、要求したライセンスキーを使用する必要があります。必ず以下を要求します。
  - 高可用性:NNMi HAリソースグループの仮想IPアドレス用のライセンスキーを取得します。このライセンスキーは、最初はプライマリサーバーで使用され、必要に応じてセカンダリサーバーで使用されます。
  - アプリケーションフェイルオーバー:プライマリサーバーの物理IPアドレスに1つと、スタンバイサーバーの物理IPアドレスに1つの、2つのライセンスキーを取得します。

  注:同じサーバーで商用ライセンスと非商用ライセンスを使用しないでください。
- 以下で入手できる各NNM iSPIのドキュメントも参照してください。
  http://h20230.www2.hp.com/selfsolve/manuals。

NNMi Ultimateライセンスに含まれている機能のリストを表示するには、『HP NNMi Softwareリリースノート』の「ライセンス」のセクションを参照してください。

恒久ライセンスキーのインストール準備

試用ライセンスでは、250ノードまでの制限が付けられています。試用ライセンスキーでNNMiを実行している場合、恒久ライセンスでサポートできる数以上のノードを管理できる場合があります。

ライセンス情報を追跡する際には、以下の点に注意してください。

- **消費量**:NNMiは、NNMiのライセンス容量限界までノードを検出および管理します (切り上げ)。
  - VMware環境:デバイスプロファイルがvnwareVMの各デバイスは、1/10のノードと同等です。
  - 他のすべてのデバイスは1つの検出されたノードと同等です。

ライセンス限度の詳細については、NNMi管理者用のヘルプの「NNMiライセンスを追跡する」を参照してください。

- 検出されたノードの数がライセンスされた容量限界に到達または超えた場合、次のいずれかが行われないかぎり、新しいノードは検出されません。
  - ライセンス拡張をインストールする。
  - 設定を確認し、NNMi検出をネットワーク環境内の重要なノードのみに限定する。次にノードを

削除し、NNMiの再検出でノードの管理対象インベントリをリセットする。

詳細については、NNMiオンラインヘルプを参照してください。

ライセンスの種類および管理対象ノードの数の確認

現在、NNMiが使用しているライセンスの種類を確認するには、以下の手順を実行します。

1. NNMiコンソールで、**[Help] > [HP Network Node Manager i Softwareについて]** の順にクリックします。
2. **[HP Network Node Manager i Softwareについて]** ウィンドウで、**[ライセンス情報]** をクリックします。
3. **[消費量]** フィールドに表示される値を探します。この値が、現在NNMiが管理しているノードの数です。

   ライセンス情報を追跡する際には、以下の点に注意してください。

   - **消費量**:NNMiは、NNMiのライセンス容量限界までノードを検出および管理します (切り上げ)。
     - VMware環境:デバイスプロファイルがvnwareVMの各デバイスは、1/10のノードと同等です。
     - 他のすべてのデバイスは1つの検出されたノードと同等です。

   ライセンス限度の詳細については、NNMi管理者用のヘルプの「NNMiライセンスを追跡する」を参照してください。

   ライセンス限度の詳細については、NNMi管理者用のヘルプの「NNMiライセンスを追跡する」を参照してください。
4. 恒久ライセンスがサポートできるノード数が、現在NNMiが管理しているノード数より少ない場合は、NNMiコンソールを使用して、あまり重要でないノードを削除します。詳細については、NNMiヘルプの「ノードの削除」を参照してください。

恒久ライセンスキーの取得およびインストール

恒久ライセンスキーを申請するには、以下の情報が必要です。

- HP製品番号や製造番号が明記されたエンタイトルメント証明書
- NNMi管理サーバーの1つのIPアドレス
- HAで動作するNNMiのライセンスの場合は、NNMi HAリソースグループの仮想IPアドレス
- お客様の企業情報もしくは団体情報

AutopassおよびHP注文番号の使用 (ファイアウォール使用時は不可)

恒久ライセンスキーを入手してインストールするには、以下の手順を実行します。

1. コマンドプロンプトで、以下のコマンドを入力し、Autopassユーザーインタフェースを開きます。

   nnmlicense.ovpl NNM -gui

2. [Autopass] ウィンドウの左側にある [**ライセンス管理**] をクリックします。
3. [**ライセンスキーのインストール**] をクリックします。
4. [**ライセンスキーの取得/インストール**] をクリックします。
5. HP注文番号を入力し、Autopassプロンプトに従ってライセンスキーの取得プロセスを完了します。
6. NNMiにより、インストールが自動的に完了します。

コマンドラインでのライセンスの追加

自動プロセスが完了しない場合は (NNMi管理サーバーがファイアウォールの背後にある場合など)、以下の手順を実行します。

1. ライセンスキーを取得するには、以下のHPパスワード配信サービスに移動します。

   https://webware.hp.com/welcome.asp

2. NNMi管理サーバーのコマンドプロンプトで以下のコマンドを入力し、システムを更新して、ライセンスデータファイルを保存します。

   nnmlicense.ovpl NNM -flicense_file

   (製品ライセンスID (NNM) では大文字と小文字が区別されます。)

   詳細については、nnmlicense.ovplのリファレンスページまたはLinuxのマンページを参照してください。

3. NNMiにより、インストールが自動的に完了します。

ライセンスキーの追加取得

NNMiのライセンス構造や、企業向けインストールにライセンス層を追加する方法の詳細については、HP営業担当者またはHewlett-Packard正規販売店にお問い合わせください。

追加のライセンスキーを取得するには、HPライセンスキー配信サービスに移動します。

https://webware.hp.com/welcome.asp

詳細については、NNMiヘルプの「ライセンス容量を拡張する」を参照してください。

開発者の方へ: NNMi開発者ツールキットを使用すると、カスタムWebサービスクライアントを統合してNNMiの機能を拡張できます。NNMi開発者ライセンスをインストールすると、NNMiによりdocフォルダーにsdk-dev-kit.jarファイルが作成されます。sdk-dev-kit.jarファイルを解凍すると、NNMi開発者ツールキットドキュメントやサンプル集を表示できます。

証明書の管理

証明書は、Webサーバーの識別情報をブラウザーに示すものです。この証明書には、自己署名するか、CA (認証機関) による署名を付けることができます。nnm.keystoreファイルでは、プライベートキーと証明書は対応する公開キーとともに格納されます。nnm.truststoreファイルには、通信する他者の証明書、または他者を識別するときに信頼する認証機関の証明書が保存されています。NNMiは、nnm.keystoreファイルとnnm.truststoreファイルの両方に自己署名証明書を含めます。

特定のNNMi機能を使用するため、NNMi管理サーバーはそれぞれの証明書を相互に共有する必要があります。この章では、NNMi管理サーバー間でこれらの証明書をコピーする方法と、nnmcertmerge.ovplスクリプトを使用してnnm.keystoreおよびnnm.truststoreファイルに証明書をマージする方法について説明します。この章では、期限の切れた証明書を新しい自己署名証明書またはCA署名証明書と置き換える方法についても説明します。

管理者は、ネットワークからNNMiへのHTTPやその他の非暗号化アクセスを無効にできます。「リモートアクセスには暗号化を必須とするようにNNMiを設定する」(264ページ)を参照してください。

この章には、以下のトピックがあります。

NNMi証明書について

このセクションでは、証明書を使用する上で参考となる用語について説明します。以下の表に示した用語をよく理解してください。

証明書関連の用語

コンセプト	説明
キーストアーとトラストストアー	**トラストストアー**: NNMiトラストストアーは、NNMiが信頼するソースから取得した公開キーを格納するnnm.truststoreファイルです。 **キーストアー**: NNMiキーストアーは、NNMiサーバーのプライベートキーをインポートするnnm.keystoreファイルです。 nnm.truststoreファイルとnnm.keystroreファイルは、以下の場所に格納されていま

証明書関連の用語 (続き)

コンセプト	説明
	す。 • Linuxの場合: $NNM_DATA/shared/nnm/certificates/ • Windowsの場合: %NNM_DATA%\shared\nnm\certificates\
デフォルトのNNMi証明書	NNMiは、デフォルトのプロパティを使用して生成される自己署名証明書とともにインストールされます。このデフォルトの証明書は、別の自己署名証明書またはCA署名の証明書に置き換えることができます。
ツール	JavaのKeytoolユーティリティを使用して証明書を生成および管理します。NNMiには、証明書をマージしてNNMiシステムでの信頼性を確立するnnmmergecert.ovplユーティリティも付属しています。このプログラムは、高可用性、フェイルオーバー、およびGNM-RNMのセットアップで使用します。
サポートされる暗号化アルゴリズム	NNMiは、RSAアルゴリズムを使用して生成された証明書を受け入れます。DSAアルゴリズムはサポートされません。
自己署名証明書	自己署名証明書は、一般にサーバーと既知のクライアントグループ間にセキュア通信を確立するために使用します。NNMiは、デフォルトのプロパティを使用して生成される自己署名証明書とともにインストールされます。 **注:** 自己署名証明書を使用するように設定されているNNMiインスタンスは、ユーザーがWebブラウザーでNNMi Webコンソールへのアクセスを試みると警告メッセージを表示します。
CA署名証明書	証明書署名要求に対する応答として受け取る署名付きサーバー証明書には、CA署名付きのNNMi証明書と、1つ以上のCA証明書が含まれます (1つ以上のCA証明書が存在する場合は証明書チェーンとも呼ばれる)。 **注:** これらの証明書は1つのファイルに入っていることもあれば、2つの別々のファイルに入っていることもあります。
ルートCA証明書	サーバーおよびユーザーの証明書の署名について信頼できる認証機関を示します。
中間CA証明書	サーバーまたはユーザーではなく、ルートCAまたは中間CA (それ自体が署名機関) のいずれかで署名される証明書。 **注:** 中間CA証明書を含め、NNMiサーバー証明書からルートCA証明書にいたるまでの証明書のリストは、証明書チェーンと呼ばれます。

既存の証明書と新規の自己署名証明書またはCA署名証明書との置き換え

自己署名証明書は、NNMiのインストール時に作成され、インストールされます。証明書の置き換えは一般に以下の目的で行います。

- デフォルトの証明書の代わりに新規の自己署名証明書またはCA署名証明書を使用する。
- 期限の切れた証明書を更新する。

証明書を置き換えるには、以下の手順を実行します。

1. 自己署名証明書を生成します。詳細については、「自己署名証明書の生成」(319ページ)を参照してください。
2. 組織でCAが署名した証明書が必要な場合は、CSR (証明書署名要求) ファイルを生成してCA署名証明書を取得します。詳細については、「CA署名証明書の生成」(320ページ)を参照してください。
3. 次のファイルを開き、`com.hp.ov.nms.ssl.KEY_ALIAS`変数を、証明書の生成時に`<alias>`に使用した値に更新します。
   - Windowsの場合: `%NNM_CONF%\nnm\props\nms-local.properties`
   - Linuxの場合: `$NNM_CONF/nnm/props/nms-local.properties`
4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

   > **注:** 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。
5. 以下の構文を使用してNNMiコンソールへのHTTPSアクセスをテストします。

   https://<fully_qualified_domain_name>:<port_number>/nnm/.

   CA署名証明書を使用した場合、ブラウザーによってCAが信頼されると、NNMiコンソールへのHTTPS接続が信頼されます。

   自己署名証明書を使用した場合、NNMiコンソールへの信頼性のないHTTPS接続についての警告メッセージがブラウザーに表示されます。

自己署名証明書の生成

自己署名証明書を生成するには、以下の手順を実行します。

1. nnm.keystoreおよびnnm.truststoreファイルが存在するNNMi管理サーバーのディレクトリに変更します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\certificates
   - Linuxの場合: $NnmDataDir/shared/nnm/certificates
2. nnm.keystoreファイルのバックアップコピーを保存します。

   注:
   - 既存のNNMi証明書を置き換える場合は、この手順を完了するまで既存の証明書を削除しないでください。暗号化された情報を新しい証明書に転送するには、インストールされた以前の証明書と新しい証明書の両方でNNMiを少なくとも1回は起動する必要があります。
   - クライアントサーバーに対してNNMi管理サーバーに新しい証明書を確実に表示するには、次の手順の説明に従って、NNMiが新しい証明書をポイントしていることを確認してください。
3. システムからプライベートキーを生成します。このプライベートキーを生成するには、keytoolコマンドを使用します。
   a. 以下のコマンドをそのまま実行します。
      - Windowsの場合: %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -genkeypair - validity 3650 -keyalg rsa -keystore nnm.keystore -storepass nnmkeypass - alias <alias_name>
      - Linuxの場合: $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -genkeypair - validity 3650 -keyalg rsa -keystore nnm.keystore -storepass nnmkeypass - alias <alias_name>

        注: エイリアス(この例では<alias_name>)は、この新規作成キーを識別する名前です。エイリアスは任意の文字列にすることができますが、HPでは、正しいバージョンを簡単に識別できるように完全修飾ドメイン名(FQDN)に続けてサフィックスを指定することをお勧めします。たとえば、myserver.mydomain-<number>やmyserver.mydomain-<date>のようにエイリアス名を指定できます。

   b. 必要な情報を入力します。

      注意: 姓名の入力を求められたら、システムのFQDNを入力してください。

      自己署名証明書が生成されます。

      CA署名証明書を取得するためには、さらにCSRファイルを生成し、CAに送信する必要があります。詳細については、「CA署名証明書の生成」(320ページ)を参照してください。

CA署名証明書の生成

CA署名証明書を取得してインストールするには、以下の手順を実行します。

1. 自己署名証明書を生成します。詳細については、「自己署名証明書の生成」(319ページ)を参照してください。
2. 以下のコマンドを実行して、CSR (証明書署名要求) ファイルを作成します。
   - Windowsの場合: `%NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -keystore nnm.keystore -certreq -storepass nnmkeypass -alias <alias_name> -file CERTREQFILE`
   - Linuxの場合:`$NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -keystore nnm.keystore -certreq -storepass nnmkeypass -alias <alias_name> -file CERTREQFILE`

   注:
   - 上記のコマンドでは、`<alias_name>`は証明書の生成時に指定したエイリアスに相当します。
   - `keytool`コマンドの詳細については、**http://www.oracle.com/technetwork/java/index.html**で「鍵および証明書管理ツール」を検索してください。
3. CA署名機関にCSRを送信します (CA署名機関が証明書ファイルに署名して返します)。各種のCA証明書についての詳細は、「CA署名証明書のタイプ」(323ページ)を参照してください。
4. これらの証明書が記録されているファイルをNNMi管理サーバーのいずれかの場所にコピーします。この例では、以下の場所にファイルをコピーします。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\certificates`
   - Linuxの場合: `$NnmDataDir/shared/nnm/certificates`
5. `nnm.keystore`および`nnm.truststore`ファイルが存在するNNMi管理サーバーのディレクトリに変更します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\certificates`
   - Linuxの場合: `$NnmDataDir/shared/nnm/certificates`
6. 以下のコマンドを実行して、証明書を`nnm.keystore`ファイルにインポートします。

   Windowsの場合:
   - **`%NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -importcert -trustcacerts -keystore nnm.keystore -storepass nnmkeypass -alias <alias_name> -file <myserver.crt>`**

   Linuxの場合:

- **$NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -importcert -trustcacerts -keystore nnm.keystore -storepass nnmkeypass -alias <alias_name> -file** <myserver.crt>

  注:

  - 上記のコマンドで、
    - <myserver.crt>は、署名付きサーバー証明書を保存した場所の完全パスに相当します。
    - <alias_name>は、証明書の生成時に指定したエイリアスに相当します。
  - **-storepass**オプションを使用し、パスワードを入力する場合、キーストアープログラムはキーストアーパスワードの入力を要求しません。**-storepass**オプションを使用しない場合は、キーストアーパスワードの入力を求められたときに**nnmkeypass**と入力してください。

7. 証明書の信頼を確認するメッセージが表示されたら、**y**

   証明書をキーストアーにインポートするときの出力例

   このコマンドによる出力形式は以下のとおりです。

   Owner:CN=NNMi_server.example.com

   Issuer:CN=NNMi_server.example.com

   Serial number:494440748e5

   Valid from:Tue Oct 28 10:16:21 MST 2008 until:Thu Oct 04 11:16:21 MDT 2108

   Certificate fingerprints:

   MD5:29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02

   SHA1:C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03

   Trust this certificate?[no]:y

   Certificate was added to keystore

8. 以下のコマンドを実行して、証明書をnnm.truststore fileファイルにインポートします。

   - Windowsの場合:

     %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -import -alias <alias_name> -keystore nnm.truststore -file <myca.crt>

   - Linuxの場合:

     $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -import -alias <alias_name> -keystore nnm.truststore -file <myca.crt>

     注:

- 上記のコマンドで、
  - <myca.crt>は、CA証明書を保存した場所の完全パスに相当します。
  - <alias_name>は、証明書の生成時に指定したエイリアスに相当します。
- -storepassオプションを使用し、パスワードを入力する場合、キーストアープログラムはキーストアーパスワードの入力を要求しません。-storepassオプションを使用しない場合は、キーストアーパスワードの入力を求められたときにnnmkeypassと入力してください。

9. トラストストアーのパスワードの入力を求められたら、**「ovpass」と入力します。**
10. トラストストアーの内容を確認します。
    - Windowsの場合:

      %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool -list -keystore nnm.truststore

    - Linuxの場合:

      $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -list -keystore nnm.truststore

    トラストストアーのパスワードの入力を求められたら、**「ovpass」と入力します。**

    トラストストアーの出力例

    トラストストアーの出力形式は以下のとおりです。

    Keystore type: jks

    Keystore provider:SUN

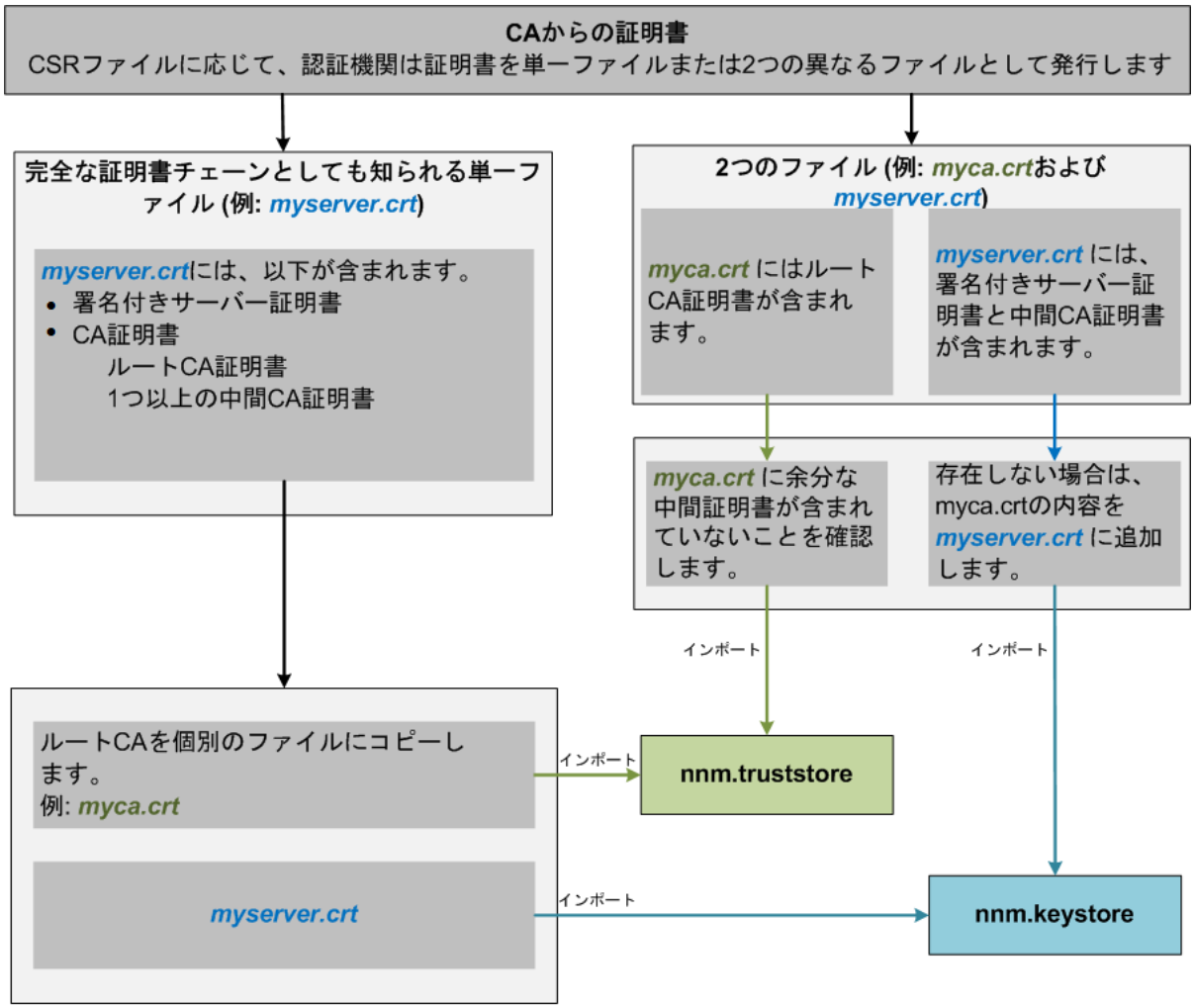
    Your keystore contains 1 entry

    nnmi_ldap, Nov 14, 2008, trustedCertEntry,

    Certificate fingerprint (MD5):29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02

    ヒント: トラストストアーには複数の証明書を含めることができます。

CA署名証明書のタイプ



注: CAによって証明書が別のフォームで返される場合は、証明書チェーンとルートCA証明書を取得する方法について、CA提供者に問い合わせてください。

認証機関 (CA) からは以下のいずれかが提供されます。

- **サーバー証明書** (CAによって署名されたNNMi証明書) と1つ以上のCA証明書が含まれる署名付きサーバー証明書ファイル。このセクションでは、署名付きサーバー証明書を`myserver.crt`として示しています。

  CA証明書には、以下のいずれかを指定できます。

  - ルートCA証明書 - サーバーおよびクライアントの証明書の署名について信頼できる機関を示します。
  - 中間CA証明書 - サーバーまたはユーザーではなく、ルートCAまたは中間CA (それ自体が署名機関) のいずれかで署名される証明書。

> **注:** 中間CA証明書を含め、NNMiサーバー証明書からルートCA証明書にいたるまでの証明書のリストは、**証明書チェーン**と呼ばれます。

- 署名付きサーバー証明書と、1つ以上のCA証明書が含まれる別のファイル。このセクションでは、署名付きサーバー証明書を`myserver.crt`、CA証明書を`myca.crt`として示しています。`myserver.crt`ファイルは、1つのサーバー証明書または証明書チェーンを含んでいる必要がありますが、`myca.crt`ファイル内にあるルートCA証明書を含んでいる必要はありません。

NNMiに新しい証明書を設定するには、証明書チェーンを`nnm.keystore`にインポートし、ルートCA証明書を`nnm.truststore`にインポートする必要があります。サーバー証明書を`nnm.keystore`ファイルにインポートする場合は`myserver.crt`ファイルを使用し、CA証明書を`nnm.truststore`ファイルにインポートする場合は`myca.crt`ファイルを使用します。

> **注:** CAによって証明書が別のフォームで返される場合は、別個の証明書チェーンとルートCA証明書を取得する方法について、CA提供者に問い合わせてください。

完全な証明書チェーンを含んでいる1つのファイルで提供された場合、そのファイルからルートCA証明書フォームを`myca.crt`ファイルにコピーします。`myca.crt`ファイルを使用して`nnm.truststore`へインポートすると、NNMiが証明書を発行したCAを信頼するようになります。

2つのファイルで提供された場合、`myca.crt`ファイルの内容を`myserver.crt`の末尾に追加します(ファイルに含まれていない場合)。また、余分な中間証明書がある場合は、それらを`myca.crt`からすべて削除します。これにより、完全な証明書チェーンを含んでいる1つのファイル`myserver.crt`と、ルートCA証明書を含んでいる1つのファイル`myca.crt`が生成されます。

> **注:** CAのみを使用している場合、一般にルートCA証明書が`nnm.truststore`に追加されます。中間CAまたはサーバー証明書を`nnm.truststore`に追加すると、それらの証明書は明示的に信頼済みとなり、取り消しなどの追加情報についてのチェックはされません。CAが要求する場合には、追加の証明書のみを`nnm.truststore`に追加してください。

以下は、CA署名機関から受け取るファイルの例です。

独立サーバーで、複数のCA証明書ファイルがある場合

```
-----BEGIN CERTIFICATE-----
Sample/AVQQKExNQU0EgQ29ycG9yYXRpb24gTHRkMRAwDgYDVQQLEwdOZXR3b3Js
eGVSZXZvY2F0aW9uTGlzdD9iYXNlP29iamVjdENsYXNzPWNSTERpc3RyaWJ1dGlw
.....
.....
TZImiZPyLGQBGRYDaW50MRIwEAYKCZImiZPyLGQBGRYCc2cxEzARBgNVBAMTCmNb
pSo6o/76yShtT7Vrlfz+mXjWyEHaIy/QLCpPebYhejHEg4dZgzWWT/lQt==
-----END CERTIFICATE-----
```

結合サーバーで、1つのファイルに複数のCA証明書がある場合

```
-----BEGIN CERTIFICATE-----
```

Sample1/VQQKExNQU0EgQ29ycG9yYXRpb24gTHRkMRAwDgYDVQQLEwdOZXR3b3Js

eGVSZXZvY2F0aW9uTGlzdD9iYXNlP29iamVjdENsYXNzPWNSTERpc3RyaWJ1dGlw

.....

.....

TZImiZPyLGQBGRYDaW50MRIwEAYKCZImiZPyLGQBGRYCc2cxEzARBgNVBAMTCmNb

pSo6o/76yShtT7Vrlfz+mXjWyEHaIy/QLCpPebYhejHEg4dZgzWWT/lQt==

-----END CERTIFICATE-----

-----BEGIN CERTIFICATE-----

Sample2/GhOdHA6Ly9jb3JwMWRjc2cyLnNnLmludC5wc2FnbG9iYWwuY29tL0Nlc

RaOCApwwggKYMB0GA1UdDgQWBBSqaWZzCRcpvJWOFPZ/Be9b+QSPyDAfBgNVHSMC

.....

.....

Wp5Lz1ZJAOu1VHbPVdQnXnlBkx7V65niLoaT90Eqd6laliVlJHj7GBriJ90uvVGu

BQagggEChoG9bGRhcDovLy9DTj1jb3JwMWRjc2cyL==

-----END CERTIFICATE-----

アプリケーションフェイルオーバー環境での証明書の使用

アプリケーションフェイルオーバーでの証明書の使用法

アプリケーションフェイルオーバー機能を設定するときには、両方のノードの`nnm.keystore`ファイルと`nnm.truststore`ファイルの内容をマージして、1個の`nnm.keystore`ファイルと1個の`nnm.truststore`ファイルにする必要があります。

以下の手順を実行し、自己署名証明書またはCA署名証明書を使用するようにアプリケーションフェイルオーバー機能を設定します。

注意: NNMiおよびアプリケーションフェイルオーバー機能で自己署名証明書を使用する場合、以下の手順を完了しないと、NNMiのプロセスがスタンバイNNMi管理サーバー (この例の`Server Y`)

で正常に起動しません。

1. Server Yで以下のディレクトリに変更します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\certificates
   - Linuxの場合: $NnmDataDir/shared/nnm/certificates
2. nnm.keystoreおよびnnm.truststoreファイルを、Server YからServer Xの一時保存場所にコピーします。以降の手順では、これらのファイル保存場所を<keystore>および<truststore>と呼びます。
3. Server Xで以下のコマンドを実行し、Server Yの証明書をServer Xのnnm.keystoreおよびnnm.truststoreファイルにマージします。

   Windowsの場合:

   nnmcertmerge.ovpl -keystore <keystore> -truststore <truststore>

   Linuxの場 合 :

   nnmcertmerge.ovpl -keystore <keystore> -truststore <truststore>
4. マージしたnnm.keystoreおよびnnm.truststoreファイルをserver Xからserver Yにコピーし、どちらのノードにもマージ済みファイルがあるようにします。これらのファイル保存場所は、以下のとおりです。
   - Windowsの場合: %NnmDataDir%\shared\nnm\certificates
   - Linuxの場合: $NnmDataDir/shared/nnm/certificates
5. Server XとServer Yの両方で以下のコマンドを実行します。完全修飾ドメイン名を含め、両方のサーバーからの表示結果が一致することを確認します。一致しない場合は続行せずに、326から327までをやり直します。

   Windowsの場合:

   %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -list -keystore %NnmDataDir%\shared\nnm\certificates\nnm.keystore -storepass nnmkeypass

   Linuxの場合:

   $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -list -keystore $NnmDataDir/shared/nnm/certificates/nnm.keystore -storepass nnmkeypass
6. Server XとServer Yの両方で以下のコマンドを実行します。完全修飾ドメイン名を含め、両方のサーバーからの表示結果が一致することを確認します。一致しない場合は続行せずに、326から327までをやり直します。

   Windowsの場合:

   %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -list -keystore %NnmDataDir%\shared\nnm\certificates\nnm.truststore -storepass ovpass

Linuxの場合:

$NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -list -keystore $NnmDataDir/shared/nnm/certificates/nnm.truststore -storepass ovpass

7. 「アプリケーションフェイルオーバー構成のNNMiの設定」(142ページ)から、アプリケーションフェイルオーバー機能の設定を続行します。

高可用性環境での証明書の使用

このセクションでは、HA環境で自己署名証明書またはCA証明書を使用するようにNNMiを設定する方法について説明します。

HAでの証明書の使用法

デフォルト証明書を使用した高可用性の設定

NNMi でHAを正しく有効にするための設定プロセスでは、プライマリクラスターノードとセカンダリクラスターノードの間でデフォルトの自己署名証明書を共有します。HA下で実行されるNNMiでデフォルトの証明書を使用するために、追加の手順を実行する必要はありません。

新しい証明書を使用した高可用性の設定

このセクションでは、newcertという新規の自己署名証明書またはCA証明書を作成します。以下の手順を実行して、この新規のCA証明書または自己署名証明書を使用するようにHAを設定します。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

ヒント: この手順は、「高可用性環境での共有NNMiデータ」(196ページ) の説明に従って、NNMiにHAを設定する前または後に実行できます。

1. 手順2を完了する前に、NNMi_HA1で以下のディレクトリに変更します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\certificates
   - Linuxの場合: $NnmDataDir/shared/nnm/certificates
2. NNMi_HA1で、以下のコマンドを実行してnewcertをnnm.keystoreファイルにインポートします。
   - Windowsの場合:**%NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool -import -alias newcert_Alias -keystore nnm.keystore -file newcert**
   - Linuxの場合:**$NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -import -alias newcert_Alias -keystore nnm.keystore -file newcert**
3. アクティブノード (NNMi_HA1) とスタンバイノード (NNMi_HA2) の両方で以下のファイルを編集します。
   - Windowsの場合: %NnmDataDir%\conf\nnm\props\nms-local.properties
   - Linuxの場合: $NnmDataDir/conf/nnm/props/nms-local.properties
4. NNMi_HA1とNNMi_HA2の両方のnms-local.propertiesファイルで、以下の行を変更します。

   com.hp.ov.nms.ssl.KEY_ALIAS = newcert_Alias
5. 変更を保存します。

グローバルネットワーク管理環境での証明書の使用

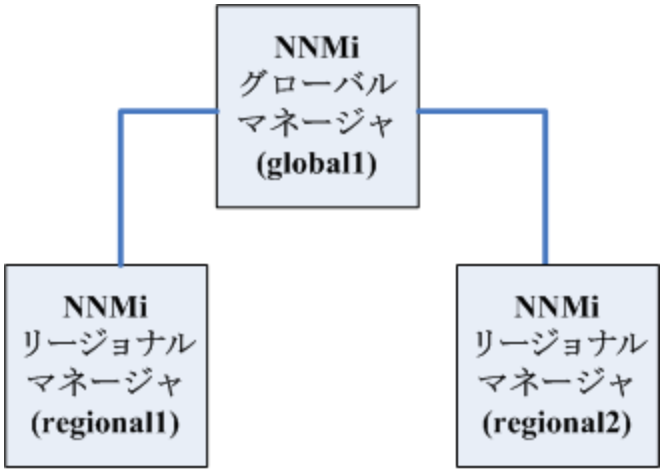
グローバルネットワーク管理環境での証明書の設定

NNMiのインストール時には、インストールスクリプトによってNNMi管理サーバーの自己署名証明書が作成されます。この証明書には、ノードの完全修飾ドメイン名を含むエイリアスが記録されています。インストールスクリプトは、この自己署名証明書をNNMi管理サーバーのnnm.keystoreおよびnnm.truststoreファイルに追加します。

以下の手順を実行し、以下の図に基づいて自己署名証明書またはCA署名証明書を使用するようにグローバルネットワーク管理機能を設定します。

開始する前に、必要な証明書がリージョナルマネージャーシステムで作成されていることを確認してください。詳細については、「既存の証明書と新規の自己署名証明書またはCA署名証明書との置き換え」(318ページ)を参照してください。

グローバルネットワーク管理



1. regional1およびregional2で以下のディレクトリに変更します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\certificates
   - Linuxの場合: $NnmDataDir/shared/nnm/certificates
2. nnm.truststoreファイルを、上記のregional1およびregional2の場所から、global1の任意の一時保管場所にコピーします。
3. global1で以下のコマンドを実行し、regional1およびregional2の証明書をglobal1のnnm.truststoreファイルにマージします。

   Windowsの場合:

   a. **nnmcertmerge.ovpl -truststore regional1_nnm.truststore_location**
   b. **nnmcertmerge.ovpl -truststore regional2_nnm.truststore_location**

   Linuxの場合:

   a. **nnmcertmerge.ovpl -truststore regional1_nnm.truststore_location**
   b. **nnmcertmerge.ovpl -truststore regional2_nnm.truststore_location**
4. global1で、以下のコマンドを以下の順序で実行します。
   a. global1のNNMi管理サーバーで**ovstop**を実行します。
   b. global1のNNMi管理サーバー で**ovstart**を実行します。

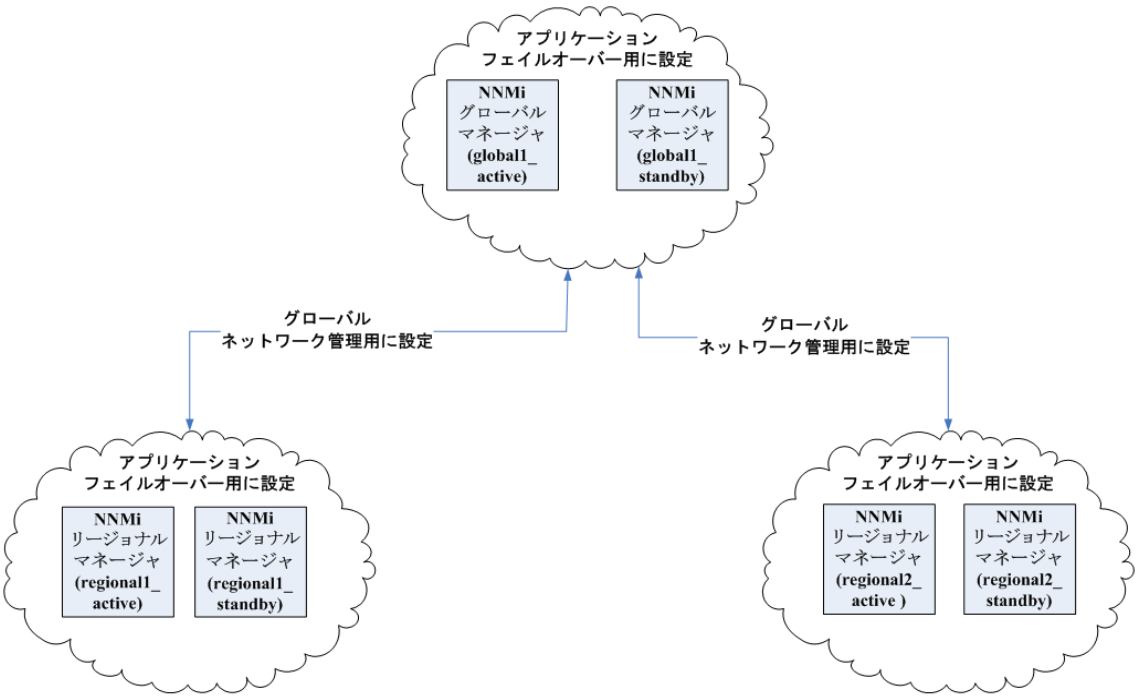
注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

フェイルオーバーが有効なグローバルネットワーク管理環境での証明書の設定

NNMiのインストール時には、インストールスクリプトによってNNMi管理サーバーの自己署名証明書が作成されます。この証明書には、ノードの完全修飾ドメイン名を含むエイリアスが記録されています。インストールスクリプトは、この自己署名証明書をNNMi管理サーバーの`nnm.keystore`および`nnm.truststore`ファイルに追加します。

この例では、以下の図に示すように、アプリケーションフェイルオーバー機能でグローバルネットワーク管理設定を使用します。

アプリケーションフェイルオーバーが有効なグローバルネットワーク管理



以下の手順を実行し、上の図に基づいてアプリケーションフェイルオーバーが有効なグローバルネットワーク管理機能を設定します。

1. 上の図に示すアプリケーションフェイルオーバークラスターごとに、「アプリケーションフェイルオーバー環境での証明書の使用」(325ページ)に示す指示に従ってください。
2. 「アプリケーションフェイルオーバーの要件」(143ページ)の指示に従ってアプリケーションフェイルオーバーを設定します。
3. 「グローバルネットワーク管理環境での証明書の設定」(328ページ)に示す`regional1_active and regional2_active`に関する指示に従ってください。

ディレクトリサービスへのSSL接続を設定する

デフォルトでは、ディレクトリサービス通信を有効にすると、NNMiは、ディレクトリサービスからデータを取得するときにLDAPプロトコルを使用します。ディレクトリサービスでSSL接続が必要な場合は、SSLプロトコルを有効にして、NNMiとディレクトリサービスの間を流れるデータを暗号化する必要があります。

SSLでは、ディレクトリサービスホストとNNMi管理サーバーの間で信頼関係を確立する必要があります。この信頼関係を確立するには、証明書をNNMiトラストストアーに追加します。証明書は、ディレクトリサービスホストの識別情報をNNMi管理サーバーに示すものです。

SSL通信用のトラストストアー証明書をインストールするには、以下の手順を実行します。

1. ディレクトリサーバーから会社のトラストストアー証明書を取得します。ディレクトリサービス管理者からこの証明書のテキストファイルのコピーを入手できます。
2. NNMiトラストストアーが格納されているディレクトリに変更します。
   - Windowsの場合: `%NnmDataDir%\shared\nnm\certificates`
   - Linuxの場合: `$NnmDataDir/shared/nnm/certificates`

   `certificates`ディレクトリから、この手順のコマンドすべてを実行します。
3. 会社のトラストストアー証明書をNNMiトラストストアーにインポートします。
   a. 以下のコマンドを実行します。
      - Windowsの場合:

        ```
        %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -import
        -alias nnmi_ldap -keystore nnm.truststore
        -file <Directory_Server_Certificate.txt>
        ```
      - Linuxの場合:

        ```
        $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -import
        -alias nnmi_ldap -keystore nnm.truststore
        -file <Directory_Server_Certificate.txt>
        ```

        `<Directory_Server_Certificate.txt>`は、会社のトラストストアー証明書です。
   b. キーストアーのパスワードの入力を求められたら、**「ovpass」と入力します。**
   c. 証明書の信頼を確認するメッセージが表示されたら、**y**

      証明書をトラストストアーにインポートするときの出力例

      このコマンドによる出力形式は以下のとおりです。

      Owner:CN=NNMi_server.example.com

      Issuer:CN=NNMi_server.example.com

      Serial number:494440748e5

      Valid from:Tue Oct 28 10:16:21 MST 2008 until:Thu Oct 04 11:16:21 MDT 2108

      Certificate fingerprints:

MD5:29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02

SHA1:C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03:7E:C4:03

Trust this certificate?[no]:y

Certificate was added to keystore

4. トラストストアーの内容を確認します。
   - Windowsの場合:

     %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -list -keystore nnm.truststore

   - Linuxの場合:

     $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -list -keystore nnm.truststore

   キーストアーのパスワードの入力を求められたら、**「ovpass」と入力します。**

   トラストストアーの出力例

   トラストストアーの出力形式は以下のとおりです。

   Keystore type: jks

   Keystore provider:SUN

   Your keystore contains 1 entry

   nnmi_ldap, Nov 14, 2008, trustedCertEntry,

   Certificate fingerprint (MD5):29:02:D7:D7:D7:D7:29:02:29:02:29:02:29:02:29:02

   ヒント: トラストストアーには複数の証明書を含めることができます。

5. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーでovstopコマンドを実行します。
   b. NNMi管理サーバーでovstartコマンドを実行します。

   注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

keytoolコマンドの詳細については、**http://www.oracle.com/technetwork/java/index.html**で「鍵および証明書管理ツール」を検索してください。

NNMiとシングルサインオン (SSO) の使用

HP Network Node Manager i Software (NNMi) シングルサインオン (SSO) を設定すると、NNMiコンソールから簡単にNNM iSPIsにアクセスできるようになります。SSOを使用してNNMiコンソールにログオンすれば、NNM iSPIsや他のHPアプリケーションにアクセスできます。再度ログインする必要はありません。SSOは、安全なアクセスレベルを維持しながら、より簡単にNNM iSPIsや他のHPアプリケーションにアクセスできるようにする機能です。NNMiコンソールからサインアウト (またはNNMiコンソールセッションがタイムアウト) した後にNNMiコンソールとは異なるNNM iSPIやほかのアプリケーションのURLにアクセスするには、サインイン資格証明を再入力する必要があります。

インストール中にSSOは無効になっています。SSOが有効になっていても、あるNNMi管理サーバーから別の管理サーバーへと参照すると、最初の管理サーバーからログアウトされ、利益はほとんどありません。これが起こらないようにするために、SSOは無効に初期設定されており、この章で説明されているように、NNMi管理サーバー間で`initString`パラメーターと`protectedDomains`パラメーターの設定を調整できます。

この章には、以下のトピックがあります。

NNMiへのSSOアクセス

複数のNNMi管理サーバー間を移動するには、以下のいずれかを実行します。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてから`ovstop`コマンドおよび`ovstart`コマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

- `nms-ui.properties`ファイルを編集して、`com.hp.nms.ui.sso.initString`と`com.hp.nms.ui.sso.protectedDomains`のパラメーター値をNNMi管理サーバー間で同じ値にします。`com.hp.nms.ui.sso.domain`パラメーターを、NNMi管理サーバーが配置されているドメインと一致するように設定してください。
  - NNMi管理サーバーを1つのネットワークドメインにしか配置していない場合は、「1つのドメインに対するSSOの有効化」(334ページ)の説明に従ってください。

- NNMi管理サーバーを複数のネットワークドメインに配置している場合、詳細については、「異なるドメインに配置されているNNMi管理サーバーに対するSSOの有効化」(335ページ)の説明に従ってください。

- nms-ui.properties fileを編集し、SSOが無効になっていることを確認します。詳細については、「SSOの無効化」(338ページ)を参照してください。

これらのアクションのいずれかが完了していないと、別のNNMi管理サーバーに移動するたびに、直前のNNMi管理サーバーから自動的にサインアウトします。

SSOとNNMiグローバルネットワーク管理機能を併用する場合、特別な考慮事項があります。詳細については、「SSOおよびアクションメニュー」(462ページ)および「グローバルネットワーク管理用にシングルサインオンを設定する」(462ページ)を参照してください。

NNMi管理サーバーのドメイン名がmycompanyのように短く、ドット (.) がない場合、NNMiコンソールによりただちにサインアウトされます。SSOブラウザークッキーの制限には、mycompany.comのように、ドット (.) が少なくとも1つ付いているドメイン名が必要です。この状況を解決するには、以下の手順を実行します。

1. 以下のファイルをテキストエディターで開きます。
   - Windowsの場合: %NNM_PROPS%/nms-ui.properties
   - Linuxの場合: $NNM_PROPS/nms-ui.properties
2. この例では、以下の文字列を検索します。

   com.hp.nms.ui.sso.domain = mycompany

   これを以下の文字列で置き換えます。

   com.hp.nms.ui.sso.domain = mycompany.com
3. 以下のコマンドを実行し、変更をコミットします。

   nnmsso.ovpl -reload

   詳細については、nnmsso.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

1つのドメインに対するSSOの有効化

1つのドメインでSSOを使用可能にするには、以下の手順を実行します。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

1. 以下のファイルを開きます。
   - Windowsの場合: %NNM_PROPS%\nms-ui.properties
   - Linuxの場合: $NNM_PROPS/nms-ui.properties
2. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.isEnabled = false

   これを以下のように変更します。

   com.hp.nms.ui.sso.isEnabled = true
3. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.domain = mycompany.com

   mycompany.comを、NNMi管理サーバーが配置されているドメインに変更します。 1つのドメインでSSOを有効にするときは、1つのドメインのみがリストされていることを確認してください。
4. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.protectedDomains = mycompany.com

   mycompany.comを、NNMi管理サーバーが配置されているドメインに変更します。 1つの保護ドメインでSSOを有効にするときは、1つの保護ドメインのみがリストされていることを確認してください。
5. 以下のコマンドを実行し、変更をコミットします。

   nnmsso.ovpl -reload

   詳細については、nnmsso.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

異なるドメインに配置されているNNMi管理サーバーに対するSSOの有効化

SSOを使用できるように複数のNNMi管理サーバーを設定できます。この例では、異なるドメインに配置されている3つのNNMi管理サーバーに対してSSOを設定する方法を説明します。SSOを使用できるように複数のNNMi管理サーバーを設定する必要がある場合に、これらのシステムが異なるドメインに配置されているときは、以下の手順を実行します。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

1. 以下のファイルを開きます。
   - Windowsの場合: %NNM_PROPS%\nms-ui.properties
   - Linuxの場合: $NNM_PROPS/nms-ui.properties
2. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.isEnabled = false

   これを以下のように変更します。

   com.hp.nms.ui.sso.isEnabled = true
3. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.domain = group1.mycompany.com

   ドメイン名に1つ以上のドット (.) があることを確認します。
4. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.protectedDomains=group1.mycompany.com

   これを以下のように変更します。

   com.hp.nms.ui.sso.protectedDomains=group1.mycompany.com, group2.yourcompany.com, group3.yourcompany.com
5. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.initString =Initialization String

   1つのSSO設定で機能するように各NNMi管理サーバーの初期化ストリングを共有する必要があります。SSO設定に含まれるすべてのNNMi管理サーバーの初期化ストリングを同じ値に変更します。
6. 以下のコマンドを実行し、変更をコミットします。

   nnmsso.ovpl -reload

   詳細については、nnmsso.ovplのリファレンスページ、またはLinuxのマンページを参照してください。
7. 手順1から手順6までをさらに2回繰り返し、残りの2つのNNMi管理サーバーを設定します。残りの各NNMi管理サーバーについては、手順3で、group2またはgroup3をgroup1に置き換えてください。

NNMiとNNM iSPIsのSSOアクセス

SSOが有効になったら、NNMiとNNM iSPIs間のSSOにはinitString設定は必要ありません。

SSOを使用するには、以下のようにNNMiにアクセスします。

- 以下の形式の正しいURLを使用します。

  <protocol>://<fully_qualified_domain_name>:<port_number>/nnm/ **<protocol>**はhttpまたはhttpsです。

  <fully_qualified_domain_name>は、NNMi管理サーバーの正式な完全修飾ドメイン名 (FQDN) です。

  <port_number>は、NNMiコンソールに接続するためのポートです。これは、NNMiのインストール時に割り当てられ、以下のファイルで指定されます。

  - Windowsの場合: `%NnmDataDir%\conf\nnm\props\nms-local.properties`
  - Linuxの場合: `$NnmDataDir/conf/nnm/props/nms-local.properties`

- 有効なアカウントを使用してNNMiにログオンします。

SSOが機能するには、NNMiとNNM iSPIsへのURLアクセスに共通するネットワークドメイン名が使用されている必要があります。さらに、IPアドレスが含まれていないURLである必要があります。NNMi管理サーバー用のFQDNがない場合は、代わりにNNMi管理サーバーのIPアドレスを使用できますが、その場合、NNM iSPIsのシングルサインオンが無効になるため、次回NNM iSPIにアクセスするときにもう一度ログオンする必要があります。

NNMi管理サーバーの正式なFQDNを判別するには、以下のいずれかの方法を使用します。

- `nnmofficialfqdn.ovpl`コマンドを使用して、インストール中に設定した正式なFQDNの値を表示します。詳細については、nnmofficialfqdn.ovplのリファレンスページ、またはLinuxのマンページを参照してください。
- NNMiコンソールで、**[ヘルプ] > [システム情報]** をクリックします。[**サーバー**] タブで、正式なFQDNステートメントを特定します。

インストール中に設定された正式なFQDNを変更する必要がある場合は、`nnmsetofficialfqdn.ovpl`コマンドを使用します。詳細については、nnmsetofficialfqdn.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

注: インストール後、システムアカウントは有効なままになっています。システムアカウントは、コマンドラインのセキュリティと復旧の目的のみに使用します。

NNM iSPIsへのSSOには、ユーザーが正式なFQDNを含むURLでNNMiコンソールにアクセスすることが要求されます。IPアドレスや短縮されたドメイン名など、正式ではないドメイン名を使用してNNMiコンソールにアクセスした場合にNNMi URLを正式なFQDNにリダイレクトするようにNNMiを設定できます。URLをリダイレクトするようにNNMiを設定する前に、該当する正式なFQDNが設定されている必要があります。詳細については、NNMiヘルプを参照してください。

NNMiでURLへのリダイレクトを可能にした後、以下の点に注意してください。

- アクセスするNNMi管理サーバーに適したホスト名を使用して、NNMiコンソールにログオンできます。たとえば、ユーザーがhttp://localhost/nnmを要求している場合、NNMiは http://host.mydomain.com/nnmなどのURLにそれをリダイレクトします。
- http://host.mydomain.com/nnmを使用してNNMiコンソールにアクセスできない場合、以下のURLを使用して、NNMiコンソールに直接アクセスしてください。

<protocol>://<fully_qualified_domain_name>:<port_number>launch?cmd=showMain
<protocol>はhttpまたはhttpsです。

<fully_qualified_domain_name>は、NNMi管理サーバーの正式な完全修飾ドメイン名 (FQDN) です。

<port_number>は、NNMiコンソールに接続するためのポートです。これは、NNMiのインストール時に割り当てられ、以下のファイルで指定されます。

- Windowsの場合: `%NnmDataDir%\conf\nnm\props\nms-local.properties`
- Linuxの場合: `$NnmDataDir/conf/nnm/props/nms-local.properties`

SSOの無効化

SSOを無効にする必要がある場合は、以下の手順を実行します。

注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてから`ovstop`コマンドおよび`ovstart`コマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

1. 以下のファイルを開きます。
   - Windowsの場合: `%NNM_PROPS%\nms-ui.properties`
   - Linuxの場合: `$NNM_PROPS/nms-ui.properties`
2. ファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.isEnabled = true

   `isEnabled`プロパティを`false`に変更します。

   com.hp.nms.ui.sso.isEnabled = false
3. 以下のコマンドを実行し、変更をコミットします。

   nnmsso.ovpl -reload

   詳細については、nnmsso.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

SSOセキュリティに関する注意

1. SSOセキュリティの `initString`パラメーターは、以下のように使用されます。

   SSOは、対象鍵暗号方式を使用してSSOトークンの検証と作成を行います。設定内の`initString`パラメーターは、秘密鍵の初期化に使用されます。アプリケーションはトークンを作成し、`initString`パラメーターを使用する各アプリケーションはそのトークンを検証します。

注: 以下は、非常に重要な情報です。

- initStringパラメーターを設定せずに、SSOを使用することはできません。
- initStringパラメーターは機密情報であり、公開、移動、永続性において、機密情報として取り扱う必要があります。
- 相互に統合するアプリケーションは、SSOを使用してinitStringを共有できます。
- initStringは最低12文字の長さです。

2. 特に必要でない限り、SSOを無効にします。
3. 最も弱い認証フレームワークを使用するアプリケーションやほかの統合アプリケーションに信頼されるSSOトークンを発行するアプリケーションは、すべてのアプリケーションの認証セキュリティレベルを判断します。

   HPは、強力で安全な認証フレームワークを使用するアプリケーションのみがSSOトークンを発行するように設定することを推奨します。
4. 対称暗号化による影響について

   SSOは、SSOトークンの発行と検証に対象鍵暗号方式を使用します。そのため、SSOを使用するアプリケーションは、同一のinitStringを共有しているその他のすべてのアプリケーションによって信頼されるトークンを発行できます。

   initStringを共有するアプリケーションが信頼されない場所にある、または信頼できない場所にアクセスできる場合に、この潜在的なリスクが浮上します。
5. ユーザーロール

   SSOでは、統合されたアプリケーション間でユーザーロールは共有されません。このため、統合されたアプリケーションはユーザーロールを監視する必要があります。HPは、すべての統合アプリケーションで、同一のユーザーレジストリ (LDAP/ADとして) を共有することを推奨します。

   ユーザーロールを管理できないと、セキュリティ違反やアプリケーションエラーが発生する場合があります。たとえば、統合アプリケーションで異なるロールに同じユーザー名が割り当てられることがあります。

   ユーザーがアプリケーションAにログオンし、コンテナーやアプリケーション認証を使用するアプリケーションBにアクセスするとします。ユーザーロールを管理できないと、そのユーザーはアプリケーションBに手動でログオンし、ユーザー名を入力しなければならなくなります。このとき、ユーザーがアプリケーションAにログオンしたときとは異なるユーザー名を入力すると、その後にアプリケーションAまたはBから3つ目のアプリケーション (アプリケーションC) にアクセスした場合にアプリケーションAまたはBに使用したユーザー名でアプリケーションCにアクセスするという予期しない動作が発生することになります。
6. 認証にIdentity Managerが使用される

   Identity Manager内の保護されていないすべてのリソースは、SSO設定に非セキュアーURL設定として設定されている必要があります。
7. SSOデモモード:

- デモの目的のみにSSOデモモードを使用します。
- セキュアーでないネットワークでのみデモモードを使用します。
- デモモードを本番に使用しないでください。デモモードと本番モードを混ぜて使用しないでください。

公開キーインフラストラクチャーユーザー認証をサポートするためのNNMiの設定

NNMiでは、公開キーインフラストラクチャー (PKI) を使用したユーザー認証がサポートされています。このため、ユーザーがパスワードを使用せずにX.509クライアント証明書でNNMiにログオンする必要があります。この章では、NNMiユーザーアカウントに証明書をマップするようにNNMiを設定する方法 (PKIユーザー認証を使用する方法) について説明します。

注: PKIユーザー認証には、Common Access Card (CAC) やPersonal Identity Verification (PIV) カードなどのスマートカードのサポートが含まれます。

PKIユーザー認証を使用できるようにNNMiを設定すると、NNMiユーザーはNNMiへのログインでNNMi固有のユーザー名とパスワードを使用する必要がありません。

この方法を使用して、NNMiは、ユーザー名を取得するためにPKI証明書を読み込みます。NNMiユーザーロールを取得するには、NNMi内でユーザーのロールを定義するか、ライトウェイトディレクトリアクセスプロトコル (LDAP) を使用するようにNNMiを設定する必要があります。

注: PKIユーザー認証ではHTTPSプロトコルが使用されます。

注: PKIユーザー認証は、ライトウェイトシングルサインオン (LW-SSO) 機能に置き換わるものです。そのため、両方を使用することはできません。詳細については、「SSOの無効化」(338ページ)を参照してください。

この章には、以下のトピックがあります。

「PKIユーザー認証のためのCLI認証の設定」(360ページ)

「PKIユーザー認証の問題のトラブルシューティング」(362ページ)

ユーザー認証方針

NNMiには、NNMiユーザーアクセス情報の定義および保存先としていくつかのオプションが用意されています。

以下の表にPKIユーザー認証で使用できるオプションを示します。

ユーザー認証方針

項目	ユーザー認証の方法	NNMiのユーザーアカウント定義	NNMiのユーザーグループ定義	グループメンバーシップの方法
混合	X.509証明書	はい	はい	NNMiユーザーアカウントのマッピング
外部	X.509証明書	いいえ	はい	LDAP

[混合] オプションでは、NNMiがユーザーグループ割り当ての定義と保存を行います。NNMiのすべてのユーザー情報を設定する方法の詳細については、NNMiヘルプの「**ユーザーアカウントの設定 (ユーザーアカウントフォーム)**」Configuring User Accounts (User Account Form) を参照してください。

[外部] オプションでは、NNMiはライトウェイトディレクトリアクセスプロトコル (LDAP) のユーザーグループ割り当てを使用します。詳細については、「NNMiとLDAPによるディレクトリサービスの統合」(375ページ)を参照してください。

PKIユーザー認証のためのNNMiの設定 (X.509証明書認証)

PKIユーザー認証のためにNNMiを設定するには、ユーザーアカウント名が、証明書に含まれるユーザー名に一致する必要があります。以下のいずれかの方法を使用してロールを設定します。

- LDAPを使用するには、「NNMiとLDAPによるディレクトリサービスの統合」(375ページ)を参照してください。
- ユーザーアカウントの追加にNNMiコンソールを使用するには、[**ユーザーアカウント**] フォームで [**ディレクトリサービスアカウント**] チェックボックスをオンにし、[**パスワード**] フィールドは空白のままにします。次に、前のマッピングルールに一致するユーザーアカウント名を使用します。

NNMiの場合は、次のファイルでPKIユーザー認証を有効化してカスタマイズします。

- Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`
- Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`

NNMiを有効化して、PKIユーザー認証 (X.509証明書認証ともいう) を必須とするには、以下の手順を実行します。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`
   - Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`
2. 以下のテキストブロックを探します。

   ```
   <realm name="console">
   <mode>FORM</mode>
   </realm>
   ```
3. 見つかった行を以下のように編集します。

   ```
   <realm name="console">
   <mode>X509</mode>
   </realm>
   ```
4. 以下のテキストブロックを探します。

   ```
   <principalMapping>
   ```
5. `<principalMapping>`セクションで項目を編集して、プリンシパルを抽出する (マップする) ようにNNMiを設定します。この手順を実行するには、証明書の形式を知る必要があります。

   注: NNMiではプリンシパルを抽出するためのいくつかのオプションをサポートしています。これらのオプションは、任意の順序および任意の数で指定できます。

   - 属性エレメントは、`SubjectDN`から`EMAILADDRESS`などのフィールドを抽出します。
     - LDAPを使用している場合、抽出された名前は、LDAPに設定されている名前に一致する必要があります。詳細については、「NNMiとLDAPによるディレクトリサービスの統合」を参照してください。
     - 内部アカウントを使用する場合、名前がNNMiユーザーアカウント名に一致する必要があります。アカウントがPKIユーザー認証のみに使用される場合、パスワードなしで「ディレクトリサービスアカウント」として作成される必要があります (NNMi [**ユーザーアカウント**] フォームを使用。[**ディレクトリサービスアカウント**] チェックボックスをオンにして、[**パスワード**] フィールドを空白のままにします)。アカウントがPKIユーザー認証とパスワードログオンの両方に使用される場合、パスワード付きの標準アカウントとして作成される必要があります。
   - `regexp`エレメントは、`SubjectDN`全体に対して正規表現を実行します。
   - `subjectAlternativeName` (SAN) エレメントは、タイプ`rfc822Name` (電子メールアドレス) と一緒に使用できます。

- タイプotherNameおよび追加のoid属性のsubjectAlternativeNameエレメント。このオプションは、通常、[Microsoft ユニバーサル プリンシパル名 (UPN)] フィールドに使用されます。

nms-auth-config.xmlファイルの<principalMapping>セクションで提供される例に加えて、以下の例を参照してください。

例1:[EMAILADDRESS] フィールドを使用するには、これらの行を以下のように編集します。

<!-- The attribute element extracts a field from the SubjectDN;

for example, EMAILADDRESS, CN, or UID.-->

<attribute>EMAILADDRESS</attribute>

例2:[EMAILADDRESS] フィールドの一部のみを抽出するには、フィールドの一部を抽出するためのより複雑な正規表現を使用する例として、以下の行を編集します。[EMAILADDRESS] フィールドの名前の部分のみを抽出するには、以下の正規表現式を使用します。

<!-- Extract the name part of the email field which appears first

in the subjectDN.If the subject is EMAILADDRESS=first.last@example.com,

CN=First Last, OU=MyGroup, O=My Company, the mapped username would be

"first.last"--> <regexp group="1">EMAILADDRESS=([^@]+).*</regexp>

例3:文字列の中間のフィールドを照合するためのより複雑な正規表現を使用する例として、以下の行を編集します。

<!--Extract the CN field which appears anywhere in the subjectDN.

Note the optional group before the CN which matches the

previous fields.If the subject is EMAILADDRESS=first.last@example.com,

CN=First Last, OU=MyGroup, O=My Company

nms-auth-config.xmlファイルの<principalMapping>セクションで提供される例に加えて、以下の例を参照してください。

例1:[EMAILADDRESS] フィールドを使用するには、これらの行を以下のように編集します。

<!-- The attribute element extracts a field from the SubjectDN; for example,

EMAILADDRESS, CN, or UID.-->

<attribute>EMAILADDRESS</attribute>

例2:[EMAILADDRESS] フィールドの一部のみを抽出するには、フィールドの一部を抽出するためのより複雑な正規表現を使用する例として、以下の行を編集します。[EMAILADDRESS] フィールドの名前の部分のみを抽出するには、以下の正規表現式を使用します。

<!-- Extract the name part of the email field which appears first in

the subjectDN.If the subject is EMAILADDRESS=first.last@example.com,

CN=First Last, OU=MyGroup, O=My Company, the mapped username would be

"first.last"-->

<regexp group="1">EMAILADDRESS=([^@]+).*</regexp>

例3:文字列の中間のフィールドを照合するためのより複雑な正規表現を使用する例として、以下の行を編集します。

<!--Extract the CN field which appears anywhere in the subjectDN.

Note the optional group before the CN which matches the previous fields.

If the subject is EMAILADDRESS=first.last@example.com, CN=First Last,

OU=MyGroup, O=My Company

Then the mapped username would be "First Last" -->

<regexp group="2">(.*,)?CN=([^,]+).*</regexp>

例4:[サブジェクトの別名] フィールドから電子メールアドレスを抽出するには、これらの行を以下のように編集します。

<!-- Extract the first match of type rfc822Name from the Subject

Alternative Name field of the certificate.-->

<subjectAlternativeName type="rfc822Name" />

例5:[サブジェクトの別名] から特定のOIDを抽出するには、これらの行を以下のように編集します。

<!-- Extract the first match of type otherName with the supplied

OID from the Subject Alternative Name field of the certificate.-->

<subjectAlternativeName type="otherName" oid="1.3.6.1.4.1.311.20.2.3" />

注: デバッグログを有効にするログコマンドは以下のとおりです。

nnmsetlogginglevel.ovpl

com.hp.ov.nms.as.server.auth.x509.NmsCertMapper FINEST

6. 変更を保存します。
7. 信頼されたCA証明書をトラストストアにすでにインストールしている場合、`nms-auth-config.xml`ファイルへの変更を即時に有効にするために、以下のスクリプトを実行します。

   nnmsecurity.ovpl -reloadAuthConfig

   証明書をまだインストールしていない場合は、以下の手順に進みます。
8. NNMi管理サーバーで、`nnm.truststore`ファイルが存在するディレクトリに変更します。

   Windowsの場合: `%NnmDataDir%\shared\nnm\certificates`

   Linuxの場合: `$NnmDataDir/shared/nnm/certificates`
9. 信頼済みCA証明書を`nnm.truststore`ファイルにインポートします。使用する必要がある証明書が`example_ca.cer`ファイルに含まれているとします。以下のコマンドを実行して、CA証明書を

NNMi nnm.truststoreファイルにインポートします。

Windowsの場合:

%NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -import -alias myca -keystore nnm.truststore -file example_ca.cer

Linuxの場合:

$NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -import -alias myca -keystore nnm.truststore -file example_ca.cer

10. NNMiサービスを再起動します。
    a. NNMi管理サーバーで `ovstop`コマンドを実行します。
    b. NNMi管理サーバーで`ovstart`コマンドを実行します。

注: HA下でファイルを変更する場合、クラスター内の両方のノードで変更を行う必要があります。HA設定を使用したNNMiでは、この変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。

NNMiは現在、PKIユーザー認証を使用するために設定されています。NNMiへのログオンにパスワードを使用できなくなります。LDAPおよびNNMiユーザーアカウントが正しく動作していること、証明書およびアカウントが、ユーザーがNNMiにアクセスできるように正しく設定されていることを確認してください。

クライアント証明書を使用したNNMiへのログオン

クライアント証明書を使用してNNMiにログオンするには、以下の手順を実行します。

1. クライアント証明書にブラウザーからアクセスできることを確認します。
2. ブラウザーで`https://<hostname>/nnm`を指定します。
3. NNMiによりアクセスが許可され、NNMiまたはLDAPアカウント設定に基づいてユーザーロールが割り当てられます。

クライアント証明書を持つユーザのアクセスの廃止

ユーザーをNNMiにアクセスさせないようにするには、以下のいずれかを実行します。

- LDAPアカウントを使用してアクセスできるようにユーザーが設定されている場合、NNMiに関連付けられたLDAPグループからユーザーを削除する。
- NNMiユーザーアカウントを使用してアクセスできるようにユーザーが設定されている場合、ユーザーグループからユーザーを削除し、ユーザーアカウントを削除する。

どちらの場合も、ユーザーはNNMiコンソールにアクセスできなくなります。

グローバルネットワーク管理環境のPKIユーザー認証の特別な考慮事項

グローバルネットワーク管理設定でNNMiを使用する場合、グローバルネットワーク管理設定に含まれるすべてのNNMi管理サーバーにPKIユーザー認証を設定します。

証明書検証 (CRLおよびOCSP)

NNMiでは、証明書の取り消しを確認する方法として、以下の2つの方法がサポートされます。

- 証明書失効リスト (CRL) - CRLとは取り消された証明書のリストで、認証機関 (CA) からダウンロードします。
- Online Certificate Status Protocol (OCSP) - OCSPは、OCSPレスポンダーというオンラインサービスを使用して、1つの証明書の取り消しを対話式に確認するプロトコルです。

CRLおよびOCSPの検証では、証明書が取り消されたユーザーのアクセス拒否という同じ結果を、2つの異なる方法で得ることができます。ブラウザーは通常多様な認証機関 (CA) に対応しているため、Webブラウザーでは一般的にOCSPが優れているとみなされますが、1つのWebサイトの確認のためにCRL全体をダウンロードしなければならないのは非効率です。

多くのクライアントを扱うことが多いサーバーでは、同じCAから取得した証明書のみを使用する場合、すべての接続に対してOCSPを確認する必要はなく、ダウンロードが1日に1回で済むCRLの確認のほうがはるかに効率的なことがあります。

OCSPおよびCRLの両方が有効にされると、NNMiはデフォルトでCRLを最初に照会します。CRLは通常寿命がはるかに長く、ネットワークの停止に対する回復力が強いため、CRLの確認が最初に実行されます。ネットワークまたはOCSPレスポンダーが停止すると、OCSPが頻繁に要求を実行し、ユーザーがログオンできなくなります。NNMiは、ネットワークまたはOCSPレスポンダーが停止した場合、動作の続行に使用するために有効なCRLを取得しようとします。

また、CRLの比較はOCSPよりも高速です。つまり、証明書をディスク上のリストと照合するほうが、各証明書を検証するためにネットワークを介して別のサーバーを照会するよりも速くなります。そのため、証明書が信頼済みエンティティで署名され、期限が切れていない場合、CRLは、その証明書が取り消されたかどうかを確認するために照会されます。取り消された場合、OCSPを確認する必要はありません。ただし、CRLを確認した後に証明書がまだ有効な場合、証明書が最近取り消されていないこと (および、証明書をリストしている更新済みのCRLがまだ使用可能でないこと) を確認するために、OCSPも照会されます。

OCSPおよびCRLの両方が有効になると、NNMiは以下をサポートします。

- NNMiが最初にCRLを照会し、OCSPが続きます (これはデフォルトの動作です)。
- CRLが使用可能でない場合、OCSPがバックアップとして使用されます。
- OCSPが使用可能でない場合、CRLがバックアップとして使用されます。

証明書検証プロトコルの一般設定

取り消された証明書をNNMiで確認する方法を設定できます。たとえば、使用するプロトコルの順序や、すべてのプロトコルを使用するかどうかを設定できます。

NNMiでは、nms-auth-config.xmlファイルを使用して、このような設定を行います。

プロトコルの順序の設定

デフォルトのNNMiでは、CRLチェックの後でOCSPチェックが実行されます。

取り消された証明書に対して行う証明書検証プロトコルチェックの順序を設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml
2. ファイルの<revocation>セクション (<revocation>タグを検索) で、以下のテキストで開始する行を探します。

   <ordering>
3. 以下のいずれかを行います。
   - CRLチェックの後でOCSPチェックを行うように指定するには、行を以下のように変更します。

     <ordering>CRL OCSP</ordering>
   - OCSPチェックの後でCRLチェックを行うように指定するには、行を以下のように変更します。

     <ordering>OCSP CRL</ordering>
4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

プロトコル要求の設定

プロトコル要求に関して、以下のいずれかを実行するようにNNMiを設定できます。

- 証明書ごとにすべての証明書検証プロトコルを確認する
- 優先される順序でプロトコルリストを確認し、有効な応答が受信された時点で停止する

プロトコル要求を設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`
2. ファイルの`<revocation>`セクション (`<revocation>`タグを検索) で、以下のテキストで開始する行を探します。

   `<mode>`
3. 以下のいずれかを行います。
   - 証明書ごとにすべてのプロトコルをNNMiで確認するには、行を以下のように変更します。

     `<mode>CHECK_ALL</mode>`
   - 優先される順序でプロトコルリストをNNMiで確認し、有効な応答が受信された時点で停止するには、行を以下のように変更します。

     `<mode>FIRST_SUCCESS</mode>`
4. `nms-auth-config.xml`ファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

CRLを使用した証明書の検証

NNMiでは、信頼されなくなった証明書によるクライアントへのアクセスを、CRLを使用して適切に拒否します。

注: 認証中にCRLに証明書のシリアル番号が見つかると、NNMiはその証明書を受け入れず、認証に失敗します。

X.509認証モードを使用している場合、NNMiではデフォルトでCRLがチェックされますが、以下のセクションの説明に従って`nms-auth-config.xml`ファイルを編集することでCRLを指定できます。

注: NNMiでは、以下の場所にCRL設定が保存されます。

- Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`
- Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`

新しい使用可能なオプションを参照するために使用できる設定ファイルのデフォルトバージョンもあります。デフォルト設定ファイルは、以下の場所に保存されています。

- Windowsの場合: `%NnmInstallDir%\newconfig\HPOvNnmAS\nmsas\conf\nms-auth-config.xml`

- **Linuxの場合:** $NnmInstallDir/newconfig/HPOvNnmAS/nmsas/conf/nms-auth-config.xml

CRLチェックの有効化および無効化

デフォルトでは、NNMiでCRLチェックが有効になっています。

CRLチェックを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml
2. ファイルの<crl>セクション (<crl>タグを検索) で、以下のテキストで開始する行を探します。

   <enabled>
3. 以下のいずれかを行います。
   - CRLチェックを有効にするには、行を以下のように変更します。

     <enabled>true</enabled>
   - CRLチェックを無効にするには、行を以下のように変更します。

     <enabled>false</enabled>
4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

CRL強制モードの変更

デフォルトでは、NNMiはCRLを強制するように設定されています。

製品のCRL強制を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml
2. ファイルの<crl>セクション (<crl>タグを検索) で、以下のテキストで開始する行を探します。

   <mode>
3. 行を以下のように編集します。

   <mode><value></mode>

<value>は以下のいずれかです。

- ENFORCE:証明書で指定されたCRLを強制する
- ATTEMPT: CRLを確認するが、CRLが使用可能でない場合はアクセスを許可する
- REQUIRE: 証明書でCRLを必須にし、強制する

注: REQUIREモードの場合、CRLが指定されていないか、ユーザーの証明書で使用できない場合は、認証に失敗します。

4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

CRLの更新頻度の変更

NNMiでCRLを更新する頻度を設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml
2. ファイルの<crl>セクション (<crl>タグを検索) で、以下のテキストで開始する行を探します。

   <refreshPeriod>
3. 行を以下のように編集します。

   <refreshPeriod><value></refreshPeriod>

   <value>は、時間または日数を表す整数です (最小値は1h)。

   たとえば、24時間の場合は「24h」と入力し、2日の場合は「2d」と入力します。
4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

CRLの最大アイドル時間の変更

CRLがアイドル状態になって (使用またはアクセスされなくなって) からNNMiでCRLを保持する期間を設定できます。

CRLの最大アイドル時間を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml

2. ファイルの<crl>セクション (<crl>タグを検索) で、以下のテキストで開始する行を探します。

   <maxIdleTime>

3. 行を以下のように編集します。

   <maxIdleTime><value></maxIdleTime>

   <value>は、時間または日数を表す整数です (最小値は1h)。

   たとえば、24時間の場合は「24h」と入力し、2日の場合は「2d」と入力します。

4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

CRLの有効期限の警告

CRLチェックを有効にすると、CRLの有効期限が切れた場合にNNMiコンソールからユーザーがロックアウトされる可能性があります。不要なロックアウトを避けるために、NNMiでは、稼働状態の警告メッセージが表示され、CRLの有効期限が切れたこと、またはまもなく有効期限が切れることを管理者に警告します。

CRLの有効期限が切れたことを示す警告 (重要警戒域の重大度) は、1つ以上のCRLの有効期限が切れたときに発生します。

CRLの有効期限が切れることを示す警告 (警戒域の重大度) は、1つ以上のCRLの残り有効期間が1/6未満になったときに発生します。たとえば、CRLの有効期間が24時間の場合、CRLの残り有効期間が4時間未満になると、NNMiに警告が表示されます。

CRLが常に最新の状態となるようにリフレッシュ期間を設定します。リフレッシュ期間を適切に設定すると、CRLサーバーが当分使用できない場合でも、ダウンロードしたCRLの残り有効期間が不足しないようになります。これにより、CRLサーバーが使用できるようになるまでNNMiで通常の操作ができるようになります。この例では、8時間のリフレッシュ期間が適切だと考えられます。

CRLの場所の変更

デフォルトのNNMiでは、証明書に組み込まれたHTTPの場所からCRLがダウンロードされます。この場所にNNMi管理サーバーからアクセスできない場合、管理者は必要なCRLをほかの方法で取得し、このCRLをローカルファイルシステムからロードするようにNNMiを設定できます。

注: 証明書の発行者が署名したCRLのみが、証明書の評価時に考慮されます。

CRLをローカルファイルシステムからロードするようにNNMiを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml

2. ファイルの <crl> セクション (<crl> タグを検索) で、以下を含むテキストブロックを探します。

   <!--

   CRLの場所の指定 (省略可能)。設定されている場合、NNMiは、このCRLと同じCAが発行したすべての証明書が次の場所にあるものとして処理します。複数のエントリが含まれる場合があります。 <location>file:///var/opt/OV/shared/nnm/certificates/myco.crl</location>

   -->

3. -->タグの後に行を挿入しオペレーティングシステムに基づいて以下のように入力します。

   Windowsの場合: `<location>file:///C:/CRLS/<crlname>.crl</location>`

   Linuxの場合: `<location>file:///var/opt/OV/shared/nnm/certificates/<crlname>.crl</location>`

4. `nms-auth-config.xml`ファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

Online Certificate Status Protocol (OCSP) を使用した証明書の検証

NNMiでは、Online Certificate Status Protocol (OCSP) を使用して、取り消された証明書を対話形式で確認できます。

PKIユーザー認証では、OCSPを使用して証明書の取り消しステータスを確認します。これは、OCSPレスポンダーにクエリーすることで行われます。OCSPレスポンダーは、以下のように特定の証明書に関する取り消し情報を迅速かつ正確に提供します。

- OCSPクライアントは、証明書ステータス要求をOCSPレスポンダーに送信します。
- OCSPクライアントは、OCSPレスポンダーからデジタル署名付きの応答が提供されるまで問題のある証明書の受け入れを停止します。
- OCSPレスポンダーは、以下のいずれかの値を返して証明書のステータスを示します。
  - Good (成功: ユーザーのアクセスは許可されます)
  - Revoked (失敗: ユーザーのアクセスは拒否されます)
  - Unknown (失敗: ユーザーのアクセスは拒否されます)

OCSPレスポンダーは証明書ごとにクエリーされますが、CRLは定期的 (1日に1回など) にダウンロードされるため、OCSP応答が対応するCRLよりも新しくなる場合があります。

注: NNMiでは、以下の場所にOCSP設定が保存されます。

- Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`
- Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`

設定ファイルのデフォルトバージョンは、新しい利用可能なオプションを参照するために使用できます。デフォルト設定ファイルは、以下の場所に保存されています。

- Windowsの場合: `%NnmInstallDir%\newconfig\HPOvNnmAS\nmsas\conf\nms-auth-config.xml`
- Linuxの場合: `$NnmInstallDir/newconfig/HPOvNnmAS/nmsas/conf/nms-auth-config.xml`

OCSPチェックの有効化および無効化

OCSPチェックを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`
2. ファイルの`<ocsp>`セクション (`<ocsp>`タグを検索) で、以下のテキストで開始する行を探します。

   `<enabled>`
3. 以下のいずれかを行います。
   - OCSPチェックを有効にするには、行を以下のように変更します。

     `<enabled>true</enabled>`
   - OCSPチェックを無効にするには、行を以下のように変更します。

     `<enabled>false</enabled>`
4. `nms-auth-config.xml`ファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

OCSP強制モードの変更

デフォルトでは、NNMiはOCSPを強制するように設定されています。

製品のOCSP強制を変更するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`
2. ファイルの `<ocsp>` セクション (`<ocsp>` タグを検索) で、以下のテキストで開始する行を探します。

   `<mode>`

3. 行を以下のように編集します。

   <mode><value></mode>

   <value>は以下のいずれかです。

   - ENFORCE:証明書で指定されたOCSPを強制する
   - ATTEMPT: OCSPを確認するが、OCSPが使用可能でない場合はアクセスを許可する
   - REQUIRE: 証明書でOCSPを必須にし、強制する

4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

nonceの有効化

(再生攻撃を回避するための) セキュリティ強化として、OCSPリクエスターは、証明書の検証要求にnonceを追加できます。nonceは、各要求に添付されるランダム番号で、暗号化を変更します。nonce機能を有効にすると、OCSPレスポンダーは、nonce値を使用して適切な応答を計算します。

注: nonceを使用すると、応答を事前計算またはキャッシュできないため、OCSPレスポンダーの負荷が増えます。一部のOCSPレスポンダーは、nonceを含む要求を受け入れない可能性があります。

注: デフォルトでは、nonce機能が無効になっています。

OCSPのnonce機能を有効にするには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml

2. ファイルの<ocsp>セクション (<ocsp>タグを検索) で、以下のテキストで開始する行を探します。

   <nonce>

3. 以下のいずれかを行います。
   - nonce機能を有効にするには、行を以下のように変更します。

     <nonce>true</nonce>

   - nonce機能を無効にして一般的な要求を使用するには、行を以下のように変更します。

     <nonce>false</nonce>

4. nms-auth-config.xmlファイルを保存します。

5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

OCSPレスポンダーのURLの指定

必要に応じて、以下のようにOCSPレスポンダーのURLを指定できます。

1. 以下のファイルを編集します。

   Windowsの場合: %NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml

   Linuxの場合: $NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml
2. ファイルの<ocsp>セクション (<ocsp>タグを検索) で、以下のテキストで開始する行を探します。

   <responder>
3. この行を以下のように編集します。

   <responder><URL></responder>

   ここで、<URL>はOCSPレスポンダーに関連付けられたURLです。
4. nms-auth-config.xmlファイルを保存します。
5. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

注: OCSP URLでは、HTTPプロトコルを使用する必要があります。

- nms-auth-config.xmlファイルでOCSP URLが指定されていない場合、NNMiは証明書自体からOCSPレスポンダーを取得しようとします。
- 証明書でOCSPレスポンダーが指定されていない場合、NNMiは<mode>設定を使用して対処法を決定します。
  - モードがENFORCEまたはATTEMPTである場合、NNMiはこの証明書のOCSP検証手順を渡します。
  - モードがREQUIREである場合、NNMiは証明書を拒否します。

NNMiログオンアクセスに使用される証明書を制限するNNMiの設定

PKIユーザー認証を使用するNNMiを使用している場合は、NNMiログオンアクセスで有効とみなされる証明書を制限できます。

NNMiでは、以下のタイプの制限がサポートされます。

- 証明書のキーの拡張使用に関する制限。これは、ハードウェアベースの証明書またはほかの特定の証明書へのNNMiアクセスを制限するために使用できます。

- 証明書の発行者に関する制限。これは、ログオン以外の目的でロードされた信頼される証明書が、ログオン証明書の作成に使用されるのを防ぐための制限です。

ログオンアクセスに使用される証明書を制限するようにNNMiを設定するには、以下の手順を実行します。

1. 以下のファイルを編集します。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`
2. 以下を含むテキストブロックを探します。

   <certificateConstraints>
3. 以下の例を参考にして、ログオンに使用される証明書を制限するようにNNMiを設定します (値は適宜置き換えてください)。

   例1:クライアント認証を必須にするには、以下のセクションを編集します。

   <!-- client authentication -->

   <extKeyUsage>1.3.6.1.5.5.7.3.2</extKeyUsage>

   例2:Microsoft スマートカードを使用してユーザーがログオンする必要があるようにするには、以下のセクションを編集します。

   <!-- Microsoft smart card logon -->

   <extKeyUsage>1.3.6.1.4.1.311.20.2.2</extKeyUsage>

   例3:特定のCAが署名した証明書のみを受け入れるようにするには、以下のセクションを編集します。

   <!-- Configures one or more trusted issuers.If this is configured, client certificates must be issued by one of these issuers to be used for client authentication -->

   <trustedIssuer>CN=MyIssuer, OU=MyOrgUnit, O=MyOrg, ST=CO, C=US</trustedIssuer>

   注: 複数のextKeyUsageエントリが指定されている場合は、証明書にすべて含める必要があります (ブール式AND)。複数のtrustIssuerエントリが指定されている場合は、1つのみを証明書の信頼される発行者にする必要があります (ブール式OR)。
4. 以下のコマンドを実行して、変更内容を有効にします。

   nnmsecurity.ovpl -reloadAuthConfig

例:スマートカードログオンを必要とするNNMiの設定

以下の例に、スマートカードログオンを必須にするため、PKIユーザー認証を使用するようにNNMiを設定する方法を示します。

注: この例では、ユーザー認証の混合方法を使用します。

この例では、以下を想定しています。

- NNMiにログオンするために、組織でスマートカードを使用している。
- スマートカードに、[サブジェクトの別名] フィールドが電子メールアドレスになっている証明書が含まれている。
- すべての証明書の取り消しを確認するために、組織でCRLを使用している。

この設定例を実行するには、以下の手順を実行します。

1. NNMiコンソールで、ゲスト権限を使用して`myusername@example.com`というユーザーを作成します。
   a. [ユーザーアカウント] ビューで、`myusername@example.com`ユーザーを作成します。

      ヒント: [**ユーザーアカウント**] フォームで、[**ディレクトリサービスアカウント**] チェックボックスをオンにし、[**パスワード**] フィールドは空白のままにします。詳細については、NNMiヘルプを参照してください。

   b. [ユーザーアカウントのマッピング] ビューで、新しいユーザーアカウントマッピングを作成し、`myusername@example.com`ユーザーを`NNMi Guest Users`ユーザーグループに割り当てます。
2. 以下のファイルを編集します。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-auth-config.xml`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-auth-config.xml`
3. 以下のテキストブロックを探します。

   ```
   <realm name="console">
   <mode>FORM</mode>
   </realm>
   ```
4. X.509証明書の認証を有効にするには、テキストを以下のように編集します。

   ```
   <realm name="console">
   <mode>X509</mode>
   </realm>
   ```
5. 以下のテキストブロックを探します。

   ```
   <principalMapping>
   ```
6. `<principalMapping>`ブロックで、証明書の [サブジェクトの別名] フィールドから最初に一致するタイプ`rfc822Name`を抽出するために、以下の行を含めます。

   ```
   <subjectAlternativeName type="rfc822Name" />
   ```
7. ファイルの`<crl>`セクション (`<crl>`タグを検索) で、以下のテキストで開始する行を探します。

   ```
   <enabled>
   ```
8. CRLチェックを有効にするには、行を以下のように変更します。

<enabled>true</enabled>

9. ファイルの<crl>セクションで、以下のテキストを含むテキストブロックを探します。

   <mode>

10. CRLを必須にして強制するには、行を以下のように変更します。

    <mode>REQUIRE</mode>

11. 以下を含むテキストブロックを探します。

    <certificateConstraints>

12. クライアント認証を必須にするには、以下のセクションを編集します。

    <!-- client authentication -->

    <extKeyUsage>1.3.6.1.5.5.7.3.2</extKeyUsage>

13. ユーザーによるスマートカードログオンを必須にするには、以下の行を追加します。

    <!-- Microsoft smart card logon -->

    <extKeyUsage>1.3.6.1.4.1.311.20.2.2</extKeyUsage>

14. nms-auth-config.xmlファイルへの変更を保存します。
15. NNMi管理サーバーで、nnm.truststoreファイルが存在するディレクトリに変更します。

    Windowsの場合: %NnmDataDir%\shared\nnm\certificates

    Linuxの場合: $NnmDataDir/shared/nnm/certificates

16. 信頼済みCA証明書をnnm.truststoreファイルにインポートします。使用する必要がある証明書がexample_ca.cerファイルに含まれているとします。以下のコマンドを実行して、CA証明書をNNMi nnm.truststoreファイルにインポートします。

    Windowsの場合: %NnmInstallDir%\nonOV\jdk\hpsw\bin\keytool.exe -import -alias myca -keystore nnm.truststore -file example_ca.cer

    Linuxの場合: $NnmInstallDir/nonOV/jdk/hpsw/bin/keytool -import -alias myca -keystore nnm.truststore -file example_ca.cer

17. ユーザーアカウント名が、証明書に含まれるユーザー名 (myusername) と一致していることを確認します。
18. NNMiサービスを再起動します。
    - NNMi管理サーバーでovstopコマンドを実行します。
    - NNMi管理サーバーでovstartコマンドを実行します。

これで、スマートカードログオンを必要とするようにNNMiが設定されました。

この例で説明している変更を設定に加えた後、nms-auth-config.xmlは以下のようになります。

```
<methods>
  <X509>
    <principalMapping>
```

```
            <subjectAlternativeName type="rfc822Name" />
        </principalMapping>
        <certificateConstraints>
            <extKeyUsage>1.3.6.1.5.5.7.3.2</extKeyUsage>
            <extKeyUsage>1.3.6.1.4.1.311.20.2.2</extKeyUsage>
            <trustedIssuer>CN=MyIssuer, OU=MyOrgUnit, O=MyOrg, ST=CO, C=US</trustedIssuer>
        </certificateConstraints>
        <revocation>
          <ordering>CRL OCSP</ordering>
           <mode>CHECK_ALL</mode>
         </revocation>
         <crl>
           <enabled>true</enabled>
            <mode>REQUIRE</mode>
            <!-- refresh CRLs every 12 hours -->
            <refreshPeriod>12h</refreshPeriod>
             <!-- remove CRLs that have not been used for 36 hours -->
             <maxIdleTime>36h</maxIdleTime>
         </crl>
         <ocsp>
           <enabled>false</enabled>
           <mode>ENFORCE</mode>
           <nonce>false</nonce>
         </ocsp>
    </X509>
  </methods>
  <realms>
    <realm name="console">
    <mode>X509</mode>
     </realm>
  </realms>
```

PKIユーザー認証のためのCLI認証の設定

承認されたユーザーは、NNMiコンソールに移動することなく、NNMiコマンドラインインタフェース (CLI) を使用してNNMi設定を行うことができます。

公開キーインフラストラクチャー (PKI) ユーザー認証は、ユーザー認証を実行するクライアント側のオペレーティングシステムとWebブラウザー設定によって異なります。このため、CLIセッションはPKIユーザー認証を使用できません。これは、コマンドがWebブラウザー環境外で実行されるためです。CLI認証をルート以外のユーザーとして有効にするため、承認されたユーザーに以下のファイルの読み取りアクセス権を付与できます (ルートユーザーにはこのファイルの読み取りアクセス権がすでに付与されています)。

Windowsの場合: `%NnmDataDir%\nmsas\NNM\conf\nms-users.properties`

Linuxの場合: `$NnmDataDir/nmsas/NNM/conf/nms-users.properties`

このファイルには、NNMi「システム」ユーザー用の暗号化されたパスワードが含まれています。このファイルを読み取りできるユーザーは、「システム」ユーザーとしてCLIコマンドを呼び出すことができます。

注: 管理者グループのメンバーとしてログオンするWindowsユーザーは`nms-users.properties`ファイルへの読み取りアクセス権をすでに持っているため、管理者グループのメンバーであるWindowsユーザーには他の設定は必要ありません。セキュリティの設定の詳細については、NNMiヘルプを参照してください。

`nms-users.properties`ファイルへの読み取りアクセス権は、Linuxの通常の`chmod`コマンドを使用して設定できます。しかし、このファイルのアクセス制御を細かく設定するには、オペレーティングシステムベースのアクセス制御リスト (ACL) を使用することをお勧めします。詳細については、「非ルートユーザーがCLIコマンドを実行できるようにするためのACLの設定」(360ページ)を参照してください。

非ルートユーザーがCLIコマンドを実行できるようにするためのACLの設定

ACLコマンドは、オペレーティングシステム間や、同じオペレーティングシステムのファイルシステムタイプ間で大幅に異なります。また、オペレーティングシステムを設定してACLを有効にする必要がある場合もあります (例: Linuxの`/etc/fstab`に`,acl`エントリを追加する)。

このセクションの例では、ext3およびext4ファイルシステムでLinux (RHELとSuSE) のACLコマンドを使用します。別のファイルシステムタイプまたはオペレーティングシステムを使用している場合、詳細については、そのオペレーティングシステムのACLドキュメントを参照してください。

この例は、オペレーティングシステムユーザー`user1`に`nms-users.properties`ファイルの読み取り権限を与えます。

注: ACL権限を設定するときには、そのファイルの権限一式を指定してください。指定する権限

によって以前の権限は上書きされます。

権限の付与

1. 以下のコマンドを使用して、現在のACLをクエリーします。

   chacl –l nms-users.properties

   出力は以下のようなものになります。

   nms-users.properties [u::rw-,u:user2:r--,u:user3:r--,g::r--,m::r--,o::---]

2. 角括弧 ([ ]) で囲まれて出力されたリストに新しい権限 (,u:user1:r--) を追加し、次のコマンドを実行します。

   ```
   chacl <results from within square brackets in the ACL list>,u:user1:r-- nms-users.properties
   ```

注: ACLでは、ユーザーレベルの制御とグループレベルの制御またはその両方を行うことができます。また、nnmiadmなどのLinuxグループを作成して、nms-users.propertiesファイルへの読み取りアクセス権をそのグループに付与することもできます。次に、そのグループにLinuxユーザーを追加したり、そのグループからLinuxユーザーを削除したりして、nms-users.propertiesファイルへのアクセス権を付与または削除します。これにより、CLIコマンドに対する「system」ユーザーとしての認証が付与または削除されます。

注意: nmsprocユーザーやnmsgrpグループの権限が妨げられるような設定ミスがあるとNNMiの機能が停止する可能性があるため、ACLを設定するときは注意してください。

ACLの一覧表示

以下のコマンドを実行します。

chacl -l nms-users.properties

権限の削除

1. 以下のコマンドを使用して、現在のACLをクエリーします。

   chacl –l nms-users.properties

2. 削除するユーザーを特定して削除します (user1を削除する場合は「,u:user1:r--」と指 定)。
3. ACLリストの残りの部分をchaclコマンドに貼り付けます。

   chacl <list results minus user1> nms-users.properties

注: nms-users.propertiesファイルパス内の各ディレクトリはアクセス可能である必要があります。通常、これらのフォルダーの権限は厳しく制限されており、アクセスできません。このパスには以下のディレクトリが含まれています。

- $NnmDataDir/nmsas
- $NnmDataDir/nmsas/NNM
- $NnmDataDir/nmsas/NNM/conf
- $NnmDataDir/nmsas/NNM/conf/props

これらのフォルダーにもACLを使用できます。または、通常のLinux chmodを使用して「search」アクセス権 (実行ビット (0711モード)) を「other」 に付与することもできます。

注: nnmrestore.ovplコマンドを実行して、NNMiバックアップから復元すると、既存のACLは上書きされます。その場合、NNMiを復元した後で、このセクションですでに説明した、ユーザーをACLに追加する手順を実行し、ACLを手動で再作成して適用する必要があります。

注: アプリケーションフェイルオーバーまたは高可用性 (HA) 環境の場合、プライマリノードにログオンして適切なACLコマンドを実行し、セカンダリノードで同じプロセスを繰り返して、手動で両方のノードにACLを設定する必要があります。

注: グローバルネットワーク管理 (GNM) 環境では、個別の各ノードに異なるユーザーが含まれる独自のACLが設定されている可能性があります。たとえば、リージョナルマネージャーのCLIアクセス権のあるユーザーにグローバルマネージャーのCLIアクセス権がない場合があります。

PKIユーザー認証の問題のトラブルシューティング

PKIユーザー認証中にエラーが発生する場合があります。エラーと考えられる原因のリストについては、以下の表を参照してください。

PKIユーザー認証のエラーと考えられる原因

エラーメッセージ	考えられる原因
401 認証されていません	HTTPSではなくHTTPが使用されている。 詳細については、「リモートアクセスには暗号化を必須とするようにNNMiを設定する」(264ページ)を参照してください。
	ユーザーに証明書がない。 詳細については、「証明書の管理」(316ページ)を参照してください。
	nnm.truststoreのCAでユーザーの証明書が信頼されていない。 詳細については、「証明書の管理」(316ページ)を参照してください。
	ユーザーの証明書が期限切れになっているか、まだ有効になっていない。

PKIユーザー認証のエラーと考えられる原因 (続き)

エラーメッセージ	考えられる原因
	詳細については、「証明書の管理」(316ページ)を参照してください。
	ユーザーの証明書が取り消されているか、取り消しチェックに失敗している。 詳細については、「証明書の管理」(316ページ)を参照してください。
	ユーザーの証明書の制約チェックに失敗している。 詳細については、「NNMiログオンアクセスに使用される証明書を制限するNNMiの設定」(355ページ)を参照してください。
403 権限がありません	マッピングされたユーザー名がNNMiまたはLDAPディレクトリサービスに存在していない。 詳細については、「PKIユーザー認証のためのNNMiの設定 (X.509証明書認証)」(341ページ)を参照してください。
	証明書プリンシパルとユーザー名のマッピングが正しくない。 詳細については、「PKIユーザー認証のためのNNMiの設定 (X.509証明書認証)」(341ページ)を参照してください。
	NNMiコンソールへのアクセスを提供するユーザーグループにユーザーが含まれていない。 詳細については、NNMiヘルプの「**セキュリティの設定**」を参照してください。

注: トラブルシューティングする場合、問題を特定しやすくするために、HTTPアクセスを無効にしてログ記録をオンにします。

NNMiで使用するTelnetおよびSSHプロトコルを設定する

[アクション] > [Telnet...(クライアントから)] メニュー項目によって、選択したノードに対するtelnetコマンドが呼び出されます (NNMiコンソールを現在実行中のWebブラウザーから)。**[アクション] > [Secure Shell...(クライアントから)]** メニュー項目によって、選択したノードに対するsecure shell (SSH) コマンドが呼び出されます (NNMiコンソールを現在実行中のWebブラウザーから)。デフォルトでは、Microsoft Internet ExplorerとMozilla FirefoxのどちらでもtelnetコマンドやSSHコマンドは定義されていないため、どちらのメニュー項目を使用する場合でもエラーメッセージが生成されます。

telnet、SSH、または両方のプロトコルを各NNMiユーザーに設定して (システムごとに)、NNMiコンソールメニュー項目を変更できます。

この章には、以下のトピックがあります。

TelnetまたはSSHメニュー項目の無効化

導入環境のNNMiユーザーが、NNMiコンソールからtelnetまたはSSH接続する必要がない場合は、それぞれのメニュー項目を無効化してNNMiコンソールから削除できます。

NNMiコンソールのメニュー項目の無効化は、NNMi管理サーバー上でNNMiコンソールにログオンするすべてのユーザーに適用されます。[**Telnet**] または [**Secure Shell**] メニュー項目を無効にするには、以下の手順を実行します。

1. [**設定**] ワークスペースで [**ユーザーインタフェース**] を展開して、[**メニュー項目**] を選択します。
2. [**メニュー項目**] ビューで、[**Telnet...(クライアントから)**] 行または [**Secure Shell...(クライアントから)**] 行を選択して、 [**開く**] アイコンをクリックします。
3. [**メニュー項目**] フォームで、[**有効にする**] チェックボックスをオフにしてから、[**作成者**] フィールドを適切な値に設定します。

   作成者値を変更すると、このメニュー項目はNNMiをアップグレードしても無効化されたままです。
4. フォームを保存し、閉じます。

詳細については、NNMiヘルプの「アクションメニューの制御」を参照してください。

Windows上のブラウザーへのTelnetまたはSSHクライアントの設定

NNMiユーザーのWebブラウザーにオペレーティングシステム提供のtelnetコマンドを設定します。この手順は、NNMiユーザーが **[アクション] > [Telnet...(クライアントから)]** メニュー項目を実行する必要がある各コンピューターおよびWebブラウザーで実行する必要があります。

NNMiユーザーのWebブラウザーにサードパーティのsshコマンドを設定します。この手順は、NNMiユーザーが **[アクション] > [Secure Shell...(クライアントから)]** メニュー項目を実行する必要がある各コンピューターおよびWebブラウザーで実行する必要があります。

このセクションの手順を完了するには、コンピューターの管理権限が必要です。特定の手順は、ブラウザーおよびオペレーティングシステムのバージョン (32ビットまたは64ビット) によって異なります。

Internet Explorerのバージョンを確認するには、**[ヘルプ] > [Internet Explorerのバージョン情報]** をクリックします。バージョン情報にテキスト **[64ビット版]** が含まれない場合、このInternet Explorerは32ビットです。

Firefoxは32ビットバージョンでのみ使用可能です。

以下の表は、各ブラウザーとオペレーティングシステムの組み合わせで使用する手順を示したものです。

WindowsでのTelnetおよびSSH設定手順のマトリックス

Webブラウザー	Windowsオペレーティングシステムアーキテクチャー	適用手順
Internet Explorer 32ビット	32ビット	• 「Windowsオペレーティングシステム提供のTelnetクライアント」(366ページ) • 「サードパーティTelnetクライアント (標準Windows)」(368ページ) • 「サードパーティSSHクライアント (標準WindowsおよびWindows上のウィンドウ)」(370ページ)
	64ビットWindows 7	• 「サードパーティTelnetクライアント (標準Windows)」(368ページ) • 「サードパーティSSHクライアント (標準WindowsおよびWindows上のウィンドウ)」(370ページ)
	64ビットWindows 7以外	• 「サードパーティTelnetクライアント (Windows上のウィンドウ)」(369ページ) • 「サードパーティSSHクライアント (標準WindowsおよびWindows上のウィンドウ)」(370ページ)
Internet Explorer 64ビット	64ビット	• 「Windowsオペレーティングシステム提供のTelnetクライアント」(366ページ) • 「サードパーティTelnetクライアント (標準Windows)」(368ページ) • 「サードパーティSSHクライアント (標準WindowsおよびWindows上のウィンドウ)」(370ページ)
Firefox	32ビット	• 「Windowsオペレーティングシステム提供のTelnetクライアント」(366ページ) • 「サードパーティTelnetクライアント (標準

WindowsでのTelnetおよびSSH設定手順のマトリックス(続き)

Webブラウザー	Windowsオペレーティングシステムアーキテクチャー	適用手順
		Windows)」(368ページ) • 「サードパーティSSHクライアント(標準WindowsおよびWindows上のウィンドウ)」(370ページ)
	64ビットWindows 7	• 「サードパーティTelnetクライアント(標準Windows)」(368ページ) • 「サードパーティSSHクライアント(標準WindowsおよびWindows上のウィンドウ)」(370ページ)
	64ビットWindows 7以外	• 「サードパーティTelnetクライアント(Windows上のウィンドウ)」(369ページ) • 「サードパーティSSHクライアント(標準WindowsおよびWindows上のウィンドウ)」(370ページ)

ヒント: このセクションのタスクの多くではWindowsレジストリの編集が必要です。レジストリを直接編集せずにシステム上で各ユーザーが実行できる.regファイルを作成できます。.regファイルの例は、「Windowsレジストリを変更するファイル例」(374ページ)を参照してください。

このセクションで説明するタスクの詳細については、以下のMicrosoftの記事を参照してください。

- Microsoft提供のtelnetクライアントをインストールする
  http://technet.microsoft.com/en-us/library/cc771275%28WS.10%29.aspx
- Windowsレジストリの概要
  http://support.microsoft.com/kb/256986
- Windowsレジストリをバックアップおよび復元する
  http://support.microsoft.com/kb/322756

Windowsオペレーティングシステム提供のTelnetクライアント

この手順は、以下の場合に適用されます。

- 32ビットオペレーティングシステム上の32ビットInternet Explorer
- 32ビットオペレーティングシステム上の32ビットFirefox
- 64ビットオペレーティングシステム上の64ビットInternet Explorer

注: Windowsオペレーティングシステムで提供されるtelnetクライアントは64ビットWindowsオペレーティングシステムで実行されるInternet Explorerの32ビットバージョンでは動作しません。これを解決するには、64ビットバージョンのInternet Explorerを使用します。Windows 64ビットオペレーティングシステムには、Internet Explorerの32ビットバージョンおよび64ビットバージョンの両方が含まれています。次のディレクトリでこれらのInternet Explorerバージョンを検索します。

- 64ビットバージョンの場合:`%ProgramFiles%/Internet Explorer`
- 32ビットバージョンの場合:`%ProgramFiles(x86)%/Internet Explorer`

Webブラウザーで使用するオペレーティングシステム提供のtelnetクライアントを設定するには、以下の手順を実行します。

1. (Microsoft Windows 7、Microsoft Vista、またはMicrosoft Windows Server専用) オペレーティングシステムに該当する手順に従い、コンピューターにオペレーティングシステムtelnetクライアントをインストールします。

   Windows 7またはVista:

   a. [コントロールパネル] で、[**プログラム**] をクリックしてから、[**プログラムと機能**] をクリックします。

   b. [タスク] で、[**Windowsの機能の有効化または無効化**] をクリックします。

   c. [Windowsの機能] ダイアログボックスで、[**Telnetクライアント**] チェックボックスをオンにして、[**OK**] をクリックします。

   Windows Serverの場合:

   a. [サーバーマネージャー] の [機能の概要] で、[**機能の追加**] をクリックします。

   b. [機能の追加ウィザード] で、[**Telnetクライアント**] チェックボックスをオンにして、[**次へ**]、[**インストール**] の順にクリックします。

2. (Internet Explorer専用) telnetを使用するInternet Explorerを有効化します。

   a. Windowsレジストリをバックアップします。

   b. Windowsレジストリエディターを使用して、[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL] キーに以下の値を追加します。

名前	タイプ	データ
iexplore.exe	REG_DWORD	0

3. URL:Telnetプロトコルファイルタイプのファイル関連付けを設定します。

   a. Windowsレジストリをバックアップします。

   b. Windowsレジストリエディターを使用して、[HKEY_CLASSES_ROOT\telnet\shell\open\command] キーを以下の値で変更します。

名前	タイプ	データ
(デフォルト)	REG_SZ	rundll32.exe url.dll,TelnetProtocolHandler %l

4. %1 (小文字のL) はtelnetに渡される引数で、通常はノードのIPアドレスまたは完全修飾ドメイン名。

   ヒント: 制御を厳しくするには、キーのバイナリへのパスを1行としてコード化できます。
   例:

   "C:\Windows\system32\rundll32.exe"

   "C:\Windows\system32\url.dll",TelnetProtocolHandler %l

5. Webブラウザーを再起動してから、ブラウザーのアドレスバーにtelnetコマンドを入力します。

   telnet://<node>

   <node>はtelnetサーバーを実行するノードのIPアドレスまたは完全修飾ドメイン名です。

   セキュリティ警告が表示される場合は、アクションを許可します。

   Firefoxで、[**今後telnetリンクを同様に処理する**] チェックボックスをオンにします。

サードパーティTelnetクライアント (標準Windows)

この手順は、以下の場合に適用されます。

- 32ビットオペレーティングシステム上の32ビットInternet Explorer
- 64ビットWindows 7オペレーティングシステム上の32ビットInternet Explorer
- 32ビットオペレーティングシステム上の32ビットFirefox
- 64ビットオペレーティングシステム上の64ビットInternet Explorer

Webブラウザーで使用するサードパーティtelnetクライアントを設定するには、以下の手順を実行します。

1. サードパーティtelnetクライアントを取得してインストールします。

   この手順では、C:\Program Files\PuTTY\putty.exeにインストールしたPuTTYクライアントを例に挙げます。PuTTYクライアントは
   http://www.putty.org から使用できます。

2. (Internet Explorer専用) telnetを使用するInternet Explorerを有効化します。
   a. Windowsレジストリをバックアップします。
   b. Windowsレジストリエディターを使用して、[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL] キーに以下の値を追加します。

名前	タイプ	データ
iexplore.exe	REG_DWORD	0

3. URL:Telnetプロトコルファイルタイプのファイル関連付けを設定します。
   a. Windowsレジストリをバックアップします。
   b. Windowsレジストリエディターを使用して、[HKEY_CLASSES_ROOT\telnet\shell\open\command] キーを以下の値で変更します。

名前	タイプ	データ
(デフォルト)	REG_SZ	"C:\Program Files\PuTTY\putty.exe" %l

   %1 (小文字のL) はtelnetに渡される引数で、通常はノードのIPアドレスまたは完全修飾ドメイン名。

   ヒント: .regファイルでは、各引用符 (") とバックスラッシュ (\) 文字はバックスラッシュ (\) 文字でエスケープします。

4. Webブラウザーを再起動してから、ブラウザーのアドレスバーにtelnetコマンドを入力します。

   telnet://<node>

   <node>はtelnetサーバーを実行するノードのIPアドレスまたは完全修飾ドメイン名です。

   セキュリティ警告が表示される場合は、アクションを許可します。

   Firefoxで、[**今後telnetリンクを同様に処理する**] チェックボックスをオンにします。

サードパーティTelnetクライアント (Windows上のウィンドウ)

この手順は、以下の場合に適用されます。

- 64ビットオペレーティングシステム上の32ビットInternet Explorer (Windows 7以外)
- 32ビットオペレーティングシステム上の64ビットFirefox

Webブラウザーで使用するサードパーティtelnetクライアントを設定するには、以下の手順を実行します。

1. サードパーティtelnetクライアントを取得してインストールします。

   この手順では、`C:\Program Files\PuTTY\putty.exe`にインストールしたPuTTYクライアントを例に挙げます。PuTTYクライアントは http://www.putty.org から使用できます。

2. (Internet Explorer専用) telnetを使用するInternet Explorerを有効化します。
   a. Windowsレジストリをバックアップします。
   b. Windowsレジストリエディターを使用して、[HKEY_LOCAL_MACHINE\

SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL] キーに以下の値を追加します。

名前	タイプ	データ
iexplore.exe	REG_DWORD	0

3. URL:Telnetプロトコルファイルタイプのファイル関連付けを設定します。
   a. Windowsレジストリをバックアップします。
   b. Windowsレジストリエディターを使用して、[HKEY_CLASSES_ROOT\Wow6432Node\telnet\shell\open\command] キーを以下の値で変更します。

名前	タイプ	データ
(デフォルト)	REG_SZ	"C:\Program Files\PuTTY\putty.exe" %l

%1 (小文字のL) はtelnetに渡される引数で、通常はノードのIPアドレスまたは完全修飾ドメイン名。

ヒント: .regファイルでは、各引用符 (") とバックスラッシュ (\) 文字はバックスラッシュ (\) 文字でエスケープします。

4. Webブラウザーを再起動してから、ブラウザーのアドレスバーにtelnetコマンドを入力します。

   telnet://<node>

   <node>はtelnetサーバーを実行するノードのIPアドレスまたは完全修飾ドメイン名です。

   セキュリティ警告が表示される場合は、アクションを許可します。

   Firefoxで、[**今後telnetリンクを同様に処理する**] チェックボックスをオンにします。

サードパーティSSHクライアント (標準Windowsおよび Windows上のウィンドウ)

この手順は、以下の場合に適用されます。

- 32ビットまたは64ビットオペレーティングシステム上の32ビットInternet Explorer
- 32ビットまたは64ビットオペレーティングシステム上の32ビットFirefox
- 64ビットオペレーティングシステム上の64ビットInternet Explorer

Webブラウザーで使用するサードパーティSSHクライアントを設定するには、以下の手順を実行します。

1. サードパーティSSHクライアントを取得してインストールします。

   この手順では、C:\Program Files\PuTTY\putty.exeにインストールしたPuTTYクライアントを例に挙げます。PuTTYクライアントは **http://www.putty.org** から使用できます。

2. PuTTYは 「ssh://<node>」 入力を正しく構文解析できないため、この例には入力引数から「ssh://」を取り除くスクリプトが含まれています。スクリプト `C:\Program Files\PuTTY\ssh.js`には、以下のコマンドが含まれます。

   ```
   host = WScript.Arguments(0).replace(/ssh:/,"").replace(/\//g,"");
   shell = WScript.CreateObject("WScript.Shell");
   shell.Run("\"c:\\Program Files\\PuTTY\\putty.exe\" -ssh " + host);
   ```

   ヒント: このスクリプトはこの例のために作成されたもので、PuTTYには含まれません。

3. sshプロトコルを定義します。
   a. Windowsレジストリをバックアップします。
   b. Windowsレジストリエディターを使用して、[HKEY_CLASSES_ROOT\ssh] キーに以下の値を追加します。

名前	タイプ	データ
(デフォルト)	REG_SZ	URL:sshプロトコル
EditFlags	REG_DWORD	2
FriendlyTypeName	REG_SZ	セキュアーシェル
URLプロトコル	REG_SZ	値なし

4. URL:sshプロトコルファイルタイプのファイル関連付けを設定します。
   a. Windowsレジストリをバックアップします。
   b. Windowsレジストリエディターを使用して、[HKEY_CLASSES_ROOT\ssh\shell\open\command] キーを以下の値で変更します。

名前	タイプ	データ
(デフォルト)	REG_SZ	"C:\Windows\System32\WScript.exe" "C:\Program Files\PuTTY\ssh.js" %l

   `%l` (小文字のL) は完全ssh引数で、プロトコル指定が含まれます。ssh.jsスクリプトはsshターゲットをPuTTYに渡します。

   ヒント: .regファイルでは、各引用符 (") とバックスラッシュ (\) 文字はバックスラッシュ (\) 文字でエスケープします。

5. Webブラウザーを再起動してから、ブラウザーのアドレスバーにsshコマンドを入力します。

   ssh://<node>

   <node>はtelnetサーバーを実行するノードのIPアドレスまたは完全修飾ドメイン名です。

   セキュリティ警告が表示される場合は、アクションを許可します。

   Firefoxで、[**今後sshリンクを同様に処理する**] チェックボックスをオンにします。

LinuxでTelnetまたはSSHを使用するFirefoxの設定

Linuxオペレーティングシステムにtelnetまたはsshプロトコルを定義してから、新規プロトコルを使用するようにFirefoxを設定します。

このセクションの手順を完了するには、コンピューターの管理権限が必要です。

詳細については、**http://kb.mozillazine.org/Register_protocol**を参照してください。

Linux上のTelnet

Linuxオペレーティングシステムでtelnetプロトコルを使用するようにFirefoxを設定するには、以下の手順を実行します。

1. telnetプロトコルを定義します。
   a. `/usr/local/bin/nnmtelnet`ファイルを以下の内容で作成します。

      ```
      #!/bin/bash
      #
      # Linux shell script called by Firefox in response to
      # telnet:// URLs for the NNMi telnet menu.
      #
      address=`echo $1 | cut -d : -f 2 | sed 's;/;;g'`
      port=`echo $1 | cut -d : -f 3`
      exec /usr/bin/xterm -e telnet $address $port
      ```

   b. 誰でも実行可能なスクリプト権限を設定します。

      chmod 755 /usr/local/bin/nnmtelnet
2. telnet用のFirefoxプリファレンスを設定します。
   a. Firefoxアドレスバーに、「`about:config`」と入力します。
   b. プリファレンスリスト内を右クリックし、[**新規**] をクリックしてから、[**ブール値**] をクリックします。
   c. プリファレンス名「`network.protocol-handler.expose.telnet`」を入力します。
   d. プリファレンス値**false**
3. 新規に定義されたプロトコルを使用するようにFirefoxを設定します。
   a. telnetリンクを参照します。

      ヒント: リンクを含む簡易HTMLファイルを作成、または **[アクション] > [Telnet...(クライアントから)]** をNNMiコンソールで使用できます。アドレスバーに直接リンクを入力しても、同じ結果にはなりません。

b. [アプリケーションの起動] ウィンドウで、[**選択**] をクリックしてから、`/usr/local/bin/nnmtelnet`を選択します。
c. [**今後telnetリンクを同様に処理する**] チェックボックスをオンにします。

Linux上のセキュアーシェル

Linuxオペレーティングシステムでsshプロトコルを使用するようにFirefoxを設定するには、以下の手順を実行します。

1. sshプロトコルを定義します。
   a. `/usr/local/bin/nnmssh`ファイルを以下の内容で作成します。

      ```
      #!/bin/bash
      #
      # Linux shell script called by Firefox in response to
      # ssh:// URLs for the NNMi SSH menu.
      #
      address=`echo $1 | cut -d : -f 2 | sed 's;/;;g'`
      port=`echo $1 | cut -d : -f 3`
      exec /usr/bin/xterm -e ssh $address $port
      ```

   b. 誰でも実行可能なスクリプト権限を設定します。

      chmod 755 /usr/local/bin/nnmssh
2. SSH用のFirefoxプリファレンスを設定します。
   a. Firefoxアドレスバーに、「`about:config`」と入力します。
   b. プリファレンスリスト内を右クリックし、[**新規**] をクリックしてから、[**ブール値**] をクリックします。
   c. プリファレンス名「`network.protocol-handler.expose.ssh`」を入力します。
   d. プリファレンス値**false**
3. 新規に定義されたプロトコルを使用するようにFirefoxを設定します。
   a. SSHリンクを参照します。

      ヒント: リンクを含む簡易HTMLファイルを作成、またはNNMiコンソールで定義した新規SSHメニュー項目を使用できます。アドレスバーに直接リンクを入力しても、同じ結果にはなりません。

   b. [アプリケーションの起動] ウィンドウで、[**選択**] をクリックしてから、`/usr/local/bin/nnmssh`を選択します。
   c. [**今後sshリンクを同様に処理する**] チェックボックスをオンにします。

Windowsレジストリを変更するファイル例

多くのNNMiユーザーがtelnetまたはsshプロトコルを使用してNNMiコンソールから管理対象ノードにアクセスする必要がある場合は、Windowsレジストリ更新を1つ以上の.regファイルで自動化することができます。このセクションには、独自の.regファイル作成の基準にできる.regファイル例が含まれます。レジストリキーは、アプリケーションとオペレーティングシステムが一致する場合と、64ビットのWindowsバージョンで32ビットのアプリケーションを実行する場合では異なるパスにあります。

詳細については、**http://support.microsoft.com/kb/310516**のMicrosoftの記事を参照してください。

nnmtelnet.regの例

このレジストリの内容例は、「Windowsオペレーティングシステム提供のTelnetクライアント」(366ページ) に適用されます。

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL]

"iexplore.exe"=dword:00000000

[HKEY_CLASSES_ROOT\telnet\shell\open\command]

@="\"C:\\Windows\\system32\\rundll32.exe\" \"C:\\Windows\\system32\\url.dll\",TelnetProtocolHandler %l"
```

nnmputtytelnet.regの例

このレジストリの内容例は、「サードパーティTelnetクライアント (標準Windows)」(368ページ) に適用されます。

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL]

"iexplore.exe"=dword:0c0000000

[HKEY_CLASSES_ROOT\telnet\shell\open\command]

@="\"C:\\Program Files\\PuTTY\\putty.exe\" %l"
```

nnmtelnet32on64.regの例

このレジストリの内容例は、「サードパーティTelnetクライアント (Windows上のウィンドウ)」(369ページ) に適用されます。

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_DISABLE_TELNET_PROTOCOL]
```

"iexplore.exe"=dword:00000000

[HKEY_CLASSES_ROOT\Wow6432Node\telnet\shell\open\command]

@="\"C:\\Program Files\\PuTTY\\putty.exe\" %l"

nnmssh.regの例

このレジストリの内容例は、 **「サードパーティSSHクライアント (標準WindowsおよびWindows上のウィンドウ)」(370ページ) に適用されます。**

Windows Registry Editor Version 5.00

[HKEY_CLASSES_ROOT\ssh]

@="URL:ssh Protocol"

"EditFlags"=dword:00000002

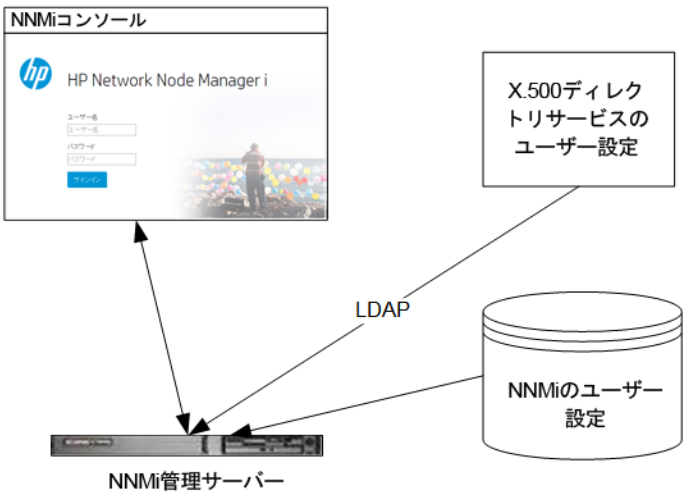
"FriendlyTypeName"="Secure Shell"

"URL Protocol"=""

[HKEY_CLASSES_ROOT\ssh\shell\open\command]

@="\"C:\\Windows\\System32\\WScript.exe\" \"c:\\Program Files\\PuTTY\\ssh.js\" %l"

NNMiとLDAPによるディレクトリサービスの統合



この章では、NNMiとディレクトリサービスを統合することにより、ユーザー名、パスワード、および必要に応じてNNMiユーザーグループの割り当ての保存場所を統合する方法について説明します。内容は以下のとおりです。

NNMiユーザーのアクセス情報と設定オプション

NNMiユーザーは、以下の項目によって定義されます。

- **ユーザー名**は、NNMiユーザーを一意に識別します。ユーザー名によってNNMiへのアクセスが許可され、インシデント割り当てを受け取ることができます。
- **パスワード**は、ユーザー名と関連付けられ、NNMiコンソールまたはNNMiコマンドへのアクセスを制御するために使用されます。
- **NNMiユーザーグループ**メンバーシップにより、提供する情報およびNNMiコンソールでユーザーが実行可能なアクションのタイプを制御します。ユーザーグループのメンバーシップに従って、ユーザーが使用可能なNNMiコマンドの制御も行われます。

NNMiには、以下の説明に従って、NNMiユーザーアクセス情報の保存先としていくつかのオプションが用意されています。以下の表に、NNMiユーザーアクセス情報を保存するデータベースを設定オプションごとに示します。

注: ユーザーが「外部」(オプション3) の使用を指定しない場合、NNMiではパスワードポリシーを適用するメカニズム (パスワード強度チェックなどのアカウント保護メカニズム) は設定されません。パスワードポリシー管理では、ユーザーにパスワード変更を定期的に求めるなどのベストプラクティスを実践することをお勧めします。

ユーザー情報の保存オプション

モード	ユーザーアカウント	ユーザーグループ	ユーザーグループメンバーシップ
内部 (オプション1)	NNMi	NNMi	NNMi
混合 (オプション2)	混合 (NNMiのアカウント名、LDAPのアカウントのパスワード)	NNMi	NNMi
外部 (オプション3)	ディレクトリサービス	両方	ディレクトリサービス

NNMiは、ライトウェイトディレクトリアクセスプロトコル (LDAP) を使用して、ディレクトリサービスと通信します。NNMiと一緒にLDAPを使用する場合、前述の表で示された以下のいずれかのモードを使用します。

- 混合モード (当初はオプション2と呼称):一部のNNMiユーザー情報をNNMiデータベースに、一部のNNMiユーザー情報をディレクトリサービスに保存

  混合モードを使用して、ユーザー名、ユーザーグループ、およびユーザーグループのマッピングをNNMiデータベースに保存するためにNNMiを設定し、ユーザー名およびパスワード (ユーザーアカウント) をディレクトリサービスに依存する必要があります。つまり、アカウント名の情報がNNMiとLDAPの両方に保存されている必要がありますが、アカウントのパスワードはLDAPのみに保存されます。
- 外部モード (当初はオプション3と呼称):すべてのNNMiユーザー情報をディレクトリサービスに保存

  外部モードを使用すると、すべてのユーザーアカウント情報がLDAPを使用して保存されるため、NNMiにユーザーアカウント情報を追加する必要はありません。

新しいユーザーアカウントを追加するときや、混合モードを使用して既存のアカウントを変更するときは、[**ディレクトリサービスアカウント**] チェックボックスをオンにする必要があります。ユーザーアカウントを設定するときは、一部のユーザーの [**ディレクトリサービスアカウント**] チェックボックスをオンにして、ほかのユーザーでは、内部モード、混合モード、および外部モードを組み合わせる方法としてオフにする、という設定を行わないでください。この設定はサポートされていません。

NNMiを、ユーザーアクセス情報の一部またはすべてを保存するディレクトリサービスと統合すると、[**システム情報**] ウィンドウの [**サーバー**] タブのユーザーアカウントおよびユーザーグループ定義ステートメントに、LDAP照会で取得した情報のタイプが示されます。

NNMiとほかのアプリケーションの間のシングルサインオン (SSO) は、NNMiユーザーアクセス情報の設定やその保存場所に関係なく機能します。

外部モード (当初はオプション1と呼称):すべてのNNMiユーザー情報をNNMiデータベースに保存

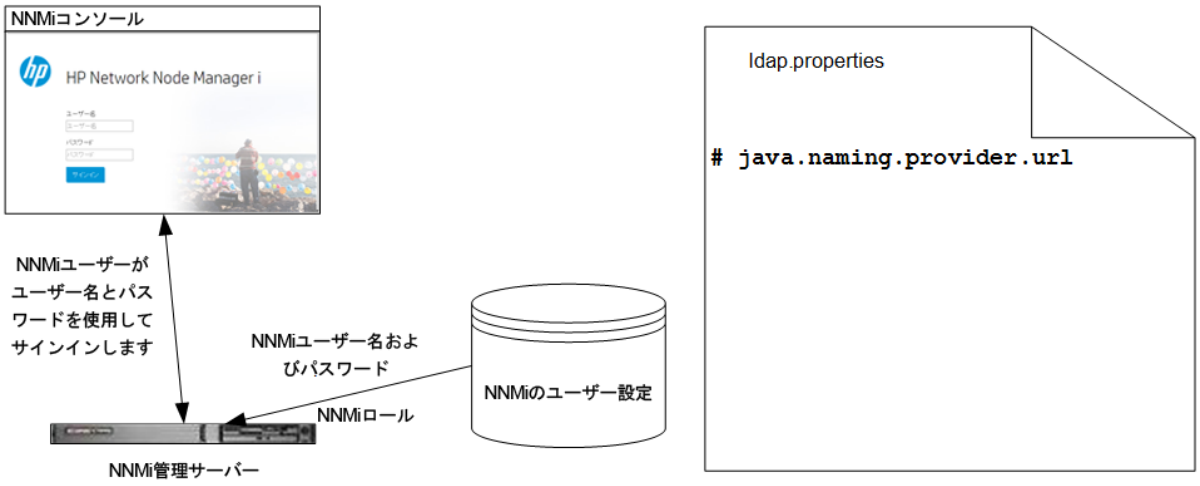
内部モードを使用した設定では、NNMiが、すべてのユーザーアクセス情報を取得するためにNNMiデータベースにアクセスします。それらの情報は、NNMi管理者がNNMiコンソールで定義およびメンテナンスします。ユーザーアクセス情報は、NNMiにとってローカルの情報となります。NNMiはディレクトリサービスにアクセスせず、NNMiは (以下の図のコメント行に示されている) `ldap.properties`ファイルを無視します。

以下の図に、このオプションの情報フローを示します。この情報フローは、以下のような状況に適しています。

- NNMiユーザーの数が少ない。
- ディレクトリサービスを使用していない。

NNMiデータベースですべてのユーザー情報を設定する方法の詳細については、NNMiヘルプの「NNMiでアクセスを制御する」を参照してください。この章を読む必要はありません。

内部モードにおけるNNMiユーザーサインインの情報フロー



混合モード(当初はオプション2と呼称):一部のNNMiユーザー情報をNNMiデータベースに、一部のNNMiユーザー情報をディレクトリサービスに保存

混合モードを使用した設定では、NNMiが、ユーザー名とパスワードを取得するためにディレクトリサービスにアクセスします。それらの情報は、NNMiの外部で定義され、ほかのアプリケーションでも使用できます。ユーザーからNNMiユーザーグループへのマッピングは、NNMiコンソールでメンテナンスします。NNMiユーザーアクセス情報の設定およびメンテナンスは、以下で説明するように共同で行われます。

- ディレクトリサービス管理者は、ディレクトリサービス内のユーザー名とパスワードをメンテナンスします。
- NNMi管理者は、(ディレクトリサービスで定義されている) ユーザー名、ユーザーグループ定義、ユーザーグループのマッピングをNNMiコンソールで入力します。
- NNMi管理者は、NNMiに対するユーザー名のディレクトリサービスデータベーススキーマを記述するNNMi `ldap.properties`ファイルを設定します(以下の図のコメント行は、NNMiがユーザーグループ情報をディレクトリサービスから引き出さないことを示しています)。

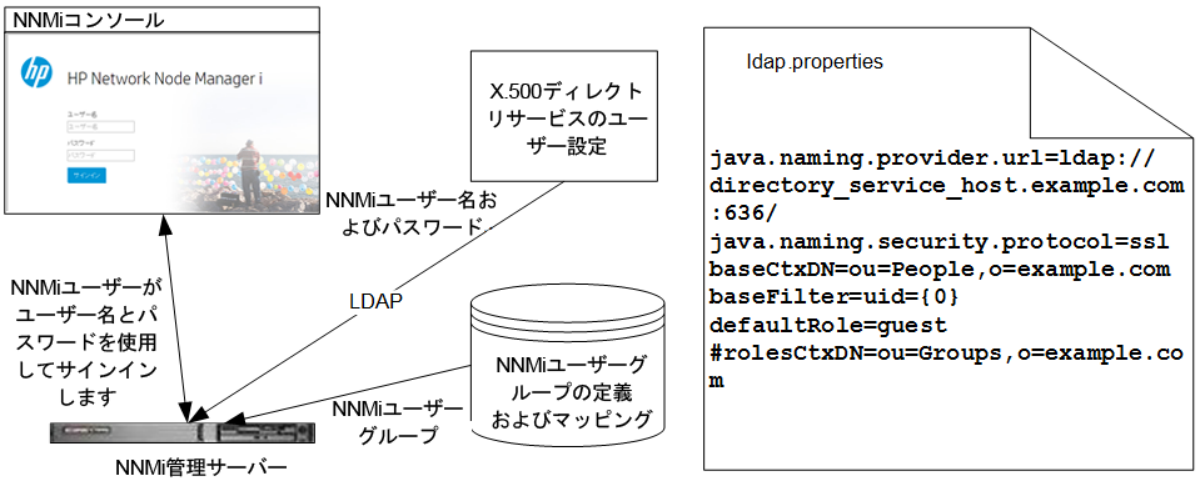
ユーザー名は、2か所で入力する必要があるため、両方の場所でユーザー名のメンテナンスを行う必要があります。

以下の図に、このオプションの情報フローを示します。この情報フローは、以下のような状況に適しています。

- NNMiユーザーの数が少なく、ディレクトリサービスを使用できる。
- ユーザーグループの変更ごとにディレクトリサービスの変更を必要とするのではなく、NNMi管理者がユーザーグループを管理する。
- ディレクトリサービスのグループ定義を簡単には拡張できない。

ユーザー名とパスワードを保存するディレクトリサービスとの統合に関する詳細については、この章の以降の説明と、NNMiヘルプの「ディレクトリサービスおよびNNMiを使用してアクセスを制御する」を参照してください。

混合モードで使用するNNMiユーザーサインインの情報フロー



外部モード(当初はオプション3と呼称):すべてのNNMiユーザー情報をディレクトリサービスに保存

外部モードを使用した設定では、NNMiが、すべてのユーザーアクセス情報を取得するためにディレクトリサービスにアクセスします。それらの情報は、NNMiの外部で定義され、ほかのアプリケーションが使用できます。1つ以上のディレクトリサービスグループでのメンバーシップにより、ユーザーのNNMiユーザーグループが決まります。

NNMiユーザーアクセス情報の設定およびメンテナンスは、以下で説明するように共同で行われます。

- ディレクトリサービス管理者は、ディレクトリサービス内のユーザー名、パスワード、グループメンバーシップをメンテナンスします。
- NNMi管理者は、ディレクトリサービスグループをNNMiユーザーグループにNNMiコンソールでマップします。
- NNMi管理者は、NNMiに対するユーザー名およびグループのディレクトリサービスデータベーススキーマを記述するNNMi `ldap.properties`ファイルを設定します。

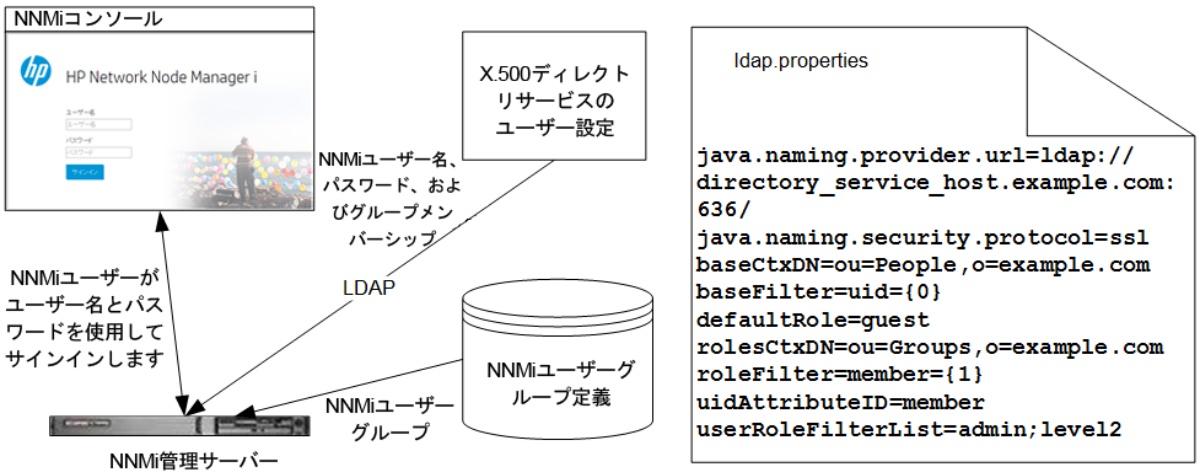
以下の図に、このオプションでの情報フローを示します。これは、NNMiにアクセスする必要があるユーザーで構成されるユーザーグループを含めるようにディレクトリサービスを変更することが可能な環境に適しています。

このオプションは混合モードの例を拡張した形態であるため、HPでは以下の設定プロセスを推奨します。

1. ディレクトリサービスからNNMiユーザー名とパスワードを取得するよう設定して検証します。
2. ディレクトリサービスからNNMiユーザーグループを取得するように設定します。

すべてのユーザー情報を保存するディレクトリサービスとの統合に関する詳細については、この章の以降の説明と、NNMiヘルプの「ディレクトリサービスを使用してアクセスを制御する」を参照してください。

外部モードで使用するNNMiユーザーサインインの情報フロー



ディレクトリサービスにアクセスするNNMiの設定

ディレクトリサービスへのアクセスは、以下のファイルで設定されています。

- Windowsの場合: `%NNM_SHARED_CONF%\ldap.properties`
- Linuxの場合: `$NNM_SHARED_CONF/ldap.properties`

このファイルの詳細については、「ldap.properties設定ファイルリファレンス」(401ページ)を参照してください。「例」(406ページ)も参照してください。

ディレクトリサービスの一般的な構造の詳細については、「ディレクトリサービスのクエリー」(389ページ)を参照してください。

混合モードで設定する場合は、以下のタスクを実行します。

- タスク1:現在のNNMiユーザー情報をバックアップする
- タスク2:省略可能。ディレクトリサービスへのセキュアー接続を設定する
- タスク3:ディレクトリサービスからのユーザーアクセスを設定する
- タスク4:ユーザー名とパスワードの設定をテストする
- タスク9:クリーンアップしてNNMiへの予期せぬアクセスを防止する
- タスク10:省略可能。ユーザーグループをセキュリティグループにマッピングする

外部モードで設定する場合は、以下のタスクを実行します。

- タスク1: 現在のNNMiユーザー情報をバックアップする
- タスク2:省略可能。ディレクトリサービスへのセキュアー接続を設定する
- タスク3:ディレクトリサービスからのユーザーアクセスを設定する
- タスク4:ユーザー名とパスワードの設定をテストする
- タスク5:タスク5: (設定オプション3のみ) ディレクトリサービスからのグループの取得を設定する

  注: ディレクトリサービスにNNMiユーザーグループを保存する場合は、NNMiユーザーグループによってディレクトリサービスを設定する必要があります。詳細については、「NNMiユーザーグループを保存するディレクトリサービスの設定」(400ページ)を参照してください。

- タスク6:タスク6: (設定オプション3のみ) ディレクトリサービスグループをNNMiユーザーグループにマッピングする
- タスク7:タスク7: (設定オプション3のみ) NNMiユーザーグループ設定をテストする
- タスク8:タスク8: (設定オプション3のみ) インシデント割り当てのNNMiユーザーグループを設定する
- タスク9:クリーンアップしてNNMiへの予期せぬアクセスを防止する
- タスク10:省略可能。ユーザーグループをセキュリティグループにマッピングする

タスク1: 現在のNNMiユーザー情報をバックアップする

NNMiデータベースのユーザー情報をバックアップします。

**nnmconfigexport.ovpl -c account -u <user>
-p <password> -f NNMi_database_accounts.xml**

タスク2 (省略可能)ディレクトリサービスへのセキュアー接続を設定する

ディレクトリサービスでSecure Socket Layer (SSL) を使用する必要がある場合は、「ディレクトリサービスへのSSL接続を設定する」(331ページ) の説明に従って、自社の証明書をNNMiトラストストアーにインポートします。

タスク3: ディレクトリサービスからユーザーアクセスを設定する

混合モードおよび外部モードの場合のみ以下のタスクを実行します。ディレクトリサービスに応じた適切な手順に従ってください。このタスクには、以下のセクションが含まれます。

- Microsoft Active Directoryの場合の簡単な方法
- 他のディレクトリサービスの場合の簡単な方法

(設定の詳細な手順については、「ユーザー識別」(394ページ)を参照してください。)

Microsoft Active Directoryの場合の簡単な方法

1. NNMiに付属するldap.propertiesファイルをバックアップしてから、そのファイルを任意のテキストエディターで開きます。
2. ファイルの内容を以下のテキストで上書きします。

   java.naming.provider.url=ldap://<myldapserver>:389/

   bindDN=<mydomain>\\<myusername>

   bindCredential=<mypassword>

   baseCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,DC=<mysuffix>

   baseFilter=CN={0}

   defaultRole=guest

   #rolesCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,DC=<mysuffix>

   roleFilter=member={1}

   uidAttributeID=member

   userRoleFilterList=admin;level2;level1

3. ディレクトリサービスにアクセスするときのURLを指定します。上のテキストには以下の行があります。

   java.naming.provider.url=ldap://<myldapserver>:389/

   <myldapserver>を、Active Directoryサーバーの完全修飾ホスト名 (例: myserver.example.com) で置き換えます。

   ヒント: 複数のディレクトリサービスURLを指定するには、各URLをスペース文字1つ ( ) で区切ります。

4. 有効なディレクトリサービスユーザーの資格証明を指定します。上のテキストには以下の行があります。

   bindDN=<mydomain>\\<myusername>
   bindCredential=<mypassword>

   以下のように置き換えます。

   - <mydomain>をActive Directoryドメインの名前で置き換えます。
   - <myusername>および<mypassword>をActive Directoryサーバーにアクセスするときに使用するユーザー名とパスワードで置き換えます。

     平文のパスワードを保存する場合は、ディレクトリサービスへの読み取り専用アクセス権を付与してユーザー名を指定してください。暗号化されたパスワードを指定する場合は、ldap.propertiesファイルに保存する前に平文のパスワードを以下のコマンドで暗号化します。

     nnmldap.ovpl -encrypt <mypassword>

> **注:** この暗号化パスワードは、その作成先のNNMiインスタンスでのみ機能します。ほかのNNMiインスタンスには使用しないでください。

詳細については、nnmldap.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

5. ディレクトリサーバードメインの中でユーザーレコードを保存する部分を指定します。上のテキストには以下の行があります。

   ```
   baseCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,
     DC=<mysuffix>
   ```

   <myhostname>、<mycompanyname>、および<mysuffix>をActive Directoryサーバーの完全修飾ホスト名のコンポーネントで置き換えます (たとえばホスト名myserver.example.comの場合は、DC=myserver,DC=example,DC=comと指定します)。

他のディレクトリサービスの場合の簡単な方法

1. NNMiに付属するldap.propertiesファイルをバックアップしてから、そのファイルを任意のテキストエディターで開きます。
2. ディレクトリサービスにアクセスするときのURLを指定します。上のテキストには以下の行があります。

   ```
   #java.naming.provider.url=ldap://<myldapserver>:389/
   ```

   以下の手順を実行します。

   - 行のコメントを解除します (#文字を削除します)。
   - <myldapserver>を、ディレクトリサーバーの完全修飾ホスト名 (例: myserver.example.com) で置き換えます。

     > **ヒント:** 複数のディレクトリサービスURLを指定するには、各URLをスペース文字1つ ( ) で区切ります。

3. ディレクトリサーバードメインの中でユーザーレコードを保存する部分を指定します。上のテキストには以下の行があります。

   ```
   baseCtxDN=ou=People,o=myco.com
   ```

   ou=People,o=myco.comをユーザーレコードを保存するディレクトリサービスドメインの部分で置き換えます。
4. NNMiにサインインするユーザー名の形式を指定します。

   上のテキストには以下の行があります。

   ```
   baseFilter=uid={0}
   ```

   uidをディレクトリサービスドメインのユーザー名属性で置き換えます。

タスク4:ユーザー名とパスワードの設定をテストする

1. ldap.propertiesファイルで、テスト用にdefaultRole=guestと設定します(この値はいつでも変更できます)。
2. ldap.propertiesファイルを保存します。
3. 以下のコマンドを実行して、NNMiにldap.propertiesファイルを再読み込みさせます。

   nnmldap.ovpl -reload
4. ディレクトリサービスで定義されているユーザー名とパスワードを使用して、NNMiコンソールにログオンします。

   ヒント: このテストは、NNMiデータベースでまだ定義されていないユーザー名を使用して実行してください。
5. NNMiコンソールのタイトルバーで、ユーザー名とNNMiロール (ゲスト) を確認します。
   - ユーザーサインインが正しく動作したら、このタスクの手順8に進みます。
   - ユーザーサインインが正しく動作しない場合は、手順6に進みます。

     ヒント: 各テストの後で、NNMiコンソールからサインアウトしてセッション資格証明をクリアします。
6. 以下のコマンドを実行し、あるユーザーの設定をテストします。

   nnmldap.ovpl -diagnose <NNMi_user>

   <NNMi_user>は、ディレクトリサービスで定義したNNMiユーザーのサインイン名で置き換えます。

   コマンド出力を検討し、適切に応答します。推奨事項は以下のとおりです。
   - タスク3が正常に完了したことを確認します。
   - 「ユーザー識別」(394ページ)の詳細な設定プロセスに従います。
7. NNMiコンソールへのサインイン時に期待する結果が表示されるまで、手順1から手順5を繰り返します。
8. ログオンできたら、設定方法を選択します。
   - NNMiユーザーグループメンバーシップをNNMiデータベースに保存する (混合モードを使用する設定) 場合は、タスク9に進みます。
   - NNMiユーザーグループメンバーシップをディレクトリサービスに保存する (外部モードを使用する設定) 場合は、タスク5に進みます。

タスク5:(外部モードのみ) ディレクトリサービスからのグループの取得を設定する

このタスクは、設定オプション3の場合に実行します。ディレクトリサービスに応じた適切な手順に従ってください。このタスクには、以下のセクションが含まれます。

- Microsoft Active Directoryの場合の簡単な方法
- 他のディレクトリサービスの場合の簡単な方法

(設定の詳細な手順については、「ユーザーグループの識別」(397ページ)を参照してください。)

Microsoft Active Directoryの場合の簡単な方法

1. ldap.propertiesファイルをバックアップしてから、そのファイルを任意のテキストエディターで開きます。
2. ディレクトリサーバードメインの中でグループレコードを保存する部分を指定します。上のテキストには以下の行があります。

   ```
   #rolesCtxDN=CN=Users,DC=<myhostname>,DC=<mycompanyname>,
     DC=<mysuffix>
   ```

   以下の手順を実行します。
   - 行のコメントを解除します (#文字を削除します)。
   - <myhostname>、<mycompanyname>、および<mysuffix>をActive Directoryサーバーの完全修飾ホスト名のコンポーネントで置き換えます (たとえばホスト名myserver.example.comの場合は、DC=myserver,DC=example,DC=comと指定します)。

他のディレクトリサービスの場合の簡単な方法

1. ldap.propertiesファイルをバックアップしてから、そのファイルを任意のテキストエディターで開きます。
2. ディレクトリサーバードメインの中でグループレコードを保存する部分を指定します。上のテキストには以下の行があります。

   ```
   #rolesCtxDN=ou=Groups,o=myco.com
   ```

   以下の手順を実行します。
   - 行のコメントを解除します (#文字を削除します)。
   - ou=Groups,o=myco.comを、ディレクトリサービスドメインのグループレコードを保存する部分で置き換えます。
3. ディレクトリサービスのグループ定義でグループメンバー名の形式を指定します。上のテキストには以下の行があります。

   ```
   roleFilter=member={1}
   ```

memberを、ディレクトリサービスドメインのディレクトリサービスユーザーIDを保存するグループ属性の名前で置き換えます。

タスク6:(外部モードのみ) ディレクトリサービスグループをNNMiユーザーグループにマップする

1. NNMiコンソールで、定義済みのNNMiユーザーグループをディレクトリサービスのユーザーグループにマップします。
   a. **[ユーザーグループ]** ビューを開きます。

      [設定] ワークスペースで **[セキュリティ]** を展開してから **[ユーザーグループ]** をクリックします。
   b. **[admin]** 行をダブルクリックします。
   c. **[ディレクトリサービス名]** フィールドに、NNMi管理者のディレクトリサービスグループの完全識別名を入力します。
   d. **[保存して閉じる]** アイコンをクリックします。
   e. **guest**、**level1**、**level2**の行ごとに手順bから手順dを繰り返します。

   ヒント: このマッピングにより、NNMiコンソールにアクセスできるようになります。NNMiコンソールにアクセスするすべてのユーザーは、この手順で指定した、定義済みのNNMiユーザーグループのうちいずれかにマッピングされているディレクトリサービスグループに含まれている必要があります。
2. ディレクトリサービスで1人以上のNNMiユーザーを含むその他のグループに、NNMiコンソールで新しいユーザーグループを作成します。
   a. **[ユーザーグループ]** ビューを開きます。

      [設定] ワークスペースで **[セキュリティ]** を展開してから **[ユーザーグループ]** をクリックします。
   b. ✱ **[新規作成]** アイコンをクリックしてから、グループの情報を入力します。
      - **[一意の名前]** は一意の値に設定します。短い名前にすることをお勧めします。
      - **[表示名]** は、ユーザーに表示される値に設定します。
      - **[ディレクトリサービス名]** は、ディレクトリサービスグループの完全識別名に設定します。
      - **[説明]** は、このNNMiユーザーグループの目的を説明するテキストに設定します。
   c. **[保存して閉じる]** をクリックします。
   d. NNMiユーザーの追加のディレクトリサービスグループごとに手順bと手順cを繰り返します。

   ヒント: このマッピングにより、NNMiコンソールのトポロジオブジェクトにアクセスできるようになります。各ディレクトリサービスグループは、複数のNNMiユーザーグループにマッピングできます。

タスク7:(外部モードのみ) NNMiユーザーグループ設定をテストする

1. ldap.propertiesファイルを保存します。
2. 以下のコマンドを実行して、NNMiにldap.propertiesファイルを再読み込みさせます。

   nnmldap.ovpl -reload
3. ディレクトリサービスで定義されているユーザー名とパスワードを使用して、NNMiコンソールにログオンします。

   注: NNMiデータベースでまだ定義されておらず、admin、level1、level2のNNMiユーザーグループにマッピングされているディレクトリサービスグループのメンバーであるユーザー名で、このテストを実行します。
4. ユーザー名とNNMiロール ([**ユーザーグループ**] ビューの [**表示名**] フィールドで定義したもの) をNNMiコンソールのタイトルバーで確認します。
   - ユーザーサインインが正しく動作したら、タスク8に進みます。
   - ユーザーサインインが正しく動作しない場合は、手順5に進みます。

   ヒント: 各テストの後で、NNMiコンソールからサインアウトしてセッション資格証明をクリアします。
5. 以下のコマンドを実行し、あるユーザーの設定をテストします。

   nnmldap.ovpl -diagnose <NNMi_user>

   <NNMi_user>は、ディレクトリサービスで定義したNNMiユーザーのサインイン名で置き換えます。

   コマンド出力を検討し、適切に応答します。推奨事項は以下のとおりです。
   - タスク5が正常に完了したことを確認します。
   - 定義済みのNNMiユーザーグループごとに、タスク6が正常に完了したことを確認します。
   - 「ユーザーグループの識別」(397ページ)の詳細な設定プロセスに従います。
6. NNMiコンソールへのサインイン時に期待する結果が表示されるまで、手順1から手順4を繰り返します。

タスク8:(外部モードのみ) インシデント割り当てのNNMiユーザーグループを設定する

1. ldap.propertiesファイルをバックアップしてから、そのファイルを任意のテキストエディターで開きます。

2. インシデントを割り当てることができるNNMiロールをNNMiオペレーターが指定するように、userRoleFilterListパラメーター値を変更します。

   ヒント: 1つ以上の定義済みNNMiユーザーグループ名の一意の名前 (「ユーザーグループの識別」(397ページ) で定義) をセミコロンで区切ったリストという形式です。

3. ldap.propertiesファイルを保存します。
4. 以下のコマンドを実行して、NNMiにldap.propertiesファイルを再読み込みさせます。

   ```
   nnmldap.ovpl -reload
   ```

5. ディレクトリサービスで定義されているユーザー名とパスワードを使用して、NNMiコンソールにログオンします。
6. 任意のインシデントビューでインシデントを選択し、**[アクション] > [割り当て] > [インシデントの割り当て]** をクリックします。userRoleFilterListパラメーターによって指定されている各NNMiロールのユーザーに、インシデントを割り当てることができることを確認します。
7. 設定した各NNMiロールにインシデントを割り当てることができるまで、手順1から手順6の操作を繰り返してください。

タスク9:クリーンアップしてNNMiへの予期せぬアクセスを防止する

1. **省略可能。**ldap.propertiesファイルで、defaultRoleパラメーターの値を変更するか、またはコメントを解除します。
2. (混合モードのみ) NNMiデータベースにユーザーグループメンバーシップを保存するには、以下の手順を実行して、NNMiデータベースのユーザーアクセス情報をリセットします。
   a. 既存のユーザーアクセス情報すべてを削除します([**ユーザーアカウント**] ビューのすべての行を削除します)。

      詳細については、NNMiヘルプの「ユーザーアカウントを削除する」を参照してください。
   b. NNMiユーザーごとに、ユーザー名の [**ユーザーアカウント**] ビューに新しいオブジェクトを作成します。
      - [**名前**] フィールドに、ディレクトリサービスに定義されているユーザー名を入力します。
      - [**ディレクトリサービスアカウント**] チェックボックスを選択します。
      - パスワードは指定しないでください。

      詳細については、NNMiヘルプの「ユーザーアカウントタスク」を参照してください。
   c. NNMiユーザーごとに、1つ以上のNNMiユーザーグループにユーザーアカウントをマップします。

      詳細については、NNMiヘルプの「ユーザーアカウントマッピングタスク」を参照してください。
   d. インシデント所有権を更新して、各割り当てインシデントが有効なユーザー名と関連付けられるようにします。

詳細については、NNMiヘルプの「インシデント割り当てを管理する」を参照してください。

3. (外部モードのみ) ディレクトリサービスのユーザーグループメンバーシップを使用するには、以下の手順を実行して、NNMiデータベースのユーザーアクセス情報をリセットします。
   a. 既存のユーザーアクセス情報すべてを削除します([**ユーザーアカウント**] ビューのすべての行を削除します)。

      詳細については、NNMiヘルプの「ユーザーアカウントを削除する」を参照してください。
   b. インシデント所有権を更新して、各割り当てインシデントが有効なユーザー名と関連付けられるようにします。

      詳細については、NNMiヘルプの「インシデント割り当てを管理する」を参照してください。

タスク10:省略可能。ユーザーグループをセキュリティグループにマッピングする

詳細については、NNMiヘルプの「セキュリティグループマッピングタスク」を参照してください。

ディレクトリサービスのクエリー

NNMiは、LDAPを使用してディレクトリサービスと通信します。NNMiが要求を送信すると、ディレクトリサービスは保存されている情報を返します。NNMiは、ディレクトリサービスに保存されている情報を変更できません。

このセクションでは以下の内容について説明します。

- 「ディレクトリサービスアクセス」(389ページ)
- 「ディレクトリサービスの情報」(390ページ)
- 「ディレクトリサービス管理者が所有する情報」(393ページ)
- 「ユーザー識別」(394ページ)
- 「ユーザーグループの識別」(397ページ)

ディレクトリサービスアクセス

LDAPは、以下の形式でディレクトリサービスに対してクエリーを実行します。

- ldap://<directory_service_host>:<port>/<search_string>
- `ldap`はプロトコル指定子です。この指定子は、ディレクトリサービスへの標準接続とSSL接続の両方で使用してください。
- <directory_service_host>は、ディレクトリサービスをホストするコンピューターの完全修飾名です。

- <port>は、LDAP通信でディレクトリサービスが使用するポートです。非SSL接続のデフォルトポートは389です。 SSL接続のデフォルトポートは636です。
- <search_string>には要求情報が指定されます。詳細については、「ディレクトリサービスの情報」(390ページ)と、以下で入手できるRFC 1959「An LDAP URL Format」を参照してください。
  labs.apache.org/webarch/uri/rfc/rfc1959.txt

WebブラウザーでLDAPクエリーをURLとして入力し、アクセス情報が正しく、検索文字列の構造が正しいことを確認できます。

ヒント: ディレクトリサービス (たとえば、Active Directory) が匿名アクセスを許可しない場合、そのディレクトリはWebブラウザーからのLDAPクエリーを拒否します。この場合は、サードパーティ製のLDAPブラウザー (Apache Directory Studioに含まれるLDAPブラウザーなど) を使用し、設定パラメーターの有効性を検証できます。

ディレクトリサービスの情報

ディレクトリサービスには、ユーザー名、パスワード、およびグループメンバーシップなどの情報が保存されています。ディレクトリサービス内の情報にアクセスするには、情報の保存場所を参照する識別名を知っている必要があります。サインインアプリケーションの場合の識別名は、可変情報 (ユーザー名など) と固定情報 (ユーザー名の保存場所など) の組み合わせです。識別名を構成するエレメントは、ディレクトリサービスの構造と内容によって決まります。

以下の例は、USERS-NNMi-Adminというユーザーグループの場合に考えられる定義を示しています。このグループは、NNMiへの管理アクセス権限を持つディレクトリサーバーのユーザーIDのリストで構成されます。以下の情報は、これらの例に関係しています。

- Active Directoryの例は、Windowsオペレーティングシステムの場合です。
- ほかのディレクトリサービスの例は、Linuxオペレーティングシステムの場合です。
- それぞれの例に示すファイルは、LDIF (lightweight directory interchange format) ファイルの一部です。LDIFファイルにより、ディレクトリサービスの情報を共有できます。
- それぞれの例の図は、ディレクトリサービスドメインをグラフィカルに表現したものです。この図は、引用したLDIFファイルに含まれる情報を拡張して表示したものです。

Active Directoryの情報構造例

この例での関心の対象は以下の項目です。

- ユーザーJohn Doeの識別名:
  `CN=john.doe@example.com,OU=Users,OU=Accounts,DC=example,DC=com`
- グループUSERS-NNMi-Adminの識別名: `CN=USERS-NNMi-Admin,OU=Groups,OU=Accounts,DC=example,DC=com`
- ディレクトリサービスユーザーIDを保存するグループ属性: `member`

LDIFファイルの引用例:

```
groups |USERS-NNMi-Admin

dn:CN=USERS-NNMi-Admin,OU=Groups,OU=Accounts,DC=example,DC=com
```

cn:**USERS-NNMi-Admin**

description:Group of users for NNMi administration.

member:CN=john.doe@example.com,OU=Users,OU=Accounts,

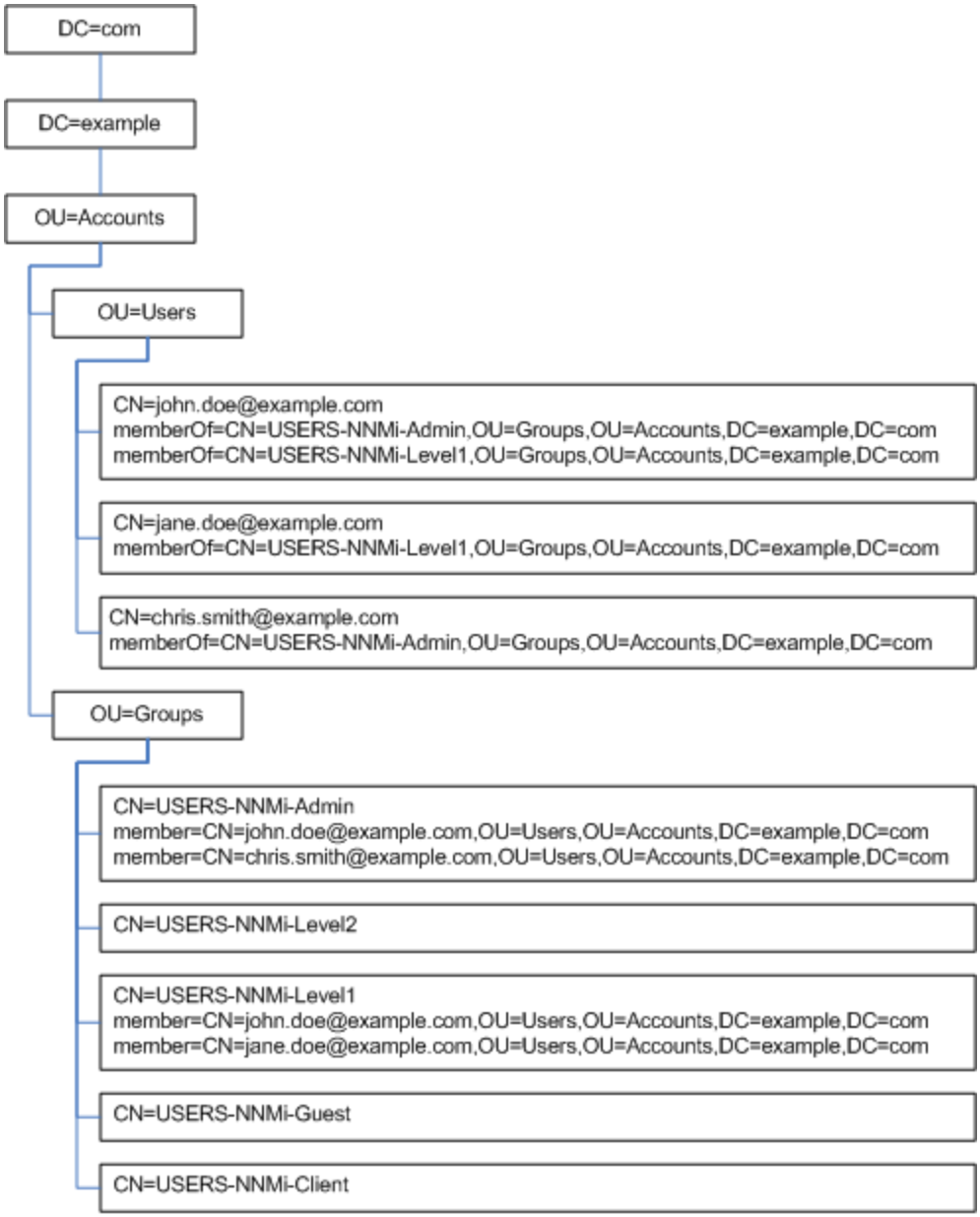
DC=example,DC=com

member:CN=chris.smith@example.com,OU=Users,OU=Accounts,

DC=example,DC=com

以下の図に、このディレクトリサービスのドメインを示します。

Active Directoryのドメイン例



他のディレクトリサービスの情報構造例

この例での関心の対象は以下の項目です。

- ユーザーJohn Doeの識別名: `uid=john.doe@example.com,ou=People,o=example.com`
- グループUSERS-NNMi-Adminの識別名: `cn=USERS-NNMi-Admin,ou=Groups,o=example.com`
- ディレクトリサービスユーザーIDを保存するグループ属性: member

LDIFファイルの引用例:

groups |USERS-NNMi-Admin

dn: cn=USERS-NNMi-Admin,ou=Groups,o=example.com

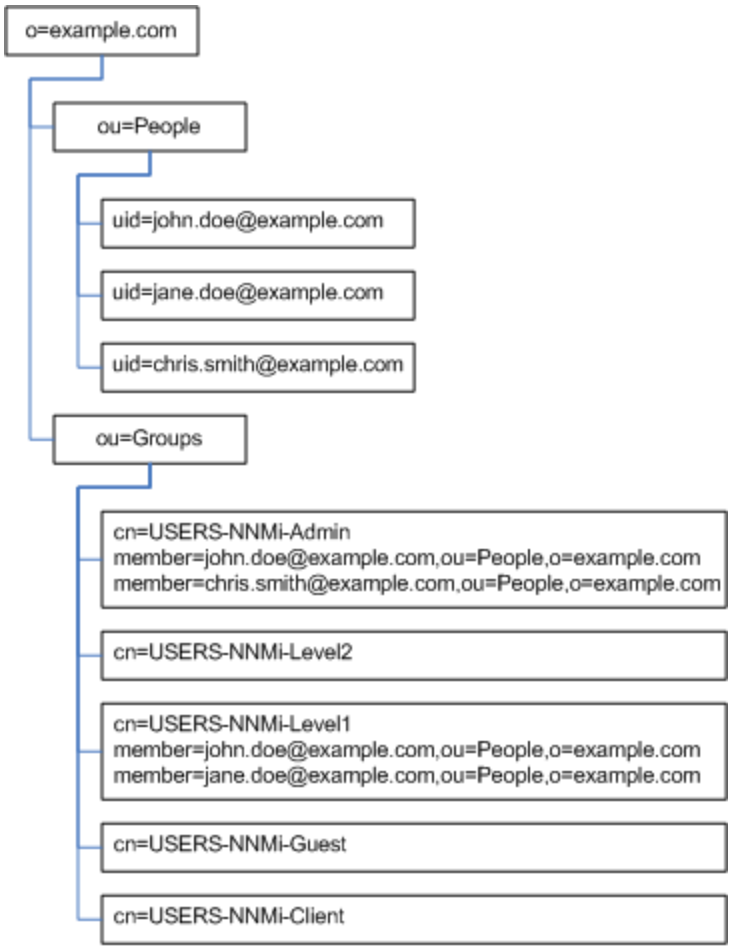
cn:**USERS-NNMi-Admin**

description:Group of users for NNMi administration.

member: uid=john.doe@example.com,ou=People,o=example.com

member: uid=chris.smith@example.com,ou=People,o=example.com

他のディレクトリサービスのドメインの例



ディレクトリサービス管理者が所有する情報

以下の表に、ディレクトリサービスにLDAPアクセスするようにNNMiを設定する前に、ディレクトリサービス管理者から取得する情報をリストします。

- ユーザー名とパスワードのみにディレクトリサービスを使用する場合は (混合モードのみ)、ユーザー名およびパスワードをディレクトリサービスから取得するための情報を収集します。
- すべてのNNMiアクセス情報にディレクトリサービスを使用する場合は (外部モードのみ)、以下のそれぞれの表の情報を収集します。

ユーザー名およびパスワードをディレクトリサービスから取得するための情報

情報	Active Directoryの例	その他のディレクトリサービスの例
ディレクトリサービスをホストするコンピューターの完全修飾名	directory_service_host.example.com	
LDAP通信でディレクトリサービスが使用するポート	• 非SSL接続の場合は389 • SSL接続の場合は636	
ディレクトリサービスでのSSL接続情報	SSL接続が必要な場合は、会社のトラストストアー証明書のコピーを取得し、「ディレクトリサービスへのSSL接続を設定する」(331ページ) を参照します。	
ディレクトリサービスに保存される1つのユーザー名の識別名 (ディレクトリサービスドメインを示す)	CN=john.doe@example.com, OU=Users,OU=Accounts, DC=example,DC=com	uid=john.doe@example.com, ou=People,o=example.com

グループメンバーシップをディレクトリサービスから取得するための情報

情報	Active Directoryの例	その他のディレクトリサービスの例
ユーザーが割り当てられているグループを識別する識別名	memberOfユーザー属性によりグループを識別します。	• ou=Groups,o=example.com • cn=USERS-NNMi-*, ou=Groups,o=example.com
グループ内のユーザーを識別する方法	• CN=john.doe@example.com, OU=Users,OU=Accounts, DC=example,DC=com • CN=john.doe@example.com	• cn=john.doe@example.com, ou=People,o=example.com • cn=john.doe@example.com
ディレクトリサービスユーザーIDを保存するグループ属性	member	member

グループメンバーシップをディレクトリサービスから取得するための情報 (続き)

情報	Active Directoryの例	その他のディレクトリサービスの例
NNMiアクセスに適用するディレクトリサービスのグループの名前	• CN=USERS-NNMi-Admin, OU=Groups,OU=Accounts, DC=example,DC=com • CN=USERS-NNMi-Level2, OU=Groups,OU=Accounts, DC=example,DC=com • CN=USERS-NNMi-Level1, OU=Groups,OU=Accounts, DC=example,DC=com • CN=USERS-NNMi-Client, OU=Groups,OU=Accounts, DC=example,DC=com • CN=USERS-NNMi-Guest, OU=Groups,OU=Accounts, DC=example,DC=com	• cn=USERS-NNMi-Admin, ou=Groups,o=example.com • cn=USERS-NNMi-Level2, ou=Groups,o=example.com • cn=USERS-NNMi-Level1, ou=Groups,o=example.com • cn=USERS-NNMi-Client, ou=Groups,o=example.com • cn=USERS-NNMi-Guest, ou=Groups,o=example.com

ユーザー識別

ユーザー識別は、混合モードと外部モードに適用されます。

ユーザー識別のための識別名は、1人のユーザーをディレクトリサービスで特定するための完全に修飾する方法です。NNMiはユーザー識別名をLDAP要求でディレクトリサービスに渡します。

ldap.propertiesファイルでのユーザー識別名は、baseFilter値とbaseCtxDN値を連結した値です。ディレクトリサービスによって返されたパスワードが、NNMiコンソールにユーザーが入力したサインインパスワードと一致する場合、ユーザーサインインが続行されます。

混合モードの場合は、以下の情報が適用されます。

- NNMiコンソールアクセスの場合、NNMiは以下の情報を検討し、可能な限り高い権限をユーザーに与えます。
  - ldap.propertiesファイルのdefaultRoleパラメーターの値
  - NNMiコンソールで定義済みのNNMiユーザーグループにおける、このユーザーのメンバーシップ
- NNMiトポロジオブジェクトアクセスの場合、NNMiは、NNMiコンソールでこのユーザーが属するNNMiユーザーグループのセキュリティグループマッピングに従ってアクセス権を与えます。

外部モードの場合は、以下の情報が適用されます。

- NNMiコンソールアクセスの場合、NNMiは以下の情報を検討し、可能な限り高い権限をユーザーに与えます。

- ldap.propertiesファイルのdefaultRoleパラメーターの値
- NNMiコンソールで定義済みのNNMiユーザーグループにマッピングされている ([**ディレクトリサービス名**] フィールド) ディレクトリサービスグループにおける、このユーザーのメンバーシップ

- NNMiトポロジオブジェクトアクセスの場合、NNMiは、このユーザーがディレクトリサービス (NNMiコンソールでNNMiユーザーグループがマッピングされている) で属するグループのセキュリティグループマッピングに従ってアクセス権を与えます。

Active Directoryでのユーザー識別例

baseFilterをCN={0}に設定し、baseCtxDNをOU=Users,OU=Accounts,DC=example,DC=comに設定し、ユーザーがjohn.doeとしてNNMiにサインインする場合、ディレクトリサービスに渡される文字列は以下のとおりです。

CN=john.doe,OU=Users,OU=Accounts,DC=example,DC=com

その他のディレクトリサービスでのユーザー識別例

baseFilterをuid={0}@example.comに設定し、baseCtxDNをou=People,o=example.comに設定し、ユーザーがjohn.doeとしてNNMiにサインインする場合、ディレクトリサービスに渡される文字列は以下のとおりです。

uid=john.doe@example.com,ou=People,o=example.com

ディレクトリサービスからのNNMiユーザーアクセスの設定 (詳細な方法)

タスク3で説明した簡単な方法が正しく動作しない場合は、以下の手順を実行します。

1. 必要なユーザー情報をディレクトリサービス管理者から取得します。
2. 適切な手順を完了し、ディレクトリサービスにおけるユーザー名の形式を確認します。
   - Active Directoryおよびその他のディレクトリサービスの場合にLDAPブラウザーを使用する方法: 「ディレクトリサービスでユーザーを識別する方法の判別 (LDAPブラウザーを使用する方法)」を参照してください。
   - ほかのディレクトリサービスの場合にWebブラウザーを使用する方法: 「ディレクトリサービスでユーザーを識別する方法の判別 (Webブラウザーを使用する方法)」を参照してください。
3. 任意のテキストエディターでldap.propertiesファイルを開きます。

   ヒント: ldap.propertiesファイルの詳細については、「ldap.properties設定ファイルリファレンス」(401ページ)を参照してください。
4. java.naming.provider.urlパラメーターを、LDAPによってディレクトリサービスにアクセスする場合のURLに設定します。

- LDAPブラウザーを使用する方法: LDAPブラウザー設定からこの情報を入手します。
- Webブラウザーを使用する方法: 「ディレクトリサービスでユーザーを識別する方法の判別 (Webブラウザーを使用する方法)」から、<directory_service_host>と<port>の値を含めます。

ヒント: 複数のディレクトリサービスURLを指定するには、各URLをスペース文字1つで区切ります。

5. ディレクトリサービスへのセキュアー通信を設定した場合は、以下の行のコメントを解除 (または追加) します。

   `java.naming.security.protocol=ssl`

6. (Active Directoryのみ) `bindDN`および`bindCredential`パラメーターを以下のように設定します。
   - `<mydomain>`をActive Directoryドメインの名前で置き換えます。
   - `<myusername>`および`<mypassword>`をActive Directoryサーバーにアクセスするときに使用するユーザー名とパスワードで置き換えます。

     平文のパスワードを保存する場合は、ディレクトリサービスへの読み取り専用アクセス権を付与してユーザー名を指定してください。
     暗号化されたパスワードを指定する場合は、`ldap.properties`ファイルに保存する前に平文のパスワードを以下のコマンドで暗号化します。

     nnmldap.ovpl -encrypt <mypassword>

     注: この暗号化パスワードは、その作成先のNNMiインスタンスでのみ機能します。ほかのNNMiインスタンスには使用しないでください。

     詳細については、nnmldap.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

7. `baseCtxDN`パラメーターを、複数のユーザーで同じになっている、識別ユーザー名のエレメントに設定します。
8. NNMiのサインインで入力するときのユーザー名が、ディレクトリサービスでユーザー名が保存されるときの方法と相関するように、`baseFilter`パラメーターを設定します。

   この値は、ユーザーごとに変更される識別ユーザー名のエレメントです。実際のユーザー名を式`{0}`で置き換えます。
9. タスク4の説明に従って設定をテストします。

ディレクトリサービスでユーザーを識別する方法の判別 (LDAPブラウザーを使用する方法)

サードパーティのLDAPブラウザーで、以下の手順を実行します。

1. ディレクトリサーバードメインの中でグループ情報を保存する領域にナビゲートします。
2. ユーザーのグループを識別し、そのグループに関連付けられているユーザーの識別名の形式を調べます。

ディレクトリサービスでユーザーを識別する方法の判別 (Webブラウザーを使用する方法)

1. サポートされるWebブラウザーで、以下のURLを入力します。

   ldap://<directory_service_host>:<port>/<user_search_string>

   - <directory_service_host>は、ディレクトリサービスをホストするコンピューターの完全修飾名です。
   - <port>は、LDAP通信でディレクトリサービスが使用するポートです。
   - <user_search_string>は、ディレクトリサービスに保存される1つのユーザー名の識別名です。

2. ディレクトリサービスのアクセステストの結果を評価します。
   - 要求が時間切れになったり、ディレクトリサービスに到達できなかったことを示すメッセージが表示される場合は、<directory_service_host>と<port>の値を確認してから、手順1を繰り返してください。
   - ディレクトリサービスに要求されたエントリが存在しないことを示すメッセージが表示された場合は、<user_search_string>の値を確認してから、手順1の操作を繰り返してください。
   - 該当するユーザーレコードが表示された場合、そのアクセス情報は正しいことになります。<user_search_string>の値は、識別ユーザー名です。

ユーザーグループの識別

ユーザーグループ識別は、外部モードに適用されます。

NNMiは、NNMiユーザーのユーザーグループを以下のように判断します。

1. NNMiiは、NNMiコンソールで設定されているすべてのユーザーグループの外部名の値をディレクトリサービスグループの名前と比較します。
2. ユーザーグループが一致する場合、NNMiは、NNMiユーザーがディレクトリサービスのそのグループのメンバーであるかどうかを判断します。

NNMiコンソールで、短いテキスト文字列により、NNMiコンソールアクセスを許可する、定義済みのNNMiユーザーグループの一意の名前が識別されます。`ldap.properties`設定ファイルの`defaultRole`および`userRoleFilterList`パラメーターも、このテキスト文字列を必要とします。以下の表では、このグループの一意の名前を表示名にマッピングしています。

NNMiユーザーグループ名のマッピング

NNMiのロール名 NNMiコンソール	**NNMi設定ファイルのユーザーグループの一意の名前およびテキスト文字列**
管理者	admin
グローバルオペレーター	globalops
オペレーターレベル2	level2

NNMiユーザーグループ名のマッピング (続き)

NNMiのロール名 NNMiコンソール	NNMi設定ファイルのユーザーグループの一意の名前およびテキスト文字列
オペレーターレベル1	level1
ゲスト	ゲスト
Webサービスクライアント	クライアント

注: NNMiグローバルオペレーターユーザーグループ (globalops) では、すべてのトポロジオブジェクトのみにアクセス権が与えられます。ユーザーがNNMiコンソールにアクセスするには、ユーザーを他のいずれかのユーザーグループ (level2、 level1、または guest) に割り当てる必要があります。

globalopsユーザーグループはデフォルトですべてのセキュリティグループにマッピングされるため、管理者はこのユーザーグループをセキュリティグループにマッピングしないようにする必要があります。

ディレクトリサービスからのユーザーグループ取得の設定 (詳細な方法)

タスク5で説明した簡単な方法が正しく動作しない場合は、以下の手順を実行します。

1. 必要なユーザー情報をディレクトリサービス管理者から取得します。
2. 適切な手順を完了し、ディレクトリサービスにおけるグループ名およびグループメンバーの形式を確認します。
   - Active Directoryの場合にLDAPブラウザーを使用する方法: 「ディレクトリサービスでグループおよびグループメンバーシップを識別する方法の判別 (Active Directoryの場合にLDAPブラウザーを使用する方法)」を参照してください。
   - ほかのディレクトリサービスの場合にLDAPブラウザーを使用する方法: 「ディレクトリサービスでグループおよびグループメンバーシップを識別する方法の判別 (他のディレクトリサービスの場合にLDAPブラウザーを使用する方法)」を参照してください。
   - ほかのディレクトリサービスの場合にWebブラウザーを使用する方法: 「ディレクトリサービスでグループを識別する方法の判別 (Webブラウザーを使用する方法)」を参照してください。
3. 任意のテキストエディターでldap.propertiesファイルを開きます。

   ヒント: ldap.propertiesファイルの詳細については、 「ldap.properties設定ファイルリファレンス」(401ページ) を参照してください。

4. rolesCtxDNパラメーターを、複数のグループで同じになっている、識別グループ名のエレメントに設定します。
5. ディレクトリサービスでグループにユーザー名が保存されるときの方法とユーザー名が相関するように、roleFilterパラメーターを設定します。実際のユーザー名を以下の式のいずれかで置き換えます。
   - サインインのために入力されたユーザー名を意味する場合は{0}を使用します (たとえば、john.doe)。
   - ディレクトリサービスによって返された認証済みユーザーの識別名を意味する場合は、{1}を使用します (たとえば、uid=john.doe@example.com,ou=People,o=example.com)。
6. uidAttributeIDパラメーターを、ユーザーIDを保存するグループ属性の名前に設定します。
7. 「ディレクトリサービスにアクセスするNNMiの設定」(380ページ) の説明に従って設定をテストします。

ディレクトリサービスでグループおよびグループメンバーシップを識別する方法の判別 (Active Directoryの場合にLDAPブラウザーを使用する方法)

サードパーティのLDAPブラウザーで、以下の手順を実行します。

1. ディレクトリサーバードメインの中でユーザー情報を保存する領域にナビゲートします。
2. NNMiにアクセスする必要があるユーザーを識別し、そのユーザーに関連付けられているグループの識別名の形式を調べます。
3. ディレクトリサーバードメインの中でグループ情報を保存する領域にナビゲートします。
4. NNMiユーザーグループに対応するグループを識別して、グループに関連付けられているユーザーの名前の形式を調べます。

ディレクトリサービスでグループおよびグループメンバーシップを識別する方法の判別 (他のディレクトリサービスの場合にLDAPブラウザーを使用する方法)

サードパーティのLDAPブラウザーで、以下の手順を実行します。

1. ディレクトリサーバードメインの中でグループ情報を保存する領域にナビゲートします。
2. NNMiユーザーグループに対応するグループを識別して、それらのグループの識別名の形式を調べます。
3. また、グループに関連付けられているユーザーの名前の形式も調べます。

ディレクトリサービスでグループを識別する方法の判別 (Webブラウザーを使用する方法)

1. サポートされるWebブラウザーで、以下のURLを入力します。

   ldap://<directory_service_host>:<port>/<group_search_string>

   - <directory_service_host>は、ディレクトリサービスをホストするコンピューターの完全修飾名です。
   - <port>は、LDAP通信でディレクトリサービスが使用するポートです。

- <group_search_string>は、ディレクトリサービスに保存されるグループ名の識別名です (例: cn=USERS-NNMi-Admin,ou=Groups,o=example.com)。

2. ディレクトリサービスのアクセステストの結果を評価します。
   - ディレクトリサービスに要求されたエントリが存在しないことを示すメッセージが表示された場合は、<group_search_string>の値を確認してから、手順1の操作を繰り返してください。
   - 該当するグループのリストが表示された場合、そのアクセス情報は正しいことになります。
3. グループのプロパティを調べ、そのグループに関連付けられえているユーザーの名前の形式を判断してください。

NNMiユーザーグループを保存するディレクトリサービスの設定

NNMiユーザーグループをディレクトリサービスに保存する場合 (外部モード) は、NNMiユーザーグループ情報を使用してディレクトリサービスを設定する必要があります。原則として、ディレクトリサービスには適切なユーザーグループがすでに含まれています。含まれていない場合、ディレクトリサービス管理者は、特にNNMiユーザーグループ割り当て用の新規ユーザーグループを作成できます。

ディレクトリサービスの設定およびメンテナンス手順は、特定のディレクトリサービスソフトウェアと企業のポリシーに応じて異なるため、ここではそれらの手順について説明していません。

ディレクトリサービス統合のトラブルシューティング

1. 以下のコマンドを実行してNNMi LDAP設定を検証します。

   nnmldap.ovpl -info

   報告された設定が期待どおりの設定ではない場合は、ldap.propertiesファイルで設定を確認してください。
2. 以下のコマンドを実行して、NNMiにldap.propertiesファイルを再読み込みさせます。

   nnmldap.ovpl -reload
3. 以下のコマンドを実行し、あるユーザーの設定をテストします。

   nnmldap.ovpl -diagnose <NNMi_user>

   <NNMi_user>は、ディレクトリサービスで定義したNNMiユーザーのサインイン名で置き換えます。

   コマンド出力を検討し、適切に応答します。

4. ディレクトリサービスに期待されるレコードが含まれていることを確認します。Webブラウザーまたはサードパーティのldapブラウザー (Apache Directory Studioに含まれるLDAPブラウザーなど) を使用して、ディレクトリサービスの情報を調べます。

   ディレクトリサービスに対するクエリーの形式に関する詳細については、以下で入手できるRFC 1959「An LDAP URL Format」を参照してください。

   http://labs.apache.org/webarch/uri/rfc/rfc1959.txt

5. ログファイルを表示し、サインイン要求が正しいことを確認して、エラーが発生しているかどうかを判断します。

   Windowsの場合: `%NnmDataDir%\log\nnm\nnm.log`

   Linuxの場合: `$NnmDataDir/log/nnm/nnm.log`

   - 以下の行のようなメッセージは、ディレクトリサービスでHTTPS通信が必要であることを示しています。この場合は、「ディレクトリサービスへのSSL接続を設定する」(331ページ)の説明に従ってSSLを有効にします。

     javax.naming.AuthenticationNotSupportedException:[LDAP: error code 13 - confidentiality required]

   - 以下の行のようなメッセージは、ディレクトリサービスとのやり取り中にタイムアウトが発生したことを示します。この場合は、`nms-ldap.properties`ファイルの`searchTimeLimit`の値を増やします。

     javax.naming.TimeLimitExceededException:[LDAP: error code 3 - Timelimit Exceeded]

ldap.properties設定ファイルリファレンス

`ldap.properties`ファイルには、ディレクトリサービスと通信し、それに対するLDAP照会を作成する場合の設定が保存されています。このファイルは以下の場所にあります。

- Windowsの場合: `%NNM_SHARED_CONF%\ldap.properties`
- Linuxの場合: `$NNM_SHARED_CONF/ldap.properties`

`ldap.properties`ファイルでは、以下の規則が適用されます。

- 行をコメントアウトするには、その行の先頭を番号記号文字 (#) にします。
- 特殊文字には、以下のルールが適用されます。
  - バックスラッシュ文字 (\)、カンマ (,)、セミコロン (;)、プラス記号 (+)、小なり記号 (<)、大なり記号 (>) を指定するには、バックスラッシュ文字でエスケープします。次に例を示します。
  - 文字列の先頭文字または末尾文字としてスペース文字 ( ) を含めるには、バックスラッシュ文字 (\) でエスケープします。
  - 文字列の先頭文字としてシャープ記号 (#) を含めるには、バックスラッシュ文字 (\) でエスケープします。

  ここで言及していない文字をエスケープしたり、引用符で囲んだりする必要はありません。

注: `ldap.properties`ファイルを編集したら、以下のコマンドを実行してNNMiにLDAP設定を再読み込みさせます。

nnmldap.ovpl -reload

以下の表に、`ldap.properties`ファイルのパラメーターの説明を示します。

注: 初期の`ldap.properties`ファイルには、以下の表にリストされたパラメーターの一部が含まれていない場合があります。必要なパラメーターを追加してください。

ldap.propertiesファイルのパラメーター

パラメーター	説明
java.naming.provider.url	ディレクトリサービスにアクセスするときのURLを指定します。 URLは、プロトコル (ldap) の後にディレクトリサービスの完全修飾ホスト名が続き、必要に応じてさらにポート番号が続く形式で指定します。例: java.naming.provider.url=ldap://ldap.example.com:389/ ポート番号を省略すると、以下のデフォルト値が適用されます。 • 非SSL接続の場合、デフォルト値は389です。 • SSL接続の場合、デフォルト値は636です。 複数のディレクトリサービスのURLを指定すると、NNMiは可能な限り最初のディレクトリサービスを使用します。そのディレクトリサービスにアクセスできない場合、NNMiはリスト内の次のディレクトリサービスにクエリーを実行し、以下同様に対処します。各URLは1つのスペース文字で区切ります。例: java.naming.provider.url=ldap://ldap1.example.com/ ldap://ldap2.example.com/ このパラメーターを設定すると、NNMiとディレクトリサービス間のLDAP通信が有効になります。LDAP通信を無効にするには、このパラメーターをコメントアウトしてからファイルを保存します。NNMiは、`ldap.properties`ファイルの設定を無視します。
	接続プロトコル指定を指定します。 • LDAP over SSLを使用するようにディレクトリサーバーが設定されている場合は、このパラメーターを`ssl`に設定します。次に例を示します。 `java.naming.security.protocol=ssl` • ディレクトリサービスでSSLが不要な場合は、このパラメーターをコメントアウトしたままにします。 詳細については、「ディレクトリサービスへのSSL接続を設定する」(331

ldap.propertiesファイルのパラメーター (続き)

パラメーター	説明
	ページ)を参照してください。
bindDN	匿名アクセスを許可しない (Active Directoryなどの) ディレクトリサービスの場合は、そのディレクトリサービスにアクセスするユーザー名を指定します。 例: bindDN=region1\\john.doe@example.com • 平文のパスワードを保存する場合は、ディレクトリサービスへの読み取り専用アクセス権を付与してユーザー名を指定してください。 例: bindCredential=PasswordForJohnDoe • 暗号化されたパスワードを指定する場合は、ldap.propertiesファイルに保存する前に平文のパスワードを以下のコマンドで暗号化します。 **nnmldap.ovpl -encrypt <mypassword>** 次に例を示します。bindCredential={ENC} uaF22C+0CF9VozBVYj8OAw== この暗号化パスワードは、その作成先のNNMiインスタンスでのみ機能します。他のNNMiインスタンスには使用しないでください。 詳細については、nnmldap.ovplのリファレンスページ、またはUNIXのマンページを参照してください。
bindCredential	bindDNが設定されている場合は、そのbindDNによって識別されるユーザー名のパスワードを指定します。次に例を示します。 bindCredential=PasswordForJohnDoe
baseCtxDN	ディレクトリサーバードメインの中でユーザーレコードを保存する部分を指定します。 形式は、ディレクトリサービスの属性名と値のカンマ区切りリストです。次に例を示します。 • baseCtxDN=CN=Users,DC=ldapserver,DC=example,DC=com • baseCtxDN=ou=People,o=example.com 詳細については、「ユーザー識別」(394ページ)を参照してください。
baseFilter	NNMiにサインインするユーザー名の形式を指定します。 形式は、ディレクトリサービスのユーザー名属性の名前と、入力したユーザーサインイン名をディレクトリサービス内の名前の形式に関連付ける文字列で構成されます。ユーザー名文字列には、式 {0} (サインイン

ldap.propertiesファイルのパラメーター (続き)

パラメーター	説明

	で入力されたユーザー名を示す) と、ユーザー名のディレクトリサービス形式を照合するために必要なほかの文字が含まれます。
	• NNMiのサインインで入力されたユーザー名がディレクトリサービスに保存されているユーザー名と同じ場合、値は置換表現になります。次に例を示します。
	  • `baseFilter=CN={0}`
	  • `baseFilter=uid={0}`
	• NNMiのサインインで入力したユーザー名がディレクトリサービスに保存されているユーザー名のサブセットになっている場合は、値に追加の文字を含めます。次に例を示します。
	  • `baseFilter=CN={0}@example.com`
	  • `baseFilter=uid={0}@example.com`
	詳細については、「ユーザー識別」(394ページ)を参照してください。

defaultRole	省略可能。LDAPに従ってNNMiにサインインするディレクトリサービスユーザーすべてに適用されるデフォルトロールを指定します。このパラメーターの値は、(NNMiデータベースまたはディレクトリサービスでの) ユーザーグループマッピングの保存場所に関係なく適用されます。 定義済みのNNMiユーザーグループにユーザーが直接設定されている場合、NNMiは、デフォルトロールおよび割り当て済みユーザーグループの権限のスーパーセットをユーザーに付与します。 有効な値は、`admin`、`level2`、`level1`、または`guest`です。 `admin` は有効な値ですが、デフォルトロールとしての`admin` の使用は慎重に検討する必要があります。 この名前は、定義済みNNMiユーザーグループ名の一意の名前です。 次に例を示します。 `defaultRole=guest` コメントアウトまたは省略すると、NNMiはデフォルトロールを使用しません。
rolesCtxDN	ディレクトリサーバードメインの中でグループレコードを保存する部分を指定します。 形式は、ディレクトリサービスの属性名と値のカンマ区切りリストです。次に例を示します。

ldap.propertiesファイルのパラメーター (続き)

パラメーター	説明
	• rolesCtxDN=CN=Users,DC=ldapserver,DC=example,DC=com • rolesCtxDN=ou=Groups,o=example.com 他のディレクトリサービス (Active Directory以外) では、検索速度を高めるため、NNMiユーザーグループを含むディレクトリサービスグループを1つ以上指定できます。グループ名にパターンがある場合は、ワイルドカードを指定できます。たとえば、ディレクトリサービスにUSERS-NNMi-administratorsやUSERS-NNMi-level1Operatorsなどの名前のグループが含まれる場合は、以下のような検索コンテキストを使用できます。 rolesCtxDN=cn=USERS-NNMi-*,ou=Groups,o=example.com このパラメーターを設定すると、LDAPを介したNNMiユーザーグループ割り当てのディレクトリサービスの照会が有効になります。 LDAPを介したNNMiユーザーグループ割り当てのディレクトリサービスの照会を無効にするには、このパラメーターをコメントアウトしてからファイルを保存します。NNMiは、ldap.propertiesファイルにある残りのユーザーグループ関連の値を無視します。 詳細については、「ユーザーグループの識別」(397ページ)を参照してください。
roleFilter	ディレクトリサービスのグループ定義でグループメンバー名の形式を指定します。 形式は、ユーザーIDのディレクトリサービスグループ属性の名前と、入力したユーザーサインイン名をディレクトリサービス内のユーザーIDの形式に関連付ける文字列で構成されます。ユーザー名文字列には、以下の式の1つと、グループメンバー名のディレクトリサービス形式を照合するために必要な他の文字が含まれています。 • 式{0}は、サインインで入力されたユーザー名を示します (たとえば、john.doe)。 サインインで入力される (短い) ユーザー名で照合するロールフィルター例: roleFilter=member={0} • 式{1}は、ディレクトリサービスによって返された認証済みユーザーの識別名を意味します (たとえば、 CN=john.doe@example.com,OU=Users,OU=Accounts, DC=example,DC=com または uid=john.doe@example.com,ou=People,o=example.com)。 (完全に) 認証されたユーザー名で照合するロールフィルター例:

ldap.propertiesファイルのパラメーター (続き)

パラメーター	説明
	roleFilter=member={1} 詳細については、「ユーザーグループの識別」(397ページ)を参照してください。
uidAttributeID	ディレクトリサービスユーザーIDを保存するグループ属性を指定します。 次に例を示します。 uidAttributeID=member 詳細については、「ユーザーグループの識別」(397ページ)を参照してください。
userRoleFilterList	省略可能。NNMiコンソールで関連ユーザーにインシデントを割り当てることができるNNMiユーザーグループを制限します。 このリストのユーザーグループは、LDAPで認証されるディレクトリサービスユーザー名のみに適用されます。このパラメーターでは、NNMiユーザーグループがNNMiコンソールで割り当てられて、NNMiデータベースに保存されるときに使用できない機能が提供されます。 1つ以上の定義済みNNMiユーザーグループ名の一意の名前をセミコロンで区切ったリストという形式です。 userRoleFilterList=admin;globalops;level2;level1
searchTimeLimit	省略可能。タイムアウト値をミリ秒単位で指定します。デフォルト値は10000 (10秒) です。NNMiユーザーサインイン中にタイムアウトになる場合は、この値を増やします。 次に例を示します。 searchTimeLimit=10000

例

Active Directoryの場合のldap.propatiesファイルの例

Active Directoryの場合の`ldap.properties`ファイルの例を以下に示します。

java.naming.provider.url=ldap://MYldapserver.example.com:389/

bindDN=MYdomain\\MYusername

bindCredential=MYpassword

baseCtxDN=CN=Users,DC=MYldapserver,DC=EXAMPLE,DC=com

baseFilter=CN={0}

defaultRole=guest

rolesCtxDN=CN=Users,DC=MYldapserver,DC=EXAMPLE,DC=com

rolesCtxDN=CN=Users,DC=MYldapserver,DC=EXAMPLE,DC=com

roleFilter=member={1}

uidAttributeID=member

userRoleFilterList=admin;level2;level1

他のディレクトリサービスの場合のldap.propatiesファイルの例

他のディレクトリサービスの場合の`ldap.properties`ファイルの例を以下に示します。

java.naming.provider.url=ldap://MYldapserver.example.com:389/

baseCtxDN=ou=People,o=EXAMPLE.com

baseFilter=uid={0}

defaultRole=guest

rolesCtxDN=ou=Groups,o=EXAMPLE.com

roleFilter=member={1}

uidAttributeID=member

userRoleFilterList=admin;level2;level1

NAT環境の重複IPアドレスの管理

NNMiでは、ネットワークアドレス変換 (NAT) ドメインの実装 (重複IPアドレスが生じる可能性があり、NAT内部/外部IPアドレスのペアを処理するNNMi設定が必要になる) を含むネットワークエリアを容易に管理できます。NNMi管理者は、テナント定義を作成して各NATドメインを識別します。NNMiでは、テナント/IPアドレスのペアを使用して各ノードを識別します。アドレスは、1つのテナントのノードグループ内で重複していない限り、重複しているとはみなされません。

注: NATドメイン統合のコンテキスト外の重複IPアドレス: 重複IPアドレス/MACアドレスのあるファイアウォールまたはロードバランサーデバイス (物理デバイスでホストされている仮想インスタンスなど) がネットワークにある場合。NNMi管理者は、ファイアウォールおよびロードバランサーのsysObjectId値を設定ファイルに入力します。これにより、NNMiは、(同じノードオブジェクトのようにすべてマージするのではなく) それらのsysObjectId値を持つノードオブジェクトの各インスタンスを正常に認識できるようになります。

NATとは

通常、ネットワークアドレス変換 (NAT) は、ローカルネットワークを外部 (パブリック) インターネットと相互接続するために使用します。具体的に言うと、NATではIPヘッダー情報を変換します (パブ

リックネットワークを通過する必要があるIPパケットの内部アドレスを外部 (パブリック) アドレスに置き換えます)。NATでは、静的または動的な外部IPアドレスを使用することによりこれを実現します。ネットワークアドレス変換はインターネットセキュリティの手段として使用されますが、送信者のIPアドレスをインターネットアクセスに使用しません。

ネットワークアドレス変換テクノロジは、より多くのIPv4アドレスを求めるニーズの高まりに対応するソリューションとして開発されました。IPアドレスの特定範囲 (RFC 1918を参照) は、内部専用として設計されています (インターネット上ではルーティングできません)。プライベートネットワークにこれらのアドレスを使用して、購入が必要なパブリックアドレスの数を削減できます。

NATの利点

NATには、以下のような利点があります。

- プライベートIPアドレスを再利用できる
- 内部アドレスを外部ネットワークから隠ぺいすることで、プライベートネットワークのセキュリティが強化される
- 多数のホストを少数のパブリック (外部) IPアドレスを使用してグローバルインターネットに接続するため、IPアドレス空間を節約できる

サポートされるNATタイプ

NNMiでは、以下のタイプのNATプロトコルがサポートされます。

- 静的NAT — 内部IPアドレスが、常に同じ外部IPアドレスにマップされるNATタイプ (各ノードは静的な内部/外部アドレスペアを持つ)。このタイプでは、Webサーバーなどの内部ホストにプライベート IPアドレスを割り当てたまま、インターネット上で到達可能な状態にすることができます。
- 動的NAT—外部IPアドレスと 内部IPアドレスの間にマップされるNATタイプ(セッションで変更できる)。内部IPアドレスは、使用可能なパブリックIPアドレスのプールから引き出されて、外部IPアドレスに動的にマップされます。通常、ネットワークのNATゲートウェイルーターで登録済みパブリックIPアドレスのテーブルが保持されています。内部IPアドレスからインターネットへのアクセスが要求されると、別の内部IPアドレスで現在使用されていないIPアドレスがルーターによって選択されます。
- 動的ポートアドレス変換 (PAT、ネットワークアドレスおよびポート変換 (NAPT) ともいう)—外部IPアドレスだけではなく、動的にポート番号も提供するNATタイプ。アドレスとポート番号を変換することで、複数の内部アドレスが1つの外部アドレスを使用してインターネット上で同時に通信できるようになります。

NNMiにNATを実装する方法

NNMiでは、テナント/IPアドレスのペアを使用して各ノードを識別することによって、NAT環境を管理します。NNMi管理者は、NATアドレスドメインごとにテナント定義を作成します。テナントにより、ノードの論理グループが識別されます。たとえば、インターネットプロバイダーのネットワークに、プライベートIPアドレスを実装した顧客が複数存在するとします。インターネットプロバイダーは、NNMi内で各顧客のノードを、個々の顧客を識別する特定のテナント名に割り当てることができます。そのテナントの論理グループ内では、以下のようになります。

- NNMi管理者は、検出シードを使用して、テナント/IPアドレスのペアを使用するテナントメンバーのノードを識別します。
- サブネット接続ルールは、各テナントのノードグループ内で独立して適用されます。
- ルーター冗長グループは、ほかのテナントノードグループから独立し、各テナント内でモニタリングされます。
- NNMiは、各テナントのノードグループ内、および定義済みのそのテナントのノードとデフォルトテナントに割り当てられたノード間でのみL2接続を検出します。
- 複数のNATドメイン (NATゲートウェイルーターなど) と相互接続するインフラストラクチャーデバイスは、すべてデフォルトテナントに割り当てます。これにより、ワークグループ (および顧客) が確認する必要があるレイヤー2接続がNNMiに表示されるようになります。
- NNMiユーザーが表示できるテナント数は、セキュリティグループによって決まります。割り当てられたセキュリティグループには、複数のテナントのノードを含めることができます。詳細については、「NNMiのセキュリティおよびマルチテナント設定」(438ページ)を参照してください。

ヒント: ネットワーク管理環境のすべてのNATドメインで、ドメインネームシステム (DNS) 名が重複しないようにすることを推奨します。

使用しているNATプロトコルによって、NNMiの実装方法や要件が異なる場合があります。たとえば、動的NATまたは動的PATを使用している場合、追加のハードウェアおよびライセンスが必要になります。NATプロトコルのタイプに基づいて、適切なセクションを参照してください。

- 「静的NATの考慮事項」(409ページ)
- 「動的NATおよび動的PATの考慮事項」(419ページ)

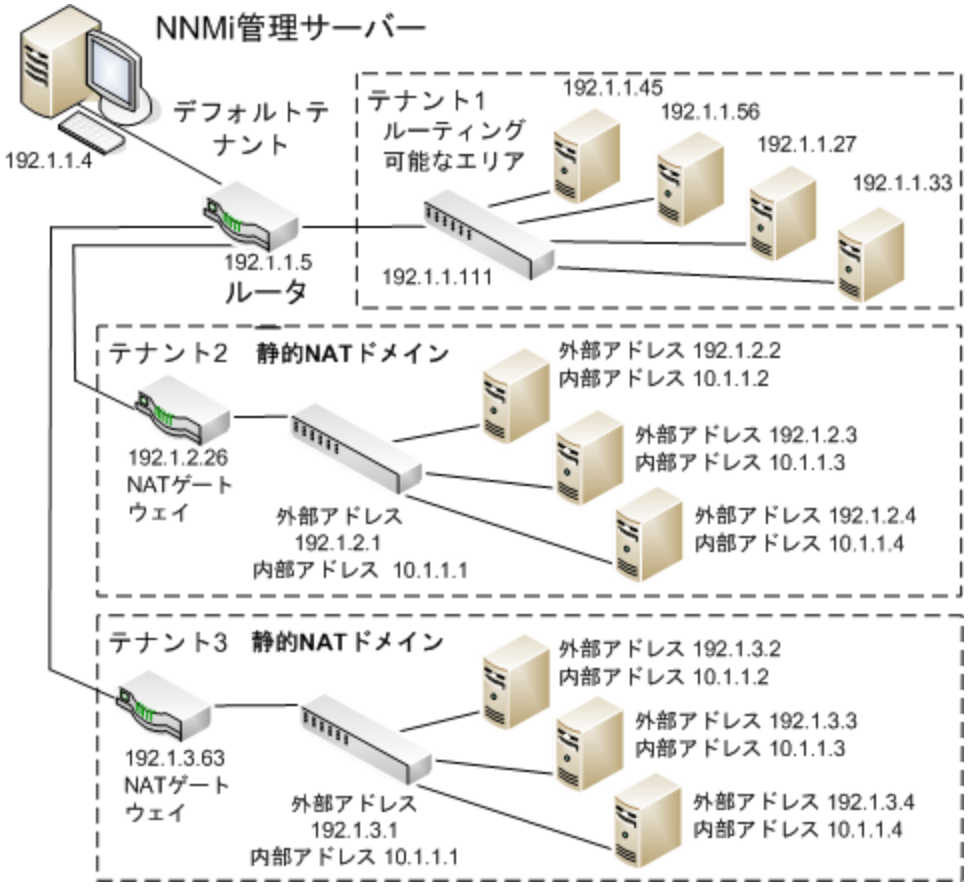
詳細については、「ネットワークアドレス変換 (NAT) 環境でのNNMiの配備」(423ページ) を参照してください。

静的NATの考慮事項

各インスタンスが一意のテナントで設定されていれば、1つのNNMi管理サーバーで任意の数の静的NATインスタンスを監視できます。テナントの詳細については、「NNMiセキュリティおよびマルチテナント」(426ページ) およびNNMiヘルプの「テナントを設定する」を参照してください。

静的NATの設定例として以下の図を参照してください。

静的NATの設定例



注: デフォルトテナントに属するノードは、任意のテナントの任意のノードにレイヤー2接続できます。デフォルトテナント以外のテナント内のノードは、同じテナントかデフォルトテナント内のデバイスにしかレイヤー2接続できません。

サブネットはテナントに固有です (サブネットは複数のテナントにまたがらない)。このメリットは、同じサブネットを異なるテナントで使用できる点にあります。

ルーター冗長グループ (RRG) はテナントをまたぐことができません。

ヒント: 複数のNATドメイン (NATゲートウェイなど) と相互接続するインフラストラクチャーデバイスは、すべてデフォルトテナントに割り当てます。これにより、ワークグループ (および顧客) が確認する必要があるレイヤー2接続がNNMiに表示されるようになります。

注: デフォルトセキュリティグループ内のデバイスはすべてのビューで表示されます。デバイスへのアクセスを制御するには、該当するデバイスをデフォルトセキュリティグループ以外のセキュリティグループに割り当てます。

静的NATのハードウェアとソフトウェアの要件

静的NATドメインの管理には、特別なハードウェアまたはソフトウェアの要件はありません。1つのNNMi管理サーバーで、NNMi、NNMi Advanced、NNMi Premium、またはNNMi Ultimateを使用する静的NATドメインをいくつでも管理できます。

重複するIPアドレスマッピング

NNMi管理サーバーが静的NATドメイン外に存在する場合は、各静的NAT内部/外部IPアドレスペアの識別に重複するアドレスマッピングを使用するメリットがあります。NNMiは、以下の方法で静的NATドメインのマッピングの外部アドレス/内部アドレスペアを使用します。

- ノードフォームに[マップされたアドレス]属性値が表示されます。
- 通信の設定とモニタリングの設定が拡張されます。これによって、NNMiは、各静的NATノードのSNMPエージェントおよび管理対象IPアドレスの状態とステータスを確実に正しく計算できます(「状態とステータスのNNMi計算」(425ページ)も参照)。
  - NNMiは、ICMP障害モニタリングのIPアドレス障害ポーリングにモニタリングの設定を正確に使用できます。
  - NNMiは、ICMP ping要求を使用して(SNMP照会に加えて)、非SNMPノードにレイヤー2およびレイヤー3が接続できるか正確に判定できます。
- トラップがNATドメインから発生する場合は、NNMiでSNMPトラップのソースノードを正確に判定できます。SNMPv1がネットワークで使用されている場合は、「静的NAT環境のSNMPトラップ」(SNMP Traps in Static NAT Environments)(240ページ)も参照してください。
- カスタムインシデント属性が正確に計算されます。
  - `cia.agentAddress` = 外部IPアドレス(パブリックアドレス)。
  - `cia.internalAddress` = インシデントのソースノードの内部IPアドレス。

注: 動的NATまたは動的PATを使用しているネットワーク管理ドメインのエリアに対してNNMiを設定している場合、[重複するIPアドレスマッピング]フォームは使用しないでください。「動的NATおよび動的PATの考慮事項」(419ページ)を参照してください。

プライベートIPアドレスの範囲

Internet Engineering Task Force (IETF) およびInternet Assigned Numbers Authority (IANA) では、以下のIPアドレス範囲をプライベートネットワーク(企業のローカルエリアネットワーク(LAN)、企業のオフィス、または住宅用のネットワークなど)用に予約しています。

IPv4プライベートアドレス範囲(RFC 1918):

- 10.0.0.0 ~ 10.255.255.255 (24ビットブロック)
- 172.16.0.0 ~ 172.31.255.255 (20ビットブロック)
- 192.168.0.0 ~ 192.168.255.255 (16ビットブロック)

IPv6プライベートアドレス範囲:

- fc00::/7アドレスブロック = RFC 4193ユニークローカルアドレス (ULA)
- fec0::/10アドレスブロック = 非推奨 (RFC 3879)

静的NATでの通信

NNMiでは、使用可能な重複するアドレスマッピングを自動的に使用して静的NAT通信用のテナント/外部IPアドレスのペアを識別することにより、静的NATファイアウォールを通して正常な通信が行われます。この利点については、「重複するIPアドレスマッピング」(411ページ) を参照してください。

静的NAT環境における管理アドレスのICMPポーリングの管理

NAT環境では、ファイアウォールにより、NNMiがノードのIPアドレス (プライベートIPアドレス) を使用してNATノードとやり取りすることがブロックされます。これを解決するには、NATアドレス (パブリックIPアドレス) を使用してNNMiと通信します。

NAT環境では、ノードの管理アドレスが、ノードでホストされるIPアドレスと異なることがあります。NNMiがNAT環境でノードを検出できるようにするには、NATアドレスを検出シードとしてNNMiに追加する必要があります。NNMiは、このNATアドレスがノードの`ipAddressTable`に存在しなくても、それを通信に使用します。

NNMiはこの機能を提供することで、誤ったノード停止中インシデントの生成を回避し、根本原因分析をより正確にします。

NAT環境における管理アドレスのICMPポーリングの有効化

NNMiでは、NAT環境に存在するノードも含めてすべてのノードのICMP管理アドレスポーリングがデフォルトで自動的に有効になります。NAT環境がある場合、この設定を無効にしないことをお勧めします。

(無効になっている場合に) 管理アドレスのICMPポーリングを有効にするには、以下の手順を実行します。

1. ワークスペースのナビゲーションパネルで、[**設定**] ワークスペースを選択して [**モニタリング**] フォルダーを展開し、[**モニタリングの設定**] を選択して [**デフォルト設定**] タブを探します。
2. [ICMP管 理アドレスポーリング] を有効にします。NNMiヘルプの「デフォルトのモニタリングを設定する」を参照してください。

SNMPエージェントに対して [**アクション**] > [**モニタリングの設定**] を実行した後にNNMiが表示する情報を確認します。表示される情報に、NNMiが管理アドレスのポーリングを有効にしているかどうかが示されます。

ICMP管理アドレスポーリングが有効になっていると、NNMiが以下のように変更されます。

- [エージェントICMP状態] フィールドが、以下のフォームに表示されます。
  - [ノード] フォーム
  - [SNMPエージェント] フォーム
  - [SNMPエージェント] テーブルビュー
- NNMiは、管理アドレスICMP状態の表示場所を変更します。NNMiは、SNMPエージェントステータスの判断方法も変更します。

以下の表に、エージェントICMPおよびIPアドレス状態のポーリングアクションを示します。NNMiは、ICMP管理アドレスポーリング設定およびICMP障害ポーリング設定に応じて、これらのアクションを実行します。

ICMP設定および結果の状態ポーリング

ICMP管理アドレスポーリング	ICMP障害ポーリング	エージェントICMP状態	IPアドレス状態
有効	無効	ポーリング	未ポーリング
有効	有効	ポーリング	ポーリング
無効	無効	未ポーリング	未ポーリング
無効	有効	未ポーリング	ポーリング

以下の表に、SNMPエージェントとICMPの応答に合わせてAPAによって決定されるSNMPエージェントステータスに対する変更点を示します。

SNMPエージェントステータスの判断

SNMPエージェント応答	管理アドレスICMP応答	SNMPエージェントステータス
応答	応答	正常域
応答	無応答	警戒域
無応答	応答	危険域
無応答	無応答	危険域

管理アドレスのICMPポーリングを有効にすると、APAは、結果とインシデントの生成時に、管理アドレスICMPの応答とSNMPエージェントの応答を考慮するようになります。

検出と静的NAT

NNMi管理者は、ネットワーク管理環境内の各静的NATドメインを識別するためにテナント定義を作成する必要があります。

スパイラル検出では、NATドメイン内の各ノードを識別するために検出シード (テナントとIPアドレスのペア) が必要です。NNMi管理者は、静的NATドメインのノードごとに検出シードを作成する必要があります。検出シードでは、ノードごとに以下の情報を指定する必要があります。

- 外部IPアドレス (外部/内部IPアドレスペアのパブリックアドレス)
- テナント名

詳細については、NNMiヘルプを参照してください。

注: 検出シードを静的NAT環境内に追加する場合 (`nnmloadseeds.ovpl`コマンドまたはNNMiコンソールを使用)、必ずノードの外部 (パブリック) IPアドレスを使用してください。詳細については、nnmloadseeds.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

ヒント: ドメインネームシステム (DNS) 名が重複しないようにすることをお勧めします。

静的NATのモニタリングの設定

ネットワーク環境によって、NNMi管理者はICMP障害モニタリングの設定の使用を選択できます (「状態とステータスのNNMi計算」(425ページ)も参照)。

- [**モニタリングの設定**] > [**ノードの設定**] タブ。ノードグループのモニタリングを設定します。[ICMP障害モニタリング] セクションで選択します (詳細については、NNMiオンラインヘルプを参照してください)。
  - 管理アドレスポーリング (デフォルトで有効な、強く推奨される機能)
  - IPアドレス障害ポーリング (省略可能)
- [**モニタリングの設定**] > [**デフォルト設定**] タブ。[ICMP障害モニタリング] セクションで選択します (詳細については、NNMiオンラインヘルプを参照してください)。

注: ネットワーク環境に動的NATドメインも設定されている場合、動的NATドメインとは異なる設定が静的NATドメインで必要になることがあるため、デフォルト設定が適切でない可能性があります。

トラップと静的NAT

NNMi管理サーバーでNATゲートウェイの背後にあるノードからSNMPトラップを受信するには、管理対象ノードを変更する必要があります。このセクションでは、SNMPv2cとSNMPv1の2種類のSNMPトラップについて説明します。

NNMiでは、受信した各トラップのソースアドレスを一義的に解決する必要があります。

SNMPv2cトラップ

以下の表に、SNMPv2cトラップの形式を示します。この表の上部のセクションはIPヘッダー、下部のセクションはSNMPトラップのProtocol Data Unit (PDU) で構成されています。

SNMPv2cトラップの形式

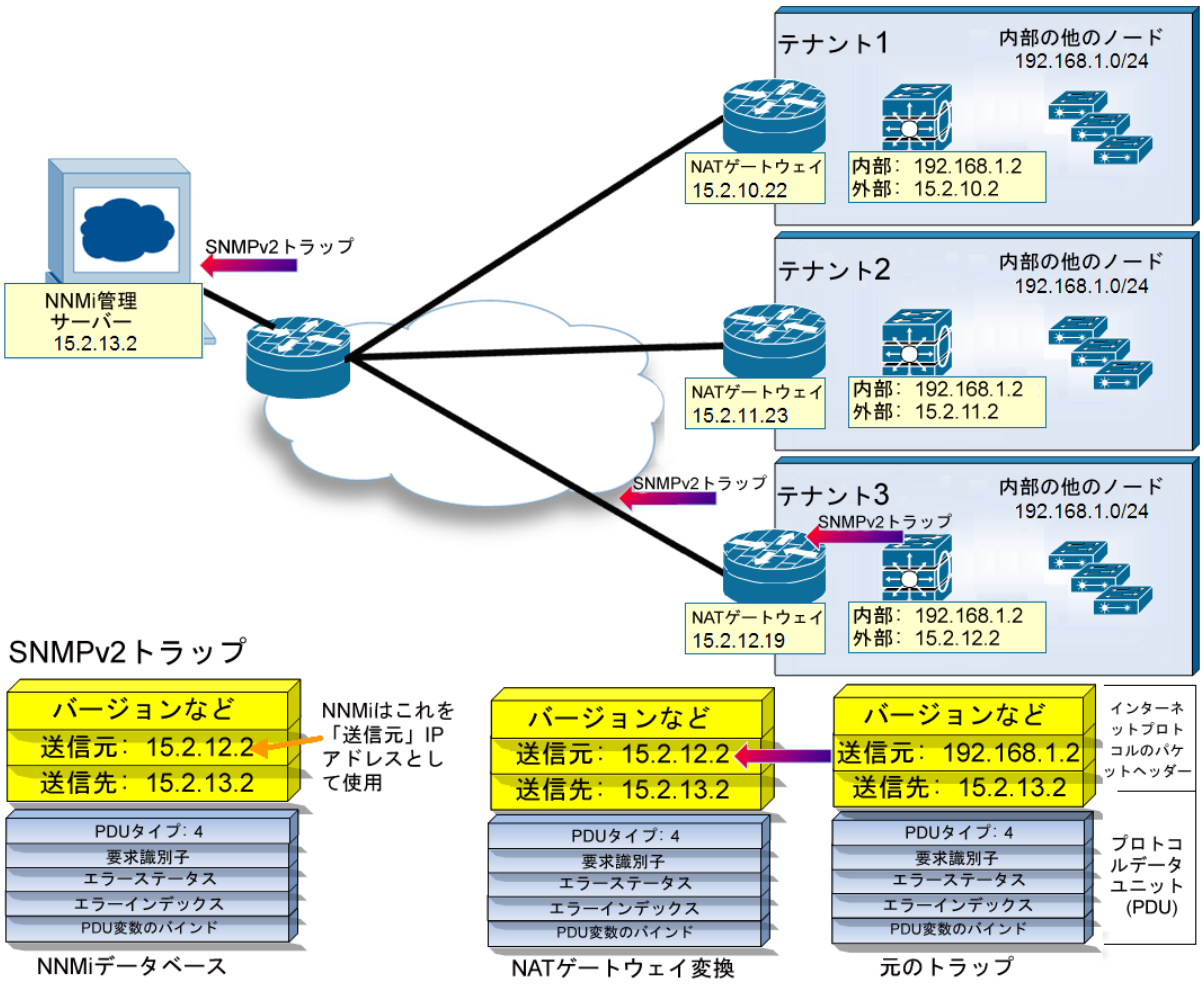
バージョンおよびその他の情報
送信元アドレス
デスティネーションアドレス
PDUタイプ:4
要求識別子
エラーステータス
エラーインデックス
PDU変数のバインド

SNMPv2cトラップのPDUには、エージェントアドレスフィールドがありません。そのため、IPパケットヘッダー内にはトラップのソースフィールドのみがあります。ソースフィールドは、NATルーターによって適切に変換されます。

ソースノードのプライベート内部IPアドレスに関連付けられているインタフェースで、NATルーターの背後にあるデバイスのすべてのトラップのソースが明らかになっていることを確認します。これで、NATゲートウェイがトラップを適切なパブリックアドレスに変換できます。

以下の図に、NATゲートウェイからの適切な変換の例を示します。NATゲートウェイによって、192.168.1.2のソースアドレスで始まるトラップのアドレスが15.2.13.2に適切に変換されます。次に、NNMi管理サーバーによってこのアドレスが適切に解決されます。

SNMPv2cの例



SNMPv1トラップ

SNMPv1トラップの場合、SNMPトラップのPDU内にエージェントアドレスが組み込まれています。以下の表に、SNMPv1トラップの形式を示します。上部のセクションはIPヘッダー、下部のセクションはSNMPトラップのPDUで構成されています。

SNMPv1トラップの形式

バージョンおよびその他の情報
送信元アドレス
デスティネーションアドレス
PDUタイプ:4
エンタープライズ

バージョンおよびその他の情報
送信元アドレス
デスティネーションアドレス
エージェントアドレス
汎用トラップコード
固有トラップコード
タイムスタンプ
PDU変数のバインド

エージェントアドレスはヘッダーではなくPDUに組み込まれているため、通常、この値はNATルーターによって変換されません。ヘッダーのアドレスを認識して、ペイロードのエージェントアドレスを無視するようにNNMiを設定するには、以下の手順を実行します。

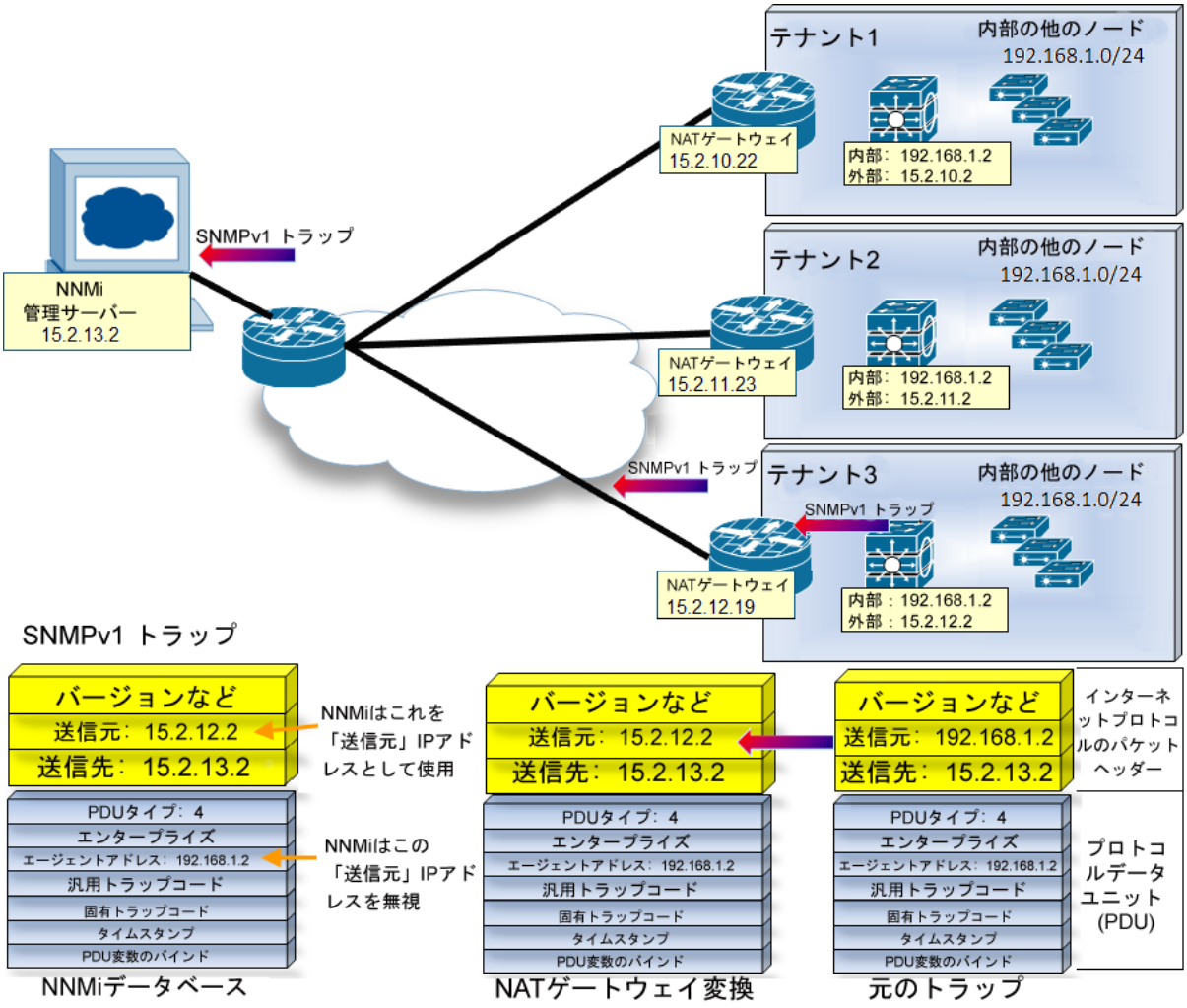
1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\nms-jboss.properties`
   - UNIXの場合: `$NNM_PROPS/nms-jboss.properties`
2. 以下の行を探します。

   `#!com.hp.nnm.trapd.useUdpHeaderIpAddress=false`
3. 以下のように値を**true**に変更して#!文字を削除します。

   `com.hp.nnm.trapd.useUdpHeaderIpAddress=true`
4. ファイルを保存してNNMiを再起動します。

以下の図に、競合するIPアドレスフィールドがNNMiで無視されるSNMPv1トラップの例を示します。

SNMPv1の例



注: NNMiでは、関連する以下のカスタムインシデント属性 (CIA) が提供されます。

- cia.agentAddress — トラップを生成したSNMPエージェントのSNMPv1トラップデータに保存されるIPアドレス。
- cia.internalAddress — 静的NATがネットワーク管理ドメインに含まれている場合、NNMi管理者は、選択したインシデントのソースノードの外部管理アドレスにマップされる内部IPアドレスを表示するようにこの属性を設定できます。

  [重複するIPアドレスマッピング] フォームを使用して、この内部アドレス (プライベートアドレス) に外部管理IPアドレス (パブリックアドレス) をマップする必要があります。詳細については、NNMiヘルプを参照してください。

サブネットと静的NAT

サブネットおよびNATに関しては、以下に注意してください。

- サブネットはテナントに固有です (サブネットは複数のテナントにまたがらない)。このメリットは、同じサブネットを異なるテナントで使用できる点にあります。
- サブネットフィルターではテナントとアドレスのペアが使用されます。
- サブネット接続ルールを設定する場合、そのルールはすべてのテナントに適用されます。サブネットのメンバーは、すべてのテナントで一意である必要があります (各ノードは1つのテナントにのみ割り当てられます)。サブネット接続ルールで、デフォルトテナントと別のテナント間にリンクを確立できます。ただし、2つのテナント間のリンクは、いずれかのテナントがデフォルトテナントである場合にのみ使用できます。

グローバルネットワーク管理:静的NATで任意

NNMiのグローバルネットワーク管理機能は、静的NATドメインの管理では任意です。1つのNNMi管理サーバーがあれば、静的NATドメインをいくつでも管理できます。

グローバルマネージャーとリージョナルマネージャーを使用する場合は、リージョナルマネージャーごとに、少なくとも1つの静的またはルーティング可能 (非変換) アドレスが存在している必要があります。これにより、NNMi管理サーバーが相互に通信することができ、通信を隠ぺいしてセキュリティを確保できます。グローバルネットワーク管理の詳細については、「グローバルネットワーク管理」(452ページ) を参照してください。

動的NATおよび動的PATの考慮事項

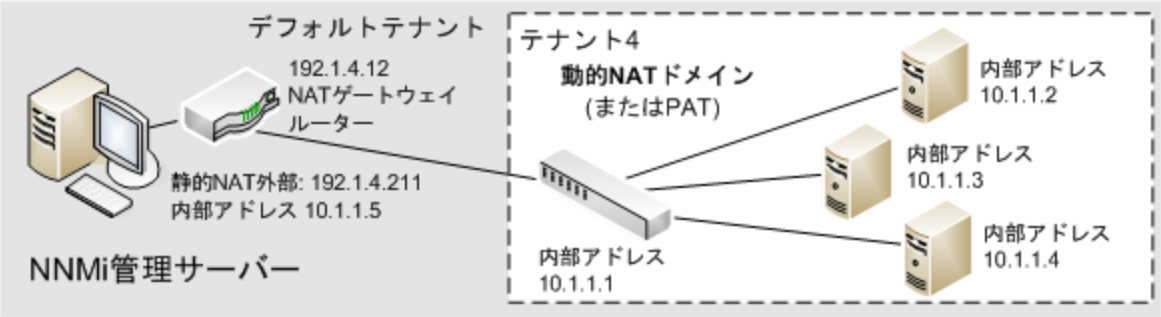
動的NATまたは動的PATの各ドメインには、独自のNNMi管理サーバーが必要です。NNMi管理サーバーは、リージョナルマネージャーとしてグローバルネットワーク管理環境に参加している必要があります。

NNMi管理者は、各NATドメインを識別するためにテナント定義を作成します。テナントは、NNMiグローバルネットワーク管理設定全体で一意である必要があります。

動的NATの以下の2つの設定例を参照してください。

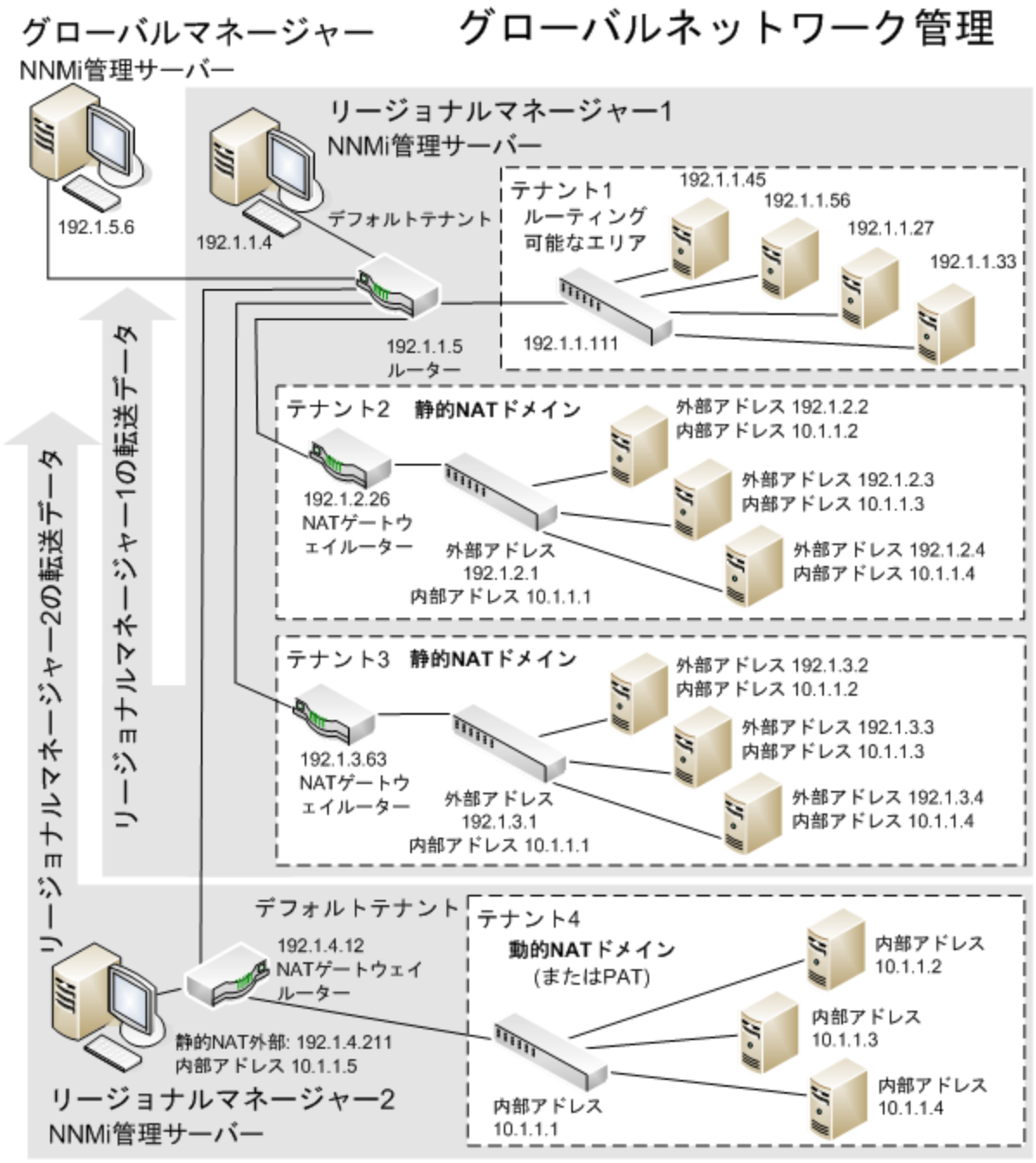
注: リージョナルマネージャーがNATファイアウォールの背後にある場合、その外部 (パブリック) アドレスは静的アドレスである必要があります。

動的NATの設定例



NAT環境内のグローバルネットワーク管理設定の例として以下の図を参照してください。

NAT環境内のグローバルネットワーク管理設定の例



デフォルトテナントに属するデバイスは、任意のテナントの任意のデバイスにレイヤー2接続できます。デフォルトテナント以外のテナント内のデバイスは、同じテナントかデフォルトテナント内のデバイスにしかレイヤー2接続できません。

ヒント: 複数のNATドメイン (NATゲートウェイなど) と相互接続するインフラストラクチャーデバイスは、すべてデフォルトテナントに割り当てます。これにより、ワークグループ (および顧客) が確認する必要があるレイヤー2接続がNNMiに表示されるようになります。

注: デフォルトセキュリティグループ内のデバイスはすべてのビューで表示されます。デバイスへのアクセスを制御するには、該当するデバイスをデフォルトセキュリティグループ以外のセキュリティグループに割り当てます。

グローバルネットワーク管理の詳細については、「グローバルネットワーク管理」(452ページ)を参照してください。テナントの設定の詳細については、NNMiヘルプの「テナントを設定する」を参照してください。

動的NATおよび動的PATのハードウェアとソフトウェアの要件

動的NATおよび動的PAT環境では、NNMi Advanced、NNMi Premium、またはNNMi Ultimateソフトウェアが必要です。

動的NATまたは動的PATで設定されたアドレスドメインごとにNNMiリージョナルマネージャーが必要です。

動的NATおよび動的PATの検出設定

NNMi管理者は、ネットワーク管理環境内の各動的NATドメインを識別するためにテナント定義を作成する必要があります。テナント名は、NNMiグローバルネットワーク管理設定全体で一意である必要があります。

スパイラル検出では、NATドメイン内の各ノードを識別するために検出シード(テナントとIPアドレスのペア)が必要です。NNMi管理者は、動的NATドメインのノードごとに検出シードを作成する必要があります。検出シードでは、ノードごとに以下の情報を指定する必要があります。

- 内部IPアドレス(外部アドレス/内部アドレスペアのパブリックアドレス)
- テナント名

注: 動的NATまたは動的PAT環境内に検出シードを追加する場合(`nnmloadseeds.ovpl`コマンドまたはグラフィカルユーザーインタフェースを使用)、必ずノードの内部IPアドレスを使用してください。

詳細については、nnmloadseeds.ovplリファレンスページ、Linuxのマニュアルページ、またはNNMiヘルプを参照してください。

動的NATのモニタリングの設定

ネットワーク環境によって、NNMi管理者はICMP障害モニタリングの設定の使用を選択できます(「状態とステータスのNNMi計算」(425ページ)も参照)。

- [**モニタリングの設定**] > [**ノードの設定**] タブ。ノードグループのモニタリングを設定します。[ICMP障害モニタリング] セクションで選択します(詳細については、NNMiオンラインヘルプを参照してください)。

- 管理アドレスポーリング (デフォルトで有効な、強く推奨される機能)
- IPアドレス障害ポーリング (省略可能)

- [**モニタリングの設定**] > [**デフォルト設定**] タブ。[ICMP障害モニタリング] セクションで選択します (詳細については、NNMiオンラインヘルプを参照してください)。

注: ネットワーク環境に静的NATドメインも設定されている場合、動的NATドメインとは異なる設定が静的NATドメインで必要になることがあるため、デフォルト設定が適切でない可能性があります。

サブネットと動的NATおよび動的PAT

動的NATまたはPAT環境でサブネットを使用する場合、以下の点に注意してください。

- サブネットはテナントに固有です (サブネットは複数のテナントにまたがらない)。

  ヒント: 同じサブネットを異なるテナントで使用できます。

- サブネットフィルターではテナントとアドレスのペアが使用されます。
- サブネット接続ルールを設定する場合、そのルールはすべてのテナントに適用されます。サブネットのメンバーは、すべてのテナントで一意である必要があります (各ノードは1つのテナントにのみ割り当てられます)。サブネット接続ルールで、デフォルトテナントと別のテナント間にリンクを確立できます。ただし、2つのテナント間のリンクは、いずれかのテナントがデフォルトテナントである場合にのみ使用できます。

グローバルネットワーク管理:動的NATおよび動的PATで必須

NNMiのグローバルネットワーク管理機能は、動的NATドメインの管理で必要です。各動的NATドメインおよび動的PATドメインには、独自のNNMiリージョナルマネージャーが必要です。

NNMiリージョナルマネージャーごとに、少なくとも1つの静的またはルーティング可能 (非変換) アドレスが存在している必要があります。これにより、NNMi管理サーバーが相互に通信することができ、通信を隠ぺいしてセキュリティを確保できます。

注: リージョナルマネージャーがNATファイアウォールの背後にある場合、その外部アドレスは静的アドレスである必要があります。

グローバルネットワーク管理の詳細については、「グローバルネットワーク管理」(452ページ)を参照してください。NNMiヘルプの「グローバルネットワーク管理のためのテナントのベストプラクティス」も参照してください。

ネットワークアドレス変換 (NAT) 環境でのNNMiの配備

NAT環境でNNMiを配備するには、以下の手順を実行します。

1. ネットワーク管理環境の各NATドメインのリストを特定して作成します。
2. 各NATドメイン内で使用されるサポート対象NATのタイプを調べます。
3. 各NATドメイン (NATドメインの内部IPアドレス領域内外) に関して、必要に応じて各NNMi管理サーバーを配備します。特別な考慮事項を参照してください。

   「静的NATの考慮事項」(409ページ)

   「動的NATおよび動的PATの考慮事項」(419ページ)
4. NNMiの [**設定**] > [**検出**] > [**テナント**] ワークスペースを使用して、各NATドメインで一意のテナント名を定義します。

   注: 配備でグローバルネットワーク管理を使用している場合、この名前はすべてのNNMi管理サーバー (リージョナルマネージャーとグローバルマネージャー) で一意である必要があります。
5. NNMiでモニタリングする必要のある各NATドメイン内のノードを決定します。
6. 静的NATドメインのみ:重複するアドレスマッピングを作成して、各ノードの割り当てられたNAT外部/内部IPアドレスのペアを識別します。重複するアドレスマッピングを作成する利点については、「重複するIPアドレスマッピング」(411ページ) を参照してください。

   以下の情報を入力します。

   - テナント名
   - 外部IPアドレス
   - 内部IPアドレス

   NNMiの [**設定**] > [**検出**] > [**重複するアドレスマッピング**] ワークスペースまたは `nnmloadipmappings.ovpl`コマンドラインツールのいずれかを使用します。

   詳細については、NNMiオンラインヘルプを参照してください。
7. ネットワーク環境のNNMi管理サーバーの配備先によっては、NNMiでノードの内部アドレスを使用する場合に、ファイアウォールによってNNMiとNATドメイン内のノードの通信がブロックされる可能性があります。そのため、[**設定**] > [**通信の設定**] 設定で、適切な [優先管理アドレス] 設定 (NATの外部または内部IPアドレス) を使用します。
8. ネットワーク環境のNATの [モニタリングの設定] 設定を確認します。
   - 「静的NATのモニタリングの設定」(414ページ)
   - 「動的NATのモニタリングの設定」(421ページ)

     [モニタリングの設定] の詳細については、NNMiオンラインヘルプを参照してください。

9. 各ノードの検出シードを設定します。

   注: 複数のNATドメイン (NATゲートウェイルーターなど) と相互接続するインフラストラクチャーデバイスは、すべてデフォルトテナントに割り当てます。

   NNMiの [**設定**] > [**検出**] > [**シード**] ワークスペースまたは`loadseeds.ovpl`コマンドラインツールのいずれかを使用します。

   - NNMi管理サーバーが内部IPアドレス領域内にある場合、内部IPアドレスを使用して検出シードを設定します。
     - ホスト名/IP (内部IPアドレスを使用)
     - テナント名
   - NNMi管理サーバーが内部IPアドレス領域外にある場合、外部IPアドレスを使用して検出シードを設定します。
     - ホスト名/IP (外部IPアドレスを使用)
     - テナント名

   詳細については、NNMiオンラインヘルプを参照してください。
10. NNMi検出で、期待どおりノードが検出されることを確認します。検出されない場合、設定 (上記) をダブルチェックします。
11. NNMi設定がチームのニーズを満たしていることを確認します。
    - 各ノードのセキュリティグループの割り当てを微調整して、NNMiコンソールで各ノードを表示できるチームメンバー/顧客を制御します。NNMiの [設定] > [セキュリティ] > [セキュリティグループ] ワークスペースを使用します。
    - これらのノードに適用される [モニタリングの設定] 設定を確認して、必要に応じて微調整します。NNMiの [**設定**] > [**モニタリング**] > [**モニタリングの設定**] ワークスペースを使用します。
12. NNMiマップにノード間の接続が期待どおりに表示されることを確認します。表示されない場合、以下の作業を行います。
    - 接続に含まれる両方のノードのテナントの割り当て (デフォルトテナントまたはその他のテナント) が正しいことを確認します。
    - [**設定**] > [**検出の設定**]**の [サブネット接続ルール**] タブの設定が正しいことを確認します。
    - 自動的に検出されない接続をNNMiで強制的に追加するには、`nnmconnedit.ovpl`コマンドラインツールを使用します。詳細については、[**NNMiオンラインヘルプ**] > [**NNMi ドキュメントライブラリ**] > [**リファレンスページ**] を参照してください。
13. 適切なNNMi管理サーバーのIPアドレスが含まれるように各ノードのSNMPエージェントのSNMPトラップ転送ルールが設定されていることを確認します。
14. 静的NATドメインのみ:NNMiの [重複するアドレスマッピング] の [内部アドレス] に関連付けられたインタフェースが、NNMi管理サーバーに送信されるすべてのトラップのソースになるように、各静的NATノードのSNMPエージェントを設定します。

15. ネットワーク環境にSNMPv1が含まれている場合、NNMi設定で必要な変更を適切に行います。
「トラップと静的NAT」(414ページ)を参照してください。

状態とステータスのNNMi計算

デフォルトのNNMiでは、NAT環境に存在するノードを含め、各ノードの管理アドレスのICMPポーリングが自動的に有効になります ([**設定**] > [**モニタリング**] > [**モニタリングの設定**]、[**デフォルト設定**] タブ、[**ICMP障害モニタリング**] セクションの [**管理アドレスポーリングを有効にする**] 設定)。NAT環境がある場合、この設定を無効にしないことをお勧めします。

注: [**インベントリ**] > [**SNMPエージェント**] ビューでSNMPエージェントを選択し、[**アクション**] > [**モニタリングの設定**] コマンドを使用します。表示される情報に、NNMiでこの管理アドレスポーリングが有効になっているかどうかが示されます。

管理アドレスポーリングが有効の場合は、[エージェントICMP状態] フィールドが、以下の場所に表示されます。

- [ノード] フォーム
- [SNMPエージェント] フォーム
- [SNMPエージェント] テーブルビュー

以下の表に、[ICMP障害モニタリング] 設定に基づいてNNMiの動作がどのように変化するかを示します。表の1番目の行に、NNMiのデフォルト設定を示します。

モニタリングの設定の内容および結果としてのState Poller動作

ICMP障害モニタリングの設定		**結果としてのNNMi動作**	
管理アドレスポーリングを有効にする	IPアドレス障害ポーリングを有効にする	エージェントICMP状態	IPアドレス状態
有効	無効	ポーリング	未ポーリング
有効	有効	ポーリング	ポーリング
無効	無効	未ポーリング	未ポーリング
無効	有効	未ポーリング	ポーリング

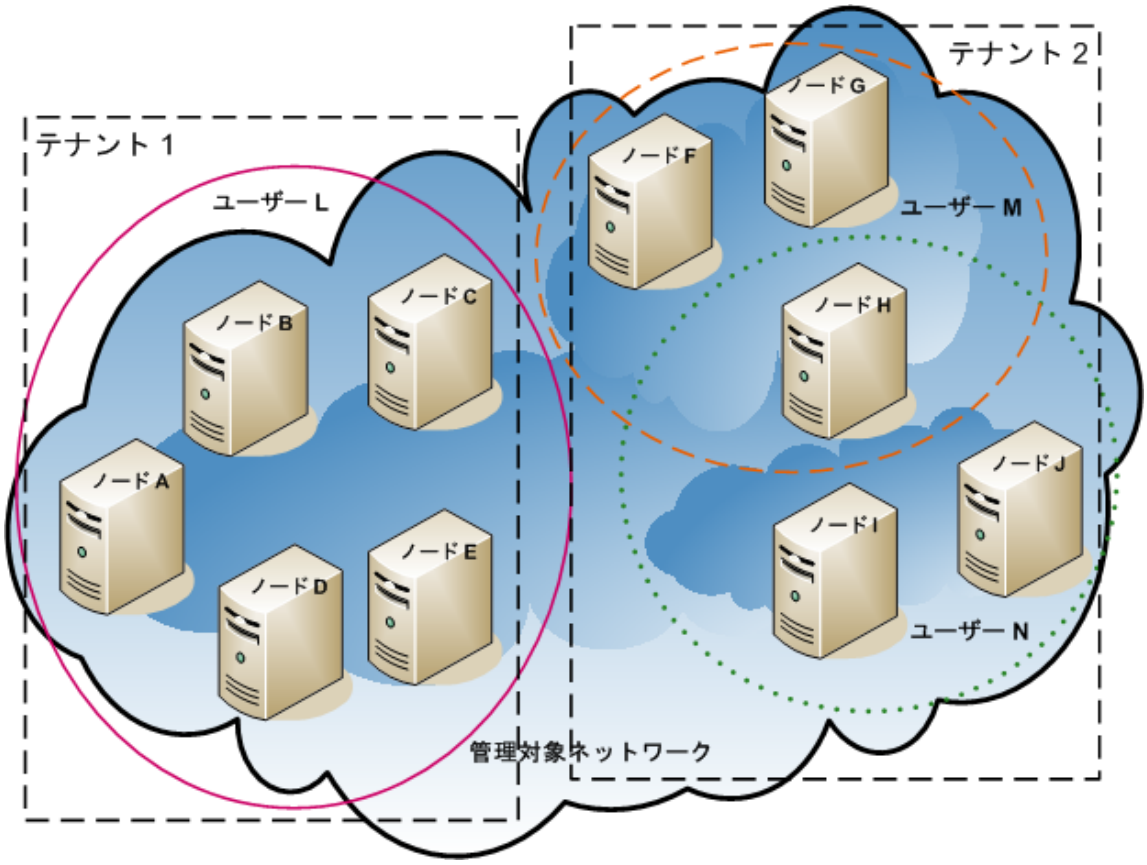
管理アドレスポーリングを有効にすると、結果の計算時とインシデントの生成時に、管理アドレスのICMP応答とSNMPエージェントの応答の両方がNNMiで考慮されます。

以下の表に、ICMP応答とSNMP応答の組み合わせによって決定される、SNMPエージェントステータスの計算を示します。

SNMPエージェントステータスの判断

SNMPエージェントの応答	管理アドレスのICMP応答	結果としてのSNMPエージェントステータス
応答	応答	正常域
応答	無応答	警戒域
無応答	応答	危険域
無応答	無応答	危険域

NNMiセキュリティおよびマルチテナント



注: NNMiでは、テナントを使用して重複アドレスドメインを含むネットワークに対応します。重複アドレスドメインは、ネットワーク管理ドメインの静的ネットワークアドレス変換 (NAT)、動的 NAT、または動的ポートアドレス変換 (PAT) 領域内に存在する可能性があります。そのようなネットワークの場合、重複アドレスドメインを異なるテナントに配置します (これはシード済み検出を使用して行います)。詳細については、「NAT環境の重複IPアドレスの管理」(407ページ)およびNNMiヘルプを参照してください。

デフォルトでは、すべてのNNMiコンソールユーザーがNNMiデータベースのすべてのオブジェクトを参照できます。使用環境でこのデフォルト設定を許容できる場合、この章を読む必要はありません。

NNMiセキュリティおよびマルチテナントでは、NNMiデータベースのオブジェクトに関する情報へのユーザーアクセスを制限できます。この制限は、ネットワークオペレーターのビューをその責任範囲に合わせてカスタマイズする場合に役立ちます。また、サービスプロバイダーがNNMiを組織ごとに設定する場合にも役立ちます。

この章では、NNMiセキュリティおよびテナントモデルについて説明し、設定の推奨事項について記載します。内容は以下のとおりです。

『HP Network Node Manager i Software Step-by-Step Guide to Using Security Groups White Paper』も参照してください。

オブジェクトのアクセス制限による影響

NNMiセキュリティを設定すると以下のような影響があります。

- トポロジインベントリオブジェクト:
  - 各NNMiコンソールユーザーには、それぞれのユーザーのNNMiユーザーアカウント設定に対応するノードのみが表示されます。
  - インタフェースなどのサブノードオブジェクトは、そのノードからアクセス制御を継承します。
  - 接続などのノード間オブジェクトは、NNMiコンソールユーザーが、関連するノードの少なくとも1つを表示できる場合にのみ表示されます。
  - NNMiコンソールユーザーには、ノードグループの中の少なくとも1つのノードにそのユーザーがアクセスできるノードグループのみが表示されます。
  - Network Performance Server (NPS) レポートの場合、NNMi管理者はインタフェースのアクセス制御の継承を選択的に上書きできます。詳細については、「NPSレポートへの選択インタフェースの追加」(451ページ)を参照してください。
- マップおよびパスビュー:
  - マップには、関与している両方のノードを表示する権限をNNMiコンソールユーザーが持っている接続が表示されます。

- パスビューでは、NNMiコンソールユーザーがアクセスできないすべての中間ノードは省略されるか、クラウドとして表示されます。

- NNM iSPI for MPLSおよびNNM iSPI for IP Multicastについては、マップとパスビューにNNMiコンソールユーザーがアクセスできないノードが含まれている場合、NNM iSPIには接続中のインタフェースとノードの名前しか表示されません。アクセスできないノードのアイコンは白色で表示され、それらのノードのステータスと詳細情報を入手できないことが示されます。

- NNM iSPI for IP Telephonyについては、マップとパスビューにNNMiコンソールユーザーがアクセスできないノードが含まれている場合、NNM iSPIには接続されているインタフェースとノードの名前しか表示されません。アクセスできないノードのアイコンにはNNMiステータスが表示されますが、アクションを行ってもすべて失敗します。

- インシデント:
  - ソースノードがNNMiトポロジ内にあるインシデントについては、NNMiコンソールユーザーには、そのユーザーがソースノードにアクセスできるインシデントのみが表示されます。

  - NNMiの稼働状態およびライセンス管理イベントのインシデントなど、ソースノードが含まれないインシデントは、1つのグループとして処理されます。NNMi管理者は、どのNNMiコンソールユーザーにそれらのインシデントが表示されるかを (ユーザーに [未解決のインシデント] セキュリティグループを関連付けることで) 決定します。

  - ソースノードがNNMiトポロジ内にないトラップから生じたインシデントは、ソースノードが含まれないインシデントと同様に処理されます。これらのインシデントを生成するようにNNMiが設定されている場合、NNMi管理者は、どのNNMiコンソールユーザーにそれらのインシデントが表示されるかを (ユーザーに [未解決のインシデント] セキュリティグループを関連付けることで) 決定します。

  注: インシデントの割り当てアクションでは、ユーザーのアクセス権はチェックされません。NNMi管理者によって、あるインシデントがそのインシデントを表示する権限を持たないNNMiコンソールユーザーに割り当てられる可能性があります。

- NNMiコンソールアクション:
  - 何も選択を行わずに実行されるアクションについては、NNMiコンソールユーザーには、そのユーザーが実行する権限を持っているアクションのみが表示されます。

  - 選択された1つ以上のオブジェクトに対して実行されるアクションの場合、NNMiコンソールユーザーは、選択されたオブジェクトに対する適切なアクセスレベルを持っている必要があります。セキュリティ設定によっては、NNMiコンソールビューに表示されている一部のオブジェクトに対して有効ではないアクションがNNMiコンソールに表示される場合もあります。これらの無効なアクションを実行すると、この制限に関するエラーメッセージが表示されます。

  - マップビューや、NNM iSPIテーブルビューおよびフォームについては、NNMiは、認識不能なノードと、NNMiトポロジ内に存在するが現在のユーザーがアクセスできないノードの区別を行うことができません。

- MIBブラウザーおよびLine Grapher:
  - NNMiコンソールユーザーは、ユーザーがアクセスできるノードのMIBデータとグラフを表示できます。
  - NNMiコンソールユーザーは、ユーザーがSNMPコミュニティ文字列を認識しているノードのMIBデータを表示できます。
- NNMiコンソール URL:

  ダイレクトURLからNNMiコンソールビューにアクセスするには、NNMiにログオンする必要があります。NNMiは、NNMiセキュリティ設定に応じてユーザーのアクセス権を適用し、それに従って、使用可能なトポロジを制限します。

NNMiセキュリティモデル

NNMiセキュリティモデルでは、NNMiデータベースのオブジェクトへのユーザーアクセスを制御できます。このモデルは、NNMiユーザーのアクセスを特定のオブジェクトやインシデントに制限するネットワーク管理組織で使用する場合に適しています。NNMiセキュリティモデルには、以下の利点があります。

- NNMiコンソールオペレーターのネットワークのビューを制限できます。オペレーターは特定のデバイスタイプまたはネットワーク領域に集中できます。
- NNMiトポロジへのオペレーターアクセスをカスタマイズできます。オペレーターアクセスのレベルは、ノードごとに設定できます。
- [ノード (すべての属性)] ビューおよびNetwork Performance Serverレポートをセキュリティグループでフィルタリングできます。
- セキュリティ設定で構成されるノードグループの設定およびメンテナンスが簡素化されます。
- NNMiテナントモデルとは独立して使用できます。

NNMiセキュリティは、以下のような場合に使用されます。

- NNMiオペレーターがサイト (カスタムマップ) 内の機器タイプに集中できるようにする。
- 特定のサイト (カスタムマップ) のノードのみが表示される各サイトビューをNNMiオペレーターに提供する。
- 導入時にノードをステージングする。NNMi管理者にはすべてのノードが表示されますが、NNMiオペレーターには導入したノードのみが表示されます。
- すべてのNOCオペレーターにフルアクセスを付与し、NOCユーザーのアクセスを制限する。
- 中央のNOCオペレーターに完全なネットワークビューを提供し、地域のNOCオペレーターのビューを制限する。

セキュリティグループ

NNMiセキュリティモデルでは、ノードへのユーザーアクセスはユーザーグループおよびセキュリティグループを介して間接的に制御されます。NNMiトポロジ内の各ノードは、1つのセキュリティグループのみに関連付けられます。セキュリティグループは複数のユーザーグループに関連付けることができます。

各ユーザーアカウントは、以下のユーザーグループにマッピングされます。

- 以下に示す事前設定された1つ以上のNNMiユーザーグループ:
  - NNMi管理者
  - NNMiグローバルオペレーター
  - NNMiレベル2オペレーター
  - NNMiレベル1オペレーター
  - NNMiゲストユーザー

  このマッピングはNNMiコンソールアクセスに必要で、これによってNNMiコンソール内で使用できるアクションが決まります。ユーザーアカウントがこれらの複数のNNMiユーザーグループにマッピングされている場合、許可されるアクションのスーパーセットがユーザーに付与されます。

  注: [NNMi Webサービスクライアント] ユーザーグループでは、NNMiコンソールへのアクセス権は付与されませんが、すべてのNNMiオブジェクトへの管理者レベルのアクセス権が付与されます。

  注: NNMiグローバルオペレーターユーザーグループ (`globalops`) では、トポロジオブジェクトのみにアクセス権が与えられます。ユーザーがNNMiコンソールにアクセスするには、ユーザーを他のいずれかのユーザーグループ (`level2`、 `level1`、または `guest`) に割り当てる必要があります。

  `globalops`ユーザーグループはデフォルトですべてのセキュリティグループにマッピングされるため、管理者はこのユーザーグループをセキュリティグループにマッピングしないようにする必要があります。

- セキュリティグループにマッピングされる0個以上のカスタムユーザーグループ

  これらのマッピングでは、NNMiデータベースのオブジェクトへのアクセスが提供されます。各マッピングには、セキュリティグループのノードに適用されるオブジェクトアクセス権限レベルが含まれています。オブジェクトアクセス権限レベルは、インタフェースやインシデントなどの関連するデータベースオブジェクトにも適用されます。たとえば、インタフェースXおよびYを含むノードへのオブジェクトオペレーターレベル1のアクセス権限があるユーザーには、以下のすべてのデータベースオブジェクトへのオブジェクトオペレーターレベル1のアクセス権限があります。

- ノードA
- インタフェースXおよびY
- ソースオブジェクトがノードA、インタフェースX、またはインタフェースYのインシデント

NNMiには、以下のセキュリティグループがあります。

- デフォルトセキュリティグループ

  新しいNNMiインストール済み環境では、[デフォルトのセキュリティグループ] がすべてのノードに対する初期セキュリティグループとして割り当てられます。デフォルトでは、すべてのユーザーに、[デフォルトのセキュリティグループ] 内のすべてのオブジェクトが表示されます。NNMi管理者は、[デフォルトのセキュリティグループ] に関連付けられるノードと、[デフォルトのセキュリティグループ] 内のオブジェクトにアクセスできるユーザーを設定できます。
- 未解決のインシデント

  [未解決のインシデント] セキュリティグループは、ソースノードがNNMiトポロジ内にない受信トラップからNNMiが作成するインシデントへのアクセス権を提供します。デフォルトでは、すべてのユーザーに、[未解決のインシデント] セキュリティグループに関連付けられたすべてのインシデントが表示されます。NNMi管理者は、[未解決のインシデント] セキュリティグループに関連付けられたインシデントにアクセスできるユーザーを設定できます。

すべてのセンサーは、ノードのセキュリティグループの割り当てを継承します。

注: 以下のベストプラクティスがNNMiセキュリティ設定に適用されます。

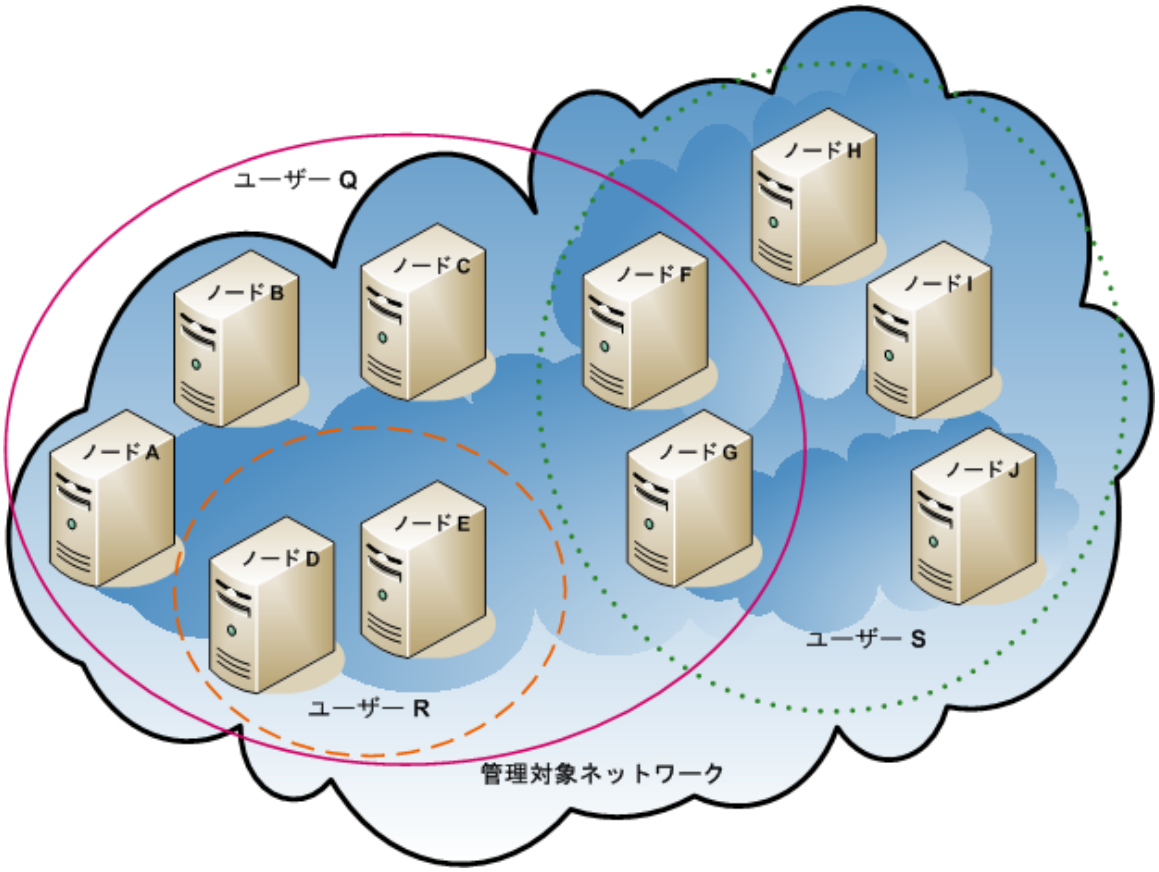
- 各ユーザーアカウントを事前設定された1つのNNMiユーザーグループのみにマッピングします。
- 事前設定されたNNMiユーザーグループをセキュリティグループにマッピングしないでください。
- [NNMi管理者] ユーザーグループにマッピングされたすべてのユーザーアカウントには、NNMiデータベースのすべてのオブジェクトに対する管理者レベルのアクセス権が付与されるため、このユーザーアカウントをほかのユーザーグループにマッピングしないでください。
- Web Service Clientロール専用のユーザーアカウントを別個に作成します。このユーザーアカウントはNNMiトポロジ全体にアクセスできるため、このユーザーアカウントは [NNMi Web Service Client] ユーザーグループにのみマッピングしてください。

セキュリティグループ構造の例

以下の図にある3つの楕円形は、このNNMiトポロジの例で、ユーザーに表示する必要のあるノードのプライマリグループを示しています。ユーザーアクセスを完全に制御するには、4つの各サブグループが一意のセキュリティグループに対応している必要があります。一意の各セキュリティグループを1つ以上のユーザーグループにマッピングして、そのセキュリティグループ内のオブジェクトに対する使用可能なユーザーアクセスのレベルを表すことができます。

セキュリティグループマッピングの例に、このトポロジにおけるセキュリティグループと考えられるカスタムユーザーグループ間のマッピングをリストします(このセキュリティモデルを実際に実装する場合、これらのカスタムユーザーグループの一部は不要になる可能性があります)。ユーザーアカウントマッピングの例に、このトポロジにおけるいくつかのユーザーアカウントとユーザーグループのマッピングをリストします。

ユーザーアクセス要件に対応するトポロジの例



セキュリティグループマッピングの例

セキュリティグループ	セキュリティグループのノード	ユーザーグループ	オブジェクトアクセス権限
SG1	A、B、C	UG1管理者	オブジェクト管理者
		UG1レベル2	オブジェクトオペレーターレベル2
		UG1レベル1	オブジェクトオペレーターレベル1
		UG1ゲスト	オブジェクトゲスト

セキュリティグループマッピングの例 (続き)

セキュリティグループ	セキュリティグループのノード	ユーザーグループ	オブジェクトアクセス権限
SG2	D、E	UG2管理者	オブジェクト管理者
		UG2レベル2	オブジェクトオペレーターレベル2
		UG2レベル1	オブジェクトオペレーターレベル1
		UG2ゲスト	オブジェクトゲスト
SG3	F、G	UG3管理者	オブジェクト管理者
		UG3レベル2	オブジェクトオペレーターレベル2
		UG3レベル1	オブジェクトオペレーターレベル1
		UG3ゲスト	オブジェクトゲスト
SG4	H、I、J	UG4管理者	オブジェクト管理者
		UG4レベル2	オブジェクトオペレーターレベル2
		UG4レベル1	オブジェクトオペレーターレベル1
		UG4ゲスト	オブジェクトゲスト

ユーザーアカウントマッピングの例

ユーザーアカウント	ユーザーグループ	ノードアクセス	注
ユーザーQ	NNMiレベル2オペレーター	なし	このユーザーには、ピンクの楕円形 (実線) に含まれるノードへのオペレーターレベル2のアクセス権限があります。
	UG1レベル2	A、B、C	
	UG2レベル2	D、E	
	UG3レベル2	F、G	

ユーザーアカウントマッピングの例 (続き)

ユーザーアカウント	ユーザーグループ	ノードアクセス	注
ユーザーR	NNMiレベル1オペレーター	なし	このユーザーには、オレンジの楕円形 (破線) に含まれるノードへのオペレーターレベル1のアクセス権限があります。
	UG2レベル1	D、E	
ユーザーS	NNMiレベル2オペレーター	なし	このユーザーには、緑の楕円形 (点線) に含まれるノードへのオペレーターレベル2のアクセス権限があります。
	UG3レベル2	F、G	
	UG4レベル2	H、I、J	
ユーザーT	NNMiレベル2オペレーター	なし	このユーザーは、トポロジの例に含まれるすべてのノードに (各権限レベルで) アクセスできます。 このユーザーには、ノードDおよびEへの管理アクセス権がありますが、管理アクセス権が必要なツールのメニュー項目は表示できません。ユーザーにNNMi管理サーバーへのアクセス権がある場合は、ノードDおよびEに対してのみ、管理アクセス権が必要なコマンドラインツールを実行できます。
	UG1ゲスト	A、B、C	
	UG2管理者	D、E	
	UG3レベル2	F、G	
	UG4レベル1	H、I、J	

NNMiテナントモデル

NNMiテナントモデルでは、トポロジ検出とトポロジデータが各テナント (組織または顧客とも呼ばれる) で完全に分離されます。このモデルは、サービスプロバイダー (特に管理対象サービスプロバイダー) や大規模エンタープライズに適しています。NNMiテナントモデルには、以下の利点があります。

- 各ノードが属する組織が明確になります。
- [ノード (すべての属性)] インベントリビューとNetwork Performance Serverレポートを、テナントとセキュリティグループでフィルタリングできます。
- 顧客データへのオペレーターアクセスを分離する規制要件に適合します。
- テナント設定で構成されるノードグループの設定およびメンテナンスが簡素化されます。
- NNMiセキュリティの設定が簡素化されます。
- アドレス変換プロトコルを使用した場合、重複しているアドレスドメインを管理できます。

NNMiマルチテナントを使用すると、同じNNMi管理サーバーで複数の顧客 (テナント) を管理するサービスプロバイダーに、異なる顧客ビューを提供することができます。

注: 各インスタンスが一意のテナントで設定されている場合、1つのNNMi管理サーバーで任意の数の静的ネットワークアドレス変換 (NAT) インスタンスを監視できます。詳細については、「NAT環境の重複IPアドレスの管理」(407ページ)およびNNMiヘルプを参照してください。

テナント

NNMiテナントモデルでは、組織という概念がセキュリティ設定に加わります。NNMiトポロジ内の各ノードが属するテナントは、1つのみです。テナントによって、NNMiデータベースが論理的に分離されます。オブジェクトアクセスはセキュリティグループで管理されます。

ノードが最初に検出されてNNMiデータベースに追加されるときに、各ノードで初期検出テナントの割り当てが発生します。シード済みのノードで、各ノードに割り当てるテナントを指定できます。NNMiによって、検出された他のすべてのノード (自動検出ルールに含まれているが直接シードされないノード) がデフォルトテナントに割り当てられます。NNMi管理者は、検出後にいつでもノードのテナントを変更できます。

各テナント定義には、初期検出セキュリティグループが含まれます。NNMiによって、この初期検出セキュリティグループが初期検出テナントとともにノードに割り当てられます。NNMi管理者は、検出後にいつでもノードのセキュリティグループを変更できます。

ヒント: ノードのテナントの割り当てを変更しても、セキュリティグループの割り当ては自動的に変更されません。

NNMiには、デフォルトテナントが備わっています。デフォルトでは、すべてのNNMiユーザーが、([デフォルトのセキュリティグループ] を介して) このテナントに関連付けられたすべてのオブジェクトにアクセスできます。

すべてのセンサーは、ノードのテナントおよびセキュリティグループの割り当てを継承します。

注: 以下のベストプラクティスがNNMiテナント設定に適用されます。

- 小規模な組織の場合、テナントごとに1つのセキュリティグループで十分です。
- 大規模な組織を複数のセキュリティグループに分割できます。

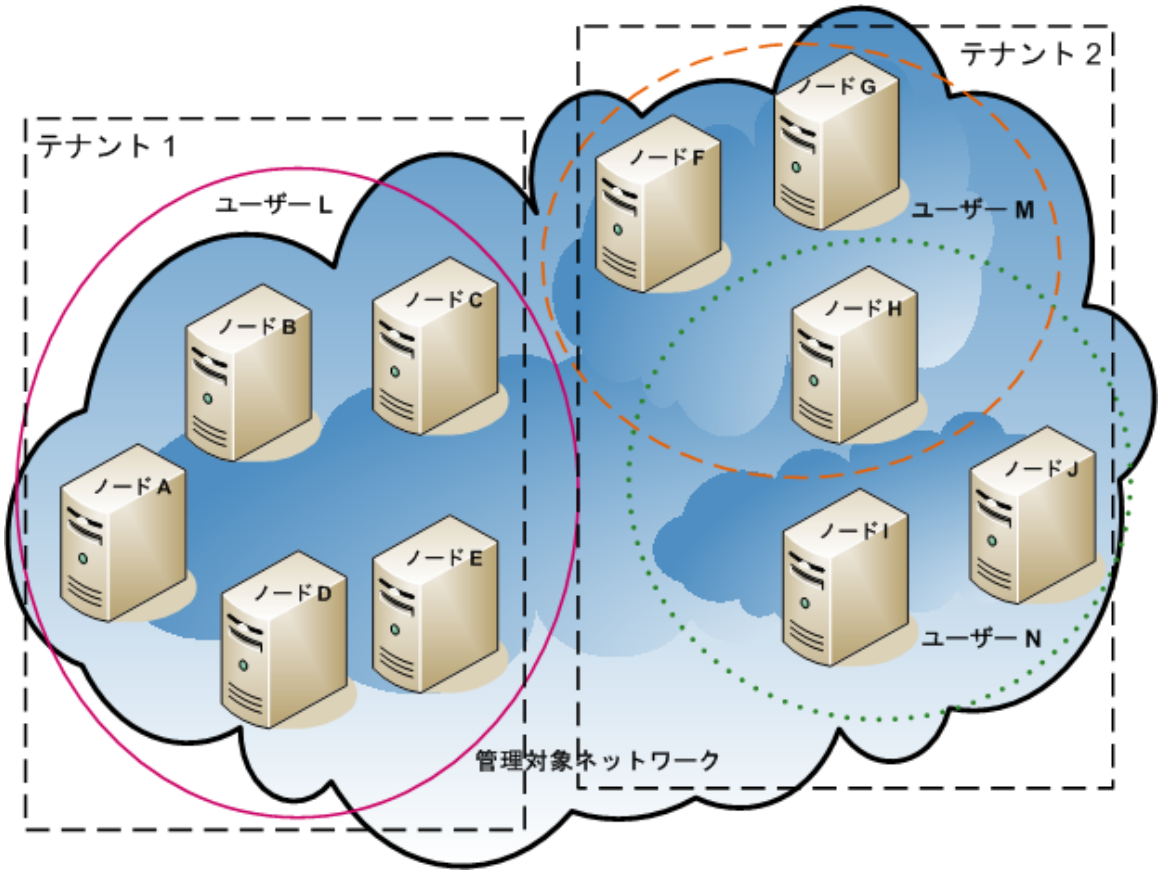
- ユーザーが組織をまたいでノードにアクセスできないようにするには、各セキュリティグループに、1つのテナントのみに対応するノードしか含まれないようにします。

テナント構造の例

以下の図に、NNMiトポロジ内に2つのテナントが含まれている様子を長方形の線で囲んで示します。これらの3つの楕円形は、ユーザーにノードを表示する必要があるプライマリグループを表しています。テナント1のトポロジは1つのグループとして管理されるため、1つのセキュリティグループのみが必要です。テナント2のトポロジは重複しているセットで管理されるため、3つのセキュリティグループに分割されます。

複数のテナントのセキュリティグループマッピングの例に、このトポロジにおけるセキュリティグループと考えられるカスタムユーザーグループ間のマッピングをリストします(このセキュリティモデルを実際に実装する場合、これらのカスタムユーザーグループの一部は不要になる可能性があります)。複数のテナントのユーザーアカウントマッピングの例に、このトポロジにおけるいくつかのユーザーアカウントとユーザーグループのマッピングをリストします。

複数のテナントのトポロジの例



複数のテナントのセキュリティグループマッピングの例

セキュリティグループ	セキュリティグループのノード	ユーザーグループ	オブジェクトアクセス権限
T1 SG	A、B、C、D、E	T1管理者	オブジェクト管理者
		T1レベル2	オブジェクトオペレーターレベル2
		T1レベル1	オブジェクトオペレーターレベル1
		T1ゲスト	オブジェクトゲスト
T2 SGa	F、G	T2_a管理者	オブジェクト管理者
		T2_aレベル2	オブジェクトオペレーターレベル2
		T2_aレベル1	オブジェクトオペレーターレベル1
		T2_aゲスト	オブジェクトゲスト
T2 SGb	H	T2_b管理者	オブジェクト管理者
		T2_bレベル2	オブジェクトオペレーターレベル2
		T2_bレベル1	オブジェクトオペレーターレベル1
		T2_bゲスト	オブジェクトゲスト
T2 SGc	I、J	T2_c管理者	オブジェクト管理者
		T2_cレベル2	オブジェクトオペレーターレベル2
		T2_cレベル1	オブジェクトオペレーターレベル1
		T2_cゲスト	オブジェクトゲスト

複数のテナントのユーザーアカウントマッピングの例

ユーザーアカウント	ユーザーグループ	ノードアクセス	注
ユーザーL	NNMiレベル2オペレーター	なし	このユーザーには、テナント1のすべてのノードをグループ化する、ピンクの楕円形 (実線) に含まれるノードへのオペレーターレベル2のアクセス権限があります。
	T1レベル2	A、B、C、D、E	
ユーザーM	NNMiレベル1オペレーター	なし	このユーザーには、テナント2のノードのサブセットをグループ化する、オレンジの楕円形 (破線) に含まれるノードへのオペレーターレベル1のアクセス権限があります。
	T2_aレベル1	F、G	
	T2_bレベル1	H	
ユーザーN	NNMiレベル2オペレーター	なし	このユーザーには、テナント2のノードのサブセットをグループ化する、緑の楕円形 (点線) に含まれるノードへのオペレーターレベル2のアクセス権限があります。
	T2_bレベル2	H	
	T2_cレベル2	I、J	

NNMiのセキュリティおよびマルチテナント設定

注: 各インスタンスが一意のテナントで設定されている場合、1つのNNMi管理サーバーで任意の数の静的ネットワークアドレス変換 (NAT) インスタンスを監視できます。詳細については、「NAT環境の重複IPアドレスの管理」(407ページ)およびNNMiヘルプを参照してください。

NNMiのセキュリティおよびマルチテナント設定は、NNMiデータベース全体に適用されます。NNMi管理者であれば、すべてのテナントのすべてのオブジェクトへのオペレーターアクセス権限を表示および設定できます。

NNMi管理者が少なくとも1つのカスタムセキュリティグループを定義すると、[**セキュリティグループ**] フィールドがすべての [**ノード**] フォームに表示されます。また、[**ノード**] および [**ノード (すべての属性)**] インベントリビューの列としても表示されます。

NNMi管理者が少なくとも1つのカスタムテナントを定義すると、[**テナント**] フィールドがすべての [**ノード**] フォームに表示されます。また、[**ノード**] および [**ノード (すべての属性)**] インベントリビューの列としても表示されます。

ノードグループ

セキュリティ設定またはマルチテナント設定の一部と適合するようにノードグループを作成するには、セキュリティグループUUID、セキュリティグループ名、テナントUUID、またはテナント名に基づいて、ノードグループの追加フィルターを指定します。これらのノードグループを使用して、監視アクションおよびインシデントライフサイクル移行アクション用のポーリングサイクルを、セキュリティグループまたはテナントごとに設定します。

ヒント: セキュリティグループとテナントの名前は変更できるため、追加フィルターにはセキュリティグループまたはテナントのUUIDを指定します。この情報は、設定フォームと、`nnmsecurity.ovpl`コマンド出力で使用できます。

ユーザーグループ:NNMiコンソールアクセス

事前に定義されたNNMiユーザーグループの1つにユーザーアカウントをマッピングすると、NNMiロールと、NNMiコンソールで表示されるメニュー項目が設定されます。各ユーザーアカウントには、そのユーザーのトポロジオブジェクトに対する最も高いオブジェクトアクセス権限とに対応するNNMiロールを付与することをお勧めします。

注: ただし、NNMi管理者はすべてのトポロジオブジェクトへのアクセス権を持つため、管理者レベルの権限を付与することは避けてください。NNMiトポロジ内の一部のノードに対してのみ、NNMiコンソールユーザーを管理者として設定するには、そのユーザーをNNMiレベル2オペレーターまたはNNMiレベル1オペレーターのユーザーグループに割り当てます(レベル1オペレーターにはレベル2オペレーターよりも低いアクセス権が与えられています)。また、オブジェクト管理者オブジェクトアクセス権限を使用して、トポロジ内のノードのサブセットを含むセキュリティグループにマッピングされたカスタムユーザーグループを作成し、ユーザーをそのグループに割り当てます。

ユーザーグループ: ディレクトリサービス

ユーザーグループメンバーシップをNNMiデータベースに保存する場合、すべてのオブジェクトアクセス設定は、NNMi設定エリア内で、ユーザーグループ、ユーザーアカウントマッピング、セキュリティグループ、およびセキュリティグループマッピングを使用して行われます。

ユーザーグループメンバーシップをディレクトリサービスに保存する場合、オブジェクトアクセス設定は、NNMi設定 (セキュリティグループおよびセキュリティグループマッピング) と、ディレクトリサービスコンテンツ (ユーザーグループメンバーシップ) の間で共有されます。NNMiデータベースに、ユーザーアカウントまたはユーザーアカウントマッピングを作成しないでください。ディレクトリサービス内の適用可能なグループごとに、NNMiデータベースに1つ以上のユーザーグループを作成してください。NNMiで、各ユーザーグループ定義の [**ディレクトリサービス名**] フィールドに、ディレクトリサービス内のそのグループの識別名を設定します。

詳細については、「NNMiとLDAPによるディレクトリサービスの統合」(375ページ)を参照してください。

設定ツール

NNMiには、マルチテナントとセキュリティを設定するためのいくつかのツールが備わっています。

セキュリティウィザード

NNMiコンソールの [**セキュリティウィザード**] は、セキュリティ設定の可視化に役立ちます。NNMiコンソール内でノードをセキュリティグループに割り当てるには、このウィザードを使用する方法が最も簡単です。[**変更概要の表示**] ページには、現在のウィザードセッションで保存されていない変更点のリストが表示されます。また、セキュリティ設定に関する潜在的な問題も示されます。

注: [**セキュリティウィザード**] は、NNMiセキュリティ設定に関してのみ使用できます。テナント情報は含まれていません。

[**セキュリティウィザード**] の使用法の詳細については、ウィザード内のNNMiヘルプリンクをクリックしてください。

NNMiコンソールフォーム

NNMiコンソール内の個々のセキュリティオブジェクトおよびマルチテナントオブジェクトのフォームは、設定の1つの側面を同時に集中的に捉える場合に便利です。これらのフォームの使用法の詳細については、各フォームのNNMiヘルプを参照してください。

[**テナント**] ビューにはNNMiマルチテナント設定情報が含まれています。このビューは、[**設定**] ワークスペースの [**検出**] の下に表示されます。各 [**テナント**] フォームには1つのNNMiテナントが記述され、現在そのテナントに割り当てられているノードが表示されます。ノードの割り当て情報は読み取り専用です。

ノードに割り当てられているテナントまたはセキュリティグループを変更するには、[**ノード**] フォームまたは`nnmsecurity.ovpl`コマンドを使用します。

以下のNNMiコンソールビューは、[**設定**] ワークスペースの [**セキュリティ**] の下に表示されます。これらのビューには、以下のNNMiセキュリティ設定情報が含まれています。

- **ユーザーアカウント**
  - 各 [**ユーザーアカウント**] フォームには1つのNNMiユーザーが記述され、そのユーザーが属するユーザーグループが表示されます。メンバーシップ情報は読み取り専用です。
  - ユーザーグループメンバーシップをディレクトリサービスに保存すると、ユーザーアカウントはNNMiコンソールに表示されません。
- **ユーザーグループ**

  各 [**ユーザーグループ**] フォームには1つのNNMiユーザーグループが記述され、そのユーザーグループにマッピングされたユーザーアカウントとセキュリティグループが表示されます。マッピング情報は読み取り専用です。
- **ユーザーアカウントのマッピング**
  - 各 [**ユーザーアカウントのマッピング**] フォームには、1つのユーザーアカウントとユーザーグループの関連付けが表示されます。

- ユーザーアカウントマッピングに変更を行っても、現在のNNMiコンソールユーザーにその変更は反映されません。現在のユーザーは、NNMiコンソールに次回ログオンしたときに、変更を受け取ります。
- ユーザーグループメンバーシップをディレクトリサービスに保存すると、ユーザーアカウントマッピングはNNMiコンソールに表示されません。

- **セキュリティグループ**

  各 [**セキュリティグループ**] フォームには1つのNNMiセキュリティグループが記述され、そのセキュリティグループに現在割り当てられているノードが表示されます。ノードの割り当て情報は読み取り専用です。
- **セキュリティグループのマッピング**
  - 各 [**セキュリティグループのマッピング**] フォームには、1つのユーザーグループとセキュリティグループの関連付けが表示されます。
  - 初期設定の後、セキュリティグループマッピングに関連付けられたオブジェクトのアクセス権限は読み取り専用になっています。セキュリティグループマッピングのオブジェクトアクセス権限を変更するには、そのマッピングを削除して、再度作成します。

コマンドライン

`nnmsecurity.ovpl`コマンドラインインタフェースは、自動操作や一括操作を行う場合に便利です。このツールは、セキュリティ設定に関する潜在的な問題のレポートも提供します。

`nnmsecurity.ovpl`オプションの多くは、カンマ区切り値 (CSV) ファイルからの入力データのロードをサポートしています。設定データは、`nnmsecurity.ovpl`コマンドで使用するために、CSV出力を生成できるファイルまたはシステムに保持できます。このコマンドは、NNMiの外部で生成されたUUIDも受け入れます。

ヒント: セキュリティグループとテナントの名前は一意である必要はないため、`nnmsecurity.ovpl`コマンドへの入力値としてセキュリティグループまたはテナントのUUIDを指定します。

以下のスクリプト例では、`nnmsecurity.ovpl`コマンドを使用して、2つのユーザーアカウントと5つのノードにセキュリティ設定を作成しています。

```
#!/bin/sh

# 2つのユーザーを作成する

nnmsecurity.ovpl -createUserAccount -u user1 -p password -role level1

nnmsecurity.ovpl -createUserAccount -u user2 -p password -role level2

# 2つのグループを作成する

nnmsecurity.ovpl -createUserGroup local1

nnmsecurity.ovpl -createUserGroup local2

#新しいユーザーグループにユーザーアカウントを割り当てる
```

```
nnmsecurity.ovpl -assignUserToGroup -user user1 -userGroup local1

nnmsecurity.ovpl -assignUserToGroup -user user2 -userGroup local2

# 2つのセキュリティグループを作成する

nnmsecurity.ovpl -createSecurityGroup secgroup1

nnmsecurity.ovpl -createSecurityGroup secgroup2

#新しいセキュリティグループに新しいユーザーグループを割り当てる

nnmsecurity.ovpl -assignUserGroupToSecurityGroup -userGroup local1
 -securityGroup secgroup1 -role level1

nnmsecurity.ovpl -assignUserGroupToSecurityGroup -userGroup local2
 -securityGroup secgroup2 -role level2

#セキュリティグループをノードに割り当てる

nnmsecurity.ovpl -assignNodeToSecurityGroup -node mplspe01 -securityGroup secgroup1

nnmsecurity.ovpl -assignNodeToSecurityGroup -node vwan_router-1 -securityGroup secgroup1

nnmsecurity.ovpl -assignNodeToSecurityGroup -node vwan_router-2 -securityGroup secgroup1

nnmsecurity.ovpl -assignNodeToSecurityGroup -node data_center_1 -securityGroup secgroup2

nnmsecurity.ovpl -assignNodeToSecurityGroup -node mplspe03 -securityGroup secgroup2
```

テナントの設定

注: 各インスタンスが一意のテナントで設定されている場合、1つのNNMi管理サーバーで任意の数の静的ネットワークアドレス変換 (NAT) インスタンスを監視できます。詳細については、「NAT環境の重複IPアドレスの管理」(407ページ)およびNNMiヘルプを参照してください。

NNMiでは、以下の方法でマルチテナントを設定できます。

- NNMiコンソールの [**テナント**] フォームは、個々のテナントを処理する際に役立ちます。
- `nnmsecurity.ovpl`コマンドラインインタフェースは、自動操作や一括操作を行う場合に便利です。このツールは、テナント設定に関する潜在的な問題のレポートも提供します。

各NNMiトポロジオブジェクトをテナント (組織) に割り当てるためにNNMiマルチテナントを定義および設定するプロセスは、循環的なプロセスです。この概略的な手順では、NNMiマルチテナントを設定するための1つの方法を説明します。

NNMiマルチテナントの設定に関しては、以下に注意してください。

- 検出されたノードにNNMiによって割り当てられるセキュリティグループは、そのノードに関連付けられたテナントの [初期検出セキュリティグループ] の値によって設定されます。
- NNMiテナントを設定しないで、NNMiセキュリティモデルを使用すると、すべてのノードがデフォルトテナントに割り当てられます。

- NNMi検出用にノードをシードするときに、そのノードが属するテナントを指定できます。自動検出ルールを使用してNNMiでノードが検出されると、NNMiによってそのノードはデフォルトテナントに割り当てられます。検出後、ノードに対するテナントの割り当てを変更できます。

NNMiマルチテナントを計画および設定するための概略的な方法を以下に示します。

1. ユーザー要件を分析して、NNMi環境で必要なテナントの数を判別します。

   1つのNNMi管理サーバーで複数のネットワークを個々に管理する場合のみ、テナントを使用することをお勧めします。
2. 管理対象のネットワークトポロジを分析して、各テナントにどのノードが属するかを判別します。
3. 各テナントのトポロジを分析して、NNMiユーザーがアクセスする必要のあるノードのグループを判別します。
4. 事前に定義されたNNMiユーザーグループと、[デフォルトのセキュリティグループ] および [未解決のインシデント] セキュリティグループの間のデフォルトの関係を削除します。

   この手順により、ユーザーが管理してはならないノードへのアクセス権が、そのユーザーに間違って付与されることがないようにします。この時点では、NNMiトポロジ内のオブジェクトにアクセスできるのはNNMi管理者のみです。
5. 特定されたテナントを設定します。
   a. 特定されたセキュリティグループを作成します。
   b. 特定されたテナントを作成します。

      テナントごとに、[デフォルトのセキュリティグループ]、またはアクセスが制限されたテナント固有のセキュリティグループのいずれかに、[初期検出セキュリティグループ] を設定します。これを行うことで、NNMi管理者がアクセス権を設定するまで、テナントの新しいノードが全体に表示されることはなくなります。
6. テナントをシードに割り当てて、検出の準備を行います。

   ヒント: ノードのグループを検出した後、[初期検出セキュリティグループ] の値を変更できます。これを行うことで、ノードをセキュリティグループに手動で再割り当てする処理が制限されます。
7. 検出が完了したら、以下の手順を実行します。
   - ノードごとにテナントを確認し、必要に応じて変更します。
   - ノードごとにセキュリティグループを確認し、必要に応じて変更します。

   「設定の確認」(445ページ)を参照してください。

セキュリティグループの設定

ヒント: NNMiをディレクトリサービスと統合して、ユーザー名、パスワード、および必要に応じてNNMiユーザーグループの割り当ての保管場所を統合する場合は、NNMiセキュリティを設定する前に、その統合の設定を実行してください。

NNMiでは、以下の方法でセキュリティを設定できます。

- NNMiコンソールの [**セキュリティウィザード**] は、セキュリティ設定の可視化に役立ちます。[**変更概要の表示**] ページには、現在のウィザードセッションで保存されていない変更点のリストが表示されます。また、セキュリティ設定に関する潜在的な問題も示されます。
- 個々のセキュリティオブジェクトに対応したNNMiコンソールのフォームは、セキュリティ設定の1つの側面を同時に集中的に捉える場合に便利です。
- `nnmsecurity.ovpl`コマンドラインインタフェースは、自動操作や一括操作を行う場合に便利です。このツールは、セキュリティ設定に関する潜在的な問題のレポートも提供します。

NNMiトポロジ内のオブジェクトに対するユーザーのアクセス権を制限するためにNNMiセキュリティを定義および設定するプロセスは、循環的なプロセスです。この概略的な手順では、NNMiセキュリティを設定するための1つの方法を説明します。

ヒント: この例では、セキュリティグループからユーザーアカウントに移動します。たとえば、ユーザーアカウントからセキュリティグループにNNMiセキュリティを設定する場合、NNMiヘルプで「セキュリティの設定例」を検索してください。

NNMiセキュリティの設定に関しては、以下に注意してください。

- 検出されたノードにNNMiによって割り当てられるセキュリティグループは、そのノードに関連付けられたテナントの [初期検出セキュリティグループ] の値によって設定されます。
- NNMiテナントを設定しないで、NNMiセキュリティモデルを使用すると、すべてのノードがデフォルトテナントに割り当てられます。

NNMiセキュリティを計画および設定するための概略的な方法を以下に示します。

1. 管理対象のネットワークトポロジを分析して、NNMiユーザーがアクセスする必要のあるノードのグループを判別します。
2. 事前に定義されたNNMiユーザーグループと、[デフォルトのセキュリティグループ] および [未解決のインシデント] セキュリティグループの間のデフォルトの関係を削除します。

   この手順により、ユーザーが管理してはならないノードへのアクセス権が、そのユーザーに間違って付与されることがないようにします。この時点では、NNMiトポロジ内のオブジェクトにアクセスできるのはNNMi管理者のみです。
3. ノードの各サブセットのセキュリティグループを設定します。特定のノードは1つのセキュリティグループにのみ属することができます。
   a. セキュリティグループを作成します。
   b. 適切なノードを各セキュリティグループに割り当てます。
4. カスタムユーザーグループを設定します。
   a. セキュリティグループごとに、NNMiユーザーアクセスの各レベルに対応するユーザーグループを設定します。
      - ユーザーグループメンバーシップをNNMiデータベースに保存しても、それらのユーザーグループにユーザーはマッピングされません。

- ユーザーグループメンバーシップをディレクトリサービスに保存する場合は、各ユーザーグループの [ディレクトリサービス名] フィールドに、ディレクトリサービス内のそのグループの識別名を設定します。

b. 各カスタムユーザーグループを、適切なセキュリティグループにマッピングします。マッピングごとに適切なオブジェクトアクセス権限を設定します。

5. ユーザーアカウントを設定します。
   - ユーザーグループメンバーシップをNNMiデータベースに保存する場合は、以下の手順を実行します。
     - NNMiコンソールにアクセスできるユーザーごとに、ユーザーアカウントオブジェクトを作成します(ユーザーアカウントを設定するプロセスは、NNMiコンソールログオンにディレクトリサービスを使用しているかどうかによって異なります)。
     - 各ユーザーアカウントを、(NNMiコンソールにアクセスするために) 事前に定義したNNMiユーザーグループの1つにマッピングします。
     - 各ユーザーアカウントを (トポロジオブジェクトにアクセスするために) 1つ以上のカスタムNNMiユーザーグループにマッピングします。
   - ユーザーグループメンバーシップをディレクトリサービスに保存する場合、各ユーザーが、事前に定義されたNNMiユーザーグループの1つ、および1つ以上のカスタムユーザーグループに属していることを確認します。
6. 「設定の確認」(445ページ)の説明に従って、設定を確認します。
7. セキュリティ設定を管理します。
   - [デフォルトのセキュリティグループ] に追加されたノードに注目し、これらのノードを適切なセキュリティグループに移動します。
   - 新しいNNMiコンソールユーザーを適切なユーザーグループに追加します。

設定の確認

セキュリティ設定が適切であるかを確認するために、設定の各側面を別個に確認します。このセクションでは、設定を確認するためのいくつかの方法を説明します。ここに記載されていない方法も使用できます。

注: NNMiには、潜在的なセキュリティ設定エラーのレポートが備わっています。これらのレポートには、NNMiコンソールの **[ツール] > [セキュリティレポート]** で、`-displayConfigReport`オプションを`nnmsecurity.ovpl`コマンドに設定してアクセスします。

セキュリティグループとノード間の割り当てを確認する

各ノードが適切なセキュリティグループに割り当てられていることを確認する方法の1つとして、セキュリティグループごとに [**ノード**] または [**ノード (すべての属性)**] インベントリビューをソートし、グループ分けを調べる方法があります。

また、-listNodesInSecurityGroupオプションをnnmsecurity.ovplコマンドに指定して使用することもできます。

ユーザーグループとセキュリティグループ間の割り当てを確認する

どのユーザーグループが各セキュリティグループにマッピングされているかを確認する方法の1つとして、ユーザーグループまたはセキュリティグループごとに [**セキュリティグループのマッピング**] ビューをソートして、グループ分けを調べる方法があります。また、各マッピングのオブジェクトアクセス権限も確認します。

あるいは、[**セキュリティウィザード**] の [**ユーザーグループとセキュリティグループのマップ**] ページで、同時に1つのユーザーグループまたはセキュリティグループを選択して、そのオブジェクトに対する現在のマッピングを確認します。

また、-listUserGroupsForSecurityGroupオプションをnnmsecurity.ovplコマンドに指定して使用することもできます。

各ユーザーがNNMiコンソールアクセス権を持っているかを確認する

NNMiコンソールアクセス権について、事前に設定されたNNMiユーザーグループ (高い方から順に表示) の1つに各ユーザーが割り当てられていることを確認します。

- NNMi管理者
- NNMiレベル2オペレーター
- NNMiレベル1オペレーター
- NNMiゲストユーザー

その他のすべてのユーザーグループ割り当てで、NNMiデータベースのオブジェクトへのアクセス権が付与されます。

注: NNMiグローバルオペレーターユーザーグループでは、トポロジオブジェクトのみにアクセス権が与えられます。globalopsユーザーがNNMiコンソールにアクセスできるユーザーグループ (level2、level1、またはguestなど) に関連付けられていない場合、そのユーザーはNNMiコンソールにはアクセスできません。

NNMiコンソールアクセス権を持たないユーザーは、[**セキュリティウィザード**] の [**変更概要の表示**] ページにリストされます。**[ツール] > [セキュリティレポート]** メニュー項目で、-displayConfigReport usersWithoutRolesオプションをnnmsecurity.ovplコマンドに設定して、この情報を得ることもできます。

注: NNMiコンソールの各 [**ツール**] および [**アクション**] メニュー項目には、デフォルトのNNMiロールが関連付けられています(各 [アクション] メニュー項目に関連付けられているデフォルトのNNMiロールを確認するには、NNMiヘルプの「NNMiに用意されているアクション」を参照してください)。NNMiが提供するメニュー項目の設定をメニュー項目に割り当てられたデフォルトのNNMiロールよりも低いレベルのロールに変更すると、NNMiはその変更を無視します。デフォルトのNNMiロールよりも低いレベルのロールが割り当てられたすべてのユーザーグループは、メニュー項目にはアクセスできません。

ユーザーとユーザーグループ間の割り当てを確認する

ユーザーグループメンバーシップを確認する方法の1つとして、ユーザーアカウントまたはユーザーグループごとに [**ユーザーアカウントのマッピング**] ビューをソートして、グループ分けを調べる方法があります。

あるいは、[**セキュリティウィザード**] の [**ユーザーアカウントとユーザーグループのマップ**] ページで、同時に1つのユーザーアカウントまたはユーザーグループを選択して、そのオブジェクトに対する現在のマッピングを確認します。

また、`-listUserGroups`オプションと`-listUserGroupMembers`オプションを`nnmsecurity.ovpl`コマンドに指定して使用することもできます。

テナントとノード間の割り当てを確認する

各ノードが適切なテナントに割り当てられていることを確認する方法の1つとして、テナントごとに [**ノード**] または [**ノード (すべての属性)**] インベントリビューをソートし、グループ分けを調べる方法があります。

現在のユーザー設定を確認する

現在ログオンしているユーザーのNNMiコンソールアクセス権を確認するには、**[ヘルプ] > [システム情報]** をクリックします。[**製品**] タブの [**ユーザー情報**] セクションに、現在のNNMiセッションに関する以下の情報がリストされます。

- NNMiデータベースのユーザーアカウント、またはアクセス対象のディレクトリサービスに定義されているユーザー名。
- NNMiロール。これは、ユーザーがマッピングされる、事前に定義されたNNMiユーザーグループ (NNMi管理者、NNMiレベル2オペレーター、NNMiレベル1オペレーター、およびNNMiゲストユーザー) の中で最も高い権限を持つものに対応します。このマッピングによって、NNMiコンソールで使用できるアクションが決まります。
- このユーザー名にマッピングされたユーザーグループ。このリストには、NNMiロールを設定する事前に設定されたNNMiユーザーグループと、NNMiデータベース内のオブジェクトへのアクセス権を付与するその他のすべてのユーザーグループが含まれています。

NNMiのセキュリティおよびマルチテナント設定のエクスポート

以下の表に、NNMiのセキュリティおよびマルチテナント設定をエクスポートするための設定エリア (`nnmconfigexport.ovpl -c`で利用可能) を示します。これらのエクスポートエリアは、特にグローバルネットワーク管理環境で、複数のNNMi管理サーバーにわたって設定を管理するのに役立ちます。

NNMiのセキュリティおよびマルチテナント設定のエクスポートエリア

設定エリア	説明
account	ユーザーアカウント、ユーザーグループ、およびユーザーアカウントとユーザーグループ間のマッピングをエクスポートします。

NNMiのセキュリティおよびマルチテナント設定のエクスポートエリア (続き)

設定エリア	説明
	複数のNNMiデータベースにわたってユーザー定義を共有するのに便利です。
セキュリティ	テナントおよびセキュリティグループをエクスポートします。 複数のNNMiデータベースにわたってセキュリティ定義を共有するのに便利です。 この情報をインポートすると、新しいオブジェクトが作成され、既存のオブジェクトが更新されますが、現在のエクスポートに含まれていないオブジェクトは削除されません。このため、ローカルで定義されたオブジェクトがNNMiデータベースに含まれている場合でも、このオプションは安全に使用できます。
securitymappings	ユーザーグループとセキュリティグループ間のマッピングをエクスポートします。 セキュリティとマルチテナント設定を完全にエクスポートするには、account、security、およびsecuritymappings設定エリアの同時エクスポートを実行してください。

NNMiセキュリティ、マルチテナント、およびグローバルネットワーク管理 (GNM)

グローバルネットワーク管理 (GNM) 環境では、ノードのテナントは、そのノードを管理するNNMi管理サーバーに設定されます。GNM環境では、指定されたノードのテナントUUIDは各グローバルマネージャーとリージョナルマネージャーで同じです。

ノードのセキュリティグループは、トポロジにそのノードが含まれる各NNMi管理サーバーに設定されます。したがって、トポロジ内のオブジェクトへのユーザーアクセスは、GNM環境の各NNMi管理サーバーに別個に設定されます。グローバルマネージャーとリージョナルマネージャーが使用するセキュリティグループ定義は、同じである場合も、異なる場合もあります。

グローバルマネージャーとリージョナルマネージャーに同様のユーザーアクセスを設定する場合、いくつかの裏技を使用して設定することもできますが、大部分の場合、各NNMi管理サーバーにカスタム設定を行う必要があります。

注: 動的ネットワークアドレス変換 (NAT) または動的ポートアドレス変換 (PAT) の各グルーにプは、NNMiグローバルネットワーク全体の管理設定内で一意のテナントに加えて、NNMiリージョナルマネージャーが必要です。「NAT環境の重複IPアドレスの管理」(407ページ)を参照してください。NNMiヘルプも参照してください。

ヒント: グローバルマネージャーにすべてのテナントとセキュリティグループを定義します。`nnmconfigexport.ovpl -c security`を使用して、テナントとセキュリティグループ定義をエクスポートします。各リージョナルマネージャーで、`nnmconfigimport.ovpl`を使用してテナントとセキュリティグループ定義をインポートします。あるいは、`nnmsecurity.ovpl`コマンドを使用して、別のNNMi管理サーバーのUUIDと同じUUIDを使用して、テナントおよびセキュリティグループを作成することができます。この推奨手順に従うことで、GNM環境内で、各テナントとセキュリティグループのUUIDを同じにすることができます。

注: ユーザーがグローバルマネージャーからNPSレポートを開始する場合、このベストプラクティスは設定の必須部分になります。

注: テナントUUIDは一意である必要がありますが、テナント名は再利用できます。NNMiは、名前が同じでUUIDが異なる2つのテナントを、共有設定を持たない2つの別個のテナントであると見なします。

ヒント: 組織ごとに1つのリージョナルマネージャーをセットアップする場合は、リージョナルマネージャーのすべてのノードを1つのテナントに入れることができます。ただし、各リージョナルマネージャーに一意のテナントを設定し、グローバルマネージャーでトポロジデータが確実に分離されるようにしてください。

リージョナルマネージャーからグローバルマネージャーに転送されたインシデントに、セキュリティ情報とテナント情報を伝達するいくつかの追加カスタムインシデント属性 (CIA) が含まれる場合があります。

このようなインシデントのソースオブジェクトが [デフォルトテナント] 以外のテナントに属している場合、転送されるインシデントには以下のCIAが含まれます。

- cia.tenant.name
- cia.tenant.uuid

このようなインシデントのソースオブジェクトが [デフォルトのセキュリティグループ] 以外のセキュリティグループに属している場合、転送されるインシデントには以下のCIAが含まれます。

- cia.securityGroup.name
- cia.securityGroup.uuid

初期GNM設定

グローバルネットワーク管理 (GNM) の初期設定後、リージョナルマネージャーは、(GNM設定に従って) リージョナルトポロジ内のノードに関する情報を使用して、グローバルマネージャーを更新します。

デフォルトテナントのみとのトポロジの同期

カスタムセキュリティグループとデフォルトテナントを持つGNM環境の場合、グローバルマネージャーでは、リモートで管理されているすべてのノードが、以下の設定でグローバルマネージャートポロジに追加されます。

- デフォルトテナント
- デフォルトテナントの [初期検出セキュリティグループ] として設定されるセキュリティグループ。

カスタムテナントとのトポロジの同期

カスタムセキュリティグループとカスタムテナントを持つGNM環境の場合、グローバルマネージャーでは、リモートで管理されているすべてのノードが、そのノードに割り当てられているテナントのUUIDを使用して、グローバルマネージャートポロジに追加されます。そのテナントUUIDがグローバルマネージャーにない場合、以下のように、GNMプロセスによってグローバルマネージャーのNNMi設定にテナントが作成されます。

- このテナントUUIDは、リージョナルマネージャーの場合と同じ値です。
- テナント名は、リージョナルマネージャーの場合と同じ値です。
- [初期検出セキュリティグループ] の値は、テナントと同じ名前のセキュリティグループに設定されます(このセキュリティグループがグローバルマネージャーにない場合、NNMiによってそのセキュリティグループが作成されます)。

グローバルマネージャーのトポロジにノードが追加されると、そのノードは、グローバルマネージャーに設定されたテナントUUIDに対応する [初期検出セキュリティグループ] に割り当てられます。このため、グローバルマネージャー上でのセキュリティグループの関連付けは、リージョナルマネージャー上でのセキュリティグループの関れ付けから独立しています。

ヒント: グローバルマネージャーでのセキュリティ設定を簡素化するために、以下をお勧めします。

- 各リージョナルマネージャーによって管理されるノードのスプレッドシートまたはその他のレコードを保持します。ノードごとに、リージョナルマネージャーとグローバルマネージャーのそれぞれに必要なセキュリティグループをメモしておきます。GNM設定が完了したら、`nnmsecurity.ovpl`コマンドを使用して、セキュリティグループの割り当ての確認および更新を行います。
- GNM環境で、複数のリージョナルマネージャーによって1つのグローバルマネージャーが更新されている場合、そのグローバルマネージャーに対してGNM設定を有効にするには、各リージョナルマネージャーから1つずつ設定を行ってください。

  該当する場合は、各リージョナルマネージャーをGNM設定に追加する前に、デフォルトテナント (またはカスタムテナント) の [初期検出セキュリティグループ] の値を変更できます。これを実行した場合、以前に設定されたリージョナルマネージャーのトポロジに新しいノードが追加されると、さまざまな結果が生じる可能性があることに注意してください。
- GNMを有効にする前に、グローバルマネージャー上で、リージョナルマネージャーで使用される各テナントの [初期検出セキュリティグループ] を、オペレーターがアクセスできない専用セキュリティグループに設定してください。これにより、グローバルマネージャー上の管

理者は、ほかのNNMiコンソールオペレーターのために、ノードを適切なセキュリティグループに明示的に移動しなくてはならなくなります。

GNMのメンテナンス

以下の表は、リージョナルマネージャーでのノードのテナントまたはセキュリティグループの割り当てへの変更が、グローバルマネージャーにどのように影響を及ぼすかを示しています。

リージョナルマネージャーでの設定変更がグローバルマネージャーに及ぼす影響

アクション	影響
リージョナルマネージャーで、ノードを別のテナントに割り当てる。	グローバルマネージャーのノードは、その別のテナントに割り当てられるように変更されます。このテナントUUIDがグローバルマネージャーにない場合は作成されます。
リージョナルマネージャーで、ノードを別のセキュリティグループに割り当てる。	グローバルマネージャーでは変更は行われません。NNMi管理者は、その変更を手動で複製するように選択できます。
リージョナルマネージャーで、テナントの設定 (名前、説明、または初期検出セキュリティグループ) を変更する。	グローバルマネージャーでは変更は行われません。NNMi管理者は、その変更を手動で複製するように選択できます。
リージョナルマネージャーで、セキュリティグループの設定 (名前または説明) を変更する。	グローバルマネージャーでは変更は行われません。NNMi管理者は、その変更を手動で複製するように選択できます。

NPSレポートへの選択インタフェースの追加

Network Performance Server (NPS) は、NNM iSPI Performance for Metricsソフトウェアとともにインストールされるデータベースサーバーです。

デフォルトで、ノードのすべてのコンポーネントは、そのノードと同じセキュリティグループに属します。個々のインタフェースに対して、このデフォルトの動作をオーバーライドし、インタフェースを別のセキュリティグループに割り当てることができます。このオーバーライドは、共有デバイスのテナント (顧客) 向けの適切なインタフェースを含むテナント固有のレポートを生成するために行います。このようにすると、各顧客には、自分のインタフェースに関するインタフェース情報が表示され、デバイス上のほかのインタフェースは表示されないようになります。

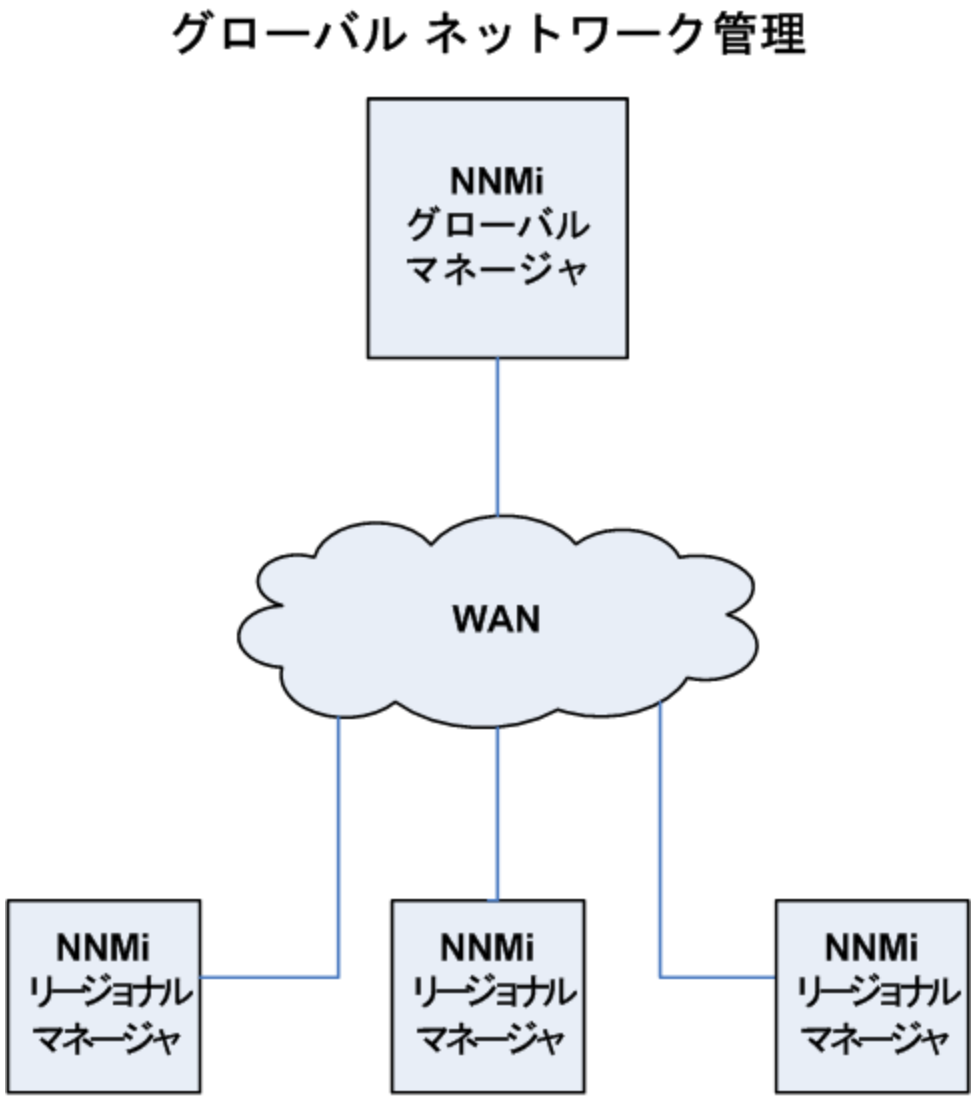
注: セキュリティグループのオーバーライドは、NPSレポートにのみ反映されます。NNMiコンソールでユーザーに表示される内容や、ユーザーが実行できる事柄には影響は及ぼされません。

インタフェースのセキュリティグループ割り当てを変更するには、[**インタフェース**] フォームの [**カスタム属性**] タブ、または`nnmloadattributes.ovpl`コマンドを使用して、`InterfaceSecurityGroupOverride`カスタム属性をインタフェースに追加します。このカスタム属性の値をセキュリティグループのUUIDに設定します。例:

```
InterfaceSecurityGroupOverride=0826c95c-5ec8-4b8c-8998-301e0cf3c1c2
```

注: インタフェースは、同時に1つのセキュリティグループにしか属すことができません。インタフェースに`InterfaceSecurityGroupOverride`カスタム属性を設定すると、そのインタフェースと、ノードが属するセキュリティグループの間の関連付けが壊れます。

グローバルネットワーク管理



この章には、以下のトピックがあります。

グローバルネットワーク管理の利点

HP Network Node Manager i Software (NNMi) を地理的位置が異なる複数のNNMi管理サーバーに導入しているとします。各NNMi管理サーバーでは、検出と監視のニーズに合うように、ネットワークの検出および監視を行っています。こうした既存のNNMi管理サーバーと設定を使用して、特定のNNMi管理サーバーをグローバルマネージャーとして指定することで、新たな検出を追加したりモニタリングの設定を変更したりせずに、集約したノードオブジェクトデータを表示することができます。

NNMiグローバルネットワーク管理機能により、地理的位置が異なるネットワークを管理しながら、複数のNNMi管理サーバーを連携させることができます。特定のNNMi管理サーバーをグローバルマネージャーとして指定し、複数のリージョナルマネージャーを集約したノードオブジェクトデータを表示します。

NNMiグローバルネットワーク管理機能には、以下の利点があります。

- グローバルマネージャーから見た、企業のネットワークの全体像を表示できます。
- 以下のように容易に設定できます。

  - リージョナルマネージャーの管理者はそれぞれ、すべてのノードオブジェクトデータを指定するか、またはグローバルマネージャーレベルで参加する特定のノードグループを指定します。
  - 各グローバルマネージャーの管理者は、情報の提供を許可するリージョナルマネージャーを指定します。
- 各サーバーごとに、インシデントの生成と管理を行うことができます (各サーバーで使用可能なトポロジのコンテキスト内で生成されます)。

詳細については、NNMiヘルプの「NNMiのグローバルネットワーク管理機能」を参照してください。

注: 動的ネットワークアドレス変換 (NAT)、動的ポートアドレス変換 (PAT)、または動的ネットワークアドレスおよびポート変換 (NAPT) の各グループには、NNMiグローバルネットワーク管理設定全体で一意のテナントに加え、NNMiリージョナルマネージャーが必要です。「NAT環境の重複IPアドレスの管理」(407ページ)を参照してください。NNMiヘルプも参照してください。

グローバルネットワーク管理が自分のネットワークの管理に適しているかどうかを判断するには

このセクションに含まれる質問は、 NNMiグローバルネットワーク管理機能がネットワーク管理の向上に役立つかどうかを判断するのに役立ちます。

マルチサイトネットワークを継続的に監視する必要がありますか?

ITグループは、複数のサイトに配備されているネットワーク機器を 週7日、24時間体制で管理していますか?NNMiのグローバルネットワーク管理機能を使用すれば、トポロジとインシデントを集約して表示し、確認することができるようになります。

重要デバイスを表示できるか?

複数の場所に配備された重要デバイスのステータスとインシデントを、1つのNNMi管理サーバーで表示できますか?

はい。リージョナルマネージャーに転送フィルターを設定します。このフィルターにより、リージョナルマネージャーからグローバルマネージャーに送信するノードオブジェクトデータを選択できます。たとえば、リージョナルマネージャーに対し転送フィルターを設定して、重要デバイスに関する情報のみをグローバルマネージャーに転送するようにできます。

ライセンスの考慮事項

NNMiライセンスキーの取得とインストールの詳細については、「NNMiのライセンス」(312ページ)を参照してください。

グローバルマネージャーおよびリージョナルマネージャーの両方で、NNMi Advancedライセンス、NNMi Premiumライセンス、またはNNMi Ultimateライセンスが必要ですか?

グローバルマネージャーとして使用するNNMi管理サーバーには、NNMi Advancedライセンス、NNMi Premiumライセンス、またはNNMi Ultimateライセンスを購入してインストールする必要があります。

NNMiリージョナルマネージャーは、NNMiライセンス、NNMi Advancedライセンス、NNMi Premiumライセンス、またはNNMi Ultimateライセンスでライセンス供与することができます。

1つの地域をカバーするのに十分なNNMiライセンスを持っています。グローバルネットワーク管理機能を使用しながら、グローバルマネージャーに必要な新しいライセンスの数を抑えることはできますか?

いいえ。 グローバルマネージャーに十分なNNMi Advancedライセンス、NNMi Premiumライセンス、またはNNMi Ultimateライセンスを購入してインストールし、グローバルマネージャーでローカルでモニタリングされるノードの数を満たすか超えるようにする必要があります。NNMiは、グローバルマネージャーのライセンスに対して、さまざまなリージョナルマネージャーのノードをカウントしません。

ライセンスを取得したノードの総数がグローバルマネージャーのNNMi Advancedライセンス容量、NNMi Premiumライセンス容量、またはNNMi Ultimateライセンス容量より多くなるように、リージョナルマネージャー用にNNMiライセンスを増やしました。グローバルマネージャーには、すべての領域のすべてのノードの完全なインベントリがありません。十分なライセンスをグローバルマネージャー用に購入した後で、グローバルマネージャーをすべてのリージョナルマネージャーと同期させて、ライセンスが不十分だったために前回省略したノードを検索して作成するにはどうしたら良いでしょうか。

グローバルマネージャーでトポロジを再同期するには、以下のいずれかを実行します。

- すべてのリージョナルマネージャーで設定されている、すべての再検出間隔の時間が経過して、すべての領域ですべてのノードが再検出されるのを待機します。リージョナルマネージャーは、すべての領域ですべてのノードを再検出したら、再検出されたノードの情報をグローバルマネージャーに送信します。グローバルマネージャーはこのノード情報を受信し、各領域でノードごとにグローバルノードを作成します。
- 各リージョナルマネージャーで`nnmnoderediscover.ovpl -all` スクリプトを実行します。

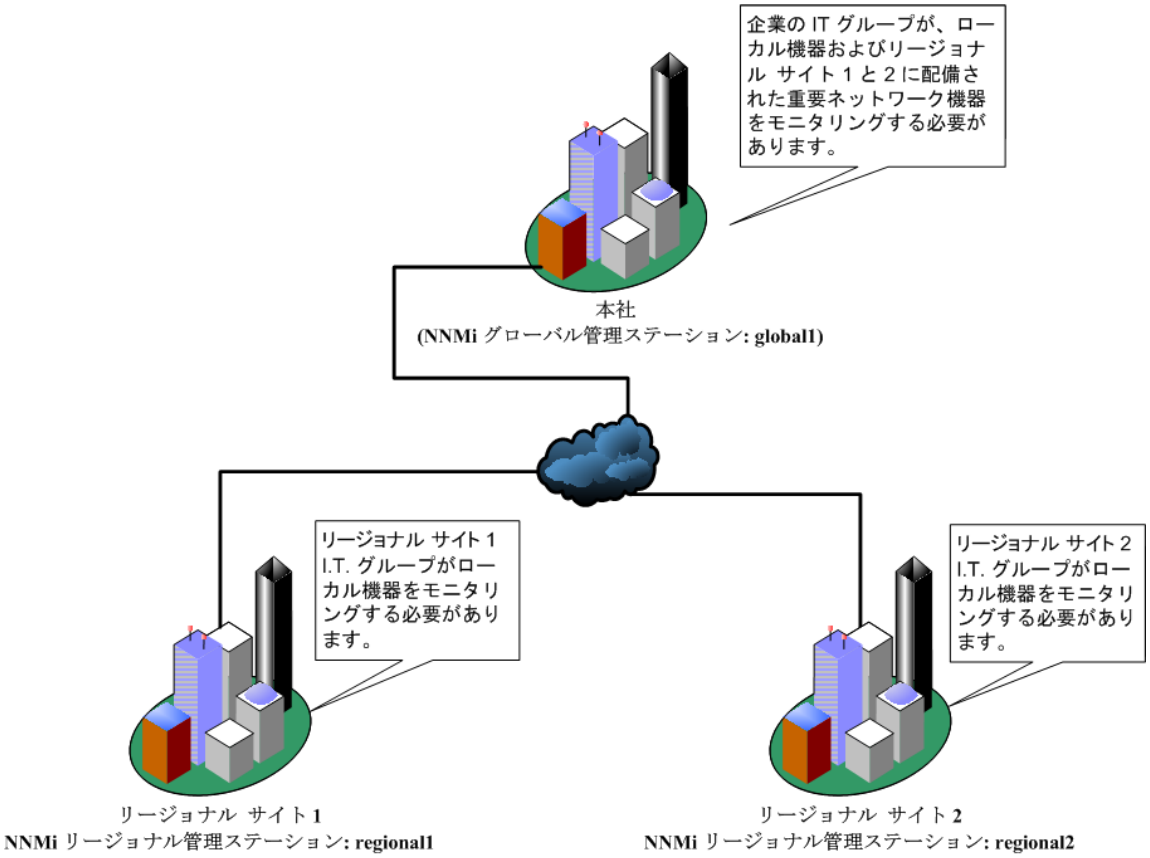
注: 2番目のオプションでは、ネットワーク上のトラフィックが増加し、NNMiマネージャーのセット全体から多くのNNMiリソースが消費されることにもなります。このオプションは、最初のNNMi検出ほどリソースの多くを消費しませんが、最初の検出を実行することに似ています。最適な方法では、ある程度の時間をおくか、現在のリージョナルマネージャーの負荷が減って正常になるのを待ち、領域ごとに間隔をおいてスクリプトを実行してから、次のリージョナルマネージャーの再検出を始めます。

実践的なグローバルネットワーク管理の例

以下の図を参照してください。この例の場合、会社には地理的位置が異なる2つの運用サイトがあります。本社は、運用サイトとは別の地理的位置にあります。つまり、全部で3か所でNNMi管理サーバーが機能しています。

本社のIT担当者が、ローカルネットワーク機器およびリージョナルサイト1と2の両方に配備された重要ネットワーク機器を、ネットワークの観点から監視する必要があります。リージョナルサイト1と2両方のIT担当者は、そぞれのサイトに配備されている重要なネットワーク機器を監視する必要があります。

ネットワークの例



要件のレビュー

この例の場合、本社、リージョナルサイト1、リージョナルサイト2のNNMi管理サーバーが、それぞれのサイトに配備された複数のルーターとスイッチを管理します。

この例では、NNMi管理サーバーをそれぞれglobal1、regional1およびregional2と見なします。

これらのNNMi管理サーバーは、それぞれの場所に配備された重要なスイッチとルーターの検出とモニタリングを行うように設定されています。

ヒント: グローバルネットワーク管理機能を使用するために、これらのサイトにあるNNMi管理サーバーでの検出を再設定する必要はありません。

注: グローバルネットワーク管理機能の設定中、`nnmbackup.ovpl`スクリプトを使って1つのNNMi管理サーバーをバックアップし、`nnmrestore.ovpl`スクリプトを使ってこのバックアップを第2のNNMi管理サーバーに復元し、この両方のNNMi管理サーバーをリージョナルNNMi管理サーバーに接続してみる場合があります。このようなことはしないでください。あるNNMi管理サーバーから2番目のNNMi管理サーバーにバックアップデータを配置すると、これらの両方のサーバーに同じデータベースUUIDが存在することになります。NNMiを第2のNNMi管理サーバーに復元した後、元のNNMi管理サーバーからNNMiをアンインストールする必要があります。

NNMiをアンインストールする前に、最新のパッチから開始して、NNMiパッチをすべて逆順で削除します。パッチの削除プロセスは、NNMi管理サーバーで実行しているオペレーティングシステムによって異なります。インストールおよび削除手順については、パッチのマニュアルを参照してください。

本社ITグループでは、リージョナルサイト1と2に配備された重要な機器のみの監視を行い、ほかのデバイスの管理はしない予定です。

以下の表に、監視のニーズをまとめます。

グローバルネットワーク管理のネットワーク要件

サイト	NNMi管理サーバー	重要なスイッチ	管理するリージョナル機器
本社	global1	15台のModel 3500yl HP Procurve Switch	各リージョナルサイトのModel 3500yl HP ProCurve Switchすべて
リージョナルサイト1	regional1	15台のModel 3500yl HP Procurve Switch	該当なし
リージョナルサイト2	regional2	15台のModel 3500yl HP Procurve Switch	該当なし

要約すると、以下のようになります。

- NNMi管理サーバー `global1`が本社をモニタリングします。
- NNMi管理サーバー `regional1`と`regional2`が各リージョナルサイトをモニタリングします。
- リージョナルサイト1と2に配備されたModel 3500yl ProCurve Switchのインシデントとデバイス情報を、本社で表示する必要があります。
- `regional1`と`regional2`の両方で、リージョナルサイト1に配備された複数の共通スイッチを管理します。

リージョナルマネージャーとグローバルマネージャーの接続

グローバルネットワーク管理接続を設定するときに、以下の情報を考慮します。

- グローバルマネージャーとすべてのリージョナルマネージャーで、同じNNMiバージョンおよびパッチレベルを使用します。異なるNNMiバージョンを使用したグローバルネットワーク管理設定はサポートされていません。
- NNMiでは、リージョナルマネージャーと通信する1つ以上のグローバルマネージャーを設定できます。たとえば、regional1と通信するために第2のグローバルマネージャー、global2が必要な場合、NNMiでは、regional1と通信するglobal1とglobal2の両方を設定できます。詳細については、『HP Network Node Manager i Softwareシステムとデバイス対応マトリックス』を参照してください。
- グローバルネットワーク管理は、1つの接続レイヤーで動作します。たとえば、この章の例では、1つの接続レイヤー、regional1と通信するglobal1とregional2と通信するglobal1について検討します。NNMiは、複数の接続レベルを設定しないでください。たとえば、global1はregional1と通信する設定にはせず、regional1はregional2と通信する設定にします。グローバルネットワーク管理機能は、この3つのレイヤー設定用に設計されています。
- 2つのNNMi管理サーバーは、相互に両方向に通信する設定にはしないでください。たとえば、global1はregional1と通信する設定にはせず、regional1はglobal1と通信する設定にします。

初期準備

このセクションでは、シナリオ例のグローバルネットワーク管理の設定に必要な初期準備について説明します。

ポート可用性:ファイアウォールの設定

グローバルネットワーク管理機能が正しく機能するためには、global1から regional1とregional2へのTCPアクセス用に、特定のウェルノウンポートが開いているかどうかを確認する必要があります。NNMiインストールスクリプトでは、デフォルトとしてポート80と443を設定します。ただし、インストール中にこれらの値は変更できます。

注: このセクションで説明した例では、global1がregional1とregional2へのTCPアクセスを確立します。ファイアウォールは、一般的に接続を開始するサーバーに基づいて設定されます。global1がregional1とregional2への接続を確立すると、トラフィックは両方向に流れます。

現在の値を確認したりポート設定を変更したりするには、以下のファイルを編集します。

- Windowsの場合: %NNM_CONF%\nnm\props\nms-local.properties
- Linuxの場合: $NNM_CONF/nnm/props/nms-local.properties

以下の表に、アクセス可能にしておく必要があるウェルノウンポートを示します。

アクセス可能にしておく必要があるソケット

セキュリティ	パラメーター	TCPポート
非SSL	nmsas.server.port.web.http	80
	nmsas.server.port.hq	4457
SSL	nmsas.server.port.web.https	443
	nmsas.server.port.hq.ssl	4459

詳細については、「NNMiおよびNNM iSPIのデフォルトポート」(506ページ)を参照してください。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

自己署名証明書の設定

`global1`と2つのリージョナルNNMi管理サーバー (`regional1`と`regional2`) 間でSSL (Secure Sockets Layer) を使用してグローバルネットワーク管理機能を使用する場合は、自己署名証明書を設定する必要があります。

NNMiのインストール中、NNMiインストールスクリプトでは、他のエンティティに対して自身を識別できるよう、NNMi管理サーバーに自己署名証明書を作成します。使用するNNMi管理サーバーには、正しい証明書を持つグローバルネットワーク管理機能を設定する必要があります。「グローバルネットワーク管理環境での証明書の使用」(328ページ)に示した手順を実行してください。

グローバルネットワーク管理でアプリケーションフェイルオーバーの設定を行う

NNMiのインストール中、NNMiインストールスクリプトでは、他のエンティティに対して自身を識別できるよう、NNMi管理サーバーに自己署名証明書を作成します。

グローバルネットワーク管理機能とともにアプリケーションフェイルオーバーを使用するには、「グローバルネットワーク管理でアプリケーションフェイルオーバーの設定を行う 」(473ページ)の説明に従って手順を実行する必要があります。

NNMi管理サーバー規模の考慮事項

この例では、グローバルネットワーク管理設定で既存のNNMi管理サーバーを使用することを想定しています。

NNMiのインストールが必要となるサーバーのサイズに関する具体的な情報については、HP Network Node Manager i Softwareインタラクティブインストールガイド、NNMiリリースノート, およびNNMi対応マトリックスを参照してください。

システムクロックの同期

global1、regional1、およびregional1サーバーをグローバルネットワーク管理設定に接続する前に、これらのNNMi管理サーバークロックを同期することが重要です。

注: グローバルネットワーク管理 (グローバルマネージャーとリージョナルマネージャー) やシングルサインオン (SSO) に属するネットワーク環境内のすべてのNNMi管理サーバーは、それぞれの内部タイムクロックを世界標準時で同期する必要があります。

たとえば、LinuxツールのNetwork Time Protocol Daemon (NTPD) や使用可能なWindowsオペレーティングシステムツールなどの時刻の同期プログラムを使用します。詳細については、NNMiヘルプの「クロック同期の問題」または「グローバルネットワーク管理のトラブルシューティング」と「クロック同期 」(474ページ) を参照してください。

注: サーバークロック同期の問題など、リージョナルマネージャーとの接続に問題がある場合、NNMiではNNMiコンソールの下部に警告メッセージが表示されます。

グローバルネットワーク管理で自己署名証明書を使用する場合のアプリケーションフェイルオーバー機能の使用法

アプリケーションフェイルオーバー設定で、自己署名証明書を使用したグローバルネットワーク管理機能を使用する場合は、「フェイルオーバーが有効なグローバルネットワーク管理環境での証明書の設定」(330ページ) の手順を実行します。

グローバルネットワーク管理における自己署名証明書の使用法

自己署名証明書を使用したグローバルネットワーク管理機能を使用する場合は、「グローバルネットワーク管理環境での証明書の使用」(328ページ)の手順を実行する必要があります。

グローバルネットワーク管理における認証機関の使用法

認証機関を使用したグローバルネットワーク管理機能を使用する場合は、「グローバルネットワーク管理環境での証明書の使用」(328ページ)の手順を実行する必要があります。

監視する重要な機器の一覧作成

各リージョナルマネージャーによって管理され、グローバルマネージャーからモニタリングされる機器のリストを作成します。たとえば、global1からモニタリングされるregional1とregional2の管理対象機器リストを作成します。この情報を転送フィルターで使用します。詳細については、「リージョナルマネージャーでの転送フィルターの設定」(465ページ)を参照してください。

ヒント: regional1とregional2からglobal1に転送する情報を制限した場合に得られる結果については、慎重に考慮する必要があります。計画を立てるときに、以下の点を考慮してください。

- global1で完全な分析を行って正確なインシデントを生成するには、regional1とregional2から得られる完全なトポロジが必要になるため、除外するデバイスが多くなりすぎないように注意します。
- 重要ではないデバイスを除外すると、global1のシステムパフォーマンスコストを節約できます。
- 重要ではないデバイスを除外すると、ソリューションの全体的な拡張性が改善され、NNMiで必要となるネットワークトラフィックを削減できます。

グローバルマネージャーとリージョナルマネージャーの管理ドメインの検討

リージョナルマネージャーからグローバルマネージャーに転送する情報を決定するために、グローバルマネージャーとリージョナルマネージャーの管理ドメインを検討します。

この例では、NNMi管理サーバー global1、regional1、およびregional2は、独自のノードセットを管理しています。この例では、後でregional1とregional2からglobal1に、それぞれが管理する機器に関する情報を転送するよう設定します。

以下の手順を実行して、global1、regional1、およびregional2が現在監視している機器を確認します。機器を確認しておくと、regional1とregional2からglobal1に転送する重要な機器を選択するときに役立ちます。

この例では、以下の手順を実行してこの情報を確認します。

1. ブラウザーでglobal1のNNMiコンソールを指します。
2. サインインします。
3. [**インベントリ**] ワークスペースをクリックします。
4. このワークスペースでglobal1が現在監視していて検出されたインベントリを確認できます。
5. ブラウザーでregional1のNNMiコンソールを指します。
6. サインインします。
7. [**インベントリ**] ワークスペースをクリックします。
8. regional1が監視しているノードを確認し、global1で監視するデバイスの一覧を作成します。
9. ブラウザーでregional2のNNMiコンソールを指します。
10. サインインします。
11. [**インベントリ**] ワークスペースをクリックします。
12. regional2が監視しているノードを確認し、global1で監視するデバイスの一覧を作成します。

NNMiヘルプトピックの確認

グローバルネットワーク管理に関するすべてのヘルプトピックを確認するには、以下の手順を実行します。

1. NNMiヘルプで、[**検索**] をクリックします。
2. [検索] フィールドに「**グローバルネットワーク管理**」と入力します。
3. [**検索**] をクリックします。

この検索により、グローバルネットワーク管理に関連する50以上のトピックが見つかります。

SSOおよびアクションメニュー

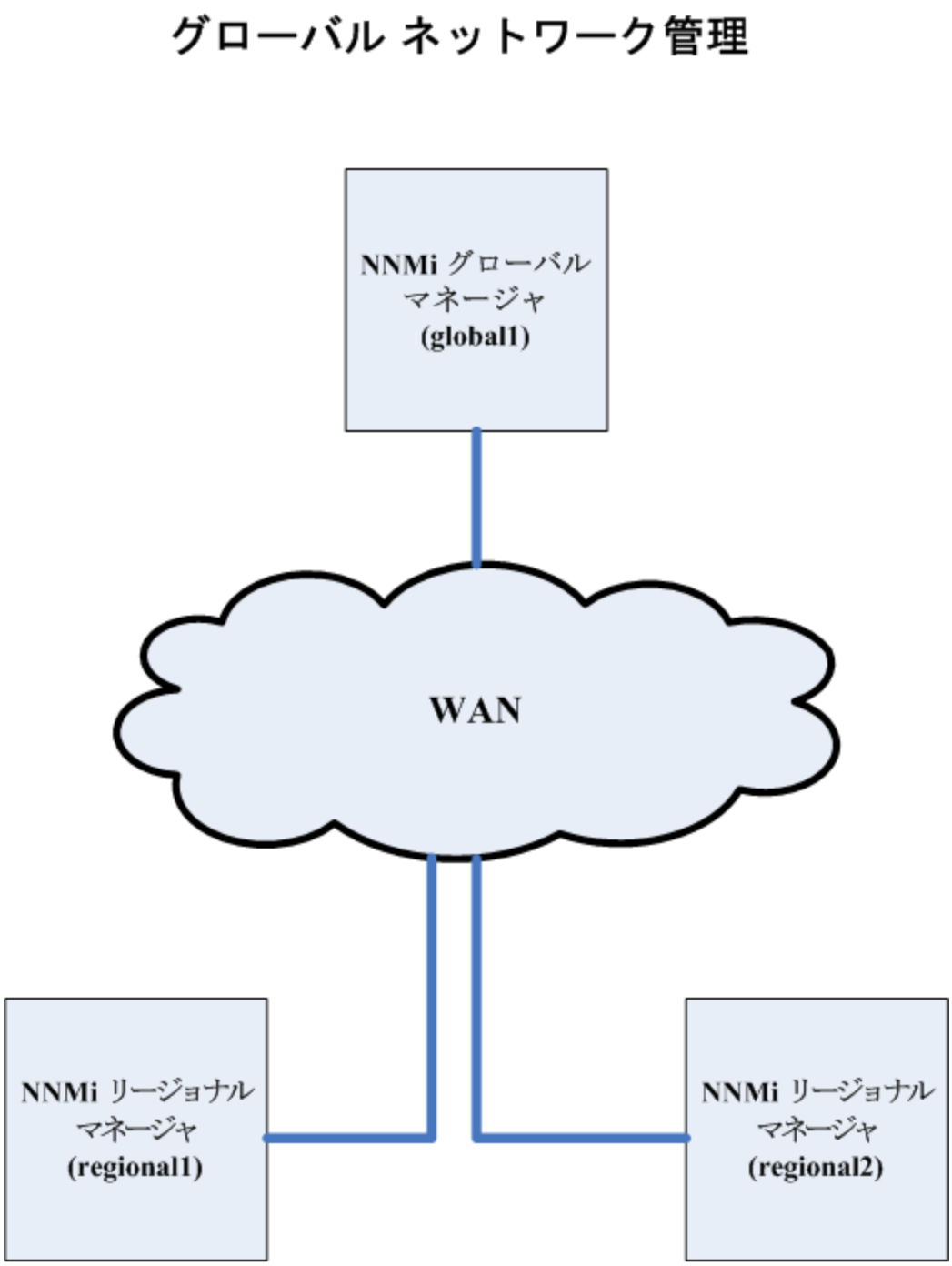
グローバルマネージャーのNNMiコンソールから、リージョナルマネージャーが管理するノードを選択した後に、[**アクション**] メニューを使用して、選択したノードに対するアクションを開始できます。

NNMi管理サーバーの間で`initString`と`domain`のパラメーターを同一にしないと、グローバルマネージャーのセッション情報は新しいセッションに渡されず、アクションは開始されません。この問題を回避するには、「グローバルネットワーク管理用にシングルサインオンを設定する」(462ページ)の設定手順に従ってください。

グローバルネットワーク管理用にシングルサインオンを設定する

NNMiシングルサインオン (SSO) を設定すると、NNMiグローバルマネージャーから簡単にNNMiリージョナルマネージャーにアクセスできるようになります。

注: グローバルマネージャーからリージョナルマネージャーに接続する前に、シングルサインオンを設定しておく必要があります。詳細については、「NNMiとシングルサインオン (SSO) の使用」(333ページ)を参照してください。



SSO機能は、NNMi管理サーバー内のユーザー名を交換しますが、パスワードやロールは交換しません。たとえば、NNMiは1つのNNMi管理サーバー (global1) の特定のユーザー名を、別のNNMi管理サーバー (regional1またはregional2) の異なるロールに関連付けます。3つのNNMi管理サーバーで、同じユーザー名に異なるパスワードが関連付けられることもあります。

グローバルマネージャーとリージョナルマネージャーが同じ管理ドメインにあり、手順4に示したように初期化ストリング値をグローバルNNMi管理サーバーからリージョナルNNMi管理サーバーにコ

ピーしないと、NNMiコンソールのアクセスに問題が起こる場合があります。これを回避するには、以下の手順を実行してSSOを正しく設定するか、「SSOの無効化」(338ページ) の説明に従ってSSOを無効にします。

SSOをグローバルネットワーク管理機能と連携させるには、以下の手順を実行します。

1. global1、regional1、regional2で以下のファイルを開きます。
   - Windowsの場合: %NNM_PROPS%\nms-ui.properties
   - Linuxの場合: $NNM_PROPS/nms-ui.properties
2. global1、regional1、およびregional2ファイルで、以下のようなセクションを探します。

   com.hp.nms.ui.sso.isEnabled = false

   これを以下のように変更します。

   com.hp.nms.ui.sso.isEnabled = true
3. global1のSSO NNMi初期化文字列を探します。nms-ui.propertiesファイルから、以下のようなセクションを特定します。

   com.hp.nms.ui.sso.initString =Initialization String
4. global1のnms-ui.propertiesファイルにある初期化ストリングの値を、regional1とregional2のnms-ui.propertiesファイルにコピーします。初期化文字列は、すべてのサーバーで同じ値を使用する必要があります。変更を保存します。

   注: グローバルNNMi管理サーバーから リージョナルNNMi管理サーバーへのInitialization String値のコピーはNNMiでサポートされます。 この操作により、グローバルマネージャーから2つのリージョナルマネージャーにInitialization String値がコピーされます。グローバルネットワーク管理機能でSSOを使用する場合は、Initialization String値のコピーは、常にグローバルマネージャーからリージョナルマネージャーに対して行ってください。

   注: グローバルマネージャーとリージョナルマネージャーが同じ管理ドメインにあり、Initialization String値をグローバルNNMi管理サーバーからリージョナルNNMi管理サーバーにコピーしない場合は、SSOを無効にして、NNMiコンソールのアクセスに問題が起こらないようにします。詳細については、「SSOの無効化」(338ページ)を参照してください。
5. global1、regional1、およびregional2が異なるドメインにある場合は、protectedDomainsの内容を変更します。変更するには、nms-ui.propertiesファイルの中から以下のようなセクションを探します。

   com.hp.nms.ui.sso.protectedDomains=group1.mycompany.com

   global1はglobal1.company1.comに、regional1はregional1.company2.comに、そして、regional2はregional2.company3.comにあるとします。global1、regional1、regional2にあるnms-ui.propertiesファイルのprotectedDomainsセクションを以下のように変更します。

   com.hp.nms.ui.sso.protectedDomains=regional1.company1.com,
   regional2.company2.com,regional3.company3.com

6. 変更を保存します。
7. global1、regional1、regional2で、以下の一連のコマンドを実行します。
   a. ovstop
   b. ovstart

注: アプリケーションフェイルオーバー設定でシングルサインオンを有効にするときに、手動で行う設定手順はありません。たとえば、アプリケーションフェイルオーバー設定でシングルサインオンを設定する場合、NNMiによりアクティブNNMi管理サーバーからスタンバイNNMi管理サーバーに上記の変更を複製されます。

リージョナルマネージャーでの転送フィルターの設定

この例では、global1はregional1とregional2の両方と通信します。グローバルマネージャーglobal1がリージョナルマネージャーregional1とregional2から受け取るノードオブジェクトデータを制御するには、regional1とregional2の両方で転送フィルターを設定する必要があります。

転送されるノードを制限する転送フィルターの設定

この例では、Model 3500yl ProCurve Switchのノード情報のみをregional1からglobal1に転送できるノードグループを作成します。新しいノードグループを作成し、グループに制限を設定するには、以下の手順を実行します。

1. NNMiコンソールのregional1の [**設定**] ワークスペースから [**ノードグループ**] をクリックします。
2. [**新規作成**] をクリックします。

   注: この例では、ノードフィルターを新規作成し、そのフィルターを使用してregional1とregional2の転送フィルターを作成する方法を説明していますが、既存のフィルターを使用して、リージョナルNNMi管理サーバーからグローバルNNMi管理サーバーへの転送フィルターを設定することもできます。

   ヒント: 独自のデバイスもフィルターも含まれていないコンテナーノードグループを作成して、このノードグループを使用して子ノードグループを指定できます。この方法を使用すると、1つのコンテナーノードグループを使用して、ノードオブジェクトデータをグローバルNNMi管理サーバーに転送できます。

3. [**デバイスフィルター**] タブをクリックします。フィルター名にglobal1と入力し、[注] フィールドに作成するフィルターの説明を入力します。
4. [**新規作成**] アイコンをクリックして、[*ノード デバイスフィルター*] フォームを開きます。

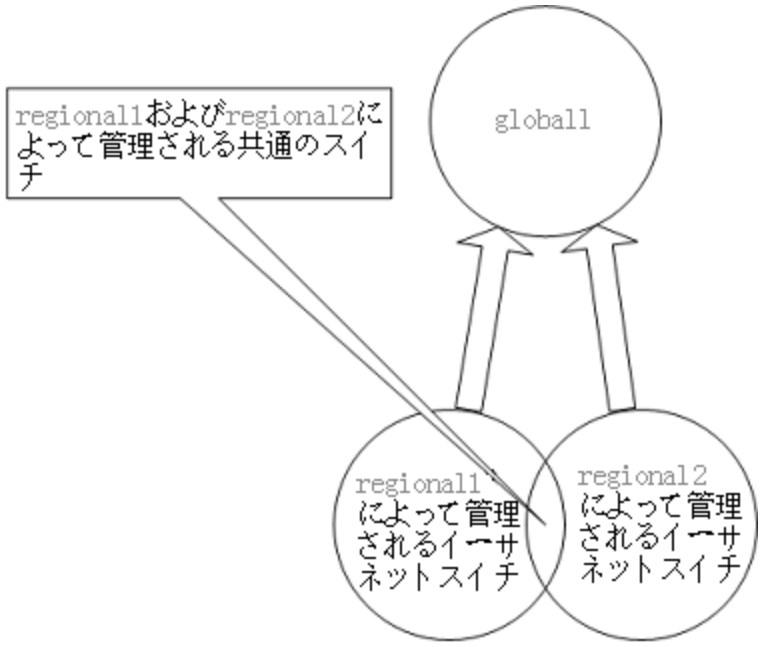
5. プルダウンメニューを使用して、[デバイスのカテゴリ] では [スイッチルーター]、[デバイスのベンダー] では [Hewlett-Packard]、[デバイスのファミリー] では [HP Procurve 3500 Fixed-port Switch] を選択します。
6. プルダウンメニューから [**クイック検索**] をクリックして、[デバイスのプロファイル] フォームを開きます。
7. 3500yl HP ProCurve Switchのプロファイルを検索して選択し、[**OK**] をクリックします。
8. 設定フォームごとに、[**保存して閉じる**] をクリックします。
9. このフィルターをテストするため、[**global1**] を選択します。
10. プルダウンメニューから、[**メンバーの表示**] をクリックします。
11. NNMiではすでにHP 3500ylスイッチが1つ検出されています。これは、作成したフィルターが、設定した特定のスイッチモデルを検索していることを示しています。次のステップでは、今作成したこのノードフィルターを使用して転送フィルターを設定します。
12. NNMiコンソールのregional1の [**設定**] ワークスペースから [**グローバルネットワーク管理**] をクリックします。
13. [**転送フィルター**] タブをクリックします。
14. [**クイック検索**] をクリックします。
15. **global1**フィルターを選択し、[**OK**] をクリックします。
16. [**保存して閉じる**] をクリックします。

これで、regional1の転送フィルターの設定作業は完了です。regional2についても手順1から手順16を実行し、「グローバルマネージャーとリージョナルマネージャーの接続」(466ページ)の説明に従って、global1をregional1とregional2に接続します。

グローバルマネージャーとリージョナルマネージャーの接続

この例では、regional1とregional2の両方で、共通のスイッチを複数管理します。

この共通のスイッチ情報をregional1かglobal1に転送するには、必要な接続を設定する必要があります。



そのためには、global1を先にregional1に接続してからregional2に接続する必要があります。この接続順により、global1はregional1をこれらの共通スイッチの監視を行うNNMi管理サーバーであるとみなします。Global1は、また、regional2から受け取るこれらの共通スイッチに関する情報を無視します。

注: この機能の動作を理解するには、まずは小さな規模で使用してから、それぞれのネットワーク管理ニーズに合わせて拡張することを推奨します。

global1を先にregional1に接続し、次にregional2に接続するには、以下の手順を実行します。

1. まず、NNMi管理サーバーのクロックをglobal1、regional1、およびregional2と同期してから、グローバルネットワーク管理設定内のこれらのサーバーを接続します。詳細については、NNMiヘルプの「クロック同期の問題」を参照してください。

   注: サーバークロック同期の問題など、リージョナルマネージャーとの接続に問題がある場合は、NNMiでは警告メッセージが表示されます。

2. global1からregional1への接続を設定します。
   a. global1のNNMiコンソールで、[**設定**] ワークスペースの [**グローバルネットワーク管理**] をクリックします。
   b. [**リージョナルマネージャ接続**] をクリックします。
   c. [**新規作成**] アイコンをクリックして、リージョナルマネージャーを新規作成します。
   d. regional1の名前と説明情報を追加します。
   e. [**接続**] タブをクリックします。
   f. [**新規作成**] アイコンをクリックします。

g. regional1の接続情報を追加します。

注: このフォームの実行に関する具体的な情報については、NNMiヘルプの [ヘルプ] > [リージョナルマネージャの接続フォームの使用法] を参照してください。

h. 各設定フォームで [**保存して閉じる**] をクリックし、変更を保存します。

3. global1からregional2への接続を確立するため、手順aから手順gまでを実行します。

global1からregional1とregional2への接続ステータスの判定

global1からregional1およびregional2への接続の状態を確認するには、以下の手順を実行します。

1. global1のNNMiコンソールで、[**設定**] ワークスペースの [**グローバルネットワーク管理**] をクリックします。
2. [**リージョナルマネージャ接続**] タブをクリックします。
3. regional1とregional2の接続ステータスを確認します。[接続されています] と表示され、正しく機能していることを意味します。

   詳細については、NNMiヘルプの「リージョナルマネージャーとの接続状態を確認する」を参照してください。

NNMiが検出を完了するまで、次のセクションには進まないでください。詳細については、『HP Network Node Manager i Softwareインタラクティブインストールガイド』の「検出の進行状況の確認」を参照してください。

global1インベントリの確認

NNMiが検出を完了するまで、このセクションは実行しないでください。詳細については、『HP Network Node Manager i Softwareインタラクティブインストールガイド』の「検出の進行状況の確認」を参照してください。

global1に転送されるノード情報regional1を表示するには、以下の手順を実行します。

1. [**インベントリ**] ワークスペースに配置されている [**管理サーバーのノード**] フォームに、global1のNNMiコンソールから移動します。
2. スイッチprocurve1.x.y.zに関する情報がregional1からglobal1に転送されたと仮定します。**regional1**を選択すると、インベントリは以下のように表示されます。

手順1から手順2を実行して、接続されているほかのリージョナルマネージャーからglobal1に渡されたデバイスインベントリも表示します。

global1とregional1との通信の切断

(一時的または完全に) グローバルマネージャー (global1など) をシャットダウンするには、グローバルマネージャーとリージョナルマネージャー間の通信を切断する必要があります。

この例では、global1では対regional1のサブスクリプションがまだアクティブであると想定します。

global1とregional1間の通信を切断するには、以下の手順を実行します。

1. global1のNNMiコンソールで、[**設定**] ワークスペースの [**グローバルネットワーク管理**] をクリックします。
2. [**リージョナルマネージャ接続**] をクリックします。
3. ステータスが [接続されています] であることを確認します。ステータスが [接続されています] ではない場合は、処理を続行する前に、NNMiヘルプの「グローバルネットワーク管理のトラブルシューティング」を参照して問題を診断します。
4. regional1を選択し、[**開く**] アイコンをクリックします。
5. [**接続**] をクリックして [**regional1.x.y.z**] を選択してから [**削除**] をクリックします。
6. [**保存して閉じる**] をクリックします。
7. [**リージョナルマネージャ接続**] タブでは、regional1の [**名前**] 属性に注意してください (大文字小文字は区別されます)。後のステップで、RemoteNNMiServerName変数にこのテキスト文字列が必要になります。
8. [**保存して閉じる**] をクリックします。
9. global1で、コマンドラインで以下のコマンドを入力します。

   nnmnodedelete.ovpl -rm regional1 -u NNMiadminUserName **-p** NNMiadminPassword
10. これらのコマンドにより、regional1から転送されたノードレコードをglobal1から削除します。コマンドでは、regional1からglobal1に転送されたノードに関連するインシデントも閉じます。詳細については、NNMiヘルプの「リージョナルマネージャーとの接続を解除する」を参照してください。
11. regional1の設定レコードを削除するには、以下を実行します。
    a. [**設定**] ワークスペースをクリックします。
    b. [**グローバルネットワーク管理**] フォームを選択します。
    c. [**リージョナルマネージャ接続**] タブを選択します。
    d. regional1を選択して [削除] アイコンをクリックします。
    e. [**保存して閉じる**] をクリックして削除を保存します。
12. regional2など、global1に接続されているほかのNNMi管理サーバーリージョナルについても手順1から手順11を実行します。

検出とデータの同期

ネットワーク管理者がネットワーク上のデバイスの追加、削除、または変更を行うと、regional1やregional2などのリージョナルサーバーはそうした変更を検出して、この章の例でのglobal1などのグローバルサーバーを更新します。regional1とregional2では、global1が管理するノードの管理モードに対して管理者が行う変更についてもglobal1に通知します。

注: 整合性を保つため、regional1とregional2はデバイスの状態の変化を検出すると、global1を継続的に更新するので、グローバルサーバーとリージョナルサーバーの両方でノードの状態が同じに保たれます。

regional1またはregional2が管理するノードに関する情報をglobal1が要求するたびに、regional1またはregional2は要求された情報をglobal1に返します。 global1からノードに直接要求することはありません。global1が検出を実行するとき、デバイスに対するSNMPクエリーは重複しません。

global1は、regional1またはregional2が検出を完了するたびに、regional1とregional2を同期します。NNMiはFDB (転送データベース) データを使用して、レイヤー2接続を計算します。FDBデータは非常にダイナミックなもので、特に、1つのグローバルサーバーに複数のリージョナルサーバーが接続しているような場合には、検出するごとに大きく異なります。

注: ユーザーが修正した属性やアプリケーションが修正した属性に対する変更は、グローバルサーバーでは同期中に更新されません。

[再検出間隔] は、各リージョナルサーバーで調整でき、global1とリージョナルマネージャーとの間の検出の精度を変更できます。[再検出間隔] が短くなるほど、検出の精度が上がり、NNMiが行うネットワークトラフィックも増えます。[再検出間隔] が長くなるほど、検出の精度は下がり、NNMiが行うネットワークトラフィックも減ります。これは、ネットワークが大きくなるほど、ユーザーが行う再検出の頻度が少なくなることを意味します。[再検出間隔] を設定するには、以下の手順を実行します。

1. regional1またはregional2のNNMiコンソールから、[**設定**] ワークスペースの [**検出の設定**] をクリックします。
2. リージョナルサーバーで検出を開始する頻度に従い、[再検出周期] を調整します。グローバルサーバーは、リージョナルサーバーが検出を完了するとすぐに検出を開始します。
3. [**保存して閉じる**] をクリックします。

リージョナルマネージャーからグローバルマネージャーへのカスタム属性の複製

NNMiでは、リージョナルマネージャーでカスタム属性を設定して、それらのカスタム属性をグローバルマネージャーに複製できます。たとえば、カスタム属性データをリージョナルマネージャーの

ノードに追加して、そのデータをグローバルマネージャーに複製した後で、そのデータを使用してそれらのノードのインシデントを強化できます。

注: NNMiでは、リージョナルマネージャーからグローバルマネージャーにノードおよびインタフェースのカスタム属性を複製できます。

NNMiコンソールで、グローバルマネージャーの [**カスタム属性の複製**] タブ ([**グローバルネットワーク管理**] 設定内) を使用してカスタム属性の複製を設定できます。

注: NNMiでは、ユーザーによる設定や入力を行わずに無番号インタフェースのカスタム属性が複製されます。詳細については、NNMiヘルプを参照してください。

また、`nnmgnmattrcfg.ovpl`コマンドラインインタフェースツールを使用して以下を実行できます。

- 複製に属性を追加する
- 複製から属性を除外する
- 一括操作用のファイルを使用する複製に属性を追加する
- 一括操作用のファイルを使用する複製から属性を除外する

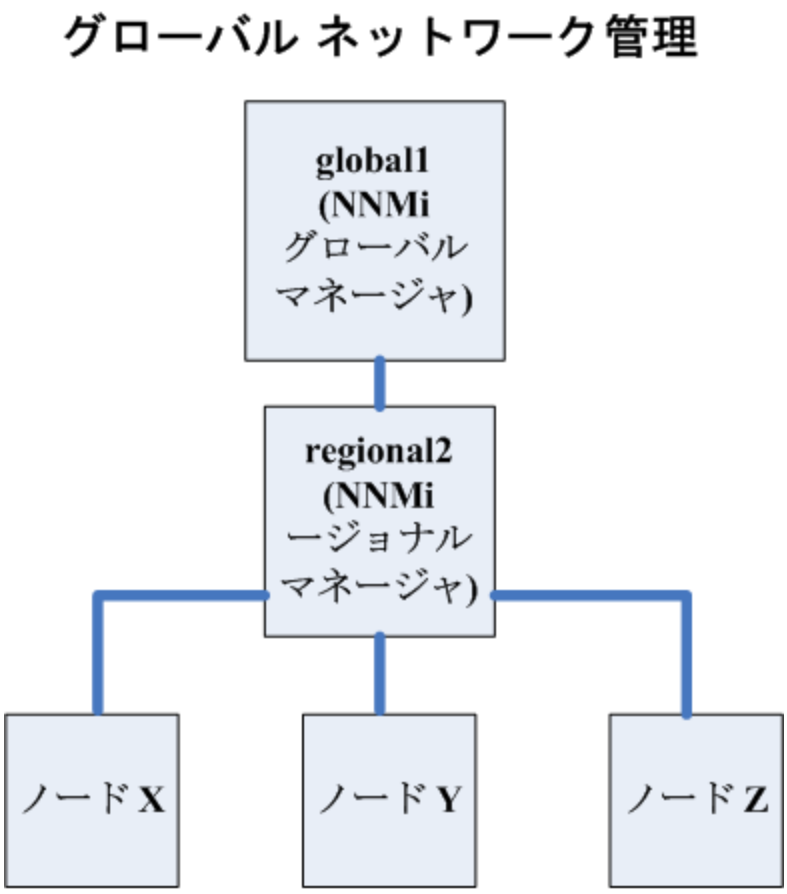
詳細については、`nnmgnmattrcfg.ovpl`のリファレンスページ、またはLinuxのマニュアルページを参照してください。

デバイスのステータスのポーリングまたは設定ポーリング

この例では、以下を前提としています (以下の図を参照)。

- リージョナルNNMi管理サーバー`regional2`は、`Node X`を検出および管理する
- グローバルNNMi管理サーバー`global1`は、リージョナルNNMi管理サーバー`regional2`と接続する

ノードのステータスのポーリングまたは設定ポーリング



global1からNode Xのステータスをポーリングするには、以下の手順を実行します。

1. global1から、[**インベントリ**] ワークスペースの [**ノード**] をクリックします。
2. ノードインベントリからNode Xを選択します。
3. **[アクション] > [ステータスのポーリング]** メニュー項目を使用して、Node Xのステータスのポーリングを要求します。
4. NNMi管理サーバーglobal1は、リージョナルNNMi管理サーバーregional2からのステータスのポーリングを要求し、結果を画面に表示します。ステータスのポーリング要求は、global1とregional2のどちらから発行しても問題はありません。ステータスのポーリングの結果は同じものが表示されます。

global1でNode Xの最新の検出情報を取得するようにするには、以下を実行してglobal1からNode Xの設定ポーリングを行います。

1. global1から、[**インベントリ**] ワークスペースの [**ノード**] をクリックします。
2. ノードインベントリからNode Xを選択します。
3. **[アクション] > [設定のポーリング]** メニュー項目を使用して、Node Xの設定ポーリングを要求します。

4. NNMi管理サーバーglobal1は、リージョナルNNMi管理サーバーregional2からの設定ポーリングを要求し、結果を画面に表示します。設定ポーリング要求は、global1とregional2のどちらから発行しても問題はありません。設定ポーリングの結果は同じものが表示されます。

グローバルマネージャーを使ったデバイスステータスの判定とNNMiインシデント生成

NNMi管理サーバー global1は、リージョナルマネージャーregional1とregional2からくるステータス変更をリッスンし、ローカルデータベースにあるステータスを更新します。

NNMi管理サーバー regional1とregional2のNNMi StatePollerサービスは、監視するデバイスの状態の値を計算します。global1は、regional1とregional2から状態の値の更新を受け取ります。global1は、自分が検出するノードにポーリングしますが、regional1とregional2によって管理されているノードにはポーリングしません。

regional1によって管理されているノードの管理モードを変更した後、global1上の管理モードも変更されます。ネットワーク管理者がregional1またはregional2によって管理されるネットワーク機器の追加、削除、変更を行うと、regional1またはregional2はそれらのネットワークデバイスの変更についてglobal1を更新します。

global1は、regional1とregional2によって転送されてきたノードオブジェクトデータなど、独自のCausal Engineとトポロジを使用してインシデントを生成します。これは、生成するインシデントが、トポロジに違いがある場合に、regional1とregional2のインシデントとは少し異なる場合があることを意味します。

フィルタリングがglobal1の接続性に影響する可能性があるため、転送フィルターをregional1やregional2に使用することは避けたほうがよいでしょう。ここで生じる差異が、global1と2つのリージョナル (regional1とregional2) との間の根本原因分析での差異になる可能性があります。ほとんどの場合、転送フィルターの使用しないことを選択すると、グローバルNNMi管理サーバーのトポロジは大きくなります。これは、より正確な根本原因分析の結果を得るのに役立ちます。

追加の設定をしないと、regional1はトラップをglobal1に転送しません。これを行うには、特定のトラップをglobal1に転送するようにregional1を設定する必要があります。HPでは、グローバルマネージャーに過剰な負荷がかからないように、リージョナルマネージャーは量の少ない、重要なトラップを転送するよう設定することをお勧めします。NNMiは、転送されたトラップがTrapStormインシデントを引き起こすような場合、転送されたトラップを削除します。NNMiコンソールでTrapStorm Management Eventの詳細を参照してください。

グローバルネットワーク管理でアプリケーションフェイルオーバーの設定を行う

グローバルマネージャーとリージョナルマネージャーの両方を、アプリケーションフェイルオーバーを使用するよう設定できます。グローバルマネージャーとリージョナルマネージャーは、アクティブなシステムを自動的に検出して接続します。

アプリケーションフェイルオーバーを認識するようglobal1を設定するには、以下の手順を実行します。

1. global1のNNMiコンソールで、[**設定**] ワークスペースの [**グローバルネットワーク管理**] をクリックします。

   この例では、以下を想定しています。

   - regional1がアプリケーションフェイルオーバー用に設定されている
   - regional1_backupがセカンダリサーバーとして設定されている

2. [**リージョナルマネージャ接続**] をクリックします。
3. regional1を選択し、[**開く**] アイコンをクリックします。
4. [**新規作成**] アイコンをクリックします。
5. [**ホスト名**]、[**HTTPポート**]、[**ユーザー名**]、および [**順序**] に値を入力します。順番の値には、regional1より大きな値を設定します。
6. 各設定フォームで [**保存して閉じる**] をクリックし、変更を保存します。

   リージョナルマネージャーが失敗すると、グローバルマネージャーは以下を実行します。

   a. プライマリに問い合わせます。
   b. プライマリからの応答がない場合、セカンダリに問い合わせます。

グローバルシステムでアクティブシステムが応答しないことを検出すると、順序の番号が最も小さいものから再接続を試みます。

グローバルネットワーク管理のトラブルシューティングのヒント

このセクションでは、以下のトラブルシューティングのトピックについて説明します。

ヒント: グローバルネットワーク管理のトラブルシューティング情報については、NNMiヘルプの「**グローバルネットワーク管理のトラブルシューティング**」も参照してください。

- 「クロック同期」(474ページ)
- 「グローバルネットワーク管理システム情報」(475ページ)
- 「グローバルマネージャーからのリージョナルマネージャー検出の同期」(475ページ)
- 「破損したglobal1上のデータベースの修復」(477ページ)

クロック同期

グローバルネットワーク管理 (グローバルマネージャーとリージョナルマネージャー) やシングルサインオン (SSO) に属するネットワーク環境内のすべてのNNMi管理サーバーは、それぞれの内部タイムク

ロックを世界標準時で同期する必要があります。たとえば、LinuxツールのNetwork Time Protocol Daemon (NTPD) や利用可能なWindowsオペレーティングシステムツールなどの時刻の同期プログラムを使用します。

NNMiコンソールの下部に次のメッセージが表示される場合の対応は、次のとおりです。

NNMi is not connected to 1 Regional Manager(s).See Help ?System Information, Global Network Management.

グローバルマネージャーの`nnm.0.0.log`ファイルに次のメッセージがないか確認します。

WARNING:Not connecting to system <serverName> due to clock difference of <number of seconds>.Remote time is <date/time>.

クロックが合わなくなり、再同期が必要です。グローバルマネージャーの`nnm.0.0.log`ファイルに次のメッセージがないか確認します。

WARNING:Not connecting to system <serverName> due to clock difference of <number of seconds>.Remote time is <date/time>.

この警告が表示されて数分以内に、NNMiはリージョナルマネージャ接続を切断します。また、NNMiコンソールの下部に次のメッセージが表示されます。

NNMi is not connected to 1 Regional Manager(s).See Help ?System Information, Global Network Management.

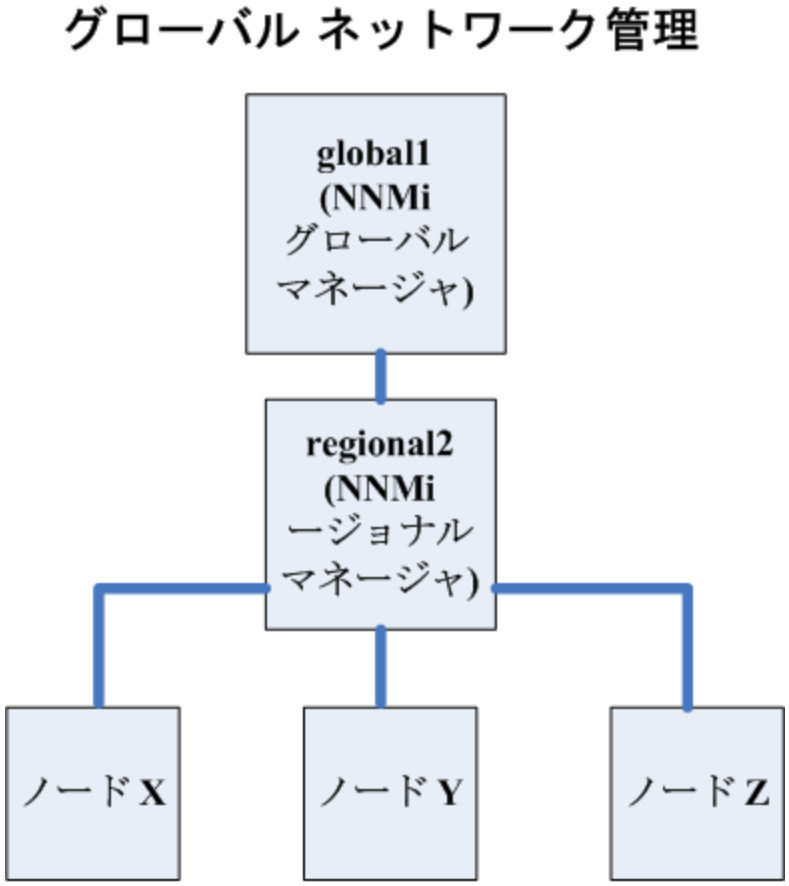
グローバルネットワーク管理システム情報

グローバルネットワーク管理接続に関する情報を表示するには、**[ヘルプ] > [システム情報]** を選択して **[グローバルネットワーク管理]** タブをクリックします。

グローバルマネージャーからのリージョナルマネージャー検出の同期

`global1`と`regional2`の間で情報に矛盾があることに気がついた場合は、`global1`から**`nnmnoderediscover.ovpl`**スクリプトを実行して、`global1`と`regional2`を同期します。実行の結果、`regional2`は新しい検出結果を使用して`global1`を更新します。

この例では、以下の図に示すネットワークを使用します。



以下のコマンドを実行してノードX、Y、およびZとglobal1を同期します。

```
nnmnoderediscover.ovpl -u username -p password -rm regional2
```

ヒント: nnmnoderediscover.ovplコマンドで-fullsyncフラグを使用して、ポーリングされるオブジェクトのすべての状態とステータスを同期することができます (ただし、この処理には時間がかかり、システム負荷が増加する可能性があります)。詳細については、nnmnoderediscover.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

- NNMiでは、手動による再同期の後にトポロジ、状態、およびステータスが自動的に再同期されます。
- 再同期中にNNMiを停止しないでください。再同期を確実に完了するには、手動による再同期の後でNNMiを数時間実行し続けます。実際の所要時間は、ノード数、状態変化の量、および再同期中に受信されたトラップデータによって異なります。
- 再同期が完了する前にNNMiを停止する必要がある場合は、再同期をもう一度実行して完了する必要があります。
- 管理サーバー全体の再同期を手動で実行するには、nnmnoderediscover.ovpl –all –fullsyncを実行します。

破損したglobal1上のデータベースの修復

global1のサービスを停止し、データベースを復元する必要がある場合、いくつかの方法があります。

1. global1のデータベースを正しく復元すると、regional1とregional2はglobal1を使用してキャッシュされた情報を同期します。global1をオンラインに戻した後、手動で行う手順はありません。
2. global1のサービスが長時間停止すると、手順1は正常に機能しないことがあります。これを解消するには、global1で**nnmnoderediscover.ovpl**スクリプトを実行してglobal1、regional1およびregional2で新たな検出を開始します。この場合、さらに迅速に更新されたステータス情報を入手するため、キーデバイスに対してステータスのポーリングを実行できます。
3. global1のデータベースを復元できない場合、**nnmsubscription.ovpl**スクリプトを使用して古いglobal1データをregional1とregional2のデータベースから消去するには、サポートに問い合わせる必要があります。

グローバルネットワーク管理とNNM iSPIsまたは第三者の統合

NNM iSPIまたは第三者の統合は、導入にあたりそれぞれ独自のガイドラインがあります。この章の例では、複数のNNM iSPIsをregional1のみ、global1のみ、またはregiona1とglobal1の両方に配備できます。その他のNNM iSPIsまたは第三者の統合については、regional1とglobal1の両方にインストールされている必要があります。詳細については、NNM iSPIまたは第三者の統合に関するドキュメントを参照してください。

HP Network Node Manager iSPI Performance for Metrics Software

NNMiがグローバルネットワーク管理環境で配備されている場合は、以下を実行する必要があります。

1. NNMi管理サーバーごとにNetwork Performance Server (NPS) の1つのインスタンスを配備します。すべてのリージョナルマネージャーおよびグローバルマネージャーには、NPSの別個のインスタンスがインストールされ、配備されている必要があります。
2. すべてのリージョナルマネージャーおよびグローバルマネージャーで、イネーブルメントスクリプトを1回実行します。

グローバルネットワーク管理とアドレス変換プロトコル

動的ネットワークアドレス変換 (NAT)、動的ポートアドレス変換 (PAT)、または動的ネットワークアドレスおよびポート変換 (NAPT) の各グループには、NNMiグローバルネットワーク管理設定全体で一意のテナントに加え、NNMiリージョナルマネージャーが必要です。「NAT環境の重複IPアドレスの管理」(407ページ)を参照してください。NNMiヘルプも参照してください。

IPv6用NNMi Advancedの設定

IPv6管理機能を使用するには、NNMi Advanced、NNMi PremiumまたはNNMi Ultimateライセンスを購入してインストールする必要があります。

NNMiのIPv6管理により、インタフェース、ノード、サブネットも含めたIPv6アドレスの検出と監視が可能になります。シームレスな統合を提供するため、NNMiはIPv4とIPv6両方のアドレスを含めるようIPアドレスモデルを拡張します。NNMiでは、可能な限りすべてのIPアドレスが等しく扱われます。IPv4アドレスに関連するほとんどの機能はIPv6アドレスについても使用可能です。ただし、いくつか例外があります。NNMiコンソールに表示されるIPv6情報の詳細については、NNMiヘルプを参照してください。

この章には、以下のトピックがあります。

機能説明

NNMi IPv6管理機能には、以下の機能があります。

- IPv6専用デバイスおよびデュアルスタックデバイスのIPv6インベントリ検出
  - IPv6アドレス
  - IPv6サブネット
  - IPv6アドレス、サブネット、インタフェース、およびノード間の関連付け
- 以下のためのネイティブIPv6 SNMP通信

  - ノードの検出
  - インタフェースの監視
  - トラップと通知の受信と転送
- デュアルスタックデバイスでのIPv4またはIPv6通信 (管理アドレス) の自動選択。NNMiコンソールを使用し、[**設定**] ワークスペースにある [**通信の設定**] で、SNMP管理アドレス設定をIPv4またはIPv6に設定します。
- IPv6アドレスフォルト 監視のためのネイティブICMPv6通信
- IPv6アドレスまたはホスト名を使用したシード済みデバイスの検出
- IPv6レイヤー3隣接検出ヒントを使用したIPv6デバイスの自動検出
- LLDP (Link Layer Discovery Protocol) IPv6隣接情報を使用するレイヤー2隣接検出ヒントを使用したIPv6デバイスの自動検出
- IPv4、IPv6情報の統合表示
  - ノード、インタフェース、アドレス、サブネット、および関連付けのインベントリビュー
  - IPv4デバイスとIPv6デバイス用のレイヤー2隣接ビューおよびトポロジマップ
  - IPv4デバイスとIPv6デバイス用のレイヤー3隣接ビューおよびトポロジマップ
  - インシデント、結果、根本原因分析
- NNMiコンソールアクション: IPv6アドレスとノードに対するpingとtraceroute
- IPv6アドレスとアドレス範囲を使用したNNMi設定
  - 通信設定
  - 検出の設定
  - モニタリングの設定
  - ノードとインタフェースグループ
  - インシデントの設定
- IPv6インベントリとインシデント用のSDK Webサービスサポート
- IPv6インタフェースに対するNNM iSPI Performance for Metricsのサポート

NNMi IPv6管理機能には、以下は含まれません。

- IPv6サブネット接続の検出
- 検出のためのIPv6 pingスィープの使用
- IPv6 ネットワーク パス ビュー (Smart Path)
- IPv6リンクローカルアドレス障害監視
- 検出シードとしてのIPv6リンクローカルアドレスの使用

必要条件

管理サーバーの仕様およびNNMiのインストールの詳細については、『NNMiデプロイメントリファレンス』、『NNMiリリースノート』、および『NNMi対応マトリックス』を参照してください。

ネイティブIPv6通信を使用するには、NNMi管理サーバーはデュアルスタックシステムであることが必要です。つまり、IPv4とIPv6両方を使用して通信するということです。

注: HP NNMiでIPv6検出を設定していて、HP Universal CMDB (HP UCMDB) 統合を使用している場合、UCMDB HP Discovery and Dependency mapping (DDM、検出および依存関係マッピング) インポートタスクは失敗します。HP NNMiでHP UCMDB統合を使用するには、IPv6検出を無効にする必要があります。

IPv6の追加要件は以下のとおりです。

- 少なくとも1つのネットワークインタフェースでIPv4を有効化し設定する必要があります。
- IPv6を有効化し、管理する必要のあるIPv6ネットワークに接続する少なくとも1つのネットワークインタフェースで、グローバルユニキャストアドレスまたは一意のローカルユニキャストアドレスを持つ必要があります。
- NNMi管理サーバーにIPv6ルートを設定し、IPv6を使用してNNMiで検出と監視を行うデバイスとNNMiが通信できるようにする必要があります。

注: IPv4専用のNNMi管理サーバーを使用することもできますが、IPv4/IPv6デュアルスタックデバイスをNNMiで完全に管理することはできなくなります。たとえば、IPv4専用管理サーバーを使用すると、NNMiはIPv6専用デバイスの検出、IPv6シードとヒントを使用した検出、およびIPv6アドレスを持つデバイス上での障害の監視はできません。

NNMi管理サーバーで使用されるDNSサーバーは、DSNからIPv6アドレスへのホスト名とIPv6アドレスからDSNへのホスト名を解決する必要があります。たとえば、AAAA DNSレコードからのホスト名とAAAA DNSへのホスト名を解決する必要があります。つまり、DNSサーバーはホスト名を128ビットIPv6アドレスにマッピングする必要があります。IPv6対応DNSサーバーが使用できない場合でも、NNMiは正しく機能しますが、NNMiではIPv6アドレスを使用するノードのDNSホスト名の判定や表示は行いません。

ライセンス

IPv6管理機能を使用するには、NNMi Advanced、NNMi PremiumまたはNNMi Ultimateライセンスを購入してインストールする必要があります。NNMiライセンスの取得とインストールの詳細については、「NNMiのライセンス」(312ページ) を参照してください。

NNMi製品には、インスタントオンライセンス用パスワードが含まれています。これは一時的なものですが、有効なNNMi Advancedライセンスです。できるだけ早く、永久ライセンスキーを入手してインストールしてください。

サポートされる設定

NNMiをサポートするオペレーティングシステム構成の詳細については、『NNMi対応マトリックス』を参照してください。

管理サーバー

以下の表に、IPv4専用およびデュアルスタック両方のNNMi管理サーバーの機能を示します。

管理サーバーの機能

機能	IPv4専用	デュアルスタック
IPv4通信 (SNMP、ICMP)	対応	対応
IPv6通信 (SNMP、ICMPv6)	非対応	対応
デュアルスタック管理ノード	対応	対応
IPv4シードを使用した検出	対応	対応
IPv6シードを使用した検出	非対応	対応
IPv4アドレスおよびサブネットインベントリ	対応	対応
IPv6アドレスおよびサブネットインベントリ	対応	対応
SNMPを使用したインタフェースステータスとパフォーマンス	対応	対応
ICMPを使用したIPv4アドレスステータス	対応	対応
ICMPv6を使用したIPv6アドレスステータス	非対応	対応
IPv6専用管理ノード	非対応	対応
IPv6シードを使用した検出	非対応	対応
IPv6アドレスおよびサブネットインベントリ	非対応	対応
SNMPを使用したインタフェースステータスとパフォーマンス	非対応	対応
ICMPv6を使用したIPv6アドレスステータス	非対応	対応

管理サーバーの機能 (続き)

機能	IPv4専用	デュアルスタック
IPv4専用管理ノード	対応	対応
IPv4シードを使用したノード検出	対応	対応
IPv4シードを使用したノード検出	対応	対応
SNMPを使用したインタフェースステータスとパフォーマンス	対応	対応
SNMPを使用したインタフェースステータスとパフォーマンス	対応	対応
IPv4アドレスおよびサブネットインベントリ	対応	対応

IPv6をサポートするSNMP MIB

NNMiでは、IPv6用の以下のSNMP MIBがサポートされています。

- RFC 4293 (現在のIETF標準)
- RFC 2465 (元のIETF提案)
- Cisco IP-MIB

NNMiのインストール

NNMiのインストール中に、インストールスクリプトがIPv6機能をアクティブにします。 ただし、必要に応じて`nms-jboss.properties`ファイルを編集し、これらのIPv6機能を手動で非アクティブにできます。

非アクティブにされた後で、IPv6機能を再度アクティブにできます。詳細については、「IPv6機能の非アクティブ化」(482ページ)および「IPv6機能の再アクティブ化」(484ページ)を参照してください。

IPv6機能の非アクティブ化

以下の手順を実行して、管理上IPv6機能を無効化することができます。

1. `nms-jboss.properties`ファイルを開きます。以下の場所を探してください。

   Windowsの場合: `%NNM_PROPS%\nms-jboss.properties`

Linuxの場合: `$NNM_PROPS/nms-jboss.properties`

注: NNMiでは、各プロパティの完全な記述を用意しており、`nms-jboss.properties`ファイルのコメントとして示しています。

2. NNMiのIPv6通信を非アクティブ化するには、以下の手順を実行します。
   a. # Enable Java IPv6 Communicationで始まるテキストを探します。
   b. 以下の行を見つけます。
   c. java.net.preferIPv4Stack=false
   d. この行を以下のように編集します。

      java.net.preferIPv4Stack=true

      行がコメント化されていないことを確認します。
3. NNMiでIPv6管理全体を非アクティブ化するには、以下の手順を実行します。
   a. # Enable NNMi IPv6 Managementで始まるテキストを探します。
   b. 以下の行を見つけます。

      com.hp.nnm.enableIPv6Mgmt=true
   c. この行を以下のように編集します。

      com.hp.nnm.enableIPv6Mgmt=false

      行がコメント化されていないことを確認します。
   d. `nms-jboss.properties`ファイルを保存して閉じます。
4. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで`ovstop`コマンドを実行します。
   b. NNMi管理サーバーで`ovstart`コマンドを実行します。

      注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてから`ovstop`コマンドおよび`ovstart`コマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。
5. 以下のコマンドを使用して、NNMiプロセスを確認します。

   ovstatus -v ovjboss

NNMiライセンスの変更の詳細については、「ライセンス」(480ページ)を参照してください。

.

非アクティブ化後のIPv6監視

IPv6管理またはIPv6通信が完全に無効になると、StatePollerサービスはICMPv6によるIPv6アドレスの監視をすぐに停止します。NNMiは、これらのアドレスのIPアドレス状態を [未ポーリング] に設定します。アドレスを選択し、このアドレスに対して [**アクション**] > [**モニタリングの設定**] を使用すると、関連する [監視設定] ルールで [IPアドレスの障害のポーリング] が有効になっている場合でも、NNMiは「障害ICMPポーリングの有効化： false」と表示します。

非アクティブ化後のIPv6インベントリ

一度NNMiが完全にIPv6インベントリを検出すると、以下の場合には、NNMiにそのインベントリを自動的に消去させることができます。

- マスターIPv6スイッチをオンにした後で、オフにしてNNMiを再起動した。

  NNMiはIPv6インベントリをすぐに削除しません。NNMiはSNMPノード のIPv6インベントリを次の検出サイクルで削除します。NNMiは 非SNMP IPv6ノードを削除しません。IPv6ノードは、NNMiインベントリから手動で削除する必要があります。
- NNMi Advancedのみ。NNMi Advancedライセンスが期限切れ、または誰かがライセンスを削除した。NNMiは、NNMiの基本ライセンスを使用します。基本ライセンスは、検出されたノードすべての管理を続行するのに十分な機能があります。

  NNMiは 非SNMP IPv6ノードすべてをインベントリからすぐに削除します。NNMiはSNMPノードをすべて再検出し、IPv6データはすべて削除します。
- NNMi Advancedのみ。NNMi Advancedライセンスが期限切れ、または誰かがライセンスを削除した。NNMiは、NNMi基本ライセンスを使用します。基本ライセンスは、検出したノードすべての管理を続行するのに十分な機能はありません。NNMiはすぐに、非SNMP IPv6ノードを削除します。

IPv6インベントリクリーンアップ時の既知の問題点

以下の状況で、IPv6インベントリが残る場合があります。

NNMiがSNMPを使用して、あるIPv6ノードを正常に管理し、次の検出の前にそのノードにアクセスできなくなったような場合です。

既存の検出システムの設計上、検出プロセスはSNMPを使用した通信ができなくなったノードを更新できません。このようにして残ったノードを削除するには、通信の問題を解決してから、NNMiコンソールの [**アクション**] > [**設定のポーリング**] コマンドを使用してそれらのノードの設定情報を取得する必要があります。ネイティブIPv6ノードの場合、NNMiコンソールから直接ノードを削除します。

IPv6機能の再アクティブ化

注: IPv6専用デバイスの検出やIPv6アドレスステータスの監視など、IPv6通信を必要とする機能では、NNMi管理サーバーにIPv6グローバルユニキャストアドレスが設定され機能することが必要で

す。

以下に示す手順で、非アクティブにされた後でIPv6機能を再度アクティブにする方法を説明します。

1. nms-jboss.propertiesファイルを編集します。以下の場所を探してください。

   Windowsの場合:%NNM_PROPS%\nms-jboss.properties

   Linuxの場合: $NNM_PROPS/nms-jboss.properties

   注: NNMiでは、各プロパティの完全な記述を用意しており、nms-jboss.propertiesファイルのコメントとして示しています。

2. # Enable NNMi IPv6 Managementで始まるテキストを探します。
3. NNMiでIPv6通信を有効化するには、以下のプロパティをコメント解除します。

   java.net.preferIPv4Stack=false

   注:プロパティをコメント解除するには、行の先頭から#!文字を削除します。

4. # Enable NNMi IPv6 Managementで始まるテキストを探します。
5. NNMiでIPv6通信全体を有効化するには、以下のプロパティをコメント解除します。

   com.hp.nnm.enableIPv6Mgmt=true

6. nms-jboss.propertiesファイルを保存して閉じます。
7. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

   注: 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

8. 以下のコマンドを使用して、NNMiプロセスを確認します。

   ovstatus -v ovjboss

   起動に成功すると、以下のように表示されます。

   object manager name: ovjboss

   状態　　実行中

   PID:　　<Process ID #>

   最後のメッセージ　　初期化が終了しました。

   exit status:　　-

   additional info:

SERVICE STATUS

CommunicationModelService サービスが起動されました

CommunicationParametersStatsService サービスが起動されました

CustomPoller サービスが起動されました

IslandSpotterService サービスが起動されました

ManagedNodeLicenseManager サービスが起動されました

MonitoringSettingsService サービスが起動されました

NamedPoll サービスが起動されました

msApa サービスが起動されました

NmsCustomCorrelation サービスが起動されました

NmsDisco サービスが起動されました

NmsEvents サービスが起動されました

NmsEventsConfiguration サービスが起動されました

NmsExtensionNotificationService サービスが起動されました

NnmTrapService サービスが起動されました

PerformanceSpiAdapterTopologyChangeService サービスが起動されました

PerformanceSpiConsumptionManager サービスが起動されました

RbaManager サービスが起動されました

RediscoverQueue サービスが起動されました

SpmdjbossStart サービスが起動されました

StagedIcmp サービスが起動されました

StagedSnmp サービスが起動されました

StatePoller サービスが起動されました

TrapConfigurationService サービスが起動されました

TrustManager サービスが起動されました

9. IPv6を再度アクティブにすると、NNMiビューには、新たに検出されたノードのIPv6インベントリが表示されます。次の検出サイクルの間に、NNMiビューにはその前の検出ノードに関連するIPv6インベントリが表示されます。
10. 必要に応じて、デュアルスタック管理ノードのSNMP管理アドレス設定を指定します。デュアルスタック管理ノードは、IPv4またはIPv6いずれかを使用して通信できるノードです。これを行うには、以下の手順を実行します。
    a. NNMiコンソールで、[**設定**] ワークスペースにある [**通信の設定**] をクリックします。
    b. [**管理アドレスの選択**] セクションを見つけます。[IPバージョン設定] フィールドで、[IPv4]、[IPv6]、または [いずれか] を選択します。

c. 変更を保存します。

d. NNMi管理サーバーを再起動します。

NNMi管理サーバーで**ovstop**コマンドを実行します。

NNMi管理サーバーで**ovstart**コマンドを実行します。

> **注:** 高可用性 (HA) でファイルの変更を行うとき、クラスターの両方のノードに変更を加える必要があります。変更によってNNMi管理サーバーを停止して再起動する必要がある場合、ノードをメンテナンスモードにしてからovstopコマンドおよびovstartコマンドを実行する必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

スピードアップを図るには、デュアルスタックノードとわかっているノードを選択し、NNMiコンソールで [**アクション**] > [**設定のポーリング**] コマンドを使用します。nnmnoderediscover.ovplスクリプトを使用して、NNMi検出キューにノードを追加することもできます。詳細については、nnmnoderediscover.ovplのリファレンスページ、またはLinuxのマンページを参照してください。

NNMi管理サーバーでIPv6通信を有効化すると、NNMiはICMPv6を使用してIPv6アドレスフォルトがないかノードの監視を開始します。

第7章: NNMiセキュリティ

この章には、以下のトピックがあります。

WebアクセスおよびRMI通信にSSL通信を設定する

NNMiには、WebアクセスおよびJava Remote Method Invocation (RMI) 通信でSecure Sockets Layer (SSL) を設定するのに使用される一連のデフォルト暗号が含まれています。暗号はnms-jboss.propertiesファイルにリストされています。

注意: HPの承認なしに暗号リストから暗号を追加または削除しないでください。これを行うと、製品に障害が発生したり、製品が動作しなくなる可能性があります。

非root Linuxユーザーへの NNMiの開始と停止の許可

注: /opt/OVディレクトリがnosuidオプションセットを含むパーティション上にある場合は、非ルートユーザー機能を利用できません。パーティションがnosuidオプションセットを使用して設定されているかどうかを判別するには、/etc/fstabを参照してください。

NNMiには、非root Linuxユーザーに NNMiの開始と停止を許可する方法があります。以下の手順を実行します。

1. ルートとして、以下のファイルを編集します。

   $NnmDataDir/shared/nnm/conf/ovstart.allow

2. NNMiの開始と停止を許可する非ルートユーザーを含めます (1行に1ユーザー)。
3. 変更を保存します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

組み込みデータベースツールのパスワードの入力

NNMiで組み込みデータベースツール (psqlなど) を実行するには、パスワードを入力する必要があります。NNMiによってデフォルトのパフワードが設定されており、ユーザーはnnmchangeembdbpw.ovplスクリプトを使用してこのパスワードを変更する必要があります。

注: nnmchangeembdbpw.ovplスクリプトを実行するには、Windowsシステムの場合は管理者、Linuxシステムの場合はrootとしてログインする必要があります。詳細については、nnmchangeembdbpw.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。

NNMiを高可用性 (HA) 環境で設定している場合、プライマリクラスターノードのみでnnmchangeembdbpw.ovplスクリプトを実行します。

プライマリクラスターノードのみで、以下の手順を実行します。

1. プライマリクラスターノードをメンテナンスモードに切り替えます。

   ノードのメンテナンスモードへの切り替えの詳細については、「メンテナンスモード」(202ページ) を参照してください。
2. 以下のコマンドでNNMiの全プロセスを停止します。

   Windowsの場合: `%NNM_BIN%\ovstop -c`

   Linuxの場合: `$NNM_BIN/ovstop -c`
3. nnmsdbmgrを再起動します。

   Windowsの場合: `%NNM_BIN%\ovstart nnmsdbmgr`

   Linuxの場合: `$NNM_BIN/ovstart nnmsdbmgr`
4. 組み込みデータベースパスワードを変更するには、nnmchangeembdbpw.ovpl スクリプトを実行します。

   Windowsの場合: `%NNM_BIN%\nnmchangeembdbpw.ovpl`

   Linuxの場合: `$NNM_BIN/nnmchangeembdbpw.ovpl`
5. セカンダリクラスターノードにコピーできるように変更を複製ディレクトリにコピーするには、nnmdatareplication.ovpl スクリプトを実行します。

Windowsの場合: Windowsの場合: %NNM_DATA%\misc\nnm\ha\nnmdatareplication.ovpl NNM

Linuxの場合: Linuxの場合: $NNM_DATA/misc/nnm/ha/nnmdatareplication.ovpl NNM

6. すべてのNNMiプロセスを再起動します。

   Windowsの場合: `%NNM_BIN%\ovstart`

   Linuxの場合: `$NNM_BIN/ovstart`

7. プライマリクラスターノードをメンテナンスモードから戻します。
8. セカンダリクラスターノードにフェイルオーバーします。

   注: Postgresパスワードが複製されるように、セカンダリクラスターノードをメンテナンスモードに切り替えてはいけません。

NNMiリソースグループがこのノードで開始されると、アプリケーションによって自動的にセカンダリクラスターノードにパスワードがコピーされます。

NNMiでSSLv3暗号化を有効化または無効化する設定

暗号化のNNMiリストは変更できます。ただし、このセクションで説明するプロパティファイルを別のディレクトリにコピーして、元の情報を必ず保存しておいてください。デフォルトでは、NNMiはSSLv3暗号化を無効化します。Webブラウザー通信の問題を解決するために、SSLv3暗号化を有効化する必要が生じることがあります。たとえば、以下のいずれかのエラーに似た接続エラーを受け取ることがあります。

- 安全な接続に失敗しました
- このページは表示できません

NNMi管理サーバー上に存在するNNM iSPIソフトウェアも使用していて、NNMiでSSLv3暗号化を有効化する場合には、各iSPIでもSSLv3を有効化する必要があります。SSLv3の有効化と無効化についての詳細は、対応する各NNM iSPIの『デプロイメントリファレンス』を参照してください。

高可用性 (HA) 下でファイル変更を行う場合、更新する必要があるserver.propertiesファイルの場所は、<Shared_Disk>`/NNM/dataDir/nmsas/NNM/server.properties`です。

SSLv3暗号化を有効化するようにNNMiを設定するには、次の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`

2. 以下の行を編集し、

   com.hp.ov.nms.ssl.PROTOCOLS = SSLv2Hello,TLSv1,TLSv1.1,TLSv1.2

   SSLv3を含めます。次に例を示します。

com.hp.ov.nms.ssl.PROTOCOLS = SSLv2Hello,TLSv1,TLSv1.1,TLSv1.2,SSLv3

注: この行に含まれるプロトコルはどれも削除できます。

3. ファイルを保存します。

注: 1つ以上のiSPIについてもSSLv3を有効化する場合は、次の手順で説明しているように、NNMi管理サーバーの停止と起動を行う前にそれらを変更してください。

4. NNMi管理サーバーを停止します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

SSLv3暗号化を有効にした後でSSLv3暗号化を無効にするには、次の手順を実行します。

1. 以下のファイルを開きます。

   Windowsの場合: `%NnmDataDir%\nmsas\NNM\server.properties`

   Linuxの場合: `$NnmDataDir/nmsas/NNM/server.properties`

2. 以下の行を編集し、

   com.hp.ov.nms.ssl.PROTOCOLS = SSLv2Hello,TLSv1,TLSv1.1,TLSv1.2,SSLv3

   SSLv3を削除します。次に例を示します。

   com.hp.ov.nms.ssl.PROTOCOLS = SSLv2Hello,TLSv1,TLSv1.1,TLSv1.2

3. ファイルを保存します。

注: SSLv3を有効化した後で1つ以上のiSPIでSSLv3を無効化する場合は、次の手順で説明しているように、NNMi管理サーバーの停止と起動を行う前にそれらを変更してください。

4. NNMi管理サーバーを停止します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

NNMi暗号化の設定

NNMiが使用する暗号化についての情報は、『NNMi強化ガイド』の「NNMi Webサーバーで使用される暗号の設定」を参照してください。

NNMiデータの暗号化

NNMiでは製品のさまざまなエリアにデータ暗号化が組み込まれています。例:

- アプリケーションフェイルオーバーは、クラスターノード間で送信されるメッセージを暗号化します。
- NNMiは、ユーザーアカウント用のパスワードを暗号化された形式でNNMiデータベースに保存します。
- グローバルネットワーク管理 (GNM) は、リージョナルマネージャーとグローバルマネージャー間で送信されるメッセージを暗号化します。

NNMiは、複数のNNMiコンポーネントに及ぶデータ暗号化の方法を使用します。NNMiデータ暗号化では、以下の暗号化タイプをサポートしています。

- 対称暗号化 — 両者が同じプライベートキーを共有します
- 非対称 — 公開キーとプライベートキーを使用した暗号化で、両者が他方の公開キーを持ちますが、自分自身のプライベートキーは保持します
- MessageDigest (ハッシュ) — 一方通行の暗号化 (復号できません) で、任意の長さの文字列が固定長の文字列に短縮されます

暗号化設定ファイル

NNMi暗号化フレームワークには、組織の暗号化設定を構成するために編集可能な一連のファイルが含まれています。ファイルは以下のフォルダーにあります。

- Windowsの場合: `%NnmDataDir%\shared\nnm\conf\crypto`
- Linuxの場合: `$NnmDataDir/shared/nnm/conf/crypto`

注意: 暗号設定ファイルは高度なユーザー向けです。暗号設定ファイルを編集する場合は細心の注意を払ってください。これらのファイルを不適切に編集すると、重大な問題が発生します。たとえば、アプリケーションフェイルオーバー用の暗号化パラメーターを変更すると、アプリケーションフェイルオーバーが機能しなくなります。同様に、システムおよびデータベースのパスワード暗号化設定を変更すると、NNMiが起動しなくなります。異なるNNMiサブシステムの暗号設定を変更する場合は、以降のセクションを参照してその手順に従ってください。

暗号設定ファイルのテキストブロック

暗号設定ファイルには、以下のテキストブロックが含まれています。

<allowed>

<allowed>ブロックは、プロバイダーのタイプ、アルゴリズム、および暗号設定ファイルを他の場所で使用できる最小のキーの長さを定義します。

注: 許可されていないアルゴリズムやキーの長さを使用しようとすると、NNMiは暗号化エラーを生成します。

ヒント: プロバイダーとは、暗号アルゴリズムの実装を可能にするベンダー (エンティティ) です。

暗号設定ファイルにリストされているアルゴリズムは、それらのファイルにリストされているプロバイダーに関連付けられています。

<default>

<default>ブロックは、サポートされているすべてのコンポーネントで使用されるデフォルト設定をリストしています。たとえば、<default>ブロックは1つの対称アルゴリズム、1つの非対称アルゴリズム、1つのダイジェストをリストしています。任意のコンポーネント用に定義されたコンポーネントブロックがある場合、そのコンポーネントはそのコンポーネントブロックで指定されたアルゴリズムを使用します (つまり、コンポーネントブロックの定義は<default>ブロックを上書きします)。そうでない場合、コンポーネントはそのコンポーネントによって使用される特定の暗号化のタイプ用のデフォルトのアルゴリズムを (<default>ブロックから) 要求します。

各コンポーネントは1つの暗号化のタイプのみを使用します (対称、非対称、またはダイジェスト)。たとえば、アプリケーションフェイルオーバーは対称暗号化のみを使用するため、アプリケーションフェイルオーバーのコンポーネントブロックで非対称またはダイジェストアルゴリズムを指定しても効果はなく、指定する必要はありません。

注: デフォルトブロックまたはコンポーネントブロックにリストされているキーのサイズは、少なくとも<allowed>ブロックにリストされているサイズにする必要があります (ただし、必要に応じてこのサイズより大きくすることはできます)。たとえば、<allowed>ブロックにAES-128が含まれている場合、AES-192も有効です。ただし、<allowed>ブロックがAES-192を指定する場合、AES-128は無効です。

暗号化およびアプリケーションフェイルオーバー

アプリケーションフェイルオーバーの暗号化設定を変更する (たとえば、暗号化アルゴリズムやキーの長さを変更する) には、以下の手順を実行します。

1. 両方のノードでovstopコマンドを実行して、NNMiおよびnnmclusterプロセスを停止します。アプリケーションフェイルオーバー用に設定されたNNMi管理サーバーでovstopコマンドを使用すると、NNMiは自動的に以下のコマンドを実行します。

   nnmcluster -disable -shutdown

2. 必要に応じてnnmcluster-crypto-config.xmlファイルを編集します。

   注: アプリケーションフェイルオーバーは対称暗号化のみを使用するため、非対称またはダ

イジェストを追加しても効果はなく、対称暗号化を削除すると障害が発生します。

3. nnmcluster-crypto-config.xmlファイルへの変更を保存します。
4. 古いキーファイルを削除します。

   ヒント: ファイルの場所はnnmcluster-crypto-config.xmlファイルで定義されています。

5. 新しいキーファイルを生成するには、以下のコマンドを実行します。

   nnmcluster -genkey

6. 編集したnnmcluster-crypto-config.xmlファイルと新しいキーファイルをクラスター内の他のノード (同じフォルダー内) にコピーします。

   これで、暗号化アルゴリズムおよびキーを定義するnnmcluster-crypto-config.xmlファイルが両方のノードで同じになります。また、キー自体も両方のノードで同じになります。

7. アクティブノードとスタンバイノードでnnmclusterを実行し、クラスターを再び開始します。

   アクティブノードでnnmcluster -daemonを実行します。

   注: ノードがアクティブになるまで待機します。

   スタンバイノードでnnmcluster -daemon を実行します。

注: 古いキーファイルを削除しないと、以下のようなエラーが発生する場合があります。

警告:新しい暗号化キーを生成するには、NNMi クラスターをシャットダウンする必要があります。

続行しますか? (y/n)

y

エラー:新しい暗号化キーの生成に失敗しました。

原因としては、キーサイズを増やしたため
現在のキーが無効である可能性があります。

既存のキーを削除し、再試行してください。

暗号化およびユーザーアカウントパスワード

注: この情報は、ライトウェイトディレクトリアクセスプロトコル (LDAP) またはCommon Access Card (CAC) アカウントには適用されません。

NNMiコンソールを使用して作成されたNNMiユーザーアカウントはNNMiデータベースに保存されます。これらのユーザーのパスワードはハッシュされ、データベースに保存されます。

ユーザーがNNMiコンソールにサインインするか、コマンドラインインタフェース (CLI) ツールを使用する場合、指定したパスワードはハッシュされ、データベースに保存されたハッシュ値と比較されま

す。ユーザーが正しいパスワードを指定すると、これらの2つのハッシュされた文字列が一致し、ユーザーは認証されます。

NNMiの従来のバージョン (9.x) はユーザーパスワードをハッシュするための暗号化アルゴリズムを使用していましたが、この方式は古くなりました。NNMi 10.00はユーザーアカウントパスワードにより強力なアルゴリズムを使用しています。ただし、ハッシュは一方向の暗号化であるため、復号化は不可能であり、NNMi 9.xから10.00へのアップグレード中に再暗号化することになります。

アップグレード時に、すべての既存のユーザーは従来の暗号化アルゴリズムを使用したデータベースに保存されたパスワードを保持します。ただし、従来のアルゴリズムを使用してハッシュされたパスワードを持つユーザーがログオンに成功すると、指定したパスワードは自動的に暗号設定ファイルで指定された新しいハッシュアルゴリズムを使用して再暗号化されます。

つまり、アップグレード後に各ユーザーが初めてログインするたびに、すべてのパスワードが少しずつ新しいアルゴリズムに更新されることになります。同じことが、将来的に暗号設定が変更された場合にも言えます。ユーザーパスワードは、次にログオンに成功したときに新しいハッシュアルゴリズムにアップグレードされます。

- ユーザーパスワードをアップグレードするには、<allowed>ブロックにリストされている従来のアルゴリズム (たとえば、MD5) が存在している必要があります。したがって、すべてのパスワードが移行されるまで<allowed>ブロックにリストされている従来のアルゴリズムを残しておいてください。
- <allowed>ブロックに従来のアルゴリズムが存在していないと、データベースでハッシュされた既存のパスワードを再ハッシュすることができません。したがって、関連付けられたユーザーはログオンできず、NNMiは新しいアルゴリズムを使用してパスワードを再暗号化できません。
- 従来のアルゴリズムを<allowed>ブロックから削除した場合、管理者は影響を受けるユーザーを削除して再作成するか、パスワードが従来のアルゴリズムで暗号化されたユーザーのそれぞれのパスワードをリセットする必要があります。

以下のコマンドを使用して、ユーザーのパスワードが暗号設定ファイルにリストされているアルゴリズムを使用しているか、またはユーザーのパスワードが暗号設定ファイルで指定されなくなった従来のアルゴリズムで暗号化されているかを判断します。

nnmsecurity.ovpl -listUserAccounts legacy

詳細については、nnmsecurity.ovpl のリファレンスページ、またはLinuxのマニュアルページを参照してください。

HP Performance Insight (OVPI) によるカスタムレポートパックのSNMP収集のNNMiへの移行

NNMiカスタムポーラー機能とHP Performance Insight (OVPI) を使用している場合、OVPIでのカスタムレポートパック収集をNNMiに移行できます。移行したOVPI収集はNNMiカスタムポーラー機能で使用できます。

NNMiカスタムポーラー機能では、SNMP MIB式を使用してNNMiがポーリングする必要のある追加情報を指定することによって、積極的にネットワーク管理を行えます。

カスタムポーラー収集は、収集 (ポーリング) する情報および収集したデータのNNMiによる処理方法を定義します。詳細については、NNMiヘルプの「カスタムポーラー収集を作成する」および「カスタムポーリングを設定する」を参照してください。『HP Network Node Manager i Softwareステップバイステップガイド (カスタムポーラーに関するホワイトペーパー)』も参照してください。

注: この手順は、OVPIでのカスタムレポートパックのSNMPベース収集のみの移行に使用します。

OVPIカスタムレポートパックに関連付けられたSNMP収集をNNMiに移行するには、以下の手順を実行します。

1. OVPIからNNMiに移行する必要がある収集ポリシーを特定します。
2. OVPI `collection_manager`ツールを使用して、OVPIサーバーからこれらのカスタムレポートパック内の収集ポリシーをエクスポートします。次に例を示します。

   注: OVPIサーバーは、収集を実行するリモートポーラーの場合とサテライトサーバーの場合があります。

   ```
   collection_manager -export <file name>
   ```

   詳細については、`collection_manager`のリファレンスページを参照してください。
3. NNMiカスタムポーラー収集に必要な追加情報を収集します。この情報は、以下のいずれかの方法を使用して`nnmmigrateovpli.ovpl`コマンドに渡すことができます。

   `nnmmigrateovpi.ovpl`コマンドライン引数として1つのTEELファイルの情報を指定する。次に例を示します。

   ```
   nnmmigrateovpi.ovpl – policyName myPolicy – teelFile /tmp/OVPI/myTeel.TEEL
   – pollInterval 300 – nodeGroup myNodeGroup
   ```

   `nnmmigrateovpi.ovpl`コマンドで-policyFile引数を使用して、1つのポリシーファイル内の複数のTEELファイルを指定する。次に例を示します。

   ```
   nnmmigrateovpi.ovpl -policyFile CP_policy_config.txt -teelDir /tmp/OVPI
   -batchFile generated_CP_commands.txt
   ```

エクスポートされるOVPI収集ポリシーファイルには、policy_name、table_name、poll_interval、datapipe_name、poll_from、user_name、server_name、group、group_server、desc列 が含 まれます。

以下の表に、このエクスポートされる情報がnnmmigrateovpi.ovplコマンドの必須情報にどのように関連するかを示します。

OVPI収集ポリシーファイルの列	nnmmigrate.ovplの必須フィールド
policy_name	ポリシー名
table_name	TEELファイル名
poll_interval	ポーリング間隔
グループ	ノードグループ

OVPI収集ポリシーファイルの情報を抽出するには、以下のLinuxコマンドを使用します。

```
cut -f1,2,3,8 -d',' ovpi_collection_policy.txt > CP_policy_config.txt
```

ここで、ovi_collection_policy.txtはエクスポートされるOVPI収集ポリシーファイル例の名前、CP_policy_config.txtはnnmcustompollerconfig.ovplコマンドへの入力に使用されるポリシーファイル (<policyfile>) 例の名前です。

4. エクスポートされるOVPI収集ポリシーファイルの内容を確認します。内容を確認するときに、以下の点に注意してください。
   - エクスポートされるOVPI収集ポリシーのtable_nameフィールドは、teel拡張子のないTEELファイル名と同じであるとみなされます。TEELファイル名がtable_nameとは異なる場合、table_nameがTEELファイル名と一致するように手動でファイルを編集する必要があります。
   - グループ名は、NNMiのノードグループには対応していない可能性があります。これらの名前が一致しない場合、以下のいずれかを実行します。
     - 移行コマンドに対する情報を指定するときに、このグループ名をNNMiノードグループ名と一致するように変更する。
     - エクスポートされるグループ名と一致するノードグループを作成する。
5. OVPI収集ポリシーで使用されるTEELファイルを探します。
6. TEELファイルをNNMiシステムの一時的な場所にコピーします。
7. nnmmigrateovpi.ovplを使用して TEELファイルに含まれるデータでカスタムポーラー収集を設定できるようにするために必要なコマンドを生成します。

ヒント: nnmmigrateovpi.ovplを使用して、1つのTEELファイルまたは複数のTEELファイルを移行できます。

詳細については、nnmmigrateovpi.ovplのリファレンスページを参照してください。

注意: 生成されるカスタムポーラー設定コマンドのいくつかのフィールドは、デフォルト値

を使用します。必要に応じて、要件を満たすようにこれらのフィールドを変更します。詳細については、nnmmigrateovpi.ovplのリファレンスページを参照してください。

以下の手順は、nnmmigrateovpi.ovplコマンドを使用して複数の収集を移行する方法の例です。この例では、前の手順の説明に従ってエクスポートされるOVPI収集ポリシーファイルがすでに作成済みで、その内容を確認していることを前提としています。

1. 以下のnnmmigrateovpi.ovplコマンドを実行します。

   nnmmigrateovpi.ovpl -policyFile <file name> -teelDir <directory where the TEEL files are present> [-batchFile <file name where generated commands are written>]

   次に例を示します。

   nnmmigrateovpi.ovpl -policyFile CP_policy_config.txt -teelDir /tmp/OVPI -batchFile generated_CP_commands.txt

2. NNMiカスタムポーラー設定コマンドnnmcustompollerconfig.ovplで、以下のように新しいバッチファイルを使用します。

   nnmcustompollerconfig.ovpl -batch <batch command file>

   次に例を示します。

   nnmcustompollerconfig.ovpl -batch generated_CP_commands.txt

   NNMiはバッチコマンドファイルに含まれる設定情報を使用して、カスタムポーラー収集を作成します。

3. NNMiコンソールからこれらのカスタムポーラー収集を表示するには、以下の手順を実行します。
   a. [**設定**] ワークスペースに移動します。
   b. [**モニタリング**] をクリックして展開します。
   c. [**カスタムポーラーの設定**] を選択します。
   d. [**カスタムポーラー収集**] タブに移動します。
      作成されたカスタムポーラー収集のリストが表示されます。

付録A: 追加情報

このセクションでは以下の付録について説明します。

アプリケーションフェイルオーバー構成のNNMiの手動設定

この付録の手順では、NNMiクラスター設定ウィザードを使用しないでアプリケーションフェイルオーバーを設定する方法を説明します。

注: Oracleデータベースでアプリケーションフェイルオーバーを使用している場合、以下の前提条件アクションを含め、この付録の設定手順を実行する必要があります。

[セカンダリサーバーのインストール] オプションを使用して、スタンバイサーバーをインストールする必要があります。スタンバイサーバーをプライマリサーバーとしてインストールした場合は、そのサーバーをアンインストールし、[セカンダリサーバーのインストール] オプションを使用して再インストールします。

NNMiをアンインストールする前に、最新のパッチから開始して、NNMiパッチをすべて逆順で削除します。パッチの削除プロセスは、NNMi管理サーバーで実行しているオペレーティングシステムによって異なります。インストールおよび削除手順については、パッチのマニュアルを参照してください。

アプリケーションフェイルオーバーを手動で設定するには、以下の手順を実行します。

1. 両方のノードで**ovstop**を実行します。
2. `nms-cluster.properties`ファイルに含まれる指示を参考にして、サーバーX (アクティブ) およびサーバーY (スタンバイ) のアプリケーションフェイルオーバー機能を設定します。以下の手順を実行します。

   注: 以下の手順では、ファイルのテキストブロックの行のコメントを解除し、テキストを変更することを**編集**と呼びます。

a. 以下のファイルを編集します。
   - Windowsの場合: %NnmDataDir%\shared\nnm\conf\props\nms-cluster.properties
   - Linuxの場合: $NnmDataDir/shared/nnm/conf/props/nms-cluster.properties

b. NNMiクラスターに一意の名前を宣言します。アクティブサーバーとスタンバイサーバーが同じ名前を使用するように設定します。

   com.hp.ov.nms.cluster.name=MyCluster

c. nms-cluster.propertiesファイルのcom.hp.ov.nms.cluster.member.hostnamesパラメーターに、クラスターのすべてのノードのホスト名を追加します。

   com.hp.ov.nms.cluster.member.hostnames = fqdn_for_active, fqdn_for_standby

   注: NNMi 9.0xでは、アプリケーションフェイルオーバー機能でUDPソリューションがサポートされ、クラスターホストはネットワークで自動的に検出されました。NNMi 9.2xからはUDPソリューションが排除され、TCPソリューションのみがサポートされます。NNMi 9.0xから移行する場合は、アプリケーションフェイルオーバーを機能させるために手順cを完了してクラスターホスト名を定義する必要があります。

d. 省略可能。nms-cluster.propertiesファイル内のその他のcom.hp.ov.nms.cluster*パラメーターを定義します。各パラメーターの変更方法については、nms-cluster.propertiesファイル内の指示に従ってください。

   注: Oracleデータベースでアプリケーションフェイルオーバーを使用している場合、NNMiではnms-cluster.propertiesファイルに含まれるデータベースパラメーターが無視されます。

3. 選択した方法に基づき、「アプリケーションフェイルオーバー環境での証明書の使用」(325ページ)に示されている指示を実行します。

   注意: アプリケーションフェイルオーバー機能を設定するときには、両方のノードのnnm.keystoreおよびnnm.truststoreファイルの内容をマージして、nnm.keystoreおよびnnm.truststoreを1つのファイルにする必要があります。方法を選択し、手順3の1セットの指示を完了する必要があります。

4. 以下のファイルをサーバーXからサーバーYにコピーします。
   - Windowsの場合:

     %NnmDataDir%\shared\nnm\conf\nnmcluster\cluster.keystore

   - Linuxの場合:

     $NnmDataDir/shared/nnm/conf/nnmcluster/cluster.keystore

5. nnmclusterコマンドをサーバーXとサーバーYの両方で実行します。

   各サーバーに、以下のように表示されます。

```
============================== Current cluster state
==============================

State ID:000000001000000005

日/時間:2011年3月15日 - 09:37:58 (GMT-0600)

クラスター名:このクラスター (キーCRC:626,187,650)

自動フェイルオーバー:有効

NNMデータベースの種類:組み込み

NNMで設定済みのACTIVEノード:NO_ACTIVE

NNMの現在のACTIVEノード:NO_ACTIVE

クラスターメンバー:

  Local?    NodeType  State                    OvStatus     Hostname/Address

  ------    --------  -----                    --------     ----------------------
-----

* REMOTE    ADMIN     n/a                      n/a
serverX.xxx.yyy.yourcompany.com/16.78.61.68:7800

  (SELF)    ADMIN     n/a                      n/a
serverY.xxx.yyy.yourcompany.com/16.78.61.71:7800
```

画面には、サーバーXとサーバーYの両方がリストされます。 両方のノードの情報が表示されない場合、それらのノードはお互いに通信していません。手順を進める前に、以下の点を確認して、修正してください。

- クラスター名が、サーバーXとサーバーYで異なっているかどうか。
- キーCRCが、サーバーXとサーバーYで異なっているかどうか。 サーバーXとサーバーYの両方で、以下のファイルの内容を確認してください。

  Windowsの場合: %NnmDataDir%\shared\nnm\conf\nnmcluster\cluster.keystore

  Linuxの場合: $NnmDataDir/shared/nnm/conf/nnmcluster/cluster.keystore
- サーバーXまたはサーバーYのファイアウォールによって、ノードの通信が妨げられているかどうか。
- nnm.keystoreファイルとnnm.truststoreファイルを確実にマージしたかどうか。このエラーが表示されるのは、**nnmcluster**コマンドを実行した後です。
- サーバーXとサーバーYで、異なるオペレーティングシステムが実行されているかどうか。たとえば、サーバーXでLinuxオペレーティングシステムが実行され、サーバーYでWindowsオペレーティングシステムが実行されている場合などです。このエラーが表示されるのは、**nnmcluster**コマンドを実行した後です。

- サーバーXとサーバーYが、異なるバージョンのNNMiを実行しているかどうか。たとえば、サーバーXがNNMi 10.01を実行しており、サーバーYがNNMi 10.01パッチ1 (リリース後) を実行している場合などです。このエラーが表示されるのは、`nnmcluster`コマンドを実行した後です。

6. サーバーXで、NNMiクラスターマネージャーを開始します。

   nnmcluster -daemon

   注: `nnmcluster -daemon`コマンドをNNMi管理サーバーXで実行すると、NNMiクラスターマネージャーが以下の起動ルーチンを実行します。

   - NNMi管理サーバー Xをクラスターに接続します。
   - 他のNNMi管理サーバーが存在しないことを検知します。
   - NNMi管理サーバーXはアクティブ状態に変わります。
   - NNMi管理サーバー X (アクティブサーバー) のNNMiサービスを開始します。
   - データベースのバックアップを作成します。

   詳細については、nnmclusterのリファレンスページ、またはLinuxのマニュアルページを参照してください。

7. サーバーXがクラスターの最初のアクティブノードになるまで数分待ちます。サーバーXで**nnmcluster -display**コマンドを実行し、表示された結果から`ACTIVE_NNM_STARTING`または`ACTIVE_SomeOtherState`の「`ACTIVE`」という語を検索します。サーバーXがアクティブノードであることを確認するまで手順8に進まないでください。
8. サーバーYでNNMiクラスターマネージャーを開始します。

   nnmcluster -daemon

   注: `nnmcluster -daemon`コマンドをNNMi管理サーバーYで実行すると、NNMiクラスターマネージャーが以下の起動ルーチンを実行します。

   - NNMi管理サーバー Yをクラスターに接続します。
   - NNMi管理サーバー Xが存在し、アクティブな状態であることが検出されます。ディスプレイに`STANDBY_INITIALIZING`と表示されます。
   - NNMi管理サーバーYのデータベースバックアップがNNMi管理サーバーXのバックアップと比較されます。 一致しない場合は、新しいデータベースバックアップがNNMi管理サーバーX (アクティブ) からNNMi管理サーバーY (スタンバイ) に送信されます。ディスプレイに`STANDBY_RECV_DBZIP`と表示されます。

- NNMi管理サーバー Yは、スタンバイ状態に該当するバックアップに最低限必要となる、トランザクションログの最小限のセットを受信します。ディスプレイにSTANDBY_RECV_TXLOGSと表示されます。
- NNMi管理サーバー Yは待機状態になり、新しいトランザクションログとハートビート信号をNNMi管理サーバー Xから受信し続けます。 ディスプレイにSTANDBY_READYと表示されます。

詳細については、nnmclusterのリファレンスページ、またはLinuxのマニュアルページを参照してください。

9. フェイルオーバーが発生した場合、サーバーXのNNMiコンソールは機能しなくなります。サーバーXのNNMiコンソールセッションを閉じて、サーバーY (新たにアクティブになったサーバー) にログオンします。NNMiユーザーに、サーバーX (アクティブNNMi管理サーバー) とサーバーY (スタンバイNNMi管理サーバー) への2つのブックマークを登録するように指示します。フェイルオーバーが発生すると、ユーザーはサーバーY (スタンバイNNMi管理サーバー) に接続できます。
10. ネットワークオペレーションセンター (NOC) の担当者に、サーバーXとサーバーYの両方にトラップを送信するようにデバイスを設定するように指示します。 サーバーX (アクティブ) が実行している間、サーバーXは転送されたトラップを処理し、サーバーY (スタンバイ) はそのトラップを無視します。

NNMi環境変数

HP Network Node Manager i Software (NNMi) には、ファイルシステム内の移動やスクリプトの作成に使用できる多数の環境変数があります。

この付録では、以下の内容を記載しています。

- 「このドキュメントで使用する環境変数」(503ページ)
- 「他の使用可能な環境変数」(504ページ)

このドキュメントで使用する環境変数

このドキュメントでは、主に以下の2つのNNMi環境変数を使用して、ファイルやディレクトリの場所を参照します。以下に示す変数はデフォルト値です。実際の値は、NNMiのインストール時に行った選択内容によって異なります。

- Windows Serverの場合:
  - %NnmInstallDir%:<drive>\Program Files (x86)\HP\HP BTO Software
  - %NnmDataDir%:<drive>\ProgramData\HP\HP BTO Software

> **注:** Windowsシステムでは、NNMiのインストールプロセスによってこれらのシステム環境変数が作成されるため、すべてのユーザーがいつでも使用できます。

- Linuxの場合:
  - `$NnmInstallDir:/opt/OV`
  - `$NnmDataDir:/var/opt/OV`

> **注:** Linuxシステムでは、これらの環境変数を使用する場合は手動で作成する必要があります。

また、このドキュメントには、NNMi管理サーバーでユーザーログオン設定を行うときに使用するNNMi環境変数も一部掲載されています。これらの変数の形式は`NNM_*`です。NNMi環境変数の詳細リストについては、「他の使用可能な環境変数」(504ページ) を参照してください。

他の使用可能な環境変数

NNMi管理者は、いくつかのNNMiファイルの場所に定期的にアクセスします。NNMiには、通常アクセスする場所へ移動するためのさまざまな環境変数を設定するスクリプトが用意されています。

NNMi環境変数の拡張リストをセットアップするには、次の例のようなコマンドを使用します。

- Windowsの場合:`"C:\Program Files (x86)\HP\HP BTO Software\bin\nnm.envvars.bat"`
- Linuxの場合: `. /opt/OV/bin/nnm.envvars.sh`

上記の各OS用のコマンドを実行した後で、「Windows OSでの環境変数のデフォルトの場所」または「Linux OSでの環境変数のデフォルトの場所」で示すNNMi環境変数を使用して、頻繁に使用するNNMiファイルの場所に移動できます。

Windows OSでの環境変数のデフォルトの場所

変数	Windows (例)
%NNM_BIN%	C:\Program Files (x86)\HP\HP BTO Software\bin
%NNM_CONF%	C:\ProgramData\HP\HP BTO Software\conf
%NNM_DATA%	C:\ProgramData\HP\HP BTO Software\
%NNM_DB%	C:\ProgramData\HP\HP BTO Software\shared\nnm\databases
%NNM_JAVA%	C:\Program Files (x86)\HP\HP BTO Software\nonOV\jdk\hpsw\bin\java.exe
%NNM_JAVA_PATH_SEP%	;
%NNM_JRE%	C:\Program Files (x86)\HP\HP BTO Software\nonOV\jdk\hpsw

Windows OSでの環境変数のデフォルトの場所 (続き)

変数	Windows (例)
%NNM_LOG%	C:\ProgramData\HP\HP BTO Software\log
%NNM_LRF%	C:\ProgramData\HP\HP BTO Software\shared\nnm\lrf
%NNM_PRIV_LOG%	C:\ProgramData\HP\HP BTO Software\log
%NNM_PROPS%	C:\ProgramData\HP\HP BTO Software\shared\nnm\conf\props
%NNM_SHARED_CONF%	C:\ProgramData\HP\HP BTO Software\shared\nnm\conf
%NNM_SHARE_LOG%	C:\ProgramData\HP\HP BTO Software\log
%NNM_SNMP_MIBS%	C:\Program Files (x86)\HP\HP BTO Software\misc\nnm\snmp-mibs
%NNM_TMP%	C:\ProgramData\HP\HP BTO Software\tmp
%NNM_USER_SNMP_MIBS%	C:\ProgramData\HP\HP BTO Software\shared\nnm\user-snmp-mibs
%NNM_WWW%	C:\ProgramData\HP\HP BTO Software\shared\nnm\www

Linux OSでの環境変数のデフォルトの場所

変数	Linux
$NNM_BIN	/opt/OV/bin
$NNM_CONF	/var/opt/OV/conf
$NNM_DATA	/var/opt/OV
$NNM_DB	/var/opt/OV/shared/nnm/databases
$NNM_JAVA	/opt/OV/nonOV/jdk/hpsw/bin/java
$NNM_JAVA_PATH_SEP	:
$NNM_JRE	/opt/OV/nonOV/jdk/nnm
$NNM_LOG	/var/opt/OV/log
$NNM_LRF	/var/opt/OV/shared/nnm/lrf
$NNM_PRIV_LOG	/var/opt/OV/log
$NNM_PROPS	/var/opt/OV/shared/nnm/conf/props
$NNM_SHARED_CONF	/var/opt/OV/shared/nnm/conf
$NNM_SHARE_LOG	/var/opt/OV/log

Linux OSでの環境変数のデフォルトの場所 (続き)

変数	Linux
$NNM_SNMP_MIBS	/opt/OV/misc/nnm/snmp-mibs
$NNM_TMP	/var/opt/OV/tmp
$NNM_USER_SNMP_MIBS	/var/opt/OV/shared/nnm/user-snmp-mibs
$NNM_WWW	/var/opt/OV/shared/nnm/www

NNMiおよびNNM iSPIのデフォルトポート

この付録では、NNMiとNNM iSPIがネットワーク通信に使用するデフォルトポートを示します。製品間でポートの競合が発生した場合は、「設定の変更」列の説明に従ってそのポート番号のほとんどを変更できます。

後続のトピックでは、個々のHP Network Management Software製品で使用されるポートについて説明されています。

- 「HP Network Node Manager i Softwareポート」(507ページ)
- 「NNM iSPI for MPLSのポート」(520ページ)
- 「NNM iSPI for IP Telephonyのポート」(524ページ)
- 「NNM iSPI for IP Multicastのポート」(528ページ)
- 「NNM iSPI Performance for Trafficのポート」(532ページ)
- 「NNM iSPI Performance for QAのポート」(541ページ)
- 「NNM iSPI Performance for MetricsおよびNPSのポート」(545ページ)
- 「NNM iSPI NETのポート」(547ページ)

HP Network Node Manager i Softwareポート

NNMiポートは以下のカテゴリに分類されます。

- NNMi管理サーバーで使用されるポート
- NNMi管理サーバーと他のシステムの通信で使用されるポート
- グローバルネットワーク管理で必須のアクセス可能ソケット

NNMi管理サーバーで使用されるポート

以下の表に、NNMiが管理サーバーで使用するポートを示します。NNMiはそれらのポートで待機します。ポートの競合が発生した場合は、「設定の変更」列の説明に従ってそのポート番号のほとんどを変更できます。詳細については、nnm.portsリファレンスページ、またはLinuxのマニュアルページを参照してください。

注: アプリケーションフェイルオーバーを正しく機能させるために、TCPポート7800-7810をオープンにしてください。アプリケーションフェイルオーバーが正しく機能するには、アクティブおよびスタンバイNNMi管理サーバーの相互のネットワークアクセスに制限のないことが必要です。

NNMi管理サーバーで使用されるポート

ポート	タイプ	名前	目的	設定の変更
80	TCP	nmsas.server.port.web.http	デフォルトHTTPポート - Web UIとWebサービスで使用 - GNM設定では、NNMiはこのポートを使	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。 インストール時に変更することもできます。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			用してグローバルマネージャーからリージョナルマネージャーへの通信を確立します。 - このポートが開くと、双方向となります。	
162	UDP	trapPort	SNMPトラップポート	nnmtrapconfig.ovpl Perlスクリプトを使用して変更します。詳細については、nnmtrapconfig.ovplのリファレンスページ、またはLinuxのマニュアルページを参照してください。
443	TCP	nmsas.server.port.web.https	デフォルトのセキュアーHTTPSポート (SSL) - Web UIとWebサービスで使用	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。
1098	TCP	nmsas.server.port.naming.rmi	- NNMiコマンドラインツールで使用され、NNMiで使	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			用されるさまざまなサービスと通信します。 - システムのファイアウォールを設定して、これらのポートへのアクセスをローカルホストのみに制限することをお勧めします。	
1099	TCP	nmsas.server.port.naming.port	- NNMiコマンドラインツールで使用され、NNMiで使用されるさまざまなサービスと通信します。 - システムのファイアウォールを設定して、これ	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			らのポートへのアクセスをローカルホストのみに制限することをお勧めします。	
3873	TCP	nmsas.server.port.remoting.ejb3	- NNMiコマンドラインツールで使用され、NNMiで使用されるさまざまなサービスと通信します。 - システムのファイアウォールを設定して、これらのポートへのアクセスをローカルホストのみに制限することをお勧めします。	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。
4444	TCP	nmsas.server.port.jmx.jrmp	- NNMiコマンドラインツー	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ルで使用され、NNMiで使用されるさまざまなサービスと通信します。 - システムのファイアウォールを設定して、これらのポートへのアクセスをローカルホストのみに制限することをお勧めします。	local.propertiesファイル (Linux) を変更します。
4445	TCP	nmsas.server.port.jmx.rmi	- NNMiコマンドラインツールで使用され、NNMiで使用されるさまざまなサービスと通信します。 - システムのファイア	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ウォールを設定して、これらのポートへのアクセスをローカルホストのみに制限することをお勧めします。	
4446	TCP	nmsas.server.port.invoker.unified	- NNMiコマンドラインツールで使用され、NNMiで使用されるさまざまなサービスと通信します。 - システムのファイアウォールを設定して、これらのポートへのアクセスをローカルホストのみに制限することをお勧めします。	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
4457	TCP	nmsas.server.port.hq	- グローバルネットワーク管理の非暗号化トラフィックで使用します。 - メッセージングでは、グローバルマネージャーからリージョナルマネージャーへ通信が行われます。 - このポートが開くと、双方向となります。	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。
4459	TCP	nmsas.server.port.hq.ssl	- グローバルネットワーク管理の暗号化トラフィックで使用します。 - メッセージ	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ングでは、グローバルマネージャーからリージョナルマネージャーへ通信が行われます。 - このポートが開くと、双方向となります。	
4712	TCP	nmsas.server.port.ts.recovery	内部トランザクションサービスのポート	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。
4713	TCP	nmsas.server.port.ts.status	内部トランザクションサービスのポート	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。
4714	TCP	nmsas.server.port.ts.id	内部トランザクションサービスのポート	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理	%NNM_CONF%\nnm\props\nms-local.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-local.propertiesファイル (Linux) を変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			サーバーに対して組み込みデータベースが待機するポートです。	
7800-7810	TCP		- アプリケーションのフェイルオーバーで使用するJGroupsポート。 - アプリケーションフェイルオーバーを使用していない場合、システムのファイアウォールを設定して、これらのポートへのアクセスを制限することをお勧めします。	%NNM_CONF%\nnm\props\nms-cluster.propertiesファイル (Windows) または$NNM_CONF/nnm/props/nms-cluster.propertiesファイル (Linux) を変更します。
8886	TCP	OVsPMD_MGMT	NNMi ovspmd (プロセスマ	/etc/servicesファイルを変更します。

NNMi管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ネージャー) 管理ポート	
8887	TCP	OVsPMD_REQ	NNMi ovsmpd (プロセスマネージャー) 要求ポート	/etc/servicesファイルを変更します。
8989	TCP	com.hp.ov.nms.events.action.server.port	アクションサーバーポート	%NnmInstallDir%\misc\nnm\props\shared\nnmaction.propertiesファイル (Windows) または $NnmInstallDir/misc/nnm/props/shared/nnmaction.propertiesファイル (Linux) を変更します。

NNMi管理サーバーと他のシステムの通信で使用されるポート

以下の表に、他のシステムとの通信でNNMiが使用するポートの一部を示します。ファイアウォールによってNNMiがこれらのシステムから分断されている場合は、そのファイアウォールでこれらのポートの多くを開く必要があります。実際のポートセットは、NNMiで使用するように設定した統合セットと、それらの統合の設定方法に応じて異なります。4列目がクライアントであればNNMiはそのポートに接続または送信し、4列目がサーバーであればNNMiはそのポートで待機します。

NNMi管理サーバーと他のシステムの通信で使用されるポート

ポート	タイプ	目的	クライアント、サーバー
80	TCP	NNMiのデフォルトHTTPポート、Web UIとWebサービスで使用	サーバー
80	TCP	NNMiが他のアプリケーションに接続するときのデフォルトHTTPポート。実際のポートはNNMiの設定によって異なります。	クライアント

NNMi管理サーバーと他のシステムの通信で使用されるポート (続き)

ポート	タイプ	目的	クライアント、サーバー
161	UDP	SNMP要求ポート	クライアント
162	UDP	SNMPトラップポート - NNMiが受信するトラップ	サーバー
162	UDP	SNMPトラップポート。トラップ転送、Northboundインタフェース、またはNetCool統合	クライアント
389	TCP	デフォルトLDAPポート	クライアント
395	UDP	nGenius Probe SNMPトラップポート	クライアント
443	TCP	NNMiが他のアプリケーションに接続するときのデフォルトのセキュアーHTTPSポート、実際のポートはNNMiの設定によって異なります。 HP OM on WindowsのデフォルトHTTPSポート	クライアント
443	TCP	デフォルトのセキュアーHTTPSポート、Web UIとWebサービスで使用	サーバー
636	TCP	デフォルトのセキュアーLDAPポート (SSL)	クライアント
1741	TCP	デフォルトのCiscoWorks LMS Webサービスポート	クライアント
4457	TCP	グローバルネットワーク管理の非暗号化トラフィックで使用します。グローバルマネージャーからリージョナルマネージャーに対して接続を行います。	クライアント、サーバー
4459	TCP	グローバルネットワーク管理の暗号化トラフィックで使用します。グローバルマネージャーからリージョナルマネージャーに対して接続を行います。	クライアント、サーバー
7800-7810	TCP	アプリケーションのフェイルオーバーで使用するJGroupsポート	クライアントとサーバー
8004	TCP	別のWebサーバーがすでにポート80を使用している場合のNNMiのデフォルト	サーバー

NNMi管理サーバーと他のシステムの通信で使用されるポート (続き)

ポート	タイプ	目的	クライアント、サーバー
		HTTPポート。Web UIとWebサービスで使用。NNMi管理サーバーの実際のHTTPポートを検証します。	
8080	TCP	NNMiと同じシステムにインストールされている場合に、NAに接続するときのデフォルトHTTPポート。 HP UCMDB Webサービスのデフォルトポート	クライアント
8443または8444	TCP	HP OM for UNIXに接続するときのデフォルトHTTPポート	クライアント
9300	TCP	NNM iSPI Performance for Metricsに接続するときのデフォルトHTTPポート	クライアント
50000	TCP	SIMに接続するときのデフォルトHTTPSポート	クライアント

注: 検出のためにICMP障害ポーリングまたはpingスィープを使用するようにNNMiを設定する場合は、ICMPパケットを通過させるようにファイアウォールを設定してください。

注: NNMi-HP OM統合のWebサービス方式は、ファイアウォールを介して機能することはありませんが、Northboundインタフェースを使用するNNMi-HP OM統合はファイアウォールを介して機能します。

グローバルネットワーク管理で必須のアクセス可能ソケット

グローバルNNMi管理サーバーからリージョナルNNMi管理サーバーに対して、以下の表に示すウェルノウンポートがアクセス可能になっている必要があります。グローバルネットワーク管理機能では、TCPアクセス用にグローバルNNMi管理サーバーからリージョナルNNMi管理サーバーに対して、これらのポートが開いている必要があります。リージョナルNNMi管理サーバーが逆に、グローバルNNMi管理サーバーに対してソケットを開くことはありません。

グローバルネットワーク管理で必須のアクセス可能ソケット

セキュリティ	パラメーター	TCPポート
非SSL	nmsas.server.port.web.http	80
	nmsas.server.port.hq	4457
SSL	nmsas.server.port.web.https	443
	nmsas.server.port.hq.ssl	4459

NNM iSPI for MPLSのポート

以下の表に、HP Network Node Manager iSPI for MPLS Softwareが管理サーバーで使用するポートを示します。ポートが競合する場合、%NnmDataDir%/nmsas/mpls/server.propertiesにあるserver.propertiesファイルを使用してこれらのポート番号のほぼすべてを変更できます。

HP Network Node Manager iSPI for MPLS Software管理サーバーで使用されるポート

ポート	タイプ	名前	目的	設定の変更
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理サーバーに対して組み込みデータベースが待機するポートです。このポートは、nms-local.propertiesファイルでNNMiに設定するポートと同じです。	N/A
24040	TCP	nmsas.server.port.web.http	Web UIで使用される、デフォルトのHTTPポート。	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
24041	TCP	nmsas.server.port.remoting.ejb3	デフォルトの	%NnmDataDir%\nmsas\mpls\server.properti

HP Network Node Manager iSPI for MPLS Software管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			EJB3リモートコネクターポート	esファイル (Windows) または $NnmDataDir/nmsas/mpls/server.properties ファイル (Linux) を変更します。
24043	TCP	nmsas.server.port.web.https	Web UIで使用される、デフォルトのセキュアーHTTPSポート (SSL)。	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.properties ファイル (Linux) を変更します。インストール時に変更することもできます。
24044	TCP	nmsas.server.port.jmx.jrmp	デフォルトのRMIオブジェクトポート (JRMP呼び出し元)	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.properties ファイル (Linux) を変更します。
24045	TCP	nmsas.server.port.invoker.unified	デフォルトのRMIリモートサーバーコネクターポート	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.properties ファイル (Linux) を変更します。
24046	TCP	nmsas.server.port.naming.port	デフォルトのブートストラップJNPサービスポート (JNDIプロバイダー)	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.properties ファイル (Linux) を変更します。インストール時に変更することもできます。
24047	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.properties ファイル (Linux) を変更します。

HP Network Node Manager iSPI for MPLS Software管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			します。	
24048	TCP	nmsas.server.port.jmx.rmi	デフォルトのRMIプール済み呼び出し元ポート	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。
24049	TCP	nmsas.server.port.naming.rmi	RMIネームサービスのデフォルトポート	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。
24092	TCP	nmsas.server.port.hq.ssl	グローバルネットワーク管理の暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。
24712	TCP	nmsas.server.port.ts.recovery	トランザクションサービスで使用するデフォルトの復旧ポート。	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。
24713	TCP	nmsas.server.port.ts.status	トランザクションサービスで使用するデフォルトのステータスポート。	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。

HP Network Node Manager iSPI for MPLS Software管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
24714	TCP	nmsas.server.port.ts.id	トランザクションサービスで使用するデフォルトポート。	%NnmDataDir%\nmsas\mpls\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/mpls/server.propertiesファイル (Linux) を変更します。

NNM iSPI for IP Telephonyのポート

以下の表に、NNM iSPI for IP Telephonyが管理サーバーで使用するポートを示します。ポートが競合する場合、%NnmDataDir%/nmsas/multicast/server.propertiesにある server.propertiesファイルを使用してこれらのポート番号のほぼすべてを変更できます。

NNM iSPI for IP Telephony管理サーバーで使用されるポート

ポート	タイプ	名前	目的	設定の変更
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理サーバーに対して組み込みデータベースが待機するポートです。このポートは、nms-local.propertiesファイルでNNMiに設定するポートと同じです。	N/A
10080	TCP	nmsas.server.port.web.http	Web UIで使用される、デフォルトのHTTPポート。	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
10083	TCP	nmsas.server.port.naming.rmi	RMIネームサービ	%NnmDataDir%\nmsas\ipt\server.propertie

NNM iSPI for IP Telephony管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			スのデフォルトポート	sファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10084	TCP	nmsas.server.port.jmx.jrmp	デフォルトのRMIオブジェクトポート (JRMP呼び出し元)	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10085	TCP	nmsas.server.port.jmx.rmi	デフォルトのRMIプール済み呼び出し元ポート	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10086	TCP	nmsas.server.port.invoker.unified	デフォルトのRMIリモートサーバーコネクターポート	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10087	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10089	TCP	nmsas.server.port.remoting.ejb3	デフォルトのEJB3リモートコネクターポート	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10092	TCP	nmsas.server.port.hq.ssl	グローバルネッ	%NnmDataDir%\nmsas\ipt\server.propertie

NNM iSPI for IP Telephony管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			トワーク管理の暗号化トラフィックで使用します。	sファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
10099	TCP	nmsas.server.port.naming.port	デフォルトのブートストラップJNPサービスポート (JNDIプロバイダー)	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。インストール時に変更することもできます。
10443	TCP	nmsas.server.port.web.https	Web UIで使用される、デフォルトのセキュアーHTTPSポート (SSL)。	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。インストール時に変更することもできます。
14712	TCP	nmsas.server.port.ts.recovery	トランザクションサービスで使用するデフォルトの復旧ポート。	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
14713	TCP	nmsas.server.port.ts.status	トランザクションサービスで使用するデフォルトのステータスポート。	%NnmDataDir%\nmsas\ipt\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。
14714	TCP	nmsas.server.port.ts.id	トランザクショ	%NnmDataDir%\nmsas\ipt\server.propertie

NNM iSPI for IP Telephony管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ンサービスで使用するデフォルトポート。	sファイル (Windows) または $NnmDataDir/nmsas/ipt/server.properties ファイル (Linux) を変更します。

NNM iSPI for IP Multicastのポート

以下の表に、NNM iSPI for IP Multicastが管理サーバーで使用するポートを示します。ポートが競合する場合、%NnmDataDir%/nmsas/multicast/server.propertiesにある server.propertiesファイルを使用してこれらのポート番号のほぼすべてを変更できます。

NNM iSPI for IP Multicast管理サーバーで使用されるポート

ポート	タイプ	名前	目的	設定の変更
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理サーバーに対して組み込みデータベースが待機するポートです。このポートは、nms-local.propertiesファイルでNNMiに設定するポートと同じです。	N/A
8084	TCP	nmsas.server.port.web.http	Web UIで使用される、デフォルトのHTTPポート。	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
14083	TCP	nmsas.server.port.naming.rmi	RMIネームサー	%NnmDataDir%\nmsas\multicast\server.propert

NNM iSPI for IP Multicast管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ビスのデフォルトポート	iesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14084	TCP	nmsas.server.port.jmx.jrmp	デフォルトのRMIオブジェクトポート (JRMP呼び出し元)	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14085	TCP	nmsas.server.port.jmx.rmi	デフォルトのRMIプール済み呼び出し元ポート	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14086	TCP	nmsas.server.port.invoker.unified	デフォルトのRMIリモートサーバーコネクターポート	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14087	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14089	TCP	nmsas.server.port.remoting.ejb3	デフォルトのEJB3リモートコネクターポート	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。

NNM iSPI for IP Multicast管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
14092	TCP	nmsas.server.port.hq.ssl	グローバルネットワーク管理の暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\multicast\server.propertiesファイル(Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル(Linux) を変更します。
14099	TCP	nmsas.server.port.naming.port	デフォルトのブートストラップJNPサービスポート (JNDIプロバイダー)	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
14102	TCP	nmsas.server.port.ts.id	トランザクションサービスで使用するデフォルトポート。	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14103	TCP	nmsas.server.port.ts.recovery	トランザクションサービスで使用するデフォルトの復旧ポート。	%NnmDataDir%\nmsas\multicast\server.propertiesファイル(Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル(Linux) を変更します。
14104	TCP	nmsas.server.port.ts.status	トランザクションサービスで使用するデフォルトのステータスポート。	%NnmDataDir%\nmsas\multicast\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。
14443	TCP	nmsas.server.port.web.https	Web UIで使用さ	%NnmDataDir%\nmsas\multicast\server.propert

NNM iSPI for IP Multicast管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			れる、デフォルトのセキュアーHTTPSポート (SSL)。	iesファイル (Windows) または $NnmDataDir/nmsas/multicast/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。

NNM iSPI Performance for Trafficのポート

NNMi SPI Performance for Trafficポートは以下のカテゴリに分類されます。

- NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート
- NNMi SPI Performance for Traffic管理サーバー (トラフィックリーフ) で使用されるポート
- NNMi SPI Performance for Traffic管理サーバーと他のシステムの通信で使用されるポート

NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート

以下の表に、NNMi SPI Performance for Traffic (トラフィックマスターコンポーネント) が管理サーバーで使用するポートを示します。ポートが競合する場合、%NnmDataDir%/nmsas/traffic-master/server.propertiesにあるserver.propertiesファイルを使用してこれらのポート番号のほぼすべてを変更できます。

NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート

ポート	タイプ	名前	目的	設定の変更
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理サーバーに対して組み込みデータベースが待機するポートです。このポートは、nms-local.propertiesファイルでNNMiに設定するポートと同じです。	N/A
12080	TCP	nmsas.server.port.web.http	Web UIで使用される、デフォルトのHTTPポート。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-

NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
				master/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
12043	TCP	nmsas.server.port.web.https	Web UIで使用される、デフォルトのセキュアーHTTPSポート (SSL)。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
12083	TCP	nmsas.server.port.naming.rmi	RMIネームサービスのデフォルトポート	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12084	TCP	nmsas.server.port.jmx.jrmp	デフォルトのRMIオブジェクトポート (JRMP呼び出し元)	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12085	TCP	nmsas.server.port.jmx.rmi	デフォルトのRMIプール済み呼び出し元ポート	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイ

NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
				ル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12086	TCP	nmsas.server.port.invoker.unified	デフォルトのRMIリモートサーバーコネクターポート	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12087	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12089	TCP	nmsas.server.port.remoting.ejb3	デフォルトのEJB3	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12092	TCP	nmsas.server.port.hq.ssl	グローバルネットワーク管理の暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-

NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
				master/server.propertiesファイル (Linux) を変更します。
12099	TCP	nmsas.server.port.naming.port	デフォルトのブートストラップJNPサービスポート (JNDIプロバイダー)	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
12712	TCP	nmsas.server.port.ts.recovery	トランザクションサービスで使用するデフォルトの復旧ポート。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12713	TCP	nmsas.server.port.ts.status	トランザクションサービスで使用するデフォルトのステータスポート。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-master/server.propertiesファイル (Linux) を変更します。
12714	TCP	nmsas.server.port.ts.id	トランザクションサービスで使用するデフォルトポート。	%NnmDataDir%\nmsas\traffic-master\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-

NNMi SPI Performance for Traffic 管理サーバー (トラフィックマスター) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
				master/server.propertiesファイル (Linux) を変更します。

NNMi SPI Performance for Traffic管理サーバー (トラフィックリーフ) で使用されるポート

以下の表に、NNMi SPI Performance for Traffic (トラフィックリーフコンポーネント) が管理サーバーで使用するポートを示します。ポートが競合する場合、%NnmDataDir%/nmsas/traffic-leaf/server.propertiesにあるserver.propertiesファイルを使用してこれらのポート番号のほぼすべてを変更できます。

NNMi SPI Performance for Traffic管理サーバー (トラフィックリーフ) で使用されるポート

ポート	タイプ	名前	目的	設定の変更
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理サーバーに対して組み込みデータベースが待機するポートです。このポートは、nms-local.propertiesファイルでNNMiに設定するポートと同じです。	N/A
11080	TCP	nmsas.server.port.web.http	Web UIで使用される、デフォルトのHTTPポート。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。

NNMi SPI Performance for Traffic管理サーバー (トラフィックリーフ) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
11043	TCP	nmsas.server.port.web.https	Web UIで使用される、デフォルトのセキュアーHTTPSポート (SSL)。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
11083	TCP	nmsas.server.port.naming.rmi	RMIネームサービスのデフォルトポート	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11084	TCP	nmsas.server.port.jmx.jrmp	デフォルトのRMIオブジェクトポート (JRMP呼び出し元)	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11085	TCP	nmsas.server.port.jmx.rmi	デフォルトのRMIプール済み呼び出し元ポート	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11086	TCP	nmsas.server.port.invoker.unified	デフォルトのRMIリモート	%NnmDataDir%\nmsas\traffic-

NNMi SPI Performance for Traffic管理サーバー (トラフィックリーフ) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			サーバーコネクターポート	leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11087	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11089	TCP	nmsas.server.port.remoting.ejb3	デフォルトのEJB3リモートコネクターポート	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11092	TCP	nmsas.server.port.hq.ssl	グローバルネットワーク管理の暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11099	TCP	nmsas.server.port.naming.port	デフォルトのブートストラップJNPサービスポート (JNDIプロバイダー)	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または

NNMi SPI Performance for Traffic管理サーバー (トラフィックリーフ) で使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
				$NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
11712	TCP	nmsas.server.port.ts.recovery	トランザクションサービスで使用するデフォルトの復旧ポート。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11713	TCP	nmsas.server.port.ts.status	トランザクションサービスで使用するデフォルトのステータスポート。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。
11714	TCP	nmsas.server.port.ts.id	トランザクションサービスで使用するデフォルトポート。	%NnmDataDir%\nmsas\traffic-leaf\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/traffic-leaf/server.propertiesファイル (Linux) を変更します。

NNMi SPI Performance for Traffic管理サーバーと他のシステムの通信で使用されるポート

以下の表に、他のシステムとの通信でNNMi SPI Performance for Trafficが使用するポートの一部を示します。ファイアウォールによってNNMi SPI Performance for Trafficがこれらのシステムから分断されている場合は、そのファイアウォールでこれらのポートの多くを開く必要があり

ます。実際のポートセットは、NNMi SPI Performance for Trafficで使用するように設定した統合セットと、それらの統合の設定方法に応じて異なります。4列目がクライアントであれば、NNMi SPI Performance for Trafficはそのポートに接続または送信し、4列目がサーバーであればNNMi SPI Performance for Trafficはそのポートで待機します。

管理サーバーと他のシステムの通信で使用されるポート

ポート	タイプ	目的	クライアントまたはサーバー
任意の空きポート	TCP	Avayaストリーミング	サーバー
任意の空きポート	TCP	RTCPサーバー	サーバー
22	TCP	Cisco/Avaya SSH通信	クライアント
22/23	TCP	Cisco FTP/SFTP通信	サーバー
23	TCP	Avaya Survivable通信	クライアント
8000 (設定可能)	TCP	.NETプロキシ (IPT付属)	クライアント
8443	TCP	Cisco AXL通信	クライアント

NNM iSPI Performance for QAのポート

以下の表に、NNMi SPI Performance for QAが管理サーバーで使用するポートを示します。ポートが競合する場合、%NnmDataDir%/nmsas/multicast/server.propertiesにある server.propertiesファイルを使用してこれらのポート番号のほぼすべてを変更できます。

NNMi SPI Performance for QA管理サーバーで使用されるポート

ポート	タイプ	名前	目的	設定の変更
5432	TCP	com.hp.ov.nms.postgres.port	このPostgreSQLポートは、このNNMi管理サーバーに対して組み込みデータベースが待機するポートです。このポートは、nms-local.propertiesファイルでNNMiに設定するポートと同じです。	N/A
54040	TCP	nmsas.server.port.web.http	Web UIで使用される、デフォルトのHTTPポート。	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.propertiesファイル (Linux) を変更します。インストール時に変更することもできます。
54041	TCP	nmsas.server.port.remoting.ejb3	リモートejbコー	%NnmDataDir%\nmsas\qa\server.propertie

NNMi SPI Performance for QA管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			ルの呼び出しで使用します。	sファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54043	TCP	nmsas.server.port.web.https	Web UIで使用される、デフォルトのセキュアーHTTPSポート (SSL)。	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。インストール時に変更することもできます。
54045	TCP	nmsas.server.port.invoker.unified	jboss remoting サービスで使用します。	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54046	TCP	nmsas.server.port.naming.port	デフォルトのブートストラップJNPサービスポート (JNDIプロバイダー)	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。インストール時に変更することもできます。
54047	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54048	TCP	nmsas.server.port.jmx.rmi	デフォルトのRMIプール済み呼び出し元ポート	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties

NNMi SPI Performance for QA管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
				ファイル (Linux) を変更します。
54049	TCP	nmsas.server.port.naming.rmi	RMIネームサービスのデフォルトポート	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.propertiesファイル (Linux) を変更します。
54084	TCP	nmsas.server.port.jmx.jrmp	デフォルトのRMIオブジェクトポート (JRMP呼び出し元)	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.propertiesファイル (Linux) を変更します。
54086	TCP	nmsas.server.port.invoker.unified	デフォルトのRMIリモートサーバーコネクターポート	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.propertiesファイル (Linux) を変更します。
54087	TCP	nmsas.server.port.hq	グローバルネットワーク管理の非暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.propertiesファイル (Linux) を変更します。
54088	TCP	nmsas.server.port.hq.ssl	グローバルネットワーク管理の暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または $NnmDataDir/nmsas/qa/server.propertiesファイル (Linux) を変更します。
54089	TCP	nmsas.server.port.remoting.ejb3	デフォルトの	%NnmDataDir%\nmsas\qa\server.propertiesファイル (Windows) または

NNMi SPI Performance for QA管理サーバーで使用されるポート (続き)

ポート	タイプ	名前	目的	設定の変更
			EJB3リモートコネクターポート	$NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54092	TCP	nmsas.server.port.hq.ssl	グローバルネットワーク管理の暗号化トラフィックで使用します。	%NnmDataDir%\nmsas\qa\server.properties ファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54712	TCP	nmsas.server.port.ts.recovery	トランザクションサービスで使用するデフォルトの復旧ポート。	%NnmDataDir%\nmsas\qa\server.properties ファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54713	TCP	nmsas.server.port.ts.status	トランザクションサービスで使用するデフォルトのステータスポート。	%NnmDataDir%\nmsas\qa\server.properties ファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。
54714	TCP	nmsas.server.port.ts.id	トランザクションサービスで使用するデフォルトポート。	%NnmDataDir%\nmsas\qa\server.properties ファイル (Windows) または $NnmDataDir/nmsas/qa/server.properties ファイル (Linux) を変更します。

NNM iSPI Performance for MetricsおよびNPSのポート

以下の表に、NNM iSPI Performance for MetricsおよびNetwork Performance Server (NPS) で必要となるポートを示します。ポートが競合する場合、これらのポート番号のほぼすべてを変更できます。

注: NNMiとNPSが共存していない場合、OSのネットワークファイル共有で使用されるネットワークポートも必要になります (LinuxではNFSサービス、WindowsではWindowsファイル共有)。

NNM iSPI Performance for MetricsおよびNPSで必要となるポート

ポート	タイプ	名前	目的	設定の変更
9300	TCP	NPS UI	Web UIとBI Webサービスで使用される、デフォルトのHTTPポート。	configureWebAccess.ovplを使用して変更します。
9301	TCP	Sybase ASE	Sybase ASE (BIコンテンツマネージャーデータベース)。同じサーバー上で実行中のプロセスによって使用されます。	変更できません。
9302	TCP	Sybase IQ Agent	Sybase IQ Agentサービス。同じサーバー上で実行中のプロセスによって使用されます。	変更できません。
9303	TCP	Sybase IQ - PerfSPI DB	すべてのNPS ExtensionPackのデータを保存するために使用するSybase IQデータベース。同じサーバー上で実行中のプロセスによって使用されます。	変更できません。
9305	TCP	NPS UI - SSL	Web UIとBI Webサービスで使用される、デフォルトのセキュアーHTTPSポート (SSL)。	configureWebAccess.ovplを使用して変更します。
9306	TCP	データベースのSQL再書き込みプロキシ - PerfSPI	BIサーバーによって使用される、Perfspiデータベースの SQL再書き込みプロキシ。同じサーバー上で実行中のプロセスによって使用されます。	変更できません。

NNM iSPI Performance for MetricsおよびNPSで必要となるポート (続き)

ポート	タイプ	名前	目的	設定の変更
		DB		
9308	TCP	Sybase ASEバックアップサーバー	BIコンテンツマネージャーのデータベースのSybase ASEバックアップサーバー。同じサーバー上で実行中のプロセスによって使用されます。	変更できません。

NNM iSPI NETのポート

以下の表に、NNM iSPI NET診断サーバーが使用するポートを示します。NNM iSPI NET診断サーバーによってHP Operations Orchestration (HP OO)がインストールされます。詳細については、『HP Operations Orchestration管理者ガイド』を参照してください。

NNM iSPI NET診断サーバーで使用されるポート

ポート	タイプ	名前	目的	設定の変更
3306	TCP	MySQLデータベースポート	MySQLデータベースへのアクセスを提供します。	変更できません。
8080	TCP	jetty httpポート	Web UIとWebサービスで使用される、デフォルトのHTTPポート。	インストール後の変更はサポートされていません。
8443	TCP	jetty SSL/httpsポート	Web UIとWebサービスで使用される、デフォルトのHTTPSポート。	インストール後の変更はサポートされていません。
9004	TCP	HP OO RASポート	HP OOリモートアクションサービスへのアクセスを提供します。	変更できません。

設定問題に関するトラブルシューティング

このセクションでは、一般的な問題とその対処法をいくつか説明します。

NNMiが、SNMPデータおよびMIB文字列を正しく解釈して表示しないことがある

症状

これは、NNMiがどの文字セットを使用してこのデータを解釈するのかを認識しないことがあることが原因です。その結果、NNMiは、sysDescription、sysContact、その他のデータなど、一部のSNMPトラップからの文字化けした文字列およびその他のoctetstringデータを表示します。

解決方法

正しい文字セットを使用してこのデータを解釈することで解決できます。

不適切な文字セットを使用しているために、SNMPトラップおよびその他のoctetstringデータが文字化けしたテキストで表示されてしまう場合は、以下の手順を実行してください。

1. 以下のファイルを編集します。
   - Windowsの場合: `%NNM_PROPS%\nms-jboss.properties`
   - Linuxの場合: `$NNM_PROPS/nms-jboss.properties`
2. 次の文字列で始まる行からコメント (`#!`文字) を削除します。

   `#!com.hp.nnm.sourceEncoding=`
3. `nms-jboss.properties`ファイルの例を使用し、ご使用の環境で現在サポートされているソースエンコーディングをカンマで区切ったリストに`com.hp.nnm.sourceEncoding` JVMプロパティを設定します。この例は、Shift_JIS、EUC_JP、UTF-8、ISO-8859-1の文字セットの組み合わせを示します。
4. 変更を保存します。
5. NNMi管理サーバーを再起動します。

   NNMi管理サーバーで`ovstop`コマンドを実行します。

   NNMi管理サーバーで`ovstart`コマンドを実行します。

   注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動

> が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

6. 変更内容をテストするには、疑わしいトラップをNNMiに再送信し、文字化け表示の問題が発生しないことを確認します。

バイナリデータ、または何らかの理由で解釈できないデータが文字化けテキストに含まれる場合は、以下の手順を実行し、16進数形式で文字列を表示するようにNNMiを設定します。

1. 以下のファイルを開きます。
   - Windowsの場合: `%NNMDATADIR%\shared\nnm\conf\nnmvbnosrcenc.conf`
   - Linuxの場合: `$NNMDATADIR/shared/nnm/conf/nnmvbnosrcenc.conf`
2. NNMiが文字化けした形式で表示するトラップOID、varbind OID値の組み合わせを追加します。バイナリデータなど、NNMiにデコードさせないvarbind値からの組み合わせも追加します。nnmvbnosrcenc.confファイルの例をテンプレートとして使用し、組み合わせを設定します。これは、NNMiに16進値を使用してインシデントフォームのカスタムインシデント属性値を表示するように指示します。
3. 変更を保存します。
4. NNMi管理サーバーを再起動します。

   NNMi管理サーバーでovstopコマンドを実行します。

   NNMi管理サーバーでovstartコマンドを実行します。

   > **注:** 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、ovstopおよびovstartコマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

5. 変更内容をテストし、この変更によって、以前文字化けしていた文字列が16進数で表示されるようになったことを確認します。

ESXiサーバーとノードではなく、LinuxサーバーがNNMiマップに表示される

症状

Net-SNMPエージェントが有効になっているLinuxサーバーでVMWAREが導入されました。

解決方法

NNMiによってESXiサーバーを検出して表示する場合は、ESXiサーバーとノードのベアーメタルインストールを完了する必要があります。詳細については、http://www.vmware.com.を参照してください。

ESXiデバイスではなく、[SNMPなし]がNNMiマップに表示される

解決方法

NNMiがESXiサーバーとノードを検出してマッピングするためには、ESXi SNMPエージェントをインストールして有効にする必要があります。ESXi SNMPエージェントをアンインストールしたか無効にした可能性があります。

解決方法

これを解決するには、ESXi SNMPエージェントをインストールするか有効にします。詳細については、http://www.vmware.com.を参照してください。

ESXiサーバー、およびESXiサーバーで動作する仮想マシンと仮想サーバーがNNMiマップに表示される

症状

NNMiでは、すべてのシステムが雲のシンボルで接続されて表示されます。これは、仮想マシンと仮想サーバーを含むESXiサーバーをNNMiマップに表示しない場合に限り問題となります。

解決方法

仮想マシンと仮想サーバーを含むESXiサーバーがNNMiに表示されないようにするには、以下の手順を実行します。

1. NNMiコンソールを開きます。
2. 削除するノードを表示しているトポロジマップに移動し、ESXiサーバー、仮想マシン、および仮想サーバーを表すノードを削除します。
3. [**設定**] ワークスペースの [**検出の設定**] をクリックします。
4. [**自動検出ルール**] タブをクリックします。
5. 新しい自動検出ルールを作成します。

6. 比較的に小さい数値を [順序] フィールドに入力し、このルールの優先順位を高くします。[含められたノードを検出する] チェックボックスがオフになっていることを確認します。
7. このルールの新しいIPアドレス範囲を追加します。
8. ESXiサーバー、仮想マシン、および仮想サーバーを表すノードの場合は、このノードのそれぞれのIPアドレスまたはIPアドレス範囲を追加し、[範囲のタイプ] を [ルールにより無視された] ではなく [ルールにより含める] に変更します。
9. [**保存して閉じる**] を3回クリックして作業を保存します。

注: ESXiサーバーとノードではなく、LinuxサーバーがNNMiマップに表示される。

NNMiが、ホスト (NNMi管理サーバー) と一致しないライセンスキーに関するメッセージを表示する。

症状

これは、NNMi管理サーバーのIPアドレスと一致しないIPアドレスで作成されたNNMiライセンスキーがインストールされた後に発生します。

解決方法

以下の手順で無効なライセンスキーを削除することで解決できます。

1. コマンドプロンプトで以下のコマンドを入力し、Autopassユーザーインタフェースを開きます。

   ```
   nnmlicense.ovpl NNM -gui
   ```

2. [Autopass] ウィンドウの左側で [**ライセンスキーの削除**] をクリックします。
3. 無効なライセンスキーを選択します。
4. [**削除**] をクリックします。

NNMを影響される製品で置き換えて、その他の影響されるNNMi製品統合に対して手順1から手順4を繰り返します。たとえばNNMi SPIネットワークエンジニアリングツールセットソフトウェアに関連するライセンスを操作するには、以下のコマンドを使用してAutopassユーザーインタフェースを開きます。

nnmlicense.ovpl iSPI-NET -gui

ライセンスの詳細については、「NNMiのライセンス」(312ページ) を参照してください。

PAgP (ポート集約プロトコル) を使用している一部のCiscoデバイスの場合、ポート集約の一部となっているリンクが停止すると、NNMiでそのデバイスのポートがポート集約の一部ではなくなったとみなされる可能性がある

症状

これにより、ポート集約のパフォーマンス低下状態がNNMiからレポートされなくなる場合がある。

解決方法

NNMi 9.0xパッチ4から、PAgPを使用するCiscoデバイスをNNMiで管理しやすくする機能が備わっています。このNNMiの機能を設定して、停止中のインタフェースがポート集約の一部としてまだ設定されているかどうかを判断できます。この機能を有効にするには、以下の手順を実行します。

1. 以下のファイルを開きます。
   - Windowsの場合: `%NNM_PROPS%\nms-disco.properties`
   - Linuxの場合: `$NNM_PROPS/nms-disco.properties`
2. `enablePagpOperDownHeuristic`エントリを特定します。このエントリは以下の行のように記述されています。

   #!com.hp.ov.nms.disco.enablePagpOperDownHeuristic=false

   `enablePagpOperDownHeuristic`を有効にするには、以下のように行を変更します。

   com.hp.ov.nms.disco.enablePagpOperDownHeuristic=true

   > **注:** 行の始めにある**#!**文字を必ず削除してください。

3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

> **注:** 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

NNMiでOracleデータベースを使用している。大きいノードグループを設定すると、ノードグループマップの生成中にエラーが発生する

症状

これは、NNMiを以下のように設定した場合に生じる可能性があります。

- NNMiでOracleデータベースを使用している。
- 子ノードグループを含む最上位レベルのノードグループを作成している。
- いずれかの子ノードグループに1000以上のメンバーが含まれている。
- これらのノードグループの [**ノードグループマップの設定**] > [**接続**] > [**ノードグループ接続**] セクションで、以下のいずれか、あるいは両方を選択している。
  - **ノードからノードグループへ**
  - **ノードグループからノードグループへ**

解決方法

これを修正するには、子ノードグループに含まれるメンバーを1000未満にするか、これらのノードグループの [**ノードグループマップの設定**] > [**接続**] > [**ノードグループ接続**] セクションで、[**ノードからノードグループへ**] または [**ノードグループからノードグループへ**] のいずれか、あるいは両方を選択しないようにします。

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Filesライブラリを誤ってNNMi管理サーバーから削除してしまった

症状

NNMiコンソールの [**SNMPv3設定**] フォームでは、SNMPv3デバイスとのやり取りに使用するプライバシプロトコルを指定できます。Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy FilesライブラリがNNMi管理サーバーにインストールされている場合に限り、AES-192、AES-256、TripleDESのプロトコルを使用できます。

解決方法

Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Filesライブラリを誤って削除してしまい、SNMPv3通信に使用するAES-192、AES-256、およびTripleDESのプライバシプロトコルを

NNMiで使用できるようにする必要がある場合、以下の手順を実行します。

1. Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Filesライブラリを、Java開発者用のOracle Technology Network Webサイトからダウンロードします。
2. ダウンロードしたパッケージを展開してから、両方のJARファイル (`local_policy.jar`および`US_export_policy.jar`) を以下の場所にコピーします。
   - Windowsの場合: `%NnmInstallDir%\nonOV\jdk\nnm\jre\lib\security`
   - Linuxの場合: `$NnmInstallDir/nonOV/jdk/nnm/jre/lib/security`
3. NNMi管理サーバーを再起動します。
   a. NNMi管理サーバーで**ovstop**コマンドを実行します。
   b. NNMi管理サーバーで**ovstart**コマンドを実行します。

注: 高可用性 (HA) でファイルの変更を行う場合は、クラスターの両方のノードに変更を加える必要があります。HA設定を使用するNNMiでは、変更でNNMi管理サーバーの停止と再起動が必要な場合、`ovstop`および`ovstart`コマンドを実行する前にノードをメンテナンスモードにする必要があります。詳細については、「メンテナンスモード」(202ページ)を参照してください。

用語集

A

account
NNMiでは、ユーザーまたはユーザーグループがNNMiにアクセスする方法を提供します。NNMiユーザーアカウントはNNMiコンソールにセットアップされ、事前定義されたユーザーロールを実装します。システムアカウントおよびユーザーロールを参照してください。

ARPキャッシュ
ARP (アドレス解決プロトコル) キャッシュは、データリンク層 (OSIレイヤー2) アドレスをネットワーク層 (OSIレイヤー3) アドレスにマップするオペレーティングシステムテーブルです。データリンク層アドレスは通常はMACアドレスですが、ネットワーク層アドレスは通常はIPアドレスです。ルールベース検出では、NNMiは、検出されたノードでARPキャッシュエントリ (ならびにほかのテクニック) を使用して、現在の検出ルールに照らしてチェックできる追加ノードを見つけます。

C

Causal Engine
因果関係ベースの方法を使用して、根本原因解析 (RCA) をネットワーク現象に適用するNNMiテクノロジ。Causal Engine RCAをトリガーするのは、状態ポーリング、SNMPトラップ、特定のインシデントの結果として検出された変更など、特定のオカレンスです。Causal EngineはRCAを使用して管理対象オブジェクトのステータスを調べ、これらオブジェクトに関する結果を明確化し、根本原因インシデントを生成します。

H

HA
このガイドでは、設定の一部に障害があっても中断されないサービスを提供するハードウェアおよびソフトウェアの設定のことです。高可用性 (HA) とは、コンポーネントに障害があった場合でもアプリケーションを実行し続けるよう冗長コンポーネントを備えた構成を意味します。NNMiは、市販されているいくつかのHAソリューションの1つをサポートするように設定できます。アプリケーションフェイルオーバーと比べてください。

HAリソースグループ
HP ServiceGuard、Veritas Cluster Server、Microsoft Cluster Serviceなどの最新の高可用性環境では、アプリケーションは、アプリケーション自体、その共有ファイルシステム、仮想IPアドレスのようなリソースの復合物として表わされます。リソースはHAリソースグループで構成されます。これはクラスター環境で実行中のアプリケーションを表します。

HP Network Node Manager i Software
ネットワーク管理の支援や統合のために設計されたHPのソフトウェア商品です (短縮形はNNMi)。ネットワークノードの継続検出、イベントの監視、ネットワーク障害管理といった機能を備えています。主にNNMiコンソールからアクセスします。

I

ICMP
中核的なインターネットプロトコルスイート (TCP/IP) の1つ。ICMP pingは、状態ポーリング用のSNMPクエリーとともにNNMiで使用されます。

iSPI
Iファミリ内のスマートプラグイン。NNM iSPIは、MPLSのような特殊テクノロジ用に、またはネットワークエンジニアリングのような特定の分野用に、NNMiに機能を追加します。

L

L2
階層化通信モデルであるOpen Systems Interconnection (OSI) のデータリンク層です。データリンク層では、ネットワークの物理リンクを介してデータの伝送を行います。NNMiレイヤー2ビューは、デバイスの物理接続に関する情報を提供します。

L3
階層化通信モデルであるOpen Systems Interconnection (OSI) のネットワーク層です。ネットワーク層は、ネットワーク上の隣接するノードのアドレスの取得、データ伝送経路の選択、サービス品質などに関与します。NNMiレイヤー3ビューは、ルーティングの観点から接続に関する情報を提供します。

M

MIB
SNMPで、管理対照ネットワークに関するデータの階層的に組織化された集合。管理情報ベース内のデータオブジェクトは管理対照デバイスの特色を参照します。NNMiは、ネットワーク管理情報を収集する場合、MIBデータオブジェクト (「MIBオブジェクト」、「オブジェクト」、「MIB」と呼ばれることもあります) を使用して、管理対象ノードとの間でSNMPクエリーを出し、またはSNMPトラップを受け取ります。

N

NNM 6.x/7.xイベント
古いNNM管理ステーションからNNMiに転送されたイベント用のNNMi用語。NNMiには、転送されたイベントからNNMiが生成するインシデントを参照するためのインシデントビューがあります。

NNM iSPI
Iファミリ内のスマートプラグイン。NNM iSPIは、MPLSのような特殊テクノロジ用に、またはネットワークエンジニアリングのような特定の分野用に、NNMiに機能を追加します。

NNMi
ネットワーク管理の支援や統合のために設計されたHPのソフトウェア商品です (短縮形はNNMi)。ネットワークノードの継続検出、イベントの監視、ネットワーク障害管理といった機能を備えています。主にNNMiコンソールからアクセスします。

NNMiコンソール
NNMiユーザーインタフェース。オペレーターや管理者は、NNMiコンソールを使用してNNMiネットワーク管理タスクを実行できます。

O

OID
SNMPで、管理情報ベースデータオブジェクトを識別する数字のシーケンス。OIDは、小数点で分離された数字で構成されます。各数字は、MIB階層のそのレベルにおける特定のデータオブジェクトを表します。OIDはMIBオブジェクト名と同等の数字です。 たとえば、MIBオブジェクト名iso.org.dod.internet.mgmt.mib-2.bgp.bgpTraps.bgpEstablishedはそのOID 1.3.6.1.2.1.15.0.1と同等です。

ovstartコマンド
NNMiの管理プロセスを起動するためのコマンド。コマンドプロンプトで起動します。ovstartのリファレンスページ、またはUNIXのマニュアルページを参照してください。

ovstatusコマンド
NNMiが管理するプロセスの現在のステータスをレポートするコマンド。NNMiコンソール ([ツール] > [NNMiステータス]) またはコマンドプロンプトで起動できます。ovstatusのリファレンスページ、またはUNIXのマニュアルページを参照してください。

ovstopコマンド
NNMiの管理プロセスを停止するためのコマンド。コマンドプロンプトで起動します。ovstopのリファレンスページ、またはUNIXのマニュアルページを参照してください。

P

Pingスィープ
ICMP ECHO要求を複数のIPアドレスに送信し、応答するノードにどのアドレスが割り当てられているか調べるネットワークプローブテクニック。ルールベース検出で有効にすると、NNMiは、設定されたIPアドレス範囲でpingスィープを使用してその他のノードを検索できます。サービス拒絶攻撃にpingスィープを使用できるので、ICMP ECHO要求をブロックするネットワーク管理者もいます。

PostgreSQL
トポロジ、インシデント、設定情報のような情報を保存するためにNNMiがデフォルトで使用するオープンソースリレーショナルデータベース。NNMiでは、ほとんどのテーブルについてPostgreSQLの代わりにOracleを使用するよう設定することもできます。

R

RCA
NNMiで、根本原因解析 (RCA) とは、ネットワーク問題の原因を調べるためにNNMiが使用する問題解決方法のクラスのことです。NNMiで、根本原因とは、関連付けられた問題の現象が処理されていない場合、すぐに実施できる問題です。NNMiは、すぐに実施できる問題についてユーザーに通知する方法、および根本原因が解決されるまで二次的問題の現象をレポートしないようにする方法の2つの主要な方法で根本原因の識別を使用します。根本原因を判別すると、管理対象オブジェクトのステータス変更、または根本原因インシデントの生成、あるいはその両方が行われることがあります。NNMiがRCAを使用する例として、管理対象ルーターで障害が発生し、NNMi管理サーバーからみてルーターの反対側にある管理対象ノードが状態ポーリングクエリーに応答できなくなることが挙げられます。NNMiはRCAを使用し、状態ポーリング障害が二次的問題の現象であるか調べます。ルーターが根本原因インシデントであることを報告し、根本原因ルーター障害が解決されるまでダウンストリームノードで発生している

問題の現象を報告することは差し控えます。

S

SNMP
OSIモデルのアプリケーション層 (レイヤー7) で機能する簡易なプロトコル。リモートユーザーは、このプロトコルによって、ネットワーク要素の管理情報を検査または変更できます。SNMPは、管理対照ノード上のエージェントプロセッサーとネットワーク管理情報を交換するためにNNMiが使う主要なプロトコルです。NNMiは、SNMPの最も一般的なバージョンであるSNMPv1、SNMPv2c、およびSNMPv3の3つをサポートしています。

SNMPトラップ
ポーリングを使ったネットワーク管理 (SNMPエージェントから請求された応答) は、処理をできるだけ簡単にするためのSNMPの設計原則です。しかし、このプロトコルは、SNMPエージェントからSNMPマネージャープロセス (この場合、NNMi) への要請されないメッセージの通信も提供します。要請されないエージェントメッセージは、「トラップ」として知られており、内部状態の変化または障害条件に応答してSNMPエージェントが生成します。NNMiは、受信したSNMPトラップ ([SNMPトラップ] インシデントの参照ビューに表示) からインシデントを生成します。

SNMPトラップストーム
要請されない大量のSNMPエージェントメッセージ。SNMPマネージャープロセス (この場合、NNMi) を圧倒する可能性があります。nnmtrapconfig.ovplコマンドを使用してNNMiにSNMPトラップストームしきい値を設定できます。受信トラップレートが指定のしきい値レートを超えるとき、NNMiは、トラップレートが再対応レート未満に下がるまでトラップをブロックします。

sysObjectID
NNMiで、ネットワーク要素のモデルまたは種類を識別するSNMPオブジェクト識別子の専門化された用語。システムオブジェクトIDは、ネットワーク要素の管理情報ベースオブジェクトの一部です。このオブジェクトは、検出の間に個別のノードからNNMiによってクエリーされます。システムオブジェクトIDによって分類できるネットワーク要素の種類の例には、HP ProCurveスイッチファミリ、HP J8715A ProCurve Switch、HP IPFシステム用のHP SNMPエージェントがあります。他のベンダーのネットワーク要素も同じようにシステムオブジェクトIDに従って分類できます。システムオブジェクトIDの重要な使用法はNNMiデバイスのプロファイルルの定義にあります。デバイスのプロファイルルは、ネットワーク要素の種類が分かると、削減できるネットワーク要素の特徴を指定します。

ア

アクティブなクラスターノード
アプリケーションフェイルオーバーまたは高可用性設定でNNMiプロセスを現在実行しているサーバー。

アドレスのヒント
SNMP ARPキャッシュクエリー、CDP、EDP、またはその他の検出プロトコルクエリー、またはpingスィープを使用してNNMiが見つけたIPアドレス。NNMiはさらに、検出ヒントとして見つかったIPアドレスについてクエリーを実行し、結果をルールベース検出内の現在の検出ルールに照らしてチェックします。

アプリケーションフェイルオーバー
NNMiで、現在アクティブなサーバーが停止した場合に、NNMiのプロセスの制御をスタンバイサーバーに移行する追加機能 (ユーザーが設定し、jbossクラスタリングサポートを利用)。

イ

インシデント
NNMiでは、ネットワークに関連するオカレンスの通知が、NNMiコンソールインシデントビューとフォームに表示されます。NNMiには、インシデント属性に基づいてユーザーがインシデントをフィルターできるようにするいくつもの [インシデントの管理] ビューと [インシデントの参照] ビューがあります。ほとんどのインシデントビューには、NNMi (管理イベントと呼ばれることもあります) が直接生成したインシデントが表示されます。NNMiには、SNMPトラップから生成されたインシデントおよびNNM 6.x/7.xイベントから生成されたインシデントを参照するビューもあります。

インターネット制御メッセージプロトコル
中核的なインターネットプロトコルスイート (TCP/IP) の1つ。ICMP pingは、状態ポーリング用のSNMPクエリーとともにNNMiで使用されます。

インタフェース
ノードをネットワークに接続するのに使われる物理ポート。

インタフェースグループ
NNMiの主要なフィルターテクニックの1つ。ただし、グループごとに、グループまたはフィルター視覚化に設定を適用する目的で、インタフェースはグループにまとめられます。インタフェースグループは、モニタリングの設定、テーブルビューのフィルタリング、およびマップビューのカスタマイズのいずれかまたはすべてに使用できます。ノードグループも参照してください。

エ

エピソード
NNMi根本原因解析で、特定の持続時間を指すのに使用する用語。この持続時間は一次的な障害によってトリガーされ、その間、二次障害は抑制されるか、または一次的障害の下で相互に関連付けられます。

オ

オブジェクト識別子
SNMPで、管理情報ベースデータオブジェクトを識別する数字のシーケンス。OIDは、小数点で分離された数字で構成されます。各数字は、MIB階層のそのレベルにおける特定のデータオブジェクトを表します。OIDはMIBオブジェクト名と同等の数字です。 たとえば、MIBオブジェクト名 iso.org.dod.internet.mgmt.mib-2.bgp.bgpTraps.bgpEstablishedはそのOID 1.3.6.1.2.1.15.0.1と同等です。

ク

クラスター
NNMiの関係では、高可用性テクノロジまたはjbossクラスター化機能の使用によってリンクされるハードウェアおよびソフトウェアのグループ化のことです。これらは、一緒に機能して、コンポーネントに過剰負荷または障害が発生した場合、機能とデータの連続性を確実にします。クラスター内のコンピューターは一般に高速LAN経由でお互いに接続されます。クラスターは、通常、

可用性かパフォーマンス、またはその両方を向上させるために導入します。

クラスターメンバーまたはノード
NNMiの関係では、NNMi高可用性またはアプリケーションフェイルオーバーをサポートするよう設定された、または設定される予定の高可用性またはjbossクラスター内のシステム。

グ

グローバルネットワーク管理
地理的に分散している1つ以上のリージョナルマネージャーからのデータを統合する1つ以上のグローバルマネージャーを持つ、NNMiの分散型の配備です。

グローバルマネージャー
分散NNMiリージョナルマネージャーサーバーからのデータを統合する、グローバルネットワーク管理配備内のNNMi管理サーバー。グローバルマネージャーは、環境全体のトポロジおよびインシデントの統合ビューを提供します。グローバルマネージャーには、NNMi Advancedライセンスが必要です。

コ

コミュニティ文字列
SNMPエージェントでSNMPクエリーを認証するために、SNMPv1およびSNMPv2cシステムで使用されるパスワードのような仕組み。コミュニティ文字列はSNMPパケット内のクリアテキストに渡されるので、パケット傍受に対して脆くなります。SNMPv3は、認証用の強力なセキュリティメカニズムを用意します。

コンソール
NNMiユーザーインタフェース。オペレーターや管理者は、NNMiコンソールを使用してNNMiネットワーク管理タスクを実行できます。

コントローラー
NNMiアプリケーションフェイルオーバーでの、マスタークラスターの状態を持つクラスターメンバーを表すJGroups用語。JGroupsにより、クラスターのどのメンバーが最下位のIPアドレスに基づくコントローラーであるかが判別されます。

シ

シード
ネットワーク検出プロセスの開始点として機能することによって、NNMiのネットワーク検出を補助するネットワークノードのことです。たとえば、管理環境内のコアルーターなどがシードになることができます。各シードは、IPアドレスやホスト名によって識別されます。ルールベース検出が設定されていない場合、NNMiの検出プロセスは指定シードのリストベース検出に制限されます。

シード済み検出
シードのリストに基づいたプロセス。シードとして指定するノードのみに関する詳細ネットワーク情報を検出し、返します。リストに基づいた検出は、特定したクエリーとタスクのネットワークインベントリのみを保守します。ルールベース検出と比べてください。検出プロセスおよびスパイラル検出も参照してください。

システムアカウント
NNMiでは、NNMiのインストール時に使用するために備わっている特別なアカウント。NNMiシステムアカウントは、インストール

終了後は、コマンドラインのセキュリティや復旧目的のみに使用されます。ユーザーアカウントと比べてください。

システムオブジェクトID
NNMiで、ネットワーク要素のモデルまたは種類を識別するSNMPオブジェクト識別子の専門化された用語。システムオブジェクトIDは、ネットワーク要素の管理情報ベースオブジェクトの一部です。このオブジェクトは、検出の間に個別のノードからNNMiによってクエリーされます。システムオブジェクトIDによって分類できるネットワーク要素の種類の例には、HP ProCurveスイッチファミリ、HP J8715A ProCurve Switch、HP IPFシステム用のHP SNMPエージェントがあります。他のベンダーのネットワーク要素も同じようにシステムオブジェクトIDに従って分類できます。システムオブジェクトIDの重要な使用法はNNMiデバイスのプロファイルルの定義にあります。デバイスのプロファイルルは、ネットワーク要素の種類が分かると、削減できるネットワーク要素の特徴を指定します。

ス

ステータス
NNMiでは、全般的な稼働状態を示す管理対象オブジェクトの属性。ステータスは、管理対象オブジェクトの未解決結果からCausal Engineによって計算されます。状態と比べてください。

スパイラル検出
NNMiの管理するネットワークのインベントリ、コンテインメント、リレーションシップ、接続についての情報などのネットワークトポロジ情報をNNMiが常時更新する処理。検出プロセス、ルールベース検出およびリストベース検出も参照してください。

ト

トポロジ (ネットワーク)
ネットワークのノードや接続などが、通信ネットワーク上でどのように配置されているのかを示す図のことです。

トラップ
ポーリングを使ったネットワーク管理 (SNMPエージェントから請求された応答) は、処理をできるだけ簡単にするためのSNMPの設計原則です。しかし、このプロトコルは、SNMPエージェントからSNMPマネージャープロセス (この場合、NNMi) への要請されないメッセージの通信も提供します。要請されないエージェントメッセージは、「トラップ」として知られており、内部状態の変化または障害条件に応答してSNMPエージェントが生成します。NNMiは、受信したSNMPトラップ ([SNMPトラップ] インシデントの参照ビューに表示) からインシデントを生成します。

ノ

ノード
ネットワーク関係で、ネットワークに接続されているコンピューターシステムやデバイス (プリンター、ルーター、ブリッジなど) のことです。SNMPクエリーに応答できるノードは最も包括的な情報をNNMiに提供しますが、NNMiは非SNMPノードの制限された管理も実行できます。

ノードグループ
NNMiの主要なフィルターテクニックの1つ。ただし、グループごとに、グループまたはフィルター視覚化に設定を適用する目的で、ノードはグループにまとめられます。ノードグループは、モニタリングの設定、テーブルビューのフィルタリング、および

マップビューのカスタマイズのいずれかまたはすべてに使用できます。インタフェースグループも参照してください。

ポ

ポート
ネットワークハードウェアの関係において、ネットワークデバイスを経由して情報の受け渡しを行うコネクターです。

ボ

ボリュームグループ
コンピューターストレージ仮想化の用語。1つの大規模ストレージエリアを形成するよう設定された1つまたは複数のディスクドライブ。NNMiがサポートするいくつかの高可用性製品は共有ファイルシステムでボリュームグループを使用します。

ユ

ユーザーアカウント
NNMiでは、ユーザーまたはユーザーグループがNNMiにアクセスする方法を提供します。NNMiユーザーアカウントはNNMiコンソールにセットアップされ、事前定義されたユーザーロールを実装します。システムアカウントおよびユーザーロールを参照してください。

ユーザーロール
NNMi管理者は、ユーザーアクセス設定の一環として、NNMiの各ユーザーアカウントに定義済みのユーザーロールを割り当てます。ユーザーロールにより、NNMiコンソールにアクセス可能なユーザーアカウント、および各ユーザーアカウントで使用可能なワークスペースとアクションが決まります。NNMiには、管理者、Webサービスクライアント、オペレーターレベル2、オペレーターレベル1、ゲストなど、プログラムによってあらかじめ定義され、変更することのできない階層型ユーザーロールがあります。ユーザーアカウントも参照してください。

リ

リージョナルマネージャー
デバイスの検出、ポーリング、およびトラップ受信を行い、情報をグローバルマネージャーに転送する、グローバルネットワーク管理配備内のNNMi管理サーバー。

リストに基づいた検出
シードのリストに基づいたプロセス。シードとして指定するノードのみに関する詳細ネットワーク情報を検出し、返します。リストに基づいた検出は、特定したクエリーとタスクのネットワークインベントリのみを保守します。ルールベース検出と比べてください。検出プロセスおよびスパイラル検出も参照してください。

ル

ルール
ルールベース検出プロセスを制限するのに使用される、ある範囲のユーザー定義IPアドレスかシステムオブジェクトID (オブジェクト識別子)、またはその両方。検出ルールは、NNMiコンソールの [自動検出ルール] の [検出の設定] 部分に設定します。ルールベース検出も参照してください。

ルールベースの検出
自動検出と呼ばれることがよくあります。NNMiは、ルールベースの検出を使用し、ユーザー指定検出ルールに従って、NNMiが

データベースに追加する必要のあるノードを探し出します。NNMiは、検出されたノードのデータ内で検出ヒントを探してから、指定の検出ルールに照らしてこれらの候補をチェックします。検出ルールは、NNMiコンソールの [自動検出ルール] の [検出の設定] 部分に設定します。リストベース検出と比べてください。

レ

レイヤー2
階層化通信モデルであるOpen Systems Interconnection (OSI) のデータリンク層です。データリンク層では、ネットワークの物理リンクを介してデータの伝送を行います。NNMiレイヤー2ビューは、デバイスの物理接続に関する情報を提供します。

レイヤー3
階層化通信モデルであるOpen Systems Interconnection (OSI) のネットワーク層です。ネットワーク層は、ネットワーク上の隣接するノードのアドレスの取得、データ伝送経路の選択、サービス品質などに関与します。NNMiレイヤー3ビューは、ルーティングの観点から接続に関する情報を提供します。

ロ

ロール
NNMi管理者は、ユーザーアクセス設定の一環として、NNMiの各ユーザーアカウントに定義済みのユーザーロールを割り当てます。ユーザーロールにより、NNMiコンソールにアクセス可能なユーザーアカウント、および各ユーザーアカウントで使用可能なワークスペースとアクションが決まります。NNMiには、管理者、Webサービスクライアント、オペレーターレベル2、オペレーターレベル1、ゲストなど、プログラムによってあらかじめ定義され、変更することのできない階層型ユーザーロールがあります。ユーザーアカウントも参照してください。

因

因果関係
あるイベント (原因) と別のイベント (影響) の間の関係を示します。イベント (影響) は最初のイベント (原因) の直接的な結果です。NNMiは、因果関係分析アルゴリズムを使用して、イベントのサイクルを分析し、ネットワーク問題を解決するソリューションを明らかにします。

仮

仮想IPアドレス
特別なネットワークハードウェアに結び付かれていないIPアドレス。現在のフェイルオーバーまたはロードバランシングのニーズに基づいて、最も該当するサーバーに中断されないネットワークトラフィックを送信するため、高可用性設定で使われます。

仮想ホスト名
仮想IPアドレスに関連付けられたホスト名。

管

管理サーバー
NNMi管理サーバーは、NNMiソフトウェアがインストールされるコンピューターシステムです。NNMiのプロセスとサービスは、NNMi管理サーバーで稼働します。(以前のNNMリビジョンはこのシステムについて「NNM管理ステーション」という用語を使用していました。)

管理情報ベース
SNMPで、管理対照ネットワークに関するデータの階層的に組織化された集合。管理情報ベース内のデータオブジェクトは管理対照デバイスの特色を参照します。NNMiは、ネットワーク管理情報を収集する場合、MIBデータオブジェクト(「MIBオブジェクト」、「オブジェクト」、「MIB」と呼ばれることもあります)を使用して、管理対象ノードとの間でSNMPクエリーを出し、またはSNMPトラップを受け取ります。

簡

簡易ネットワーク管理プロトコル (SNMP)
OSIモデルのアプリケーション層(レイヤー7)で機能する簡易なプロトコル。リモートユーザーは、このプロトコルによって、ネットワーク要素の管理情報を検査または変更できます。SNMPは、管理対照ノード上のエージェントプロセッサーとネットワーク管理情報を交換するためにNNMiが使う主要なプロトコルです。NNMiは、SNMPの最も一般的なバージョンであるSNMPv1、SNMPv2c、およびSNMPv3の3つをサポートしています。

結

結論
NNMiで、管理対象オブジェクト用にCausal Engineがステータスと根本原因インシデントを決定した方法を明らかにする、Causal Engineが生成および使用するサポート詳細。

検

検出シード
ネットワーク検出プロセスの開始点として機能することによって、NNMiのネットワーク検出を補助するネットワークノードのことです。たとえば、管理環境内のコアルーターなどがシードになることができます。各シードは、IPアドレスやホスト名によって識別されます。ルールベース検出が設定されていない場合、NNMiの検出プロセスは指定シードのリストベース検出に制限されます。

検出のヒント
SNMP ARPキャッシュクエリー、CDP、EDP、またはその他の検出プロトコルクエリー、またはpingスィープを使用してNNMiが見つけたIPアドレス。NNMiはさらに、検出ヒントとして見つかったIPアドレスについてクエリーを実行し、結果をルールベース検出内の現在の検出ルールに照らしてチェックします。

検出プロセス
NNMiが、ネットワークノードを管理下におくために、これらの情報を収集するプロセス。初期検出は、まずデバイスインベントリの情報を収集し、次にネットワーク接続情報を収集するという2つのフェーズのプロセスで実行されます。最初の検出の後も検出プロセスは継続されます。つまり、リストベース検出では、シードリスト内のデバイスは、設定が変更されると更新されます。ルールベース検出では、新しいデバイスは現在の検出ルールに合致すると追加されます。検出プロセスは、NNMiコンソールまたはコマンドラインから、デバイスまたはデバイスセットについてオンデマンドで開始できます。スパイラル検出、ルール

ベース検出およびリストベース検出も参照してください。

検出ルール

ルールベース検出プロセスを制限するのに使用される、ある範囲のユーザー定義IPアドレスかシステムオブジェクトID (オブジェクト識別子)、またはその両方。検出ルールは、NNMiコンソールの [自動検出ルール] の [検出の設定] 部分に設定します。ルールベース検出も参照してください。

公

公開キー証明書

ネットワークセキュリティおよび暗号化で使用されます。デジタル署名を組み込み、公開キーと識別情報を結合するファイルです。証明書は、公開キーが個人または組織に属することの確認に使われます。NNMiはSSL証明書を使います。これにはクライアントとサーバーの通信の認証と暗号化のために、公開キーおよびプライベートキーが含まれています。

高

高可用性

このガイドでは、設定の一部に障害があっても中断されないサービスを提供するハードウェアおよびソフトウェアの設定のことです。高可用性 (HA) とは、コンポーネントに障害があった場合でもアプリケーションを実行し続けるよう冗長コンポーネントを備えた構成を意味します。NNMiは、市販されているいくつかのHAソリューションの1つをサポートするように設定できます。アプリケーションフェイルオーバーと比べてください。

根

根本原因インシデント

[相関処理特性] 属性が [根本原因] に設定されているNNMiインシデント。NNMiは、関連問題の現象が処理されていない場合、根本原因解析 (RCA) を使用して現象を解決するすぐ実施できる課題として根本原因インシデントを確定します。根本原因解析を参照してください。

根本原因解析

NNMiで、根本原因解析 (RCA) とは、ネットワーク問題の原因を調べるためにNNMiが使用する問題解決方法のクラスのことです。NNMiで、根本原因とは、関連付けられた問題の現象が処理されていない場合、すぐに実施できる問題です。NNMiは、すぐに実施できる問題についてユーザーに通知する方法、および根本原因が解決されるまで二次的問題の現象をレポートしないようにする方法の2つの主要な方法で根本原因の識別を使用します。根本原因を判別すると、管理対象オブジェクトのステータス変更、または根本原因インシデントの生成、あるいはその両方が行われることがあります。NNMiがRCAを使用する例として、管理対象ルーターで障害が発生し、NNMi管理サーバーからみてルーターの反対側にある管理対象ノードが状態ポーリングクエリーに応答できなくなることが挙げられます。NNMiはRCAを使用し、状態ポーリング障害が二次的問題の現象であるか調べます。ルーターが根本原因インシデントであることを報告し、根本原因ルーター障害が解決されるまでダウンストリームノードで発生している問題の現象を報告することは差し控えます。

自

自動検出
自動検出と呼ばれることがよくあります。NNMiは、ルールベースの検出を使用し、ユーザー指定検出ルールに従って、NNMiがデータベースに追加する必要のあるノードを探し出します。NNMiは、検出されたノードのデータ内で検出ヒントを探してから、指定の検出ルールに照らしてこれらの候補をチェックします。検出ルールは、NNMiコンソールの [自動検出ルール] の [検出の設定] 部分に設定します。リストベース検出と比べてください。

障

障害ポーリング
主要なNNMiモニタリングアクティビティ。このアクティビティでは、NNMiは、管理対象の各オブジェクトの状態を調べるために、管理対象インタフェース、IPアドレス、SNMPエージェントすべてに関し、ステータスMIBのSNMP読み取り専用クエリーかICMP ping、またはその両方を発行します。ユーザーは、NNMiコンソールの [設定] ワークスペースの [モニタリングの設定] で、さまざまなインタフェースグループ、ノードグループ、ノードすべてについて実行された障害ポーリングの種類をカスタマイズできます。障害ポーリングは状態ポーリングのサブセットです。

状

状態
NNMiでは、一般的に、MIB II ifAdminStatus、MIB II ifOperStatus、パフォーマンス、または可用性に関連する自己報告された管理対象オブジェクト応答について、「状態」という用語を使用します。ステータスと比べてください。

状態ポーリング
NNMiのState Pollerが実行する指令された監視。障害、パフォーマンス、コンポーネント稼働状態、管理対象オブジェクトの可用性データを取得するためにICMP pingとSNMPクエリーを使います。障害ポーリングも参照してください。

組

組み込みデータベース
NNMiに組み込まれたデータベース。NNMiは、ほとんどのテーブルについて、組み込みデータベースの代わりに外部のOracleデータベースを使うよう設定することもできます。PostgreSQLも参照してください。

未

未接続インタフェース
NNMiの観点から、NNMiが検出したほかのデバイスに接続されていないインタフェースのこと。デフォルトでは、NNMiがモニタリングする未接続インタフェースはIPアドレスのあるもののみであり、[ルーター] ノードグループのノードに含まれます。

領

領域
NNMiにおいて、タイムアウト値やアクセス資格認定のような通信設定を行うためにグループにまとめられたデバイスです。

論

論理ボリューム

個別のファイルシステムまたはデバイススワップ空間として使用できるボリュームグループ内の任意のサイズの容量を指すコンピューターストレージ仮想化の用語。NNMiがサポートするいくつかの高可用性製品は共有ファイルシステムで論理ボリュームを使用します。

ドキュメントのフィードバックを送信

このドキュメントに関するご意見については、電子メールでドキュメントチームまでご連絡ください。このシステムで電子メールクライアントが設定されていれば、このリンクをクリックすることで、以下の情報が件名に記入された電子メールウィンドウが開きます。

デプロイメントリファレンスに関するフィードバック (Network Node Manager i Software 10.10)

電子メールの本文にご意見、ご感想を記入の上、[送信] をクリックしてください。

電子メールクライアントが利用できない場合は、上記の情報をコピーしてWebメールクライアントの新規メッセージに貼り付け、network-management-doc-feedback@hpe.com にお送りください。

フィードバックをお寄せください