

LAB02: Onboarding Windows and Linux Servers to Azure Arc

Student Lab Manual

Table of Contents

Exercise 1 - Examine the existing Arc-connected machines

Task 1 - Use the Azure portal to examine you Arc-enabled machines inventory

Task 2 - Examine the virtual machines that you will Arc-enable

Exercise 2 - Onboard a Windows and a Linux machine to Azure Arc

Task 1 - Generate a script to automate the download and installation of the Azure Arc connected machine agent for a Windows machine, and connect the machine to Azure Arc

Task 2 - Connect a Linux machine to Azure Arc using the direct internet access (without a proxy)

Task 3 - Connect a linux machine to Azure Arc using a proxy and Arc Gateway

====

Exercise 1 - Examine the existing Arc-connected machines

Objective

The deployment process that you have walked through in Lab01 should have set up a number of VMs running on Hyper-V in the ArcBox-Client machine. Two of these machines have been connected to Azure Arc for you by the set script. In this exercise you will verify that these two machines are indeed Arc-enabled and you will identify the other machines that you will Arc-enable.

Estimated Time to Complete This Lab

10 minutes

Explanation

The Arc-connected machines can be viewed on the Azure portal.

====

Task 1: Use the Azure portal to examine you Arc-enabled machines inventory

- [] Enter "Machines - Azure Arc" in the top search bar in the Azure portal and select it from the displayed services.

Machines - Azure Arc

All Services (99+) Marketplace (1) Documentation (99+) Resources (0) Resource Groups (0)

Azure Active Directory (0)

Services

Machines - Azure Arc (2)

Azure Database for MySQL servers
Kubernetes - Azure Arc
SQL Server databases - Azure Arc
SQL managed instances - Azure Arc

Virtual machines
Azure Active Directory
Azure Cosmos DB

Marketplace

Virtual machine - Azure Arc (preview)

Documentation

Quickstart - Connect hybrid machine with Azure Arc-enabled serv...
Connect hybrid machines to Azure at scale - Azure Arc
Connect hybrid machines to Azure using a deployment script - Az...
Overview of the Azure Connected Machine agent - Azure Arc
Azure Arc-enabled server configurations - Azure Architecture Cen...
Azure Connected Machine agent deployment options - Azure Arc
Connected Machine agent network requirements - Azure Arc
Connect machines at scale using Group Policy with a PowerShell s...

Continue searching in Azure Active Directory

Give feedback

- [] You should see the machines that are connected to Arc already: *Arcbox-Ubuntu-01* and *ArcBox-Win2K25*.

Home > Azure Arc

Azure Arc | Machines

Overview All Azure Arc resources

Filter for any field... Subscription equals all Resource group equals all Location equals all

Have Windows Server 2012 machines? Keep machines reaching the end of their support lifecycle protected by enabling Windows Server Update Services (WSUS).

Machines

Kubernetes clusters

Host environments

Data services

Internet of Things (IoT)

Application services

Licenses

Management

DevOps

Showing 1 to 2 of 2 records.

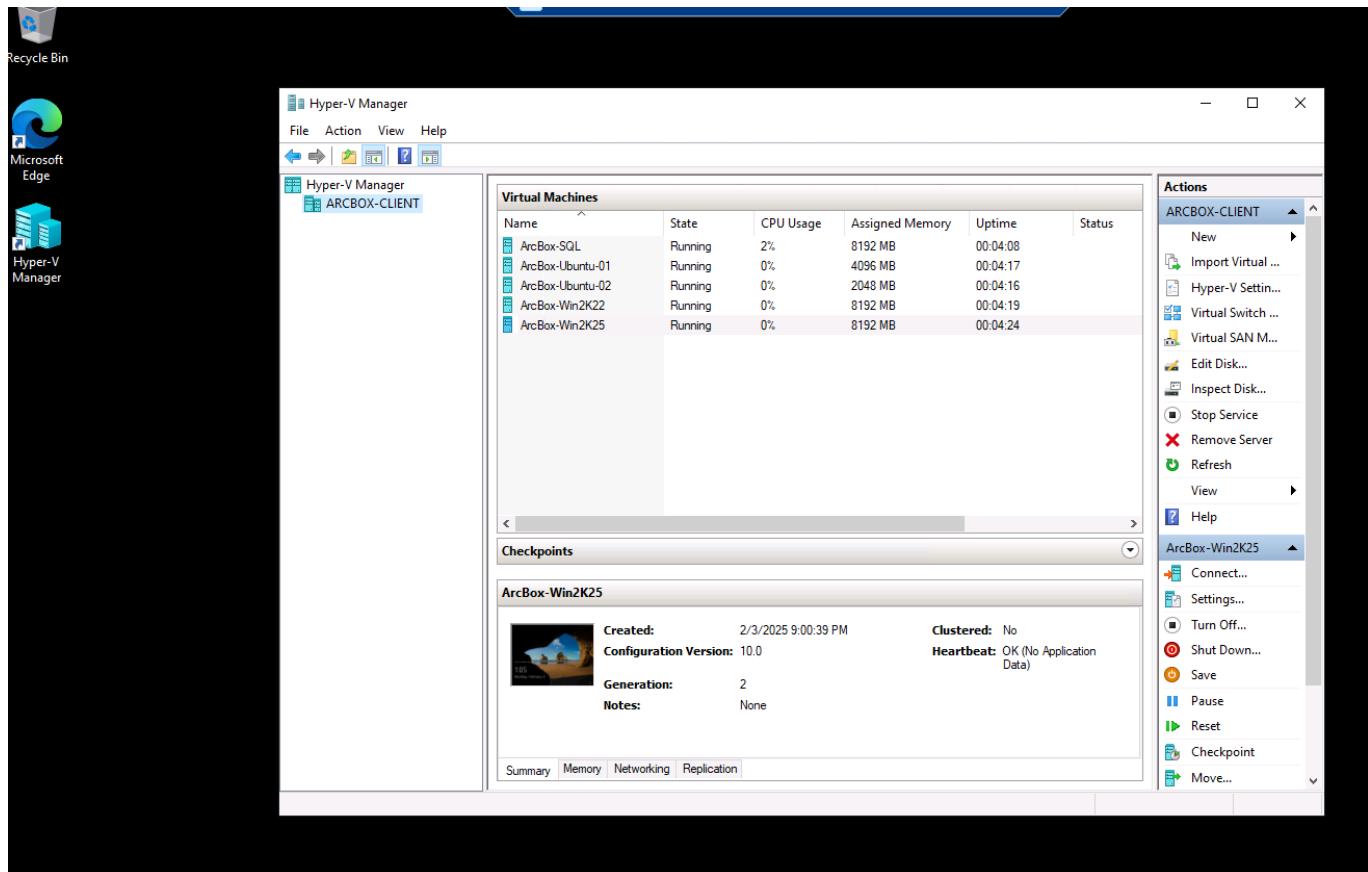
Name ↑	Kind ↑↓	Arc agent status ↑↓
Arcbox-Ubuntu-01		Connected
Arcbox-Win2k25		Connected

Task 1 has been completed

====

Task 2: Examine the virtual machines that you will Arc-enable

1. [] You want to connect 2 more machines running as VMs in the ArcBox-Client. You can see these (ArcBox-Win2K22 and ArcBox-Ubuntu-02) by running the Hyper-V Manager in the ArcBox-Client (after you have connected to it with RDP as explained earlier in Lab01).



Task 2 has been completed

====

Exercise 2 - Onboard a Windows and a Linux machine to Azure Arc

Objective

In this exercise you will onboard the Windows machine ArcBox-Win2K22 and the Linux machine ArcBox-Ubuntu-02 to Azure Arc using the Service Principal onboarding method.

Estimated Time to Complete This Lab

60 minutes

Explanation

Connecting machines in your hybrid environment directly with Azure can be accomplished using different methods, depending on your requirements and the tools you prefer to use. One method to connect the machines to Azure Arc is to use a Microsoft Entra ID service principal. This service principal is a special limited management identity that has only the minimum permission necessary to connect machines to Azure. This method is safer than using a higher privileged account like a Subscription Contributor and follows access control security best practices. The service principal is used only during onboarding; it is not used for any other purpose.

The Azure Connected Machine agent for Linux and Windows communicates outbound securely to Azure Arc over TCP port 443. By default, the agent uses the default route to the internet to reach Azure services. You can optionally configure the agent to use a proxy server if your network requires it. There are a number of URLs that must be available in order to install and use the Connected Machine agent and onboard the machine to Azure Arc. The proxy must be configured to allow all these URLs including ones containing wildcards. However, the number of URLs can be consolidated using the Azure Arc Gateway service which reduces the number of URLs to a small set of endpoints that need to be allowed by the proxy.

In Task 1 of this exercise you will learn how to onboard a Windows machine to Azure Arc using direct internet access (without a proxy).

You will then learn how to onboard a Linux machine to Azure Arc using either the direct internet access (without a proxy) in task 2 or using a proxy together with the Azure Arc Gateway in task 3.

[!Important] For onboarding the Linux machine to Azure Arc, it is important to follow either Task 2 or Task 3.

Documentation

[Azure Connected Machine agent deployment options.](#)

[Connect hybrid machines to Azure at scale.](#)

[Connected Machine agent network requirements](#)

[Simplify network configuration requirements with Azure Arc gateway](#)

====

Task 1: Generate a script to automate the download and installation of the Azure Arc connected machine agent for a Windows machine, and to connect the machine to Azure Arc

1. [] From the Azure portal go to the "Machines - Azure Arc" page and select "Add/Create" at the upper left, then select "Add a machine".

The screenshot shows the Azure Arc Machines dashboard. On the left, there's a navigation menu with 'Machines' selected. In the center, there's a search bar and a 'Resource group equals all' filter. A prominent red box highlights the 'Add a machine' button, which is described as 'Connect and manage an existing server or virtual machine from any of your environments'. Below it are two other options: 'Create a machine in a connected host environment' and 'Create a virtual machine in your connected host environments'. At the bottom, there's a table listing a single machine named 'Arcbox-Ubuntu-01'.

2. [] In the next screen, go to "Add multiple servers" and click on "Generate script".

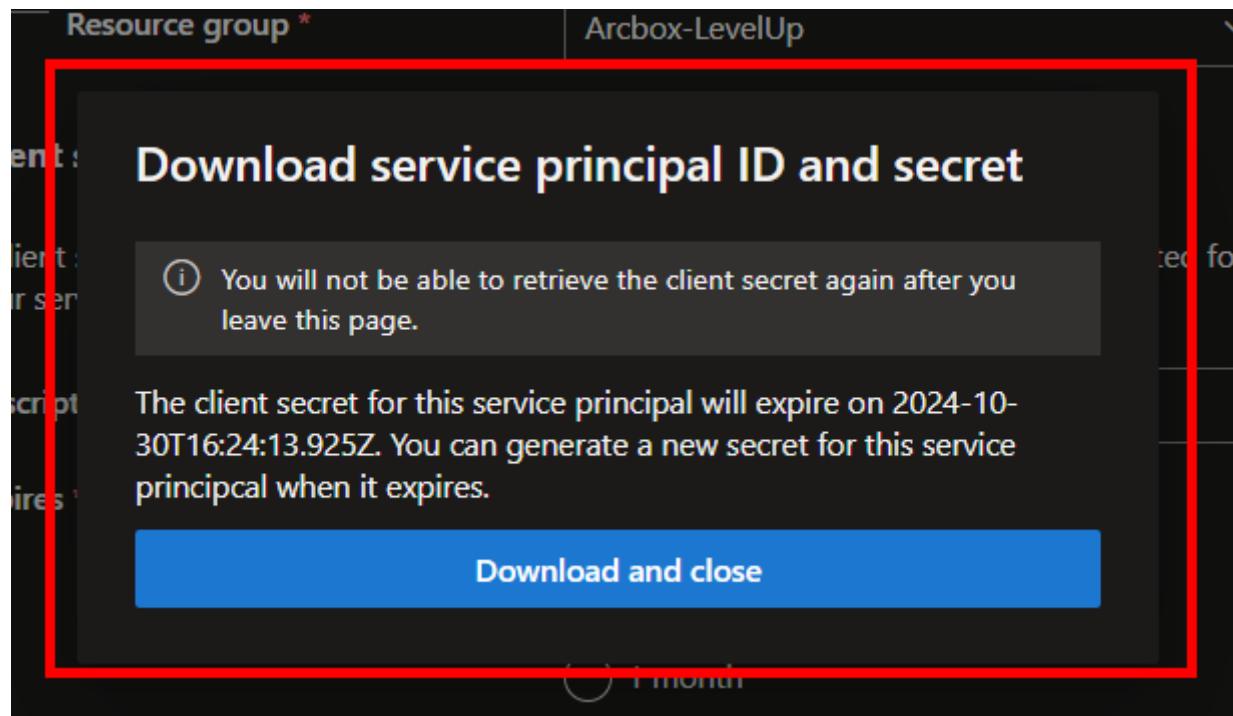
This screenshot shows the 'Add servers with Azure Arc' page. It features five cards: 'Add a single server', 'Add multiple servers' (which is highlighted with a red box), 'Add Windows Server with installer', 'Add servers from AWS', and 'Add servers from Update Management'. Each card has a 'Generate script' button at the bottom.

3. [] Fill in the Resource Group, Region, Operating System (Windows), keep Connectivity as "Public endpoint". Keep the *Connect SQL Server* option ticked. In the Authentication section click on the "Create new" link under the "Service Principal" dropdown and then enter a name (ex:ArcOnboarding) and choose "Azure Connected Machine Onboarding" as the role and click "Create".

The left side of the image shows the 'Add multiple servers with Azure Arc' configuration page. It includes fields for 'Subscription' (Visual Studio Enterprise Subscription), 'Resource group' (ArcBox, highlighted with a red box), 'Region' (US Central US), 'Operating system' (Windows), 'SQL Server' (checkbox checked), and 'Connectivity method' (radio button selected for 'Public endpoint'). A red arrow points from the 'Resource group' field on this page to the 'Resource group' field on the right.

The right side of the image shows the 'New Azure Arc service principal' creation page. It has sections for 'Service principal details' (Name: ArcOnboarding, highlighted with a red box), 'Client secret' (Description: v1, Expires: 1 day), and 'Role assignment' (Roles: Azure Connected Machine Onboarding, highlighted with a red box).

4. [] Click on "Download and close" to save your new service principal ID and secret as a text file (you will use this in a future step).



5. [] Select your newly created "ArcOnboarding" service principal from the dropdown menu.

The screenshot shows the "Authentication" section. It includes a descriptive text about using a service principal for onboarding multiple servers. Below is a dropdown menu for "Service principal" with "Arconboarding" selected and a "Create new" option. The entire section is highlighted with a red box.

6. [] Then click on "Download and run script" and scroll down till you see the "Download" button, and next to it click on the icon that looks like two pieces of paper stacked on each other to copy your onboarding script.

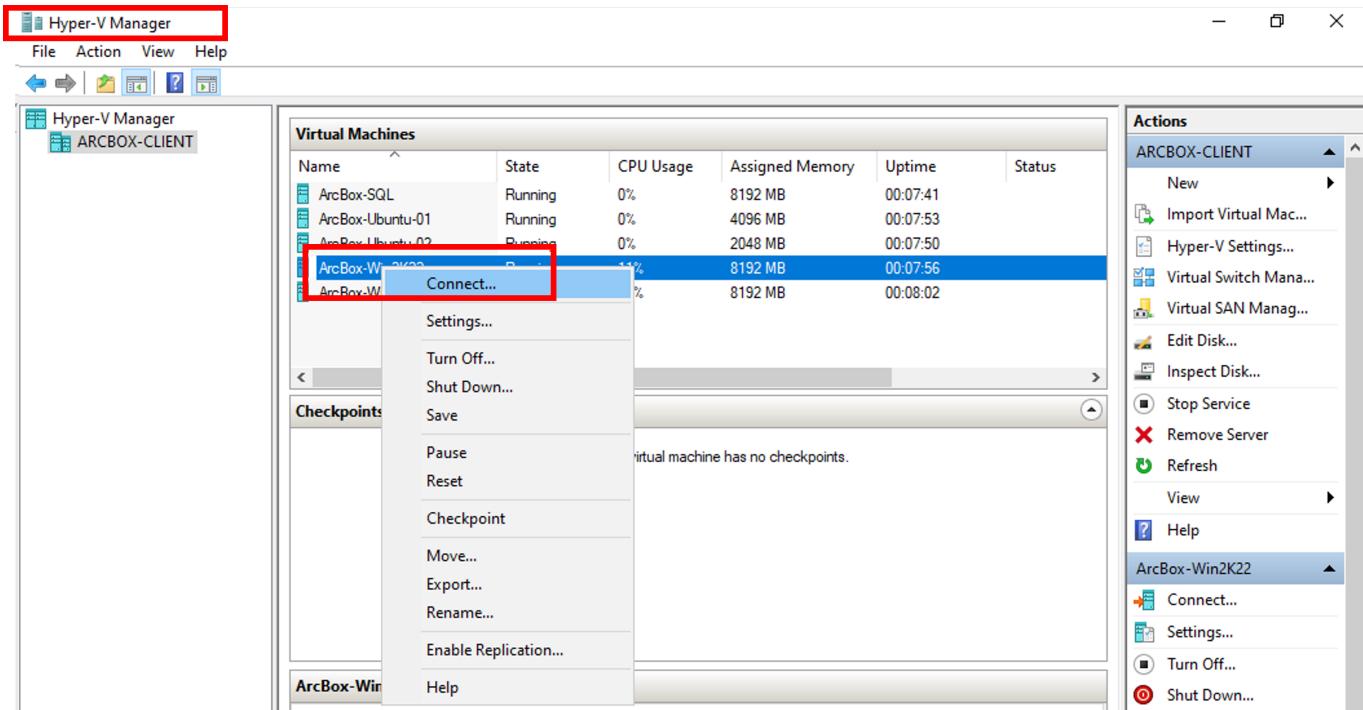
```

1 $global:scriptPath = $myinvocation.mycommand.definition
2
3 function Restart-AsAdmin {
4     $pwshCommand = "powershell"
5     if ($PSVersionTable.PSVersion.Major -ge 6) {
6         $pwshCommand = "pwsh"
7     }
8
9     try {
10        Write-Host "This script requires administrator permissions to install the Azure Connected Machine
11        Agent. Attempting to restart script with elevated permissions..."
12        $arguments = "-NoExit -Command `"$& '$scriptPath`"""
13        Start-Process $pwshCommand -Verb runAs -ArgumentList $arguments
14        exit 0
15    } catch {
16        throw "Failed to elevate permissions. Please run this script as Administrator."
17    }
18
19    try {
20        if (-not ([Security.Principal.WindowsPrincipal] [Security.Principal.WindowsIdentity]::GetCurrent()).
21            IsInRole([Security.Principal.WindowsBuiltInRole]::Administrator)) {
22            if ([System.Environment]::UserInteractive) {
23                Restart-AsAdmin
24            } else {
25                throw "This script requires administrator permissions to install the Azure Connected Machine
26                Agent. Please run this script as Administrator."
27            }
28        }
29
30        # Add the service principal application ID and secret here

```

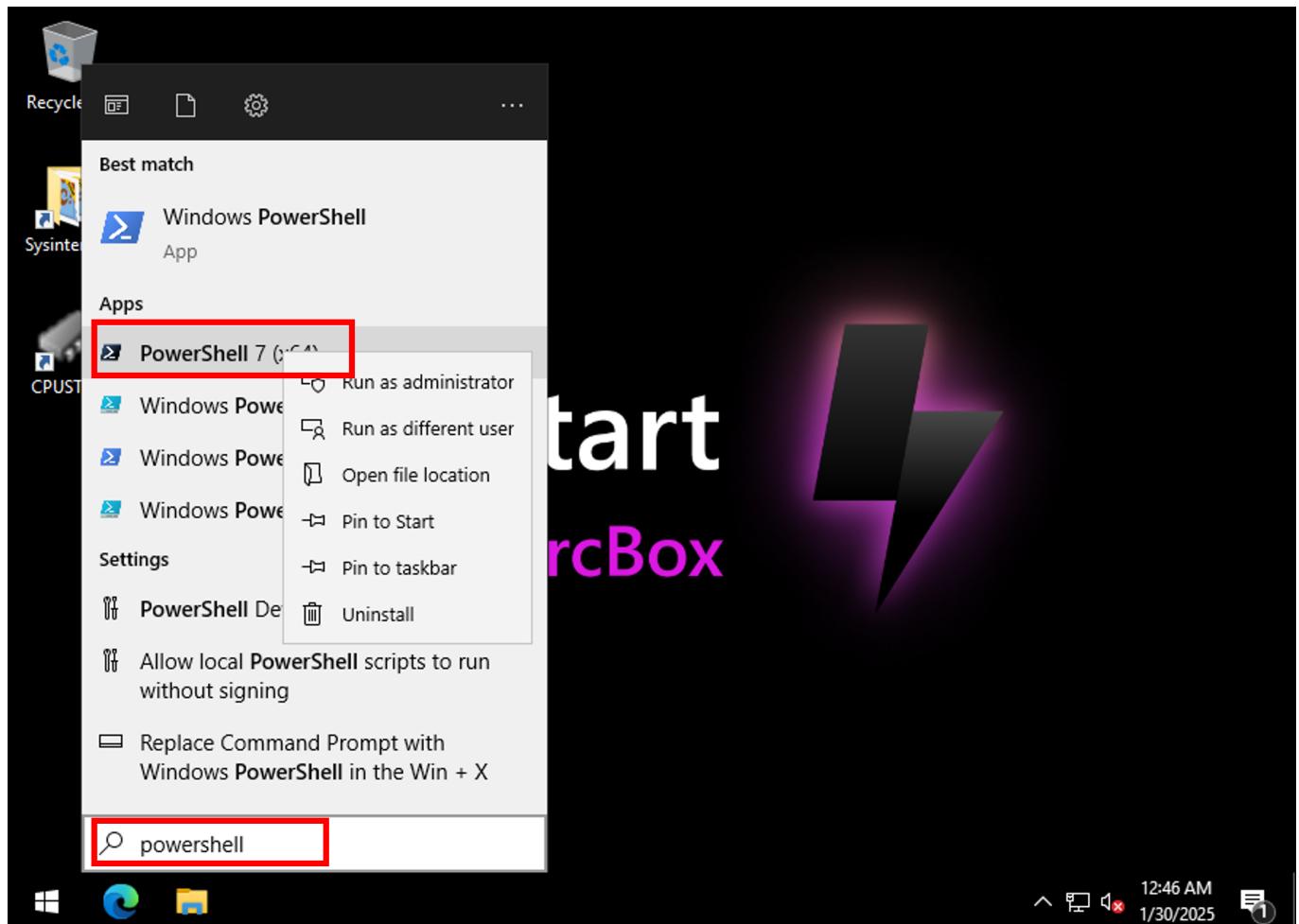
[Download](#)

7. [] Go to the ArcBox-Client machine via RDP and from Hyper-V manager right-click on the *ArcBox-Win2K22* VM and click "Connect" (Administrator default password is JS123!!). Next you will need to open an editor (Visual Studio Code or Notepad for example) and paste the content of the onboarding script.



8. [] Fill in the Service Principal Id, if it is not already populated, and the secret from the text file that was downloaded above into the script where you see "ENTER SECRET HERE". Start Powershell 7 and paste

the script (after you have added the secret). This will execute the script.



```
Administrator: PowerShell 7 (x64)
>>> $ogBody | ConvertTo-Json) | out-null;
>>> Write-Host -ForegroundColor red $_.Exception;
>> }
VERBOSE: Installing Azure Connected Machine Agent
VERBOSE: PowerShell version: 7.4.6
VERBOSE: Total Physical Memory: 8192 MB
VERBOSE: .NET Framework version: 4.8.4161
VERBOSE: Checking if this is an Azure virtual machine
VERBOSE: Error The request was canceled due to the configured HttpClient.Timeout of 1 seconds elapsing. checking if we are in Azure
VERBOSE: Downloading agent package from https://gb1.his.arc.azure.com/azcmagent/latest/AzureConnectedMachineAgent.msi to C:\Users\ADMINI~1\AppData\Local\Temp\2\AzureConnectedMachineAgent.msi
VERBOSE: Installing agent package
Installation of azcmagent completed successfully
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% [==>]
    INFO    Creating resource i
e Resource ID=/subscriptions/
t HybridCompute/machines/ArcBox-Win2K22
  60% [=====>]
  80% [=====>]
100% [=====]
    INFO    Connected machine to Azure
```

9. [] On successful completion a message is displayed to confirm the machine is connected to Azure Arc.
We can also verify that our Windows machines are all now connected in the Azure portal (Machines - Azure Arc).

The screenshot shows the Azure Arc | Machines page. On the left, there's a navigation sidebar with options like Overview, All Azure Arc resources, and Azure Arc resources (Machines is selected). The main area displays a table of connected machines. The table has columns for Name, Kind, Arc agent status, and Resource group. Three machines are listed: Arcbox-Ubuntu-01 (Kind: VM), ArcBox-Win2K22 (Kind: VM), and ArcBox-Win2K25 (Kind: VM). All three machines are marked as Connected and belong to the ArcBox resource group. A red box highlights the 'Machines' section in the sidebar and the list of three machines.

10. [] **Optional:** If you intend to attempt the Arc-enabled SQL server labs within this workshop, then you might want to run the same script inside the *ArcBox-SQL* VM to onboard this server ready for the coming labs, provided that you had selected the SQL onboarding option ticked when you created the script in the portal.

Task 1 has been completed

====

Task 2: Connect a Linux machine to Azure Arc using the direct internet access (without a proxy)

[!Important] Follow either this task or the following task to onboard the Linux machine.

1. [] From the Azure portal go to the "Machines - Azure Arc" page and select "Add/Create" at the upper left, then select "Add a machine".

The screenshot shows the Azure Arc | Machines page. The 'Machines' section in the sidebar is highlighted with a red box. In the main area, there's a callout box pointing to the 'Add a machine' button, which is also highlighted with a red box. The callout box contains the text: 'Connect and manage an existing server or virtual machine from any of your environments'. Below this, there's another option: 'Create a machine in a connected host environment'.

2. [] In the next screen, go to "Add multiple servers" and click on "Generate script".

The screenshot shows the 'Add servers with Azure Arc' interface. It has four main sections:

- Add a single server**: This section is described as generating a script for one server at a time.
- Add multiple servers**: This section is highlighted with a red box. It describes generating a script for multiple servers using a service principal.
- Add Windows Server with installer**: This section involves using an installer to onboard a single Windows Server.
- Add servers from AWS**: This section connects Amazon Web Services accounts to Azure Arc.

Each section has a 'Generate script' button and a 'Learn more' link.

3. [] Fill in the required details but this time choose Linux for the Operating System box and choose the service principal in the "Authentication" dropdown that you created in the previous exercise. Then download the script to your local machine (or you can copy the content into the clipboard).

4. [] Make sure that the Service Principal Id is filled in, then add the client secret to the script using your editor. Also add the following 3 lines to the script **just below the last export statement** (to allow onboarding of Azure linux machines):

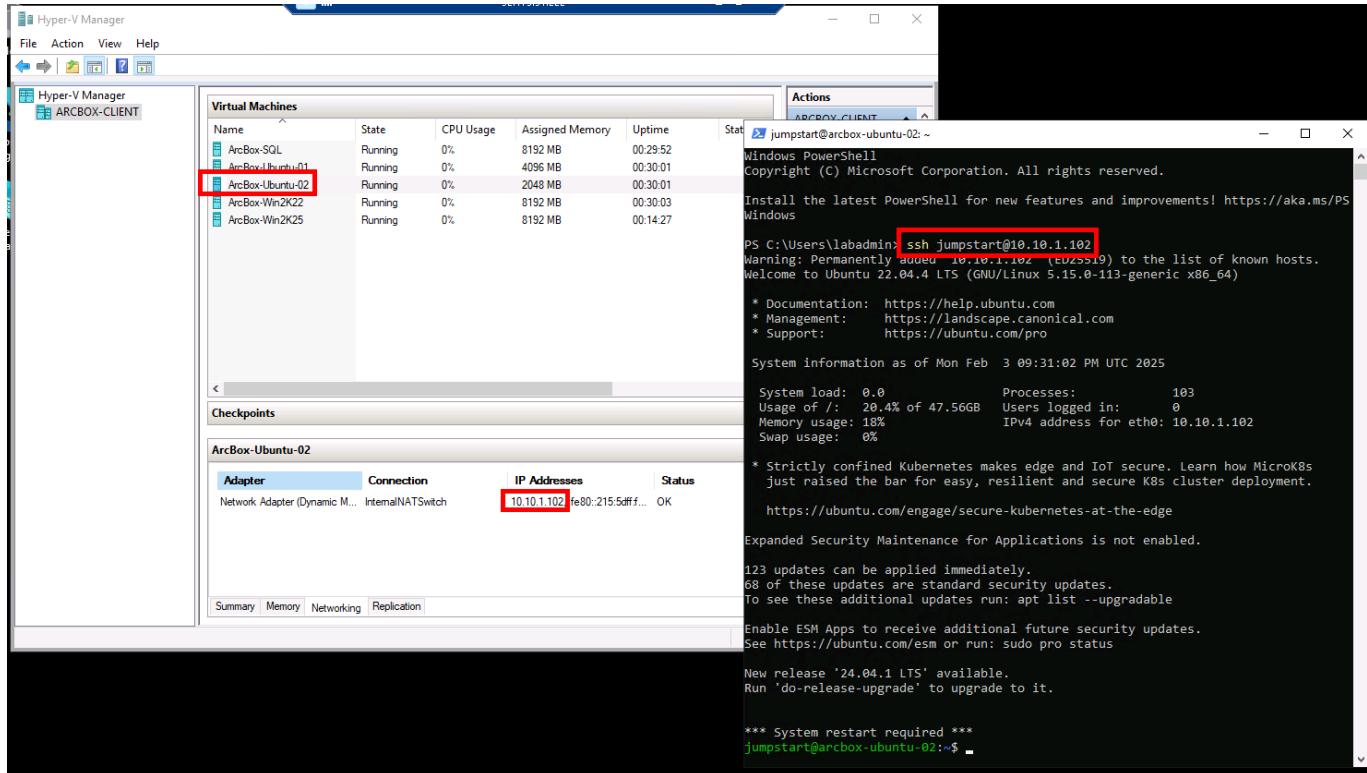
```
sudo ufw --force enable
sudo ufw deny out from any to 169.254.169.254
sudo ufw default allow incoming
```

5. [] Connect the ArcBox-Client machine, and from the "Networking" tab on Hyper-v Manager find the IP address of the Linux machine.

The screenshot shows the Hyper-V Manager application. On the left, the 'Virtual Machines' list is displayed, showing several running VMs. One VM, 'ArcBox-Ubuntu-02', is selected and highlighted with a red box. On the right, the 'Actions' pane for 'ARCBOX-CLIENT' is open, showing various management options. Below it, the 'ArcBox-Ubuntu-02' details pane is shown, with the 'Networking' tab selected. A red arrow points to the 'IP Addresses' table, which lists the IP address '10.10.1.102'. Another red box highlights this IP address.

Adapter	Connection	IP Addresses	Status
Network Adapter (Dynamic M...)	InternalNATSwitch	10.10.1.102, fe80::215:5dff...	OK

6. [] SSH into the ArcBox-Ubuntu-02 machine using "PowerShell". If you are prompted for a password then use *JS123!!*:



```
ssh jumpstart@<Enter IP Address of the Linux machine>
```

7. [] Create an empty onboarding script file using the nano editor, and paste the script content from your local machine. Once you are in the editor, navigate using the arrow keys and not the mouse.

```
nano onboardingscript.sh
```

8. [] Save the file (Ctrl-O then Enter) and exit (Ctrl-X). Now you can run the script:

```
sudo bash ./onboardingscript.sh
```

9. [] Wait for the script to finish successfully. A message should confirm that the machine is now Arc-connected. You can also verify that our Linux machine is connected in the Azure portal (Machines - Azure Arc).

```
60% [=====>      ]
80% [=====>      ]
100% [=====]
INFO  Connected machine to Azure
```

Task 2 has been completed

====

Task 3: Connect a linux machine to Azure Arc using a proxy and Arc Gateway

[!Important] Follow this task if you have not already onboarded the Linux machine using the previous task.

1. [] From the Azure portal search for *Arc gateways* and select from the search results, then click on the *Create* button.

The screenshot shows the Azure portal search interface. The search bar at the top contains the text "arc gateways" and is highlighted with a red box. Below the search bar, there are four filter buttons: "All" (selected), "Services (27)", "Documentation (99+)", and "More (4)". Under the "Services" heading, the "Arc gateways" option is listed and highlighted with a red box. Below it are other service options: "APC Gateways", "NAT gateways", and "Application gateways". Under the "Documentation" heading, several articles are listed: "Simplify network configuration requirements with Azure Arc gateway (preview) - A...", "Overview of Azure Arc gateway for Azure Local, version 23H2 (preview) - Azure Lo...", "Simplify network configuration requirements with Azure Arc gateway (preview) - A...", and "How to simplify network configuration requirements with Azure Arc gateway (Publ...". At the bottom of the screenshot is a dark footer bar with a blue arrow icon and the text "Continue searching in Microsoft Entra ID".

2. [] Fill in the details of the resource group name, choose a name for your Arc Gateway and choose the region of your lab resources. Then move to the *Review + create* tab and create the gateway.

Create an Arc gateway resource

Basics Tags Review + create

For customers who use enterprise firewalls or proxies to manage outbound traffic, the Arc gateway enables you to onboard infrastructure to Azure via Arc with only 7 endpoints required.

Resource details

Select a name, location, subscription, and resource group for your Arc gateway resource. Your Arc gateway resource does not need to be in the same subscription, resource group, or location as the Arc-enabled resources that will be associated with the Arc gateway resource.

Subscription * ⓘ

Resource group * ⓘ Select existing item...
Create new

Name *

Location * (Europe) UK South

3. [] The deployment will take few minutes and when it is done choose *Go to resource*.

gatewaytest1.ArcGateway.1739390188385 | Overview

Deployment

Search X ⏪

Delete Cancel Redeploy Download Refresh

Overview Inputs Outputs Template

✓ Your deployment is complete

Deployment name : gatewaytest1.ArcGateway.1739390188385
Subscription :
Resource group : ArcBox

Deployment details

Next steps

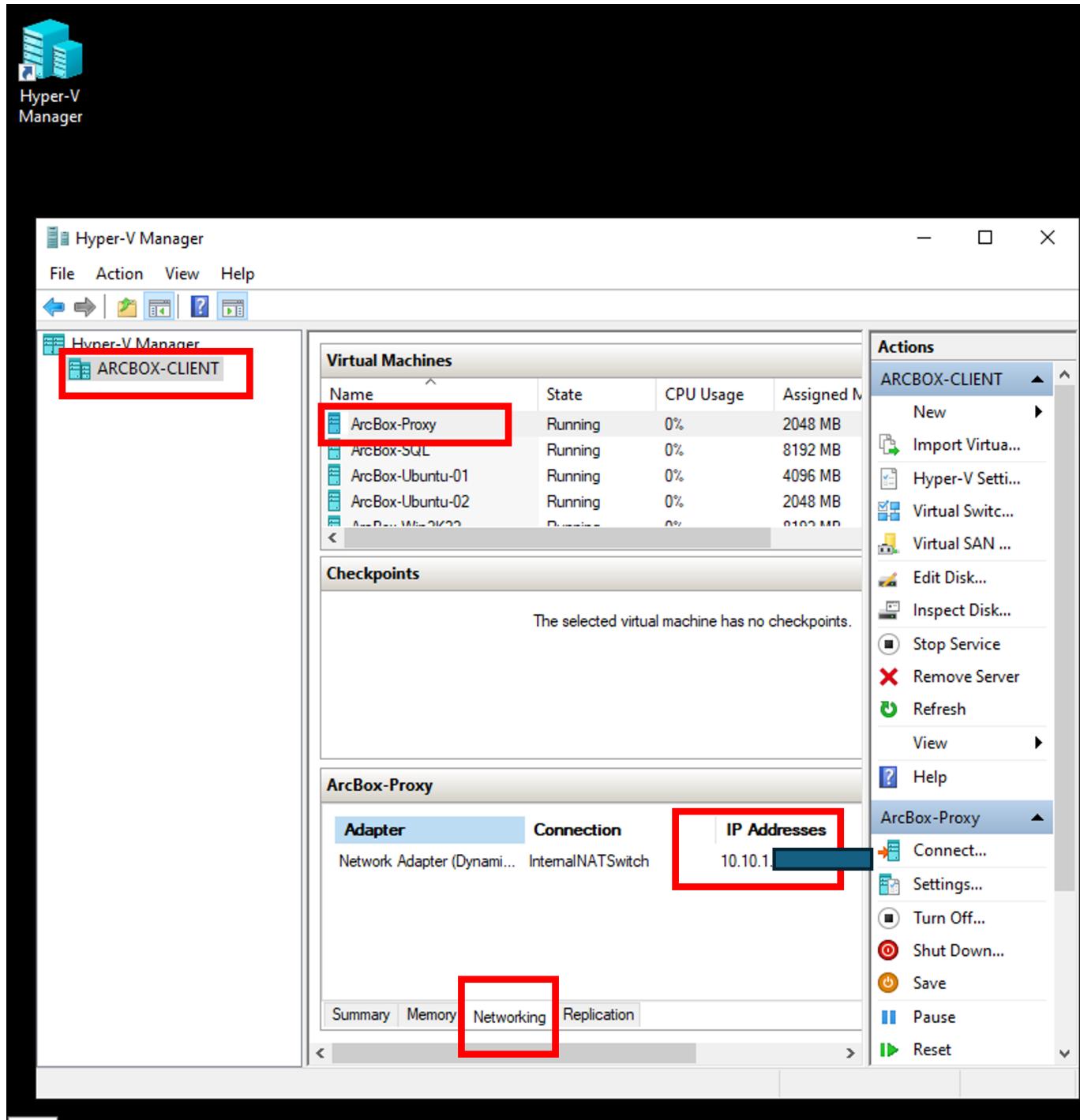
Go to resource

4. [] Find the Gateway URL and make a note of it to be used later.

Resource ID : /subscriptions//ArcBox/provider...
Gateway URL : 87c.gw.arc.azure.com

JSON View

5. [] From the *ArcBox-Client* machine open *Hyper-V Manager*, click the *ArcBox-Client* server, click the *ArcBox-Proxy* under Virtual Machines. From the *Networking* tab find the IPV4 address of the proxy. Make a note of this IP address.



6. [] From the *ArcBox-Client* machine use Powershell 7 to ssh into the proxy server *ArcBox-Proxy* using the IP address you found in the previous steps.

```
ssh jumpstart@<Enter IP Address of the proxy machine>
```

7. [] Once you have logged into the proxy machine edit the *whitelist.txt* file which contains the list of links allowed by the proxy. These should be similar to the links stated in [Arc Gateway documentation](#). If any of the links is missing then add it to the whitelist. Once you are in the editor, navigate using the arrow

keys and not the mouse. Make sure that you fill in the Arc gateway URL that you have from the earlier steps, and also the region of your deployment.

```
sudo nano /etc/squid/whitelist.txt
```

```
jumpstart@proxy: ~
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-131-generic x86_64)

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

System information as of Thu Feb 13 07:03:29 PM UTC 2025

 System load:  0.02          Processes:           101
 Usage of /:   20.0% of 47.56GB  Users logged in:    0
 Memory usage: 12%          IPv4 address for eth0: 10.10.1.104
 Swap usage:   0%

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

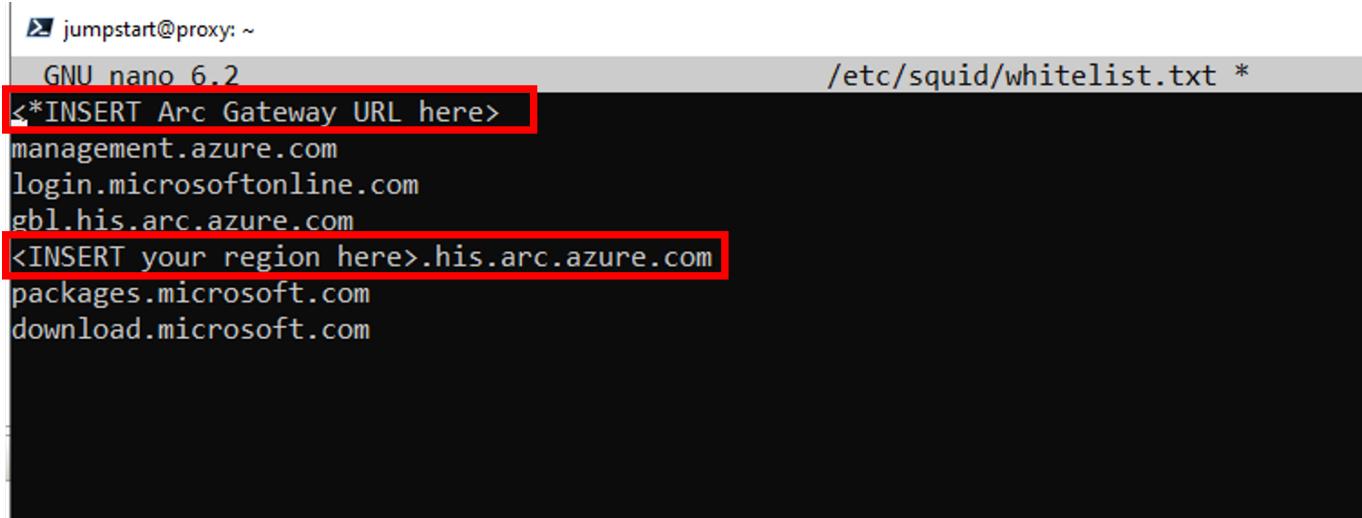
Expanded Security Maintenance for Applications is not enabled.

58 updates can be applied immediately.
1 of these updates is a standard security update.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Thu Feb 13 19:00:41 2025 from 10.10.1.1
jumpstart@proxy:~$ sudo nano /etc/squid/whitelist.txt
```



```
jumpstart@proxy: ~
GNU nano 6.2
/etc/squid/whitelist.txt *
<*INSERT Arc Gateway URL here>
management.azure.com
login.microsoftonline.com
gbl.his.arc.azure.com
<INSERT your region here>.his.arc.azure.com
packages.microsoft.com
download.microsoft.com
```

8. [] Save the file (Ctrl-O then Enter) and exit (Ctrl-X).
9. [] In order for the new whitelist to be active, restart the Squid proxy service.

```
sudo systemctl restart squid
```

10. [] **Optional:** If you want to have a look at how the Squid proxy server configuration uses the whitelist then open the file /etc/squid/squid.conf and examine the settings, including the port used 3128 which is the default Squid proxy port.
11. [] From the *ArcBox-Client* machine open *Hype-V Manager*, click the *ArcBox-Client* server, click the *ArcBox-Ubuntu-02* under Virtual Machines. From the *Networking* tab find the IPV4 address of the machine. Make a note of this IP address. Then ssh into it from Powershell 7.

```
ssh jumpstart@<Enter IP Address of the ArcBox-Ubuntu-02 machine>
```

12. [] Configure the proxy settings on the ArcBox-Ubuntu-02 machine.

```
sudo nano /etc/environment
```

13. [] Add the following 3 lines to the file making sure that you use the IP address of the **proxy server!** After you have done, Save the file (Ctrl-O then Enter) and exit (Ctrl-X).

```
export http_proxy=<proxy IP Address>:3128
export https_proxy=<proxy IP Address>:3128
export no_proxy="localhost,127.0.0.1,::1"
```

```
jumpstart@arcbox-ubuntu-02: ~
GNU nano 6.2                                     /etc/environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"
export http_proxy="10.10.1.10:3128"
export https_proxy="10.10.1.10:3128"
export no_proxy="localhost,127.0.0.1,::1"
```

14. [] Type *EXIT* then re-login again to the *ArcBox-Ubuntu-02* for the changes to take effect.

```
ssh jumpstart@<Enter IP Address of the ArcBox-Ubuntu-02 machine>
```

15. [] Optional: Test that your proxy is blocking internet access

```
jumpstart@arcbox-ubuntu-02: ~
jumpstart@arcbox-ubuntu-02:~$ curl -I http://bing.com
HTTP/1.1 403 Forbidden
Server: squid/5.9
Mime-Version: 1.0
Date: Thu, 13 Feb 2025 13:40:33 GMT
Content-Type: text/html; charset=utf-8
Content-Length: 3485
X-Squid-Error: ERR_ACCESS_DENIED 0
Vary: Accept-Language
Content-Language: en
X-Cache: MISS from proxy
X-Cache-Lookup: NONE from proxy:3128
Via: 1.1 proxy (squid/5.9)
Connection: keep-alive

jumpstart@arcbox-ubuntu-02:~$
```

16. [] From the Azure portal go to the "Machines - Azure Arc" page and select "Add/Create" at the upper left, then select "Add a machine".

The screenshot shows the Azure Arc | Machines page. On the left, there's a navigation sidebar with options like Overview, All Azure Arc resources, and Azure Arc resources (Machines, Kubernetes clusters, Host environments, Data services, Internet of Things (IoT)). The 'Machines' option is highlighted with a red box. In the main content area, there's a search bar and filter options (Resource group equals all, Location equals all). Below that, there are two main options: 'Add a machine' (highlighted with a red box) and 'Create a machine in a connected host environment'. A table lists existing machines, including 'Arcbox-Ubuntu-01' which is connected to the 'Arcbox' resource group and running Ubuntu 22.04 LTS. The table includes columns for Name, Kind, Arc agent status, Resource group, Subscription, Operating system, Defender extensi..., and Monitoring ex...

17. [] In the next screen, go to "Add multiple servers" and click on "Generate script".

The screenshot shows the 'Add servers with Azure Arc' page. It features five options: 'Add a single server' (with 'Generate script' and 'Learn more' buttons), 'Add multiple servers' (highlighted with a red box), 'Add Windows Server with installer' (with 'Download installer' and 'Learn more' buttons), 'Add servers from AWS' (with 'Add servers' and 'Learn more' buttons), and 'Add servers from Update Management' (with 'Add servers' and 'Learn more' buttons). The 'Add multiple servers' section contains a brief description: 'To add multiple servers to Azure, we will generate a script that handles authentication through a service principal. You will see that and other prerequisites next.'

18. [] Fill in the required details and choose Linux for the Operating System box. Set the connectivity method to *Proxy server*. Enter the URL of the proxy server and name of the Arc gateway as shown in the screenshots but make sure that they match with your own settings (the IP address of the proxy server that you observed earlier). Also Choose the service principal in the "Authentication" dropdown that you created in the previous exercise.

The screenshot shows the 'Add multiple servers with Azure Arc' configuration page. It has tabs for Basics, Tags, and Download and run script. The Basics tab is active. It includes sections for Project details, Server details, and SQL Server.

Project details: Selects the subscription (MPDG-Iod5008779) and resource group (ArcBox).

Server details: Sets the Region to (Europe) UK South and the Operating system to Linux.

SQL Server: Has a 'Connect SQL Server' checkbox and a note about automatically connecting instances to Azure Arc.

Connectivity method

Choose how the connected machine agent running in the server should connect to the Internet. This setting only applies to the Arc agent. Proxy settings for extensions are configured separately.

Connectivity method *

Public endpoint

Proxy server

Private endpoint

(i) The Arc gateway (preview) reduces the number of URLs that you must allow in your Proxy servers to use Arc. To use the Arc gateway (preview) feature, a gateway resource is required. [Learn more](#)

Proxy server URL * ⓘ

Gateway resource ⓘ
Create new

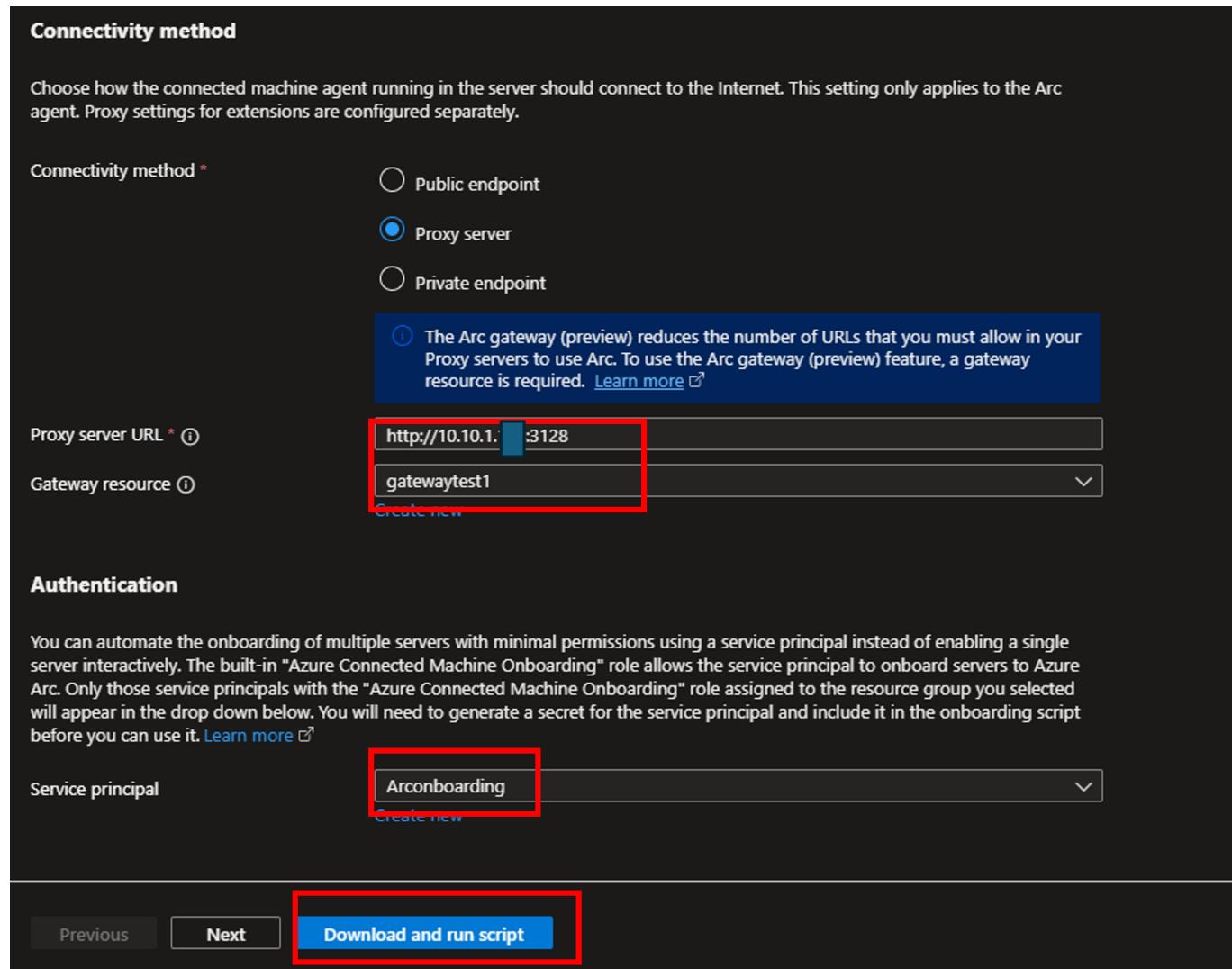
Authentication

You can automate the onboarding of multiple servers with minimal permissions using a service principal instead of enabling a single server interactively. The built-in "Azure Connected Machine Onboarding" role allows the service principal to onboard servers to Azure Arc. Only those service principals with the "Azure Connected Machine Onboarding" role assigned to the resource group you selected will appear in the drop down below. You will need to generate a secret for the service principal and include it in the onboarding script before you can use it. [Learn more](#)

Service principal
Create new

Download and run script

Previous



19. [] On the *Download and run script* tab notice the reminder to ensure that the required endpoints are not blocked by the proxy server. These were the links that you added to the proxy whitelist earlier. Choose the *Basic script* option.

Basics Tags Download and run script

1. Ensure the network connectivity requirements are met

If outbound connectivity is restricted by your proxy server, make sure the URLs listed below are not blocked.

URL	Description
87c.gw.arc.azure.com	Your Arc gateway URL
management.azure.com	Azure Resource Manager Endpoint, required for ARM control channel
login.microsoftonline.com	Microsoft Entra ID's endpoint, for acquiring identity access tokens
gbl.his.arc.azure.com	The cloud service endpoint for communicating with Arc Agents
<region>.his.arc.azure.com	The cloud service endpoint for communicating with Arc Agents
packages.microsoft.com	Required to acquire Linux based Arc agent payload, only needed to connect Linux servers to Arc
download.microsoft.com	Used to download the Windows installation package

Before you run the script, make sure your server meets the following requirements.

- HTTPS access to Azure services
All servers require access to port 443 and a set of outbound URLs for the Azure Arc agents to properly function.
- Local administrator permission

2. Select deployment method

Deployment method

Basic script

20. [] Copy the script or download it. You should see that your proxy IP address has been configured in the script.

```

20  output=$(wget https://gbl.his.arc.azure.com/azcmagent-linux -e use_proxy=yes -e https_proxy="http://10.10.
1.104:3128" -O "$LINUX_INSTALL_SCRIPT" 2>&1);
21  if [ $? != 0 ]; then wget -qO- -e use_proxy=yes -e https_proxy="http://10.10.1.104:3128" --method=PUT
--body-data='{"subscriptionId": "'$subscriptionId'", "resourceGroup": "'$resourceGroup'", 
"tenantId": "'$tenantId'", "location": "'$location'", "correlationId": "'$correlationId'", 
"authType": "'$authType'", "operation": "'onboarding'", "messageType": "'DownloadScriptFailed'", 
"message": "'$output'"}' "https://gbl.his.arc.azure.com/log" &> /dev/null || true; fi;
22  echo "$output";
23
24  # Install the hybrid agent
25  bash "$LINUX_INSTALL_SCRIPT" --proxy "http://10.10.1.104:3128";
26
27  # Run connect command
28  sudo azcmagent connect --service-principal-id "$ServicePrincipalId" --service-principal-secret
"$ServicePrincipalClientSecret" --resource-group "$resourceGroup" --tenant-id "$tenantId" --location
"$location" --subscription-id "$subscriptionId" --cloud "$cloud" --gateway-id "$gatewayId" --tags
'ArcSQLServerExtensionDeployment=Disabled' --correlation-id "$correlationId";
29

```

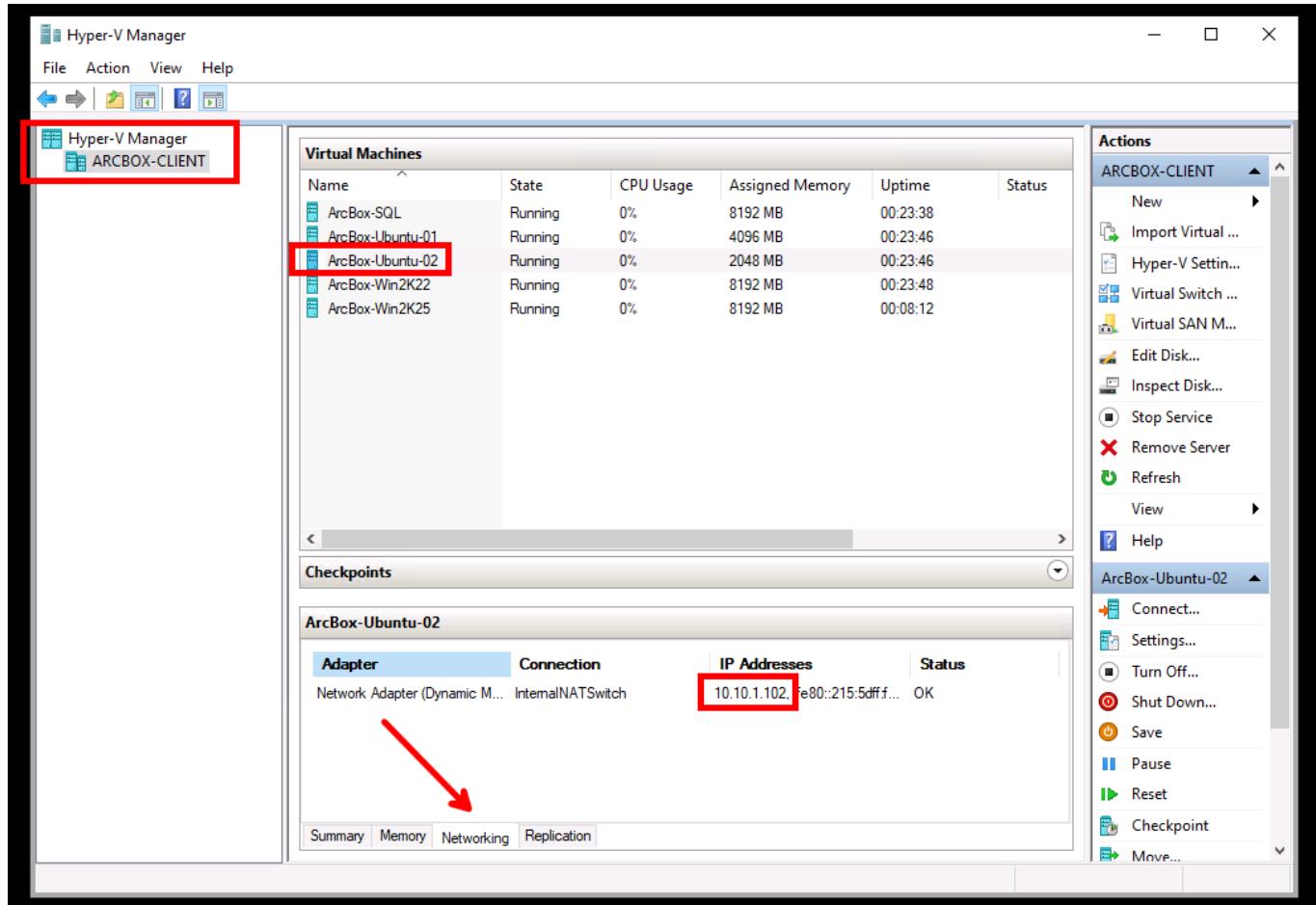
[Download](#)



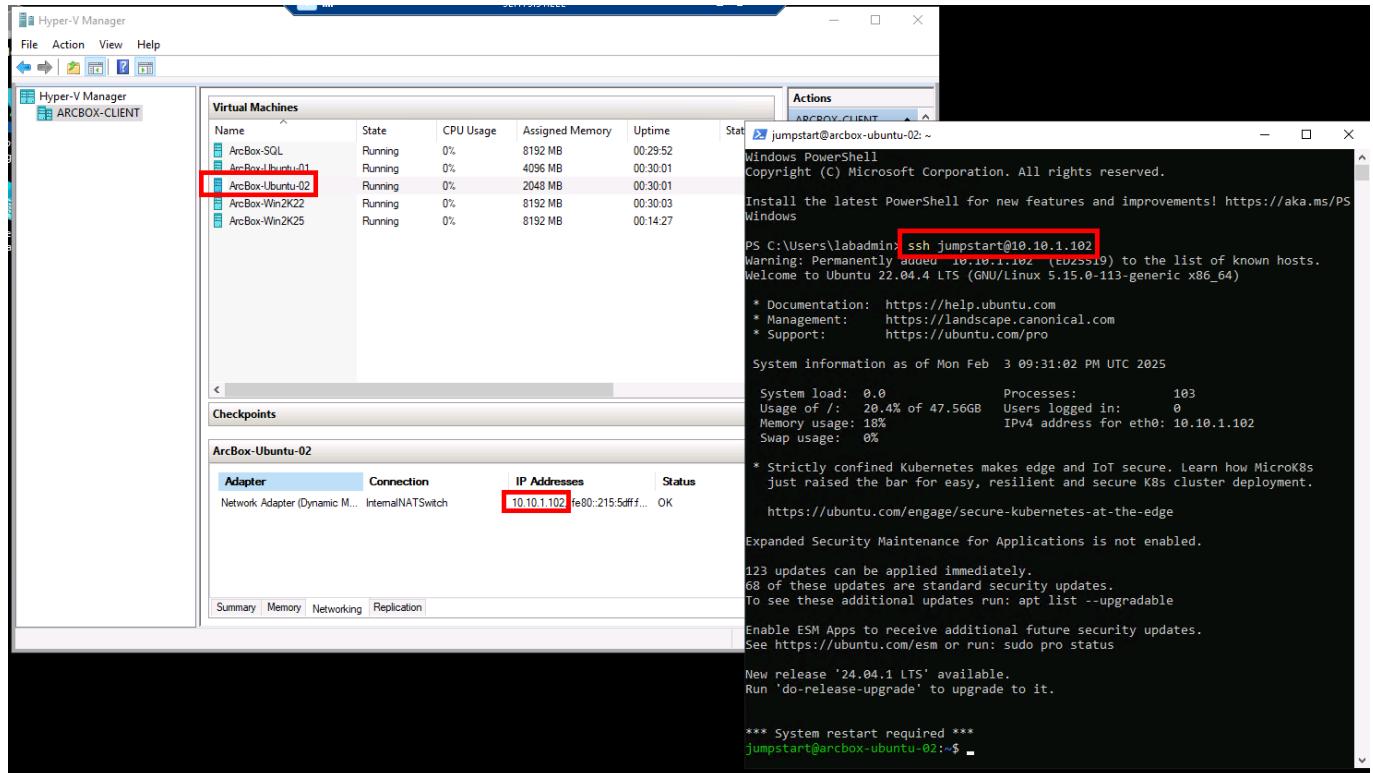
21. [] In your local editor, paste the script and make sure that the Service Principal Id and the client secret are filled in. Also add the following 3 lines to the script **just below the last export statement** (to allow onboarding of Azure linux machines):

```
sudo ufw --force enable
sudo ufw deny out from any to 169.254.169.254
sudo ufw default allow incoming
```

22. [] Connect the ArcBox-Client machine. If you are still connected to the *ArcBox-Ubuntu-02* machine with ssh session then ignore the following 2 steps.
23. [] From the "Networking" tab on Hyper-v Manager find the IP address of the Linux machine *ArcBox-Ubuntu-02*.



24. [] SSH into the *ArcBox-Ubuntu-02* machine using "PowerShell". If you are prompted for a password then use *JS123!!*:



```
ssh jumpstart@<Enter IP Address of the Linux machine>
```

25. [] Create an empty onboarding script file using the nano editor, and paste (Right-click) the script content from your local machine. Once you are in the editor, navigate using the arrow keys and not the mouse.

```
nano onboardingscript.sh
```

26. [] Save the file (Ctrl-O then Enter) and exit (Ctrl-X). Now you can run the script:

```
sudo bash ./onboardingscript.sh
```

27. [] Wait for the script to finish successfully. A message should confirm that the machine is now Arc-connected. You can also verify that our Linux machine is connected in the Azure portal (Machines - Azure Arc).



28. [] You can confirm that the *ArcBox-Ubuntu-02* is associated with your Arc gateway by navigating to the Azure portal page of the Arc gateway and examining the *Associated resources*.

The screenshot shows the 'Associated resources' section of the Azure Arc gateway interface. The left sidebar includes options like Overview, Activity log, Access control (IAM), Tags, and Associated resources, with 'Associated resources' being the active tab and highlighted by a red box. The main area displays a table with columns: Name (with an up-down arrow), Kind, Connection status, Resource group, and Subscription. One row is visible, showing 'arcbox-ubuntu-02' under 'Name', 'Connected' under 'Connection status', and a blue bar under 'Subscription'. A search bar and an 'Add filter' button are also present at the top of the main area.

Task 2 has been completed

====

Congratulations, you have completed all tasks in this lab

Click **Next** for the next lab or **Go back to the main table of content**