

Azure Stack HCI のプロキシ事情 23H2対応版

Windows Server and Cloud User Group Japan
後藤 諭史 (Satoshi GOTO)

自己紹介

- ▶ 後藤 諭史 (Satoshi GOTO)
- ▶ 国内Sierでプリセールスやっています
- ▶ 仮想化製品が主な専門分野です。
が、基本的には雑用係
- ▶ Microsoft MVP - Cloud and Datacenter Management
(Jul.2012 - Jun.2025)
Microsoft MVP – Microsoft Azure (Jul.2024 - Jun.2025)

お約束ですが.....



本セッション資料ですが、個人で準備した環境において、個人的に実施した検証／結果を基に記載しています。
あくまで個人の意見／見解であり、所属する会社の正式な回答／見解ではない事に留意してください。

お約束ですが.....その 2



クラウドサービスを取り扱っているため、セッション当日（2024/09/11）時点の情報となります。
セッション終了直後、いきなり仕様が変更される場合もありますのでご了承ください。

エンタープライズ環境あるある

Internetへはプロキシ経由でしかアクセスできません

Azure Stack HCI は「Azureのサービス」 したがって、Azureにはアクセスできない

① Important

Azure Express Route and Azure Private Link are not supported for Azure Stack HCI, version 23H2 or any of its components as it is not possible to access the public endpoints required for Azure Stack HCI, version 23H2.

<https://learn.microsoft.com/en-us/azure-stack/hci/concepts/firewall-requirements>

Azure Express Route および Azure Private Linkは、Azure Stack HCI 23H2に必要なパブリックエンドポイントにアクセスできないため、Azure Stack HCI 23H2 またはそれらのコンポーネントではサポートされていません。

**プロキシ環境下では、プロキシ経由での
Azure接続が必須**

プロキシ環境で涙を流したことはありませんか？

というわけで、この辺りを整理

Windows環境におけるプロキシ設定の種類

▶ WinInet

- ブラウザなどの対話型アプリケーション用のプロキシ設定
- GUIでは「設定」→「ネットワークとインターネット」→「プロキシ」で設定可能
- Azure Stack HCI ではSet-WinInetProxyコマンドレットの使用が推奨されている

▶ WinHttp

- OSのサービスなどで利用されるシステム通信用のプロキシ設定
- コマンドプロンプトの「netsh winhttp proxy」コマンドで設定
- PowerShellでは「Set-WinHttpProxy」コマンドレットで設定可能

▶ 環境変数

- WinInet、WinHttpに依存しないアプリケーションに対する、環境変数でのプロキシ設定
- Azure Stack HCI においては、Azure Resource BridgeとAKSのプロキシ設定として使用

プロキシの設定内容

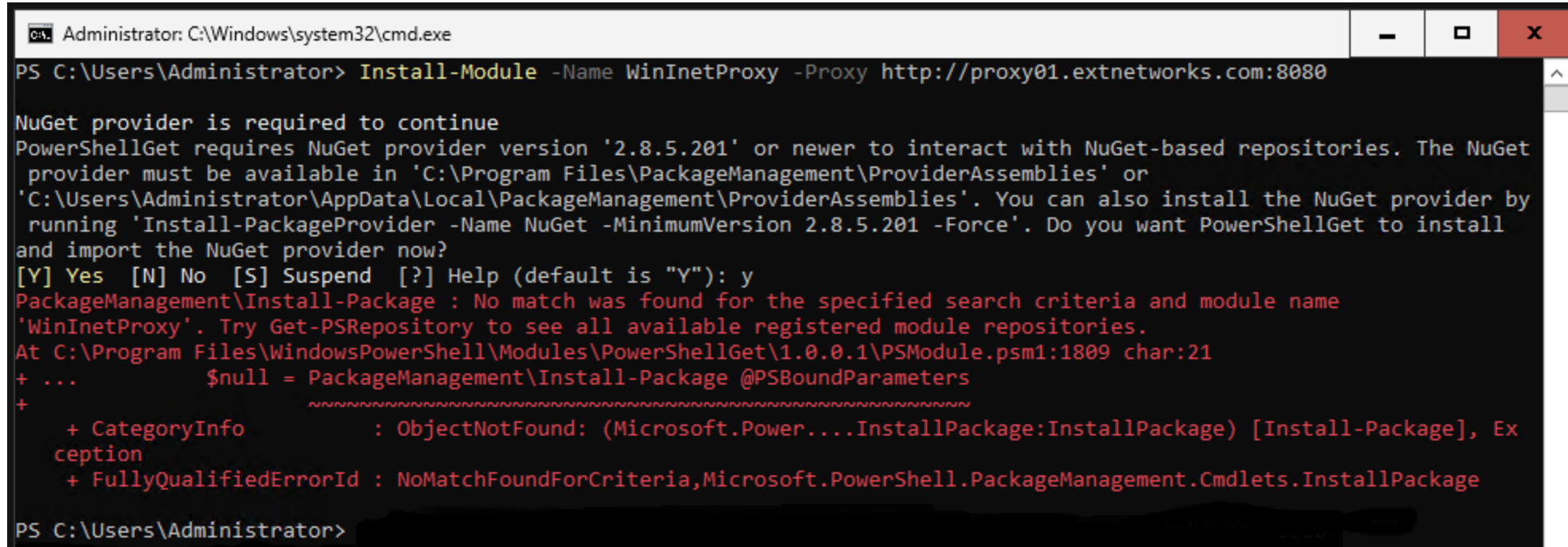
- ▶ プロキシサーバーのアドレス
 - 「http://proxy01.extnetworks.com:8080」などのURI+ポート番号形式で指定
- ▶ 除外設定（プロキシバイパスとかバイパスリスト）
 - イン트라ネットなどのプロキシを経由しない通信先を指定
 - 無設定だと、WS-ManagementなどのHTTPベースの通信もプロキシにいてしまうので、いろいろ大変
 - ドメインサーバーあての通信や自ネットワークのIPアドレスなどを除外対象に

認証付きプロキシ

- ▶ プロキシを使用する際にユーザーID/Passwordによる認証を必要とするプロキシサーバー
 - IP Addressとユーザー名を紐づける場合、最もお手軽な手法
- ▶ Azure Stack HCI においては、ユーザーIDとPasswordを用いた認証付きプロキシは非サポート
 - ユーザーIDとPasswordを記憶するのはセキュリティ的な問題がある
 - そのため、ユーザーID認証は非サポート
- ▶ Azure Stack HCI ノード、その他の関連するIPアドレスからの通信は、認証除外すべき

WinInetProxyのインストール

```
Install-Module -Name WinInetProxy -Proxy http://proxy01.extnetworks.com:8080
```



```
C:\Windows\system32\cmd.exe
PS C:\Users\Administrator> Install-Module -Name WinInetProxy -Proxy http://proxy01.extnetworks.com:8080

NuGet provider is required to continue
PowerShellGet requires NuGet provider version '2.8.5.201' or newer to interact with NuGet-based repositories. The NuGet
provider must be available in 'C:\Program Files\PackageManagement\ProviderAssemblies' or
'C:\Users\Administrator\AppData\Local\PackageManagement\ProviderAssemblies'. You can also install the NuGet provider by
running 'Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force'. Do you want PowerShellGet to install
and import the NuGet provider now?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y
PackageManagement\Install-Package : No match was found for the specified search criteria and module name
'WinInetProxy'. Try Get-PSRepository to see all available registered module repositories.
At C:\Program Files\WindowsPowerShell\Modules\PowerShellGet\1.0.0.1\PSModule.psm1:1809 char:21
+ ...          $null = PackageManagement\Install-Package @PSBoundParameters
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (Microsoft.Power....InstallPackage:InstallPackage) [Install-Package], Ex
ception
+ FullyQualifiedErrorId : NoMatchFoundForCriteria,Microsoft.PowerShell.PackageManagement.Cmdlets.InstallPackage

PS C:\Users\Administrator>
```

- ▶ Install-Moduleコマンドレットの「-Proxy」オプションを使用してプロキシを指定
- ▶ NuGetのインストール後パッケージの検出で失敗する.....

PowerShell モジュールリポジトリの登録

```
Administrator: C:\Windows\system32\cmd.exe
PS C:\Users\Administrator> Install-Module -Name WinInetProxy -Proxy http://proxy01.extnetworks.com:8080

NuGet provider is required to continue
PowerShellGet requires NuGet provider version '2.8.5.201' or newer to interact with NuGet-based repositories. The NuGet
provider must be available in 'C:\Program Files\PackageManagement\ProviderAssemblies' or
'C:\Users\Administrator\AppData\Local\PackageManagement\ProviderAssemblies'. You can also install the NuGet provider by
running 'Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force'. Do you want PowerShellGet to install
and import the NuGet provider now?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y
PackageManagement\Install-Package : No match was found for the specified search criteria and module name
'WinInetProxy'. Try Get-PSRepository to see all available registered module repositories.
At C:\Program Files\WindowsPowerShell\Modules\PowerShellGet\1.0.0.1\PSModule.psm1:1809 char:21
+ ...          $null = PackageManagement\Install-Package @PSBoundParameters
+ ~~~~~
+ CategoryInfo          : (0) [Install-Package], Ex
ception
+ FullyQualifiedErrorId : NoMatchFoundForCriteria,Microsoft.PowerShell.PackageManagement.Cmdlets.InstallPackage

PS C:\Users\Administrator> Get-PSRepository
WARNING: Unable to find module repositories.
PS C:\Users\Administrator> Register-PSRepository -Default -InstallationPolicy Trusted -Proxy http://proxy01.extnetworks.
com:8080
PS C:\Users\Administrator> Get-PSRepository
```

Name	InstallationPolicy	SourceLocation
PSGallery	Untrusted	https://www.powershellgallery.com/api/v2

```
PS C:\Users\Administrator>
```

PowerShell モジュール リポジトリが見つからない

PowerShell モジュール リポジトリを登録。
その際に「-Proxy」オプションでプロキシを指定

PowerShell モジュール リポジトリが登録される

改めてWinInetProxyのインストール

```
Administrator: C:\Windows\system32\cmd.exe
PS C:\Users\Administrator> Get-PSRepository

Name                InstallationPolicy SourceLocation
----                -
PSGallery            Untrusted         https://www.powershellgallery.com/api/v2

PS C:\Users\Administrator> Install-Module -Name WinInetProxy -Proxy http://proxy01.extnetworks.com:8080

Untrusted repository
You are installing the modules from an untrusted repository. If you trust this repository, change its
InstallationPolicy value by running the Set-PSRepository cmdlet. Are you sure you want to install the modules from
'PSGallery'?
[Y] Yes  [A] Yes to All  [N] No  [L] No to All  [S] Suspend  [?] Help (default is "N"): y
PS C:\Users\Administrator>
```

PSGalleryを信頼してインストール

- ▶ 先にRegister-PSRepositoryコマンドレットを実行してリポジトリを取得してから、Install-Moduleコマンドレットを実行すると無駄がないかも

WinInetのProxy設定

```
Set-WinInetProxy -ProxySettingsPerUser 0 -ProxyServer http://proxy01.extnetworks.com:8080 `
-ProxyBypass "localhost;127.0.0.1;10.*.*.*;192.168.*.*;*.extnetworks.com;AzHost11;AzHost12;AzStackHCI10"
```



```
Administrator: C:\Windows\system32\cmd.exe - powershell
PS C:\Users\Administrator> Set-WinInetProxy -ProxySettingsPerUser 0 -ProxyServer http://proxy01.extnetworks.com:8080 -ProxyBypass "localhost;127.0.0.1;10.*.*.*;192.168.*.*;*.extnetworks.com;AzHost11;AzHost12;AzStackHCI10"
Start proxy Configuration
Proxy is machine wide
AutoDetect is 0
PACUrl is
ProxyServer is http://proxy01.extnetworks.com:8080
ProxyBypass is localhost;127.0.0.1;10.*.*.*;192.168.*.*;*.extnetworks.com;AzHost11;AzHost12;AzStackHCI10
Entered WriteProxySettingsHelper
Entered WriteProxySettingsHelper

Successfully set proxy
PS C:\Users\Administrator>
```

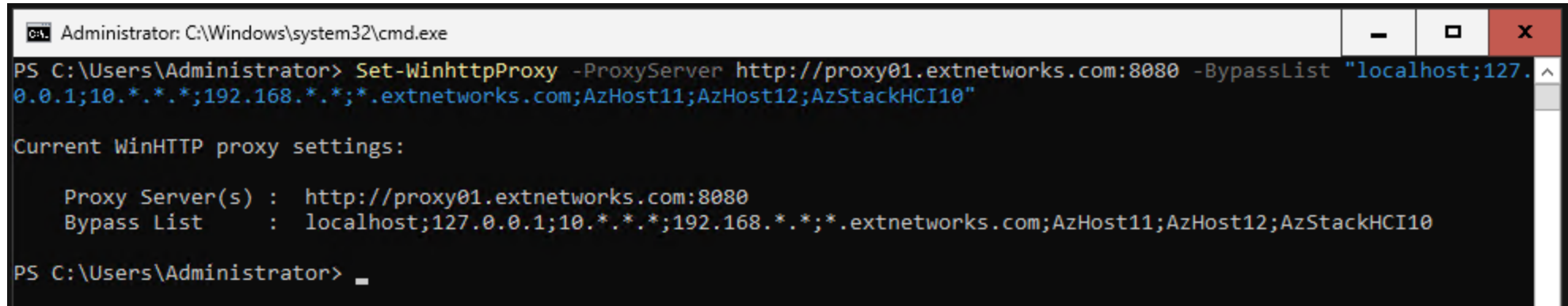
- ▶ 「-ProxySettingsPerUser」 オプションで「0」を指定することでマシン全体に設定
- ▶ 必要な対象をバイパスリストに追加。なお「<Local>」は現在非サポートなので注意

必要なバイパス設定

- ▶ 各サーバーの IP アドレスとクラスターのIPアドレス
- ▶ インフラストラクチャ ネットワークに対して定義した IP アドレス。
Azureポータルでデプロイ時に設定するであろう8個のIPアドレスを定義する必要あり
→ この辺は「192.168.*.*」等でまとめてしまうのも手
- ▶ 各サーバーの NetBIOS 名およびクラスターのNetBIOS 名
→ ベタで指定するか、「AzHost*」とまとめてしまうか
- ▶ 任意のホストまたはサブドメインのドメイン名またはドメイン名と
アスタリスク (*) ワイルドカード
→ 「*.extnetwork.com」のようにドメイン名を指定

WinHttpのプロキシ設定

```
Set-WinhttpProxy -ProxyServer http://proxy01.extnetworks.com:8080 `
-BypassList "localhost;127.0.0.1;10.*.*.*;192.168.*.*;*.extnetworks.com;AzHost11;AzHost12;AzStackHCI10"
```



```
C:\Windows\system32\cmd.exe
PS C:\Users\Administrator> Set-WinhttpProxy -ProxyServer http://proxy01.extnetworks.com:8080 -BypassList "localhost;127.0.0.1;10.*.*.*;192.168.*.*;*.extnetworks.com;AzHost11;AzHost12;AzStackHCI10"

Current WinHTTP proxy settings:

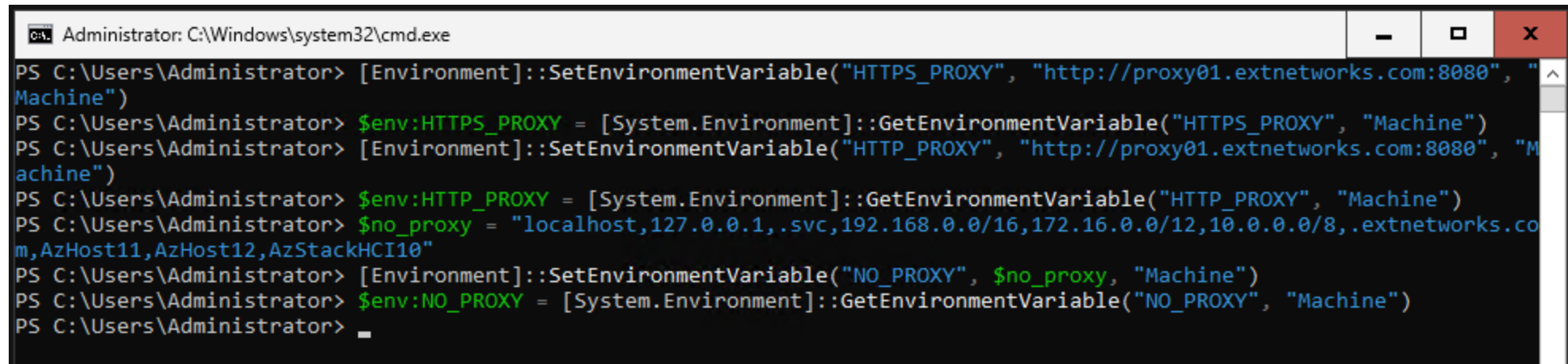
Proxy Server(s) : http://proxy01.extnetworks.com:8080
Bypass List     : localhost;127.0.0.1;10.*.*.*;192.168.*.*;*.extnetworks.com;AzHost11;AzHost12;AzStackHCI10

PS C:\Users\Administrator> _
```

- ▶ netshコマンドでも実施可能（同じ結果になります）
- ▶ 必要な対象をバイパスリストに追加。Set-WinInetProxyコマンドレットで登録した内容と同一のものが推奨
- ▶ 「<Local>」は現在非サポートなので注意

環境変数でのプロキシ設定

```
[Environment]::SetEnvironmentVariable("HTTPS_PROXY", "http://proxy01.extnetworks.com:8080", "Machine")
$env:HTTPS_PROXY = [System.Environment]::GetEnvironmentVariable("HTTPS_PROXY", "Machine")
[Environment]::SetEnvironmentVariable("HTTP_PROXY", "http://proxy01.extnetworks.com:8080", "Machine")
$env:HTTP_PROXY = [System.Environment]::GetEnvironmentVariable("HTTP_PROXY", "Machine")
$no_proxy = "localhost,127.0.0.1,.svc,192.168.0.0/16,172.16.0.0/12,10.0.0.0/8,.extnetworks.com,AzHost11,AzHost12,AzStackHCI10"
[Environment]::SetEnvironmentVariable("NO_PROXY", $no_proxy, "Machine")
$env:NO_PROXY = [System.Environment]::GetEnvironmentVariable("NO_PROXY", "Machine")
```

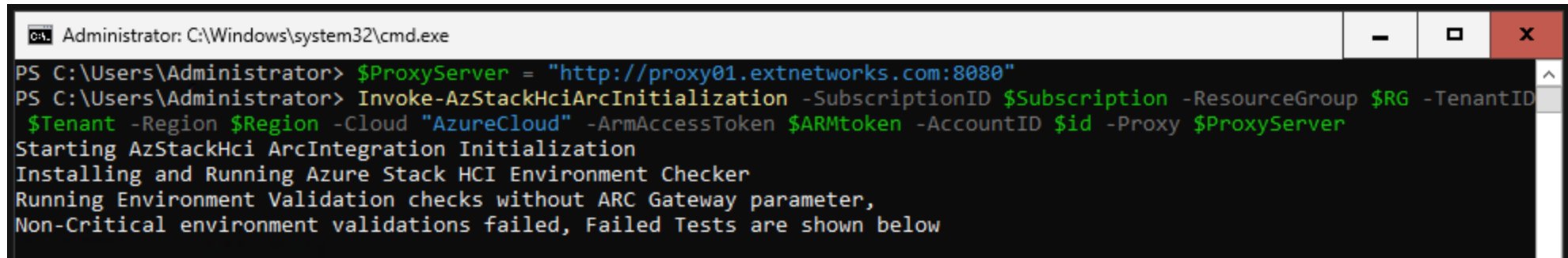


```
C:\Windows\system32\cmd.exe
PS C:\Users\Administrator> [Environment]::SetEnvironmentVariable("HTTPS_PROXY", "http://proxy01.extnetworks.com:8080", "Machine")
PS C:\Users\Administrator> $env:HTTPS_PROXY = [System.Environment]::GetEnvironmentVariable("HTTPS_PROXY", "Machine")
PS C:\Users\Administrator> [Environment]::SetEnvironmentVariable("HTTP_PROXY", "http://proxy01.extnetworks.com:8080", "Machine")
PS C:\Users\Administrator> $env:HTTP_PROXY = [System.Environment]::GetEnvironmentVariable("HTTP_PROXY", "Machine")
PS C:\Users\Administrator> $no_proxy = "localhost,127.0.0.1,.svc,192.168.0.0/16,172.16.0.0/12,10.0.0.0/8,.extnetworks.com,AzHost11,AzHost12,AzStackHCI10"
PS C:\Users\Administrator> [Environment]::SetEnvironmentVariable("NO_PROXY", $no_proxy, "Machine")
PS C:\Users\Administrator> $env:NO_PROXY = [System.Environment]::GetEnvironmentVariable("NO_PROXY", "Machine")
PS C:\Users\Administrator>
```

- ▶ バイパスリストの書き方に注意。「<Local>」は現在非サポート
- ▶ プライベートアドレスは、AKSのためにすべて除外する必要あり
- ▶ ワイルドカードとして「*」はサポートされていないので「.」を使用
- ▶ 「.SVC」はLinux 表記の AKS 内部サービス通信用で、Arc Resource BridgeとAKSに必要

Azure Arcのセットアップ

```
$ProxyServer = "http://proxy01.extnetworks.com:8080"  
Invoke-AzStackHciArcInitialization -SubscriptionID $Subscription -ResourceGroup $RG -TenantID $Tenant `  
-Region $Region -Cloud "AzureCloud" -ArmAccessToken $ARMtoken -AccountID $id -Proxy $ProxyServer
```



```
Administrator: C:\Windows\system32\cmd.exe  
PS C:\Users\Administrator> $ProxyServer = "http://proxy01.extnetworks.com:8080"  
PS C:\Users\Administrator> Invoke-AzStackHciArcInitialization -SubscriptionID $Subscription -ResourceGroup $RG -TenantID $Tenant `  
$Tenant -Region $Region -Cloud "AzureCloud" -ArmAccessToken $ARMtoken -AccountID $id -Proxy $ProxyServer  
Starting AzStackHci ArcIntegration Initialization  
Installing and Running Azure Stack HCI Environment Checker  
Running Environment Validation checks without ARC Gateway parameter,  
Non-Critical environment validations failed, Failed Tests are shown below
```

- ▶ Invoke-AzStackHciArcInitializationコマンドレットの「-Proxy」オプションでプロキシサーバーを指定する

Azure Arcのセットアップ中

```
Administrator: C:\Windows\system32\cmd.exe

HealthCheckSource : ArcInitialization\Connectivity\12fe93b8
Name              : AzStackHci_Connectivity_Proxy_ByPassList_Recommendations
DisplayName       : AZHOST11 Proxy ByPassList Recommendations
Tags              : {[Service, System]}
Title             : Proxy ByPassList Recommendations
Status            : FAILURE
Severity          : WARNING
Description        : Checks that all nodes are configured with recommended proxy bypass list items
Remediation        : https://learn.microsoft.com/en-us/azure-stack/hci/manage/configure-proxy-settings
TargetResourceID  : AZHOST11/Proxy_Bypass_List
TargetResourceName : WinHttp
TargetResourceType : Proxy Settings
Timestamp         : 11/2/2024 12:30:47 PM
AdditionalData     : {[Detail, Missing bypass list entries .svc,10.0.0.0/8,172.16.0.0/12,192.168.0.0/16 in WinHttp on
                    AZHOST11.], [Status, FAILURE], [TimeStamp, 11/02/2024 12:30:47], [Resource,
                    Proxy_ByPassList_Recommendations_AZHOST11]}...
```

ちょいちょいプロキシ設定が登場

```
Administrator: C:\Windows\system32\cmd.exe

Installing agent package
Installation of azcmagent completed successfully
0
Configuring proxy on agent : http://proxy01.extnetworks.com:8080
Config property proxy.url set to value http://proxy01.extnetworks.com:8080
Connecting to Azure using ARM Access Token
Connected to Azure successfully
Microsoft.HybridCompute RP already registered, skipping registration
Microsoft.GuestConfiguration RP already registered, skipping registration
Microsoft.HybridConnectivity RP already registered, skipping registration
Microsoft.AzureStackHCI RP already registered, skipping registration
INFO    Connecting machine to Azure... This might take a few minutes.
INFO    Testing connectivity to endpoints that are needed to connect to Azure... This might take a few minutes.
20% [==> ]
30% [===> ]
```


引き続きAzureポータルからデプロイ実施

Microsoft Azure
リソース、サービス、ドキュメントの検索 (G+/)
Copilot

ホーム > Azure Arc | Azure Stack HCI >

Azure Stack HCI のデプロイ

基本
構成
ネットワーク
管理
セキュリティ
詳細設定
タグ
検証
確認および作成

開始する前に、Active Directory ドメインを準備し、このシステム内のすべてのサーバーを Azure Arc に接続してしてください。 [詳細情報](#)

プロジェクトの詳細

デプロイされているリソースとコストを管理するサブスクリプションを選択します。フォルダーのようなリソースグループを使用して、すべてのリソースを整理し、管理します。

サブスクリプション * ①

リソースグループ * ①

インスタンスの詳細

クラスター名は、基になる 1 台以上のサーバーを管理する代わりに、この Azure Stack HCI システムを全体として管理するために後で使用します。暗号化キー、ローカル管理者の資格情報、BitLocker 回復キーなど、このシステムのシークレットを安全に保存するには、空のキーコンテナを作成します。 [詳細情報](#)

クラスター名 * ①

リージョン * ①

キー コンテナ * ①

☒ 新しいキー コンテナの作成
これにより、このクラスターの新しいキー コンテナが作成されます。

☐ 既存のキー コンテナを選択します
これにより、同じリソースグループ内のキー コンテナを複数のクラスターと

確認および作成
< 前へ
次へ: 構成

バイパスリストが不完全だと……

```
Administrator: C:\Windows\system32\cmd.exe
TCP 192.168.100.11:50389 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50390 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50391 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50392 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50393 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50394 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50395 192.168.100.4:9389 TIME_WAIT
TCP 192.168.100.11:50397 192.168.100.81:8080 ESTABLISHED
TCP 192.168.100.11:50398 192.168.100.81:8080 ESTABLISHED
TCP 192.168.100.11:50399 192.168.100.81:8080 ESTABLISHED
TCP 192.168.100.11:50408 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50409 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50410 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50411 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50419 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50420 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50429 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50430 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50434 192.168.100.12:5985 ESTABLISHED
TCP 192.168.100.11:50435 192.168.100.12:5985 ESTABLISHED
TCP 192.168.100.11:50436 192.168.100.12:5985 ESTABLISHED
TCP 192.168.100.11:50439 192.168.100.11:5985 ESTABLISHED
TCP 192.168.100.11:50440 192.168.100.11:5985 ESTABLISHED
TCP 192.168.100.11:50441 192.168.100.81:8080 TIME_WAIT
TCP 192.168.100.11:50442 192.168.100.11:5985 ESTABLISHED
TCP 192.168.100.11:50443 192.168.100.81:8080 ESTABLISHED
TCP 192.168.100.11:50444 192.168.100.81:8080 ESTABLISHED
TCP [::]:80 [::]:0 LISTENING
TCP [::]:135 [::]:0 LISTENING
TCP [::]:445 [::]:0 LISTENING
```

WS-Managementの通信もプロキシ経由になってしまい、セットアップでエラーが発生する

```
Administrator: C:\Windows\system32\cmd.exe
TCP 192.168.100.11:49991 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:49999 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50000 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50001 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50002 192.168.100.81:8080 ESTABLISHED
TCP 192.168.100.11:50026 192.168.100.4:389 ESTABLISHED
TCP 192.168.100.11:50034 192.168.100.4:135 ESTABLISHED
TCP 192.168.100.11:50035 192.168.100.4:49668 ESTABLISHED
TCP 192.168.100.11:50042 192.168.100.211:55000 TIME_WAIT
TCP 192.168.100.11:50055 192.168.100.211:65000 TIME_WAIT
TCP 192.168.100.11:50080 192.168.100.81:8080 ESTABLISHED
TCP 192.168.100.11:50111 192.168.100.211:65000 ESTABLISHED
TCP 192.168.100.11:50112 192.168.100.211:55000 ESTABLISHED
TCP 192.168.100.11:50184 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50185 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50186 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50187 192.168.100.12:45000 ESTABLISHED
TCP 192.168.100.11:50188 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50189 192.168.100.12:5985 TIME_WAIT
TCP 192.168.100.11:50190 192.168.100.12:5985 TIME_WAIT
```

このあたりも

プロキシサーバーログ

環境変数により、リソースブリッジに対してもプロキシの設定がなされ、プロキシ経由で通信していることが確認できる

```
172.16.10.81 - root@localhost:/var/log/squid VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
1730619288.418 202 192.168.100.213 TCP_TUNNEL/200 7901 CONNECT mcr.microsoft.com:443 - HIER_DIRECT/150.171.70.10 -
1730619288.729 671 192.168.100.11 TCP_TUNNEL/200 6809 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.33 -
1730619290.733 669 192.168.100.11 TCP_TUNNEL/200 6809 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.33 -
1730619291.594 3913 192.168.100.213 TCP_TUNNEL/200 34885655 CONNECT eastasia.data.mcr.microsoft.com:443 - HIER_DIRECT/150.171.69.10 -
1730619291.987 854 192.168.100.12 TCP_TUNNEL/200 6799 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.34 -
1730619292.774 670 192.168.100.12 TCP_TUNNEL/200 6810 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.34 -
1730619293.187 36146 192.168.100.213 TCP_TUNNEL/200 48947434 CONNECT eastasia.data.mcr.microsoft.com:443 - HIER_DIRECT/150.171.69.10 -
1730619293.758 663 192.168.100.12 TCP_TUNNEL/200 6809 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.34 -
1730619297.779 674 192.168.100.12 TCP_TUNNEL/200 6807 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.34 -
1730619297.781 670 192.168.100.12 TCP_TUNNEL/200 6809 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.34 -
1730619298.104 30253 192.168.100.213 TCP_TUNNEL/200 7300 CONNECT mcr.microsoft.com:443 - HIER_DIRECT/150.171.70.10 -
1730619299.133 31040 192.168.100.213 TCP_TUNNEL/200 13145 CONNECT mcr.microsoft.com:443 - HIER_DIRECT/150.171.70.10 -
1730619299.705 668 192.168.100.12 TCP_TUNNEL/200 7003 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.34 -
[root@localhost squid]#
```

といいつつも

プロキシを経由せずにダイレクトアクセスを試みている形跡もある

日付/時刻	送信元	デバイス	宛先	アプリケーション名	結果
2024/11/03 17:03:03	192.168.100.213	02:ec:64:0b:00:00	13.107.246.47 (pmc-prod-afd-endpoint-evdhh8f8byhsezf...	HTTPS	拒否
2024/11/03 17:02:55	192.168.100.213	02:ec:64:0b:00:00	13.107.246.47 (pmc-prod-afd-endpoint-evdhh8f8byhsezf...	HTTPS	拒否
2024/11/03 17:02:51	192.168.100.213	02:ec:64:0b:00:00	13.107.246.47 (pmc-prod-afd-endpoint-evdhh8f8byhsezf...	HTTPS	拒否
2024/11/03 17:02:49	192.168.100.213	02:ec:64:0b:00:00	13.107.246.47 (pmc-prod-afd-endpoint-evdhh8f8byhsezf...	HTTPS	拒否
2024/11/03 17:02:48	192.168.100.213				
2024/11/03 17:02:28	192.168.100.6				
2024/11/03 17:02:28	192.168.100.6				
2024/11/03 17:01:52	192.168.100.220				
2024/11/03 17:01:52	192.168.100.220				
2024/11/03 17:01:52	192.168.100.220				
2024/11/03 17:01:43	192.168.100.213				
2024/11/03 17:01:31	192.168.100.220				
2024/11/03 17:01:29	MSOL_BE41DCF3D65				
2024/11/03 17:01:09	192.168.100.213				
2024/11/03 17:00:57	192.168.100.220				
2024/11/03 17:00:53	192.168.100.213				
2024/11/03 17:00:44	192.168.100.213				
2024/11/03 17:00:40	192.168.100.213				
2024/11/03 17:00:38	192.168.100.213				

```

172.16.10.81 - root@localhost:/var/log/squid VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) ヘルプ(H)
1730621007.994 9092 192.168.100.12 TCP_TUNNEL/ABORTED/200 12887 CONNECT onetech.azureedge.net:443 - HIER_DIRECT/117.18.232.200
1730621007.995 9194 192.168.100.12 TCP_TUNNEL/ABORTED/200 5355 CONNECT go.microsoft.com:443 - HIER_DIRECT/23.32.225.160
1730621009.352 1215 192.168.100.12 TCP_TUNNEL/200 6803 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.5
1730621011.579 75174 192.168.100.213 TCP_TUNNEL/200 7637 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.30
1730621013.436 2307 192.168.100.12 TCP_TUNNEL/200 6802 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.5
1730621014.803 650 192.168.100.12 TCP_TUNNEL/200 6800 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.5
1730621018.877 165073 192.168.100.213 TCP_TUNNEL/200 13032 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.1
1730621020.758 704 192.168.100.11 TCP_TUNNEL/200 7284 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.5
1730621022.634 135030 192.168.100.213 TCP_TUNNEL/200 10012 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.18
1730621022.987 91449 192.168.100.213 TCP_TUNNEL/200 9039 CONNECT g3-prod-dm2-012-sb.servicebus.windows.net:443 - HIER_DIRECT/40.78.147.229
1730621023.706 165649 192.168.100.213 TCP_TUNNEL/200 12914 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.18
1730621024.744 690 192.168.100.11 TCP_TUNNEL/200 6802 CONNECT eastus-shared.prod.warm.ingest.monitor.core.windows.net:443 - HIER_DIRECT/20.62.132.5

```

プロキシ設定が正しければ

Microsoft Azure

リソース、サービス、ドキュメントの検索 (G+/)

Copilot

ホーム > AzStackHCI10

AzStackHCI10 | デプロイ

Azure Stack HCI

検索

更新 デプロイの再開

このデプロイのテンプレートを保存するには、[ここをクリックしてください。](#)

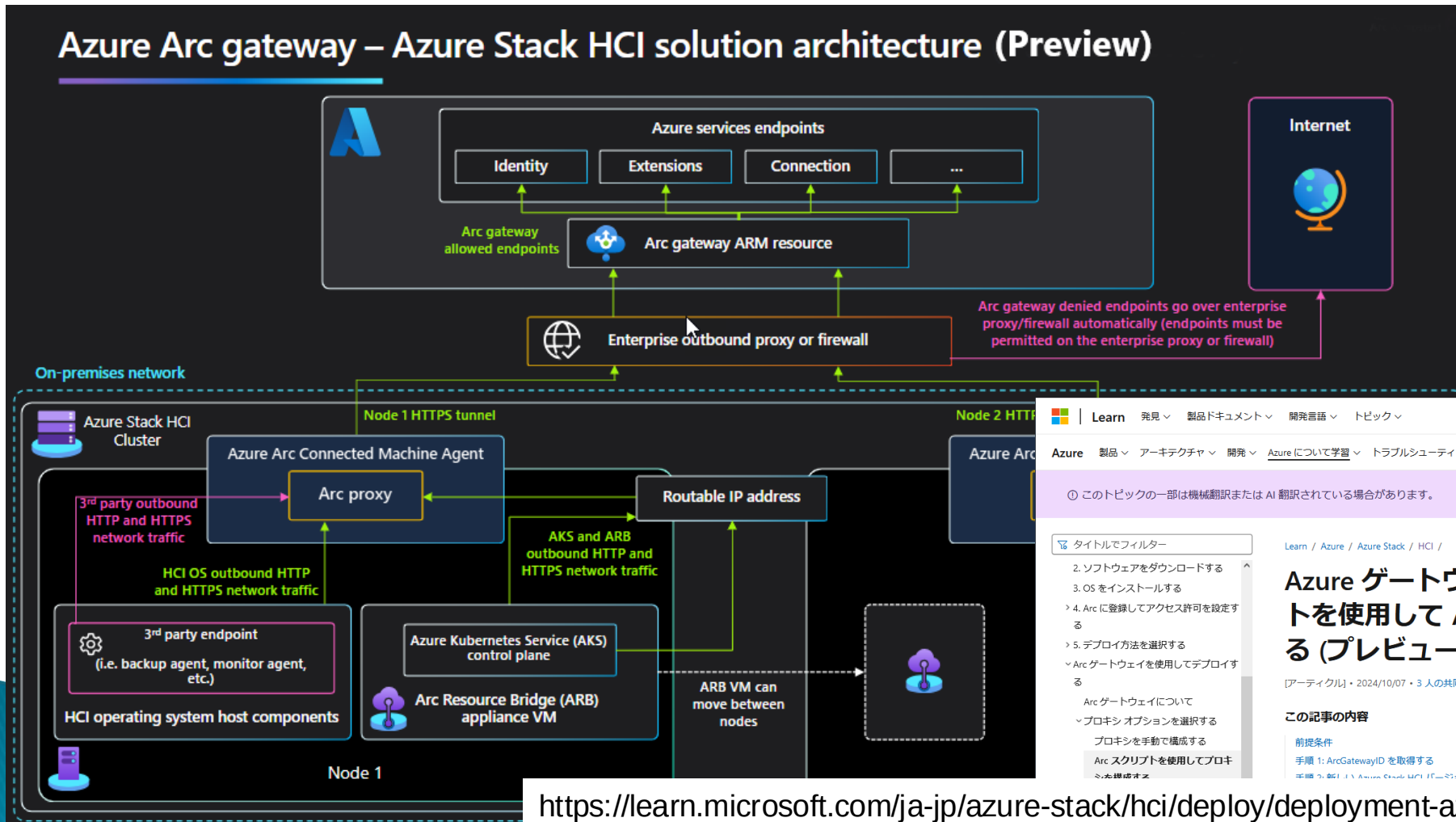
名前	説明	状態	開始時刻	終了時刻
Deploy Azure Stack HCI	Deploy the Azure Stack HCI syste...	成功	2024/11/3 午後2:16	2024/11/3 午後5:13
Check and resolve requi...	Check and resolve deployment r...	成功	2024/11/3 午後2:16	2024/11/3 午後2:18
Validate environment	Validate the environment using t...	成功	2024/11/3 午後2:18	2024/11/3 午後2:18
Resolve requirement	Resolve deployment requirements.	成功	2024/11/3 午後2:34	2024/11/3 午後2:34
Install OpenSSH Client	Install OpenSSH Client.	成功	2024/11/3 午後2:46	2024/11/3 午後2:46
Install OS updates	Install OS updates on all node an...	成功	2024/11/3 午後2:49	2024/11/3 午後2:49
Clean up post update	Clean up post OS update.	成功	2024/11/3 午後2:57	2024/11/3 午後3:18
EvaluateProxyConfigurat...	Check if proxy is enabled on the ...	成功	2024/11/3 午後3:18	2024/11/3 午後3:18
Validate network setting:	Validate network settings for serv...	成功	2024/11/3 午後3:18	2024/11/3 午後3:18
Configure settings on se	Configure settings on servers.	成功	2024/11/3 午後3:18	2024/11/3 午後3:32
Adjust the number of inf	Scale the number of infrastruc...	成功	2024/11/3 午後3:32	2024/11/3 午後3:32
Prepare servers for secur	Prepare servers to apply WDAC s...	成功	2024/11/3 午後3:32	2024/11/3 午後3:32
Apply security settings o	Apply security settings on servers.	成功	2024/11/3 午後3:32	2024/11/3 午後3:33
Configure networking	Configure the host networking s...	成功	2024/11/3 午後3:44	2024/11/3 午後3:52

本例の所要時間は
2h57m

デプロイ完了 (2hとか3hとか.....)

プロキシ設定めんどくさい……

▶ そんなあなたにAzure Arc Gateway (たぶん)



まとめ

- ▶ プロキシ環境下でも Azure Stack HCI は導入可能です。
- ▶ プロキシの設定をどれか一つでも忘れると、デプロイ失敗するので注意。
あとバイパス設定も重要。忘れるとデプロイ失敗するので必ず設定を確認
- ▶ Azure Arc Gateway もちょっと期待。検証実施予定（次回かな？）
- ▶ よいAzure Stack HCI ライフを！

リファレンス

- ▶ Azure Stack HCI のドキュメント
<https://learn.microsoft.com/ja-jp/azure-stack/hci/>
- ▶ Azure Stack HCI バージョン 23H2 のプロキシ設定を構成する
<https://learn.microsoft.com/ja-jp/azure-stack/hci/manage/configure-proxy-settings-23h2>

ご清聴、ありがとうございました。

**ご不明点等ございましたら、
お気軽にご質問ください。**