



テクニカルホワイトペーパー

Veritas InfoScale Enterprise 7.4.1 for RHEL on Azure
構築手順書
サブネットを跨いだ FSS + OverlayIP 切替 編

2020 年 3 月

ベリタステクノロジーズ合同会社
テクノロジーセールス本部

VERITAS™

The truth in information.

免責事項

ベリタステクノロジーズ合同会社は、この文書の著作権を留保します。また、記載された内容の無謬性を保証しません。Veritas InfoScale は将来に渡って仕様を変更する可能性を常に含み、これらは予告なく行われることもあります。なお、当ドキュメントの内容は参考資料として、読者の責任において管理/配布されるようお願いいたします。

目次

| | |
|---|-----------|
| 免責事項 | 2 |
| 1. はじめに | 5 |
| 本書の目的..... | 5 |
| 2. システム構成 | 6 |
| 本書の前提となるシステム構成..... | 6 |
| AZURE 上に RHEL を構築する場合の注意点..... | 7 |
| 3. インストール | 11 |
| インストール..... | 11 |
| 4. VXVM と VXFS の設定 | 19 |
| ディスクの確認とイニシャライズ..... | 19 |
| ディスクのイニシャライズと EXPORT..... | 19 |
| VXVM のディスクグループの作成..... | 20 |
| VXVM のボリュームと VXFS のファイルシステムの作成..... | 21 |
| 既存のサービスグループ:CVM への設定追加..... | 22 |
| 5. PYTHON SDK FOR AZURE のインストールと AZURE 関連の各種設定 | 24 |
| PYTHON SKD FOR AZURE のインストール..... | 24 |
| サービスプリンシパルの作成と権限付与..... | 25 |
| INFOSCALE のパラメーター設定に必要な情報の確認..... | 29 |
| 6. クラスターの設定 | 33 |
| アプリケーションを管理するサービスグループのセットアップ..... | 33 |
| APACHE のインストールと必要な設定..... | 33 |
| 異なるサブネット間でネットワーク経路を切り替える為の設定..... | 35 |
| APACHE のクラスタリング..... | 37 |
| 7. 動作確認と切り替えテスト | 41 |
| アプリケーションを管理するサービスグループの手動切り替え..... | 41 |
| 障害によるサービスグループの自動切り替え..... | 42 |
| 障害ノードの復旧..... | 42 |
| アプリケーション(APACHE)障害..... | 42 |

| | |
|--------------------|-----------|
| 8. 付録 | 43 |
| MAIN.CF の内容..... | 43 |

1. はじめに

本書の目的

本書は、InfoScale Enterprise 7.4.1 を用いて、Azure 上の異なるサブネット上に配備された 2 台の RHEL 仮想マシンを跨いで FSS を用いた仮想ミラーリングを行い、且つ OverlayIP によるルートテーブルの切り替えによってネットワークを切り替える 2 ノード稼働待機型クラスター構成の構築の理解を目的に作成されています。尚、クラスタリング対象のアプリケーションとして apache を用いています。

2. システム構成

本書の前提となるシステム構成

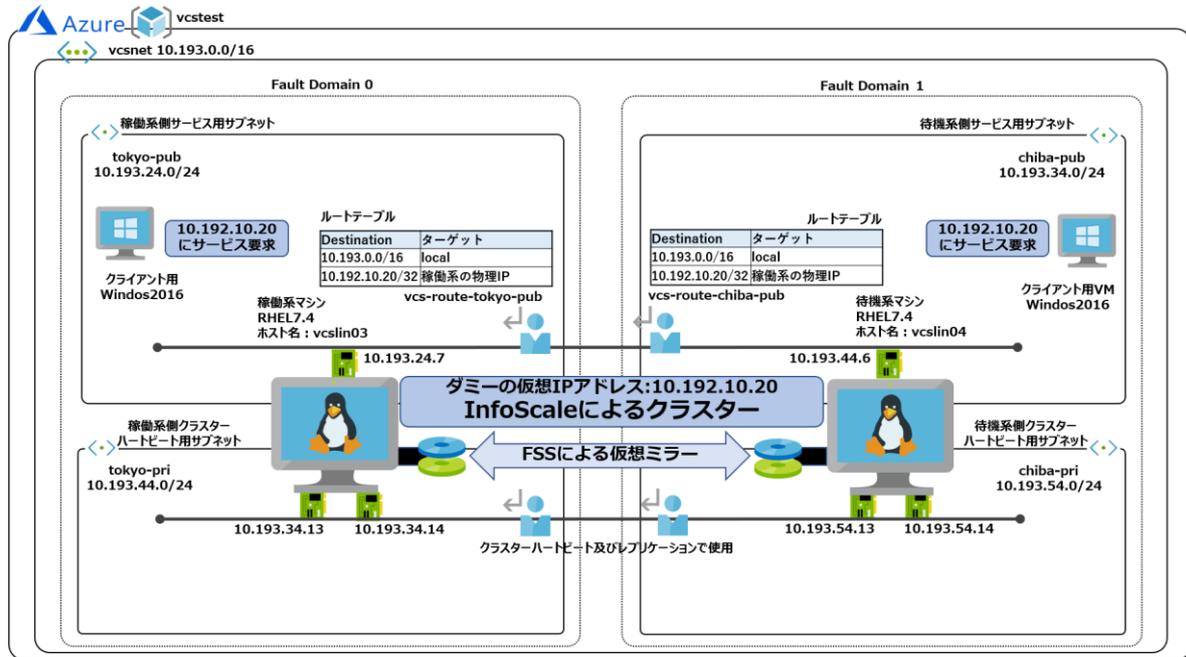


図 1 InfoScale 構築後の構成イメージ

Azure 上に RHEL を構築する場合の注意点

InfoScale は、Azure 上の RHEL での稼働を保証しています。ただし、InfoScale を構築する場合は、以下の条件を満たすように RHEL を構築してください。

- **2つのノード間で適切な通信ができるよう firewall や セキュリティグループの設定を調整する事：**
InfoScale はインストーラー実行時や稼働時に、以下の通信を行います。Firewall や Azure のセキュリティグループを適切に変更してください。
 - ping
 - ssh(Port22)
 - クラスタースタートアップ (Port50000, 50001)
- **RHEL のインスタンスに ssh を用いて root でパスワードを用いてログインできること：** InfoScale のインストール時は、1 台のノードから push install を行いますが、その際に他のノードに ssh を用いて root でパスワードを用いてログインします。Azure 上で deploy される RHEL は、デフォルトで ssh を用いて root でパスワードを用いてログインできるようになっていません。/etc/passwd を変更してパスワードを用いて root で ssh ができるようにしてください。
- **yum が使用できること：** Azure 上で deploy される RHEL には、InfoScale が必要とするパッケージの幾つかがインストールされていません。そのため、InfoScale のインストーラーの中で yum を用いて必要なパッケージをインストールします。適切なネットワークの設定（踏み台サーバーを経由して yum のサーバーにアクセス等）もしくは yum のリポジトリの設定を行い、yum が使用できるようにしてください。
- **Python SDK for Azure が使用できること：** InfoScale は、プライベート IP を管理したりルートテーブルを切り替えるために Python SDK for Azure を使用して IAM 経由でコントロールします。InfoScale がインストールされるノードが IAM に接続できるようにし、RHEL 仮想マシンを IAM 経由でコントロールできるようにしてください。
- **データ配置用のブロックストレージが、両方のノードに割り当てられている事：** Azure 上のディスク（ブロックストレージ）はリージョン等を跨ったインスタンス（RHEL 仮想マシン）間で切り替える事ができません。その為、本構成では仮想ミラーリングを使用します。InfoScale によるレプリケーション対象として、各ノード最低 1 つのデータ格納用ディスクを RHEL 仮想マシンに紐付けてください。

- /opt に 4Gbyte 以上の空き領域があること：Azure 上に配備される RHEL のバージョンによっては、/opt ファイルシステムの容量が 2Gbyte しかありません。InfoScale のインストール時は、/opt ファイルシステムに 4Gbyte 以上の空き領域が必要ですので、/opt ファイルシステムを拡張してください。

```
# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        7.9G  0  7.9G  0% /dev
tmpfs           7.9G  0  7.9G  0% /dev/shm
tmpfs           7.9G  9.2M 7.9G  1% /run
tmpfs           7.9G  0  7.9G  0% /sys/fs/cgroup
/dev/mapper/rootvg-rootlv 2.0G  69M 2.0G  4% /
/dev/mapper/rootvg-usrlv  10G  1.4G 8.7G 14% /usr
/dev/mapper/rootvg-home1v 1014M  33M 982M  4% /home
/dev/mapper/rootvg-varlv  8.0G 165M 7.9G  3% /var
/dev/mapper/rootvg-optlv  2.0G 140M 1.9G  7% /opt
/dev/mapper/rootvg-tmplv  2.0G  33M 2.0G  2% /tmp
/dev/sda2        494M 106M 388M 22% /boot
/dev/sda1        500M  9.7M 491M  2% /boot/efi
/dev/sdb1        20G  2.1G 17G 12% /mnt/resource
tmpfs            1.6G  0  1.6G  0% /run/user/1000
tmpfs            1.6G  0  1.6G  0% /run/user/0

# lvextend -L 10g /dev/rootvg/optlv
Size of logical volume rootvg/optlv changed from 2.00 GiB (512 extents) to 10.00 GiB (2560 extents).
Logical volume rootvg/optlv successfully resized.

# xfs_growfs /dev/mapper/rootvg-optlv
meta-data=/dev/mapper/rootvg-optlv isize=512    agcount=4, agsize=131072 blks
         =                       sectsz=4096   attr=2, projid32bit=1
         =                       crc=1        finobt=0 spinodes=0
data     =                       bsize=4096   blocks=524288, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal              bsize=4096   blocks=2560, version=2
         =                       sectsz=4096   sunit=1 blks, lazy-count=1
realtime =none                  extsz=4096   blocks=0, rtextents=0
data blocks changed from 524288 to 2621440

# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        7.9G  0  7.9G  0% /dev
tmpfs           7.9G  0  7.9G  0% /dev/shm
tmpfs           7.9G  9.1M 7.9G  1% /run
tmpfs           7.9G  0  7.9G  0% /sys/fs/cgroup
/dev/mapper/rootvg-rootlv 2.0G  69M 2.0G  4% /
/dev/mapper/rootvg-usrlv  10G  1.4G 8.7G 14% /usr
/dev/mapper/rootvg-home1v 1014M  33M 982M  4% /home
/dev/mapper/rootvg-varlv  8.0G 165M 7.9G  3% /var
/dev/mapper/rootvg-optlv  10G 141M 9.9G  2% /opt
/dev/mapper/rootvg-tmplv  2.0G  33M 2.0G  2% /tmp
/dev/sda2        494M 106M 388M 22% /boot
/dev/sda1        500M  9.7M 491M  2% /boot/efi
/dev/sdb1        20G  2.1G 17G 12% /mnt/resource
tmpfs            1.6G  0  1.6G  0% /run/user/1000
```

- サービスネットワークとは別にクラスターハートビート及びレプリケーション用のネットワークを持ち、稼働系と待機系で疎通が可能である事：FSS による仮想ミラーリングによる InfoScale によるクラスターでは、サービス用ネットワークと別にクラスターハートビートネットワークが 2 系統必要です。他の構成の場合は、クラスターハートビートネットワークは 1 系統で十分ですが、FSS の場合はミラーリングによる完全同期を行う関係上、パフォーマンスを向上させるために 2 系統必要です。Azure 上の RHEL 仮想マシンでは、NIC を追加する事は簡単にできますが、RHEL 仮想マシン間でサブネットが異なる場合は OS 側でルーティングの設定を行わないと疎通できません。下記は、5 ページの構成図において、10.193.44.0 のネットワークに接続された 1 号機の eth1 (10.193.44.13) と、10.193.54.0 のネットワークに接続された 2 号機の eth1 (10.193.54.13) のペア、及び 1 号機の eth2 (10.193.44.14) と、2 号機の eth2 (10.193.54.14) のペアが、それぞれ疎通できるようにルーティングの設定を行う例です。Azure の場合、デフォルトゲートウェイの IP アドレスは各ネットワークの最初の IP アドレス (例えば 10.193.44.0 のネットワークなら、10.193.44.1) に固定されておりますので注意が必要です。尚、下記の nmcli コマンドによるルーティングの設定は reboot すると無効になってしまいますので、起動スクリプトに含める等の対応により OS 起動時に必ず有効になるように設定してください。

1 号機の例

```
# ifconfig -a
# ifconfig -a
eth0: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 10.193.24.7 netmask 255.255.255.0 broadcast 10.193.24.255
    . . . (途中略) . . .

eth1: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 10.193.44.13 netmask 255.255.255.0 broadcast 10.193.44.255
    inet6 fe80::afd8:2567:74ba:49a6 prefixlen 64 scopeid 0x20<link>
    ether 00:0d:3a:52:5e:34 txqueuelen 1000 (Ethernet)
    RX packets 2 bytes 762 (762.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10 bytes 1308 (1.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth2: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 10.193.44.14 netmask 255.255.255.0 broadcast 10.193.44.255
    inet6 fe80::feba:d22e:9003:f2a1 prefixlen 64 scopeid 0x20<link>
    ether 00:0d:3a:52:5c:08 txqueuelen 1000 (Ethernet)
    RX packets 2 bytes 762 (762.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10 bytes 1308 (1.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    . . . (以下略) . . .

# nmcli device modify eth1 +ipv4.routes "10.193.54.13/32 10.193.44.1"
Connection successfully reapplied to device 'eth1'.
# nmcli device modify eth2 +ipv4.routes "10.193.54.14/32 10.193.44.1"
Connection successfully reapplied to device 'eth2'.
```

| Kernel IP routing table | | | | | | |
|-------------------------|--------------------|------------------------|------------|------------|----------|---------------|
| Destination | Gateway | Genmask | Flags | Metric | Ref | Use Iface |
| 0.0.0.0 | 10.193.24.1 | 0.0.0.0 | UG | 100 | 0 | 0 eth0 |
| 10.193.24.0 | 0.0.0.0 | 255.255.255.0 | U | 100 | 0 | 0 eth0 |
| 10.193.44.0 | 0.0.0.0 | 255.255.255.0 | U | 100 | 0 | 0 eth2 |
| 10.193.44.0 | 0.0.0.0 | 255.255.255.0 | U | 101 | 0 | 0 eth1 |
| 10.193.54.13 | 10.193.44.1 | 255.255.255.255 | UGH | 100 | 0 | 0 eth1 |
| 10.193.54.14 | 10.193.44.1 | 255.255.255.255 | UGH | 100 | 0 | 0 eth2 |
| 168.63.129.16 | 10.193.24.1 | 255.255.255.255 | UGH | 100 | 0 | 0 eth0 |
| 169.254.169.254 | 10.193.24.1 | 255.255.255.255 | UGH | 100 | 0 | 0 eth0 |

2 号機の例

```
# ifconfig -a
eth0: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 10.193.34.6 netmask 255.255.255.0 broadcast 10.193.34.255
    . . . (途中略) . . .

eth1: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 10.193.54.13 netmask 255.255.255.0 broadcast 10.193.54.255
    inet6 fe80::1abe:5a0f:811f:1f15 prefixlen 64 scopeid 0x20<link>
    ether 00:0d:3a:50:9d:13 txqueuelen 1000 (Ethernet)
    RX packets 2 bytes 762 (762.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 13 bytes 1518 (1.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth2: flags=4163<UP, BROADCAST, RUNNING, MULTICAST> mtu 1500
    inet 10.193.54.14 netmask 255.255.255.0 broadcast 10.193.54.255
    inet6 fe80::b281:95e3:1b57:80fb prefixlen 64 scopeid 0x20<link>
    ether 00:0d:3a:51:58:a3 txqueuelen 1000 (Ethernet)
    RX packets 2 bytes 762 (762.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 13 bytes 1518 (1.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0 . . . (以下略) . . .

# nmcli device modify eth1 +ipv4.routes "10.193.44.13/32 10.193.54.1"
Connection successfully reapplied to device 'eth1'.
# nmcli device modify eth2 +ipv4.routes "10.193.44.14/32 10.193.54.1"
Connection successfully reapplied to device 'eth2'.

# route -n
Kernel IP routing table
Destination    Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0        0.0.0.0         0.0.0.0         UG    100    0      0 eth0
10.193.34.0    0.0.0.0         255.255.255.0   U     100    0      0 eth0
10.193.44.13  10.193.54.1    255.255.255.255 UGH   100    0      0 eth1
10.193.44.14  10.193.54.1    255.255.255.255 UGH   100    0      0 eth2
10.193.54.0    0.0.0.0         255.255.255.0   U     100    0      0 eth2
10.193.54.0    0.0.0.0         255.255.255.0   U     101    0      0 eth1
168.63.129.16 10.193.34.1     255.255.255.255 UGH   100    0      0 eth0
169.254.169.254 10.193.34.1    255.255.255.255 UGH   100    0      0 eth0
```

3. インストール

インストール

ここでは、InfoScale Enterprise のインストールを行います。

メディアイメージ配下の “installer” を実行してください

```
# ./installer
```

“I” (Install) を指定して、先に進んでください

```

Veritas InfoScale Storage and Availability Solutions 7.4.1 Install Program

Task Menu:

P) Perform a Pre-Installation Check      I) Install a Product
C) Configure a Product Component        G) Upgrade a Product
O) Perform a Post-Installation Check    U) Uninstall a Product
L) License a Product                    S) Start a Product
D) View Product Descriptions           X) Stop a Product
R) View Product Requirements           ?) Help

Enter a Task: [P, I, C, G, O, U, L, S, D, X, R, ?] I

```

InfoScale Enterprise をインストールするので、“4”を指定し、Configuration の選択の部分では“4”の SFCFSHA を指定し、ライセンスに同意して、先に進んでください。

```

Veritas InfoScale Storage and Availability Solutions 7.4.1 Install Program

1) Veritas InfoScale Foundation
2) Veritas InfoScale Availability
3) Veritas InfoScale Storage
4) Veritas InfoScale Enterprise
b) Back to previous menu

Select a product to install: [1-4, b, q, ?] 4

Would you like to configure InfoScale Enterprise after installation? [y, n, q] (n) y

1) Cluster Server (VCS)
2) Storage Foundation (SF)
3) Storage Foundation and High Availability (SFHA)
4) Storage Foundation Cluster File System HA (SFCFSHA)
5) Storage Foundation for Oracle RAC (SF Oracle RAC)

Select a component to configure: [1-5, q] 4

This product may contain open source and other third party materials that are subject
to a separate license. See the applicable Third-Party Notice at
https://www.veritas.com/about/legal/license-agreements

Do you agree with the terms of the End User License Agreement as specified in the EULA/en/EULA.pdf file
present on media? [y, n, q, ?] y

```

インストールすべきホスト名を指定して、先に進んでください。複数のノードに同時にインストール可能です。

```
Enter the system names separated by spaces: [q, ?] vcslin03 vcslin04
```

途中略

キーレスライセンス: "2" を選択し、"4" の「Veritas InfoScale Enterprise」を指定して先に進んでください。

```

Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

To comply with the terms of our End User License Agreement, you have 60 days to either:

* Enter a valid license key matching the functionality in use on the systems
* Enable keyless licensing and manage the systems with a Management Server. For more details visit
http://www.veritas.com/community/blogs/introducing-keyless-feature-enablement-storage-foundation-ha-51.
The
product is fully functional during these 60 days.

1) Enter a valid license key(Key file path needed)
2) Enable keyless licensing and complete system licensing later

How would you like to license the systems? [1-2, q] (2) 2

1) Veritas InfoScale Foundation
2) Veritas InfoScale Availability
3) Veritas InfoScale Storage
4) Veritas InfoScale Enterprise
b) Back to previous menu

Which product would you like to register? [1-4, b, q] (4) 4
Registering keyless key ENTERPRISE on Veritas InfoScale Enterprise
Successfully registered ENTERPRISE keyless key on vcslin03
Successfully registered ENTERPRISE keyless key on vcslin04

```

InfoScale7.4.1 から、Veritas Telemetry Collector が実装されました。これは、顧客が望む場合に限り、InfoScale の情報をネットワーク経由でベリタスに送付し、ベリタス主体でライセンス管理等を行えるようにするオプション機能です。もちろん使用しない事も可能です。インストーラーでは、この機能を使用する前提でエッジサーバーの IP とポートの情報を要求します。本件では、Veritas Telemetry Collector は使用しませんので、エッジサーバーの IP アドレスとポートは実在しない物を指定します。実在しない物を指定しても、インストールは正しく行われ、InfoScale も問題なく機能しますので、ご安心ください。

```

The Veritas Cloud Receiver (VCR) is a preconfigured, cloud-based edge server deployed by Veritas. Enter
telemetry.veritas.com to use the Veritas Cloud Receiver as an edge server for your environment.
Enter the edge server's hostname/ip: [q, ?] 172.31.3.115
CPI WARNING V-9-40-1382 Could not ping the Edge server 172.31.3.115 from following hosts: vcslin03 vcslin04
Please make sure 172.31.3.115 is accessible from vcslin03 vcslin04 for telemetry collector to work.
Installer will proceed.

Press [Enter] to continue:
Enter the edge server's port number: [q, ?] 8080

```

本書の前提では、I/O Fencing は構成しないので、“n” を指定して先に進んでください。

```
I/O Fencing

It needs to be determined at this time if you plan to configure I/O Fencing in enabled or disabled mode, as well as help in determining the number of network interconnects (NICs) required on your systems. If you configure I/O Fencing in enabled mode, only a single NIC is required, though at least two are recommended.

A split brain can occur if servers within the cluster become unable to communicate for any number of reasons. If I/O Fencing is not enabled, you run the risk of data corruption should a split brain occur. Therefore, to avoid data corruption due to split brain in CFS environments, I/O Fencing has to be enabled.

If you do not enable I/O Fencing, you do so at your own risk

See the Administrator's Guide for more information on I/O Fencing

Do you want to configure I/O Fencing in enabled mode? [y, n, q, ?] (y) n
```

Enter を押して次に進んでください。

```
Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

To configure VCS, answer the set of questions on the next screen.

When [b] is presented after a question, 'b' may be entered to go back to the first question of the configuration set.

When [?] is presented after a question, '?' may be entered for help or additional information about the question.

Following each set of questions, the information you have entered will be presented for confirmation. To repeat the set of questions and correct any previous errors, enter 'n' at the confirmation prompt.

No configuration changes are made to the systems until all configuration questions are completed and confirmed.

Press [Enter] to continue:
```

クラスター名をしていきます。これは任意なので、今回は vcsazure3 を指定しています。

```
Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

To configure VCS for InfoScale Enterprise the following information is required:

A unique cluster name
Two or more NICs per system used for heartbeat links
A unique cluster ID number between 0-65535

One or more heartbeat links are configured as private links
You can configure one heartbeat link as a low-priority link

All systems are being configured to create one cluster.

Enter the unique cluster name: [q, ?] vcsazure3
```

FSS を用いる場合、**ハートビートは必ず LLT over UDP** を選択してください。使用する NIC は eth1 と eth2 です。あらかじめハートビート用に設定しておいた IP アドレス（1 号機は eth1:10.193.44.13 と eth2:10.193.44.14、2 号機は eth1:10.193.54.13 と eth2:10.193.54.14）、使用するポートは eth1 間は 50000、eth2 間は 50001 を指定します。VVR を用いるクラスターではハートビート用の NIC が 1 つで構成可能ですが、Azure 上の FSS の場合はハートビート用の NIC が 2 つ必要です。

```

Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

1) Configure the heartbeat links using LLT over Ethernet
2) Configure the heartbeat links using LLT over UDP
3) Configure the heartbeat links using LLT over TCP
4) Configure the heartbeat links using LLT over RDMA
5) Automatically detect configuration for LLT over Ethernet
b) Back to previous menu

How would you like to configure heartbeat links? [1-5, b, q, ?] (5) 2

Discovering NICs on vcslin03 ..... Discovered eth0 eth1 eth2

Enter the NIC for the first private heartbeat link on vcslin03: [b, q, ?] (eth1)
Some configured IP addresses have been found on the NIC eth1 in vcslin03, Do you want
to choose one for the first private heartbeat link? [y, n, q, ?] (y)
Please select one IP address:
1) 10.193.44.13/24
b) Back to previous menu

Please select one IP address: [1-1, b, q, ?] (1)
Enter the UDP port for the first private heartbeat link on vcslin03: [b, q, ?] (50000)

Enter the NIC for the second private heartbeat link on vcslin03: [b, q, ?] (eth2)
Some configured IP addresses have been found on the NIC eth2 in vcslin03, Do you want
to choose one for the second private heartbeat link? [y, n, q, ?] (y)
Please select one IP address:
1) 10.193.44.14/24
b) Back to previous menu

Please select one IP address: [1-1, b, q, ?] (1)
CPI WARNING V-9-40-3255 Veritas recommends that different heartbeat links in one system
use IP addresses that are in different subnets.
Are you sure you want to continue? [y, n, q, ?] (y)
Enter the UDP port for the second private heartbeat link on vcslin03: [b, q, ?] (50001)

Do you want to configure an additional low-priority heartbeat link? [y, n, q, b, ?] (n)

Are you using the same NICs for private heartbeat links on all systems? [y, n, q, b, ?] (y)
Some configured IP addresses have been found on the NIC eth1 in vcslin04, Do you want
to choose one for the first private heartbeat link? [y, n, q, ?] (y)
Please select one IP address:
1) 10.193.54.13/24
b) Back to previous menu

Please select one IP address: [1-1, b, q, ?] (1)
CPI WARNING V-9-40-3256 Veritas recommends that the two IP addresses which constitute
one heartbeat link are in the same subnet.
Are you sure you want to continue? [y, n, q, ?] (y)
The UDP Port for this link: 50000

Some configured IP addresses have been found on the NIC eth2 in vcslin04, Do you want
to choose one for the second private heartbeat link? [y, n, q, ?] (y)
Please select one IP address:
1) 10.193.54.14/24
b) Back to previous menu

... 次ページに続く ...

```

```

Please select one IP address: [1-1, b, q, ?] (1)
CPI WARNING V-9-40-3256 Veritas recommends that the two IP addresses which constitute
one heartbeat link are in the same subnet.
Are you sure you want to continue? [y, n, q, ?] (y)
The UDP Port for this link: 50001

Checking media speed for eth1 on vcslin03 ..... 50000Mb/s
Checking media speed for eth2 on vcslin03 ..... 50000Mb/s
Checking media speed for eth1 on vcslin04 ..... 50000Mb/s
Checking media speed for eth2 on vcslin04 ..... 50000Mb/s
    
```

クラスターID は、0～65535 からランダムに指定されます。同一ネットワークセグメント上でクラスターIDが重複しない必要があります。チェックが通ったら次に進みます。

```

Enter a unique cluster ID number between 0-65535: [b, q, ?] (56719)

The cluster cannot be configured if the cluster ID 56719 is in use by another cluster.
Installer can perform a check to determine if the cluster ID is duplicate. The check
will take less than a minute to complete.

Would you like to check if the cluster ID is in use by another cluster? [y, n, q] (y) y

Checking cluster ID ..... Done

Duplicated cluster ID detection passed. The cluster ID 56719 can be used for the cluster.

Press [Enter] to continue:
    
```

内容をチェックして、問題なければ先に進んでください。

```

Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

Cluster information verification:

Cluster Name: azurevcs3
Cluster ID Number: 56719

Private Heartbeat NICs for vcslin03:
link1=eth1 over UDP
ip 10.193.44.13 netmask 255.255.255.0 port 50000
link2=eth2 over UDP
ip 10.193.44.14 netmask 255.255.255.0 port 50001

Private Heartbeat NICs for vcslin04:
link1=eth1 over UDP
ip 10.193.54.13 netmask 255.255.255.0 port 50000
link2=eth2 over UDP
ip 10.193.54.14 netmask 255.255.255.0 port 50001

LLT tagging:

When configuring LLT over UDP or TCP, tags are used in LLT configuration file with set-addr option.
Due to this, LLT configuration file may use local NIC identifiers referring to remote IP addresses.

Is this information correct? [y, n, q, ?] (y) y
    
```

本書では、インストール時にクラスターの仮想 IP を設定する必要はないので “n” を答えて先に進んでください。

```

Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

The following data is required to configure the Virtual IP of the Cluster:

    A public NIC used by each system in the cluster
    A Virtual IP address and netmask

Do you want to configure the Virtual IP? [y, n, q, ?] (n) n

```

本書では、VCS をセキュアモードで設定しないので、“n” に続いて “y” を指定し、先に進んでください。

```

Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

We recommend that you run Cluster Server in secure mode.

Running VCS in Secure Mode guarantees that all inter-system communication is encrypted, and users are verified with security credentials.

When running VCS in Secure Mode, NIS and system usernames and passwords are used to verify identity. VCS usernames and passwords are no longer utilized when a cluster is running in Secure Mode.

Would you like to configure the VCS cluster in secure mode? [y, n, q, ?] (y) n

CPI WARNING V-9-40-3394 We recommend that you install the cluster in secure mode. This ensures that communication between cluster components is encrypted and cluster information is visible to specified users only.

Are you sure that you want to proceed with non-secure installation? [y, n, q] (n) y

```

VCS のオペレーションを行うためのユーザーとパスワードを設定します。この ID は、複数の VCS 同士で連携したり、OS に root 以外でログインしたユーザーで VCS のオペレーションを行う際に必要です。本書では、デフォルトのユーザー：admin のみを設定しています。

```

Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

The following information is required to add VCS users:

    A user name
    A password for the user
    User privileges (Administrator, Operator, or Guest)

Do you wish to accept the default cluster credentials of 'admin/password'? [y, n, q] (y) y

Do you want to add another user to the cluster? [y, n, q] (n) n

```

内容を確認し、“y” を指定して先に進んでください。

```
Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

VCS User verification:

  User: admin   Privilege: Administrators

  Passwords are not displayed

Is this information correct? [y,n,q] (y) y
```

本書では、SMTP 通知は行わないので “n” を指定して先に進んでください。

```
Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

The following information is required to configure SMTP notification:

  The domain-based hostname of the SMTP server
  The email address of each SMTP recipient
  A minimum severity level of messages to send to each recipient

Do you want to configure SMTP notification? [y,n,q,?] (n) n
```

本書では、SNMP 通知は行わないので “n” を指定して先に進んでください。

```
Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

The following information is required to configure SNMP notification:

  System names of SNMP consoles to receive VCS trap messages
  SNMP trap daemon port numbers for each console
  A minimum severity level of messages to send to each console

Do you want to configure SNMP notification? [y,n,q,?] (n) n
```

本書では、グローバルクラスターの設定は行わないので “n” を指定して先に進んでください。

```
Veritas InfoScale Enterprise 7.4.1 Install Program
vcslin03 vcslin04

The following data is required to configure the Global Cluster Option:

  A public NIC used by each system in the cluster
  A Virtual IP address and netmask

Do you want to configure the Global Cluster Option? [y,n,q,?] (n) n
```

インストールを続行するには、関連プロセスやドライバーの再起動が必要です。“y” を指定して先に進んでください。

```
All InfoScale Enterprise processes that are currently running must be stopped
Do you want to stop InfoScale Enterprise processes now? [y, n, q, ?] (y) y
```

途中略

```
Storage Foundation and High Availability Startup completed successfully

The updates to VRTSaslapm package are released via the SORT web page: https://sort.veritas.com/asl. To make
sure you have the latest version of VRTSaslapm (for up to date ASLs and APMs), download and install the latest
package from the SORT web page.

After configuring the cluster, you can configure application monitoring using Veritas InfoScale Operation
Manager.
To launch the High Availability Configuration Wizard:
1. Log on to the Veritas InfoScale Operation Manager Management Server domain.
2. In the Veritas InfoScale Operation Manager home page, click the Availability icon from the list of
perspectives.
3. Locate the cluster and then right-click on the cluster or on one of the systems under the cluster.
4. Click Configure Application.

Installation procedures and diagnostic information are saved in the log files under directory
/var/tmp/installer-2019053110370EF. This information helps us identify and resolve failed operations performed
by the installer. Would you like to send the information to us to help improve installation in the future?
[y, n, q, ?] (y) n

Checking online updates for Veritas InfoScale Enterprise 7.4.1

    No updates available for Veritas InfoScale Enterprise 7.4.1

    Visit https://sort.veritas.com for more information.

ssh is configured in password-less mode on vcslin04

Do you want to cleanup the communication for the systems vcslin04? [y, n, q] (n) y

    Cleanup the communication for the system vcslin04 ..... Done

installer log files, summary file, and response file are saved at:

    /opt/VRTS/install/logs/installer-202004030338QNM

Would you like to view the summary file? [y, n, q] (n)
#
```

これで、InfoScale のインストールは終了です。インストール時に問題が発生した場合は、インストーラーの最後で表示された/opt/VRTS/install/logs 配下のログを収集の上ベリタスに連絡してください。

4. VxVM と VxFS の設定

ディスクの確認とイニシャライズ

まず、VxVM から認識されているディスクの確認とイニシャライズを行います。

オペレーションの準備

オペレーションを始める前に、両方のノードで PATH と MANPATH の設定を行ってください。

```
# PATH=$PATH:/opt/VRTS/bin export PATH
# MANPATH=$MANPATH/opt/VRTS/man export MANPATH
```

ディスクのイニシャライズと export

1 号機で認識できているディスクを確認してください。OS のディスク以外に、1 本のディスクが見えています。データ用にこのディスクを使用します。

```
# vxdisk list
DEVICE          TYPE          DISK          GROUP          STATUS
xvda            auto:none    -             -             online invalid
01_xen-vd0_1   auto:none    -             -             online invalid
```

1 号機で認識できているディスクを確認してください。vxdisk list の出力結果によると、3 本のディスク (sda, 10-193-24-7_disk_0, 10-193-24-7_disk_2) が見えています。vxdisk path の出力と見比べる事で、10-193-27-4_disk_2 の実態は sdb、10-193-24-7_disk_0 の実態は sdc であることが分かりました。また、df の出力結果から sda と sdb は OS の領域として使用されている事がわかりましたので、データ用に使用できるディスクは sdc すなわち 10-193-24-7_disk_0 であると特定できます。

```
# vxdisk list
DEVICE          TYPE          DISK          GROUP          STATUS
sda            auto:none    -             -             online invalid
10-193-24-7_disk_0 auto:none    -             -             online invalid
10-193-24-7_disk_2 auto:none    -             -             online invalid
# vxdisk path
SUBPATH          DANAME          DMNAME          GROUP          STATE
sda              sda             -              -             ENABLED
sdc              10-193-24-7_disk_0 -              -             ENABLED
sdb              10-193-24-7_disk_2 -              -             ENABLED
# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2       32G   8.3G  24G   27% /
devtmpfs        3.9G   0    3.9G   0% /dev
tmpfs           3.9G   0    3.9G   0% /dev/shm
tmpfs           3.9G   8.4M  3.9G   1% /run
tmpfs           3.9G   0    3.9G   0% /sys/fs/cgroup
/dev/sda1       497M  103M  394M  21% /boot
/dev/sdb1       40G   2.1G  36G   6% /mnt/resource
tmpfs           797M   0    797M   0% /run/user/0
```

前述のディスクをインシャライズし、その後で export してください。

```
# vxdisksetup -i 10-193-24-7_disk_0
# vxdisk export 10-193-24-7_disk_0
```

2 号機でも、同じようにディスクのインシャライズと export を行ってください。

```
# vxdisk list
DEVICE          TYPE          DISK          GROUP          STATUS
sda             auto:none    -             -             online invalid
10-193-34-6_disk_0 auto:none    -             -             online invalid
10-193-34-6_disk_2 auto:none    -             -             online invalid
# vxdisk path
SUBPATH          DANAME          DMNAME          GROUP          STATE
sda             sda             -             -             ENABLED
sdc             10-193-34-6_disk_0 -             -             ENABLED
sdb             10-193-34-6_disk_2 -             -             ENABLED
# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda2       32G   5.1G  27G  16% /
devtmpfs        3.9G   0    3.9G   0% /dev
tmpfs           3.9G   0    3.9G   0% /dev/shm
tmpfs           3.9G   8.5M  3.9G   1% /run
tmpfs           3.9G   0    3.9G   0% /sys/fs/cgroup
/dev/sda1       497M  103M  394M  21% /boot
/dev/sdb1       40G   2.1G  36G   6% /mnt/resource
tmpfs           797M   0    797M   0% /run/user/0
# vxdisksetup -i 10-193-34-6_disk_0
# vxdisk export 10-193-34-6_disk_0
```

1 号機及び 2 号機で、vxdctl enable でディスクの再走査を行った後に、再度 vxdisk list を実行すると、他方のノードのディスクが見えるようになります。下記は、1 号機の出力例です。

```
# vxdisk list
DEVICE          TYPE          DISK          GROUP          STATUS
sda             auto:none    -             -             online invalid
10-193-24-7_disk_0 auto:cdsdisk -             -             online exported
10-193-24-7_disk_2 auto:none    -             -             online invalid
10-193-34-6_disk_0 auto:cdsdisk -             -             online remote
```

VxVM のディスクグループの作成

次に、ディスクグループを作成します。ディスクグループは、Linux の LVM で言う所の Volume Group に相当し、VxVM でボリューム（仮想デバイス）を作成して使用する場合は、ボリュームは必ずいずれかのディスクグループに所属している必要があります。本書のクラスターは、FSS 構成であるため、双方のノードの内蔵ディスクをお互いにシェアした fss ディスクグループを作成します。

ディスクグループの作成

1 号機で vxdg コマンドを用いて、以下の例に従って fssdg という名前のディスクグループを作成します。ディスクとして 10-193-24-7_disk_0 と 10-193-34-6_disk_0 の 2 本を使用しています。

```
vxdg -s -o fss init fssdg 10-193-24-7_disk_0 10-193-34-6_disk_0
```

ディスクグループの作成に成功すると、`vxdisk list` コマンドの出力は以下のようになり、従来空欄だった“GROUP”の列にディスクグループ名である“fssdg”が表示されます。

```
# vxdisk list
DEVICE          TYPE          DISK          GROUP          STATUS
sda             auto:none     -             -             online invalid
10-193-24-7_disk_0 auto:cdsdisk  10-193-24-7_disk_0 fssdg         online exported shared
10-193-24-7_disk_2 auto:none     -             -             online invalid
10-193-34-6_disk_0 auto:cdsdisk  10-193-34-6_disk_0 fssdg         online shared remote
```

VxVM のボリュームと VxFS のファイルシステムの作成

次に、ボリュームを作成します。ボリュームは、Linux の LVM で言う所の Volume に相当します。ボリュームを作製したら、そこにファイルシステムを構築します。この作業は 1 号機のみで行います。

まず、VxVM のボリュームを作製します。下記例では、“fssdg”というディスクグループ内の“10-193-24-7_disk_0”と“10-193-34-6_disk_0”の 2 本のディスクを用いて、“vol01”という名前の容量 3Gbyte のミラーボリュームを作成しています。

```
# vxassist -g fssdg make vol01 3g layout=mirror 10-193-24-7_disk_0 10-193-34-6_disk_0
```

作製したボリュームは、`vxprint` コマンドで確認できます。

```
# vxprint
Disk group: fssdg

TY NAME          ASSOC          KSTATE  LENGTH  PLOFFS  STATE  TUTILO  PUTILO
dg fssdg         fssdg         -       -       -       -       -       -
dm 10-193-24-7_disk_0 10-193-24-7_disk_0 - 10411776 - - - -
dm 10-193-34-6_disk_0 10-193-34-6_disk_0 - 10411776 - REMOTE - -
v vol01          fsgen         ENABLED 6291456 - ACTIVE - -
pl vol01-01      vol01         ENABLED 6291456 - ACTIVE - -
sd 10-193-24-7_disk_0-01 vol01-01 ENABLED 6291456 0 - - -
pl vol01-02      vol01         ENABLED 6291456 - ACTIVE - -
sd 10-193-34-6_disk_0-01 vol01-02 ENABLED 6291456 0 - - -
dc vol01_dco     vol01         - - - - - -
v vol01_dcl      gen           ENABLED 67840 - ACTIVE - -
pl vol01_dcl-01  vol01_dcl    ENABLED 67840 - ACTIVE - -
sd 10-193-24-7_disk_0-02 vol01_dcl-01 ENABLED 67840 0 - - -
pl vol01_dcl-02  vol01_dcl    ENABLED 67840 - ACTIVE - -
sd 10-193-34-6_disk_0-02 vol01_dcl-02 ENABLED 67840 0 - - -
```

作製されたボリューム上にファイルシステムを作製します。指定するデバイスパス名は /dev/vx/rdsk までが固定で、その後にディスクグループ名（下記例では fssdg）、最後にボリューム名（下記例では vol01）、というフォーマットです。

```
# mkfs -t vxfs /dev/vx/rdsk/fssdg/vol01
version 15 layout
6291456 sectors, 3145728 blocks of size 1024, log size 16384 blocks
rcq size 1024 blocks
largefiles supported
maxlink supported
WORM not supported
```

既存のサービスグループ:cvm への設定追加

本構成では、インストール時に SFCFSHA を指定しているため、自動的に cvm というサービスグループが作成されています。作成したファイルシステムを mount して使用するためには、このサービスグループにリソースを追加する必要があります。この作業は、特に断りがない限り 1 号機のみで行います。

まず、クラスターの構成を編集可能にするモード設定を行ってください。

```
# haconf -makerw
```

下記では、使用するディスクグループ:fssdg とボリューム vol01 を管理・監視するリソース:voldg を追加しています。最後に、このリソースの online の条件が、既に構成済みのリソースである cvm_clus の online である事を設定しています。

```
# hares -add voldg CVMVolDg cvm
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
# hares -modify voldg CVMDiskGroup fssdg
# hares -modify voldg CVMActivation sw
# hares -modify voldg CVMVolume vol01
# hares -modify voldg Critical 0
# hares -modify voldg Enabled 1
# hares -link voldg cvm_clus
```

次に、ファイルシステムを mount するリソース : mount を作製します。このリソースの online の条件が、既に構成済みのリソースである voldg の online である事を設定しています。尚、このリソースを作製する際に、1 号機と 2 号機で mount point : /data を作製しておいてください。

```
# hares -add mount CFMount cvm
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
# hares -modify mount MountPoint /fss_aws
# hares -modify mount BlockDevice /dev/vx/dsk/fssdg/vol01
# hares -modify mount Enabled 1
# hares -link mount voldg
#
# mkdir /data
```

サービスグループ: cvm を 1 号機と 2 号機で online にします。

```
# hagr -online cvm -any
VCS NOTICE V-16-1-50735 Attempting to online group on system vcslin03
VCS NOTICE V-16-1-50735 Attempting to online group on system vcslin04
```

Online になったかどうかは、hastatus と df で確認してください。

```
# hastatus -sum

-- SYSTEM STATE
-- System          State          Frozen
A vcslin03         RUNNING        0
A vcslin04         RUNNING        0

-- GROUP STATE
-- Group           System          Probed    AutoDisabled  State
B cvm              vcslin03        Y         N              ONLINE
B cvm              vcslin04        Y         N              ONLINE

# df -h
Filesystem          Size  Used Avail Use% Mounted on
/dev/sda2           32G   8.3G  24G  27% /
devtmpfs            3.9G   0    3.9G   0% /dev
tmpfs               3.9G   0    3.9G   0% /dev/shm
tmpfs               3.9G   8.4M  3.9G   1% /run
tmpfs               3.9G   0    3.9G   0% /sys/fs/cgroup
/dev/sda1           497M  103M  394M  21% /boot
/dev/sdb1           40G   2.1G  36G   6% /mnt/resource
tmpfs               797M   0    797M   0% /run/user/0
/dev/vx/dsk/fssdg/vol01 3.0G   36M  2.8G   2% /data
```

5. Python SDK for Azure のインストールと Azure 関連の各種設定

ここでは、InfoScale と Azure が連携するための様々な設定を行います。まず、InfoScale から Azure にアクセスするために Python SKD for Azure をクラスターの全ノードにインストールします。加えて、Azure を IAM 経由でコントロールするためのサービスプリンシパルを作成して必要な権限を付与します。また、InfoScale のパラメーター設定に必要なテナント ID 等の情報を確認します。

Python SKD for Azure のインストール

両方のノードで Python SKD for Azure をインストールします。インストールが必要なモジュールは合計 7 個あり、いずれも InfoScale の専用コマンドを用いて行います。

azure.mgmt.compute をインストールします。

```
# /opt/VRTSpython/bin/pip3.5 install azure.mgmt.compute==3.0.0
Collecting azure.mgmt.compute==3.0.0
  (中略)
Successfully installed PyJWT-1.7.1 adal-1.2.2 azure-common-1.1.25 azure-mgmt-nspkg-3.0.2 azure-nspkg-3.0.2
azure.mgmt.compute-certifi-2019.11.28 cffi-1.14.0 chardet-3.0.4 cryptography-2.8 idna-2.9 importlib-
metadata-1.5.0 isodate-0.6.0 jeepney-0.4.3 keyring-20.0.1 msrest-0.6.11 msrestazure-0.4.34 oauthlib-3.1.0
pyparser-2.20 python-dateutil-2.8.1 requests-2.23.0 requests-oauthlib-1.3.0 secretstorage-3.1.2 six-
1.14.0 urllib3-1.25.8 zipp-1.2.0
```

azure-mgmt-network をインストールします。

```
# /opt/VRTSpython/bin/pip3.5 install azure-mgmt-network==1.7.1
  (中略)
Successfully installed azure-mgmt-network-1.7.1
```

azure.mgmt.dns をインストールします。本書の構成では DNS の切り替えは行わないので省略も可能ですが、要件が変更になる場合に備えてインストールしておくことを推奨します。

```
# /opt/VRTSpython/bin/pip3.5 install azure.mgmt.dns==1.0.0
  (中略)
Successfully installed azure-common-1.1.4 azure.mgmt.dns msrest-0.4.29
```

azure.storage をインストールします。本書の構成ではブロックストレージの切り替えは行わないので省略も可能ですが、要件が変更になる場合に備えてインストールしておくことを推奨します。

```
# /opt/VRTSpython/bin/pip3.5 install azure.storage==0.33.0
  (中略)
Successfully installed azure.storage
```

msrest をインストールします。

```
# /opt/VRTSpython/bin/pip3.5 install --force-reinstall msrest==0.4.29
(中略)
Successfully installed certifi-2019.11.28 chardet-3.0.4 idna-2.9 isodate-0.6.0 msrest-0.4.29 oauthlib-3.1.0 requests-2.23.0 requests-oauthlib-1.3.0 six-1.14.0 urllib3-1.25.8
```

msrestazure をインストールします。

```
# /opt/VRTSpython/bin/pip3.5 install --force-reinstall msrestazure==0.4.19
(中略)
Successfully installed PyJWT-1.7.1 adal-0.4.7 certifi-2019.11.28 cffi-1.14.0 chardet-3.0.4 cryptography-2.8 idna-2.9 importlib-metadata-1.5.0 isodate-0.6.0 jeepney-0.4.3 keyring-20.0.1 msrest-0.4.29 msrestazure-0.4.19 oauthlib-3.1.0 pycparser-2.20 python-dateutil-2.8.1 requests-2.23.0 requests-oauthlib-1.3.0 secretstorage-3.1.2 six-1.14.0 urllib3-1.25.8 zipp-1.2.0
```

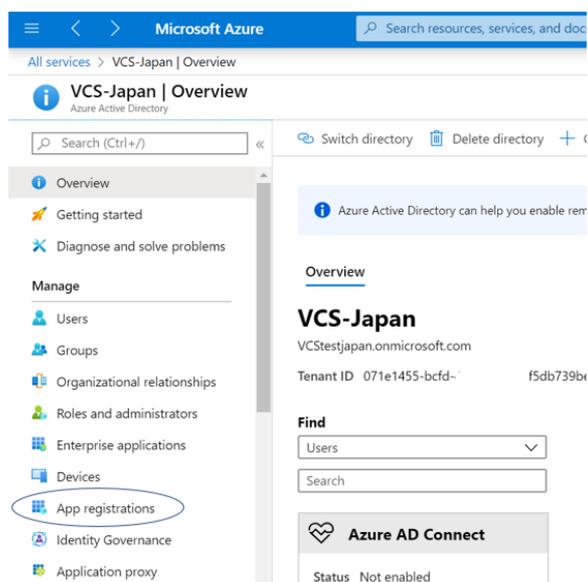
azure-common をインストールします。

```
# /opt/VRTSpython/bin/pip3.5 install --force-reinstall azure-common==1.1.4
(中略)
Successfully installed azure-common-1.1.4 azure-nspkg-3.0.2
```

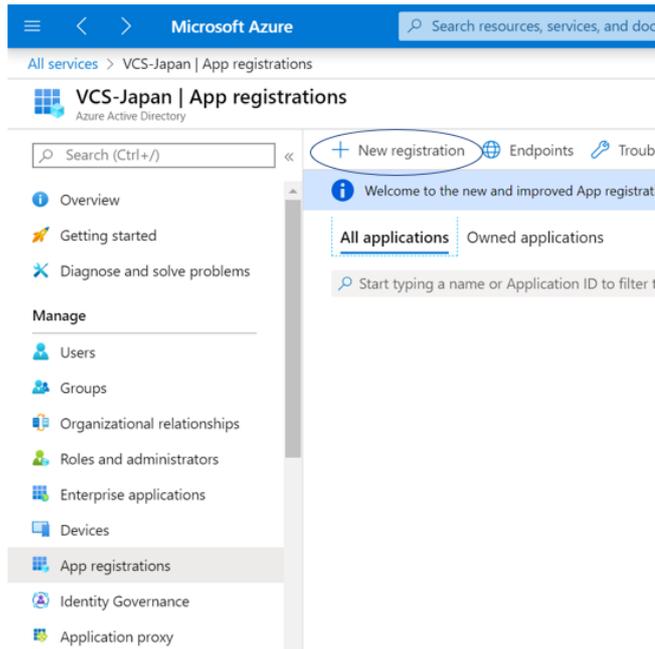
サービスプリンシパルの作成と権限付与

ここでは、InfoScale から IAM 経由で Azure のルーティングテーブルの更新等を行うためのサービスプリンシパルを作成し、必要な権限を付与します。この作業は、Azure Portal から行います。

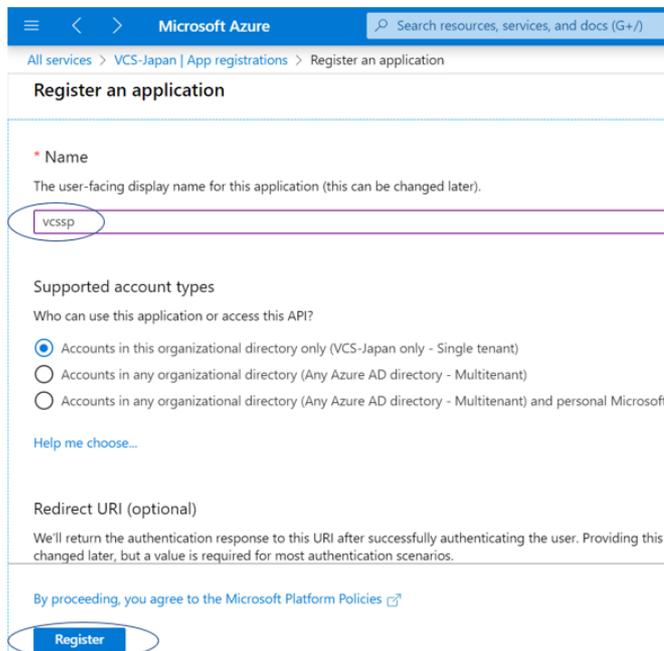
まず、Azure Portal のメニューから “App registration” をクリックしてください。



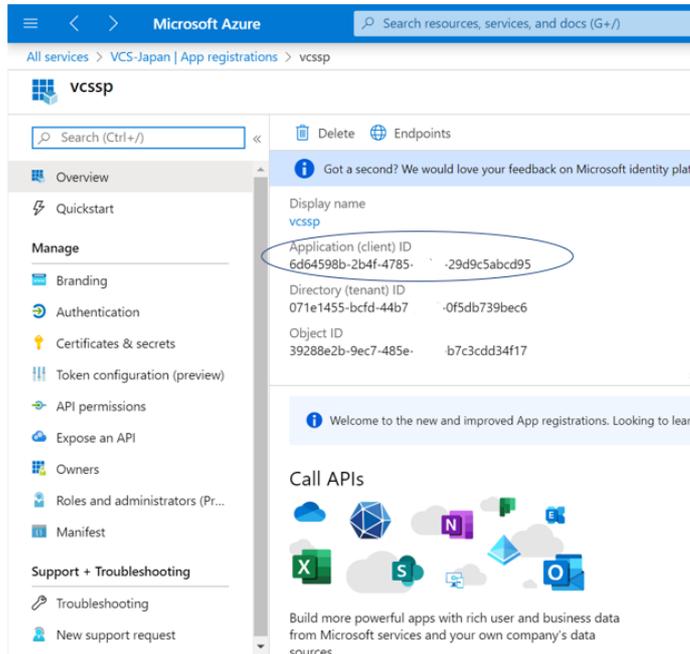
メニューが出ますので、“New Registration” を選択してください。



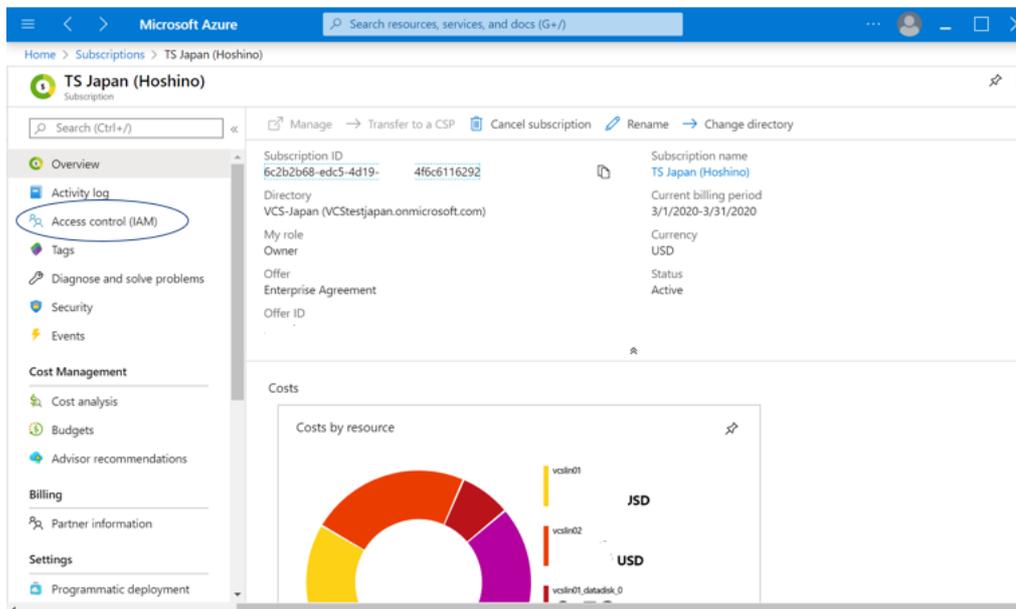
名前を入力し（この例では“vcssp”）Account Type は“Single Tenant”にして、“Register” をクリックし、登録してください。



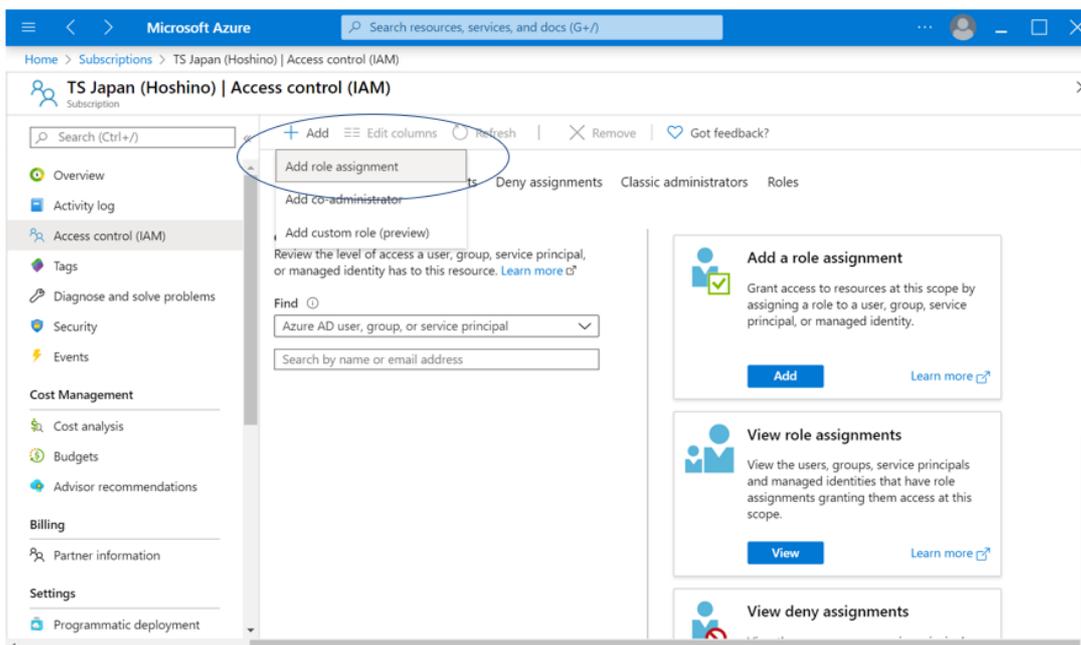
登録が完了したら、Application ID（Client ID）を控えておいてください。下記例では、控えるべきIDは "6d64598b-2b4f-4785-xxxx-29d9c5abcd95" です。



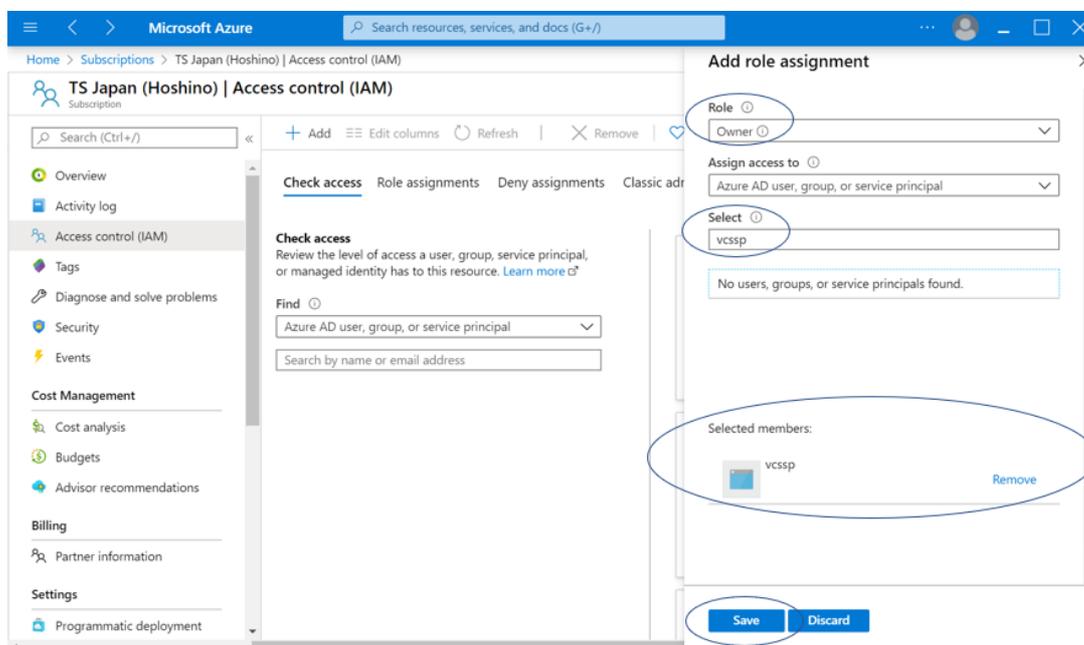
次に、必要な権限を付与します。まず、Azure Portal のメニューから "Access Control (IAM)" をクリックしてください。



メニューが出ますので、“Add role assignment” を選択してください。



先に作成したサービスプリンシパル：vcssp を選択し、必要な権限（Role）を付与してください。下記の例では分かりやすくするために“Owner”つまり、フルコントロールを可能にしていますが、本来であれば必要な作業（ルーティングテーブルの更新）のみが可能になる権限をプルダウンから選択します。適切な設定を選んだら“Save”をクリックしてください。

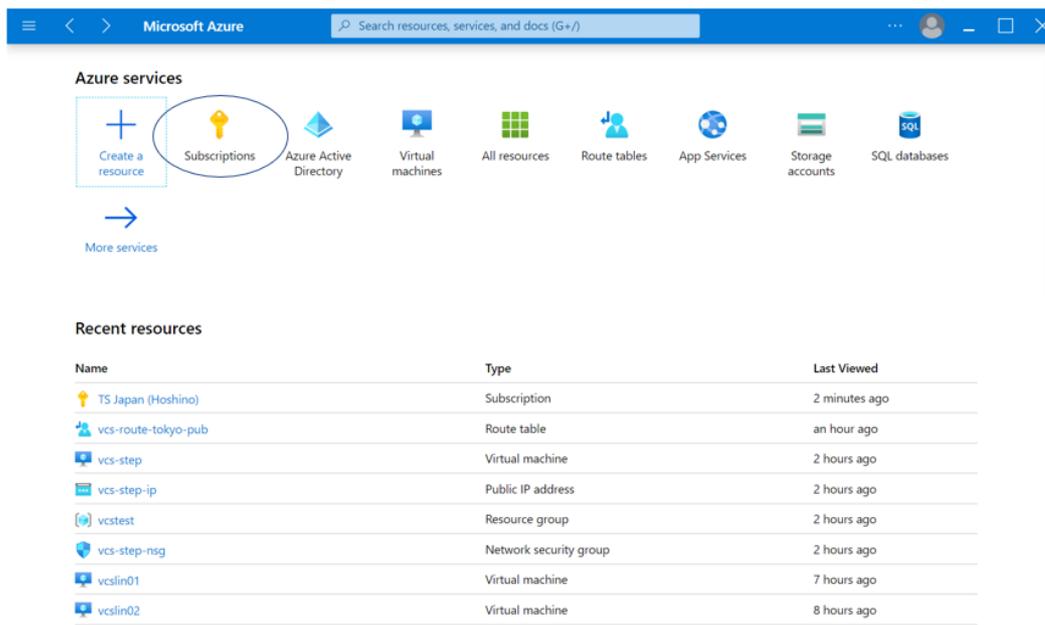


以上で、サービスプリンシパルの作成と権限付与は完了です。

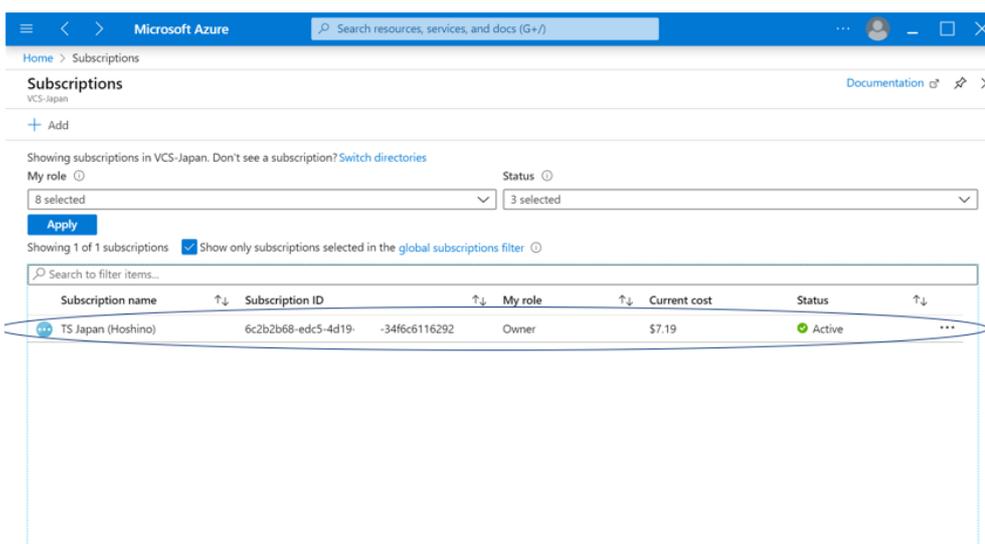
InfoScale のパラメーター設定に必要な情報の確認

InfoScale を Azure と連携されるための基本情報として、サブスクリプション ID、TenantID、SecretKey、ClientID の 4 つの情報が必要で、これらは AzureAuth Agent のパラメーターとして登録されます。ClientID に関しては、先の工程で確認済みですので、ここではサブスクリプション ID、TenantID、SecretKey の 3 つの情報を確認する方法を説明します。

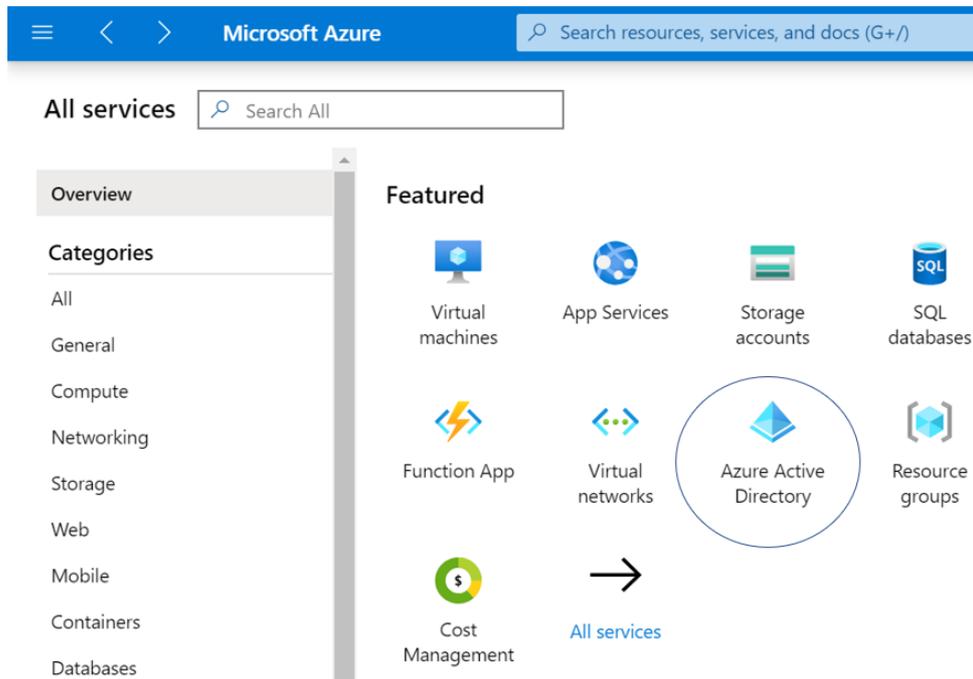
まず、サブスクリプション ID を確認します。Azure Portal のメニューから “Subscription” をクリックしてください。



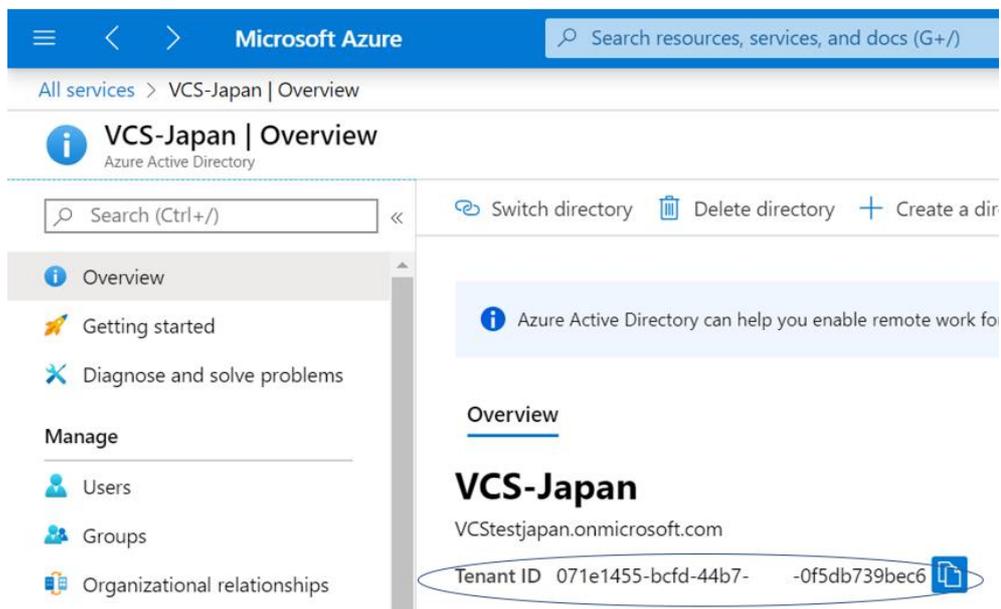
下記例では、控えるべき ID は “6c2b2b68-edc5-4d19-xxxx-34f6c6116292” です。



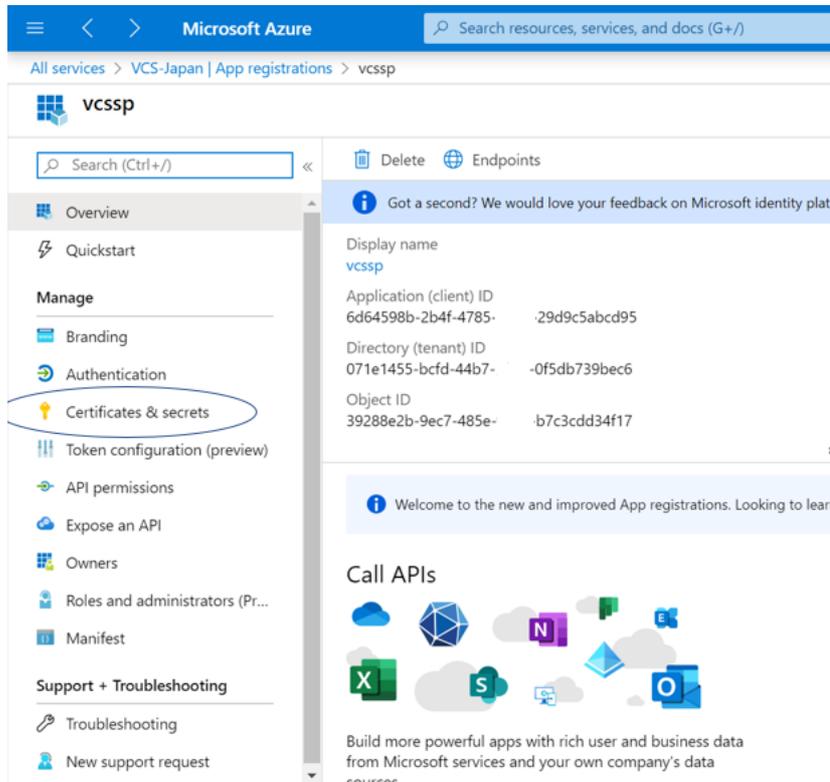
次に、TenantIDを確認します。Azure Portal のメニューから “Azure Active Directory” をクリックしてください。



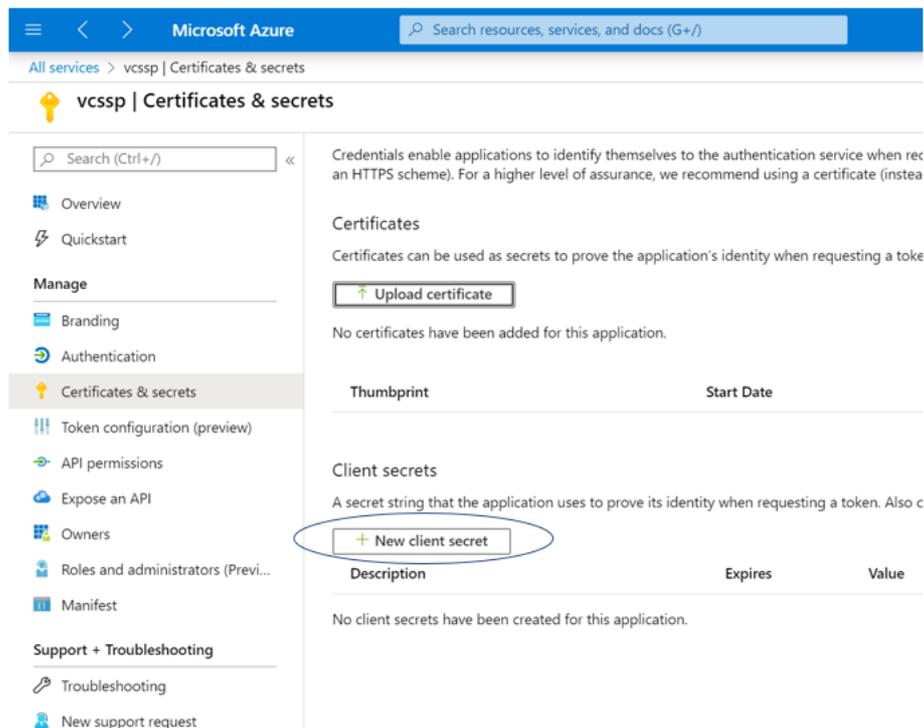
下記例では、控えるべき ID は “071e1455-bcfd-44b7-xxxx-0f5db739bec6” です。



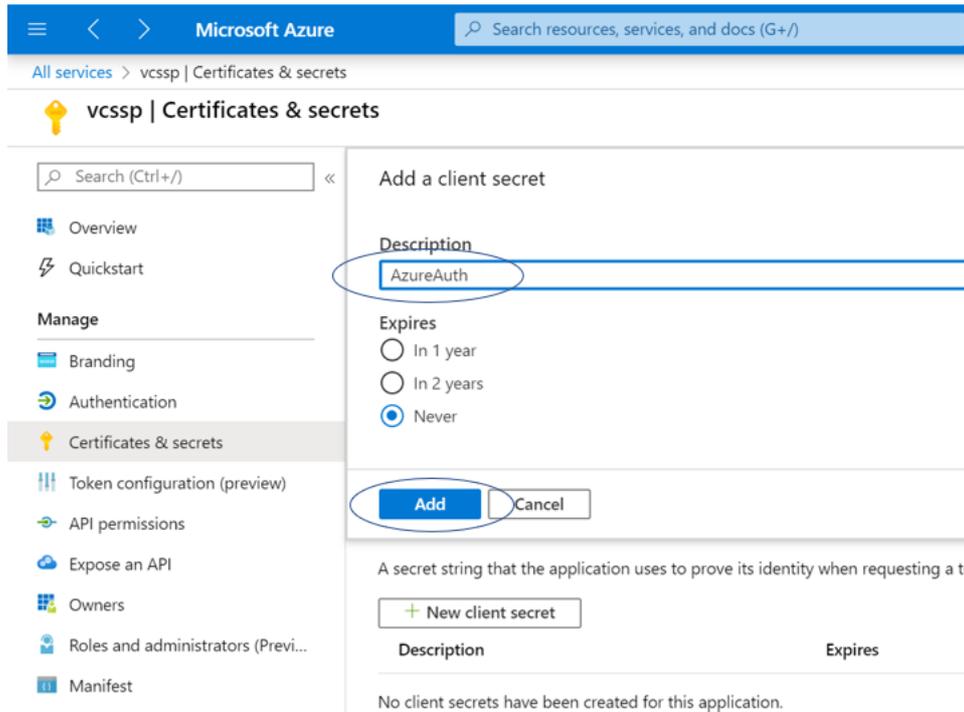
最後に、SecretKey を作成します。Azure Portal のメニューから、26 ページで作成したサービスプリンシパル：vcssp を表示し、“Certifications & secrets” をクリックしてください。



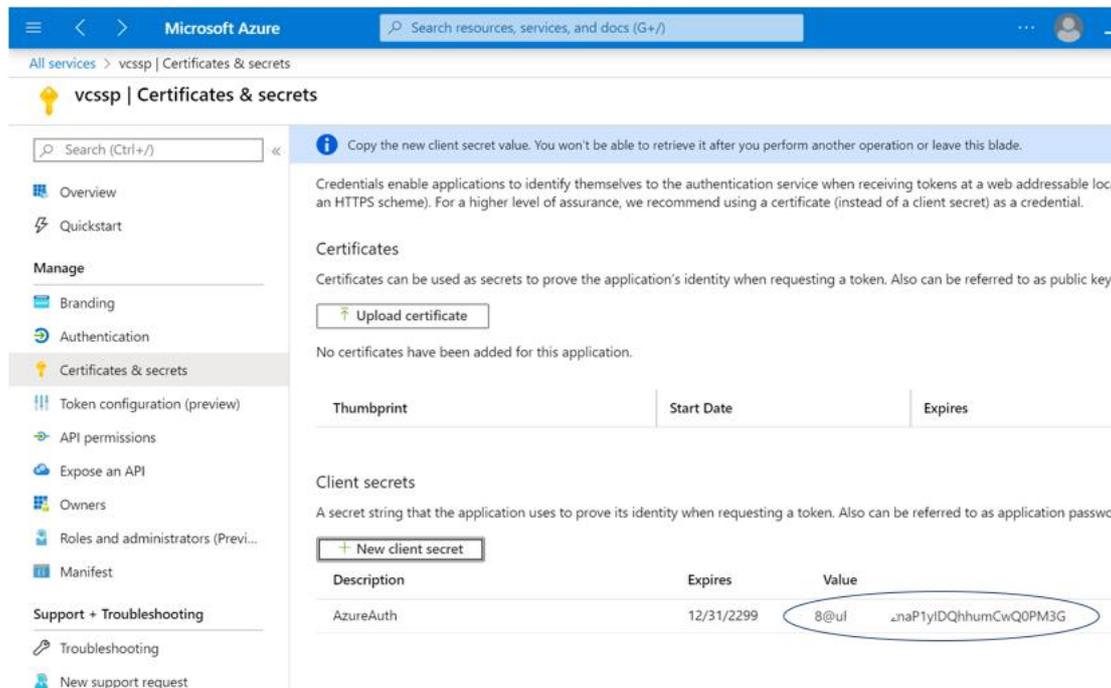
“New client secret” をクリックしてください。



分かりやすい Description を指定し（下記例では AzureAuth）、期限を選び（下記例では無期限：Never）、"Add" をクリックして SecretKey を作成してください。



下記例では、控えるべき SecretKey は "8@uxxxxxnaP1yIDQhhumCwQ0PM3G" です。



6. クラスターの設定

ここでは、Azure 上の RHEL 上で、アプリケーション及びその稼働に必要な個々のコンポーネントが、InfoScale によって適切に監視/リカバリーされるための設定を行います。

アプリケーションを管理するサービスグループのセットアップ

ここでは、アプリケーションを管理するサービスグループを作製します。このサービスグループは、稼働系でアプリケーションを立ち上げ、クライアントからの通信経路を確立します。この作業も、どちらか一方のノードで行います。

サービスグループ：application を作製します。このサービスグループは、稼働待機型になります。従って、クラスターが立ち上がるとデフォルトで 1 号機のみで自動的にオンラインになります。また、このサービスグループが動作する為には、同じノード内で、サービスグループ：cvm が動作している必要がある為、サービスグループ：cvm に「online local firm」で依存させます。

```
# hagr -add application
VCS NOTICE V-16-1-10136 Group added; populating SystemList and setting the Parallel attribute
recommended before adding resources
# hagr -modify application Parallel 0
# hagr -modify application SystemList vcslin03 0 vcslin04 1
# hagr -modify application AutoStartList vcslin03 vcslin04
# hagr -link application cvm online local firm
```

apache のインストールと必要な設定

本書では、クラスタリング対象のアプリケーションとして apache を用いています。ここでは、apache のインストールとクラスタリングの為に必要な apache の微調整を行います。

まず、以下の例に従って両方のノードで apache をインストールしてください。

```
# yum install httpd
Loaded plugins: amazon-id, product-id, rhui-lb, search-disabled-repos, subscription-manager
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.6-90.el7 will be installed
--> Processing Dependency: httpd-tools = 2.4.6-90.el7 for package: httpd-2.4.6-90.el7.x86_64
--> Running transaction check
--> Package httpd-tools.x86_64 0:2.4.6-90.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch          Version        Repository      Size
=====
Installing:
httpd                  x86_64        2.4.6-90.el7   rhel-7-server-rpms 1.2 M
Installing for dependencies:
httpd-tools            x86_64        2.4.6-90.el7   rhel-7-server-rpms 91 k

以下略
:
```

apache のコンテンツは、デフォルトで /var/www 配下に格納されます。しかし、クラスタリングする場合は、クラスター間で切り替え可能な共有ストレージ領域内に格納されべきです。まず、1 号機で /var/www ディレクトリを 仮想共有ストレージ領域である /data 配下に移動し、/var/www にアクセスすると /data/www を参照するようにシンボリックリンクを設定します。その後、apache を InfoScale からのみコントロールするために、OS による自動起動が行われないようにします。2 号機では、シンボリックリンクの作成のみを行ってください。

```
# systemctl disable httpd.service
# mv /var/www /data
# cd /var
# ln -s /data/www ./www
# ls -l | grep www
lrwxrwxrwx. 1 root root    9 Mar 25 08:16 www -> /data/www
```

本構成は FSS によるクラスターファイルシステムを使用しているため、上記の 1 号機の作業によって 2 号機の /data 配下でも www が見えるようになります。従って、2 号機では、/var/www の削除とシンボリックリンクの作成を行ってください。

```
# systemctl disable httpd.service
# cd /var
# rm -rf ./www
# ln -s /data/www ./www
# ls -l | grep www
lrwxrwxrwx. 1 root root    9 Apr  3 04:37 www -> /data/www
```

インストールと apache 関連の微調整の後に、以下の例に従って apache の動作確認を行います。まず、"systemctl start httpd.service" で apache が立ち上がり、プロセス ID が /etc/httpd/run/httpd.pid に格納される事を確認します。次に、"systemctl stop httpd.service" で apache が停止し、プロセス ID ファイル：/etc/httpd/run/httpd.pid が削除される事を確認します。

```
# ps -ef | grep httpd
root    29275 17759  0 05:58 pts/0    00:00:00 grep --color=auto httpd

# systemctl start httpd.service

# ps -ef | grep httpd
root    29484      1  0 05:59 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache  29485 29484  0 05:59 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache  29486 29484  0 05:59 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache  29487 29484  0 05:59 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache  29488 29484  0 05:59 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache  29489 29484  0 05:59 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
root    29504 17759  0 05:59 pts/0    00:00:00 grep --color=auto httpd
# cat /etc/httpd/run/httpd.pid
29484

# systemctl stop httpd.service

# ps -ef | grep httpd
root    29767 17759  0 06:00 pts/0    00:00:00 grep --color=auto httpd

# cat /etc/httpd/run/httpd.pid
cat: /etc/httpd/run/httpd.pid: No such file or directory
```

これで、apache をクラスタリングする為の準備が整いました。

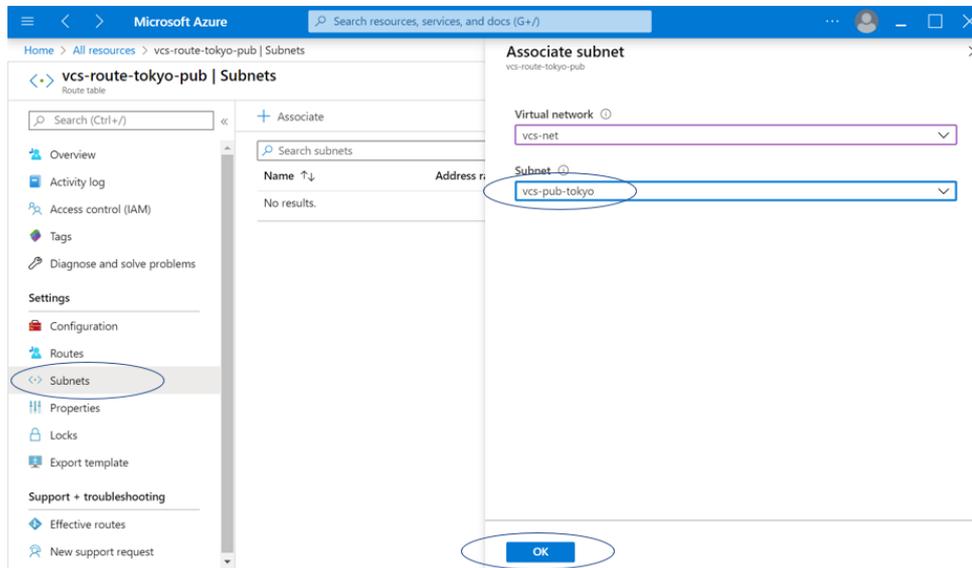
異なるサブネット間でネットワーク経路を切り替える為の設定

次に、異なるサブネット間でネットワーク経路を切り替える為の設定を行います。サブネットが異なる場合、同じ PrivateIP を切り替えることはできません。PublicIP を切り替えることはできますが、通常、クラスタリングを行うような重要なシステムに PublicIP を振る事は一般的ではありません。このような条件下で、クラスタが切り替わったことをクライアントに意識させずに業務を継続するには、特別な手法が必要です。InfoScale は、Azure のルートテーブルを切り替える事によって、この課題を解決します。これから説明する例では、クラスタが配置される Virtual Network 内のクライアントは 10.192.10.20 という IP を用いて、Web サーバーである apache にアクセスを試みます。InfoScale は、クラスタが配置される Virtual Network 内の全ルートテーブルを更新し、10.192.10.20 のターゲットを、クラスタの稼働系ノードの物理 IP にリダイレクトします。クラスタが切り替わると、新たに稼働系になったノードの物理 IP に 10.192.10.20 向けの packets がリダイレクトされるように、ルートテーブルを更新します。また、クラスタの稼働系では、10.192.10.20 が立ち上がっており、クライアントに packets を送信する際の送信元アドレスになります。ここで使用する IP : 10.192.10.20 を OverlayIP とよび、この方式を OverlayIP 方式とよびます。尚、**OverlayIP は必ず Virtual Network に関連付けられた CIDR ブロックに含まれない IP にしてください**。本書の例では、OverlayIP を VPC に関連付けられた CIDR ブロック : 10.193.0.0/16 に含まれないような IP : 10.192.10.20 にしています。

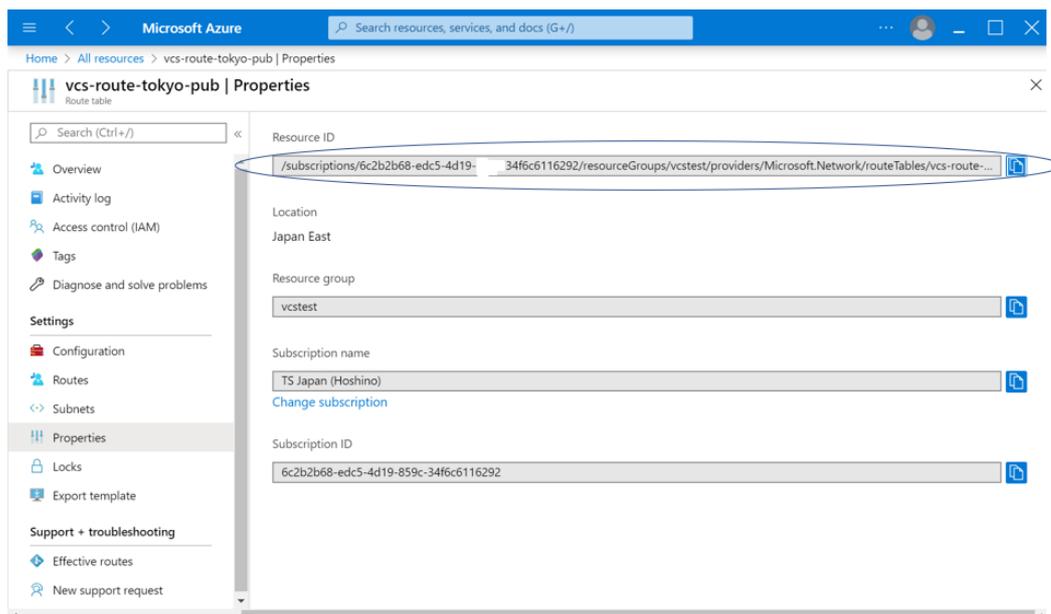
OverlayIP 方式を使用するには、ルーティングテーブルを作成し、クラスタにサービス要求を行うクライアントが配置されるサブネットに関連付ける必要があります。その上で、InfoScale からそのルーティングテーブルを更新する事で、クライアントからのサービス要求が、クラスタの稼働系の物理 IP にリダイレクトされるようになります。まず、ルーティングテーブルを作成します。Azure Portal から Create Routing Table を指定します。下記例では、“vcs-route-tokyo-pub” という名前のルートテーブルを作成しています。

The screenshot shows the 'Create route table' form in the Microsoft Azure portal. The 'Name' field is highlighted with a red oval and contains the text 'vcs-route-tokyo-pub'. Other fields include 'Subscription' (TS Japan (Hoshino)), 'Resource group' (vcstest), and 'Location' ((Asia Pacific) Japan East). The 'Virtual network gateway route propagation' toggle is set to 'Enabled'.

ルートテーブルを作成したら、クラスターにサービス要求を行うクライアントが配置されるサブネットに関連付けます。ルートテーブルを指定し “subnet” をクリックすると関連付けるサブネットを選択できます。下記例では “vcs-tokyo-pub” というサブネットに関連付けています。クラスターにサービス要求を行うクライアントが配置されるサブネットが複数ある場合は、ルートテーブルの作成とサブネットへの関連付けを複数回行ってください。



InfoScale からルートテーブルを更新するためには、パラメーターとしてルートテーブル ID を設定する必要がありますので、ルートテーブル ID を確認しておきます。ルートテーブルを指定し “Properties” をクリックすると ID が確認できます。控えるべき ID は “ /subscriptions/6c2b2b68-edc5-4d19-xxxx-34f6c6116292/resourceGroups/vctest/providers/Microsoft.Network/routeTables/vcs-route-tokyo-pub” です。



これで、OverlayIP 方式を使用するための Azure 側の設定と確認は完了です。

apache のクラスタリング

ここからは、apache のクラスタリング設定を行います。具体的には、apache の起動・停止・監視を行うリソースの他に、apache へのアクセス経路を切り替える OverlayIP や NIC の設定を行います。この作業は、一方のノードから行います。

まず、以下の例に従って、サービスグループ： application 内に apache を制御するリソース:app を作製します。apache を起動する際に実行するコマンドとして "/bin/systemctl start httpd.service" を、apache を停止する際に実行するコマンドとして "/bin/systemctl stop httpd.service" を、apache の稼働確認をプロセス ID ファイル：/etc/httpd/run/httpd.pid によって行うように設定しています。

```
# hares -add app Application application
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
# hares -modify app StartProgram "/bin/systemctl start httpd.service"
# hares -modify app StopProgram "/bin/systemctl stop httpd.service"
# hares -modify app PidFiles /etc/httpd/run/httpd.pid
# hares -modify app Enabled 1
```

次に、Python SDK for Azure を経由して IAM 経由で Azure をコントロールする為の認証を行うリソースを登録します。このリソース登録には、32 ページで確認した SecretKey が必要ですが、そのまま InfoScale のパラメータとして設定するのではなく、暗号化して設定します。これで InfoScale の設定ファイルが漏洩しても SecretKey が漏洩する事はありません。下記のように vcsencrypt コマンドを使用して、キーを暗号化します。暗号化されたキーは "flnNftKnITkjTpnJunLjuTjtPrtRgpFlgTonOnjPgrNrnTkrSniTmpGl fPfnSliNm" であり、これを後のリソース登録時に使用します。

```
# /opt/VRTSvcs/bin/vcsencrypt -agent 8@uxxxxxnaPlylDQhhumCwQ0PM3G" です
flnNftKnITkjTpnJunLjuTjtPrtRgpFlgTonOnjPgrNrnTkrSniTmpGl fPfnSliNm
```

準備が整ったので、認証を行うリソース：auth を作製します。このリソースには 29～32 ページで確認したサブスクリプション ID、TenantID、ClientID の 3 つの情報に加えて、上記で暗号化した SecretKey を登録します。

```
# hares -add auth AzureAuth application
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
# hares -modify auth SubscriptionId 6c2b2b68-edc5-4d19-xxxx-34f6c6116292
# hares -modify auth TenantId 071e1455-bcfd-44b7-xxxx-0f5db739bec6
# hares -modify auth ClientId 6d64598b-2b4f-4785-xxxx-29d9c5abcd95
# hares -modify auth SecretKey ¥
flnNftKnITkjTpnJunLjuTjtPrtRgpFlgTonOnjPgrNrnTkrSniTmpGl fPfnSliNm
# hares -modify auth Enabled 1
```

次に、ルートテーブルを更新するリソース：azureip を作製します。OverlayIP には 10.192.10.20 を指定します。Device には eth0 を指定します。RouteTableResourceIds には、36 ページで確認したルートテーブル ID を指定します。AzureAuthResName には、全ページで設定したリソース auth を指定します。

```
# hares -add azureip AzureIP application
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
# hares -modify azureip OverlayIP 10.192.10.20
# hares -modify azureip NICDevice eth0
# hares -modify azureip RouteTableResourceIds /subscriptions/6c2b2b68-edc5-4d19-xxxx-34f6c6116292/resourceGroups/vctest/providers/Microsoft.Network/routeTables/vcs-route-tokyo-pub
# hares -modify azureip AzureAuthResName auth
# hares -modify azureip Enabled 1
```

OverlayIP に関するリソースを設定したら、そのもととなる IP を立ち上げるリソース：ip を作製します。当然ながら使用する IP アドレスは 10.192.10.20 です。Device には eth0 を指定します。繰り返になりますが、OverlayIP は必ず Virtual Network に関連付けられた CIDR ブロックに含まれない IP にしてください。

```
# hares -add ip IP application
VCS NOTICE V-16-1-10242 Resource added. Enabled attribute must be set before agent monitors
# hares -modify ip Address 10.192.10.20
# hares -modify ip Device eth0
# hares -modify ip NetMask 255.255.255.0
# hares -modify ip Enabled 1
```

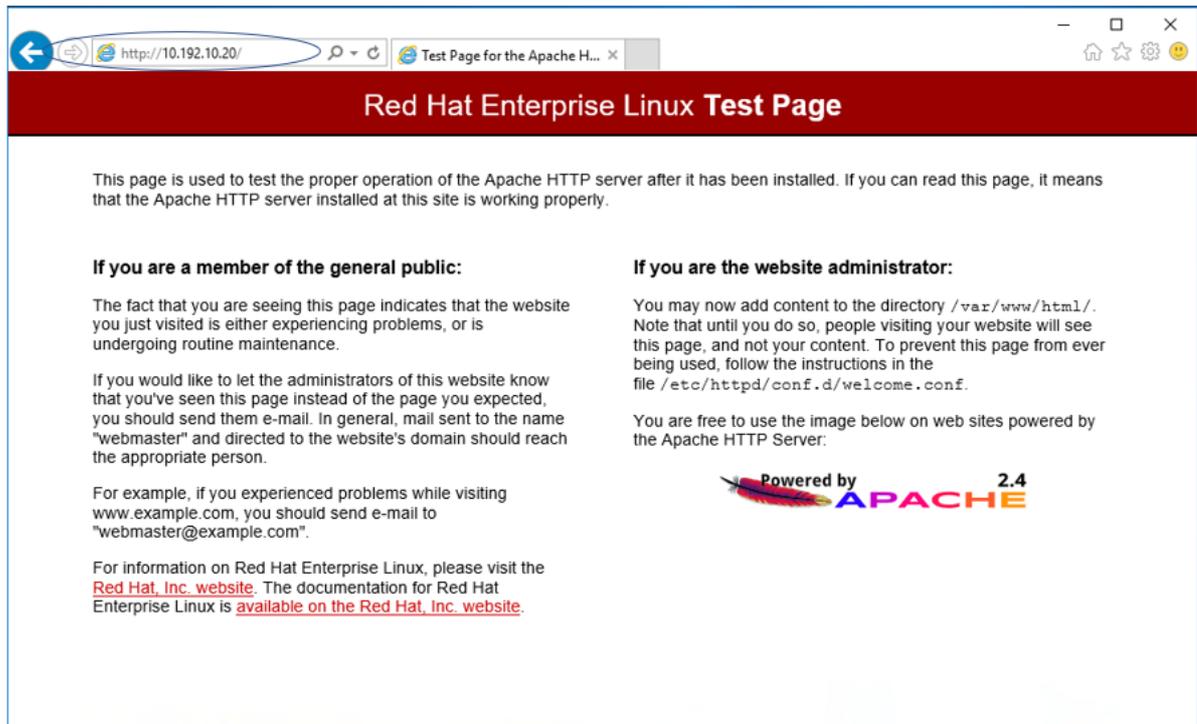
最後に、各リソースの依存関係を指定します。azureip の online 条件は auth の online、ip の online 条件は azureip の online、app の online 条件は ip なので、以下のように設定します。

```
# hares -link azureip auth
# hares -link ip azureip
# hares -link app ip
```

これで、サービスグループ: application の設定は完了です。早速、このサービスグループを 1 号機で online にします。

```
# hagr -online application -sys vcslin03
```

クライアントのブラウザから、OverlayIP : 10.192.10.20 を指定して Web アクセスできることを確認してください。



InfoScale が稼働しているノードで、サービスグループが 1 号機で online になっており、1 号機で OverlayIP : 10.192.10.20 が立ち上がっており、且つ apache が動作していることを確認します。

```
# hastatus -sum

-- SYSTEM STATE
-- System          State          Frozen
A vcslin03         RUNNING       0
A vcslin04         RUNNING       0

-- GROUP STATE
-- Group           System          Probed    AutoDisabled  State
B application     vcslin03       Y         N              ONLINE
B application     vcslin04       Y         N              OFFLINE
B cvm             vcslin03       Y         N              ONLINE
B cvm             vcslin04       Y         N              ONLINE

# ifconfig -a | grep 10.192.10.20
    inet 10.192.10.20 netmask 255.255.255.0 broadcast 10.192.10.255

# ps -ef | grep httpd
root      35357      1  0 06:51 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache   35358  35357  0 06:51 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache   35359  35357  0 06:51 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache   35360  35357  0 06:51 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache   35361  35357  0 06:51 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache   35362  35357  0 06:51 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
root     36115  15098  0 06:53 pts/0    00:00:00 grep --color=auto httpd
```

Azure Portal にて、ルートテーブルの内容を確認します。10.192.10.20 のターゲットが、1 号機の物理 IP である 10.193.24.7 になっています。

The screenshot shows the Azure Portal interface for a route table. The left sidebar contains navigation options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings, Configuration, Routes, Subnets, Properties, Locks, Export template, Support + troubleshooting, Effective routes, and New support request. The main content area shows the route table details for 'vcs-route-tokyo-pub'. Under the 'Routes' section, there is a table with the following data:

| Name | Address prefix | Next hop |
|---------------------------|-----------------|-------------|
| vcs10_192_10_20_azurevcs3 | 10.192.10.20/32 | 10.193.24.7 |

最後に、必ずクラスターの構成をディスクに反映させ、編集不可にするモード設定を行ってください。これを行わないと、サーバー再起動やクラスターの停止時に、そこまでの設定変更が無効になってしまいます。

```
# haconf -dump -makero
```

これで、全ての設定は完了です。

7. 動作確認と切り替えテスト

最後に、Azure 上の RHEL 上で、apache 及びその稼働に必要な個々のコンポーネントが、InfoScale によって適切に監視/リカバリーされるかを確認するための簡単なテストを行います。

アプリケーションを管理するサービスグループの手動切り替え

下記は、GUI から application のサービスグループを、手動で vcslin04 に切り替えるオペレーション例です。

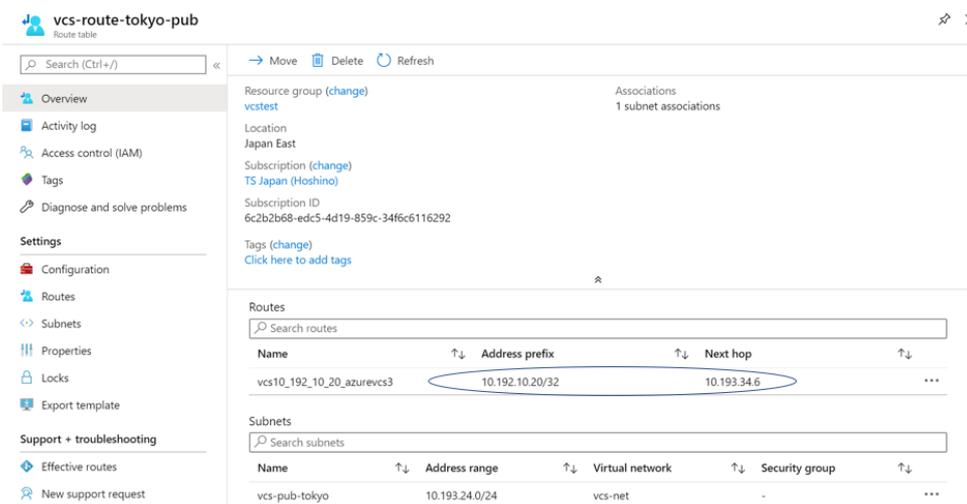
```
# hagr -switch application -to vcslin04
```

2号機への切り替えが正常に完了すると、hastatus の結果は以下のようになります。

```
# hastatus -sum
-- SYSTEM STATE
-- System          State          Frozen
A vcslin03         RUNNING        0
A vcslin04         RUNNING        0

-- GROUP STATE
-- Group           System         Probed         AutoDisabled   State
B application      vcslin03       Y              N               OFFLINE
B application      vcslin04       Y              N               ONLINE
B replication      vcslin03       Y              N               ONLINE
B replication      vcslin04       Y              N               ONLINE
```

クライアントのブラウザから、OverlayIP：10.192.10.20 を指定して Web アクセスできることを確認すると共に、Azure Portal でルートテーブルの 10.192.10.20 のターゲットが 2号機の物理 IP である 10.193.34.6 に変更されている事を確認してください。



障害によるサービスグループの自動切り替え

apache が稼働している方の RHEL インスタンスを、Azure Portal から停止してください。その後で、生き残ったノードに正しくサービスグループが切り替わって、クライアントから 10.192.10.20 に Web アクセスできる事を確認します。また、Azure Portal でルートテーブルが更新された事も確認してください。

障害ノードの復旧

上記のテストで停止したノードを Azure Portal から再立ち上げし、立ち上がったノードにサービスグループを手動で切り替えてください。この前と後で、各ノードの ifconfig や df や hastatus の表示及びルートテーブルの内容がどう変わったか確認してください。

```
# hagr -switch application -to vcslin04
```

アプリケーション (apache) 障害

apache が稼働しているノードで、以下のコマンドで apache を強制終了してください。その後、障害が検知され、他方のノードに正しくサービスグループが切り替わっていることを確認してください。

```
# systemctl stop httpd.service
```

8. 付録

main.cf の内容

本書の手順で構成した場合、main.cf ファイルは以下の内容となります。尚、main.cf のパスは、`/etc/VRTSvcs/conf/config/main.cf` です。

```
include "OracleASMTypes.cf"
include "types.cf"
include "CFSTypes.cf"
include "CRSResource.cf"
include "CSSD.cf"
include "CVMTTypes.cf"
include "Db2udbTypes.cf"
include "MultiPrivNIC.cf"
include "OracleTypes.cf"
include "PrivNIC.cf"
include "SybaseTypes.cf"

cluster azurevcs3 (
    UserNames = { admin = gmnFmhMjnInnLvnHmk }
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
)

system vcslin03 (
)

system vcslin04 (
)

group application (
    SystemList = { vcslin03 = 0, vcslin04 = 1 }
    AutoStartList = { vcslin03, vcslin04 }
)

    Application app (
        StartProgram = "/bin/systemctl start httpd.service"
        StopProgram = "/bin/systemctl stop httpd.service"
        PidFiles = { "/etc/httpd/run/httpd.pid" }
    )

    AzureAuth auth (
        SubscriptionId = 6c2b2b68-edic5-4d19-859c-34f6c6116292
        ClientId = 6d64598b-2b4f-4785-b6ba-29d9c5abcd95
        SecretKey = flNftKnITkjTpnJunLjuTjtPrtRgpFlgTonOnjPgrNrnTkrSniTmpGI fPfnSliNm
        TenantId = 071e1455-bcfd-44b7-9940-0f5db739bec6
    )

    AzureIP azureip (
        NICDevice = eth0
        OverlayIP = "10.192.10.20"
        RouteTableResourceIds = {
            "/subscriptions/6c2b2b68-edic5-4d19-859c-
34f6c6116292/resourceGroups/vctest/providers/Microsoft.Network/routeTables/vcs-route-tokyo-pub" }
        AzureAuthResName = auth
    )

    IP ip (
        Device = eth0
        Address = "10.192.10.20"
        NetMask = "255.255.255.0"
    )
```

```

requires group cvm online local firm
app requires ip
azureip requires auth
ip requires azureip

// resource dependency tree
//
//   group application
//   {
//     Application app
//     {
//       IP ip
//       {
//         AzureIP azureip
//         {
//           AzureAuth auth
//         }
//       }
//     }
//   }

group cvm (
  SystemList = { vcslin03 = 0, vcslin04 = 1 }
  AutoFailOver = 0
  Parallel = 1
  AutoStartList = { vcslin03, vcslin04 }
)

CFMount mount (
  MountPoint = "/data"
  BlockDevice = "/dev/vx/dsk/fssdg/vol01"
)

CFSfsckd vxfsckd (
)

CVMCluster cvm_clus (
  CVMClustName = azurevcs3
  CVMNodeId = { vcslin03 = 0, vcslin04 = 1 }
  CVMTransport = gab
  CVMTimeout = 200
)

CVMVolDg voldg (
  Critical = 0
  CVMdiskGroup = fssdg
  CVMVolume = { vol01 }
  CVMActivation = sw
)

CVMvxconfigd cvm_vxconfigd (
  Critical = 0
  CVMvxconfigdArgs = { syslog }
)

ProcessOnOnly vxattachd (
  Critical = 0
  PathName = "/bin/sh"
  Arguments = "- /usr/lib/vxvm/bin/vxattachd root"
  RestartLimit = 3
)

cvm_clus requires cvm_vxconfigd
mount requires voldg

```

```
voldg requires cvm_clus
vxfsckd requires cvm_clus

// resource dependency tree
//
//   group cvm
//   {
//     CFMount mount
//     {
//       CVMVoldg voldg
//       {
//         CVMCluster cvm_clus
//         {
//           CVMVxconfigd cvm_vxconfigd
//         }
//       }
//     }
//     ProcessOnOnly vxattachd
//     CFSfsckd vxfsckd
//     {
//       CVMCluster cvm_clus
//       {
//         CVMVxconfigd cvm_vxconfigd
//       }
//     }
//   }
// }
```

ベリタステクノロジーズについて

Veritas Technologies はエンタープライズデータ管理のグローバルリーダーです。複雑化した IT 環境においてデータ管理の簡素化を実現するために、世界の先進企業 50,000 社以上、Fortune 500 企業の 90 パーセントが、ベリタスのソリューションを導入しています。ベリタスのエンタープライズ・データサービス・プラットフォームは、お客様のデータ活用を推進するため、データ保護の自動化とデータリカバリを実現して、ビジネスに不可欠なアプリケーションの可用性を確保し、複雑化するデータ規制対応に必要なインサイトを提供します。ベリタスのソリューションは信頼性とスケーラビリティに優れ、500 以上のデータソースと 50 のクラウドを含む 150 以上のストレージ環境に対応しています。



ベリタステクノロジーズ合同会社

<https://www.veritas.com/ja/jp>

〒107-0052 東京都港区赤坂 1-11-44 赤坂インターシティ 4F

ベリタスセールスインフォメーションセンター（法人のお客様向け製品購入に関する相談窓口）

■電話受付時間：10:00～12:00, 13:00～17:00（土、日、祝日、年末年始を除く）

■電話番号：0120-907-000（IP 電話からは 03-4531-1799）