



Microsoft Azure

Microsoft Azure 自習書シリーズ No.16

コマンドラインによる Microsoft Azure の管理

PowerShell 編

Published: 2015 年 2 月 28 日

Updated: 2015 年 2 月 28 日

Cloudlive, Inc.



Cloudlive

本書に含まれる情報は本書の制作時のものであり、将来予告なしに変更されることがあります。提供されるソフトウェアおよびサービスは市場の変化に対応する目的で随時更新されるため、本書の内容が最新のものではない場合があります。本書の記述が実際のソフトウェアおよびサービスと異なる場合は、実際のソフトウェアおよびサービスが優先されます。Microsoft および Cloudlive は、本書の内容を更新したり最新の情報を反映することについて一切の義務を負わず、これを行わないことによる責任を負いません。また、Microsoft および Cloudlive は、本書の使用に起因するいかなる状況についても責任を負いません。この状況には、過失、あらゆる破損または損失（業務上の損失、収益または利益などの結果的な損失、間接的な損失、特別の事情から生じた損失を無制限に含む）などが含まれます。

Microsoft、SQL Server、Visual Studio、Windows、Windows Server、MSDN は米国 Microsoft Corporation および、またはその関連会社の、米国およびその他の国における登録商標または商標です。

その他、記載されている会社名および製品名は、各社の商標または登録商標です。

© Copyright 2014 Microsoft Corporation. All rights reserved.

更新履歴

バージョン	更新日	内容
v1.00	2015/2/28	・初版リリース

目次

STEP 1.	コマンドラインによる管理の概要	6
1.1	コマンドライン ツール 概要	7
•	概要	7
1.2	本自習書のゴール	8
STEP 2.	前提条件	9
2.1	前提条件	10
•	必要な環境	10
•	前提知識	10
2.2	事前準備	11
•	Microsoft Azure サブスクリプションの準備	11
STEP 3.	初期設定	12
3.1	インストール方法	13
•	Web Platform Installer の取得	13
•	Microsoft Azure PowerShell コマンドレットの起動	15
•	ヘルプの表示	16
•	PowerShell の設定	17
•	PowerShell ISE	17
3.2	サブスクリプションへの接続	18
•	AAD による認証	18
•	管理証明書による認証	22
STEP 4.	ストレージの操作	25
4.1	ストレージ アカウントの作成	26
4.2	デフォルトのストレージ アカウントを設定する	28
4.3	BLOB の操作	29
•	コンテナの作成	29
•	ファイルのアップロード	30
•	BLOB 一覧の取得	31
•	ファイルのダウンロード	31
STEP 5.	仮想ネットワークの作成	32
5.1	仮想ネットワークの作成	33
•	仮想ネットワークが定義されているか確認する	33
•	構成ファイルを作成または修正する	33
•	設定を反映する	35
STEP 6.	仮想マシンの作成	36
6.1	仮想マシン イメージの取得	37
•	仮想マシン イメージの一覧取得	37
6.2	仮想マシンの作成	39
•	仮想マシンの状態を確認する	41

コマンドラインによる Microsoft Azure の管理 PowerShell 編

6.3	仮想マシンへのリモートデスクトップ接続.....	42
STEP 7.	仮想マシンへの操作	43
7.1	仮想マシンへの操作概要.....	44
7.2	仮想マシンにデータ ディスクを追加	45
7.3	エンドポイントの追加	47
7.4	仮想マシンの管理者情報のリセット	49
	• VM エージェントがインストールされているか確認する	49
	• VM エージェントをインストールする	49
	• 管理者情報をリセットする.....	50
7.5	仮想マシンの再起動、シャットダウン、	51
	• 仮想マシンの再起動	51
	• 仮想マシンのシャットダウン	51
	• 仮想マシンの削除	52
Appendix I	REST API の呼び出し	53
I.1	サービス管理 REST API とは	54
I.2	API の呼び出し方法.....	55
	• API 概要	55
	• API 呼び出し	56
Appendix II	管理 API ライブラリの利用	60
II.1	管理 API ライブラリとは.....	61
II.2	API 利用方法	62
	• ライブラリの入手	62
	• API 呼び出し	62

STEP 1. コマンドラインによる管理の概要

この STEP では Microsoft Azure および Azure 仮想マシンの概要について説明します。

この STEP では、次のことを学習します。

- ✓ コマンドラインツールの概要
- ✓ 本自習書のゴール

1.1 コマンドライン ツール 概要

➡ 概要

Microsoft Azure で提供されるサービスを管理するにはいくつかの方法が用意されていますが、自習書ではこの中から、**Microsoft Azure PowerShell** コマンドレット を利用して、Azure を操作、管理する方法を学習します。

- 管理ポータル（Web ブラウザ）
- Microsoft Azure PowerShell モジュール
- Microsoft Azure クロス プラットフォーム コマンドライン インターフェイス
- 管理 API ライブラリ（.NET / Java / PHP / Node.js / Python / Ruby）
- サービス管理 REST API

PowerShell とは、Windows 上で動作するスクリプティング環境です。コマンドプロンプトで対話的に操作する方法に加え、スクリプトを記述し自動的にバッチ処理することもできます。

Microsoft Azure PowerShell コマンドレットは、PowerShell から Azure を管理するためのコマンドレット群です。Azure PowerShell コマンドレット を利用すれば、管理ポータルと同様なタスクを実行することができます。また、いくつかの操作は管理ポータルでは操作できず、Azure PowerShell コマンドレット を利用しなければならない場合もあります。

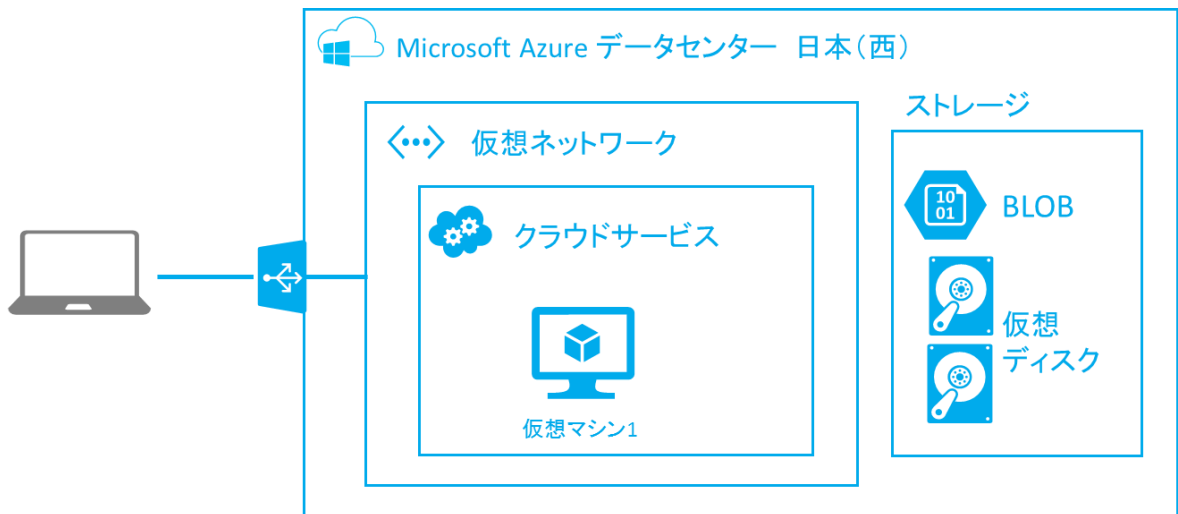
このように、Microsoft Azure をより効率的に行うためには Microsoft Azure PowerShell コマンドレットは必要不可欠なものです。

1.2 本自習書のゴール

Microsoft Azure PowerShell コマンドレットは、執筆時点で 400 種類以上のコマンドレットがあります。すべてをこの自習書で説明することはできないため、よくあるシナリオとしてストレージ アカウント、仮想ネットワーク、仮想マシンの作成までを説明していきます。

おおよその手順は、以下の通りです。

1. サブスクリプションへの接続
2. ストレージ アカウントの作成と、簡単な操作
3. 仮想ネットワークの作成
4. 作成したストレージと仮想ネットワークを利用して仮想マシンを新規作成
5. 作成した仮想マシンへの操作



それに加えて本書の最後では、REST API による Azure 管理方法を説明します。

STEP 2. 前提条件

この STEP では、この自習書で実習を行うために必要な前提について説明します。

この STEP では、次のことを学習します。

- ✓ 前提条件
- ✓ 事前準備

2.1 前提条件

➡ 必要な環境

この自習書で実習を行うために必要な環境は以下のとおりです。

- Microsoft Azure サブスクリプション
コマンドラインツールで Microsoft Azure を操作するために必要です。
- Windows クライアント
PowerShell 管理ツールをインストールするために必要です。

➡ 前提知識

この自習書で実習を行うために必要な知識は以下のとおりです。

- Microsoft Azure 仮想マシンを操作したことがあるかた、または同程度の知識
- PowerShell に関する基本的な知識と操作方法

上記の通り、管理ポータル等で Microsoft Azure 仮想マシンをの操作をしたことがあるかたを前提としていますので、ご了承ください。まだ仮想マシンを操作したことがないかたは、自習書「**No.01 Microsoft Azure 仮想マシンの立ち上げ、基本的な使用方法（Windows 編）**」をご参照ください。

2.2 事前準備

➡ Microsoft Azure サブスクリプションの準備

この自習書を進めるには、Microsoft Azure サブスクリプションをあらかじめ契約しておく必要があります。

既に有効な Microsoft アカウント および Microsoft Azure サブスクリプションをお持ちの場合、この事前作業はスキップしてください。

1. Microsoft アカウントの準備

以下の URL をブラウザで開き、新しく Microsoft アカウントを作成します。

Microsoft アカウント登録手続き

<http://www.microsoft.com/ja-jp/msaccount/signup/default.aspx>

2. Microsoft Azure サブスクリプションの作成

以下の URL をブラウザで開き、手順に従って Microsoft Azure サブスクリプションを作成します。

Microsoft Azure サブスクリプション申し込み Step by Step

<http://msdn.microsoft.com/ja-jp/windowsazure/ee943806.aspx>

サブスクリプション作成後、Microsoft Azure 管理ポータルに接続し、手順 1 で作成した Microsoft アカウントを使用してサインインできれば事前作業は完了です。

Microsoft Azure 管理ポータル

<https://manage.windowsazure.com/>

STEP 3. 初期設定

この STEP では、Microsoft Azure PowerShell コマンドレットのインストールと認証について説明します。

この STEP では、次のことを学習します。

- ✓ インストール方法
- ✓ サブスクリプションへの接続

3.1 インストール方法

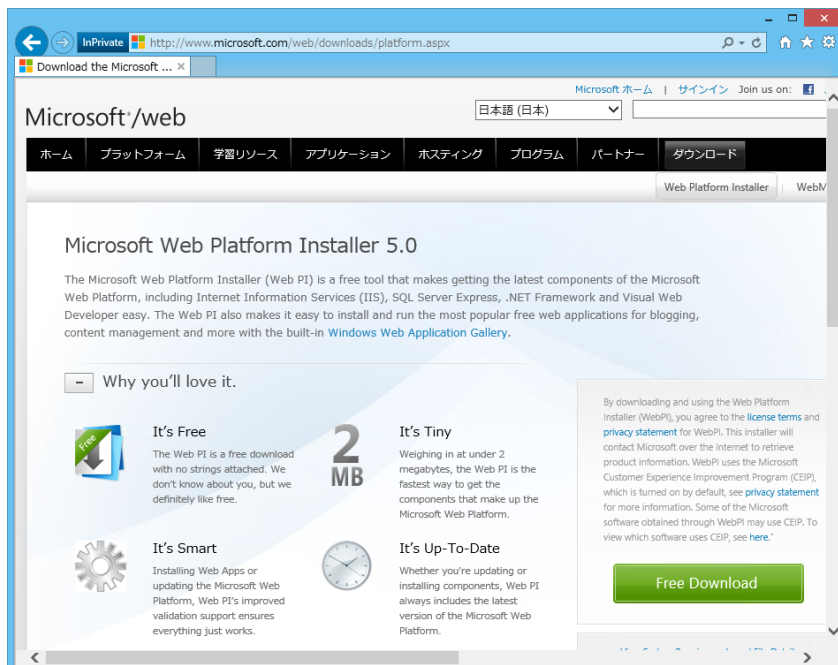
はじめに Microsoft Azure PowerShell モジュールのインストールを行います。

▼ Web Platform Installer の取得

以下の Web ページから入手できる Web Platform Installer をインストールします。

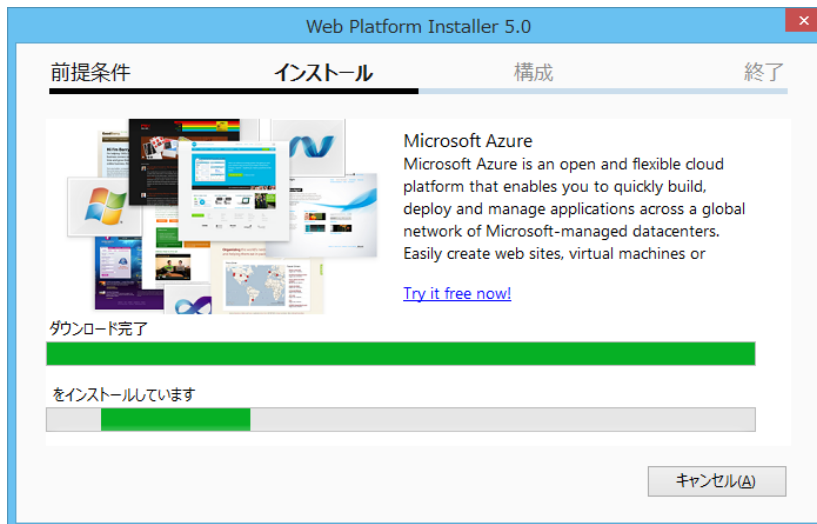
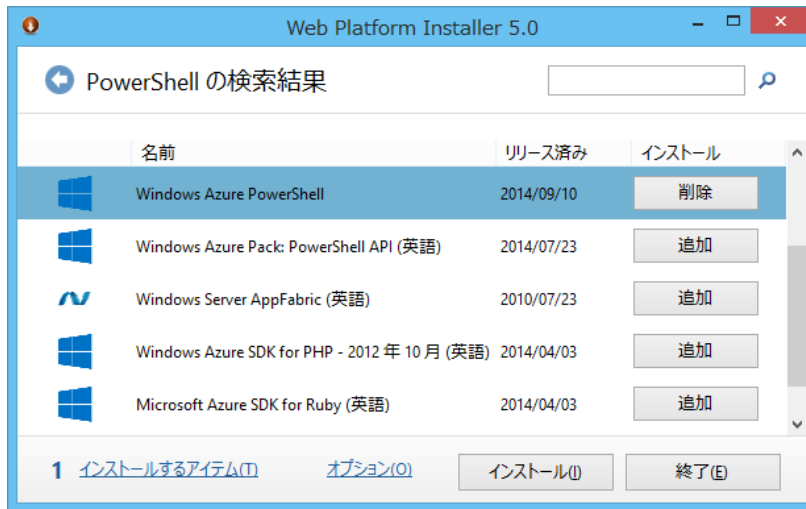
- <http://www.microsoft.com/web/downloads/platform.aspx>

1. ブラウザを起動し、Web Platform Installer のページを開きます。右下の「Free Download」をクリックし、実行します。

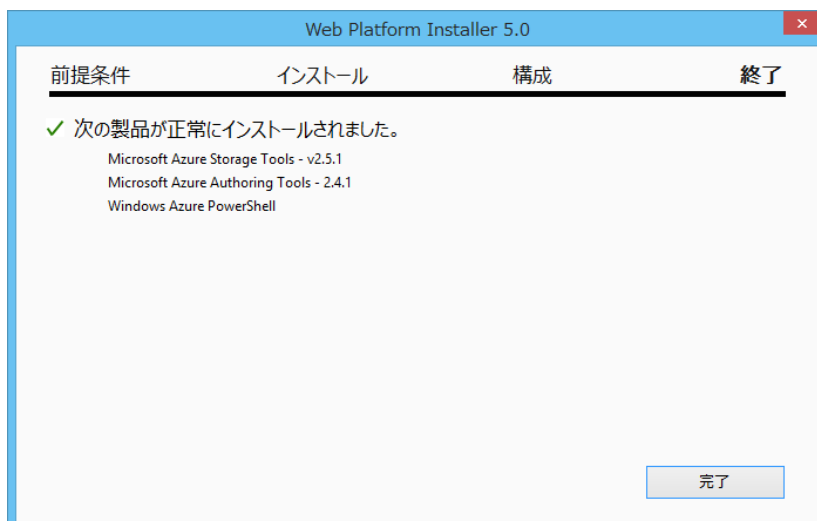


コマンドラインによる Microsoft Azure の管理 PowerShell 編

2. Windows Azure PowerShell を選択し追加ボタンをクリックします。引き続きインストールボタンをクリックするとインストールが開始されます。

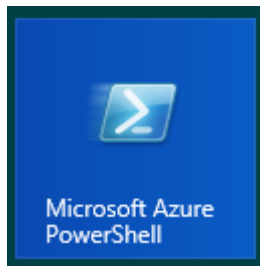


インストールが終了したら完了ボタンをクリックします。

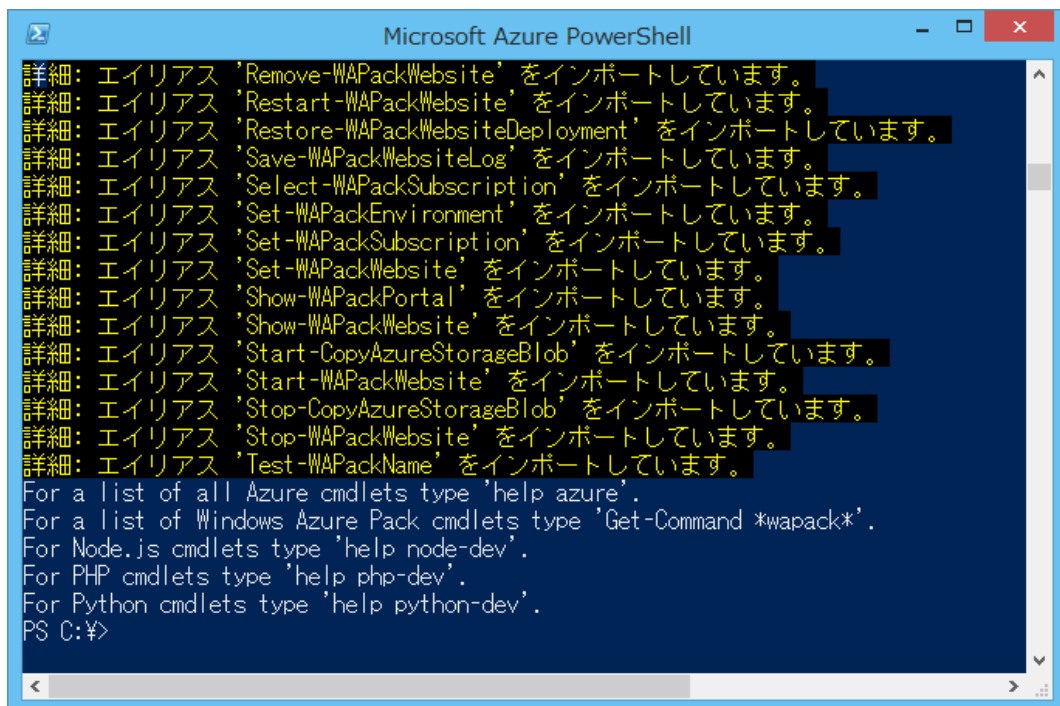


➡ Microsoft Azure PowerShell コマンドレットの起動

1. スタートメニューから Microsoft Azure PowerShell を選択します（Windows 8.1 での例）。



2. Microsoft Azure PowerShell コマンドプロンプトが起動します。これで、Azure コマンドレットが利用可能になります。



これで Azure コマンドレットが利用できる準備が整いました。

➡ ヘルプの表示

最初にヘルプに表示方法について説明しておきます。

Azure コマンドレットに含まれるコマンド一覧を表示したい場合は、Get-Help コマンドを利用します。

```
Get-Help Azure
```

Name	Category	Module	Synopsis
Add-AzureAccount	Cmdlet	Azure	Adds t...
Add-AzureCacheWorkerRole	Cmdlet	Azure	Add a ...
Add-AzureCertificate	Cmdlet	Azure	Upload...
Add-AzureDataDisk	Cmdlet	Azure	Adds a...
Add-AzureDisk	Cmdlet	Azure	Adds a...
Add-AzureDjangoWebRole	Cmdlet	Azure	Create...
Add-AzureDns	Cmdlet	Azure	Add-Az...
Add-AzureEndpoint	Cmdlet	Azure	Add en...
Add-AzureEnvironment	Cmdlet	Azure	Create...
Add-AzureHDInsightConfigValues	Cmdlet	Azure	Adds a...
Add-AzureHDInsightMetastore	Cmdlet	Azure	Adds a...
Add-AzureHDInsightStorage	Cmdlet	Azure	Adds a...
Add-AzureInternalLoadBalancer	Cmdlet	Azure	Add-Az...
Add-AzureNodeWebRole	Cmdlet	Azure	Create...
Add-AzureNodeWorkerRole	Cmdlet	Azure	Create...
Add-AzurePHPWebRole	Cmdlet	Azure	Create...
Add-AzurePHPWorkerRole	Cmdlet	Azure	Create...
Add-AzureProvisioningConfig	Cmdlet	Azure	Adds t...
Add-AzureTrafficManagerEndpoint	Cmdlet	Azure	Adds a...
Add-AzureVhd	Cmdlet	Azure	Upload...

特定のコマンドについて詳細を知りたい場合は、特定のコマンド名を指定すると、詳細な引数などが確認できます。操作に迷った場合は、ヘルプや下記の Azure コマンドレット リファレンスを参考にしてください。

```
Get-Help Get-AzureVM
```

- Azure コマンドレット リファレンス
<http://msdn.microsoft.com/ja-jp/library/azure/jj554330.aspx>

▼ PowerShell の設定

後半の手順で自作の PowerShell スクリプトを実行する手順があります。PowerShell のデフォルトでは、自作のスクリプト実行が無効化されています。以下の手順で有効化をお願いします。

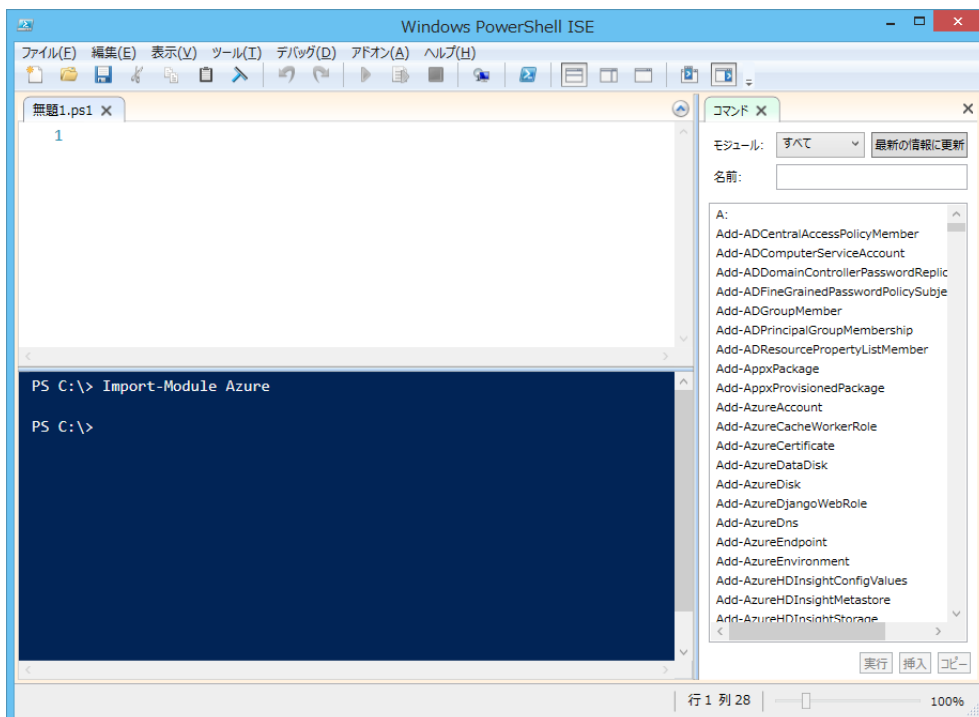
1. PowerShell を管理者権限で実行する。
2. **Set-ExecutionPolicy** コマンドレットを実行し、**ExecutionPolicy** に RemoteSigned を指定して、自作スクリプトが実行できるように権限を変更します。

```
Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
```

PowerShell には、32bit 版と 64bit 版が存在し、それぞれ設定が独立しているので注意してください。

▼ PowerShell ISE

PowerShell には統合開発環境として、PowerShell ISE と呼ばれるエディタとコンソールが一体となった便利なツールが提供されています。コマンドレットは引数の補完が効くため便利に利用できます。Azure コマンドレットは、**Import-Module Azure** で読み込むことができます。



3.2 サブスクリプションへの接続

Microsoft Azure を Azure コマンドレットから操作するためには資格情報が必要です。資格情報を認証する手段として、2 つの方法が用意されています。

- Azure Active Directory (AAD) による認証
- 管理証明書を利用した認証

最初に AAD 方式を説明し、次に管理証明書を利用した認証方法を説明します。

➡ AAD による認証

はじめに AAD による認証方法を説明します。

AAD 方式では、サブスクリプションを保有している Microsoft アカウントで Azure にログインすると、AAD により認証¹されます。ただし認証の有効時期は 12 時間であるため、有効期限が切れたら再度ログインしなおす必要があります。

ここでは対話的に認証する方法を説明しますが、バージョン 0.8.6 以降では、Credential オプションにより、資格情報を引数として指定できるようになりました。ただし組織アカウントにのみに対応しています。

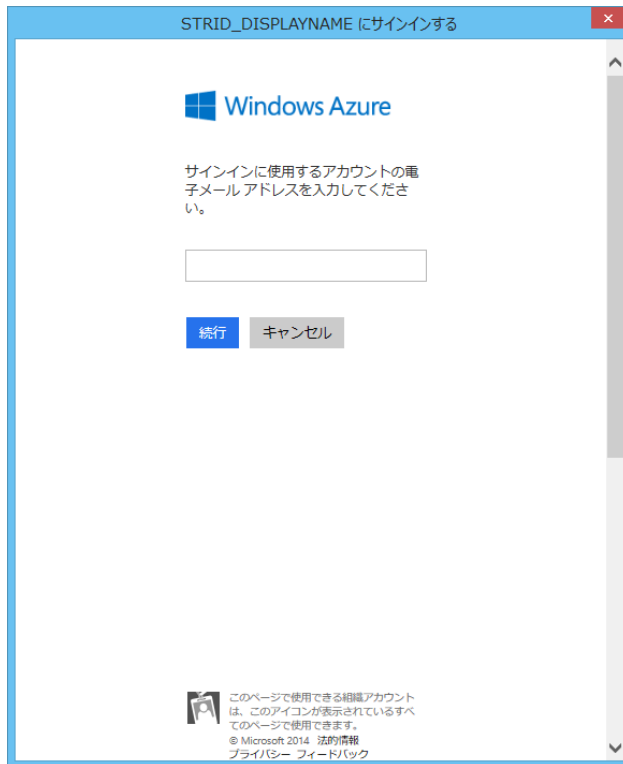
1. **Add-AzureAccount** コマンドを入力します。

```
Add-AzureAccount
```

¹AAD の既定アカウントとして、サブスクリプション作成時の Microsoft アカウントが登録されているので、この情報で認証されることになります。

コマンドラインによる Microsoft Azure の管理 PowerShell 編

2. 認証画面が表示されます。



組織アカウントに紐付いている場合、以下のダイアログが表示される可能性があります。適宜選択してください。



Microsoft アカウントとパスワードを入力して認証します。

Microsoft アカウントへのサインイン

サインイン

Microsoft アカウント [詳細](#)

☐ サインインしたままにする

[サインイン](#)

[アカウントにアクセスできない場合 一時使用コードを使ってサインイン](#)

©2014 Microsoft [プライバシーと Cookie](#) [新規登録](#)

3. 認証が完了し Azure コマンドレットで操作可能となります。

```
PS C:\> Add-AzureAccount
詳細: Account "azstudy@outlook.com" has been added.
詳細: Subscription "AzureStudy" is selected as the default subscription.
詳細: To view all the subscriptions, please use Get-AzureSubscription.
詳細: To switch to a different subscription, please use Select-AzureSubscription.

Id                                Type            Subscriptions    Tenants
--                                -
azstudy@outlook.com              User            cada499a-d476-4192-95fc-1ccb8316b058 523dfb19-
```

PS C:\> █

コマンドラインによる Microsoft Azure の管理 PowerShell 編
いくつかの補足事項を以下に示します。

- サブスクリプション情報は、**Get-AzureSubscription** コマンドで確認できます。また複数のサブスクリプションが紐付いている場合は、**Select-AzureSubscription** コマンドでどのサブスクリプションに対して操作するか明示的に指定することができます。
- また、認証情報を一端削除したい場合は、**Remove-AzureAccount** コマンドを利用します（アカウントの認証情報が削除されるだけで、アカウント自身が削除されるわけではありません）。

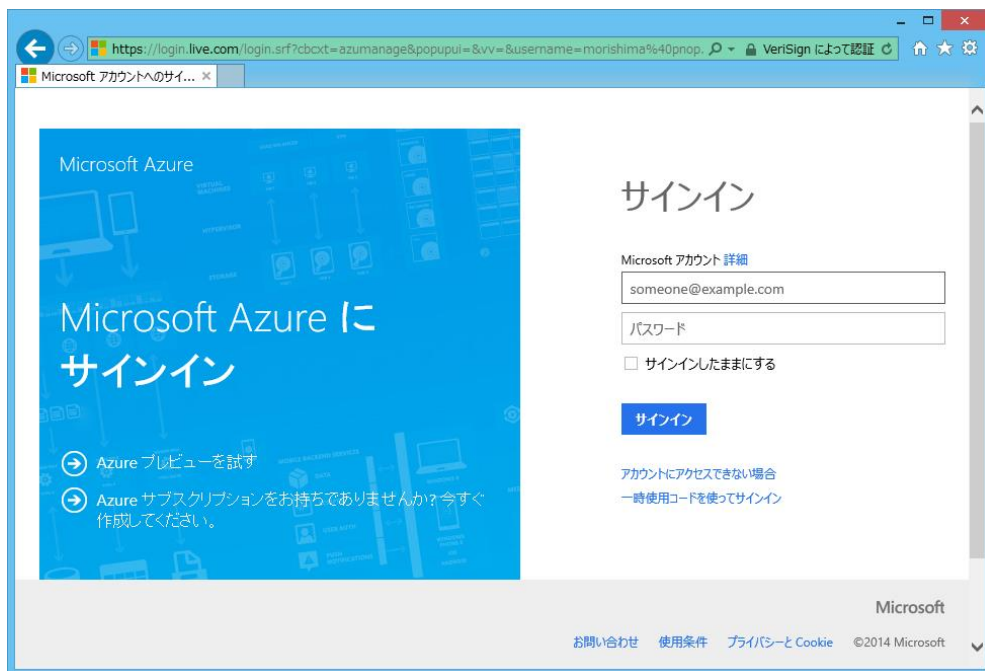
▼ 管理証明書による認証

管理証明書による認証は、サブスクリプション情報を一度ダウンロードしてインポートしておけば、再度操作する必要はありません。したがって、一度この方法で設定しておけば、スクリプトを使った自動化などが行いやすくなります。

1. **Get-AzurePublishSettingsFile** コマンドを入力します。

```
Get-AzurePublishSettingsFile
```

2. ブラウザが起動するので、Microsoft アカウントでサインインします。



3. サブスクリプションファイルのダウンロード画面が表示されます。任意の場所にファイルを保存してください。

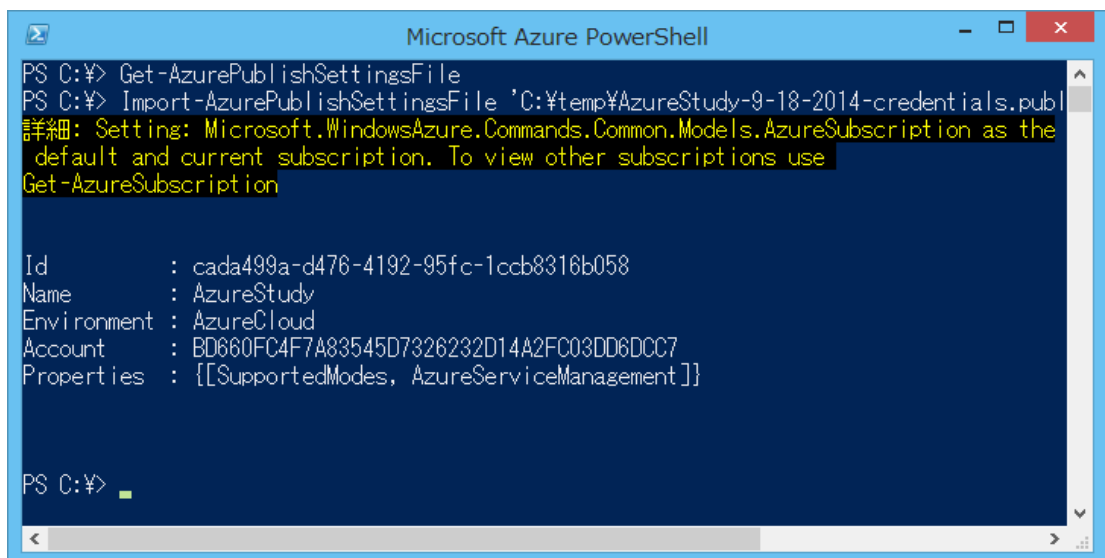
サブスクリプションを複数保持している場合は、任意のサブスクリプションを選択できます。管理したいサブスクリプションを選択して、ボタンをクリックしてください。また、サブスクリプションが一つの場合は、そのような画面は表示されません。

コマンドラインによる Microsoft Azure の管理 PowerShell 編



4. **Import-AzurePublishSettingsFile** コマンドで、保存したサブスクリプションファイルをインポートします。

```
Import-AzurePublishSettingsFile -PublishSettingsFile '保存したファイル名'
```



5. 保存したファイルを削除します。

保存したファイルが第三者の手に渡ると、Azure を自由に操作されてしまいますのでご注意ください。

コマンドラインによる Microsoft Azure の管理 PowerShell 編

6. 管理証明書は、管理ポータルの設定画面から確認できます。また、管理ポータルから削除すれば、その証明書を利用しているユーザーは利用できなくなります。

設定					
サブスクリプション		管理証明書	管理者	アフィニティグループ	使用状況
名前	状態	サブスクリプション	サブスクリプション ID	サムプリント	有効期限 ↑
AzureStudy-9-18-20...	✓ 作成済み	AzureStudy	cada499a-d476-419...	BD660FC4F7A83545...	2015/09/18

以下に、いくつかの補足事項を示します。

- 管理証明書を利用した場合、本手順を実施すれば管理証明書の利用期限が切れるまで利用できます。本手順で作成した証明書の有効期限は 1 年となっています。
- **Get-AzurePublishSettingsFile** コマンドレットによる操作を何回も実施すると、その度に Azure 管理ポータル側に管理証明書が作られます。管理証明書の数には上限があるので、注意してください。
- インポートした情報を削除するには、**Remove-AzureSubscription** コマンドで削除できます。

STEP 4. ストレージの操作

この STEP では、ストレージ アカウントの作成と簡単な BLOB 操作について説明します。

この STEP では、次のことを学習します。

- ✓ ストレージ アカウントの作成
- ✓ ストレージ アカウントのデフォルト設定
- ✓ BLOB の操作

4.1 ストレージ アカウントの作成

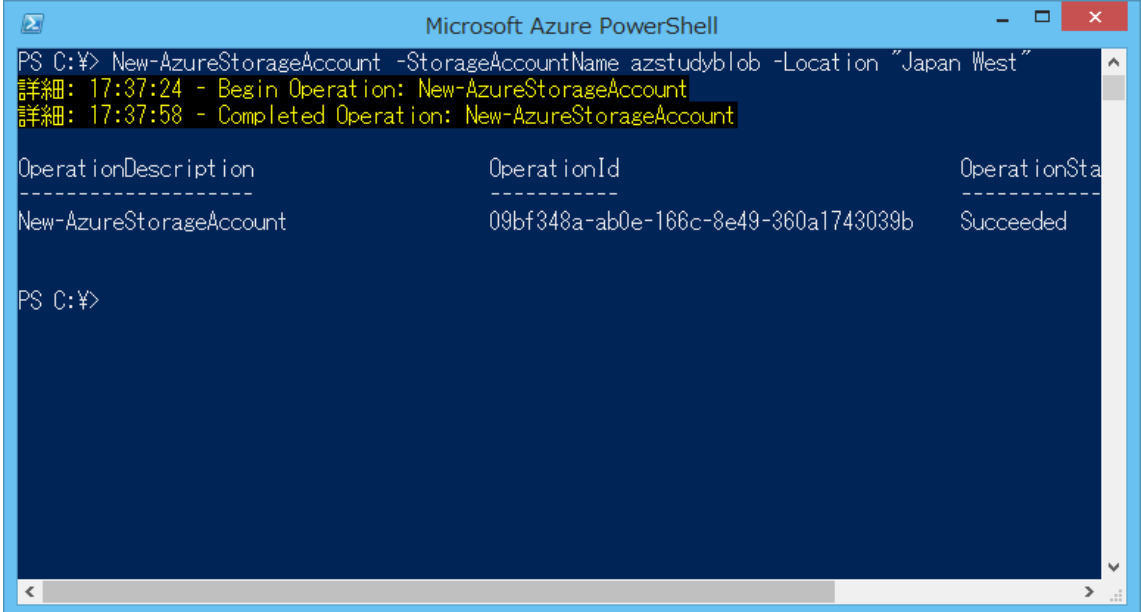
ストレージ アカウントを作成します。

1. **New-AzureStorageAccount** コマンドレット で新規にストレージ アカウントを作成します。

StorageAccountName 引数 には、ストレージ アカウント（Azure で一意な名前）を指定し、**Location** 引数には、作成したい DC の場所を指定します。

```
New-AzureStorageAccount -StorageAccountName "azstudyblob" -Location "Japan West"
```

2. **OperationStatus** が、**Succeeded** と表示されれば成功です。



The screenshot shows a PowerShell window titled "Microsoft Azure PowerShell". The command executed is `New-AzureStorageAccount -StorageAccountName azstudyblob -Location "Japan West"`. The output shows the operation starting at 17:37:24 and completing at 17:37:58. Below the command, a table displays the operation details:

OperationDescription	OperationId	OperationStatus
New-AzureStorageAccount	09bf348a-ab0e-166c-8e49-360a1743039b	Succeeded

The prompt `PS C:\>` is visible at the bottom of the console.

Location 引数に指定できる DC 名は、**Get-AzureLocation** コマンドレットで取得できます。
以下は実行例です。

```
Get-AzureLocation | Format-Table Name
```

```
Name
```

```
----
```

```
South Central US
```

```
Central US
```

```
East US 2
```

```
East US
```

```
West US
```

```
North Europe
```

```
West Europe
```

```
East Asia
```

```
Southeast Asia
```

```
Japan West
```

4.2 デフォルトのストレージ アカウントを設定する

Azure コマンドレットでストレージ アカウントを扱う場合、サブスクリプションのデフォルトのストレージ アカウントとして設定しておく必要があります。個別にストレージ アカウントを指定できる場合もありますが、デフォルトとして設定しておくると暗黙的に設定しておいたストレージ アカウントが使用されます。

これ以降の手順では、サブスクリプションにデフォルトのストレージ アカウントが設定されている前提で操作を行いますので注意してください。

1. 以下のコマンドを実行して、サブスクリプションにストレージ アカウントが設定されているか確認します。何も表示されない場合は未設定です。

```
(Get-AzureSubscription).CurrentStorageAccountName
```

2. **Set-AzureSubscription** コマンドレットを実行します。**SubscriptionName** 引数には現在のサブスクリプション名、**CurrentStorageAccountName** 引数に先ほど作成したストレージ アカウント名を指定します。

```
Set-AzureSubscription -SubscriptionName "AzureStudy" -CurrentStorageAccountName "azstudyblob"
```

4.3 BLOB の操作

Azure コマンドレットから BLOB の操作を行うことができます。ここではいくつかの簡単な BLOB を操作するためのコマンドレットを説明します。

➡ コンテナの作成

コンテナの作成は、**New-AzureStorageContainer** コマンドレットを使用します。**Name** 引数にコンテナ名を指定します。

```
New-AzureStorageContainer -Name "foo"
```

コンテナ名一覧を取得するには、**Get-AzureStorageContainer** コマンドレットを使用します。以下は実行例です。

```
> Get-AzureStorageContainer
```

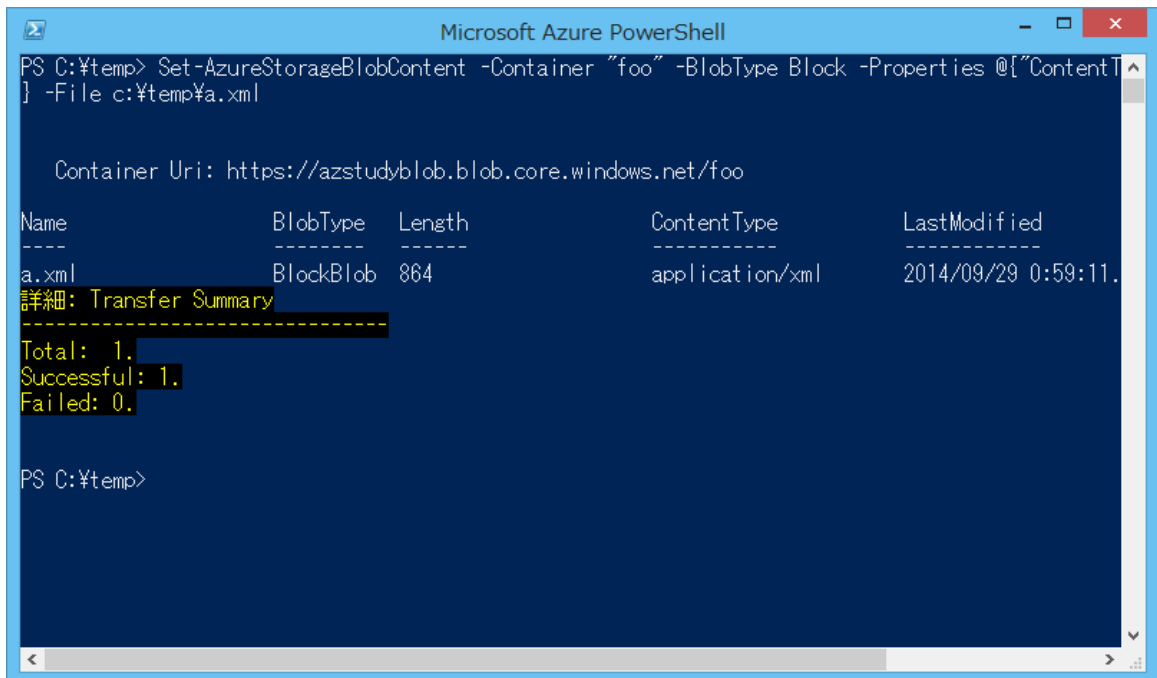
```
Blob End Point: https://azstudyblob.blob.core.windows.net/
```

Name	PublicAccess	LastModified
----	-----	-----
foo	Off	2014/09/25 7:49:54 +00:00
fuga	On	2014/09/25 7:29:22 +00:00
hoge	Off	2014/09/25 7:29:19 +00:00
vhds	Off	2014/09/23 7:58:06 +00:00

▼ ファイルのアップロード

ファイルを BLOB にアップロードするには、**Set-AzureStorageBlobContent** コマンドレットを使用します。**Container** 引数にコンテナ名、**BlobType** 引数には BLOB のタイプ (Block または Page。通常のファイルは Block を指定)、**File** 引数にはアップロードしたいファイルを指定します。

```
Set-AzureStorageBlobContent -Container "foo" -BlobType Block -Properties @{"ContentType" = "application/xml"} -File c:¥temp¥a.xml.
```



The screenshot shows a PowerShell window titled "Microsoft Azure PowerShell". The command entered is `Set-AzureStorageBlobContent -Container "foo" -BlobType Block -Properties @{"ContentType" = "application/xml"} -File c:¥temp¥a.xml`. The output shows the container URI and a table of upload results.

Name	BlobType	Length	ContentType	LastModified
a.xml	BlockBlob	864	application/xml	2014/09/29 0:59:11.

Below the table, a yellow box highlights the "Transfer Summary" section:

```
詳細: Transfer Summary
-----
Total: 1.
Successful: 1.
Failed: 0.
```

ちなみに削除は、**Remove-AzureStorageBlob** コマンドレットを使用します。

➡ BLOB 一覧の取得

特定のコンテナ内の BLOB 一覧を取得するには、**Get-AzureStorageBlob** コマンドを使用します。Container 引数にコンテナ名を指定して実行します。

```
> Get-AzureStorageBlob -Container "foo"
```

```
Container Uri: https://azstudyblob.blob.core.windows.net/foo
```

Name	BlobType	Length	ContentType	LastModified
----	-----	-----	-----	-----
a.xml	BlockBlob	864	application/xml	2014/09/26 3:34:19 +...

➡ ファイルのダウンロード

BLOB に保存されたコンテンツをダウンロードするには、**Get-AzureStorageBlobContent** コマンドレットを使用します。**Container** 引数にコンテナ名、**Blob** 引数に BLOB 名、**Destination** 引数に保存先ファイル名を指定します。

```
Get-AzureStorageBlobContent -Container "foo" -Blob a.xml -Destination c:¥temp¥z.xml
```

```
Microsoft Azure PowerShell
PS C:¥temp> Get-AzureStorageBlobContent -Container "foo" -Blob a.xml -Destination c:¥temp¥z.xml

Container Uri: https://azstudyblob.blob.core.windows.net/foo

Name          BlobType Length      ContentType      LastModified
-----
a.xml         BlockBlob 864        application/xml  2014/09/29 0:59:11.
詳細: Transfer Summary
-----
Total: 1.
Successful: 1.
Failed: 0.

PS C:¥temp>
```

STEP 5. 仮想ネットワークの作成

この STEP では、仮想ネットワークの作成について説明します。

この STEP では、次のことを学習します。

- ✓ 仮想ネットワークの作成

5.1 仮想ネットワークの作成

ストレージ アカウントに続き仮想ネットワークを作成します。利用しているサブスクリプションに、既に仮想ネットワークが作成されている場合と、まだ作成されていない場合で手順が異なりますので注意してください。

➡ 仮想ネットワークが定義されているか確認する

仮想ネットワークの存在するかの確認は、もちろん管理ポータルからもできますが、**Get-AzureVNetConfig** コマンドレットで確認することもできます。**ExportFile** 引数に出力したいファイル名を指定し、加えて **Verbose** 引数を付けて実行します。

```
Get-AzureVNetConfig -ExportToFile "c:¥temp¥vnet.xml" -Verbose
```

もし、ResourceNotFound エラーが表示された場合は、仮想ネットワークは存在しません。エラーが表示されない場合は、仮想ネットワーク 構成ファイルが出力されています。

➡ 構成ファイルを作成または修正する

仮想ネットワーク 構成ファイルが存在しなかった場合、以下のファイルを新規に作成します。

構成ファイルが存在した場合は、VirtualNetworkSite 要素（太字部分）を、VirtualNetworkSites 要素配下にコピーしてください。**既存の設定を上書きしないよう注意してください。**

```
<?xml version="1.0" encoding="utf-8"?>
<NetworkConfiguration xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://schemas.microsoft.com/ServiceHosting/2011/07/NetworkConfigura
tion">
  <VirtualNetworkConfiguration>
    <VirtualNetworkSites>
      <VirtualNetworkSite name="azurestudy-vnet" Location="Japan West">
        <AddressSpace>
          <AddressPrefix>10.0.0.0/8</AddressPrefix>
        </AddressSpace>
        <Subnets>
```

```

    <Subnet name="Subnet-1">
      <AddressPrefix>10.0.0.0/24</AddressPrefix>
    </Subnet>
    <Subnet name="Subnet-2">
      <AddressPrefix>10.0.1.0/24</AddressPrefix>
    </Subnet>
  </Subnets>
</VirtualNetworkSite>
</VirtualNetworkSites>
</VirtualNetworkConfiguration>
</NetworkConfiguration>

```

ここで出現する主要な要素の説明は以下の通りです。

要素	概要
VirtualNetworkSite	仮想ネットワークを定義します。Name 属性に仮想ネットワーク名を、Location 属性には作成する DC の場所を指定します。 ここでは、日本（西）に作成しています。
AddressSpace/AddressPrefix	仮想ネットワーク サイトに使用するアドレス空間を定義します。プライベート IP アドレス空間を複数指定することができます。
Subnet	仮想ネットワーク内のサブネットを定義します。Name 属性にサブネット名を指定します。
Subnet/AddressPrefix	サブネットのアドレス空間を定義します。Addressspace/AddressPrefix で指定したアドレスのサブセットを指定する必要があり、複数していただけます。

詳細なスキーマは、以下の「Azure 仮想ネットワークの構成スキーマ」を参照してください。

<http://msdn.microsoft.com/ja-jp/library/azure/jj157100.aspx>

設定を反映する

構成ファイルの準備ができたなら、**Set-AzureVNetConfig** コマンドレットを使用して仮想ネットワークの設定を反映します。

```
Set-AzureVNetConfig -ConfigurationPath C:¥temp¥vnet.xml
```

Get-AzureVNetSite コマンドレットを実行するか、管理ポータルで作成を確認できます。

```

Microsoft Azure PowerShell
PS C:¥temp> Get-AzureVNetSite
詳細: 10:13:31 - Begin Operation: Get-AzureVNetSite
詳細: 10:13:34 - Completed Operation: Get-AzureVNetSite

AddressSpacePrefixes : [10.0.0.0/8]
AffinityGroup         :
DnsServers            : []
GatewayProfile        :
GatewaySites         :
Id                   : eaa712bb-086f-4842-a30c-4b4ee7c660ba
InUse                 : False
Label                :
Name                 : azurestudy-vnet
State                : Created
Subnets             : {Subnet-1, Subnet-2}
OperationDescription : Get-AzureVNetSite
OperationId          : a97192b2-62c6-116c-a0bf-c734b229bb77
OperationStatus      : Succeeded

PS C:¥temp>

```

ネットワーク			
仮想ネットワーク ローカル ネットワーク DNS サーバー			
名前	状態	サブスクリプション	場所
azurestudy-vnet →	✓ 作成済み	AzureStudy	日本 (西)

STEP 6. 仮想マシンの作成

この STEP では、この自習書で実習を行うために必要な前提について説明します。

この STEP では、次のことを学習します。

- ✓ 仮想マシンイメージの取得
- ✓ 仮想マシンの作成
- ✓ 仮想マシンへのリモートデスクトップ接続

6.1 仮想マシン イメージの取得

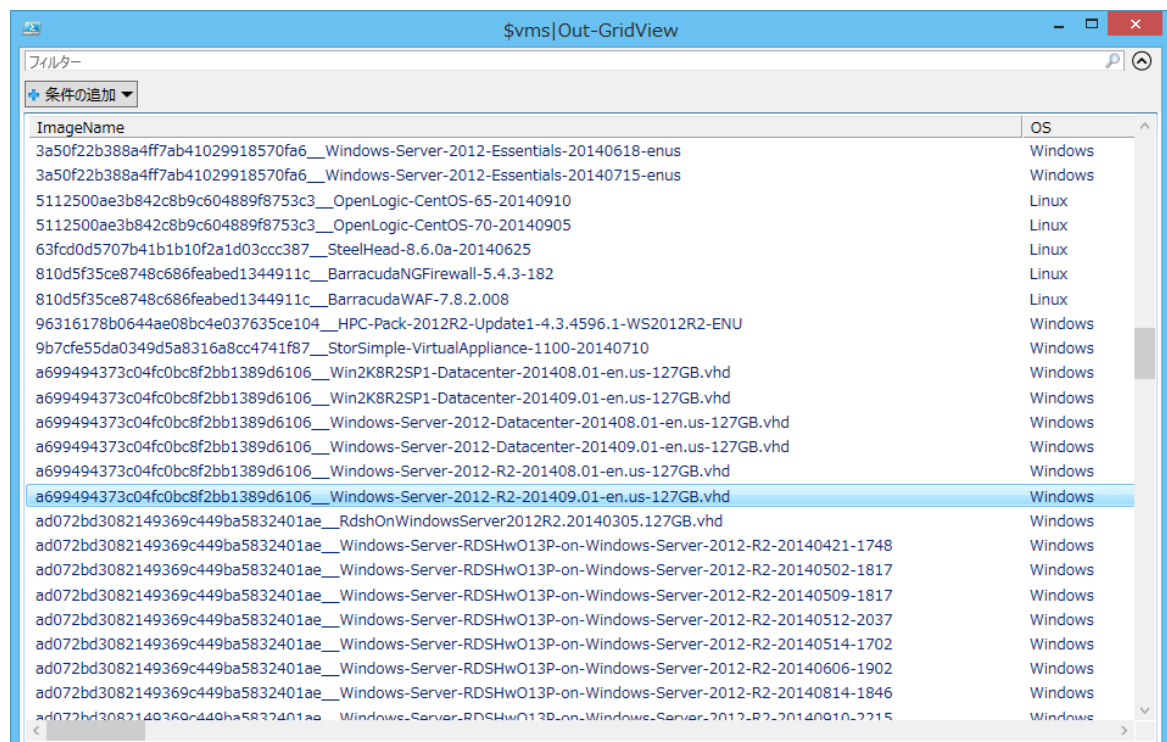
GUI での操作では、仮想マシンイメージは一覧から選択するだけですが、CUI 操作の場合は、コマンドでイメージ名を取得する必要があります。イメージ一覧には、Linux ベースのものや、3rd パーティが提供する選択不可能なイメージも存在するので注意してください。

▼ 仮想マシン イメージの一覧取得

仮想マシン イメージの一覧取得には、**Get-AzureVMImage** コマンドレットを使用します。ただし引数なしで実行すると、多数の VM イメージの情報がコンソールに出力されてしまうだけになってしまいます。したがって、以下のように **Out-GridView** コマンドレットをパイプで接続すると GUI で参照でき便利です。

```
Get-AzureVMImage | Out-GridView
```

以下、Out-GridView で表示した例です。多数の VM イメージが提供されていることが分かります。



コマンドラインによる Microsoft Azure の管理 PowerShell 編
 また特定の VM イメージ情報を取り出したいときは、PowerShell の **Where-Object** コマンドレット等と組み合わせると便利です。本自習書では、Windows Server 2012 R2 のイメージを利用するため、**Label** プロパティにて判別しています。

以下は実行例です。

```
> Get-AzureVMImage | Where-Object { $_.Label -like "Windows Server 2012 R2*" } |
Sort-Object -Descending PublishedDate | Select-Object -First 1
```

ImageName : a699494373c04fc0bc8f2bb1389d6106__Windows-Server-2012-R2-201411.01-en.us-127GB.vhd
 OS : Windows
 MediaLink :
 LogicalSizeInGB : 128
 AffinityGroup :
 Category : Public
 Location : East Asia;Southeast Asia;North Europe;West Europe;Japan West;Central US;East US;East US 2;South Central US;West US
 Label : Windows Server 2012 R2 Datacenter, September 2014
 Description : At the heart of the Microsoft Cloud OS vision, Windows Server 2012 R2 brings Microsoft's experience delivering global-scale cloud services into your infrastructure. It offers enterprise-class performance, flexibility for your applications and excellent economics for your datacenter and hybrid cloud environment. This image includes Windows Server 2012 R2 Update.

<< 以下 省略 >>

実際に仮想マシン作成時に必要なものは、**ImageName** プロパティの値です。

6.2 仮想マシンの作成

仮想マシンイメージ名が取得できたところで、先ほど作成した仮想ネットワーク上に新しい仮想マシンを作成します。

仮想マシンが利用するディスクを保存するストレージ アカウントは、前述したとおり、サブスクリプションに設定したデフォルトのストレージ アカウント上に作成されます（vhds というコンテナ配下に任意の名前で作成されます）。

仮想マシンの作成は対話的に行うより、スクリプトとして実行したほうが完結に記述し実行できるため、ここでは、以下のスクリプトを記述し実行してください。

作成の流れとしては、仮想マシンの構成情報オブジェクトを複数のコマンドレットをパイプして作成し、**New-AzureVM** コマンドレットに引数として渡す流れとなります。

以下をテキストエディタを開き、**CreateVM.ps1** という名前で保存します。管理者ユーザー名、パスワード、仮想ネットワーク名、VM 名などは適宜変更します。

```
# 設定情報
$vmUser   = "管理者ユーザー名"
$vmPsswd  = "パスワード"
$vnnetName = "azurestudy-vnet"
$vmName   = "azurestudy02"

# Windows 2012 Server R2 の VMイメージ取得
$vmImage = Get-AzureVMImage | Where-Object { $_.Label -like "Windows Server 2012 R2*" } | Sort-Object -Descending PublishedDate | Select-Object -First 1

# VMの構成設定
$vmConfig = `
    New-AzureVMConfig -Name $vmName -InstanceSize "Basic_A1" -ImageName $vmImage.ImageName | `
    Add-AzureProvisioningConfig -Windows -AdminUsername $vmUser -Password $vmPsswd | `
    Set-AzureSubnet -SubnetNames "Subnet-1" | `
    Set-AzureStaticVNetIP -IPAddress "10.0.0.22"

# VMの作成
New-AzureVM -ServiceName $vmName -VNetName $vnnetName -VMs $vmConfig -Location "Japan West"
```

コマンドラインによる Microsoft Azure の管理 PowerShell 編
保存したスクリプト CreateVM.ps1 を実行します。Succeed と表示されれば正常に実行完了です。

```

Microsoft Azure PowerShell
PS C:\temp> .\CreateVM.ps1
詳細: 15:08:09 - Begin Operation: Get-AzureVMImage
詳細: 15:08:27 - Completed Operation: Get-AzureVMImage
詳細: 15:08:27 - Begin Operation: Get-AzureVMImage
詳細: 15:08:28 - Completed Operation: Get-AzureVMImage
警告: The specified DNS name is already taken.
詳細: 15:08:42 - Begin Operation: New-AzureVM - Create Deployment with VM azurestudy02
詳細: 15:09:48 - Completed Operation: New-AzureVM - Create Deployment with VM azurestudy02

OperationDescription              OperationId                        OperationStatus
-----
New-AzureVM                      1be912e0-aa4c-1a1b-9dc9-8a947e1c8e43  Succeeded

PS C:\temp>

```

スクリプト内の各コマンドレットの動きについては、以下の通りです。

コマンドレット	概要
New-AzureVMConfig	仮想マシンの構成オブジェクトを作成します。仮想マシン名やインスタンスサイズ、作成したいイメージ名を指定します。
Add-AzureProvisioningConfig	仮想マシンのプロビジョニング構成を追加します。 この例では、Windows 引数で Windows 仮想マシンであることを明示し、引き続き AdminUser 引数で管理者ユーザー名と、Passowrd 引数でパスワードを指定します。
Set-AzureSubnet	仮想マシンの配置先サブネットを指定します。 サブネットはあらかじめ作成されている必要があります。
Set-AzureStaticVNetIP	仮想マシンのプライベート IP アドレスを指定します。 この例では、IPAddress 引数に指定した"10.0.0.5"に固定されます。このコマンドレットを指定しない場合は、DHCP により任意の IP アドレスが割り当てられます。
New-AzureVM	指定された仮想マシン構成オブジェクトを元に、Azure 仮想マシンを作成します。 ServiceName 引数でクラウドサービス名、VNetName 引数で仮想ネットワーク名、Location 引数で DC の場所を指定します。

▼ 仮想マシンの状態を確認する

作成した仮想マシンの状態を確認するには、**Get-AzureVM** コマンドレットを使用します。状態が、**ReadyRole** であれば準備完了です。

```
> Get-AzureVM
```

ServiceName	Name	Status
-----	----	-----
azurestudy02	azurestudy02	ReadyRole

詳細な情報を参照したい場合は、**ServiceName** 引数をつけて明示的にサービス名を指定します。

```
Microsoft Azure PowerShell
PS C:\temp> Get-AzureVM -ServiceName azurestudy02 -Name azurestudy02
詳細: 15:45:47 - Completed Operation: Get Deployment

DeploymentName      : azurestudy02
Name                : azurestudy02
Label              :
VM                 : Microsoft.WindowsAzure.Commands.ServiceManagement.Model.Persisten
InstanceStatus      : ReadyRole
IpAddress           : 10.0.0.5
InstanceStateDetails :
PowerState          : Started
InstanceErrorCode    :
InstanceFaultDomain : 0
InstanceName        : azurestudy02
InstanceUpgradeDomain : 0
InstanceSize        : Basic_A1
HostName            : azurestudy02
AvailabilitySetName  :
DNSName             : http://azurestudy02.cloudapp.net/
Status              : ReadyRole
GuestAgentStatus     : Microsoft.WindowsAzure.Commands.ServiceManagement.Model.GuestAgen
ResourceExtensionStatusList : {Microsoft.Compute.BGInfo}
```

当然管理ポータルからも確認できます。

仮想マシン			
インスタンス イメージ ディスク			
名前	↑	状態	サブスクリプション 場所
azurestudy02	→	✓ 実行中	AzureStudy 日本 (西)

6.3 仮想マシンへのリモートデスクトップ接続

仮想マシンへリモートデスクトップ接続するには、**Get-AzureRemoteDesktopFile** コマンドレットを使用します。

ServiceName 引数と、**Name** 引数で接続したい仮想マシンを指定し、**LocalPath** 引数で RDP ファイルの保存先を指定します。保存したファイルを開けば、リモートデスクトップ接続が開始されますので、仮想マシン作成時にしていたユーザー名とパスワードを指定して、仮想マシンに接続します。

```
Get-AzureRemoteDesktopFile -ServiceName "azurestudy02" -Name "azurestudy02" -LocalPath "c:¥temp¥vm.rdp"
```

LocalPath 引数の変わって、**Launch** 引数を指定して、直接リモートデスクトップ接続を起動することもできます。

STEP 7. 仮想マシンへの操作

この STEP では、この自習書で実習を行うために必要な前提について説明します。

この STEP では、次のことを学習します。

- ✓ 仮想マシンへの操作概要
- ✓ 仮想マシンにデータ ディスクを追加
- ✓ 仮想マシンへのエンドポイントの追加
- ✓ 仮想マシンの管理者情報リセット
- ✓ 仮想マシンの再起動、シャットダウン、削除

7.1 仮想マシンへの操作概要

本章では、Azure コマンドレットを使用した仮想マシンへの操作方法を説明します。

仮想マシンへの操作は複数のコマンドレットをパイプで連結して操作する方法が一般的です。よくあるパターンとしては、以下の流れでコマンドレットを実行します。

1. **Get-AzureVM** コマンドレットにより仮想マシン情報のオブジェクトを取得する。
2. 仮想マシンへの操作コマンドを実行する。例えば次節で説明する、**Add-AzureDataDisk** コマンドレットなど。
3. **Update-AzureVM** コマンドレットを実行して、仮想マシンを更新する。

7.2 仮想マシンにデータ ディスクを追加

仮想マシンへデータディスクを追加する方法を説明します。

前述した流れのとおり、はじめに **Get-AzureVM** コマンドレットで仮想マシン情報を取得し、**Add-AzureDataDisk** コマンドレットでデータディスクの追加を指示し、**Update-AzureVM** コマンドレットで仮想マシンの構成を更新します。

Add-AzureDataDisk コマンドレットでは、**CreateNew** 引数で新規作成を指示し、**DiskSizeInGB** でデータディスクの容量を指示し、**LUN** では論理ディスク番号を 0～15 の間で指定します。

データディスクは、サブスクリプションに設定されたデフォルトのストレージ アカウント上に作成されます。

```
Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" | Add-AzureDataDisk -CreateNew -DiskSizeInGB 30 -DiskLabel "DataDisk" -LUN 0 | Update-AzureVM
```

Get-AzureDataDisk コマンドレットで仮想マシンに接続されたデータディスク情報を確認できます。以下は実行例です。

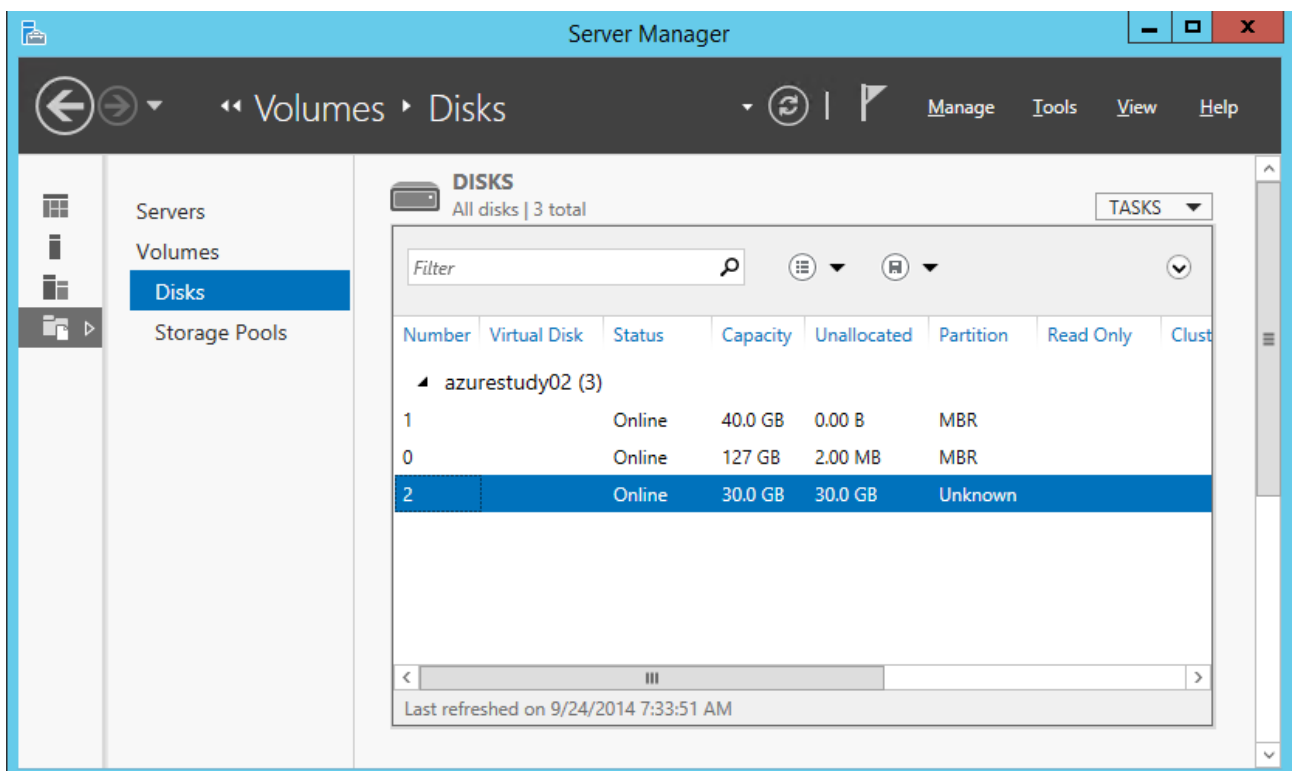
```
> Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" | Get-AzureDataDisk

HostCaching      : None
DiskLabel        : DataDisk
DiskName         : azurestudy02-azurestudy02-0-201409240632280212
Lun              : 0
LogicalDiskSizeInGB : 30
MediaLink        : https://azstudyblob.blob.core.windows.net/vhds/azurestudy02-azurestudy
                  02-DataDisk-2014-9-24-588.vhd
SourceMediaLink  :
ExtensionData    :
```

コマンドラインによる Microsoft Azure の管理 PowerShell 編
他と同様に管理ポータル（仮想マシンのダッシュボード）からも確認できます。

ディスク			
ディスク	種類	ホスト キャッシュ	V. 
azurestudy02-azurestudy02-0-...	OS ディスク	読み取り/書き込み	https:
azurestudy02-azurestudy02-0-...	データ ディスク	なし	https:

仮想マシンに接続し、サーバermanageで確認します。30GB のデータディスクが接続されていることが確認できます。後は通常の Windows 操作を実施しデータディスクを初期化しマウントすれば、利用可能になります。



7.3 エンドポイントの追加

エンドポイントの追加は、**Add-AzureEndpoint** コマンドレットを使用します。管理ポータルによる GUI 操作では複数のエンドポイントを一括して定義することはできませんが、PowerShell スクリプトで実行すれば簡単に定義できます。

ここでは、以下のスクリプトのどおり、5000 から 5010 番の TCP エンドポイントを 10 個追加し、最後に仮想マシンを更新するサンプルを説明します。

Name 引数でエンドポイント名、**Protocol** 引数 では、Tcp もしくは Udp を指定できます。**PublicPort**/**LocalPort** にはポート番号を指定します。

```
$startPort = 5000
$endPort   = 5010
$prefix    = "Port-"

$vm = Get-AzureVM -ServiceName azurestudy02 -Name azurestudy02

For($port = $startPort; $port -le $endPort; $port++)
{
    $name = $prefix + $port
    $vm = $vm | Add-AzureEndpoint -Name $name -Protocol tcp -PublicPort $port -LocalPort $port
}

$vm | Update-AzureVM
```

Get-AzureEndpoint コマンドレットで仮想マシンのエンドポイント情報を取得できます。

```
Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" | Get-AzureEndpoint |
Out-GridView

# コンソールに出力した場合は、Format-Table コマンドレットを使用する
Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" | Get-AzureEndpoint |
Format-Table
```

コマンドラインによる Microsoft Azure の管理 PowerShell 編

以下、Out-GridView での表示例です。

LBSetName	LocalPort	Port	Name	Protocol	Vip	ProbePath	ProbePort
	5,000	5,000	Port-5000	tcp	23.100.106.250		0
	5,001	5,001	Port-5001	tcp	23.100.106.250		0
	5,002	5,002	Port-5002	tcp	23.100.106.250		0
	5,003	5,003	Port-5003	tcp	23.100.106.250		0
	5,004	5,004	Port-5004	tcp	23.100.106.250		0
	5,005	5,005	Port-5005	tcp	23.100.106.250		0
	5,006	5,006	Port-5006	tcp	23.100.106.250		0
	5,007	5,007	Port-5007	tcp	23.100.106.250		0
	5,008	5,008	Port-5008	tcp	23.100.106.250		0
	5,009	5,009	Port-5009	tcp	23.100.106.250		0
	5,010	5,010	Port-5010	tcp	23.100.106.250		0
	5,986	56,728	PowerShell	tcp	23.100.106.250		0
	3,389	51,337	RemoteDes...	tcp	23.100.106.250		0

管理ポータルからの確認もできます。

名前	プロトコル	パブリック ポート	プライベート ポート
Port-5000	TCP	5000	5000
Port-5001	TCP	5001	5001
Port-5002	TCP	5002	5002
Port-5003	TCP	5003	5003
Port-5004	TCP	5004	5004
Port-5005	TCP	5005	5005
Port-5006	TCP	5006	5006
Port-5007	TCP	5007	5007
Port-5008	TCP	5008	5008
Port-5009	TCP	5009	5009
Port-5010	TCP	5010	5010
PowerShell	TCP	56728	5986
RemoteDesktop	TCP	51337	3389

7.4 仮想マシンの管理者情報のリセット

仮想マシンの管理者情報をリセットする方法を説明します。ここでいう管理者情報とは仮想マシン作成時に指定してユーザー名とパスワードです。

リセットするためには、仮想マシンに Azure VM エージェントがインストールされている必要があります。過去に作成した仮想マシンなどはインストールされていない可能性があります。

➡ VM エージェントがインストールされているか確認する

Get-AzureVM コマンドレットで取得できるオブジェクトの **VM** プロパティの **ProvisionGuestAgent** プロパティを参照します。

```
(Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02").VM.ProvisionGuestAgent
```

結果が True なら VM エージェントはインストールされていますが、False の場合はインストールされていません。

➡ VM エージェントをインストールする

VM エージェントがインストールされていない場合は以下の手順でインストールできます。現時点では明示的に無効化しないかぎり VM エージェントはデフォルトでインストールされます。

1. 仮想マシンにリモートデスクトップ接続しログインします。
2. 以下のページから VM エージェントのインストーラ(MSI)をダウンロードします。
<http://go.microsoft.com/fwlink/?LinkID=394789&clcid=0x409>
3. 仮想マシン上で VM エージェントをインストールします。ウィザードにしたがってインストールすれば問題ありません。
4. 以下の手順で仮想マシンの構成を更新します。

```
$vm = Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02"  
$vm.VM. ProvisionGuestAgent = $TRUE  
$vm | Update-AzureVM
```

▼ 管理者情報をリセットする

仮想マシンの資格情報をリセットするには、**Set-AzureVMAccessExtension** コマンドレットを使用します。**UserName** 引数に管理者ユーザー名、**Password** 引数に新しいパスワードを指定します。

```
Get-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" | Set-AzureVMAccessExtension -UserName "azureuser" -Password "somepassword!" | Update-AzureVM
```

仮想マシン作成時と異なるユーザー名を指定すると、仮想マシン作成時に作成した初期ユーザー名から変更されますが、ホームディレクトリ等の変更ありません。

7.5 仮想マシンの再起動、シャットダウン、

最後に仮想マシンの再起動、シャットダウン削除について説明します。

➡ 仮想マシンの再起動

仮想マシンの再起動は、**Restart-AzureVM** コマンドレットを使用します。

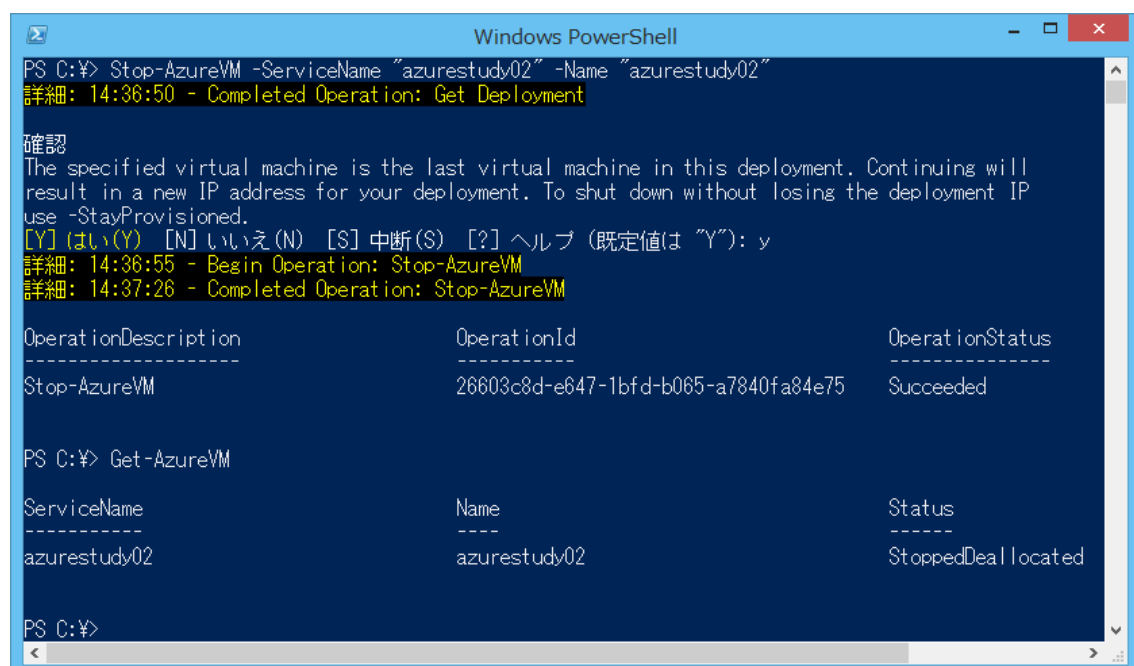
```
Restart-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02"
```

➡ 仮想マシンのシャットダウン

仮想マシンのシャットダウンには、**Stop-AzureVM** コマンドレットを使用します。最後のデプロイメントの場合、IP アドレスが開放される旨のメッセージが表示されますが、Y を押して続行します。

```
Stop-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02"
```

Get-AzureVM コマンドレットで状態を確認したときに、**StoppedDeallocated** となっていれば停止済みかつ IP アドレスが解除済みで、課金されない状態となっています。**StayProvisioned** 引数を指定すると IP アドレスは解放されませんが、課金されたままの状態となってしまうので、注意してください。



```
Windows PowerShell
PS C:\> Stop-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02"
詳細: 14:36:50 - Completed Operation: Get Deployment

確認
The specified virtual machine is the last virtual machine in this deployment. Continuing will
result in a new IP address for your deployment. To shut down without losing the deployment IP
use -StayProvisioned.
[Y] はい(Y) [N] いいえ(N) [S] 中断(S) [?] ヘルプ (既定値は "Y"): y
詳細: 14:36:55 - Begin Operation: Stop-AzureVM
詳細: 14:37:26 - Completed Operation: Stop-AzureVM

OperationDescription      OperationId                OperationStatus
-----
Stop-AzureVM              26603c8d-e647-1bfd-b065-a7840fa84e75  Succeeded

PS C:\> Get-AzureVM

ServiceName      Name      Status
-----
azurestudy02     azurestudy02  StoppedDeallocated

PS C:\>
```

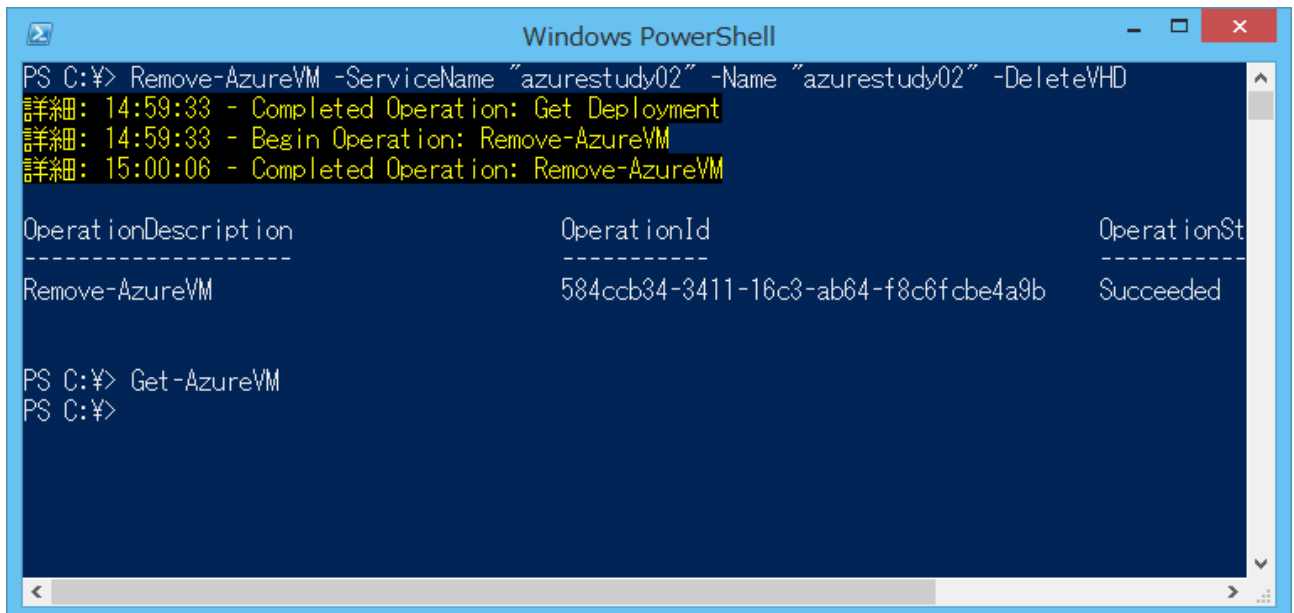
コマンドラインによる Microsoft Azure の管理 PowerShell 編
停止した仮想マシンを開始するには、**Start-AzureVM** コマンドレットを使用します。

➡ 仮想マシンの削除

仮想マシンを削除するには、**Remove-AzureVM** コマンドレットを使用します。**DeleteVHD** 引数を付加すると同時に BLOB に展開された OS イメージファイルも削除されます。

```
Remove-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" -DeleteVHD
```

Get-AzureVM コマンドレット確認し、仮想マシンが無くなっていれば削除完了です。



```
Windows PowerShell
PS C:\> Remove-AzureVM -ServiceName "azurestudy02" -Name "azurestudy02" -DeleteVHD
詳細: 14:59:33 - Completed Operation: Get Deployment
詳細: 14:59:33 - Begin Operation: Remove-AzureVM
詳細: 15:00:06 - Completed Operation: Remove-AzureVM

OperationDescription          OperationId                    OperationSt
-----
Remove-AzureVM                584ccb34-3411-16c3-ab64-f8c6fcbe4a9b  Succeeded

PS C:\> Get-AzureVM
PS C:\>
```

Appendix I REST API の呼び出し

この STEP では、サービス管理 REST API の使用方法について説明します。

この STEP では、次のことを学習します。

- ✓ サービス管理 REST API とは
- ✓ API の呼び出し方法

I.1 サービス管理 REST API とは

Azure を管理するための低レベル な API として、Service Management REST API （サービス管理 REST API）が公開されています。

ここまで説明してきた、Azure PowerShell コマンドレット も、最終的にはこれらの API 群を利用して実装されています。

本章では、サービス管理 REST API の簡単な使用方法を説明します。ここでは、C#を使って記述しますが、他言語で呼び出しを行う場合についても、基本的な作法は同じです。

リファレンスについては、以下の URL を参照してください。

- サービス管理 REST API リファレンス
<http://msdn.microsoft.com/ja-jp/library/azure/ee460799.aspx>

サービス管理 API を呼び出すためには、Azure Active Directory による認証か、証明書による認証が必要です。本書では証明書による認証方法で実施します。「3.2 サブスクリプションへの接続 管理証明書による認証」を実施していれば、必要な証明書が Azure およびクライアント PC に作成されていますので、この手順を実施しておいてください。

詳しくは、以下の URL を参照してください。

- サービス管理要求の認証
<http://msdn.microsoft.com/ja-jp/library/azure/ee460782.aspx>

I.2 API の呼び出し方法

▶ API 概要

ここでは、簡単なサンプルとして List Cloud Services API を呼び出してみます。API の詳細は、以下の URL を参照します。

- List Cloud Services
<http://msdn.microsoft.com/ja-jp/library/azure/ee460781.aspx>

リファレンスを参照すると、メソッドと要求 URI が説明されています。URL 中の **<subscription-id>** には、利用しているサブスクリプション ID を設定します。サブスクリプション ID は、**Get-AzureSubscription** コマンド等で取得できます。

API によっては、他のパラメータを要求するものがありますので、それらは任意に設定します。

メソッド	要求 URI
GET	<a href="https://management.core.windows.net/<subscription-id>/services/hostedservices">https://management.core.windows.net/<subscription-id>/services/hostedservices

また呼び出しには、**x-ms-version** ヘッダが必須となっています。バージョン管理については、以下の URL を参照してください。

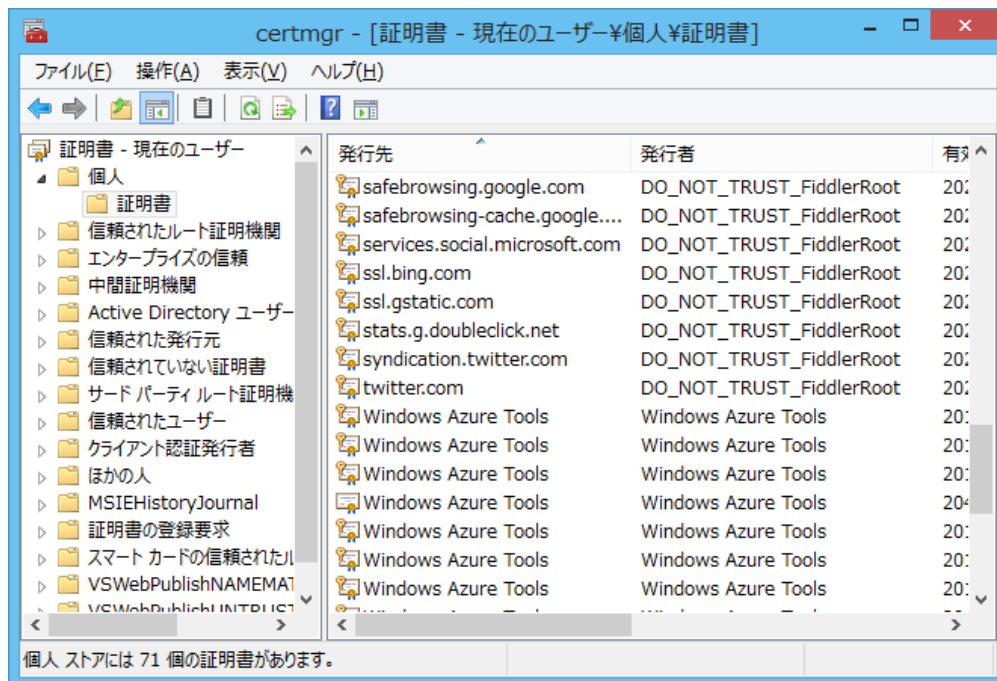
- サービス管理のバージョン管理
<http://msdn.microsoft.com/ja-jp/library/azure/gg592580.aspx>

➡ API 呼び出し

はじめに証明書を取得します。以下の例では、Windows の証明書ストアから、証明書のサムプリントをキーに取得しています。

```
// 証明書の取得
var store = new X509Store(StoreName.My, StoreLocation.CurrentUser);
store.Open(OpenFlags.ReadOnly);
var collection = store.Certificates.Find(X509FindType.FindByThumbprint,
    "4D4FFC91FCE55B6287E2EA3ECF2BB43455DF0FA2", false);
var x509 = collection[0];
store.Close();
```

Windows の証明書は、ユーザー証明書の管理ツールから確認できます。



コマンドラインによる Microsoft Azure の管理 PowerShell 編

取得した証明書は、**WebRequestHandler** インスタンスの **ClientCertificates** プロパティに追加します。

また、**x-ms-version** ヘッダ には、サポートされているバージョン日付けを設定します。

最後に API を呼び出し、結果を取得します。結果のスキーマはリファレンスに記述されてます。ほとんどの API は XML を返却するため、XML パーサーでパースすれば中身を解析することができます。

```
// URLの設定
var subscriptionId = "020fe900-c88b-4a6d-85e5-5777f8edeba6";
var url = string.Format("https://management.core.windows.net/{0}/services/hostedservices", subscriptionId);

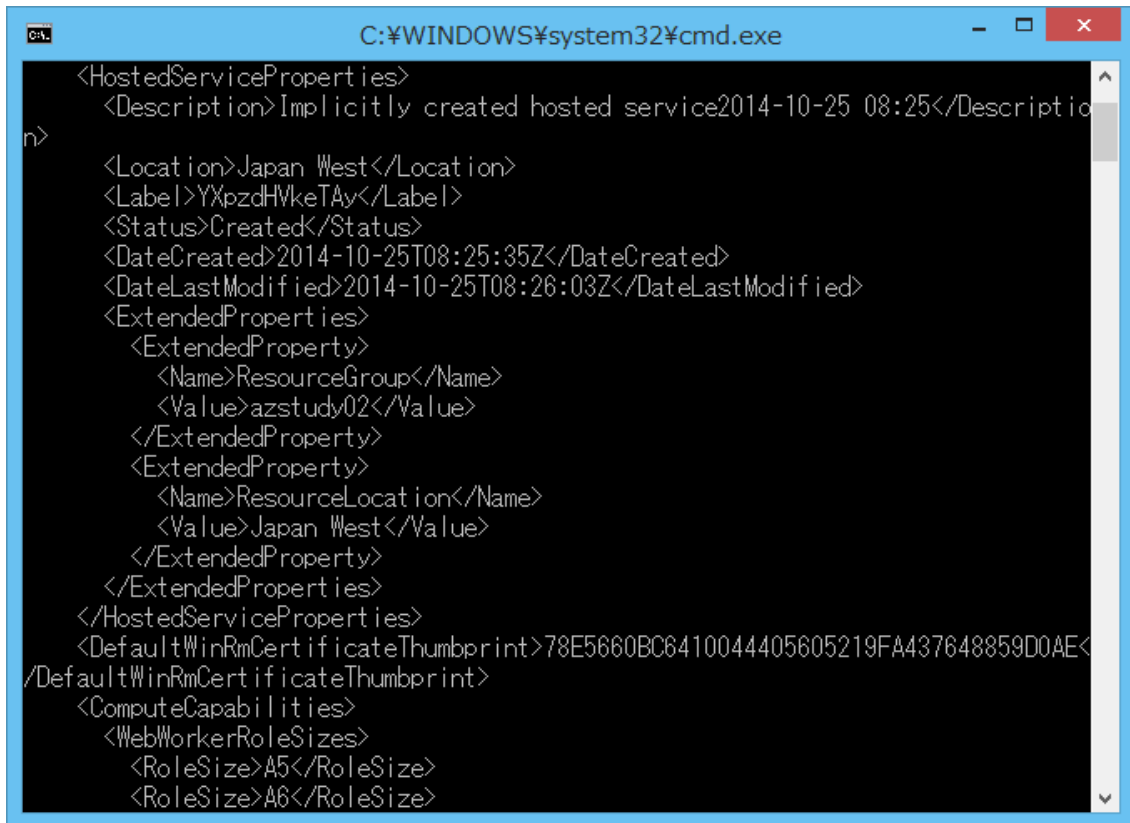
// 証明書の追加
var handler = new WebRequestHandler();
handler.ClientCertificates.Add(x509);

// REST APIの呼び出し
var client = new HttpClient(handler);
client.DefaultRequestHeaders.Add("x-ms-version", "2014-06-01");
var response = await client.GetStringAsync(url);

// XMLをパースして出力
var xml = XElement.Parse(response);
Console.WriteLine(xml.ToString());
```

コマンドラインによる Microsoft Azure の管理 PowerShell 編

実行例は以下の通りです。



```

C:\WINDOWS\system32\cmd.exe

n>
<HostedServiceProperties>
  <Description>Implicitly created hosted service2014-10-25 08:25</Description>
  <Location>Japan West</Location>
  <Label>YXpzdHVkeTAy</Label>
  <Status>Created</Status>
  <DateCreated>2014-10-25T08:25:35Z</DateCreated>
  <DateLastModified>2014-10-25T08:26:03Z</DateLastModified>
  <ExtendedProperties>
    <ExtendedProperty>
      <Name>ResourceGroup</Name>
      <Value>azstudy02</Value>
    </ExtendedProperty>
    <ExtendedProperty>
      <Name>ResourceLocation</Name>
      <Value>Japan West</Value>
    </ExtendedProperty>
  </ExtendedProperties>
</HostedServiceProperties>
<DefaultWinRmCertificateThumbprint>78E5660BC6410044405605219FA437648859D0AE<
/DefaultWinRmCertificateThumbprint>
<ComputeCapabilities>
  <WebWorkerRoleSizes>
    <RoleSize>A5</RoleSize>
    <RoleSize>A6</RoleSize>
  </WebWorkerRoleSizes>
</ComputeCapabilities>

```

これらの REST API をラップした各言語用のライブラリも提供されているため、それらを利用する方法もあります。

Appendix II 管理 API ライブラリの利用

この STEP では、管理 API ライブラリ的使用方法について説明します。

この STEP では、次のことを学習します。

- ✓ サービス管理 REST API とは
- ✓ API の呼び出し方法

II.1 管理 API ライブラリとは

Appendix I では、Azure を管理するための低レベルな API を紹介しましたが、管理 API ライブラリとは、これらの REST API を各言語毎に用意したライブラリでラップしたものです。

主要な言語毎に用意されており、独自の管理ツールを柔軟に作成することができます。また直接 REST API を利用するよりも、簡単に管理 API を呼び出すことができます。

用意されている言語とライブラリ取得先は以下の通りです。すべて GitHub 上にてオープンソースとして公開されています。

- .NET Framework (C# , VB など)
<https://github.com/Azure/azure-sdk-for-net>
- Java
<https://github.com/Azure/azure-sdk-for-java>
- PHP
<https://github.com/Azure/azure-sdk-for-php>
- Node.js
<https://github.com/Azure/azure-sdk-for-node>
- Python
<https://github.com/Azure/azure-sdk-for-python>
- Ruby
<https://github.com/Azure/azure-sdk-for-ruby>

本書では、.NET Framework 向けライブラリを C#から利用した方法を説明します。.NET Framework 向けのリファレンスは、以下の URL を参照してください。

- Management Libraries for NET
[http://msdn.microsoft.com/en-us/library/dn602775\(v=azure.11\).aspx](http://msdn.microsoft.com/en-us/library/dn602775(v=azure.11).aspx)

II.2 API 利用方法

管理 API の仕様は多岐にわたるため、ここでは簡単なサンプルとしてクラウドサービスの情報を取得する API を呼び出してみます。REST API と同じく証明書を利用した認証を行います。

PowerShell で提供されている操作であれば、ほぼ同様な仕組みが実装されていますので、詳細はリファレンスを参照してください。

➡ ライブラリの入手

ライブラリは Nuget 経由でインストールすることができます。詳しくは、以下の URL を参照してください。

- Microsoft Azure Management Libraries 2.0.0
<https://www.nuget.org/packages/Microsoft.WindowsAzure.Management.Libraries>

➡ API 呼び出し

初めに管理 API を呼び出すためのクライアントクラスを作成します。

サブスクリプション ID と証明書情報から **CertificateCloudCredentials** オブジェクトを生成します。

次に **CertificateCloudCredentials** オブジェクトを引数に、**ComputeManagementClient** オブジェクトを生成します。このクライアントクラスは、クラウドサービスを管理するためのクラスです。他にも、ストレージや仮想ネットワーク、Web サイトなど管理対象のコンポーネント毎に用意されています。詳細は、リファレンスを参照してください。

```
// 初期設定
var x509 = <<証明書情報>>
var subscriptionId = "021fe701-c78f-5b7f-9ae4-5666f7acebaf";

// 証明書から認証情報を取得
CertificateCloudCredentials creds =
    new CertificateCloudCredentials(subscriptionId, x509);
ComputeManagementClient client =
    CloudContext.Clients.CreateComputeManagementClient(creds);
```

コマンドラインによる Microsoft Azure の管理 PowerShell 編
つぎに、**ComputeManagementClient** クラスの **HostedServices** プロパティを経由し、**Get** メソッドを呼び出すと、クラウドサービスの情報を取得できます。

```
var hs = client.HostedServices.Get("azurestudy02");  
  
Console.WriteLine("Name      : " + hs.ServiceName);  
Console.WriteLine("Label     : " + hs.Properties.Label);  
Console.WriteLine("Location : " + hs.Properties.Location);  
Console.WriteLine("Status  : " + hs.Properties.Status);
```

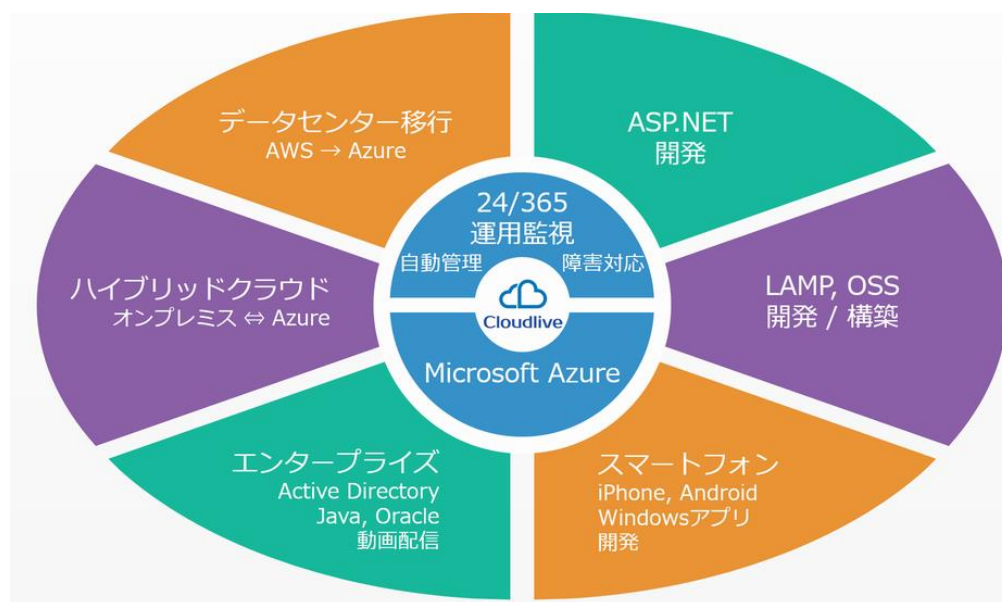
執筆者プロフィール

Cloudlive 株式会社 (<http://www.cloudlive.jp>)



皆様が Microsoft Azure の恩恵を受け、最大限に活用できるよう、支援することをミッションとした企業です。24/365 の運用監視や、各種コンサルティング、開発支援を行っています。

Azure の 2008 年プレビュー時から、Azure 事業に取り組んでおり、Windows, Linux とともに日本 TOP のノウハウと実績を持ちます。Microsoft Azure MVP 経験者が 4 名在籍しており、Microsoft 本社へフィードバックや情報交換も頻繁に行うとともに、変化の速いクラウド業界において最新のノウハウを提供します。お困りの点がありましたら、ぜひご相談ください。本書に対する感想や、ご意見もお待ちしています。



安心、安全の運用監視

24時間365日 Microsoft Azure を監視



ノウハウに基づく、最適なプラン、構成を提案

Microsoftテクノロジーに限らず、Linux/OSSの実績も豊富



Microsoft Azureスペシャリストによるサービス提供

Microsoft Azure MVP経験者4名 + 経験豊富なメンバー



初回アセスメント無料

ちょっとしたわからないことも、まずはご相談ください