



## Cisco CloudCenter リリース 4.8.x 統合ガイド

初回更新日: 2017 年 7 月 19 日

### シスコシステムズ合同会社

〒107-6227 東京都港区赤坂 9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先: シスココンタクトセンター

0120-092-255 (フリーコール、携帯・PHS 含む)

電話受付時間: 平日 10:00~12:00、13:00~17:00

<http://www.cisco.com/jp/go/contactcenter/>

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェア ライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコとこれら各社は、商品性の保証、特定目的への準拠の保証と権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワークポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <http://www.cisco.com/go/trademarks>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2017 Cisco Systems, Inc. All rights reserved.

1. ACI .....	2
2. CloudHSM.....	5
3. SMTP メール サーバ.....	5
4. コールアウト スクリプト.....	6
5. Infoblox .....	15
6. Jenkins.....	16
7. SSO .....	22
7.1 SSO AD.....	23
7.2 SAML SSO.....	23
7.3 使用例: ADFS SAML SSO.....	27
8. ServiceNow .....	30
8.1 アーキテクチャ概要.....	30
8.2 統合の概要 .....	32
8.3 リリースノート v1.4 .....	34
8.4 インストールと設定 v1.4.....	35

# ACI

- 概要
- ACI の基礎
- 可用性
- 利点
- 統合の要件
  - APIC の要件
  - CloudCenter の要件
  - VMware vSphere の要件
- 拡張機能の使用

## 概要

CloudCenter のユーザは、設定済みアプリケーション プロファイルを使用して、アプリケーションのインフラストラクチャに依存しないモデルを作成できます。モデル化すると、Cisco CloudCenter プラットフォームと Cisco [アプリケーション セントリック インフラストラクチャ \(ACI\)](#) が連携し、自動化されたエンドツーエンドのプロビジョニングをアプリケーションのコンピューティング、ストレージ、およびネットワーク設定と、それらに必要な一連のコンポーネントに提供します。

## ACI の基礎

ACI ポリシー モデルの詳細については、『[Cisco ACI Fundamentals Guide](#)』を参照してください。

## 可用性

CloudCenter と ACI の統合は **VMware** クラウド環境で使用できます。

CloudCenter は次の [APIC](#) リリースをサポートしています。

- Cisco APIC、リリース 1.0
- Cisco APIC、リリース 1.1
- Cisco APIC、リリース 1.2
- Cisco APIC、リリース 2.0 (仮想分散スイッチ – DVS モードのみ)
- Cisco APIC、リリース 2.1

## 利点

CloudCenter と ACI の統合により、次の利点がもたらされます。

- 完全に自動化された ACI ポリシー オブジェクトの作成機能を使用できる。
- プログラミングやアプリケーション コードの変更、クラウドに固有のスクリプトの記述、あるいはネットワークに関する専門知識を必要とすることなく、ネットワークのマイクロセグメンテーションによるセキュリティと効率性を実現できる。
- 完全に統合された Cisco ACI ネットワーク ポリシーと設定により、ユーザがアプリケーションをセルフサービスまたはオンデマンドで導入・管理できる。

## 統合要件

CloudCenter プラットフォームは、オーバーレイ インフラストラクチャのエンドツーエンドでのプロビジョニングとアプリケーションの導入を自動化します。ACI では、これに次のリソースのプロビジョニングと管理が含まれています。



CloudCenter で設定する APIC テナントにこれらのリソースを作成する権限があることを確認します。

- アプリケーション ネットワーク プロファイル (ANP)
- エンドポイント グループ (EPG)
- コントラクト
- サブジェクト/フィルタ

APIC でアプリケーションをプロビジョニングし、設定するための CloudCenter プラットフォームの前提条件として、まず Cisco ACI の動作環境に必要な次の要件を満たします。

- リーフ スイッチ プロファイル、スイッチ セレクタ、インターフェイス プロファイル、およびポリシー グループ
- VLAN プール
- VMware の仮想マシン マネージャ (VMM) ドメイン
- 外部インターネット接続用に L3 アウトで設定された新しいテナントとブリッジのドメインへのルーティングが可能な IP サブネット
- ルーティング プロトコル
- VRF

## APIC の要件

Cisco Application Policy Infrastructure Controller (Cisco APIC) は HTTP と HTTPS の両方で機能します。

- HTTPS: デフォルトでは、Cisco APIC は、UI と REST APIS のいずれでも HTTPS のみをリスンします。APIC は、APIC ホスト名に対応する有効な SSL 証明書により設定されていることを確認します。
- HTTP: APIC の HTTP アクセスを有効にし、ホスト名または IP アドレスのいずれかを使用してアクセスできるようにします。

環境の健全性を確保するには、次の手順を実行します。

1. APIC UI を使用して、EPG が 1 つある新しいアプリケーション ネットワーク プロファイルを手動で追加します。
2. 新しい VMware 仮想分散スイッチ (vDS) ポート グループがプロビジョニングされており、APIC UI に表示されることを確認します。
3. vCenter の UI を使用して、作成済みのポート グループをポイントするネットワークが設定された新しい VM をプロビジョニングするか、または複製します。
4. ストリクト モードで操作する場合、VM に SSH/RDP でアクセスできません。
  - a. ポート 22/3389 のコントラクトを、ステップ 1 の EPG をプロバイダーとして作成します。
  - b. ステップ 4a で作成したコントラクトが使用する新しい L3 アウト設定を作成します。
5. ステップ 3 で起動した VM に SSH/RDP でアクセスし、CloudCenter バンドル リポジトリと AMQP サーバにアクセスできることを確認します。

## CloudCenter の要件

ACI の拡張機能で使用される CCO は、対応する APIC エンドポイントにアクセス可能である必要があります。

## VMware vSphere の要件

要件	詳細
VMware vCenter 5.0/5.5/6.0 の動作環境	VMware vSphere の最小バージョンは v5.0 ですが、vSphere v5.5 U2 が最適です。
CloudCenter プラットフォームは、VMware のプライベート データセンターへの仮想マシンのプロビジョニングを自動化します。	CloudCenter プラットフォームは、vCenter 設定への <b>アクセス クレデンシャル</b> を必要とします。
すべての ESX ホストが ACI リーフ スイッチに物理的に接続されている必要があります。	データセンターに必要なインストール要件は次のとおりです。 <ul style="list-style-type: none"> <li>• 中規模のインスタンスを 10 以上実行できる物理 ESX ホスト</li> <li>• ESX クラスタ (クラスタが 1 つのホストのみで構成されている可能性もあります)</li> <li>• 100 GB 以上の空き領域があるデータストア (または DRS サポート用のデータストア クラスタ)</li> </ul>
ESXi ホストが Cisco UCS ベースの場合	<ul style="list-style-type: none"> <li>• CMM の VLAN を vNIC テンプレートにマッピングする必要があります。</li> <li>• ファブリックからのアップリンクは、リーフ スイッチへのトランキング VLAN と相互接続する必要があります。</li> </ul>

## 拡張機能の使用

ACI 拡張機能を作成して CCO の機能を拡張し、ACI 環境でネットワークをプロビジョニングすることができます。その後に、**拡張機能**を使用して次の CloudCenter リソースを設定できます。

- **導入環境フロー**: ACI の拡張機能は、導入環境でも統合されるため、各クラウド アカウントがその拡張機能を使用するかどうかを決定できます。CCM がクラウド プロバイダーに要求を行う必要はありません。詳細については、「[Deployment Environment Defaults](#)」を参照してください。
- **アプリケーションの導入レベル**: アプリケーションを導入するときにアプリケーション プロファイルに組み込まれるテナントと VMM のドメインを設定します。[外部ルーテッド ネットワーク (External Routed Network)] フィールド (レイヤ 3 アウト) を APIC 設定に合わせて設定すると、そのテナント ネットワークに接続することができます。

## USE ACI EXTENSION

ON 

## \* APIC EXTENSION

CiscoLabACI\_ByAdmin01

## \* VIRTUAL MACHINE MANAGER

VMM

## \* APIC TENANT

CliQr-Dev-3

## L3 OUT

External

- **NIC:** [Cisco ACI] タブを選択すると、[エンドポイント グループ (EPG) の種類 (Endpoint Group (EPG) Type)] からいずれかのオプションを任意で選択できます。

NIC 1

## NETWORK TYPE

VMware

Cisco ACI

## \* END POINT GROUPT (EPG) TYPE

New EPG

## \* BRIDGE DOMAIN

CliQr-BD

## CONTRACTS

CliQr-Dev-3/default

common/maple-l3-out

CliQr-Dev-3/imported

- [既存の EPG (Existing EPG)]: APIC セットアップから事前設定された EPG を使用します。
- [新しい EPG (New EPG)]: この導入用に新しい EPG を作成します。必要に応じて、この新しい EPG で使用される契約またはインターフェイスを選択することもできます。
- [ブリッジドメイン テンプレート (Bridge Domain Template)]: 選択したテンプレートを使用して新しいブリッジドメインを作成します。
- **外部サービスとしての ACI:** 外部サービスを含んでいるアプリケーションを導入する場合は、このサービス層の [詳細設定 (Advanced)] セクションで ACI 拡張機能を設定すると、APIC サービス グラフ テンプレートを使用することができます。

## Advanced

## ACI

## Extensions

APIC

## Tenants

CliQr-Dev-2

## Service Graph Template

f6

ACI service insertion between dependent service

# CloudHSM

- 概要
- 要件
- 参考資料

## 概要

CloudCenter プラットフォームは AWS クラウド ハードウェア セキュリティ モジュール (CloudHSM) をサポートします。このモジュールは、セキュア キーストレージを提供し、不正開封防止ハードウェア モジュール内での暗号操作を可能にするハードウェア アプライアンスです。



## 要件

CloudHSM を使用するには、システム管理者としてログインする際に、次の要件に従う必要があります。

- Luna プロバイダー (lunaProvider.jar ファイル) 用に CCM を設定し、このファイルが `/usr/local/tomcat/lib` ディレクトリにコピーされており、Tomcat がこのライブラリを使用できるようにします。Tomcat を再起動して、lunaProvider ライブラリが自動的に使用されるようにする必要があります。
- 各テナントは、一意の暗号キーが必要です。
- CloudHSM サーバとやり取りする CCM インスタンスは、CCM と同じ VPC 内に存在している必要があります。
- CloudHSM への接続を再確立する前に、CCM をリブートします。

## 参考資料

- Safenet Luna WebHelp
- AWS CloudHSM
- AWS CloudHSM 開始ガイド
- AWS CloudHSM フォーラム
- Connecting Multiple VPCs with EC2 Instances (SSL)

# SMTP メール サーバ

- 概要
- メール プロパティ ファイル
- SMTP 設定プロセス

## 概要

ここでは、SMTP メール サーバを設定し、CCM を使用して電子メールを送信する方法について説明します。



CloudCenter プラットフォームは、TLS ポートをサポートしていません。CloudCenter プラットフォームは、SMTP メール サーバの設定では SSL ポートのみをサポートしています。



パスワード文字列を暗号化形式で追加します。

## メール プロパティ ファイル

mail.properties ファイルは、ローカル Tomcat サーバ(/usr/local/tomcat/webapps/ROOT/WEB-INF/mail.properties)にあります。

```
# The hostname or IP address of your SMTP server
# Currently mob-gen.com email domain is hosted by gmail
# Gmail requires smtp over ssl, do not modify these settings
mail.smtp.host=smtp.gmail.com
mail.smtp.auth=true
mail.smtp.port=465
mail.smtp.socketFactory.port=465
mail.smtp.socketFactory.class=javax.net.ssl.SSLSocketFactory
mail.smtp.socketFactory.fallback=false

# Email user to authenticate to gmail
mail.user.number=1

mail.user.1=<your_osmosix_email_addr>
mail.password.1=<your_email_password>
from.mail.user.1=<your_cloudcentererrtech_email_addr>
from.mail.username.1=
```

## SMTP 設定プロセス

CCM の SMTP メール サーバを設定するには、次のプロセスを実行します。

1. mail.properties ファイルを開きます。  
**vi /usr/local/tomcat/webapps/ROOT/WEB-INF/mail.properties**
2. このファイル内で必要な設定を変更します。  
たとえば、Gmail を使用する場合は、次の行のみを変更します。

```
mail.user.1=<your_osmosix_email_addr>

mail.password.1=<your_email_password>

from.mail.user.1=<your_cloudcentertech_email_addr>

from.mail.username.1=
```

同様に、mail.properties ファイルで必要な行を変更して、CCM からメールにアクセスできるようにします。

3. Tomcat を再起動します。  
**/etc/init.d/tomcat restart**

## コールアウト スクリプト

- 概要
- 権限
- サポート対象のコールアウト トピック
- callout.conf ファイルでサポートされる属性
  - コールアウト スクリプトの環境変数
  - 各コールアウト スクリプトの設定
- vmNaming コールアウト スクリプト
- IPAM コールアウト スクリプト
  - サポート対象のプロパティ
  - OS 固有の IPAM プロパティ
  - IPAM コールアウト スクリプトの例
- ipamDealloc コールアウト スクリプト
- コールアウト ワークフローの例

## 概要

VM の導入ライフサイクルのさまざまな段階で、CloudCenter プラットフォームは、プロビジョニング プロセスの動作を制御する機能をサポートします。動作を制御する各種ライフサイクル ポイントを「トピック」と呼びます。動作はトピックに割り当てるコールアウト経由でスクリプトにより制御します。コールアウトに共通する使用例は、IP アドレスの割り当て (IPAM) トピック時に IP アドレスを取得し、IP シャットダウン トピック (ipamDealloc) 時に IP アドレスの割り当てを解除するための IPAM ツールの照会です。この使用例の実装例については、「[Infoblox の統合](#)」ページを参照してください。



コールアウトは **CCO** ごとに設定され、CCO からプロビジョニングされた各 VM に適用されます。異なる動作が必要な場合は、コールアウト スクリプト内で制御ロジック (if/then/case) を使用します。

各コールアウト スクリプトは次のようになります。

- 同じパラメータと入力変数を使用します。
- 別の変数を公開し、相互に排他的です。必要なときに任意のスクリプトを実行できます。
- クラウド タイプ、導入環境などのさまざまな環境変数にアクセスできます (CloudCenter プラットフォームから実行する場合)。



完全なリストは、`/usr/local/osmosix/callout/<name>/logs/<timestamp>` のコールアウト スクリプト ログで入手できます。

- 設定ファイル (callout.conf) と実行するスクリプトの 2 つの主要部分から構成されています。これらのファイルを CCO の `/usr/local/osmosix/callout/<script topic>/<files>` パスに配置します。使用するサブフォルダの名前は任意ですが、**ベスト プラクティスはそのコールアウトのトピックの名前を使用することです**。次に例を示します。  
`/usr/local/osmosix/callout/ipam/<files>`

## 権限

セキュリティを強化した実装の一部として、すべてのコールアウト スクリプトは **cliqruser** として実行されます。

既存のコールアウト スクリプトは変更しなくても動作し続けます。障害が発生した場合は、以下の項目を確認してください。

コールアウトの アクセス権	依存関係
ワークスペース権限	<b>chown -R cliqruser:cliqruser /usr/local/osmosix/callout</b>
スクリプトのシバン行	<ul style="list-style-type: none"> <li>• Bash: <b>#!/bin/bash</b></li> <li>• Python: <b>#!/usr/bin/python2.7</b></li> </ul>
<b>cliqruser</b> 権限	<ul style="list-style-type: none"> <li>• スクリプト内で使用するファイルおよびディレクトリに対する読み取りおよび書き込み機能</li> <li>• スクリプト内で使用する任意のコマンドを発行する機能</li> </ul>

## サポート対象のコールアウトトピック

これらのスクリプトのそれぞれについて、以下の項で説明します。



各スクリプトの説明に直接リンクしている前出の目次を使用してください。

スクリプト トピック	フォルダ	コールアウト スクリプト ファイル	説明	サポート対象のクラウド
<b>vmNaming</b>	/usr/local/osmosix/callout/vm naming/	/usr/local/osmosix/callout/vm naming/callout.conf	<p>各ノードを起動する前に呼び出されます。</p> <p>このスクリプトは、各ジョブにすべての名前変数(アプリケーションの名前、階層の名前、選択したイメージ)を提供(スクリプトに挿入)します。</p>	<p>AzureRM、OpenStack、VMware、AWS、および Alibaba</p> <p>サポート対象のクラウドでサポートされるコールアウトの詳細については、「<a href="#">VM (Node) Name Config</a>」を参照してください。</p> <p>▼ <a href="#">VMware の微妙な違い</a></p> <p>[インスタンスの命名戦略 (Instance Naming Strategy)] ドロップダウンの [ホスト名コールアウト (Hostname Callout)] オプションは、ゲスト OS 内に <b>osHostna me</b> と <b>vmName</b> を設定します。これら 2 つの設定は同じです。</p> <ul style="list-style-type: none"> <li>● vCenter 設定として</li> <li>● Linux (CentOS7) の場合。</li> </ul> <p>▼ <a href="#">Amazon の微妙な違い</a></p> <p>[インスタンスの命名戦略 (Instance Naming Strategy)] ドロップダウンの [ホスト名コールアウト (Hostname Callout)] オプションは、ゲスト OS 内に <b>osHostna me</b> と <b>vmName</b> を設定します。これら 2 つの設定は Windows と Linux 両方で同じです。</p> <p>▼ <a href="#">Alibaba の微妙な違い</a></p> <p>Windows と Linux 両方のインスタンスにおいて、文字列内のどの場所にもハイフン文字を使用することはできません。</p>
<b>ipam</b>	/usr/local/osmosix/callout/ipam/	/usr/local/osmosix/callout/ipam/callout.conf	ネットワークおよび OS に固有の設定	<p>OpenStack、VMware、および AWS。</p> <p>▼ <a href="#">VMware の微妙な違い</a></p> <p><b>osHostname</b> 設定:</p> <ul style="list-style-type: none"> <li>● IPAM コールアウトの場合は必須ではありません。</li> <li>● Windows でのみ動作します。</li> <li>● Linux の設定は <b>vmNaming</b> 設定で上書きされます。</li> </ul> <p>▼ <a href="#">Amazon の微妙な違い</a></p> <p>CloudCenter は、<b>IP address</b> と <b>network</b> と <b>mask</b> を使用して、指定したサブネット内に DHCP スコープを設定します。</p>
<b>ipamDealloc</b>	/usr/local/osmosix/callout/ipamDealloc/	/usr/local/osmosix/callout/ipamDealloc/callout.conf	ノードが破棄される直前に呼び出されます。	OpenStack、VMware、および AWS。

## callout.conf ファイルでサポートされる属性

このコールアウト スクリプトは、各行で <key>=<value> を使用して、標準的な Java プロパティ ファイル形式をサポートします。3 つのトピックすべてについて、次の callout.conf ファイルの例を参照してください。

- **topic=vmNaming**

```
[root@centos7-base ipam]# cat ../vmnaming/callout.conf
name=vmNamingExample
type=exec
topic=vmNaming
debug=true
executable=run.sh
reinject=true
disabled=false
```

- **topic=ipam**

```
[root@centos7-base ipam]# cat callout.conf
name=ipamExample
type=exec
topic=ipam
debug=true
executable=ipam.py
einject=true
disabled=false
```

- **topic=ipamDealloc**

```
[root@centos7-base ipam]# cat ../ipamDealloc/callout.conf
name=ipamDeallocExample
type=exec
topic=ipamDealloc
debug=true
executable=run.sh
reinject=true
disabled=false
```

## コールアウト スクリプトの環境変数

コールアウト スクリプトは入力パラメータとして環境変数を受け入れます。変数のリストはノードの種類によって異なります。次の表に例を表示します。

変数	サンプル値または種類
eNV_osName	Linux、Windows
eNV_vmNaming	vmNaming モジュールによって渡された文字列、または CloudCenter プラットフォームによって自動生成された文字列
eNV_JOB_ID	アプリケーション VM(のみ)を識別する整数
eNV_launchUserId	スクリプトまたはモジュールを起動する担当者のユーザ ID を識別する整数
eNV_launchUserName	スクリプトまたはモジュールを起動する担当者のユーザ名を識別する文字列



アプリケーション VM に対するすべてのジョブ アプリケーション設定も、eNV 変数として使用できます。詳細については、「[CloudCenter-Defined Parameters](#)」を参照してください。

**ベスト プラクティス**

デバッグ レベルをオンにしてデバッグ ログを確認し(「[Locate Log Files](#)」を参照)、使用可能なすべての変数のリストを表示します。

## 各コールアウト スクリプトの設定

callout.conf ファイルで各スクリプトを個別に設定します。

これらのコールアウト スクリプトそれぞれを、**テナント レベルではなく**、リージョン レベルで CCO ごとに設定できます。次に、vmNaming コールアウト スクリプトを追加する設定手順の例を示します。



クラウド リージョンのコールアウトまたは属性に変更を加えると、変更を有効にするために CCO を再起動する必要があります。

コールアウト スクリプトは /usr/local/osmosix/callout/callout\_name/ フォルダにあります。callout\_name は対応するコールアウトの名前です。

コールアウト スクリプトを追加するには、次のプロセスに従います。

1. CCO に次のディレクトリを作成します。

```
/usr/local/osmosix/callout/vmnaming/
```

2. このディレクトリに次のファイルを作成します。

```
/usr/local/osmosix/callout/vmnaming/callout.conf
```

3. スクリプトのファイルを作成します。

```
/usr/local/osmosix/callout/vmnaming/<script name>
```

4. アクセス許可を必ず実行します。

```
chmod 777 <script>
```

5. callout.conf ファイルでこのファイルを参照します。

## vmNaming コールアウト スクリプト

vmNaming スクリプトのサポート対象の環境変数:

変数	サンプルの値またはタイプ
eNV_JOB_ID	integer(アプリケーション VM のみ)
eNV_launchUserId	integer
eNV_launchUserName	string

vmNaming スクリプトのサポート対象のキー:

CloudCenter の必須キー	説明
vmName	VM の名前

VM 命名コールアウト スクリプトの出力例:

```
run.sh

#!/bin/bash

echo "vmName=`uuidgen`"
```

このスクリプトで VM の名前を変更することができます。サポート対象のクラウド(上記の表を参照)のいずれかを使用して VM の名前を変更するには、「VM (Node) Name Config」を参照してください。

## IPAM コールアウト スクリプト



「VM Name Config」>「Instance IPAM Strategy」セクションで示されているように、IPAM 命名戦略のリージョンを有効にする必要もあります。

統合の一環として、IPAM を作成し、CCO の起動時に動的に呼び出されるコールアウト スクリプトを組み込みます。モジュールは動的にロードまたはリロード(自動ロード)したり、CCO 起動時にロードしたりできます。デフォルトでは、自動ロードは無効になっています。

IPAM モジュールのコールアウト スクリプトには、次のようなパラメータが含まれています。

- DNS サーバリスト
- DNS サフィックス リスト
- vNIC の数
- vNIC の IP アドレスの数
- vNIC のネットマスクの数
- VM 名

スクリプトを実行すると、そのクラウドに対するすべての導入で IPAM モジュールが管理する IP アドレスが検出されます。

IPAM コールアウト スクリプトのオプションは、VMware および OpenStack でサポートされます。

▼ [Amazon の微妙な違い](#)

CloudCenter は、**IP address** と **network** と **mask** を使用して、指定したサブネット内に DHCP スコープを設定します。

コールアウト スクリプトのパスは /usr/local/osmosix/callout で、各モジュールはスクリプト パスの下のサブフォルダです。

### 例

```
UserClusterName="cluster01"

eNV_Cloud_Setting_UserDataCenterName="dc02"

eNV_NumTasks="1" eNV_UseBatchTaskList="0"

eNV_Cloud_Setting_UserResourcePoolName="resourcepool1"

eNV_Cloud_Setting_UserClusterName="cluster01"
```

## サポート対象のプロパティ

各コールアウト スクリプトに対して出力される複数のキーと値のペア。

キー	説明	必須かどうか
osHostname	OS ホスト名	○ AWS および OpenStack の場合はサポートされません。
DnsServerList	DNS サーバリスト(カンマ区切り)	
DnsSuffixList	DNS サフィックス リスト(カンマ区切り)	
nicCount	仮想 NIC (vNIC) の数	○ nicCount パラメータでは、一度に 1 つの NIC の使用のみが許可されます。

nicIP_n	N 番目の vNIC の IP アドレス	○
nicNetmask_n	N 番目の vNIC のネットマスク	○
nicGateway_n	N 番目の vNIC のゲートウェイ (CCO) IP アドレス	○
nicDnsServerList_n	N 番目の vNIC の DNS サーバリスト (カンマ区切り)	○
nicUseDhcp_0	IPAM スクリプトの一部として、nicIP_n と nicDnsServerList_n にダミー値を指定します。ただし、これらの値は DHCP の設定によって上書きされます。	IPAM コールアウトを使用しており、アドレッシングが DHCP を使用するように割り当てられている場合は必須。
ANY	このプロパティは、reinject の設定が true の場合にサポートされます。  例: myCustomParam=myValue	




VM 設定に複数の NIC が含まれている場合、CloudCenter プラットフォームは NIC ごとに 1 つの IPAM コールを実行します。前述のとおり、\_n サフィックスが付いたキーを使用することにより、各 NIC に複数の IP を割り当てることもできます。

## OS 固有の IPAM プロパティ

各コールアウト スクリプトに対して出力される複数のキーと値のペア。

IPAM プロパティ	Linux	Windows	必須かどうか
timezone	VMware の場合は <i>サポートされません</i> 。  AWS および OpenStack の場合は <i>サポートされません</i> 。	未使用	○
timeZoneld	未使用	このタイム ゾーンの Windows インデックス ID。  <div>  Windows に固有の VMware IPAM 設定スクリプトの場合、導入完了後未定の期間が経過した後でのみ、変更内容が有効になることに注意してください。 </div> AWS: インスタンスのタイミングは内部的に管理されるため、効果はありません。	○
fullName	不要	管理者ユーザの名前	○
organization	不要	組織の名前 (文字列)	○
productKey	不要	Windows のプロダクト キー	○
setAdminPassword	不要	管理者パスワード	○
changeSid	未使用	Microsoft SID に対する true または false の値  changeSid オプションを <b>true</b> に設定する必要があります。	○
deleteAccounts	未使用	true または false の値	○
dynamicPropertyName	未使用	任意のプロパティの予約済み名前ホルダー	○
dynamicPropertyValue	未使用	任意のプロパティの予約済み値ホルダー	○
custSpec	VMware のゲスト カスタマイズ仕様名	VMware のゲスト カスタマイズ仕様名	なし
hwClockUTC	VMware および OpenStack の場合は <i>サポートされません</i> 。  AWS: クロックは内部的に管理されるため、効果はありません。	VMware および OpenStack の場合は <i>サポートされません</i> 。  AWS: クロックは内部的に管理されるため、効果はありません。	○

domainName	Linux ドメイン名  <div> domainName を指定しない場合は、CloudCenter プラットフォームでデフォルト値の <b>mydomain</b> に設定します。  デフォルト値の <b>mydomain</b> を使用しない場合は、この値を必ず指定してください。  AWS および OpenStack の場合はサポートされません。</div>	自動的にドメインに参加するために使用	○
domainAdminName	未使用	自動的にドメインに参加するために使用	
domainAdminPassword	未使用	自動的にドメインに参加するために使用	
workgroup	未使用	VM を配置するワークグループ。  3 つのドメイン値のいずれかが欠落している場合は、ワークグループ キーが必要です。  3 つのドメイン値がすべて存在する場合、ワークグループは必要ありません。	

## Windows に固有の例

**run.sh**

```
#!/bin/bash

echo "setAdminPassword=abcd"
echo "timeZoneId=10 *"
echo "fullName=Enterprise ABCD"
echo "organization=ABCD"
echo "productKey=..."
echo "changeSid=true"
```

## IPAM コールアウト スクリプトの例

**run.sh**

```
#!/bin/bash

echo "DnsServerList=8.8.8.8,10.0.0.100"

echo "nicCount=2"
echo "nicIP_0=10.0.0.100"
echo "nicDnsServerList_0=1.2.3.4,5.6.7.8"
echo "nicGateway_0=10.0.0.1"
echo "nicNetmask_0=255.255.255.0"

echo "nicIP_1=192.1.0.100"
echo "nicDnsServerList_1=10.10.10.10"
echo "nicGateway_1=192.1.0.1"
echo "nicNetmask_1=255.255.255.0"

echo "domainName=test.org"
echo "hwClockUTC=true"
echo "timeZone=Canada/Eastern"
echo "osHostname=testhost1"
```

## ipamDealloc コールアウト スクリプト

ipamDealloc スクリプトでは、環境をクリーンアップできるようにし、reinject 設定によってサポートされているカスタム プロパティのみと連動します。

ipamDealloc の例:

### run.sh

```
#!/bin/bash

./delete_record_by_ip.sh $IP
```

このスクリプトからの出力は通知にすぎないため、CloudCenter プラットフォームはこの出力を検索しません。

## コールアウト ワークフローの例

### IPAM コールアウトの例

```
#!/usr/local/bin/python2.7
import infoblox, sys, requests, os, random
requests.packages.urllib3.disable_warnings()

#Assign command line arguments to named variables
hostname = os.environ['vmNaming']
domain = "vm.cloudcenter.com"
fqdn = hostname + "." + domain
network = "10.49.18.0/23" #sys.argv[2]
netmask = "255.255.254.0"
gateway = "10.49.19.254"
dns_server = "10.48.112.33,10.52.112.19"

#Setup connection object for Infoblox
iba_api = infoblox.Infoblox('10.49.9.163', 'admin', 'infoblox', '1.4',
iba_dns_view='VM-view', iba_network_view='default', iba_verify_ssl=False)

try:
    #Create new host record with supplied network and fqdn arguments
    ip = iba_api.create_host_record(network, fqdn)
    print "DnsServerList="+dns_server
    print "nicCount=1"
    print "nicIP_0=" + ip
    print "nicDnsServerList_0="+dns_server
    print "nicGateway_0="+gateway
    print "nicNetmask_0="+netmask
    print "domainName="+domain
    print "HWClockUTC=true"
    print "timeZone=Canada/Eastern"
    print "osHostname="+hostname
    print "infobloxFQDN="+fqdn
except Exception as e:
    print e
```



# Infoblox

- 概要
- IPAM 統合の前提条件
- IPAM モジュールの設定
- Infoblox API
- コールアウト設定ファイル

## 概要

CloudCenter プラットフォームは、IP アドレス管理 (IPAM) の統合をサポートし、導入に向けての IP アドレスの管理を行います。

ここでは、CCO 上での複数のコールアウト スクリプトの実行による、IPAM プロバイダーである [Infoblox](#) との統合について説明します。各コールアウト スクリプトの詳細については、「[コールアウト スクリプト](#)」を参照してください。

CloudCenter プラットフォームが IP アドレスと DNS 名の割り当てについてどのように Infoblox をサポートするかの簡単なデモンストレーションについては、Infoblox の[統合に関するビデオ](#)を参照してください。

## IPAM 統合の前提条件

クラウド サポートの詳細については、「[コールアウト スクリプト](#)」を参照してください。



クラウド リージョンのコールアウトまたは属性に変更を加えると、変更を有効にするために CCO を再起動する必要があります。

## IPAM モジュールの設定

1. モジュール テンプレートを取得して、`/usr/local/osmosix/callout` に保存するには、[CloudCenter サポート](#)にお問い合わせください。
2. テスト環境に応じて、コールアウト スクリプトを変更します。
3. スリープ ジョブをモデル化し、環境変数を追加し、ジョブを実行してコールアウト ログを確認し、変数が正しくエクスポートされることを確認します。
4. 多層 Web アプリケーションをモデル化し、環境変数を追加し、ジョブを実行してコールアウト ログを確認し、変数が正しくエクスポートされることを確認します。

## Infoblox API

Infoblox モジュールと統合するには、[Infoblox-API-Python](#) モジュールを使用します。

次に、Infoblox-API-Python モジュールを使用するスクリプトの例を示します。このスクリプトには、**python-requests 2.5** が必要です。また、このスクリプトはコールアウトから直接呼び出すことができます。

### createHost.py

```
#!/usr/local/bin/python2.7
import infoblox, sys
#Check to see if command line included enough arguments.
if (len(sys.argv) < 3):
    print "Usage: createHost.py <fqdn> <network CIDR>"
    quit()
#Assign command line arguments to named variables
fqdn = sys.argv[1]
network = sys.argv[2]
#Setup connection object for Infoblox
iba_api = infoblox.Infoblox('10.110.1.45', 'admin', 'infoblox', '1.6', 'default',
'default', False)
try:
    #Create new host record with supplied network and fqdn arguments
    ip = iba_api.create_host_record(network, fqdn)
    print "nicCount=1"
    print "nicIP_1=" + ip
except Exception as e:
    print e
```

## コールアウト設定ファイル

次に、コールアウトと Infoblox アプリケーションの統合の例を示します。

### callout.conf

```
name=infoblox
type=exec
topic=ipam
debug=true
executable=createHost.py
reinject=true
disabled=false
```

## Jenkins

- 概要
- [CloudCenter Jenkins プラグイン](#)
- 前提条件
- [CloudCenter Jenkins プラグインのインストール](#)
- [jenkinsBuildId マクロ](#)
- [すべてのビルドでの新規導入の作成](#)
- [既存の導入の更新](#)

## 概要

事前にモデル化された Jenkins プロジェクト (Git/SVN からソースコードを取得する Maven など) では、CloudCenter Jenkins プラグインを使用して CloudCenter と統合できます。

このファイルを手動でコピーする必要はありません。Jenkins のユーザがこのプラグインを使用できるように、CloudCenter がダウンロード URL を提供します。ダウンロード場所については、[CloudCenter サポート](#)にお問い合わせください。

## CloudCenter Jenkins プラグイン

CloudCenter Jenkins プラグインは、[サポート対象のクラウド](#)上での導入をユーザが Jenkins サーバから直接開始できるようにして、Jenkins と CloudCenter 間の完全な統合を実現します。

さらに、ユーザは各階層に対してアップグレード スクリプトを指定することで、既存の導入をアップグレードできます。

## 前提条件

- Jenkins に慣れていない場合は、ソース リポジトリとして Github を使用し、Jenkins で Maven プロジェクトを設定します。詳細については、<http://www.youtube.com/watch?v=ITQGi5jzjvo> を参照してください。
- CloudCenter Jenkins プラグインを使用するには、サポート対象の Jenkins バージョン値が **1.624 以上** である必要があります。
- Jenkins サーバに必要な Java のバージョンは **Java 7** である必要があります。
- デフォルトのクラウド設定を **環境内** で設定して、アプリケーションを展開するときに Jenkins プラグインがそれらのクラウド設定を取得して使用できるようにする必要があります。
- CloudCenter 4.8.x の Jenkins プラグインは、CloudCenter 4.8.x に関連付けられた API を必要とします (たとえば、Jenkins プラグインはジョブ導入に v2 CloudCenter API を必要とし、v2 API は CloudCenter 4.8.x でのみ使用可能です)。
- Jenkins サーバと CCM の間で正常に接続できるようにするために、必ず CCM からの証明書を Jenkins の信頼できる Java キーストアにインポートします (詳細については、「[Certificate Authentication](#)」を参照)。

## CloudCenter Jenkins プラグインのインストール

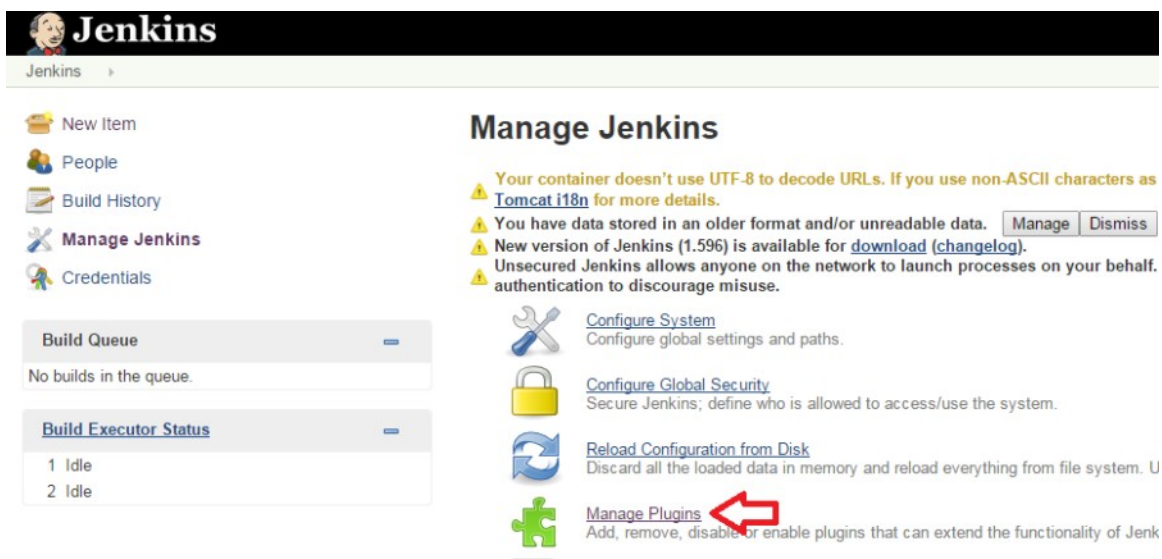
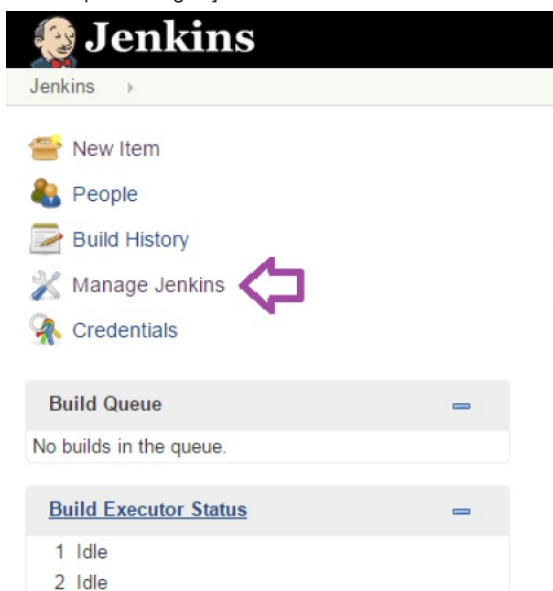
CloudCenter Jenkins プラグインをインストールするには、次の手順を実行します。

- [CloudCenter サポート](#)に問い合わせ、CloudCenter Jenkins プラグインのダウンロード リンクを取得します。
- 管理者クレデンシャルを使用して CCM にログインします。
- Jenkins ユーザ用に **API 管理(アクセス)キー**を生成します。



- アプリケーションをモデル化し、このユーザが Jenkins ビルド サーバからアーティファクトにアクセスできるようにします。

5. Jenkins で、[Jenkins の管理 (Manage Jenkins)] > [プラグインの管理 (Manage Plugins)] > [詳細設定 (Advanced)] > [プラグインのアップロード (Upload Plugin)] に移動し、CloudCenter Jenkins プラグインをアップロードして、インストールします。



The screenshot shows the Jenkins Plugin Manager interface. The top navigation bar includes 'Jenkins' and 'Plugin Manager'. The left sidebar has links for 'Back to Dashboard' and 'Manage Jenkins'. The main content area has tabs for 'Updates', 'Available', 'Installed', and 'Advanced'. The 'Advanced' tab is active, displaying the 'HTTP Proxy Configuration' section with input fields for 'Server', 'Port', 'User name', 'Password', and 'No Proxy Host'. Below this is a 'Submit' button. The 'Upload Plugin' section follows, with a red arrow pointing to the 'Choose File' button. The text below the button reads: 'You can upload a .hpi file to install a plugin from outside the central plugin repository.' The 'File:' field shows 'CliQrJenkinsClient.hpi' and an 'Upload' button is at the bottom.

The screenshot shows the Jenkins Update Center interface. The top navigation bar includes 'Jenkins' and 'Update center'. The left sidebar has links for 'Back to Dashboard', 'Manage Jenkins', and 'Manage Plugins'. The main content area is titled 'Installing Plugins/Upgrades'. It shows a progress bar for 'CliQrJenkinsClient' with a 'Success' status. Below the progress bar, there are two links: 'Go back to the top page (you can start using the installed plugins right away)' and 'Restart Jenkins when installation is complete and no jobs are running'.

6. CliQrJenkinsPlugin をインストールした後に、既存または新規のプロジェクトに移動してポストビルド ステップを設定し、Maven プロジェクトを使用して Git/SVN からソースコードを Jenkins ビルドに取得します。

Jenkins > gg > configuration

### Build History

[trend](#)

#2	Jan 7, 2015 7:50 PM
#1	Jan 7, 2015 7:48 PM

[RSS for all](#)
[RSS for failures](#)

☐ Disable build (no new builds will be executed until the project is re-enabled.)  
☐ Execute concurrent builds if necessary

### Advanced Project Options

### Source Code Management

☒ None  
☐ CVS  
☐ CVS Projectset  
☐ Subversion

### Build Triggers

☐ Build after other projects are built  
☐ Build periodically  
☐ Poll SCM

### Build

Add build step ▾

CliQr Application Deployment Client

Execute Windows batch command

Execute shell

Invoke Ant

Invoke top-level Maven targets

- CloudCenter アプリケーション導入クライアントを Jenkins に設定して、ビルド システムからの統合と導入 (新規または既存ノードでのアップグレード) を続行します。

Jenkins > Deploy\_PetClinic > configuration

AccessKey:  [?](#)  
CliQr REST API key for the above User [Validate Connection](#)

☒ Deploy to Project

Project Name:

Project Phase:

CliQr Project's Phase:  [?](#)

Application Version:  [?](#)

Deployment Environment:  [?](#)

CliQr's Deployment Environment:  [?](#)

Cloud Type:

Cloud Account:

Instance Type:

Tags:

AppParameters:  [?](#)  
separated list of Key Value pairs which will be available in userenv of the VM's. Sample Input key=value.key1=value1. You can also create a file with name as appParams under \$WORKSPACE with multiple lines of key=value which will be passed automatically.

### Application Deployment Configuration

[Add another Tier Deployment Configuration](#)

**Choose to copy your binaries**

☐ Copy Binaries to External Location  
☒ Don't copy Binaries


**Choose the Deployment Behavior**

☒ Create a new Deployment on every Build  
☐ Stop Old Deployment if exists  
☐ Update Existing Deployment (Create new if doesn't Exist)  
☒ Wait For Deployment And Export Details

Waits for Deployment to go into Running state and exports CliQr Job Details into \$WORKSPACE/userenv file

[Add build step ▾](#)  
[Save](#) [Apply](#) [Delete](#)

[CloudCenter アプリケーション導入クライアント (CloudCenter Application Deployment Client)] ページのパラメータのリストを次の表に示します。

パラメータ	説明
Cisco CloudCenter CCM URL	CCM サーバの IP アドレス。信頼できる証明書 (詳細については、「 <a href="#">Certificate Authentication</a> 」を参照してください) が CCM サーバに追加されていることを確認します (自己署名証明書ではなく)。
[ユーザ名 (Username)]	[Manage Access Key] セクションにリストされたユーザ名 (詳細については「 <a href="#">API Authentication</a> 」を参照してください)。
AccessKey	[アクセスキーの管理 (Manage Access Key)] セクションにリストされたユーザの API 管理キー (詳細については、「 <a href="#">API Management Key</a> 」を参照してください)。
Deploy to Project	このフラグを使用して、一般的な導入環境ではなく、プロジェクトに展開します (詳細については、「 <a href="#">Projects and Phases</a> 」を参照してください)。
Project Name	[Deploy to Project] フラグを選択した場合は、このパラメータが表示され、プロジェクトのリストが CCM サーバから取得されます。このパラメータ値に基づいて、このプロジェクトに関連付けられているアプリケーションのみが、[Application Version] フィールドから除外されます。
Project Phase	プロジェクト名に基づいて、そのプロジェクトのすべてのフェーズのドロップダウン リストが表示されます。[Deployment Environment] フィールドにこの値を使用し、プロジェクトをフィルタリングします。
アプリケーション	CCM UI にリストされたアプリケーションのドロップダウン リストから、導入に必要なアプリケーションを選択します。 <div> クレデンシャルを入力した後は、<b>アプリケーション管理 API</b> のロードにしばらく時間がかかる場合があるため、その間は待機します。</div>
アプリケーションバージョン (Application Version)	アプリケーションに基づいて、このドロップダウン リストからアプリケーションのバージョンを選択します。
Deployment Environment	アプリケーションの導入に必要な導入環境を選択します。CloudCenter は各導入のデフォルト設定 (デフォルト クラウド、デフォルト インスタンス タイプなど) を使用するため、これらのデフォルト設定すべてを必ず確認してください。
Cloud Type	導入環境に存在するクラウド タイプのドロップダウン リストからクラウド タイプを 1 つ選択します。
Cloud Account	アプリケーションを展開するクラウド アカウント。
Instance Type	デフォルトのインスタンス タイプとして単一インスタンス タイプを選択します。
タグ	このジョブに関連付けられているタグのカンマ区切りのリスト。
AppParameters	グローバル パラメータとして渡すキーと値のペアのカンマ区切りのリスト。例: abcd=wow, cdef=cliqrRocks  CloudCenter には \$BUILD_ID、\$BUILD_NUMBER、\$BUILD_TAG、\$JOB_NAME、および使用可能な場合は他のプラグインの \$BUILD_TIMESTAMP が含まれています。  このセクションで変数を追加して、他のジョブが共有しているパラメータを取得することができます。例: abc=\${BUILD_PARENT_NUMBER} または abc=\$BUILD_PARENT_NUMBER  また、キーと値のパラメータ ペアが複数行含まれている \$WORKSPACE/appParams ファイルからパラメータを取得するオプションもあります。それらのパラメータを使用すると、CCM UI に表示せずにパスワードやその他の機密情報を渡すことができます。
Choose the binaries to be Copied	ファイルを外部の場所へコピーする必要があるかどうかを識別します <ul style="list-style-type: none"><li>[Copy Binaries to External Location]: すべてのバイナリを /tmp/app1/latest で使用できます</li><li>[Don't copy Binaries]: バイナリは他の場所にコピーされません</li></ul>

Choose the Deployment Behavior	<p>導入に対して一度しか実行できません。</p> <ul style="list-style-type: none"> <li>[Create a new Deployment on every Build]: このオプションではすべてのビルドにまったく新しい導入を作成します。</li> <li>[Stop Old Deployment if exists]: このオプションではすべてのビルドにまったく新しい導入を作成し、古い導入が存在すれば停止します。</li> <li>[Update Existing Deployment (Create new if doesn't Exist)]: <ul style="list-style-type: none"> <li>同じプロジェクトの以前のビルド時に CloudCenter Jenkins プラグインから起動した以前の導入を更新します。</li> <li>以前の導入が進行中で(ジョブ)、実行状態になっていない場合、CloudCenter Jenkins プラグインは導入が実行状態になるまで待機してから更新をトリガーします(詳細については「<a href="#">Deployment and VM States</a>」を参照してください)。</li> <li>以前の導入がエラーとなった場合、またはその導入が CCM サーバから停止またはキャンセルされた場合、CloudCenter Jenkins プラグインは更新プロセスの一部として新しい導入を起動します。</li> <li>これが最初のビルドである場合は、CloudCenter Jenkins プラグインが新しい導入を作成し、次の成功したビルドから既存の導入を使用します。</li> <li>UpdateScripts: ここに記載する順序で実行される tierName:scriptToExecute スクリプトのカンマ区切りのリスト。 例: AppCluster:/shared/app/petclinic/update.sh, Database:/shared/app/updatemysql.sh, AppCluster:/shared/app/startServer.sh</li> </ul> </li> <li>[Wait For Deployment And Export Details]: 導入が実行状態になるのを待機してから、ジョブの詳細を \$WORKSPACE/userenv ファイルにエクスポートします</li> </ul>
--------------------------------	---

## jenkinsBuildId マクロ

更新スクリプトでは、更新導入時にコピーするバイナリをポイントする引数として、\$BUILD\_ID または %jenkinsBuildId% を渡すことができます。



%jenkinsBuildId% マクロは、アプリケーションに固有のマクロではありません。この CloudCenter で定義されたマクロは、Jenkins プラグインを使用して起動する導入に適用されます。

jenkinsBuildId マクロは、アプリケーションの導入の userenv に Jenkins のビルド ID を渡すために主に使用されます。Jenkins プラグインがトリガーする導入では userenv に自動的に jenkinsBuildId が設定され、リポジトリ/ストレージ内の正しいバイナリをポイントするために使用されます。たとえば、Web サーバに /shared/app/petclinic/latest/ に設定された以前の war ファイル パスがある場合、この war ファイル (petclinic.war) はこのマクロを使用して /shared/app/petclinic/%jenkinsBuildId%/petclinic.war をポイントできるようになります。

更新導入の場合、jenkinsBuildId マクロは既存の導入に渡す必要がある値を変更します。これは、userenv に導入時の古い jenkinsBuildId があるためです。

CloudCenter がターゲット ロケーションで作成するフォルダ名は、ランダムなタイムスタンプ値ではなく、jenkinsBuildId 値を使用するようになります。

## すべてのビルドでの新規導入の作成

このオプションは、すべてのビルドにまったく新しい導入を作成します。

## 既存の導入の更新

既存の導入を更新する場合は、次のようになります。

- 同じプロジェクトの以前のビルド時に Jenkins プラグインから起動した以前の導入を更新します。
- 以前の導入ジョブがまだ進行中で、実行状態になっていない場合、プラグインは実行状態になるまで待機してから更新をトリガーします。
- 以前の導入がエラー状態で終了するか、またはその導入が CCM から停止またはキャンセルされた場合、プラグインは更新の一部として新たな導入を起動します。
- 初めてのビルドの場合、このプラグインは新規導入を作成し、次に成功したビルドには、この既存の導入を使用します。
- セキュリティ グループ エラーになった場合は、複数のシステムが同じアカウントにアクセスしているかどうかを必ず確認してください。

## SSO

- SSO AD
- SAML SSO
- 使用例: ADFS SAML SSO



## SSO AD の統合

- 概要
- ユーザ認証
- 削除されたユーザの処理

### 概要

一部の企業には独自の Active Directory (AD) や、その他の類似するセットアップがあり、それらのクレデンシャルを使用して外部アプリケーションやプラットフォームにログインする場合があります。CloudCenter は直接的な AD 認証をサポートしていません。その代わりに、サービス プロバイダー (SP) としての CloudCenter と、ADFS などのお客様の ID プロバイダー (IDP) との間でシングル サインオン (SSO) を使用して統合をサポートします。

CloudCenter は各ベンダーがテナントとしてモデル化されているマルチテナント モデルをサポートします。テナントにはルート階層ツリーの構造が 1 つあります。各テナントには、専用のユーザ セットがあります。それらのユーザの 1 人がテナント管理者 (ルート管理者またはプラットフォーム管理者とも呼ぶ) であり、特殊な管理者権限を持っています。



**CloudCenter は LDAP や AD に対して直接認証されません。**

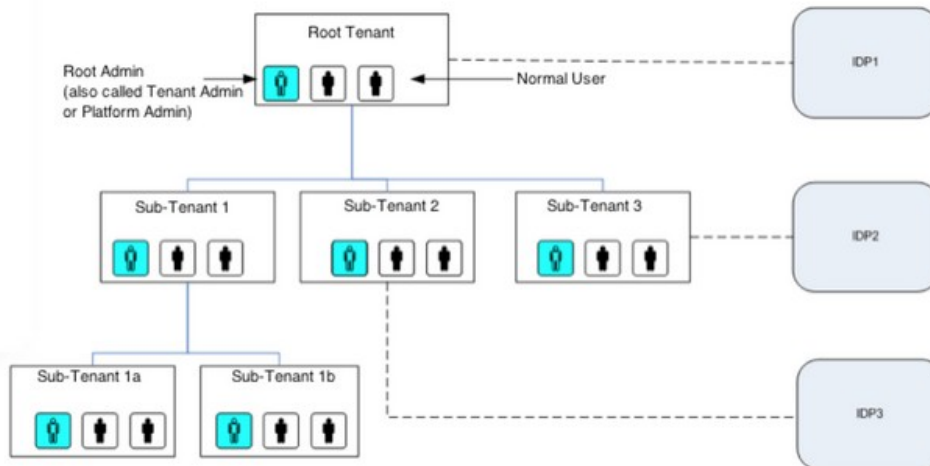
CloudCenter は、SAML 2.0 プロトコルをサポートする SSO ID プロバイダー (IDP) を通じて LDAP や AD とのみやり取りします (Ping ID、ADFS、Shibboleth など)。

CloudCenter を使用して SSO を実装するには、次の必要があります。

1. CCM を設定して、認証を SSO IDP にリダイレクトする必要があります。
2. また、ユーザ アクティベーション プロファイルに対して、(SAML IDP によって返された) 追加のユーザのカスタム プロパティをマッピングする必要もあります。
3. これらのステップをすべて正常に実行すると、CloudCenter は適切なユーザ グループ メンバーシップと追加のロールおよび権限を自動的に割り当てます。

### ユーザ認証

外部 ID プロバイダー (IDP) によって認証されるユーザ (ユーザ X) であっても、ユーザ X には CCM VM のユーザ データベース内に対応する存在が必要です。SSO 環境では、ユーザ X が IDP によって認証され、初めて CCM VM を使用すると、プラットフォーム管理者がテナントとテナント管理者を作成していた場合においては、ユーザ X の認証が CCM ユーザ データベース内に作成されます。



各テナントは、次のように独自の SSO をポイントできます。

- 専用のエイリアス ホスト名を持ち、外部 IDP を使用してユーザを認証するように、各テナントを設定できます。
- 各テナントとユーザには、外部組織とユーザに関連付けるための `externalId` があります。

### 削除されたユーザの処理

IDP データベースからユーザを削除すると、削除されたユーザは CloudCenter にはログインできませんが、設定や関連付けられた依存関係は CloudCenter プラットフォームにそのまま維持されます。

## SAML SSO の統合

- 概要
- CloudCenter サポート
- SAML 認証設定
- 参考資料

### 概要



**CloudCenter は LDAP や AD に対して直接認証されません。**

CloudCenter は、SAML 2.0 プロトコルをサポートする SSO ID プロバイダー(IDP)を通じて LDAP や AD とのみやり取りします(Ping ID、ADFS、Shibboleth など)。

CloudCenter を使用して SSO を実装するには、次の必要があります。

1. CCM を設定して、認証を SSO IDP にリダイレクトする必要があります。
2. また、ユーザ アクティベーション プロファイルに対して、(SAML IDP によって返された)追加のユーザのカスタム プロパティをマッピングする必要もあります。
3. これらのステップをすべて正常に実行すると、CloudCenter は適切なユーザ グループ メンバーシップと追加のロールおよび権限を自動的に割り当てます。

CCM インスタンスは Security Assertion Markup Language(SAML)2.0 SSO を Spring Security SAML Extension を通じてサポートします。

システム管理者(**SysAdmin**)は、ルートレベルまたはテナントレベルで SAML 統合を設定できます。この統合を正確に設定するには、導入に応じて、ルート テナントまたはサブテナントに関する次の情報が必要です。

- テナント設定情報
- アクティベーション プロファイル情報

### CloudCenter サポート

詳細については、[CloudCenter サポート](#)にお問い合わせください。

### SAML 認証設定

SSO を使用するようにテナントを設定するには、次の手順を実行します。

1. テナントを作成します(「[Sub-Tenant Configuration](#)」を参照)
  - a. [短縮名 (Short Name)]: 空白文字および特殊文字を使用せずに文字列を入力します。
  - b. [外部 ID (External Id)]: テナントが関連付けられている外部システムの組織の ID を入力します。
  - c. [テナント (Tenant)]: テナントの CCM サーバドメイン名エイリアス。これは、SSO の観点から、サービス プロバイダー (SP) のエンドポイントとして機能します。
2. 新しく作成したテナント管理者としてログインし、[アクティベーション プロファイル](#)を作成します。
3. [テナント情報 (Tenant Info)] タブをクリックし、新しく作成したアクティベーション プロファイルをデフォルトのアクティベーション プロファイルとして選択します。
4. IDP 情報を収集して、IDP が提供するサブジェクト属性を確認し、ユーザ プロパティのマッピング ブランに対して IDP サブジェクト属性を作成します。
5. システム管理者 (SysAdmin) としてログインし(「[Admin Users](#)」 > 「[Login as a System Admin](#)」を参照)、[ベンダー管理者の管理 (Manage Vendor Admins)] タブをクリックし、[認証設定 (Authentication Settings)] アクション項目を選択します。
6. [IDP 設定 (IDP Settings)] セクション、[SP 設定 (SP Settings)] セクション、および [属性マッピング (Attribute Mappings)] セクションに情報を入力して [送信 (Submit)] をクリックします。

**SAML Authentication Settings**

**Vendor Information**

Vendor Name: Training Tenant

**IDP Settings**

IDP Name: your\_idp  
Provide the string to identify the IDP.

IDP Metadata URL: http://shibdev.ciqr.com/ldp/shibboleth  
Provide the URL for the IDP metadata.

IDP Metadata File: <idpname>-metadata.xml  
If no IDP metadata URL is provided, specify the path of the IDP metadata file.

**SP Settings**

Entity ID: your.company.com  
Provide target authentication domain name.

Default SSO Binding: post

Logout Target URL: http://your.company.com  
Provide the URL that will be redirected to after logout.

**Attribute Mappings**

Mapped by Friendly Name: ☐

First Name: cn

Last Name: sn

Company Name: company

Email: email

User Group:

Activation Profiles Reference:

1st Level Vendor External ID:

2nd Level Vendor External ID:

**Submit** **Cancel**

- [IDP メタデータ (IDP Metadata)]: CloudCenter と IDP 間の相互の信頼関係を確立します。
- [エンティティ ID (Entity ID)]: この認証のターゲットドメイン名。
- [ログアウト URL (Logout URL)]: 会社の SAML ページにログインする場合は、ログインしたユーザが SAML ページからログアウトする際にそのユーザをリダイレクトするページの URL を指定する必要があります。CloudCenter 4.8.0 はユーザをカスタム ログアウト ページにリダイレクトします (企業によって設定されている場合)。
- [属性マッピング (Attribute Mapping)] セクション: CloudCenter 内のユーザ属性にマッピングされる IDP からのフィールドです。これらのフィールドについて不明な点がある場合は、IDP 管理者に問い合わせてください。少なくとも、名、姓、電子メール アドレス、外部ユーザ ID は入力する必要があります。
- [ユーザ グループ (User Group)]: IDP からフィールド名を指定します。これは、ユーザが属するグループを決定するために使用されます。
- [アクティベーション プロファイル参照 (Activation Profiles Reference)]: メタデータ内の属性を特定して、デフォルト プロファイルではなく、関連付けられたアクティベーション プロファイルを選択します。

- [第 1 および第 2 レベルのベンダー外部 ID (1st/2nd Level Vendor External ID)]: SSO 自己署名ユーザの正しいテナントへの配置を自動化するために使用します。
  - SSO ユーザの適切なテナントへの配置を自動化するためにサブテナントを作成する場合に、外部 ID の設定とともに使用します。次に例を示します。

## Attribute Mappings

Mapped by Friendly Name: ☐

First Name:

Last Name:

Company Name:

Email:

User External ID:

1st Level Vendor External ID:

2nd Level Vendor External ID:

- 3 つのレベルの階層を持つ組織の場合:
  - 最上位組織 = **ルート組織**
    - 組織に関係なく各ユーザには一意のユーザ ID (UID) があります。
  - 第 1 レベルの組織 (サブテナント) = **Company1**
    - **Company1** の各ユーザには一意の外部識別子、**Company1\_uid** があります。
  - 第 2 レベルの組織 (サブテナント) = **Company1\_c** (それぞれの会社には複数の部門または顧客が存在する場合と、存在しない場合があります)
    - 各部門または顧客ユーザには一意の外部識別子、**Company1\_c\_uid** があります。
- **Company1\_uid** と **Company1\_c\_uid** の組み合わせを使用してユーザの組織を見つけます。
  - **Company1\_uid** と **Company1\_c\_uid** が ない ユーザはルート組織に配置されます。
  - **Company1\_uid** のみのユーザは次のように配置されます。
    - 第 1 レベルのサブテナント
    - 外部 ID がユーザの **Company1\_uid** と一致
  - **Company1\_uid** と **Company1\_c\_uid** の 両方 を持つユーザは次のように配置されます。
    - 第 2 レベルのサブテナント
    - 外部 ID
      - 第 2 レベルのサブテナントがユーザの **Company1\_c\_uid** と一致
      - 第 1 レベルのサブテナントがユーザの **Company1\_uid** と一致
- デフォルトの動作では、ベンダー ID にマッピングされたフィールドの値が空白の場合、ユーザは**ルート** テナントに配置されます。
- デフォルトのマッピングを設定し、[ベンダー ID (Vendor ID)] フィールドに値を指定することによってユーザをブロックすることはできますが、テナントの値に一致しません。



第 1 レベルと第 2 レベルのベンダーの値と、いずれのテナントの外部 ID にも一致しない外部 ID をマッピングした場合、それらのユーザにはアクセス権は与えられません。

7. SP メタデータをダウンロードし、それを IDP 管理者に送信して SP を登録します。

## 参考資料

IDP ループの SP グループへのマッピングの例については、[Ping ID ナレッジ ベース](#) を参照してください。

## ADFS SAML SSO の統合

- 概要
- ドメインおよびポータルの確認
- CloudCenter サポート
- SAML 認証設定
- ADFS の信頼設定

### 概要



**CloudCenter は LDAP や AD に対して直接認証されません。**

CloudCenter は、SAML 2.0 プロトコルをサポートする SSO ID プロバイダー (IDP) を通じて LDAP や AD とのみやり取りします (Ping ID、ADFS、Shibboleth など)。

CloudCenter を使用して SSO を実装するには、次が必要です。

1. CCM を設定して、認証を SSO IDP にリダイレクトする必要があります。
2. また、ユーザ アクティベーション プロファイルに対して、(SAML IDP によって返された) 追加のユーザのカスタム プロパティをマッピングする必要もあります。
3. これらのステップをすべて正常に実行すると、CloudCenter は適切なユーザ グループ メンバーシップと追加のロールおよび権限を自動的に割り当てます。

CCM インスタンスは Security Assertion Markup Language (SAML) 2.0 SSO を Spring Security SAML Extension を通じてサポートします。

システム管理者 (SysAdmin) は、ルート レベルまたはテナント レベルで SAML 統合を設定できます。この統合を正確に設定するには、導入に応じて、ルート テナントまたはサブテナントに関する次の情報が必要です。

- テナント情報
- アクティベーション プロファイル情報

### ドメインおよびポータルの確認

次の情報が正確であることを確認します。

- CCM のタイムゾーンと時刻 (および関連付けによるその他すべてのアプライアンス) が AD ドメイン コントローラと一致している。
- FQDN ポータル ページ (<https://cloud.core.enterprise.com> など) のログオンが正確である。

### CloudCenter サポート

詳細については、[CloudCenter サポート](#)にお問い合わせください。

### SAML 認証設定

SSO を使用するようテナントを設定するには、次の手順を実行します。

1. テナントを作成します (「[Sub-Tenant Configuration](#)」を参照)
  - a. [短縮名 (Short Name)]: 空白文字および特殊文字を使用せずに文字列を入力します。
  - b. [外部 ID (External Id)]: テナントが関連付けられている外部システムの組織の ID を入力します。
  - c. [テナント \(Tenant\)](#): テナントの CCM サーバドメイン名エイリアス。これは、SSO の観点から、サービス プロバイダー (SP) のエンドポイントとして機能します。
2. 新しく作成したテナント管理者としてログインし、[アクティベーション プロファイル](#)を作成します。
3. [ベンダー情報 (Vendor Info)] タブをクリックし、新しく作成したアクティベーション プロファイルをデフォルトのアクティベーション プロファイルとして選択します。
4. システム管理者 (Sys Admin) としてログインし (「[Admin Users](#)」 > 「[Login as a System Admin](#)」を参照)、[ベンダー管理者の管理 (Manage Vendor Admins)] タブをクリックし、このテナントの [認証設定 (Authentication Settings)] アクションドロップダウンを選択します。

Name	Short Name	URL	Email	Authentication	Actions
Clear Technology Inc.	clear	http://www.clear.com	info@clear.com	DEFAULT	Action List
Marketing	marketing			DEFAULT	Action List
Sales	sales			DEFAULT	Action List
Human Resources	HumanResources			DEFAULT	Action List
Documentation	docs			DEFAULT	Action List
Documentation Search	Doc			DEFAULT	Action List
Training Portal	Training			DEFAULT	Action List
CloudIQ	cloudiq		cloudiq@clear.com	DEFAULT	Action List
Unlabeled	Unlabeled			DEFAULT	Action List
Education	edu			DEFAULT	Action List
IDP Settings	IdP Settings			DEFAULT	Action List
UK	UK			DEFAULT	Action List

5. [IDP 設定 (IDP Settings)] に情報を入力します。
  - a. [IDP 名 (IDP Name)] (名前の例ではサポート対象の AD ドメインを示しています)
  - b. [IDP メタデータ URL (IDP Metadata URL)]: CloudCenter プラットフォームと IDP 間の相互の信頼関係を確立します (現在、HTTPS はサポートされていないため、HTTP を使用します)。
  - c. [IDP メタデータ ファイル (IDP Metadata File)] (該当する場合)。
6. [SP 設定 (SP Settings)] に情報を入力します。
  - a. [エンティティ ID (Entity ID)]: この認証のターゲットドメイン名 (ログイン ページの DNS 名が必要です)
  - b. [デフォルトの SSO (Default SSO)] バインディングは、ポスト時はそのままにします
  - c. [ログアウト ターゲット URL (Logout Target URL)]: 会社の SAML ページにログインする場合は、ログインしたユーザが SAML ページからログアウトする際にそのユーザをリダイレクトするページの URL を指定する必要があります ([エンティティ ID (Entity ID)] と同じである可能性があります)。
7. [属性マッピング (Attribute Mappings)] セクションに情報を入力します。これらは、CloudCenter プラットフォーム内のユーザ属性にマッピングされる IDP からのフィールドです。これらのフィールドについて不明な点がある場合は、IDP 管理者に問い合わせてください。少なくとも、名、姓、電子メール アドレス、外部ユーザ ID は入力する必要があります。
  - a. [名 (First Name)] のマッピングを入力します (<http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname>)
  - b. [姓 (Last Name)] のマッピングを入力します (<http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname>)
  - c. [電子メール (Email)] のマッピングを入力します (<http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress>)
  - d. [ユーザ グループ (User Group)] のマッピングを入力します (<http://schemas.xmlsoap.org/claims/Group>)
  - e. メタデータ ファイルをダウンロードします。
8. [送信 (Submit)] をクリックします。

## ADFS の信頼設定

ADFS の信頼設定を行い、対応する要求規則を編集するには、次の手順を実行します。

1. AD FS Manager で、[AD FS] > [Trust Relationship] > [Relying Party Trusts] を選択し、[Add Relying Party Trust] をクリックして [Add Relying Party Trust] ウィザードを開きます。
2. [Welcome] ページで [Start] をクリックします。
3. [Select Import Data from a file] ページで、sp-xxxxx.xml を参照し、選択します。
4. [Next] をクリックします。
5. [Display name] に表示名を指定します。
6. [Next] をクリックします。
7. [Configure Multi-factor Authentication Now?] ページで [I do not want to configure multi-factor authentication settings for this relying party trust at this time] を選択します。
8. [Next] をクリックします。
9. [Choose Issuance Authorization Rules] ページで、[Permit all users to access this relying party] を選択します。
10. [Next] をクリックします。
11. [Ready to Add Trust] ページで、新しい信頼当事者証明のプロパティを入力し、[Next] をクリックして、信頼当事者証明情報を保存します。
12. [Finish] ページで、[Close] をクリックします。このアクションにより、[Edit Claim Rules] ボックスが自動的に表示されます。
13. [プロパティ (Properties)] をクリックします。
14. [Advanced] タブの [Secure hash algorithm] で [SHA-1] を選択し、[OK] をクリックします。
15. リストで、要求規則を作成する信頼をクリックします。
16. 選択した信頼を右クリックし、[Edit Claim Rules] をクリックします。
17. [Select Rule Template] ページの [Claim rule template] でリストから [Send LDAP Attributes as Claims] をクリックし、[Next] をクリックします。
18. [Configure Rule] ページの [Claim rule name] の表示名フィールドに **Get Attributes** と入力します。
19. [Mapping of LDAP attributes to outgoing claim types] で、次の **LDAP 属性** を選択し、対応する **出力方向の要求の種類タイプ** をドロップダウンリストから選択します。
  - a. [Given-Name] = [Given Name]
  - b. [Surname] = [Surname]
  - c. [E-Mail-Addresses] = [E-Mail Address]
  - d. [Token-Groups - Unqualified Names] = [Group]

Edit Rule - Get attributes
✕

You can configure this rule to send the values of LDAP attributes as claims. Select an attribute store from which to extract LDAP attributes. Specify how the attributes will map to the outgoing claim types that will be issued from the rule.

Claim rule name:

Rule template: Send LDAP Attributes as Claims

Attribute store:

Mapping of LDAP attributes to outgoing claim types:

	LDAP Attribute (Select or type to add more)	Outgoing Claim Type (Select or type to add more)
▶	Given-Name ▼	Given Name ▼
	Surname ▼	Surname ▼
	E-Mail-Addresses ▼	E-Mail Address ▼
	Token-Groups - Unqualified Names ▼	Group ▼
*	▼	▼

View Rule Language...

OK
Cancel

20. [OK] をクリック
21. 別の規則を入力方向の要求のトランスフォーム テンプレートに追加します。[Select Rule Template] ページの [Claim rule template] で、リストから [Transform an Incoming Claim] を選択し、[Next] をクリックします。
22. 規則に SAM to NameID という名前を指定して、次の値をマッピングします。
  - a. [Incoming claim type] = [E-Mail Address]
  - b. [Outgoing claim type] = [Name ID]
  - c. [Outgoing name ID format] = [Email]



23. [OK] をクリック

これで、ADFS SAML SSO の統合が設定されました。

## ServiceNow

- [アーキテクチャの概要](#)
- [統合の概要](#)
- [リリースノート v1.4](#)
- [インストールと設定 v1.4](#)

### CloudCenter: ServiceNow の統合のアーキテクチャの概要

- [概要](#)
- [アーキテクチャ](#)
- [統合の要件](#)
- [機能](#)

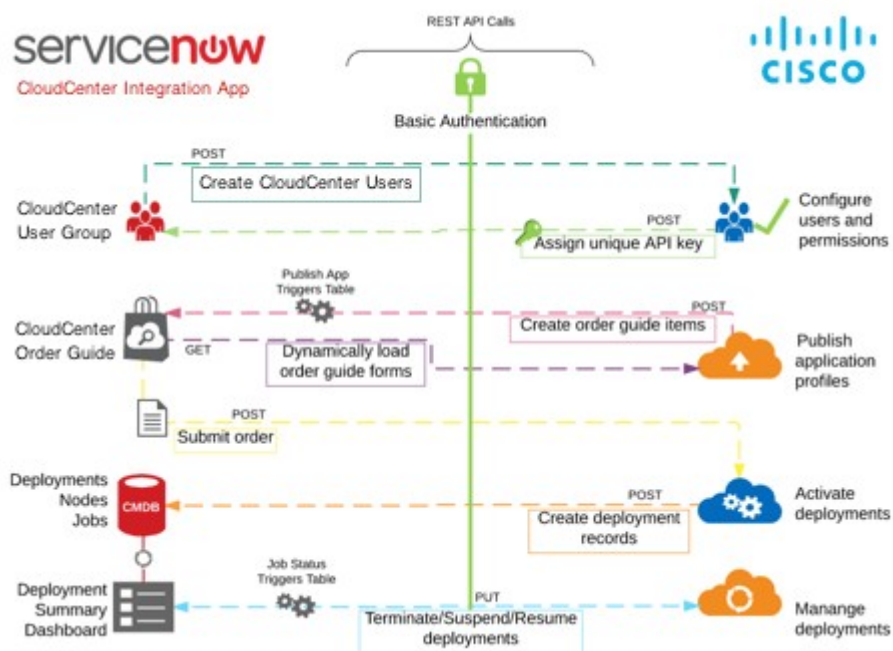
#### 概要

Cisco CloudCenter には ServiceNow によって認定された統合アプリケーションがあり、ServiceNow の App Store からダウンロードできます。統合アプリケーションは、ServiceNow の CMS 内と、「プライベート アプリケーション スコープ」空間内で開発されます。これによって ServiceNow 環境の他の部分に対するリソースの可用性が制限されますが、統合アプリケーションがグローバル スコープに影響を与えず、顧客の ServiceNow 環境の全体的な整合性を維持するようにします。



## アーキテクチャ

次の図に、Cisco CloudCenter と ServiceNow 間の通信を示します。すべての通信が REST API コールを通じて行われます。



- ユーザの作成:** ServiceNow のユーザレコードは、設定プロセス時に設定されたグループに追加されます。ユーザがグループに追加されると、アウトバンド REST コールが実行され、対応するユーザレコードを CloudCenter に作成してアクティベートし、CloudCenter API への後続のコールの認証に使用される API キーが ServiceNow ユーザに割り当てられます。
- パブリッシュ:** アプリケーションが CloudCenter から ServiceNow にパブリッシュされると、ServiceNow に対してインバンド REST コールが実行され、アプリケーションプロファイルが作成されます。ServiceNow への最初のパブリッシュの後、アプリケーションが CloudCenter で更新されている場合は、管理者は ServiceNow にアプリケーションを再パブリッシュする必要があります。
- 注文:** アプリケーションがカタログを通じて要求されると、関連付けられた導入の設定項目 (CI) と論理 CI グループが作成されます。その後、アウトバンド REST コールが実行され、導入 CI が更新され、関連付けられたノードとジョブ CI が作成されます。
- 導入管理:** CloudCenter に導入関連の更新がある場合は、インバンド REST コールが実行され、それらの更新が関連付けられた CI にプッシュされます。

## 統合要件

コンポーネント	要件	詳細
ServiceNow	バージョン サポート	統合アプリケーションでは、ServiceNow が Geneva または Helsinki である必要があります。Helsinki のサポートは統合アプリケーションのバージョン 1.2 から開始されました。
	ポート	CloudCenter Manager は、MID サーバが使用されていない場合は、SSL を介して ServiceNow に直接到達でき、またその逆も可能である必要があります。
	MID サーバ(任意)	MID サーバのサポートは、ServiceNow から CCM へのインバンドトラフィックに使用できます。CCM から ServiceNow へのアウトバンドトラフィックは直接使用できる必要があります。
Cisco CloudCenter	CloudCenter の動作環境	「 <a href="#">Manual Installation</a> 」を参照してください。
	バージョン サポート	統合アプリケーションは、Cisco CloudCenter 4.6.0 以降をサポートしています。
	ポリシー定義	実装ガイドに定義されているとおり、アプリケーションのパブリッシュにはアクション ポリシーを Cisco CloudCenter に作成する必要があります。
	テナント管理者の API キー	ServiceNow から CloudCenter への API 通信の場合

## 機能

- **MID サーバ**: 統合は、MID サーバの有無を問わず、ServiceNow 環境をサポートします。
- **アプリケーション セキュリティ**: すべての API キーがパスワード(2 方向暗号化)内部タイプを使用して暗号化されます。CloudCenter へのコールには、ユーザに関連付けられた API キーを使用した基本的な認証が必要です。
- **アプリケーション コンポーネント**: 統合アプリケーションは、多数のスクリプト インクルード、ビジネス ルール、および一部の ServiceNow テーブルの拡張などのテーブルを作成します。
- **変更されたコンポーネント**: 統合では ServiceNow 内の発注書コンポーネントが変更されます。
- **アプリケーションのカスタマイズ**: 統合アプリケーションをカスタマイズできます。顧客が自社のローカル インストールに変更を行う場合、それらの変更は、今後の製品の更新によってローカルの変更が上書きされる場合を考え、更新セットで維持されるようにすることを推奨します。
- **製品(アプリケーション)カタログ**: 統合では Angular を使用して開発された新しい製品カタログが追加されます。統合アプリケーションは OOB ServiceNow カatalogを使用しません。
- **見積もりの生成**: 統合は CloudCenter に保存されたイメージまたはクラウド コストのデータに依存してチェックアウト フォームに見積もりを提供します。見積もりを概算として使用し、導入コストを決定します。
- **導入の運用コスト**: 統合は、各導入の運用コストを最初の見積もりプロセス時に取得したコスト データに基づいて維持します。

## CloudCenter と ServiceNow の統合の概要

- [概要](#)
- [統合の詳細](#)
- [利点](#)
- [機能](#)

### 概要

Cisco CloudCenter を使用すると、サービス マネージャとアプリケーション所有者は、モデル化は 1 回、任意の場所に展開のアプリケーション プロファイルを簡単に作成し、ServiceNow にパブリッシュすることができます。

ServiceNow を使用すると、ユーザはサポート対象のデータセンターまたはクラウドへの導入を、各オプションのコストの対照比較とともに要求することができます。

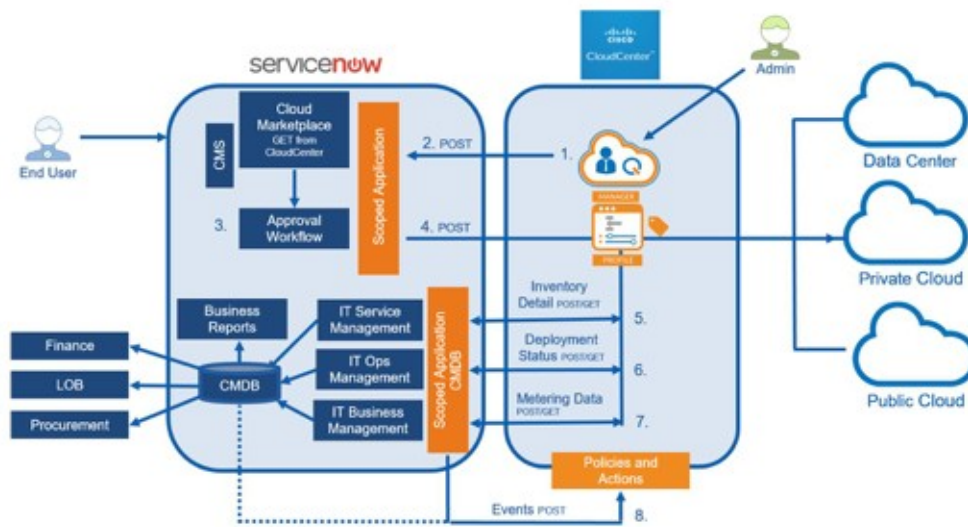
Cisco CloudCenter と ServiceNow の統合は、IT サービス管理 (ITSM)、IT 運用管理 (ITOM)、および IT ビジネス管理 (ITBM) プロセスを拡張してアプリケーションの導入と管理を組み込む一方で、企業全体の管理機能とガバナンス機能を追加します。

### 統合の詳細

Cisco CloudCenter は、ServiceNow によって認定され、ServiceNow の App Store からダウンロードできる統合アプリケーションを提供します。この統合により、CloudCenter で作成されたアプリケーションやサービスが ServiceNow のカスタム カタログにパブリッシュされて、指定された ServiceNow ユーザが使用できるようになります。これらのユーザは、カタログを参照し、アプリケーションを設定してカートに追加し、クラウド間で導入コストを比較し、希望するクラウドにアプリケーションを導入することができます。

導入のためにアプリケーションを送信すると、統合アプリケーション内でそのステータスをトラッキングできます。さらに、一時停止、再開、または終了などのライフサイクル アクションを、CloudCenter からのリアルタイムの更新を使用してアクティブな導入上で実行できます。ユーザは、ServiceNow を使用したまま、導入済みの仮想マシンに SSH または RDP で接続することもできます。

これらの機能はすべて、ServiceNow の CMS 内で完全に機能し、開発された一連のサンプル UI ページを実装することによって可能となります。ユーザ インターフェイスは、わかりやすく、カスタマイズが可能であるとともに、クラウドの導入と管理のほとんどを簡単にできるようにします。統合アプリケーションのセットアップ ウィザードの一部として、各 ServiceNow 環境が CloudCenter 環境と簡単にリンクされ、ServiceNow と Cisco CloudCenter 間のすべての通信が保護され、REST API コールを通じて実行されます。



## 利点

CloudCenter-ServiceNow 統合アプリケーションには次の利点があります。

- ServiceNow による認定:** 統合アプリケーションは ServiceNow によって認定されており、ServiceNow のお客様は App Store からダウンロードできます。このアプリケーションは、グローバル スコープではなく、ServiceNow アプリケーションのスコープ内で開発されており、既存の ServiceNow の実装への影響が最小限に押さえられています。
- DevOps を促進:** 新しいインフラストラクチャやアプリケーションの導入をさらに効率的に取得できるようにすることで、DevOps を促進します。
- 使いやすさ:** シンプルかつ効果的なユーザ インターフェイスによって、クラウドの導入の複雑さを排除します。ServiceNow のエンド ユーザに単一の場所を提供することで、追加ツールのためのトレーニングの必要性を削減します。
- コスト削減:** 導入や終了をスケジュールすることで、コストを低減し、インフラストラクチャリソースを最適化します。
- 互換性の向上:** 統合アプリケーションは MID サーバを利用することで ServiceNow 環境との互換性が得られます。
- アプリケーションのパブリッシュ:** サービス マネージャは、クラウドに依存しないアプリケーション プロファイルを、設定済みまたはインポートしたイメージ、アプリケーション サービス、コンテナを使用して Cisco CloudCenter モデラー内で簡単にモデル化し、それを ServiceNow に対してパブリッシュして使用されるようにすることができます。
- サービス要求:** ServiceNow では、指定されたユーザは、アプリケーションおよびサービスを選択し、ショッピング カートに追加し、開始日と終了日を入力して、利用可能なクラウドごとに提示されるコストの対照比較に基づいて導入場所を選択できます。
- ワークフローの承認:** 要求を予想コストとともに承認者にルーティングすることで、その意思決定を促進します。
- 導入の自動化:** ServiceNow は Cisco CloudCenter ソリューションを呼び出し、ターゲットのクラウド API との直接通信を通じてクラウドのインフラストラクチャリソースをプロビジョニングします。次に、アプリケーション イメージ、サービス、およびデータを導入し、設定します。
- ステータスの更新:** ServiceNow は Cisco CloudCenter ソリューションからステータスと進行中のインフラストラクチャ パフォーマンスの更新を受け取ります。サードパーティ製のアプリケーション パフォーマンス モニタリングおよびサービス保証ツールからの情報を使用して、ServiceNow はインシデントをログに記録するか、または新しい修復ワークフローを開始することができます。

## 機能

- Cisco CloudCenter から ServiceNow へのアプリケーションのワンクリック パブリッシュ。
- 単一の仮想マシン、または完全なアプリケーション スタックのプロビジョニングと管理。
- カスタマイズを容易にするために Angular 内で開発され、OOB ServiceNow カタログにはリンクされていない製品(アプリケーション)カタログの提供。
- ショッピング カートのエクスペリエンス。
- 発注時の設定済みクラウドの導入コストの対照比較。
- 後で使用するための見積書の保存、または財務処理のためのダウンロード。
- 承認ワークフローと電子メール通知。ガバナンスを目的とした Cisco CloudCenter 内でのタグの使用。
- 展開および終了のスケジュールリング。
- アクティブな導入の一時停止、再開、終了などのライフサイクル アクション。
- 導入済みの仮想マシンへのリモート接続。
- ServiceNow と Cisco CloudCenter 間の通信に REST API を使用。

## CloudCenter/ServiceNow v1.4 の統合のリリースノート

- リリース日
- アーキテクチャ
- インストール
- アップグレード
- セキュリティ
- サポートされる CloudCenter プラットフォーム
- サポートされる ServiceNow プラットフォーム
- 新機能
- 拡張機能
- その他の更新
- 廃止された項目
- 既知の問題
- 解決済みの問題

### リリース日

2017 年 4 月 28 日

CloudCenter 4.8.0 のサポート: 2017 年 5 月 15 日

### アーキテクチャ

更新なし。

### インストール

更新なし

### アップグレード

ServiceNow v1.2 または v1.3 から v1.4 にアップグレードするには、アップグレード ファイルとともに提供される CM: v1.2 - 1.4: 修正スクリプトを実行する必要があります。

### セキュリティ

更新なし

### サポートされる CloudCenter プラットフォーム

- Cisco CloudCenter 4.8.0
- Cisco CloudCenter 4.7.1.1
- Cisco CloudCenter 4.6.x

### サポートされる ServiceNow プラットフォーム

- ServiceNow Helsinki
- ServiceNow Istanbul



ServiceNow Geneva のサポートは、v1.4 では使用できません。

### 新機能

v1.4 で追加された新機能はありません。

## 拡張機能

機能拡張	説明
[サービス インスタンス (Service Instances)] セクション	<ul style="list-style-type: none"> <li>フィルタ オプションとしてステータス [再開中 (Resuming)] を使用できます。</li> <li>[リモート接続 (Remote Connection)] には次の詳細が含まれています。 <ul style="list-style-type: none"> <li>プライベート IP (Private IP)</li> <li>パブリック IP (Public IP)</li> <li>ノード ID (Node ID)</li> <li>ノード ステータス (Node Status)</li> </ul> </li> </ul>
発注プロセス	<i>webservice</i> および <i>path</i> の各グローバル パラメータのサポートの追加。

## その他の更新

- ServiceNow、Istanbul バージョンのサポートを追加しました。
- IE11 ブラウザと Firefox ブラウザのサポートを改善しました。
- rest\_service* ロールと *model\_manager* ロールが *Cliqr* ユーザ グループから削除されました。

## 廃止された項目

なし。

## 既知の問題

Cloud Marketplace v1.4 には以下の既知の問題があります。

ServiceNow から導入する場合、[Public IP] の値はデフォルトで *True* に設定され、CloudCenter での値が *False* に設定されていると導入は失敗します。この問題を回避するには、[CloudCenter サポート チーム](#)にお問い合わせください。

## 解決済みの問題

以下の問題は、Cloud Marketplace v1.4 で解決済みまたは対処済みです。

- 問題:** 導入ステータスが [進行中 (In-Progress)] または [エラー (Error)] である場合に、導入を拡張するためのオプションが表示されていました。  
**解決:** [サービス インスタンスの詳細 (Service Instance details)] ページの [拡張 (Extend)] オプションは、導入ステータスが *導入済み (Deployed)* または *中断 (Suspended)* である場合にのみ表示されます。
- 問題:** 親ノードの [OS 名 (OS Name)] は常にデフォルトで [Linux] になっていました。  
**解決:** 親ノードの [OS 名 (OS Name)] はプリセットされなくなりました。
- 問題:** *x\_cqt\_cliqr\_publish\_app\_trigger* テーブルの owner フィールドの文字列長が原因で、より長い名前を持つユーザの場合にアプリケーションを ServiceNow にパブリッシュできませんでした。  
**解決:** デフォルトの文字列値は 40 文字から 100 文字に拡大されました。

## ServiceNow と CloudCenter の統合 v1.4 のインストールと設定

- 可用性
- CloudCenter の前提条件
- Cisco CloudCenter の設定
  - 1. カスタム アクションの作成
  - 2. ポリシーの作成
- ServiceNow の設定
  - 1. CloudCenter-ServiceNow 統合アプリケーションのダウンロードとインストール
  - 2. CloudCenter-ServiceNow 統合アプリケーションの設定
  - オプション設定
  - マルチテナント ユーザ設定の微妙な違い
- CloudCenter と ServiceNow との統合の検証
- その他の注意事項

## アベイラビリティ

**Integration–CloudCenter**、リリース 1.4 は、以下の **ServiceNow** リリースでサポートされます。

- ServiceNow、Istanbul
- ServiceNow、Helsinki

**Integration–CloudCenter**、リリース 1.4 は、以下の **CloudCenter** リリースでサポートされます。

- CloudCenter 4.8
- CloudCenter 4.7
- CloudCenter 4.6

## CloudCenter の前提条件

1. ガバナンス モードを有効にする。
2. 次のリソースを **自分のテナントのすべてのユーザ**と共有する(各リソースの [共有 (Share)] ポップアップ内)。
  - a. システム タグ
  - b. 導入環境
  - c. アプリケーション プロファイル

## Cisco CloudCenter の設定

ServiceNow の統合用に CloudCenter を設定するには、段階的なプロセスに従う必要があります。

### 1. カスタム アクションの作成

Cisco CloudCenter から ServiceNow にアプリケーション プロファイルをパブリッシュするカスタム アクションを作成する必要があります。カスタム アクションを作成するには、Cisco CloudCenter で次の手順を実行します。

1. 管理者権限を使用して、Cisco CloudCenter にログインします。
2. 左側のナビゲーションから [ポリシー (Policies)] を選択します。
3. [カスタム アクション (Custom Actions)] タブを選択します。
4. [カスタム アクションの追加 (Add Custom Action)] をクリックし、各フィールドを次のように定義します。

フィールド	値
[名前 (Name)]	ServiceNow へパブリッシュ
説明	このフィールドには空白のままにします
ユーザに表示 (Visible to User)	[有効 (Enabled)]
オブジェクト	Application
アクション タイプ	Web サービスの呼び出し
[プロトコル (Protocol)]	HTTPS
Web サービス URL	<yourServiceNowInstance.com>/api/now/table/x_cqt_cliqr_publish_app_trigger
[ユーザ名 (Username)]	rest.admin (このユーザは後で ServiceNow に作成されます)
[パスワード (Password)]	rest.admin ユーザのパスワードを入力します。
HTTP 要求タイプ (Http Request Type)	POST
コンテンツ タイプ (Content Type)	JSON
コマンド パラメータ (Command Param)	このフィールドには空白のままにします



**本文 (Body)**

```
{
  "app_id": "%appId%", "app_name": "%appName%",
  "latest_app_version": "%latestAppVersion%",
  "owner_id": "%ownerId%",
  "owner": "%owner%"
}
```

## 2. ポリシーの作成

3 つのアクション ポリシーを作成して、アプリケーションの導入のステータスを ServiceNow に伝達する必要があります。各ポリシーについては、次に示す共通設定と個別のポリシー設定を使用します。3 つの個別ポリシーを作成するために、次の手順を 3 回繰り返します。

新しい**それぞれの**ポリシーを作成するには、Cisco CloudCenter で次の手順を実行します。

1. 左側のナビゲーションから [ポリシー (Policies)] を選択します。
2. [ポリシー (Policies)] タブを選択します。
3. [アクション ポリシーの追加 (Add Action Policy)] をクリックし、各フィールドを次のように定義します。

**共通設定**

フィールド	値
[名前 (Name)]	下記の「 <i>個別のポリシーの設定</i> 」セクションを参照してください。
説明	このフィールドには空白のままにします
次に関して実行: (Execute For)	アプリケーションの導入
次のイベント時に (On Event)	下記の「 <i>個別のポリシーの設定</i> 」セクションを参照してください。
アクション タイプ	Web サービスの呼び出し
[プロトコル (Protocol)]	HTTPS
Web サービス URL	<yourServiceNowInstance.com>/api/now/table/x_cqt_cliqr_job_status_trigger
[ユーザ名 (Username)]	rest.admin (このユーザは後で ServiceNow に作成されます)
[パスワード (Password)]	rest.admin ユーザのパスワードを入力します。
HTTP 要求タイプ (Http Request Type)	POST
コンテンツ タイプ (Content Type)	JSON
コマンド パラメータ (Command Param)	このフィールドには空白のままにします
本文 (Body)	下記の「 <i>個別のポリシーの設定</i> 」セクションを参照してください。
Auto Enabled for shared users (共有ユーザに対して自動的に有効化)	[有効 (Enabled)]
このポリシーのユーザによる無効化を制限する (Restrict users from disabling this Policy)	[有効 (Enabled)]

## 個別のポリシーの設定

ポリシー	次のイベント時に (On Event)	本文 (Body)
1. SNOW_job_status_changed	ステータス変更	<pre>{   "job_id": "%jobId%",   "job_name": "%jobName%",   "job_type": "%jobType%",   "app_name": "%appName%",   "owner": "%owner%",   "status": "%status%",   "changed_on": "%ChangedOn%",   "new_status": "%NewStatus%" }</pre>
2. SNOW_job_deployed	導入済み	<pre>{   "job_id": "%jobId%",   "job_name": "%jobName%",   "job_type": "%jobType%",   "app_name": "%appName%",   "owner": "%owner%",   "status": "%status%",   "deployed_on": "%DeployedOn%" }</pre>
3. SNOW_job_canceled	キャンセル (Canceled)	<pre>{   "job_id": "%jobId%",   "job_name": "%jobName%",   "job_type": "%jobType%",   "app_name": "%appName%",   "owner": "%owner%",   "status": "%status%",   "cancelled_on": "%CancelledOn%" }</pre>

4. 3 つのポリシーすべてに対してこの手順を繰り返し、これらのポリシーを **自分のテナント内のすべてのユーザ** と共有します。

## ServiceNow の設定

CloudCenter-ServiceNow 統合アプリケーションを設定するには、段階的なプロセスに従う必要があります。

### 1. CloudCenter-ServiceNow 統合アプリケーションのダウンロードとインストール

CloudCenter-ServiceNow 統合アプリケーションをダウンロードしてインストールするには、次の手順を実行します。

- CloudCenter-ServiceNow 統合アプリケーションをダウンロードする要求を送信します。
  - Web サイト <https://store.servicenow.com> にアクセスします。
  - Integration - CloudCenter** を検索します。
  - カタログから [Integration - CloudCenter] アプリケーションを選択します。
  - [Contact Seller] を選択して、アプリケーションをダウンロードする承認を受けます。
  - シスコがアプリケーションのダウンロードを承認すると、ServiceNow 管理者は確認電子メールを受け取ります。確認電子メールでリンクをクリックして、アプリケーションを **取得** します。
- ServiceNow に CloudCenter アプリケーションをインストールします。
  - アプリケーションが承認されたら、左側のフィルタ オプションを使用して、[System Applications] > [Applications] > [Downloads] > [Integration - CloudCenter] に移動します。
  - [All versions] をクリックします。



- c. 適切なバージョンを選択し、インストールします。



## 2. CloudCenter-ServiceNow 統合アプリケーションの設定

CloudCenter-ServiceNow 統合アプリケーションを設定するには、次の手順を実行します。

1. Procurement プラグインを有効にします。
  - a. フィルタを使用し、[Plugins] に移動します。
  - b. プラグイン モジュール内で **Procurement** を検索します。
  - c. [Procurement - com.snc.procurement] を見つけてアクティブ化します。
  - d. 完了したら、[Close and Reload Form] をクリックします。



2. 証明書をインストールします。



自己署名証明書を Cisco CloudCenter で使用する場合は、ServiceNow に証明書を追加して通信を有効にする必要があります。

- a. フィルタを使用して、[System Definition] > [Certificates] に移動します。
  - b. 種類に [Trust Store Cert] を使用して、新しい X.509 証明書を追加します。
3. Cisco CloudCenter との接続を設定します。



このステップでは、ネットワーク上の ServiceNow 環境と通信できる CloudCenter Manager (CCM) があると仮定します。

- a. ログインした ServiceNow ユーザが **u\_cliqr\_admin** または **x\_cqt\_cliqr.global\_tenant\_admin** のいずれかのグループに含まれていることを確認します。
- b. フィルタを使用し、[Cloud Marketplace Configuration] に移動します。
- c. Cisco CloudCenter との接続を確立するには、[General Settings] セクションの次の必須フィールドに入力します。参考までに、下のスクリーンショットを参照してください。

全般設定	その他の情報
CCM URL	https://<ciscoCloudCenterURL>.com 形式を使用して、Cisco CloudCenter の URL を入力します。
[ユーザ名 (Username)]	Cisco CloudCenter のデフォルトのユーザ名である <b>cliqradmin</b> を入力するか、または別の管理者アカウントを入力します。
API Key	上記で使った管理者アカウントの API キーを入力します。API キーを取得するには、Cisco CloudCenter にログインします。次に、[管理者 (Admin)] > [ユーザ (Users)] に移動し、使用するアカウントの [API キーの管理 (Manage API Key)] をクリックします。

The screenshot shows the 'Cloud Marketplace Configuration' page. The 'General Settings' section includes fields for 'CCM URL' (https://62.89.222.75), 'Username' (cliqadmin), and 'API Key' (XXXXXXXXXXXXXXXXXXXX). There is an 'Enable PO' checkbox. The 'Cloud and Instance Mapping' section has an 'Add mappings' button. A 'Submit' button is at the bottom right.

- d. [General Settings] セクションに入力したら、[Submit] をクリックします。[Cloud Marketplace Configuration] ページがリロードされ、[Cloud and Instance Mapping] という新しいセクションにアクセスできるようになります。同じページに留まり、次の項のステップを実行します。



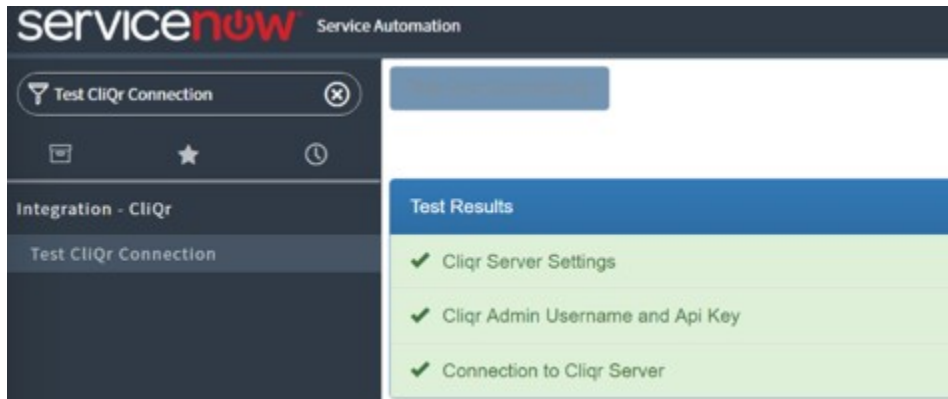
この手順は重要です。[Cloud and Instance Mapping] セクションが表示されない場合は、ServiceNow は Cisco CloudCenter と通信できません。

4. クラウドとインスタンスのマッピングを設定します。  
a. [Cloud and Instance Mapping] セクションの次の必須フィールドに入力します。参考までに、下のスクリーンショットを参照してください。

クラウドとインスタンスのマッピング	その他の情報
Cloud Family	ドロップダウンを使用してクラウドを選択します。Cisco CloudCenter で有効になっているクラウドのみを選択できます。
地域	ドロップダウンを使用して、Cisco CloudCenter で有効になっているリージョンを選択します。
Instance Mapping	統合アプリケーションの注文書には、ユーザがインスタンスのサイズを選択できるようにシンプルなアプローチが使用されています。このドロップダウンを使用して、小規模、中規模、大規模、超大規模に適したインスタンス タイプ値を関連付けます。その他のクラウドを設定するには、プラス  アイコンをクリックします。

The screenshot shows the 'Cloud and Instance Mapping' section. It includes a text field with 'XXXXXXXXXXXXXXXXXXXX', an 'Enable PO' checkbox, and a dropdown menu for 'Cloud Family' set to 'Amazon'. Below are several dropdown menus for instance sizes: 'Small' (m1.small), 'Medium' (m1.medium), 'Large' (m3.medium), and 'Extralarge' (m4.large). A 'Submit' button is at the bottom right.

- b. [Cloud and Instance Mapping] セクションの入力を完了したら、[Submit] ボタンをクリックして設定を保存します。
5. 接続の確認。
- a. フィルタを使用して、[Integration - CloudCenter] > [Test CliQr Connection] に移動します。
- b. [Test CliQr Connectivity] をクリックします。
- c. 次の図に示すように、3 つのテストがすべて成功する必要があります。成功しなかった場合は、このアプリケーションに対する MID サーバの設定の接続詳細を確認します。詳細については、「オプション設定」セクションを参照してください。



6. 統合ユーザの作成: ServiceNow で次のように新規ユーザを作成:

フィールド	値
ユーザ ID (User ID)	rest.admin
名	Rest
姓	Admin
E メール	不要
[パスワード (Password)]	<enter a secure password>
Web service access only	[有効 (Enabled)]
Internal integration user	[有効 (Enabled)]
ロール	次のロールを追加します。 <ul style="list-style-type: none"> <li>• u_cliqr_admin</li> <li>• rest_service</li> </ul>

7. 設定後のチェックリストに記載されている項目に従って、統合を確認します。アプリケーションをアップグレードまたはインストールすると、次のグループおよびロールが自動的にロードされます。

グループ名 (Group Name)	ロール	その他の情報
CliQr	N/A	<p>このグループ内のユーザは、そのユーザ アカウントが初めて CloudCenter 内に作成されたときにデフォルトの (最初の) CloudCenter アクティベーション プロファイルに関連付けられます。</p> <p>単一アクティベーション プロファイルを使用する場合、すべてのクラウド マーケットプレイス ユーザがこのグループに含まれている必要があり、グループ名は変更できません。</p> <p>CloudCenter が単一アクティベーション プロファイルで設定されていて (デフォルト)、マルチテナントは有効になっていない (デフォルト) 場合、これ以上の設定は不要です。</p>

a. 以下のグループを設定します (ユーザが適切なロールを持つようにするために推奨)。

グループ名 (Group Name)	ロール	その他の情報
Cloud Marketplace Consumers	u_cliqr_requester	このグループ内のユーザには、クラウド マーケットプレイスへのアクセス権が付与されています。このグループに追加されるユーザは、CloudCenter で自動作成されて ServiceNow に保存される、対応するユーザ アカウントと API キーを持ちます。
Cloud Marketplace Approvers	x_cqt_cliqr.approver	このグループ内のユーザには、クラウド マーケットプレイス内のダッシュボードに [Approvals] タブも表示されます。
Cloud Marketplace Admins	u_cliqr_admin	このグループ内のユーザは、クラウド マーケットプレイスの設定タスクを実行できます。

- b. [Integration - CloudCenter] > [Group Properties] の下に次のグループ プロパティが存在することを確認します。

グループ	ロール	その他の情報
CliQr - <AP_NAME>	N/A	<p>特定のユーザが別のアクティベーション プロファイルに関連付けられる場合、またはこのアプリケーションに対してマルチテナントが有効になっている場合は、必要に応じて追加のグループを作成します。</p> <p>追加のアクティベーション プロファイル(またはテナント)ごとに、左側の命名規則に従って、関連付けられたグループを作成します。各 CloudCenter アクティベーション プロファイルは、関連付けられたグループ マッピングを ServiceNow に追加することも必要とします。</p> <ol style="list-style-type: none"> <li>アクティベーション プロファイル ID(profile_id)を、<a href="#">アクティベーション プロファイル表示 API</a> を使用して CloudCenter プラットフォームから取得します。</li> <li>[Group Properties] テーブルに移動し、以下の詳細を使用してレコードを追加します。 <ol style="list-style-type: none"> <li>グループ: <b>CliQr</b> - &lt;アプリケーション プロファイル名&gt;</li> <li>名前: <b>activation_profile</b></li> <li>値: &lt;プロファイル ID&gt; (<a href="#">アクティベーション プロファイル表示 API</a> から取得)</li> </ol> </li> </ol>
Cloud Marketplace Global Tenant Admins	<ul style="list-style-type: none"> <li>u_cliqr_admin</li> <li>x_cqt_cliqr.global_tenant_admin</li> </ul>	<p>このグループは、マルチテナントが有効になっている場合にのみ作成します。</p> <p>このグループ内のユーザは、ServiceNow 内のサブテナントの同期と設定、および ServiceNow ユーザのテナント ID の修正を行うことができます。</p>

8. 調達モジュールの使用は任意ですが、アクティブな発注書(PO)を使用する場合はこのオプションが必要になります。
- インストール時にサンプル PO がロードされます。
  - PO を有効にする前に、PO (サンプルの場合も含む)の所有者を設定する必要があります。
  - 有効になったら、すべての要求に関して PO 所有者から承認を得る必要があります。

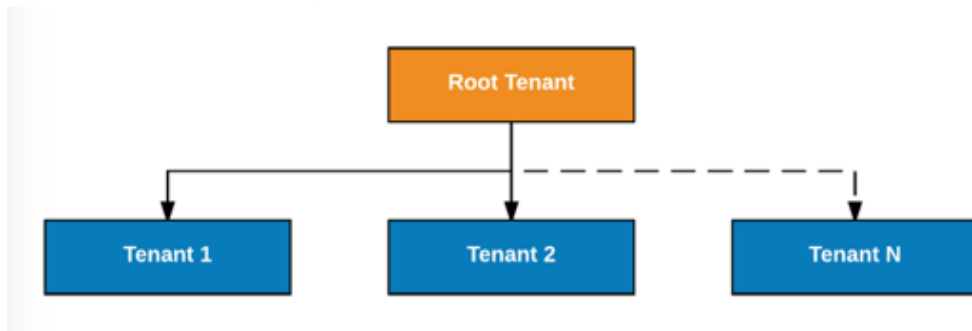
## オプション設定

このセクションで示されているオプションの設定手順により、電子メールを有効にし、MID サーバをサポートすることができます。

- 承認ワークフローの電子メール通知に対して電子メールを有効にする必要があります。既存の ServiceNow のインストールにはすでにこれが設定されている場合があります。
  - フィルタを使用して、[System Properties] > [Email Properties](ServiceNow のバージョンによっては [Email])に移動します。
  - アウトバウンドとインバウンドの電子メール送信が有効になっていることを確認します。
- MID サーバに対するサポートは次のとおりです。
  - [MID Server] > [Capabilities] に移動し、REST、SSH、および SOAP の機能が追加されていることを確認します。
  - [Integration - CloudCenter] > [Settings] > [Properties] の下に次のプロパティを作成します。

プロパティ名	値
Mid_server_name	<configured_rest_server_name>

3. マルチテナント設定 – 省略可能。[Cisco CloudCenter 内で設定されている場合は必須です](#)。
- マルチテナント設定は省略可能で、Cisco CloudCenter 内でも設定されている場合にのみ必要です。ServiceNow アプリケーションは、次の図に示すように、単一階層の複数テナント モデルのみをサポートします。



- 上の図のようなテナント モデルを使用して CloudCenter を設定します。テナントごとに次の項目が設定されていることを確認します (「[CloudCenter の設定](#)」セクションを参照)。
  - アプリケーション プロフィールをパブリッシュするためのカスタム アクション
  - ログインまたはホームページの固有のロゴ
  - 適切な契約と使用計画を持つアクティベーション プロファイル(デフォルト アクティベーション プロファイルとして設定)。

- c. **x\_cqt\_cliqr.global\_tenant\_admin** ロールを持つユーザとして ServiceNow にログインし、[Cloud Marketplace Configuration] ページでマルチテナントを有効にします。

**Username**

**API Key**

☒ Enable MultiTenancy

☐ Enable PO

- d. 変更を保存します。
- e. [List Sub-Tenants] を選択して、サブテナント設定プロセスを開始します。最初は、テナントリストは空になっています。

**Cloud MarketPlace Configuration**

Root Tenant Configuration / List SubTenants

[Sync Tenants](#) [Back](#)

Tenant Name	Parent Tenant Name	Tenant Username
No items to display		

- f. [Sync Tenants] を選択してリストに入力します。

**Cloud MarketPlace Configuration**

Root Tenant Configuration / List SubTenants

[Sync Tenants](#) [Back](#)

Tenant Name	Parent Tenant Name	Tenant Username	
Tenant2	Cisco Systems	tenant2_iU4	<a href="#">Configure</a>
Tenant1	Cisco Systems	tenant1_HU4	<a href="#">Configure</a>
Tenant3	Cisco Systems	tenant3_gV4	<a href="#">Configure</a>

1 - 3 displayed, 3 in total

- g. サブテナントごとに、[Configure] を選択し、API キーを設定し、クラウドとインスタンスのマッピングを設定します。

**Cloud MarketPlace Configuration**

Root Tenant Configuration / List SubTenants / Configuring SubTenant

[Back](#)

### General Settings

**CCM URL**

**Username**

**API Key**

### Cloud and Instance Mapping

Amazon ☒

Cloud Family

Amazon

Cloud Region

US East (Virginia)

Small

d2.xlarge

Medium

db.t1.micro

Large

db.t1.micro

Extralarge

cr1.8xlarge

[Submit](#)

## マルチテナント ユーザ設定の微妙な違い

マルチテナント モデルに関して、以下の要件と結果に注意してください。

1. マルチテナントが有効になっている場合は、ユーザごとに **tenant\_id** フィールドを設定する必要があります。



テナント ID が設定されていない場合、ユーザはクラウド マーケットプレイスを使用できません。

- ユーザの **tenant\_id** は、CloudCenter で設定されているテナント ID、および ServiceNow の [Tenant] テーブルに表示される短縮名に一致する必要があります。
- **tenant\_id** が空白のユーザは、ルート テナントに含まれるとみなされます。
- 最初のインストールでは **tenant\_id** フィールドは表示されません。[User] フォームをカスタマイズして、ビューに **tenant\_id** フィールドを追加する必要があります。
- LDAP を使用している場合は、ユーザをインポートするときに **tenant\_id** フィールドを自動マッピングします。詳細については、[http://wiki.servicenow.com/index.php?title=Setting\\_up\\_the\\_LDAP\\_Import\\_Map#gsc.tab=0](http://wiki.servicenow.com/index.php?title=Setting_up_the_LDAP_Import_Map#gsc.tab=0) を参照してください。

## CloudCenter と ServiceNow との統合の検証

CloudCenter-ServiceNow 統合アプリケーションを検証するには、次の手順を実行します。

1. テスト ユーザを作成します。
  - a. インストールを確認するには、ServiceNow に次のテスト ユーザ アカウントを作成します。ユーザ アカウントを作成するには、電子メールアドレスが必要です。

ユーザ名 (User Name)	グループ メンバーシップ	説明
consumer.user	i. Cloud Marketplace Consumers ii. CliQr	このテスト ユーザを使用して注文を送信します。
approver.user	i. Cloud Marketplace Approvers ii. CliQr	このテスト ユーザを使用して注文要求を承認または拒否します。

- b. 新しい発注書が上記の [Create a Purchase Order] セクションに作成されました。この発注書の [Assigned To] フィールドをユーザ **approver.user** に更新します。
2. アプリケーション プロファイルを ServiceNow にパブリッシュします。
    - a. Cisco CloudCenter の [アプリケーション (Applications)] タブでアプリケーション プロファイルを見つけます。
    - b. アプリケーションのドロップダウンをクリックし、テナント内のすべてのユーザと共有します。
    - c. アプリケーションのドロップダウンをクリックし、[ServiceNow へパブリッシュ (Publish to ServiceNow)] を選択します。
    - d. ServiceNow で、[Integration - CloudCenter] > [Data Tables] > [Application Profiles] に移動します。パブリッシュ済みのアプリケーションを含むレコードがあるはずです。
  3. 発注するには、次の手順を実行します。
    - a. <https://<yourServiceNowInstance>.com/cloud-marketplace> にアクセスします。
    - b. **consumer.user** としてログインします。
    - c. [Store] をクリックします。
    - d. カタログからアプリケーションを選択し、注文を設定して送信します。
    - e. **approver.user** としてログインし、要求を承認します。
    - f. 注文が承認されると、指定された導入開始日時に基づいてその注文が導入されます。
    - g. [サービス インスタンス (Service Instances)] タブを参照し、注文のステータスが [導入済み (Deployed)] であることを確認します。

## 追加の注意事項

統合アプリケーションのインストールによって、タスク テーブルにフィールドが追加されます。これは線形動作であるため、環境内の既存のデータに応じて、ロードにしばらく時間がかかる場合があります。