zpe

nodegrid

User Guide

ZPE Systems, Inc.

4.1

ユーザーガイド

Nodegrid Serial Console™

Nodegrid Services Router™

Nodegrid Gate SR™

Nodegrid Bold SR™

Nodegrid Link SR™

Nodegrid Manager™

この文書はバージョン 4.1.x に対応しています。

米国の通知

警告: コンプライアンスを担当する当事者によって明示的に承認されていない本ユニットの変更または改造によって、機器を操作するユーザーの権限が無効になる場合があります。

注:この機器は、FCC規則のPart15に従い、クラスAのデジタルデバイスの制限に準拠していることがテストされ、確認されています。これらの制限は、機器が商用環境で動作する場合に、有害な干渉から合理的に保護するように設計されています。この装置は、無線周波数エネルギーを生成、使用、放射することが可能です。取扱説明書通りに設置/使用しないと、無線通信に有害な干渉を引き起こす可能性があります。住宅地でのこの機器の操作は、有害な干渉を引き起こす可能性があり、その場合、ユーザーは、自己費用で干渉を修正する必要があります。

カナダの通知

> このクラスAのデジタル装置は、カナダのICES-003に準拠しています。
> Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

欧州連合の通知

> これは、クラスAの製品です。この機器は、国内環境で無線干渉を引き起こす場合があります。このような場合、ユーザーは適切な措置を講じる必要があります。

日本の通知

> この装置は、クラスA情報技術装置です。 この装置を 家庭環境で使用すると電波妨害を引き起こすことが あります。
> この場合には使用者が適切な対策を講ずるよう要求されることがあります。VCCI - A

目次

はじめに

Nodegrid 4.1 ユーザーマニュアルは、Nodegrid Platformバージョン4.1、および Nodegrid Serial Console シリーズ、Nodegrid Services Router、Nodegrid Gate SR、Nodegrid Bold SR、Nodegrid Link SRなどのサポートユニットについて記載しています。

製品概要

Nodegrid Serial Console

Nodegrid Serial Console製品ラインは、サーバ、ネットワークルータとスイッチ、ストレージ、PDU、UPS、およびシリアルポートを持つその他のデバイスを含むシリアルポート接続を介して接続されたデバイスを統合・管理します。

Nodegrid Serial Console - S シリーズ

*Nodegrid Serial Console (Sシリーズ) は、あらゆるモダン/レガシー混合環境に適合するように作られています。*自動センシングポートを使用すると、ストレート ケーブルやレガシーアダプターを使用する場合でも、あらゆる環境でSシリーズの Console Server を使用できます。

- 自動スイッチ(CiscoまたはレガシーピンアウトI)
- 16/32/48 シリアルポート
- 追加の USB ポート
- 工場出荷時にアップグレードが可能なCPUとRAM
- 1U 19"ラック標準ユニット
- シングル AC、デュアル AC、デュアル DC

ハードウェアの仕様

アイテム	説明
CPU	Intel x86_64 デュアルコア CPU
メモリとストレージ	4 GB の DDR3 DRAM、32 GB mSATA SSD
インターフェース	2 GB (10/100/1000BT) RJ45 または 2 SFP+ ファイバ インターフェースのイーサネットインターフェースは、1 GB/2.5GB/10GB モジュールと互換性があります 16、32、48 RS-232 シリアルポート、RJ45 @ 230.400 bps 最大/ポート。 1 RJ45 RS-232 Serial Consoleポート 1 USB 3.0 ホスト、1 USB 2.0 ホスト、および 12 USB 2.0 ホスト、タイプ A コネクタ 1 HDMI
電源	シングル/デュアル AC 100-240 VAC、50/60 Hz デュアルDC: 40-63 VDC 消費電力 45 W 標準
物理的	フロント-リア 取り付けブラケット サイズ(L x W x H): 443 x 312 x 43 mm (17.4 x 12.3 x 1.7 インチ)、1U 重量: 4.9 kg (10.8 lb)、オプションに応じて フロントツーバックまたはバックツーフロントファン (交換可能)
環境的	操作：0°C ～ 50°C (32 ～ 122°F)、5-95% RH、結露なし ストレージ: -20°C ～ 67° C (-4 ～ 153°F)、10-90% RH、結露なし

インターフェースフロント

ポート	説明
HDMI	HDMIインターフェース
USB	USB 2.0ポート
PWR	電源LED 緑: · オン - 通常、 · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし
RST	リセットボタン: <3s システムリセット、 >10s 設定 工場出荷時へのリセットとシステムリセット
FAN	ファン
USB	1 x USB 2.0 ポート、12 x USB 1.1 ポート

インターフェースバック

ポート	説明
電源	シングルまたはデュアル電源ソケット
シリアル	シリアル インターフェース · オレンジ LED - DCD/DTR - オン: ポートオープン および/または ケーブル接続済み、オフ: 準備ができていません ·緑 LED - RX/TX - 点滅: データアクティビティ、オフ: アクティビティなし
	ネットワークインターフェイス 銅: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 -1000BaseT リンク速度

ETH0/SFP0	·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害 SFP 1Gb/10Gb: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH1/SFP1	ネットワークインターフェイス 銅: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 -1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害 SFP 1Gb/10Gb: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
コンソール	コンソール MGMT インターフェース ·オレンジ LED 電源障害用 - 点滅: 電源の故障/オフ (デュアル電源モデル用)、オフ: 通常 ·緑 LED システムアクティビティ - 点滅: 通常、オフまたはオン: アクティビティなし
USB	1×USB 3.0

Nodegrid Serial Console - Rシリーズ

*Nodegrid Serial Console (Rシリーズ)*は、Cisco、Arista、Dell、Palo Alto Networks、Juniperなどの主要なハードウェア環境に適合するように作られています。Rシリーズ Serial Consoleは、既に構築された標準ラックのアップグレードやレトロフィットに最適です。

- Cisco ピンアウト デバイスの場合
- 16/32/48/96 シリアルポート
- 1U 19"ラック標準ユニット
- シングル AC、デュアル AC、およびデュアル DC

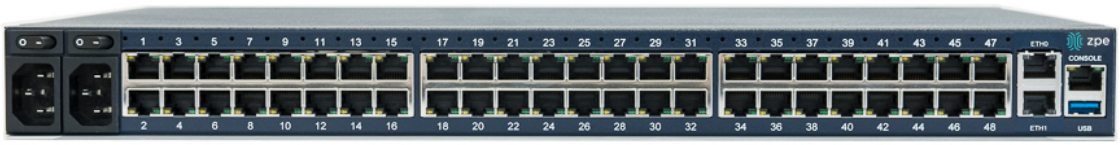
ハードウェアの仕様

アイテム	説明
CPU	Intel Atom x86_64 デュアルコア @ 1.75 GHz CPU
メモリとストレージ	4 GB の DDR3 DRAM、32 GB mSATA SSD
インターフェース	2 GB (10/100/1000BT) RJ45 または 2 SFP+ ファイバ インターフェースのイーサネットインターフェースは、1 GB/2.5GB/10GB モジュールと互換性があります 16、32、48、96 RS-232 シリアルポート、RJ45 @ 230.400 bps 最大/ポート。 1 RJ45 RS-232 Serial Consoleポート 1 USB 3.0 ホストおよび 2 USB 2.0 タイプ A コネクタのホスト 1 HDMI
電源	シングル/デュアル AC 100-240 VAC、50/60 Hz デュアルDC: 40-63 VDC 電力消費 45 W (96 ポート)
物理的	フロント-リア 取り付けブラケット サイズ(L x W x H): 443 x 312 x 43 mm (17.4 x 12.3 x 1.7 インチ)、1U 重量: オプションに応じて 4.9 kg (10.8 lb)
環境的	操作： 0°C ～ 50° C (32°F ～ 122°F)、湿度: 5-95% (結露なきこと) ストレージ: 20°C ～ 67°C (-4°F ～ 153°F)、湿度: 10-90% （結露なきこと）

インターフェースフロント

ポート	説明
HDMI	HDMIインターフェース
USB	2 x USB 2.0 ポート
PWR	電源LED 緑: · オン - 通常、 · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし
RST	リセットボタン: <3s システムリセット、 >10s 設定 工場出荷時へのリセットとシステムリセット

インターフェースバック



ポート	説明
電源	シングルまたはデュアル電源ソケット
シリアル	シリアル インターフェース ·オレンジ LED - DCD/DTR - オン: ポートオープン および/または ケーブル接続済み、オフ: 準備ができていません ·緑 LED - RX/TX - 点滅: データアクティビティ、オフ: アクティビティなし
ETH0/SFP0	ネットワークインターフェイス 銅: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 -1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害

	SFP 1Gb/10Gb: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH1/SFP1	ネットワークインターフェイス 銅: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 -1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害 SFP 1Gb/10Gb: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
コンソール	コンソール MGMT インターフェース ·オレンジ LED - 電源障害用- 点滅: 電源の故障/オフ (デュアル電源モデル用)、オフ: 通常 ·緑 LED - システムアクティビティ - 点滅: 通常、オフまたはオン: アクティビティなし
USB	USB 3.0

Nodegrid Serial Console - Tシリーズ

NODEGRID SERIAL CONSOLE (T シリーズ) は、レガシーデバイスを使用した環境に適合するように作られており、レガシーコンソールサーバと直接交換できます。

- レガシーデバイスの場合
- 16/32/48/96 シリアルポート
- 1U 19"標準ユニット
- シングル AC、デュアル AC、およびデュアル DC

ハードウェアの仕様

アイテム	説明
CPU	Intel Atom x86_64 デュアルコア @ 1.75 GHz CPU
メモリとストレージ	4 GB の DDR3 DRAM、32 GB mSATA SSD
インターフェース	2 GB (10/100/1000BT) RJ45 または 2 SFP+ ファイバ インターフェースのイーサネットインターフェースは、1 GB/2.5GB/10GB モジュールと互換性があります 16、32、48、96 RS-232 シリアルポート、RJ45 @ 230.400 bps 最大/ポート。 1 RJ45 RS-232 Serial Consoleポート 1 USB 3.0 ホストおよび 2 USB 2.0 タイプ A コネクタのホスト 1 HDMI
電源	シングル/デュアル AC 100-240 VAC、50/60 Hz デュアルDC: 40-63 VDC 電力消費 45 W (96 ポート)
物理的	フロント-リア 取り付けブラケット サイズ(L x W x H): 443 x 312 x 43 mm (17.4 x 12.3 x 1.7 インチ)、1U 重量: オプションに応じて 4.9 kg (10.8 lb)
環境的	操作：0 ～ 50° C (32 ～ 122°F)、5-95% RH、結露なし ストレージ: -20 ～ 67° C (-4 ～ 153°F)、10-90% RH、結露なし

インターフェースフロント

ポート	説明
HDMI	HDMIインターフェース
USB	2 x USB 2.0 ポート
PWR	電源LED 緑: · オン - 通常、 · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし
RST	リセットボタン: <3s システムリセット、 >10s 設定 工場出荷時へのリセットとシステムリセット

インターフェースバック

ポート	説明
電源	シングルまたはデュアル電源ソケット
シリアル	シリアル インターフェース ·オレンジ LED - DCD/DTR - オン: ポートオープン および/または ケーブル接続済み、オフ: 準備ができていません ·緑 LED - RX/TX - 点滅: データアクティビティ、オフ: アクティビティなし
ETH0/SFP0	ネットワークインターフェイス 銅: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 -1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害 SFP 1Gb/10Gb: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH1/SFP1	ネットワークインターフェイス 銅: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 -1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害 SFP 1Gb/10Gb: ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
コンソール	コンソール MGMT インターフェース ·オレンジ LED - 電源障害 - 点滅: 電源の故障/オフ (デュアル電源モデル用)、オフ: 通常 ·緑 LED - システムアクティビティ - 点滅: 通常、オフまたはオン: アクティビティなし
USB	USB 3.0

Nodegrid Services Router ファミリ

Nodegrid Services Router は、ソフトウェア定義ネットワーキング (SDN)、帯域外 (OOB) 管理、DevOps、携帯電話のフェールオーバー、ドッカー、SD-WAN、リモート/ブランチ オフィス、小売店舗、およびネットワーク機能仮想化 (NFV) 機能用に設計されたプラットフォーム機器です。

Nodegrid Services Router

*NODEGRID Services Router*は、ソフトウェア定義ネットワーキング(SDN)、帯域外(OOB)管理、DevOps、携帯電話のフェールオーバー、ドッカー、SD-WAN、リモート/ブランチオフィス、小売店舗、ネットワーク機能仮想化 (NFV) 機能用に設計されたモジュラーオープンプラットフォーム機器です。

- オープンフレームワーク、Modular Services Router
- プラグ式拡張モジュール - 5スロット付き
- GbE、シリアル、SFP+ 10GbE、PoE+、USB、M.2/SATA+アンテナ、ストレージ、追加コンピューティング用モジュール
- 1U 19"標準ユニット
- コントロール プレーンとデータ プレーンの分離

ハードウェアの仕様

アイテム	説明
CPU	Intel Multi-core x86_64 CPU
メモリとストレージ	8 GB DDR4 DRAM、32 GB mSATA SSD (工場出荷時のアップグレード可能)
インターフェース	2 SFP+ イーサネット 2 GB イーサネット 1 RJ45 RS-232 Serial Consoleポート 1 USB 3.0 1 USB 2.0 1 HDMI
電源	シングル/デュアル AC 100-240 VAC、50/60 Hz デュアルDC: 36-75 VDC 消費電力 90W 標準
物理的	フロント-リア 取り付けブラケット サイズ(L x W x H): 438 x 332 x 43mm (17.2 x 13.1 x 1.7 インチ)、1U 重量: オプションに応じて、4.9 kg (10.8 lb) エアエキゾーストまたはエアインテイクファン (交換可能)
環境的	操作： 0°C ～ 45°C (32°F ～ 113°F)、5-95% RH、結露なし ストレージ: -20°C ～ 67°C (-4°F ～ 153°F)、10-90% RH、結露なし

インターフェースフロント

ポート	説明
スロット1	モジュール用スロット
スロット2	モジュール用スロット
スロット3	モジュール用スロット
SFP+ 0	ネットワーク インターフェース ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンク/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
SFP+ 1	ネットワーク インターフェース ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンク/ケーブル切断/イーサネット障害 ·右/緑 - 10Gb リンク速度 ·右/オレンジ - 1Gb リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH0	ネットワーク インターフェース ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンク/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH1	ネットワーク インターフェース ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害

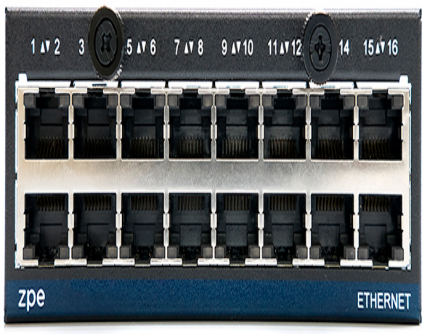
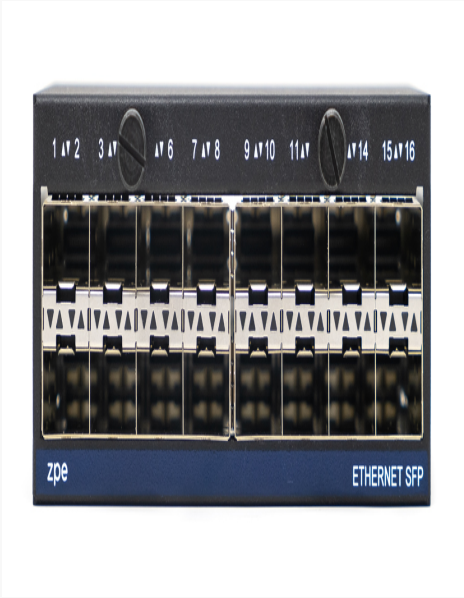
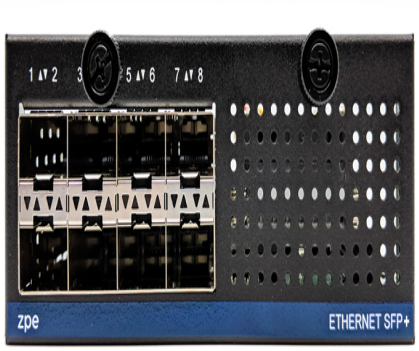
コンソール	コンソール MGMT インターフェース ·オレンジ LED - 電源障害用 - 点滅: 電源の故障/オフ (デュアル電源モデル用)、オフ: 通常 ·緑 LED - システムアクティビティ - 点滅: 通常、オフまたはオン: アクティビティなし
USB	USB 3.0
RST	リセットボタン: <3s システムリセット >10s 設定 工場出荷時へのリセットとシステムリセット

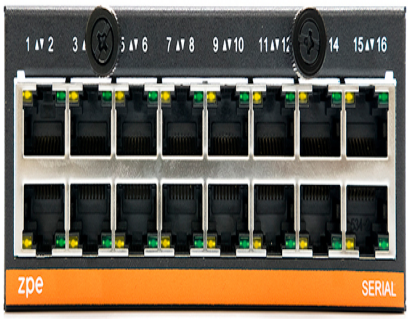
インターフェースバック

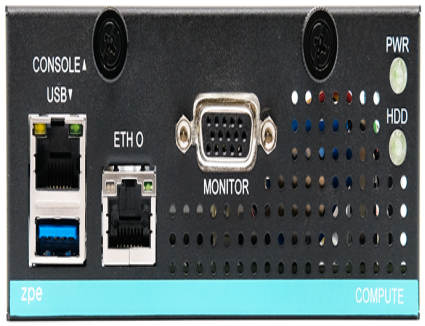
ポート	説明
スロット4	モジュール用スロット (モデルによって異なります)
スロット5	モジュール用スロット (モデルによって異なります)
USB	2 x USB 2.0 ポート
HDMI	HDMIインターフェース
PWR	電源LED 緑: · オン - 通常、 · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし
FAN	ファン
電源ソケット	デュアル電源ソケット
電源	シングルまたはデュアル電源ソケット

Nodegrid Services Router 拡張モジュール

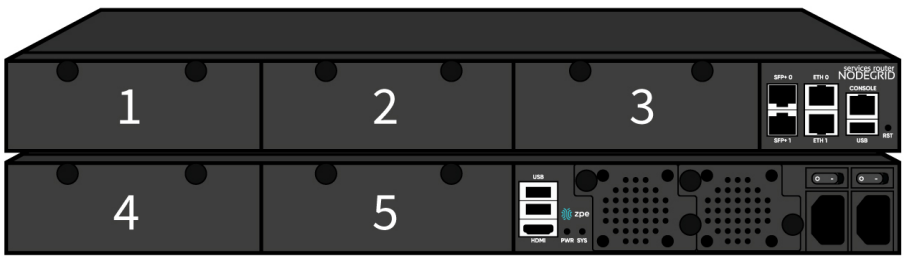
Nodegrid Services Router には、機能拡張のために高い柔軟性を提供する、モジュール用スロットが最大 5 つ備わります。

モジュール	画像	仕様
16 ポート 1GbE		1000BASE-T Cat5e 以上
16 ポート SFP 1GbE		すべての SFP モジュールをサポート
8 ポート SFP+ 10GbE		すべての SFP+ モジュールをサポート

8 ポート PoE+		ポートあたりの最大電力量は 25.5W 合計最大150W PoE+ 利用可能 設定可能な電力バジェット
16 ポート シリアル		RJ45 シリアルロールポート最大 230.400 bps
16 ポート USB		USB 2.0 インターフェース タイプ A
M.2 携帯電話 +アンテナ		最大 2x 4G/LTE モデム用

M.2 SATA		最大 2x mSATA ストレージ モジュール用
ストレージ		2.5" SATA (HDD/SDD) ストレージ用
演算		演算モジュール (カード上のサーバ) は、独立した演算機能を提供します。

拡張モジュール互換性チャート



拡張カード	スロット1	スロット2	スロット3	Slot4	Slot5
16 ポート GbE イーサネット	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
16 ポート SFP	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
16ポートシリアル	✔	✔	✔	✔	✔
16ポートUSB	✔	✔	✔	✔	✔
M.2 携帯電話 / Wi-Fi	✔	✔	✔	✔	✔
8 ポート SFP+	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
8 ポート POE+	✔	✔	✔	–	–
演算	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
ストレージ*	–	–	–	✔	✔
M.2 SATA*	–	–	–	✔	✔

> 注:
>
> (*) Nodegrid Services Router は、最大2台の SATA ドライブをサポートしており、2 枚のストレージカードまたは 1 枚のM.2 SATAカードに分割できます。
>
> (**) **セキュア絶縁モード**では、カードを通常のスロットに配置する場合と同様に管理できますが、ネットワークトラフィックは他のすべてのスロットから分離されます。

Nodegrid Gate SR

*Nodegrid Gate SR*は、ネットワークに俊敏性をもたらします。データセンターとブランチの両方に最適な、Nodegrid Gate SR は、小型フォームファクタで膨大な電力を供給し、真に堅牢で動的でセキュアなインフラストラクチャ管理ソリューションを実現します。Nodegrid Gate SRの設定と管理は、ZPE Cloudを介して行います。

- ZPE Cloudを使用し、ブランチ全体に安全で迅速、一貫性のある展開を実現
- ソフトウェア定義ネットワーキング、ネットワーク機能仮想化、ゲストOS、Kubernetes、ドッカ-機能
- 安全で集中管理されたリモートデバイスへのアクセスと制御により、MTTR、ダウンタイム、コストを最小限に抑えます。
- オープンな業界標準ハードウェアと使いやすいソフトウェアにより、サイトの信頼性が向上
- 遠隔地での迅速で簡単なセットアップのための、ゼロタッチプロビジョニング(ZTP)
- ZPE Cloud および ZPE Systems Nodegrid Manager を統合した、ベンダーニュートラルな総合管理ソリューション
- ダイレクト Linux Shell、HTML5 クロスデバイス Web アクセス、およびコマンドラインインターフェース
- 最新の64-bit Linux Kernelによる、迅速なセキュリティパッチ適用と広範なソフトウェアの可用性
- Kubernetes と Docker に最適化された、迅速で柔軟なスクリプトとアプリケーション統合
- 実用的なリアルタイムデータに基づく拡張オートメーション
- 4G/LTEモデムへのフェールオーバー
- ゲートウェイおよびマルチルーティングテーブル機能
- SSL VPN と IPsec
- リモートサイト用の予備 IP を DHCP サーバ に置き換えるか、現在のルータを完全に置き換えます
- ファイアウォール–内蔵、チェックボックスをオンにします
- 安全 – 選択可能な暗号化されたクリプトグラフィックプロトコルと、暗号スイート レベル、設定チェックサム™
- 電源制御と監視 – 誤動作が発生する前に最適でない IT デバイスの状態に関するアラートを受け取り、その問題を自動で解決します。
- オーケストレーション - Puppet、Chef、Ansible、RESTful、ZPE Cloud
- 内部カードを介してWiFiホットスポットを準備するか、PoE+ポートを介して自分のAP (アクセスポイント) を追加します
- 高い接続性を実現する高密度でフレキシブルなインターフェース

ハードウェアの仕様

アイテム	説明
CPU	Intel Multi-core x86_64 CPU
メモリとストレージ	8-32 GB の DDR4 DRAM、32 GB SATADOM SSD (アップグレード可能)
インターフェース	4 PoE+ GB (10/100/1000BT) 内蔵スイッチ付きRJ45のイーサネットインターフェイス 4 GB (10/100/1000BT) スイッチを内蔵したRJ45のイーサネットインターフェイス 8 RJ45 シリアルポート 2 SFP+ (10G) 1 Rj45 コンソールポート 2 タイプAのUSB 3.0 ホスト 2 タイプAの USB 2.0 ホスト 2 GPIO 1 デジタル出力ポート 1 リレーポート 1 Wi-Fi スロット (クライアントまたはサーバ) オプション 2 携帯電話 スロット (4G/LTE) デュアル SIM付き - オプション 1 HDMI ポート
電源	36V - 75VDCデュアル電源入力 - 冗長アクティブ/パッシブ入力、最高電圧がアクティブになります。 AC電源アダプタ (アドオン) 100-240V ～50-60Hz、1.2A、動作温度 -25°C～60°C 消費電力 45W 標準
物理的	フロント-リア取り付けブラケット サイズ (L x W x H): 241.3 x 260.4 x 44.5 mm (9.5 x 10.25 x 1.75 インチ) 重量: 0.9 kg (2 lb) 出荷重量: 3.6 kg (8.0 lb) 出荷 (L x W x H): 349 x 375 x 178 mm (13.75 x 14.75 x 7 インチ)
環境的	操作： -20°C ～ 60°C (-4 ～ 140°F)、5～95% RH、結露なし ストレージ: -20°C ～ 67°C (-4 ～ 153°F)、10-90% RH、結露なし

インターフェースフロント

定義	説明
DIO0	デジタル I/O TTL レベル 5.5V 最大 @ 64mA
DIO1	デジタル I/O TTL レベル 5.5V 最大 @ 64mA
OUT0	信号 MOSFET デジタル出力 2.5V ～ 60V @ 500mA 最大
リレー出力	NC リレーコンタクト 最大 24V @1A
コンソール	コンソール MGMT インターフェース
USB	2×USB 2.0
HDMI	モニターインターフェース
チャネル A	チャネル Aの信号強度インジケータ
チャネル B	チャネル Bの信号強度インジケータ
PWR	電源 LED 緑: · オン - ノーマル · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし
RST	リセットボタン: <3s システムリセット >10s 工場出荷時へのデフォルトセッテイングとシステムリセット
電源スイッチ	電源オン/オフスイッチ

インターフェースバック

ポート	

	説明
PWR	電源 LED 緑: · オン - ノーマル · オフ - 電源がオフです
V2- / GND / V2+	外部電源用電源コネクタ 36V - 75VDC デュアル電源入力(冗長)
V1- / GND / V1+	外部電源用電源コネクタ 36V - 75VDC デュアル電源入力(冗長)
PoE+	4 x PoE+ ネットワーク インターフェース、1～4の番号付き ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
NET	4 x ネットワーク インターフェース、5～8の番号付き ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
SFP+ 0	SFP+ ネットワーク インターフェース 0 ·左/黄 - オン: リンクアップ、オフ: リンクなし/ケーブル切断 ·右/緑 - オン: リンクアップ、点滅: アクティビティ、オフ: リンクなし/ケーブル切断
SFP+ 1	SFP+ ネットワーク インターフェース 1 ·左/黄 - オン: リンクアップ、オフ: リンク/ケーブル切断なし ·右/緑 - オン: リンクアップ、点滅: アクティビティ、オフ: リンクなし/ケーブル切断
ETH0	ネットワーク インターフェース ·左/黄色 - オン: リンクアップ、点滅: データアクティビティ, オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - オン: 1000BaseT リンク速度、オフ: 100/10BaseT リンク速度またはオフ
USB	2 x USB 3.0 ポート
シリアル	シリアル インターフェース 1～8 ·オレンジ LED - DCD/DTR - オン: ポートオープン および/または ケーブル接続済み、オフ: 準備ができていません ·緑 LED - RX/TX - 点滅: データアクティビティ、オフ: アクティビティなし

Nodegrid Bold SR

Nodegrid Bold SR は、ネットワークの EDGE でリモートデバイスと IoT デバイスを安全にアクセスおよび制御するために設計されたオープンプラットフォーム装置です。Bold SR は、携帯電話のフェールオーバー、ネットワーク機能仮想化 (NFV)、および SD-WAN を中心とするソフトウェア定義ネットワーキングをサポートします。

- 高さ 1U、コンパクトサイズ、高い処理力
- ソフトウェア定義ネットワーキングに最適
- ネットワーク機能仮想化
- 携帯電話のフェールオーバー
- Wi-Fi ホットスポットとクライアント
- マルチインターフェース

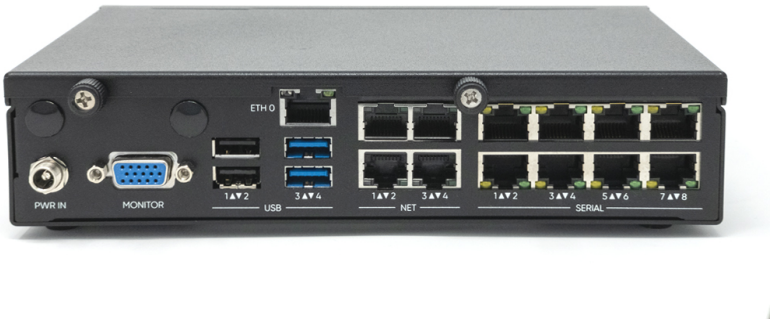
ハードウェアの仕様

アイテム	説明
CPU	Intel Multi-core x86_64 CPU
メモリとストレージ	4 GB の DDR3 DRAM、32 GB の SATADOM SSD (アップグレード可能)
インターフェース	1 GB (10/100/1000BT) RJ45 上イーサネットインターフェース 4 GB (10/100/1000BT) RJ45 上、内蔵スイッチ付きイーサネットインターフェース 8 RJ45 RS-232 シリアルポート 1 RJ45 RS-232 コンソールポート 1 USB 3.0 ホスト タイプA 2 USB 2.0 ホスト タイプA 1 Wi-Fi – オプション 2 携帯電話スロット デュアル SIM 付き– オプション 1 VGA ポート
電源	外部 100-240 VAC 経由 12 VDC、50/60 Hz アダプタ 12 VDC 外部 48 VDC アダプタ経由 標準電力消費 25 W
物理的	フロント-リア 取り付けブラケット サイズ(L x W x H): 142 x 201 x 44 mm (5.5 x 7.9 x 1.73 インチ) 重量: 1.2 kg (2.6 lb)
環境的	操作：-20°C ～ 50°C (-4°F ～ 122°F)、20～90% RH、結露なし ストレージ: -20°C ～ 67°C (-4°F ～ 153°F)、10-90% RH、結露なし

インターフェースフロント

ポート	説明
チャネル A	チャネル Aの信号強度インジケータ
チャネル B	チャネル Bの信号強度インジケータ
コンソール	コンソール MGMT インターフェース
PWR	電源LED 緑: · オン - 通常、 · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし
RST	リセットボタン: <3s システムリセット、 >10s 設定 を工場出荷時へのリセットとシステムリセット
電源スイッチ	電源オン/オフスイッチ

インターフェースバック



ポート	説明
PWR IN	外部電源用電源ソケット
モニター	VGA インターフェース
	ネットワーク インターフェース

ETH0	·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンク/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
USB	2 x USB 2.0 ポート 2 x USB 3.0 ポート
ETH1	ネットワーク インターフェース (NET) ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH2	ネットワーク インターフェース (NET) ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH3	ネットワーク インターフェース (NET) ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
ETH4	ネットワーク インターフェース (NET) ·左/緑 - 点滅: データアクティビティ、オン: 準備完了、オフ: リンクなし/ケーブル切断/イーサネット障害 ·右/緑 - 1000BaseT リンク速度 ·右/オレンジ - 100BaseT リンク速度 ·右/オフ - リンクなし/ケーブル切断/イーサネット障害
シリアル	シリアル インターフェース 1～8 ·オレンジ　LED - DCD/DTR - オン: ポートオープン および/または ケーブル接続済み、オフ: 準備ができていません ·緑 LED - RX/TX - 点滅: データアクティビティ、オフ: アクティビティなし

Nodegrid Link SR

*Nodegrid Link SR*は、ブランチネットワークに敏捷性をもたらし、コンパクトな設計に驚異的なパワーが詰め込まれています。真に堅牢でダイナミック、そして安全なインフラストラクチャ管理を可能にします。ZPE Cloud を介して Link SR を設定・管理し、ブランチ/IoT/M2M/キオスク/ATM/遠隔地を迅速で簡単に稼働させます。

- ZPE Cloudを使用し、ブランチ全体に安全で迅速、一貫性のある展開を実現
- 携帯電話ゲートウェイと Wi-Fi アクセスポイント (AP) を PoE または電源アダプタによる電源入力と組み合わせます
- ソフトウェア定義ネットワーキング、ネットワーク機能仮想化、ゲストOS、Kubernetes、ドッカ-機能
- 安全で集中管理されたリモートデバイスへのアクセスと制御により、MTTR、ダウンタイム、コストを最小限に抑えます。
- オープンな業界標準ハードウェアと使いやすいソフトウェアにより、サイトの信頼性が向上
- 遠隔地での迅速で簡単なセットアップのための、ゼロタッチプロビジョニング(ZTP)
- ZPE Cloud および ZPE Systems Nodegrid Manager を統合した、ベンダーニュートラルな総合管理ソリューション
- ダイレクト Linux Shell、HTML5 クロスデバイス Web アクセス、およびコマンドラインインターフェース
- 最新の64-bit Linux Kernelによる、迅速なセキュリティパッチ適用と広範なソフトウェアの可用性
- Kubernetes と Docker に最適化された、迅速で柔軟なスクリプトとアプリケーション統合
- 実用的なリアルタイムデータに基づく拡張オートメーション

- 4G/LTEモデムへのフェールオーバー
- リンクウェイおよびマルチルーティングテーブル機能
- SSL VPN と IPsec
- リモートサイト用の予備 IP を DHCP サーバ に置き換えるか、現在のルータを完全に置き換えます
- ファイアウォール–内蔵、チェックボックスでオンにします
- 安全で – 選択可能な暗号化されたクリプトグラフィックプロトコルと、暗号スイート レベル、設定チェックサム™
- 電源制御と監視 は、誤動作が発生する前に最適でない IT デバイスの状態に関するアラートを受け取り、その問題を自動で解決します。
- オーケストレーション - Puppet、Chef、Ansible、RESTful、ZPE Cloud
- 高い接続性を実現する高密度でフレキシブルなインターフェース

ハードウェアの仕様

アイテム	説明
CPU	Intel Multi-core x86_64 CPU
メモリとストレージ	4-8 GB の DDR3 DRAM、32 GB SATADOM SSD (アップグレード可能)
インターフェース	1 GB (10/100/1000BT) RJ45 のイーサネット インターフェース PoE 付き 1 SFP (1G) イーサネット 1 RJ45 シリアルポート 1 RJ45 コンソールポート 2 USB 2.0 ホスト タイプ A 2 GPIO 2 デジタル出力ポート 1 Wi-Fi スロット (クライアントまたはサーバスロット) オプション 1 携帯電話 スロット (4G/LTE) デュアルSIM付き - オプション 1 VGA ポート
電源	10V - 57VDC 電源入力 AC 電源アダプタ (アドオン) 100-240V～50-60Hz、1.5A PoE 電源入力 消費電力 15 W 標準
物理的	DINレールと壁取り付け可能 サイズ(L x W x H): 170 x 130 x 55 mm (6.69 x 5.11 x 2.16 インチ) 重量: 1.58 kg (2.3 lb) 出荷重量: 1.58 kg (3.5 lb) 出荷 (L x W x H): 228.6 x 342.9 x 88.9 mm (9 x 13.5 x 3.5 インチ)
環境的	操作：0 ～ 60°C (32～140° F)、5～95% RH、結露なし。 ストレージ: -20 ～ 67° C (-4 ～ 153°F)、10-90% RH、結露なし

トップ

定義	説明
BARS	信号強度インジケータ
PWR	電源 LED 緑: · オン - ノーマル · オフ - 電源がオフです
SYS	システム LED 緑: · 点滅 - 通常 · 高速点滅 - RSTボタンの受信確認 · オフまたはオン - アクティビティなし

インターフェースフロント

定義	説明
SFP 0	SFP ネットワーク インターフェース 0 ·左/黄 - 点滅: データアクティビティ、オン: リンクアップ、オフ: リンクなし/ケーブル切断 ·右/緑 - オン: 1000BaseT リンク速度、オフ: リンクなし/ケーブル切断
シリアル	シリアルインターフェース 1 ·オレンジ LED - DCD/DTR - オン: ポートオープンおよび/またはケーブル接続済み、オフ: 準備ができていません ·緑 LED - RX/TX - 点滅: データアクティビティ、オフ: アクティティナなし
コンソール	コンソール MGMT インターフェース
USB	2×USB 2.0
VGA	モニターインターフェース

インターフェースバック

定義	説明
電源スイッチ	電源オン/オフスイッチ
V1- / GND / V1+	外部電源用コネクタ 10V-57VDC 電源入力
ETH0	1 GB (10/100/1000BT) PoE を持つイーサネット · ·左/黄 - オン: リンクアップ、点滅: データアクティビティ、オフ: リンク/ケーブルなし ·右/緑 - オン: 1000BaseT リンク速度、オフ: 10/100BaseT リンク速度
DIO0	デジタル I/O TTL レベル 5.5V 最大 @ 64mA
DIO1	デジタル I/O TTL レベル 5.5V 最大 @ 64mA
OUT0	信号 MOSFET デジタル出力 2.5V ～ 60V @ 500mA 最大
OUT1	信号 MOSFET デジタル出力 2.5V ～ 60V @ 500mA 最大
RST	リセットボタン: <3sシステムリセット >10s 工場出荷時へのデフォルトとシステム リセット

Nodegrid Manager

*Nodegrid Manager *は、コンピューティング、ネットワーク、ストレージ、およびスマート電力資産を制御するための統合ソリューションを提供します。

ハードウェア要件

アイテム	説明
CPU	最低 2 x Intel Multi-core x86_64 CPU
メモリとストレージ	4 GB RAM、最低 32 GB HDD
インターフェース	最低 1 GB イーサネット インターフェース
対応ハイパーバイザー	VMWare ESX、Linux KVM、Oracle Virtualbox -- Linux OS

インストール

ハードウェアのインストール

ボックスの起動方法についての簡単な説明は、ボックス内のユニットと共に提供される付録 - クイック インストール ガイドを参照してください。

ボックスの内容

各ユニットには複数のアクセサリーが付属します。以下の表に、ボックスの内容を示します。

モデル	取り付けブラケット	電源ケーブル	ループバッククアダプター	コンソールアダプター	ネットワークケーブル	クイックスタートガイドと安全シート
Nodegrid Serial Console - Tシリーズ	はい	はい	レガシー 	Z000036	はい	はい
Nodegrid Serial Console - R シリーズ - TxxR	はい	はい	Cisco 	Z000015	はい	はい
Nodegrid Serial Console - S シリーズ - TxxS	はい	はい	レガシー/Cisco 	Z000015 Z000036	はい	はい
Nodegrid Services Router	はい	はい	レガシー/Cisco 	Z000014 Z000015	はい	はい
Nodegrid Bold Services Router	いいえ	外部電源	レガシー/Cisco 	Z000014 Z000015	はい	はい

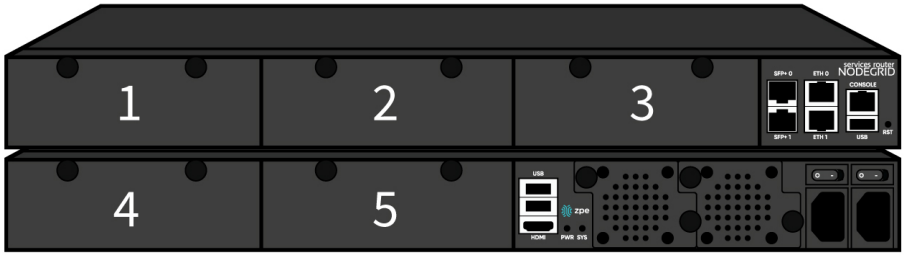
Nodegrid Services Router 用モジュールの取り付け

Nodegrid Services Router は、各種モジュールに対応しています。ユニットの電源を入れる前に取り付けてください。モジュールはコンポーネントの損傷を防ぐために、ESD 保護された環境に取り付ける必要があります。次の手順に従ってカードをインストールします。

- Nodegrid Services Router の電源をオフにします
- Nodegrid Services Router の電源がオフになっていることを確認します
- モジュールを取り付けるスロットをカバーする、ブランキングパネルのネジを外します。

- カードをボックスから出して、適切なスロットに挿入します
- 付属のネジでカードを固定します
- これで、Nodegrid Services Router をオンにできます。

> **注:**ブランキングパネルは、後で使用するために保管しておく必要があります。熱効率と安全性向上のために、未使用の各スロットはブランキングパネルで覆う必要があります。



拡張カード	スロット1	スロット2	スロット3	スロット4	スロット5
16 ポート GbE イーサネット	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
16 ポート SFP	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
16ポートシリアル	✔	✔	✔	✔	✔
16ポートUSB	✔	✔	✔	✔	✔
M.2 携帯電話 / Wi-Fi	✔	✔	✔	✔	✔
8 ポート SFP+	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
8 ポート POE+	✔	✔	✔	–	–
演算	✔	✔	✔	セキュア絶縁モード **	セキュア絶縁モード **
ストレージ*	–	–	–	✔	✔
M.2 SATA*	–	–	–	✔	✔

> **注:**
>
> (*) Nodegrid Services Router は、最大 2 台の SATA ドライブをサポートしており、2 枚のストレージカードまたは 1 枚の M.2 SATA カードに分割できます。
>
> (**) **安全絶縁モード** は、通常のスロットにカードを挿入する場合と同様に管理できますが、ネットワークトラフィックは他のスロットから分離されます。

ラック取り付け

ラック取り付けブラケットに同梱されているすべてのユニットは、標準の19インチラックに収まるように取り付け可能です。2 つのラック取り付けブラケットは、セクション (ボックスの内容) に記載されているように、ボックスに入っています。これ以降、本書では、[ラックまたはキャビネット] を [ラック] と呼びます。

- 付属のネジ (ブラケットごとに 5 本) でラック取り付けブラケットを Nodegrid ユニットに取り付けます。

> **注:**一部のユニットはファンによってアクティブに冷却されるため、ファンの風向きを正しい方向に設定するために、ユニットをラックに適切に取り付ける必要があります。ファンの風向きは、ユニットの部品番号から指定できます。

モデル	部品番号	冷却済み	気流	
Nodegrid Serial Console - Tシリーズ	NSC-Txx-xxxx-xxx	パッシブ	該当なし	
Nodegrid Serial Console - Rシリーズ	NSC-TxxR-xxxx-xxx	パッシブ	該当なし	
Nodegrid Serial Console - S シリーズ	NSC-TxxS-xxxx-xxx-F	アクティブ	フロント-バック (吸気)	
Nodegrid Serial Console - S シリーズ	NSC-TxxS-xxxx-xxx-B	アクティブ	バック-フロント (排気)	
Nodegrid Services Router	NSR-xxxx-xxx	アクティブ	フロント-バック (排気)	
Nodegrid Services Router	NSR-xxxx-xxx	アクティブ	バック-フロント (吸気)	
Nodegrid Bold Services Router	BSR-xx-xxxx	パッシブ	該当なし	

- ユニットをラック内の割り当てられたスペースに置きます。

- 適切なラックネジ (付属しません) で締めてユニットを固定します。

ネットワーク接続

モデルとバージョンに応じて、ユニットには最低 2 つの銅線イーサネットポートか、2 つの SFP+ポートが備わります。 目的のネットワーク ケーブル (CAT5e、CAT6、CAT6A) をネットワークスイッチポートから、ユニットで使用可能な任意のネットワーク ポートに接続します。 SFP+ポートの備わるモデルの場合は、ユニットの電源を入れる前にSFP+モジュールを取り付け、適切なケーブルを接続します。

電源コードの接続

Nodegrid ユニットには、1 つまたは複数の電源装置 (AC/DC) が搭載されています。すべての電源装置を適切なケーブルでラック PDU など使用可能な電源に接続します。 ユニットの電源装置が 1 つだけの場合、電源障害の冗長性は利用できません。 電源装置を 2 つ備えたユニットは、電源障害に対して冗長性があります。両方の電源装置を 2 つの独立した電源に接続する必要があります。

> **注 - Nodegrid Services Router PoE 対応**: PoE 対応の Nodegrid Services Router では、2 番目の電源装置が PoE 機能に電力を供給するために使用されるため、停電用の冗長性を提供することはできません。

すべての電源装置が電源に適切に接続されると、電源装置はオンになります。

(DC 電源ポートの詳細については、付録 - DC 電源を参照してください)。

ターゲット装置の接続

シリアルターゲット装置の接続

> **注**: EMC の問題を回避するために、すべてのポート接続に高品質のネットワークケーブルを使用してください。

ユニットのシリアルポートとシリアルデバイス'のコンソールポートの間で使用されるケーブルとアダプタは、そのピン配置によって異なります。

ルータ、スイッチ、サーバなど、最新のシリアル デバイスには、そのコンソールポートとして DB9、RJ45、または USB ポートが備わります。ポートのピン配置については、シリアルデバイスコンソールの製造元'のマニュアルを参照してください。RJ45 ポートのコンソールポートの場合、Cisco のようなピン配置を使用する場合があります。

ユニットのシリアルポートとシリアルデバイスのコンソールポートに応じて使用されるケーブル配線については、以下の表を参照してください。

モデル	ポートタイプ	ピンアウト	デバイス ポート - RJ45 (レガシー)	デバイス ポート - RJ45 (Cisco)	デバイス ポート - DB9	デバイス ポート - USB
Nodegrid Serial Console - Tシリーズ	RJ45	レガシー	CAT5e ケーブル	CAT5e ケーブルと **Z000039** クロスオーバーアダプタ	CAT5e ケーブルと **Z000036** クロスオーバーアダプタ	USB
Nodegrid Serial Console - Rシリーズ	RJ45	Cisco	-	CAT5e ケーブル	CAT5e ケーブルと **Z000015** クロスオーバーアダプタ	USB
Nodegrid Serial Console - S シリーズ	RJ45	自動検出 (レガシー/Cisco)	CAT5e ケーブル	CAT5e ケーブル	CAT5e ケーブルと **Z000015** クロスオーバーアダプタ	USB
Nodegrid Services Router	RJ45	Cisco	-	CAT5e ケーブル	CAT5e ケーブルと **Z000015** クロスオーバーアダプタ	USB
Nodegrid Bold Services Router	RJ45	Cisco	-	CAT5e ケーブル	CAT5e ケーブルと **Z000015** クロスオーバーアダプタ	USB

シリアル デバイス'の RJ45 のピン配置が、Cisco のような配置と異なる場合、またはお使いのシリアルデバイスをユニットに接続する上でご不明な点がある場合、 ZPE Systems のテクニカルサポート にお気軽にお問い合わせください。

IP ターゲット デバイスの接続

> **注**: EMC の問題を回避するために、すべてのポート接続に高品質のネットワークケーブルを使用してください。

すべての IP ベースのターゲットデバイスは、Nodegrid ユニットのネットワークインターフェースへの直接接続、既存のネットワークインフラストラクチャを介した接続の両方が可能です。 ターゲットデバイスが直接接続されている場合、イーサネット接続に標準ネットワークケーブル (CAT 5、CAT6、CAT6e)、または適切なファイバケーブルを使用できます。

Nodegrid への接続

コンソール ポートを介した接続

付属の CAT5e と RJ45-DB9 **Z000036** アダプタ/ケーブルを使用して、Nodegrid ユニットと通信します。CAT5e ケーブルの一端を Nodegrid コンソールポートに接続します。もう一方の端を RJ45-DB9 アダプタに接続し、お使いのノートパソコンまたは PC の DB9 COM ポートに接続します (お使いのノートパソコンや PC に DB9 COM ポートがない場合は、USB-DB9 アダプタを使用します (付属しません)) 。

シリアルアプリケーション (例: xterm、TeraTerm、Putty、SecureCRT) をお使いのノートパソコン/PC で実行して、その COM ポート (使用される COM ポートについてのシステム情報を参照してください) へのターミナルセッションを、115200bps、8 bits、パリティなし、1 ストップビット、フロー制御設定なしで開始します。

ETH0 を介した接続

ETH0 インターフェースは、デフォルトで DHCP 要求を聞き取るように設定されています。DHCP サーバが使用できない場合、ユニットはデフォルトの IP アドレス **192.168.160.10**を使用します。ユニットは、 ブラウザ https://[DHCP ASSIGNED IP]、https://192.168.160.10、または ssh クライアントによってアクセスが可能です。

設定	値
DHCP	有効
フォールバック IP	はい
デフォルト IP	192.168.160.10/24
デフォルトの URL	https://192.168.160.10
デフォルトの ssh	`ssh admin@192.168.160.10`

Wi-Fi を介した接続

Nodegrid は、適切な Wi-Fi デバイスが接続されている場合に備えて、Wi-Fi ホットスポットとして機能するように事前設定されています。これは、内蔵 Wi-Fi モジュール、または USB Wi-Fi アダプタのいずれかです。

Nodegrid は、SSID **Nodegrid**を使用して、Wi-Fi ネットワークを自動表示します。デフォルトの WPA 共有キーは、**Nodegrid** です。Nodegrid は、IP アドレスをクライアントに自動で提供はしません。192.168.162.1/24 の範囲で有効な IP アドレスを持つクライアントを設定します。 これで、ブラウザの Https://192.168.162.1 から、または ssh を介して、ユニットにアクセスできるようになりました。

設定	値
SSID	Nodegrid
WPA共有キー	Nodegrid
デフォルトのネットワーク	192.168.162.1/24
デフォルトの URL	https://192.168.162.1
デフォルトの ssh	`ssh admin@192.168.162.1`

KVM ポートを介した接続

Nodegrid ユニットは、KVMインターフェースを介して直接設定や管理が可能です。モニターを HDMI ケーブルでユニットの HDMI インターフェースに接続します。

Nodegrid Bold SR は、HDMI インターフェースの代わりに VGA ポートを提供します。

> **注**: HDMI から DVI-D のアダプタを使用して、DVI-D モニターの接続を可能にします。

USB キーボードとマウスを、使用可能な USB ポートに接続します。

> **注**: キーボードとマウスは Linux でサポートされている必要があり、Windows **のみ**のデバイスはサポートされていません。この制限は、USB ワイヤレス ドングルを使用するデバイスに主に影響します。

ログインプロンプトが表示されます。

I/O ポート (GPIO)

Nodegrid Gate SR は、2 つのデジタル I/O ポート (DIO0、DIO1)、1 つのデジタル出力ポート (OUT0)、および 1 つのリレー ポート (1A@24V) をサポートします。
Nodegrid Link SR は、2 つのデジタル I/O ポート (DIO0、DIO1) と 2 つのデジタル出力ポート (OUT0、OUT1) をサポートします。

DIO0 および **DIO1** の両方は、出力または入力として独立して設定可能です。DIO0 および DIO1 は、TTL レベル (5.5V 最大 @ 64mA)と JESD 22 を超える ESD 保護を備えたオープンドレインデジタル I/O ポートです。
DIO ポートが入力用に設定されている場合、次の意味を持ちます:

- 接点が開いている場合、高またはデジタル '1' 。
- 設定が閉じている場合、低またはデジタル '0' 。

> **注**: DIO0およびDIO1ポートを入力用に設定することは、ドアの閉鎖、振動、水、煙センサーのような乾接点アプリケーションに最適です。

DIO ポートが出力として設定される場合、以下のように出力されます:

- 高に設定した場合、TTL 高として;
- 低に設定した場合、TTL 低として。

> **注**: DIO0およびDIO1ポートを出力用に設定することで、低電圧/電流アプリケーションの制御に使用できます。

OUT0 および **OUT1** は、高電圧デジタル出力です。各ポートは、内部でシグナル MOSFET に接続されています。出力ポートは通常オープン (NO) で、500mA、2.5V～60Vの電圧範囲に対応できます。 OUT ポートが以下である場合:

- 高に設定すると、有効/アクティブになり、出力 OUT を地面に引き下げます。
- 低に設定すると、無効/非アクティブになり、出力 OUT が開いたままになります。

> **注**: OUT0 と OUT1 は、リレー回路のように、接続された電源回線を地面に引き下げるために使用できます。

Nodegrid Gate SR の **RELAY** ポートは、通常クローズド (NC) リレーで、1A で最大 24V のディレーティング値を持ちます。ただし、RELAY の仕様に従い、60 W、125VA の最大スイッチング電力; 220VDC、250VACの最大スイッチング電圧; 抵抗性負荷での最大スイッチング電流 2A をサポートします。
RELAY の主な役割は、**電源制御アラーム**として動作することです。RELAY が閉じている場合、Nodegrid

Gate SR が単一の電源から電力を供給されているか、まったく電源が供給されていないことを示します。したがって、この RELAY が閉じている時に Nodegrid Gate SR が両方の入力電源から電力を供給されている場合、少なくとも 1 つの入力電源で障害が発生していることを示します。
オプションで、外部デバイスを制御するためにソフトウェア制御 (オープン/クローズ) に従うように RELAY 機能を変更することができます。以下は可能なリレーの状態です:

- オープン、リレー接点が開きます。
- クローズ、リレー接点が閉じます。

I/O ポート設定は、`System :: I/O Ports`の下にあります。I/O ポートのステータスとその他のハードウェア情報は、`Tracking :: HW Monitor`の下にあります。

安全上の理由から、各ポートで定義された最大電圧/電流を超えないようにしてください。

Nodegrid Manager のインストール

Nodegrid Manager ソフトウェアは、ISO ファイルからインストールします。次の 3 段階のプロセスに沿ってインストールします:

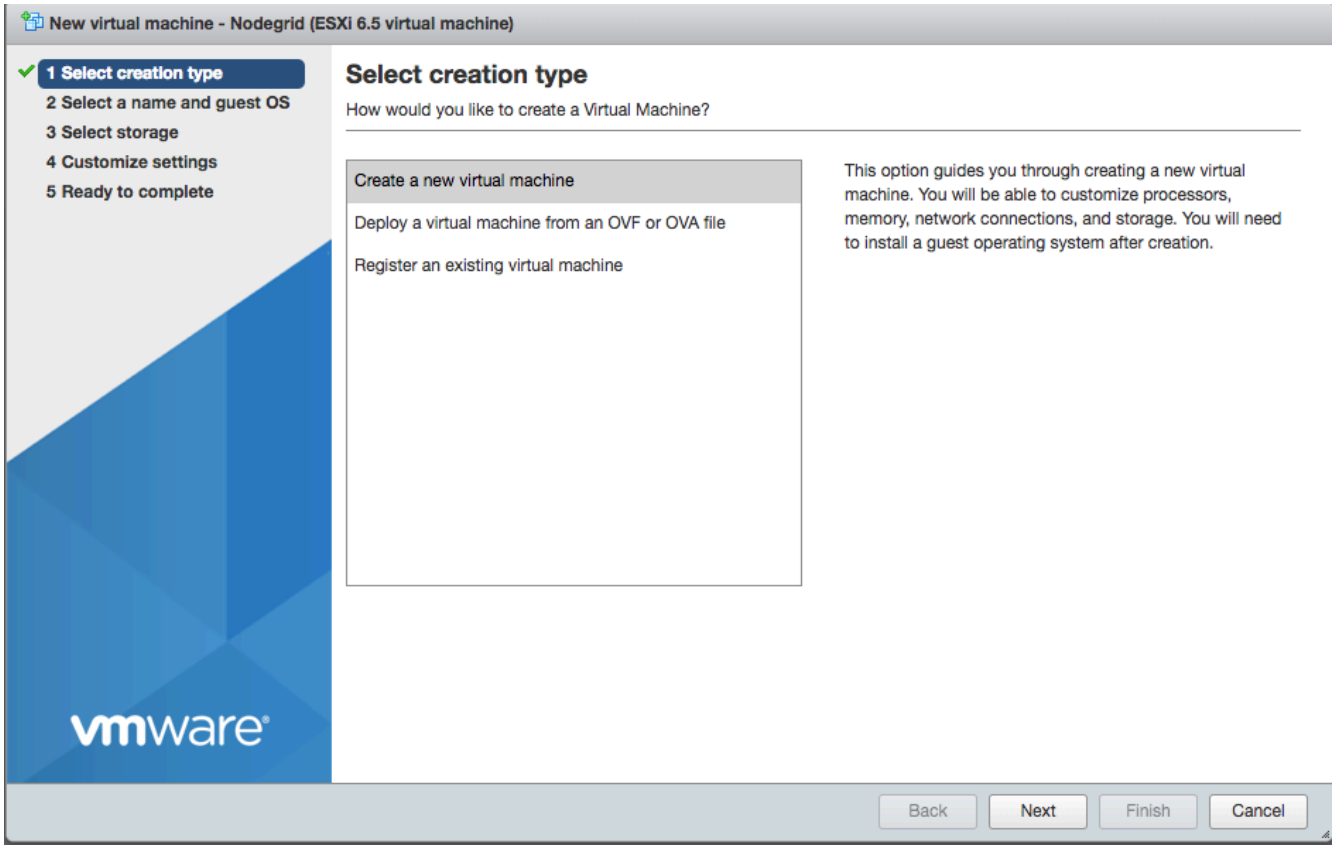
- 仮想マシンを作成します
- ソフトウェアをインストールするために、ISO ファイル/CD から起動します
- 新しく作成された仮想マシンから再起動・起動します。

最小要件

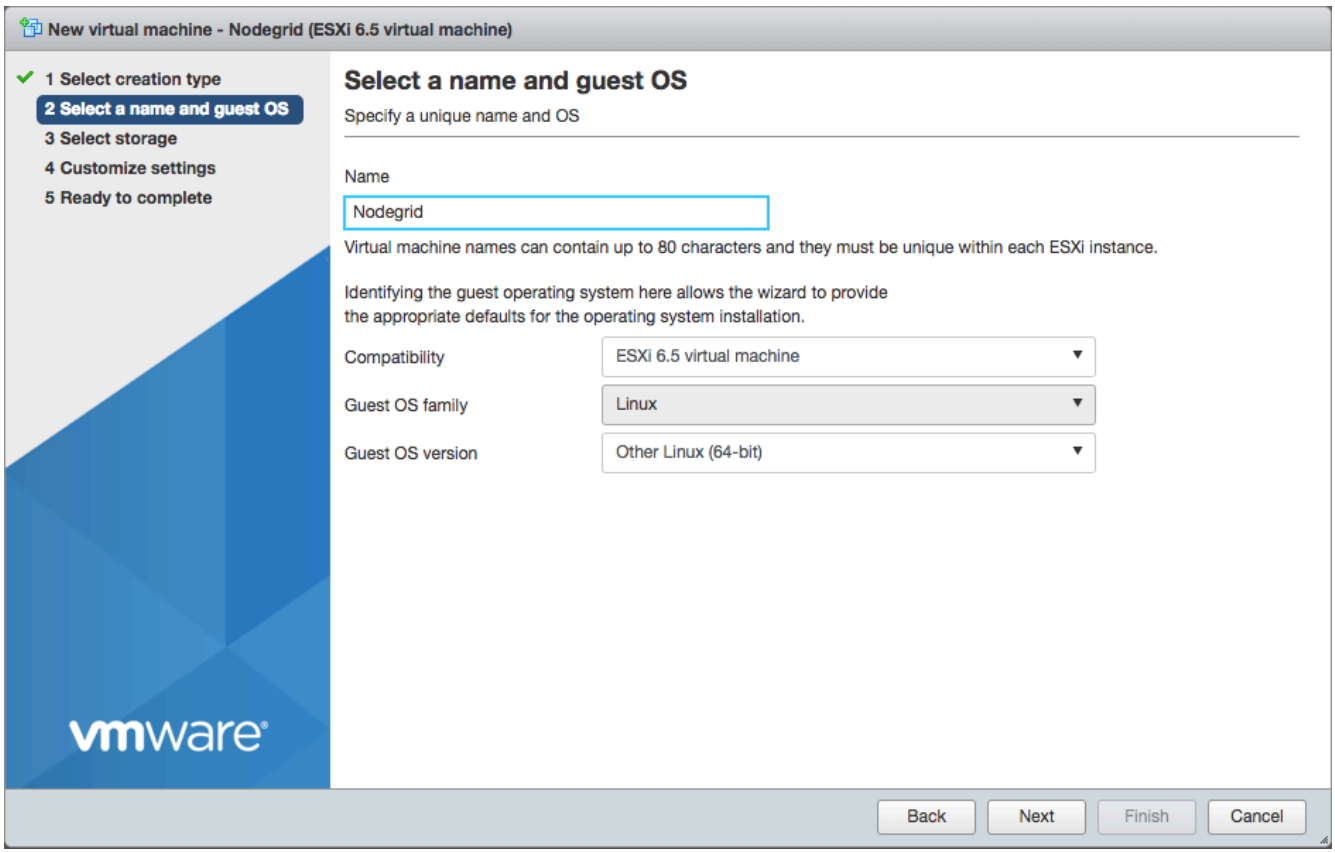
- ESXi 4.1以上
- 32 GB ハードドライブ (LSI ロジックパラレルコントローラを介して接続)
- 4 GB メモリ (8 GB 推奨)
- 2 ネットワーク アダプタ (E1000 アダプタ推奨)

仮想マシンの作成 - VMWare

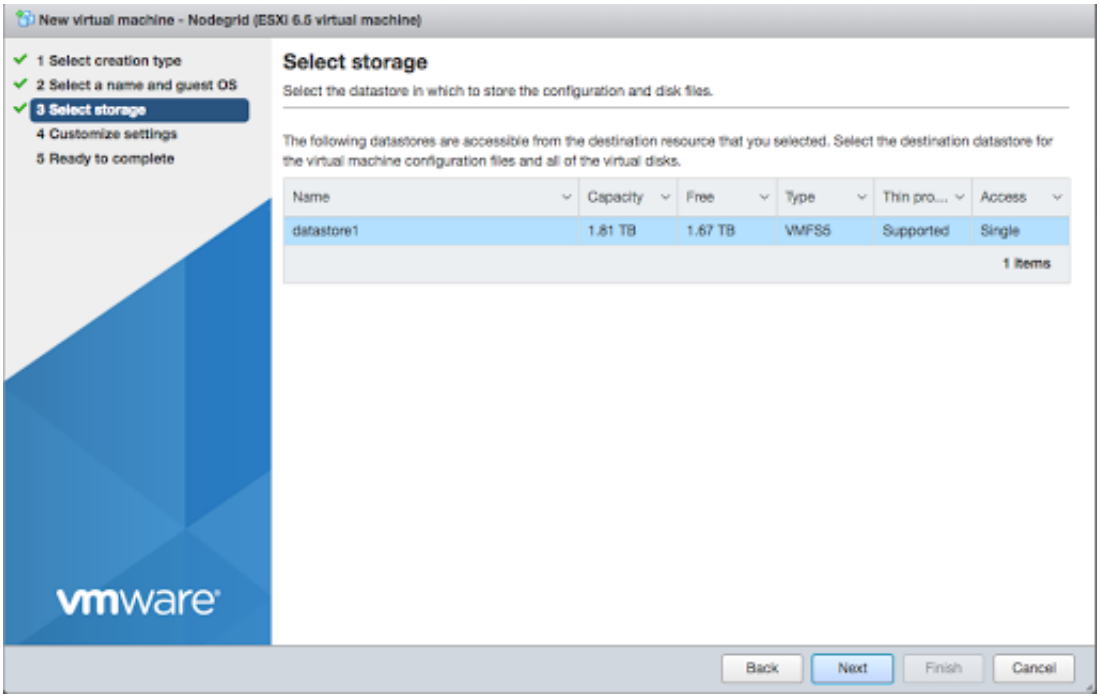
- ESXi vSphere 画面から、*[新しい仮想マシンの作成]* リンクをクリックします
- 仮想マシンの設定については、[新しい仮想マシンを作成] をクリックし、_[次へ] _をクリックします



- Nodegrid Manager 仮想マシンのための適切な*[名前]*を選び、 select as *[ゲスト OS ファミリー]*として**Linux**、*[ゲスト OS バージョン]*として**他の Linux (64ビット)**を選択し、 *[次へ]*をクリックします

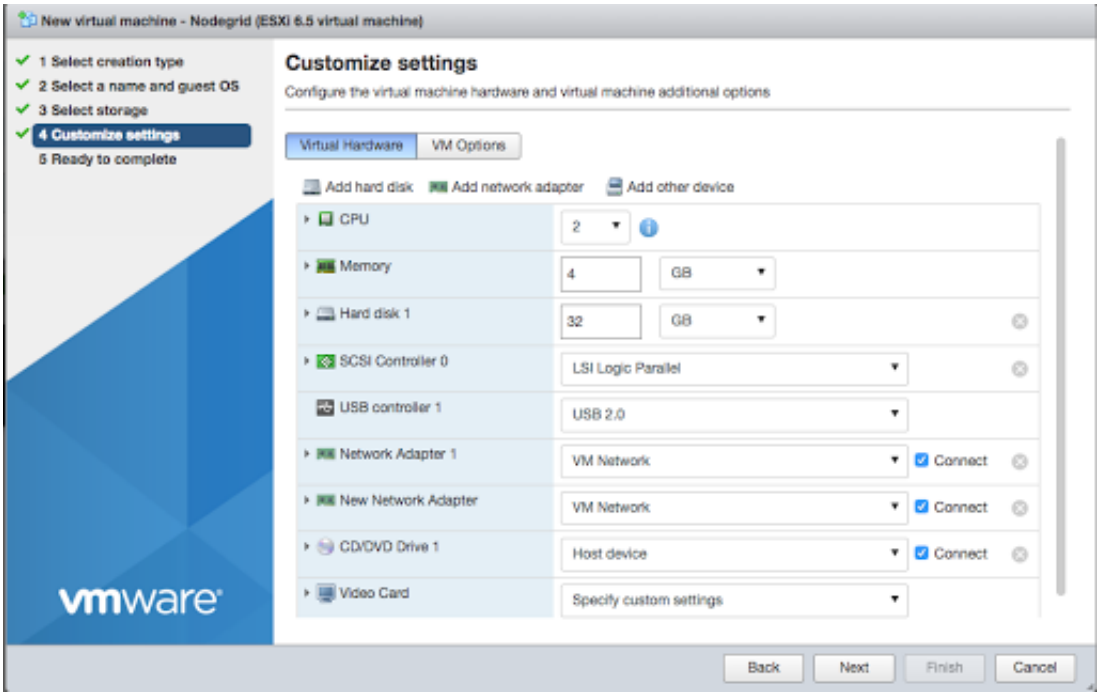


- 新しい仮想マシン用に作成するデータ ストレージの容量を選択し、*[次へ]*をクリックします

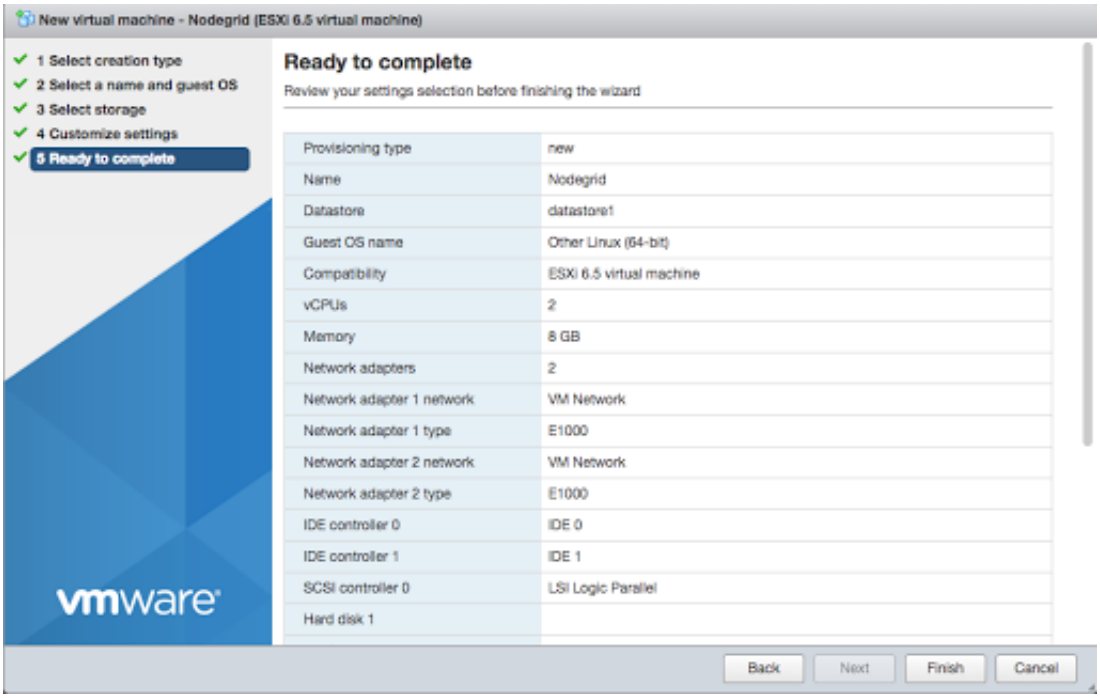


- *[設定のカスタマイズ]*画面で、次の設定を入力します:
  CPU: _**2**
メモリ: ***4GB***
  ハードディスク*:* ***32GB***
  SCSI コントローラ*:LSI ロジックパラレル*
  ネットワークアダプタ_: **E1000** タイプの**2**
  _[次へ] _をクリックします

> 注: この値は最小設定であり、必要に応じて調整する必要があります。



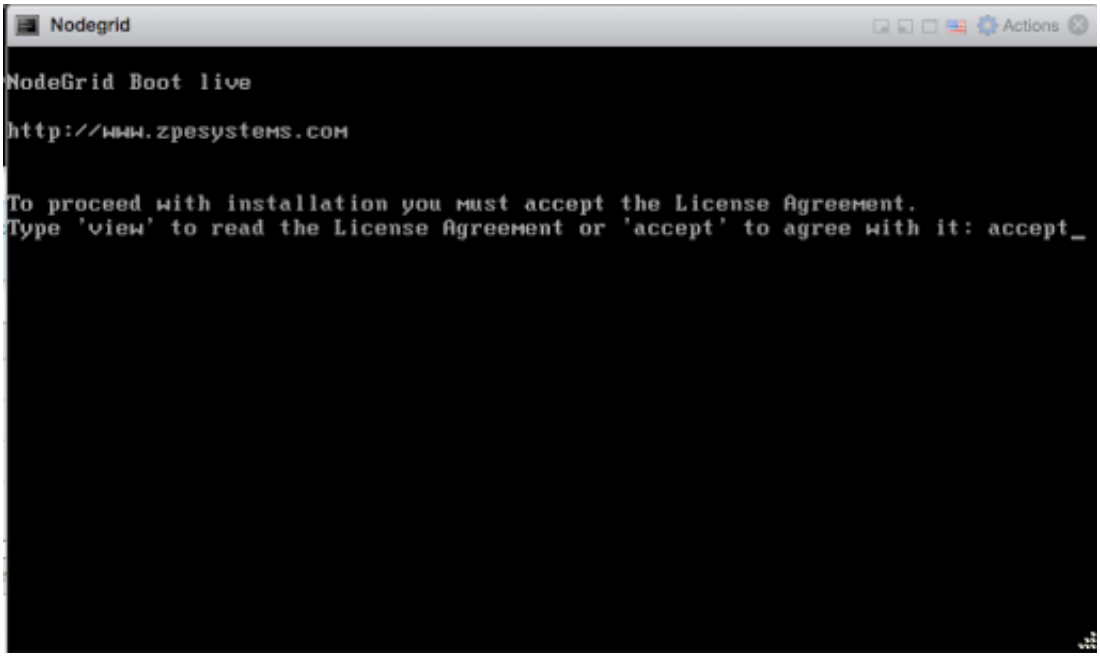
- *[完了]*をクリックして、ESXi サーバ上の仮想マシンの設定を完了します。



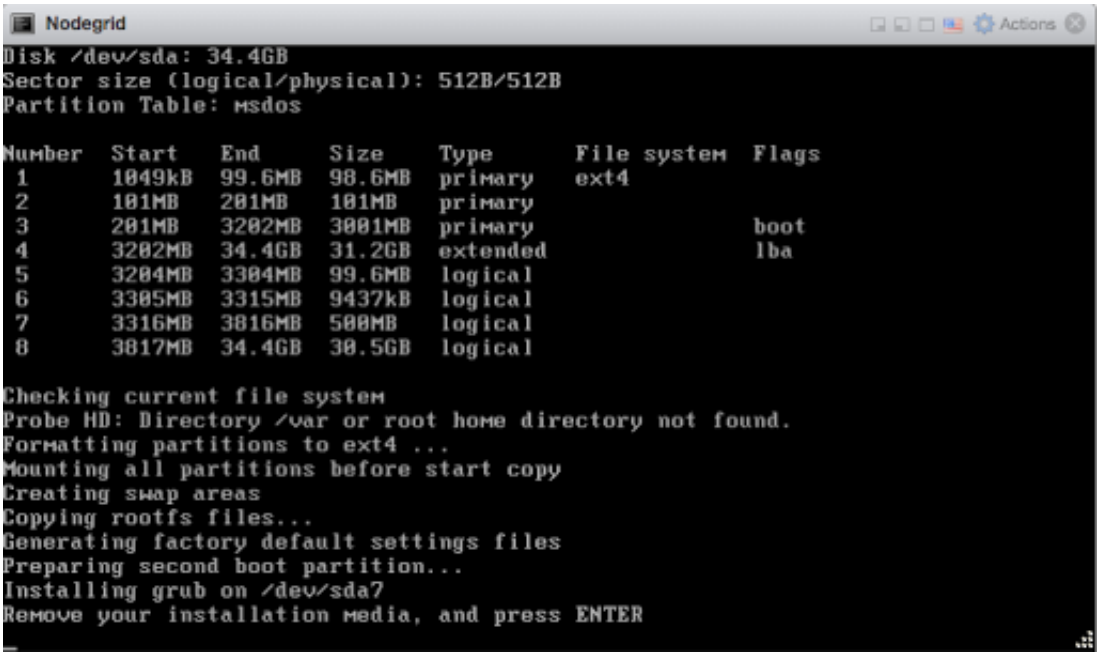
Nodegrid Manager のインストール

Nodegrid Manager ソフトウェアのインストール方法:

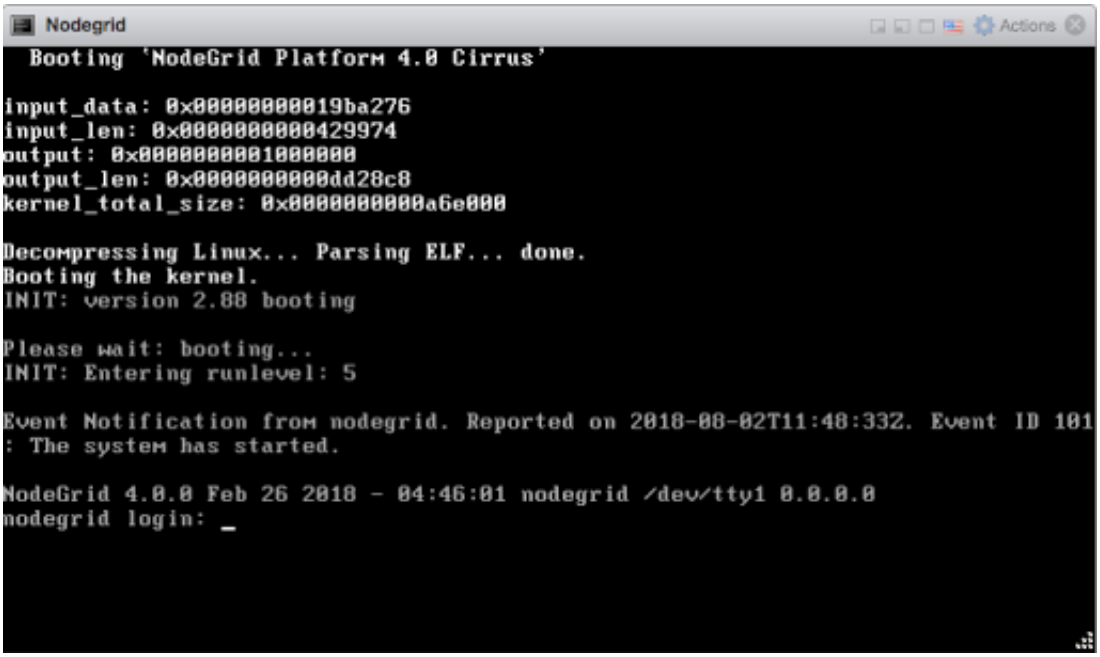
- 仮想マシンの概要画面の*[コンソール]*タブをクリックします
- 電源を入れます。オペレーティングシステムがインストールされていないので、仮想マシンを起動できません
- *CD/DVD* アイコンをクリックし、システム内の Nodegrid Manager ISO ファイルのロケーションを選択します
- コンソール領域の *[CTL-ALT-INSERT]* をクリックして仮想マシンを再起動します
- 仮想マシンの Console Server ソフトウェアは、ブートプロンプトで開始されます。ブートプロンプトで、 *[ENTER]*キーを押すか、待機します。画像は解凍された後読み込まれます
- 画像が起動したら、コンソールの指示に従います。インストールを続行するには、EULA に同意する必要があります。**[同意]**を入力してください



- インストールの過程で、ファイルは仮想マシンにコピーされ、システムが自動で再起動して Nodegrid Manager が起動します。*[ENTER]* をクリックして画像を起動するか、画像が自動で起動するのを待ちます。



- 画像を起動すると、Nodegrid Manager の新規コピーが使用可能になり、設定の準備が整います。



ネットワークの初期設定

Nodegrid Platform がオンになると、起動メッセージ、次にログインプロンプトが表示されます。

デフォルトの管理者ユーザー名は**admin**、 デフォルトのパスワードも**admin**です。 管理者ユーザーは、Web インターフェース (HTTPS) または CLI (SSH) を介して、コンソール ポートを介して Nodegrid Platform にアクセスできます。その他のアクセス方法を有効にできます。

スーパーユーザーは、**root**であり、デフォルトのパスワードは**root**です。 このルートユーザーは、Linux OS に SHELL アクセスできますが、Web インターフェースにはアクセスできません。

Nodegrid Platform は、デフォルトで、DHCP IP 設定が有効化された状態で設定されます。

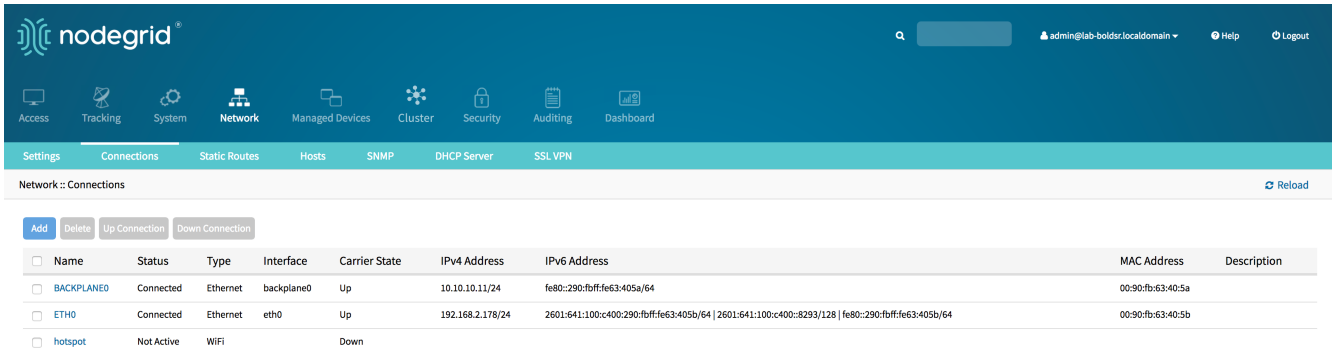
> **注:**Nodegrid Platform は、DHCP サーバに障害が発生した場合や使用できない場合、ETH0 で 192.168.160.10 で応答します。

現在の IP アドレスを識別します

現在割り当てられている IP アドレス ログインを管理者ユーザーとしてNodegrid Platformに識別し、[ネットワーク接続] 画面に移動します。

現在の IP アドレスを識別します - WebUI

- デフォルトのパスワード**admin**で、**admin**ユーザーとしてログインします
- ネットワークに移動します :: 接続



現在の IP アドレスを識別 - CLI

- デフォルトのパスワード**admin**で、**admin**としてログイン
- 現在の設定を`show /system/network_statistics/`で表示

出力例:

```
[admin@nodegrid /]# show /settings/network_connections/
  name         status       type       interface   carrier state  ipv4 address
ipv6 address                   mac address        description
  ==========   ==========   ========   ==========  =============  =================
============================  =================  ===========
  BACKPLANE0   connected    ethernet   eth0        up             192.168.10.252/24
fe80::290:fbff:fe5b:72bc/64  e4:1a:2c:5b:72:bc
  ETH0         connected    ethernet   backplane0  up             192.168.29.3/24
fe80::290:fbff:fe5b:72bd/64  e4:1a:2c:5b:72:bd
  hotspot      not active   wifi                   down
```

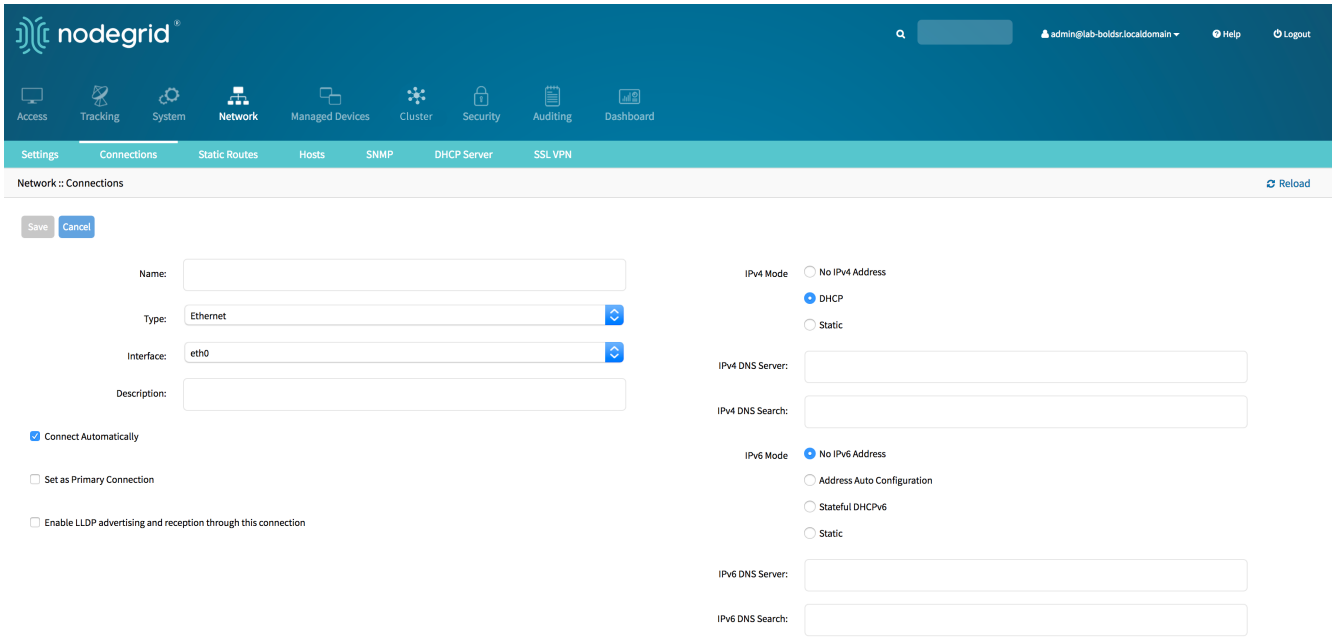
静的 IP アドレスの定義

- DHCP サーバをネットワークで使用できない場合、または動的 IP から静的 IP に変更したい場合、ネットワークパラメータを設定します。

> **注:** 以下の例では、通信に IPv4 を使用します。IPv6 は、Nodegrid Platform で完全にサポートされており、同じメニューで適切な設定がご利用いただけます。

静的 IP アドレスの定義 - Web UI

- ネットワークに移動:: 接続
- 設定するインターフェースをクリック
- 必要な詳細を提供



- [保存] をクリック

静的 IP アドレスの定義 - CLI

- 目的のネットワークインターフェースに移動します

```
[admin@Nodegrid /]# cd settings/network_connections/ETH0/
```

- ネットワークインターフェースの設定

```
[admin@Nodegrid ETH0]# set ipv4_mode=static
[admin@Nodegrid ETH0]# set ipv4_address=<IP_ADDRESS> ipv4_bitmask=<BITMASK>
ipv4_gateway=<GATEWAY>
[admin@Nodegrid ETH0]# commit
```

例:

```
[admin@Nodegrid /]# cd settings/network_connections/ETH0/
[admin@Nodegrid ETH0]# set ipv4_mode=static
[admin@Nodegrid ETH0]# set ipv4_address=10.0.0.10 ipv4_bitmask=24
ipv4_gateway=10.0.0.1
[admin@Nodegrid ETH0]# show
name: ETH0
type: ethernet
ethernet_interface = eth0
connect_automatically = yes
set_as_primary_connection = no
enable_lldp = no
```

```
ipv4_mode = static
ipv4_address = 10.0.0.10
ipv4_bitmask = 24
ipv4_gateway = 10.0.0.1
ipv4_dns_server =
ipv4_dns_search =
ipv6_mode = address_auto_configuration
ipv6_dns_server =
ipv6_dns_search =
[admin@Nodegrid ETH0]# commit
```

必要に応じて、他のインターフェースについても同じ手順に従います。

インターフェース

WebUI

Nodegrid Platform は、WebUI のビルドを介してアクセスできます。このインターフェースは、すべてのターゲットデバイスへの完全なアクセスとプラットフォームの設定と管理を可能にします。

Web UI は、モバイルブラウザを含む HTML5 に対応するすべての最新ブラウザに対応しています。Internet Explorer 11、Edge、Chrome、Firefox は定期的にテストされているブラウザです。

WebUI は、以下の一般的なストラクチャを提供します。

メニュー	アイコン	説明
アクセス		アクセスメニューから、すべてのユーザーが管理対象デバイスに簡単にアクセスできます。これにより、適切なアクセス許可を持つユーザーは、セッションを開始し、電源を制御し、デバイスのロギングの詳細を確認することが可能です。
トラッキング		追跡メニューには、システム使用率やシリアルポートの統計など、一般的な統計情報とシステム情報の概要が表示されます。
システム		システムのメニューでは、管理者は Nodegrid Platform で一般的な管理タスク (ファームウェアの更新、バックアップ、復旧、ライセンスなど) を実行できます。
ネットワーク		[ネットワーク] メニューから、すべてのネットワークインターフェースと機能へのアクセスと管理が可能です。
管理対象デバイス		管理者は、Nodegrid Platform を介して管理する必要があるデバイスを、このメニューから追加、設定、および削除することができます。
クラスタ		管理者は、クラスタメニューから Nodegrid Cluster 機能を管理できます。
セキュリティ		[セキュリティ] メニューには、Nodegrid Platform のユーザーアクセスと一般的なセキュリティを制御する設定オプションが用意されています。
監査		管理者は、このメニューから監査レベル、ロケーション、一部のグローバルロギング設定を管理できます。
ダッシュボード		ダッシュボードでは、ユーザーと管理者はダッシュボードとレポートを作成して表示できます。
アプリケーション		アプリケーションのメニューは、有効な仮想化ライセンスを使用可能な場合にのみ表示されます。管理者は、適切なライセンスで NFV および Docker アプリケーションを管理および制御できます。

CLI

CLI インターフェースを介して Nodegrid Platform にアクセスできます。CLI には、ssh クライアントかコンソールポートを介してプラットフォームに接続してアクセスします。このインターフェースは、プラットフォームのすべてのコンソールターゲットセッション、設定、および管理へのアクセスを可能にします。CLI ストラクチャは、主に WebUI ストラクチャに従います。

CLI は、以下の一般的なストラクチャを提供します。

フォルダ	説明
/アクセス	アクセスメニューから、すべてのユーザーが管理対象デバイスに簡単にアクセスできます。これにより、適切なアクセス許可を持つユーザーは、セッションを開始し、電源を制御し、デバイスのロギングの詳細を確認することが可能です。
/システム	このフォルダーには、Web UI から追跡メニューとシステムメニューを組み合わせた機能が用意されています。追跡機能には、システム使用率やシリアルポートの統計など、一般的な統計情報とシステム情報の概要が表示されます。システムの機能により、管理者は Nodegrid Platform で一般的な管理タスク (ファームウェアの更新、バックアップ、復旧、ライセンスなど) を実行できます。
/設定	このフォルダでは、システム、セキュリティ、監査、および管理対象デバイスの設定と設定オプションにアクセスできます。

CLI では多くのコマンドとオプションが提供されていますが、一般的な使用法として、ユーザー/管理者は、手始めにいくつかの基本的なコマンドに分割することが可能です。

CLI コマンド	説明
`TAB TAB`	ダブル TAB キーの組み合わせで、現在有効であり使用可能なすべてのコマンド、設定、またはオプションのリストが表示されます
`ls`	コマンド `ls` は、現在のフォルダのストラクチャを一覧表示します
`show`	[有効な場合に表示] コマンドは、表形式のビューに現在の設定を表示します
`set`	すべての変更と設定は、`set オプション=値` の一般的な形式の設定コマンドで開始され、複数設定は次のような追加のペア `オプション=値` を提供することによって組み合わせが可能になります `set オプション1=値1 オプション2=値2`
`commit`	ほとんどの変更は、直接保存されたり有効化されたりすることはなく、設定への変更は、`commit` コマンドで保存・； 有効化される前に、`show` コマンドでレビューすることが可能です。その変更はまだ有効ではなく、保存する必要があるものは、コマンドプロンプトの前の次のようなサイン `+` によって CLI に示されます。 `[+admin@nodegrid /]#`
`cancel` または `revert`	設定をコミットして保存する必要がない場合、`revert` または `cancel` コマンドを使用して変更を元に戻すことができます。

例

```
[admin@nodegrid /]# ls
```

```
access/
system/
settings/
[admin@nodegrid /]# show
[admin@nodegrid /]# show /access/
  name                    status
  ======================  =========
  Device_Console_Serial   Connected
[admin@nodegrid /]# set settings/devices/ttyS2/access/ mode=on-demand
[+admin@nodegrid /]# set settings/devices/ttyS2/access/ rs-
232_signal_for_device_state_detection=
CTS   DCD   None
[+admin@nodegrid /]# set settings/devices/ttyS2/access/ rs-
232_signal_for_device_state_detection=DCD enable_hostname_detection=yes
[+admin@nodegrid /]# commit
[admin@nodegrid /]#
```

Shell

Nodegrid Platform では、オペレーティングシステムの Shell に直接アクセスできます。このアクセスは、デフォルトでは**ルート**ユーザー (直接) と**管理者**ユーザー (CLI から) に限られています。 Shell への直接アクセス権は、特定グループのユーザーに付与できます ([グループ] セクションを参照)。これは、システムオートメーションプロセスで使用するために直接 Shell にアクセスする必要があるユーザーに役立ちます。 Nodegrid は、ssh キー認証による許可ばかりでなく、このような使用もサポートします。 Shell へのアクセス要件を確認し、必要に応じてアクセスを制限することをお勧めします。Shell へのアクセスは、高度なユースケースに対して提供されるものであるため、注意して使用する必要があります。Shell を通して Nodegrid Platform の設定に加えられた変更は、プラットフォームの一般的な動作に悪影響を及ぼす可能性があります。

デバイスアクセス

このページ`Access`には、使用可能なすべてのターゲットデバイスの概要が表示されます。これにより、ユーザーは管理対象デバイスに簡単に接続し、現在のデバイスの状態を確認し、ターゲットデバイスを検索することができます。表示されるターゲットデバイスは、ユーザーの権限と Nodegrid Cluster ノードの現在の状態によって決定されます。

デバイスセッション

Web UI にログインした後にユーザーが使用できる最初のビューは、`Access`ビューです。 このビューには、ユーザーがアクセスでき、使用可能なすべてのターゲットの概要が表示されます。各ターゲットは、現在の接続状態と使用可能な接続タイプを示します。

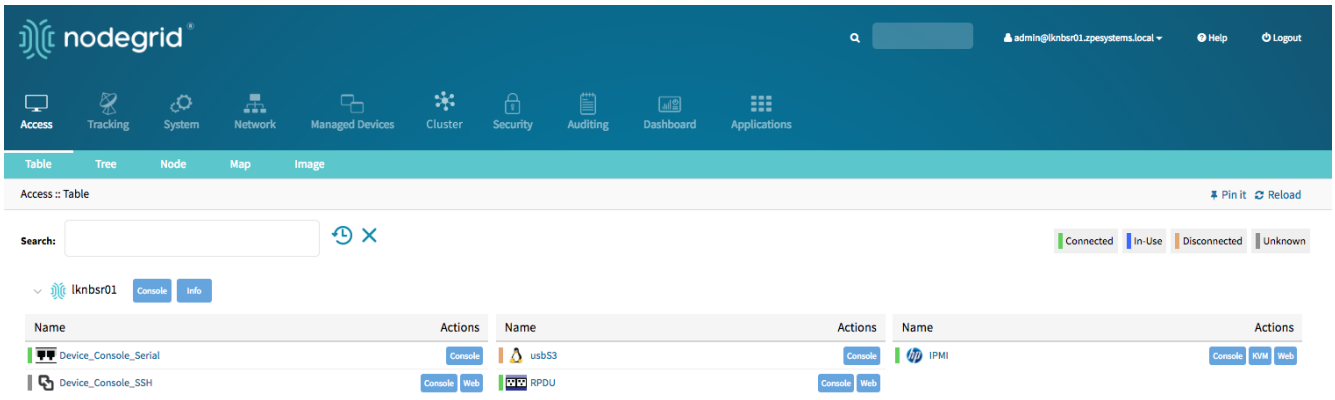
接続状態は次のとおりです:

状態	インジケータの色	アイコン	説明
接続中	緑	Connected	Nodegrid はターゲットデバイスに正常に接続でき、セッションで使用可能
使用中	青	In-Use	デバイスは現在使用中
切断済み	オレンジ	Disconnected	Nodegrid はターゲットデバイスに正常に接続できず、セッションで使用不能
原因不明	グレイ	Unknown	接続状態が不明です。これは、接続モードが**オンデマンド**のターゲットデバイス、または検出プロセスが完了していない**新しいターゲット**デバイスのデフォルトの状態です。

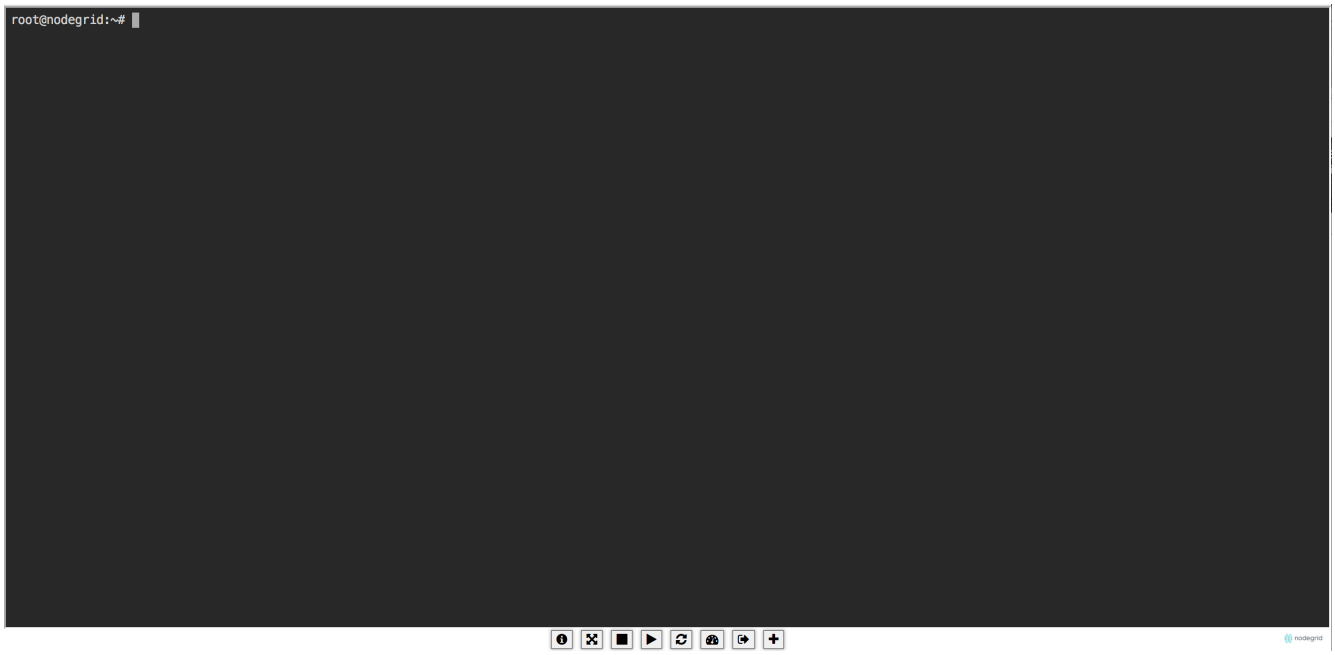
デバイスセッションは、このロケーションから直接開始できます。

デバイスセッション - Web UI

WebUI からデバイスセッションを開始するための、複数のオプションが提供されています。画面`Access`では、利用可能なターゲットセッションを直接表示し、接続ボタンをクリックするだけで新しいセッションを開始できます。



これにより、ターゲットセッションが確立される新規ウィンドウが開きます。



ウィンドウの下部には、ターゲットセッションとターゲットデバイスをさらに制御可能なボタンが表示されます。使用可能なオプションは、接続タイプとデバイスの設定によって異なります。

オプション	説明
	[情報] オプションには、現在のデバイスの詳細が表示されます。
	[フルスクリーン] でウインドウが拡張し、全画面表示になります。セッションウィンドウ自体は、その最大サイズを超えて拡張されることはありません。
	[電源オフ] オプションは、接続されているラック PDU または IPMI デバイスを介して、ターゲットデバイスの **電源をオフ**にします。
	[電源オン] オプションは、接続されているラック PDU または IPMI デバイスを介して、ターゲットデバイスの **電源をオン**にします。
	[リセット] オプションは、接続されているラック PDU または IPMI デバイスを介して、ターゲットデバイスで**電源サイクル**を実行します。
	電源ステータスには、接続されたラック PDU または IPMI デバイスによって返されたデバイスの現在の電源ステータスが表示されます。
	このオプションは、現在アクティブなセッションを終了させます
	プラス記号は、画面下部にあるコマンドラインオプションを拡大/縮小します。

セッションのウィンドウを閉じると、ターゲットデバイスも閉じられます。

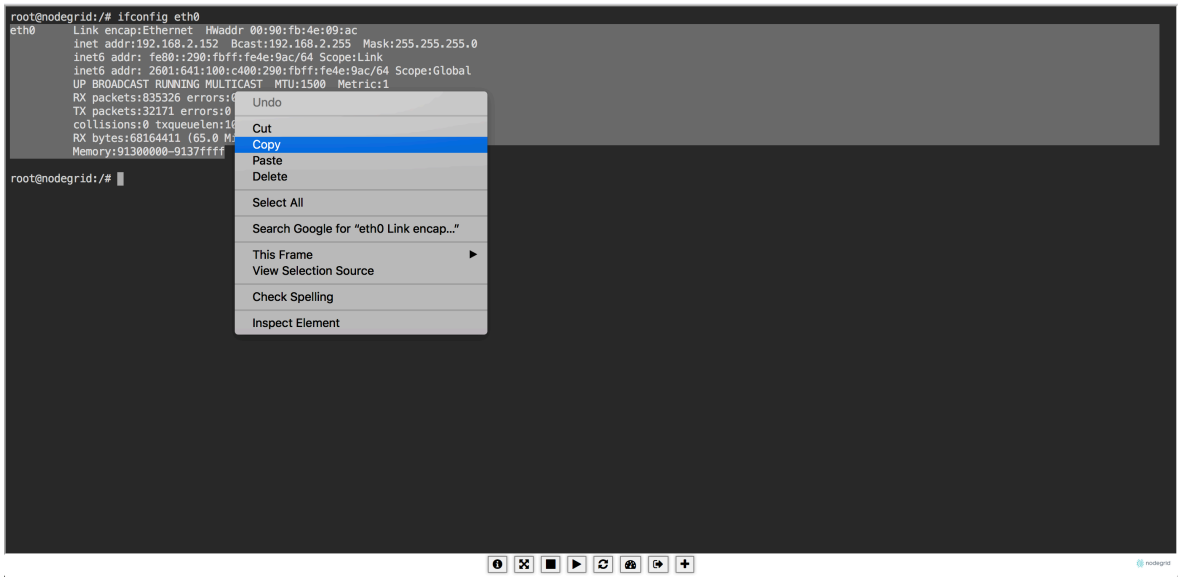
コピー＆ペースト

Nodegrid **は、他のアプリケーションと同様、HTML5 グラフィカルデバイスセッションウィンドウとデスクトップ環境との間で、テキストのコピーとペーストが可能です。**

ただし、オペレーティングシステムの特殊性により、コピー＆ペーストの操作を有効にするために、以下のようなキーの明確な組み合わせが必要となることにご注意ください。

- Windows および Linux では、コピーには`Ctrl+Ins`を、ペーストには`Shift+Ins`を使用します。
- Mac では、コピーには`Cmd+C`を、ペーストには`Cmd+V`を使用します。

テキストをハイライトし、右クリックでメニューを開くか、ショートカットを使用します。



デバイスセッション - CLI

アクセスビューは、`access`メニューの CLI で使用でき、ユーザーは、`cd /access`でこのメニューに直接移動できます。現在使用可能なターゲットを表示するには、コマンド`show`を使用します。

例:

```
[admin@nodegrid access]# show
  name                    status
  ======================  =========
  Device_Console_SSH      Connected
  Device_Console_Serial   InUse
  IPMI                    Connected
  RPDU                    Connected
  usbS2                   Connected
```

デバイスセッションは、ここから`connect`コマンドで直接開始できます。 使用: `connect <target name>`

例:

```
[admin@nodegrid access]# connect Device_Console_Serial
[Enter '^Ec?' for help]
[Enter '^Ec.' to cli ]

login:
```

> 注: CLI から開始できるのは、コンソールセッションかテキストベースのインターフェースを提供するセッションのみです。

接続が確立されると、ユーザーは、エスケープシーケンス `^Ec` を使用するか、`^O` セッションをさらに制御します。

> メモ： エスケープシーケンスは、デバイスの設定で変更できます。

以下のオプションを使用できます。

オプション	エスケープシーケンス	説明
.	`^Ec.`	現在のセッションを切断
g	`^Ecg`	現在のユーザーグループ情報を表示
l	`^Ecl`	デバイス設定で定義されているブレーク信号を送信
w	`^Ecw`	現在接続されているユーザーを表示
<cr>	`^Ec<cr>`	無視/中止コマンド信号を送信
k	`^Eck`	シリアルポート (速度データ　ビット パリティ ストップ ビット フロー)
b	`^Ecb`	ブロードキャストメッセージを送信。エスケープシーケンスが送信された後でメッセージを入力できます。
i	`^Eci`	現在のシリアルポート情報を表示
s	`^Ecs`	現在のセッションを読み取り専用モードに変更
a	`^Eca`	現在のセッションを読み取り/書き込みモードに変更
f	`^Ecf`	現在のセッションを強制的に読み取り/書き込みモードにします
z	`^Ecz`	特定の接続されたユーザーセッションを切断
?	`^Ec?`	このメッセージを印刷します

電源制御オプションは、管理対象となるラック PDU に接続されるターゲットや、IMPI を介した電源制御を提供するターゲットで使用できます。電源メニューは、次のようにして開始できます `^O`

```
Power Menu - Device_Console_Serial
Options:
1. Exit
2. Status
3. On
4. Off
5. Cycle

Enter option:
```

デバイス情報

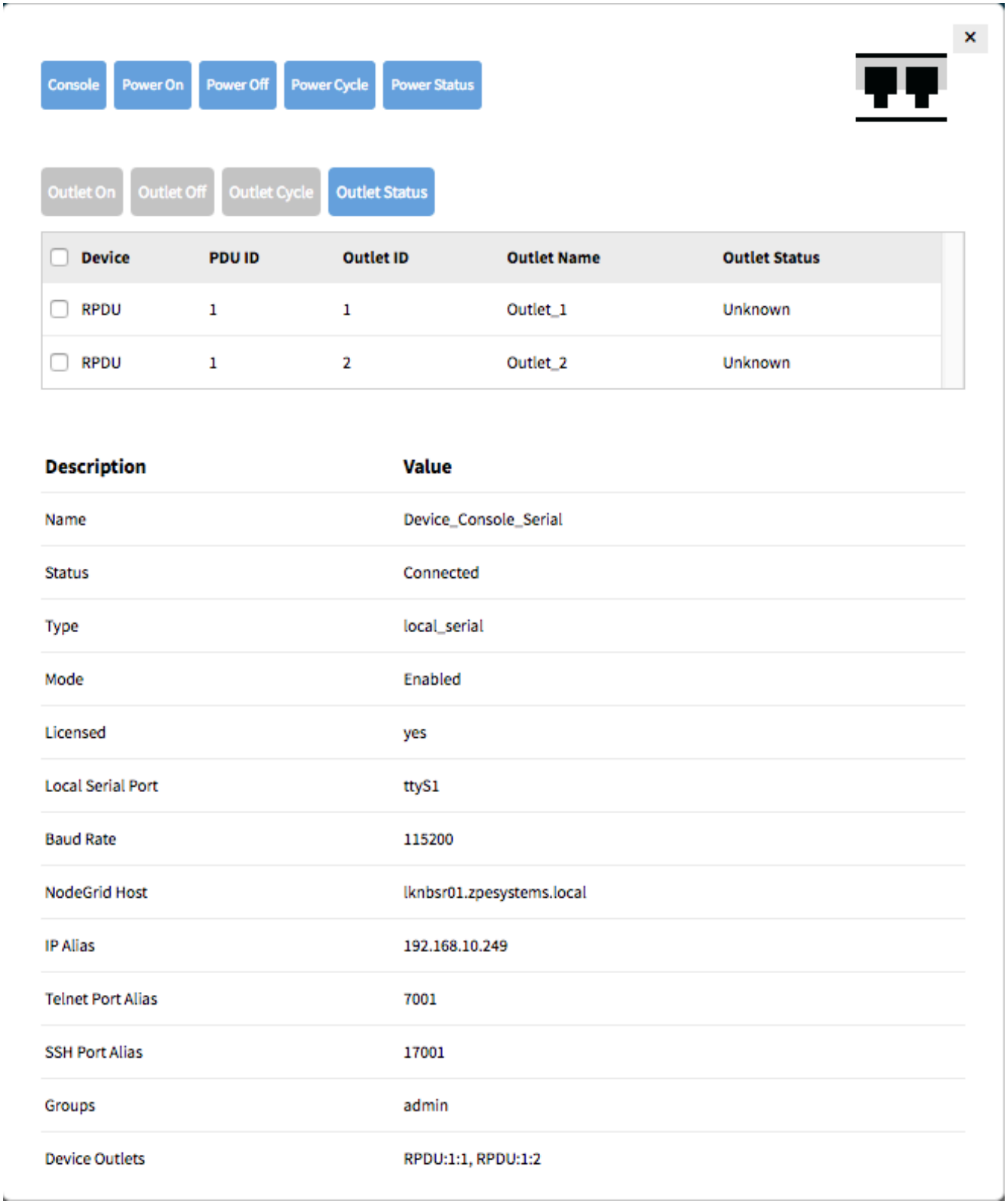
Nodegrid Platform によって維持される各デバイスには、システムに多数のデバイス情報が保存されています。ユーザーはこの情報を閲覧でき、システム内で検索することも可能です。これは、特定のターゲットを識別したい場合に特に便利です。

保存されている情報は、自動的に検出された値、デバイス設定中に設定された値、および管理者によってデバイスに関連付けられた追加情報を組み合わせたものです。

デバイス情報は、WebUI でターゲット名をクリックするか、CLI 内のデバイスに移動して、特定デバイスの Access ビューに表示できます。

デバイス情報の表示 - Web UI

- 移動先 `Access:: Table`
- ターゲット*名*をクリックすると、デバイスの詳細が表示されます



デバイス情報の表示 - CLI

- 移動先 `cd /access/`
- コマンド `show` を使用してデバイスの詳細を表示

```
[admin@nodegrid /]# cd /access/
[admin@nodegrid access]# show Device_Console_Serial/
name: Device_Console_Serial
status: Connected
```

デバイスビュー

WebUI は、ターゲットデバイスを表示・アクセスするための各種方法を提供します。デフォルトで、すべてのユーザーにテーブル ビューが表示され、ここからすべてのターゲットに簡単にアクセスすることができます。その他のビューも使用でき、アクセシビリティやデバイスのステータスの視覚化を向上させます。 次のビューを使用できます:

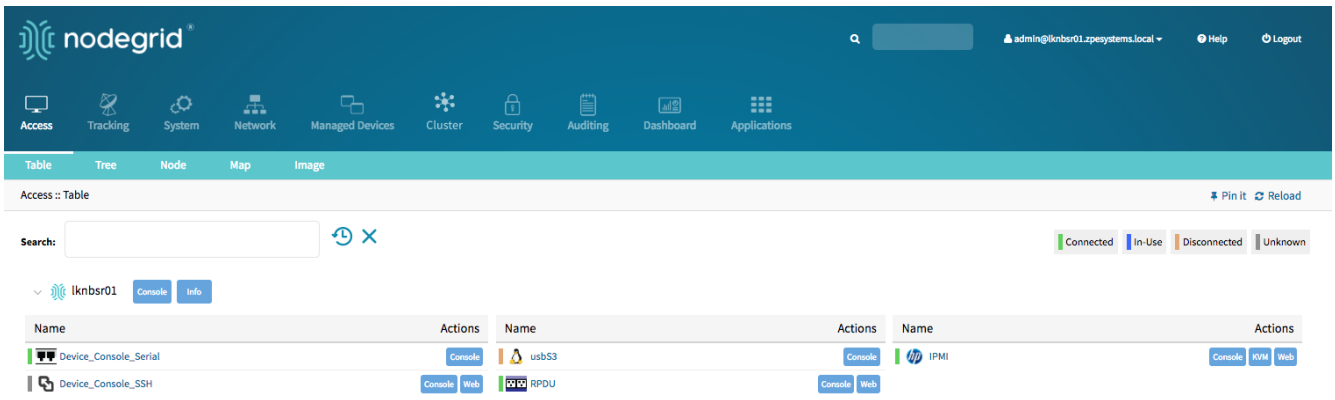
- テーブル ビュー
- ツリー ビュー
- ノードビュー
- マップビュー
- 画像ビュー

各ユーザーは、ログイン後に表示されるデフォルトのビューを変更できます。これには、ユーザーは設定ビューを開き、`Pin It`ボタンを使用します。 

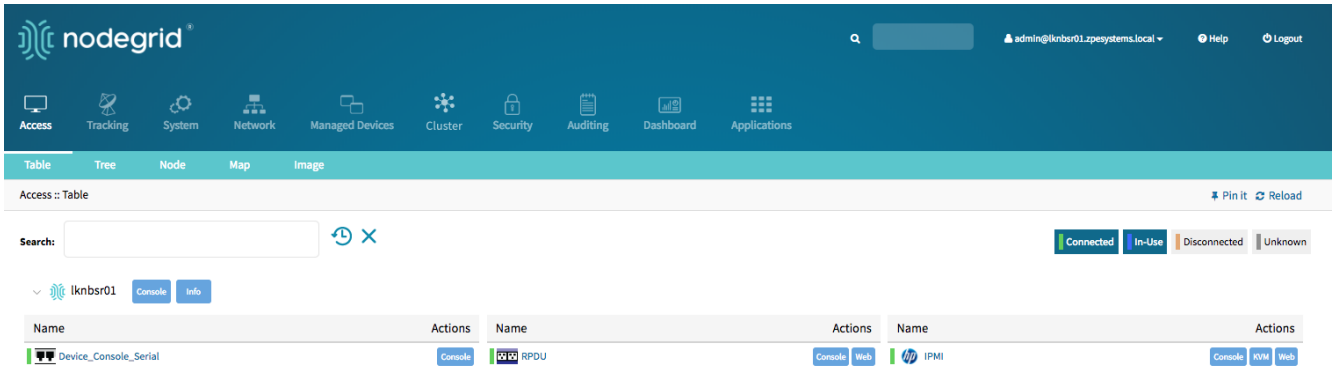
> 注: テーブル ビューは、CLI で使用できる唯一のビューです。

テーブル ビュー

テーブルビューで、すべてのターゲットデバイスとそのデバイスセッションに簡単にアクセスできます。これは、各デバイスの現在の状態を簡単に説明するテーブルビューを提供します。ビューには、ユニットに現在接続されているすべてのデバイスと、クラスタ機能を介して使用できる他のすべてのターゲットが表示されます。



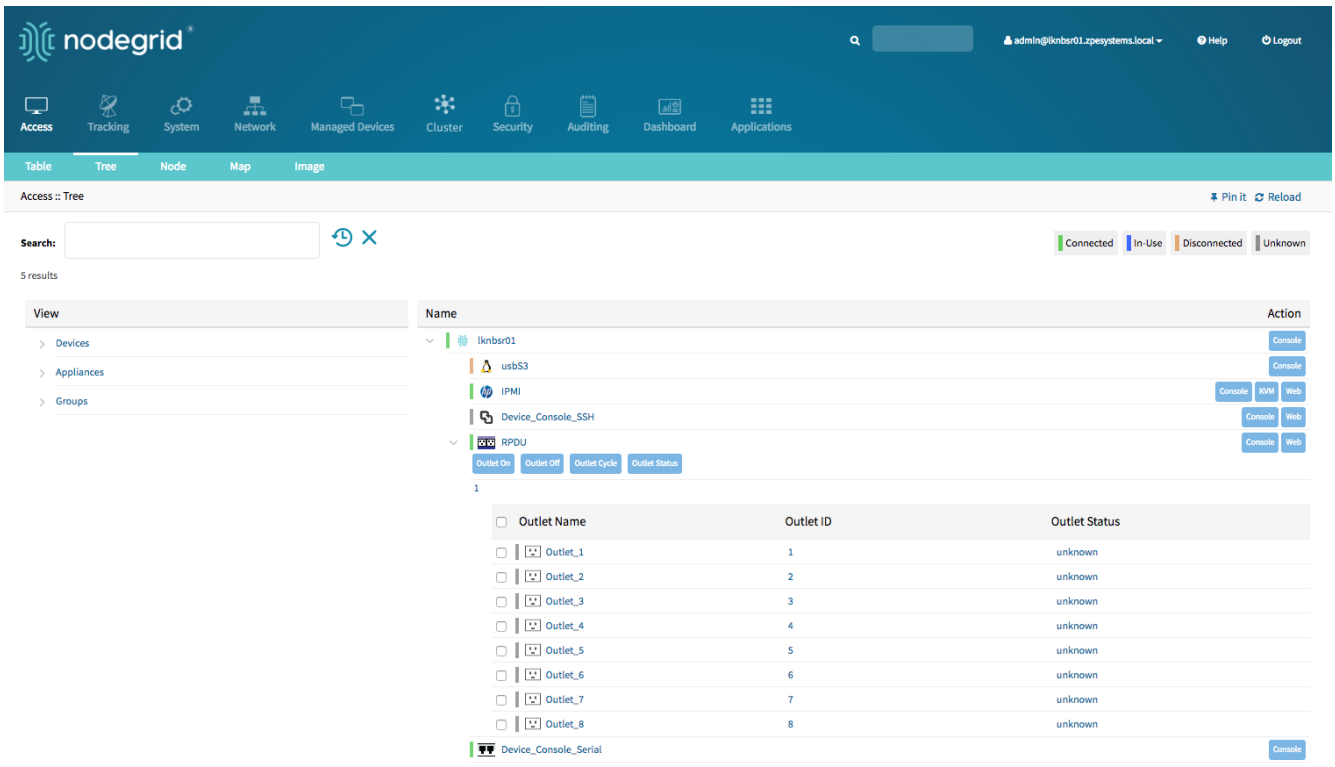
ビューでは、現在のデバイスのステータスやその他の検索条件で現在のリストをフィルタリングできます。 現在のデバイスのステータスでフィルタリングするには、右上隅のデバイスステータスアイコンをクリックします。以下の例では、接続状態 (接続中と使用中) でデバイスをフィルタリングしています



より高度な検索オプションは、検索フィールドから使用できます。詳細については、デバイスの検索を参照してください。

ツリー ビュー

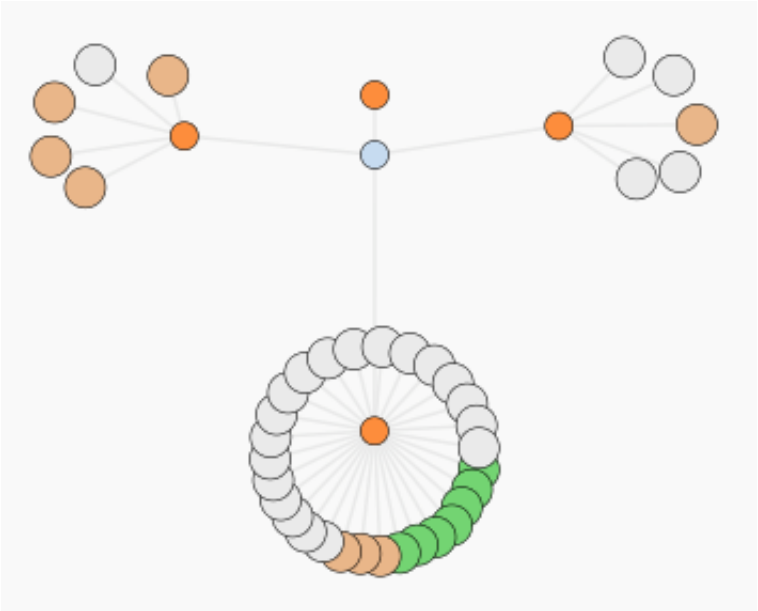
ツリー ビューには、Nodegrid セットアップの物理階層に基づくすべてのターゲットが表示され、各ターゲットの接続を開始できます。Nodegrid 名、都市名、データセンター名、行とラックなど、そのロケーションに基づいたターゲットデバイスに簡単にアクセスできます。*ビュー*セクションは、ロケーションとデバイスのタイプに基づくフィルタを提供します。



より高度な検索オプションは、検索フィールドから使用できます。詳細については、デバイスの検索を参照してください。

ノードビュー

ノードビューでは、接続されている Nodegrid ユニットの周囲にすべてのターゲットデバイスを配置し、クラスタ内のすべてのターゲットと Nodegrid ユニットの完全な概観が簡単に得られます。ビューでターゲットノードをクリックすると、ターゲットデバイス情報と接続にアクセスできます。

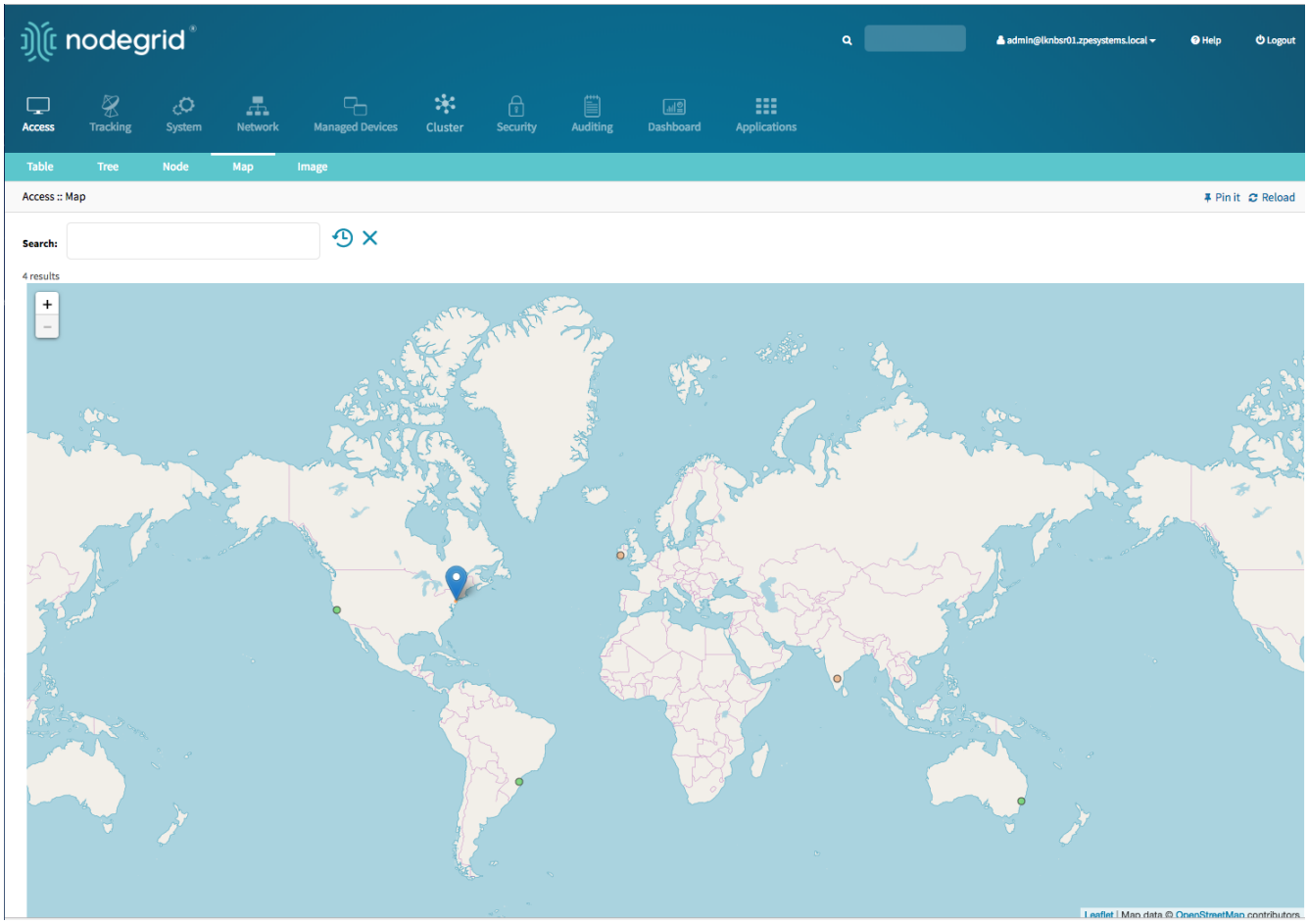


より高度な検索オプションは、検索フィールドから使用できます。詳細については、デバイスの検索を参照してください。

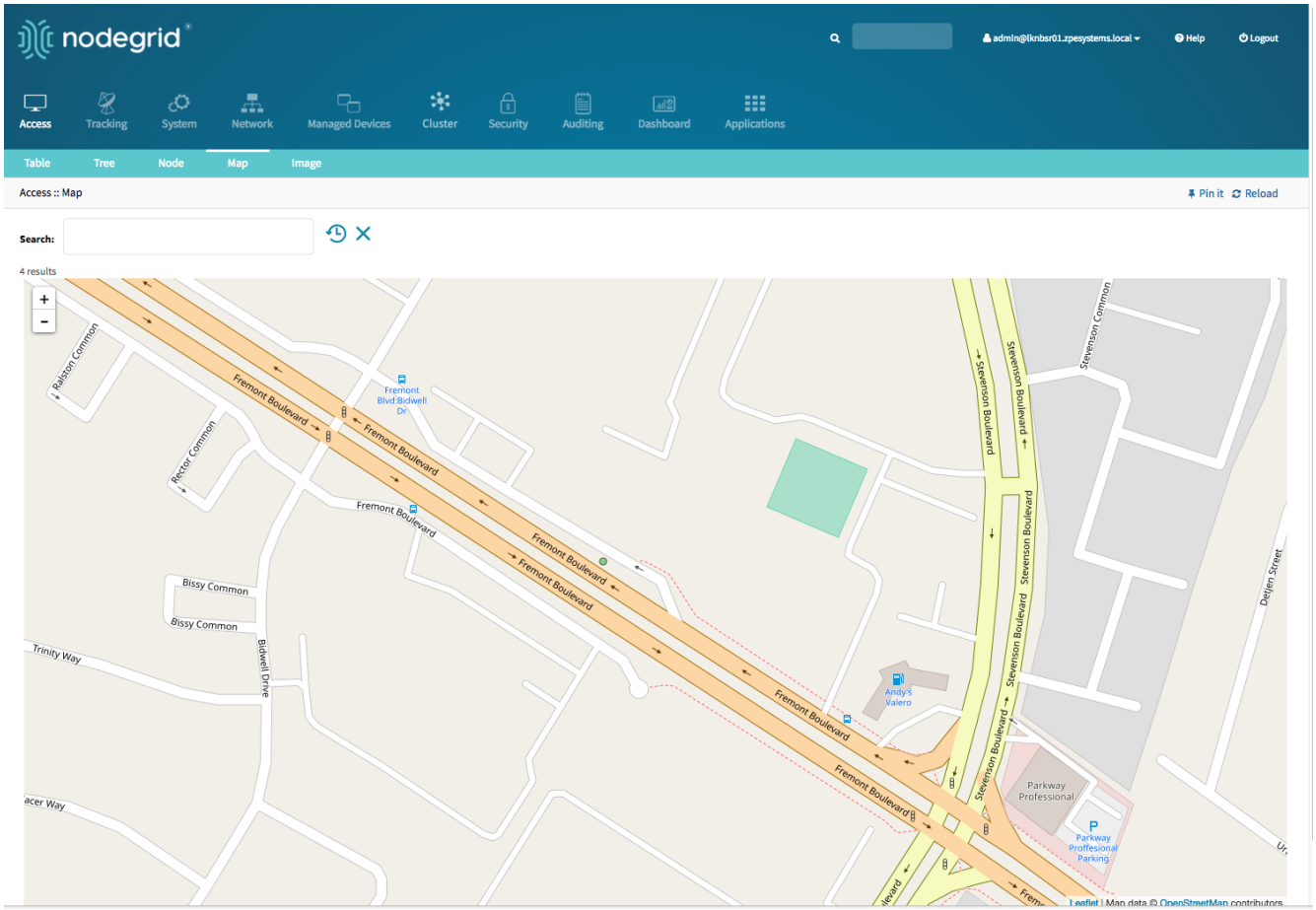
マップビュー

マップビューでは、グローバルマップ上のデバイスの現在のステータスを確認でき、 クラスタ内のすべてのターゲットと Nodegrid ユニットの完全な概観が得られます。マップビューは、正確な位置情報を建物レベルまで表示できます。ビューでは、ターゲットノードをクリックしてターゲットデバイスの情報と接続にアクセスできます。

グローバルビュー


nodegrid
admin@lknbsr01.zpesystems.local
Help
Logout
Access
Tracking
System
Network
Managed Devices
Cluster
Security
Auditing
Dashboard
Applications
Table
Tree
Node
Map
Image
Access :: Map
Pin it
Reload
Search:
4 results
Leaflet | Map data © OpenStreetMap contributors

ストリートビュー



より高度な検索オプションは、検索フィールドから使用できます。詳細については、デバイスの検索を参照してください。

画像ビュー

画像ビューでは、Nodegrid ユニット、ターゲット デバイス、および関連情報のカスタムビューを表示できます。プロフェッショナルサービスによる実装が必要です。詳細については、カスタマー サポート support@zpesystem.com にお問い合わせください。

検索

Nodegrid Platform は、簡単に検索でき、必要な情報やターゲットデバイスにアクセスできる、高度な検索機能を提供します。

デバイス検索

デバイス検索は、すべてのデバイスビューで使用でき、各ビューでターゲットデバイスを簡単に検索およびフィルタリングできます。

デバイス検索は、各ビューの左上隅にある検索フィールドと、CLI のアクセスメニューのコマンド `search` を使用して WebUI でアクセスできます。NodeIQ™自然言語検索では、カスタムフィールドを含むデバイスプロパティフィールドを検索できます。この機能は、スタンドアロンユニットの他、Cluster 設定内のすべての Nodegrid ユニット全体で自然に動作します。システムでは、デバイスの変更、新しく追加されたデ

バイス、およびバックグラウンドでのプロパティに関するすべての情報が自動で更新されます。

検索フィールドでは、以下のキーワードがサポートされています。

キーワード	説明
[検索文字列]	検索対象となる文字列の一部または全部を表す検索文字列
AND	複数の検索文字列を AND で組み合わせます
OR	複数の検索文字列を OR で組み合わせます。複数の検索文字列用のデフォルトの検索挙動
NOT	検索文字列に一致するすべてのターゲットは返されません
[フィールド名]	特定のフィールド名に対する検索文字列の制限を許可します

> 注: キーワード AND、OR および NOT は大文字と小文字を区別するため、「and」、「or」、「not」は検索文字列として識別されます。

標準フィールドデータとカスタムフィールドデータ (“管理者”グループなどのグループを含む)、IP アドレス、または特定デバイスを検索するには、以下の例に従ってください:

AND を使用した例

“PDU AND IPMI”


```
[admin@nodegrid search]# search "PDU AND IPMI"

search: PDU AND IPMI
results: 1 result
page: 1 of 1

[admin@nodegrid search]# show
  name   status   action
  ====   ======   ======
  IPMI   -
```

OR を使用した例

"PDU OR IPMI"


```
[admin@nodegrid access]# search "PDU OR IPMI"

search: PDU OR IPMI
results: 4 results
page: 1 of 1

[admin@nodegrid search]# show
  name                  status  action
  ====================  ======  ======
  IPMI                  -
  RPDU                  -
  Device_Console_SSH    -
  Device_Console_Serial -
```

"PDU IPMI"


```
[admin@nodegrid access]# search "PDU IPMI"

search: PDU IPMI
results: 4 results
page: 1 of 1

[admin@nodegrid search]# show
  name                  status  action
  ====================  ======  ======
  IPMI                  -
  RPDU                  -
  Device_Console_SSH    -
  Device_Console_Serial -
```

NOT の例

"PDU AND NOT IPMI"


```
[admin@nodegrid search]# search "PDU AND NOT IPMI"

search: PDU AND NOT IPMI
results: 3 results
page: 1 of 1

[admin@nodegrid search]# show
  name                  status  action
  ====================  ======  ======
  RPDU                  -
  Device_Console_SSH    -
  Device_Console_Serial  -
```

フィールド名を使用した例

"name:PDU"


```
[admin@nodegrid search]# search "name:PDU"

search: name:PDU
results: 1 result
page: 1 of 1

[admin@nodegrid search]# show
  name  status  action
  ====  ======  ======
  RPDU  -
```

グローバル検索

グローバル検索オプションは、WebUI で使用できます。[検索] フィールドは、現在のユーザー情報とログアウトオプションの横にある画面の上部にあります。グローバル検索は、デバイス検索と同様に機能し、同じキーワードをサポートします。検索はすべての画面で利用でき、すべてのターゲットデバイスとターゲットセッションに簡単にアクセスできます。

デバイス管理 (管理対象デバイス)

*管理対象デバイス*セクションでは、ターゲットデバイスの設定、作成、削除が可能です。Nodegrid Platformは、シリアル、USB、またはネットワーク経由で接続されるターゲットデバイスをサポートします。現在、ネットワークベースのデバイス Telnet、SSH、HTTP/S、IMPI バリエーション、および SNMP では、次のプロトコルがサポートされています。

有効化や作成、新しいターゲットデバイスを使用するための各種オプションがあります。手動で有効化/作成、または検出が可能です。

システムに追加された各管理対象デバイスは、プールからのライセンスを 1 つ使用します。各ユニットは、物理ポートの数をカバーするのに十分な永久ライセンスを備えて出荷されているので、物理ポートを使用するためにそれ以上のライセンスは必要ありません。追加デバイスの管理を可能にするために、ユニットにライセンスを追加することができます。ライセンスの有効期限が切れたりシステムから削除されたりすると、ライセンスの合計を超えるデバイスは、ステータスが “非ライセンス”の状態に変更されます。その情報はシステムに保持されますが、ライセンスのないデバイスはアクセスページに表示されず、ユーザーはそれらに接続できません。アクセスページにはライセンスされたデバイスのみが表示され、アクセスと管理に使用できます。*管理対象デバイス*ビューの右上隅に、システム内の全ライセンス、使用中の全ライセンス、使用可能な全ライセンスが表示されます。詳細は、ライセンスを参照してください。

Nodegrid Platform は、以下のタイプの管理対象デバイスをサポートします。

- RS232 プロトコルを使用するコンソール接続。Nodegrid Console Server および Nodegrid Services Router ファミリーでサポートされています。
- 以下を使用するサービスプロセッサーデバイス:
    - IPMI 1.5
    - IPMI 2.0
    - HP iLO
    - Oracle/SUN iLOM
    - IBM IMM
    - Dell DRAC
    - Dell iDRAC
- ssh プロトコルを使用したConsole Server接続
- 以下を使用したConsole Server接続
    - Vertiv ACS クラシック ファミリー
    - Vertiv ACS6000 ファミリー
    - Lantronix Console Serverファミリー
    - Opengear Console Serverファミリー
    - Digi Console Serverファミリー
    - Nodegrid Console Serverファミリー
- 以下を使用した KVM (キーボード、ビデオ、マウス) スイッチ
    - Vertiv DSR ファミリー
    - Vertiv MPU ファミリー

    - Aten Enterprise KVM ファミリー
    - Raritan KVM ファミリー
    - ZPE システム KVM モジュール
- ラック PDU
    - APC
    - CPI
    - Cyberpower
    - Baytech
    - Eaton
    - Enconnex
    - Vertiv (PM3000 および MPH2)
    - Raritan
    - Ritttal
    - Servertech
- Cisco UCS
- Netapp
- Infrabox
- 以下からの仮想マシンセッション
    - VMWare
    - KVM
- センサー
    - ZPE システムの温度と湿度センサー

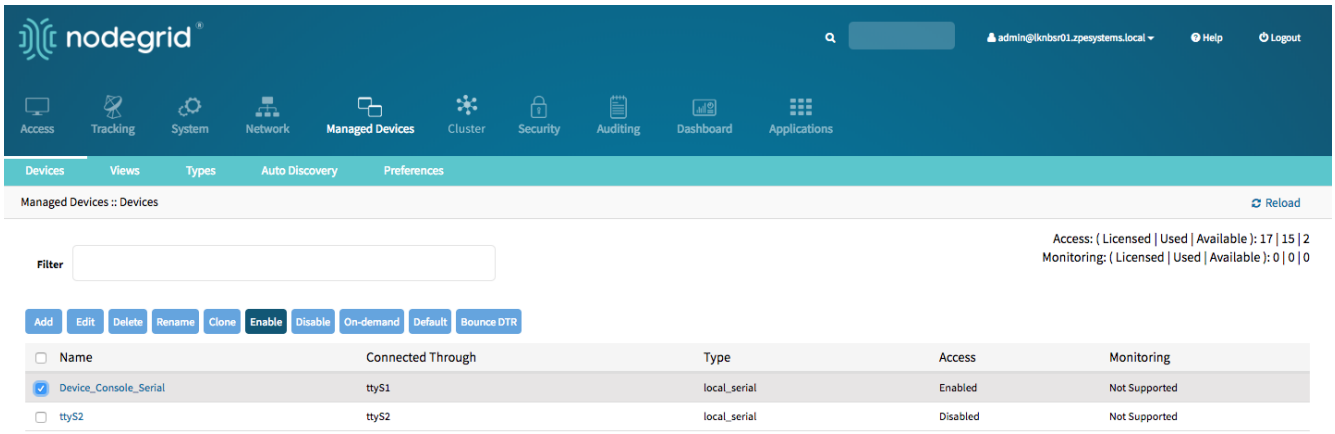
管理対象デバイスの設定

新規デバイスは、*デバイス*メニューに追加できます。メニューには、次のオプションを提供します:

- `enable`、`disable`、`reset`、および物理ポートに接続されたデバイス`configure`
- `add`、`configure`、およびネットワーク接続を介して接続されたデバイス`delete`
- `rename` 既存のデバイス/ポート
- `clone` 既存のデバイス
- 接続モードをすばやく変更するには `On-Demand`
- シリアルポートの信号`Bounce DTR`を送信するには

これらのタスクのいずれかを実行するには、そのボタンをクリックする、まずデバイスを選択し WebUI のボタンをクリックする、または CLI のコマンドを使用します。

WebUI 有効化ポート 1 の例



CLI 名前変更ポート 2 の例

```
[admin@nodegrid devices]# rename ttyS2
[admin@nodegrid {devices}]# set new_name=Port2
[admin@nodegrid {devices}]# commit
```

シリアルデバイス

Nodegrid Platform は、利用可能なシリアルおよびUSBインターフェースを介して、RS-232 シリアル接続をサポートします。ポートは自動検出されて [デバイス] メニューに表示され、直接使用することができます。ターゲットデバイスへのアクセスを提供するには、各ポートを有効にして設定する必要があります。

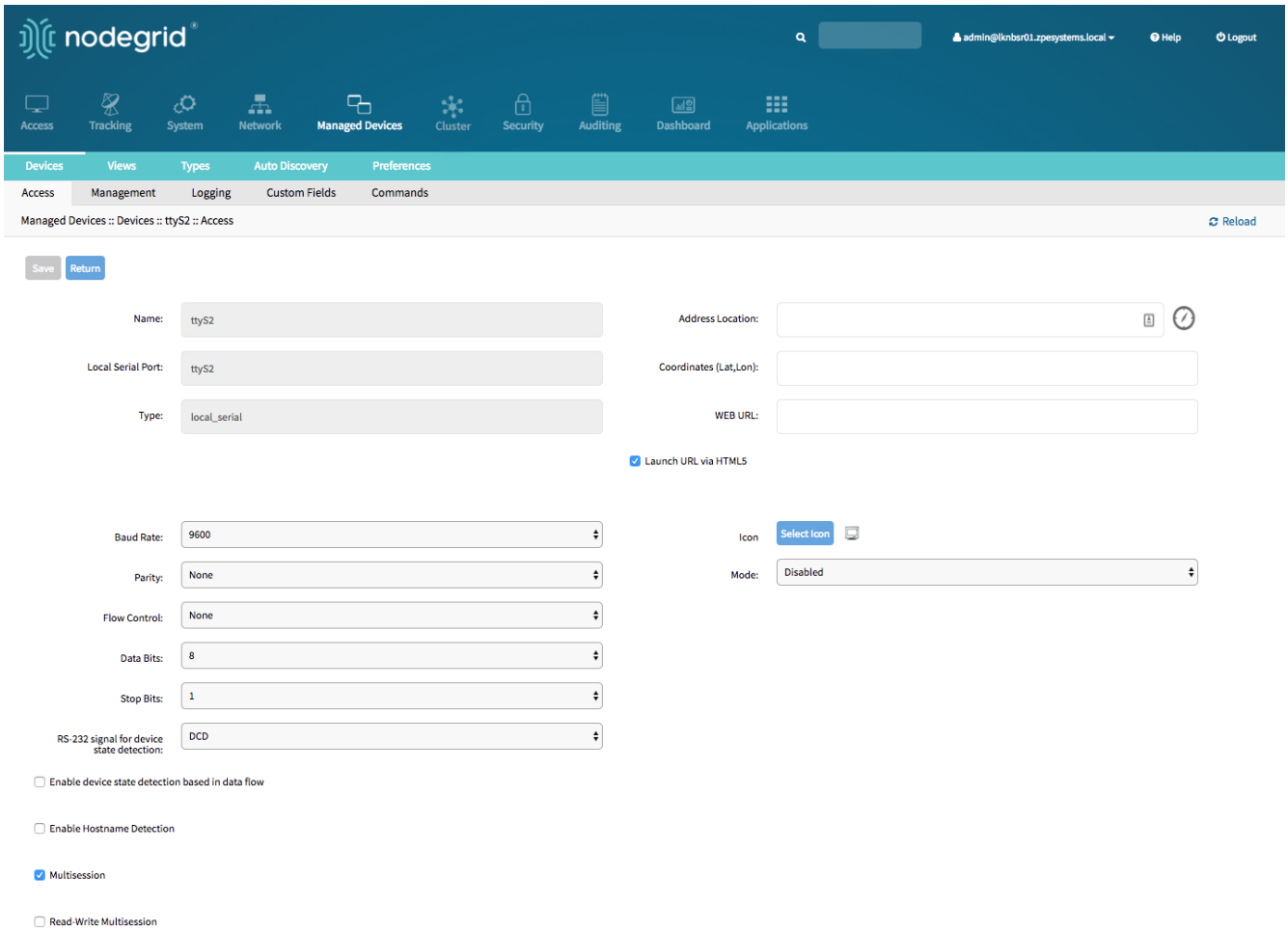
Nodegrid ポートを設定する前に、ターゲットデバイスのコンソールポートの設定を製造元に確認してください。大半のデバイスは、ポートのデフォルトである 9600,8,N,1 の設定を使用します

Nodegrid Console Server S シリーズは、ピン配置 (レガシーおよびCisco) と接続速度を自動的に検出することで、設定プロセスを簡素化する高度な自動検出をサポートします。

Serial Devices の設定 - WebUI

- 移動先 `Managed Devices:: Devices`
- ポートをクリックするか、ポートを選択して `Edit` をクリックします。複数ポートを選択可能
- 次の値を確認します:
    - `Baud Rate` ターゲットデバイスの設定に一致する正しい速度に設定されているかどうか、`Auto`
    - `Parity` 指定可能な値: *なし* (デフォルト)、*奇数*、または*偶数*
    - `Flow Control` 指定可能な値: なし (デフォルト)、ソフトウェア、ハードウェア
    - `Data Bits` 指定可能な値: *5*、*6*、*7*、*8* (デフォルト)
    - `Stop Bits` 指定可能な値: *1*
    - `RS-232 signal for device state detection` 指定可能な値: *DCD* (デフォルト)、*なし*、*CTS*
    - `Mode` 指定可能な値: *有効*、*オンデマンド*、*無効*

- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください



シリアルデバイスの設定 - CLI

- 移動先 `/settings/devices`
- ポート名のコマンド `edit` を使用して、ポート設定を変更します。複数ポートを定義可能
- コマンド `show` を使用して、現在の値を表示f
- コマンド `set` を使用して以下の値を調整します:
  - `baud_rate` ターゲットデバイスの設定に一致する正しい速度に設定されているかどうか、`Auto`
  - `parity` 指定可能な値: *なし* (デフォルト)、*奇数*、または*偶数*
  - `flow_control` 指定可能な値: *なし* (デフォルト)、*ソフトウェア*、*ハードウェア*
  - `data_bits` 指定可能な値: *5*、*6*、*7*、*8* (デフォルト)
  - `stop_bits` 指定可能な値: *1*
  - `rs-232_signal_for_device_state_detection` 指定可能な値: *DCD* (デフォルト)、*なし*、*CTS*
  - `mode` 指定可能な値: *有効*、*オンデマンド*、*無効*
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください

- `commit`コマンドで、設定を定義します。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# edit ttyS2
[admin@nodegrid {devices}]# show
name: ttyS2
type: local_serial
address_location =
coordinates =
web_url =
launch_url_via_html5 = yes
baud_rate = 9600
parity = None
flow_control = None
data_bits = 8
stop_bits = 1
rs-232_signal_for_device_state_detection = DCD
enable_device_state_detection_based_in_data_flow = no
enable_hostname_detection = no
multisession = yes
read-write_multisession = no
icon = terminal.png
mode = disabled
skip_authentication_to_access_device = no
escape_sequence = ^Ec
power_control_key = ^O
show_text_information = yes
enable_ip_alias = no
enable_second_ip_alias = no
allow_ssh_protocol = yes
ssh_port =
allow_telnet_protocol = yes
telnet_port = 7002
allow_binary_socket = no
data_logging = no
[admin@nodegrid {devices}]# set mode=enabled baud_rate=Auto
[admin@nodegrid {devices}]# commit
```

サービスプロセッサデバイス

Nodegrid Platform は、 IPMI 1.5、IMPI 2.0、Hewlett Packard ILO's、Oracle/SUN iLOM's、IBM IMM's、Dell DRAC および iDRAC などの複数の IPMI ベースのサービスプロセッサをサポートしています。

Nodegrid でこれらのデバイスを管理するには、ターゲットデバイスへの有効なネットワーク接続が必要です。これは、Nodegrid 自体の専用ネットワークインターフェースを介して、または既存のネットワーク接続を介して行うことができます。

Nodegrid は、サービスプロセッサの次の機能をサポートします。

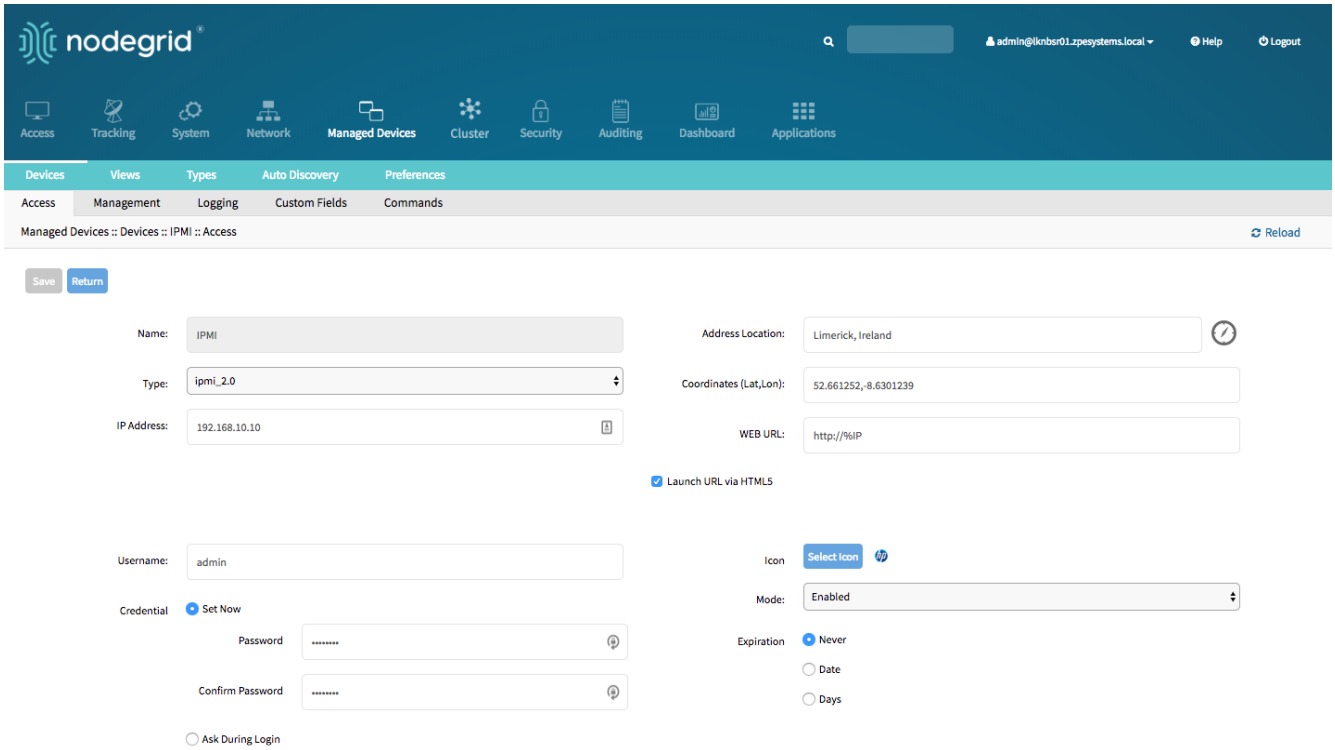
> 注: サービスプロセッサの機能によっては、一部の機能を使用できない場合があります。

- Serial Over LAN (SOL)
- Web インターフェース
- KVM セッション
- 仮想メディアのサポート
- 動力管理
- データロギング
- イベントロギング
- ラック PDU を介した電源制御

SOL 経由でコンソールにアクセスする場合は、サーバ上で BIOS コンソールリダイレクトと OS コンソールリダイレクト (通常 Linux OS 用) も有効にする必要があります。

サービスプロセッサデバイスの追加 - WebUI

- `Managed Devices:: Devices` に移動し、
- ボタン `Add` をクリックして、デバイスをシステムに追加します。この例の目的として、以下の情報を提供します:
- 追加するサーバの名前を入力します。
- サービスプロセッサの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- `Type` フィールドで、使用中のサービスプロセッサに一致するタイプを選択します。指定可能な値: *ipmi1.5*、*ipmi2.0*、*ilo*、*ilom*、*imm*、*drac*、idrac6
- ログイン時にユーザー資格情報を提供する場合、サービスプロセッサの `username` と `password` を入力するか、 `Ask During Login` オプションを選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



サービスプロセッサデバイスの追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値: *ipmi1.5*、*ipmi2.0*、*ilo*、*ilom*、*imm*、*drac*、idrac6
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、サービスプロセッサの `password`、または `Ask During Login` オプションを選択します
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=IPMI
[admin@nodegrid {devices}]# set type=ipmi_2.0
[admin@nodegrid {devices}]# set ip_address=192.168.10.11
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
```

```
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

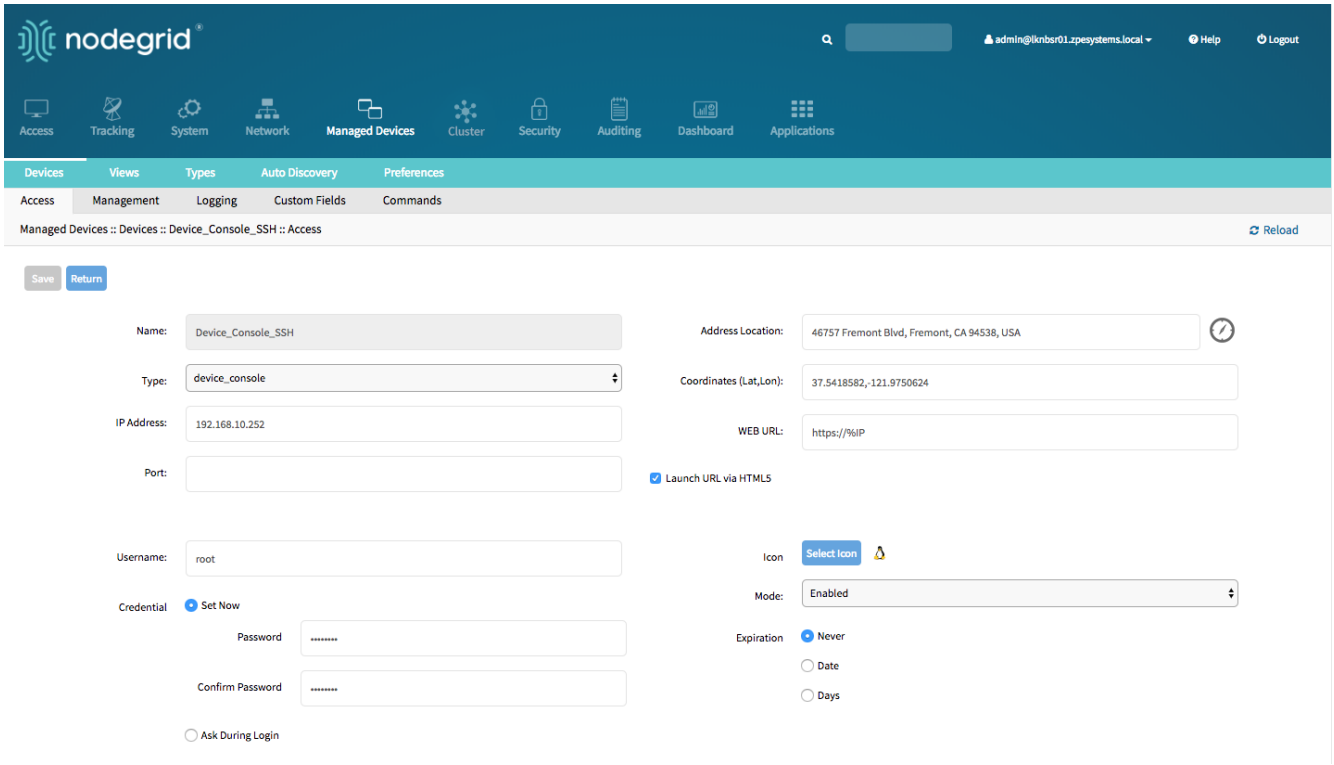
SSH を備えたデバイス

このソリューションは、SSH を介してターゲットデバイスの管理をサポートします。Nodegrid は、これらのデバイスの以下の機能に対応します:

- コンソールセッション
- データロギング
- カスタムコマンド
- Web セッション
- ラック PDU を介した電源制御

SSH でデバイスを追加 - WebUI

- `Managed Devices:: Devices` へ移動し、
- ボタン `Add` をクリックして、デバイスをシステムに追加します。
- 追加するサーバーの名前を入力します。
- デバイスの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- `Type` フィールドで、使用中の ssh または telnet に一致するタイプを選択します。指定可能な値: デバイスコンソール_
- ログイン時にユーザー資格情報を提供する場合、ssh サーバの `username` と `password` を入力するか、オプション `Ask During Login` を選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



SSH を備えたデバイスを追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値: デバイスコンソール_
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、または `Ask During Login` オプションを選択します
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Device_Console_SSH
[admin@nodegrid {devices}]# set type=device_console
[admin@nodegrid {devices}]# set ip_address=192.168.10.252
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
```

```
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

Console Server

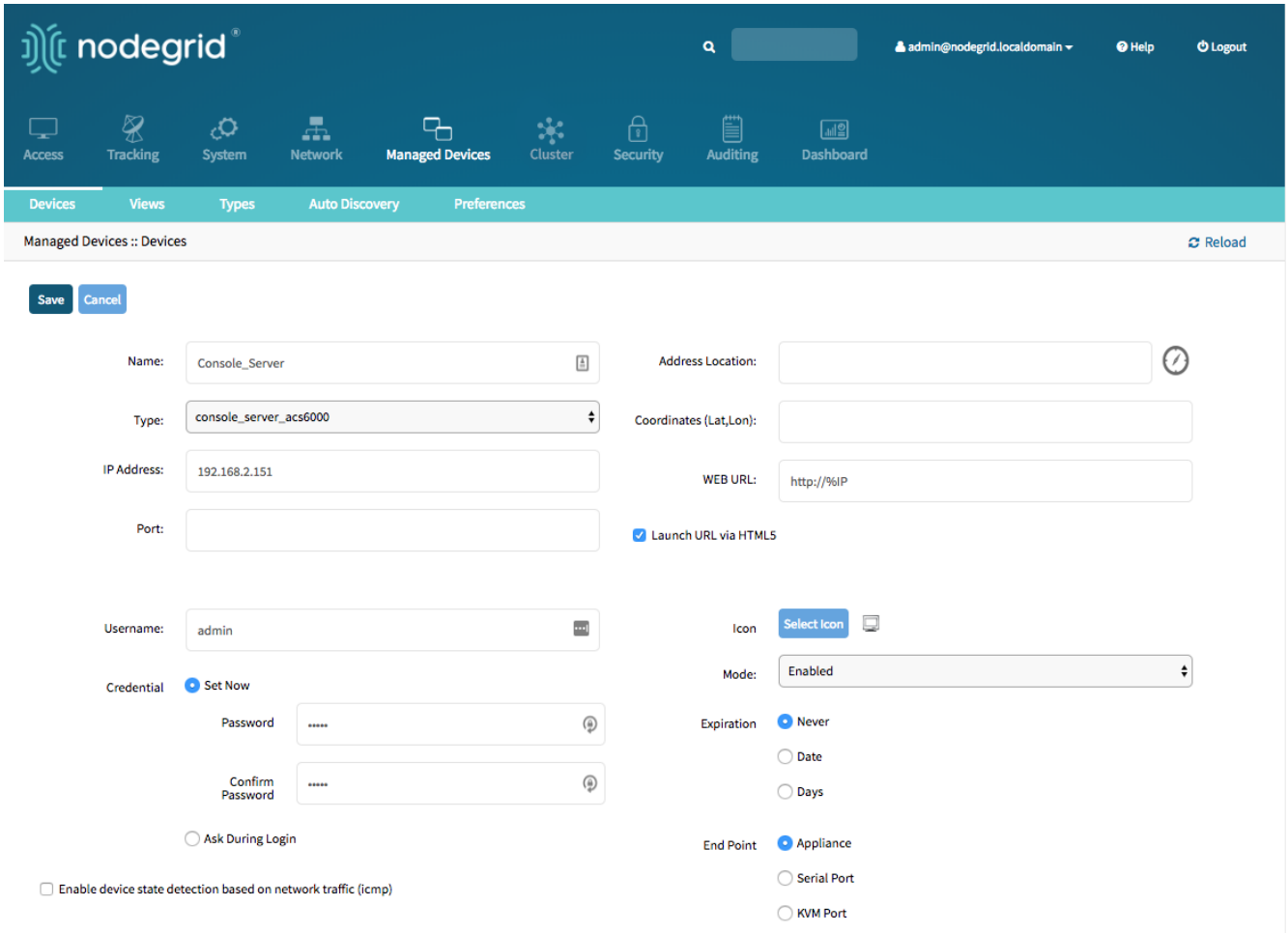
このソリューションは、Avocent および Servertech の Console Server を含む、さまざまなベンダーの複数のサードパーティ製 Console Server をサポートします。これらのデバイスは Nodegrid Platform に追加でき、接続されたターゲットを Nodegrid アプライアンスに直接接続したかのように使用できます。 サードパーティの Console Server は、2ステップで追加できます。最初のステップでサードパーティアプライアンスが Nodegrid に追加され、次のステップで有効なすべてのポートが Platform に追加されます。

Nodegrid は、これらのデバイスの以下の機能に対応します:

- コンソールセッション
- データロギング
- カスタムコマンド
- Web セッション
- ラック PDU を介した電源制御

Console Server の追加 - WebUI

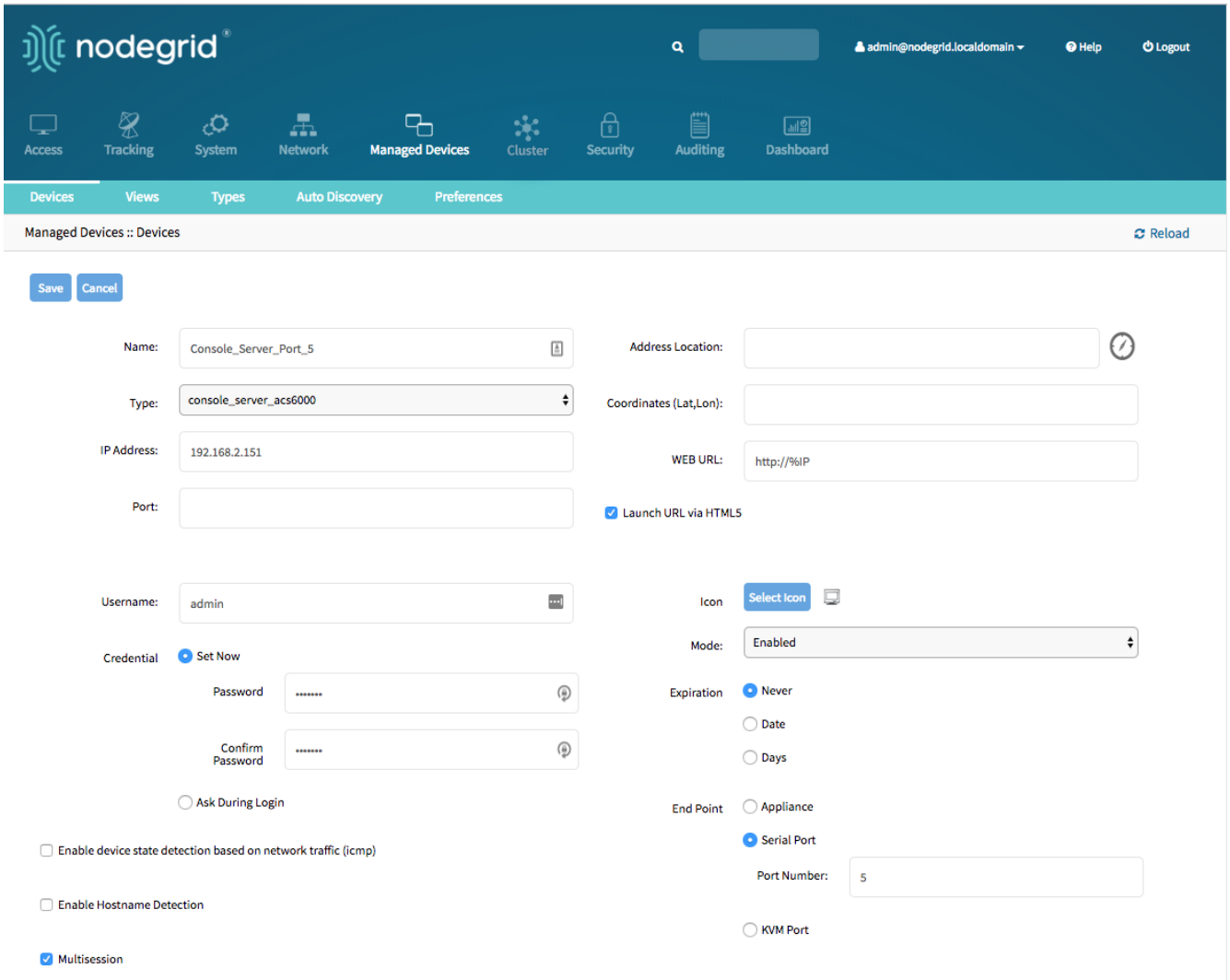
- `Managed Devices:: Devices` へ移動し、
- `Add` ボタンをクリックして、デバイスをシステムに追加します。
- 追加する Console Server の名前を入力します。
- Console Server の IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- フィールド `Type` で、Console Server に一致するタイプを選択します。 指定可能な値: *Console Server_nodegrid*、*Console Server_acs*、*Console Server_acs6000*、*Console Server_lantronix*、*Console Server_opengear*、Console Server_digicp_
- ログイン時にユーザー資格情報を提供する場合、Console Server の `username` と `password` を入力するか、オプション `Ask During Login` を選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



Console Server ポートの追加 - WebUI

- `Managed Devices:: Devices` へ移動し、
- `Add` ボタンをクリックして、デバイスをシステムに追加します。
- 追加する Console Server ポートの名前を入力します。
- Console Server の IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- フィールド `Type` で、Console Server に一致するタイプを選択します。指定可能な値:Console Server_nodegrid、Console Server_acs、Console Server_acs6000、Console Server_lantronix、Console Server_opengear、*Console Server_digicp*
- ログイン時にユーザー資格情報を提供する場合、Console Server の `username` と `password` を入力するか、オプション `Ask During Login` を選択します
- `End Point` *シリアルポート*として選択し、ポート番号を入力します
- [保存] ボタンをクリックします。

> 注: ポートは自動で検出・追加されます。詳細は、自動検出セクションを参照してください



Console Server の追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値: *console_server_acs*, *console_server_acs6000*,*console_server_lantronix*,*console_server_opengear*,console_server_digicp
    _
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、またはオプション `Ask During Login` を選択します
  - `endpoint` アプライアンスとして定義する必要があります
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Console_Server
[admin@nodegrid {devices}]# set type=console_server_acs6000
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set end_point = appliance
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

Console Server ポートの追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値: *console_server_acs*, *console_server_acs6000*,*console_server_lantronix*,*console_server_opengear*,console_server_digicp
    _
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、または `Ask During Login` オプションを選択します
  - `endpoint` *シリアル*ポート_として定義する必要があります
  - `port_number` ポート番号として定義する必要があります
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

> 注: ポートは自動で検出・追加されます。詳細は、自動検出セクションを参照してください

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Console_Server_Port_5
[admin@nodegrid {devices}]# set type=console_server_acs6000
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set end_point = serial_port
[admin@nodegrid {devices}]# set port_number = 5
```

```
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

KVM スイッチ

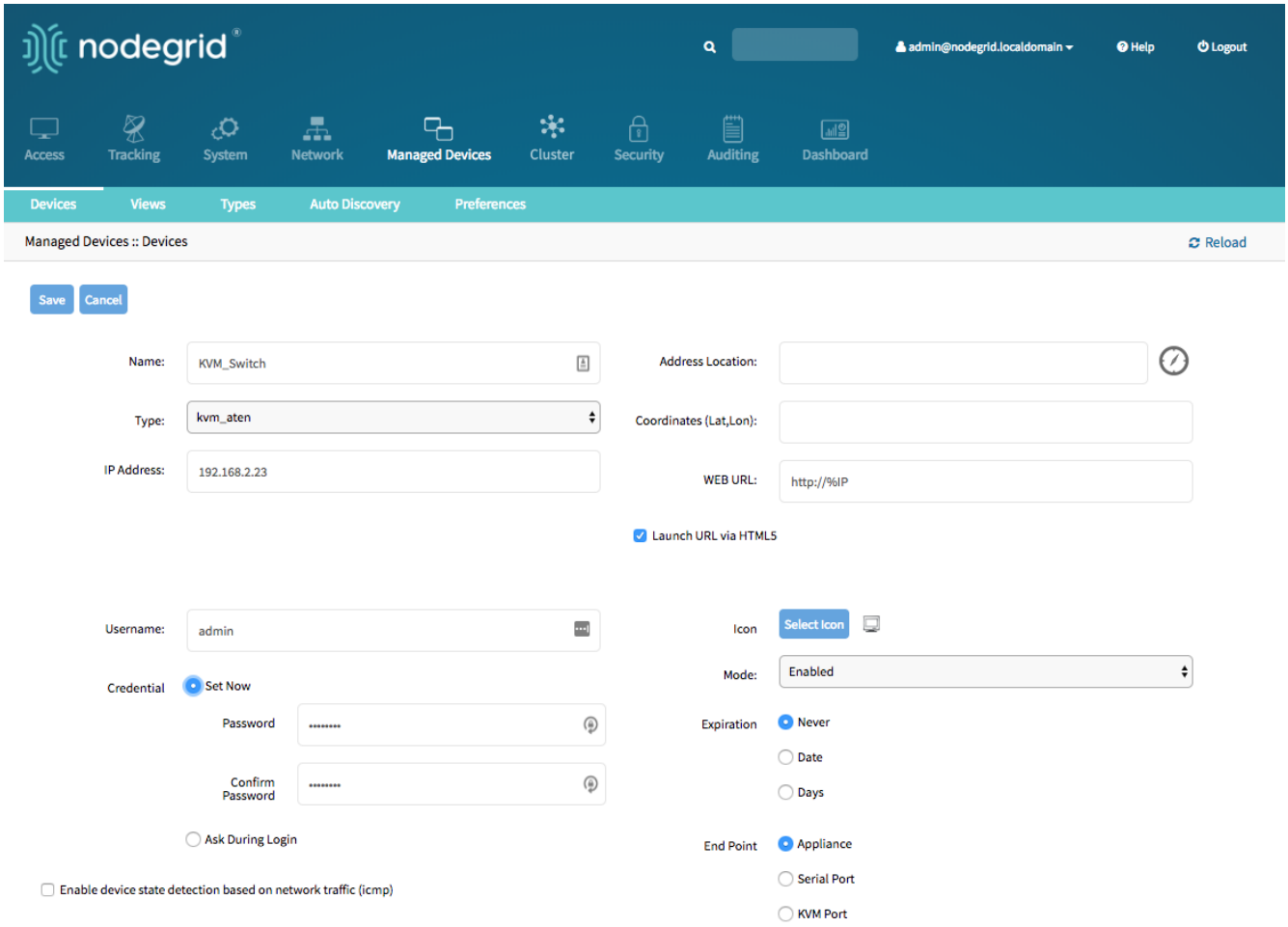
このソリューションは、Avocent および Raritan の製品を含む、さまざまなベンダーの複数のサードパーティ KVM スイッチをサポートします。これらのデバイスは Nodegrid Platform に追加でき、接続されたターゲットを Nodegrid アプライアンスに直接接続したかのように使用できます。 サードパーティの KVM スイッチは、2ステップで追加できます。最初のステップでサードパーティアプライアンスが Nodegrid に追加され、次のステップで有効なすべてのポートが Platform に追加されます。

Nodegrid は、これらのデバイスの以下の機能に対応します:

- KVMセッション
- Web セッション
- ラック PDU を介した電源制御

KVM スイッチの追加 - WebUI

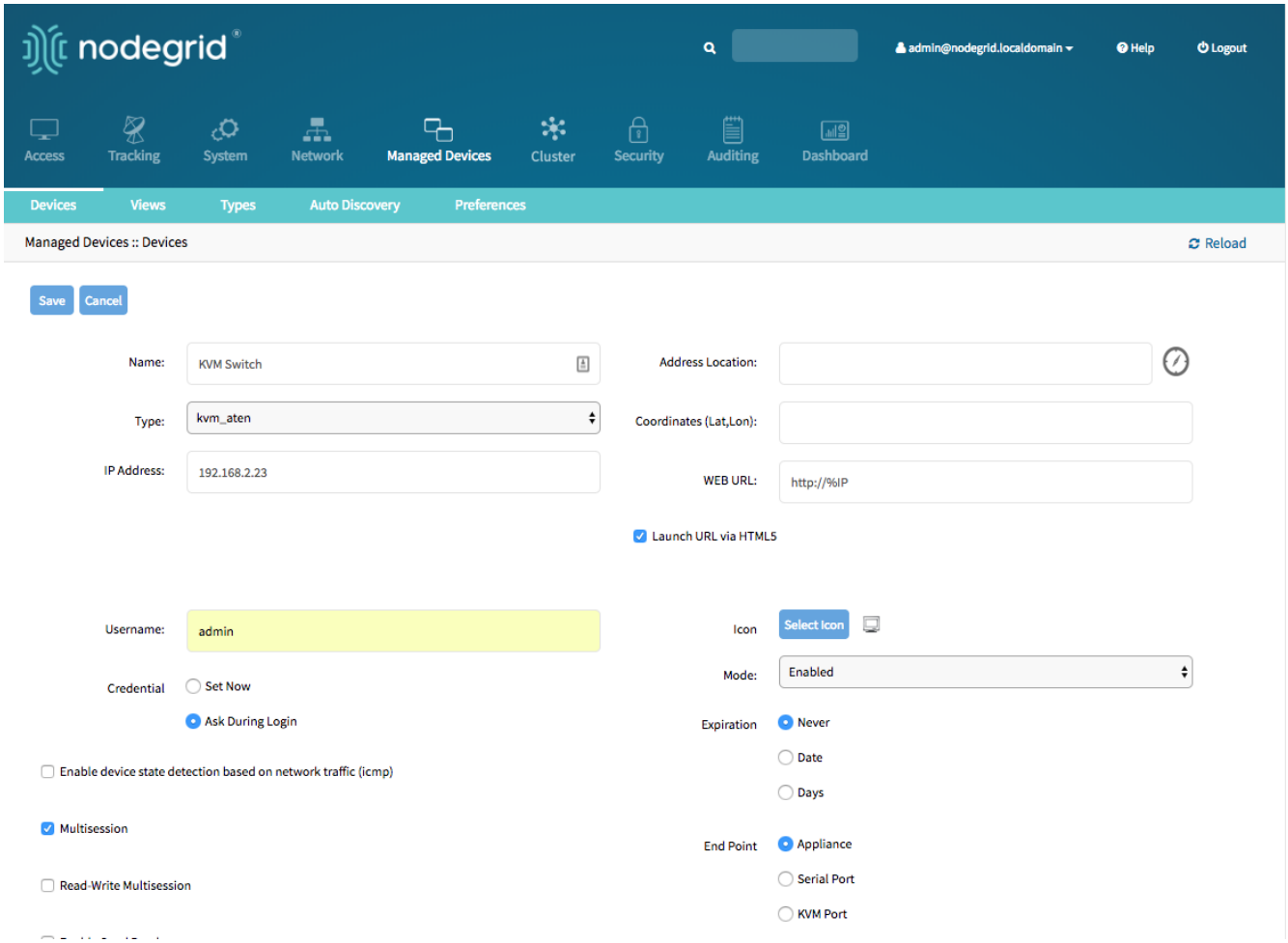
- `Managed Devices:: Devices`へ移動し、
- `Add`ボタンをクリックして、デバイスをシステムに追加します。
- 追加する KVM スイッチの名前を入力します。
- KVM スイッチの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- フィールド`Type`で、KVM スイッチに一致するタイプを選択します。指定可能な値: *kvm_dsr*、*kvm_mpu*、*kvm_aten*、*kvm_raritan*
- ログイン時にユーザー資格情報を提供する場合、KVM スイッチの`username`と`password`を入力するか、オプション`Ask During Login`を選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



KVM スイッチポートの追加 - WebUI

- `Managed Devices:: Devices` へ移動し、
- ボタン `Add` をクリックして、デバイスをシステムに追加します。
- 追加する KVM スイッチポートの名前を入力します。
- KVM スイッチの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- フィールド `Type` で、KVM スイッチに一致するタイプを選択します。指定可能な値: *kvm_dsr*、*kvm_mpu*、*kvm_aten*、*kvm_raritan*
- ログイン時にユーザー資格情報を提供する場合、KVM スイッチの `username` と `password` を入力するか、オプション `Ask During Login` を選択します
- `End Point` *KVM ポート*として選択し、ポート番号を入力します
- [保存] ボタンをクリックします。

> 注: ポートは自動で検出・追加されます。詳細は、自動検出セクションを参照してください



KVM スイッチの追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値: *kvm_dsr*、*kvm_mpu*、*kvm_aten*、*kvm_raritan*
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、または `Ask During Login` オプションを選択します
  - `endpoint` *アプライアンス*として定義する必要があります
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=KVM_Switch
[admin@nodegrid {devices}]# set type=kvm_aten
```

```
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set end_point = appliance
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

KVM スイッチポートの追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値: *kvm_dsr*、*kvm_mpu*、*kvm_aten*、*kvm_raritan*
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、または `Ask During Login` オプションを選択します
  - `endpoint` *シリアルポート_*として定義する必要があります
  - `port_number` ポート番号として定義する必要があります
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

> 注: ポートは自動で検出・追加されます。詳細は、自動検出セクションを参照してください

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Console_Server_Port_5
[admin@nodegrid {devices}]# set type=kvm_aten
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set end_point = kvm_port
[admin@nodegrid {devices}]# set port_number = 1
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin
```

```
[admin@nodegrid {devices}]# commit
```

ラック PDU

このソリューションは、APC、Avocent、Baytech、CPI、Cyberpower、Eaton、Enconnex、Geist、Liebert、Raritan、Rittal、Servertech の製品を含む、さまざまなベンダーの複数のサードパーティ製ラック PDU をサポートしています。これらのデバイスは、NodegridPlatform に追加してラック PDU に接続できます。ラック PDU が機能をサポートしている場合は、電源制御が可能です。 次に、コンセントを特定のターゲットデバイスに関連付けることが可能です。これにより、ユーザーはこのターゲットデバイスの特定の電源コンセントを直接制御できます。

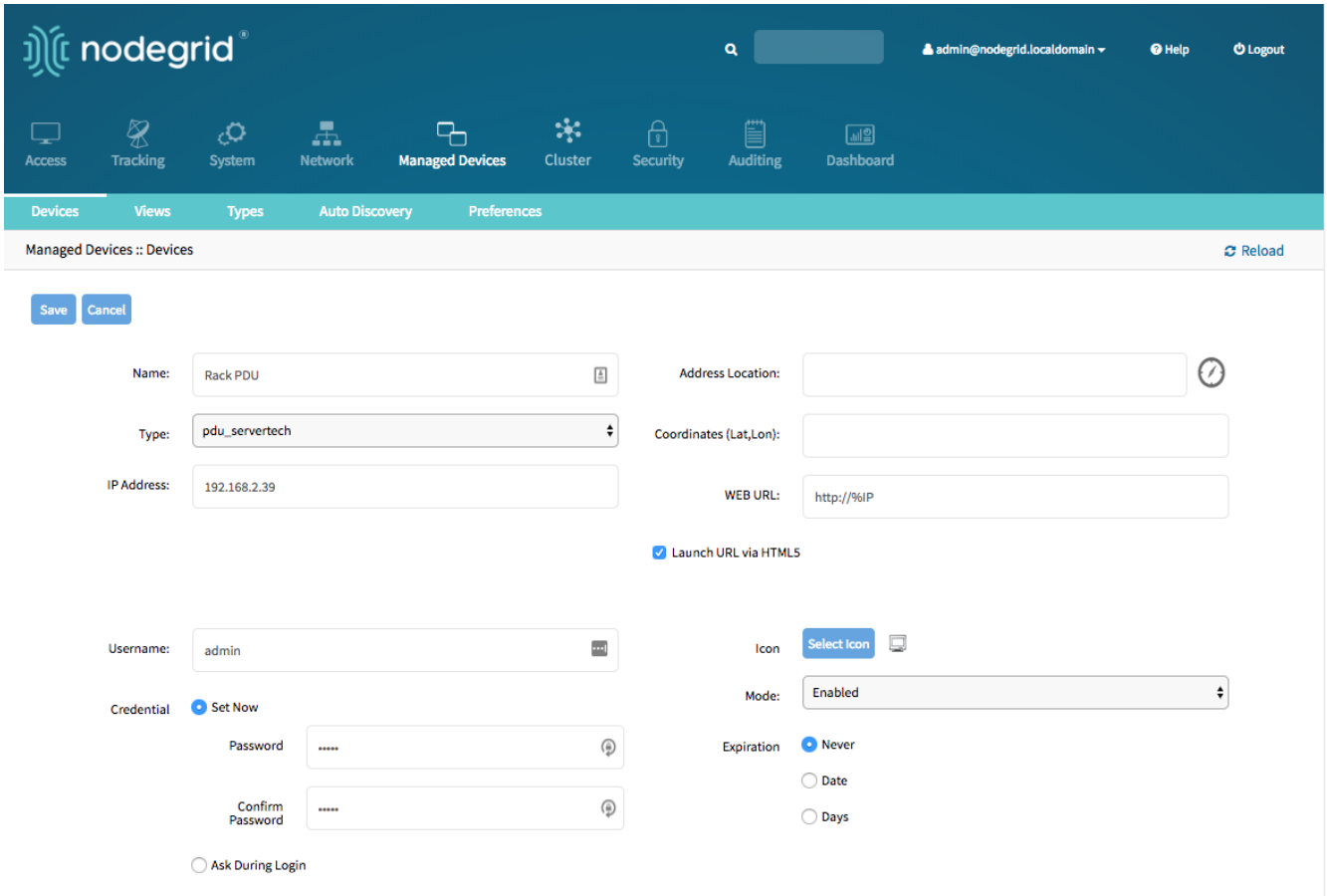
Nodegrid は、これらのデバイスの以下の機能に対応します:

- コンソールセッション
- データロギング
- カスタムコマンド
- Web セッション
- 出力電源制御

> 注: 電源制御機能が、ラック PDU でサポートされている必要があります。特定モデルで機能が使用可能かどうかは、ラック PDU のマニュアルを確認してください。

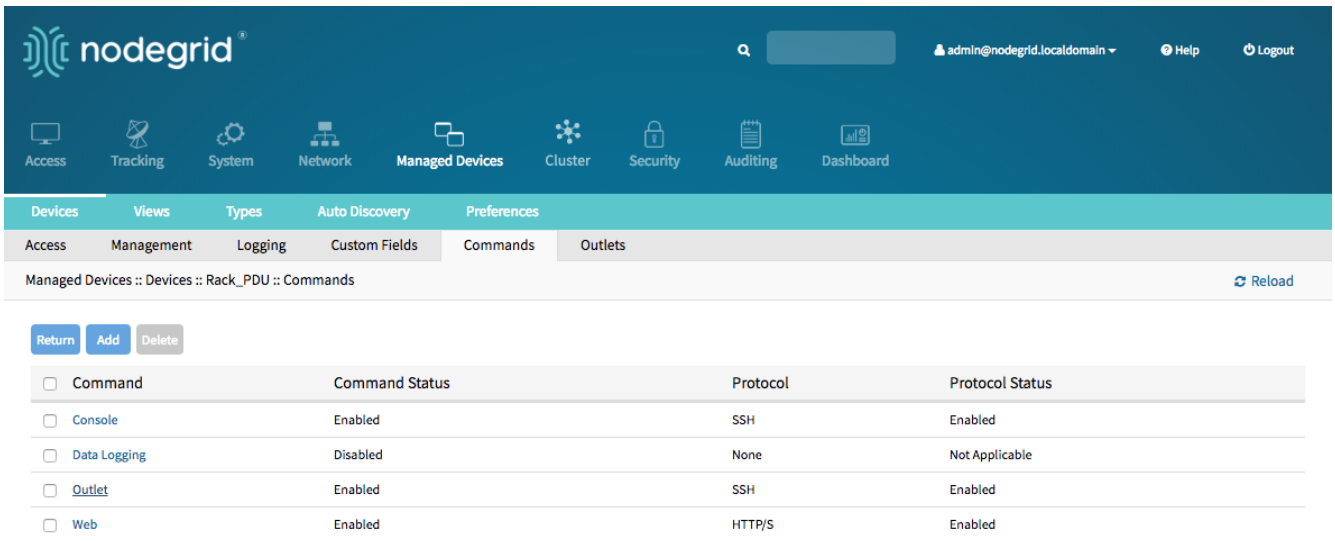
ラック PDU - WebUI

- 移動 `Managed Devices:: Devices`
- ボタン `Add` をクリックして、デバイスをシステムに追加します。
- 追加するラック PDU の名前を入力します。
- ラック PDU の IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- フィールド `Type` で、ラック PDU に一致するタイプを選択します。指定可能な値: *pdu_apc*、*pdu_baytech*、*pdu_eaton*、*pdu_mph2*、*pdu_pm3000*、*pdu_cpi*、*pdu_raritan*、*pdu_geist*、*pdu_servertech*、*pdu_enconnex*、*pdu_cyberpower*、*pdu_rittal*
- ログイン時にユーザー資格情報を提供する場合、ラック PDU の `username` と `password` を入力するか、オプション `Ask During Login` を選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

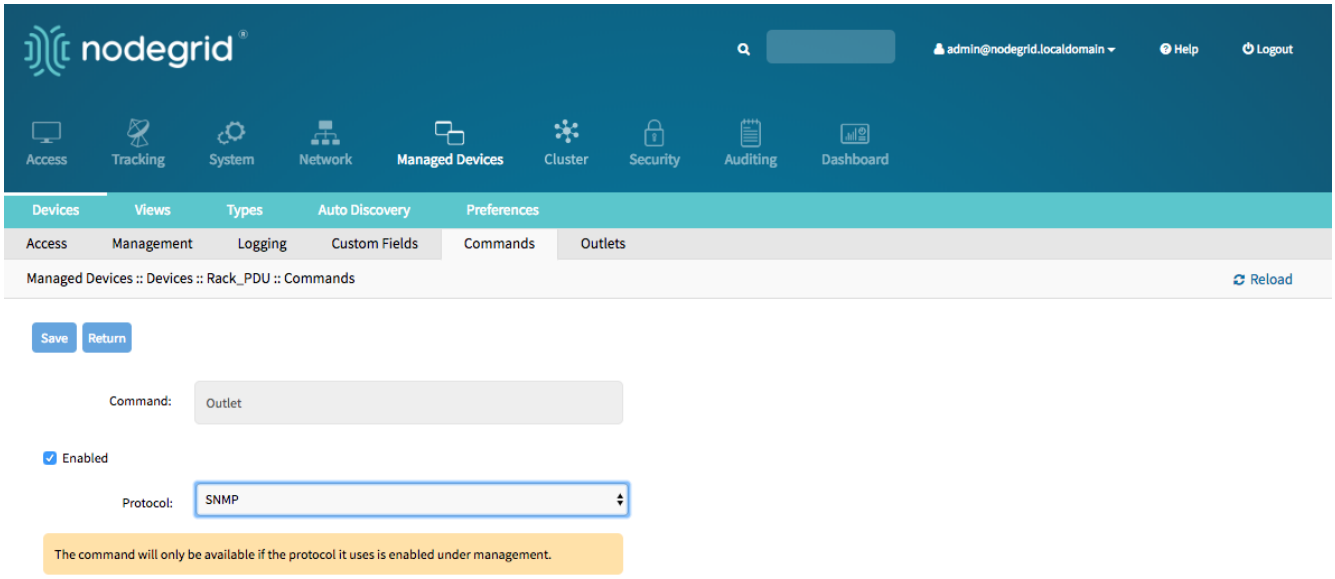


> 注: デフォルトで、Nodegrid はssh/telnet を使用してラック PDU と通信します。これらのインターフェースを使用する ラック PDU の応答時間は、通常非常に遅くなります。このため、可能であれば、ラック PDU との通信に SNMP を使用することをお勧めします。

- 移動 `Managed Devices:: Devices`
- 新しく追加されたラック PDU の `Name` をクリックします
- メニュー `Commands` に移動して*コンセント*をクリックします

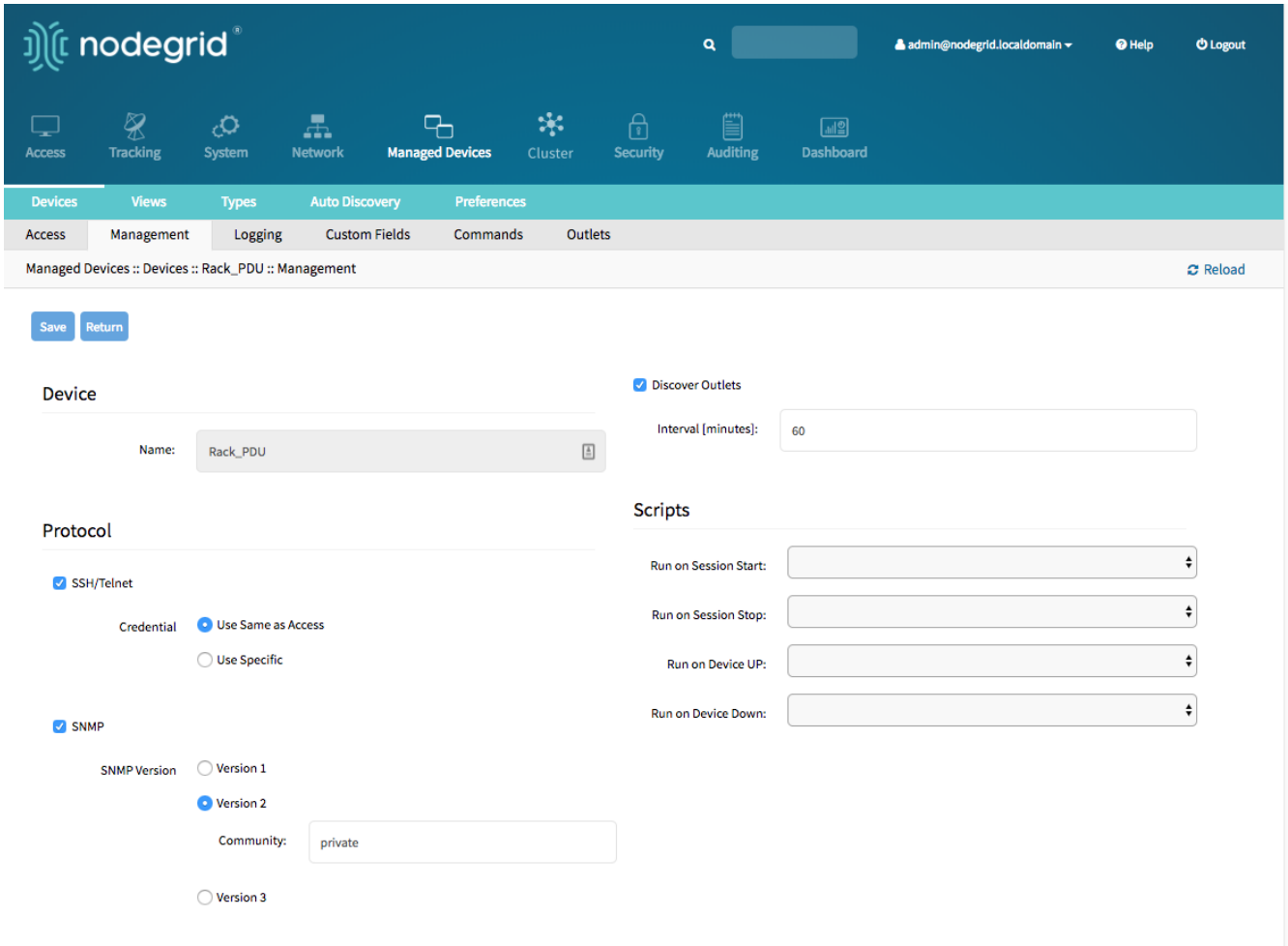


- `Protocol` を *SNMP*に変更し、 *保存*をクリックします



- メニュー`Management`に移動し、ラック PDU の設定に合わせて SNMP 値を更新します。以下をクリック `Save`

> 注: 読み取りと書き込みアクセスを提供する SNMP の詳細を使用します。読み取り専用の資格情報を使用すると、Nodegrid Platform は電源出力を制御できません。



- ラック PDU 出力は自動検出されます。ラック PDU によっては、このプロセスに数分間かかる場合があります。

ラック PDU の追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義
  - `name`
  - `type` 指定可能な値:
    pdu_apc、*pdu_baytech*、*pdu_eaton*、*pdu_mph2*、*pdu_pm3000*、*pdu_cpi*、*pdu_raritan*、*pdu_geist*、*pdu_servertech*、*pdu_enconnex*、*pdu_cyberpower*、*pdu_rittal*
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、または `Ask During Login` オプションを選択します
  - `endpoint` *アプライアンス*として定義する必要があります
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

> 注: デフォルトで、Nodegrid は ssh/telnet を使用してラック PDU と通信します。これらのインターフェースを使用するラック PDU の応答時間は、通常非常に遅くなります。このため、可能であれば、ラック PDU との通信には SNMP を使用することをお勧めします。

- 移動先 `/settings/devices/<device name>/commands/outlet`
- プロトコルを SNMP に変更します
- 移動先 `/settings/devices/<device name>/management`
- SNMP を有効にし、必要な SNMP バージョンと詳細を選択します
- 変更を保存 `commit`

> 注: 読み取りと書き込みアクセスを提供する SNMP の詳細を使用します。読み取り専用の資格情報を使用すると、Nodegrid Platform は電源出力を制御できません。

- ラック PDU 出力は自動検出されます。ラック PDU によっては、このプロセスに数分間かかる場合があります。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Rack_PDU
[admin@nodegrid {devices}]# set type=pdu_servertech
[admin@nodegrid {devices}]# set ip_address=192.168.2.39
[admin@nodegrid {devices}]# set credential=ask_during_login

or
```

```
[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
[admin@nodegrid /]# cd /settings/devices/Rack_PDU/commands/outlet
[admin@nodegrid outlet]# set protocol=snmp
[admin@nodegrid outlet]# cd /settings/devices/Rack_PDU/management/
[admin@nodegrid management]# set snmp=yes
[+admin@nodegrid management]# snmp_version = v2
[+admin@nodegrid management]# snmp_commmunity = private
[+admin@nodegrid management]# commit
```

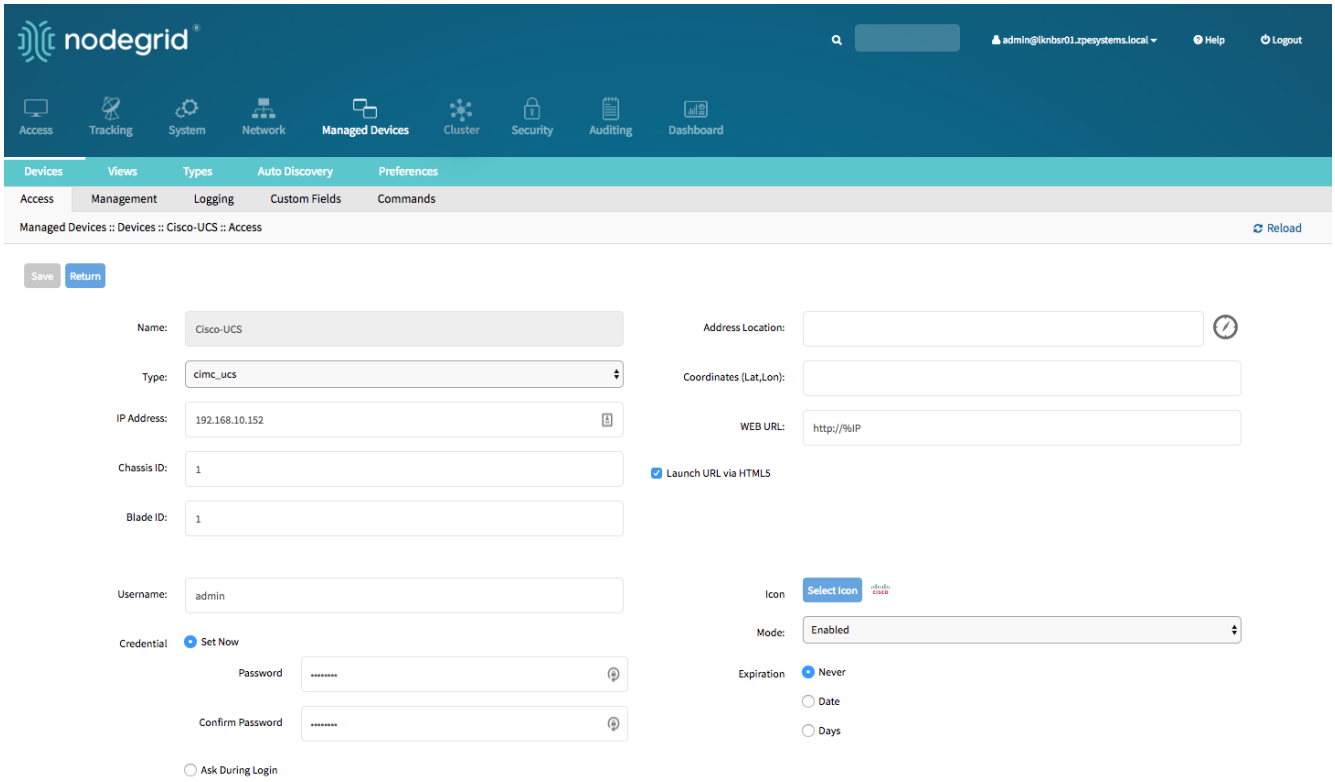
Cisco UCS

このソリューションは、コンソールポートと管理インターフェースを通じて Cisco UCS を管理します。Nodegrid は、これらのデバイスの以下の機能に対応します:

- コンソールセッション
- データロギング
- イベントロギング
- Cisco UCS アプライアンスを介した電源制御
- ウェブセッション
- カスタムコマンド

Cisco UCS の追加 - WebUI

- `Managed Devices:: Devices` へ移動し、
- `Add` ボタンをクリックして、デバイスをシステムに追加します。
- Cisco UCS ブレードの名前を入力して追加します。
- ブレードシャーシの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- `Type` フィールドで、アプライアンスに一致するタイプを選択します。指定可能な値: *CIMC_UCS*
- `Chassis ID` およびブレードを表す `Blade ID` を入力します
- ログイン時にユーザー資格情報を提供する場合、ブレードシャーシの `username` と `password` を入力するか、 `Ask During Login` オプションを選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



Cisco UCS の追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義します
  - `name` 追加されるブレード
  - `type` 指定可能な値: *CIMC_UCS*
  - `ip_address` ブレードシャーシ
  - `chassis_id` ブレードシャーシ
  - `blade_id` ブレードサーバ
  - `username` 次に、ログイン時にユーザー資格情報を提供する場合、ブレードシャーシの `password` を入力するか、 `Ask During Login` オプションを選択します
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Cisco-UCS
[admin@nodegrid {devices}]# set type=cimc_ucs
[admin@nodegrid {devices}]# set ip_address=192.168.10.151
[admin@nodegrid {devices}]# set chassis_id=1 blade_id=1s
[admin@nodegrid {devices}]# set credential=ask_during_login
```

```
or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

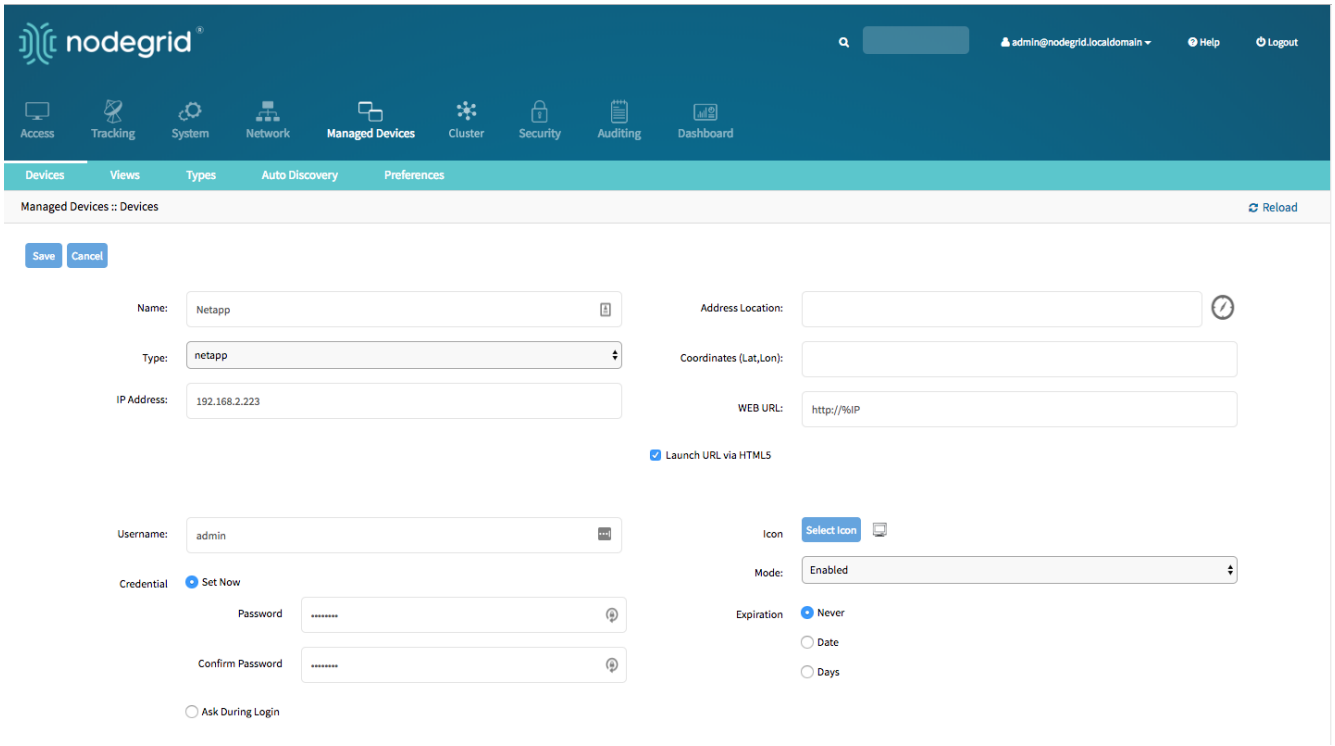
Netapp

このソリューションは、その管理インターフェースを介して Netapp アプライアンスの管理をサポートします。Nodegrid は、これらのデバイスの以下の機能に対応します:

- コンソールセッション
- データロギング
- イベントロギング
- Netappアプライアンスを介した電源制御
- ウェブセッション
- カスタムコマンド
- ラック PDU を介した電源制御

Netapp の追加 - WebUI

- `Managed Devices:: Devices`へ移動し、
- `Add`ボタンをクリックして、デバイスをシステムに追加します。
- 追加したいアプライアンスの名前を入力します。
- デバイスの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- `Type`フィールドで、Netapp アプライアンスに一致するタイプを選択します。指定可能な値: *Netapp*
- ログイン時にユーザー資格情報を提供する場合、`username`と`password`を入力するか、 `Ask During Login`オプションを選択します
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



Netapp の追加 - CLI

- 移動先 `/settings/devices`
- `add` コマンドで、新規デバイスを作成します
- `set` コマンドで、以下の設定を定義します
  - `name`
  - `type` 指定可能な値: *Netapp*
  - `ip_address`
  - `username` および、ログイン時にユーザー資格情報を提供する場合、デバイスの `password`、または `Ask During Login` オプションを選択します
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Netapp
[admin@nodegrid {devices}]# set type=netapp
[admin@nodegrid {devices}]# set ip_address=192.168.10.250
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
```

```
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit
```

Infrabox

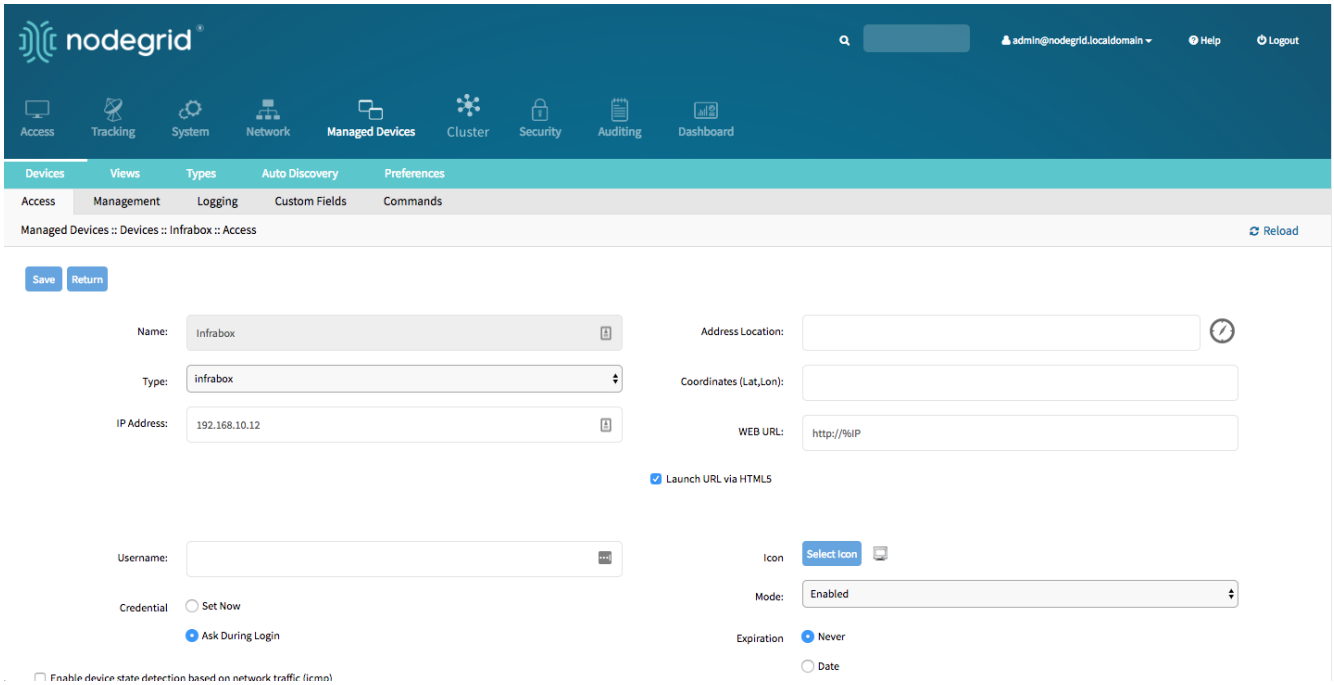
このソリューションは、 InfraSolution のラックソリューションアプライアンス (Infrabox) のスマートアクセスコントロールに対応します。Nodegrid は、これらのデバイスの以下の機能に対応します:

- ドア制御
- ウェブセッション
- ラック PDU を介した電源制御

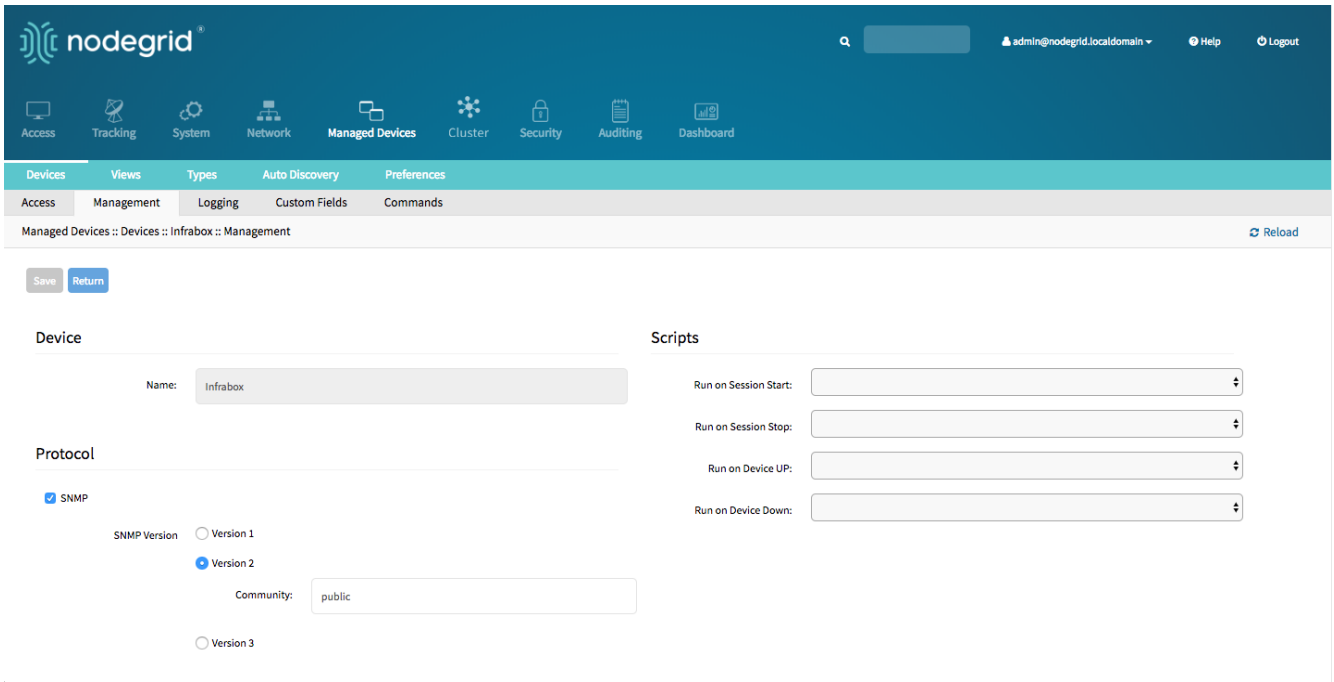
> 注: アプライアンスへの通信には、アプライアンスで SNMP を設定する必要があります。

Infrabox の追加 - WebUI

- `Managed Devices:: Devices`へ移動し、
- `Add`ボタンをクリックして、デバイスをシステムに追加します。
- 追加したいアプライアンスの名前を入力します。
- デバイスの IP アドレスを入力します。Nodegrid Platform で IP アドレスに到達可能なことを確認します。
- `Type`フィールドで、Infrabox アプライアンスに一致するタイプを選択します。 指定可能な値: *Infrabox*
- `Ask During Login`を選択し、ユーザー資格情報を提供しない
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



- `Management`メニューに移動し、アプライアンスの設定に合わせて SNMP 値を更新します
- 以下をクリック `Save`



Infrabox の追加 - CLI

- 移動先 `/settings/devices`
- `add`コマンドで、新規デバイスを作成します
- `set`コマンドで、以下の設定を定義します
  - `name`
  - `type`指定可能な値: *Infrabox*
  - `ip_address`
  - `username`および、ログイン時にユーザー資格情報を提供する場合、デバイスの`password`、または`Ask During Login`オプションを選択します
- 変更を保存 `commit`
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。
- 移動先 `/settings/devices/<Device>/management/`
- `set`コマンドで、SNMP 値を定義します
  - `snmp_version`
  - `snmp_community`
- 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Infrabox
```

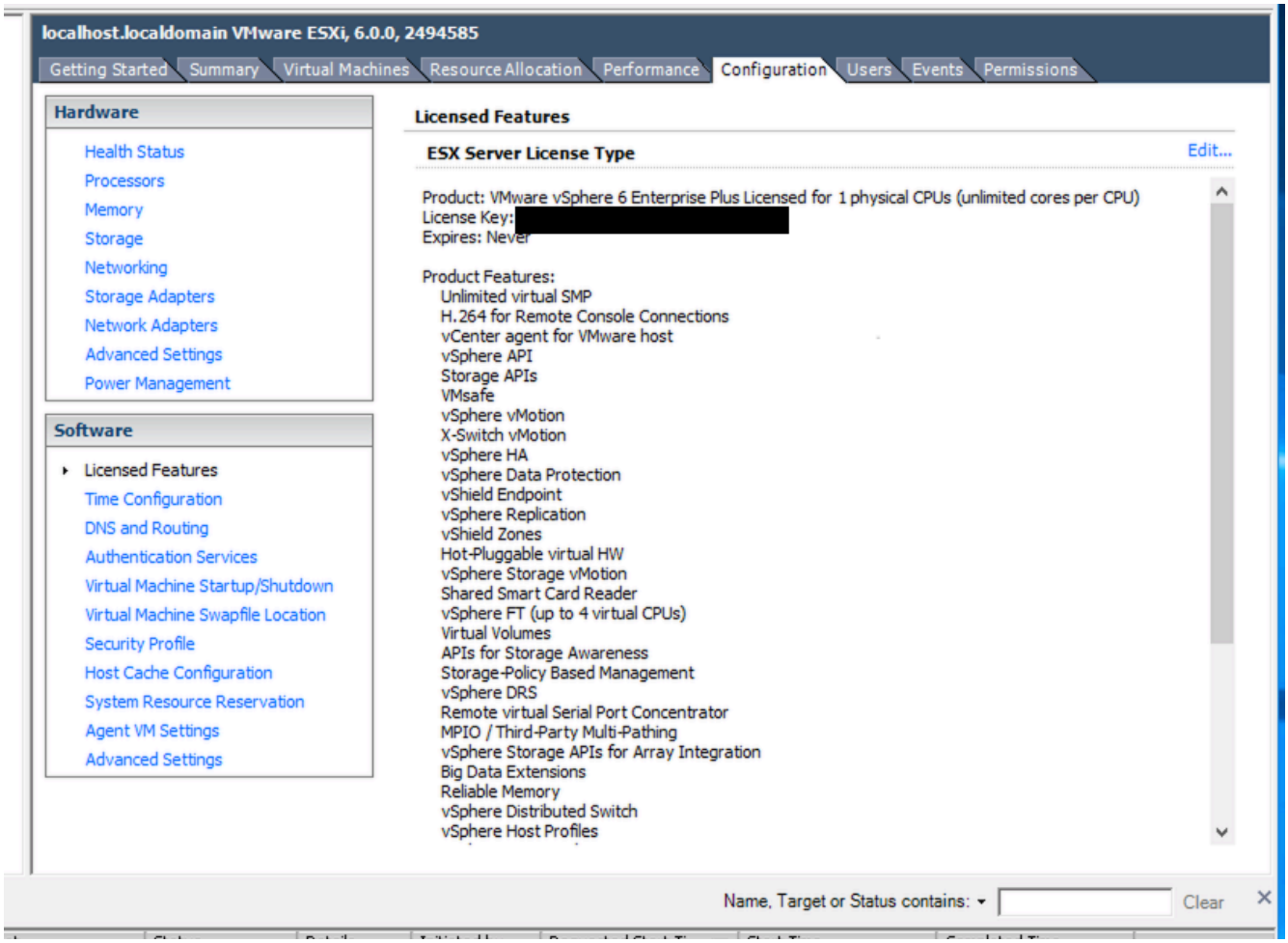
```
[admin@nodegrid {devices}]# set type=infrabox
[admin@nodegrid {devices}]# set ip_address=192.168.10.250
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# commit

[admin@nodegrid outlet]# cd /settings/devices/Infrabox/management/
[admin@nodegrid management]# set snmp=yes
[+admin@nodegrid management]# snmp_version=v2
[+admin@nodegrid management]# snmp_commmunity=private
[+admin@nodegrid management]# commit
```

仮想マシン

このソリューションは、VMWare 仮想マシンと KVM 仮想マシンの管理をサポートします。Nodegrid は、これらのデバイスの以下の機能に対応します:

- MKS セッション (VMWare マシン専用)
- 仮想Serial Consoleセッション (VMWare マシン専用)
- コンソールセッション (KVM マシン専用)
- ハイパーバイザーを介した電源制御
- デバイスへの Web セッション

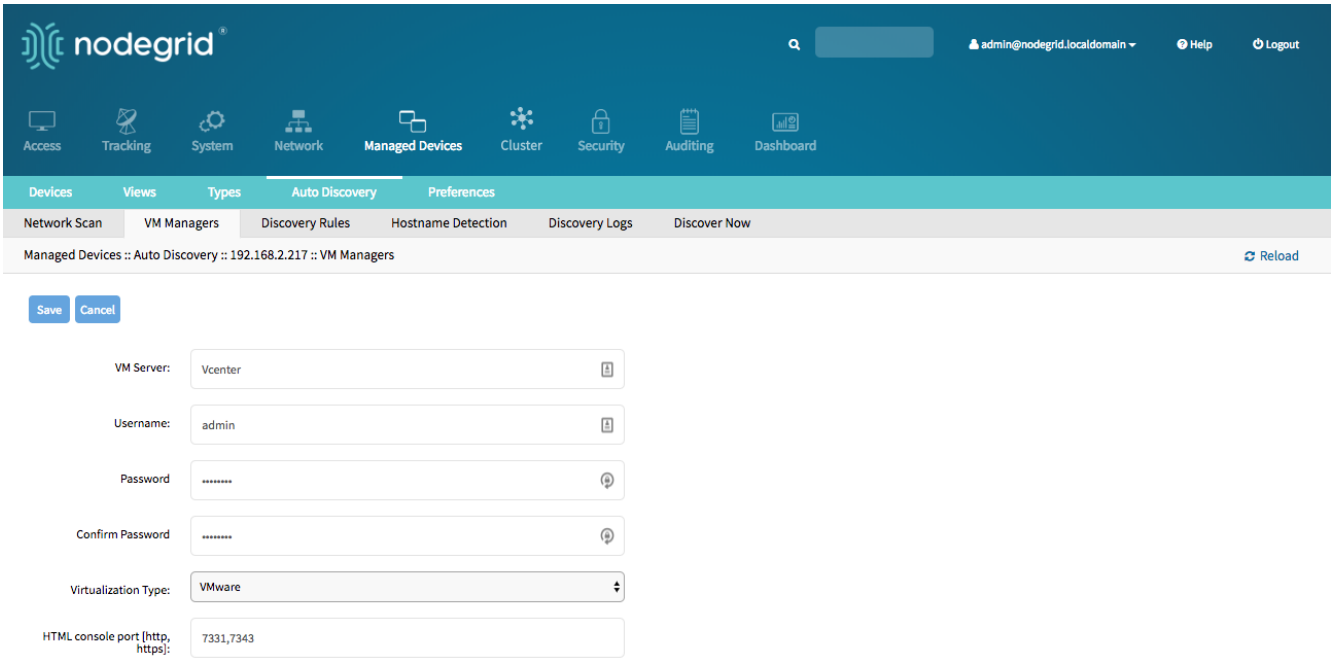
システムは、ESX への直接接続、または VSphere サーバへの接続をサポートします。接続が直接行われた場合、ESX サーバが [Vmware ホスト用 vCenter エージェント] 機能をサポートする必要があり、ESX サーバライセンスを介して有効化できます。ESX サーバがこの機能をサポートしているかどうかを確認するには、ESX ホストにログインし、[ライセンス機能] セクションに移動します。ホストにサポートされている使用可能なライセンスと機能を次に示します。



> **注**: VMWare 仮想マシンで vSPC オプションを使用するには、ポートを仮想マシン上で設定する必要があります。次の付録を参照下さい VM サーバでの仮想シリアルポート (vSPC) の設定

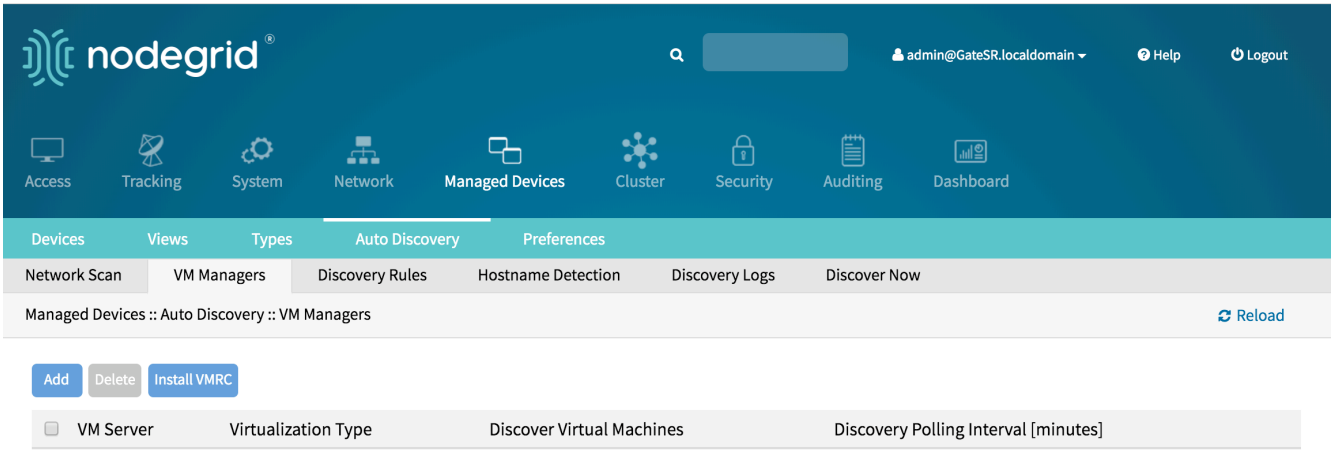
VMWare 仮想マシンの追加 - WebUI

- VM Manager を定義
  - 移動先 `Managed Devices :: Auto Discovery :: VM Managers`
  - `Add` をクリックして、新しい VM Manager を定義します
  - フィールド `VM Server` に vCenter/ESXi IP または FQDN を提供します
  - サーバの `Username` および `Password` を定義します
  - 必要に応じて `HTML console port` を調整します
  - 以下をクリック `Save`



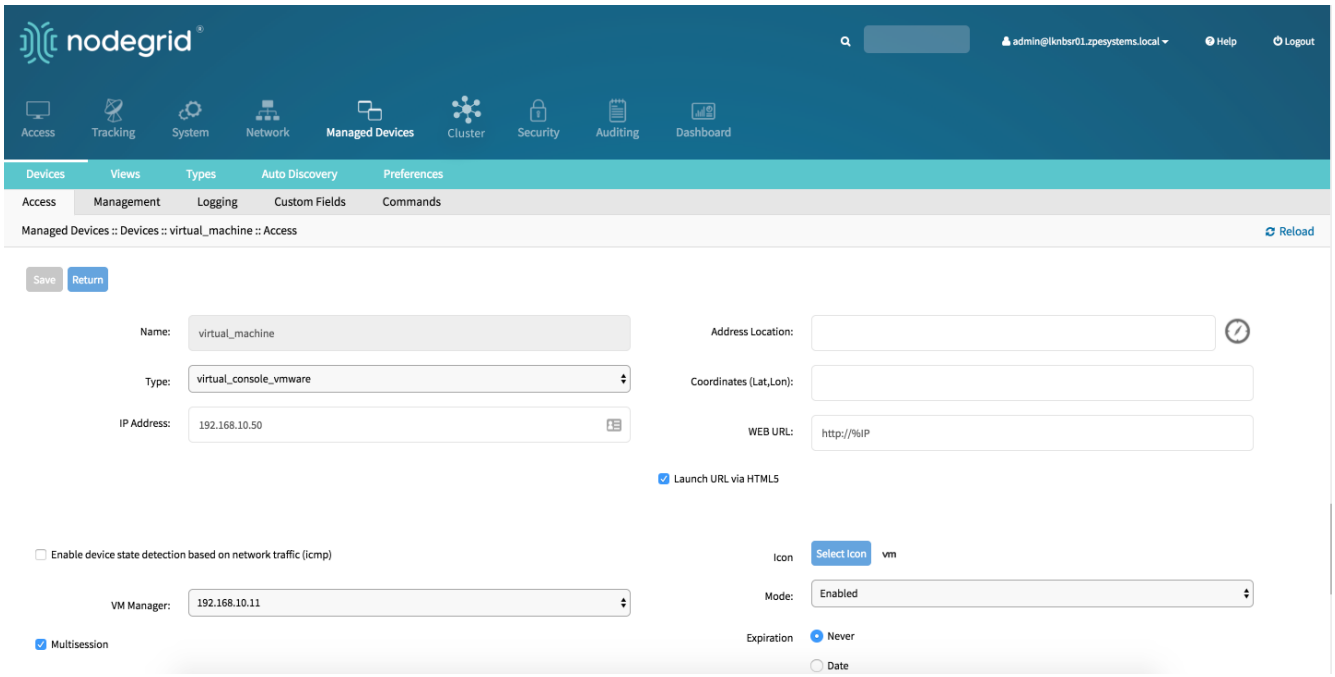
VMRC のインストール - WebUI

管理対象デバイス::自動検出::VM マネージャーの [VMRC をインストール] (Vmware リモートコンソール) をクリックすると、適切な作動グラフィカルデバイス接続と仮想マシンへのコンソールアクセスが可能になります。



- デバイスの作成
  - 移動先 `Managed Devices:: Devices`
  - `Add` ボタンをクリックして、デバイスをシステムに追加します。
  - 管理する仮想マシンの名前を入力します。名前はハイパーバイザーの名前と一致する必要があります
  - オプション: 仮想マシンの IP アドレスを入力します
  - `Type` フィールドで、仮想マシンと同じタイプを選択します。指定可能な値: *仮想*コンソール *VMware*
  - フィールド `VM Manager` で、マシンが実行する適切なハイパーバイザー を選択します

- [保存] ボタンをクリックします。



VMWare 仮想マシンの追加 - CLI

- VM Manager を定義
  - 移動先 `/settings/auto_discovery/vm_managers/`
  - この `add` コマンドで、VM マネージャーを作成します
  - `set` コマンドで、以下の設定を定義します
    - `vm_server` : vCenter/ESXi IP または FQDN を提供します
    - `username` を定義し、 `password`
    - 必要に応じて `html_console_port` を調整します
  - 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/auto_discovery/vm_managers/
[admin@nodegrid vm_managers]# add
[admin@nodegrid {vm_managers}]# set vm_server=vCenter
[admin@nodegrid {vm_managers}]# set username=admin
[admin@nodegrid {vm_managers}]# set password=password
[admin@nodegrid {vm_managers}]# commit
```

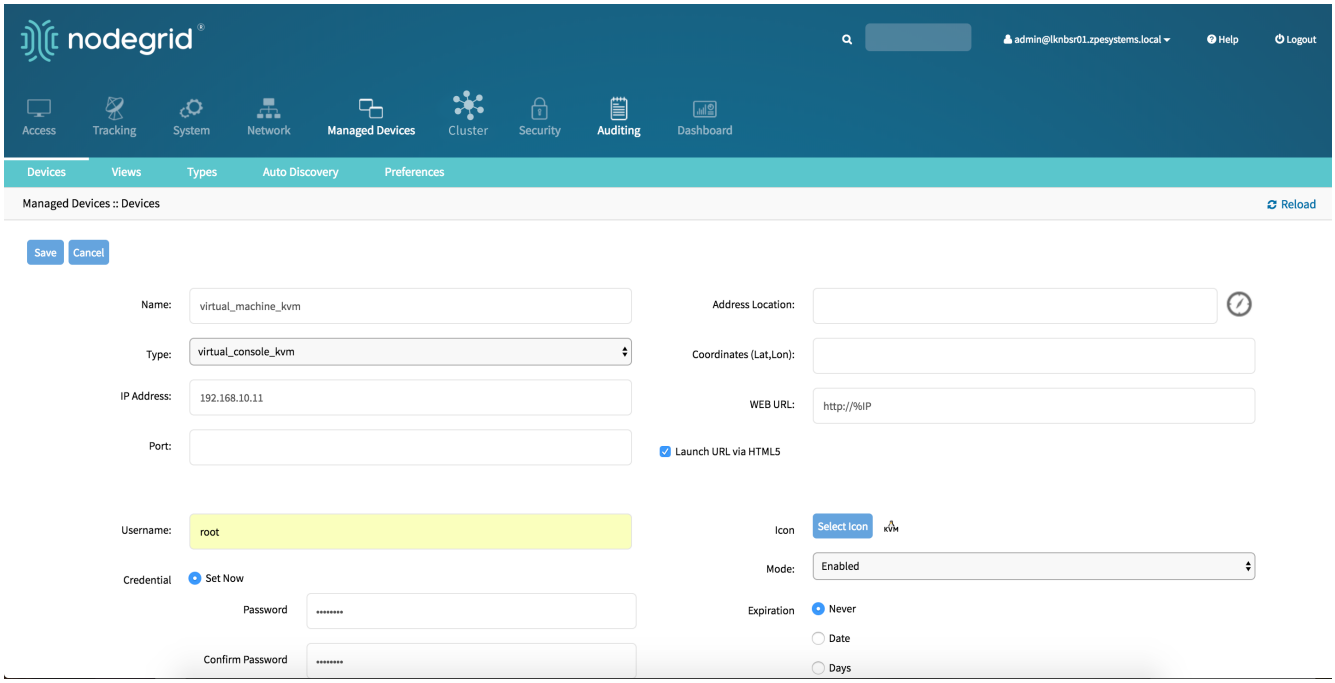
- デバイスの作成
  - 移動先 `/settings/devices`
  - `add` コマンドで、新規デバイスを作成します
  - `set` コマンドで、以下の設定を定義します

        - `name`
        - `type`指定可能な値: *仮想*コンソール*VMware*
        - オプションで、ターゲットデバイスとして`ip_address`
        - `vm_manager`、 既存の VM マネージャに設定
    - 変更を保存 `commit`
    - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Virtual_Machine
[admin@nodegrid {devices}]# set type=virtual_console_vmware
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set vm_manager=192.168.10.11
[admin@nodegrid {devices}]# commit
```

KVM 仮想マシンの追加 - WebUI

- デバイスの作成
    - 移動先 `Managed Devices:: Devices`
    - `Add`ボタンをクリックして、デバイスをシステムに追加します。
    - 管理する仮想マシンの名前を入力します。名前はハイパーバイザーの名前と一致する必要があります
    - KVM ハイパーバイザーの IP アドレスを入力します
    - KVM ハイパーバイザーのユーザー名とパスワードを提供します
    - `Type`フィールドで、仮想マシンと同じタイプを選択します。指定可能な値: *仮想*コンソール*kvm*
    - [保存] ボタンをクリックします。



KVM 仮想マシンの追加 - CLI

- デバイスの作成
  - 移動先 `/settings/devices`
  - `add` コマンドで、新規デバイスを作成します
  - `set` コマンドで、以下の設定を定義します
    - `name` または仮想マシンの場合、これはハイパーバイザー上のマシンの名前と一致する必要があります。
    - `type` 指定可能な値: *仮想コンソール*kvm
    - `ip_address` KVM ハイパーバイザー
    - KVM ハイパーバイザーの `password` と `username` を提供します
  - 変更を保存 `commit`
  - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=virtual_machine_kvm
[admin@nodegrid {devices}]# set type=virtual_console_vmware
[admin@nodegrid {devices}]# set ip_address=192.168.10.11
[admin@nodegrid {devices}]# set username=root
[admin@nodegrid {devices}]# set password=password
[admin@nodegrid {devices}]# commit
```

Nodegrid デバイス

USB センサ

Nodegrid USB 温度および湿度センサは、Nodegrid システムで自動検出されます。デバイスのタイプは自動で`usb_sensor`に調整されます。デバイスが検出された後で、デバイスを有効にする必要があります。この後デバイスの準備が整い、監視とアラーム管理に使用可能となります。

KVM ドングル

USB KVM ドングルを使用すると、VGA および USB 接続を介してレガシーサーバに KVM セッションを確立できます。ドングルは、接続されるとすぐにシステムによって自動検出されます。デバイスを有効にするだけです。

Bluetooth

Nodegrid Platform は、主に監視と IoT アプリケーション向けの Bluetooth デバイスをサポートします。Bluetooth 機能は、Nodegrid Service Router ファミリーで使用できる Nodegrid WiFi モジュールを介して提供されます。

デフォルトで、Bluetooth 機能は無効になっているため、使用する前に手動で有効化する必要があります。

現在、次のコマンドを実行することで、管理者ユーザーが Shell を通じてサービスを有効化することができます。

```
[admin@nodegrid /]# shell sudo su -
root@nodegrid:~#sed -i s/^BLUETOOTH_ENABLED=0/BLUETOOTH_ENABLED=1/g
/etc/default/bluetooth
root@nodegrid:~#sed -i s/^#AutoEnable=true/AutoEnable=true/g
/etc/bluetooth/main.conf
root@nodegrid:~#sed -i s/^#InitiallyPowered=true/InitiallyPowered=true/g
/etc/bluetooth/main.conf
```

その後、Bluetooth デバイスを Nodegrid とペアリングして、監視や IoT アプリケーションで使用できます。

`bluetoothctl`コマンドを使用して、デバイスをペアリングすることが可能です

```
root@nodegrid:~#bluetoothctl bluetoothctl
[bluetooth]# devices
Device 00:16:94:1A:EA:2C Sensor
[bluetooth]# pair 00:16:94:1A:EA:2C
Attempting to pair with 00:16:94:1A:EA:2C
Pairing successful
[bluetooth]# connect 00:16:94:1A:EA:2C
Attempting to connect to 00:16:94:1A:EA:2C
Connection successful
[bluetooth]# quit
```

自動検出

Nodegrid Platform では、ネットワークデバイス、コンソールサーバの有効化されたコンソール、KVM スイッチ、仮想シリアルポート (VMWare) および仮想マシン (VMWare) を自動検出して追加できます。

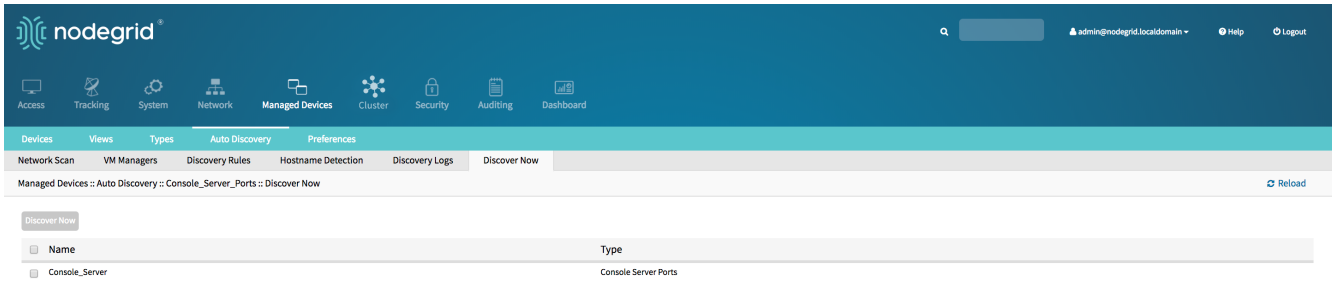
この機能は、既存のデバイスからそのプロファイルに一致するデバイスを検出してクローン作成し、動的アクセスグループを構築します。この機能で最良の結果を得るために、クローン作成プロセスで参照するデバイスが、正しく設定されていることを確認してください。デバイスにアクセスして、ユーザー名、パスワード、および IP アドレスが正しいことを確認します。ログファイルを監査して、データログとイベントログの設定が正しいことを確認します。イベントをシミュレートし、通知が作成されたかどうかを確認して、データログとイベントログに基づいてイベントが検出されていることを確認します。デバイスが、正しいアクセス権を備えた適切な承認グループ内にあることを確認します。

自動検出は、以下の一般的なプロセスに従います:

- テンプレートデバイスを作成します。このデバイスは、検出されたデバイスへの接続の詳細を除き、すべての設定をクローンするために使用されます。エンドデバイスに表示されるすべての設定作業に便利です。

> 注: ターゲットデバイスのタイプごとに、テンプレートデバイスを作成する必要があります。

- ネットワーク デバイスの場合は以下を作成 `Network Scan`
- 仮想マシンの場合は以下を作成 `Virtual Manager`
- この手順では、`Discovery Rule` を作成するすべてのデバイスに対して、テンプレートデバイスと検出ルールがリンクされ、検出されたすべてのデバイスでどのアクションが実行されるかが決定されます。
- 検出プロセスを開始します。この手順は、追加されたデバイスのタイプに応じて自動で実行されます。検出プロセスは、アプライアンスがプラットフォームに追加された時に自動で開始され、`/settings/auto_discovery/discover_now/` WebUICLI 内または `Managed Devices:: Auto Discovery:: Discover Now` CLI から、任意の時点で手動で開始することができます

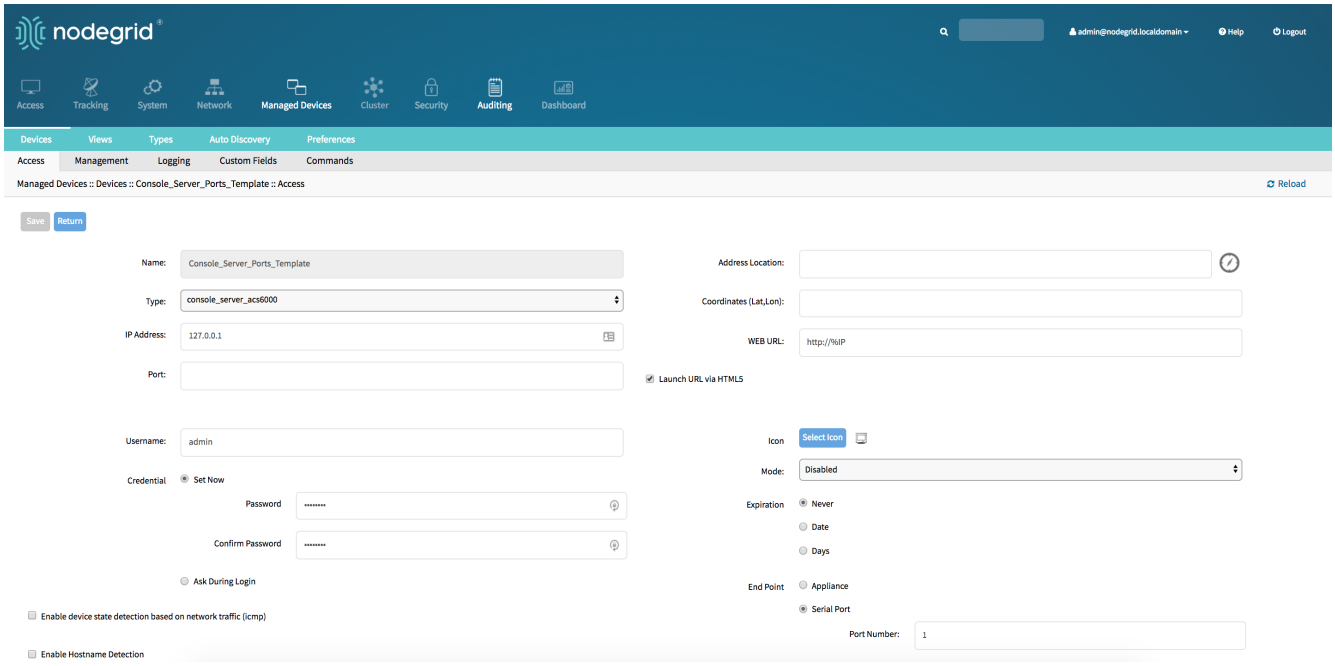


Console Server および KVM スイッチポートの自動検出

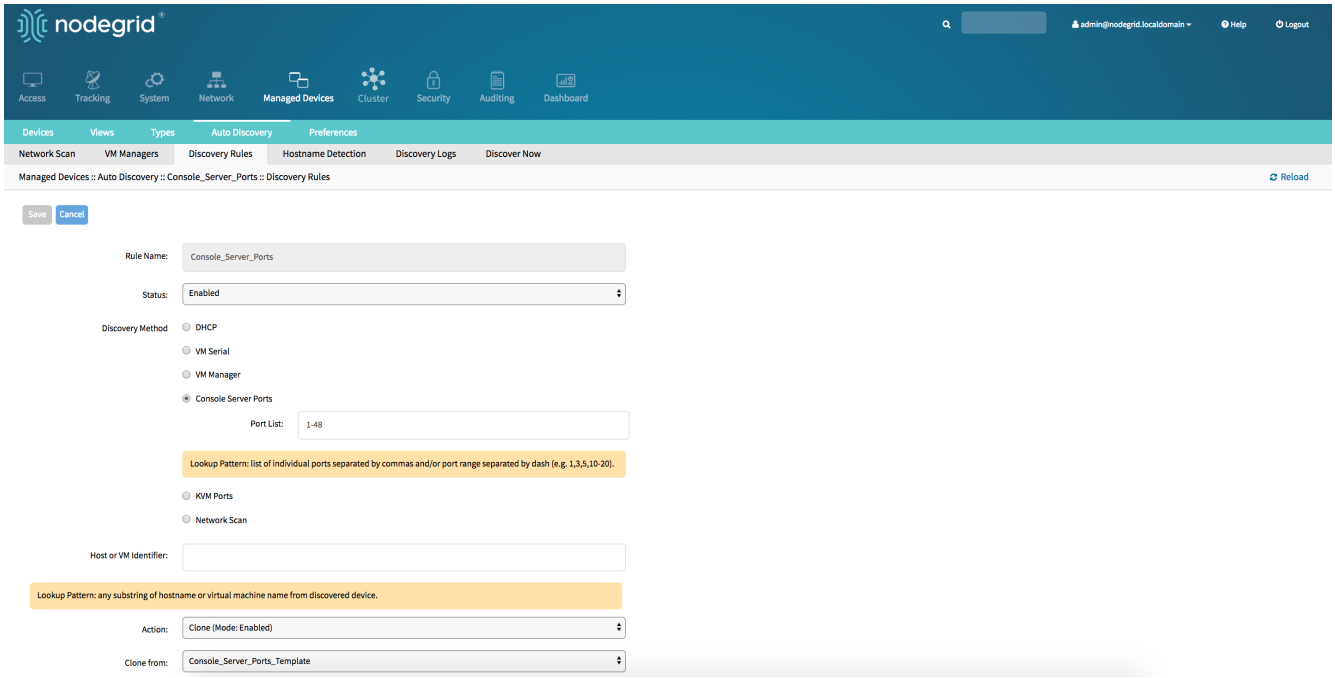
自動検出プロセスを使用すると、サードパーティ製コンソールサーバポートおよび KVM スイッチポートの備わる管理対象デバイスを自動で追加・設定できます。このプロセスで、管理対象アプライアンス上のすべての有効なポートが検出されます。Console Server アプライアンスおよび KVM スイッチは、ネットワークデバイスプロセスで検出できます。ネットワークデバイスの自動検出を参照してください。

Console Server と KVM スイッチ ポートの自動検出 - WebUI

- テンプレートデバイスの作成
  - 移動先 `Managed Devices:: Devices`
  - `Add` ボタンをクリックして、デバイスをシステムに追加します。
  - 追加したいテンプレートの名前を入力します
  - IP アドレスに 127.0.0.1 と入力します
  - フィールド `Type` で、Console Server に一致するタイプを選択します。指定可能な値: *console_server_acs*, *console_server_acs6000*,*console_server_lantronix*,*console_server_opengear*,console_server_digicp
    _
  - 選択 `Ask During Login`
  - *シリアルポート*または*KVM ポート*のいずれかを `End Point` として選択し、ポート番号を入力します
  - `Mode` *無効*を選択すると、デバイスがアクセスページに表示されないようにします
  - [保存] ボタンをクリックします。
  - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

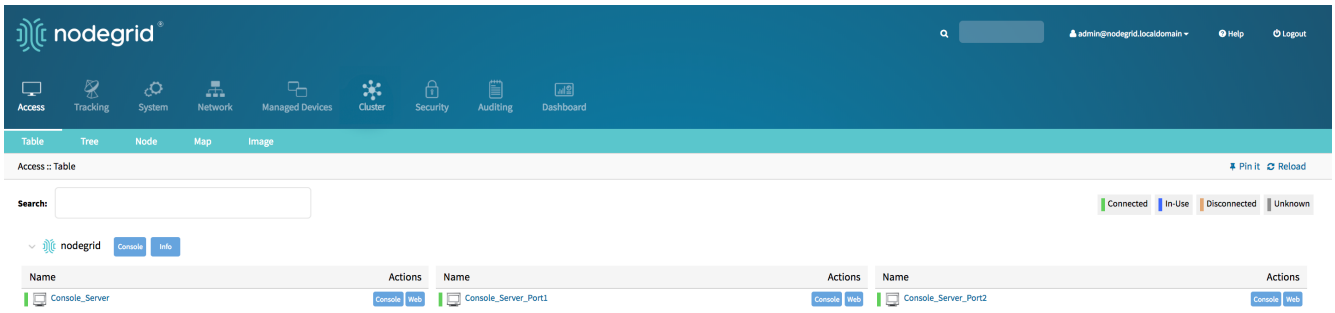


- 検出ルールを作成します
  - 移動先 `Managed Devices:: Auto Discovery:: Discovery Rules`
  - `Add` をクリックして、新しい [検出ルール] を追加します
  - 検出ルールのための `Name` を入力します
  - 検出ルールのための `Status` を選択します。使用可能な値: *有効*、*無効*
  - `Discovery Method` として、*Console Server ポート* または *KVM ポート* のいずれかを選択します
  - `Port List` については、スキャンする必要があるポートのリストを提供します。例1,3,5,10-20
  - パラメータ `Host or VM Identifier` を使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
  - `Action` については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローン (モード:有効)*、*クローン (モード:オンデマンド)*、*クローン (モード:検出済み)*、*検出されたデバイスを破棄*
  - `Clone from field` で、前に作成されたテンプレートデバイスを選択します
  - `Save` をクリックしてルールを作成します



- Console Servers または KVM スイッチアプライアンスを作成します。詳細は、 Console Servers の追加を参照してください。
- アプライアンスが作成されると、Nodegrid Platform は作成された`Discovery Rules`に基づいて接続されたデバイスの検出を自動的に開始します。このプロセスは、完了するのに数分かかります。

  <--working-->



Console Server と KVM スイッチ ポートの自動検出 - CLI

- テンプレートデバイスの作成
  - 移動先 `/settings/devices`
  - `add`コマンドで、新規デバイスを作成します
  - `set`コマンドで、以下の設定を定義します
    - `name`
    - `type`指定可能な値: *console_server_acs*, *console_server_acs6000*,*console_server_lantronix*,*console_server_opengear*,console_server_digicp_
    - `ip_address` 127.0.0.1 として
    - ユーザー認証を以下のように設定します `Ask During Login`

        - `endpoint` *シリアルポートまたはKVMポート*として定義する必要があります
        - `port_number` ポート番号として定義する必要があります
        - `mode`を*無効*に設定します
    - 変更を保存 `commit`
    - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Console_Server_Port_Template
[admin@nodegrid {devices}]# set type=console_server_acs6000
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set end_point=serial_port
[admin@nodegrid {devices}]# set port_number=1
[admin@nodegrid {devices}]# set credential=ask_during_login
[admin@nodegrid {devices}]# set mode=disabled
[admin@nodegrid {devices}]# commit
```

- 検出ルールを作成します
    - 移動先 `/settings/auto_discovery/discovery_rules/`
    - `add`コマンドで、探索ルールを作成する
    - `set`コマンドで、以下の設定を定義します
        - `rule_name` 検出ルールの場合
        - `status` 検出されたルールの場合に考えられる値: *有効*、*無効*
        - `method` を*コンソールサーバポート*または*KVMポート*_のいずれかに設定します
        - `port_list` スキャンする必要があるポートのリストを提供。例1,3,5,10-20
        - `host_identifier` パラメータを使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
    - `action`については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローンモード有効化*、*クローンモードオンデマンド*、*クローンモード検出*、*破棄デバイス*_
    - `clone_from` 以前に作成されたテンプレートデバイスに設定
    - 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/auto_discovery/discovery_rules/
[admin@nodegrid discovery_rules]# add
[admin@nodegrid {discovery_rules}]# set rule_name=Console_Server_Ports
[admin@nodegrid {discovery_rules}]# set status=enabled
[admin@nodegrid {discovery_rules}]# set method=console_server_ports
[admin@nodegrid {discovery_rules}]# set port_list=1-48
[admin@nodegrid {discovery_rules}]# set action=clone_mode_enabled
[admin@nodegrid {discovery_rules}]# set clone_from=Console_Server_Ports_Template
[admin@nodegrid {discovery_rules}]# commit
```

- Console Servers または KVM スイッチアプライアンスを作成します。詳細は、 Console Servers の追加を参照してください。
- アプライアンスが作成されると、Nodegrid Platform は作成された `Discovery Rules` に基づいて接続されたデバイスの検出を自動的に開始します。このプロセスは、完了するのに数分かかります。

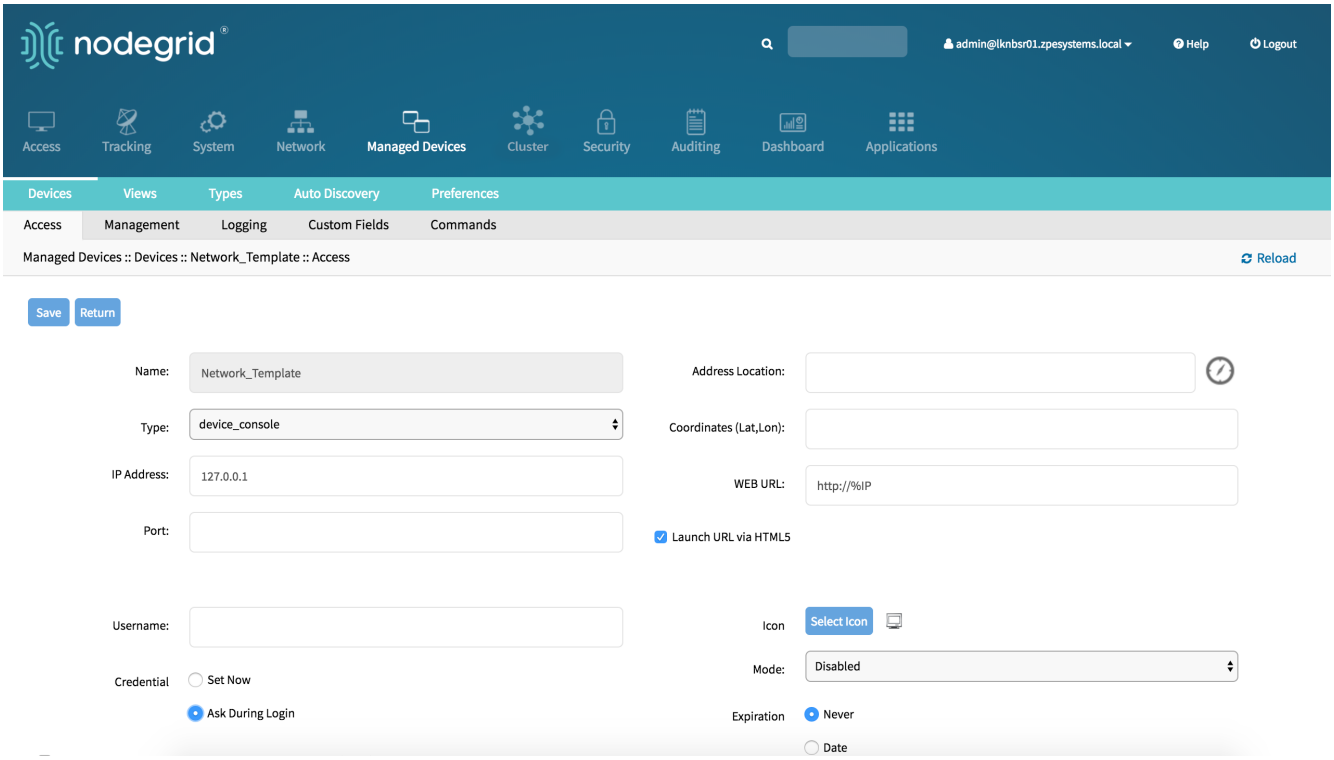
ネットワークデバイスの自動検出

ネットワークアプライアンスを自動的に検出し、Nodegrid Platform に追加できます。これには、Telnet、SSH、ICMP、Console Servers、KVMスイッチ、または IMPI プロトコルなどをサポートするアプライアンスが含まれます。 アプライアンスは、3 つの個別メソッドを使用して検出できます。それらは組み合わせたり、独自に使用することが可能です:

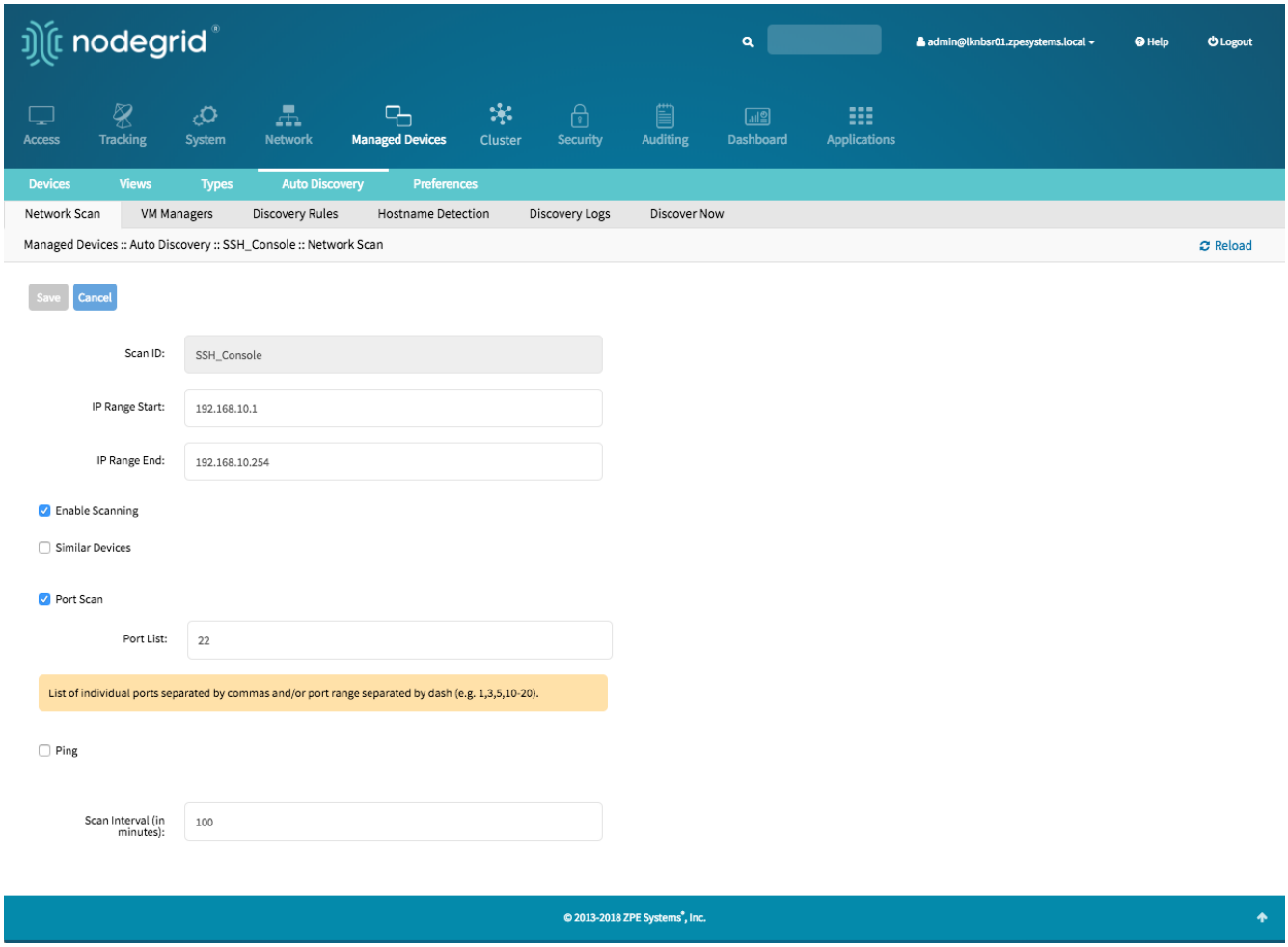
- 類似デバイス (ドロップダウンメニューからデバイスの1つを選択)、
- ポートをスキャンし、[ポート リスト] フィールドにポートのリストを入力します、
- Ping

ネットワーク デバイスの自動検出 - WebUI

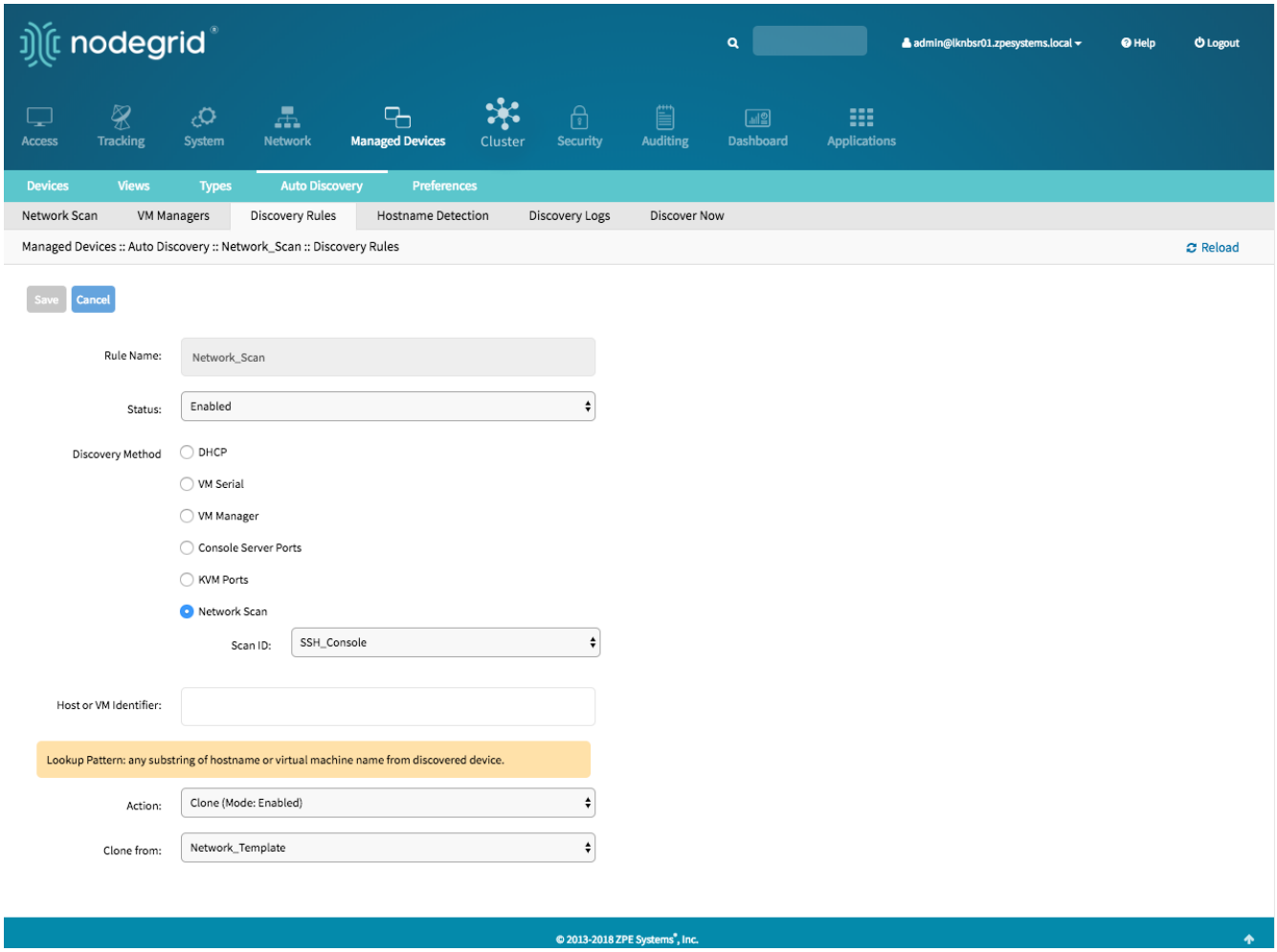
- テンプレートデバイスの作成
  - 移動先 `Managed Devices:: Devices`
  - `Add` ボタンをクリックして、デバイスをシステムに追加します。
  - 追加したいテンプレートの名前を入力します
  - IP アドレスに 127.0.0.1 と入力します
  - フィールド `Type` で、Console Server に一致するタイプを選択します。指定可能な値: *device_console*、*ilo*、*imm*、*drac*、*idrac6*、*ipmi1.5*、*impi2.0*、*ilom*、*cimc_ucs*、*netapp*、*infrabox*、*pdu**
  - ログイン時にユーザー資格情報を提供する場合、`username` と `password` を入力するか、 `Ask During Login` オプションを選択します
  - `Mode` *無効*を選択すると、デバイスがアクセスページに表示されないようにします
  - [保存] ボタンをクリックします。
  - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



- ネットワークスキャンの作成
  - 移動先 `Managed Devices:: Auto Discovery:: Network_Scan`
  - `Add` をクリックして、新しいネットワークスキャンを作成
  - の名前を入力 `Scan ID`
  - `IP Range Start` でスキャンするために、IP 範囲を定義 `IP Range End`
  - 次の 3 つのスキャン方法の 1 つ以上を選択して定義:
    - `Similar Devices` については、デバイスの識別に使用される既存のテンプレートを選択
    - `Port Scan` については、デバイス上で到達可能なポートのリストを定義
    - `ping` については、それ以上の設定は必要ありません
  - `Enable Scanning` ルールを有効化し、数分で範囲指定できる `Scan Interval` を定義



- 検出ルールを作成します
  - 移動先 `Managed Devices:: Auto Discovery:: Discovery Rules`
  - `Add` をクリックして、新しい [検出ルール] を追加します
  - 検出ルールのための `Name` を入力します
  - 検出ルールのための `Status` を選択します。使用可能な値: *有効*、*無効*
  - `Discovery Method` として、 *ネットワークスキャン*を選択
  - `Scan ID` のために、作成したネットワークスキャン ID を選択
  - パラメータ `Host or VM Identifier` を使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
  - `Action` については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローン(モード:有効)*、*クローン(モード:オンデマンド)*、*クローン(モード:検出済み)*、*検出されたデバイスを破棄*
  - `Clone from field` で、前に作成されたテンプレートデバイスを選択します
  - `Save` をクリックしてルールを作成します



- Nodegrid Platform は、作成された`Discovery Rules`に基づいて、デバイスの自動検出を開始します。このプロセスは、完了するのに数分かかります。

ネットワークデバイスの自動検出 - CLI

- テンプレートデバイスの作成
  - 移動先 `/settings/devices`
  - `add`コマンドで、新規デバイスを作成します
  - `set`コマンドで、以下の設定を定義します
    - `name`
    - `type`指定可能な値:
      device_console、*ilo*、*imm*、*drac*、*idrac6*、*ipmi1.5*、*impi2.0*、*ilom*、*cimc_ucs*、*netapp*、*infrabox*、*pdu**
    - `ip_address` 127.0.0.1 として
    - ログイン時にユーザー資格情報を提供する場合、`username`と`password`を設定するか、`Ask During Login`オプションを選択します
    - `mode`を*無効*に設定します
  - 変更を保存 `commit`

    - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Network_Template
[admin@nodegrid {devices}]# set type=device_console
[admin@nodegrid {devices}]# set ip_address=127.0.0.1
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# set mode=disabled
[admin@nodegrid {devices}]# commit
```

- ネットワークスキャンの作成
    - 移動先 `/settings/auto_discovery/network_scan/`
    - `add` コマンドで、ネットワークスキャンを作成する
    - `set` コマンドで、以下の設定を定義します
        - `scan_id` ネットワークスキャンの名前を入力する
        - `ip_range_start` でスキャンすべきネットワーク範囲を定義する `ip_range_end`
        - `enable_scanning` を *はい* に設定し、スキャンを有効にする
        - 3 つのスキャン方法から 1 つ以上を定義:
            - `similar_devices` を使用するために、 `device` を設定して、既存のデバイスかテンプレートの一つに一致させる
            - `port_scan` を使用するために、 `port_list` をデバイス上で到達可能なポートのリストに設定します
            - `ping` を使用するために、それ以上の設定は不要
        - 分単位で範囲設定できる `scan_interval` を設定する

```
[admin@nodegrid /]# cd /settings/auto_discovery/network_scan/
[admin@nodegrid network_scan]# add
[+admin@nodegrid {network_scan}]# set scan_id=SSH_Console
[+admin@nodegrid {network_scan}]# set ip_range_start=192.168.10.1
[+admin@nodegrid {network_scan}]# set ip_range_end=192.168.10.254
[+admin@nodegrid {network_scan}]# set enable_scanning=yes
[+admin@nodegrid {network_scan}]# set similar_devices=yes
```

```
[+admin@nodegrid {network_scan}]# set device= network_template
[+admin@nodegrid {network_scan}]# set port_scan=yes
[+admin@nodegrid {network_scan}]# set port_list=22
[+admin@nodegrid {network_scan}]# set ping=no
[+admin@nodegrid {network_scan}]# set scan_interval=100
[+admin@nodegrid {network_scan}]# commit
```

- 検出ルールを作成します
  - 移動先 `/settings/auto_discovery/discovery_rules/`
  - `add` コマンドで、探索ルールを作成する
  - `set` コマンドで、以下の設定を定義します
    - `rule_name` 検出ルールの場合
    - `status` 検出されたルールの場合に考えられる値: *有効*、*無効*
    - `method` *ネットワークスキャン_*に設定します
    - `scan_id` 以前作成したネットワーク スキャン ID を選択する
    - `host_identifier` パラメータを使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
  - `action` については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローンモード有効化*、*クローンモードオンデマンド*、*クローンモード検出*、*破棄デバイス_*
  - `clone_from` 以前に作成されたテンプレートデバイスに設定
  - 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/auto_discovery/discovery_rules/
[admin@nodegrid discovery_rules]# add
[admin@nodegrid {discovery_rules}]# set rule_name=Network_Scan
[admin@nodegrid {discovery_rules}]# set status=enabled
[admin@nodegrid {discovery_rules}]# set method=network_scan
[admin@nodegrid {discovery_rules}]# set scan_id=SSH_Console
[admin@nodegrid {discovery_rules}]# set action=clone_mode_enabled
[admin@nodegrid {discovery_rules}]# set clone_from=Network_Template
[admin@nodegrid {discovery_rules}]# commit
```

- Nodegrid Platform は、作成された `Discovery Rules` に基づいて、デバイスの自動検出を開始します。このプロセスは、完了するのに数分かかります。

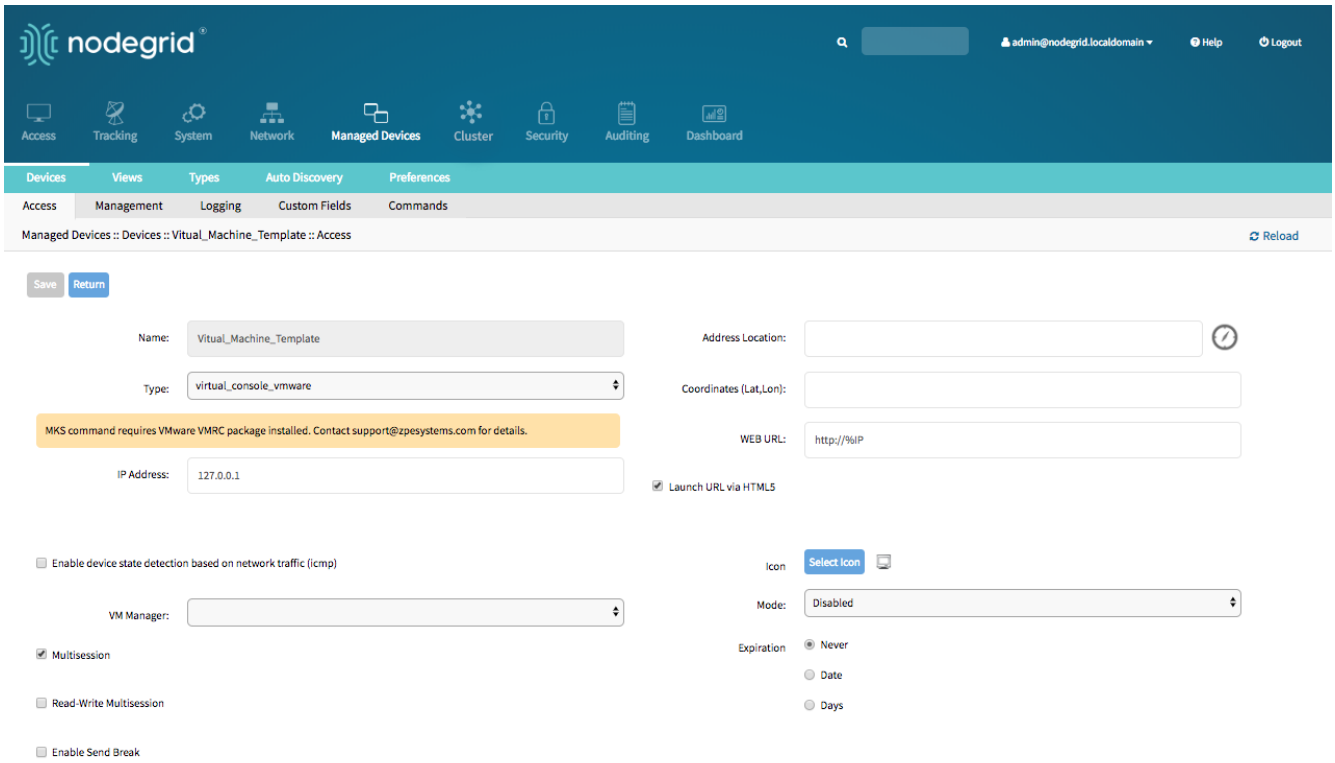
仮想マシンの自動検出

VMWare vCenter によって管理される仮想マシン、または ESXi 上で実行される仮想マシンを検出・管理することが可能です。このプロセスは、定期的に vCenter または ESXi ホストをスキャンし、新しく追加された仮想マシンを検出します。 仮想マシンは、*仮想*コンソール*vmware* または *仮想*シリアル*ポート*のいずれかのタイプとして追加できます。[付録VM サーバでの仮想シリアルポート (vSPC) の設定を参照して下さい。

> 注: ESXi 無料版はサポートされていません。

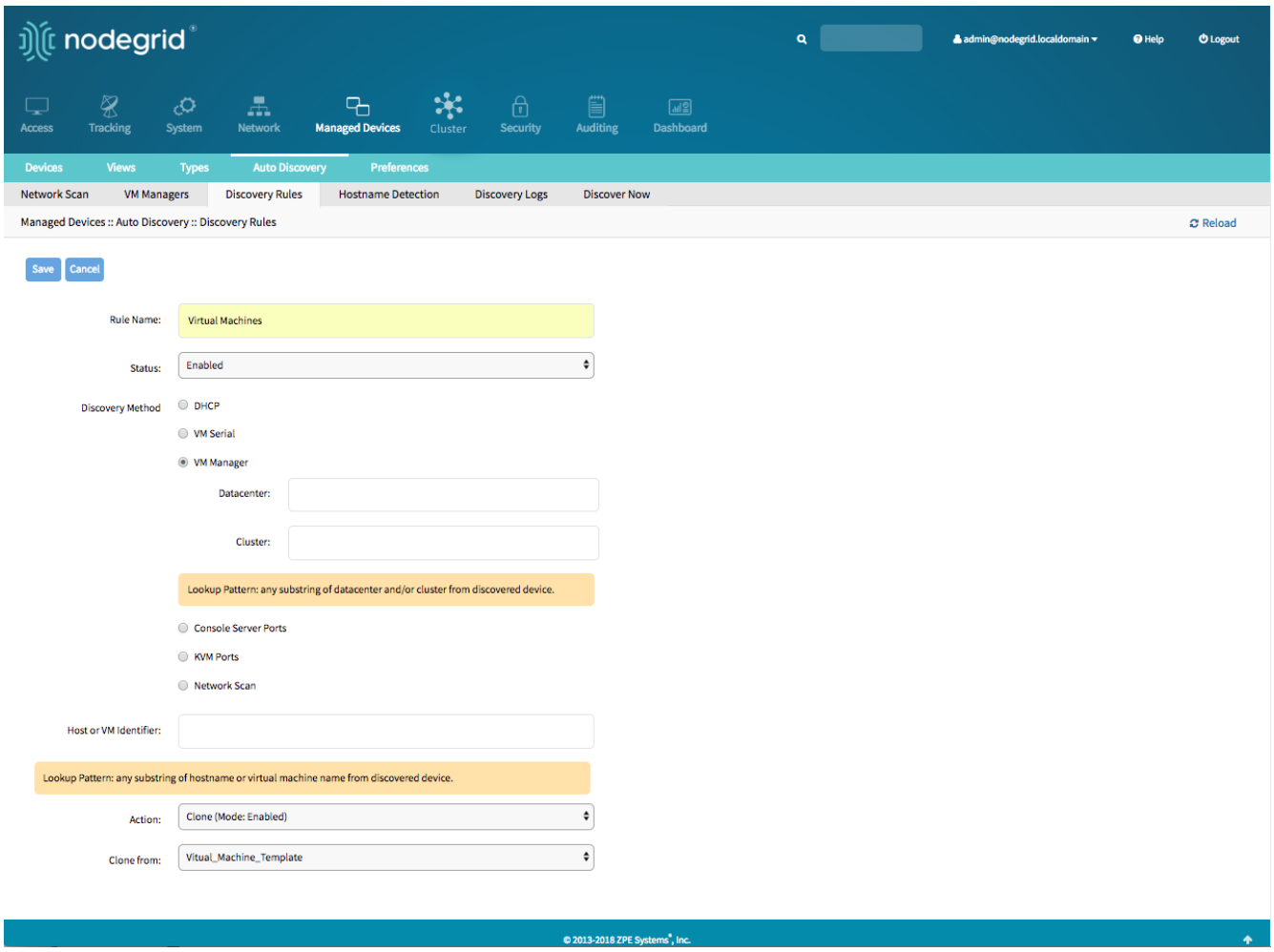
仮想マシンの自動検出 - WebUI

- テンプレートデバイスの作成
  - 移動先 `Managed Devices:: Devices`
  - `Add` ボタンをクリックして、デバイスをシステムに追加します。
  - 追加したいテンプレートの名前を入力します
  - IP アドレスに 127.0.0.1 と入力します
  - `Type` フィールドで、仮想マシンと同じタイプを選択します。指定可能な値: *仮想*コンソール *VMware*
  - ログイン時にユーザー資格情報を提供する場合、`username` と `password` を入力するか、 `Ask During Login` オプションを選択します
  - `Mode` *無効*を選択すると、デバイスがアクセスページに表示されないようにします
  - [保存] ボタンをクリックします。
  - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



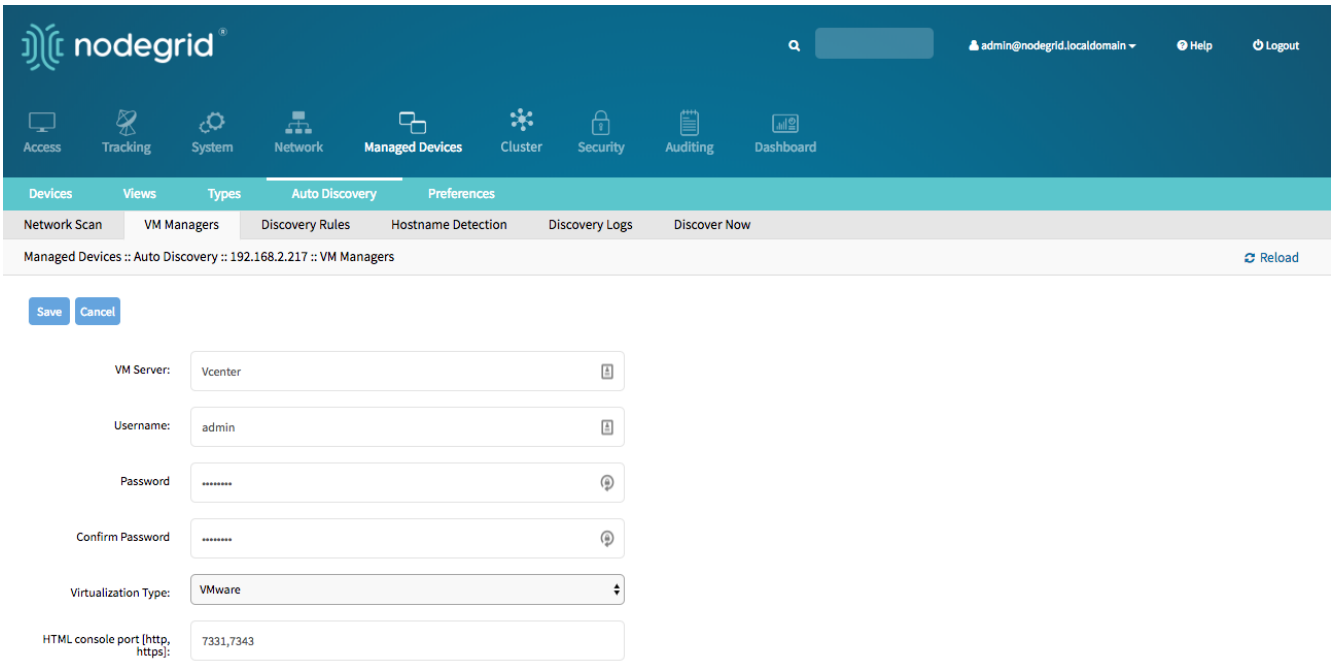
- 検出ルールを作成します
  - 移動先 `Managed Devices:: Auto Discovery:: Discovery Rules`

- `Add`をクリックして、新しい検出ルールを追加します
- 検出ルールのための`Name`を入力します
- 検出ルールのための`Status`を選択します。使用可能な値: *有効*、*無効*
- `Discovery Method`として、 *VM Manager*を選択します
- これらの特定エントリーのスキャンをフィルタするには、オプションで、フィールド`Datacenter`および `Cluster` を使用します。
- パラメータ`Host or VM Identifier`を使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
- `Action`については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローン(モード:有効)*、*クローン(モード:オンデマンド)*、*クローン(モード:検出済み)*、*検出されたデバイスを破棄*
- `Clone from field`で、前に作成されたテンプレートデバイスを選択します
- `Save`をクリックしてルールを作成します

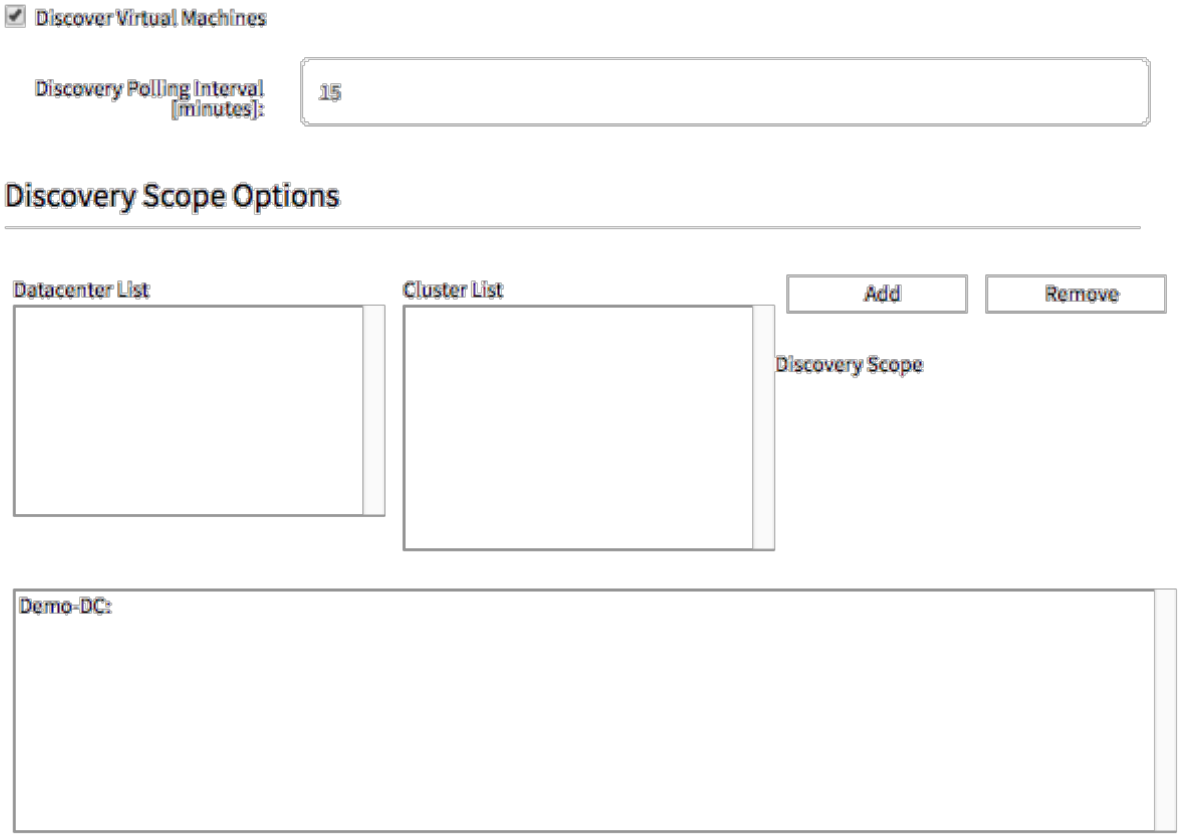


- VM Manager を定義
  - 移動先`Managed Devices :: Auto Discovery :: VM Managers`
  - `Add`をクリックして、新しい VM Manager を定義する
  - フィールド`VM Server`に vCenter/ESXi IP または FQDN を提供します
  - サーバの`Username`および `Password` を定義します

- 必要に応じて`HTML console port`を調整します
- 以下をクリック `Save`



- 仮想マシンの検出を可能にする
  - これで、Nodegrid Platform は、vCenter または ESXi システムに接続します。これには数分かかる場合があります
  - 新しく作成および接続されたものをクリックします `VM Manager`
  - 仮想マシン検出オプションを設定するためにオプション`Discover Virtual Machines`を有効化し、最小の`Data Center`および `Discovery Polling Interval`として定義します。
  - 以下をクリック `Save`



仮想マシンの自動検出 - CLI

- テンプレートデバイスの作成
  - 移動先 `/settings/devices`
  - `add` コマンドで、新規デバイスを作成する
  - `set` コマンドで、以下の設定を定義します
    - `name`
    - `type` 指定可能な値: *virtual_console_vmware*
    - `ip_address` 127.0.0.1 として
    - `mode` を *無効* に設定します
  - 変更を保存 `commit`
  - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Virtual_Machine_Template
[admin@nodegrid {devices}]# set type=virtual_console_vmware
[admin@nodegrid {devices}]# set ip_address=192.168.2.151
[admin@nodegrid {devices}]# set mode=disabled
[admin@nodegrid {devices}]# commit
```

- 検出ルールを作成します
  - 移動先 `/settings/auto_discovery/discovery_rules/`
  - `add` コマンドで、検索ルールを作成します
  - `set` コマンドで、以下の設定を定義します
    - `rule_name` 検出ルールの場合
    - `status` 検出されたルールの場合に考えられる値: *有効*、*無効*
    - `method` _vm_manager に設定 _
    - データセンターおよびクラスタに基づいてフィルタを定義するには、`datacenter` と `cluster` を使用します
    - `host_identifier` パラメータを使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります
  - `action` については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローンモード有効化*、*クローンモードオンデマンド*、*クローンモード検出*、*破棄*デバイス_
  - `clone_from` 以前に作成されたテンプレートデバイスに設定
  - 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/auto_discovery/discovery_rules/
[admin@nodegrid discovery_rules]# add
[admin@nodegrid {discovery_rules}]# set rule_name=Virtual_Machine
[admin@nodegrid {discovery_rules}]# set status=enabled
[admin@nodegrid {discovery_rules}]# set method=vm_manager
[admin@nodegrid {discovery_rules}]# set action=clone_mode_enabled
[admin@nodegrid {discovery_rules}]# set clone_from=Vitual_Machine_Template
[admin@nodegrid {discovery_rules}]# commit
```

- VM Manager を定義
  - 移動先 `/settings/auto_discovery/vm_managers/`
  - この `add` コマンドで、VM マネージャーを作成します
  - `set` コマンドで、以下の設定を定義します
    - `vm_server` : vCenter/ESXi IP または FQDN を提供します

        - `username`を定義し、 `password`
        - 必要に応じて`html_console_port`を調整します
    - 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/auto_discovery/vm_managers/
[admin@nodegrid vm_managers]# add
[admin@nodegrid {vm_managers}]# set vm_server=vCenter
[admin@nodegrid {vm_managers}]# set username=admin
[admin@nodegrid {vm_managers}]# set password=password
[admin@nodegrid {vm_managers}]# commit
```

- 仮想マシンの検出を可能にします
    - これで、Nodegrid Platform は、vCenter または ESXi システムに接続します。これには数分かかる場合があります
    - 新しく作成したものをクリックする `VM Manager`
    - 仮想マシン検出オプションを設定するためにオプション`Discover Virtual Machines`を有効化し、最小の`Data Center`および `Discovery Polling Interval`として定義します。
    - `Save` をクリック

```
[admin@nodegrid 192.168.2.217]# show
vm server: 192.168.2.217
username = Administrator@zpesystems.com
password = ********
type = VMware
html_console_port = 7331,7343
discover_virtual_machines = yes
interval_in_minutes = 15
discovery_scope =  Demo-DC!
```

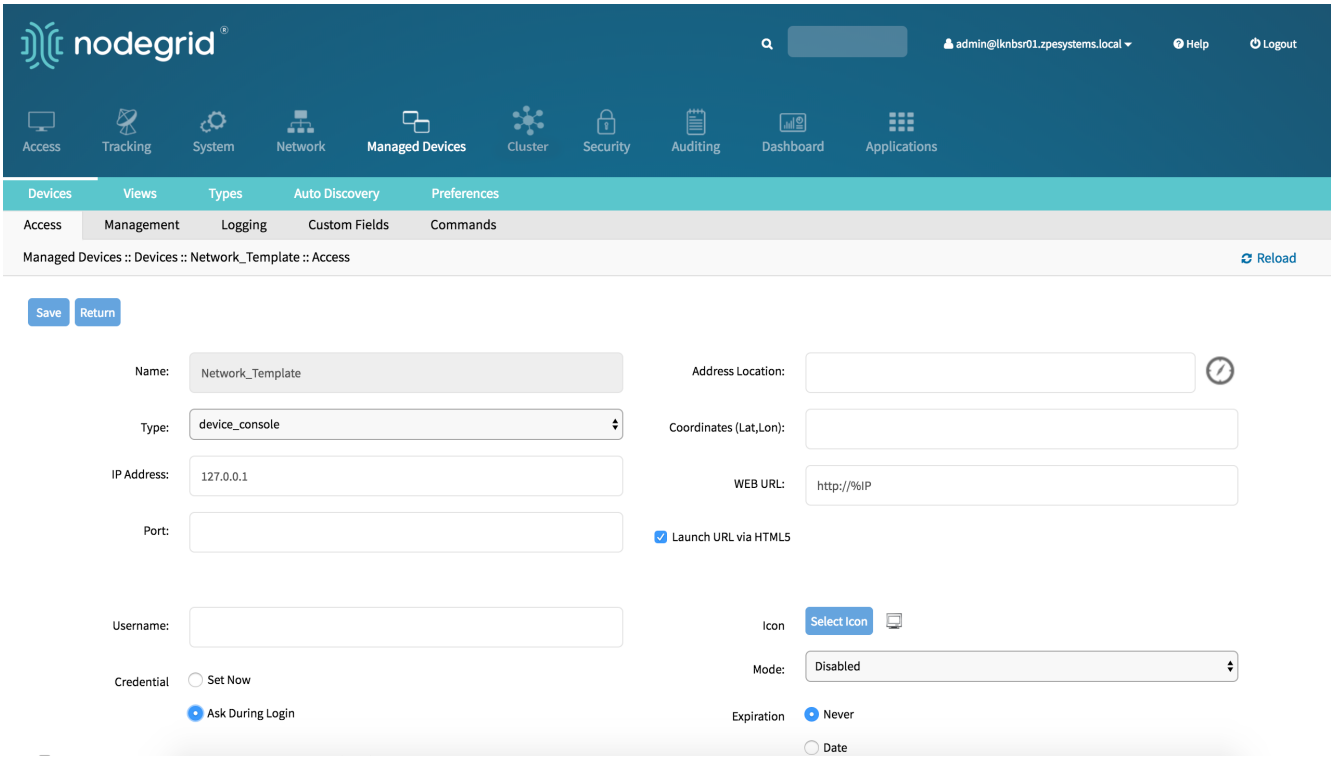
DHCP クライアントの自動検出

Nodegrid Platform は、管理ネットワーク内のクライアントの DHCP サーバとして使用できます。これらのデバイスは自動的に検出され、Nodegrid Platformに追加されます。 この機能は、ローカル Nodegrid Platform から DHCP リースを受け取る DHCP クライアントのみをサポートします。DHCP サーバの設定方法の詳細は、DHCP サーバを参照してください。

DHCP クライアントの自動検出 - Web UI

- テンプレートデバイスの作成
    - 移動先 `Managed Devices:: Devices`
    - ボタン`Add`をクリックして、デバイスをシステムに追加します。
    - 追加したいテンプレートの名前を入力します

- IP アドレスに 127.0.0.1 と入力します
- `Type` フィールドで、必要なマシンと同じタイプを選択します。
- ログイン時にユーザー資格情報を提供する場合、`username` と `password` を入力するか、`Ask During Login` オプションを選択します
- `Mode` *無効*を選択すると、デバイスがアクセスページに表示されないようにします
- [保存] ボタンをクリックします。
- オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。



- 検出ルールを作成します
  - 移動先 `Managed Devices:: Auto Discovery:: Discovery Rules`
  - `Add` をクリックして、新しい [検出ルール] を追加します
  - 検出ルールのための `Name` を入力します
  - 検出ルールのための `Status` を選択します。使用可能な値: *有効*、*無効*
  - `Discovery Method` として、 *DHCP*を選択する
  - オプションで、`MAC Address` を使用して、これらの特定エントリをフィルター処理する
  - パラメータ `Host or VM Identifier` を使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
  - `Action` については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローン(モード:有効)*、*クローン(モード:オンデマンド)*、*クローン(モード:検出済み)*、*検出されたデバイスを破棄*
  - `Clone from field` で、前に作成されたテンプレートデバイスを選択します
  - `Save` をクリックしてルールを作成します

ルールが作成され、DHCP アドレスを受け取るか、DHCP アドレスリースを更新するとすぐに、デバイスは自動的にシステムに追加されます。アドレスリース更新のデフォルト値は、10 分ごとです。

DHCP クライアントの自動検出 - CLI

- テンプレートデバイスの作成
  - 移動先 `/settings/devices`
  - `add` コマンドで、新規デバイスを作成します
  - `set` コマンドで、以下の設定を定義します
    - `name`
    - `type` 指定可能な値:
      device_console、*ilo*、*imm*、*drac*、*idrac6*、*ipmi1.5*、*impi2.0*、*ilom*、*cimc_ucs*、*netapp*、*infrabox*、*pdu**
    - `ip_address` 127.0.0.1 として
    - ログイン時にユーザー資格情報を提供する場合、`username` と `password` を設定するか、`Ask During Login` オプションを選択します
    - `mode` を*無効*に設定します
  - 変更を保存 `commit`
  - オプションで、この時デバイスの表示と動作を制御する設定を調整することができます。詳細については、デバイスの設定を参照してください。

```
[admin@nodegrid /]# cd /settings/devices
[admin@nodegrid devices]# add
[admin@nodegrid {devices}]# set name=Network_Template
[admin@nodegrid {devices}]# set type=device_console
[admin@nodegrid {devices}]# set ip_address=127.0.0.1
[admin@nodegrid {devices}]# set credential=ask_during_login

or

[admin@nodegrid {devices}]# set credential=set_now
[admin@nodegrid {devices}]# set username=admin password=admin

[admin@nodegrid {devices}]# set mode=disabled
[admin@nodegrid {devices}]# commit
```

- 検出ルールを作成します
  - 移動先 `/settings/auto_discovery/discovery_rules/`
  - `add` コマンドで、検索ルールを作成します
  - `set` コマンドで、以下の設定を定義します

        - `rule_name` 検出ルールの場合
        - `status` 検出されたルールの場合に考えられる値: *有効*、*無効*
        - `method` *dhcp*に設定します
        - オプションで、フィールド`mac_address`を使用して、これらの特定のエントリをフィルター処理します
        - `host_identifier` パラメータを使用して、フィルターをさらに適用することができます。値が指定されている場合は、ポート名の一部が値と一致する必要があります。
    - `action`については、新しいデバイス検出時に実行すべきアクションを選択します。指定可能な値: *クローンモード有効化*、*クローンモードオンデマンド*、*クローンモード検出*、*破棄デバイス_*
    - `clone_from` 以前に作成されたテンプレートデバイスに設定します
    - 変更を保存 `commit`

```
[admin@nodegrid /]# cd /settings/auto_discovery/discovery_rules/
[admin@nodegrid discovery_rules]# add
[admin@nodegrid {discovery_rules}]# set rule_name=Network_Scan
[admin@nodegrid {discovery_rules}]# set status=enabled
[admin@nodegrid {discovery_rules}]# set method=dhcp
[admin@nodegrid {discovery_rules}]# set mac_address=00:0C:29
[admin@nodegrid {discovery_rules}]# set action=clone_mode_enabled
[admin@nodegrid {discovery_rules}]# set clone_from=Network_Template
[admin@nodegrid {discovery_rules}]# commit
```

デバイスの設定

ほとんどのデバイスは、追加の設定オプションとセッティングに対応しています。このセクションでは、これらのセッティングとその設定方法について取り上げます。

ホスト名検出

この機能により、ログインプロンプト、プロンプト、またはバナーに基づいて、ターゲットデバイスのホスト名 (ネットワークまたはシリアル) を自動で検出できます。

デフォルトですでにいくつかのプローブがあり、以下のほとんどのデバイスのタイプと一致します: PDU、NetApp、Console Servers、Device Consoles、Service Processors。

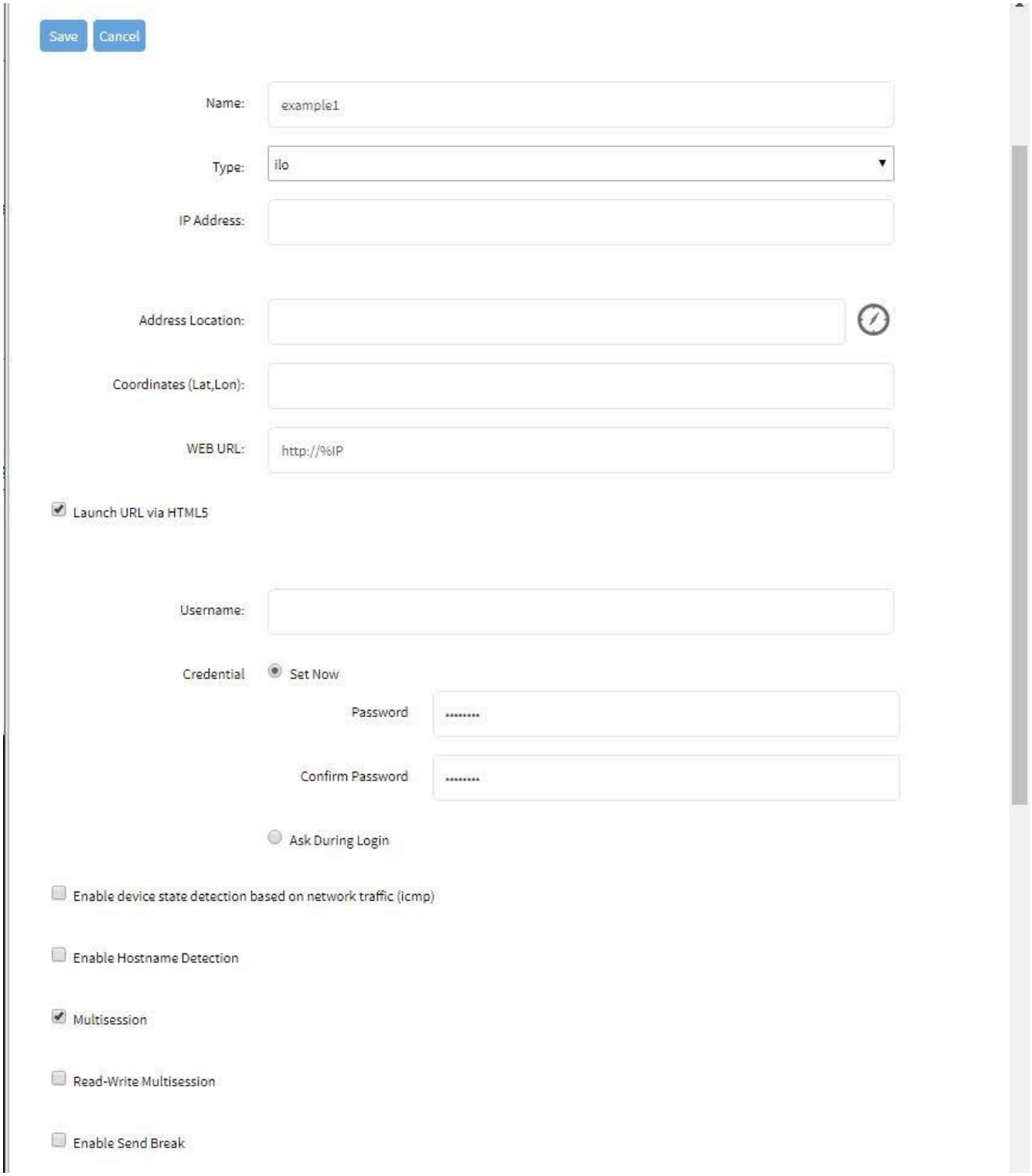
Nodegrid は最初のプローブを送信し、一致を待ちます。一致しない場合は、2 番目以降のプローブが一致するまで順に送信されます。一致すると、そのデバイスのプロービングは停止します。

ホスト名検出を設定する

ほとんどの場合、必要な設定は、ターゲットデバイスで機能を有効化することだけです。このためには、`Managed Devices` (WebUI) または`settings/devices/` (CLI) セクションのデバイスに移動し、その機能を有効にします。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲット デバイス] をクリックして、設定にアクセスします
- 横にあるボックスにチェックを入れます `Enable Hostname Detection`
- 設定を保存します



CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `enable_hostname_detection` を**はい**に設定します
- 変更を確定する

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set enable_hostname_detection=yes
[+admin@nodegrid /]# commit
```

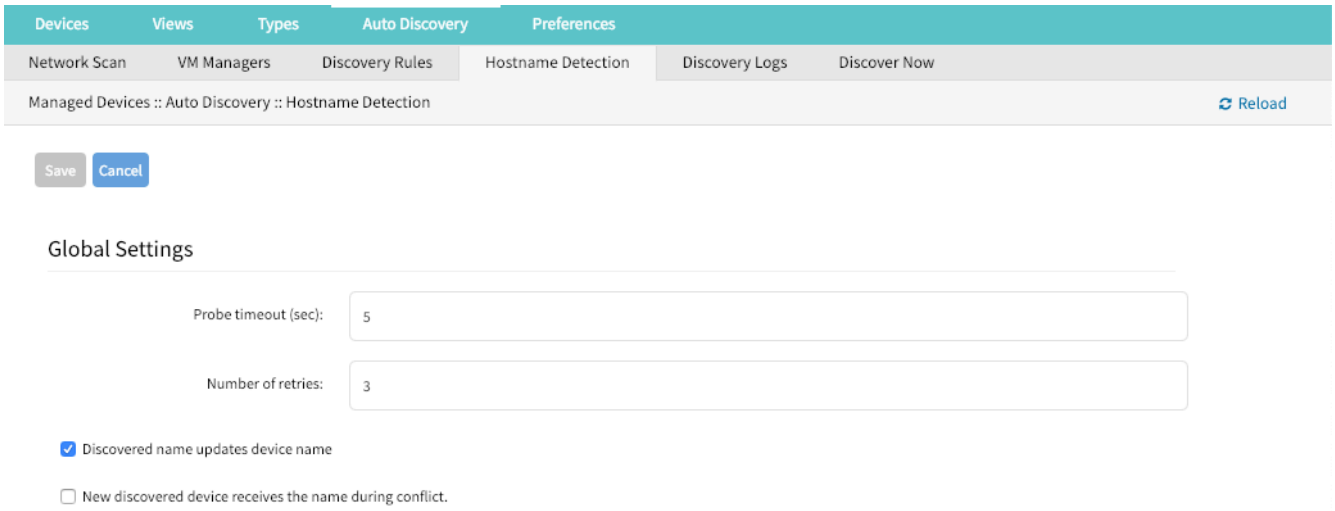
[ホスト名検出] のグローバル設定

また、各ターゲットデバイスの機能を有効にするために、以下の設定を `Managed Devices:Auto Discovery:Hostname Detection` (WebUI) または `/settings/auto_discovery/hostname_detection` (CLI) で調整することができます

以下のグローバルセッティングを設定できます

- プローブのタイムアウト: プローブ送信後に Nodegrid が出力を待機する際の、秒単位でのタイムアウト
- 再試行数: 出力が利用できなかった場合に、プローブがデバイスに再送される回数
- 検出された名前でデバイス名をアップデートする: この設定はデフォルトで有効になっていますが、無効にすることで、一致するものが検出された場合でもデバイス名は更新されなくなります。
- 新しく検出されたデバイスが、コンフリクト中にその名前を受信する: このオプションを有効にできます。複数のデバイスが同じ名前を持つ場合、この名前で検出された最新のデバイスがその名前を受け取ります。
- プローブ: デバイスに送信される [文字列 (テキスト)] の組み合わせです。その後、この出力は既存の [一致文字列] と照合されます。既存のプローブの調整や新しいプローブの作成が可能です。
- [一致] とは、プローブの出力と一致する RegEx 式です。一致した場合、ホスト名が抽出され、デバイス名が更新されます。既存の一致の調整や新しい一致の作成が可能です。

一般的な設定

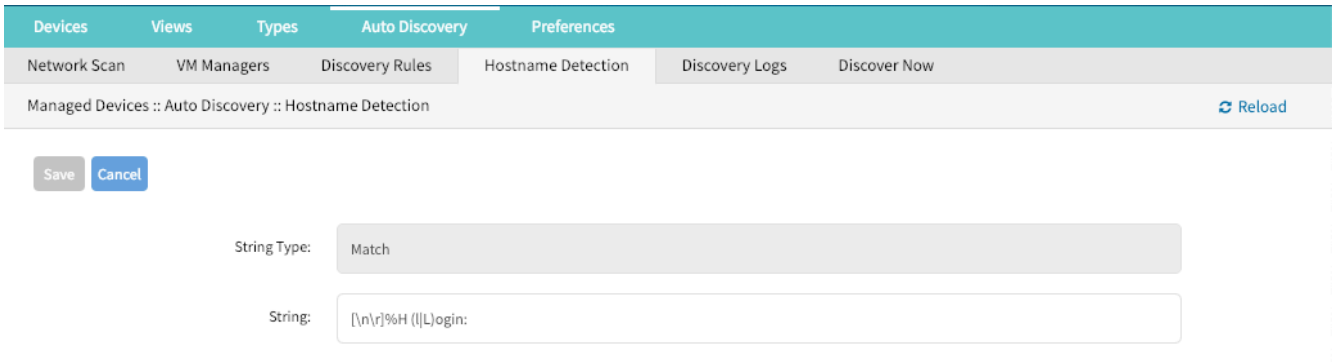


プローブまたは一致の作成

WebUI

- 移動先 `Managed Devices:Auto Discovery:Hostname Detection`
- [追加] をクリックする
- 以下から値を選択する `String Type`
- [一致] または [プローブ] になる [文字列] を指定します。

> 注: [一致] については、RegEx 式が許可されます。変数`%H`を使用してホスト名の位置を示します



CLI

- 移動先 `/settings/auto_discovery/hostname_detection/string_settings`
- タイプ `add`
- コマンド`set`を使用して、一致またはプローブとして`string_type`を定義する
- コマンド`set`を使用して、プローブまたは一致文字列を定義する
- 有効化し、変更を以下で保存する`commit`

> 注: [一致] については、RegEx 式が許可されます。変数`%H`を使用してホスト名の位置を示します

マルチセッション

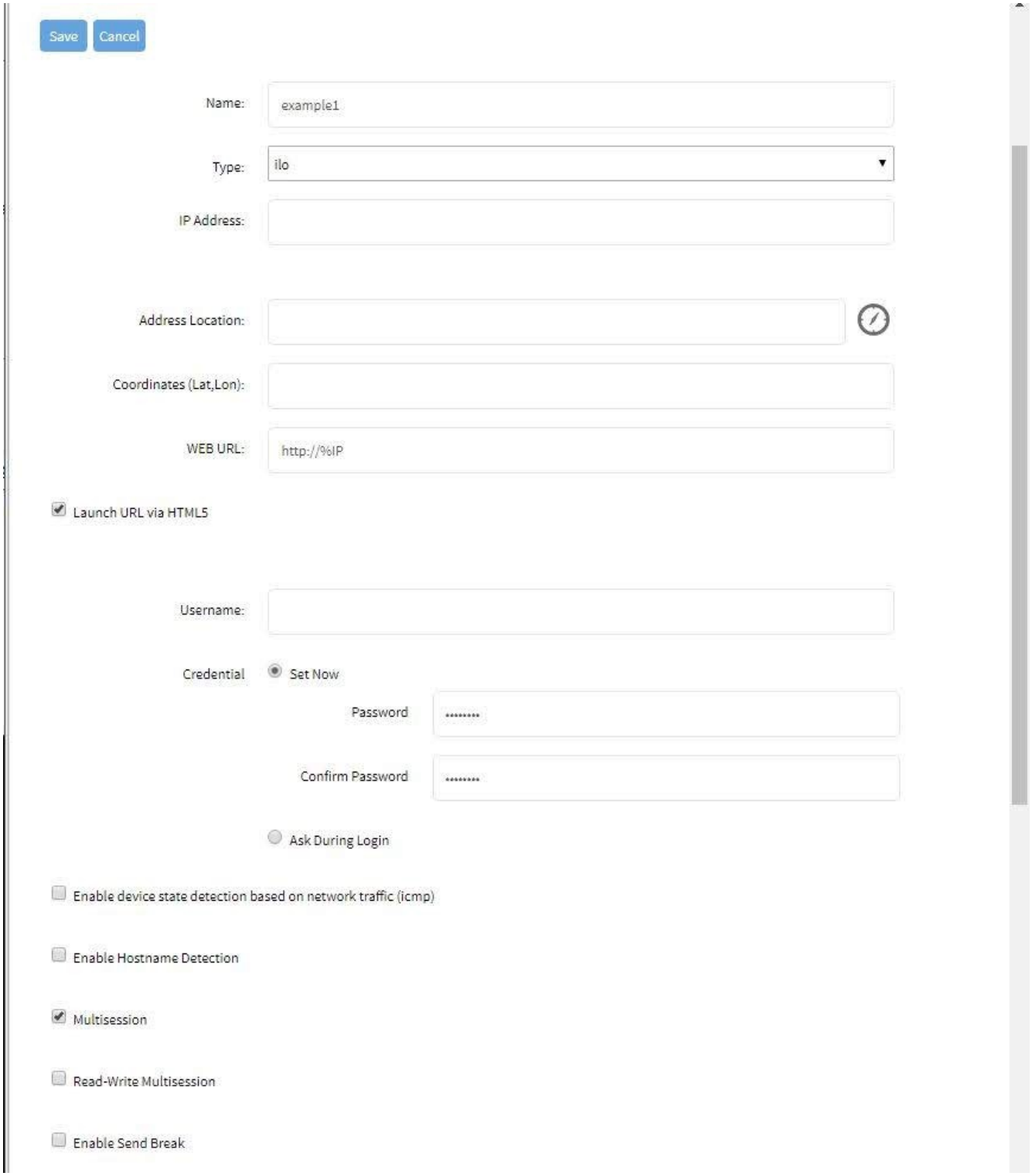
`Multisessions` 複数のユーザーが同時に同じデバイスにアクセスできるようにします。すべてのユーザーが同じ出力を表示できるようになります。デフォルトで、最初のユーザーに読み取り/書き込みアクセス権が付与され、他のすべてのユーザーにはセッションへの読み取り専用のアクセス権が付与されます。このオプション`Read-Write Multisession`を有効にすることでこの動作は変更でき、接続されているすべてのユーザーにセッションへの読み取り/書き込みアクセス権が付与されます。 この場合、一度に 1 人のユーザーだけに書き込みアクセス権が与えられ、セッションで最初にキー入力しようとしているユーザーに書き込み権が自動的に切り替わります。

コンソールセッションメニューから、セッション中に接続されている全ユーザーを確認できます (以下参照: [ブレーキシグナル])。この機能は、デバイスコンソールセッションで使用できます。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲット デバイス] をクリックして、設定にアクセスします

- 横にあるボックスにチェックを入れます `Multisessions`
- オプション：横にあるボックスにチェックを入れます `Read-Write Multisession`
- 設定を保存します



CLI

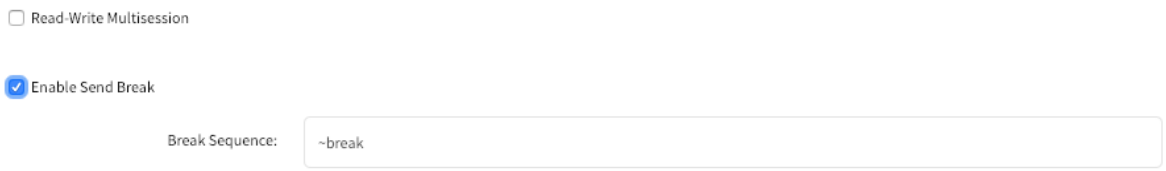
- 移動先 `/settings/devices/<Device Name>/access`
- `multisession` を**はい**に設定

- オプション: `write_multisession` を**はい**に設定
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set multisession=yes
[+admin@nodegrid /]# set read-write_multisession=yes
[+admin@nodegrid /]# commit
```

[ブレーキシグナル]

このオプションを使用すると、ssh コンソールセッションを介してブレーキシグナルを送信できます。この機能で、デバイスごとの有効化とブレーキシーケンスの設定が行えます。



WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲット デバイス] をクリックして、設定にアクセスします
- 横にあるボックスにチェックを入れます `Enable Send Break`
- 必要に応じて `Break Sequence` を調整します

CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `enable_send_break` を**はい**に設定
- 必要に応じて `break_sequence` を調整します
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set enable_send_break=yes
[+admin@nodegrid access]# set break_sequence=~break
[+admin@nodegrid access]# commit
```

エスケープシーケンス

エスケープシーケンスを使用すると、現在のセッションからエスケープしてメニューを表示したり、電源メニューの表示のような特定タスクを直接実行することが可能になります。

Nodegrid は、2 つのエスケープシーケンスをサポートしています。1 つは通常セッションメニュー用、2 つ目は (設定されている場合) ターゲットデバイスの直接電源制御を可能にする電源メニュー用です。(以下参照: 電力制御)

どちらのエスケープ シーケンスも、デフォルト値で事前設定されていますが、必要に応じて変更可能です。

	デフォルトシーケンス	
エスケープシーケンス	^Ec	CTRL+SHIFT+E c
電源制御キー	^O	CTRL+SHIFT+O

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲット デバイス] をクリックして、設定にアクセスします
- 必要に応じて、`Escape Sequence` または `Power Control Key` を調整します

CLI

- 移動先 `/settings/devices/<Device Name>/access`
- 必要に応じて、`escape_sequence` または `power_control_key` を調整します
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set escape_sequence=^Ec
[+admin@nodegrid access]# set power_control_key=^O
[+admin@nodegrid access]# commit
```

ユーザー認証を無効にする

デフォルトで、ターゲットデバイスにアクセスすると、ユーザーは最初に Nodegrid ユニットに対して認証を行う必要があり、その後デバイスを介して接続されます。特定の理由でこれが不要な場合、この機能で特定デバイスの Nodegrid 認証を無効にできます。

> 注: これにより、このデバイスのすべての Nodegrid 認証方法が無効になります。ターゲットデバイスで適切な認証メカニズムが設定されていることを確認します。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲット デバイス] をクリックして、設定にアクセスします
- 横にあるボックスにチェックを入れます `Skip authentication to access device (NONE authentication)`

CLI

- 移動先 `/settings/devices/<Device Name>/access`
- 認証を無効にするには、`skip_authentication_to_access_device`をはいに設定します
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set skip_authentication_to_access_device=yes
[+admin@nodegrid access]# commit
```

SSH / Telnetポート

これらの機能により、管理者はターゲットデバイス用の特定の ssh ポートまたは telnet ポートを定義できます。

デフォルトで各ターゲットデバイスには、基本ポートとしてポート 7000 とポート番号を持つ固有の telnet ポートが割り当てられています。ssh 接続の場合、デフォルトのポートが、すべてのデフォルトでの接続に使用されます。

SSH ポートと Telnet ポートは、必要に応じて調整できます。

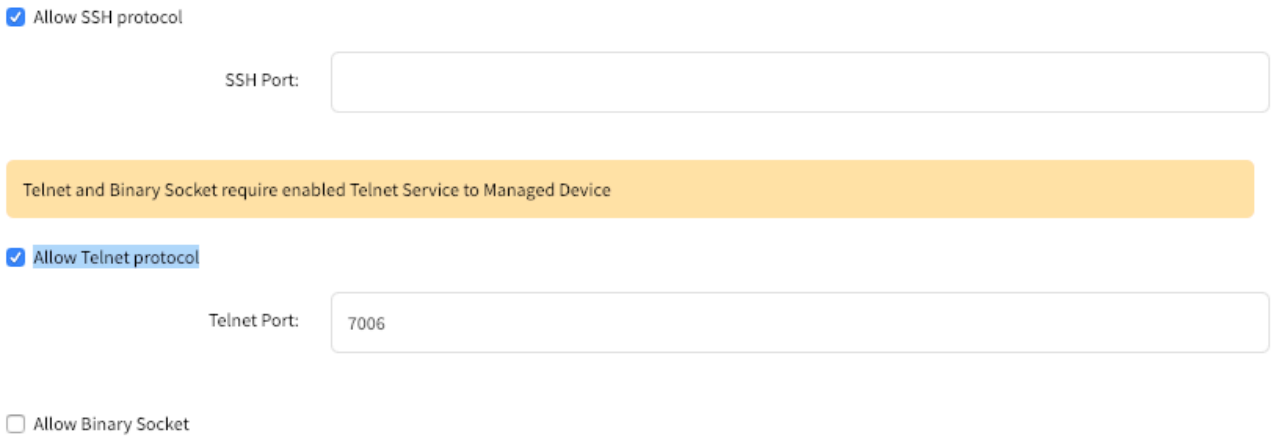
> 注: SSHv1 は非推奨です。現在は SSHv2 のみをサポートしています。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- ボックスの横の`Allow SSH protocol`または`Allow Telnet protocol`をクリックします。

  > 注: どちらのオプションもデフォルトで有効化されています。

  - 必要に応じてポート番号を指定または調整します



CLI - SSH

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、 `allow_ssh_protocol` を [はい] に設定します
- `set` コマンドで、 `ssh_port` 番号を定義します
- `commit` 変更

CLI - Telnet

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、 `allow_telnet_protocol` を [はい] に設定します
- `set` コマンドで、 `telnet_port` 番号を定義します
- `commit` 変更

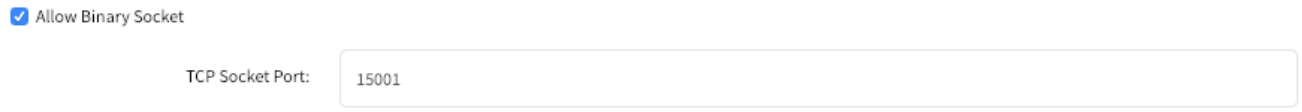
```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set allow_ssh_protocol=yes
[+admin@nodegrid access]#set ssh_port=17001
[+admin@nodegrid access]#set allow_telnet_protocol=yes
[+admin@nodegrid access]#set telnet_port=7001
[+admin@nodegrid access]#commit
```

バイナリソケット

バイナリ ソケット機能を使用すると、物理的に接続されているかのように、サードパーティシステムのデバイスへの直接アクセスが可能になります。信号は直接送信され、telnet または ssh プロトコルでカプセル化されません。特定のポートを割り当てる必要があります。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- 横にあるボックスにチェックを入れます `Allow Binary Socket`
- 必要に応じてポート番号を指定または調整します



CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、 `allow_binary_socket` を [はい] に設定します
- `set` コマンドで、 `tcp_socket_port` 番号を定義します
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set allow_binary_socket=yes
[+admin@nodegrid access]#set tcp_socket_port=15001
[+admin@nodegrid access]#commit
```

IP エイリアス

コンソールセッションは、WebUI や CLI から開始することも、ssh/telnet クライアントを介して直接開始することも可能です。ssh クライアントが使用される場合、特定のターゲットデバイスにアクセスするために、デフォルトでパラメータとしてターゲットデバイス名をパスします。

ポートエイリアスによって、ユーザーは IP アドレスを使用してターゲットデバイスに接続できます。各 IP エイリアスは、必要に応じて telnet ポートとバイナリポートの定義をサポートします。

Nodegrid ソリューションは、ターゲットデバイスごとに最大 2 つの IP アドレスエイリアスの割り当てが可能です。この機能は、IPv4、IPv6 アドレスに対応します。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- 横にあるボックスにチェックを入れます `Enable IP Alias`
    - 有効な IP アドレスを指定します
    - IP アドレスにアクセスできる有効なネットワークインターフェースを選択します。
    - `Allow the Telnet` 必要に応じて、インターフェース用のプロトコル
        - 必要に応じて、ポートを調整します
    - `Allow Binary Socket` 必要に応じて、直接アクセス用
        - 必要に応じて、ポートを調整します
- 次の手順を繰り返します `Enable Second IP Alias`

CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、 `enable_ip_alias` を [はい] に設定します
- `set` コマンドで、必要に応じて次の値を定義します
  - ip_alias - IP アドレス
  - インターフェース - 使用されるネットワークインターフェース
  - ip_alias_telnet - telnet の有効化/無効化
  - ip_alias_telnet_port ポート - 使用される Telnet ポート
  - ip_alias_binary - インターフェースがバイナリソケット接続をサポートする必要がある場合
- 次の手順を繰り返します `enable_second_ip_alias`
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set enable_ip_alias=yes
[+admin@nodegrid access]#set ip_alias=192.168.10.249
[+admin@nodegrid access]#set interface=eth0
[+admin@nodegrid access]#set ip_alias_telnet=yes
[+admin@nodegrid access]#set ip_alias_telnet_port=23
[+admin@nodegrid access]#set ip_alias_binary=no
[+admin@nodegrid access]#set ip_alias_binary_port=15001
[+admin@nodegrid access]#commit
```

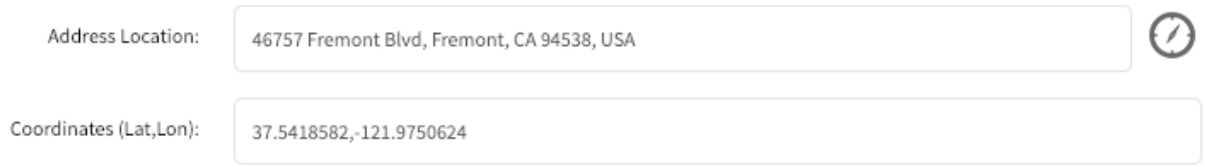
位置

各デバイスは、位置に関連付けることができます。位置の詳細は、デバイスとそのステータスをマップビューに表示するために使用されます。

位置は、住所の詳細、または経度と緯度の値を直接使用して定義できます。位置の値が住所を介して提供される場合、経度と緯度に翻訳するために、インターネット接続が必要になります。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- フィールド `Address Location` に住所を入力し、右のロケータアイコンを押して、緯度と経度の座標を識別します。
- または、有効な緯度と経度の座標を直接指定します



CLI

> 注: CLI は、住所を参照して有効な緯度と経度の座標に変換する機能をサポートしていません。

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、有効な緯度と経度の座標を指定します
- またはアドレスを指定します

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]# set coordinates="37.5418582,-121.9750624"
[+admin@nodegrid access]#set address_location="46757 Fremont Blvd, Fremont, CA
94538, USA"
[+admin@nodegrid access]#commit
```

Web URL

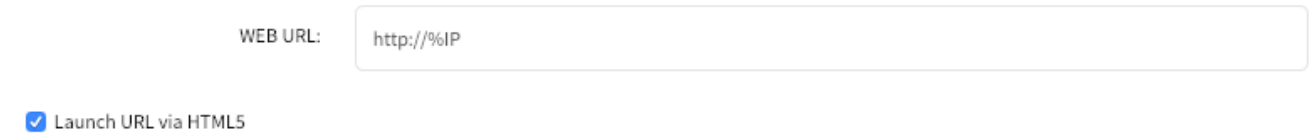
Web URL は、デバイスごとに定義できます。URL は、デフォルトで各デバイスで使用できる `Web` コマンドに使用されます。

すべての IP ベースのセッション用に定義されたデフォルトの URL は `http://%IP` で、これにより `%IP` は、各デバイスに対して定義されたIPアドレス値によって置き換えられます。 デフォルトで、URL はクライアントに転送される HTML5 フレーム内で開かれます。これにより、デバイスをネットワークに公開することなく、セキュリティで保護されていないデバイス Web インターフェースを通過することができます。

この機能を無効にすることによって、これを制御することができます。 `Launch URL via HTML5`

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- 必要に応じて、 `WEB URL` の値を調整します
- 必要に応じて、設定から HTML5 ウィンドウでの URL の起動を有効または無効にします `Launch URL via HTML5`



CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、以下を調整します `web_url`
- HTML5 ウィンドウでの URL の起動を、設定で有効または無効にします `launch_url_via_html5`

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]#set web_url="https://%IP"
[+admin@nodegrid access]#set launch_url_via_html5=yes
[+admin@nodegrid access]#commit
```

アイコン

各デバイスのタイプを表すアイコンを定義できます。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- `Select Icon` をクリックして希望のアイコンを選択し、デバイスアイコンを調整します

CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、`icon` を有効な値に調整します。 タブの使用-この時点でtab-tab を使って有効値のリストを表示します。
- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]#set icon=switch.png
[+admin@nodegrid access]#commit
```

モード

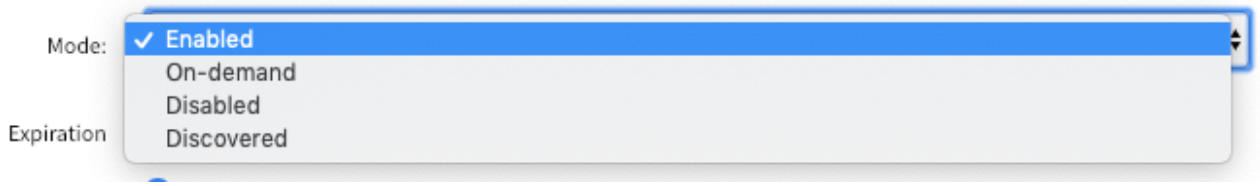
デバイス `Mode` は、Nodegrid Platform によるデバイスの管理方法と、デバイスのステータスを確認する方法を定義します。システムは 4 種類のモードに対応しています。

モード	説明
無効	このモードでは、デバイスは無効です。セッションを開くことができず、Nodegrid はデバイスに到達可能かどうかの確認を行いません。
有効	このモードでは、デバイスは有効で、セッションを開始できます。Nodegrid は、デバイスに到達可能かどうかを確認します。
オンデマンド	このモードでは、デバイスは有効で、セッションを開始できます。Nodegrid は、デバイスに到達可能かどうかを確認しません
検出済み	このモードでは、デバイスは無効です。セッションを開くことができず、Nodegrid はデバイスに到達可能かどうかの確認を行いません。このモードは、デバイスが検出プロセスを通じてシステムに追加されたことを示します。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- ドロップダウンメニューから有効なモードを選択して、デバイスモードを設定します

> 注: [検出済み] は、システムステータスのみなので選択できません



CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、 `mode` を以下のいずれかに調整します
  - 有効
  - 無効
  - オンデマンド
- 変更を `commit` します

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]#set mode=enabled
[+admin@nodegrid access]#commit
```

有効期限

各デバイスに有効期限または有効日数を指定すると、その日以降デバイスは自動的に使用できなくなります。そのデフォルト値は`Never`で、このような場合、デバイスとそのデータはシステム内に、 管理者ユーザーがそれを削除するまで保存されます。

日付 –デバイスは、指定された日付まで使用可能です。 その日以降は`Disabled`モードに設定され、管理者ユーザーはその後 10 日間はアクションを実行できます。10 日後、デバイスとそのデータはシステムから削除されます。

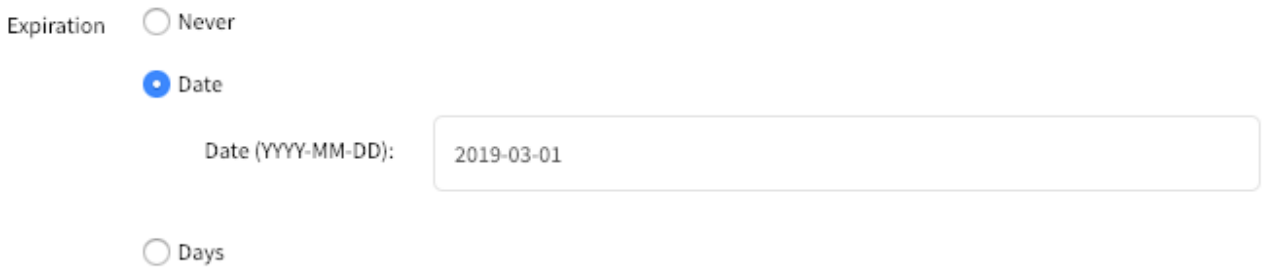
日付 – は、*タイムアウト* – に似ていますが、指定された日数を経過してもデバイスの設定が更新されない場合、デバイスとそのデータはシステムから削除されます。 これは、デバイスの使用とは関係ありません。

VM がコンスタントに追加、移動、削除される ESXi サーバと同期するために**日付**と**日数**の両方がほとんどの VM デバイスに適用され、Nodegrid に管理されるデバイスライセンスが使用可能になります。

> 注: この機能は、IP ベースのデバイスでのみ使用できます。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- `Never`、`Expiration Date`、または`Expiration Days`のいずれかを選択し、適切な値を指定します。
  - 日付: 日付は、年-月-日 (年年年-月月-日日) の形式で指定します
  - 日数： 値は 1 から 999999999 の間で指定します



CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set`コマンドで、 `expiration`を以下のいずれかに調整します
  - `never`
  - `date`
    - `set`コマンドで、有効期限日を指定します `expiration_date`
  - `days`
    - `set`コマンドで、0 から 9999999999 までの有効日数を指定します `expiration_days`

- `commit` 変更

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]#set expiration=date
[+admin@nodegrid access]#set expiration_date=2020-01-01
[+admin@nodegrid access]#commit

or

[admin@nodegrid /]#set expiration=days
[+admin@nodegrid access]#set expiration_days=5
[+admin@nodegrid access]#commit

or

[admin@nodegrid /]#set expiration=never
```

デバイスステータスの検出

Nodegrid は、**有効モード**のすべてのデバイスで、デバイスが現在使用可能かどうかを示すデバイスステータスの検出をサポートします。

シリアルデバイス

デフォルトで、Nodegrid はシリアルデバイスのDCD または CTS 信号を使用します。特定デバイスにこれらの信号が存在しない場合は、代わりにデータフローを使用するようにデバイスステータスの検出を変更できます。この場合、ステータスはデバイスによって送信される実際のデータに基づいて決定されます。

この機能を使用するには、機能`Enable device state detection based in data flow`を有効化する必要があります。

IPデバイス

IP ベースのデバイスのデフォルトのメカニズムは、アクティブな ssh セッションをデバイスに確立して監視します。

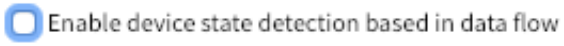
さらに、ICMP (ping) チェックを有効にして、デバイスが有効かどうかを確認できます。

この機能を使用するには、機能`Enable device state detection based on network traffic (icmp)`を有効化する必要があります。

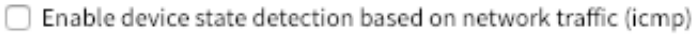
WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして、設定にアクセスします
- ボックスをチェックしてデバイスのステータス検出を有効にします

- シリアルの場合: `Enable device state detection based in data flow`



- その他のデバイスの場合: `Enable device state detection based on network traffic (icmp)`



CLI

- 移動先 `/settings/devices/<Device Name>/access`
- `set` コマンドで、デバイスのステータス検出を有効化します
  - シリアルの場合: `enable_device_state_detection_based_in_data_flow`
  - その他のデバイスの場合: `enable_device_state_detection_based_on_network_traffic`
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/access/
[admin@nodegrid /]#set enable_device_state_detection_based_in_data_flow=yes

or

[admin@nodegrid /]#set enable_device_state_detection_based_on_network_traffic=yes

[+admin@nodegrid access]#commit
```

[デバイスステータスの変更] へのカスタムスクリプトの実行

この機能を使用すると、ユーザーは特定デバイスのステータス変更にカスタムスクリプトを割り当てることができます。この機能は、イベント通知を超えたステータスの変更に対して特定のアクションを実行する必要がある場合に主に使用されます。

以下のステータス変更が使用されます:

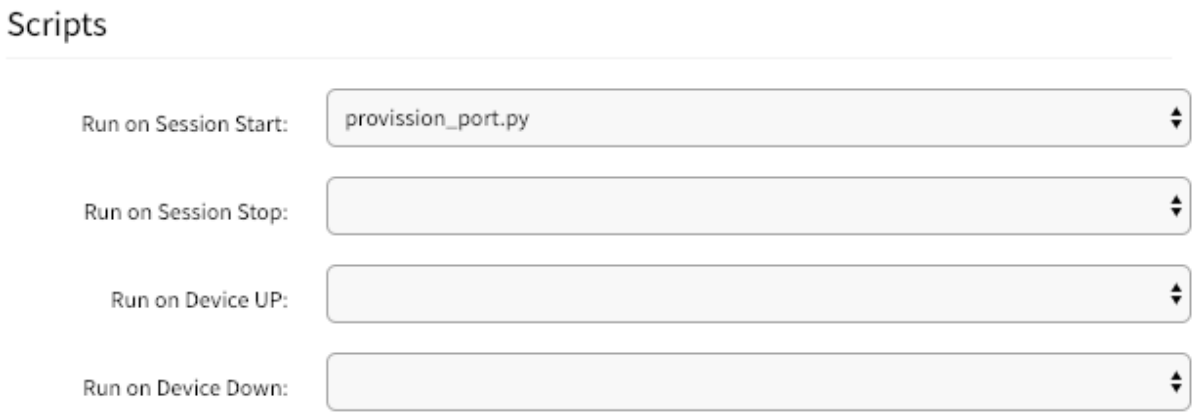
- セッション開始
- セッション停止
- デバイス アップ
- デバイス ダウン

スクリプトは、顧客またはプロフェッショナルサービス契約を通じて作成・提供される必要があります。

スクリプトをデバイスステータスに割り当てる前に、Nodegrid にコピーする必要があります。スクリプトをフォルダ `/etc/scripts/access` に配置する必要があります。 各スクリプトは、ユーザー権限で実行可能となる必要があります。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、以下に移動します `Management`
- `Scripts` セクションで、使用可能なスクリプトをデバイスステータスに割り当てます
  - `Run on Session Start`
  - `Run on Session Stop`
  - `Run on Device UP`
  - `Run on Device Down`



CLI

- 移動先 `/settings/devices/<Device Name>/management`
- `set` コマンドで、スクリプトをデバイスステータスに割り当てます
  - `on_session_start`
  - `on_session_stop`
  - `on_device_up`
  - `on_device_down`
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/management/
[admin@nodegrid /]#set on_session_start=sessionstart.sh
[+admin@nodegrid management]#commit
```

データロギング

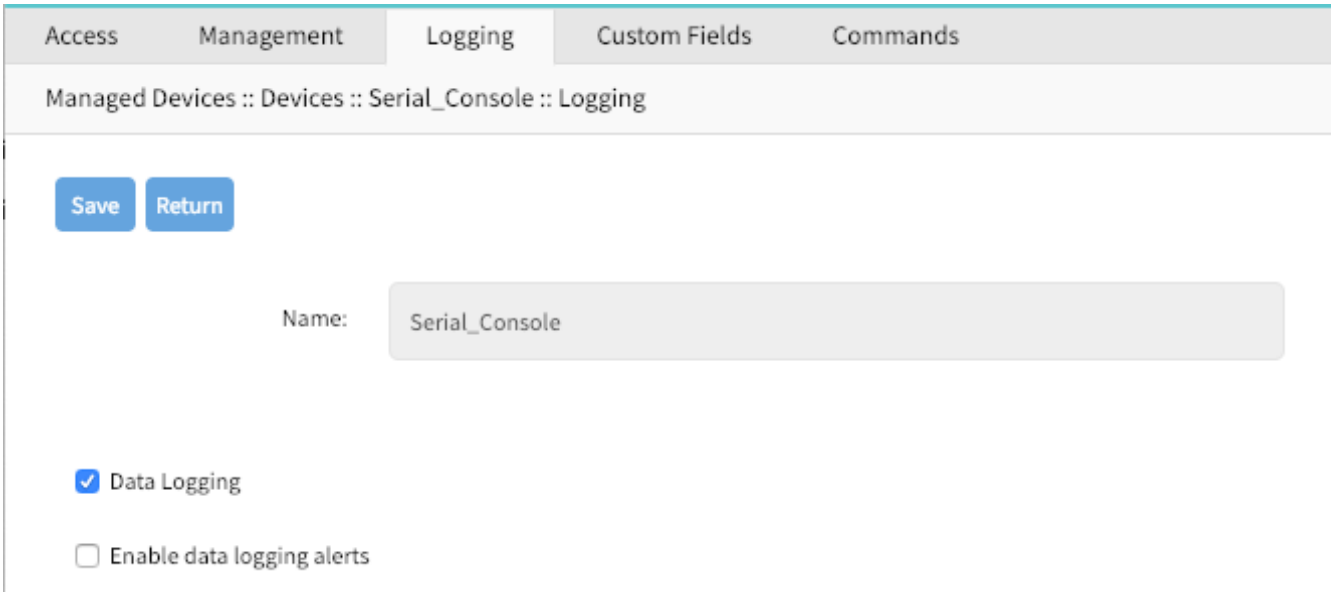
> 注: この機能は、シリアルセッションや ssh ベースのセッションなど、すべてのテキストベースのセッションで使用できます。

`Data Log`機能を有効にすることで、デバイスからデータログを収集するようにシステムを設定します。データログは、デバイスが送受信するすべての情報をキャプチャします。デバイスが有効モードの場合、ユーザーがデバイスに接続していない場合でも、データログはデータを収集します。これにより、コンソールセッションに送信されるシステムメッセージのロギングが可能になります。

収集されたデータログは、`Auditing`設定に応じて Nodegrid のローカルに保存されるか、リモートで保存されます。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、以下に移動します `Logging`
- オプションを有効にします `Data Logging`



CLI

- 移動先 `/settings/devices/<Device Name>/logging`
- `set`コマンドで、`data_logging`値を`yes`に変更します、または `no`
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/logging/
[admin@nodegrid /]#set data_logging=yes
[+admin@nodegrid logging]#commit
```

イベントロギング

> 注: この機能は、サービスプロセッサおよび IPMI セッションで使用できます。

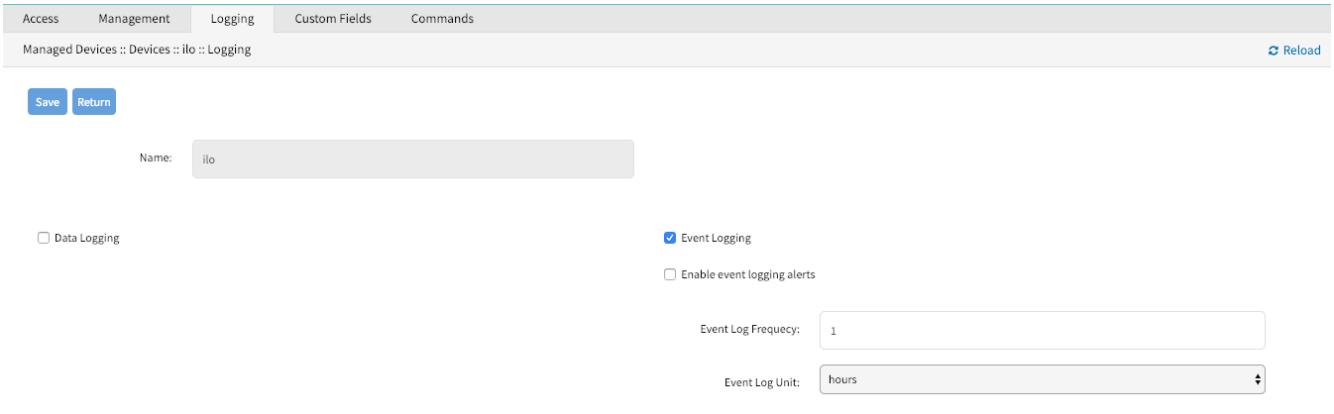
この機能を有効にすると、システムはサービスプロセッサイベントログデータ収集用に設定されます。収集されるデータのタイプは、サービスプロセスの機能と設定によって異なります。

設定`Log Frequency`と`Log unit`から、情報の収集頻度を制御できます。 収集間隔は、1 分から 9999 時間の範囲で指定できます。

収集されたデータログは、`Auditing`設定に応じて Nodegrid のローカルに保存されるか、リモートで保存されます。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、以下に移動します `Logging`
- オプションを有効にします `Event Logging`
- 必要に応じて、`Event Log Frequency`または`Event Log Unit`の値を調整します



CLI

- 移動先 `/settings/devices/<Device Name>/logging`
- `set`コマンドで、`event_logging`値を`yes`に変更します、または `no`
- 必要に応じて、`set`コマンドで`event_log_frequency`と`event_log_unit`を調整します
  - `event_log_frequency` 範囲 1 - 9999
  - `event_log_unit` オプション`hours`または `minutes`
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/ipmi/logging/
[admin@nodegrid /]#set event_logging=yes
[+admin@nodegrid logging]#set event_log_frequency=1
[+admin@nodegrid logging]#set event_log_unit=hours
[+admin@nodegrid logging]#commit
```

アラート文字列とカスタムスクリプト

データとイベントログ機能は、情報の収集だけでなく、イベント通知を作成しこれらのイベントに基づいてカスタムスクリプトを実行することができます。これは、アラート文字列を定義することでアーカイブされます。アラート文字列は、シンプルなテキストの一致、またはデータ収集時にデータソースストリームに対して評価される正規表現パターン文字列にすることができます。イベントは一致するたびに生成さ

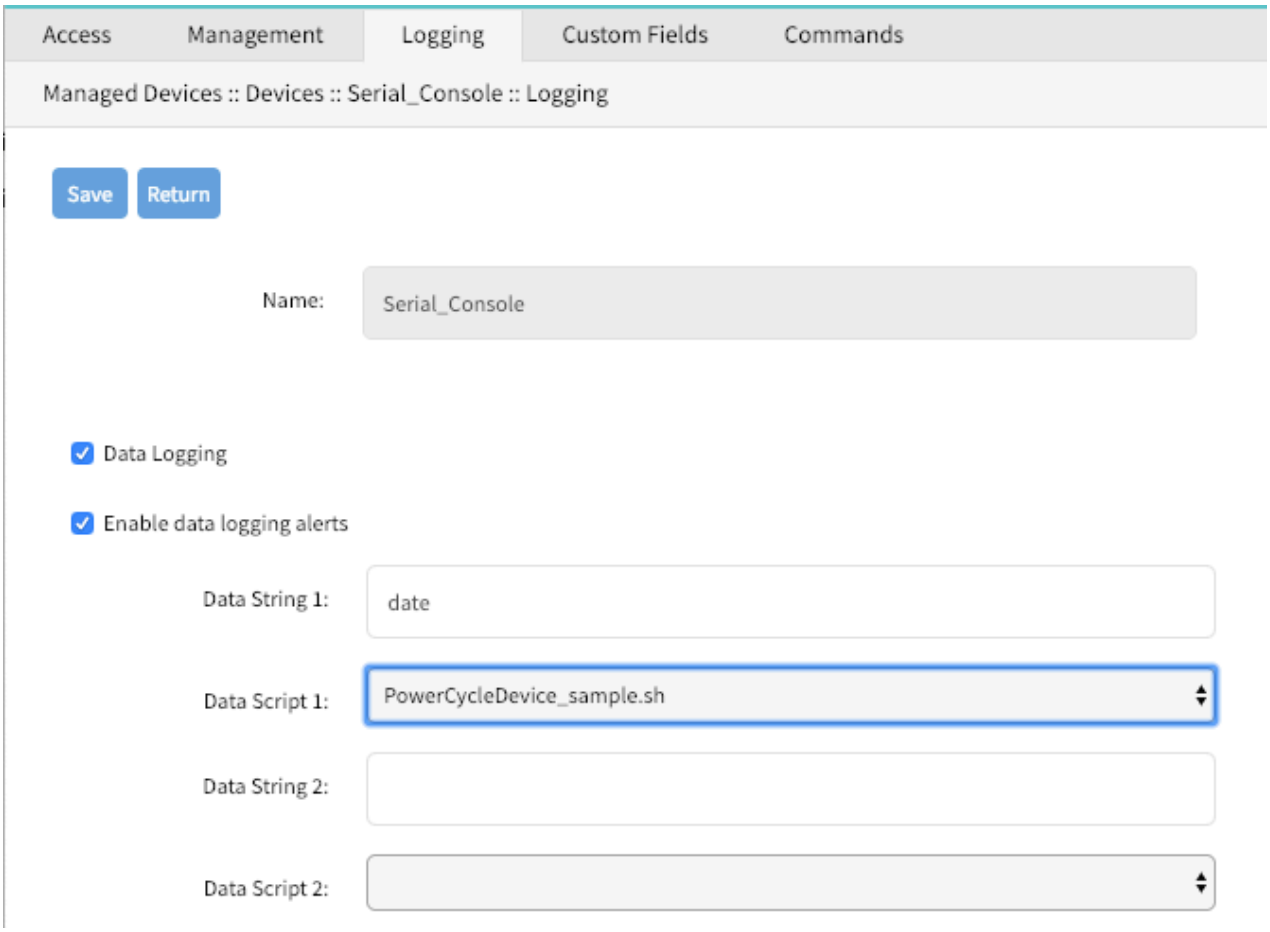
れます。

スクリプトは、顧客またはプロフェッショナルサービス契約を通じて作成・提供される必要があります。

スクリプトをデバイスステータスに割り当てる前に、Nodegrid にコピーする必要があります。スクリプトは、データログイベント用フォルダ `/etc/scripts/datalog`、またはイベントログ用フォルダ `/etc/scripts/events` に配置する必要があります。各スクリプトは、ユーザー権限で実行可能となる必要があります。

WebUI - データロギングアラート

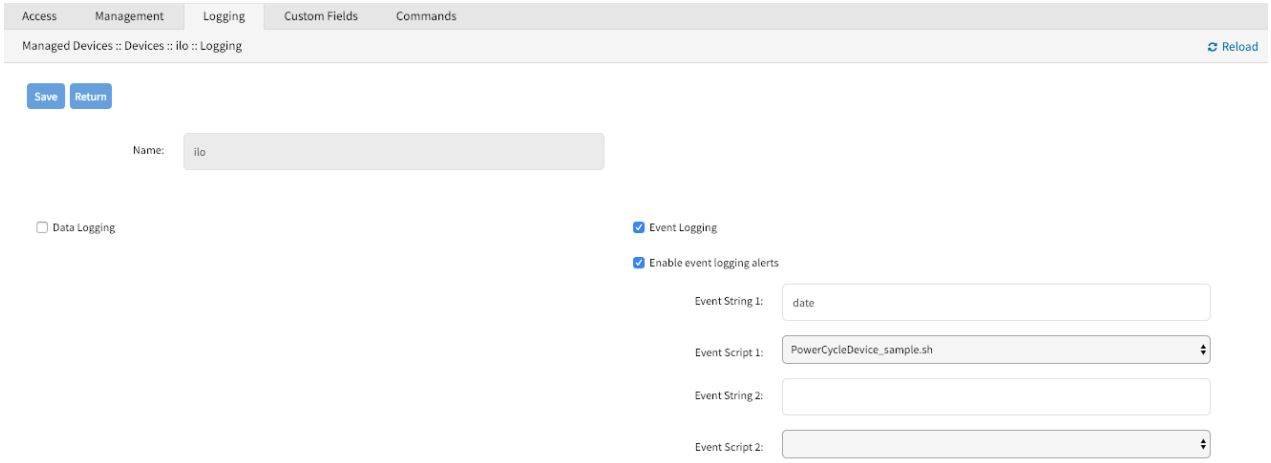
- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、以下に移動します `Logging`
- オプションを有効にします `Data Logging`
- オプションを有効にします `Enable data logging alerts`
- データストリームに一致する `Data String` 分をひとつ定義します
- カスタムスクリプトを実行する必要がある場合、定義された `Data Script` の使用可能なスクリプトを選択します



WebUI - イベントログアラート

- 移動先 `Managed Devices:: Devices`

- [ターゲットデバイス] をクリックして設定にアクセスし、 `Logging` に移動します
- オプションを有効にします `Event Logging`
- オプションを有効にします `Enable event logging alerts`
- データストリームに一致する `Event String` 分をひとつ定義します
- カスタムスクリプトを実行する必要がある場合、定義された `Event Script` の使用可能なスクリプトを選択します



CLI - Data Logging Alerts

- 移動先 `/settings/devices/<Device Name>/logging`
- `set` コマンドで、`data_logging` 値を `yes` に変更します
- `set` コマンドで、`enable_data_logging_alerts` 値を `yes` に変更します
- データストリームに一致する文字列 `data_string_1` または正規表現を定義します
- カスタムスクリプトを実行する必要がある場合に、使用可能なスクリプト `data_script_1` を定義します
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/Device_Console_Serial/logging/
[admin@nodegrid /]#set data_logging=yes
[+admin@nodegrid logging]#set enable_data_logging_alerts=yes
[+admin@nodegrid logging]#set data_string_1="String"
[+admin@nodegrid logging]#set data_script_1=ShutdownDevice_sample.sh
[+admin@nodegrid logging]#commit
```

CLI - Event Logging Alerts

- 移動先 `/settings/devices/<Device Name>/logging`
- `set` コマンドで、`event_logging` 値を以下に変更します `yes`
- 必要に応じて、`set` コマンドで `event_log_frequency` と `event_log_unit` を調整します
  - `event_log_frequency` 範囲 1 - 9999
  - `event_log_unit` オプション `hours` または `minutes`

- `set`コマンドで、`enable_event_logging_alerts`値を `yes` に変更します
- データストリームに一致する文字列`event_string_1`または正規表現を定義します
- カスタムスクリプトを実行する必要がある場合に、使用可能なスクリプト`event_script_1`を定義します
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/ipmi/logging/
[admin@nodegrid /]#set event_logging=yes
[+admin@nodegrid logging]#set event_log_frequency=1
[+admin@nodegrid logging]#set event_log_unit=hours
[+admin@nodegrid logging]#set enable_event_logging_alerts=yes
[+admin@nodegrid logging]#set event_string_1="String"
[+admin@nodegrid logging]#set event_script_1=PowerCycleDevice_sample.sh
[+admin@nodegrid logging]#commit
```

カスタムフィールド

カスタムフィールドを使用すると、ユーザーはデバイスに追加情報を割り当てることができます。この情報は、デバイスの概要ページで各デバイスに表示され、完全に検索できます。

カスタム情報は、キー/値のペアとして保存されます。

WebUI

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、 `Custom Fields`に移動します
- `Add`をクリックして、新しいカスタムフィールドを作成します
    - フィールド名を指定します
    - フィールド値を指定します

CLI

- 移動先 `/settings/devices/<Device Name>/custom_fields`
- `add`コマンドで、新しいカスタムフィールドを作成します
- `set`コマンドで、`field_name`を定義します `field_value`
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/Serial_Console/custom_fields/
[admin@nodegrid /]#add
[+admin@nodegrid custom_fields]#set field_name=Custom_Field_Example
[+admin@nodegrid custom_fields]#set field_value="A Value"
[+admin@nodegrid custom_fields]#commit
```

コマンドとカスタムコマンド

各デバイスのタイプは、ユーザーがデバイスにアクセスして操作できるコマンドのコレクションを提供します。大半のユーザーにとってデフォルト設定で十分なため、推奨されるオプションです。デフォルト設定が十分でない場合は、管理者ユーザーがこれを無効にするか、既存のコマンドを変更して、デフォルトで有効化されていない既存のコマンドを追加するか、カスタムコマンドをデバイスに割り当てることができます。コマンド機能で行われた変更は、すべてのユーザーに影響を与えるので注意が必要です。管理者ユーザーが特定のユーザーまたはグループで特定コマンドの使用を制限したい場合、ユーザーとグループの認証によりこれを実行できます。

デバイスで使用可能なコマンドは、デバイスのタイプによって異なります。例えば、(サービスプロセッサ KVM セッションサポートを有効にする) `KVM` コマンドは、サービスプロセッサデバイスでのみ使用でき、出力コマンドはデバイスの全タイプで使用できます。

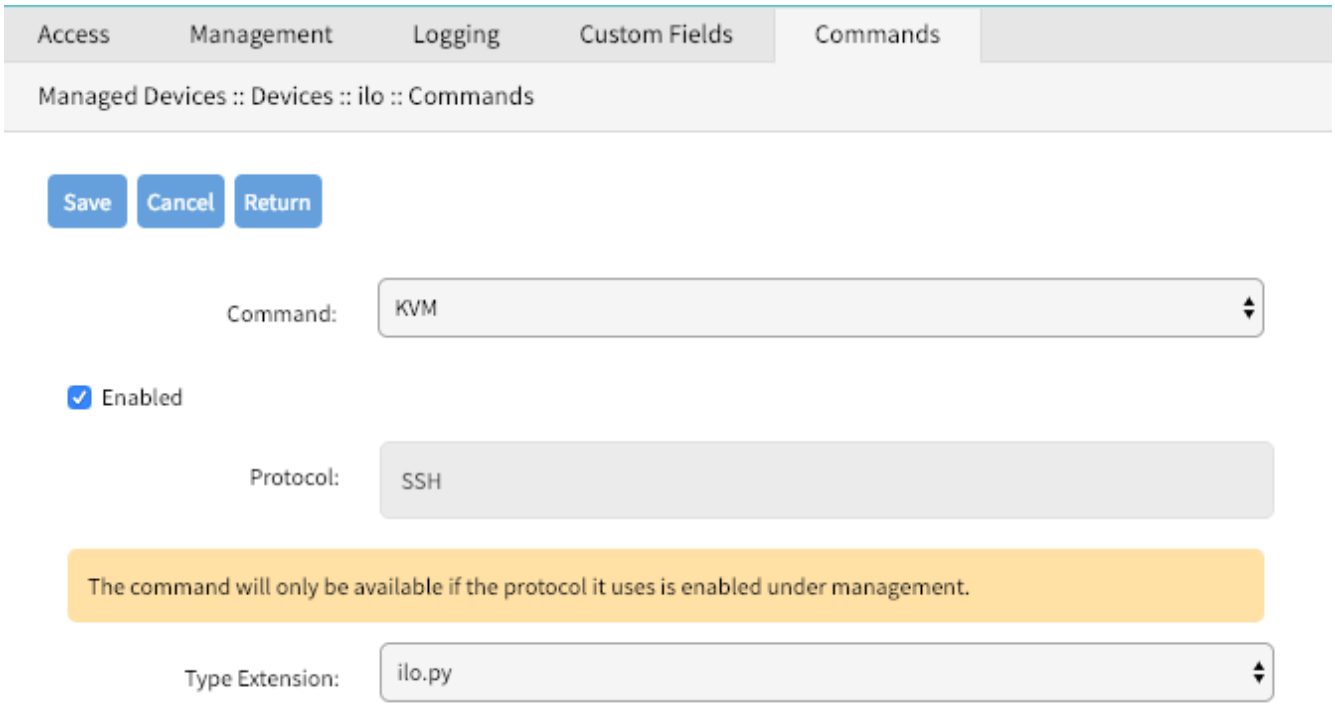
カスタムコマンドは、すべてのタイプのデバイスで使用でき、カスタムスクリプトを介して提供されます。カスタムコマンドは、追加のセッションオプションから、デバイスで実行する必要がある特定のカスタムタスクまで、幅広い各種機能に対応できます。

スクリプトは、顧客またはプロフェッショナルサービス契約を通じて作成・提供される必要があります。

> 注: カスタムコマンドは WebUI と CLI を介して実行できますが、現在カスタムコマンドは、WEBUI ではなく CLI のみにフィードバックと出力を提供できます。

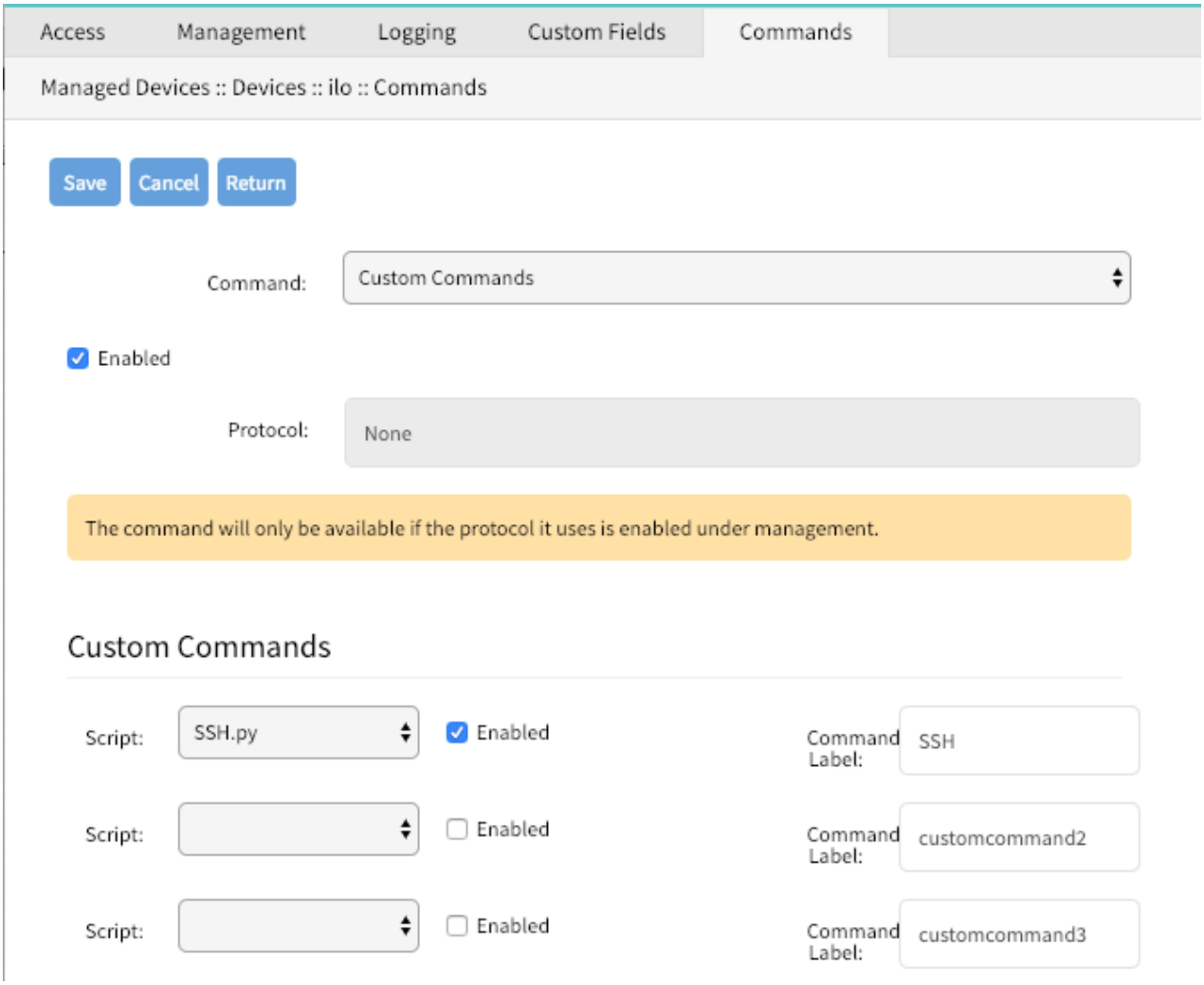
WebUI - ジェネリック

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、以下に移動します `Commands`
- `Add` をクリックしてコマンドを選択し、デバイスに関連付けます
- 既存のコマンドをクリックして、コマンドを変更または無効にします



WebUI - カスタムコマンド

- 移動先 `Managed Devices:: Devices`
- [ターゲットデバイス] をクリックして設定にアクセスし、以下に移動します `Commands`
- `Add` をクリックして選択します `Custom Commands`
    - 使用可能なスクリプトのリストから `Script` を選択します
    - [有効化] をクリックして、特定のコマンドを有効化します
    - スクリプト内のコマンドオプションに一致するようにコマンドラベルを調整します



CLI - Custom Commands

- 移動先 `/settings/devices/<Device Name>/commands`
- `add` コマンドで、新しいカスタムフィールドを作成します
- `set` コマンドで、`field_name` を定義します `field_value`
- `commit` で変更を保存します

```
[admin@nodegrid /]# /settings/devices/Serial_Console/commands/
[admin@nodegrid /]#add
[+admin@nodegrid commands]#set command=custom_commands
[+admin@nodegrid commands]#set custom_command_enabled1=yes
[+admin@nodegrid commands]#set custom_command_script1=SSH.py
[+admin@nodegrid commands]#set custom_command_label1=SSH
[+admin@nodegrid commands]#commit
```

ツリービューの設定

管理対象デバイスで :: 管理者は、デバイスを関連付けることができるツリー構造を調整および作成できます。この機能は、ユーザーがデバイスを検出してアクセスするのに役立つ特定の組織構造/物理構造を表示するのに役立ちます。

さらに、グループを使用して、ラックやルームレベルなどの監視値を集計するために使用できます。

デバイスのタイプ

デバイスのタイプの設定から、管理者は既存のデバイスのタイプのカスタマイズされたバージョンを調整または作成できます。これは、デバイスのタイプのデフォルト値が、顧客の現在のデフォルト値と一致しない場合に役立ちます。

既存のデバイスのタイプを複製、編集、または削除することで、デフォルトの通信プロトコルなどの値を必要に応じて調整できます。これらの設定は、現在特定のデバイスタイプを使用しているすべてのデバイスで、自動的に有効になります。

設定

`Preference`メニューで、管理者は`Power Menu`および`Session Preferences`オプションをさらに定義できます。 これらはグローバル設定であり、すべてのセッションに影響します。

電源メニュー設定

電源メニューの設定オプションで、管理者はコンソールセッションに表示される電源メニューの順序とラベル付けを定義できます。

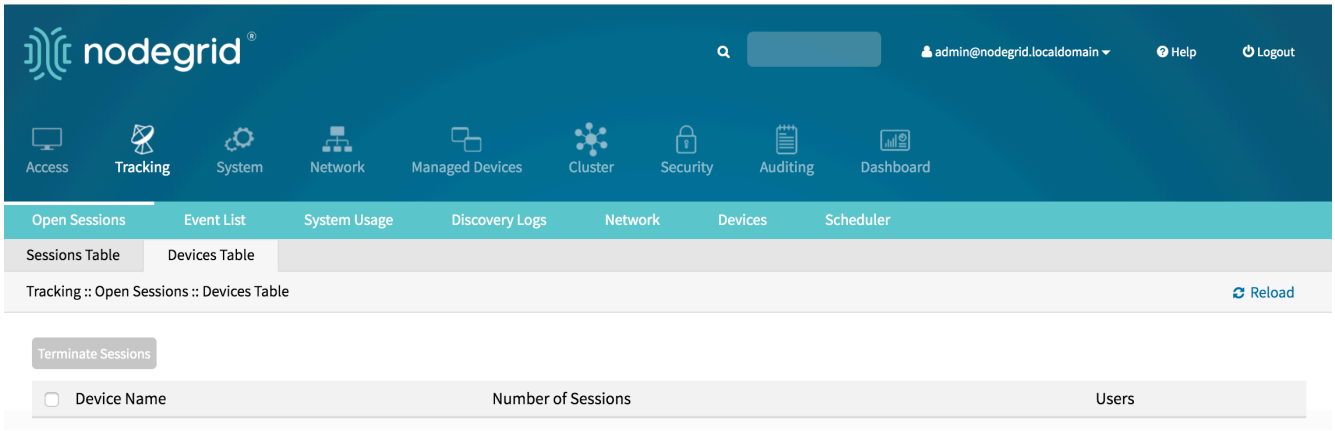
セッションの設定

セッションの設定セクションは、コンソールセッションのセッション`Disconnect HotKey`を定義できます。 通常この機能は、コンソールセッション (カスケードコンソールセッションとも呼ばれる) 内からコンソールセッションを開始する場合に役立ちます。この場合、チェーン内のすべてのセッションを終了させることなく、特定のコンソールセッションだけを終了することが困難な場合があります。ホットキーは、チェーン内の一定量のコンソールセッションから特に切断するためのオプションをユーザーに提供します。現在のセッションから始めて、チェーンをバックアップします。

デフォルトで値は定義されていません。

トラッキング

*トラッキング機能*は、システムおよびオープンセッション、イベントリスト、ルーティングテーブル、システム使用状況、検出ログ、LLDP、シリアル統計などの接続デバイスに関する情報を提供します。



オープンセッション

`Open Sessions`ページには、接続されているユーザーとデバイスセッションの概要が表示されます。

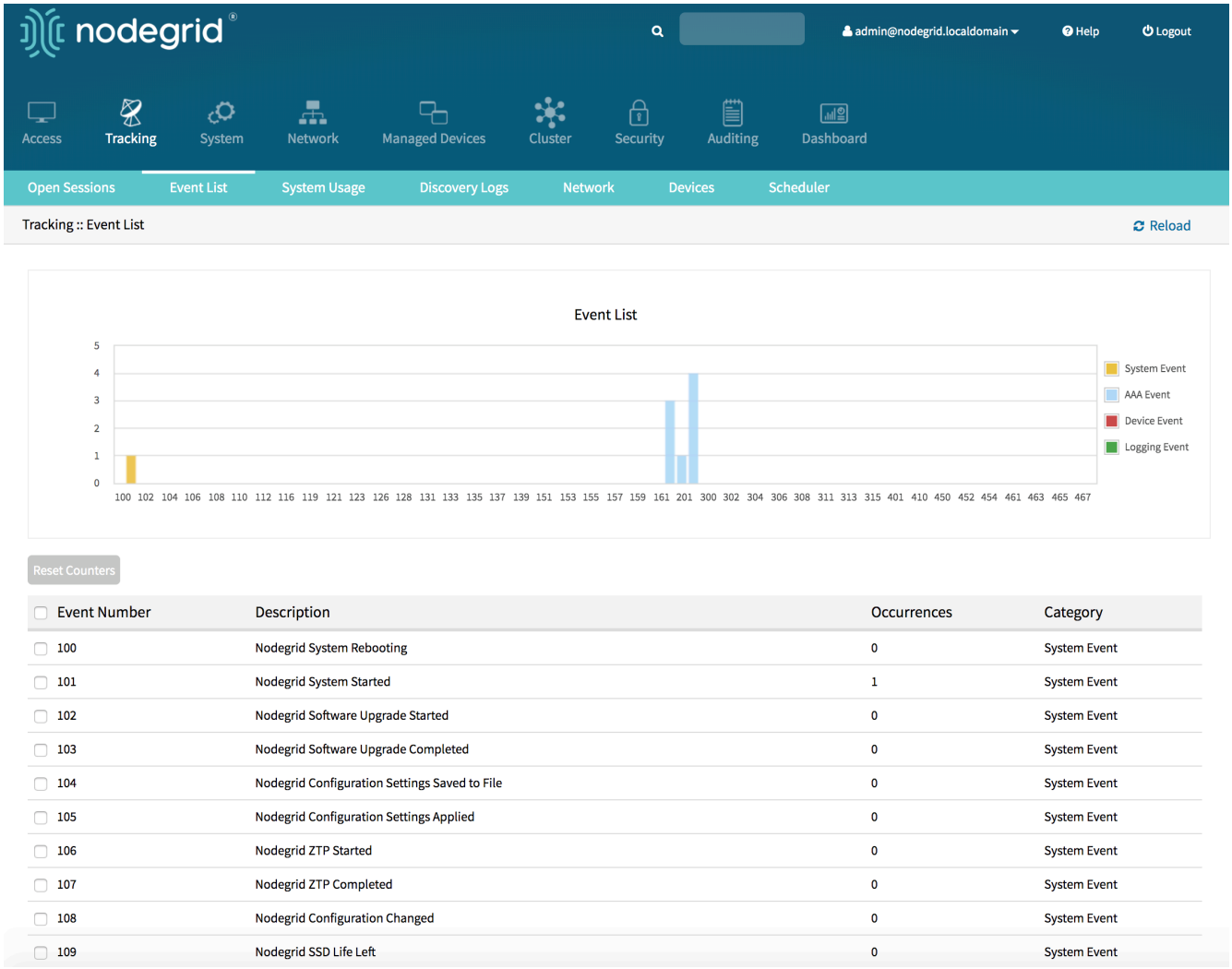
`Sessions Table`メニューには、システムに有効に接続しているすべてのユーザーが表示され、それらのユーザーがどこから接続しているか、またその滞在時間を表示します。

`Device Table`メニューには、有効なデバイスセッション、接続されているセッションの量、および接続されているユーザーに関する情報が表示されます。

ユーザーが承認グループによるアクセス許可を持っている場合、そのユーザーはセッションを終了することができます。

イベントリスト

`Event List`メニューは、システムイベント発生に関する統計情報を 表示します。イベントを選択して、現在のカウンタをリセットできます。



Nodegrid システムに登録されたイベントの全リストは以下のとおりです。必要に応じて、新しいイベントを追加できます。

	イベント番号	説明	発生	カテゴリ
	100	Nodegrid システム再起動中	0	システムイベント
	101	Nodegrid システム開始	1	システムイベント
	102	Nodegrid ソフトウェアのアップグレード開始	0	システムイベント
	103	Nodegrid ソフトウェアのアップグレード完了	0	システムイベント
	104	Nodegrid コンフィギュレーション設定をファイルに保存	0	システムイベント

	105	Nodegrid コンフィギュレーション設定を適用	0	システムイベント
	106	Nodegrid ZTP開始	0	システムイベント
	107	Nodegrid ZTP完了	0	システムイベント
	108	Nodegrid コンフィギュレーション変更	0	システムイベント
	109	Nodegrid SSDに残る寿命	0	システムイベント
	110	Nodegrid ローカルユーザーをシステムデータストアに追加	0	システムイベント
	111	Nodegrid ローカルユーザーをシステムデータストアから削除	0	システムイベント
	112	Nodegrid ローカルユーザーをシステムデータストアで改変	0	システムイベント
	115	Nodegrid セッション終了	0	システムイベント
	116	Nodegrid セッション タイムアウト済み	0	システムイベント
	118	Nodegrid 電源状態が変更した	0	システムイベント
	119	ユーザーが Nodegrid 電源の可聴アラームを停止	0	システムイベント
	120	Nodegrid 利用率超過	0	システムイベント
	121	Nodegrid サーマルスロットル温度上昇	0	システムイベント
	122	Nodegrid サーマルスロットル温度下降中	0	システムイベント
	123	Nodegrid サーマル温度警告	0	システムイベント
	124	Nodegrid サーマル温度臨界	0	システムイベ

				ント
	126	Nodegrid ファンのステータス変更	0	システムイベント
	127	ユーザーが Nodegrid ファンの可聴アラームを停止	0	システムイベント
	128	Nodegrid のローカルシリアルポートの総数が不一致	0	システムイベント
	130	Nodegrid ライセンスを追加	0	システムイベント
	131	Nodegrid ライセンスを除去	0	システムイベント
	132	Nodegrid ライセンスのコンフリクト	0	システムイベント
	133	Nodegrid ライセンス不足	0	システムイベント
	134	Nodegrid ライセンスの期限切れ	0	システムイベント
	135	Nodegrid Shell 開始	0	システムイベント
	136	Nodegrid Shell 停止	0	システムイベント
	137	Nodegrid Sudo実行	0	システムイベント
	138	Nodegrid SMS実行	0	システムイベント
	139	Nodegrid SMS無効	0	システムイベント
	150	Nodegridクラスタピアがオンライン	0	システムイベント
	151	Nodegridクラスタピアがオフライン	0	システムイベント
	152	Nodegridクラスタでピアがサインオン	0	システムイベント

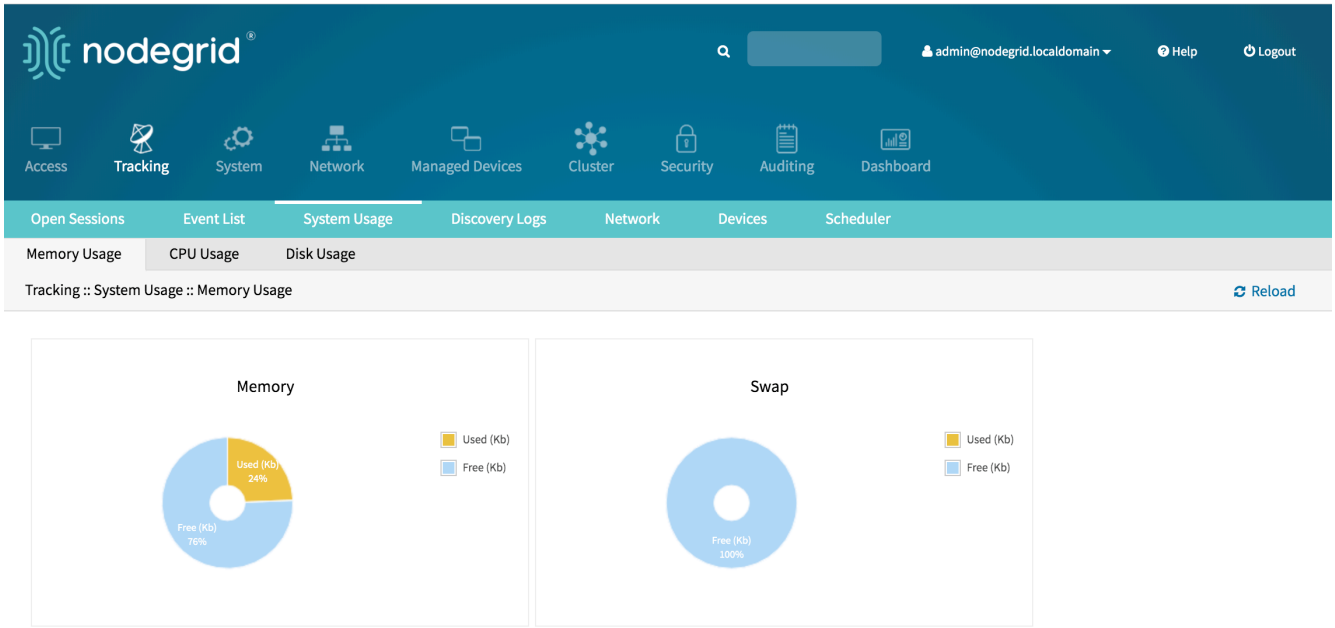
	153	Nodegridクラスタでピアがサインオフ	0	システムイベント
	154	Nodegridクラスタピアを除去	0	システムイベント
	155	Nodegridクラスタピアがコーディネーターになりました	0	システムイベント
	156	Nodegridクラスタコーディネーターがピアになりました	0	システムイベント
	157	Nodegridクラスタコーディネーターを削除	0	システムイベント
	158	Nodegridクラスタコーディネーターを創出	0	システムイベント
	159	Nodegridクラスタピアを設定	0	システムイベント
	160	Nodegrid検索が利用不可能	0	システムイベント
	161	Nodegrid検索を復元した	0	システムイベント
	200	Nodegridユーザーがログインした	3	AAAイベント
	201	Nodegridユーザーがログアウトした	1	AAAイベント
	202	Nodegridシステムの認証の失敗	4	AAAイベント
	300	Nodegridデバイスのセッションが開始	0	デバイスイベント
	301	Nodegridデバイスのセッションが停止	0	デバイスイベント
	302	Nodegridデバイスを作成	0	デバイスイベント
	303	Nodegridデバイスを削除	0	デバイスイベント
	304	Nodegridデバイス リネーム済み	0	デバイスイベント
	305	Nodegridデバイスをクローン化	0	デバイスイベント

	306	Nodegridデバイスをアップ	0	デバイスイベント
	307	Nodegridデバイスをダウン	0	デバイスイベント
	308	Nodegridデバイスセッション終了	0	デバイスイベント
	310	Nodegrid電源オンのコマンドがデバイスで実行された	0	デバイスイベント
	311	Nodegrid電源オフのコマンドがデバイスで実行された	0	デバイスイベント
	312	Nodegridパワーサイクルのコマンドがデバイスで実行された	0	デバイスイベント
	313	Nodegridサスペンド コマンドがデバイスで実行された	0	デバイスイベント
	314	Nodegridリセット コマンドがデバイスで実行された	0	デバイスイベント
	315	Nodegridシャットダウン コマンドがデバイスで実行された	0	デバイスイベント
	400	Nodegridシステム アラート検出	0	ロギングイベント
	401	Nodegrid アラート文字列をデバイスセッションで検出	0	ロギングイベント
	402	Nodegrid イベントログ文字列をデバイスイベントログで検出	0	ロギングイベント
	410	NodegridシステムNFS 失敗	0	ロギングイベント
	411	NodegridシステムNFS 回復済み	0	ロギングイベント
	450	Nodegridデータポイント状態高臨界	0	ロギングイベント
	451	Nodegridデータポイント状態高警告	0	ロギングイベント
				ロギングイベ

	452	Nodegridデータポイント状態正常	0	ント
	453	Nodegridデータポイント状態低警告	0	ロギングイベント
	454	Nodegridデータポイント状態低臨界	0	ロギングイベント
	460	Nodegridドアをアンロックしました	0	ロギングイベント
	461	Nodegridドアをロックした	0	ロギングイベント
	462	Nodegridドア開放	0	ロギングイベント
	463	Nodegridドア閉鎖	0	ロギングイベント
	464	Nodegridドアアクセス拒否	0	ロギングイベント
	465	Nodegridドアアラームがアクティブ	0	ロギングイベント
	466	Nodegridドアアラームが非アクティブ	0	ロギングイベント
	467	Nodegrid PoE電源異常	0	ロギングイベント
	468	Nodegrid PoE電力バジェット超過	0	ロギングイベント

システム使用率

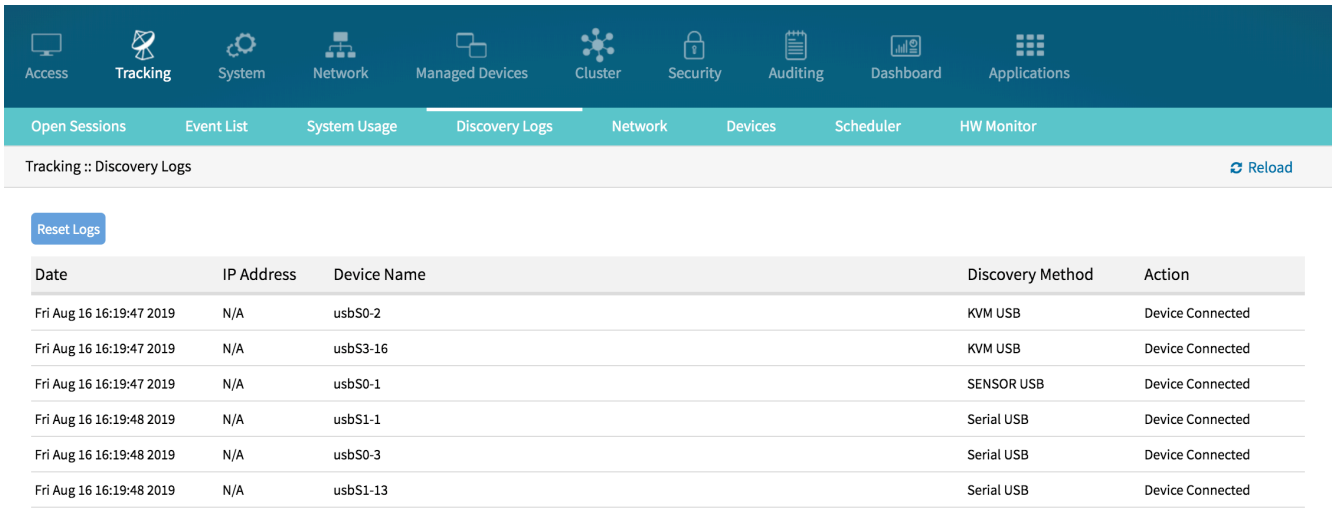
この`System Usage`ページには、現在のシステムの`Memory Usage`、`CPU Usage`、および`Disk usage`についての情報が表示されます。



Memory Type	Total (Kb)	Used (Kb)	Free (Kb)
Mem	8048328	1959868	6088460
Swap	1048572	0	1048572

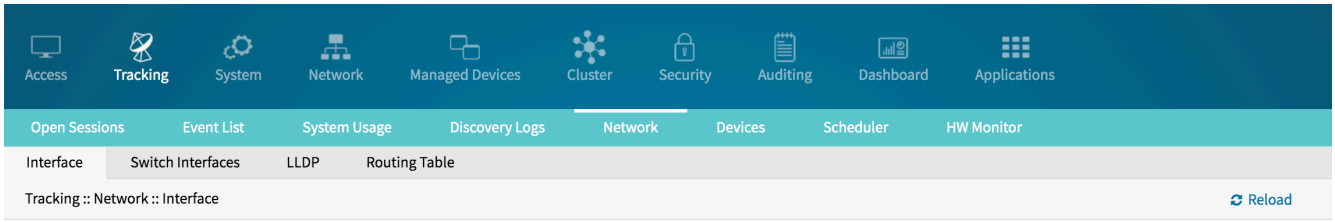
検出ログ

この`Discovery Logs`ページには、自動検出用の管理対象デバイス' で設定された検出プロセスのログが表示されます。



Date	IP Address	Device Name	Discovery Method	Action
Fri Aug 16 16:19:47 2019	N/A	usbS0-2	KVM USB	Device Connected
Fri Aug 16 16:19:47 2019	N/A	usbS3-16	KVM USB	Device Connected
Fri Aug 16 16:19:47 2019	N/A	usbS0-1	SENSOR USB	Device Connected
Fri Aug 16 16:19:48 2019	N/A	usbS1-1	Serial USB	Device Connected
Fri Aug 16 16:19:48 2019	N/A	usbS0-3	Serial USB	Device Connected
Fri Aug 16 16:19:48 2019	N/A	usbS1-13	Serial USB	Device Connected

ネットワーク統計



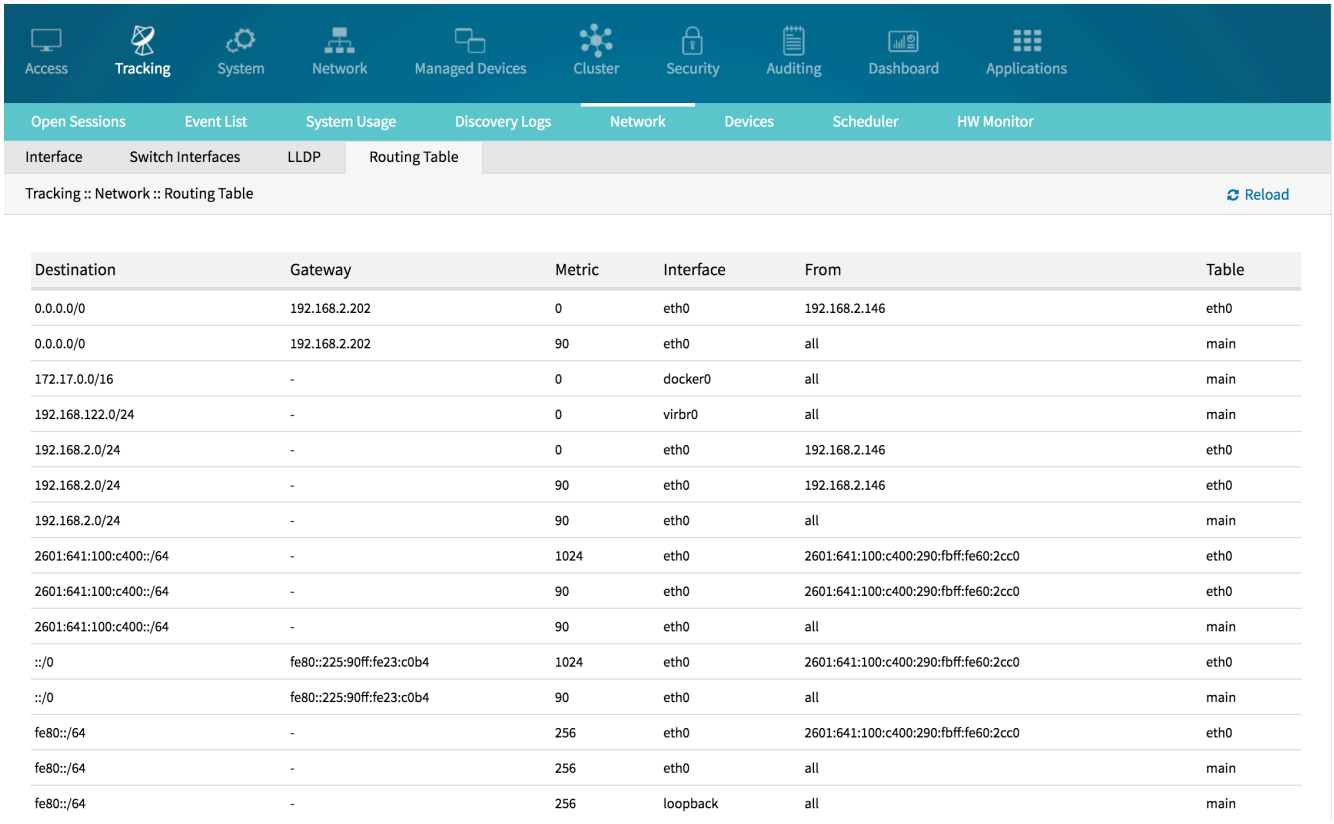
IfName	IfIndex	State	Rx Packets	Tx Packets	Collisions	Dropped	Errors
backplane0	5	Up	0	15291	0	0	0
backplane1	6	Up	0	15333	0	0	0
docker0	12	Down	0	0	0	0	0
eth0	7	Up	1073398	175196	0	0	0
eth1	8	Down	0	0	0	0	0
loopback	3	Up	0	32	0	0	0
loopback0	4	Up	0	32	0	0	0
tap0	9	Down	0	0	0	4208	0
virbr0	13	Down	0	0	0	0	0
virbr0-nic	14	Down	0	0	0	0	0

`Network`統計ページには、 ネットワーク`Interface`情報、 `LLDP`および`Routing Table`の詳細が表示されます。

この`Interface`ページには、状態、パッケージカウンタ、衝突、ドロップ、エラーなどのネットワークインターフェースに関する統計情報が表示されます。

この`LLDP`ページには、LAN 上でその ID と機能を宣伝しているデバイスが表示されます。ネットワーク接続で設定することで、Nodegrid で`LLDP advertising and reception through this connection`を有効にすることができます。

この`Routing Table`ページには、Nodegrid がネットワーク通信のために従うルーティングルールが表示されます。また、追加された任意の静的ネットワークルートも含まれています。

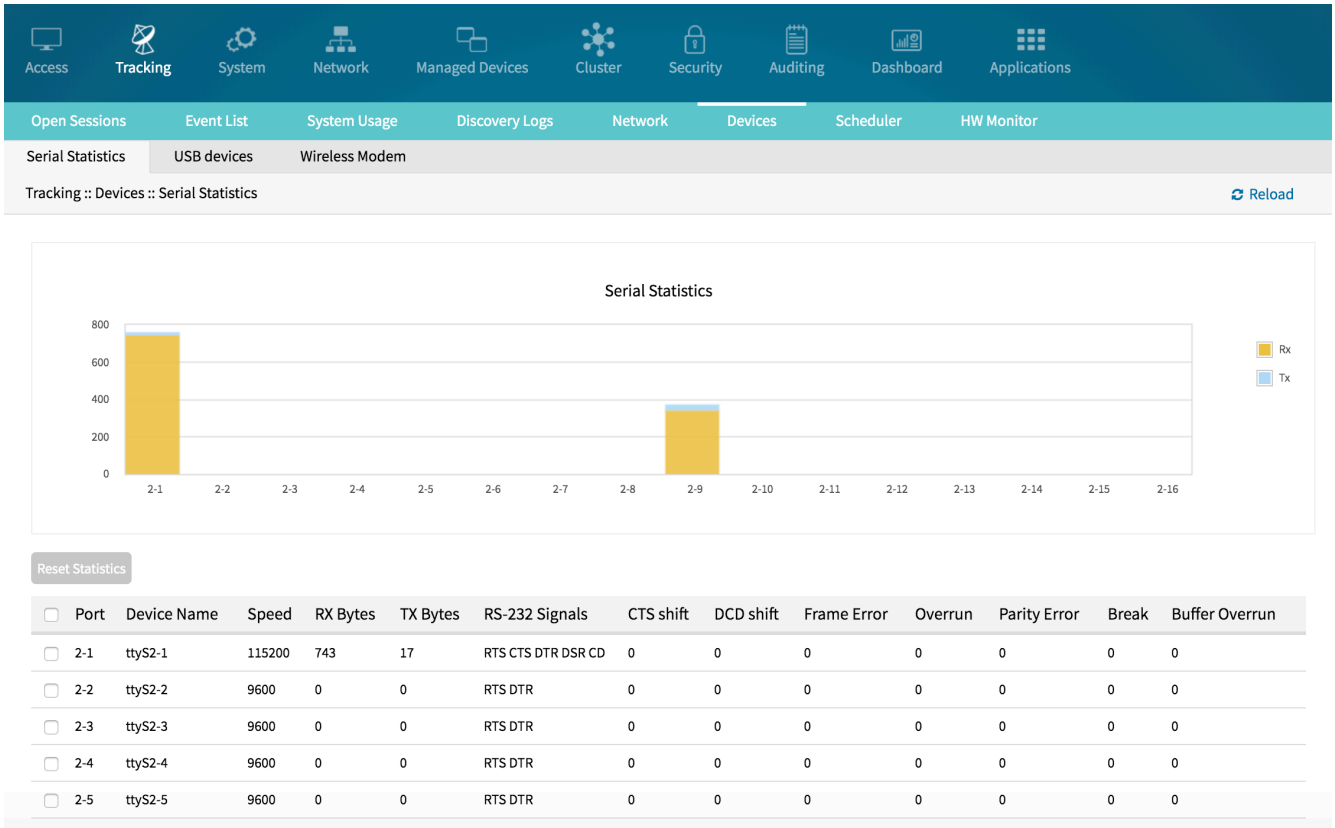


Destination	Gateway	Metric	Interface	From	Table
0.0.0.0/0	192.168.2.202	0	eth0	192.168.2.146	eth0
0.0.0.0/0	192.168.2.202	90	eth0	all	main
172.17.0.0/16	-	0	docker0	all	main
192.168.122.0/24	-	0	virbr0	all	main
192.168.2.0/24	-	0	eth0	192.168.2.146	eth0
192.168.2.0/24	-	90	eth0	192.168.2.146	eth0
192.168.2.0/24	-	90	eth0	all	main
2601:641:100:c400::/64	-	1024	eth0	2601:641:100:c400:290:fbff:fe60:2cc0	eth0
2601:641:100:c400::/64	-	90	eth0	2601:641:100:c400:290:fbff:fe60:2cc0	eth0
2601:641:100:c400::/64	-	90	eth0	all	main
::/0	fe80::225:90ff:fe23:c0b4	1024	eth0	2601:641:100:c400:290:fbff:fe60:2cc0	eth0
::/0	fe80::225:90ff:fe23:c0b4	90	eth0	all	main
fe80::/64	-	256	eth0	2601:641:100:c400:290:fbff:fe60:2cc0	eth0
fe80::/64	-	256	eth0	all	main
fe80::/64	-	256	loopback	all	main

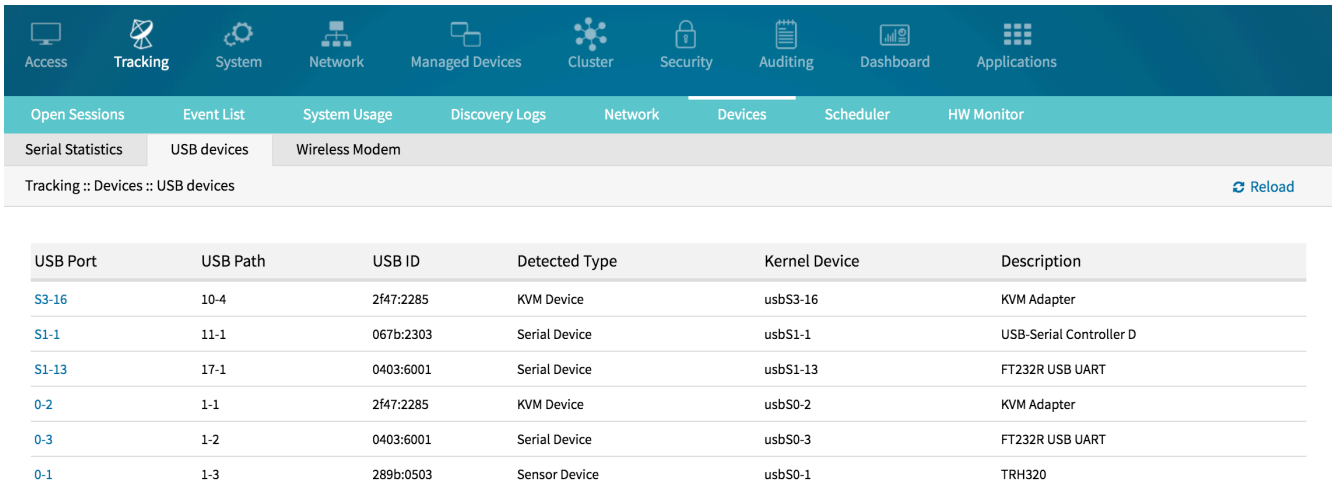
デバイス統計

この`Devices`ページには、シリアルデバイスや USB デバイス、ワイヤレスモデムなど、物理的に接続されたデバイスの接続統計情報が表示されます。使用可能なオプションは、各 Nodegrid ユニットによって異なります。

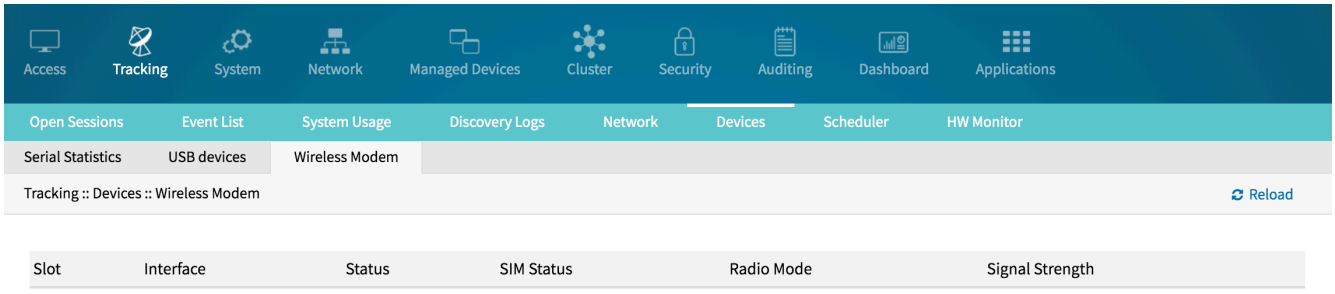
この`Serial Statistics`ページには、送受信データ、RS232 信号、エラーなど、シリアルポートの接続性に関する統計情報が表示されます。



この`USB devices`ページには、接続された USB デバイスと初期化されたドライバに関する詳細が表示されます。



この`Wireless Modem`ページには、スロット、SIM ステータス、および信号強度に関する情報が表示されます。



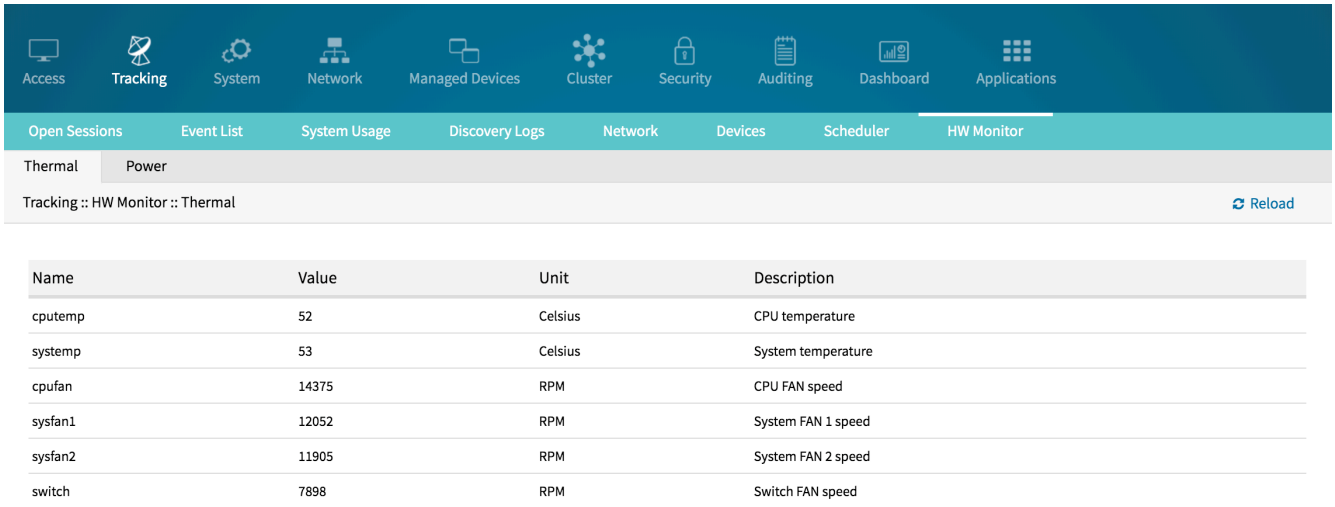
スケジューラ

この`Scheduler`ページには、スケジュールされたタスクの実行日時、実行者、イベントやエラーについての情報がが表示されます。

HWモニター

この`HW Monitor`ページには、Nodegrid システム情報が表示されます。 `Thermal` 現在の CPU 温度、システム温度、およびファン速度 (使用可能な場合) を表示します。 `Power` セクションは、電流の状態や電力消費量など、現在の電源についての情報が表示されます。この`I/O Ports`セクションは、Nodegrid Gate SR や Nodegrid Link SR など、GPIO ポートを備えたデバイスでのみ使用できます。GPIO ポートの現在のステータスが表示されます。

(NSRの例)

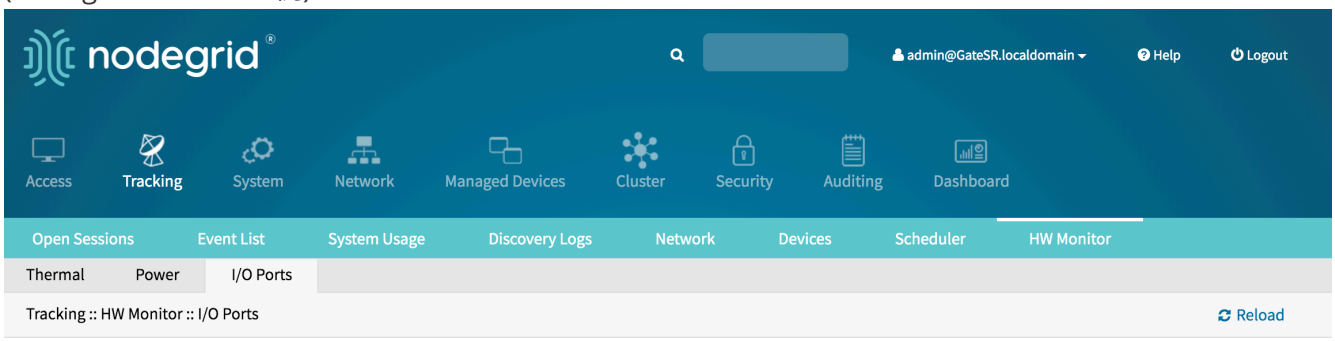


Name	Value	Unit	Description
cputemp	52	Celsius	CPU temperature
systemp	53	Celsius	System temperature
cpufan	14375	RPM	CPU FAN speed
sysfan1	12052	RPM	System FAN 1 speed
sysfan2	11905	RPM	System FAN 2 speed
switch	7898	RPM	Switch FAN speed

I/O ポート (GPIO)

このページには、GPIO ポートのステータスが表示されます。Nodegrid Gate SR や Nodegrid Link SR など、GPIO ポートを備えたモデルでのみ使用できます。

(Nodegrid Gate SR の例)



Name	Value	Direction	Description
OUT0	Low	Output	lab's door
Relay	Open	Output	my relay
DIO0	Low	Input	dio0Input My test with space $#@
DIO1	Low	Output	dio1output high

システム

システム設定では、ライセンスキー、一般的なシステム設定、ファームウェアの更新、バックアップと復元など、システム固有の設定を行うことができます。

ライセンス

[システム] をクリックすると、[ライセンス] タブに移動します。このタブには、この Nodegrid に登録されているすべてのライセンスと、その他の関連情報、ライセンス キー、有効期限、アプリケーションなどが表示されます。右上隅には、使用済みおよび使用可能なライセンスの数が表示されます。このページで、ライセンスの追加または削除が可能です。ライセンスの有効期限が切れるか削除されると、ライセンスの合計を超えるデバイスのステータスは [ライセンスなし] に変更されますが、その情報はシステムに保持されます。ただし、ライセンスのないデバイスはアクセスページに表示されません。

Nodegrid へのアクセスおよび制御には、管理対象デバイスごとにライセンスが必要です。Nodegrid の各シリアルポートに必要なライセンスは、製品に含まれています。

管理対象デバイスは、アクセスと制御のために Nodegrid の下に定義された任意の物理デバイスまたは仮想デバイスです。

システム設定

メインシステム設定はこのタブから行います。

- アドレスの位置と座標
- オンラインヘルプ URL
- セッション アイドル タイムアウト
- 適用されるリビジョンタグと最新プロファイル
- ログインロゴ画像とバナーメッセージ
- シリアルポートとライセンスの使用率
- Nodegrid Serial Console速度、ボードレート
- デュアル電源の表示
- ネットワーク起動パラメータと ISO 画像 URL

Nodegrid の位置

この Nodegrid の有効なアドレスの位置を入力し、このフィールドの右側にある小さなコンパスアイコン/ボタンをクリックして、そのアドレスの緯度と経度を下の [座標] フィールドに入力します。

[ヘルプの位置] フィールドは、ユーザーマニュアルの代替 URL の位置です。管理者は、ユーザーマニュアルをダウンロードして、Nodegrid によってアクセス可能な特定の位置に投稿できます。Nodegrid WebUI の右上にある小さい [ヘルプ] アイコン/ボタンをクリックすると、この URL で参照されているファイルで新しい Web ページが開きます。

セッション アイドル タイムアウト

これは、開いているセッションが非アクティブ化によってタイムアウトするまでの秒数です。 新しいセッションが期限切れにならないようにする場合は、ゼロ値を入力します。このフィールドの設定変更は、新しいセッションにのみ有効になります。既存のセッションは、セッション開始時に指定されたそれぞれのタイムアウト値に従って続行します。この設定は、すべての telnet、SSH、HTTP、HTTPS、およびコンソールセッションに適用されます。

Nodegrid の設定

この`Revision Tag`フィールドでは、設定参照タグとして使用される自由形式の文字列を定義できます。このフィールドは、手動または自動化された変更管理プロセスを介して更新することができます。

`Latest Profile Applied`は、ZTP プロセスまたは ZPE クラウドを通じて、最後に適用されたプロファイルを通知します。

ログインロゴ画像

この機能を使用して、Nodegrid の WebUI ログイン ページで使用するロゴ画像を変更します。新しい画像ファイルの形式は .png または .jpg で、ローカルデスクトップまたはリモートサーバ (FTP、TFTP、SFTP、SCP、HTTP、および HTTPS) からアップロードできます。それぞれの URL 形式、ユーザー名、パスワードの入力が必要となります。`<PROTOCOL>://<ServerAddress>/<Remote File>`.

アップロード後、ブラウザのキャッシュを更新して新しい画像を表示します。

ログインバナー

Nodegrid は、Telnet、SSHv2、HTTP、HTTPS、およびコンソールログインにログインバナーが表示されるように設定し、システムにログインする前にユーザーにメッセージを表示することが可能です。管理者は、デフォルトのバナー (以下) の編集とカスタマイズが可能です。

デフォルトのログインバナー:

```
WARNING: This private system is provided for authorized use only and it may be
monitored for all lawful purposes to ensure its use. All information including
personal information, placed on or sent over this system may be
monitored and recorded. Use of this system, authorized or unauthorized, constitutes
consent to monitoring your session. Unauthorized use may subject you to criminal
prosecution. Evidence of any such unauthorized use may be used for administrative,
criminal and/or legal actions.
```

使用率

各ボックスをクリックしてチェックを入れ、望ましい使用率を入力して、ライセンスとローカルシリアルポートの使用率を監視可能にします。指定したパーセンテージに達すると、イベントが生成されます。デフォルト値は 90% です。

コンソールポート

ローカルコンソールポートのボーレートを設定します。デフォルト値は、115.200 bps に設定されています。

電源装置

デュアル電源装置 (オン/オフ) の状態を表示し、片方の電源が落ちた時に (適切なチェックボックスをオンにして) アラーム音を発するようにします。

アラームの状態を確認するには、このページ`System::Preferences`の左上の`Acknowledge Alarm State`をクリックします。

ネットワークブート

Nodegrid は、ネットワークの ISO イメージから起動するように設定できます。ユニットの IPv4 アドレス、ネットマスク、使用するイーサネットインターフェース (eth0 または eth1)、および ISO イメージにアクセスできる URL を入力します。 `http://ServerIPAddress/PATH/FILENAME.ISO`

PXE ブート

Nodegrid は PXE ブート (プリブート実行環境) をサポートしています。PXE は UEFI (統合拡張可能なファームウェアインターフェース) の一部を形成し、ネットワークサーバから起動時に取得した適切なソフトウェアイメージを起動します。これは、データセンターで最も推奨される、OS の起動、インストール、およびデプロイに最適な方法のひとつです。

Nodegrid において、PXE ブートはデフォルトで有効になっていますが、`Security::Services`下の Web ページを介して、またはスコープ`/settings/services`内の CLI を介して無効にすることができます。 以下の例は、インストールされている Apache web サーバ、tftpd-hpa サービス、および Nodegrid 4.1.x を使用して Linux (Ubuntu) で DHCP/PXE サーバを設定する方法を示しています。PXE、DHCP、および TFTP サーバがインストールされている必要があります。

- Nodegrid ネットワークブートファイル (tarball) をダウンロードします - サポートに連絡してファイルを入手してください
- Nodegrid ネットワークブート tar.gz (tarball) ファイルを DHCP サーバにコピーし、2 つのディレクトリ (nodegrid 4.1.xx と boot) を作成する tar ファイルを解凍するか、ディレクトリを作成して tar ファイルをそのディレクトリに保存してから tarball ファイルを解凍します (つまり、cd /var/lib/tftpboot/PXE ディレクトリ)

```
Example:
root@ubuntu-srv1:~# cd /var/lib/tftpboot/
root@ubuntu-srv1:/var/lib/tftpboot# ls -l
drwxrwxr-x 2 root  root       4096 Apr 24 03:20 nodegrid-4.1.xx
root@ubuntu-srv1:/var/lib/tftpboot# ls -l nodegrid-4.1.xx
total 558468
-rw-r--r-- 1 root root  22270823 Apr 24 03:19 initrd
-rw-rw-r-- 1 root root 544343672 Apr 24 03:19 rootfs.img.gz
-rw-rw-r-- 1 root root         7 Apr 24 03:19 version
-rw-r--r-- 1 root root   5242832 Apr 24 03:19 vmlinuz
      root@ubuntu-srv1:/var/lib/tftpboot#
```

- dhcpd.conf ファイルを編集し、ホスト定義セクションにこれらの行を追加します。`fixed-address` 値は、Nodegrid ユニットの MAC アドレスと一致する必要があり、`hardware ethernet` は、Nodegrid ユニットの IP です。

```
host PXEboot_NSC {
   hardware ethernet e4:1a:2c:56:02:9e;
   fixed-address 192.168.22.61;
   option tftp-server-name "192.168.22.201";
   next-server 192.168.22.201;
option bootfile-name "PXE/boot/grub/i386-pc/core.0";
 # option bootfile-name "nodegrid-4.1.xx/boot/grub/i386-pc/core.0";
  option domain-name "zpesystems.com";
  option domain-name-servers 192.168.22.205, 75.75.75.75, 75.75.76.76;
   option routers 192.168.22.202;
}
```

- Web サーバ (Apache など) で、cd/var/www を使用して、ネットワークブートを実行するファイルへのソフトリンクを作成します: ln -s <ファイル名のディレクトリ>およびディレクトリにリンクするファイル名。

```
  Example:
root@ubuntu-srv1:/var/www# pwd
root@ubuntu-srv1:/var/www#
root@ubuntu-srv1:/var/www# ln -sf /var/lib/tftpboot/PXE/nodegrid-4.1.xx/ nodegrid-
4.1.xx
```

- DHCP サーバを再起動します

```
sudo service isc-dhcp-server restart
```

- tftpd-hpa プロセスを再起動します
- Nodegrid を開始します。これにより、Nodegrid のネットブートイメージが特定の Nodegrid にイン

ストールされます。

日時

正確な時刻を自動で取得するためにネットワークタイムプロトコル (NTP) サーバを設定するか、日付と時刻を手動設定します。NTP はデフォルト設定であり、NTP プール内の任意のサーバから日付と時刻を取得します。手動設定モードでは、Nodegrid は独自のクロックを使用して日付と時刻を提供します。ページを更新すると、現在のシステム時刻を確認できます。

Nodegrid は、`NTP Authentication`と`Cellular Tower Synchronization`をサポートしています。

NTP 認証は、受け取ったタイムスタンプが信頼できるソースによって生成されたものであることを確認し、不正なアクティビティや傍受から保護して Nodegrid の安全性を向上させます。

セルタワーからの日付と時刻の同期は、キャリアネットワークから正確な時刻を直接取得できるため、非常に便利です。

ローカルタイムゾーンは、ドロップダウンメニューからも設定できますが、デフォルト値は UTC です。

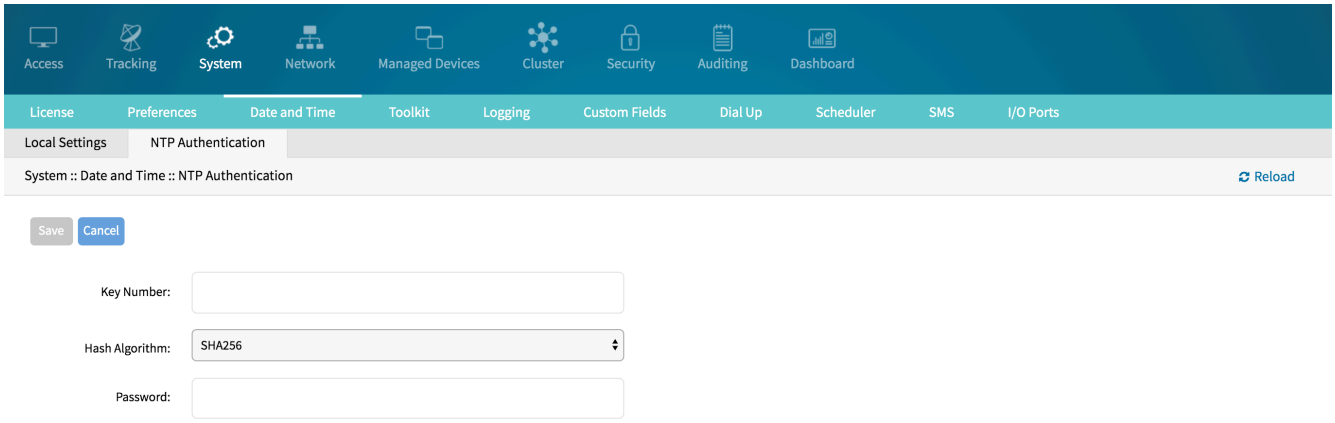
> 注: イベントログのすべてのタイムスタンプは、UTC で表示されます。

NTP 認証

NTP では、時間の同期に関連する安全性へのリスクを軽減するための多くの対策が講じられています。認証は、そのうちのひとつです。これによりクライアントは、応答が不正に生成または傍受されたのものではなく、予期していたソースから生成されたものであることを確認できます。 認証は、サーバとクライアント間で合意されたキーまたはパスワードのリストに基づいています。サーバとクライアント間の任意の通信には、メッセージに合意済みのキーの暗号化されたバージョンが追加されます。サーバ/クライアントは、受信した任意の通信に追加されたキーの暗号を解除して、必要なアクションを実行する前に、それが合意済みのキーのいずれかに一致することを確認できます。

WebUI での管理者として、次の情報を提供するために`System :: Date and Time :: NTP Authentication`に移動します:

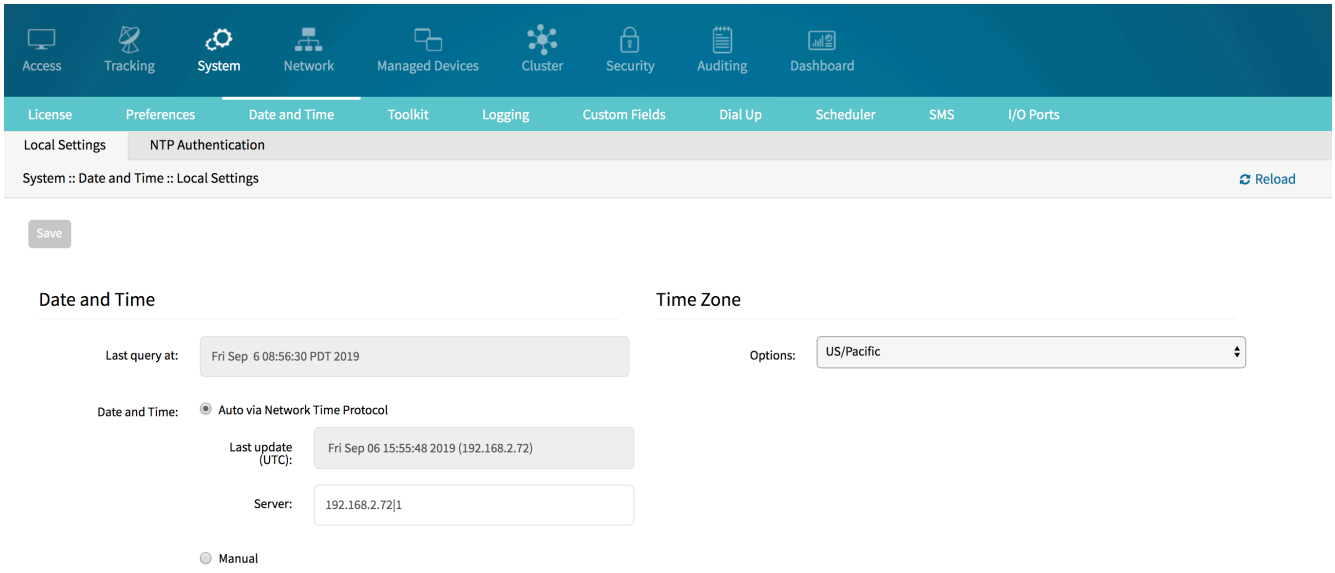
- キー番号/キー ID: キー/パスワードを識別する番号
- ハッシュアルゴリズム/タイプ: 使用される暗号化ハッシュ関数
- パスワード/キー値: パスワードは、ハッシュアルゴリズムと共に使用され、NTP パケット内のメッセージ認証コード (MAC) を生成および検証します。



各キー番号を、システム > 日付と時刻 > NTP 認証で入力します

- キー番号: 1 から 2^32-1 の範囲内の任意の符号なし整数
- ハッシュアルゴリズム: 'MD5'、'RMD160'、'SHA1'、'SHA256'、'SHA384'、'SHA512'、 'SHA3-224','SHA3-256','SHA3-384','SHA3-512'
- パスワード: パスワードは、空白を含まない文字列として、または 'HEX:' のプレフィックス付きの16進数として指定することもできます。

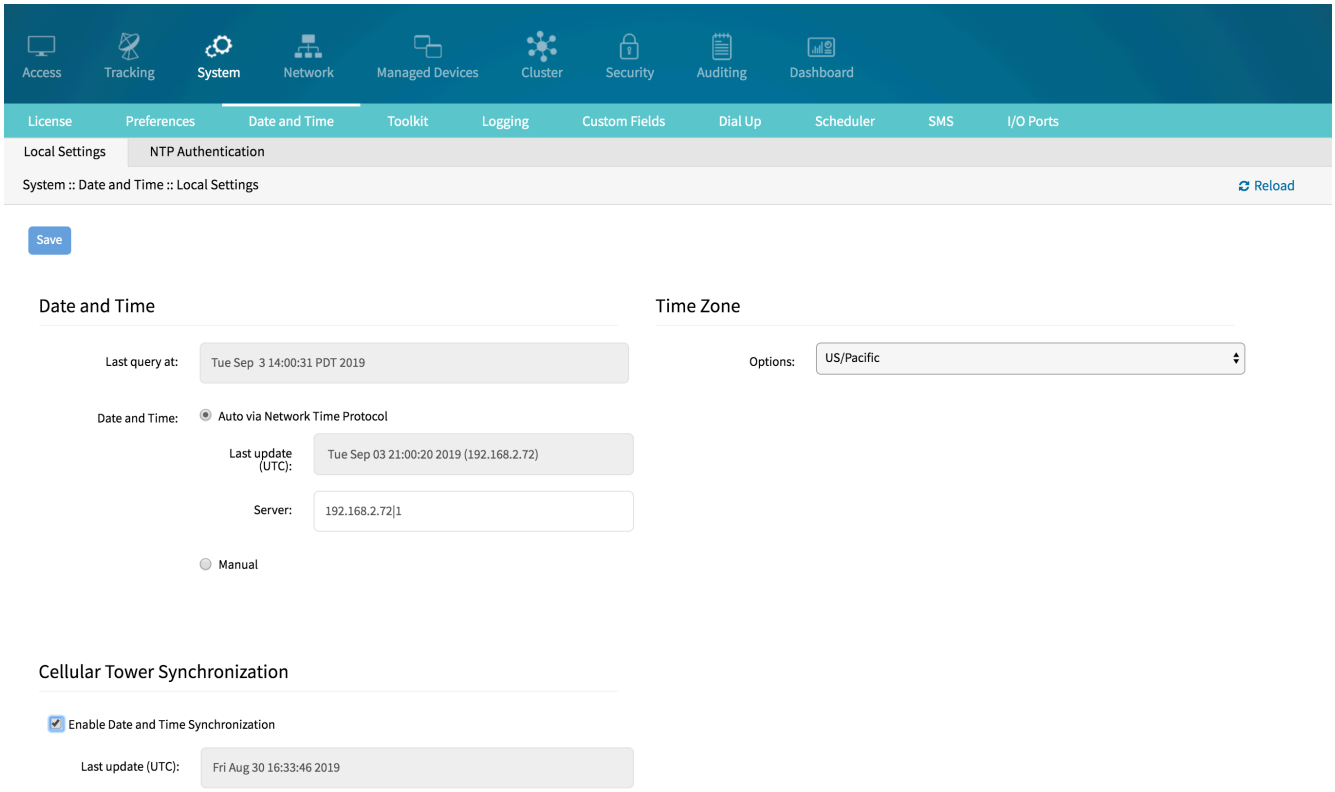
NTP サーバとそのキー番号をリンクします



リンクは、`System :: Date and Time :: Local Settings`で、またはシステム > 日付と時刻 > ローカル設定で提供されます。 サーバアドレスとそのキー番号の間に、|区切り記号 '|' (パイプ) を使用します。

携帯電話基地局との同期

この機能は、有効な SIM カードがインストールされているワイヤレスモデムカードを備えたユニットでサポートされています。SIM カードがキャリアネットワークに登録されている場合、Nodegrid は携帯電話基地局から日付と時刻を取得できます。 この機能を有効にするには、適切なチェックボックスをオンにします。



NTP と Cellular Tower の両方を同期できます。いずれかのソースから受信した最後の日時が適用されます。このアプローチにより、接続フェールオーバー設定の日付/時刻情報も受け取れます。

ロギング

システムロギング機能を使用すると、Nodegrid へのすべての CLI セッションのデータロギングが可能になり、後の検査や監査のために記録されます。

収集されたデータログは、`Auditing`設定に応じて Nodegrid のローカルに保存されるか、リモートで保存されます。

データロギング機能では、情報収集に加えて、イベント通知を作成できます。これは、アラート文字列を定義することでアーカイブされます。アラート文字列は、シンプルなテキストの一致、またはデータ収集時にデータソースストリームに対して評価される正規表現パターン文字列にすることができます。イベントは一致するたびに生成されます。

カスタムフィールド

このセクションでは、検索可能なカスタムフィールドをユニットに追加します。

この機能を使用して、デフォルトで使用できない情報を追加します。Nodegrid システムでは、カスタムフィールドを作成して、デバイスの情報の一部にすることができます。

ダイヤルアップ

デバイスやコールバックユーザーにダイヤルするためのパラメータが、ここで設定されます。ログインおよび PPP 接続機能もドロップダウンメニューから定義されます。

スケジューラ

管理者は、スケジューラでタスクやスクリプトを計画に沿って実行できます。これは、メンテナンスタスクやエンドデバイスを含む自動化タスクに使用できます。

実行する必要があるタスクは、Nodegrid にある`cli`ファイル、またはスクリプトファイルの一部である必要があります。ユーザーがアクセス可能で実行可能であるファイルである必要があります。

設定	値	説明
タスク名	文字列	タスク名
タスクの説明	文字列	タスクの説明が概要に表示されます
ユーザー	文字列	スクリプトファイルにアクセスできる、有効な **ローカル**ユーザーである必要があります。デフォルト: daemon
実行するコマンド	文字列	実行される Shell コマンド。cli ファイルを実行するために、次の設定を使用できます。 `cli -f <path><cli file name>`
分	整数	タスクを実行する場合の分 (0 から 59)。デフォルト: *(任意)
時間	整数	タスクを実行する場合の時間 (0 から 23)。デフォルト: *(任意)
月の日付	整数	タスクを実行する場合の日 (1 から 31)。デフォルト: *(任意)
月	整数	タスクを実行する月 (1 から 12)。デフォルト: *(任意)
曜日	整数	タスクを実行する曜日 (0 から 6、日曜日から土曜日)。デフォルト: *(任意)

> 注: cli ファイルは、Nodegrid cli コマンドのみを含むテキストファイルです。

スケジューラの日付と時刻の例:

タスクを毎日 (00:01) に実行

分	**1**
時間	0
月の日付	*
月	*
曜日	*

毎週土曜日の 23:45 にタスクを実行

分	**45**
時間	23
月の日付	*
月	*
曜日	6

1 時間ごとに実行

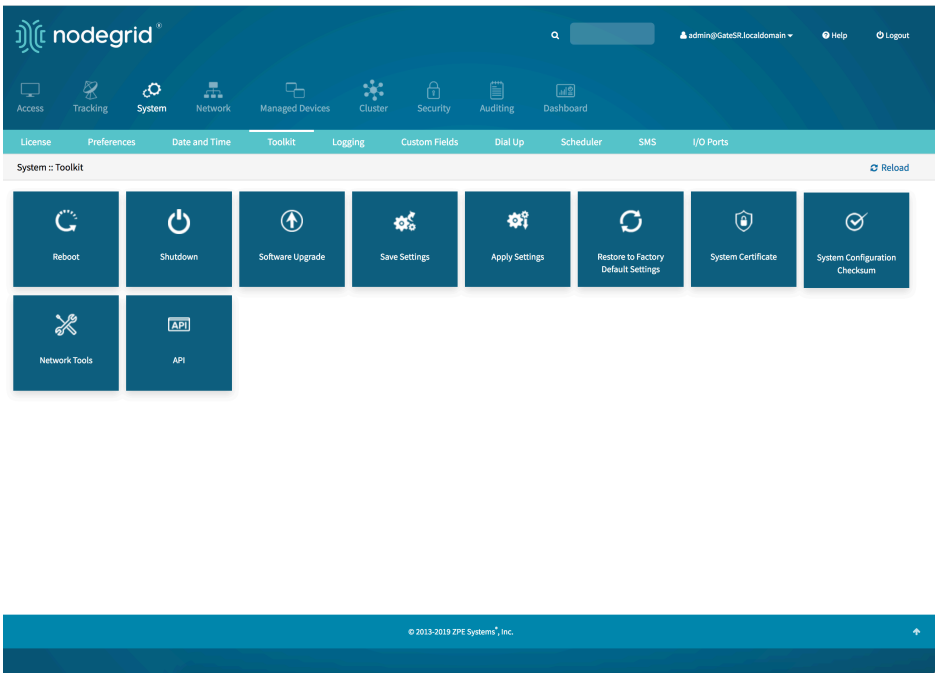
分	**0**
時間	*
月の日付	*
月	*
曜日	*

システムメンテナンス

システムメンテナンス機能は、`System::Toolkit`ページで使用できます。このツールキットは、次の操作を実行するために使用されます。

- 再起動
- シャットダウン
- ソフトウェアのアップグレード
- 設定を保存します
- 設定を適用する

- 工場出荷時の設定に戻す
- システム証明書
- システム コンフィギュレーション チェックサム
- ネットワークツール
- API



再起動とシャットダウン

再起動コマンドとシャットダウンコマンドは、Nodegrid の正常なシャットダウンと再起動を可能にし、システムはすべてのアクティブなセッションが切断されることを伝える警告メッセージを表示します。

ユニットの再起動中に、オペレーティングシステムが自動的に再起動します。シャットダウンすると、オペレーティングシステムは停止状態になります。この時点で、電源装置をオフにするか、ユニットから電源コードを取り外して、ユニットへの電源を切断するのが安全です。ユニットの電源をオンに戻すには、停止してから復元する必要があります。

ソフトウェアのアップグレード

ソフトウェアのアップグレードには 3 つの方法があり、デバイス自体から、デバイスに接続されているコンピュータから、またはリモートサーバから行えます。新しいソフトウェアの ISO 画像は、これらの特定の場所に事前に読み込まれている必要があります。

- Nodegrid デバイス自体からアップグレードするには、新しいソフトウェアの ISO ファイルを`/var/sw`に配置します。
- Nodegrid に接続されているお使いのローカルコンピュータからアップグレードするには、`Local Computer`ラジオボタンをクリックし、アップグレードに使用するファイルを選択します。

- リモートサーバからアップグレードするには、`Remote Server`ラジオボタンをクリックして、サーバの URL、求められるユーザー名とパスワードを入力します。FTP、TFTP、SFTP、SCP、HTTP、および HTTPS プロトコルがサポートされています。サーバアドレスには、IP アドレスまたはホスト名/FQDN を指定できます。IPv6 を使用する場合は、かっこ [ ] を使用します。

  例:

  `ftp://192.168.22.21/downloads/Nodegrid_v4.1.0_20191225.iso`

ダウングレードの場合は、工場出荷時のデフォルト設定を適用するか、保存されている設定を復元するかを選択できます。

設定を保存します

現在の設定は、Nodegrid 自体、デバイスに接続されているローカルコンピュータ、またはリモートサーバに保存できます。設定に任意の (意味のある) 名前を付けると、"/backup" ディレクトリに保存されます。

サーバアドレスには、IP アドレスまたはホスト名/FQDN を指定できます。IPv6 を使用する場合は、かっこ [...] を使用します。

FTP、TFTP、SFTP、および SCP プロトコルがサポートされています。

設定を適用する

保存された設定は、Nodegrid 自体から、デバイスに接続されたローカルコンピュータから、またはリモートサーバから読み込まれ、ユニットの新しい設定となる Nodegrid に適用できます。 サーバアドレスには、IP アドレスまたはホスト名/FQDN を指定できます。IPv6 を使用する場合は、かっこ [...] を使用します。

FTP、TFTP、SFTP、SCP、HTTP および HTTPS プロトコルがサポートされています。

工場出荷時の設定に戻す

このオプションは、すべての設定を工場出荷時のデフォルト値に復元するために使用されます。すべてのログファイルを消去するかどうかの選択が可能です。

システム コンフィギュレーション チェックサム

この機能を使用して、現在の特定の設定のチェックサムベースラインを作成します。これは管理者にとって、設定が変更されたかどうかを定期的に確認するためのクイックツールとなります。クリックして、実行中の設定を保存されたベースラインと比較します。すべての設定が一致した場合 (すべてOK) の主な結果は [合格] になり、不一致が検出された場合は [失敗] で、その場所を指定します。

MD5 と SHA256 は現在サポートされています。

システム証明書

証明書は、Nodegrid に接続されたローカルコンピュータ、またはリモートサーバから読み込むことができます。ローカルコンピュータから読み込む場合は、ファイルを選択します。それ以外の場合はリモートサーバの URL と、適切なユーザー名とパスワードを入力します。

証明書が適用されると、Web サーバが再起動され、アクティブなセッションが切断される可能性がありますのでご注意ください。

FTP、TFTP、SFTP、SCP、HTTP、および HTTPS のプロトコルがサポートされています。

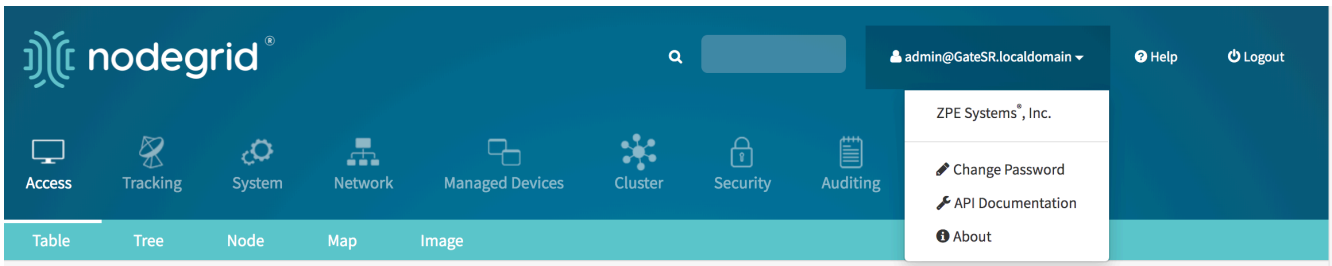
ネットワークツール

このページでは、コマンドラインを使用するのとまったく同じように、「ping」、「traceroute」、「DNS lookup] の重要なネットワークツールを提供します。コマンド出力は、ページの下部に表示されます。

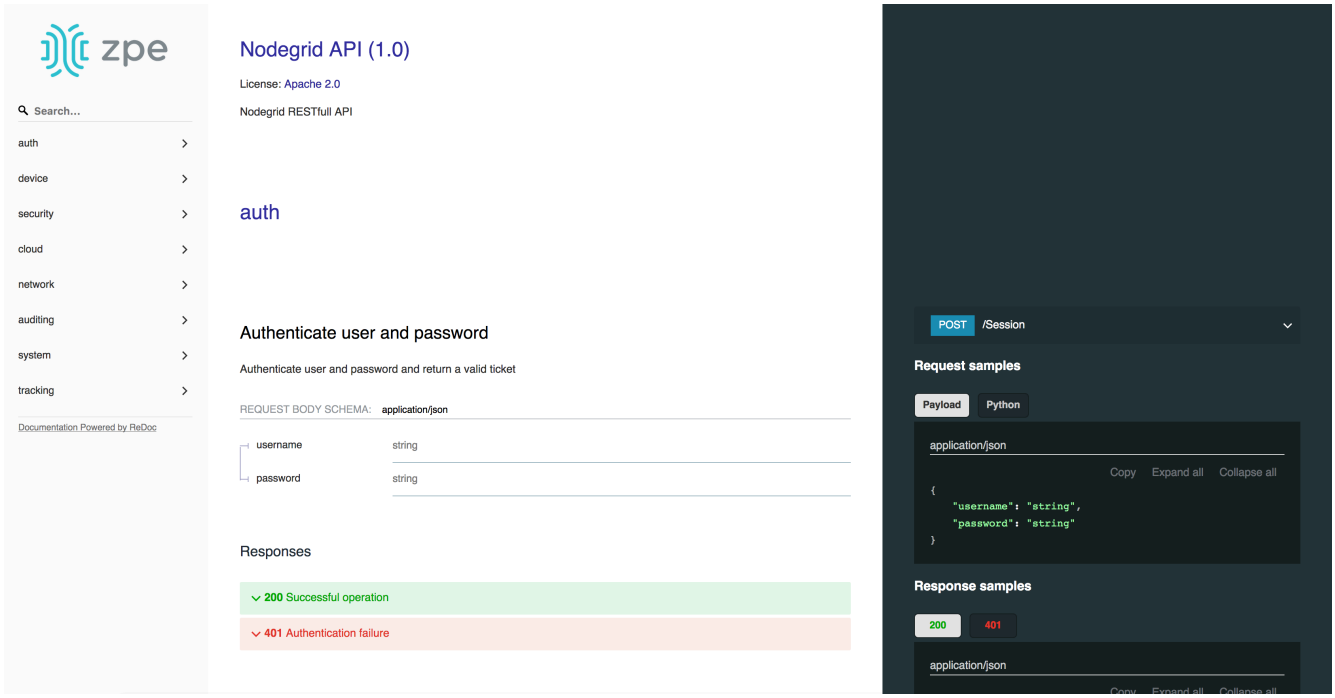
API

RESTful API

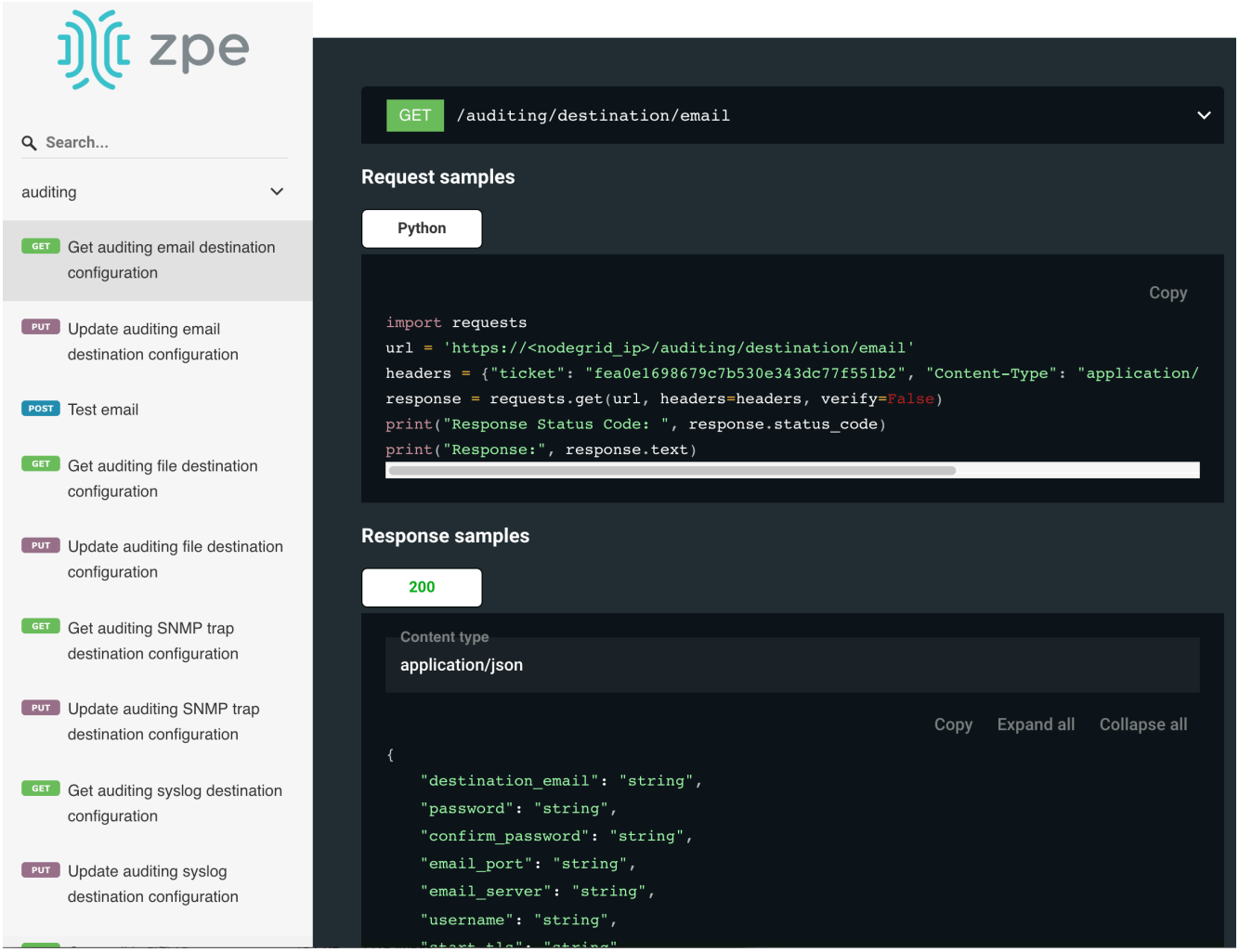
Nodegrid Platform は、読み取りと変更、および Nodegrid の設定に使用できる RESTful API を提供します。API ドキュメントは Nodegrid に埋め込まれており、システム > ツールキット > API またはメインの WEB ページの右上隅にあるプルダウン USER メニューから使用できます (プルダウンして [API ドキュメント] をクリックします)。



下にスクロールして、API 要求と応答の一例を参照してください。



例 「Eメールの送信先設定の監査を取得する」



> 注: API ドキュメントは、各 Nodegrid Platformで参照できます。 `https://<Nodegrid IP>/api_doc.html`

gRPC

Nodegrid Platformは gRPC フレームワークをサポートしています。このサービスはデフォルトで無効です。gRPC のサポートを有効にするには、[システム - セキュリティ] セクションを参照してください。

gRPC は、単純なサービス定義と構造化データを使用した、非常にスケーラブルでパフォーマンスベースの RPC フレームワークです。

Nodegrid は、次の 4 つのサービス定義を実装します:

- get_request (APIRequest) - データの読み取り、戻り値 (APIReply) を許可
- post_request (APIRequest) - コマンドの実行、エントリの追加、戻り値 (APIReply) を許可
- put_request (APIRequest) - エントリの選択/更新を必要とするコマンドの実行、戻り値 (APIReply) を許可
- delete_request (APIRequest) - 既存のデータセットの削除、セッションの破棄、戻り値 (APIReply) を許可

すべての **APIRequest** は、次の 3 つの引数を必要とします:

1. path - 使用される gRPC パス。
2. ticket - リクエストの認証チケット。
3. data - 構造化されたデータ、json 形式。

3 つの引数はすべて、既存の REST API と同じ構造に従う必要があります。詳細は`https://<Nodegrid IP>/api_doc.html`を参照してください。

すべての**APIReply**は、2 つの引数を返します:

1. message - json 形式で構造化されたデータ。
2. status_code - int32 ナンバーとしてのstatus_code。

基本的な例:

post_request - 認証 - セッションチケットを返します

```
post_request({path: '/v1/Session', data: '{"username": "admin", "password":
"admin"}'}, [...]
```

get_request - ネットワークの詳細を取得

```
get_request({path: '/v1/network/connections', ticket: 'xxxxxxxxxxxx'}, [...]
```

post_request - SMS ホワイトリストに電話番号を追加

```
post_request({path: '/v1/system/sms/whitelist', ticket: 'xxxxxxxxxxxx', data
'{"name": "phone1", "phone_number": "+111111111111"}' }, [...]
```

Putrequest_request - SMS ホワイトリストで既存の値を更新

```
put_request({path: '/v1/system/sms/whitelist/phone1', ticket: 'xxxxxxxxxxxx', data
'{"phone_number": "+122222222222"}' }, [...]
```

Deleterequest_request - SMS ホワイトリストで既存の値を削除

```
delete_request({path: '/v1/system/sms/whitelist', ticket: 'xxxxxxxxxxxx', data
'{"whitelists": [ "phone1", "phone2" ]}' }, [...]
```

SMS トリガのアクション

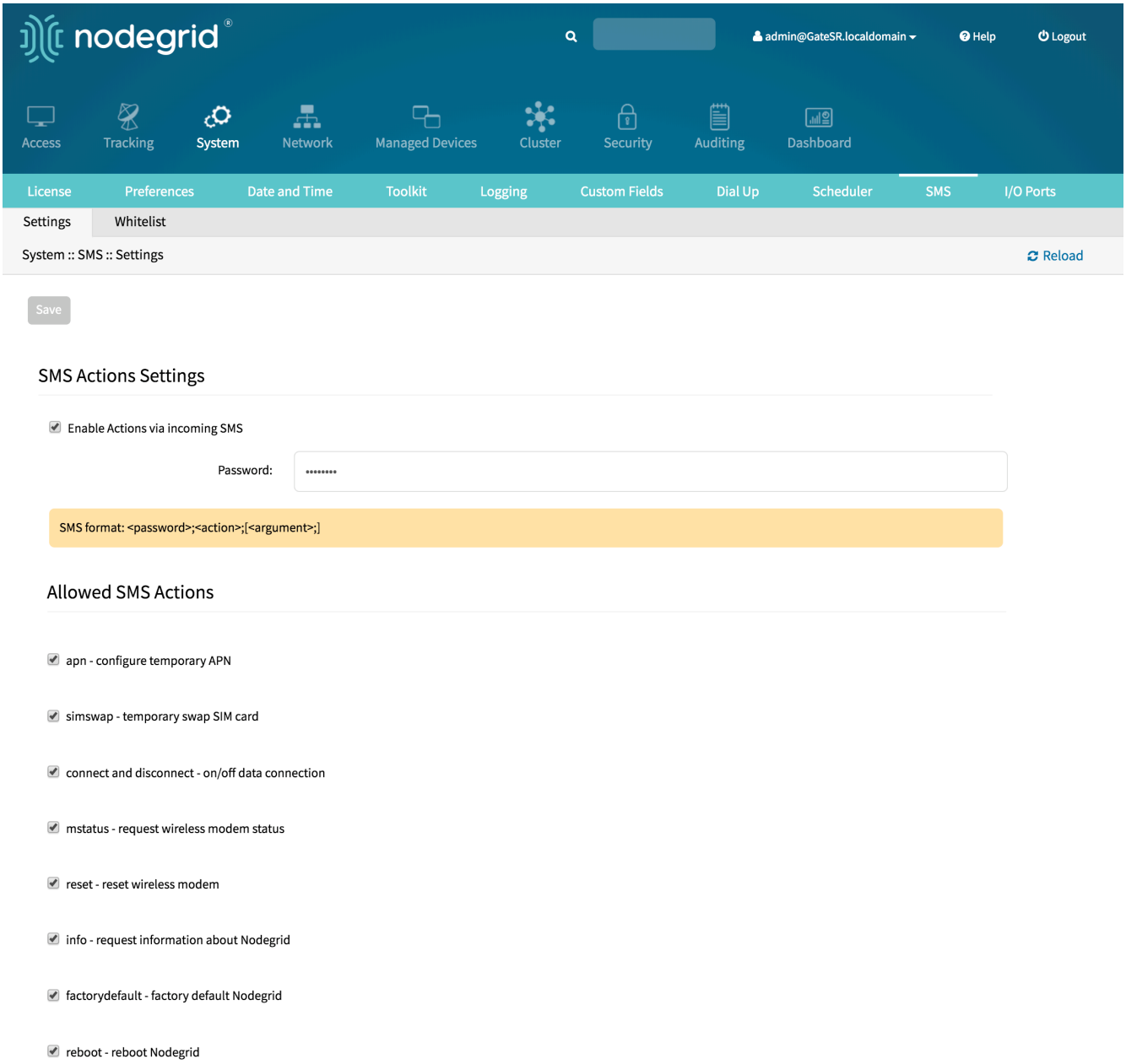
ユーザーは、SMS 着信メッセージを介して Nodegrid でリモートアクションを実行できます。SMS メッセージ認証は有効でなければならず、許可されたアクションのみを実行できます。この機能は、SMS メッセージの送受信が可能な携帯電話接続を必要とし、携帯電話モジュールがインストールされているユニットで使用できます。

この機能は、M2 カード EM7565 M2/ワイヤレス モデムを搭載した Nodegrid Services Router、Bold SR、Gate SR、Link SR などの SMS 対応モデルでサポートされています。 デフォルトで、`Disabled`は`Enable Actions via incoming SMS`です。これがデフォルトの状態で有効になっている場合 (パスワードなし)、Nodegrid はすべての電話番号から SMS によってトリガされたアクションを受け入れ、ETH0 の MAC アドレスをデフォルトのパスワードとして使用します。

> 注: SMS オプションは、SIM カードとプランが SMS 可能である必要があり、これはサービスプロバイダーに確認できます。必要に応じて複数の SMS で応答できるアクションもあるため、このサービスのコストを確認することをお勧めします。

SMS設定

設定	値	説明
入来SMSを通じてアクションを有効化する	文字列	デフォルトで無効
許可されたSMSアクション		SMS によってトリガされるアクション
apn - 一時的APNを設定する	TRUE/FALSE	一時的な APN を設定できます。
simswap - 一時的交換 SIM カード	True/False	SIM カードのフェールオーバーをトリガーします
接続および切断 - オン/オフ データ接続	True/False	モデムをトリガして接続または切断します
mstatus - ワイヤレスモデムのステータスを要求する	TRUE/FALSE	現在のモデムのステータスが返されます
リセット - ワイヤレスモデムをリセットします	TRUE/FALSE	モデムのリセットをトリガーします
情報 - Nodegrid についての情報を要求します	TRUE/FALSE	返された情報について
factorydefault - 工場出荷時設定の Nodegrid	TRUE/FALSE	Nodegrid アプライアンスの工場出荷時デフォルト値がトリガされます
再起動 - Nodegrid を再起動する	TRUE/FALSE	Nodegrid の再起動がトリガされます



SMS アクションとメッセージの例

SMS アクションとその後の応答の形式は、以下のリストに示されています。一部のアクションは、応答を必要としない場合があります。

```
Message format: < password >;< action >;< argument >;
   Response: <response>;
```

```
1. connect: try to power on data connection:
< password >;connect;
Connect action started;
```

```
2. disconnect: drop current data connection
< password >;disconnect;
Disconnect action started;
```

```
3. reset: reset wireless modem
< password >;reset;
Modem Reset will start soon;
```

```
4. apn: configure temporary APN
< password >;apn;<new apn>;
```

```
5. mstatus: request modem status
< password >;mstatus;
Service:< LTE|WCDMA >;RSSI:< value dbm >;SIM:< sim number in use >;State:< status
>;APN:< apn in use >;IP addr:< ip address when connected >
```

```
6. simswap:  swap sim card temporary
< password >;simswap;<timeout for sim to register in secs. max 180>;
Modem will reset to swap sim;
```

```
7. info: request Nodegrid information
< password >;info;
Model: < Nodegrid model >; Serial Number: < Nodegrid serial number >; Version: <
firmware version >;
```

```
8. reboot: reboot Nodegrid
< password >;reboot;
Nodegrid will reboot soon;
```

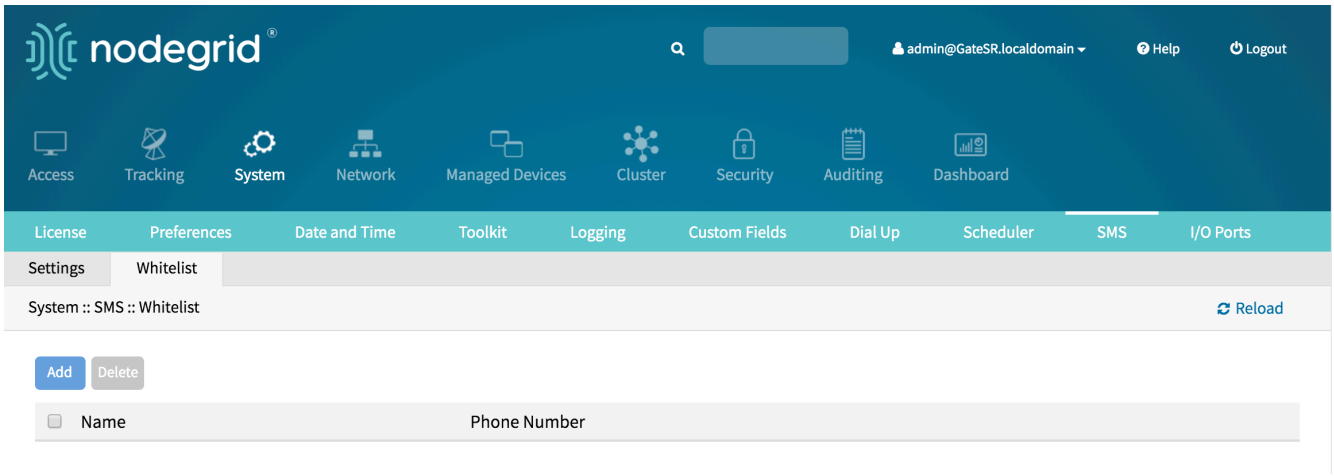
```
9. factorydefault: restore Nodegrid configuration to factory default
< password >;factorydefault;
Nodegrid will restore configuration to factory default and reboot;
```

SMS ホワイトリスト

SMS ホワイトリスト表で、管理者は SMS アクショントリガの送信が許可されている電話番号の追加、削除、変更が可能です。他のすべての電話番号からの要求は無視されます。ホワイトリストにエントリを追加するには、`Add`をクリックしてホワイトリストに登録するアイテムを入力します。

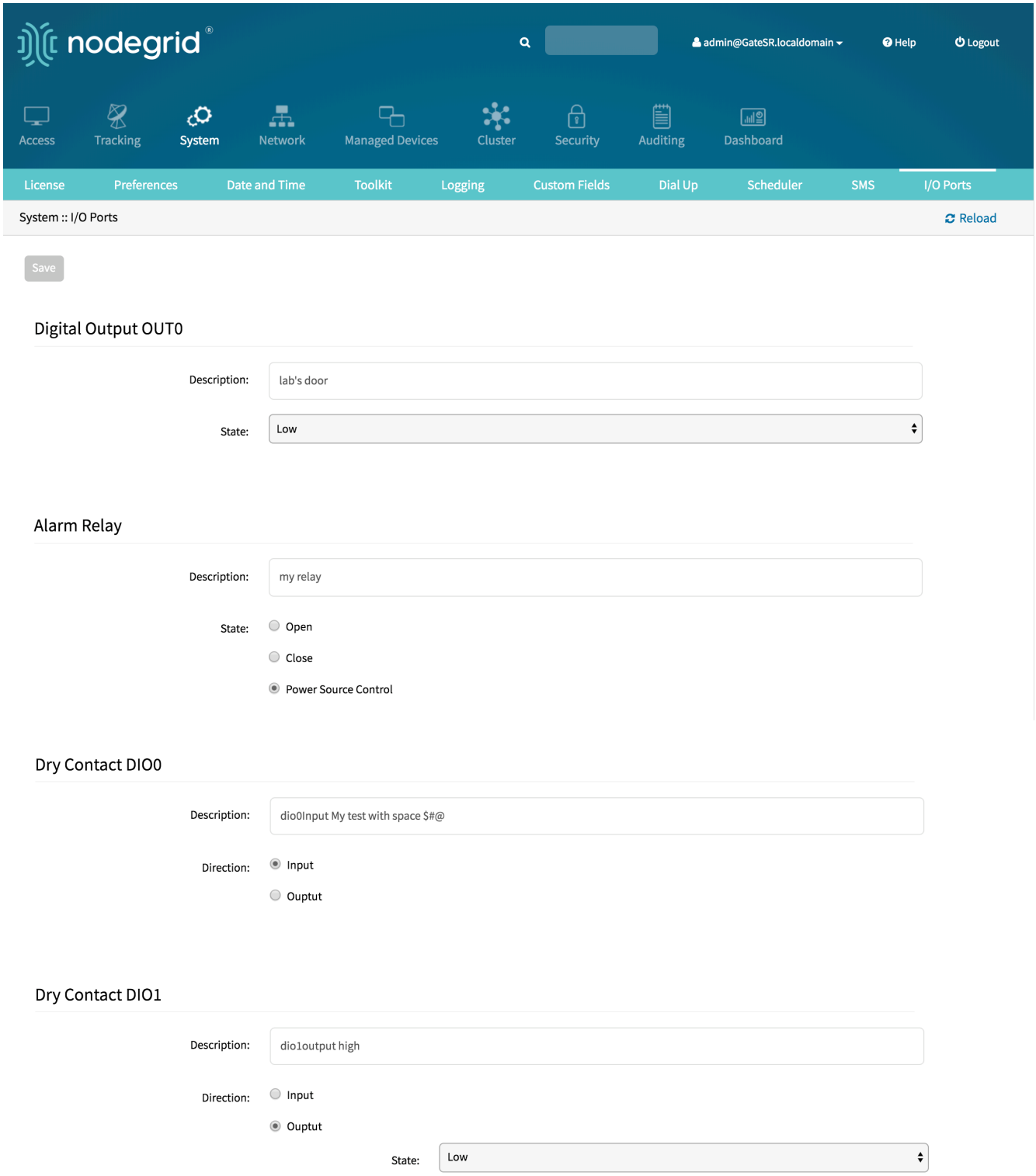
設定	値	説明
名前	文字列	名前
電話番号	電話番号	許可された電話番号

> 注: ホワイトリスト表が空の場合、すべての電話番号からのリクエストを受け付けます。



デジタル I/O

GPIO (デジタル I/O ポート) を搭載した Nodegrid モデルでは、システム`System :: I/O Ports`内に`I/O Ports`タブがあります。 このページでは、デジタル出力の状態と、入力/出力としての DIO0 と DIO1 を設定できます。DIO0/DIO1 を出力として設定すると、その状態を**低**または**高**に設定できます。


nodegrid
admin@GateSR.localdomain
Help
Logout
Access
Tracking
System
Network
Managed Devices
Cluster
Security
Auditing
Dashboard
License
Preferences
Date and Time
Toolkit
Logging
Custom Fields
Dial Up
Scheduler
SMS
I/O Ports
System :: I/O Ports
Reload
Save
Digital Output OUT0
Description:
lab's door
State:
Low
Alarm Relay
Description:
my relay
State:
Open
Close
Power Source Control
Dry Contact DIO0
Description:
dio0Input My test with space $#@
Direction:
Input
Ouptut
Dry Contact DIO1
Description:
dio1output high
Direction:
Input
Ouptut
State:
Low

ネットワーク

ネットワークメニューで、管理者はネットワーク、LTE、WIFI インターフェースの設定、ボンディングや VLAN の詳細の設定など、すべてのネットワークに関連したセッティングを設定・調整できます。

設定

ネットワーク設定メニューで、管理者はユニットホストとドメイン名の定義、複数のインターフェース間でのネットワークフェイルオーバーの設定、IP 転送の有効化、およびループバックアドレスの設定を行うことができます。

ホスト名とドメイン名

ユニットホスト名とドメイン名は、ネットワーク設定メニューで定義できます。両方の設定に適切な値を指定します。

ネットワークフェールオーバー

ネットワークフェイルオーバーオプションにより、管理者は 2 または 3 つの異なるネットワークインターフェース間で、自動的にフェールオーバーできます。

各フェールオーバー設定について、管理者は次の設定を定義できます:

設定	オプション	説明
一次接続	インターフェース	使用可能なすべてのネットワークインターフェースのリスト。1 つを選択する必要があります
二次接続/三次接続	インターフェース	使用可能なすべてのネットワークインターフェースのリスト。1 つを選択する必要があります
トリガ	到達不能プライマリ接続 IPv4 デフォルトゲートウェイ 到達不能 IP アドレス	設定に基づいて、システムはデフォルトゲートウェイまたは指定可能なアドレスのうちのいずれかの可用性をチェックします。
フェールオーバーへの再試行に失敗した回数	数	失敗回数はトリガアドレスに到達しようとします。この値は、フェールオーバーをトリガするために使用されます。
リカバーの再試行に成功した回数	番号	トリガアドレス到達への試行に成功した回数。この値は、フォールバックをトリガするために使用されます。
再試行の間隔 (秒)	番号	試行の間の待機時間

システムは、フェールオーバーインターフェースの動的 DNS の設定をサポートします。

IPv4およびIPv6プロファイル

IP フォワーディング

IP フォワーディングは、ネットワークインターフェース間のネットワークトラフィックをルーティングするために使用できます。ルーティングトラフィックの動作は、ファイアウォール設定を使用してさらに調整することが可能です。

IP フォワーディングは、IPv4 および IPv6 に対して個別に有効化できます。

ループバックアドレス

Nodegrid システムでは、必要に応じて IPv4 および IPv6 のループバック アドレスを設定できます。設定されたアドレスは、/32 (IPv4) または /128 (IPv6) のビットマスクで割り当てられます。

リバースパスフィルタリング

この`Reverse Path Filtering`設定により、管理者は Nodegrid デバイスのリバースパスフィルタリングの動作を設定できます。Nodegrid では、これが*厳密モード*にデフォルト設定されています。これは DDoS 攻撃の一部の形式からシステムを保護するため、大半の環境に推奨されます。

動的ルーティングプロトコルやその他の特定のネットワークの設定シナリオにおいて、この値を変更する必要がある場合があります。以下のオプションを使用できます:

値	説明
無効	送信元アドレスの検証は実行されません。
Strict モード	インターフェースが最適なリターンパスを表す場合、Nodegrid への各着信パケットは、ルーティングテーブルに対してテストされます。パケットをルーティングできない、または最良のリターンパスではない場合、それはドロップされます。
Loose モード	各着信パケットは、ルーティングテーブルに対してのみテストされます。パケットをルーティングできない場合は、ドロップされます。これにより、非対称ルーティングシナリオが可能になります。

複数ルーティングテーブル

Nodegrid では複数のルーティングテーブルがサポートされており、特定のネットワークインターフェースまたは IP クライアントに特定のルーティングの詳細を割り当てることができます。この機能はデフォルトで有効化されています。管理者は、必要に応じてこの機能を無効にするオプションを使用できます。

ネットワーク接続の設定

ネットワーク接続設定で、管理者は既存のネットワーク設定を編集、追加、削除できます。Nodegrid ソリューションは、既存のすべての物理インターフェースを自動的に追加します。モデルに応じて、以下の物理インターフェースが存在します。

インターフェース	モデル	物理インターフェース
ETH0	すべて	eth0
ETH1	Nodegrid Serial Consoles、Nodegrid Services Router	eth1
BACKPLANE0	Nodegrid Bold SR、Nodegrid Services Router	backplane0 は、スイッチポートと sfp0 (Nodegrid Services Router) への接続を提供します
BACKPLANE1	Nodegrid Services Router	backplane1 は、sfp1 への接続を提供します
ホットスポット	すべて	インターフェースがバインドされたワイヤレス アダプタ (使用可能な場合)

管理者は、各インターフェースの以下の設定を定義できます

設定	説明
説明	インターフェースの説明
一次接続として設定する	インターフェースをユニットのプライマリ接続として定義します。プライマリに設定できるインターフェースは 1 つだけです
この接続を介して LLDP アドバタイズおよびレセプションを有効化します	インターフェースを介して LLDP アドバタイズを有効化します
(IPv4/IPv6) モード	インターフェースに使用する IP モードを定義します。使用可能なのは No (IPv4/IPv6) アドレス DHCP (IPv4) アドレス自動設定 (IPv6) ステートフル DHCPv6 静的 (IPv4/IPv6)
(IPv4/IPv6) アドレス	モードが静的に設定されている場合は、静的 IP アドレスを定義します
(IPv4/IPv6) ビットマスク	モードが静的に設定されている場合は、静的 IP ビットマスクを定義します
(IPv4/IPv6) ゲートウェイ	モードが静的に設定されている場合は、静的 IP ゲートウェイを定義します (オプション)
(IPv4/IPv6) DNS サーバ	この接続に使用する DNS サーバを定義します。モードが静的に設定されている場合は、静的 IP ゲートウェイを定義します (オプション)
(IPv4/IPv6) DNS 検索	DNS ルックアップに使用されるドメイン名を定義します

また、既存の物理インターフェースには、追加のインターフェースを定義できることから、より高度な設定オプションが可能になります。次のインターフェースの種類がサポートされています。

インターフェース	説明
ボンディング	フェールオーバー目的で複数のインターフェースのボンディングが可能です
イーサネット	追加の物理インターフェースの設定が可能です
モバイルブロードバンド GSM	使用可能な LTE モデム接続を設定できます
VLAN	このオプションを使用すると、物理インターフェースにバインドされている VLAN インターフェースの設定が可能になります。
WiFi	このオプションを使用すると、WIFI インターフェースを WIFI クライアントまたはホットスポットとして設定できます。WiFi インターフェースの名前は、デフォルトで既に存在します。 `hotspot`
ブリッジ	1 つまたは複数の物理インターフェースのブリッジインターフェースを作成できます。

ボンディングインターフェース

ボンディングインターフェースを使用すると、2 つの物理ネットワークインターフェースを 1 つのインターフェースに結合できます。結合内のすべての物理インターフェースは、1 つのインターフェースとして機能します。これにより、インターフェースへの物理接続が中断された場合に、2 つのインターフェース間でアクティブなフェールオーバーが可能になります。内蔵機能*ネットワークフェールオーバー*は、同じ目的で使用できます。主な違いは、ボンディングインターフェースがリンク層で動作する替わりに、より多くの機能を可能にする、内蔵機能*ネットワークフェールオーバー*が IP 層上で動作することです。

> 注: 内蔵機能*ネットワークフェールオーバー*とボンディングは、組み合わせることができます。

管理者は、各ボンディングインターフェースの IP アドレス、ビットマスク、以下の特定の設定などの通常のネットワーク設定を定義できます。

設定	説明
一次インターフェース	一次ネットワークインターフェース
二次インターフェース	二次ネットワークインターフェース
ボンディングモード	ボンドモードを使用するように設定できますが、有効なオプションは アクティブバックアップ - パケットは 1 つのアクティブなインターフェースを介してのみ送信され、これによりフェールオーバーが可能になります ラウンドロビン - パケットは両方のインターフェースを介してラウンドロビン方式で送信されます。このモードでは、負荷分散とフェールオーバーが可能です
モニタリングをリンクする	リンク監視モードの指定が可能です。有効なオプションは MII ARP
監視中の頻度 (ms)	ミリ秒でインターフェースのリンク状態監視頻度を定義できます。値は MII モードでのみ有効です。
リンクアップ遅延 (ms)	リンクが検出されインターフェースがアップされる前に、ミリ秒の遅延を定義できます。値は MII モードでのみ有効です。
リンクダウン遅延 (ms)	リンクダウンが検出されインターフェースがダウンする前に、ミリ秒の遅延を定義できます。値は MII モードでのみ有効です。
ARPターゲット	ARP 監視リクエストの送信に使用される IP ターゲットを定義できます。値は ARP モード用に定義する必要があります。
ARP確証	ARP 検証に使用するインターフェースを定義できます。オプションは なし アクティブ バックアップ すべて
ボンドフェールオーバー MAC ポリシー	MAC アドレスのフェールオーバーポリシーを定義できます。可能な値は 一次インターフェース 現在アクティブなインターフェース アクティブインターフェースのフォローです

イーサネットインターフェース

物理インターフェースをシステムに追加した後に、イーサネットインターフェースを追加・設定できます。これは、ネットワーク分離をより良くサポートするためにシステムが 2 つ以上のインターフェースを持つ場合の Nodegrid Manager のインストールに該当する場合があります。

モバイルブロードバンド GSM インターフェース

モバイルブロードバンド インターフェースは、モバイルブロードバンドモデムをユニットで使用できる場合に設定できます。Nodegrid Services Router および Nodegrid Bold SR は内蔵モデムをサポートしています。その他のすべてのユニットについては、外付けモデムを使用できます。作成されたインターフェースで、フェールオーバーオプションで最も一般的に使用されるインターネット接続を確立できます。これがISP でサポートされている場合は、ユーザーとリモートシステムは、モバイル接続を介してデバイスに直接アクセスできます。

> 注: 内蔵モデムは、アクティブ/パッシブ SIM フェールオーバーをサポートします。SIM-2 の設定は、内蔵モデムでのみサポートされます。

各モバイルブロードバンド GSM インターフェースについては、管理者は IP アドレス、ビットマスク、次の SIM 固有の設定などの通常のネットワーク設定を定義できます。これらの設定は、ISP 固有であり、モデム接続を設定する前に ISP からリクエストする必要があります。

設定	説明
SIM-1 ユーザー名	SIM のロックを解除するためのユーザー名
SIM-1 パスワード	SIM のロックを解除するためのパスワード
SIM-1 アクセスポイント名 (APN)	アクセスポイント名
SIM-1 暗証番号 (PIN)	SIM のロックを解除するための PIN
第二SIMカードを有効化する	このオプションで、2 番目の SIM カードを設定できます。サポートのみ
アクティブな SIM カード	使用する一次 SIM カードを定義できます。
SIM-2 ユーザー名	SIM のロックを解除するためのユーザー名
SIM-2 パスワード	SIM のロックを解除するためのパスワード
SIM-2 アクセスポイント名 (APN)	アクセスポイント名
SIM-2 暗証番号 (PIN)	SIM のロックを解除するための PIN

VLAN インターフェース

VLAN インターフェースで、Nodegrid システムは、特定の VLAN ID を使用してネットワークトラフィックにネイティブタグを付けることができます。このためには、VLAN インターフェースを作成する必要があります。VLAN インターフェースは、Nodegrid ソリューションの他のネットワークインターフェースと同じ動作と設定を可能にします。新しいインターフェイスは特定の物理インターフェイスにバインドされ、管理者はVLAN IDを定義できます。

WIFI インターフェース

Nodegrid ソリューションは、WiFi クライアントまたはアクセスポイントとして Nodegrid を使用できます。このためには、互換性のある WiFi モジュールをインストールする必要があります。

WIFI アクセスポイント

WiFi モジュールが存在する場合、Nodegrid ソリューションをアクセスポイントとして設定する、`hotspot`インターフェースがデフォルトで定義されています。

Nodegrid をアクセスポイントとして使用するには、既存の値を必要とされる新しい値に変更します。

WIFI クライアント

Nodegrid を使用するために、WiFi クライアントは、設定に移動してオプション`Connect Automatically`を無効にし、既存の`hotspot`接続を無効にする必要があります。 この時点で、`hotspot`インターフェースがダウンしていることを確認します。

この後、Nodegrid がクライアントとして機能できる新しい WiFi インターフェースを作成できます。

WIFI設定

Wifi 設定は、現在`No Security`または `WPA2 Personal` セキュリティ設定オプションをサポートしています。

以下の WiFi 固有の設定が利用可能です。

設定	説明
WiFi SSID	使用する SSID
WiFi BSSID	使用するアクセスポイントの MAC アドレス
隠れネットワーク	有効にすると、SSID はブロードキャストされません
WiFiセキュリティ	セキュリティを以下のいずれかに設定できます セキュリティなし WPA2 パーソナル
WPA共有キー	WPA2 パーソナルがセキュリティとして定義されている場合、共有キーを定義できます

ブリッジインターフェース

ブリッジは、1 つ以上のインターフェースを横断する仮想スイッチを作成するために、全システムのインターフェースとして機能します。スイッチはネットワークインターフェースに対して完全に透過的であり、追加のセットアップは必要ありません。ブリッジネットワークの最も一般的な用途は、ブリッジインターフェースとして NFV とともに、Nodegrid ソリューション上で実行されている任意の NFV のために、外部システムと Nodegrid システム自体による簡単なネットワークアクセスを提供することです。

ブリッジネットワークインターフェースでは、すべてのイーサネットインターフェースと同じネットワーク設定オプションを使用できます。

設定	説明
ブリッジ インターフェース	物理インターフェースのコンマ区切りリスト
スパニングツリープロトコルを有効化する	インターフェースのスパニングツリープロトコルを有効化できます
Hello タイム (秒)	HELLO パケットが送信される秒数。この設定は、スパニングツリーが有効な場合に使用されます。
転送遅延 (秒)	パケット転送遅延を定義できます。この設定は、スパニングツリーが有効な場合に使用されます。
最大年齢 (秒)	パッケージの最大年齢を定義できます。この設定は、スパニングツリーが有効な場合に使用されます。

アナログモデムインターフェース

管理者は、アナログモデムインターフェースで、既存のアナログモデムと必要な PPP 接続についての詳細を設定できます。このオプションを正常に設定するには、サポートされているアナログモデムを Nodegrid システムに接続する必要があります。

以下の設定が可能です。

設定	説明
ステータス	ステータスは接続ステータスを定義します。オプションは 有効化 無効化です
デバイス名	検出されたモデム名、例 `ttyUSB0`
速度	モデムへのシリアル接続速度
PPPダイアルアウト番号	
最初のチャット	このオプションでは、必要に応じて特定の AT init 文字列を定義できます
PPP アイドル タイムアウト (秒)	設定は、接続が自動的に切断された後の接続アイドルタイムアウトを定義します。0 秒は、接続が自動で切断されないことを示します。
PPP IPv4/IPv6Address	この設定では、PPP 接続のための Ipv4 アドレスの定義が可能です。以下のオプションが使用できます アドレスなし ローカル設定 - ローカルとリモート IP アドレスの設定を許可 リモートピアからの設定を受理
PPP認証	この設定では、PPP 認証オプションの定義が可能です。可能なオプションは、なし ローカルシステムによる定義 - `PAP`、`CHAP`、`EAP`の認証プロトコルの定義を許可 リモートピアによる定義 - リモートユーザー名とパスワードの定義を許可

スタティックルート

静的ルート機能で、静的ルートの定義と管理が可能です。ルートは、IPv4 と IPv6 用に作成でき、特定のネットワークインターフェースに割り当てられます。次のオプションがあります。

設定	説明
接続	ルートが関連付けられているネットワーク接続の選択が可能です
タイプ	IP タイプの定義が可能です。オプションは IPv4 IPv6 です
送信先IP	宛先 IP またはネットワークの定義が可能です
送信先ビットマスク	関連するビットマスクを xxx.xxx.xxx.xxx または xx の形式で定義できます。 例: 255.255.255.0 24
ゲートウェイIP	ゲートウェイアドレスを定義できます
メトリック	ルーティングメトリック値を定義できます。通常のルートのデフォルト値は 100 です

手動ホスト名

ホスト名機能を使用すると、ホストのファイル内のエントリと同等の手動ホスト名定義の設定と管理が可能になります。

以下のオプションがあります。

設定	説明
IPアドレス	ターゲットホスト IP アドレスを定義できます。IPv4 および IPv6 形式がサポートされています
ホスト名	ターゲットのホスト名を定義できます
エイリアス	追加のホスト名エイリアスを定義できます

DHCPサーバ

DHCP 機能を使用すると、ターゲットデバイスの DHCP サーバの設定と管理が行えます。DHCP サーバはデフォルトでは設定されておらず、アクティブではありません。DHCPスコープが定義されると、一般的な DHCP のスコープに一致するインターフェースに接続されたすべてのターゲットデバイスへの IP アドレスの提供を開始します。

DHCP サーバは、2 段階で設定されます。最初に、一般的な DHCP スコープとコンフィギュレーションが設定され、作成されます。2番目のステップでは、特定ホストの IP アドレス予約 (`Hosts`) の他、IP アドレスのサーバに使用される IP アドレスの範囲 (`Network Range`) を定義できます。

以下のオプションがあります。

設定	説明
サブネット	使用される IP アドレスサブネットネットワーク。これは、設定されたインターフェースのコンフィギュレーションと一致する必要があります。
ネットマスク	xxx.xxx.xxx.xxx 形式で定義されたサブネットのネットワークマスク
ドメイン	スコープのドメイン名を定義できます
ドメイン名サーバ (DNS)	スコープの DNS サーバを定義できます
ルータ IP	スコープのデフォルトのゲートウェイを定義できます
ネットワーク範囲 - IP アドレスの開始	切断される最初の IP アドレスを定義できます
ネットワーク範囲 - IP アドレス終了	提供される最後の IP アドレスを定義できます
ホスト - ホスト名	IP アドレス予約のホスト名を定義できます
ホスト - HW アドレス	IP アドレス予約が適用される MAC アドレスを定義できます
ホスト - IPアドレス	定義された MAC アドレスに一致する特定のホストに割り当てられる IP アドレスを定義できます

ネットワークスイッチの設定

Nodegrid サーバルータアプライアンスで、内蔵ネットワークスイッチを設定できます。カードとポートを有効にする各ネットワークの高度なネットワーク設定が可能です。現在サポートされている機能には、個々のポートの有効化と無効化、タグ付き (アクセス) とタグなし (トランク) ポートの作成があります。

ネットワーク接続を提供する各カード、Backplane 0/1 および SFP0/1 は、スイッチに直接接続されます。インターフェース Backplane0/1 および SFP0/1 は、デフォルトで有効であり、ZTP、PXE、および DHCP 要求を提供または使用するためにデフォルトで使用可能です。他のすべてのネットワークインターフェースはデフォルトで無効になっています。

すべてのポートは VLAN1 に属し、VLAN2 に属する Backplane1 と SFP1 を除くすべての有効なインターフェース間での直接通信を提供します。

インターフェーススイッチ

インターフェーススイッチは、すべてのスイッチポートの概要、現在のステータスを提供し、有効化、無効化、現在の VLAN 関連 (タグ付き・タグなし) の表示、およびポート VLAN ID の設定が可能です。

ポート VLAN ID は、すべての着信タグなしパケットに割り当てられます。ポート VLAN ID は、その ID が一致する他のポートにパケットを転送するために使用されます。

スイッチポートインターフェースは、ポートが属する VLAN インターフェースを明確に識別します。ほとんどの一般的なシナリオでは、ポートはアクセスポートに相当するタグなしポートか、トランク ポートに相当するタグ付きポートです。

VLAN 設定

VLAN オプションで、管理者は VLAN を作成、削除、管理し、必要に応じてポートを割り当てることができます。デフォルトで、VLAN 1 と VLAN 2 が存在します。すべてのポートは、デフォルトで VLAN 1 に属しますが、デフォルトで VLAN2 に属する BACKPLANE1 および SFP1 を除きます。

タグなし/アクセスポート

タグなしまたはアクセスポートとして特定の VLAN にポートを割り当てるには、ポートを有効にしてから、PORT VLAN ID を目的の VLAN に変更します。これにより、ポートは自動的に VLAN およびタグなしのポートに割り当てられます。

> 注: ポートを割り当てる前に VLAN が存在する必要があります

タグ付き/トランクポート

タグ付きポートにより、着信パケットは VLAN タグを伝送することが可能です。タグ付きポートは、割り当てられた VLAN に属する任意のパケットを受け入れます。これらは、主に複数のスイッチ間のトランク接続を作成するために使用されます。ポートをタグ付きポートとして割り当てるには、タグ付き VLAN としてポートに 1 つ以上の VLAN を追加する必要があります。これは、VLAN 設定を使用して行うことができます。タグ付けされたポート用のポート VLAN ID は、割り当てられた VLAN の 1 つと一致するか、ブランクにする必要があります。この場合、タグなしトラフィックはこのポートで受け入れられません。

> 注: ポートを割り当てる前に VLAN が存在する必要があります

バックプレーンポート

バックプレーン設定は、Nodegrid Platform に直接公開されるスイッチインターフェースを制御します。Nodegrid が既存のスイッチポートまたは VLAN のいずれかと通信するには、少なくとも 1 つのバックプレーンインターフェースが特定の VLAN の一部である必要があります。バックプレーン設定では、現在の VLAN アソシエーションが再び表示され、バックプレーンインターフェース用のポート VLAN ID を設定できます。

VPN

Nodegrid ソリューションは、複数の VPN オプションをサポートしており、システムはさまざまなシナリオで VPN サーバまたはクライアントとして機能することが可能です。現在システムは、ホスト間、サイト間などの SSL VPN クライアントおよびサーバ オプションと IPSec 設定オプションをサポートしています。

SSL VPN

Nodegrid は各種 SSL 設定オプションをサポートしており、システムはお客様の設定とセキュリティのニーズに応じて、SSL クライアントまたは SSL サーバとして機能することが可能です。

SSL VPN クライアント

SSL VPN クライアント設定オプションは、主にフェールオーバーシナリオで使用され、これにより、メインの安全な接続が安全性の低い接続タイプにフェイルオーバーされます。次に、VPN トンネルが両側間のトラフィック保護のために使用されます。Nodegrid が SSL VPN クライアントとして設定されると、この設定はネットワークインターフェース (オプション) にバインドされ、バインドされたインターフェースが開始されるとすぐに、VPN トンネルが自動で確立されます。各種接続とインターフェースの詳細をサポートする複数のクライアント設定を追加できます。

> 注: 設定に応じて複数のファイルが必要です。これは設定が完了する前に表示する必要があります。すべてのファイルを **/etc/openvpn/CA**に配置する必要があります

クライアント設定には、次のオプションがあります。

設定	説明
名前	接続名
ネットワーク接続	トンネルがバインドされるネットワークインターフェースを選択できます。
ゲートウェイIPアドレス	SSL VPN サーバの IP アドレスまたは FQDN
ゲートウェイ TCPポート	接続に使用される TCP ポートのデフォルト値は 1194 です。
接続プロトコル	サポートされている接続プロトコルは、 UTP TCP です。
トンネルMTU	トンネルインターフェースの MTU サイズ
HMAC/メッセージ ダイジェスト アルゴリズム	リストから HMAC 接続アルゴリズムを選択できます
サイファー アルゴリズム	リストから接続暗号アルゴリズムを選択できます
LZOデータ圧縮アルゴリズムを使用する	データ圧縮をサポートするために有効にできます
認証方法	ユーザー認証方法を定義できます。オプションは TLS 静的キー パスワード

	パスワードと TLS です
TLS - CA 証明書	SSL サーバで使用される CA 証明書
TLS - クライアント証明書	SSL サーバによって認識される証明書
TLS - クライアント秘密鍵	クライアント認証秘密鍵
静的キー - 秘密	使用される秘密
静的キー - ローカルエンドポイント (ローカル IP)	VPN 接続用ローカル IP アドレス
静的キー - リモートエンドポイント (リモート IP)	VPN 接続のリモート IP アドレス
パスワード - ユーザー名	接続ユーザー名
パスワード - パスワード	接続パスワード
パスワード - CA 証明書	SSL サーバで使用される CA 証明書*ファイル*
パスワードとTLS - ユーザー名	接続ユーザー名
パスワードとTLS - パスワード	接続パスワード
パスワードと TLS - CA 証明書	SSL サーバで使用される CA 証明書*ファイル*
パスワードと TLS - クライアント証明書	SSL サーバによって認識されるクライアント証明書
パスワードと TLS - クライアント秘密キー	クライアント認証秘密鍵

SSL VPN サーバ

Nodegrid は、SSL VPN サーバとして機能するように設定できます。デフォルトで、サーバは無効になっています。サーバが設定されて起動すると、*SSL サーバの状態*ページに、一般的なサーバのステータスと接続されたクライアントの概要が表示されます。

> 注: 設定に応じて複数のファイルが必要です。これは設定が完了する前に表示する必要があります。すべてのファイルを **/etc/openvpn/CA**に配置する必要があります

以下のサーバ設定オプションがあります

設定	**説明**
ステータス	デフォルト値は、`Enabled`で、完全に設定された後にサーバを起動するために、`Disabled`に設定する必要があります

IPアドレスを聞きます	この設定により、定義されたサーバがこのインターフェースで受信するクライアント要求にのみ応答する場合、リスニング IP アドレスを定義できます。
ポート番号を聞く	この設定では、着信接続のリスニングポートが定義されます。デフォルト値は次の値です `1194`
プロトコル	この値は、使用されるプロトコルを定義します。使用可能なオプションは UDP TCP です
トンネルMTU	トンネルに使用される MTU を定義できます。デフォルト値は次の値です `1500`
同時トンネル数	同時 SSL クライアントセッションの合計数を定義できます。デフォルト値は次の値です `256`
IPアドレス	このセクションでは、トンネルの IP アドレス設定を定義できます。使用可能なオプションは、 ネットワーク ポイントとポイント ポイントとポイント IPv6 です
IP アドレス - ネットワーク - IPv4 トンネル (NetAddr ネットマスク)	トンネルに使用される IPv4 ネットワークアドレスとネットワークマスクを定義できます
IP アドレス - ネットワーク - IPv6 トンネル (NetAddr/ビットマスク):	トンネルに使用される IPv4 ネットワークアドレスとネットワークマスクを定義できます
IP アドレス - ポイントツーポイント - ローカルエンドポイント (ローカル IP)	ポイントツーポイント接続のローカル IPv4 IP アドレスを定義できます
IP アドレス - ポイントツーポイント - リモートエンドポイント (リモート IP)	ポイントツーポイント接続のリモート IPv4 IP アドレスを定義できます
IP アドレス - ポイントツーポイント IPv6 - ローカルエンドポイント (ローカル IP)	ポイントツーポイント接続のローカル IPv6 IP アドレスを定義できます
IP アドレス - ポイントツーポイント IPv6 - リモートエンドポイント (リモート IP)	ポイントツーポイント接続のリモート IPv6 IP アドレスを定義できます
認証方法	これにより、必要な認証方法を選択できます。使用可能なオプションは TLS 静的キー パスワード

	パスワードと TLS です
TLS - CA 証明書	使用する CA 証明書を選択できます
TLS - サーバ証明書	使用するサーバ証明書を選択できます
TLS - サーバキー	サーバ証明書に属する秘密キーを選択できます
TLS - ディフィー ヘルマン	Diffie Hellman キーを選択できます
静的キー - 秘密	使用する秘密を選択できます
静的キー - ディフィー ヘルマン	Diffie Hellman キーを選択できます
パスワード - CA 証明書	使用する CA 証明書を選択できます
パスワード - サーバ証明書	使用するサーバ証明書を選択できます
パスワード - サーバキー	サーバ証明書に属する秘密キーを選択できます
パスワード - ディフィー ヘルマン	Diffie Hellman キーを選択できます
パスワードと TLS - CA 証明書	使用する CA 証明書を選択できます
パスワードと TLS - サーバ証明書	使用するサーバ証明書を選択できます
パスワードと TLS - サーバキー	サーバ証明書に属する秘密キーを選択できます
パスワードとTLS - ディフィー ヘルマン	Diffie Hellman キーを選択できます
HMAC/メッセージ ダイジェスト	リストから HMAC 接続アルゴリズムを選択できます
暗号	リストから接続暗号アルゴリズムを選択できます
最小TLSバージョン	予想される接続 TLS の最小バージョン。サポートされる値は以下の通りです Non TLS 1.0 TLS 1.1 TLS 1.2 TLS 1.3 です
LZOデータ圧縮アルゴリズムを使用する	圧縮されたすべてのトンネルトラフィックを有効にした時
ゲートウェイのリダイレクト (クライアントが生成したすべてのトラフィックをトンネルに強制的に通します)	有効にすると、クライアントから送信されるすべてのトラフィックがトンネルを通って強制的に送信されます。

IPSEC VPN

Nodegrid ソリューションは、IPSec トンネルの設定をサポートします。システムは、ホストとホスト、ホストとサイト、サイトとサイト、ロードウォリアーの設定に対してさまざまな設定オプションをサポートします。

> 注: Nodegrid ノードは、インターネットに直接公開されます。アプライアンスの保護を強くお勧めします。内蔵機能は、次のように使用できます:
>
> - ファイアウォールの設定
> - Fail-2-Ban 有効化
> - すべてのデフォルトのパスワードを、強いパスワードに変更
> - 不要なサービスを無効化

認証方法

IPSec および Nodegrid ソリューションと共に、複数の認証方法が使用できます。事前共有キーや RSA キーなど、これらのいくつかは実装が非常に簡単ですが、大規模なセットアップでは柔軟性が制限されます。一方証明書はより多くの初期設定とセットアップを必要としますが、大規模なセットアップを簡単に管理・維持できる柔軟性と一貫性を提供します。

事前共有キー

事前共有キーは、IPSec 接続をセキュリティ保護するためのシンプルで最も安全性の低い方法です。事前共有キーは、秘密の文字の組み合わせです。両方のノードは、同じ秘密を共有する必要があります。Nodegrid は、32 文字以上の長さの事前共有キーをサポートします。最長ははるかに高いですが、他のベンダーとの互換性から、以下の例では 64 ビットの長さを使用します。一般に、事前共有の時間が長いほど、セキュリティが向上します。

RSA キー

RSA キー/Raw RSA キーは、一般に単一または少数のホスト間の静的設定に使用されます。設定の一部として、お互いの RSA キーを持つように手動で設定されたノード。

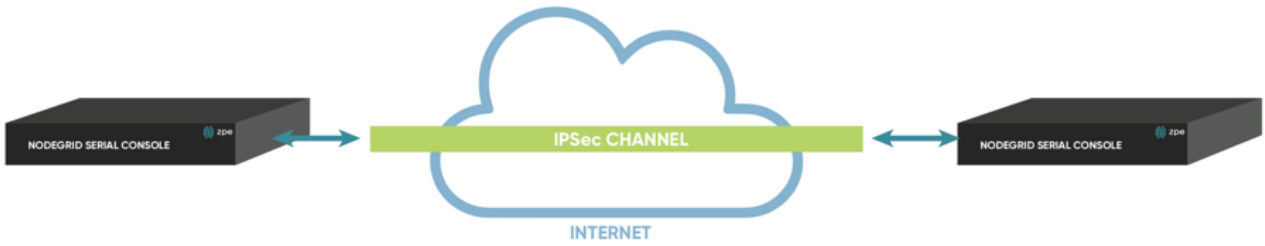
X.509 証明書

X.509 証明書の認証は、通常、小規模なものから多数のノードを持つ大規模な展開に使用されます。個々のノードの RSA キーは、中央証明機関 (CA) によって署名されます。証明機関は、特定のノードに対する信頼の失効など、ノード間の信頼関係を維持するために使用されます。Nodegrid ソリューションは、この目的のパブリックおよびプライベート CA をサポートします。 さらに、Nodegrid ソリューションは、IPSec 通信用の独自の証明機関をホスト・管理するために使用されます。

接続シナリオ

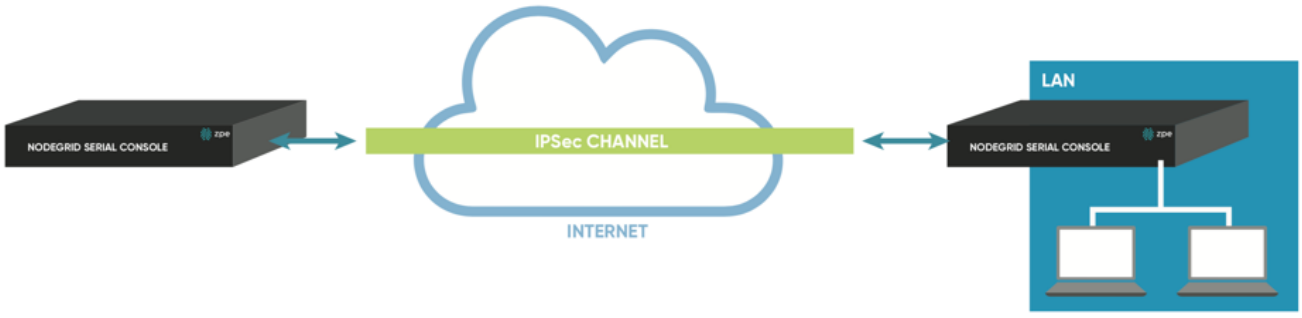
IPSec は、単なる 2 ノード間の通信から、単一ノードから複数ノードへの通信、関連するノードに限定された通信、または直接関係するノードを超えたノードの背後にあるネットワークアクセスへの拡張など、多くの接続シナリオをサポートします。多数の通信オプションがあるため、最も一般的なシナリオが例として示されています。

ホストとホスト



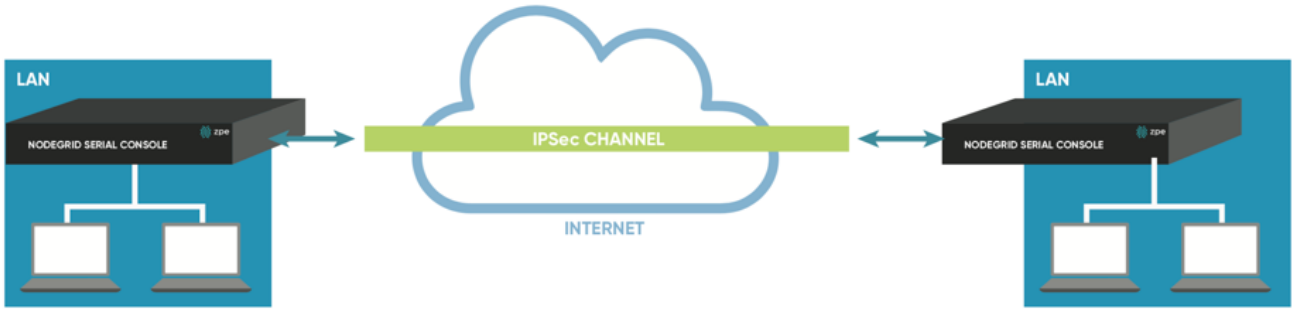
ホスト間通信とは、2 つのノードが VPN トンネルを開き、直接接続することを意味します。トンネルを介して交換される通信は、ホスト間の直接通信に限定されます。いかなるパッケージもルーティングまたは転送されません。これは基本的に、2 ノード間のポイントツーポイント通信です。

サイトとホスト



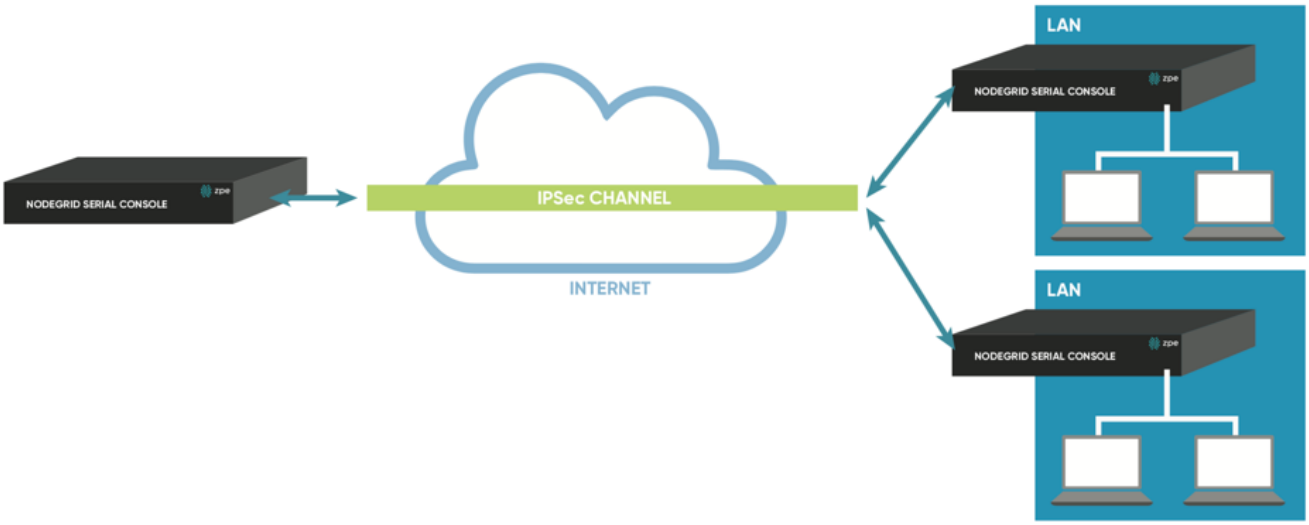
ホストとサイト間の通信シナリオでは、1 つのノードが 2 番目のノードへの VPN トンネルを確立します。通信は、特定のードへの 1 つのサイトに制限され、他方では、2 番目のノードからアクセス可能なサブネットの範囲のすべてのデバイスに制限されます。

サイトとサイト



サイト間通信では、トンネルは 2 ノード間で確立される前と同様で、通信は両側でサブネットを指定することができ、接続の両側にあるデバイス間の通信を可能にします。

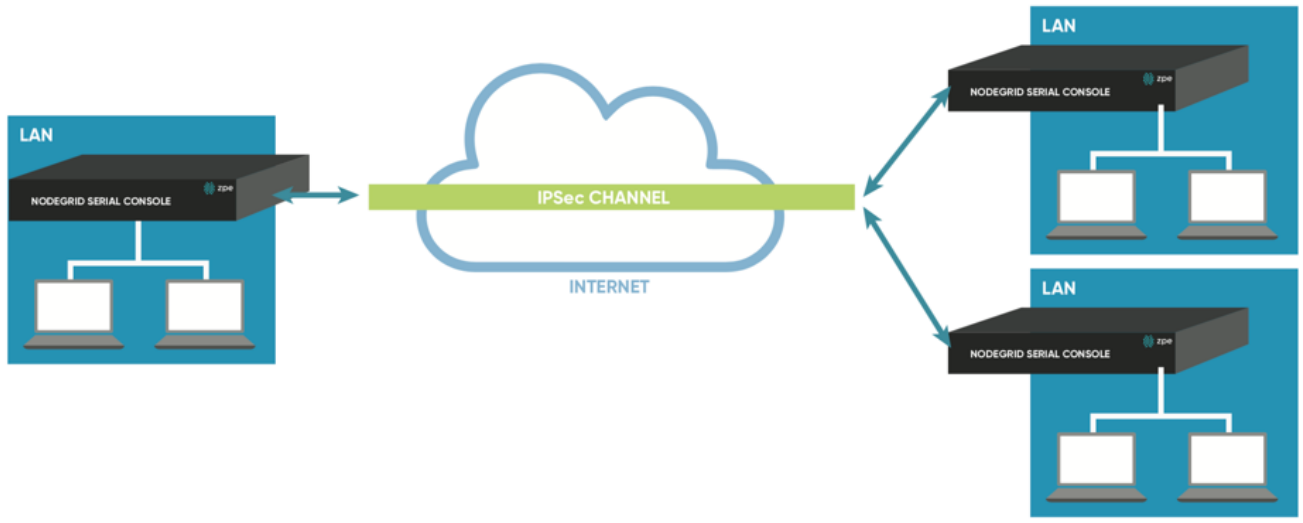
ホストとマルチサイト



マルチサイトの通信シナリオは、ホスト間で個々の VPN 接続を作成するか、特定のマルチサイト設定によって作成することができます。その後、接続設定のスケーラビリティと管理性が大幅に向上します。

ホストとマルチサイトの通信では、複数ノードが同じノードに接続できます。この一般的なシナリオは、リモートオフィスがメインオフィスに VPN 接続している場合です。この特定のシナリオでは、通信はノード 1 つと遠隔地で指定されたサブネット上のデバイスに制限されます。

サイトとマルチサイト



おそらくこのシナリオは、エンタープライズ VPN セットアップの中で最も一般的な形式です。これは [ホストとマルチサイト] のオプションに似ていますが、どちらかの側の特定のサブネットへの通信が許可され、これにより West ノードは任意のサイトで指定されたすべてのサブネットにアクセスが可能ですが、リモートサイトは West ノードによって公開されるサブネットにのみアクセスできます。

	ホストとホスト	サイトとホスト	サイトとサイト	ホストとマルチサイト	サイトとマルチサイト
事前共有キー	可能	可能	可能	可能	可能
RSA キー	推奨	推奨	推奨	可能	可能
X.509 証明書	推奨	推奨	推奨	推奨	推奨

IPSec の設定

このセクションでは、必要な接続を設定するために使用可能な一般的な設定手順の概要を説明します。

- Nodegrid の準備。参照：IPSec 用 Nodegrid ノードの準備方法
- 認証方法の 1 つが準備されていることを確認します
  - IPSec の事前共有キーの作成方法
  - IPSec の RSA キーの作成方法
  - IPSec の証明書の作成方法

> 注: 生産環境では、RSA キーまたは証明書認証を使用することをお勧めします。事前共有キーは簡単に設定でき、テスト環境の出発点として適しています。

- IPSec 設定ファイルを作成します。 設定例は、以下で確認できます:
  - 事前共有キー
    - 事前共有キーを使用して IPSec ホストとホストトンネルを設定する方法
    - 事前共有キーを使用して IPSec ホストとサイトトンネルを設定する方法
    - 事前共有キーを使用して IPSec サイトとサイトトンネルを設定する方法
  - RSA キー
    - RSA キーを使用して IPSec ホストとホストトンネルを設定する方法
    - RSA キーを使用して IPSec ホストとサイトトンネルを設定する方法
    - RSA キーを使用して IPSec サイトとサイトトンネルを設定する方法
  - 認証
    - 証明書を使用して IPSec ホストとホストトンネルを設定する方法
    - 証明書を使用して IPSec ホストとサイトトンネルを設定する方法
    - 証明書を使用して IPSec サイトとサイトトンネルを設定する方法
- 必要に応じて、設定ファイルとキーをすべてのノードに配布および交換します
- 接続をテストする

Nodegrid ソリューションで IPSec を使用する方法についての詳細は、ナレッジ ベースを参照してください。

高度なネットワーク機能

VRRP (仮想ルータ冗長プロトコル) サポート

Nodegrid Platform は、内蔵された仮想ルータ冗長プロトコル(VRRP) をサポートしています。このプロトコルを使用すると、Nodegrid は仮想ルータ インターフェースの一部となり、ルータの冗長性を確保できます。これは主に、デフォルトゲートウェイの自動フェールオーバーサポートのために使用されます。デフォルトでは、プロトコルは設定されず、サービスは実行されていません。このサポートを有効にするには、サービスを設定する必要があります。管理者は、Shell を使用してこれを行うことができます。

> 注: VRRP は、Nodegrid OS に直接公開されるネットワークインターフェースでのみ使用できます。たとえば、Nodegrid Services Router カード上の個々のスイッチ ポートは使用できません。

VRRP サポートは、keepalived サービスを通じて実装されます。サービスの公式文書はこちらからご覧いただけます。

このサービスの設定ファイルは、`/etc/keepalived/` にあり、有効に設定を行うために最低限 `keepalived.conf` が必要です。その後、次のコマンドでサービスを開始できます。

```
/etc/init.d/keepalived start
```

次にシステムを起動する時に keepalived を自動で開始させるには、次のコマンドを実行します。 `update-rc.d -s keepalived defaults 90`

認証

認証は、あなたが誰であるか、またはあなたが誰であると主張するかを検証するプロセスであり、通常は資格情報を使用して行われます。ほとんどの場合、資格情報はユーザー名とパスワードの形式を取ります。

認証は重要なセキュリティ対策の一部を構成し、証明を補完します。資格情報を使用して認証されると、その認証によりアクセス可能な範囲が決定されます。例: 特定のディレクトリ、電源、シリアルデバイス。

Nodegridには、ユニット、ネットワーク、セキュリティ、認証、許可、管理の対象となるデバイス、その他のユーザーへのフルアクセス権と全設定権を持つ [*admin*] という名前の組み込み管理者ユーザーアカウントがあります。この特別なユーザーアカウント [*admin*] は削除できず、デフォルトのパスワード [*admin*] を持ちます。

> 注: セキュリティ上の理由から、管理者は、WebUI の右上隅にあるユーザー名の下のプルダウンメニューの *パスワードの変更*オプションを使用して、初回ログイン時にデフォルトのパスワードを変更することを強くお勧めします。

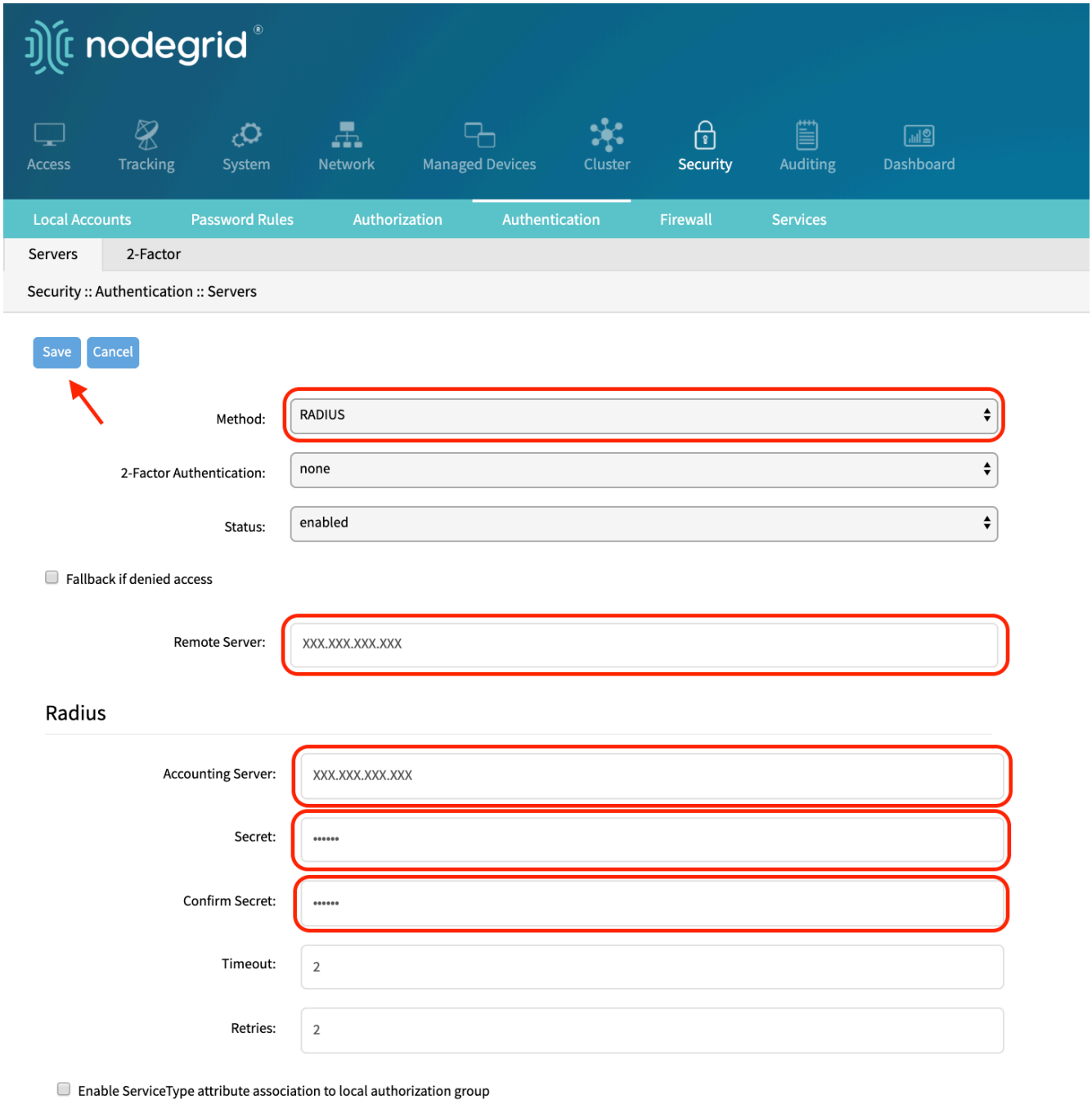
Nodegrid Platformは、ローカルユーザーとグループ、および外部ユーザーとグループの認証を完全にサポートします。ユーザーとグループの外部認証は、LDAP/AD、Tacacs+、Radius および Kerberos を介して行えます。

すべてのユーザーは、有効なすべての管理対象デバイスにデフォルトでアクセス可能です。詳細設定認証は、`Services`の下のオプション`Device access enforced via user group authorization`を選択して有効化できます。

割り当てられているグループに基づいて、これらのユーザーは Nodegrid Web ポータル管理属性へのアクセスが制限されています。ユーザー権限は、権限グループでプロファイルとアクセス権を設定することで変更できます。グループ *管理者*に属するユーザーは、管理者ユーザーと同じ管理者権限を持ちます。 各ユーザーは、Nodegrid または外部認証サーバに特定のユーザーアカウントを持っている必要があります。1 つ以上の権限グループにユーザーを割り当てることができます。

サーバを追加する

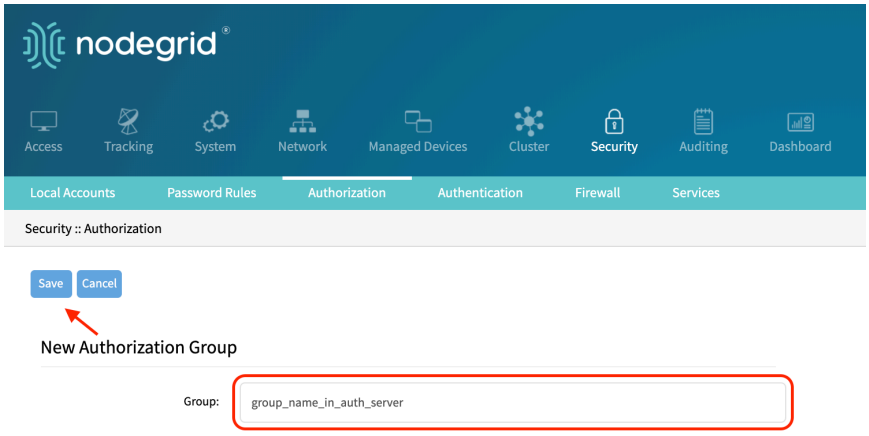
`Security :: Authentication :: Servers`に移動し、認証サーバを追加して、任意のサーバのユーザーをグループに関連付けます。



グループを追加

`Security :: Authorization`に移動して、`Add`ボタンをクリックすると、有効なグループが追加されます。 既にユーザーが関連付けられているグループをいくつか追加することを忘れないでください。次のフィールドを設定・保存してグループを追加します:

- グループ: `group_name_in_auth_server`



ローカルアカウント

新しいローカル ユーザーは、 `Security` の `Local Accounts` 下に追加、削除、変更、およびロックできます。 管理者は、次回のログイン時のパスワードの強制変更や、ユーザーアカウントの有効期限を設定できます。有効化オプションに関係なく、ユーザーはいつでもパスワードを変更できます。この機能は、すべてのユーザーとその情報を一覧表示します。

- ユーザー名とパスワード
- ハッシュ形式のパスワード (オプション)
- アカウントの有効期限 (オプション)
- グループ、 ユーザーが属するグループ

ローカルユーザーの管理

ローカルユーザーの管理は、 `Security :: Local Accounts` で行います。以下のオプションを使用できます。

- `Add` - 新規ユーザーを追加できます
- `Edit` - 既存のユーザー設定を変更できます
- `Delete` - 既存のユーザーを削除できます
- `Lock` - 既存のユーザーをロックすることができ、これによりアカウントを削除することなく、ユーザーのログインを防ぐことができます
- `Unlock` - 既存のロックされたユーザーアカウントのロックを解除できます

ローカルユーザーの追加

- `Security :: Local Accounts` に移動すると、すべてのローカルユーザーが表示されます
- `Add` をクリックすると、ローカルユーザー情報画面が表示されます
- 新しいユーザー名とパスワードを入力します
    - パスワードがハッシュ形式の場合、 `Hash Format Password` チェックボックスにチェックを入れ、 以下の**ハッシュ形式のパスワード**を表示します
- 必要に応じて、
    - アカウントの有効期限日を入力します

    - *ログイン時にパスワードの変更が必要*のチェックボックスをオンにします
- 使用可能なユーザーグループにユーザーを追加するには、左側のボックスからグループ名を選択し、`Add`をクリックします。ボックスからユーザーグループを削除するには、そのグループを選択してクリックします `Remove`
- `Save`をクリックします。

ハッシュ形式のパスワード

あなたが管理者で、単純なパスワードを使用したくない場合は、この機能を使用して、替わりにハッシュ形式のパスワードを使用することができます。これは、ユーザーの実際のパスワードを含めたり表示するスクリプトを避けるために、スクリプトを使用する場合に特に有益かもしれません。

ただし、設定したハッシュパスワードジェネレータを使用して、事前にハッシュパスワードを個別に生成する必要があることにご注意ください。Linux で一般的なハッシュジェネレータの例としては、OpenSSL、chpasswd、mkpasswd、MD5、SHA256、SHA512 などがあります。

Nodegrid は、独自の OpenSSL 実装をこの目的にも使用できます。Nodegridの OpenSSL バージョンの使用例

```
root@nodegrid:~# openssl passwd -1 -salt mysall
Password:
$1$mysall$YBFr9On0wjde5be32mC1g1
```

パスワード ルール

すべてのローカルユーザー アカウントには、パスワードルールが適用されます。これらは`Security :: Password Rules`から調整できます。 管理者は、*パスワードの複雑さ*と*パスワードの有効期限*の値を、*最短日数*、*最長日数*および*警告日*のセットとして設定できます。

以下の設定を調整できます。

設定	値	説明
パスワード複雑性を確認する	TRUE FALSE	パスワードの複雑性のルールを有効または無効にします。デフォルト値は無効です
パスワードの複雑性 - 最小桁数	番号	パスワードに含める必要がある最小桁数。デフォルト値: 0
パスワードの複雑さ - 大文字の最低数	番号	パスワードに含める必要がある大文字の最小数。デフォルト値: 0
パスワードの複雑性 - 特殊文字の最小数	番号	パスワードに含める必要がある特殊文字の最小数。デフォルト値: 0
パスワードの複雑性 - 最小サイズ	番号	パスワードに含まれる最小の文字数。デフォルト値: 8
パスワードの複雑さ - 履歴に記憶させるパスワード数	番号	パスワード履歴に保存されるパスワードの数。この数のパスワードを再利用できなくします。デフォルト値: 1
パスワードの有効期限 - 最小日数	日数	パスワードを変更する前に有効にする必要がある日数。デフォルト値: 0
パスワードの有効期限 - 最大日数	番号	パスワードを変更する前に有効にできる最大日数。デフォルト値: 99999
パスワードの有効期限 - 警告日数	番号	パスワードの有効期限が切れる前にユーザーが通知される日数。デフォルト値: 7

グループ

Nodegrid は、ユーザーグループを使用して複数のローカルユーザーとリモートユーザーを 1 つのローカルグループにまとめます。これは、ユーザーアクセス許可や管理者アクセス許可などのシステム全体の管理ロール/許可を割り当てるために使用されます。さらに、グループは特定のターゲットデバイスへのアクセス許可を付与するために使用されます。外部認証プロバイダに対して認証されたユーザグループは、ローカルグループにマップされ、これにより、割り当てられたローカルグループのアクセス許可がリモートグループに割り当てられます。

ユーザーが複数のグループのメンバーである場合、結合されたアクセス権が有効になります。

管理者は、グループを追加および削除し、そのアクセス許可を変更できます。Nodegridに初めてログインすると、デフォルト設定で**管理者**と**ユーザー**の 2 つのグループが表示されますが、削除することはできません。

グループの管理

Nodegrid Platform には、デフォルトでアクセス権限を持つ 2 つのデフォルトグループが含まれています。`admin`は、管理者ユーザーにシステムとターゲットへの完全なアクセス権を付与します。 `user` グループは、ファイングレイン認証が無効になっている場合 (デフォルト)、すべてのメンバーにすべてのターゲットへのフルアクセス権を付与します。ファイングレイン認証が有効になっている場合、`user`グループメンバーは、デフォルトでどのターゲットデバイスにもアクセスできません。

管理者は、 `Security :: Authorization`でグループを作成、編集、削除できます。

ユーザーグループの作成

- `Security :: Authorization`に移動して、すべてのグループを表示します
- `Add`をクリックして、新しいグループ名を入力し、 `Save`

この時点で、グループのプロパティとアクセス許可を変更するためにグループが作成されます。次にグループ名をクリックします。

ローカルユーザーをグループに追加します

- `Security :: Authorization`に移動して、すべてのグループを表示します
- メンバーを追加するグループの名前をクリックします
- `Members`をクリックします。これにより、グループに既に含まれるメンバーのリストが表示されます。
- `Add`をクリックすると、左側のボックスに追加できるローカルユーザーのリストが表示されます。
- ユーザーを選択し`Add`をクリックして、選択したユーザーを右のボックスのこのグループに移動させます。
- 逆の場合は、`Remove`をクリックして、このグループ内の選択したユーザーを削除し、それをローカルユーザーのボックスに戻します。

システムのアクセス許可と設定をグループに割り当てます

- `Security :: Authorization`に移動して、すべてのグループを表示します
- メンバーを追加するグループの名前をクリックします
- `Profile` をクリック

ユーザーグループには、複数の追加のシステムアクセス許可を割り当てることができます。すべてのグループはデフォルトで`user`アクセス許可を持ち、彼らに`Access`テーブルへのアクセス権を付与し、特定のターゲットアクセス許可に基づきターゲットデバイスへの接続を可能にします。

以下のシステムアクセス許可を割り当てることができます。

> 注: 同じグループに複数のアクセス許可を割り当てることができます。

許可	説明
トラックシステム情報	追跡情報へのアクセス権を付与します。セクションを参照してください。 `Tracking`
セッションを終了する	端末ユーザーおよびデバイスセッションへのアクセス許可を付与します
ソフトウェア アップグレードと再起動システム	システムのアップグレードと再起動を実行するためのアクセス許可を付与します
システムを設定します	システム設定を変更するための管理者権限を付与します
ユーザーアカウントを設定する	認証設定を変更するためのアクセス権限を付与します。
設定を適用、保存する	設定を保存するためのアクセス権限を付与します
Shell アクセス	システム Shell へのアクセス権限を付与します

以下の設定が行えます

設定	値	説明
許可	システム情報の追跡 セッション終了 ソフトウェアのアップグレードとシステムの起動 システム設定 ユーザーアカウントの設定 設定の適用 & 保存 Shell アクセス	システム許可
設定システム許可を読み出し専用に制限する	True False	付与されたシステム設定は表示されますが、変更はできません
デバイスへのメニュー駆動型アクセス	True False	ssh の Nodegrid への直接接続が確立されると、グループのメンバーにターゲットメニューが表示されます。
カスタム セッション タイムアウト	TRUE FALSE	カスタムセッション時間を有効にします
タイムアウト [秒]	数	セッションタイムアウト (秒単位)
スタートアップ・アプリケーション	CLI Shell	このグループのユーザーが ssh を介して Nodegrid ユニットに接続する時に、管理者がデフォルトの開始アプリケーションを設定できるようにします。デフォルト: CLI
メールイベントの送信先	Eメールアドレス	イベントが送信されるEメールアドレスのリスト

外部グループを割り当てます

外部グループをローカルグループに割り当てる必要があります。これにより、リモートグループに正しいアクセス許可が割り当てられるようにします。外部グループを割り当てるには、以下の手順に従います

> 注: この手順は、LDAP、AD、Kerberos グループに必要です。Radius および Tacacs 認証プロバイダは、外部グループ/ユーザーをローカルグループにリンクするための他の方法を提供します。

- `Security :: Authorization`に移動して、すべてのグループを表示します
- メンバーを追加するグループの名前をクリックします
- `Remote Groups`をクリック
- ローカルグループに割り当てられる、コンマで区切られた外部グループ名を一覧表示します
- `Save` をクリック

デバイスのアクセス許可を割り当てます

ファイングレイン認証が有効になっている場合は、特定のデバイスにアクセスするための権限をグループに割り当てる必要があります。これを行うには、特定のデバイスをグループに追加し、ターゲットへの適切なアクセス権限を設定します。複数のデバイスを同時に追加でき、アクセス許可を一緒に設定できます。

> 注: 電源コンセントを制御するためのアクセス許可は、`Outlets`アクセス許可を介して付与され、以下を介しては付与されません `Devices`

アクセス許可は、各グループが必要に応じて追加、削除、および編集できます。

- `Security :: Authorization`に移動して、すべてのグループを表示します
- メンバーを追加するグループの名前をクリックします
- `Devices`をクリック
- `Add`をクリック
- 管理対象デバイスを左側の使用可能なデバイスリストから右側の認証済みデバイスのリストに移動するには、名前をダブルクリックするか、デバイスを選択して、*追加*をクリックします。
- デバイスをダブルクリックするか、削除するデバイスを選択した後に削除ボタンをクリックすることで、デバイスを右側のボックスから削除できます。
- 必要なデバイス権限を選択します
- `Save` をクリック

以下のアクセス許可を割り当てることができます

許可	値	説明
セッション	読み取りと書き込み可能 読み取り専用 アクセス権なし	シリアルセッション、または ssh セッションへのアクセス許可 (コンソール)
電源	電源制御 電源ステータス アクセスなし	IPMI を介した電源制御の許可
ドア	ドアコントロール ドアステータス アクセスなし	ドア制御の許可

MKS	TRUE FALSE	MKS セッションへのアクセス
デバイスをリセットする	TRUE FALSE	デバイスセッションをリセットする許可
KVM	TRUE FALSE	KVM セッションへのアクセス
SPコンソール	TRUE FALSE	IPMI コンソールセッションへのアクセス (LAN 上のシリアル)
仮想メディア	TRUE FALSE	IPMI デバイスへの仮想メディアセッションを確立するためのアクセス
アクセスログ監査	TRUE FALSE	IPMI デバイスのアクセスログを読み取るためのアクセス
アクセスログクリア	TRUE FALSE	IPMI デバイスのアクセスログを消去するための許可
イベントログ監査	TRUE FALSE	デバイス固有のイベントログを読み取るための許可
イベントログクリア	TRUE FALSE	デバイス固有のイベントログを消去するための許可
モニタリング	TRUE FALSE	監視機能へのアクセス許可
センサデータ	TRUE FALSE	センサーデータの読み取り許可
カスタムコマンド	TRUE FALSE	カスタムコマンドの実行許可

電源コンセントの許可の割り当て

ラック PDU からの電源コンセントのためのアクセス許可は、デバイスのオン/オフを切り替える電力が、データセンターまたはリモートロケーションの稼働に重大な影響を及ぼす可能性があるため、個別に制御されます。許可の割り当ては、デバイスのアクセス許可に似ています。

- `Security :: Authorization` に移動して、すべてのグループを表示します
- メンバーを追加するグループの名前をクリックします
- `Outlets` をクリック
- `Add` をクリック

- 管理対象デバイスを左側の使用可能なデバイスリストから右側の認証済みデバイスのリストに移動するには、名前をダブルクリックするか、デバイスを選択して、*追加*をクリックします。
- デバイスをダブルクリックするか、削除するデバイスを選択した後に削除ボタンをクリックすることで、デバイスを右側のボックスから削除できます。
- 必要なデバイス権限を選択します
    - 電源制御 - コンセントのオン/オフ切り替え権限
    - 電源ステータス - 現在のコンセントのステータスの表示許可
    - アクセスなし
- `Save` をクリック

外部認証プロバイダ

Nodegrid では、Platform 上で簡単に外部認証を有効にできます。次の方法でユーザー認証を設定できます:

- アクティブディレクトリと LDAP (軽量ディレクトリアクセスプロトコル)、
- TACACS+ (ターミナルアクセスコントローラ アクセス制御システム プラス)、
- RADIUS (リモート認証 ダイヤルイン ユーザーサービス)
- Kerberos (ID を証明するチケットに基づく)

外部ユーザーが Nodegrid Platform にアクセスできるようにするには、特定の認証プロバイダとは別に、次の手順を実行する必要があります

- 内部グループの作成
- グループへの権限の割り当て
- 外部認証プロバイダの追加、以下参照
- 外部グループの内部グループへのマッピング

認証プロバイダは、`Security :: Authentication` セクションで追加、削除、変更することができます。このセクションでは、現在設定されているすべての認証プロバイダが表示され、認証プロバイダの作成、削除、変更、および順序付けが行えます。 認証プロバイダの順序に従い、最初のユーザー認証に使用されるプロバイダが決まります。認証に失敗すると、ユーザーのアクセスが拒否されるか、次の認証プロバイダで再試行されることがあります。認証プロバイダの設定 `Fallback if denied access` はこれを制御します。 この機能が有効になっていると、次のプロバイダが使用されます。無効にすると、結果に基づいてユーザーアクセスが許可または拒否されます。

> 注: プロバイダを任意の時点でユーザーの認証に使用できない場合、そのプロバイダはスキップされ、次のプロバイダが使用されます。

Nodegrid にアクセスするすべてのユーザーは、グループのメンバーである必要があります。ユーザーをグループメンバーとして識別できない場合は、デフォルトのグループが使用されます。これはデフォルトの `user` グループです。使用されるグループは、`Default Group` オプションを使用して調整できます。

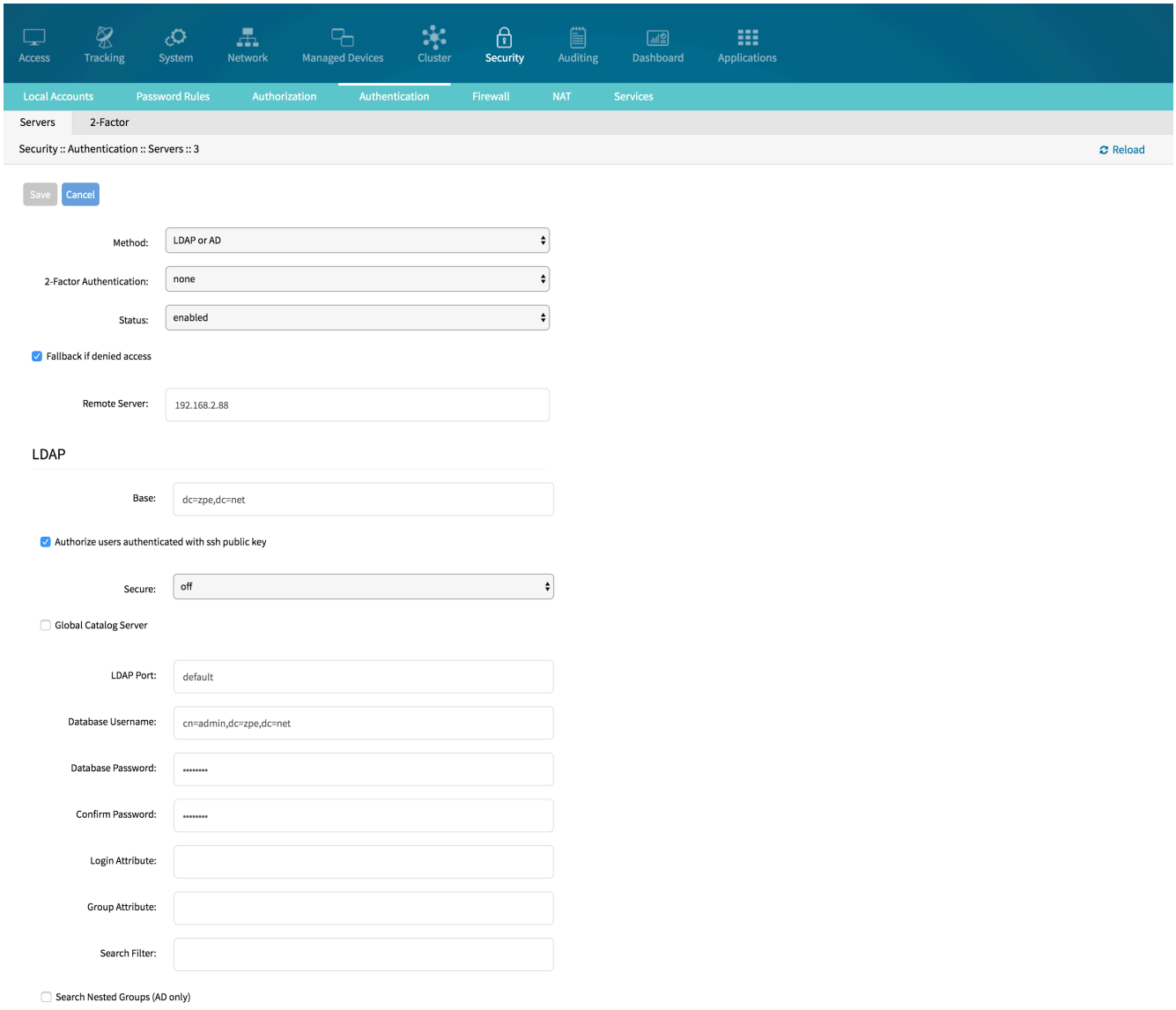
次のセクションでは、さまざまな外部認証プロバイダを追加および設定する方法について取り上げます。

LDAP およびアクティブディレクトリ

LDAP プロトコルはオープンスタンダードであり、多種多様な実装があります。すべて似ていますが、わずかなバリエーションがあります。この LDAP の例は、OpenLDAP 実装に基づいています。

Microsoft のActive Directory は、LDAP 最大の広範に使用されている実装の1つであり、企業の内部組織を反映した非常に複雑な認証プロバイダ構造の実装を可能にします。

LDAP または Active Directory 認証サーバを設定するには、以下の情報が必要です。このページでは、`Fallback if denied access``Authorize users authenticated with ssh public key`や`Search Nested Groups (AD only)`などの機能を有効にできます。



フィールド	値	説明
ステータス	TRUE FALSE	デフォルト値は**有効**です。つまり、プロバイダはユーザーの認証に使用されます。
アクセスが拒否された	有効また	デフォルトは、**無効**です。プロバイダを使用できない場合は、この機

場合フォールバック	は無効	能を**有効**にすることをお勧めします。
リモートサーバ	LDAP サーバまたはドメインの FQDN または IP	Nodegrid は、DNS 要求を介した Active Directory サーバの決定をサポートします。つまり、特定の Active Directory サーバを指定するか、有効な Active Directory ドメインを指定できます。後者の場合、システムは DNS 結果に基づいて最も近いサーバに接続します。
ベース	ベース DN	このフィールドには、ルート DN またはサブレベル DN を含めることが可能です。この DN は、ユーザーまたはグループの検索に使用される最高ポイントを示します。
ssh パブリックキーで認証されたユーザーを許可します	有効または無効	デフォルトで無効
セキュア	オン、オフ、または Start_TLS	デフォルトは**オフ**で、Nodegrid と LDAP サーバ間のすべてのトラフィックは、暗号化されずに送信されます。**オン**をお勧めします。(この機能は、サーバでサポートされている必要があります)
グローバルカタログサーバ	TRUE FALSE	プロバイダが Active Directory グローバルカタログサーバの使用を有効化した時
データベースユーザー名	ユーザー名の検索	ディレクトリでの検索に使用できる完全修飾ユーザー名。LDAP サーバがディレクトリの参照に認証を必要とする場合にのみ必要です
データベースパスワードとパスワードの確認	検索ユーザーのパスワード	LDAP サーバがディレクトリの参照に認証を必要とする場合にのみ必要です
ログイン属性	フィールドはユーザー名を識別します	ユーザー名を含む属性フィールド。Active Directory の場合、これはデフォルトで `sAMAccountName` です。
	フィール	

グループ属性	ドはグループ名を識別します	グループ識別子を含む属性フィールド。Active Directory の場合、これはデフォルトで`memberOf`です
検索フィルタ	LDAP 実装後の検索フィルタ	
ネストされたグループを検索 (AD のみ)	有効または無効	デフォルトで無効

OpenLDAP サーバの設定例

フィールド	**値**
ステータス	TRUE
アクセスが拒否された場合フォールバック	TRUE
リモートサーバ	192.168.1.1
ベース	dc=zpe、dc=net
セキュア	Off
グローバル カタログ サーバ	FALSE
データベースユーザー名	cn=admin、dc=zpe、dc=net
ログイン属性	cn
グループ属性	memberUID

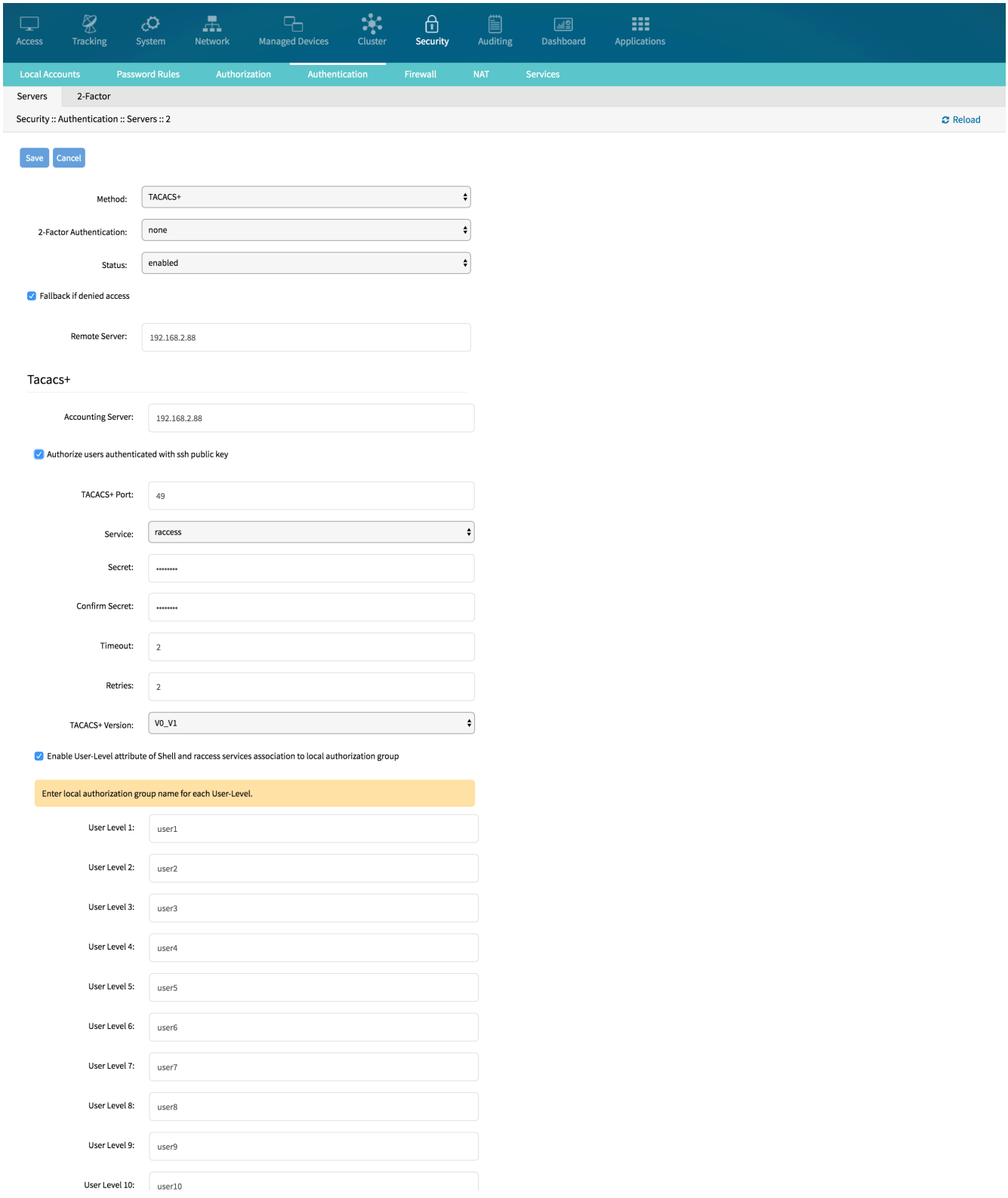
Active Directory サーバの設定例

フィールド	値
ステータス	TRUE
アクセスが拒否された場合フォールバック	TRUE
リモートサーバ	192.168.1.1
ベース	dc=zpesystems、dc=com
セキュア	TLSを起動する
グローバル カタログ サーバ	TRUE
データベースユーザー名	cn=Administrator、cn=Users、dc=zpesystems、dc=com
ログイン属性	sAMAccountName
グループ属性	memberOf

LDAP および Active Directory のセットアップ方法の詳細については、Active Directory または LDAP 認証プロバイダを設定する方法を参照してください。

TACACS +

ターミナル アクセス コントローラ アクセス制御システム プラス (TACACS+) は、Ciscoが開発したプロトコルで、1993 年にオープンスタンダードとしてリリースされました。TACACS から派生したものの、TACACS+ は、認証、許可、およびアカウンティング (AAA) サービスを処理する別個のプロトコルです。TACACS+ およびその他の柔軟な AAA プロトコルは、以前のものをほぼ置き換えました。 このページでは、 `Fallback if denied access` 、 `Authorize users authenticated with ssh public key` 、および `Enable User-Level attribute of Shell and raccess services association to local authorization group` などのオプションを設定にできます。


Access
Tracking
System
Network
Managed Devices
Cluster
Security
Auditing
Dashboard
Applications
Local Accounts
Password Rules
Authorization
Authentication
Firewall
NAT
Services
Servers
2-Factor
Security :: Authentication :: Servers :: 2
Reload
Save
Cancel
Method:
TACACS+
2-Factor Authentication:
none
Status:
enabled
Fallback if denied access
Remote Server:
192.168.2.88
Tacacs+
Accounting Server:
192.168.2.88
Authorize users authenticated with ssh public key
TACACS+ Port:
49
Service:
raccess
Secret:
Confirm Secret:
Timeout:
2
Retries:
2
TACACS+ Version:
V0_V1
Enable User-Level attribute of Shell and raccess services association to local authorization group
Enter local authorization group name for each User-Level.
User Level 1:
user1
User Level 2:
user2
User Level 3:
user3
User Level 4:
user4
User Level 5:
user5
User Level 6:
user6
User Level 7:
user7
User Level 8:
user8
User Level 9:
user9
User Level 10:
user10

フィールド	値	説明
ステータス	有効 無効	デフォルト値は**有効**です。つまり、プロバイダはユーザーの認証に使用されます。
アクセスが拒否された場合フォールバック	有効または無効	デフォルトは、**無効**です。プロバイダを使用できない場合は、この機能を**有効**にすることをお勧めします。
リモートサーバ	IPアドレス	
アカウンティング サーバ	IPアドレス	
sshパブリックキーで認証されたユーザーを許可します	有効または無効	デフォルトで無効
TACACS+ ポート	TCPポート	デフォルトのポート 49
サービス	ppp Shell アクセス	TACACS によって使用される認証サービス。デフォルト値は `raccess` です
秘密/秘密を確認	秘密	
タイムアウト	番号	数秒で通信はタイムアウトします。デフォルト値: 2
再試行	番号	接続が失敗する前の再試行の回数
TACACS+ バージョン	V0 V1 V0_V1 V1_V0	使用される TACACS バージョン。デフォルトの値は `v1` です
ローカル認可グループへの Shell およびアクセスサービスのユーザーレベル属性を有効化します	TRUE FALSE	
ユーザーレベル 1 - 10	Nodegrid グループ名	

RADIUS

RADIUS は、アプリケーション層で実行するクライアント/サーバプロトコルで、TCP または UDP のいずれかをトランスポートとして使用できます。ポート 1812 で動作し、ユーザーに集中化した認証、許可、およびアカウンティング (AAA) 管理を提供します。

Nodegrid Platform では、さまざまな方法で Radius ユーザーを Nodegrid グループに割り当てることができます。以下のオプションがあります:

- Radius サービスタイプは、認証プロバイダーの設定を使用して、Nodegrid グループに割り当てることができます。
- Radius サーバでは、この属性`Framed-Filter-ID`を使用してユーザーを Nodegrid グループに割り当てることができます。 例: `Framed-Filter-ID = “group_name=<ng-groupname>[,<ng-groupname1>];”`
- `Framed-Filter-ID`の他に、Nodegrid は、認証目的で使用できるベンダー固有属性 (VSA) もサポートしています。Radius サーバで次の 2 つのプロパティを定義する必要があります。 -- VENDOR ZPE 42518 -- ATTRIBUTE ZPE-User-Groups 1 string

Nodegrid Platform が承認する各ユーザーには、ZPE-User-Groups 属性が割り当てられている必要があります。この値は、コンマで区切られた Nodegrid グループ名のリストです。

FreeRadius サーバの設定例。

1. 以下のコンテンツで、[/usr/share/freeradius/dictionary.zpe]ファイルを作成します。

```
VENDOR  ZPE  42518
BEGIN-VENDOR ZPE
    ATTRIBUTE ZPE-User-Groups 1 string
END-VENDOR   ZPE
```

2. 以下のように dictionary.zpe を含む行を追加して、ファイル [/usr/share/freeradius/dictionary] を編集します。ロケーションは一例です。

```
$INCLUDE dictionary.zpe
$INCLUDE dictionary.jradius
```

3. ユーザーのグループを割り当てて、/etc/freeradius/users でユーザーを設定します。属性 [Framed-Filter-ID] (前と同様) や、新しい属性 [ZPE-User-Groups] を定義できます。

> 注: 両方の属性が定義されている場合、[ZPE-User-Groups] が優先されます。

```
rad-edmond       Cleartext-Password := "*****"
        Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Framed-Filter-Id = "group_name=filter-grp1, filter-grp2;",
        ZPE-User-Groups = "vsa-grp1, vsa-grp2",
        Framed-MTU = 1500,
        Framed-Compression = Van-Jacobsen-TCP-IP
```

フィールド	値	説明
ステータス	TRUE FALSE	デフォルト値は**有効**です。 つまり、プロバイダはユーザーの認証に使用されます。
アクセスが拒否された場合フォールバック	有効または無効	デフォルトは、**無効**です。プロバイダを使用できない場合は、この機能を**有効**にすることをお勧めします。
リモートサーバ	IPアドレス	
アカウンティング サーバ	IPアドレス	
秘密 / 秘密の確認	秘密	
タイムアウト	番号	数秒で通信はタイムアウトします。デフォルト値: 2
再試行	番号	接続が失敗する前の再試行の回数
ローカル認可グループへのサービス タイプ属性関連付けを有効化	TRUE FALSE	Nodegrid ローカルグループへの Radius サービス タイプの割り当てを許可
サービスタイプ ログイン	Nodegrid グループ名	
枠組みされたサービスタイプ	Nodegrid グループ名	
サービスタイプ コールバック ログイン	Nodegrid グループ名	
枠組みされたサービスタイプ コールバック ログイン	Nodegrid グループ名	
サービスタイプ アウトバウンド	Nodegrid グループ名	
管理上のサービスタイプ	Nodegrid グループ名	

Kerberos

Kerberos認証とは、*チケット*を使用して、セキュリティで保護されていないネットワークを通じて通信するノードが、セキュリティで保護された方法で相互に ID を証明可能にする、コンピュータネットワーク認証プロトコルです。主にクライアント–のサーバ モデルとして設計され、相互認証を提供します。ユーザーとサーバ両方が互いの ID を確認します。対称キー暗号化に基づいて構築され、信頼できるサード パーティを必要とし、オプションで公開キー暗号化を使用できます。デフォルトで UDP ポート 88 を使用します。

フィールド	値	コメント
ステータス	TRUE FALSE	デフォルト値は**有効**です。つまり、プロバイダはユーザーの認証に使用されます。
アクセスが拒否された場合フォールバック	有効または無効	デフォルトは、**無効**です。プロバイダを使用できない場合は、この機能を**有効**にすることをお勧めします。
リモートサーバ	IPアドレス	
レルムドメイン名	Kerberos レルム名	
ドメイン名	ドメイン名	

RSA SecurID 2 要素認証

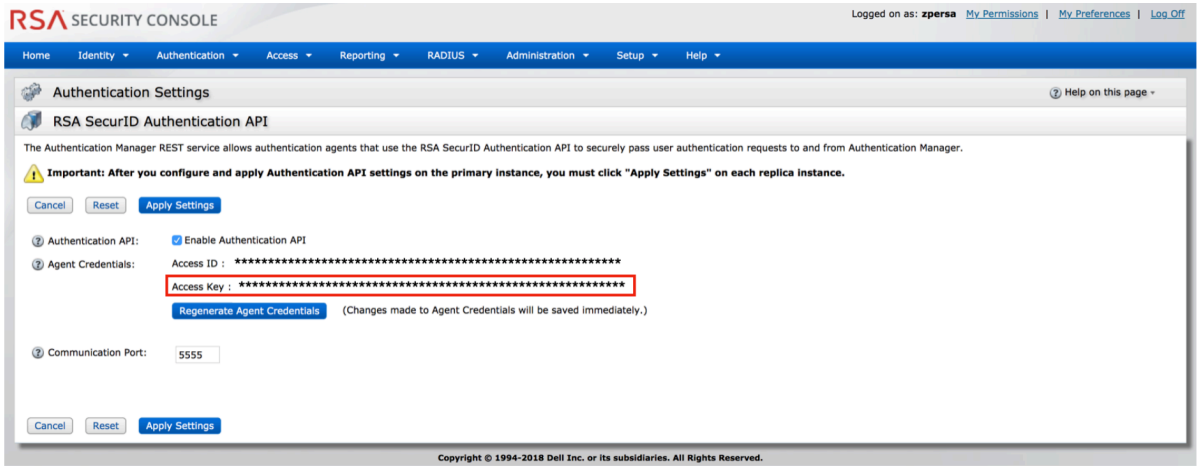
このセクションでは、Nodegrid と RSA セキュリティコンソールで必要な 2 要素認証の設定について取り上げます。

Nodegrid の設定: Web インターフェース

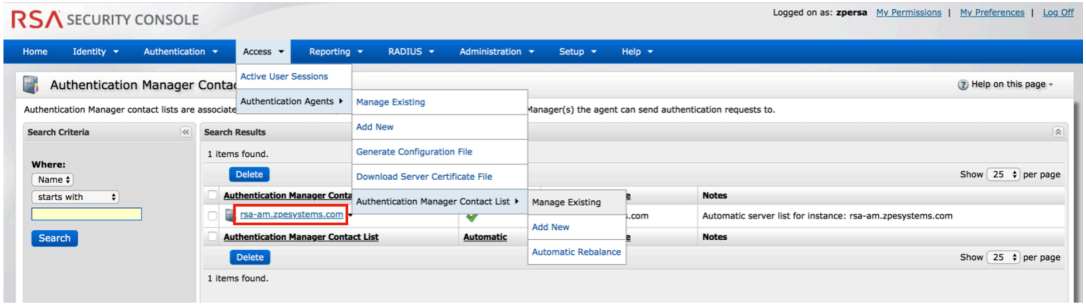
SecurID サーバの追加

- Nodegrid Web インターフェースで管理者としてログインします
- [セキュリティ] アイコンをクリックし、次に [認証] タブをクリックします
- [2 要素] タブをクリックし、次に [追加] ボタンをクリックします
- 以下の例に従い、すべてのフィールドに入力します (以下の注を参照)。
    - 名前: この名前は、ユーザーの SecurID システムを識別します。例 SecurID
    - Rest URL: SecurID 認証 API にアクセスするための URL。https://:5555/mfa/v1_1/authn のフォーマットに従います
    - レプリカを有効化する: サーバにフェールオーバーする Rest Service URL。最大 15 のレプリカを使用できます。1 行に 1 つ。 例: rsa-am-replica2.zpesystems.com:4444, 192.168.2.229:5555

- クライアントキー: RSA セキュリティコンソールから入手できます。SecurID セキュリティコンソールからアクセスキーをコピー/ペーストします。アクセスキーは RSA SecurID 認証 API で入手できます (システム設定の下)

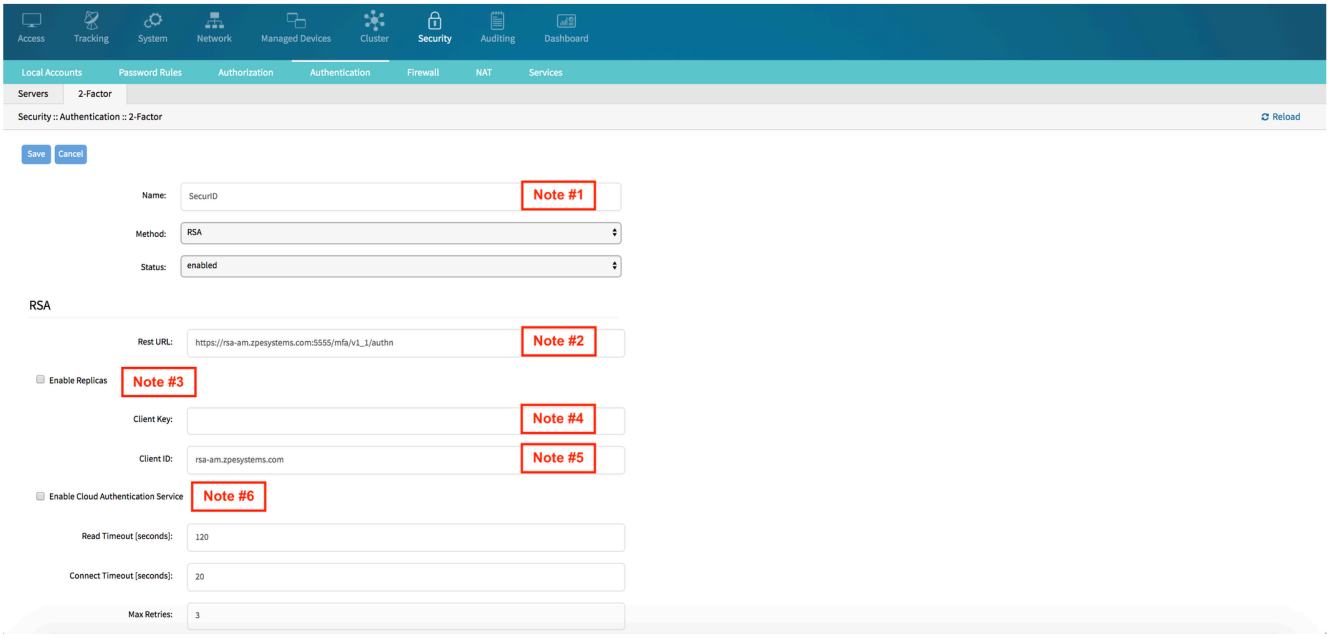


- クライアント ID: 認証マネージャの連絡先リストからサーバノード名を取得します。



- クラウド認証サービスを有効化: 有効にすると、2 つのフィールドが表示されます。サービスが正常に動作するために、これら 2 つのフィールドが必要です。

  - ポリシー ID: クラウド管理コンソールで設定されたアクセスポリシー名。この名前は、クラウド認証サービススーパー管理者から取得します。
  - テナント ID: クラウド管理コンソールで作成されたテナント ID 名。この名前は、クラウド認証サービススーパー管理者から取得します。

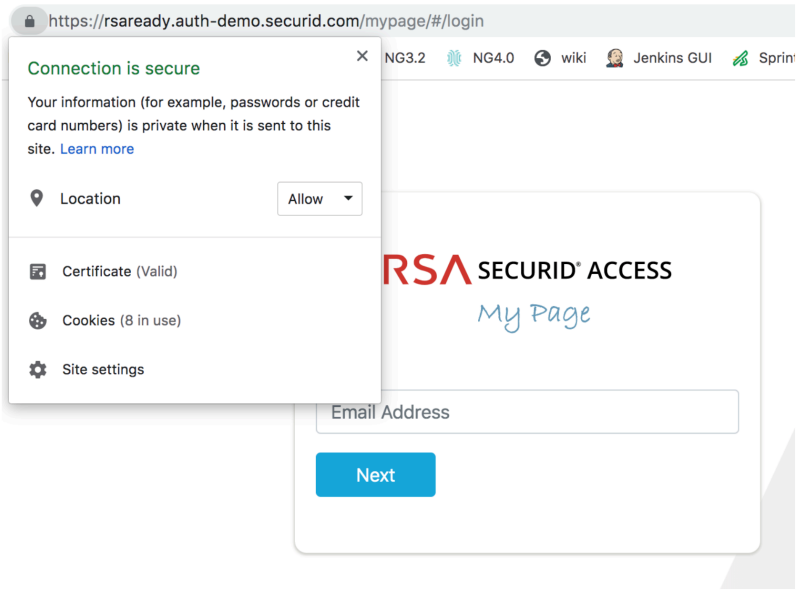


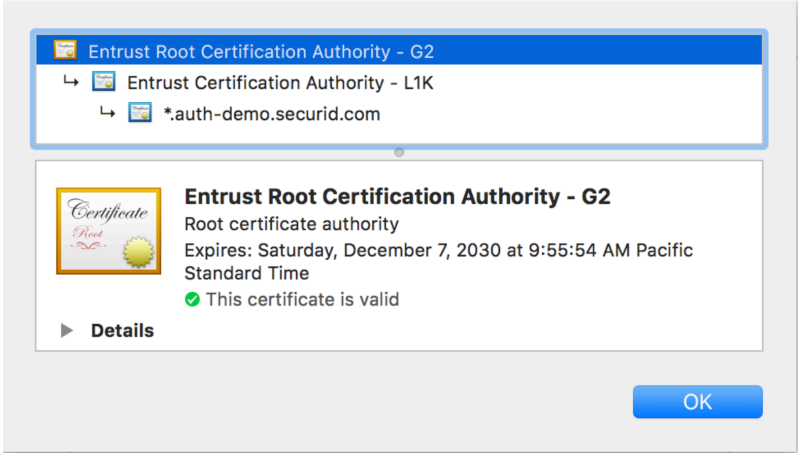
- [保存] をクリックします

SecurID サーバにアクセスするための証明書を設定します

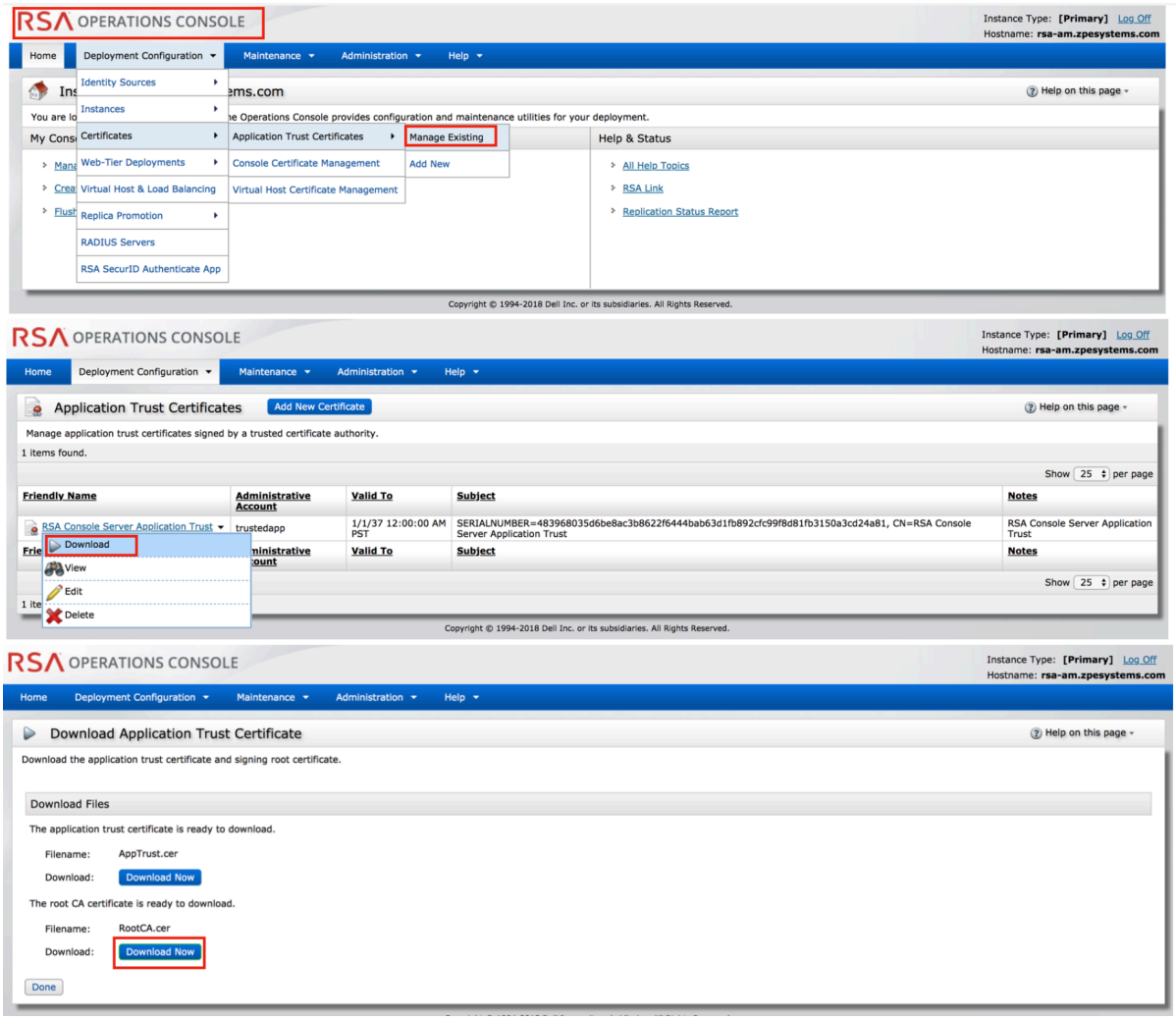
- RSA サーバがクラウド認証を使用する場合 a. RSA SecurID アクセスに移動し、URL の横にある鍵アイコンをクリックします。

  b.`Certificate`をクリックします。このポップアップが表示されます。最初/一番上の証明書をクリックし、デスクトップにドラッグしてコピーします。コピーした証明書はお使いのワークステーションで使用でき、Nodegrid に直接アップロードできます。Nodegrid は、それを予期した証明書形式に自動で変換します。





- クラウドを使用していない場合は、**RSA オペレーションコンソール**から署名ルート証明書をダウンロードします。

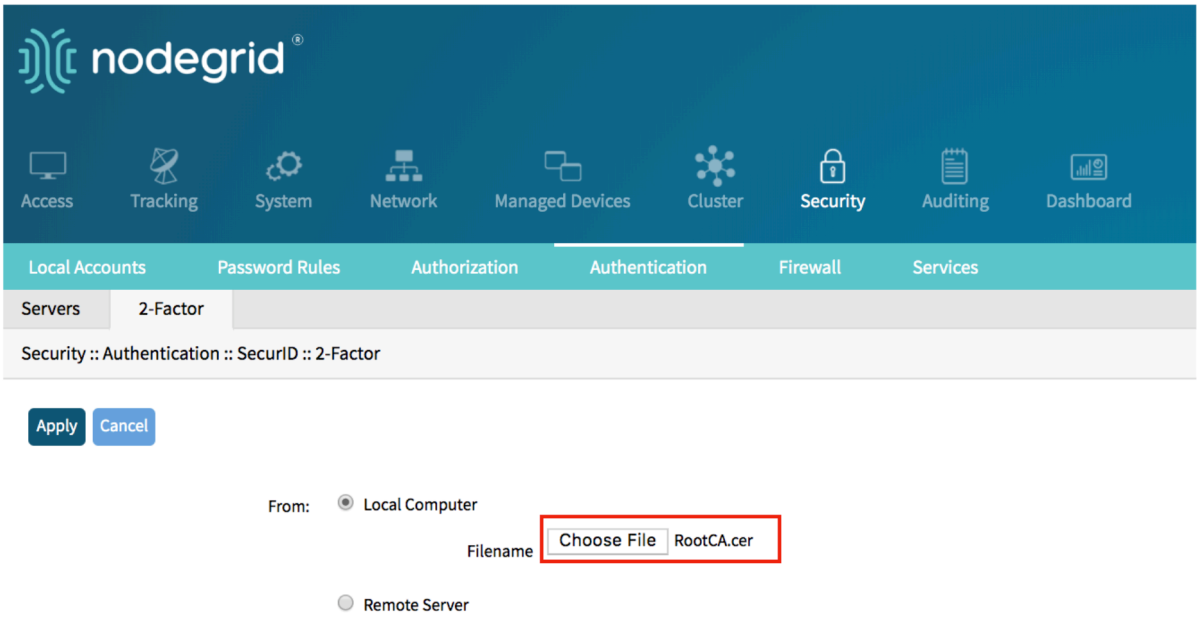


ダウンロードした証明書ファイル (RootCA.cer) はお使いのワークステーションで使用でき、Nodegrid に直接アップロードできます。Nodegrid は、それを予期した証明書形式に自動で変換します。

- 必要に応じて、Nodegrid Web インターフェースで管理者として再度ログインします
- [セキュリティ] アイコンをクリックし、次に [認証] タブをクリックします
- [2 要素] タブをクリックし、上記の手順で追加された SecurID サーバを表すリンクをクリックしま

す。

- ‘証明書’ ボタンをクリックし、‘ローカルコンピュータ’ オプションを選択し、‘ファイルを選択’ をクリックします
- ワークステーションファイルシステムを参照して、ダウンロードした証明書ファイル (つまり、RootCA.cer ファイル) を見つけます。インポートするには、‘適用’をクリックします

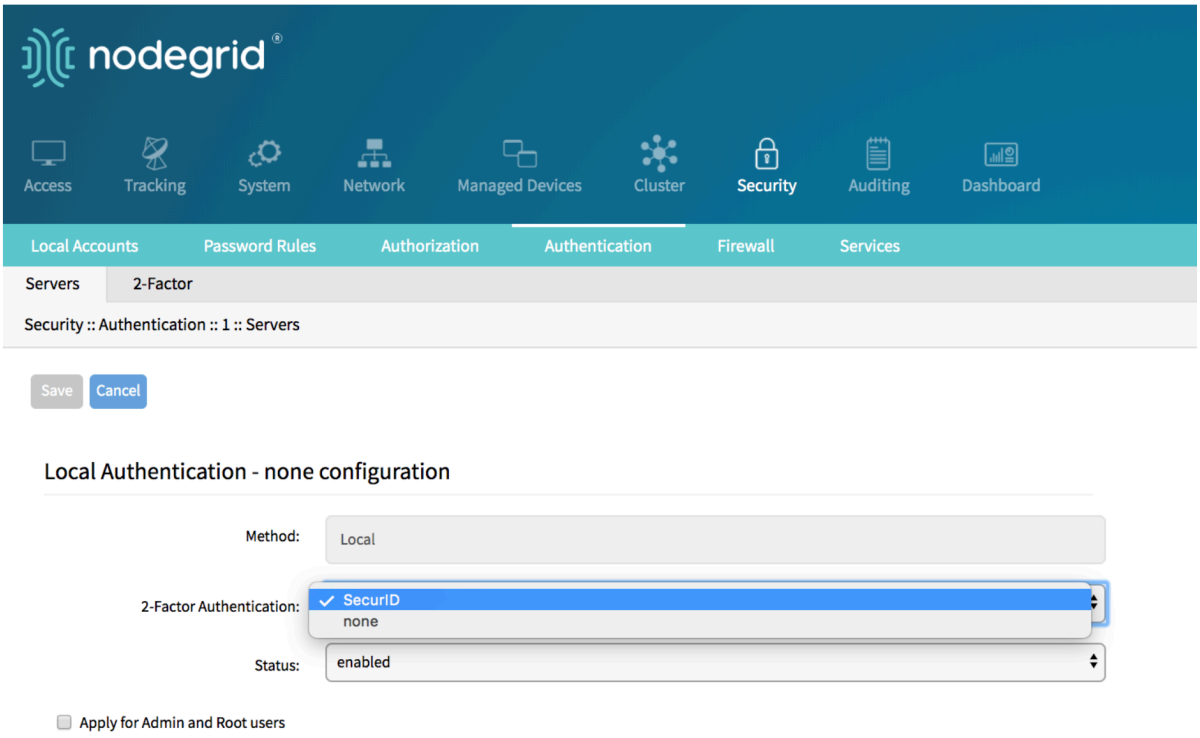


2 要素認証の認証方法への割り当て

RSA SecurID 2 要素認証は、Nodegrid でサポートされている認証方法のどれにでも追加できます: ローカル、LDAP/AD、Radius、Tacacs、またはKerberos。

Nodegridは、認証サーバ’の順序に従ってユーザーを認証します。.この方法が成功すると (つまりユーザーが認証されると)、このような方法にこのような設定がある場合、Nodegrid は 2 要素認証を開始します。

次にユーザーは、RSA SecurID から直接要求を受け取り、このユーザーのために RSA セキュリティコンソールで設定されたトークンコードと PIN を提供します。このプロセスは、Web ブラウザ、SSH、Telnet、またはコンソールポートを介してログインするユーザーに適用されます。



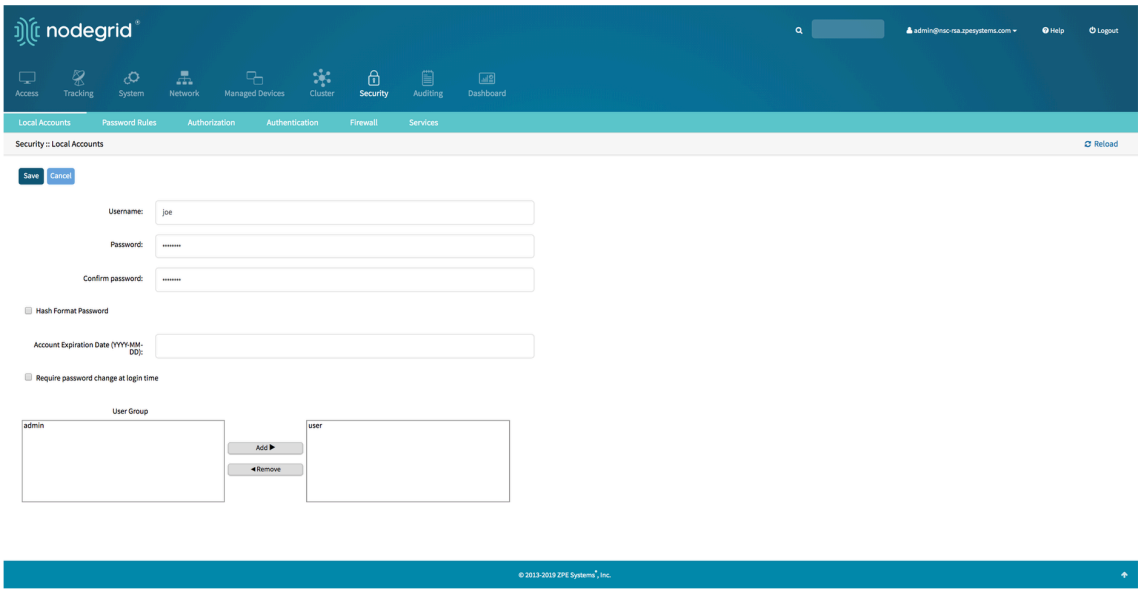
ローカル認証方法では、2 要素認証を適用またはスキップできることにご注意ください。これにより、ローカル Nodegrid 管理者は、RSA セキュリティコンソールでカウンターパートユーザーを設定することなくログインできます。

ユーザー

2 要素認証が有効になると、ユーザーはアクセス権を得るために資格情報とパスコードを提供する必要があります。つまり、ログインを許可されたユーザーは、RSA セキュリティコンソールでも設定される必要があります。

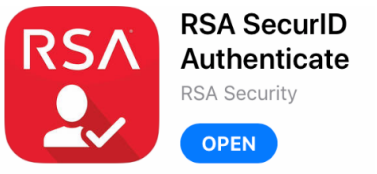
Nodegrid のローカルアカウントでのユーザーの設定方法:

- Nodegrid Web インターフェースで管理者として再度ログインします。
- [セキュリティ] アイコンをクリックし、'ローカルアカウント' タブをクリックし、'追加'をクリックします。
- ユーザー　の名前とパスワードを入力し、'保存'をクリックします



同じユーザーを RSA SecurID で設定し、トークンを割り当てる必要があります。

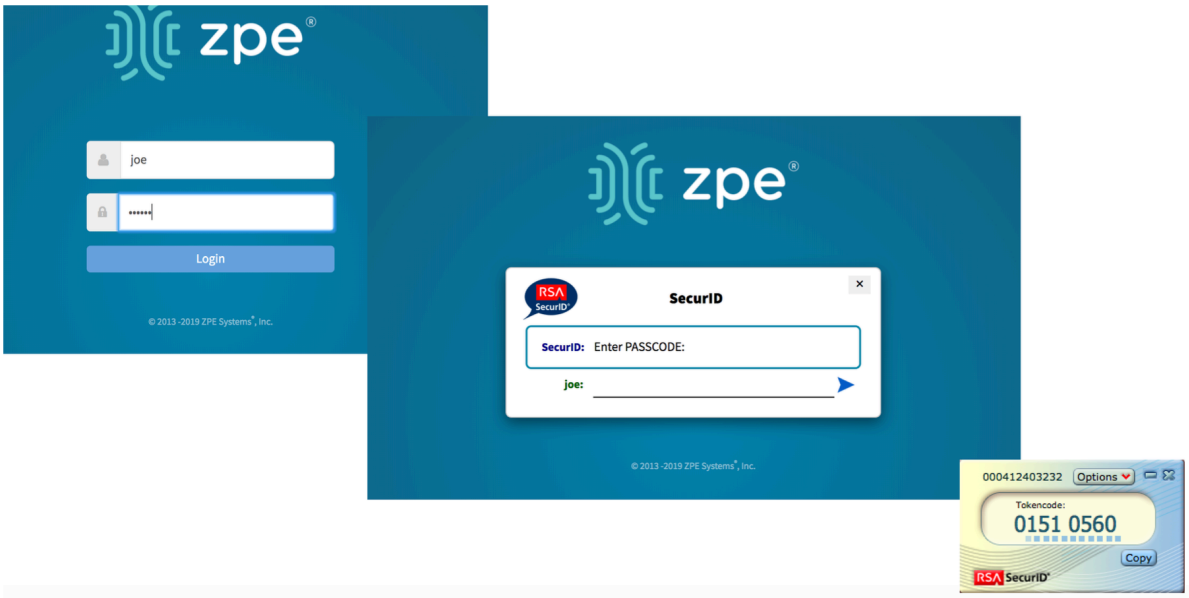
認証アプリ (クラウド認証サービス専用)



- アプリのダウンロード: RSA SecurID 認証
- ワークステーションで、RSA SecurID アクセスに移動してログインします。次に、手順に従ってデバイスを登録します。

ログイン

Nodegrid にログインするには、まずユーザーの資格情報を提供します。ログインプロセスで、2 要素認証が必要な場合は常に、SecurID によって直接要求された情報の入力が求められます。



SSHKey 認証

Nodegrid Platform では、ユーザーは ssh キーを認証に使用できます。この機能は主に、オートメーションシステムでパスワードを入力することなくユニットに安全にアクセスできるように設計されているため、Shell に直接アクセスすることで動作するように設計されており、sshキーを使用する各ユーザーには、ローカルホームディレクトリが必要です。すべてのローカル、LDAP、ADおよび Tacacs+ ユーザーがこの機能を利用できます。

> 注: Radius ユーザーは認証に ssh キーを使用できません。

ユーザーの ssh キー認証を設定するには、次の手順に従う必要があります。

- 移動先 `Security :: Authorization`
- グループを作成するか、既存のグループを使用します
- グループ `Profile` オプションに移動します。
- この `Startup application` 値を**Shell**に設定することで、このグループに属するすべてのユーザーは、SSH 経由の接続での CLI アクセスではなく、デフォルトで shell へのアクセスを取得します
- 移動先 `Security :: Local Accounts`
- ローカル ユーザーを作成し、新しく作成したグループにユーザーを追加します
- これでユーザーは、デフォルトの ssh ツールを使用して、ssh キーを Nodegrid などにコピーできるようになりました `ssh-copy-id`
- この時点以降、ユーザーは認証に ssh キーを使用できるようになります。

オプションの手順

- ユーザーがデフォルトで Shell アクセスではなく、CLI アクセスを必要とする場合、この時点で新しく作成されたグループからこのユーザーを削除することができます。
- LDAP、AD、TACACS+ などの外部認証プロバイダによってユーザーが認証される必要がある場合は、ローカルユーザーアカウントをロックできます。
    - 移動先 `Security :: Local Accounts`

- ユーザーを強調表示し、以下をクリックします `Lock` ユーザーは引き続き sskhey を認証に使用できますが、外部認証プロバイダを使用したグループの権限に基づき、その許可が適用されます。

セキュリティ

ファイアウォール

Nodegrid は、管理者によって設定された場合、ファイアウォールとして機能します。内蔵されたデフォルトのチェーンが、IPv4 用に 3 つ、IPv6 用に 3 つ、合計 6 つあります。これらは、出力、入力、転送パケットを受け入れます。必要に応じて、追加のユーザー チェーンを作成、および削除できます。チェーンごとに、デフォルトのポリシーを設定できます。デフォルトのポリシーは`Accept`パッケージに設定されます。デフォルトのチェーンは削除できません。

チェーン名をクリックして、チェーンごとにルールを作成できます。これにより、チェーンに属するすべての既存のルールが一覧表示されます。ルールは、作成、削除、変更が可能です。ルールには次の設定があります。詳細については、iptables の文書を参照してください。

設定	値	説明
ターゲット	受理 ドロップ 拒否 ログ リターン	
送信元IP/マスク	IP アドレスとマスク	
送信元IP/マスク用マッチを反転する	TRUE FALSE	
送信先IP/マスク	IP アドレスとマスク	
送信先IP/マスク用マッチを反転する	TRUE FALSE	
入力インタフェース	任意の 使用可能なインターフェース	リストの値を 1 つ選択できます。
入力インタフェース用マッチを反転する	TRUE FALSE	
出力インタフェース	任意の 使用可能なインターフェース	リストの値1 つを選択できます。
出力インタフェース用マッチを反転する	TRUE FALSE	
	新規 確立された	1 つ以上の状態を選択

状態マッチを有効化する	関連する 無効	できます
状態マッチを反転する	TRUE FALSE	
フラグメント	すべてのパケットとフラグメント フラグメント化されていないパケットと最初のパケット 2 番目以降のパケット	リストの値1 つを選択できます。
共に拒否するもの	ネットワーク到達不能 ホスト到達不能 ポート到達不能 プロトコル到達不能 ネットワーク禁止 ホスト禁止 管理禁止 TCP リセット	
プロトコル	数値 TCP UDP ICMP	
プロトコル - 数値 - プロトコル番号	プロトコル番号	
プロトコル - TCP - ソースポート	ポート番号	
プロトコル - TCP - 宛先ポート	ポート番号	
プロトコル - TCP - TCP フラグ SYN	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ ACK	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ FIN	任意の 設定 設定解除	
	任意の	

プロトコル - TCP - TCP フラグ RST	設定 設定解除	
プロトコル - TCP - TCP フラグ URG	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ PSH	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグの逆一致	TRUE FALSE	
プロトコル - UDP - ソースポート	ポート番号	
プロトコル - UDP - 宛先ポート	ポート番号	
プロトコル - ICMP - ICMP タイプ	任意の エコー応答 宛先到達不能 ネットワーク到達不能 ホスト到達不能 プロトコル到達不能 ポート到達不能 断片化が必要 ソースルートに失敗しました ネットワーク不明 ホスト不明 ネットワーク禁止 TOS ネットワーク到達不能 TOS ホスト到達不能 通信禁止 ホスト優先違反 優先カットオフ ソースクエンチ リダイレクト ネットワーク リダイレクト ホストリダイレクト TOS ネットワーク リダイレクト TOS ホストリダイレクト エコーリクエスト ルータ広告	

	ルータ要請 時間超過 トランジット中の TTL ゼロ 再アセンブリ中の TTL ゼロ パラメータの問題 不正な IP ヘッダー 必要なオプションの欠落 タイムスタンプリクエスト タイムスタンプ応答 アドレスマスクのリクエスト アドレスマスク応答	
プロトコル - ICMP - ICMP タイプの逆一致	TRUE FALSE	
プロトコル用マッチを反転する	TRUE FALSE	
送信元ポート用マッチを反転する	TRUE FALSE	
送信先ポート用マッチを反転する	TRUE FALSE	
ログ レベル	デバグ 情報 通知 警告 エラー クリティカル アラート 緊急	
ログ プレフィックス	ログプレフィックス文字列	
ログTCPシーケンス番号	TRUE FALSE	
TCPパケット番号からのログ オプション	TRUE FALSE	
IPパケット番号からのログ オプション	TRUE FALSE	

NAT

Nodegrid は、管理者によって設定された場合、ファイアウォールとして機能します。NAT セクションでは、NAT テーブルのルールを定義でき、ネットワーク アドレス変換ルール (NAT) の定義に使用できます。内蔵されたデフォルトのチェーンが、IPv4 用に 4 つ、IPv6 用に 4 つ、合計 8 つあります。これらは、プレルーティング、出力、入力、ポストルーティングパケットを受け入れます。デフォルトのチェーンは削除できません。

チェーン名をクリックして、チェーンごとにルールを作成できます。これにより、チェーンに属するすべての既存のルールが一覧表示されます。ルールは、作成、削除、変更が可能です。ルールには次の設定があります。詳細については、iptables の文書を参照してください。

設定	値	説明
ターゲット	受理 ドロップ 拒否 ログ リターン	
送信元IP/マスク	IP アドレスとマスク	
送信元IP/マスク用マッチを反転する	TRUE FALSE	
送信先IP/マスク	IP アドレスとマスク	
送信先IP/マスク用マッチを反転する	TRUE FALSE	
入力インタフェース	任意の 使用可能なインターフェース	リストの値1 つを選択できます。
入力インタフェース用マッチを反転する	TRUE FALSE	
出力インタフェース	任意の 使用可能なインターフェース	リストの値1 つを選択できます。
出力インタフェース用マッチを反転する	TRUE FALSE	
状態マッチを有効化する	新規 確立された 関連する 無効	1 つ以上の状態を選択できます
状態マッチを反転する	TRUE FALSE	

フラグメント	すべてのパケットとフラグメント フラグメント化されていないパケットと最初のパケット 2 番目以降のパケット	リストの値1 つを選択できます。
共に拒否するもの	ネットワーク到達不能 ホスト到達不能 ポート到達不能 プロトコル到達不能 ネットワーク禁止 ホスト禁止 管理禁止 TCP リセット	
プロトコル	数値 TCP UDP ICMP	
プロトコル - 数値 - プロトコル番号	プロトコル番号	
プロトコル - TCP - ソースポート	ポート番号	
プロトコル - TCP - 宛先ポート	ポート番号	
プロトコル - TCP - TCP フラグ SYN	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ ACK	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ FIN	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ RST	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグ URG	任意の 設定 設定解除	

プロトコル - TCP - TCP フラグ PSH	任意の 設定 設定解除	
プロトコル - TCP - TCP フラグの逆一致	TRUE FALSE	
プロトコル - UDP - ソースポート	ポート番号	
プロトコル - UDP - 宛先ポート	ポート番号	
プロトコル - ICMP - ICMP タイプ	任意の エコー応答 宛先到達不能 ネットワーク到達不能 ホスト到達不能 プロトコル到達不能 ポート到達不能 断片化が必要 ソースルートに失敗しました ネットワーク不明 ホスト不明 ネットワーク禁止 TOS ネットワーク到達不能 TOS ホスト到達不能 通信禁止 ホスト優先違反 優先カットオフ ソースクエンチ リダイレクト ネットワーク リダイレクト ホストリダイレクト TOS ネットワーク リダイレクト TOS ホストリダイレクト エコーリクエスト ルータ広告 ルータ要請 時間超過 トランジット中の TTL ゼロ 再アセンブリ中の TTL ゼロ パラメータの問題 不正な IP ヘッダー 必要なオプションの欠落	

	タイムスタンプリクエスト タイムスタンプ応答 アドレスマスクのリクエスト アドレスマスク応答	
プロトコル - ICMP - ICMP タイプの逆一致	TRUE FALSE	
プロトコル用マッチを反転する	TRUE FALSE	
送信元ポート用マッチを反転する	TRUE FALSE	
送信先ポート用マッチを反転する	TRUE FALSE	
ログ レベル	デバグ 情報 通知 警告 エラー クリティカル アラート 緊急	
ログ プレフィックス	ログプレフィックス文字列	
ログTCPシーケンス番号	TRUE FALSE	
TCPパケット番号からのログ オプション	TRUE FALSE	
IPパケット番号からのログ オプション	TRUE FALSE	

サービス

サービスページでは、システム上で実行する`Active Services`、システム自体への`ZPE Cloud`、`Managed Devices`、`Intrusion Prevention`、`SSH`用の一般的なサービス設定、`Web Service`設定、および Web サービス用の`Cryptographic Protocols`を定義できます。

システムのセキュリティーレベルは、これにより設定できます。たとえば、Telnet や HTTP などのセキュリティで保護されていないプロトコルや、システムへのアクセスが許可されている SSH バージョンを無効にできます。

ZPE Cloud

ZPE Cloud は、Nodegrid 製品用のクラウドベースの管理プラットフォームです。あらかじめ設定されたデバイスをブランチに発送する必要はありません。ZPE Cloud は、初期展開、設定、継続的な管理を単純化し、分かりやすく操作が簡単な豊富な分析結果と共に、展開全体に 360°の可視性を提供します。

ZPE Cloud と Nodegrid デバイスを組み合わせることで、ステージ設定や事前設定なしで IT デバイスを出荷できます。IT デバイスは、安全対策のブランチで使用される場合に限り設定します。NOC の安全性から ZPE Cloud を介した均質な自動プロビジョニングを展開します。

ZPE Cloud は、すべての Nodegrid 製品をひとつのプラットフォームにまとめます。すべての Nodegrid 製品で使用できる“リセット”ボタンを使用して、お使いの Nodegrid を ZPE Cloud に再接続します。

この`ZPE Cloud`セクションで、ユニット上のクラウドサービスを設定できます。以下の設定を使用できます:

設定	値	説明
ZPEクラウドを有効化する	TRUE FALSE	デフォルトで、Nodegrid SR ファミリー (NSR、GSR、BSR、および LSR) は有効化されています 注: Nodegrid Serial Console のデフォルト値は無効です。
ファイル保護を有効化する	TRUE FALSE	デフォルトで無効です。 有効にすると、ファイルの整合性と発生元を検証するために、このパスワードに基づく認証ハッシュが必要になります。
ファイル暗号化を有効化する	TRUE FALSE	デフォルトで無効です。 有効にすると、ファイル保護の下で定義されたパスワードを使用して、その所有者が ZIP 経由でファイルを暗号化する必要があります。

:警告: **Nodegrid v4.1**より前に出荷された Nodegrid ユニットの場合、ZPE Cloud にユニットを登録するには、`root`で次のコマンドを実行する必要があります。

zpe_cloud_enroll の使用

スクリプトは、以下に示すように 3 つの引数の組み合わせで呼び出せます:

```
root@ZPECloudNSR2:~# zpe_cloud_enroll -h
Usage: zpe_cloud_enroll [options]
ZPE Cloud Enrollment

Options:
  -v, --version         Displays version information.
  -h, --help            Displays this help.
  -c <customer-code>    ZPE Cloud customer code to enroll device.
  -k <enrollment-key>   ZPE Cloud customer enrollment key.
  -r                    Read customer enrollment key from barcode.
```

引数なし

引数が指定されていない場合、デバイスは以下と `customer code` `enrollment key` の入力を要求します:

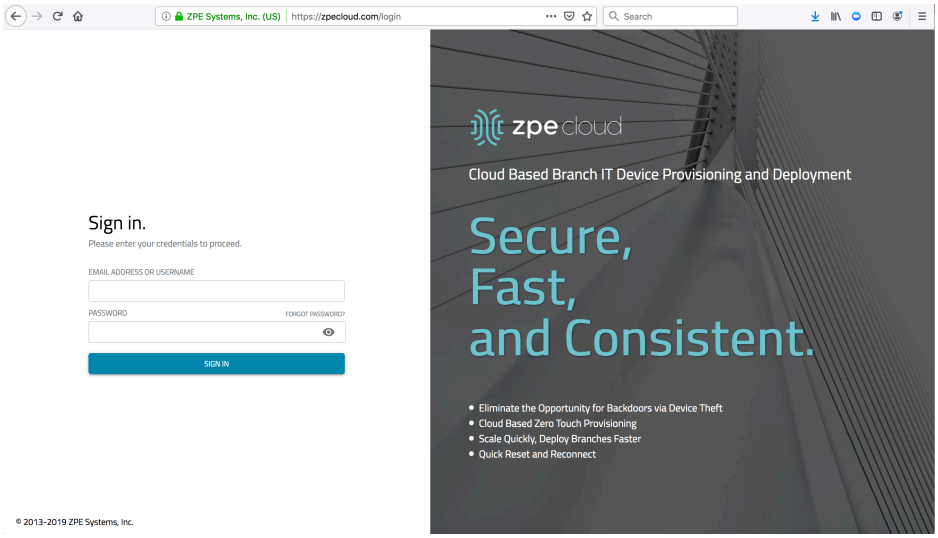
```
root@ZPECloudNSR2:~# zpe_cloud_enroll
Enter your customer code: 2
Customer Code:  "2"
Enter your enrollment key: example_key
```

引数 (顧客コードと登録キー)

この場合、顧客コード (-c) および登録キー (-k) は、以下のように提供されます スクリプト引数:

```
zpe_cloud_enroll -c 2 -k example_key
```

ユニットで ZPE Cloud を有効にすると、すべての登録デバイスを管理する www.zpecloud.com にアクセスできるようになります。クラウド管理ポータルでは、会社登録と管理者ユーザー アカウントが必要です。



アクティブ サービス

この`Active Services`ページでは、システムで有効化されるサービスと、使用するネットワーク ポートを制御できます。

以下の設定を使用できます:

設定	値	説明
USBデバイスの検出を有効化	TRUE FALSE	デフォルトで有効
RPCを有効化	TRUE FALSE	NFS 共有アクセスに必要
gRPCを有効化	TRUE FALSE	gRPC プロトコルのサポートデフォルトで無効
FTPサービスを有効化	TRUE FALSE	
SNMPサービスを有効化	TRUE FALSE	デフォルトで有効
Nodegrid への Telnet サービスを有効化	TRUE FALSE	
Telnet TCPポート	ポート番号	デフォルト値: 23
管理対象デバイスへの Telnetサービスを有効化	TRUE FALSE	
ICMPエコー応答を有効化	TRUE FALSE	デフォルトで有効
IP上でUSBを有効化	TRUE FALSE	
仮想化サービスを有効化	TRUE FALSE	NFV または Docker アプリを実行するには、有効化する必要があります。どちらの機能もライセンスが必要です
TCPポートをクラスタする	ポート番号	デフォルト値: 9966
自動クラスタ登録を有効化	TRUE FALSE	
	ポー	

検索エンジン TCP ポート	ト番号	デフォルト値: 9300
検索エンジン高レベル暗号スイートを有効化	TRUE FALSE	
VMシリアルアクセスを有効化	TRUE FALSE	デフォルトで有効
VMシリアルポート	ポート番号	デフォルト値: 9977
vMotionタイムアウト[秒]	秒単位の数値	デフォルト値: 300
ゼロ タッチ プロビジョニングを有効化	TRUE FALSE	デフォルトで有効
PXE (Preboot eXecution Environment) を有効化	TRUE FALSE	デフォルトで有効

管理対象デバイス

この`Managed Devices`セクションでは、一般的な側面の制御と、管理対象デバイスを制御するサービスを使用できます。 以下の設定が可能です。

設定	値	説明
ユーザーグループ認可で強制されたデバイスアクセス	TRUE FALSE	この機能を有効にすると、ユーザーは、ユーザーが属する認証グループの下にリストされているデバイスにのみアクセスできるようになります。この機能が有効になっていないと、ユーザーは Nodegrid に登録されているすべてのデバイスを使用でき、制限なくアクセスできます。
自動検出を有効化する	TRUE FALSE	この機能により、ネットワーク上の管理対象デバイスの自動検出が可能になります。
自動検出ルールで制御されるDHCP リース	TRUE FALSE	この機能を有効にすると、DHCP サーバは、自動検出プロセスを通じて検出されたデバイスにのみサーバリースを行います。この機能は、`Enable AutoDiscovery`が有効になっている場合にのみ使用できます。

侵入防止

この`Intrusion Prevention`セクションでは、 `Fail 2 Ban`や`Rescue Mode`のような、システムへの不正アクセスを防止するメカニズムの設定が可能です。 以下の設定が可能です。

設定	値	説明
複数認証に失敗したホストをブロック	TRUE FALSE	
期間ホストはブロックされたままになります(分)	分数	システムがネットワーク上で到達できない時間。デフォルト値:10
認証の失敗をモニターするタイムフレーム(分)	分数	失敗した認証試行がカウントされ、カウンタがリセットされるまでの時間。デフォルト値:10
ホストをブロックする認証の失敗の数	番号	ホストがブロックされる`Number of authentication fails to block host`までに失敗した認証試行の回数。 デフォルト値:5
レスキューモードは認証が必要です	TRUE FALSE	この機能を有効にすると、レスキュー モードでは root などのローカルユーザーアカウントを介した認証が必要になります。
パスワード保護されたブート*	TRUE FALSE	この機能を有効にすると、BIOS と Grub の編集に、ここで定義したパスワードに基づく認証が必要になります。

> **パスワード保護ブート**は、特許出願中の機能で、Nodegrid OS が BIOS と通信し、BIOS パスワードが不正に変更されるの防ぎます。同じパスワードで、不正な変更から Grub も保護できます。

SSH

この`SSH`セクションでは、Nodegrid システムへのアクセスを制御する SSH サービスの設定が行えます。以下の設定が可能です。

SSHv1 の明示指定はなくなります。現在は SSHv2 のみをサポートしています。

設定	値	説明
SSHがルートアクセスを許可します	TRUE FALSE	SSH を介したルートアクセスを許可。デフォルトで有効。
SSH TCPポート	ポート番号	デフォルト値: 22
SSHサイファー	暗号の許可リスト	デフォルト値: 空白、 Nodegrid でサポートされているすべての暗号を許可
SSH MACs	MAC アドレスの許可リスト	デフォルト値: ブランク、 全てのシステムの ssh を介した Nodegrid へのアクセスを許可
SSH KexAlgorithms	主要な交換アルゴリズムの許可リスト	デフォルト値: 空白

ウェブサービス

この`Web Service`セクションでは、Web サーバの設定が行えます。以下の設定が可能です。

設定	値	説明
HTTPアクセスを有効化する	TRUE FALSE	デフォルト値: 有効
HTTPポート	ポート番号	デフォルト値: 80
HTTPSアクセスを有効化する	TRUE FALSE	デフォルト値: 有効
HTTPSポート	ポート番号	デフォルト値: 443
HTTPをHTTPSにリダイレクトする	TRUE FALSE	デフォルト値: 有効

暗号プロトコル

`Cryptographic Protocols`では、Web サーバにアクセスするためにサポートされている暗号の設定が可能です。以下の設定が可能です。

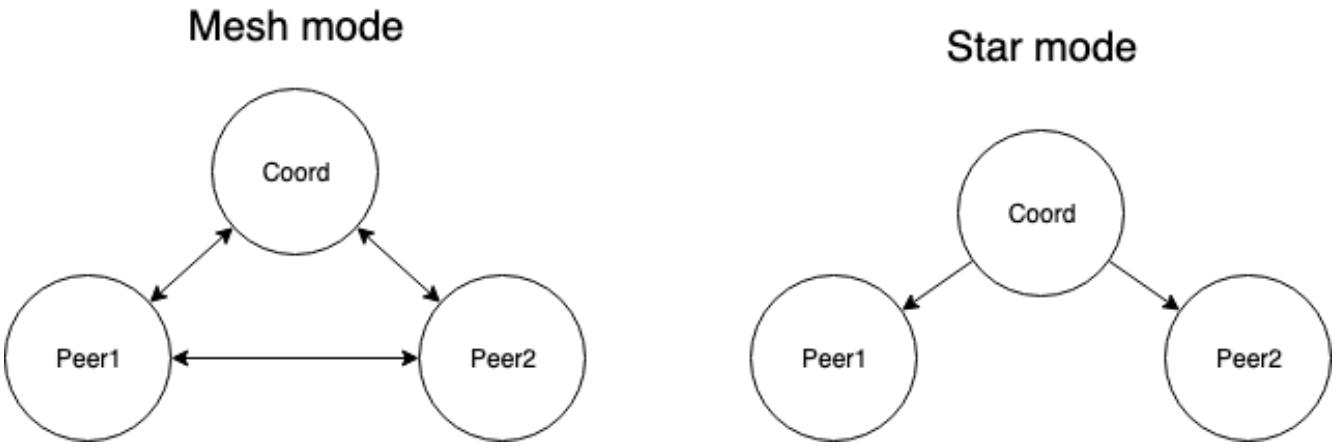
設定	値	説明
TLSv1.3	TRUE FALSE	デフォルト値: 有効
TLSv1.2	TRUE FALSE	デフォルト値: 有効
TLSv1.1	TRUE FALSE	デフォルト値: 有効
TLSv1	TRUE FALSE	デフォルト値: 無効
暗号スイートレベル	高 中 低 カスタム	デフォルト値: 中

クラスタ

クラスタは、他の Nodegrid プラットフォーム間で安全で復元力のある接続を確立する Nodegrid の機能です。クラスタリングを有効にすることで、複数の Nodegrid システムが、他のノードからすべての管理対象デバイスを簡単に管理・アクセスできるようになります。 Nodegrid は、クラスタ資産の検索によって、クラスタのアクセス管理を一層容易にします。任意の Nodegrid ノードにログインすることで、ユーザーは、Nodegrid が管理するすべての企業ネットワークとクラスタを単一のインターフェースで検索できます。これにより、垂直・水平両方向のスケーラビリティが可能になります。

2 種類のクラスタリングトポロジー:

- **STAR** - デフォルトのオプション。STAR 設定では、Nodegrid の 1 ユニットが、コーディネーターとセントラルノードとして機能します。他のすべてのピアは、スターフォーメーションでコーディネーターに接続します。コーディネータのみが、設定上のすべてのピアと接続されたデバイスのリストを表示できます。このオプションにより、ピアとして機能する Nodegrid ユニットへのシステムリソースの需要を軽減させながら、コーディネーターとして機能する Nodegrid ユニットからの集中アクセスと可視化を可能にします。
- **MESH** - MESH 設定では、Nodegrid の 1 ユニットが、コーディネーターとして機能し、すべての Nodegrid ユニット (コーディネーターとピア) は、クラスタ内のすべての接続されたデバイスを相互に確認することが可能になります。このオプションでは分散アクセスが可能になるため、すべてのユニットがピアと接続デバイスの全リストを保持する必要があり、クラスタリング上のすべてのユニットに同等のシステムリソースを要求します。この設定は、50 ユニット未満のクラスタリングにお勧めします。



ピアの概要

この `Peers` ページには、クラスタに登録されているすべての Nodegrid ユニットが一覧表示されます。

表は、各 Nodegrid の名前、IP アドレス、タイプ、および他のピアとの通信状況を示します。

ピアは、エントリを選択し、`Remove` ボタンをクリックすることで削除できます。 Nodegrid がコーディネーターの場合、表から削除することはできません。

クラスタ設定

このセクションでは、クラスタ機能と追加サービスの`Peer Management`と`License Pool`を有効化して設定することが可能です。

> 注: クラスタ機能には、クラスタ内の各ノード用のソフトウェアライセンスが必要です。

クラスタを有効化

クラスタ機能は、チェックボックス`Enable Cluster`をクリックすることで有効にできます。 各クラスタには、ピアシステムの登録を調整・制御する`Coordinator`が 1 つ必要です。

クラスタを`Type`コーディネーターとして設定する必要がある最初のユニット。その後、他の全てのユニットは、*ピア*のタイプに設定できます。`Coordinator`の役割は、ピアのコーディネーターのタイプを選択することで、後で別のシステムに変更できます。その後変更はシステムに自動で反映されます。

Nodegrid がコーディネーターである場合は、チェックボックス`Allow Enrollment`がオンであることを確認し、`Cluster Name`と`Pre-Shared Key`を提供して、ピアをクラスタに登録できるようにします。 また、Star か Meshとして`Cluster Mode`を選択して、必要なクラスタリングのタイプを設定します。

`Cluster Name`と`Pre-Shared Key`は、ピア'の設定で使用されるのでご注意ください。

Nodegrid がピアの場合は、コーディネータ'の`Cluster Name`、`Coordinators's Address`、および`Pre-Shared Key`を入力します。

他の Nodegrid システムが、他のノードからのすべての管理対象デバイスを管理、アクセス、検索できるようにするには、`Enable Clustering`チェックボックスをオンにします。

> 注: **MESH**では、コーディネーターはピアを登録するためにのみ必要になります。すべての Nodegrid システムがクラスタに登録されると、コーディネーターをピアとして設定して、他のユニットの登録を防ぐことができます。

自動登録

この`Automatic Enrollment`機能により、管理者は既存のクラスタで使用できるようになった新しい Nodegrid システムを自動で追加できます。この機能により、デフォルトで`Peers`を検出できます。`Pre-Shared Key`設定は、コーディネーターとピアで同じである必要があります。これはデフォルトで*nodegrid-key*に設定されています。この`Interval [seconds]`値は、コーディネーターにのみ適用され、潜在的なピアへの招待の送信頻度を調整します。 これは、定義されたネットワークリストに基きます。

`Coordinator`を有効にして設定した後、管理者ユーザーは、他の Nodegrid システムがネットワーク上に存在するさまざまな IP を追加できます。検出プロセスのネットワーク範囲を追加するには、クラスタ設定の下の`Automatic Enrollment Range`ページにそれらを追加します。

> 注: システムは、Nodegrid ユニットが特定の Ip上で検出され、それがクラスタに追加されるまで、すべての IP に継続的に招待を送信しますので、潜在的に Nodegrid ユニットである自動登録範囲だけに IP を追加することをお勧めします。

これにより、コーディネーターは、この範囲の任意の Nodegrid システムと通信してクラスタに追加するため、各 Nodegrid ノードに移動してピアとして設定する必要がなくなります。

ライセンスプール

この`License Pool`機能により、クラスタ内のすべてのソフトウェアライセンスの集中管理が可能になります。このために、少なくとも 1 つのユニットを (**STAR**でコーディネーターである必要があります)`License Pool Server`として設定する必要があり、他の全てのユニットは、デフォルトで`License Pool Clients`として設定されます。

ライセンスプールクライアントは、`License Pool Server`から必要なライセンスを自動で要求します。ライセンスプールサーバは、ライセンスの可用性を確認し、要求されたライセンスが利用可能な場合はこれを割り当てます。クライアントは、サーバ`Renew Time [days]`に依存するライセンスを更新します。クライアントが長時間使用できなくなり、サーバ`Lease Time [days]`,を超えた場合、ライセンスはクライアント上で無効となり、プールに戻ります。`Lease Time [days]`.オプションは、7 ～ 30 日の値を受理します。

現在リースされているライセンスは、`System :: Licenses`セクションのライセンスプールサーバで表示できます。

> 注: 各 Nodegrid ユニットには、5 つの追加テストターゲットライセンスが付属します。テストライセンスは、ターゲットライセンスがシステムに追加されると、自動的に付与されます。これは、ターゲットライセンスがライセンスプールサーバを介して適用される場合にも当てはまります。つまり、システムが最初にターゲットライセンスを要求する時、現在使用されているテストライセンスをカバーするために 5 つの追加ライセンスを要求するという意味です。

ピア管理

この`Peer Management`機能により、クラスタ内の Nodegrid ユニットのファームウェアを一元的にアップグレードすることが可能です。この機能を有効にするには、`Enable Peer Management`を選択します。

クラスタ`Management`ページでは、リモート Nodegrid ユニットのソフトウェアアップグレードプロセスをセントラルロケーションから開始できます。ユニットに適用されるファームウェアは、URL を介して利用可能なセントラルロケーションでホストする必要があります。

> 注: URL には、リモートサーバの IP またはホスト名、ファイルパス、および ISO ファイルを含める必要があります。 例:
> `ftp://192.168.2.200/nodegrid/Nodegrid_Platform_v3.1.0_20160127.iso`

このページには、クラスタ内のすべての Nodegrid システムが一覧表示されます。管理ステータスが*アイドル*の必要なノードを選択します。 ステータスが*無効*と表示されている場合は、Nodegridの`Peer Management`機能が*無効*になっていることを意味します。 選択したら、*ソフトウェアのアップグレード*ボタンをクリックします。`Remote Server`を選択して、`URL`、`Username`、`Password`を入力します。 このオプション`Format partitions before upgrade`は、ファームウェアのアップグレードを実行する前に、Nodegrid ユニットのハードドライブをフォーマットします。

ソフトウェアをダウングレードする場合は、`Restore configuration saved on version upgrade`か`Apply factory default configuration`を選択できます。

監査設定

監査機能を使用すると、作成されたイベントを以下の 4 つの異なる宛先に送信できます: Eメール、ファイル、SNMP Trap、Syslog。また、データログとイベントログを、ローカルで格納、NFS を介してリモートで保存、または Syslog サーバに送信できます。

データロギング

データロギング機能を使用すると、Nodegrid システムやターゲットデバイスから送受信されるデータ ストリームをキャプチャできます。データロギング機能の一般的な設定は、`Auditing :: Settings`で行えます。以下の設定が可能です。

設定	値	説明
ファイル送信先を有効化する	TRUE FALSE	この機能を有効にすると、すべてのデータログが、`Auditing Destinations`の下の定義済みのファイルロケーションに保存されます。デフォルト値: 有効
Syslogの送信先を有効化する	TRUE FALSE	この機能を有効にすると、すべてのデータログが、`Auditing Destinations`の下の定義済みの Syslog ロケーションに送信されます。デフォルト値: 無効
ログされたすべてのラインにタイムスタンプを追加する	TRUE FALSE	この機能を有効にすると、各データログ行にタイムスタンプが追加されます。
タイムスタンプ書式	UTC 現地時間	使用するタイムスタンプのタイムゾーンを定義します。デフォルト値: UTC

イベント

Nodegrid システムは、システム設定とデバイス設定に基づいて、イベントを自動作成します。デフォルトで、すべてのイベントがローカルファイルシステムに保存されます。この動作は、`Auditing :: Events`で調整できます。管理者は、記録すべき宛先イベントとイベントカテゴリを設定できます。

システムは、以下の 4 つのイベントカテゴリをサポートしており、個別に管理できます:

- システムイベント
- AAAイベント
- デバイスイベント
- ロギングイベント

> 注: `Tracking :: Event List`には、リストされているすべてのイベントと、それらが属するカテゴリが表示されています。

これらの各イベントカテゴリでは、4 つのイベント宛先のいずれかにイベントを送信する、またはどれにも送信しないように設定できます。イベント宛先:

- ファイル - これは、ローカルファイルストレージか NFS ファイルストレージです
- Syslog - これは、ローカルの Syslog またはリモートストレージです
- SNMPトラップ
- Eメール

送信先

ファイル

データログは、デフォルトでローカルに保存されているファイルに書き込まれます。ファイルの宛先とアーカイブの設定は以下で設定可能です `Auditing :: Destinations :: File`

> 注: NFSは、セキュリティでRPCサービスを有効化する必要があります :: サービス

以下のオプションを使用できます。

設定	値	説明
送信先	ローカル NFS	
NFS - NFS サーバ	NFS サーバの IP アドレス	
NFS - NFS パス	NFS ルートディレクトリへのパス	各ユニットには、独自のルートディレクトリが必要です。
ファイルサイズ [Kbytes]	ファイルサイズ [単位 Kbytes]	ファイルがローテーションされるファイルサイズ。有効な値は0 (無効) ～ 2048 Kbです。デフォルト値: 1024。
アーカイブ数	番号	破棄される前に保存する必要があるアーカイブファイルの数。デフォルト値: 10 最大値: 99
(NFS) 時間ごとにアーカイブ [HH:MM]	時刻形式 HH:MM	ファイルアーカイブがローテーションされる時間。デフォルト値: 空白

Syslog

Syslog 宛先を使用して、データ ログとイベント通知を保存できます。システムは、ローカル Syslog 宛先、またはリモート IPv4 および IPv6 宛先をサポートします。

以下のオプションを使用できます。

設定	値	説明
システムコンソール	TRUE FALSE	Syslog イベントは、Nodegrid システムコンソールポートセッションに表示されます。 デフォルトで、このオプションは有効です。
管理者セッション	TRUE FALSE	Syslog イベントが表示され、Nodegrid システムに公開されている任意の管理者 セッションが表示されます。 デフォルトで、このオプションは無効です。
IPv4リモートサーバ	IPアドレス	1 つ以上の IP アドレスを指定できます。 アドレスはコンマで区切る必要があります。
IPv4アドレスまたはホスト名	TRUE FALSE	デフォルトで無効です
IPv6リモートサーバ	IPアドレス	1 つ以上の IP アドレスを指定できます。 アドレスはコンマで区切る必要があります。
IPv6アドレスまたはホスト名	TRUE FALSE	デフォルトで無効です
イベントファシリティ	Log Local 0 Log Local 1 Log Local 2 Log Local 3 Log Local 4 Log Local 5	イベントの Syslog ロギングファシリティを定義します
データロギングファシリティ	Log Local 0 Log Local 1 Log Local 2 Log Local 3 Log Local 4	データログの Syslog ロギングファシリティを定義します

	Log Local 5	

SNMPトラップ

任意のトリガされたイベントは、SNMP トラップを介して既存の NMS システムに送信できます。Nodegrid システムは、SNMP v2 と v3 トラップをサポートします。Nodegrid システムの MIB ファイルは、ファームウェアファイルと共に使用できます。

MIBファイルの場所:

```
root@nodegrid:~# ls -l /usr/local/mibs/
total 104
-rw-r--r-- 1 root root 36940 Nov 20  2017 NodeGrid-MIB.asn
-rw-r--r-- 1 root root 61403 Nov 20  2017 NodeGrid-TRAP-MIB.asn
-rw-r--r-- 1 root root  2732 Nov 20  2017 ZPESystems.smi
```

> 注: SNMP3 INFORM メッセージは現在サポートされていません。

以下のオプションを使用できます。

設定	値	説明
SNMPエンジンID	なし	システムの Engine ID を表示します
サーバ	IPv4 または IPv6 IP アドレス	
トランスポートプロトコル	UDP-IPv4 TCP-IPv4 UDP-IPv6 TCP-IPv6	トラップの送信に使用されるプロトコル。デフォルトは UDP-IPv4 です。
ポート	TCPポート	デフォルト値は 161 です。
トラップバージョン	バージョン 2c バージョン 3	使用する SNMP バージョン
バージョン 2c - コミュニティ	コミュニティ名	
バージョン 3 - ユーザー名	ユーザー名	
バージョン 3 -セキュリティ レベル	noAuthNoPriv authNoPriv authPriv	
バージョン 3 -認証アルゴリズム	MD5 SHA	
バージョン 3 -認証パスワード	パスワード	
バージョン 3 -プライバシーアルゴリズム	DES AES	
バージョン 3 -プライバシーパスフレーズ	パスフレーズ	

Eメール通知

イベントはEメールでEメールアドレスに送信できます。 以下のオプションを使用できます。

設定	値	説明
サーバ	SMTP サーバアドレス	
ポート	使用する TCP ポート	デフォルトのポートは 25 です
ユーザー名	ユーザー名	
パスワード	パスワード	
パスワードを確認してください	パスワード	
送信先メール	Eメールアドレス	イベントの送信先となるEメールアドレスをターゲットにします
TLSを起動する	TRUE FALSE	通信に TLS を使用する必要があります

モニタリング

監視機能により、Nodegrid は、Nodegrid センサーに接続されている、またはプロトコルとして SNMP または IPMI をサポートしている管理対象デバイスから、センサーデータを監視および収集できます。

収集されたデータは、`Monitoring Templates`を通じて定義、および制御され、その設定中に監視対象デバイスに割り当てられます。

監視テンプレートのカスタマイズ

既存の監視テンプレートは数多く存在し、通常はユーザーの要件を満たします。必要に応じて、これらのテンプレートをカスタマイズできます。

すべてのテンプレートは、SNMP または IPMI のいずれかの監視データの収集に使用されるプロトコルに従って、`/etc/collectd.templates`ディレクトリのサブディレクトリに配置されるテキスト ファイルです。

- `/etc/collectd.templates/snmp`
- `/etc/collectd.templates/ipmi`

これらのディレクトリ内の任意の新規ファイルは、ユーザーインターフェースに自動表示されます。

SNMP テンプレート

新しい SNMP テンプレートを作成するには、Shell にルートとしてログインします。新しいテンプレートの開始点として、既存のテンプレートのひとつのコピーを作成します。

各 SNMP テンプレートファイルには、2 種類のサブセクションがあります:

- 日付
  - データポイントごとにエントリ 1 つ、それぞれが固有の ID で識別されます。
- ホスト
  - 1 つのエントリで、SNMP パラメータ、収集間隔、および収集するデータポイントを定義します。

テンプレートファイルには、関心のあるデータポイントのみを含める必要があり、他のすべてのデータポイントはファイルから削除できます。

次の表は、データ入力の設定と可能な値について説明するものです。

設定	値	説明
データ	Nodegrid システムによって収集されるデータポイントの内部名。固有の名前である必要があります。	名前にはスペースを含めないでください。例 Data "pdu_in_cur" Data "pdu_in_vol"
タイプ	温度<br ファンの速度 湿度 カウンター *残り時間* 電圧 電流 電力 皮相*電力* *電力*要素 頻度	データのタイプ
表	TRUE FALSE	OID が表の一部であるかどうか反映されます
インスタンス	true false	**表**がTRUEの場合: 対応する値に関連付けられている名前のリストを取得するためにウォークされる NMP OID プレフィックス。たとえば、PDU では、これがコンセント名である場合があります。**表**がFALSEの場合: 値に関連付けられる [インスタンスの] 名前を文字列として指定します。
インスタンスプレフィックス	文字列	*オプション*。インスタンスの先頭に付いた文字列で、二重引用符で囲まれています。
値	true false	**表**がTRUEの場合: 値のリストを取得するためにウォークされる SNMP OID プレフィックス。 **表**がFALSEの場合: 単一の値の取得に使用される SNMP OID。
スケール	10進値	*オプション*。永続化する前に取得した値に乗算する10進値。

例

```
    <Data "pdu_in_cur">
      Type "current"
      Table true
      Instance ".1.3.6.1.4.1.476.1.42.3.8.40.20.1.20"
      Values ".1.3.6.1.4.1.476.1.42.3.8.40.20.1.130"
      Scale 0.01
    </Data>
```

SNMP テンプレートでのホスト入力は、`Collect`設定での調整のみが必要です。値リストには、収集する必要があるすべてのデータ入力のリストを含める必要があります。リストされているすべてのデータ入力には、対応するデータ入力の定義が必要です。

IPMI 検出テンプレート

IPMI の'検出'テンプレートは、IPMI デバイスで使用可能なすべてのセンサーを自動検出します。

テンプレートには、サブセクション 1 つ 、ホスト、および目的のオプションだけが含まれます:

設定	値	説明
AuthType	none md2 md5 straight	IPMI プロトコルの認証タイプ。デフォルトでは、最も強力なものをネゴシエートします。
権限	コールバック ユーザー オペレータ 管理者	IPMI プロトコルの権限レベル。デフォルトは `admin` です。
センサー	収集されるセンサーの名前	**IgnoreSelected**に応じて、収集または無視するセンサーを選択します。複数回の定義が可能であり、それぞれが1つのセンサーを選択します。
IgnoreSelected	true false	True の場合、**センサー**によって選択されたセンサーは収集されません。 False の場合は、**センサー**によって選択されたセンサーのみを収集します。
スケール	"<センサー名>" <乗数>	*オプション*。永続化する前に取得した値に乗算する10進値。

監視を有効化

監視は、デバイスごとに有効になります。この設定は、管理対象デバイス設定の一部です。監視を有効にするには、次の手順が必要です。

- `Managed Device`セクション内のデバイスに移動します。
- 特定のデバイス上で`Management`セクションに移動します
- SNMP や IPMI などの必要な監視プロトコルを有効にして設定します
- 監視を有効にし、テンプレートと収集間隔を割り当てます。

ダッシュボード

Nodegrid は、イベントの詳細、管理対象デバイスの詳細、およびシステムと管理対象デバイスからの監視データを視覚的に表示するダッシュボードツールを提供します。さまざまな目的に合ったいくつかのダッシュボードを柔軟に作成し、電力消費、電圧(V)、電流(A)、温度、ファン速度などの管理対象デバイスのデータポイントを監視できます。最後の 15 分、最後の 1 時間、最終日、今週、今月、過去 5 年間など、異なる期間のデータを表示できます。

ダッシュボードガイドは、必要に応じて拡張できるシンプルで便利なダッシュボードを作成し、ユーザーのニーズに合ったダッシュボード作成の出発点となります。

> 注: ダッシュボード機能は、WebUI を介してのみ使用できます。

データポイントの探索

このセクションは必須ではありませんが、収集されたデータが保存されていることを確認する方法と、収集されるデータの詳細について取り上げています。 収集された生データポイントは、以下の手順で確認できます。

- 以下をクリック `Dashboard`
- 以下をクリック `Discover`
- 必要なインデックスのパターンを選択します
  - `logstash-*` 監視対象データが含まれます
  - `*_date_*` イベント通知が含まれます
- デフォルトで、定義された時間枠内で収集されたすべてのデータが表示されます。
  - 必要に応じて時間枠を調整します
  - `Search` フィールドを使用して、特定デバイスまたはデータポイントを検索します
- 収集されたデータポイントを確認し、使用可能なフィールドを調査します。

> 注: 収集されたデータは、保存される前にバッファリングされるため、データを視覚化する前に数回の収集サイクルが必要になります。

次のフィールドを検索式で使用できます。

データポイントフィールド (`logstash-*` インデックス)

フィールド	値	説明
ホスト	デバイス名	監視対象となるデバイスの名前。
プラグイン	snmp ipmi nominal aggregation	コレクションプラグインの名前
プラグイン_インスタンス	合計 平均	プラグインが必要とする場合、 データを収集するプラグインのインスタンス。集約プラグインに存在
収集_タイプ	温度<br ファンの速度 湿度 カウンター *残り時間* 電圧 電流 電力 皮相*電力* *電力*要素 頻度	測定のタイプ
インスタンスの_タイプ	データポイント名	測定に関連付けられた要素の名前

デバイスフィールド (`logstash-*` インデックス)

フィールド	値	説明
名前	デバイス名	監視対象となるデバイスの名前。
モード	有効 オンデマンド 無効	デバイスの動作モード
タイプ	デバイスのタイプ	`Managed Devices`で割り当てられたデバイスのタイプ
ファミリー	ilo drac ipmi_1.5 ilmi_2.0 cimc_ucs device_console pdu	デバイスのファミリー
addr_ロケーション	アドレス	
座標	座標	
ip	IPアドレス	
mac	MACアドレス	デバイスの MAC アドレス (既知の場合)。
エイリアス	IP アドレスエイリアス	
グループ	グループのリスト	デバイスへのアクセス権を付与した認証グループ
ライスセンス取得済み	はい いいえ	デバイスのライセンスの状態
ステータス	接続 切断 使用中 不明	デバイスの現在の状態
nodegrid	Nodegrid ホスト名	デバイスを制御する Nodegrid のホスト名
カスタムフィールド		デバイス用に設定された任意のカスタムフィールド

イベント フィールド (`*_date_*` インデックス)

フィールド	値	説明
event_id	番号	イベントID番号
event_msg	テキスト	イベント メッセージ
ホスト	Nodegrid ホスト名	イベントが発生した Nodegrid のホスト名。
メッセージ	テキスト	フルメッセージテキスト

ビジュアライゼーションの作成

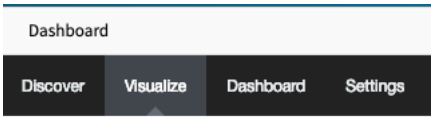
ビジュアライゼーションにより、収集したデータをダッシュボードに表示できます。ビジュアライゼーションには、データを表示・集計するための各種オプションが備わっています。次のセクションは、使用可能なオプションの小さいサブセットを取り上げ、カスタマイズされたビジュアライゼーション作成プロセスの開始点となることを目指します。

折れ線グラフ

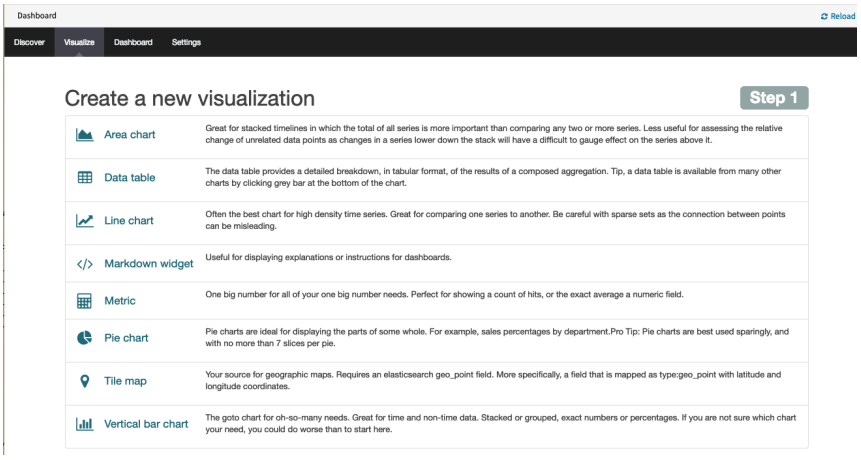
折れ線グラフは、グラフに沿ってデータポイントを視覚化できます。このグラフは、最も一般的に使用されるグラフの一つです。

折れ線グラフを作成するための一般的な手順は、以下のプロセスで概説します。

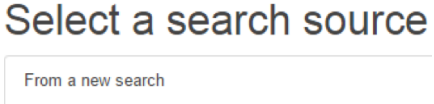
- 以下をクリック `Visualize`



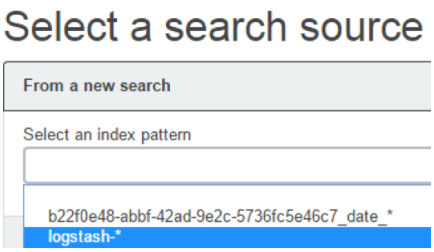
- 使用するビジュアライゼーションのスタイルを選択します。この例では、`Line Chart`を作成します。



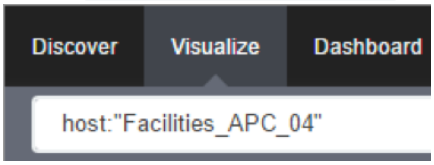
- `Select a search source` これには、**新しい検索から**をクリックします。



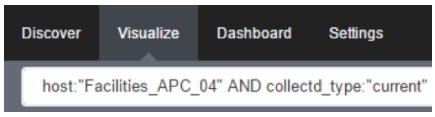
- **Logstash-***を以下として選択します `index pattern`



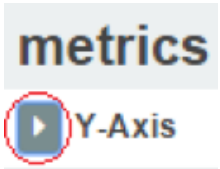
- 検索フィールドに`host:"<device name>"`などの検索式を入力して、視覚化するデータポイントを選択します。



検索式を拡張して絞り込むことができます。

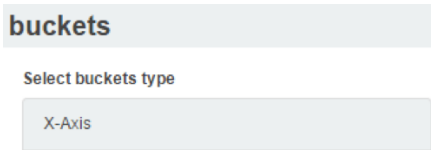


- Y 軸の左側にある`Arrow`をクリックして拡張します。

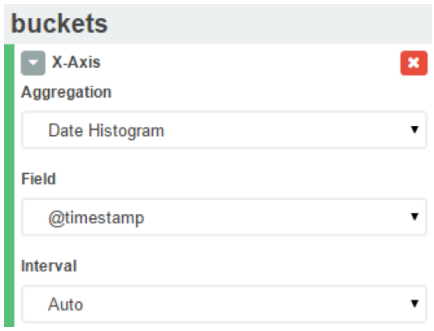


- `Aggregation`の**平均**とフィールドの**値**を選択します。

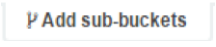
- X 軸をクリックします。



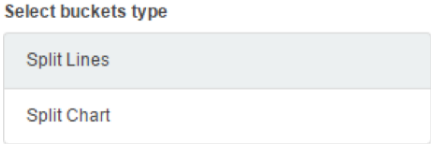
- **日付ヒストグラム**を`Aggregation`として選択します。`Field`と`Interval`はデフォルトのままにします。 以下は複数線の折れ線グラフを設定するためにデータポイントを分割するための手順なので、ビジュアライゼーションに折れ線グラフ 1 本だけを使用する場合は、次のサブステップを飛ばします。



  - `Add sub-buckets`をクリックして、複数のデータポイントを追加します。

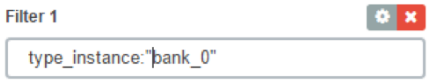


  - `Split Lines`をクリックします。

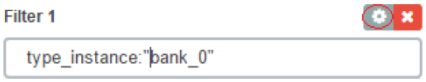


- **フィルタ**を`Sub Aggregation`として選択します。

- 検索式を入力して、視覚化する要素を選択します。

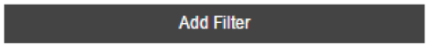


- 必要に応じて、以下をクリックしてラベルを関連付けます `settings icon`



- **ラベル**を指定します

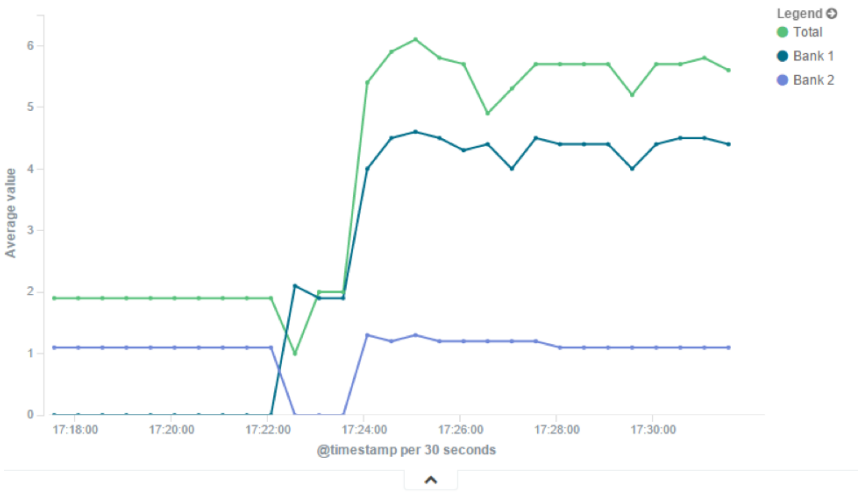
- `Add Filter`をクリックして、ビジュアライゼーションに別の要素を追加します。



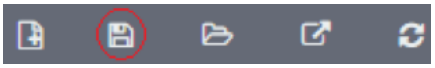
  - これらの手順を繰り返して、必要なすべての要素を追加します。

- 緑色の矢印をクリックすると、指定した設定に基づいてグラフを更新できます。

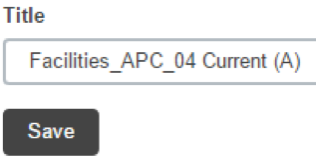
- グラフには、指定した設定が反映されます。



- `Save` アイコンをクリックして、ビジュアライゼーションを保存します。



- ビジュアライゼーションに**タイトル**を付けて、`Save` をクリックします。

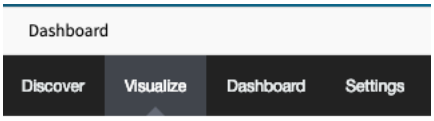


面グラフ

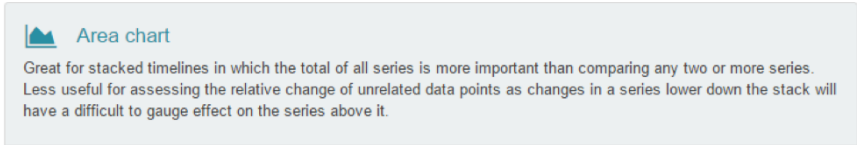
面グラフは、PDU のコンセントなど、さまざまな関連エンティティの測定値を蓄積する場合に便利です。

> 注: 先に進む前に、`Line Chart` セクションを確認してください。 `Area Chart`

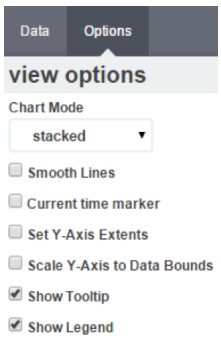
- 以下をクリック `Visualize`



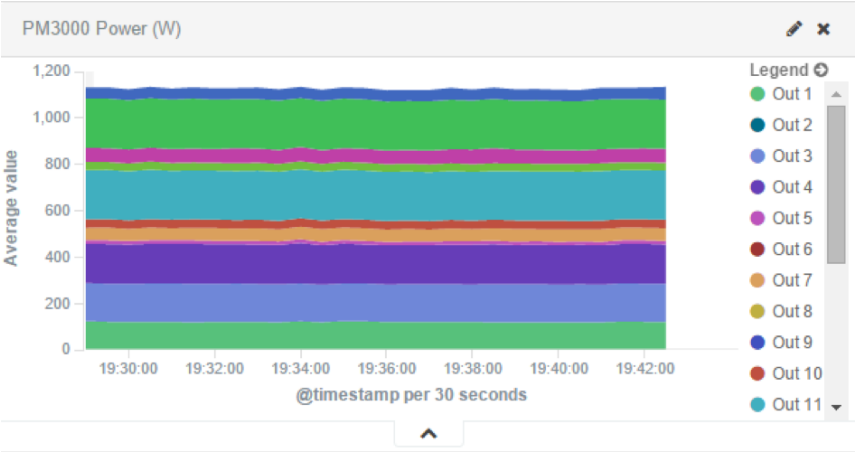
- 使用するビジュアライゼーションのスタイルを選択します。この例では、`Area Chart` を作成します。



- ビジュアライゼーションのオプションを、`Chart Mode` が**蓄積される**ように設定します。



- このビジュアライゼーションは、このような外観です

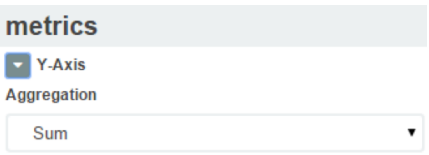


- すべての検索式は、ビジュアライゼーションの作成に使用されるデータポイントを選択または制限するために使用されます。これらは、ビジュアライゼーション全体のフィルター、サブ集計フィルタ、またはダッシュボード全体のフィルターとして使用できます。

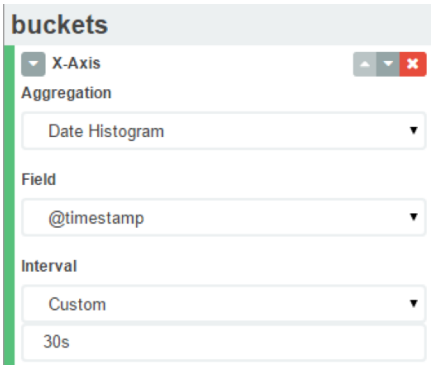
  これらの検索式は、データポイント'に限定されず、タイプ、IP アドレス、認証グループ、カスタムフィールドなど、Nodegrid のデバイスに関連付けられているフィールドを参照することもできます。

  たとえば、ラック PDU の選択で各コンセントによって提供される電流を収集するには、ひとつはカスタムフィールド“rack:abc”、もうひとつはカスタムフィールド“rack:xyz”です。

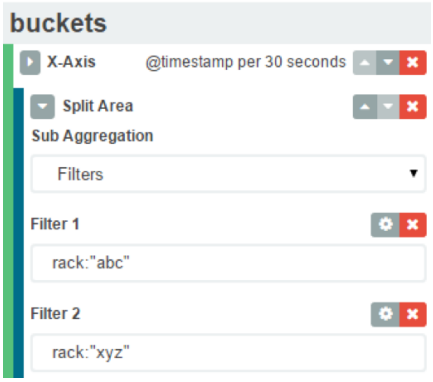
- 各ラック PDU のコンセントによって提供される電流の合計を表示するには、以下の設定を使用します。
  - **合計**としてのビジュアライゼーション `Aggregation`



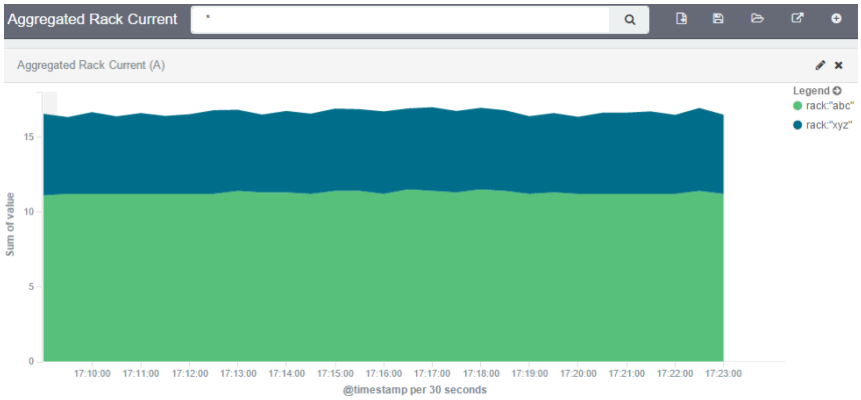
- 収集期間に一致する`buckets`間隔



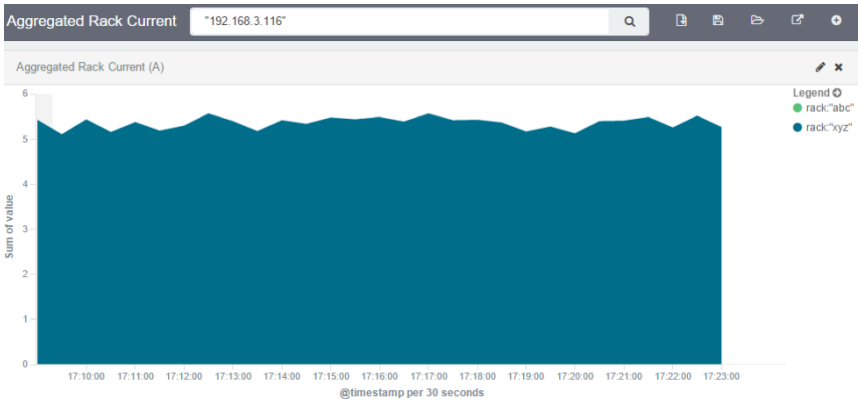
- `Sub-Aggregation`フィルタはカスタム フィールドに設定されます



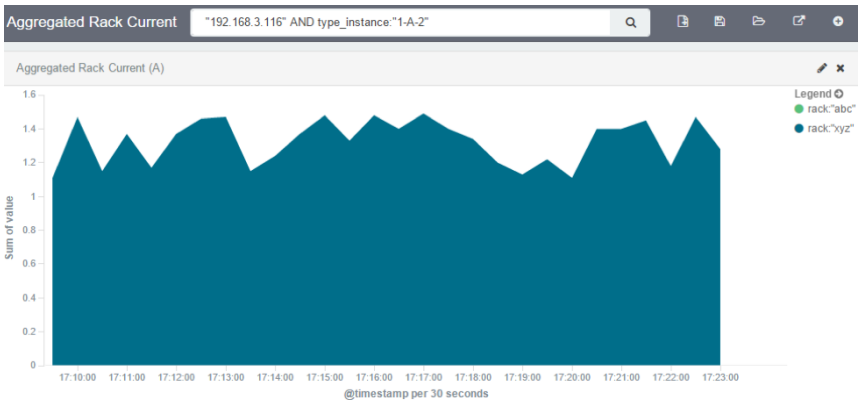
- ビジュアライゼーションの結果はこのような外観になります



- IP アドレスを使用して、1 つのラック PDU の値だけを表示するには、フィルタを追加します。



- 追加のフィルタは、すべて同じビジュアライゼーションから必要に応じて使用できます。

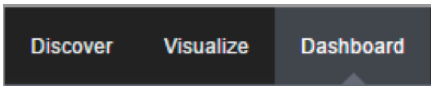


> 注: 面グラフを使用する場合は、電力消費者と電力供給者、またはラック PDU' の入出力電力を混ぜることで、同じ測定値を 2 回使用しないように特に注意します。

ダッシュボードの作成

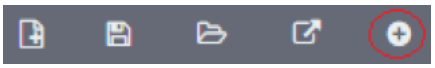
ダッシュボードは、1 つ以上のビジュアライゼーションのコレクションです。変更、または新しくダッシュボードを作成することもできます。次の手順では、新しいダッシュボードの作成方法を概説します。

- ダッシュボードをクリックします。

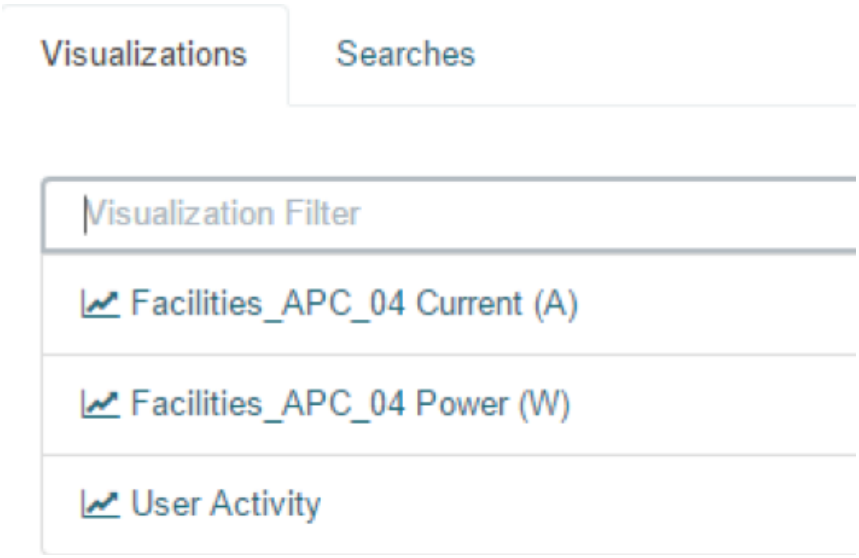


- 新しいダッシュボードのアイコンをクリックします。

- ビジュアライゼーション追加アイコンをクリックします。

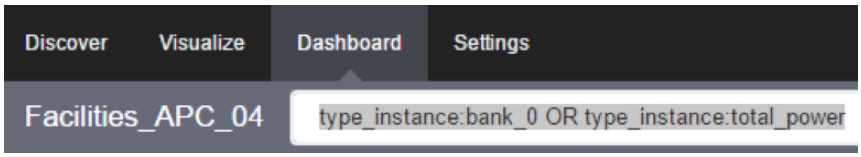


- ここには、以前保存したビジュアライゼーションが表示されます。ダッシュボードに追加するビジュアライゼーションをクリックします。
- 必要なビジュアライゼーションがすべて追加されるまで、前の手順を繰り返します。

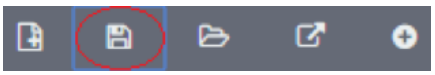


- 必要に応じて、グラフのサイズ変更や再配置を行います。

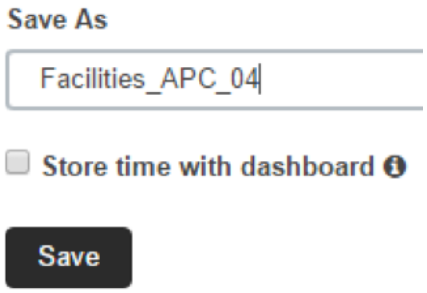
- 該当する場合は、フィルタをダッシュボードに追加できます。



- 保存アイコンをクリックします。



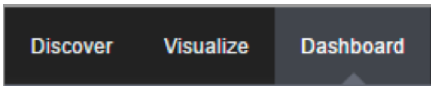
- ダッシュボード名を入力し、保存をクリックします。



ダッシュボードの検査

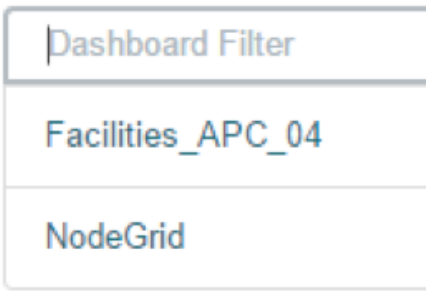
この時点から、以下の手順に従ってダッシュボードを開いて表示できます。

- ダッシュボードをクリックします。



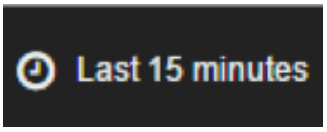
- フォルダアイコンをクリックします。

- ダッシュボード名をクリックします。ダッシュボードフィルタに検索式を入力して、 ダッシュボードを検索できます。

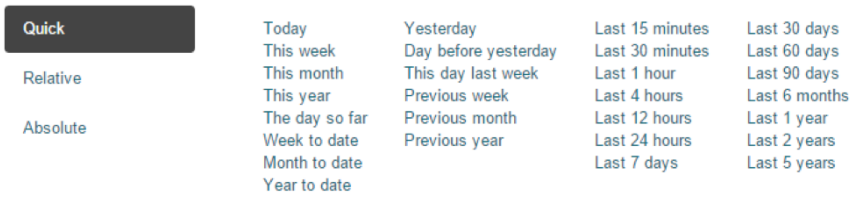


- 選択したダッシュボードが表示されます。

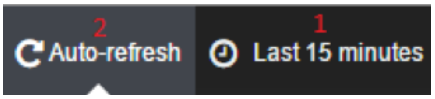
- 表示時間枠は、時計アイコンをクリックして調整できます。



- 新しい時間枠を選択します。



- 自動更新アイコンをクリックしてダッシュボードを自動更新し、更新頻度を選択することができます。



アプリケーション

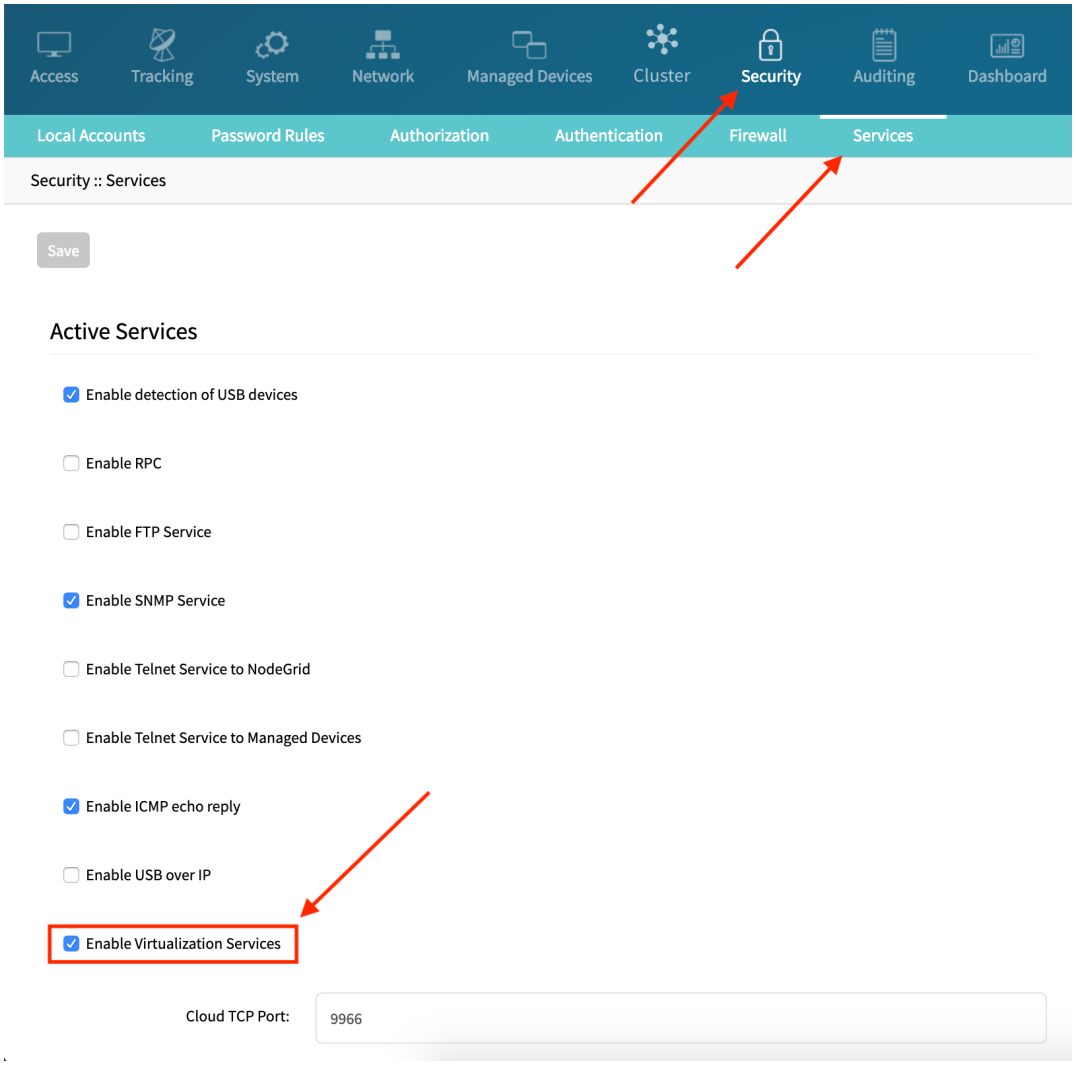
Nodegrid Platform では、追加のアプリケーションを実行できます。これは主に、特定アプリケーションをエンドデバイスの近くで実行するなど、ソフトウェア機能を拡張するために使用されます。監視と SD-WAN の分野で最も一般的に使用されます。すべての Nodegrid ユニットはこの機能をサポートしますが、このServices Router ファミリは、アプリケーション実行用に特に設計されており、幅広い接続オプションを提供します。

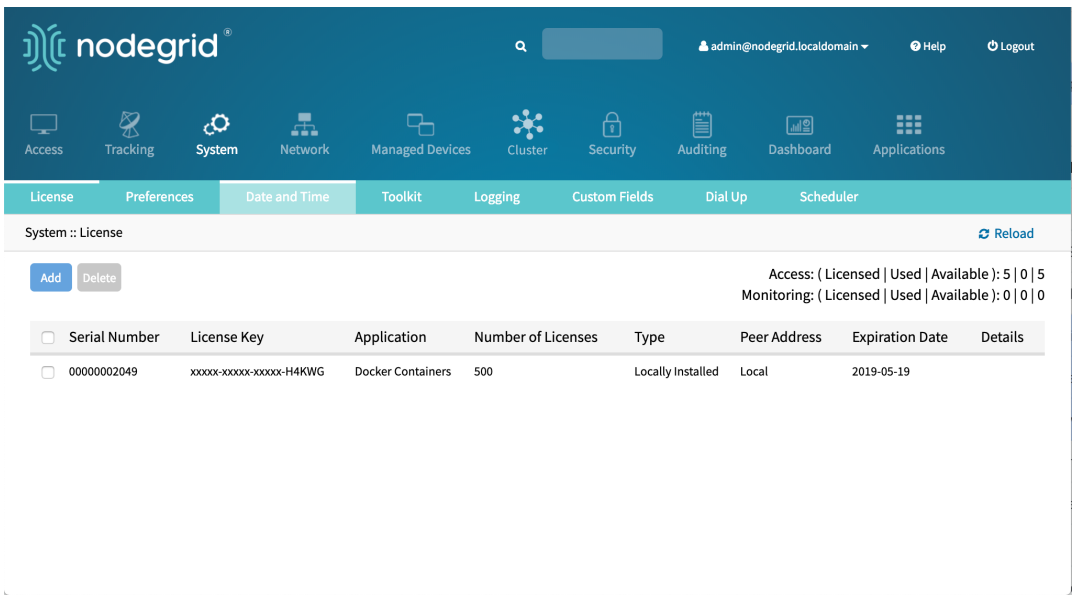
> 注: アプリケーション機能を使用するには、追加のライセンスをインストールする必要があります。仮想化サービスはデフォルトで無効になっており、以下で有効化する必要があります `Services`

Docker アプリケーション

Docker は、分散型アプリケーションの構築、配信、実行を行うためのオープンプラットフォームです。Nodegrid Platform を使用して、管理者は Docker アプリケーションを実行できます。このプラットフォームでは、**Docker Hub**から Docker アプリケーションを引き出し、Docker コンテナの開始・停止を行えます。

> 注: [仮想化サービスの有効化] は、セキュリティで有効にする必要があります:: NFV または Docker アプリを実行するためのサービス。どちらの機能も、ライセンスが必要です (システム :: ライセンス)。





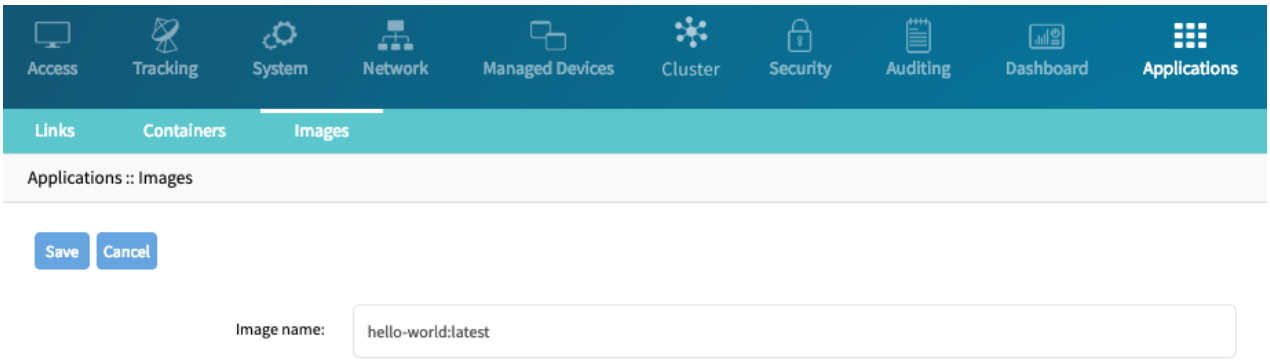
> 注: 現在、Docker アプリケーションの管理は WebUI を介してのみ行えます。WebUI は、Docker コンテナを管理するための基本的なインターフェースを提供します。管理者は docker コマンドラインツールで、より高度な機能を使用できます。

Docker イメージ

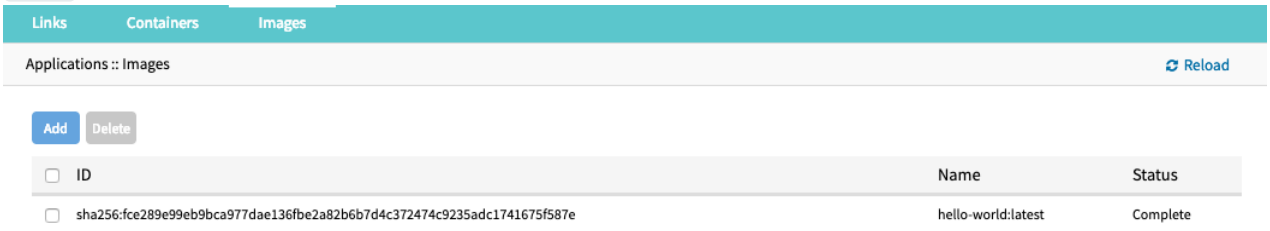
この`Applications :: Images`セクションで、管理者は特定の Docker コンテナイメージをダウンロードおよび削除できます。それらは、**Docker Hub**から直接ダウンロードできます。これには、Nodegrid から Docker Hub への直接ネットワークアクセスが必要です。

新しいイメージは、以下の手順に従ってダウンロードできます。

- 移動先 `Applications :: Images`
- 以下をクリック `Add`
- ダウンロードするイメージを提供します。特定のバージョンを、`:`記号を使用してダウンロードすることができます



- `Save`をクリックすると、イメージがダウンロードされます。



Docker コンテナ

この`Applications :: Containers`セクションで、管理者は既存のイメージに基づいてコンテナを、Nodegrid システムに追加できます。コンテナは、必要に応じて開始、停止、および削除できます。

詳細については、Docker 作成の公式文書を参照してください。

> 注: コンテナ作成後、自動では開始しません。

コンテナを追加するには、次の手順に従います。

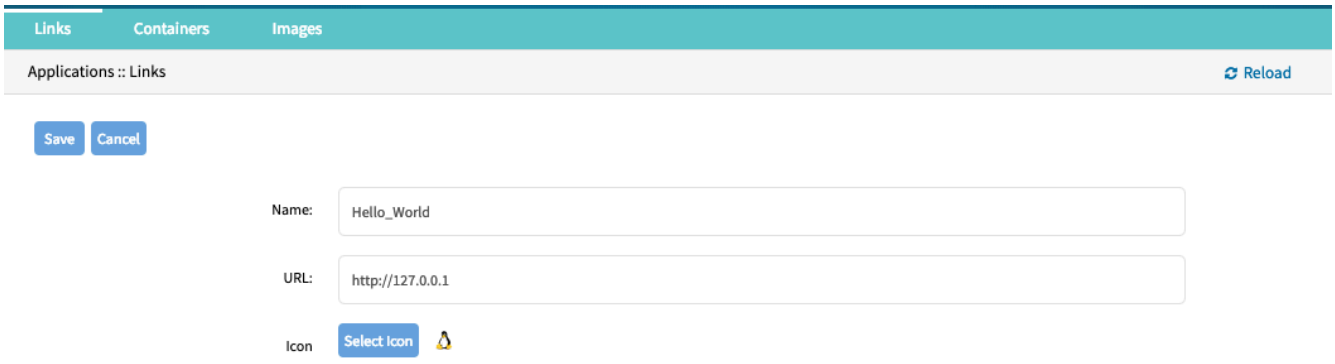
- 移動先 `Applications :: Containers`
- [追加] をクリックする
- 以下の情報を提供します

設定	値	説明
画像名	Docker イメージの名前	有効なイメージ名のリストは以下で閲覧できます `Applications :: Images`
コマンド	コマンド	コンテナ内で実行されるオプションのコマンド
ホスト名	ホスト名	コンテナに割り当てられるホスト名
ドメイン	ドメイン名	コンテナに割り当てられるドメイン名
コンテナ名	名前	Docker コンテナの名前
CPU	量	コンテナに割り当てられる CPU の量
メモリ (MB)	量	コンテナに割り当てられた RAM の量
引数	引数	コンテナの作成時に使用されるオプション

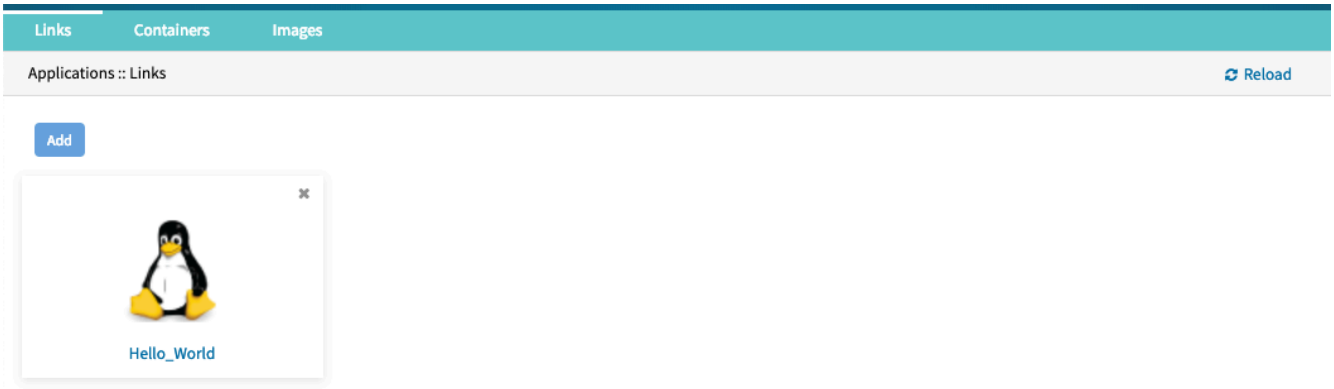
アプリケーションリンク

アプリケーションリンクで、管理者は実行中のコンテナやその他のアプリケーションへの簡単な Web リンクを作成できます。

- `Applications :: Links` へ移動する
- `Add` をクリック
- リンクに `Name` を付けます
- `URL` フィールドに有効な URL を入力します
- `Select Icon` で使用するアイコンを選択します



- これで、リンクが利用可能になります



> 注: アプリケーションによっては、作成されたアプリケーションのターゲットデバイスを作成すると有利な場合があります。

ネットワーク機能仮想化

Nodegrid Platform で、管理者は追加の NFV またはその他の仮想マシンを実行できます。コマンドラインインターフェースを介して、広範な設定オプションを使用できます。

詳細については、**テクニカルサポート**にお問い合わせください。

付録

テクニカルサポート

弊社のテクニカルサポートスタッフは、ライセンスを受けたお客さまの Nodegrid 製品に操作上の問題またはインストール上の問題が発生した場合に、サポートを提供しています。可能な限り迅速にサポートを受けるために、以下の手順に従ってください:

- このマニュアルの関連セクションを参照し、示されている推奨手順で問題を解決できるかどうか確認します。
- オンラインヘルプドキュメントは、www.zpesystems.com/supportを参照してください。
- ナレッジベースやその他の役立つリンクについては、ヘルプセンターのウェブサイトをご覧ください。

サポートチケットの送信

オンラインでサポートリクエストチケットを送信するには、次の手順に従ってください:

- ページの右上隅にある`Submit a request`リンクをクリックします。
- リクエストフォームに必要な情報を入力します。問題や質問の内容をできるだけ詳しく記入してください。
- 添付ファイルがある場合は、ファイルを追加するか、ドロップ領域にファイルをドロップします。
- チェックボックス`I'm not a robot`をオンにします。
- `Submit` をクリックします

リクエストの受信を確認するメールが ZPE Systems から届き、サポートスタッフがそれを確認します。E メールにはチケット番号も記載されています。必要に応じて後で参照できるように、チケット番号をメモしてください。

更新とパッチ

重要なセキュリティパッチのアナウンス、今後のファームウェアの更新、およびその他の技術情報を自動で受信するために、こちらから**The Loop**にサインアップしてください。

www.zpesystems.com/loop/

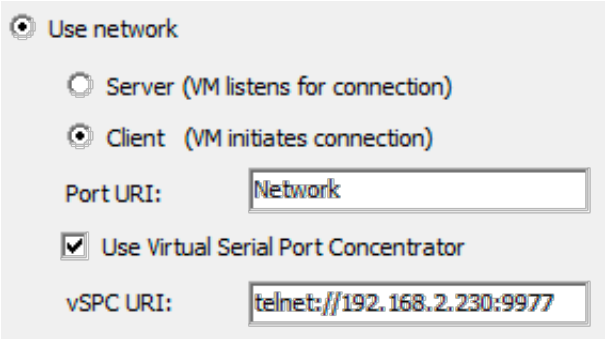
VM サーバでの仮想シリアル ポート (vSPC) の設定

Vmware 仮想マシン vSPC データを Nodegrid Platform にリダイレクトするには、仮想マシンのシリアルポートを以下に示すように設定する必要があります:

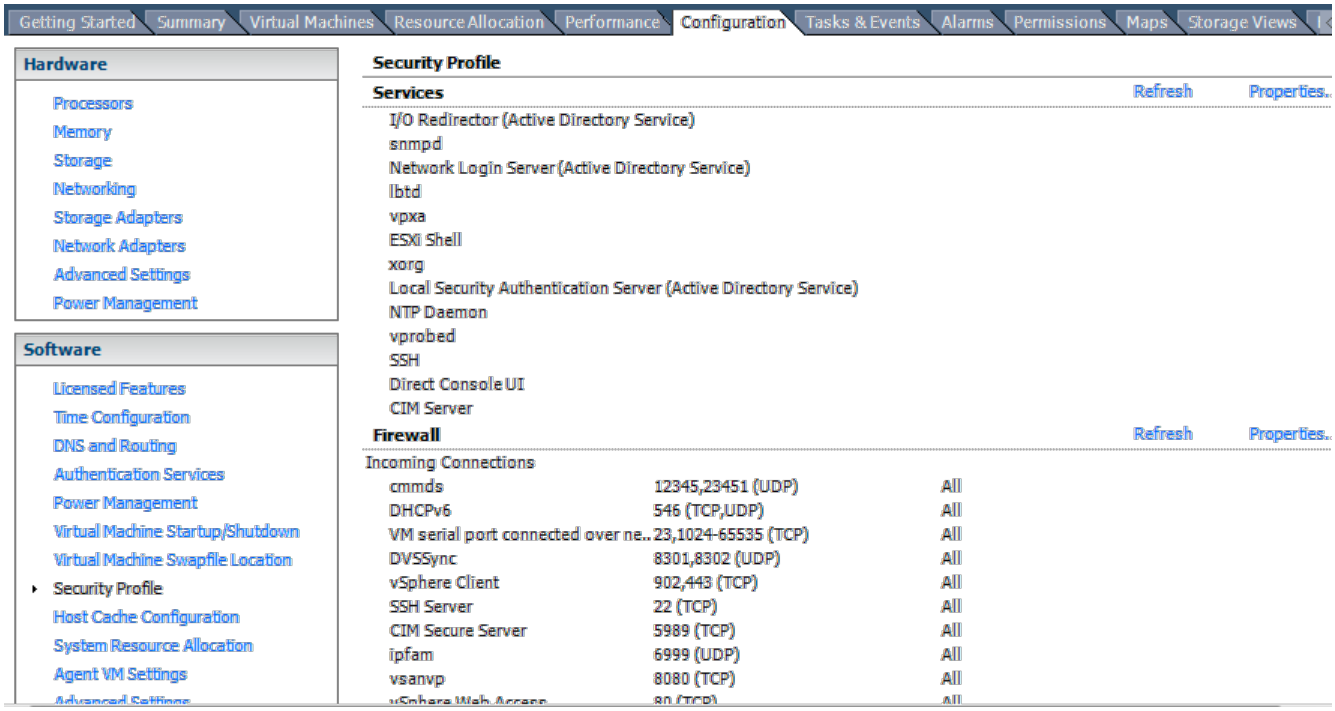
- ESXi 設定 (vSphereTM) に移動します。接続する仮想マシンを選択し、[仮想マシン設定の編集] リンクをクリックします。
- [追加] をクリックします。仮想マシンをオフにする必要があります。
- [シリアル マネージャ デバイス] をクリックし、ポップアップウィンドウの [次へ] をクリックしま

す。

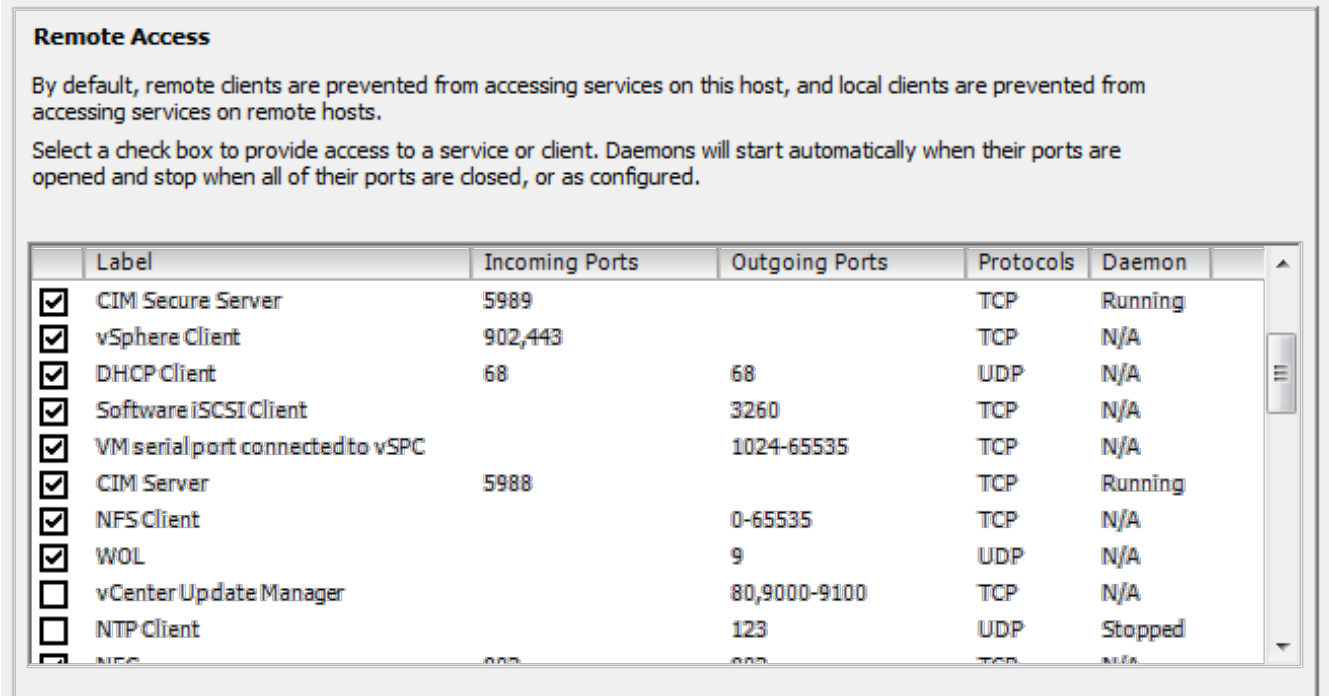
- [ネットワーク経由で接続] をクリックし、[次へ] をクリックします。
- クライアントの選択 (VM が接続を開始します) – はデフォルトです
- ポートURIに <group_id> と入力します。group_idは、自動検出時に同じグループのサーバを関連付けるために使用できる識別子です。このフィールドは省略可能です。
- vSPC URI で、telnet://<IP or Nodegrid Manager hostname>:9977 と入力します
- [完了] をクリックします。



- 最後に、ESXi ファイアウォールで vSPC ポートが有効になっていることを確認します。これを確認するには、ESXi 設定に移動し、[セキュリティプロファイル] を選択し、[プロパティ] をクリックします。



- リモートアクセスページで、vSPC に接続されている VM シリアル ポートに関連するチェック ボックスをオンにします。発信ポートには 1024 以上の TCP ポート範囲が必要であり、ポート範囲には vSPC URI フィールド (デフォルトで 9977) で使用される TCP ポートを含む必要があります 。



発信ポートの範囲を変更するには、ESXi コマンドラインに接続し、以下のコマンドを実行します:

```
~ #
~ # vi /etc/vmware/firewall/service.xml
```

ポートセクションを編集します:

```
<!-- Remote serial port with vSPC: all remote serial port traffic is initiated
<service id="0030">
  <id>vSPC</id>
  <rule id='0000'>
    <direction>outbound</direction>
    <protocol>tcp</protocol>
    <porttype>dst</porttype>
    <port>
      <begin>1024</begin>
      <end>65535</end>
    </port>
  </rule>
  <enabled>false</enabled>
  <required>false</required>
</service>
```

変更を保存し、ファイアウォールサービスを再起動します:

```
~ #
~ #   esxcli network firewall refresh
```

VMware ファイアウォールの詳細は、VMware ナレッジベースを参照してください。

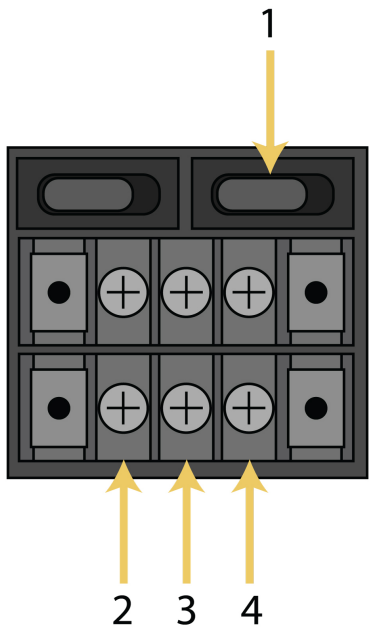
DC 電源

DC 電源は、次の 3 本のワイヤを使用して DC 電源装置に接続されます。リターン (RTN)、グラウンド (⏚) および 48 VDC。

> **警告** 電源が Nodegrid の DC 電源要件を満たしていることが重要です。先に進む前に、お使いの電源タイプが正しいものであり、DC 電源ケーブルが良好な状態であることを確認してください。これを怠ると、人身傷害や機器の損傷を引き起こす可能性があります。

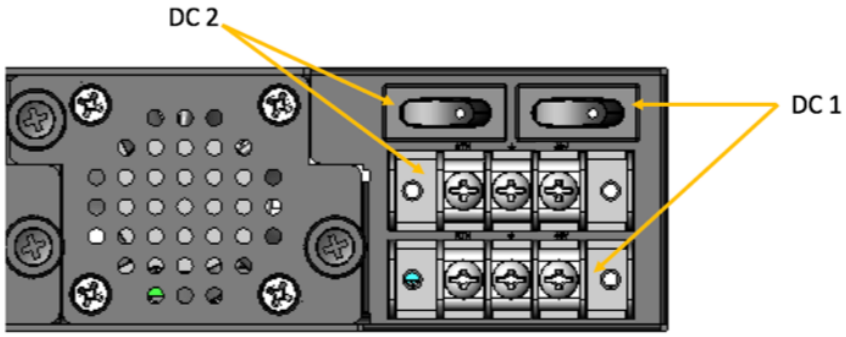
> **警告** DC 電源からの配線は、電源のプラスのワイヤ (通常は赤) が接地され、ホットワイヤ (通常は黒) が -48VDC を運ぶ、テレコムラックで特に混乱を招く可能性があります。疑わしい場合は、接続作業を進めず、資格を持つ電気技師にご相談ください。正しい接続を行わないと、人身傷害や機器の損傷を引き起こす可能性があります。

基礎

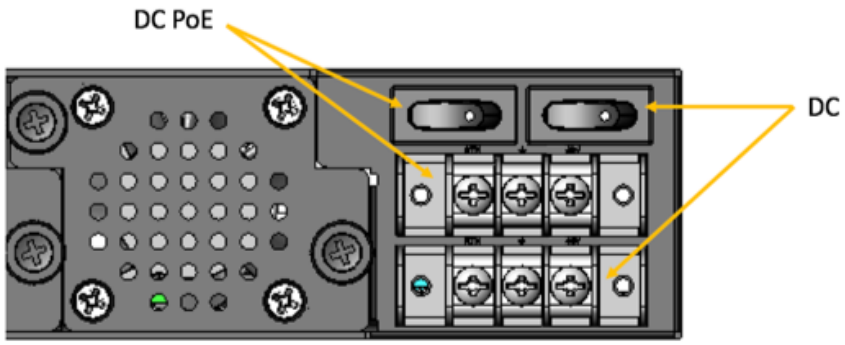


図：デュアル DC 電源接続端子ブロック

番号	説明
1	電源スイッチ
2	RTN (リターン)
3	グラウンド (⏚)
4	48 VDC



図：DC アソシエーション - 端子電源とスイッチ



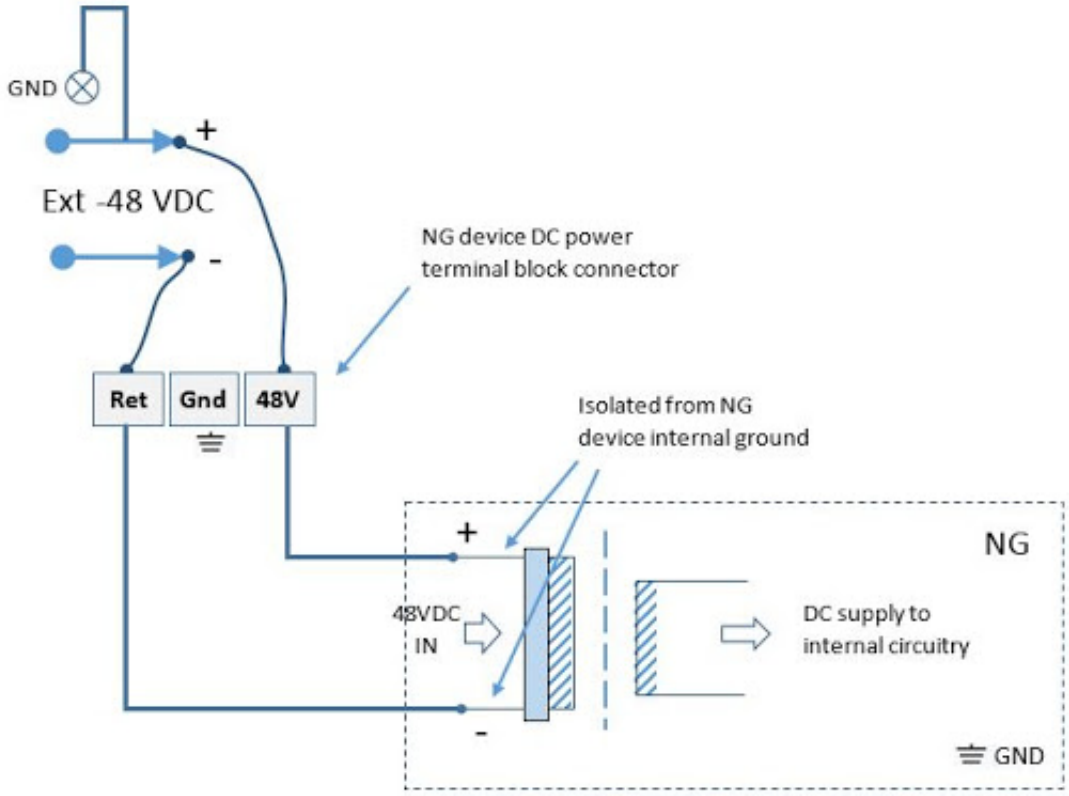
図：NSR シングル DC + PoE 電源接続端子ブロック

DC 電源で Nodegrid ユニットに電力を供給するには:

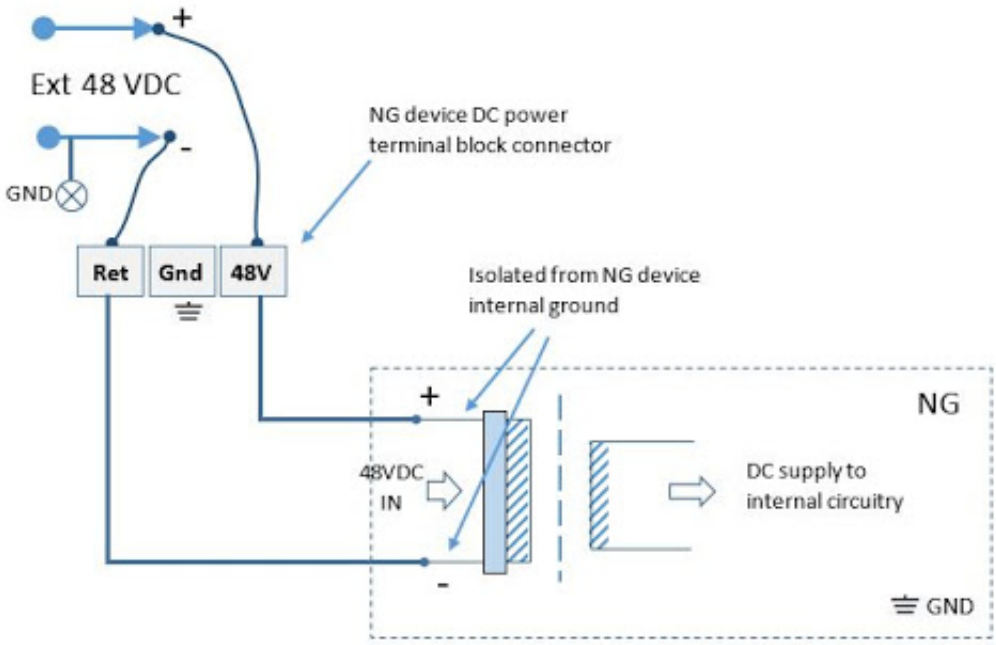
1. ユニットの電源が切れていることを確認します。
2. DC 電源ケーブルが電源に**接続されていない**ことを確認します。 通電しているワイヤでは**絶対に作業しない**でください。
3. DC 電源ブロックから保護カバーを左右にスライドさせて取り外します。
4. 3 本の DC 電源接続端子ネジをすべて緩めます。
5. リターンリードを RTN 端子に接続し、グラウンドリードを⏚端子に接続し、48 VDCリードは48 VDC端子に導き、ネジを締めます。
6. 保護カバーをスライドさせて DC 端子ブロックの上に戻します。
7. ユニットにデュアル入力 DC 端子がある場合は、2 番目の端子ブロックでも手順 3 ～ 6 を繰り返します。
8. DC 電源ケーブルを DC 電源に接続し、DC 電源をオンにします。
9. シリアルクライアント (115200 8N1 に設定) をコンソールポート (Teraterm、puTTY など) に接続します (オプション)
10. ユニットの電源をオンにします。接続しているシリアルクライアントでの起動メッセージを再確認します。
11. 接続されているデバイスの電源スイッチをオンにします。
12. DC 電源ケーブルを DC 電源に接続し、DC 電源をオンにします。
13. ユニットの電源をオンにします。

14. 接続されているデバイスの電源スイッチをオンにします。

-48VDC 電源の場合

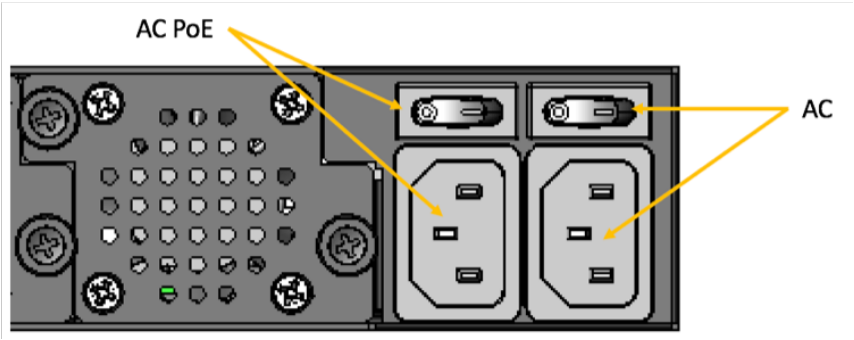


+48VDC 電源の場合



AC 電源

PoE+ サポートを備えた NSR モデルの AC ダイアグラム



図：NSR シングル AC + PoE 電源入力とスイッチ

シリアルポートのピンアウト

次の表にシリアルポートのピンアウト情報が表示されます。

Cisco のようなピンアウト

ピン	信号名	入出力
1	CTS	IN
2	DCD	IN
3	RxD	IN
4	GND	該当なし
5	GND	該当なし
6	TxD	OUT
7	DTR	OUT
8	RTS	OUT

レガシーピンアウト

ピン	信号名	入出力
1	RTS	OUT
2	DTR	OUT
3	TxD	OUT
4	GND	該当なし
5	CTS	IN
6	RxD	IN
7	DCD	IN
8	未使用	該当なし

安全性

製品の安全性に関する情報は、以下のリンクをご参照ください。

- Nodegrid Serial Console
- Nodegrid Services Router
- Nodegrid Bold SR

クイック インストール ガイド

製品のインストール情報については、以下のリンクを参照してください。

- Nodegrid Serial Console
- Nodegrid Services Router
- Nodegrid Bold SR

RoHS

RoHSに関する情報は、以下のリンクを参照してください。

- Nodegrid Serial Console
- Nodegrid Services Router
- Nodegrid Bold SR

データの永続性

通常の操作では、キーストローク、管理対象デバイス出力、および製品を通過するデバイス監視データに起因するユーザーデータは、設定でデータロギングと監視が有効になっている場合、不揮発性デバイスメモリに保存されることがあります。

Nodegrid デバイスには、次のメモリ デバイスが含まれます:

1. **BIOS** メモリサイズ: 64MB メモリタイプ: NOR フラッシュ 揮発性: 不揮発性 ユーザーデータ: いいえ
2. **フラッシュディスク** メモリサイズ: 32 GB または 64 GB。その他のカスタムサイズもご利用いただけます。 メモリタイプ: SSD 揮発性: 不揮発性 ユーザーデータ: はい パーティション/データ: sda2 - ユニット設定 sda5 - バックアップ設定 sda8 - ユーザーホームディレクトリとログファイル
3. **RAM** メモリサイズ: 4 GB または 8 GB メモリタイプ: DDR3 揮発性: 揮発性 ユーザーデータ: はい

Nodegrid ユニットの不揮発性メモリからユーザーデータを削除するには、次の 2 つの方法があります。

- **ソフト除去**: ファイルを削除し、フラッシュディスクに工場出荷時のデフォルト設定をインストールします。
- **ハード除去**: フラッシュディスクを完全に消去します。この手順によって、フラッシュディスク上のすべてのデータが破棄され、データ復旧サービスによっても、回復は不能になります。その後、Nodegrid ソフトウェアをネットワーク経由で再インストールする必要があります。

ソフト除去

次の手順で、Nodegrid の不揮発性メモリを消去します:

- 1. Nodegrid ユニットをシャットダウンし、電源をオフにします
- 2. ネットワークから Nodegrid ユニットを取り外します (ユニットのイーサネットケーブルを切断する)
- 3. Nodegrid ユニットに接続されている USB ストレージデバイスと USB ネットワークデバイスをすべて切断します
- 4. 次のいずれかのオプションを使用して Nodegrid ユニットにアクセスします:
    - a. RJ-45 コンソールアダプタと、端末またはワークステーションに接続されたストレートネットワークケーブルを使用した、Nodegrid コンソールポート。
    - b. HDMI モニターと USB キーボードに接続した HDMI ポートと USB ポート。

- 5. Nodegrid ユニットの電源を入れ、次のメニューで [Nodegrid Manager - レスキューモード] を選択する:

```
*****************************************************************************
*Nodegrid Manager <version>                                                 *
*Nodegrid Manager <version> - Factory Default Settings                      *
*Nodegrid Manager <version> - Rescue Mode       <--                         *
*Nodegrid Manager <version> - Network boot                                  *
*Nodegrid Manager <version> (verbose)                                       *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*                                                                          *
*****************************************************************************
```

```
` Use the * and * keys to select which entry is highlighted.
 [Enter] キーを押して、選択した OS を起動し、 [e] でコマンドを編集してから起動します
 または `c' for a command-line.`
```

- 6. プロンプト ("bash-4.3#") で、このコマンドを実行してすべてのファイルを消去し、工場出荷時の設定をを読み込み、引用符(")なしで 実行します: 'apply_settings --factory-and-cleanlogs -f -h'
- 7. 次のメッセージを待ちます:

> 工場出荷時に完了した設定を適用します。
> INIT: 切り替え中 [...] 再起動: システム停止

- 8. ユニットの電源を切る。

ハード除去 - 安全消去

次の手順で、Nodegrid ユニットの不揮発性メモリを消去します。

- 1. Nodegrid ユニットをシャットダウンし、電源をオフにします
- 2. ネットワークから Nodegrid ユニットを取り外します (ユニットのイーサネットケーブルを切断する)
- 3. Nodegrid ユニットに接続されている USB ストレージデバイスと USB ネットワークデバイスをすべて切断します
- 4. 次のいずれかのオプションを使用して Nodegrid ユニットにアクセスします:
    - a. RJ-45 コンソールアダプタと、端末またはワークステーションに接続されたストレートネットワークケーブルを使用した、Nodegrid コンソールポート。

    - b. HDMI モニターと USB キーボードに接続した HDMI ポートと USB ポート。
- 5. ユニットの電源を入れる
- 6. BIOS 設定ページが画面に表示されたら、[Esc] キーを押す。
- 7. Grubメニューで [Nodegrid Platform - セキュア消去] を選択します:

```
                            GNU GRUB version 2.00

 +----------------------------------------------------------------------------+
 |Nodegrid Platform - Chain boot                                              |
 |Nodegrid Platform - Rescue Mode                                             |
 |Nodegrid Platform - Secure Erase  <--                                       |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 |                                                                            |
 +----------------------------------------------------------------------------+
```

```
  `Use the ^ and v keys to select which entry is highlighted.
  [Enter] キーを押して、選択した OS を起動し、 [e] でコマンドを編集してから起動します
  または `c' for a command-line.`
```

- 8. システムからすべてのデータを永久に消去するには、[Secure Erase] を選択します:

> Nodegrid Boot live-セキュア消去
> この手順を使用すると、SSD上のすべてのデータが破壊され、データ復旧サービスを使っても回復不能になります。 この手順を実行すると、システムソフトウェアは存在しなくなるため、ネットワーク経由で再インストールする必要があります。
> 消去」と入力してSSDをセキュアに消去するか、「キャンセル」を選択して再起動します

注: セキュア消去では、消去コマンドを実行する前に、ユニットの電源をサイクルする（電源をオフにしてからオンにする）必要があります。そうしないと、以下のメッセージが表示され、システムが停止して、ユーザーに電源サイクルを要求します:

> 操作はサポートされていません。ユニットは、消去コマンドの前に電源をサイクルする必要があります。
> システムの停止を待ち、ユニットの電源をサイクルします。
> [4.614365] reboot: System halted
>
> [4.614365] reboot：システムが停止しました

- 9. 確認

> セキュア消去は、一度確認されるとキャンセルできません。
> 「yes」を入力してセキュア消去を確認します。

- 10. 「システムが停止しました」というメッセージを待ちます

> SSDの安全な消去が開始されます...
> security_password="PasSWorD"
>
> /dev/sda:
> SECURITY_SET_PASS コマンドの発行, password="PasSWorD", user=user, mode=high
> security_password="PasSWorD"
>
> /dev/sda:
> SECURITY_ERASE コマンドの発行, password="PasSWorD", user=user
>
> セキュア消去が完了しました。システムが停止しています...
> [29.083186] reboot: System halted
>
> [29.083186] reboot: システムが停止しました

- 11. ユニットの電源を切ります。

「ボラティリティ・ステートメント」

ボラティリティ・ステートメントのコピーは、こちらからアクセスしてください。 Letter of Volatility

クレジット

Contact us

セールス: sales@zpesystems.com

サポート: support@zpesystems.com

ZPE Systems, Inc.
46757 Fremont Blvd.
Fremont, CA 94538
USA

www.zpesystems.com