IBM API Connect 2018.4.1 for VMware

クラスター構成インストール・ガイド

目次

はじめに

当インストール・ガイドは、オンプレミスの IBM API Connect (以下「APIC」という) V2018.4.1 for VMware のクラスター構成をおこなうためのインストール・ガイドです。

当ガイドは、IBM Knowledge Center に記載のガイド[1]に基づいて作成しています。

インストール作業は、おおよそ以下の 3 つのステップからなります。

インストール準備

APIC ソフトウェアのダウンロードや、クラスター導入前提環境の準備

***API Connect* のインストール**

Install Assist (APICUP) を使用したパラメーター定義と ISO ファイル作成、インストール

***API Connect* の構成**

GUI を使用した APIC 間のコンポーネント構成

前提

APIC V2018.4.1 のインストールをはじめるにあたり、以下の事項を前提としています。

1. IBM API Connect バージョン 2018 の要件[2]

- ハイパーバイザーは、製品要件を満たす VMware ESXi 6.0 または 6.5 を使用
- APIC 各コンポーネントのサーバーは、製品最小要件を満たすハードウェアリソースを使用

コンポーネント	プロセッサー	メモリー	ディスクスペース
管理サーバー	4 vCPU	16 GB	250 GB
分析サーバー	2 vCPU	16 GB	200 GB
ポータルサーバー**	4 vCPU	8 GB	150 GB
ゲートウェイサーバー	4 vCPU	8 GB	32 GB

**ポータルサーバーのリソースは、ターゲットとするサイト数に応じて考慮が必要です。詳しくは、IBM Knowledge Center「VMware 環境での開発者ポータルのデプロイ」[3] をご参照ください。

以下は、APIC バージョン 2018 のインストールおよび構成時から必須となる周辺システムです。

- APIC インストールやリストア、アップグレードを行うための Install Assist 導入クライアント
- E メール通知のための SMTP サーバー
- エンドポイント名およびホスト名を解決する DNS サーバー
- クラスター時刻同期のための NTP サーバー

2. 証明書

APIC 各コンポーネントで使用する SSL サーバー証明書は、特に明示的に設定をしない場合、製品のデフォルトの証明書が使用されます。

デフォルトの証明書は、APIC サブシステム（管理サブシステム、分析サブシステム、ポータルサブシステムを指す。以降、同表記はゲートウェイサブシステムを除く。）のインストール時に Install Assist ツールである APICUP インストーラーのコマンドによって自動生成されます。

本ガイドでは、製品のデフォルトの証明書を使用します。ご自身で証明書を用意して適用する場合は、IBM Knowledge Center「証明書の操作」[4] をご参照ください。

3. 各エンドポイントと証明書の適用概要[5]

APIC のインストール時に、各サブシステム毎に APIC 構成に必要な 1 つ以上のエンドポイントを作成して、証明書または TLS 相互通信を構成することになります。（図 1）

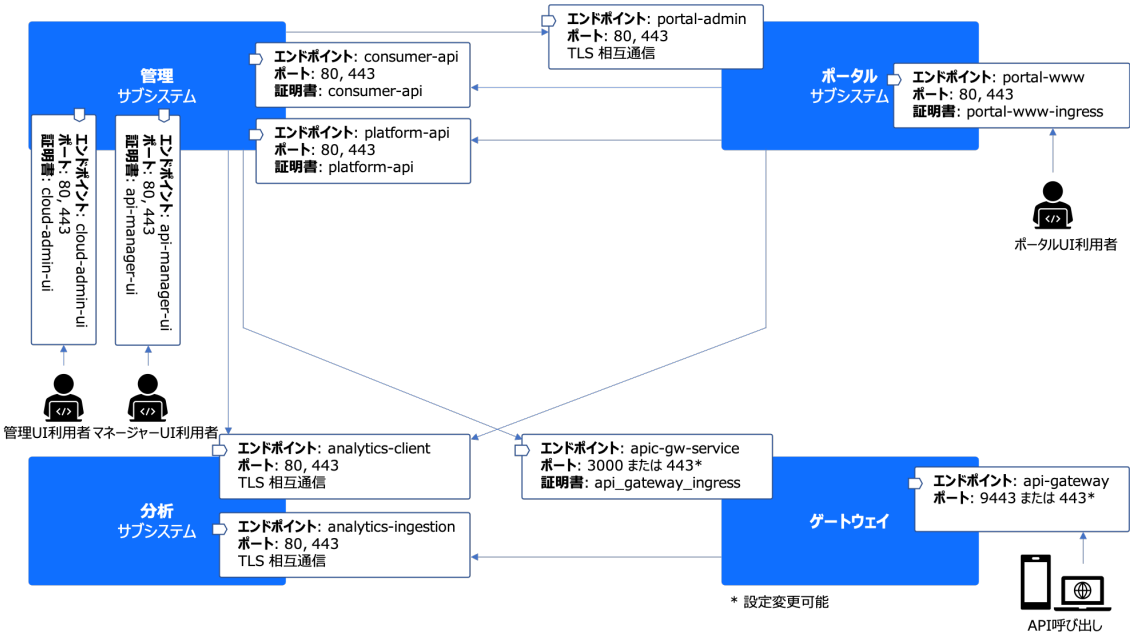


図 1. API Connect サブシステムのエンドポイント、証明書、TLS 相互通信

APIC サブシステムの各エンドポイントは APICUP インストーラーによって構成を行い、APIC サブシステム毎の ISO ファイルに設定します。また、後続の API Connect の構成手順のトポロジー構成において、ゲートウェイも含めたエンドポイント設定をします。

サブシステム	エンドポイント	概要
管理サブシステム	cloud-admin-ui	管理サーバーの Cloud マネージャーUI エンドポイント
	api-manager-ui	管理サーバーの API マネージャーUI エンドポイント
	consumer-api	管理サーバーのコンシューマーAPI 稼働エンドポイント
	platform-api	管理サーバーの管理・プロバイダーAPI 稼働エンドポイント
ポータルサブシステム	portal-admin	管理サーバーとの通信用エンドポイント
	portal-www	ポータル Web サイト URL のエンドポイント
分析サブシステム	analytics-client	管理サーバー、ポータルサーバーからの参照エンドポイント
	analytics-ingestion	ゲートウェイサーバーからのデータプッシュ用エンドポイント
ゲートウェイサブシステム	apic-gw-service	管理サーバーとの通信用エンドポイント
	api-gateway	API 呼び出しエンドポイント

4. ファイアウォール要件[6]

- APIC の各コンポーネント配置は、ゲートウェイサーバーと、ポータルサイトアクセス用に構成するリバース・プロキシーサーバーを DMZ に配置する構成が一般的です。
- 本ガイドはインストール構成の検証を目的として環境を構築しているため、対象サーバーの DMZ への配置ならびにファイアウォールの設定は考慮しておりません。
- 各サブシステムで必要なポート一覧は、「ファイアウォール要件」の文末脚注リンク先をご参照ください。

5. VMware へのインストールと構成のための要件[7]

VMware へデプロイするための要件

ISO を作成するためのユーティリティーをサポートする OS を用意します。APICUP インストーラーは、以下のユーティリティーを使用します。本ガイドでは、Linux にインストールします。

OS	ユーティリティー
Linux	mkisofs
macOS	hdiutil
Windows	mkisofs が使用可能なツール（CDRTools など）

また、SSH ログイン用の暗号鍵を生成するツール（open-ssh など）も別途用意します。

IBM API Connect は、下表のパッケージから構成されます。

本ガイドでは、ファームウェアバージョン **v2018.4.1.5** を使用します。

- Install Assist は、必ずインストールする APIC パッケージのファームウェアと同一のバージョンを使用します。
- APIC は、ゲートウェイサービスを提供するために IBM® DataPower® Gateway を使用します。
- IBM DataPower Gateway のファームウェア・バージョンも、APIC と一致する必要があります。

パッケージ	IBM API Connect サブシステムファイル
IBM API Connect® Management for VMware	management_lts_2018.4.1.5.ova
IBM API Connect Analytics for VMware	analytics_lts_2018.4.1.5.ova
IBM API Connect Developer Portal for VMware	portal_lts_2018.4.1.5.ova
IBM API Connect Install Assist	apicup-linux_lts_v2018.4.1.5
IBM DataPower Gateway Non-production for VMware**	idg2018410.lts.nonprod.ova idg2018415.lts.scrypt4

**本ガイドは Non-production を使用します。本番利用の場合はパッケージを production に読み替えてください。

サブシステム毎に、構成に必要なネットワーク設定情報を定義します。
本ガイドでは、各サーバーを 2 つのインターフェース（eth0, eth1）を使用して、ひとつの考え方としてクラスター内・クラスター間通信と、API/UI の通信をわけて構成します。

なお、分析サブシステムはエンドポイントが 1 つのインターフェースしか持たないため、ここではクラスター内・クラスター間通信に分類してインターフェース（eth0）を使用するように構成しています。

各サブシステムのクラスター内・クラスター間通信用（traffic-iface）に eth0 を使用します。（図 2）

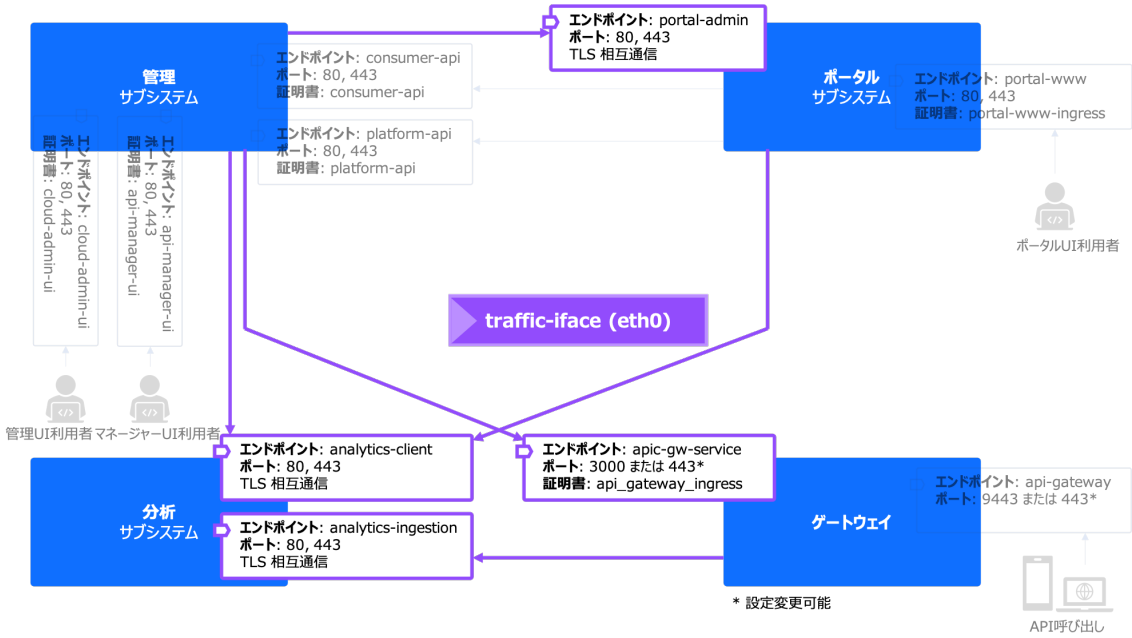


図 2. 各サブシステムクラスター間通信のインターフェース

各サブシステムの API や UI 通信用（public-iface）に eth1 を使用します。（図 3）

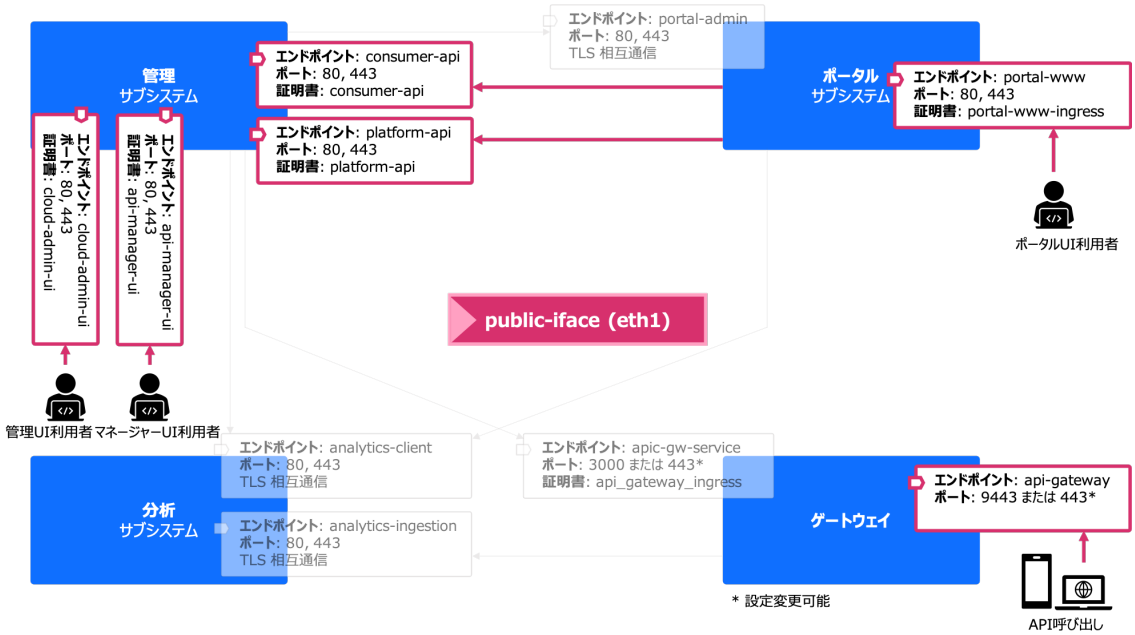


図 3. 各サブシステム API/UI 通信のインターフェース

サーバー毎にネットワーク設定情報を定義します。

- ホスト名は、クラスター内・クラスター間通信をおこなう eth0 インターフェースに対応するように定義します。
- ホスト名と FQDN は、小文字で入力する必要があります。
- 指定するホスト名には、ワイルドカード別名またはホスト別名が必要です。これにより、別々のエンドポイントが確実に連携して動作するようになります。（例: *.ドメイン名）
- eth1 インターフェースは静的経路を定義しています。

カテゴリー	FQDN		IP アドレス（CIDR）	
	ホスト名	ドメイン名	eth0 インターフェース	eth1 インターフェース
管理サブシステム	mgmt01	apic.com	9.68.85.87/24	9.68.84.194/24
	mgmt02	apic.com	9.68.85.88/24	9.68.84.195/24
	mgmt03	apic.com	9.68.85.89/24	9.68.84.196/24
分析サブシステム	analyt01	apic.com	9.68.85.90/24	N/A
	analyt02	apic.com	9.68.85.91/24	N/A
	analyt03	apic.com	9.68.85.92/24	N/A
ポータルサブシステム	ptl01	apic.com	9.68.85.93/24	9.68.83.14/24
	ptl02	apic.com	9.68.85.94/24	9.68.83.15/24
	ptl03	apic.com	9.68.85.95/24	9.68.83.16/24
ゲートウェイサブシステム	gwy01	apic.com	9.68.85.96/24	9.68.85.99/24
	gwy02	apic.com	9.68.85.97/24	9.68.85.100/24
	gwy03	apic.com	9.68.85.98/24	9.68.85.101/24

カテゴリー	IP アドレス
DNS	9.68.85.106
デフォルト・ゲートウェイ	9.68.85.1

VMware の構成要件

APIC は NFS 上にはデプロイできません。

APIC V2018 の OVA は、Kubernetes によって環境が構成されています。

- Kubernetes ポッドと Kubernetes サービス・ネットワーク用に、IP アドレスレンジが予約されています。
- 各 APIC サブシステムのホスト IP アドレスはこれら IP アドレスのレンジ外である必要があります。
- ホスト IP アドレスが競合する場合は、初期構成時にこれら IP アドレスのレンジを変更することができます。

Kubernetes ポッドおよびサービス・ネットワーク予約 IP アドレスレンジ
172.16.0.0/16 および 172.17.0.0/16

APIC サブシステムには、dev モードと standard モードオプションがあります。

- dev モードは、シングル構成の開発・テスト用のデプロイメントオプションです。クラスター構成はサポートしません。
- standard モードは、クラスター構成のプロダクション用のデプロイメントオプションです。

本ガイドはクラスター構成ですので、standard モードオプションでデプロイします。明示的に設定しない場合、サブシステムのデフォルトモードは dev モードになります。

6. ロードバランサー構成[8]

APIC クラスター構成では、ゲートウェイを含め各サブシステム 3 ノード以上のクラスターとロードバランサーで構成をおこなうことが推奨されています。

- クラスターとして 3 ノード以上が必要な理由は、APIC サブシステムが High Availability (HA) 構成において quorum ベースの技術を採用しているためです。
- ロードバランサーを構成する理由は、各クラスターが共通で持つエンドポイントに対する通信を、クラスター内に振り分けるようにするために、各 APIC サブシステムの前段にロードバランサーを配置する必要があるためです。

各エンドポイントは、ロードバランサーの IP アドレスで解決するよう DNS 登録します。

APIC サブシステム間の TLS 相互通信をサポートするために、ロードバランサーは **SSL パススルー**かつ **L4 ロードバランシング**で構成します。(詳しくは、文末脚注 8 の参考文献を参照)

- 本ガイドでは、ロードバランサーは HAProxy を使用して構成しています。
- 図 1 で前述したすべてのエンドポイントを経由するように、ロードバランサーを構成します。（図 4）
- 本ガイドで構成した HAProxy の内容は、付録に添付していますのでご参考ください。

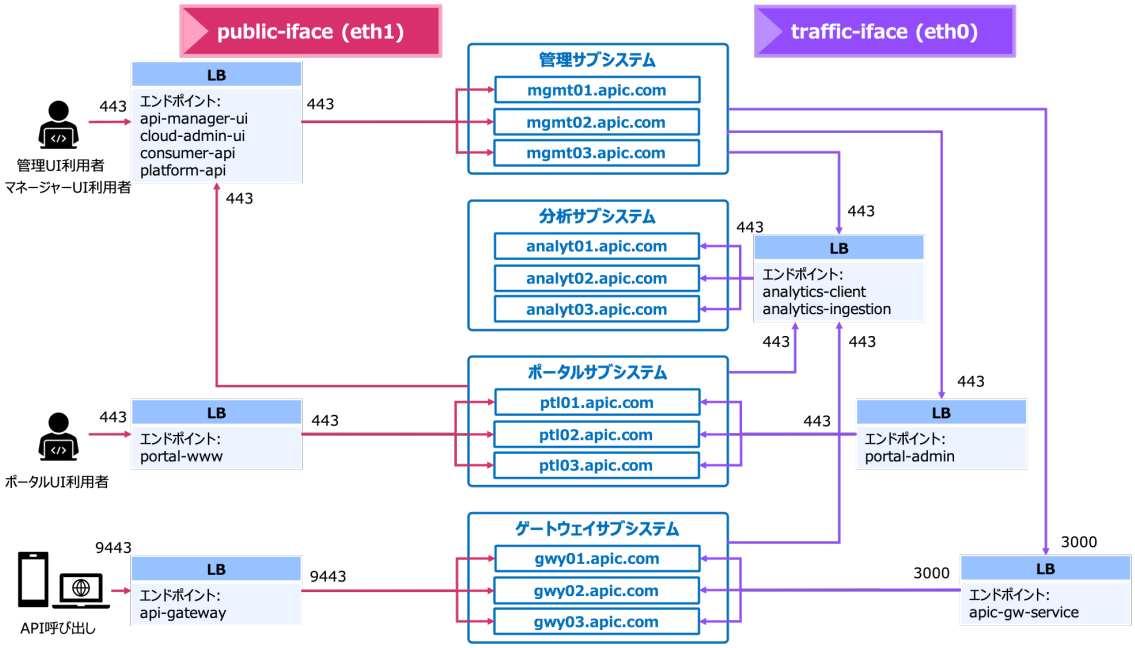


図 4. ロードバランサー論理構成

インストール準備[9]

1. IBM API Connect パッケージのダウンロード

- Passport Advantage®および IBM Fix Central から、導入 OS に合う IBM API Connect Install Assist パッケージと、最新の IBM API Connect パッケージをダウンロードします。
- 同サイトから、IBM DataPower Gateway の OVA ファイルと Fix Pack をダウンロードします。

2. Install Assist のインストール

ダウンロードした Install Assist パッケージのファイル「apicup-linux_lts_v2018.4.1.5」の名前を「apicup」（実行形式）にリネームし、任意のディレクトリー（例えば、/usr/local/bin）に配置して Path をとおします。

```
$ ls -l /home/admin/
total 38420
-rw-r--r-- 1 admin admin 39339008   5 月  29 14:02 apicup-linux_lts_v2018.4.1.5
$
$ mv /home/admin/apicup-linux_lts_v2018.4.1.5 /usr/local/bin/apicup
$
$ ls -l /usr/local/bin
$ total 38420
$ -rw-r--r-- 1 admin admin 39339008   5 月  29 14:02 apicup
$
$ tail .profile
...

PATH="$HOME/bin:$HOME/.local/bin:$PATH"
PATH=$PATH:/usr/local/bin
```

パーミッションを変更して実行権限を付与します。

```
$ chmod +x apicup
$ total 38420
$ -rwxr-xr-x 1 admin admin 39339008   5 月  29 14:04 apicup
```

以下のコマンドでバージョンを確認します。初回実行時はライセンス承諾を行います。

```
$ apicup version
Please review the license for API Connect by running "apic licenses" command or accessing
https://ibm.biz/apictoolkitlic.
Accept the license for API Connect [Y/N] Y
APIConnect 2018.4.1.5-ifix1.0
Installer 5.0.0
```

3. プロジェクト・ディレクトリーの作成

Install Assist をインストールしたクライアントで任意のディレクトリーを作成し、初期化をおこないます。ここでは、プロジェクト名を「apic415」としています。

```
$ mkdir apic415
$
$ apicup init apic415/
Creating project in apic415/ directory
```

重要

APICUP のプロジェクトは単一のプロジェクト・ディレクトリーを作成、使用してください。複数のプロジェクトは複数の証明書チェーンのアンマッチを引き起こします。

APIC の初期インストールで作成した APICUP のプロジェクトは、APIC のデータベースのリストアおよびアップグレードにも使用します。

プロジェクト・ディレクトリーには、クラスターに関する永続的な情報が含まれているため、このプロジェクト・ディレクトリー無しにはリストアやアップグレードが実行できません。

いつでもこのプロジェクト・ディレクトリーを取得できる場所にバックアップしておくことを推奨します。

初期化をおこなうと、「apiconnect-up.yml」ファイルがプロジェクト・ディレクトリーに生成されます。

```
$ cd apic415/
$
$ ls -l
total 4
-rw-rw---- 1 admin admin 174  5 月  29 14:14 apiconnect-up.yml
```

重要

apiconnect-up.yml ファイルは、安全で永続的な場所に保管する必要があります。このファイルには、テキスト形式で公開されるパスワード情報およびその他の情報が入っています。プロジェクト・ディレクトリーが安全であることを確認してください。

ISO 作成ユーティリティー「mkisofs」の実行ファイルに Path を通します。

SSH ログイン用の SSH 暗号鍵を生成します。

```
$ ssh-keygen -t rsa
/home/admin/.ssh/id_rsa
/home/admin/.ssh/id_rsa.pub
```

4. DNS 登録

エンドポイントの構成と登録

エンドポイントは、以下のフォーマットで FQDN、すべて小文字で指定します。

endpointname.domain

各サブシステムのエンドポイントがクラスター内のサーバーに割り振られるように、すべてロードバランサーの IP アドレスを指定して DNS 登録します。(図 4 を参照)

エンドポイント	エンドポイント名	値
cloud-admin-ui	cloud-admin-ui.apic.com	mgmt*.apic.com にルーティングする LB VIP
api-manager-ui	api-manager-ui.apic.com	mgmt*.apic.com にルーティングする LB VIP
consumer-api	consumer-api.apic.com	mgmt*.apic.com にルーティングする LB VIP
platform-api	platform-api.apic.com	mgmt*.apic.com にルーティングする LB VIP
portal-admin	portal-admin.apic.com	ptl*.apic.com にルーティングする LB VIP
portal-www	portal-www.apic.com	ptl*.apic.com にルーティングする LB VIP
analytics-client	analytics-client.apic.com	analyt*.apic.com にルーティングする LB VIP
analytics-ingestion	analytics-ingestion.apic.com	analyt*.apic.com にルーティングする LB VIP
apic-gw-service	apic-gw-service.apic.com	gwy*.apic.com にルーティングする LB VIP
api-gateway	api-gateway.apic.com	gwy*.apic.com にルーティングする LB VIP

ホスト名は、以下のフォーマットで FQDN、すべて小文字で DNS 登録します。

インターフェースは、eth0 インタフェースを指定しています。

hostname.domain

サブシステム	ホスト名	値
管理サブシステム	mgmt01.apic.com	9.68.85.87
	mgmt02.apic.com	9.68.85.88
	mgmt03.apic.com	9.68.85.89
分析サブシステム	analyt01.apic.com	9.68.85.90
	analyt02.apic.com	9.68.85.91
	analyt03.apic.com	9.68.85.92
ポータルサブシステム	ptl01.apic.com	9.68.85.93
	ptl02.apic.com	9.68.85.94
	ptl03.apic.com	9.68.85.95
ゲートウェイサブシステム	gwy01.apic.com	9.68.85.96
	gwy02.apic.com	9.68.85.97
	gwy03.apic.com	9.68.85.98

API Connect のインストール

1. VMware 環境での管理サブシステムのデプロイ[10]

必要情報	値
サーバー1 の IP アドレス（eth0） サーバー2 の IP アドレス（eth0） サーバー3 の IP アドレス（eth0）	9.68.85.87/24 9.68.85.88/24 9.68.85.89/24
サーバー1 のホスト名 サーバー2 のホスト名 サーバー3 のホスト名	mgmt01.apic.com mgmt02.apic.com mgmt03.apic.com
サーバー・ドメイン名	apic.com
サーバー1 の IP アドレス（eth1） サーバー2 の IP アドレス（eth1） サーバー3 の IP アドレス（eth1）	9.68.84.194/24 9.68.84.195/24 9.68.84.196/24
DNS サーバー	9.68.85.106
デフォルト・ゲートウェイ	9.68.85.1
イーサネット・インターフェース名	eth0, eth1
Platform API エンドポイント	platform-api.apic.com
Consumer API エンドポイント	consumer-api.apic.com
Cloud Admin UI エンドポイント	cloud-admin-ui.apic.com
API Manager UI エンドポイント	api-manager-ui.apic.com

1.1. ISO ファイルの生成

1. APICUP 実行環境で、プロジェクト・ディレクトリーに移動します。

```
$ cd apic415
```

2. 管理サブシステムを作成します。

```
$ apicup create subsys mgmt management
```

- mgmt は、作成する管理サブシステム ID です。スペースを含まない小文字の英数字である必要がありあす。
- management は、管理サブシステムを作成することを表します。

これにより、プロジェクト・ディレクトリーの初期化で生成した「apiconnect-up.yml」ファイルに、各サブシステムのインストール・パラメーターが追記されていくことになります。

3. apiconnect-up.yml ファイルを構成します。

以下のコマンドで、 管理サブシステムの現行値を確認することができます。

```
$ apicup subsys get mgmt
```

まだサブシステムを構成していない場合は、このコマンドによってエラーが返されます。また、値を更新していない場合は、使用できるデフォルト値がある場合、デフォルト値がリストされます。以下の例のようになります。

```
$ apicup subsys get mgmt
Appliance settings
==================

Name                    Value       Description
----                    -----       ------
additional-cloud-init-file          (Optional) Path to additional cloud-init yml file
data-device             sdb         VM disk device (usually `sdb` for SCSI or `vdb` for VirtIO)
default-password                    (Optional) Console login password for `apicadm` user
dns-servers             []          List of DNS servers
extra-values-file                   (Optional) Path to additional configuration yml file
k8s-pod-network         172.16.0.0/16   (Optional) CIDR for pods within the appliance
k8s-service-network     172.17.0.0/16   (Optional) CIDR for services within the appliance
mode                    dev
public-iface            eth0        Device for API/UI traffic (Eg: eth0)
search-domain           []          List for DNS search domains
ssh-keyfiles            []          List of SSH public keys files
traffic-iface           eth0        Device for cluster traffic (Eg: eth0)

...
```

```
 ...

 Subsystem settings
==================

Name                                Value          Description
----                                -----          ------
az-name                             default-az     Availability Zone name
cassandra-backup-auth-pass                         (Optional) Server password for DB backups
cassandra-backup-auth-user                         (Optional) Server username for DB backups
cassandra-backup-host                              (Optional) FQDN for DB backups server
cassandra-backup-path               /backups       (Optional) path for DB backups server
cassandra-backup-port               22             (Optional) Server port for DB backups
cassandra-backup-protocol           sftp           (Optional) Protocol for DB backups (sftp/ftp/objstore)
cassandra-backup-schedule           0 0 * * *      (Optional) Cron schedule for DB backups
cassandra-max-memory-gb             4              Memory limit for DB
cassandra-postmortems-auth-pass                    (Optional) Server Password for DB metrics
cassandra-postmortems-auth-user                    (Optional) Server username for DB metrics server
cassandra-postmortems-host                         (Optional) FQDN for DB metrics server
cassandra-postmortems-path          /postmortems   (Optional) path for DB metrics server
cassandra-postmortems-port          22             (Optional) Server port for DB metrics
cassandra-postmortems-schedule
                                    0 0 * * *      (Optional) Cron schedule for DB metrics
cross-az-peers                      []             (Optional) IP addresses of nodes in other AZs
migration-admin-auth-pass                          (Optional) Cloud admin password
migration-admin-auth-user                          (Optional) Cloud admin username
migration-backup-auth-pass                         (Optional) Server password for V5 data
migration-backup-auth-user                         (Optional) Server username for V5 data
migration-backup-host                              (Optional) Server FQDN for V5 data
migration-backup-input-path                        (Optional) Server path for V5 input mapping
migration-backup-logs-path                         (Optional) Server path for migration logs
migration-backup-path                              (Optional) Server path for V5 backup data
migration-backup-port               22             (Optional) Server port for V5 data
migration-enabled                   false          (Optional) Enable/Authorize migration jobs
migration-realm                                    (Optional) Cloud admin user login realm


Endpoints
=========

Name                                Value          Description
----                                -----          ------
api-manager-ui                                     FQDN of API manager UI endpoint
cloud-admin-ui                                     FQDN of Cloud admin endpoint
consumer-api                                       FQDN of consumer API endpoint
platform-api                                       FQDN of platform API endpoint


Error: Subsystem validation failure. Run with --validate to see details
$
```

a. 管理サーバーにログインするためのハッシュ・パスワードを作成します。mkpasswd を使用します。

```
$ mkpasswd --method=sha-512 --rounds=4096 Passw0rd!
```

ハッシュ値が出力されるのでコピーします。

```
$6$rounds=4096$tOYaF7YcJ$KsXRqYkzH59nnif1DqrXYkTjN1w7UY/EEjidXcysy6AgmdLth9bx7QuNpXvvLFttg29.h6Z/dPKhvvbTlW/RM.
```

b. ハッシュ・パスワードを設定します。

フォーマット: apicup subsys set *mgmt* default-password='*hashed_password*'

```
$ apicup subsys set mgmt default-password='$6$rounds=4096$tOYaF7YcJ$KsXRqYkzH59nnif1DqrXYkTjN1w7UY/EEjidXcysy6AgmdLth9bx7QuNpXvvLFttg29.h6Z/dPKhvvbTlW/RM.'
```

c. DNS サーバーを設定します。

```
$ apicup subsys set mgmt dns-servers=9.68.85.106
```

d. モードオプションを設定します。

```
$ apicup subsys set mgmt mode=standard
```

e. サーチドメインを設定します。

```
$ apicup subsys set mgmt search-domain=apic.com
```

f. SSH 公開鍵ファイルのパスを設定します。

```
$ apicup subsys set mgmt ssh-keyfiles=/home/admin/.ssh/id_rsa.pub
```

[オプション] バックアップが必要な場合は、スケジュール済みバックアップを構成します。[11]

```
$ apicup subsys set mgmt cassandra-backup-path=/home/sftp-user/sftp/backups
```

```
$ apicup subsys set mgmt cassandra-backup-host=apic520.apic.com
```

```
$ apicup subsys set mgmt cassandra-backup-auth-user=sftp-user
```

```
$ apicup subsys set mgmt cassandra-backup-auth-pass=sftp-user
```

ヒント

- ベスト・プラクティスとして、サービス間の同期を確保するために、管理サブシステムとポータルサブシステムの両方を同時にバックアップしてください。
- スケジュールは、ローカル・タイム・ゾーンを例えば JST に変更しても **UTC** 時間で稼働しますのでご注意ください。

[オプション] syslog によるログ収集が必要な場合は、syslog の構成をおこなうことができます。[12]

g. platform API エンドポイントを設定します。

```
$ apicup subsys set mgmt platform-api=platform-api.apic.com
```

h. consumer API エンドポイントを設定します。

```
$ apicup subsys set mgmt consumer-api=consumer-api.apic.com
```

i. cloud admin UI エンドポイントを設定します。

```
$ apicup subsys set mgmt cloud-admin-ui=cloud-admin-ui.apic.com
```

j. API Manager UI エンドポイントを設定します。

```
$ apicup subsys set mgmt api-manager-ui=api-manager-ui.apic.com
```

4. ホストを追加します。

フォーマット: `apicup hosts create mgmt *hostname.domainname hd_password*`

hd_password は、Linux Unified Key Setup によって管理サービスのストレージを暗号化するために使用されるパスワードです。このパスワードは保管時にハッシュされます。

管理サーバー1

```
$ apicup hosts create mgmt mgmt01.apic.com Passw0rd!
```

管理サーバー2

```
$ apicup hosts create mgmt mgmt02.apic.com Passw0rd!
```

管理サーバー3

```
$ apicup hosts create mgmt mgmt03.apic.com Passw0rd!
```

5. インターフェースを作成します。

フォーマット: apicup iface create *mgmt hostname.domainname physical_network_id host_ip_address/subnet_mask gateway_ip_address*

フォーマットの性質上、追加したインターフェースにもデフォルト・ゲートウェイを設定することになるため、後の手順で静的経路を設定します。

管理サーバー1

```
$ apicup iface create mgmt mgmt01.apic.com eth0 9.68.85.87/255.255.255.0 9.68.85.1

$ apicup iface create mgmt mgmt01.apic.com eth1 9.68.84.194/255.255.255.0 9.68.84.1
```

管理サーバー2

```
$ apicup iface create mgmt mgmt02.apic.com eth0 9.68.85.88/255.255.255.0 9.68.85.1

$ apicup iface create mgmt mgmt02.apic.com eth1 9.68.84.195/255.255.255.0 9.68.84.1
```

管理サーバー3

```
$ apicup iface create mgmt mgmt03.apic.com eth0 9.68.85.89/255.255.255.0 9.68.85.1

$ apicup iface create mgmt mgmt03.apic.com eth1 9.68.84.196/255.255.255.0 9.68.84.1
```

6. public-iface と traffic-iface を設定します。

```
$ apicup subsys set mgmt public-iface=eth1
$ apicup subsys set mgmt traffic-iface=eth0
```

7. 設定したホストを確認します。

```
$ apicup hosts list mgmt
mgmt01.apic.com
    Device  IP/Mask                 Gateway
    eth0    9.68.85.87/255.255.255.0    9.68.85.1
    eth1    9.68.84.194/255.255.255.0   9.68.84.1
mgmt02.apic.com
    Device  IP/Mask                 Gateway
    eth0    9.68.85.88/255.255.255.0    9.68.85.1
    eth1    9.68.84.195/255.255.255.0   9.68.84.1
mgmt03.apic.com
    Device  IP/Mask                 Gateway
    eth0    9.68.85.89/255.255.255.0    9.68.85.1
    eth1    9.68.84.196/255.255.255.0   9.68.84.1
```

8. 追加したインターフェースに静的経路を設定します。[13]

```
$ touch mgmt-cloud-init.yml
$
$ vi mgmt-cloud-init.yml
$
$ cat mgmt-cloud-init.yml
bootcmd:
 - ip route add 0.0.0.0/0 via 9.68.84.1 dev eth1
$
$ apicup subsys set mgmt additional-cloud-init-file mgmt-cloud-init.yml
```

9. 設定した構成を検証します。すべての設定値にチェックマークが付いていることを確認します。

```
$ apicup subsys get mgmt --validate
Appliance settings
==================

Name                          Value
----                          -----
additional-cloud-init-file    mgmt-cloud-init.yml                                ✓
data-device                   sdb                                                ✓
default-password
              $6$rounds=4096$tOYaF7YcJ$KsXRqYkzH59nnif1DqrXYkTjN1w7UY/EEjidXcysy6Ag
mdLth9bx7QuNpXvvLFttg29.h6Z/dPKhvvbTlW/RM.                                       ✓
dns-servers                   [9.68.85.106]                                      ✓
extra-values-file                                                                ✓
k8s-pod-network               172.16.0.0/16                                      ✓
k8s-service-network           172.17.0.0/16                                      ✓
mode                          standard                                           ✓
public-iface                  eth1                                               ✓
search-domain                 [apic.com]                                         ✓
ssh-keyfiles                  [/home/admin/.ssh/id_rsa.pub]                      ✓
traffic-iface                 eth0                                               ✓

...
```

```
...

Subsystem settings
==================

Name                              Value
----                              -----
az-name                           default-az                    ✓
cassandra-backup-auth-pass        c2Z0cC11c2Vy                  ✓
cassandra-backup-auth-user        sftp-user                     ✓
cassandra-backup-host             apic520.apic.com              ✓
cassandra-backup-path             /home/sftp-user/sftp/backups  ✓
cassandra-backup-port             22                            ✓
cassandra-backup-protocol         sftp                          ✓
cassandra-backup-schedule         0 0 * * *                     ✓
cassandra-max-memory-gb           4                             ✓
cassandra-postmortems-auth-pass                                 ✓
cassandra-postmortems-auth-user                                 ✓
cassandra-postmortems-host                                      ✓
cassandra-postmortems-path        /postmortems                  ✓
cassandra-postmortems-port        22                            ✓
cassandra-postmortems-schedule    0 0 * * *                     ✓
cross-az-peers                    []                            ✓
migration-admin-auth-pass                                       ✓
migration-admin-auth-user                                       ✓
migration-backup-auth-pass                                      ✓
migration-backup-auth-user                                      ✓
migration-backup-host                                           ✓
migration-backup-input-path                                     ✓
migration-backup-logs-path                                      ✓
migration-backup-path                                           ✓
migration-backup-port             22                            ✓
migration-enabled                 false                         ✓
migration-realm                                                 ✓


Endpoints
=========

Name                              Value
----                              -----
api-manager-ui                    api-manager-ui.apic.com       ✓
cloud-admin-ui                    cloud-admin-ui.apic.com       ✓
consumer-api                      consumer-api.apic.com         ✓
platform-api                      platform-api.apic.com         ✓
```

10. ISO ファイルを作成します。--out パラメーターで指定した値のディレクトリーが作成されます。

フォーマット: apicup subsys install *mgmt* --out *mgmt*plan-out

```
$ apicup subsys install mgmt --out mgmtplan-out
```

11. 作成された ISO ファイルを確認します。

各管理サーバーに対する 3 つの ISO ファイルが作成されたことを確認します。

```
$ cd mgmtplan-out/
$
$ ls -ltr
total 1328
-rw-------  1 admin admin     162  6 月  22 17:04 meta.yml
-rw-------  1 admin admin    1679  6 月  22 17:04 appliance-client.key
-rw-------  1 admin admin    1151  6 月  22 17:04 appliance-client.crt
-rw-------  1 admin admin    1074  6 月  22 17:04 appliance-client-ca.pem
drwxr-x---  2 admin admin    4096  6 月  22 17:04 mgmt01.apic.com
-rw-rw-r--  1 admin admin  440320  6 月  22 17:04 mgmt01.apic.com.iso
drwxr-x---  2 admin admin    4096  6 月  22 17:04 mgmt02.apic.com
-rw-rw-r--  1 admin admin  440320  6 月  22 17:04 mgmt02.apic.com.iso
drwxr-x---  2 admin admin    4096  6 月  22 17:04 mgmt03.apic.com
-rw-rw-r--  1 admin admin  440320  6 月  22 17:04 mgmt03.apic.com.iso
-rw-------  1 admin admin    3791  6 月  22 17:04 instructions.txt
```

重要

ホスト名、**エンドポイント名**および**インターフェース設定**は、APIC サブシステムのインストール後に値を変更して ISO を再作成し、再読み込みをしても設定が反映されません。もう一度、システムをインストールし直す必要がありますのでご注意ください。

1.2. OVF テンプレートのデプロイ

1. VMware の vSphere Web Client にログインします。

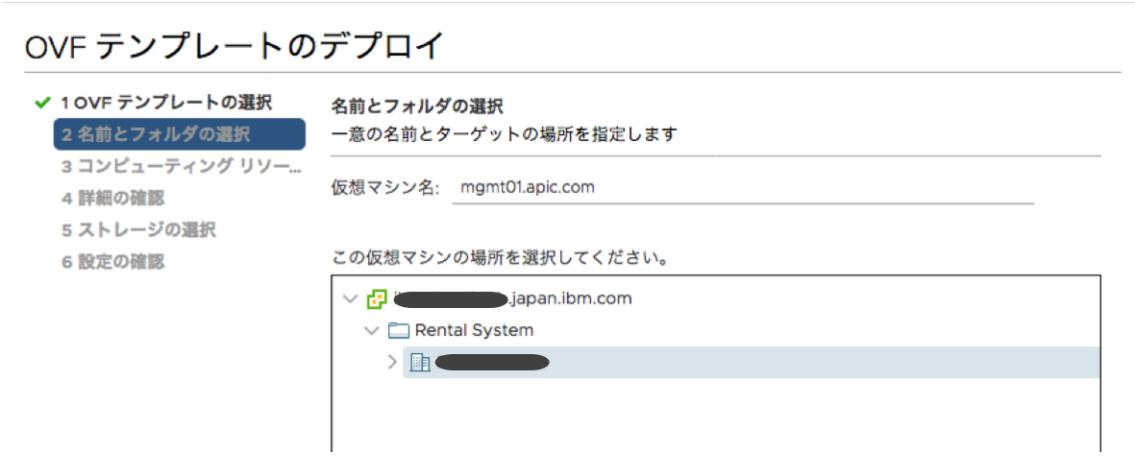
2. vSphere ナビゲーターを使用して、OVA ファイルをデプロイするディレクトリーに移動します。

3. ディレクトリーを右クリックし、**OVF テンプレートのデプロイ**を選択します。

4. OVF テンプレートのデプロイ・ウィザードを完了します。

   a. インストール準備でダウンロードした **management_lts_2018.4.1.5.ova** テンプレートを選択します。**NEXT** をクリックします。

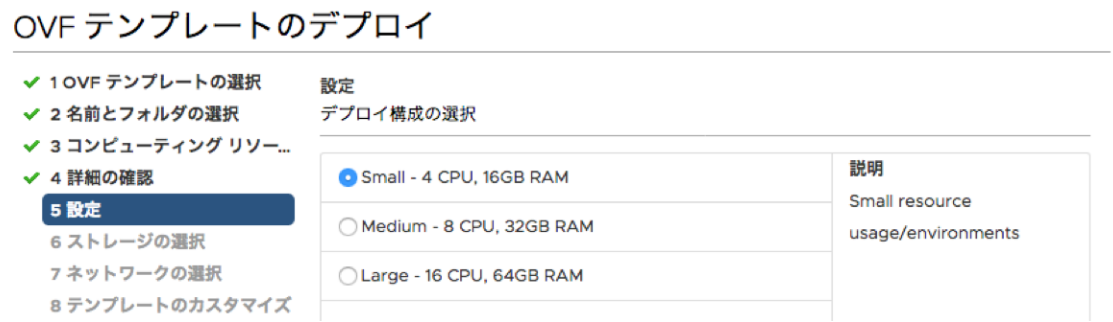
   b. ファイルの名前とロケーションを入力します。**NEXT** をクリックします。



c. テンプレートのリソースを選択します。**NEXT** をクリックします。

d. テンプレートの詳細を確認します。**NEXT** をクリックします。

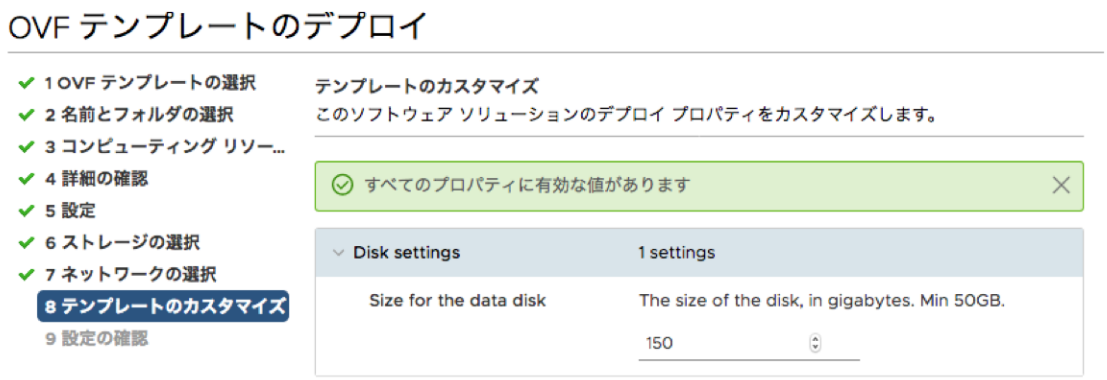
e. 構成のサイズを選択します。**NEXT** をクリックします。



f. ストレージ設定を選択します。**NEXT** をクリックします。

g. ネットワークを選択します。**NEXT** をクリックします。

h. 必要に応じてテンプレートをカスタマイズします。（データディスクのサイズ）



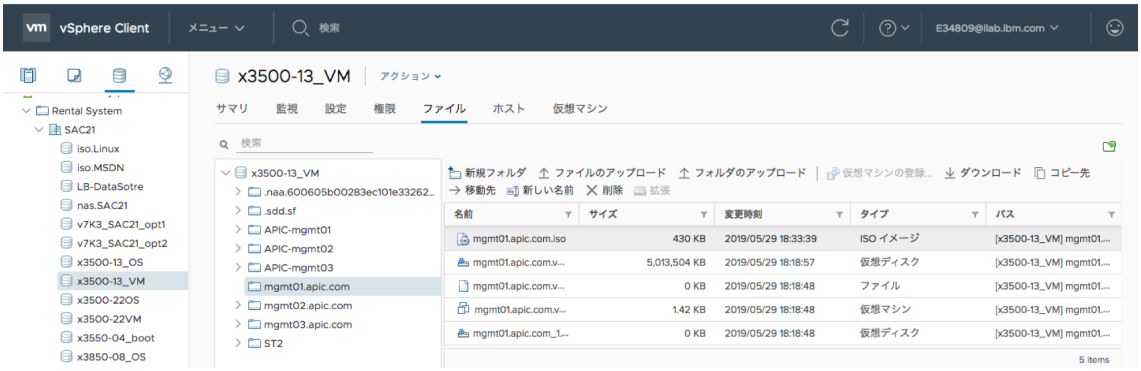
i. 設定を確認して **FINISH** をクリックし、仮想マシンをデプロイします。

1.3. ISO のデータ・ストアへのアップロード

1. ナビゲーターで**ストレージ**タブを選択します。

2. データ・ストアに移動します。

3. **ファイルのアップロード**を選択します。



4. ポップアップから、作成した ISO ファイルを選択します。

5. ISO ファイルを、デプロイした各管理サーバーのデータ・ストアにアップロードします。

1.4. ISO ファイルを使用したテンプレートの構成

1. ナビゲーターで**仮想マシンおよびテンプレート**を選択します。

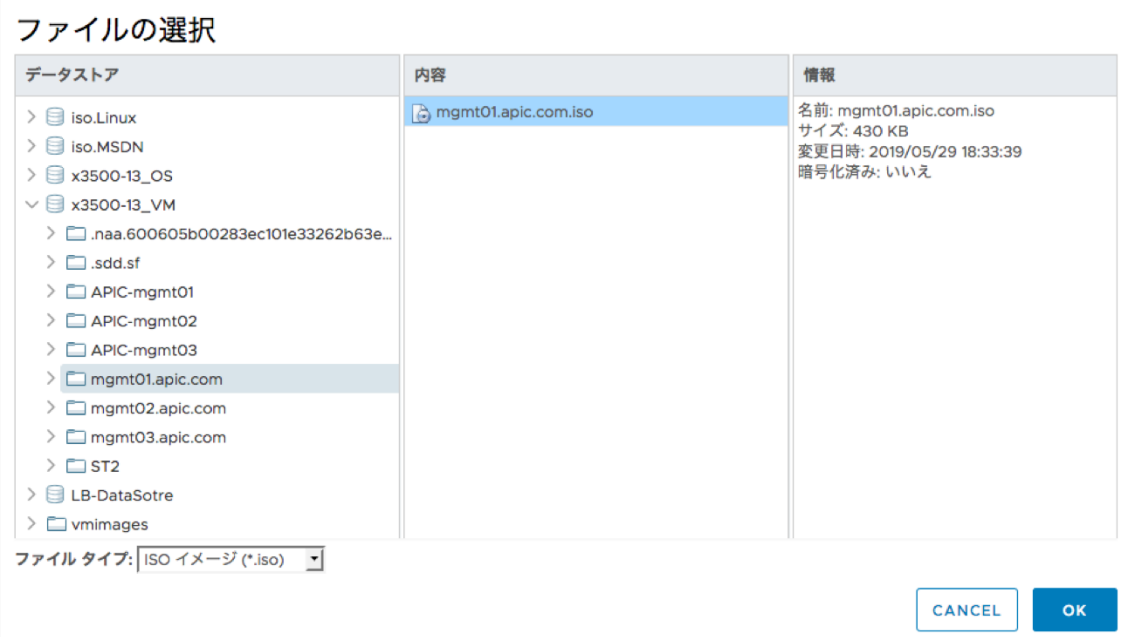
2. デプロイした仮想マシンを見つけて選択します。

3. 右クリックして**アクション**から**設定の編集...**を選択します。

4. **仮想ハードウェア**タブで、**CD/DVD ドライブ 1** を展開します。

5. **接続...(電源オン時に接続)**をチェックします。

6. **データストア ISO ファイル**を選択して、アップロードしたファイルを見つけて **OK** を選択します。



7. アイコン・バーで再生ボタンを選択して、仮想マシンを開始します。システムの可用性およびダウンロードの速度によっては、インストールが完了するまでに数分間かかることがあります。

管理サーバー2，管理サーバー3 についても 1.2.から同様の手順で構成します。

1.5. インストールの状況確認

1. SSH 公開鍵のパスを指定したクライアントから SSH ツールを使用して管理サーバーにログインします。

yes を選択して、接続を続行します。 ホスト名がホストのリストに自動的に追加されます。

```
$ ssh 9.68.85.87 -l apicadm
Enter passphrase for key '/home/admin/.ssh/id_rsa':
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-145-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

7 packages can be updated.
0 updates are security updates.


Last login: Thu Jun  6 00:58:34 2019 from 9.68.85.106

apicadm@mgmt01: ~$
```

2. apic status コマンドを実行して、インストールが完了したこと、およびシステムが正常に稼働していることを確認します。

> **ヒント**
>
> システム可用性によりますが、管理サブシステムの初回の構成が完了するまでにおおよそ 1 時間前後かかります。

```
$ sudo apic status
[INFO] Log level: info

Cluster members:
- mgmt01.apic.com (9.68.85.87)
  Type: BOOTSTRAP_MASTER
  Install stage: DONE
  Upgrade stage: UPGRADE_DONE
  Subsystem detail:
  Docker status:
    Systemd unit: running
  Kubernetes status:
    Systemd unit: running
    Kubelet version: mgmt01 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
  Etcd status: pod etcd-mgmt01 in namespace kube-system has status Running
  Addons: calico, dns, helm, kube-proxy, metrics-server, nginx-ingress,
- mgmt02.apic.com (9.68.85.88)
  Type: MASTER
  Install stage: DONE
  Upgrade stage: UPGRADE_DONE
  Subsystem detail:
  Docker status:
    Systemd unit: running
  Kubernetes status:
    Systemd unit: running
    Kubelet version: mgmt02 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
  Etcd status: pod etcd-mgmt02 in namespace kube-system has status Running
  Addons: calico, kube-proxy, nginx-ingress,
- mgmt03.apic.com (9.68.85.89)
  Type: MASTER
  Install stage: DONE
  Upgrade stage: UPGRADE_DONE
  Subsystem detail:
  Docker status:
    Systemd unit: running
  Kubernetes status:
    Systemd unit: running
    Kubelet version: mgmt03 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
  Etcd status: pod etcd-mgmt03 in namespace kube-system has status Running
  Addons: calico, kube-proxy, nginx-ingress,

...
```

...

Etcd cluster state:
- etcd member name: mgmt01.apic.com, member id: 1600303482436774943, cluster id: 16312812273776362902, leader id: 3575094720705846274, revision: 126175, version: 3.2.26
- etcd member name: mgmt02.apic.com, member id: 3575094720705846274, cluster id: 16312812273776362902, leader id: 3575094720705846274, revision: 126175, version: 3.2.26
- etcd member name: mgmt03.apic.com, member id: 2167557956339866796, cluster id: 16312812273776362902, leader id: 3575094720705846274, revision: 126175, version: 3.2.26

Pods Summary:

NODE	NAMESPACE	NAME	READY	STATUS	REASON
mgmt03	default	apiconnect-a7s-proxy-bc767f6b7-89dwx	1/1	Running	
mgmt01	default	apiconnect-a7s-proxy-bc767f6b7-8gcx7	1/1	Running	
mgmt02	default	apiconnect-a7s-proxy-bc767f6b7-m2csn	1/1	Running	
mgmt01	default	apiconnect-apiconnect-cc-66ffv	1/1	Running	
mgmt02	default	apiconnect-apiconnect-cc-backup-1559779200-29jkr	0/1	Succeeded	
mgmt02	default	apiconnect-apiconnect-cc-cassandra-stats-1559773800-ql7jn	0/1	Succeeded	
mgmt03	default	apiconnect-apiconnect-cc-cassandra-stats-1559777400-qgx86	0/1	Succeeded	
mgmt03	default	apiconnect-apiconnect-cc-cassandra-stats-1559781000-tg28x	0/1	Succeeded	
mgmt03	default	apiconnect-apiconnect-cc-hm5r2	1/1	Running	
mgmt02	default	apiconnect-apiconnect-cc-repair-1559782800-t98f9	1/1	Running	
mgmt02	default	apiconnect-apiconnect-cc-znqhr	1/1	Running	
mgmt03	default	apiconnect-apim-schema-init-job-wdj6k	0/1	Succeeded	
mgmt03	default	apiconnect-apim-v2-5b95f8c75-8n929	1/1	Running	
mgmt02	default	apiconnect-apim-v2-5b95f8c75-8pzth	1/1	Running	
mgmt01	default	apiconnect-apim-v2-5b95f8c75-vv4qk	1/1	Running	
mgmt01	default	apiconnect-client-dl-srv-69b985b9ff-64jfl	1/1	Running	
mgmt03	default	apiconnect-client-dl-srv-69b985b9ff-drvwk	1/1	Running	
mgmt02	default	apiconnect-juhu-cf45bd47-jnpfm	1/1	Running	
mgmt03	default	apiconnect-juhu-cf45bd47-mtld7	1/1	Running	
mgmt01	default	apiconnect-juhu-cf45bd47-p2mb8	1/1	Running	
mgmt01	default	apiconnect-ldap-59bc575849-2qjqs	1/1	Running	
mgmt02	default	apiconnect-ldap-59bc575849-4bzc2	1/1	Running	
mgmt03	default	apiconnect-ldap-59bc575849-79k2r	1/1	Running	
mgmt02	default	apiconnect-lur-v2-886b685f7-6jpn6	1/1	Running	
mgmt01	default	apiconnect-lur-v2-886b685f7-9fq58	1/1	Running	
mgmt03	default	apiconnect-lur-v2-886b685f7-wn2vn	1/1	Running	
mgmt02	default	apiconnect-ui-7755d8fd8b-7qv4g	1/1	Running	
mgmt01	default	apiconnect-ui-7755d8fd8b-b7dt6	1/1	Running	
mgmt02	default	cassandra-operator-cassandra-operator-6f5d868f54-m84m9	1/1	Running	

...

```
...

mgmt01    kube-system    calico-node-8c45c                                  2/2    Running
mgmt02    kube-system    calico-node-jxs4d                                  2/2    Running
mgmt03    kube-system    calico-node-lh972                                  2/2    Running
mgmt01    kube-system    coredns-688c84959f-5mqll                           1/1    Running
mgmt01    kube-system    coredns-688c84959f-snr96                           1/1    Running
mgmt01    kube-system    etcd-mgmt01                                        1/1    Running
mgmt02    kube-system    etcd-mgmt02                                        1/1    Running
mgmt03    kube-system    etcd-mgmt03                                        1/1    Running
mgmt01    kube-system    ingress-nginx-ingress-controller-69lkl             1/1    Running
mgmt02    kube-system    ingress-nginx-ingress-controller-mghxt             1/1    Running
mgmt03    kube-system    ingress-nginx-ingress-controller-vthrv             1/1    Running
mgmt01    kube-system    ingress-nginx-ingress-default-backend-78fc87c466-gfsg9   1/1    Running
mgmt01    kube-system    kube-apiserver-mgmt01                              1/1    Running
mgmt02    kube-system    kube-apiserver-mgmt02                              1/1    Running
mgmt03    kube-system    kube-apiserver-mgmt03                              1/1    Running
mgmt01    kube-system    kube-apiserver-proxy-mgmt01                        1/1    Running
mgmt02    kube-system    kube-apiserver-proxy-mgmt02                        1/1    Running
mgmt03    kube-system    kube-apiserver-proxy-mgmt03                        1/1    Running
mgmt01    kube-system    kube-controller-manager-mgmt01                     1/1    Running
mgmt02    kube-system    kube-controller-manager-mgmt02                     1/1    Running
mgmt03    kube-system    kube-controller-manager-mgmt03                     1/1    Running
mgmt01    kube-system    kube-proxy-4h5nr                                   1/1    Running
mgmt03    kube-system    kube-proxy-g8dh5                                   1/1    Running
mgmt02    kube-system    kube-proxy-qbmqd                                   1/1    Running
mgmt01    kube-system    kube-scheduler-mgmt01                              1/1    Running
mgmt02    kube-system    kube-scheduler-mgmt02                              1/1    Running
mgmt03    kube-system    kube-scheduler-mgmt03                              1/1    Running
mgmt01    kube-system    metrics-server-dd8468b44-dkgxx                     1/1    Running
mgmt01    kube-system    tiller-deploy-7c8cdfc855-lt75b                     1/1    Running

apicadm@mgmt01: ~$
```

ヒント

各サーバーの状況から以下を確認します。

- **Install stage: DONE** および **Upgrade stage: UPGRADE_DONE** になったこと
- **Etcd cluster state:**にクラスターが構成されていること

- ポッドの Status が **Running** となり、コンテナー数の分母と分子が一致していること

または、

- ポッドの Status が **Succeeded** または **Completed** となっていること

各ポッドの詳細は、IBM Developer に掲載されている「API Connect V2018 Whitepaper」[14]をご参照ください。
(https://developer.ibm.com/apiconnect/2019/02/08/api-connect-v2018-deployment-whitepaper-now-available/)

3. ローカル・タイム・ゾーンを設定します。

```
$ sudo timedatectl set-timezone Asia/Tokyo
```

4. NTP を設定します。

```
$ sudo sed -i 's/#NTP=/NTP=<NTP_Server_address>/g' /etc/systemd/timesyncd.conf
$
$ sudo systemctl restart systemd-timesyncd.service
```

2. VMware 環境での分析サブシステムのデプロイ[15]

必要情報	値
サーバー1 の IP アドレス（eth0） サーバー2 の IP アドレス（eth0） サーバー3 の IP アドレス（eth0）	9.68.85.90/24 9.68.85.91/24 9.68.85.92/24
サーバー1 のホスト名 サーバー2 のホスト名 サーバー3 のホスト名	analyt01.apic.com analyt02.apic.com analyt03.apic.com
サーバー・ドメイン名	apic.com
DNS サーバー	9.68.85.106
デフォルト・ゲートウェイ	9.68.85.1
イーサネット・インターフェース名	eth0
Analytics client エンドポイント	analytics-client.apic.com
Analytics ingestion エンドポイント	analytics-ingestion.apic.com

前提で記載したとおり、分析サブシステムは eth1 は構成はしていません。

2.1. ISO ファイルの生成

1. APICUP 実行環境で、プロジェクト・ディレクトリーに移動します。

```
$ cd apic415
```

2. 分析サブシステムを作成します。

```
$ apicup create subsys analyt analytics
```

- analyt は、作成する分析サブシステム ID です。スペースを含まない小文字の英数字である必要がありあす。
- analytics は、分析サブシステムを作成することを表します。

apiconnect-up.yml ファイルが更新されます。

3. apiconnect-up.yml ファイルを構成します。

以下のコマンドで、 分析サブシステムの現行値を確認することができます。

```
$ apicup subsys get analyt
```

まだサブシステムを構成していない場合は、このコマンドによってエラーが返されます。また、値を更新していない場合は、使用できるデフォルト値がある場合、デフォルト値がリストされます。以下の例のようになります。

```
$ apicup subsys get analyt
Appliance settings
==================

Name                    Value       Description
----                    -----       ------
additional-cloud-init-file          (Optional) Path to additional cloud-init yml file
data-device             sdb         VM disk device (usually `sdb` for SCSI or `vdb` for VirtIO)
default-password                    (Optional) Console login password for `apicadm` user
dns-servers             []          List of DNS servers
extra-values-file                   (Optional) Path to additional configuration yml file
k8s-pod-network         172.16.0.0/16   (Optional) CIDR for pods within the appliance
k8s-service-network     172.17.0.0/16   (Optional) CIDR for services within the appliance
mode                    dev
public-iface            eth0        Device for API/UI traffic (Eg: eth0)
search-domain           []          List for DNS search domains
ssh-keyfiles            []          List of SSH public keys files
traffic-iface           eth0        Device for cluster traffic (Eg: eth0)


Subsystem settings
==================

Name                    Value       Description
----                    -----       ------
enable-message-queue    false       (Optional) Enable Analytics Message Queue Service
es-max-memory-gb        16          Memory limit for elastic search


Endpoints
=========

Name                    Value       Description
----                    -----       ------
analytics-client                    FQDN of Analytics client/UI endpoint
analytics-ingestion                 FQDN of Analytics ingestion endpoint


Error: Subsystem validation failure. Run with --validate to see details
```

a. 分析サーバーにログインするためのハッシュ・パスワードを作成します。mkpasswd を使用します。

```
$ mkpasswd --method=sha-512 --rounds=4096 Passw0rd!
```

ハッシュ値が出力されるのでコピーします。

```
$6$rounds=4096$TQTD.Ixs$0xNl4Mg7ei95sn9IOPYOuTw.n6OpESnqAyX1MCY5UHwHjT7VEofZQdFqO.ZXCrUePgLQBY3Mz7jx8ZBxxmvjY/
```

b. ハッシュ・パスワードを設定します。

フォーマット: apicup subsys set *analyt* default-password='*hashed_password*'

```
$ apicup subsys set analyt default-password='$6$rounds=4096$TQTD.Ixs$0xNl4Mg7ei95sn9IOPYOuTw.n6OpESnqAyX1MCY5UHwHjT7VEofZQdFqO.ZXCrUePgLQBY3Mz7jx8ZBxxmvjY/'
```

c. DNS サーバーを設定します。

```
$ apicup subsys set analyt dns-servers=9.68.85.106
```

d. モードオプションを設定します。

```
$ apicup subsys set analyt mode=standard
```

e. サーチドメインを設定します。

```
$ apicup subsys set analyt search-domain=apic.com
```

f. SSH 公開鍵ファイルのパスを設定します。

```
$ apicup subsys set analyt ssh-keyfiles=/home/admin/.ssh/id_rsa.pub
```

g. analytics client エンドポイントを設定します。

```
$ apicup subsys set analyt analytics-client=analytics-client.apic.com
```

h. analytics ingestion エンドポイントを設定します。

```
$ apicup subsys set analyt analytics-ingestion=analytics-ingestion.apic.com
```

[オプション] syslog によるログ収集が必要な場合は、syslog の構成をおこなうことができます。

4. ホストを追加します。

フォーマット: apicup hosts create analyt *hostname.domainname hd_password*

hd_password は、Linux Unified Key Setup によって分析サービスのストレージを暗号化するために使用されるパスワードです。このパスワードは保管時にハッシュされます。

分析サーバー1

```
$ apicup hosts create analyt analyt01.apic.com Passw0rd!
```

分析サーバー2

```
$ apicup hosts create analyt analyt02.apic.com Passw0rd!
```

分析サーバー3

```
$ apicup hosts create analyt analyt03.apic.com Passw0rd!
```

5. インターフェースを作成します。

フォーマット: apicup iface create analyt *hostname.domainname physical_network_id host_ip_address/subnet_mask gateway_ip_address*

分析サーバー1

```
$ apicup iface create analyt analyt01.apic.com eth0 9.68.85.90/255.255.255.0 9.68.85.1
```

分析サーバー2

```
$ apicup iface create analyt analyt02.apic.com eth0 9.68.85.91/255.255.255.0 9.68.85.1
```

分析サーバー3

```
$ apicup iface create analyt analyt03.apic.com eth0 9.68.85.92/255.255.255.0 9.68.85.1
```

6. public-iface と traffic-iface を設定します。

```
$ apicup subsys set analyt public-iface=eth0
$ apicup subsys set analyt traffic-iface=eth0
```

7. 設定したホストを確認します。

```
$ apicup hosts list analyt
analyt01.apic.com
    Device          IP/Mask                 Gateway
    eth0            9.68.85.90/255.255.255.0    9.68.85.1
analyt02.apic.com
    Device          IP/Mask                 Gateway
    eth0            9.68.85.91/255.255.255.0    9.68.85.1
analyt03.apic.com
    Device          IP/Mask                 Gateway
    eth0            9.68.85.92/255.255.255.0    9.68.85.1
```

8. 設定した構成を検証します。すべての設定値にチェックマークが付いていることを確認します。

```
$ apicup subsys get analyt --validate
Appliance settings
==================

Name                            Value
----                            -----
additional-cloud-init-file                                                  ✓
data-device                     sdb                                         ✓
default-password
            $6$rounds=4096$MumC.E4Fu7$MRuHEU8wHfmo/kc7Xlb5.m7SpaQCqYOzeinfCo3
19YdYkVwfE1gQc531a0WLZq3CMEUVNxiDsJzipUM6HCeRX/                             ✓
dns-servers                     [9.68.85.106]                               ✓
extra-values-file                                                           ✓
k8s-pod-network                 172.16.0.0/16                               ✓
k8s-service-network             172.17.0.0/16                               ✓
mode                            standard                                    ✓
public-iface                    eth0                                        ✓
search-domain                   [apic.com]                                  ✓
ssh-keyfiles                    [/home/admin/.ssh/id_rsa.pub]               ✓
traffic-iface                   eth0                                        ✓


Subsystem settings
==================

Name                            Value
----                            -----
enable-message-queue            false                                       ✓
es-max-memory-gb                16                                          ✓


Endpoints
=========

Name                            Value
----                            -----
analytics-client                analytics-client.apic.com                   ✓
analytics-ingestion             analytics-ingestion.apic.com                ✓
```

9. ISO ファイルを作成します。--out パラメーターで指定した値のディレクトリーが作成されます。

フォーマット: apicup subsys install *analyt* --out *analyt*plan-out

```
$ apicup subsys install analyt --out analytplan-out
```

11. 作成された ISO ファイルを確認します。

各分析サーバーに対する 3 つの ISO ファイルが作成されたことを確認します。

```
$ cd analytplan-out/
$
$ ls -ltr
total 1208
-rw-------  1 admin admin     164   6 月   5 15:11 meta.yml
-rw-------  1 admin admin    1679   6 月   5 15:11 appliance-client.key
-rw-------  1 admin admin    1151   6 月   5 15:11 appliance-client.crt
-rw-------  1 admin admin    1070   6 月   5 15:11 appliance-client-ca.pem
drwxr-x---  2 admin admin    4096   6 月   5 15:11 analyt01.apic.com
-rw-rw-r--  1 admin admin  401408   6 月   5 15:11 analyt01.apic.com.iso
drwxr-x---  2 admin admin    4096   6 月   5 15:11 analyt02.apic.com
-rw-rw-r--  1 admin admin  401408   6 月   5 15:11 analyt02.apic.com.iso
drwxr-x---  2 admin admin    4096   6 月   5 15:11 analyt03.apic.com
-rw-rw-r--  1 admin admin  401408   6 月   5 15:11 analyt03.apic.com.iso
-rw-------  1 admin admin    3522   6 月   5 15:11 instructions.txt
```

重要

ホスト名、**エンドポイント名**および**インターフェース設定**は、APIC サブシステムのインストール後に値を変更して ISO を再作成し、再読み込みをしても設定が反映されません。もう一度、システムをインストールし直す必要がありますのでご注意ください。

2.2. OVF テンプレートのデプロイ

※ここからは、管理サブシステムと同様の手順です。

1. VMware の vSphere Web Client にログインします。

2. vSphere ナビゲーターを使用して、OVA ファイルをデプロイするディレクトリーに移動します。

3. ディレクトリーを右クリックし、**OVF テンプレートのデプロイ**を選択します。

4. OVF テンプレートのデプロイ・ウィザードを完了します。

   a. インストール準備でダウンロードした **analytics_lts_2018.4.1.5.ova** テンプレートを選択します。**NEXT** をクリックします。

   b. ファイルの名前とロケーションを入力します。**NEXT** をクリックします。

   c. テンプレートのリソースを選択します。**NEXT** をクリックします。

   d. テンプレートの詳細を確認します。**NEXT** をクリックします。

   e. 構成のサイズを選択します。**NEXT** をクリックします。

   f. ストレージ設定を選択します。**NEXT** をクリックします。

   g. ネットワークを選択します。**NEXT** をクリックします。

   h. 必要に応じてテンプレートをカスタマイズします。（データディスクのサイズ）

   i. 設定を確認して **FINISH** をクリックし、仮想マシンをデプロイします。

2.3. ISO のデータ・ストアへのアップロード

1. ナビゲーターで**ストレージ**タブを選択します。

2. データ・ストアに移動します。

3. **ファイルのアップロード**を選択します。

4. ポップアップから、作成した ISO ファイルを選択します。

5. ISO ファイルを、デプロイした各分析サーバーのデータ・ストアにアップロードします。

2.4. ISO ファイルを使用したテンプレートの構成

1. ナビゲーターで**仮想マシンおよびテンプレート**を選択します。

2. デプロイした仮想マシンを見つけて選択します。

3. 右クリックして**アクション**から**設定の編集...**を選択します。

4. **仮想ハードウェア**タブで、**CD/DVD ドライブ 1** を展開します。

5. **接続...(電源オン時に接続)**をチェックします。

6. **データストア ISO ファイル**を選択して、アップロードしたファイルを見つけて **OK** を選択します。

7. アイコン・バーで再生ボタンを選択して、仮想マシンを開始します。システムの可用性およびダウンロードの速度によっては、インストールが完了するまでに数分間かかることがあります。

分析サーバー2，分析サーバー3 についても 1.2.から同様の手順で構成します。

2.5. インストールの状況確認

1. SSH 公開鍵のパスを指定したクライアントから SSH ツールを使用して分析サーバーにログインします。

yes を選択して、接続を続行します。 ホスト名がホストのリストに自動的に追加されます。

```
$ ssh 9.68.85.90 -l apicadm
Enter passphrase for key '/home/admin2019/.ssh/id_rsa':
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-145-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

7 packages can be updated.
0 updates are security updates.


Last login: Thu Jun  6 01:14:10 2019

apicadm@analyt01: ~$
```

2. apic status コマンドを実行して、インストールが完了したこと、およびシステムが正常に稼働していることを確認します。

ヒント
システム可用性によりますが、分析サブシステムの初回の構成が完了するまでにおおよそ 50 分前後かかります。

```
$ sudo apic status
 [INFO] Log level: info

Cluster members:
- analyt01.apic.com (9.68.85.90)
   Type: BOOTSTRAP_MASTER
   Install stage: DONE
   Upgrade stage: UPGRADE_DONE
   Subsystem detail:
   Docker status:
      Systemd unit: running
   Kubernetes status:
      Systemd unit: running
      Kubelet version: analyt01 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
   Etcd status: pod etcd-analyt01 in namespace kube-system has status Running
   Addons: calico, dns, helm, kube-proxy, metrics-server, nginx-ingress,
- analyt02.apic.com (9.68.85.91)
   Type: MASTER
   Install stage: DONE
   Upgrade stage: UPGRADE_DONE
   Subsystem detail: Done
   Docker status:
      Systemd unit: running
   Kubernetes status:
      Systemd unit: running
      Kubelet version: analyt02 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
   Etcd status: pod etcd-analyt02 in namespace kube-system has status Running
   Addons: calico, kube-proxy, nginx-ingress,
- analyt03.apic.com (9.68.85.92)
   Type: MASTER
   Install stage: DONE
   Upgrade stage: UPGRADE_DONE
   Subsystem detail:
   Docker status:
      Systemd unit: running
   Kubernetes status:
      Systemd unit: running
      Kubelet version: analyt03 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
   Etcd status: pod etcd-analyt03 in namespace kube-system has status Running
   Addons: calico, kube-proxy, nginx-ingress,
Etcd cluster state:
- etcd member name: analyt01.apic.com, member id: 14881586479866460943, cluster id: 1801279189310374730, leader id:
5263820929367456097, revision: 12087, version: 3.2.26
- etcd member name: analyt02.apic.com, member id: 5263820929367456097, cluster id: 1801279189310374730, leader id:
5263820929367456097, revision: 12087, version: 3.2.26
- etcd member name: analyt03.apic.com, member id: 7987654815146784218, cluster id: 1801279189310374730, leader id:
5263820929367456097, revision: 12087, version: 3.2.26

...
```

```
...

Pods Summary:
NODE        NAMESPACE       NAME                                                  READY     STATUS      REASON
analyt03    default         apic-analytics-analytics-client-69c48df9cc-6h4gt      1/1       Running
analyt01    default         apic-analytics-analytics-client-69c48df9cc-rbxfv      1/1       Running
analyt02    default         apic-analytics-analytics-cronjobs-rollover-1559790900-hh77x   0/1       Succeeded
analyt03    default         apic-analytics-analytics-ingestion-6695d6f5f7-4m7x2   1/1       Running
analyt02    default         apic-analytics-analytics-ingestion-6695d6f5f7-m2shb   1/1       Running
analyt03    default         apic-analytics-analytics-mtls-gw-dcf9576dc-jzd6h      1/1       Running
analyt02    default         apic-analytics-analytics-mtls-gw-dcf9576dc-kqrht      1/1       Running
analyt03    default         apic-analytics-analytics-operator-64bf5c944c-kv9dd    1/1       Running
analyt03    default         apic-analytics-analytics-storage-basic-gpwlb          1/1       Running
analyt01    default         apic-analytics-analytics-storage-basic-jqghp          1/1       Running
analyt02    default         apic-analytics-analytics-storage-basic-tczzm          1/1       Running
analyt02    kube-system     calico-node-82b2q                                     2/2       Running
analyt03    kube-system     calico-node-9j9j5                                     2/2       Running
analyt01    kube-system     calico-node-ph5pn                                     2/2       Running
analyt01    kube-system     coredns-688c84959f-lvqj2                              1/1       Running
analyt01    kube-system     coredns-688c84959f-m4zhq                              1/1       Running
analyt01    kube-system     etcd-analyt01                                         1/1       Running
analyt02    kube-system     etcd-analyt02                                         1/1       Running
analyt03    kube-system     etcd-analyt03                                         1/1       Running
analyt03    kube-system     ingress-nginx-ingress-controller-dwcw5                1/1       Running
analyt02    kube-system     ingress-nginx-ingress-controller-hf8wv                1/1       Running
analyt01    kube-system     ingress-nginx-ingress-controller-qfhj4                1/1       Running
analyt01    kube-system     ingress-nginx-ingress-default-backend-78fc87c466-sv7sk   1/1       Running
analyt01    kube-system     kube-apiserver-analyt01                               1/1       Running
analyt02    kube-system     kube-apiserver-analyt02                               1/1       Running
analyt03    kube-system     kube-apiserver-analyt03                               1/1       Running
analyt01    kube-system     kube-apiserver-proxy-analyt01                         1/1       Running
analyt02    kube-system     kube-apiserver-proxy-analyt02                         1/1       Running
analyt03    kube-system     kube-apiserver-proxy-analyt03                         1/1       Running
analyt01    kube-system     kube-controller-manager-analyt01                      1/1       Running
analyt02    kube-system     kube-controller-manager-analyt02                      1/1       Running
analyt03    kube-system     kube-controller-manager-analyt03                      1/1       Running
analyt01    kube-system     kube-proxy-fq6w4                                      1/1       Running
analyt03    kube-system     kube-proxy-hsw5v                                      1/1       Running
analyt02    kube-system     kube-proxy-llzff                                      1/1       Running
analyt01    kube-system     kube-scheduler-analyt01                               1/1       Running
analyt02    kube-system     kube-scheduler-analyt02                               1/1       Running
analyt03    kube-system     kube-scheduler-analyt03                               1/1       Running
analyt01    kube-system     metrics-server-dd8468b44-mlgxz                        1/1       Running
analyt01    kube-system     tiller-deploy-7c8cdfc855-nc4x2                        1/1       Running

apicadm@analyt01: ~$
```

ヒント

各サーバーの状況から以下を確認します。

- **Install stage: DONE** および **Upgrade stage: UPGRADE_DONE** になったこと
- **Etcd cluster state:**にクラスターが構成されていること

- ポッドの Status が **Running** となり、コンテナー数の分母と分子が一致していること

または、

- ポッドの Status が **Succeeded** または **Completed** となっていること

3. ローカル・タイム・ゾーンを設定します。

```
$ sudo timedatectl set-timezone Asia/Tokyo
```

4. NTP を設定します。

```
$ sudo sed -i 's/#NTP=/NTP=<NTP_Server_address>/g' /etc/systemd/timesyncd.conf
$
$ sudo systemctl restart systemd-timesyncd.service
```

3. VMware 環境でのポータルサブシステムのデプロイ[16]

必要情報	値
サーバー1 の IP アドレス（eth0） サーバー2 の IP アドレス（eth0） サーバー3 の IP アドレス（eth0）	9.68.85.93/24 9.68.85.94/24 9.68.85.95/24
サーバー1 のホスト名 サーバー2 のホスト名 サーバー3 のホスト名	ptl01.apic.com ptl02.apic.com ptl03.apic.com
サーバー・ドメイン名	apic.com
サーバー1 の IP アドレス（eth1） サーバー2 の IP アドレス（eth1） サーバー3 の IP アドレス（eth1）	9.68.83.14/24 9.68.83.15/24 9.68.83.16/24
DNS サーバー	9.68.85.106
デフォルト・ゲートウェイ	9.68.85.1
イーサネット・インターフェース名	eth0, eth1
Portal admin エンドポイント	portal-admin.apic.com
Portal web エンドポイント	portal-www.apic.com

Portal admin エンドポイントと、Portal web エンドポイントは別の DNS ネームで構成する必要があります。

3.1. ISO ファイルの生成

1. APICUP 実行環境で、プロジェクト・ディレクトリーに移動します。

```
$ cd apic415
```

2. ポータルサブシステムを作成します。

```
$ apicup create subsys ptl portal
```

- ptl は、作成するポータルサブシステム ID です。スペースを含まない小文字の英数字である必要がありあす。
- portal は、ポータルサブシステムを作成することを表します。

apiconnect-up.yml ファイルが更新されます。

3. apiconnect-up.yml ファイルを構成します。

以下のコマンドで、 ポータルサブシステムの現行値を確認することができます。

```
$ apicup subsys get ptl
```

まだサブシステムを構成していない場合は、このコマンドによってエラーが返されます。また、値を更新していない場合は、使用できるデフォルト値がある場合、デフォルト値がリストされます。以下の例のようになります。

```
$ apicup subsys get ptl
Appliance settings
==================

Name                    Value       Description
----                    -----       ------
additional-cloud-init-file          (Optional) Path to additional cloud-init yml file
data-device             sdb         VM disk device (usually `sdb` for SCSI or `vdb` for VirtIO)
default-password                    (Optional) Console login password for `apicadm` user
dns-servers             []          List of DNS servers
extra-values-file                   (Optional) Path to additional configuration yml file
k8s-pod-network         172.16.0.0/16   (Optional) CIDR for pods within the appliance
k8s-service-network     172.17.0.0/16   (Optional) CIDR for services within the appliance
mode                    dev
public-iface            eth0        Device for API/UI traffic (Eg: eth0)
search-domain           []          List for DNS search domains
ssh-keyfiles            []          List of SSH public keys files
traffic-iface           eth0        Device for cluster traffic (Eg: eth0)


Subsystem settings
==================

Name                    Value       Description
----                    -----       ------
site-backup-auth-pass               (optional) Server password for portal backups
site-backup-auth-user               (optional) Server username for portal backups
site-backup-host                    (optional) FQDN for portal backups server
site-backup-path        /site-backups   (optional) Path for portal backups
site-backup-port        22          (optional) port for portal backups server
site-backup-protocol    sftp        (Optional) Protocol for portal backups (sftp/ftp/objstore)
site-backup-schedule    0 2 * * *   (optional) Cron schedule for portal backups


Endpoints
=========

Name                    Value       Description
----                    -----       ------
portal-admin                        FQDN of Portal admin endpoint
portal-www                          FQDN of Portal web endpoint


Error: Subsystem validation failure. Run with --validate to see details
```

a. ポータルサーバーにログインするためのハッシュ・パスワードを作成します。mkpasswd を使用します。

```
$ mkpasswd --method=sha-512 --rounds=4096 Passw0rd!
```

ハッシュ値が出力されるのでコピーします。

```
$6$rounds=4096$hwY0BkUrV$4.OX7s.9G8gQ8pHS7Wfy9p8KmDuMBg2L7RenSPVDRIemBnCoUhmV8QH3KM42yyh83ySjFDJALLfE5G2P22lvg/
```

b. ハッシュ・パスワードを設定します。

フォーマット: apicup subsys set *ptl* default-password='*hashed_password*'

```
$ apicup subsys set mgmt default-password='$6$rounds=4096$hwY0BkUrV$4.OX7s.9G8gQ8pHS7Wfy9p8KmDuMBg2L7RenSPVDRIemBnCoUhmV8QH3KM42yyh83ySjFDJALLfE5G2P22lvg/'
```

c. DNS サーバーを設定します。

```
$ apicup subsys set ptl dns-servers=9.68.85.106
```

d. モードオプションを設定します。

```
$ apicup subsys set ptl mode=standard
```

e. サーチドメインを設定します。

```
$ apicup subsys set ptl search-domain=apic.com
```

f. SSH 公開鍵ファイルのパスを設定します。

```
$ apicup subsys set ptl ssh-keyfiles=/home/admin/.ssh/id_rsa.pub
```

[オプション] バックアップが必要な場合は、スケジュール済みバックアップを構成します。[17]

```
$ apicup subsys set ptl site-backup-host=apic520.apic.com
```

```
$ apicup subsys set ptl site-backup-port=22
```

```
$ apicup subsys set ptl site-backup-auth-user=sftp-user
```

```
$ apicup subsys set ptl site-backup-auth-pass=sftp-user
```

```
$ apicup subsys set ptl site-backup-path=/home/sftp-user/sftp/site-backups
```

```
$ apicup subsys set ptl site-backup-protocol=sftp
```

```
$ apicup subsys set ptl site-backup-schedule="0 2 * * *"
```

ヒント

- ベスト・プラクティスとして、サービス間の同期を確保するために、管理サブシステムとポータルサブシステムの両方を同時にバックアップしてください。
- スケジュールは、ローカル・タイム・ゾーンを例えば JST に変更しても **UTC** 時間で稼働しますのでご注意ください。

[オプション] syslog によるログ収集が必要な場合は、syslog の構成をおこなうことができます。

g. portal admin エンドポイントを設定します。

```
$ apicup subsys set ptl portal-admin=portal-admin.apic.com
```

h. portal web エンドポイント（ポータル URL）を設定します。

```
$ apicup subsys set ptl portal-www=portal-www.apic.com
```

4. ホストを追加します。

フォーマット: apicup hosts create ptl *hostname.domainname hd_password*

hd_password は、Linux Unified Key Setup によってポータルサービスのストレージを暗号化するために使用されるパスワードです。このパスワードは保管時にハッシュされます。

ポータルサーバー1

```
$ apicup hosts create ptl ptl01.apic.com Passw0rd!
```

ポータルサーバー2

```
$ apicup hosts create ptl ptl02.apic.com Passw0rd!
```

ポータルサーバー3

```
$ apicup hosts create ptl ptl03.apic.com Passw0rd!
```

5. インターフェースを作成します。

フォーマット: apicup iface create *ptl hostname.domainname physical_network_id host_ip_address/subnet_mask gateway_ip_address*

フォーマットの性質上、追加したインターフェースにもデフォルト・ゲートウェイを設定することになるため、後の手順で静的経路を設定します。

ポータルサーバー1

```
$ apicup iface create ptl ptl01.apic.com eth0 9.68.85.93/255.255.255.0 9.68.85.1

$ apicup iface create ptl ptl01.apic.com eth1 9.68.83.14/255.255.255.0 9.68.83.1
```

ポータルサーバー2

```
$ apicup iface create ptl ptl02.apic.com eth0 9.68.85.94/255.255.255.0 9.68.85.1

$ apicup iface create ptl ptl02.apic.com eth1 9.68.83.15/255.255.255.0 9.68.83.1
```

ポータルサーバー3

```
$ apicup iface create ptl ptl03.apic.com eth0 9.68.85.95/255.255.255.0 9.68.85.1

$ apicup iface create ptl ptl03.apic.com eth1 9.68.83.16/255.255.255.0 9.68.83.1
```

6. public-iface と traffic-iface を設定します。

```
$ apicup subsys set ptl public-iface=eth1
$ apicup subsys set ptl traffic-iface=eth0
```

7. 設定したホストを確認します。

```
$ apicup hosts list ptl
ptl01.apic.com
    Device  IP/Mask                 Gateway
    eth0    9.68.85.93/255.255.255.0    9.68.85.1
    eth1    9.68.83.14/255.255.255.0    9.68.83.1
ptl02.apic.com
    Device  IP/Mask                 Gateway
    eth0    9.68.85.94/255.255.255.0    9.68.85.1
    eth1    9.68.83.15/255.255.255.0    9.68.83.1
ptl03.apic.com
    Device  IP/Mask                 Gateway
    eth0    9.68.85.95/255.255.255.0    9.68.85.1
    eth1    9.68.83.16/255.255.255.0    9.68.83.1
```

8. 追加したインターフェースに静的経路を設定します。

```
$ touch ptl-cloud-init.yml
$
$ vi ptl-cloud-init.yml
$
$ cat ptl-cloud-init.yml
bootcmd:
 - ip route add 0.0.0.0/0 via 9.68.83.1 dev eth1
$
$ apicup subsys set ptl additional-cloud-init-file ptl-cloud-init.yml
```

9. 設定した構成を検証します。すべての設定値にチェックマークが付いていることを確認します。

```
$ apicup subsys get ptl --validate
Appliance settings
==================

Name                          Value
----                          -----
additional-cloud-init-file    ptl-cloud-init.yml                                  ✓
data-device                   sdb                                                 ✓
default-password
              $6$rounds=4096$hwY0BkUrV$4.OX7s.9G8gQ8pHS7Wfy9p8KmDuMBg2L7RenSPV
DRIemBnCoUhmV8QH3KM42yyh83ySjFDJALLfE5G2P22lvg/                                   ✓
dns-servers                    [9.68.85.106]                                      ✓
extra-values-file                                                                 ✓
k8s-pod-network               172.16.0.0/16                                       ✓
k8s-service-network           172.17.0.0/16                                       ✓
mode                          standard                                            ✓
public-iface                  eth1                                                ✓
search-domain                 [apic.com]                                          ✓
ssh-keyfiles                  [/home/admin/.ssh/id_rsa.pub]                       ✓
traffic-iface                 eth0                                                ✓

...
```

```
...

Subsystem settings
==================

Name                        Value                                 
----                        -----                                 
site-backup-auth-pass       c2Z0cC11c2Vy                       ✓  
site-backup-auth-user       sftp-user                          ✓  
site-backup-host            apic520.apic.com                   ✓  
site-backup-path            /home/sftp-user/sftp/site-backups  ✓  
site-backup-port            22                                 ✓  
site-backup-protocol        sftp                               ✓  
site-backup-schedule        0 2 * * *                          ✓  


Endpoints
=========

Name                        Value                                 
----                        -----                                 
portal-admin                portal-admin.apic.com              ✓  
portal-www                  portal-www.apic.com                ✓  
```

10. ISO ファイルを作成します。--out パラメーターで指定した値のディレクトリーが作成されます。

フォーマット: apicup subsys install *ptl* --out *ptl*plan-out

```
$ apicup subsys install ptl --out ptlplan-out
```

11. 作成された ISO ファイルを確認します。

各ポータルサーバーに対する 3 つの ISO ファイルが作成されたことを確認します。

```
$ cd ptlplan-out/
$
$ ls -ltr
total 1220
-rw-------  1 admin admin     161   6 月  22 16:21 meta.yml
-rw-------  1 admin admin    1675   6 月  22 16:21 appliance-client.key
-rw-------  1 admin admin    1147   6 月  22 16:21 appliance-client.crt
-rw-------  1 admin admin    1074   6 月  22 16:21 appliance-client-ca.pem
drwxr-x---  2 admin admin    4096   6 月  22 16:21 ptl01.apic.com
-rw-rw-r--  1 admin admin  403456   6 月  22 16:21 ptl01.apic.com.iso
drwxr-x---  2 admin admin    4096   6 月  22 16:21 ptl02.apic.com
-rw-rw-r--  1 admin admin  403456   6 月  22 16:21 ptl02.apic.com.iso
drwxr-x---  2 admin admin    4096   6 月  22 16:21 ptl03.apic.com
-rw-rw-r--  1 admin admin  403456   6 月  22 16:21 ptl03.apic.com.iso
-rw-------  1 admin admin    3774   6 月  22 16:21 instructions.txt
```

重要

ホスト名、**エンドポイント名**および**インターフェース設定**は、APIC サブシステムのインストール後に値を変更して ISO を再作成し、再読み込みをしても設定が反映されません。もう一度、システムをインストールし直す必要がありますのでご注意ください。

3.2. OVF テンプレートのデプロイ

※ここからは、管理サブシステムと同様の手順です。

1. VMware の vSphere Web Client にログインします。

2. vSphere ナビゲーターを使用して、OVA ファイルをデプロイするディレクトリーに移動します。

3. ディレクトリーを右クリックし、**OVF テンプレートのデプロイ**を選択します。

4. OVF テンプレートのデプロイ・ウィザードを完了します。

   a. インストール準備でダウンロードした **portal_lts_2018.4.1.5.ova** テンプレートを選択します。**NEXT** をクリックします。

   b. ファイルの名前とロケーションを入力します。**NEXT** をクリックします。

   c. テンプレートのリソースを選択します。**NEXT** をクリックします。

   d. テンプレートの詳細を確認します。**NEXT** をクリックします。

   e. 構成のサイズを選択します。**NEXT** をクリックします。

   f. ストレージ設定を選択します。**NEXT** をクリックします。

   g. ネットワークを選択します。**NEXT** をクリックします。

   h. 必要に応じてテンプレートをカスタマイズします。（データディスクのサイズ）

   i. 設定を確認して **FINISH** をクリックし、仮想マシンをデプロイします。

3.3. ISO のデータ・ストアへのアップロード

1. ナビゲーターで**ストレージ**タブを選択します。

2. データ・ストアに移動します。

3. **ファイルのアップロード**を選択します。

4. ポップアップから、作成した ISO ファイルを選択します。

5. ISO ファイルを、デプロイした各分析サーバーのデータ・ストアにアップロードします。

3.4. ISO ファイルを使用したテンプレートの構成

1. ナビゲーターで**仮想マシンおよびテンプレート**を選択します。

2. デプロイした仮想マシンを見つけて選択します。

3. 右クリックして**アクション**から**設定の編集...**を選択します。

4. **仮想ハードウェア**タブで、**CD/DVD ドライブ 1** を展開します。

5. **接続...(電源オン時に接続)**をチェックします。

6. **データストア ISO ファイル**を選択して、アップロードしたファイルを見つけて **OK** を選択します。

7. アイコン・バーで再生ボタンを選択して、仮想マシンを開始します。システムの可用性およびダウンロードの速度によっては、インストールが完了するまでに数分間かかることがあります。

ポータルサーバー2, ポータルサーバー3 についても 1.2.から同様の手順で構成します。

3.5. インストールの状況確認

1. SSH 公開鍵のパスを指定したクライアントから SSH ツールを使用してポータルサーバーにログインします。**yes** を選択して、接続を続行します。 ホスト名がホストのリストに自動的に追加されます。

```
$ ssh 9.68.85.93 -l apicadm
Enter passphrase for key '/home/admin/.ssh/id_rsa':
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-145-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud

7 packages can be updated.
0 updates are security updates.


Last login: Wed Jun  5 08:05:43 2019 from 9.68.85.106

apicadm@ptl01: ~$
```

2. apic status コマンドを実行して、インストールが完了したこと、およびシステムが正常に稼働していることを確認します。

情報
システム可用性によりますが、分析サブシステムの初回の構成が完了するまでにおおよそ 30 分前後かかります。

```
$ sudo apic status
[INFO] Log level: info

Cluster members:
- ptl01.apic.com (9.68.85.93)
   Type: BOOTSTRAP_MASTER
   Install stage: DONE
   Upgrade stage: UPGRADE_DONE
   Subsystem detail: Done
   Docker status:
      Systemd unit: running
   Kubernetes status:
      Systemd unit: running
      Kubelet version: ptl01 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
   Etcd status: pod etcd-ptl01 in namespace kube-system has status Running
   Addons: calico, dns, helm, kube-proxy, metrics-server, nginx-ingress,
- ptl02.apic.com (9.68.85.94)
   Type: MASTER
   Install stage: DONE
   Upgrade stage: UPGRADE_DONE
   Subsystem detail:
   Docker status:
      Systemd unit: running
   Kubernetes status:
      Systemd unit: running
      Kubelet version: ptl02 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
   Etcd status: pod etcd-ptl02 in namespace kube-system has status Running
   Addons: calico, kube-proxy, nginx-ingress,
- ptl03.apic.com (9.68.85.95)
   Type: MASTER
   Install stage: DONE
   Upgrade stage: UPGRADE_DONE
   Subsystem detail: Done
   Docker status:
      Systemd unit: running
   Kubernetes status:
      Systemd unit: running
      Kubelet version: ptl03 (4.4.0-145-generic) [Kubelet v1.13.5, Proxy v1.13.5]
   Etcd status: pod etcd-ptl03 in namespace kube-system has status Running
   Addons: calico, kube-proxy, nginx-ingress,
Etcd cluster state:
- etcd member name: ptl01.apic.com, member id: 16784920490433818745, cluster id: 12802501948044056823, leader id:
16784920490433818745, revision: 10897, version: 3.2.26
- etcd member name: ptl02.apic.com, member id: 14868614643968380330, cluster id: 12802501948044056823, leader id:
16784920490433818745, revision: 10897, version: 3.2.26
- etcd member name: ptl03.apic.com, member id: 3232128550836151414, cluster id: 12802501948044056823, leader id:
16784920490433818745, revision: 10897, version: 3.2.26

...
```

```
...

Pods Summary:
NODE         NAMESPACE        NAME                                                READY     STATUS    REASON
ptl03        default          apic-portal-apic-portal-db-9492x                    2/2       Running
ptl02        default          apic-portal-apic-portal-db-qxm7v                    2/2       Running
ptl01        default          apic-portal-apic-portal-db-wxrn9                    2/2       Running
ptl03        default          apic-portal-apic-portal-nginx-7d5f8dd996-56lrb      1/1       Running
ptl02        default          apic-portal-apic-portal-nginx-7d5f8dd996-b7tsx      1/1       Running
ptl01        default          apic-portal-apic-portal-nginx-7d5f8dd996-rsckh      1/1       Running
ptl01        default          apic-portal-apic-portal-www-4q7fx                   2/2       Running
ptl02        default          apic-portal-apic-portal-www-7f7dd                   2/2       Running
ptl03        default          apic-portal-apic-portal-www-c2vr6                   2/2       Running
ptl03        kube-system      calico-node-cs8vc                                   2/2       Running
ptl02        kube-system      calico-node-knhc4                                   2/2       Running
ptl01        kube-system      calico-node-w5tjx                                   2/2       Running
ptl01        kube-system      coredns-688c84959f-2fx6n                            1/1       Running
ptl01        kube-system      coredns-688c84959f-swmgn                            1/1       Running
ptl01        kube-system      etcd-ptl01                                          1/1       Running
ptl02        kube-system      etcd-ptl02                                          1/1       Running
ptl03        kube-system      etcd-ptl03                                          1/1       Running
ptl01        kube-system      ingress-nginx-ingress-controller-6zkt9              1/1       Running
ptl02        kube-system      ingress-nginx-ingress-controller-fsjrc              1/1       Running
ptl03        kube-system      ingress-nginx-ingress-controller-wdpbc              1/1       Running
ptl01        kube-system      ingress-nginx-ingress-default-backend-78fc87c466-999wg   1/1       Running
ptl01        kube-system      kube-apiserver-proxy-ptl01                          1/1       Running
ptl02        kube-system      kube-apiserver-proxy-ptl02                          1/1       Running
ptl03        kube-system      kube-apiserver-proxy-ptl03                          1/1       Running
ptl01        kube-system      kube-apiserver-ptl01                                1/1       Running
ptl02        kube-system      kube-apiserver-ptl02                                1/1       Running
ptl03        kube-system      kube-apiserver-ptl03                                1/1       Running
ptl01        kube-system      kube-controller-manager-ptl01                       1/1       Running
ptl02        kube-system      kube-controller-manager-ptl02                       1/1       Running
ptl03        kube-system      kube-controller-manager-ptl03                       1/1       Running
ptl03        kube-system      kube-proxy-ph2d6                                    1/1       Running
ptl02        kube-system      kube-proxy-pwqfs                                    1/1       Running
ptl01        kube-system      kube-proxy-xxk2d                                    1/1       Running
ptl01        kube-system      kube-scheduler-ptl01                                1/1       Running
ptl02        kube-system      kube-scheduler-ptl02                                1/1       Running
ptl03        kube-system      kube-scheduler-ptl03                                1/1       Running
ptl01        kube-system      metrics-server-dd8468b44-ltnfs                      1/1       Running
ptl01        kube-system      tiller-deploy-7c8cdfc855-v2d67                      1/1       Running

apicadm@ptl01: ~$
```

ヒント

各サーバーの状況から以下を確認します。

- **Install stage: DONE** および **Upgrade stage: UPGRADE_DONE** になったこと
- **Etcd cluster state:**にクラスターが構成されていること

- ポッドの Status が **Running** となり、コンテナー数の分母と分子が一致していること

または、

- ポッドの Status が **Succeeded** または **Completed** となっていること

3. ローカル・タイム・ゾーンを設定します。

```
$ sudo timedatectl set-timezone Asia/Tokyo
```

4. NTP を設定します。

```
$ sudo sed -i 's/#NTP=/NTP=<NTP_Server_address>/g' /etc/systemd/timesyncd.conf
$
$ sudo systemctl restart systemd-timesyncd.service
```

これで、API Connect の構成をするための APIC サブシステムの準備ができました。

次に、IBM DataPower Gateway を使用してゲートウェイサーバーのインストールと API ゲートウェイ・サービスのための構成をおこないます。

4. API Connect 用の DataPower ゲートウェイの構成[18]

前提

- 互換性のあるバージョンの IBM DataPower Gateway および API Connect を使用していることを確認します。
- API Connect および DataPower は、2 つのタイプのゲートウェイ構成をサポートしています。**DataPower Gateway (v5 互換)**は、API Connect バージョン 5.x で利用できたゲートウェイ・サポートと同じサポートを提供します。**DataPower API Gateway** は、拡張されたパフォーマンス重視型のゲートウェイです。詳しくは、API Connect のゲートウェイ・タイプを参照してください。[19]
- 本ガイドでは、ゲートウェイ・タイプとして **DataPower API Gateway** を構成します。

> 重要
>
> DataPower Gateway (v5 互換)は、 バージョン 2018.4.1.4 から非推奨になったことが発表されました。[20]
>
> 今後、ゲートウェイ・タイプとして DataPower API Gateway を使用することをおすすめします。

必要情報	値
サーバー1 の IP アドレス（eth0） サーバー2 の IP アドレス（eth0） サーバー3 の IP アドレス（eth0）	9.68.85.96/24 9.68.85.97/24 9.68.85.98/24
サーバー1 のホスト名 サーバー2 のホスト名 サーバー3 のホスト名	gwy01.apic.com gwy02.apic.com gwy03.apic.com
サーバー・ドメイン名	apic.com
サーバー1 の IP アドレス（eth1） サーバー2 の IP アドレス（eth1） サーバー3 の IP アドレス（eth1）	9.68.85.99/24 9.68.85.100/24 9.68.85.101/24
DNS サーバー	9.68.85.106
デフォルト・ゲートウェイ	9.68.85.1
イーサネット・インターフェース名	eth0, eth1
ゲートウェイ・サービス（ディレクター）エンドポイント	apic-gw-service.apic.com
ゲートウェイ API エンドポイント	api-gateway.apic.com

4.1. OVF テンプレートのデプロイ[21]

1. VMware の vSphere Web Client にログインします。

2. vSphere ナビゲーターを使用して、OVA ファイルをデプロイするディレクトリーに移動します。

3. ディレクトリーを右クリックし、**OVF テンプレートのデプロイ**を選択します。

4. OVF テンプレートのデプロイ・ウィザードを完了します。

   a. インストール準備でダウンロードした **idg2018410.lts.nonprod.ova (環境にあわせてパッケージを読み替えてください)** テンプレートを選択します。**NEXT** をクリックします。

   b. ファイルの名前とロケーションを入力します。**NEXT** をクリックします。

   c. テンプレートのリソースを選択します。**NEXT** をクリックします。

   d. テンプレートの詳細を確認します。**NEXT** をクリックします。

   e. 使用許諾契約書の条項を読んで同意にチェックを入れます。**NEXT** をクリックします。

   f. 構成のサイズを選択します。**NEXT** をクリックします。

   g. ストレージ設定を選択します。**NEXT** をクリックします。

   h. ネットワークを選択します。**NEXT** をクリックします。

   i. 必要に応じてテンプレートをカスタマイズします。

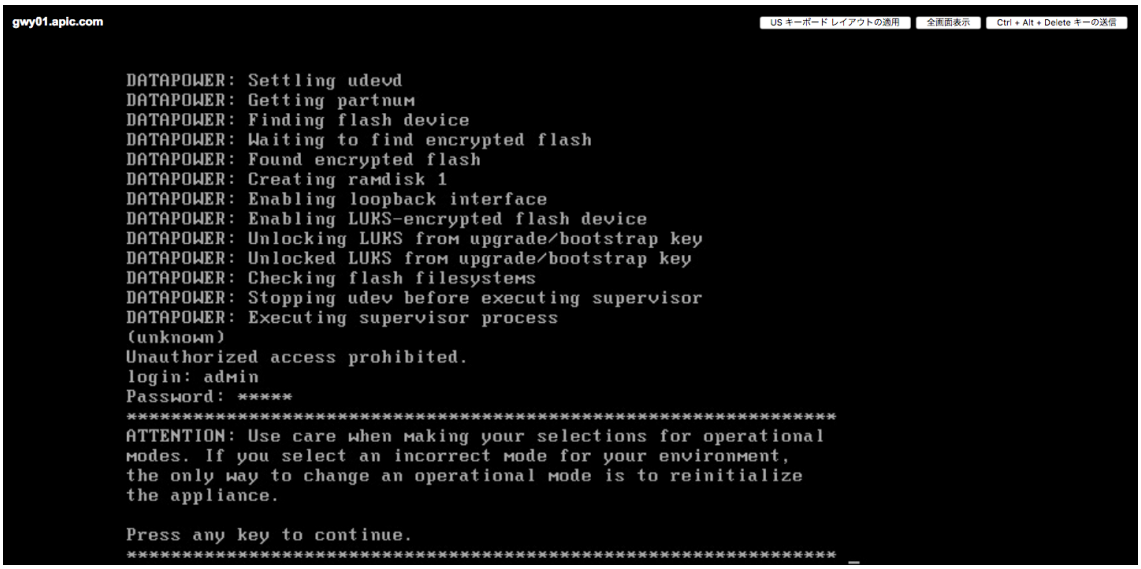
   j. 設定を確認して **FINISH** をクリックし、仮想マシンをデプロイします。

4.2. DataPower Gateway の初期化[22]

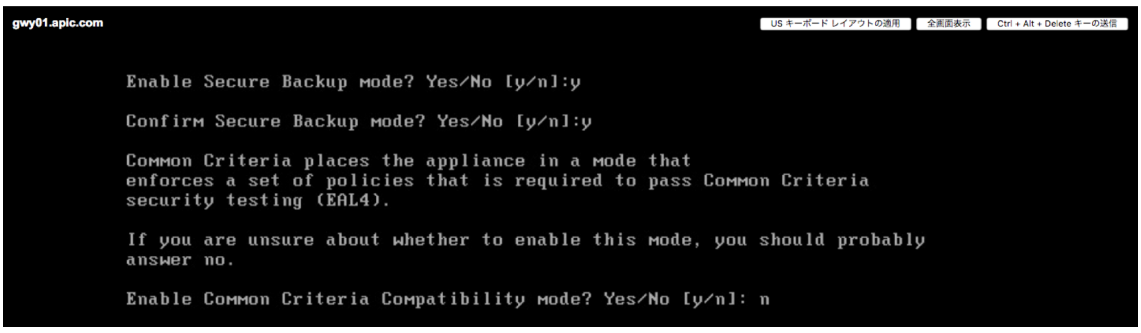
1. VMware の vSphere Web Client にログインします。

2. デプロイした DataPower を選択して、**Web コンソールの起動**をクリックします。

3. プロンプトで、**login: admin** (ローカル・ユーザー・アカウント名)、**Password: admin** (デフォルト・パスワード) と入力します。

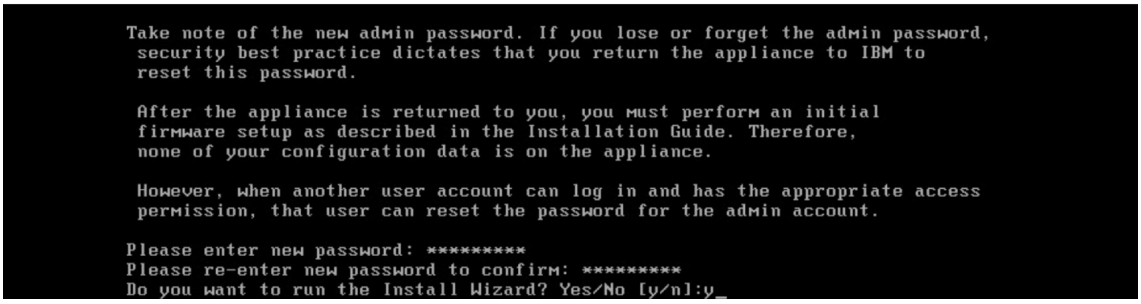


4. セキュア・バックアップを作成する場合は、セキュア・バックアップ・モード: **y** と入力します。
5. コモン・クライテリア互換モードは、通常は必要ありませんので **n** と入力します。



6. 新規パスワードを入力して確認します。

7. 基本ファームウェア構成はここまでです。Do you want to run the Installation Wizard? プロンプトが表示されるので、**y** と入力して、インストール・ウィザードを開始します。

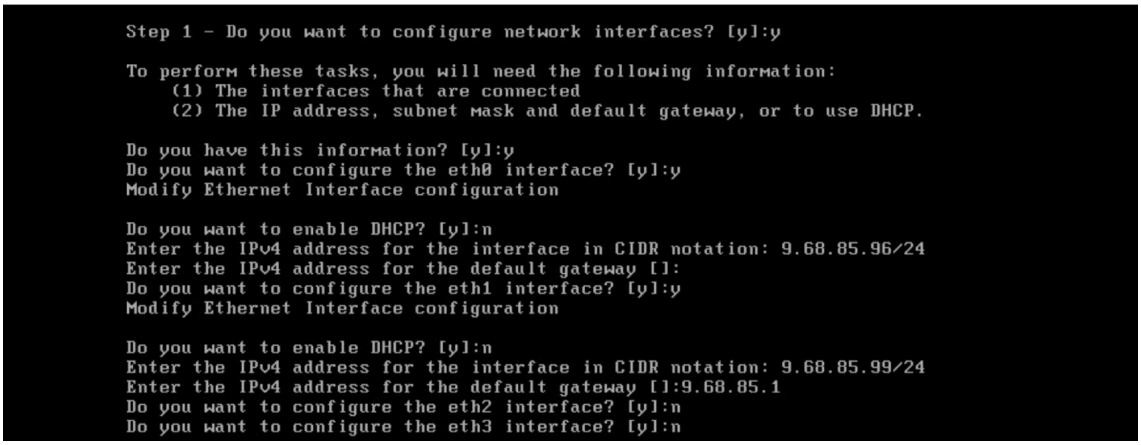


ヒント

不注意でプロンプトに n と入力した場合は、次のコマンドを入力することで、インストール・ウィザードを開始できます。

```
# configure terminal
# startup
```

8. Step 1 - **y** と入力して、ネットワーク・インターフェースを構成します。



9. Step 2 - **y** と入力して、DNS サービスを構成します。

```
Step 2 -  Do you want to configure network services? [y]:y
Do you want to configure DNS? [y]:y

This configuration requires the IP address of the DNS server.

Do you have this information? [y]:y
Enter the IP address of the DNS server: 9.68.85.106
Modify DNS Settings configuration
```

10. Step 3 - **y** と入力して、アプライアンス名を入力します。

```
Step 3 -  Do you want to define a unique system identifier for the appliance? [y
]:y
Enter a unique system identifier: gwy01
Modify System Settings configuration
```

11. Step 4 - **y** と入力して、SSH アクセスを構成します。

```
Step 4 - Do you want to configure remote management access? [y]:y

These configurations require the IP address of the local interface that manages
the appliance.

Do you have this information? [y]:y
Do you want to enable SSH? [y]:y
Enter the local IP address [0 for all]: 0
Enter the port number [22]: 22

%       Pending

SSH service listener enabled

Do you want to enable WebGUI access [y]:_
```

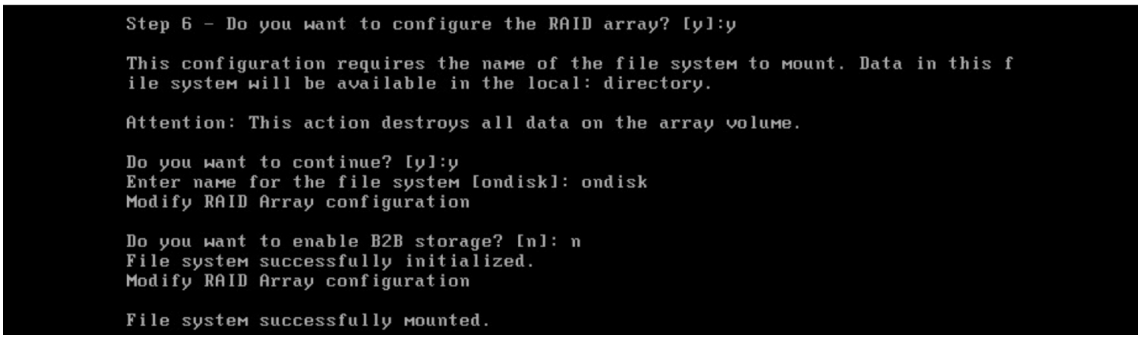
12. 続いて、**y** と入力して、Web 管理インターフェースを構成します。

```
Do you want to enable WebGUI access [y]:y
Enter the local IP address [0 for all]: 0
Enter the port number [9090]: 9090
Modify Web Management Service configuration
```

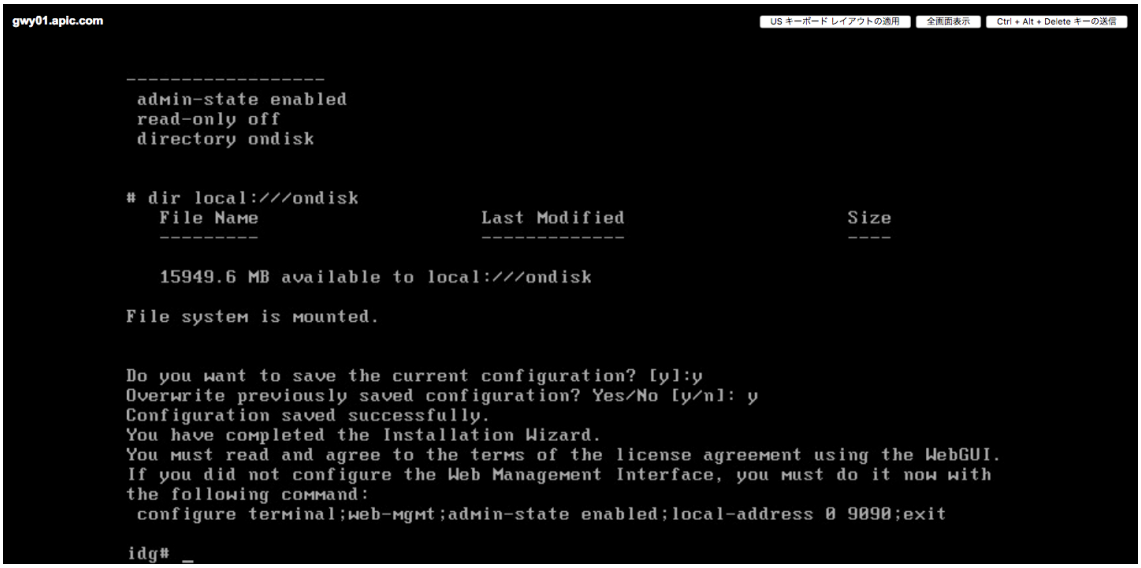
13. Step 5 - パスワード・リセット用のアカウントは、本ガイドでは構成しないので **n** と入力します。

```
Attention: If the password for the admin account is lost or forgotten, you will
have to delete the virtual machine. However, if another user account can log in
and if that account has the appropriate access permission, that user can reset t
he password for the admin account.

Note: If you specify an existing user account, you will change the password for
this account.

Step 5 - Do you want to configure a user account that can reset passwords? [y]:
n
Skipping the configuration of a user account that can reset passwords.

Note: Configuration of the RAID array is required to use the RAID array in this
appliance.
```

14. Step 6 - **y** と入力して、RAID アレイを構成します。B2B ストレージは本ガイドでは使用しませんので **n** と入力します。



15. **y** と入力して、構成した内容を保存します。先程構成した WebGUI を使用して、使用条件に同意するようプロンプト表示されます。



16. 使用条件に同意します。Web 管理インターフェースにアクセスするためのアドレスには HTTPS プロトコルが使用され、その形式は https://*IP_address*:*port* になります。

4.3. ローカル・タイム・ゾーンの設定

1. SSH で DataPower にログインします。

2. 以下のコマンドで、ローカル・タイム・ゾーンを設定します。

```
idg# top; config; timezone; name JST-9; exit
```

4.4. NTP サービスの設定

1. SSH で DataPower にログインします。

2. 以下のコマンドで、NTP サービスを設定します。

```
idg# top; config; ntp-service; remote-server <NTP_Server_address>; admin-state enabled;
exit;
```

3. NTP サーバーから時刻同期ができたことを確認します。

```
idg# show ntp-refresh

    Last server tried: 9.189.13.75
Last result received: Success!
  Time after refresh: Tue Jun 11 16:04:06 2019
          Local Time: Tue Jun 11 16:04:15 2019
```

4.5. DataPower ファームウェアのアップグレード[23]

1. DataPower の Web 管理インターフェースにアクセスして **default** ドメインにログインします。

2. 検索フィールドに、system と入力します。

3. 検索結果から、**System Control** をクリックします。

4. **Boot Image** セクションを探します。

5. **Upload** をクリックして、ダウンロードしたファームウェア・イメージ **idg2018415.lts.scrypt4** をアップロードします。

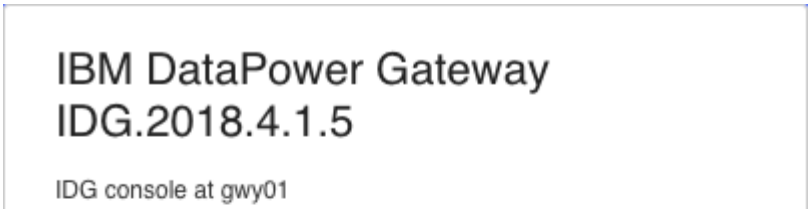
6. **Firmware File** リストから、アップロードしたファームウェア・イメージを選択します。

7. **I accept the terms of the license agreements** にチェックを入れます。

8. **Boot Image** をクリックします。

9. プロンプトに従います。アップグレード中は、ファームウェア・イメージをロードし、サーバーが再始動されます。本ガイドの構成ですとおおよそ 20 分前後でアップグレードが完了します。

10. DataPower の Web 管理インターフェースにアクセスして、ファームウェア・バージョンがアップグレードされたことを確認します。



ゲートウェイサーバー2，ゲートウェイサーバー3 についても、4.1.から同様の手順で構成します。

全台共通

4.6. XML 管理インターフェースの有効化

1. SSH で DataPower にログインします。

2. 以下のコマンドで XML 管理インターフェースを有効化します。

```
idg# top; config; xml-mgmt; admin-state enabled; exit; write mem;
```

全台共通

4.7. DataPower のアプリケーション・ドメインの作成

1. SSH で DataPower にログインします。

2. 以下のコマンドでアプリケーション・ドメイン ***apiconnect*** を作成します。名前は任意です。**default** ドメインを *apiconnect* ドメインから visible になるように構成し、作成したアプリケーション・ドメインにスイッチします。

```
idg# top; config; domain apiconnect; visible default; exit; write mem;
```

以降、設定はすべてアプリケーション・ドメインでおこないます。

[オプション] API Connect ゲートウェイ・サービス・プロセスの追加ロギング設定

1. 以下のコマンドで apic-gw-service に関するログを取得することができます。

```
idg[apiconnect](config)# logging target gwd-log

New Log Target configuration

idg[apiconnect](config logging target gwd-log)# type file
idg[apiconnect](config logging target gwd-log)# format text
idg[apiconnect](config logging target gwd-log)# timestamp syslog
idg[apiconnect](config logging target gwd-log)# size 50000
idg[apiconnect](config logging target gwd-log)# local-file logtemp:///gwd-log
idg[apiconnect](config logging target gwd-log)# event apic-gw-service debug
idg[apiconnect](config logging target gwd-log)# exit;
idg[apiconnect](config)# write mem;
```

注意: 本番環境ではデバッグ・レベルに設定しないようにしてください。

全台共通

4.8. 構成シーケンスの定義

1. API Connect ゲートウェイ・サービスでは、構成シーケンスを使用して DataPower を構成し、API Connect で定義されている API を実装します。
    - Location profile: local:///
    - Configuration execution interval: 3000

```
idg[apiconnect](config)# config-sequence "apiconnect"

New Configuration Sequence configuration

idg[apiconnect](config config-sequence apiconnect)# location "local:///"
idg[apiconnect](config config-sequence apiconnect)# watch "on"
idg[apiconnect](config config-sequence apiconnect)# run-sequence-interval 3000
idg[apiconnect](config config-sequence apiconnect)# delete-unused "on"
idg[apiconnect](config config-sequence apiconnect)# match "(.*)¥.cfg$"
idg[apiconnect](config config-sequence apiconnect)# summary "Toolkit Reboot configuration"
idg[apiconnect](config config-sequence apiconnect)# exit;
idg[apiconnect](config)# write mem;
```

全台共通

4.9. 自己署名証明書の作成

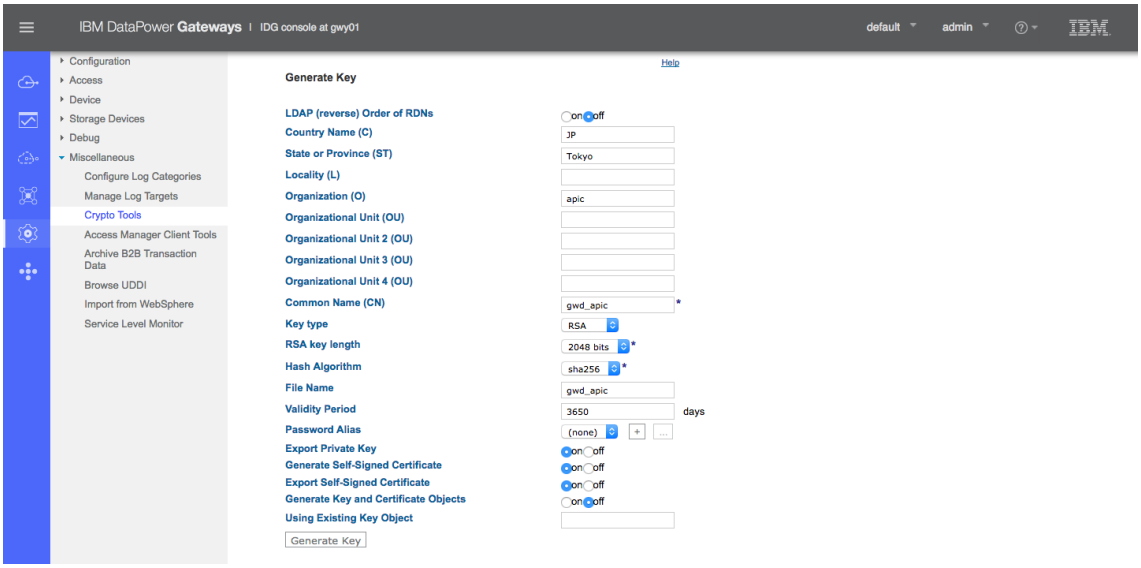
管理サーバーと API ゲートウェイ・サービス・プロセスの間のトラフィックを保護するために使用される自己署名証明書と秘密鍵を作成します。DataPower を使用するか、**OpenSSL** などの他のツールを使用して、証明書と秘密鍵を生成できます。本ガイドでは、DataPower の WebGUI を使用して作成します。

ゲートウェイサーバー2, ゲートウェイサーバー3 でも同様の証明書、鍵を使用する場合は、本手順はいずれかのサーバーで 1 度だけおこないます。

1. WebGUI から、DataPower の **default ドメイン**にログインします。

2. 検索フィールドに **Crypto Tools** と入力して選択します。

3. 以下の項目を入力して、**Generate Key** をクリックします。

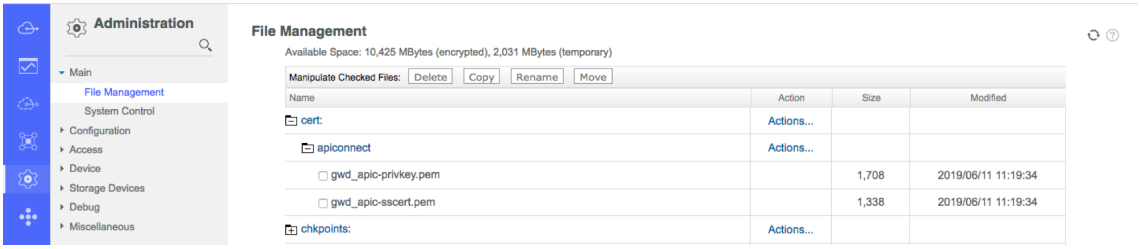


4. **Generate an RSA key pair and a CSR** の確認をクリックします。

5. 検索フィールドに **File Management** と入力して選択します。**temporary:///**フォルダに、生成した証明書と暗号鍵があるので右クリックしてダウンロードします。

- gwd_apic-privkey.pem – 秘密鍵ファイル
- gwd_apic-sscert.pem – 自己署名証明書ファイル

6. **cert:///apiconnect** フォルダの右にある **Actions...**ボタンをクリックし、ダウンロードした 2 ファイルをアップロードします。



全台共通

4.10. 暗号オブジェクトの定義

ここから、SSH の apiconnect ドメインに戻ります。

1. アップロードしたファイルから、暗号鍵と暗号証明書オブジェクトを作成します。

```
idg[apiconnect](config)# crypto;

Crypto configuration mode
idg[apiconnect](config-crypto)# key gwd_apic-privkey cert:///gwd_apic-privkey.pem

Creating key 'gwd_apic-privkey'

idg[apiconnect](config-crypto)# certificate gwd_apic-sscert cert:///gwd_apic-sscert.pem

Creating certificate 'gwd_apic-sscert'

idg[apiconnect](config-crypto)# exit;

Exiting Crypto Configuration mode
idg[apiconnect](config)# write mem;
```

2. ID 資格情報オブジェクトを設定し、暗号鍵と暗号証明書を関連付けます。

```
idg[apiconnect](config)# crypto;

Crypto configuration mode
idg[apiconnect](config-crypto)# idcred apic-gw-service-idcred gwd_apic-privkey gwd_apic-
sscert;

Creating Identification Credentials 'apic-gw-service-idcred'

idg[apiconnect](config-crypto)# exit;

Exiting Crypto Configuration mode
idg[apiconnect](config)# write mem;
```

3. 設定した ID 資格情報を使用して、**SSL クライアント・プロファイル**を作成します。

```
idg[apiconnect](config)# crypto;

Crypto configuration mode
idg[apiconnect](config-crypto)# ssl-client gwd_client;

New SSL Client Profile configuration

idg[apiconnect](config ssl-client gwd_client)# reset

idg[apiconnect](config ssl-client gwd_client)# protocols TLSv1d2

idg[apiconnect](config ssl-client gwd_client)# idcred apic-gw-service-idcred

idg[apiconnect](config ssl-client gwd_client)# no validate-server-cert

idg[apiconnect](config ssl-client gwd_client)# exit;

idg[apiconnect](config-crypto)# exit;

Exiting Crypto Configuration mode
idg[apiconnect](config)# write mem;
```

4. 同じく設定した ID 資格情報を使用して、**SSL サーバー・プロファイル**を作成します。

```
idg[apiconnect](config)# crypto;

Crypto configuration mode
idg[apiconnect](config-crypto)# ssl-server gwd_server;

New SSL Server Profile configuration

idg[apiconnect](config ssl-server gwd_server)# reset

idg[apiconnect](config ssl-server gwd_server)# protocols TLSv1d2

idg[apiconnect](config ssl-server gwd_server)# idcred apic-gw-service-idcred

idg[apiconnect](config ssl-server gwd_server)# validate-client-cert off

idg[apiconnect](config ssl-server gwd_server)# exit;

idg[apiconnect](config-crypto)# exit;
```

全台共通

4.11. ゲートウェイ・サービス・クラスター構成のためのゲートウェイ・ピアリング構成

1. ゲートウェイ・サービスをクラスターとして構成するために、ゲートウェイ・ピアリングを作成します。

- **パーシスタンス・ロケーション**は、**memory** に設定します。
- プライマリーにするサーバーと、それ以外のサーバーは、異なるプライオリティを設定します。設定する数値が低いほど、プライオリティが高くなります。
- **プライマリー**は、**peer は構成しない**でください。ただし、**enable-peer-group** は **on** にします。
- **peer** は、クラスター内通信なので APIC サブシステムに合わせて **eth0 インターフェース**で定義します。

情報
パーシスタンス・ロケーションの値は、物理 DataPower アプライアンスの場合は **RAID** に設定して、仮想 DataPower アプライアンスの場合は**メモリー**に設定します。**RAID** と**ローカル**の両方の設定は再始動後も保持されますが、**メモリー**設定は保持されません。**ローカル**はセキュアなオプションではないことに注意してください。

プライマリー・サーバー (ゲートウェイサーバー1)

```
config;
gateway-peering gwd_peering;
  admin-state enabled
  local-address 9.68.85.96
  local-port 16380
  monitor-port 26380
  priority 50
  enable-ssl off
  enable-peer-group on
  persistence memory
exit;
write mem;
```

セカンダリー・サーバー(ゲートウェイサーバー2)

```
config;
gateway-peering gwd_peering;
  admin-state enabled
  local-address 9.68.85.97
  local-port 16380
  monitor-port 26380
  priority 100
  enable-ssl off
  enable-peer-group on
  peer 9.68.85.96
  peer 9.68.85.98
  persistence memory
exit;
write mem;
```

セカンダリー・サーバー(ゲートウェイサーバー3)

```
config;
gateway-peering gwd_peering;
   admin-state enabled
   local-address 9.68.85.98
   local-port 16380
   monitor-port 26380
   priority 101
   enable-ssl off
   enable-peer-group on
   peer 9.68.85.96
   peer 9.68.85.97
   persistence memory
exit;
write mem;
```

2. 以下のコマンドで、ピアリング・オブジェクトがアップになったことを確認します。

```
idg[apiconnect]# show gateway-peering-status

 Address      Configuration name Pending updates Replication offset Link status Primary
 ---------- ------------------ --------------- ------------------ ----------- -------
 9.68.85.96 gwd_peering        0                 302946682          ok          yes
 9.68.85.97 gwd_peering        0                 302946539          ok          no
 9.68.85.98 gwd_peering        0                 302946539          ok          no
```

プライマリーのみ

4.12. API ゲートウェイ・サービスの作成 (プライマリー・ゲートウェイサーバー1)

1. API ゲートウェイ・サービスを作成します。SSL クライアント、SSL サーバープロファイルとゲートウェイ・ピアリングを設定します。

> 情報
> DataPower API Gateway タイプの場合は、**v5-compatibility-mode** を **off** に設定します。

```
config;
apic-gw-service;
   admin-state enabled
   local-address 9.68.85.96
   local-port 3000
   api-gw-address 9.68.85.99
   api-gw-port 9443
   ssl-client gwd_client
   ssl-server gwd_server
   gateway-peering gwd_peering
   v5-compatibility-mode off
exit;
write mem;
```

- **local-address** は、管理サブシステムとのクラスター間通信インターフェースです。ここでは **eth0** を指定します。
- **api-gw-address** は、API ゲートウェイ・サービスが API 呼び出しでリッスンするインターフェースです。ここでは **eth1** を指定します。

2. サービス・プロセスがアップになったことを確認します。これには少し時間がかかる場合があります。

```
idg[apiconnect]# show apic-gw-service;

apic-gw-service: default [up]
------------------------
 admin-state enabled
 local-address 9.68.85.96
 local-port 3000
 ssl-client gwd_client   [up]
 ssl-server gwd_server   [up]
 api-gw-address 9.68.85.99
 api-gw-port 9443
 gateway-peering gwd_peering   [up]
 v5-compatibility-mode off
 slm-mode autounicast
 ip-unicast
```

セカンダリーのみ

4.13. API ゲートウェイ・サービスの作成 (セカンダリー・ゲートウェイサーバー2、3)

1. ゲートウェイサーバー1 で構成した、4.12.のステップを繰り返します。

セカンダリー・サーバー(ゲートウェイサーバー2)

```
config;
apic-gw-service;
  admin-state enabled
  local-address 9.68.85.97
  local-port 3000
  api-gw-address 9.68.85.100
  api-gw-port 9443
  ssl-client gwd_client
  ssl-server gwd_server
  gateway-peering gwd_peering
  v5-compatibility-mode off
exit;
write mem;
```

セカンダリー・サーバー(ゲートウェイサーバー3)

```
config;
apic-gw-service;
  admin-state enabled
  local-address 9.68.85.98
  local-port 3000
  api-gw-address 9.68.85.101
  api-gw-port 9443
  ssl-client gwd_client
  ssl-server gwd_server
  gateway-peering gwd_peering
  v5-compatibility-mode off
exit;
write mem;
```

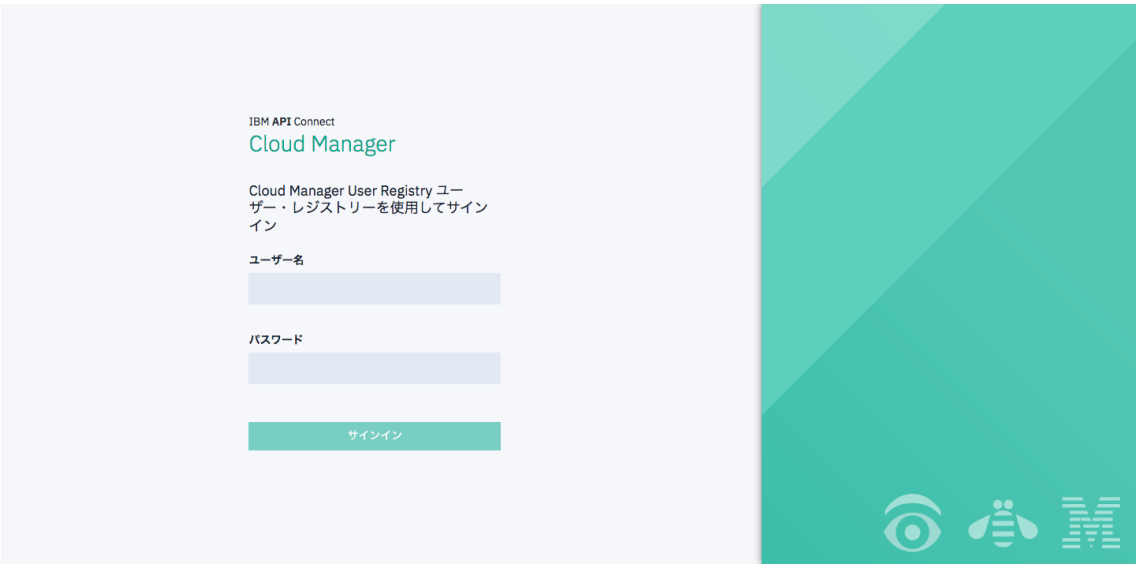
これで、API Connect でゲートウェイ・サービスを構成するためのゲートウェイサーバーの準備ができました。

API Connect の構成

1. クラウド・コンソール・ユーザー・インターフェースへのアクセス[24]

1. ブラウザで https://<*Cloud Admin UI* のエンドポイント *URL*>/admin にアクセスし、クラウド・コンソール・ユーザー・インターフェース（以後、CMC）が表示されることを確認します。

※インストール前に定義した各エンドポイントに対するアクセスは FQDN である必要であるため、ブラウザアクセスするクライアントで解決できるよう構成する必要があります。



2. ユーザー名: **admin** 、パスワード: **7iron-hide** (デフォルト・パスワード)を入力してサインインボタンをクリックし、パスワードを変更します。

3. **Cloud Manager へようこそ**画面が表示されれば、ログイン成功です。

2. 通知のための E メール・サーバーの構成[25]

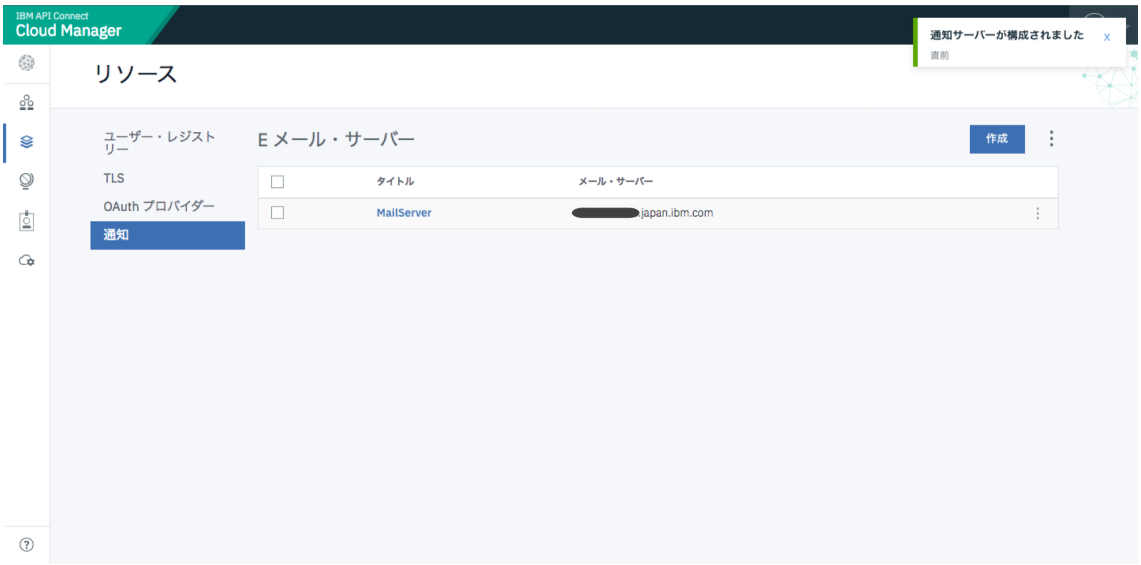
1. ホーム画面から**リソースの管理**をクリックして、メニュー一覧から**通知**をクリックします。

2. **作成**をクリックして、E メール・サーバーを作成します。

3. E メール・サーバーの作成で、E メール・サーバー構成のタイトル、E メールの送信に使用する SMTP サーバーのアドレスとポートを入力し、「保存」ボタンをクリックします。

※SMTP サーバーにユーザー認証がある場合は、ユーザー名とパスワードも入力します。

4. **通知サーバーが構成されました**と表示されたら、E メール・サーバーの構成は完了です。



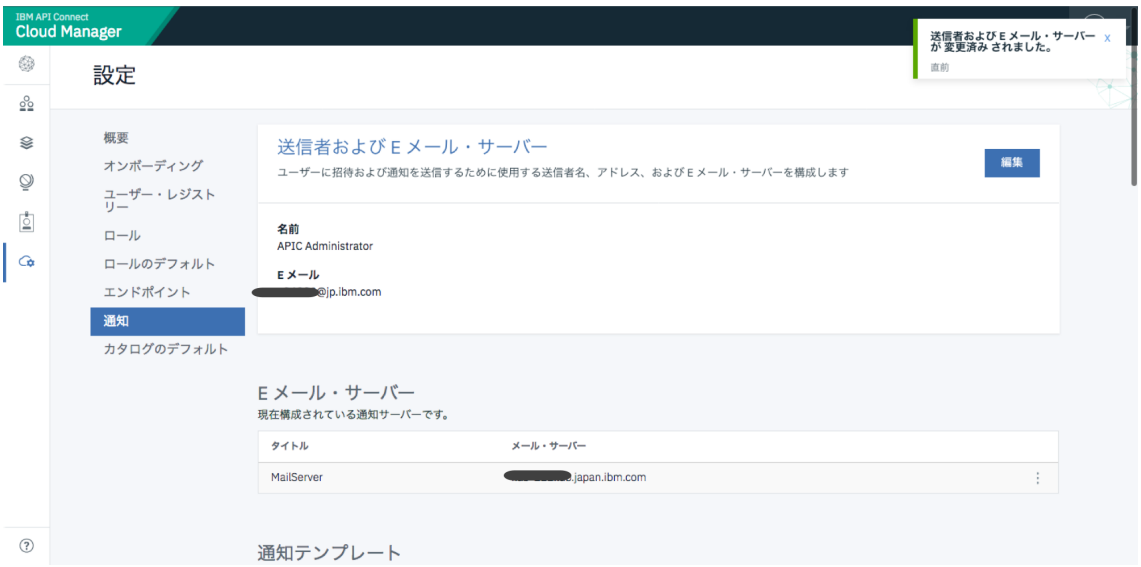
3. 通知の構成[26]

構成した E メール・サーバーを、APIC のイベントに関する自動 E メール通知のサーバーとして登録します。

1. メニューバーから、**設定** ＞ **通知**をクリックします。

2. **編集**をクリックして、送信者および E メール・サーバーを構成します。**保存**をクリックします。

3. **送信者および E メール・サーバーが変更済みされました。**と表示されれば、通知の構成は完了です。



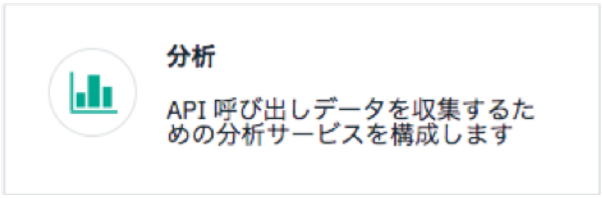
4. トポロジーの定義[27]

4.1. 分析サービスの登録[28]

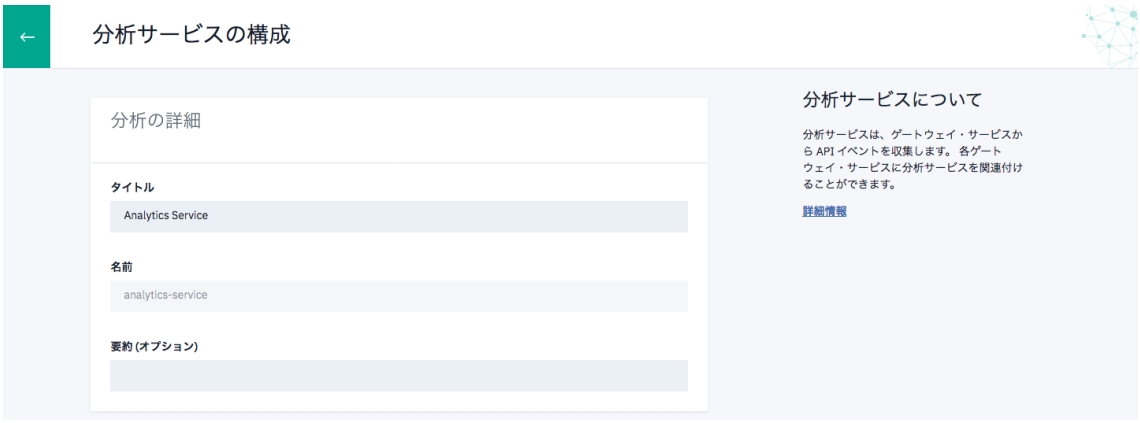
1. ホーム画面から、**トポロジーの構成**をクリックします。

2. **サービスの登録**をクリックします。

3. **分析**をクリックします。



4. **分析の詳細**を編集します。
   - タイトル: *Analytics Service*
   - 名前: *analytics-service*



5. **管理エンドポイント**を編集します。
   - エンドポイント: *https://analytics-client.apic.com*
   - TLS クライアント・プロファイル（オプション）: *Analytics client TLS client profile*

6. **保存**をクリックします。

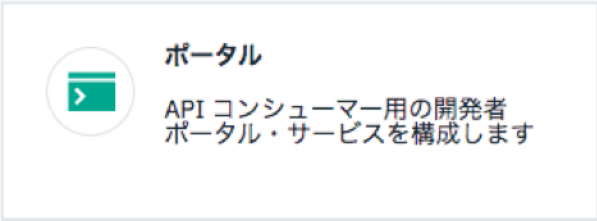
7. 分析サービスが登録されたことを確認します。

4.2. ポータルサービスの登録[29]

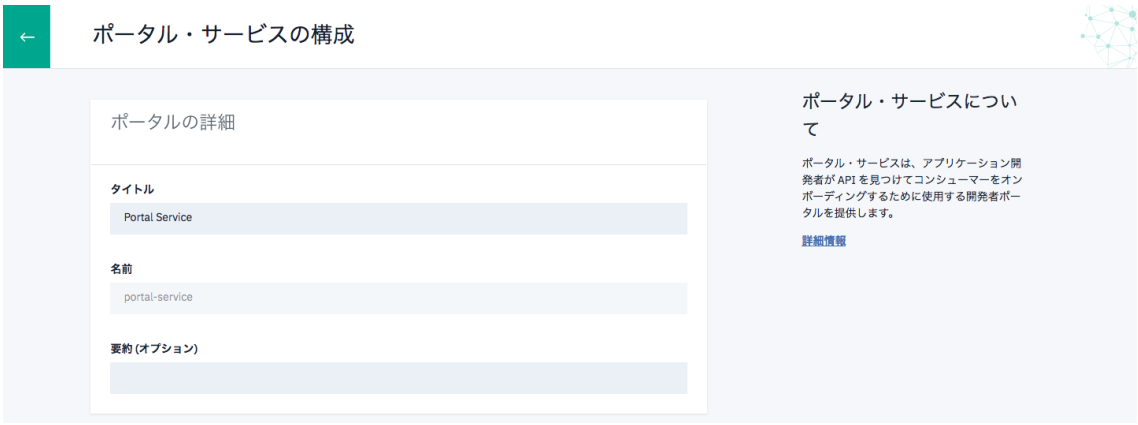
1. ホーム画面から、**トポロジーの構成**をクリックします。

2. **サービスの登録**をクリックします。

3. **ポータル**をクリックします。

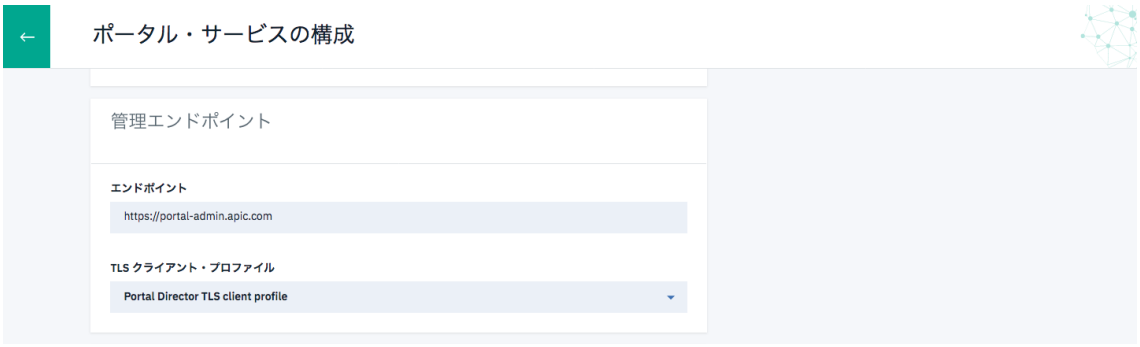


4. **ポータルの詳細**を編集します。
   - タイトル: *Portal Service*
   - 名前: *portal-service*



5. **管理エンドポイント**を編集します。

- エンドポイント: *https://portal-admin.apic.com*
- TLS クライアント・プロファイル（オプション）: *Portal Director TLS client profile*



6. **ポータル Web サイトの URL** を編集します。

- ポータル Web サイトの URL: *https://portal-www.apic.com*

7. **保存**をクリックします。

8. ポータルサービスが登録されたことを確認します。

4.3. ゲートウェイ・サービスの登録[30]

1. ホーム画面から、**トポロジーの構成**をクリックします。

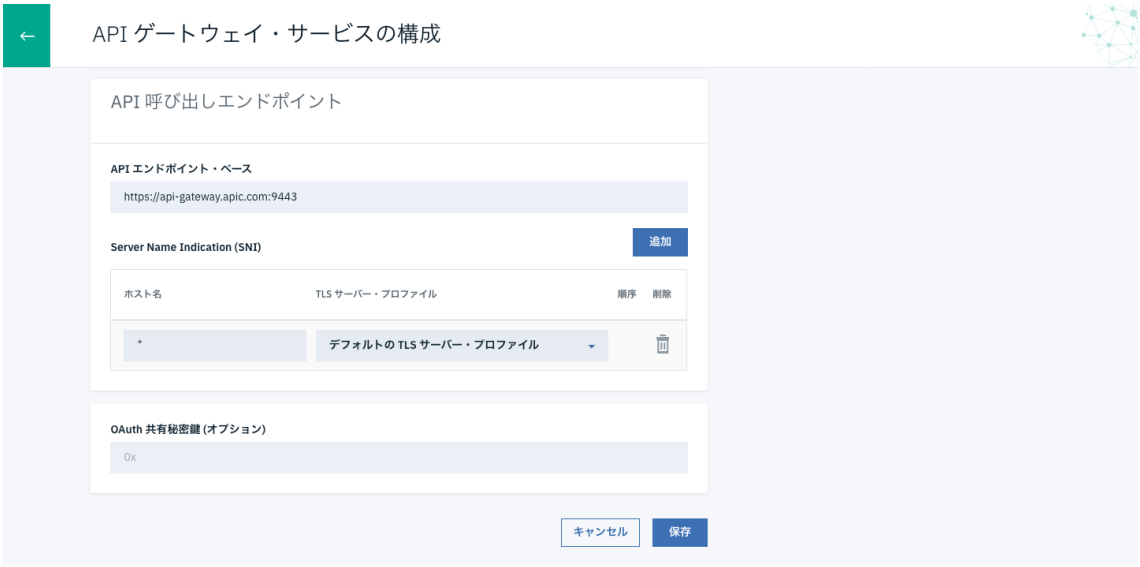
2. **サービスの登録**をクリックします。

3. **DataPower API Gateway** をクリックします。

4. **ゲートウェイの詳細**を編集します。
   - タイトル: *Gateway Service*
   - 名前: *gateway-service*

5. **管理エンドポイント**を編集します。API ゲートウェイ・サービスで定義したポートも入力します。
   - エンドポイント: *https://apic-gw-service.apic.com:3000*
   - TLS クライアント・プロファイル（オプション）: *Default TLS client profile*

6. **API 呼び出しエンドポイント**を編集します。API ゲートウェイ・サービスで定義したポートも入力します。
   - API エンドポイント・ベース: *https://api-gateway.apic.com:9443*



7. OAuth 共有秘密鍵はデフォルト値のままにします。

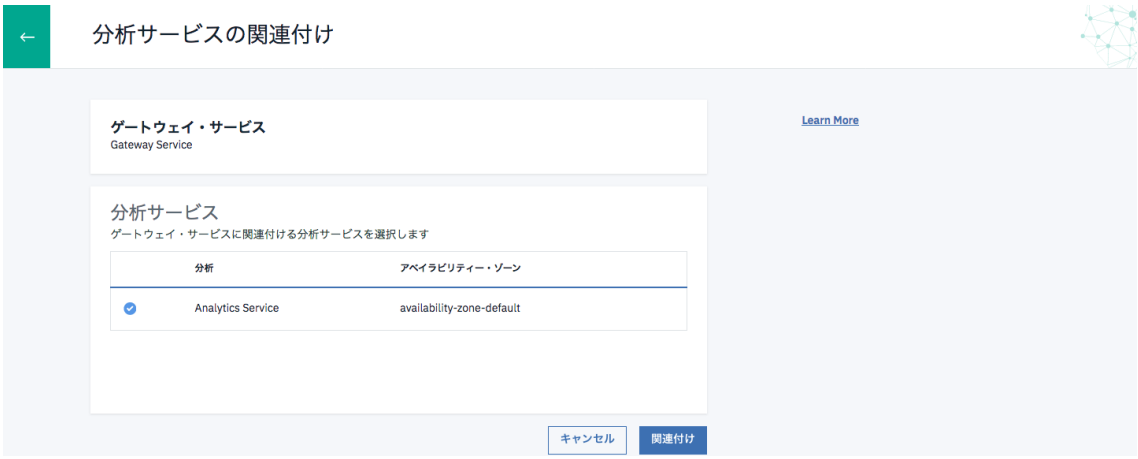
8. **保存**をクリックします。

4.4. ゲートウェイ・サービスへの分析サービスの関連付け[31]

1. ホーム画面から、**トポロジーの構成**をクリックします。

2. ゲートウェイ・サービスに表示される、**分析サービスの関連付け**をクリックします。

3. 作成した分析サービスにチェックを入れ、**関連付け**をクリックします。・



4. これで、ゲートウェイ・サービスで処理した API の分析データが、関連付けた分析サービスにプッシュされるようになります。

以上で、APIC を使用するためのインストールとトポロジー構成が完了しました。

以降の API の作成および公開、製品カタログの管理、公開された API 製品の使用などについては、

Knowledge Center の各チュートリアル[32]をご参照ください。

付録

HAProxy の構成例

- HAProxy バージョン: HA-Proxy version 1.5.18 2016/05/10
- HAProxy 構成: /etc/haproxy/haproxy.cfg

```
# This sample HAProxy configuration file configures one HAProxy node to distribute traffic to
# Management, Portal, Analytics, and Gateway clusters. Another option is to configure one
HAproxy
# node per cluster.

global
#           log localhost           local0
            log /dev/log            local1 notice
            chroot        /var/lib/haproxy
            pidfile       /var/run/haproxy.pid
#           stats socket /run/haproxy/admin.sock mode 660 level admin
            stats timeout 30s
            user haproxy
            group haproxy
            daemon

            stats socket /var/lib/haproxy/stats

            # Default SSL material locations
            ca-base /etc/ssl/certs
            crt-base /etc/ssl/private

            # Default ciphers to use on SSL-enabled listening sockets.
            # For more information, see ciphers(1SSL). This list is from:
            #   https://hynek.me/articles/hardening-your-web-servers-ssl-ciphers/
            ssl-default-bind-ciphers
ECDH+AESGCM:DH+AESGCM:ECDH+AES256:DH+AES256:ECDH+AES128:DH+AES:ECDH+
3DES:DH+3DES:RSA+AESGCM:RSA+AES:RSA+3DES:!aNULL:!MD5:!DSS
            ssl-default-bind-options no-sslv3

defaults
            log             global
            mode        http
            option        httplog
            option        dontlognull
            timeout connect 5000
            timeout client   50000
            timeout server   50000
#           errorfile 400 /etc/haproxy/errors/400.http
#           errorfile 403 /etc/haproxy/errors/403.http
#           errorfile 408 /etc/haproxy/errors/408.http
#           errorfile 500 /etc/haproxy/errors/500.http
#           errorfile 502 /etc/haproxy/errors/502.http
#           errorfile 503 /etc/haproxy/errors/503.http
#           errorfile 504 /etc/haproxy/errors/504.http
```

```
######## frontend CONFIGURATION ########
frontend front
    mode tcp
    option tcplog
    #
    # Map to the hostname and TCP port for the Management load balancer.
    # In this example, the hostname for the load balancer is 9.68.85.104.
    #
    bind 9.68.85.104:443
    tcp-request inspect-delay 5s
    tcp-request content accept if { req_ssl_hello_type 1 }

        #
        # The value for the Management endpoints as defined in the apiconnect-up.yml
        # file using the apicup installer. In this example, the endpoints are api-manager-
ui.apic.test,
        # cloud-admin-ui.apic.test, consumer-api-ui.apic.test, and
        # platform-api-ui.apic.test. Standard SNI structure specifies
        # whether the INCOMING request is for api-manager or cloud-admin or for consumer-
api or platform-api
        # then use "be_management".
        #
    use_backend be_management if { req_ssl_sni -i api-manager-ui.apic.com OR req_ssl_sni
-i cloud-admin-ui.apic.com }
    use_backend be_management if { req_ssl_sni -i consumer-api.apic.com OR req_ssl_sni -i
platform-api.apic.com }
    use_backend be_portal_traffic if { req_ssl_sni -i portal-admin.apic.com }
    use_backend be_portal_public if { req_ssl_sni -i portal-www.apic.com }
    use_backend be_analytics if { req_ssl_sni -i analytics-ingestion.apic.com OR req_ssl_sni -
i analytics-client.apic.com }
            #
            # be_management is defined to point management traffic to the cluster
            # containing three management nodes
            #
backend be_management
    mode tcp
    option tcplog
    balance roundrobin
#     option ssl-hello-chk

    #
    # One entry per Management node in the cluster.
    # Hostname and TCP Port for each Management node.
    #
    server management0 9.68.84.194:443 check inter 10s rise 2 fall 2
    server management1 9.68.84.195:443 check inter 10s rise 2 fall 2
    server management2 9.68.84.196:443 check inter 10s rise 2 fall 2


backend be_portal_traffic
    mode tcp
```

```
    option tcplog
    balance roundrobin
#     option ssl-hello-chk

    #
    # One entry per Portal node.
    # Hostname and TCP Port for the Portal node.
    #
   server portal0 9.68.85.93:443 check inter 10s rise 2 fall 2
   server portal1 9.68.85.94:443 check inter 10s rise 2 fall 2
   server portal2 9.68.85.95:443 check inter 10s rise 2 fall 2


backend be_portal_public
    mode tcp
    option tcplog
    balance roundrobin
#     option ssl-hello-chk

    #
    # One entry per Portal node.
    # Hostname and TCP Port for the Portal node.
    #
    server portal0 9.68.83.14:443 check inter 10s rise 2 fall 2
    server portal1 9.68.83.15:443 check inter 10s rise 2 fall 2
    server portal2 9.68.83.16:443 check inter 10s rise 2 fall 2


backend be_analytics
    mode tcp
    option tcplog
    balance roundrobin
#     option ssl-hello-chk
    #
    # One entry per Analytics node.
    # Hostname and TCP Port for the Analytics node.
    #
    server analytics0 9.68.85.90:443 check inter 10s rise 2 fall 2
    server analytics1 9.68.85.91:443 check inter 10s rise 2 fall 2
    server analytics2 9.68.85.92:443 check inter 10s rise 2 fall 2

listen hastats 0.0.0.0:8080
    mode http
    maxconn 64
    timeout connect 5000
    timeout client 10000
    timeout server 10000
    stats enable
    stats show-legends
    stats uri /haproxy?hastats

    stats auth user:password
```

```
frontend front3000
      mode tcp
      option tcplog
      #
      # Map to the hostname and TCP port for the Management load balancer.
      # In this example, the hostname for the load balancer is 9.68.85.104.
      #
      bind 9.68.85.104:3000
      tcp-request inspect-delay 5s
      tcp-request content accept if { req_ssl_hello_type 1 }

          #
          # The value for the Management endpoints as defined in the apiconnect-up.yml
          # file using the apicup installer. In this example, the endpoints are api-manager-
ui.apic.test,
          # cloud-admin-ui.apic.test, consumer-api-ui.apic.test, and
          # platform-api-ui.apic.test. Standard SNI structure specifies
          # whether the INCOMING request is for api-manager or cloud-admin or for consumer-
api or platform-api
          # then use "be_management".
          #
      use_backend be_gateway_3000 if { req_ssl_sni -i apic-gw-service.apic.com }

frontend front9443
      mode tcp
      option tcplog
      #
      # Map to the hostname and TCP port for the Management load balancer.
      # In this example, the hostname for the load balancer is 9.68.85.104.
      #
      bind 9.68.85.104:9443
      tcp-request inspect-delay 5s
      tcp-request content accept if { req_ssl_hello_type 1 }

          #
          # The value for the Management endpoints as defined in the apiconnect-up.yml
          # file using the apicup installer. In this example, the endpoints are api-manager-
ui.apic.test,
          # cloud-admin-ui.apic.test, consumer-api-ui.apic.test, and
          # platform-api-ui.apic.test. Standard SNI structure specifies
          # whether the INCOMING request is for api-manager or cloud-admin or for consumer-
api or platform-api
          # then use "be_management".
          #
      use_backend be_gateway_9443 if { req_ssl_sni -i api-gateway.apic.com }


backend be_gateway_3000
      mode tcp
      option tcplog
      balance roundrobin
```

```
#      option ssl-hello-chk
       #
       # One entry per Gateway node.
       # Hostname and TCP Port for the Gateway node.
       #
       server gateway3 9.68.85.96:3000 check inter 10s rise 2 fall 2
       server gateway4 9.68.85.97:3000 check inter 10s rise 2 fall 2
       server gateway5 9.68.85.98:3000 check inter 10s rise 2 fall 2

backend be_gateway_9443
       mode tcp
       option tcplog
       balance roundrobin
#      option ssl-hello-chk

       #
       # One entry per Gateway node.
       # Hostname and TCP Port for the Gateway node.
       #
       server gateway0 9.68.85.99:9443 check inter 10s rise 2 fall 2
       server gateway1 9.68.85.100:9443 check inter 10s rise 2 fall 2
       server gateway2 9.68.85.101:9443 check inter 10s rise 2 fall 2
```

参照

[1] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/SSMNED_2018/com.ibm.apic.install.doc/installing_vm.html [Accessed 17 Jun. 2019].

[2] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/overview_apimgmt_requirements.html [Accessed 17 Jun. 2019].

[3] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/tapim_portal_ova_install.html [Accessed 17 Jun. 2019].

[4] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/certs_overview_vm.html [Accessed 17 Jun. 2019].

[5] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/SSMNED_2018/com.ibm.apic.install.doc/capic_deploy_overview_vm.html [Accessed 18 Jun. 2019].

[6] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/overview_apimgmt_portreqs_vmware.html [Accessed 17 Jun. 2019].

[7] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/overview_vmware_reqs.html [Accessed 20 Jun. 2019].

[8] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/HA_topology_ova.html [Accessed 20 Jun. 2019].

[9] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/overview_installing_ova_first_steps.html [Accessed 20 Jun. 2019].

[10] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/overview_installing_mgmtvm_apimgmt.html [Accessed 9 Jul. 2019].

[11] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/tapic_db_backup_ova.html [Accessed 10 Jul. 2019].

[12] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/logging_ova.html [Accessed 10 Jul. 2019].

[13] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/tapim_add_static_route_ova.html [Accessed 10 Jul. 2019].

[14] Developer.ibm.com. (2019). *API Connect V2018 Whitepaper Now Available - API Connect*. [online] Available at: https://developer.ibm.com/apiconnect/2019/02/08/api-connect-v2018-deployment-whitepaper-now-available/ [Accessed 11 Jul. 2019].

[15] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/overview_installing_analytics_ova.html [Accessed 11 Jul. 2019].

[16] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/tapim_portal_ova_install.html [Accessed 11 Jul. 2019].

[17] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/tapic_db_backup_portal.html [Accessed 11 Jul. 2019].

[18] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.install.doc/tapic_install_datapower_gateway.html [Accessed 11 Jul. 2019].

[19] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.overview.doc/rapic_gateway_types.html#reference_rww_k4s_zdb [Accessed 11 Jul. 2019].

[20] Www-01.ibm.com. (2019). *IBM Deprecated and removed features in versions 2018.4 and earlier of IBM DataPower Gateways products*. [online] Available at: https://www-01.ibm.com/support/docview.wss?uid=swg21634531#dr2018 [Accessed 11 Jul. 2019].

[21] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/SS9H2Y_7.7.0/com.ibm.dp.doc/virtual_deployingvmware.html [Accessed 11 Jul. 2019].

[22] Ibm.com. (2019). *IBM Knowledge Center Error*. [online] Available at: https://www.ibm.com/support/knowledgecenter/SS9H2Y_7.7.0/com.ibm.dp.doc/virtual_installingsoftwareonvirtualappliance.html [Accessed 11 Jul. 2019].

[23] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SS9H2Y_7.7.0/com.ibm.dp.doc/firmware_image_applying.html [Accessed 11 Jul. 2019].

[24] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/login.html [Accessed 11 Jul. 2019].

[25] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/config_emailserver.html [Accessed 11 Jul. 2019].

[26] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/task_cmc_config_notifications.html [Accessed 11 Jul. 2019].

[27] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/api_create.html [Accessed 12 Jul. 2019].

[28] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/config_analytics.html [Accessed 12 Jul. 2019].

[29] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/config_portal.html [Accessed 12 Jul. 2019].

[30] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/config_gateway.html [Accessed 12 Jul. 2019].

[31] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.cmc.doc/associate_analytics.html [Accessed 12 Jul. 2019].

[32] Ibm.com. (2019). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SSMNED_2018/com.ibm.apic.tutorials.doc/tutorials_home.html [Accessed 12 Jul. 2019].