

WORKSHOP

Data in use Protection Compass

Keep the cape in the Cloud and on the Edge

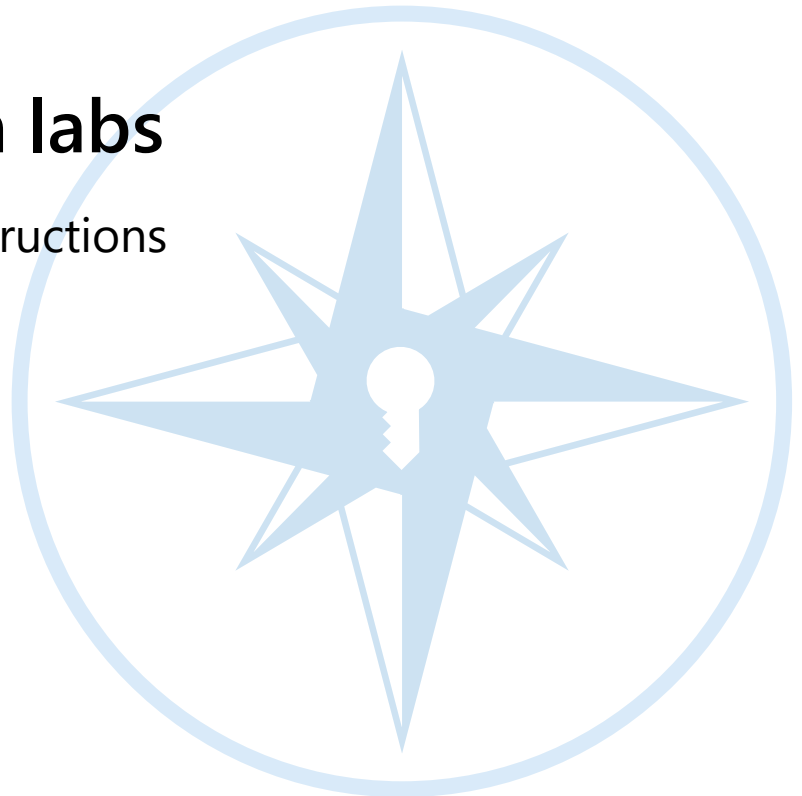
N

Available hands-on labs

Hands-on lab deployment instructions

Version 1.0 (Alpha) - June 2020

<https://aka.ms/DataInUseProtectionWS>



Information in this document, including URL and other Internet Web site references, is subject to change without notice. Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place or event is intended or should be inferred. Complying with all applicable copyright laws is the responsibility of the user.



Attribution 4.0 International (CC BY 4.0)

Microsoft and any contributors grant you a license to this document under the [Creative Commons Attribution 4.0 International Public License](#), see the [LICENSE](#) file, and grant you a license to any code in the repository under the MIT License, see the [LICENSE-CODE](#) file.

Microsoft, Windows, Microsoft Azure and/or other Microsoft products and services referenced in the document may be either trademarks or registered trademarks of Microsoft in the United States and/or other countries. The license for this document does not grant you rights to use any Microsoft names, logos, or trademarks. Microsoft's general trademark guidelines can be found at <http://go.microsoft.com/fwlink/?LinkID=254653>.

Privacy information can be found at <https://privacy.microsoft.com/en-us/>

Microsoft and any contributors reserve all other rights, whether under their respective copyrights, patents, or trademarks, whether by implication, estoppel or otherwise.

Table of contents

ABSTRACT AND LEARNING OBJECTIVES.....	1
Overview.....	1
Hands-on lab requirements.....	1
Check your quota on the Azure portal	2
PREPARE YOUR ENVIRONMENT FOR THE HANDS-ON LABS.....	3
Provision a resource group.....	3
Create a storage account.....	3
Create one container inside your storage account.....	4
Copy all the files into your vhds containers.....	Error! Bookmark not defined.
Download the scripts.....	5
Create the configuration file.....	6
CREATE THE DEVTEST LABS FOR THE HANDS-ON LABS.....	8
Run the main scripts (in order of logical execution).....	8
USE THE DEVTEST LABS FOR THE HANDS-ON LABS	11
Access your DevTest Lab.....	11
Start a VM.....	11
Connect to a VM	11
Connect to a Windows VM	11
Copy the provided SSH key pair.....	12
Connect to a Linux VM	12
Stop a VM	12
REMOVE YOUR ENVIRONMENT FOR THE HANDS-ON LABS.....	13
Delete the Resource group in which you placed your Azure resources	13
APPENDIX A: UTILITY SCRIPTS.....	14
APPENDIX B: HANDS-ON LABS' RESOURCES	16
List of labs.....	16
Confidential Computing (CC) lab	16
Homomorphic Encryption (HE) lab.....	16
Virtual machine (VM) type and disk type	16

- VM type 16
- Managed HDD Disk Type (Standard)..... 17
- Size of VHDS..... 17
- Azure DevTest Labs JSON files..... 18
- List of credentials 19
 - Windows credentials 19
 - Linux credentials for SSH connections 19
- APPENDIX C: DOWNLOAD THE VHD FILES TO RUN THEM ON-PREMISES..... 20
- APPENDIX D: ADD/REMOVE VMS ACCORDING TO A PLANNING21

Abstract and learning objectives

Overview

The **Data in use Protection Compass** is a customer workshop that covers various applicable techniques and capabilities to ensure for their (most) sensitive data in use adequate levels of confidentiality and privacy as part of their Microsoft's cloud journey or in the Edge. It is designed for you to help our customers select the best suited approach(es) depending on their ultimate goals and related requirements.

This workshop provides a series of hands-on labs to illustrate in practice all the concepts presented and discussed in the tracks it comprises, i.e. the so-called "Common Track", and the two specific tailored for purpose more advanced tracks, namely the "Confidential Computing Track", and the "Homomorphic Encryption Track".

In turn, each specific track provides a series of exercises tailored for purpose so that the audience can get started with the covered subjects.

Once deployed in an Azure subscription, these hands-on labs can be implemented on their own.

This document provides all the hands-on lab setup instructions required to successfully do so.

Hands-on lab requirements

- A [Microsoft account](#).
- An Azure subscription. The subscription being used for the hands-on labs must be pay-as-you-go, EA or MSDN.
 - Trial subscriptions will not work.
 - Enough credit to allow a student to run VMs for 5 hours as an estimated.
- Adequate quotas for the required virtual machines (VMs), see next section.

It's recommended to use Azure to deploy and run the virtual machines (VMs) through the [Azure DevTest Labs](#).

However, you can also run the VMs on-premises. To do so, you can download the VHD files, and instantiate them on top of Hyper-V. **Please note that some VHD images require an Intel Xeon processor with the Intel Software Guard Extensions (SGX) technology.**

Deploying the VMs on an Azure DevTest Lab environment will take a number of hours to deploy. If updates are missing, update artifacts can take additional hours to apply updates on virtual machines.

IMPORTANT: Please create the storage account and the DevTest Labs **in the same Azure Region**, otherwise, you will have to pay extra-money for the egress traffic. The Available hands-on labs in this guide features the **UK South** Azure region. As of this writing, this is indeed one of the three currently regions where the [DCsv2-Series Family of VMs](#) is currently available, the two others being Canada Central and US East. This will evolve over the time. Please check out the [current availability of the DCsv2-Series Family](#) to set the appropriate Azure region depending on your own specific location and requirements.

Check your quota on the Azure portal

This step is important!

By default, as indicated above, the quota in place may not be enough to deploy the Data in use Protection Compass hands-on labs. Moreover, it can take few hours or few days to have the quota increased for the subscription. Plan accordingly.

1. Log in to the [Azure portal](#).
2. From left navigation area, select **All services**.
3. Select **Subscriptions**.
4. Select the targeted subscription for the hands-on lab.
5. Click **Usage + Quotas**
6. Select **the region** that you want to check against.
7. Check the following quotas:
 - Dv3 Family vCPUs
 - DCsv2 Family vCPUs
 - Public IP Addresses

Prepare your environment for the hands-on labs

You will set up your environment, and thus deploy all the available hands-on labs using the provided Azure DevTest Labs scripts.

Provision a resource group

You will create an Azure resource group for all the resources to setup for these hands-on labs.

1. In the [Azure Portal](#), select **Resource groups**, select **+Add**, then enter the following settings in the **Create a resource group** blade:
 - a. **Subscription:** Select the subscription you are using for this hands-on lab
 - b. **Resource group:** Enter **COMPASS-HOL-RG**.
 - a. **Region:** Select the **UK South** location as stated before for the resource group. All the hands-on labs' resources will be created here.

The screenshot shows the 'Create a resource group' blade in the Azure Portal. At the top, there are tabs for 'Basics', 'Tags', and 'Review + create'. Below the tabs, there is a description of a resource group. Under 'Project details', the 'Subscription' is set to 'Souscription NTO-France Microsoft Azure' and the 'Resource group' is set to 'COMPASS-HOL-RG'. Under 'Resource details', the 'Region' is set to '(Europe) UK South'.

2. Select **Review + create**.
3. Once the validation passed, click **Create**.
4. When invited, click **Go to resource group**.

Create a storage account

Inside the resource group created above, you will now create a new storage account.

1. In the [Azure Portal](#), select **Storage accounts**, select **+Add**, then enter the following settings in the **Create storage account** blade:
 - a. **Subscription:** Select the subscription you are using for this hands-on lab
 - b. **Resource group:** Use existing and select **COMPASS-HOL-RG**.
 - b. **Storage account name:** Choose a unique name. For example, **mycompasshols**.
This will be your Destination Storage Account. It will be later referred as to <DestStorageAccount>
 - c. **Location:** Select the **UK South** location as for the above resource group

- d. **Performance:** Leave Standard selected
- e. **Account kind:** Leave StorageV2 (general purpose v2) selected
- f. **Replication:** Select Locally-redundant storage (LRS)
- g. **Access tier (default):** Leave Hot selected

Basics Networking Advanced Tags Review + create

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. [Learn more about Azure storage accounts](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Resource group * [Create new](#)

Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

Storage account name * ✓

Location *

Performance ☒ Standard ☐ Premium

Account kind

Replication

Access tier (default) ☐ Cool ☒ Hot

- 2. Click **Review + create**
- 3. Once the validation passed, click **Create**.
- 4. When invited, click **Go to resource**.

Create one container inside your storage account

Inside the storage account blade:

- 1. Click **Container**.
- 2. Click + **Container**. Enter the following settings in the **New container** blade.
 - a. **Name:** type "vhds".
 - b. **Public access level:** Leave Private (no anonymous access) selected.
- 3. Click **Create**.
- 4. On the left pane, Under Settings, click **Access keys**. Take a note the content of Key1.
It will be your Destination Storage Key. It will be later referred as to <DestStorageKey>

Prepare and upload the virtual machine images

Proceed as per documentations to upload the expected files to the above storage account:

- **Confidential Computing hands-on lab – Hands-on lab setup.**
- **Homomorphic Encryption hands-on lab – Hands-on lab setup.**

At this stage, all the required files should be available in your account storage:

<https://compasswshols.blob.core.windows.net/hol-vhds>

Download the scripts

Important note Az and AzureRM cmdlets modules cannot be imported in the same session or used in the same script or runbook. If you are running PowerShell in an environment you control, you can use the 'Uninstall-AzureRm' cmdlet to remove all AzureRm modules from your machine.

[See here](#) for more information.

1. Open a PowerShell console with admin privileges.
2. Install Azure PowerShell (Az Module on Windows with PowerShell). See [Install Azure PowerShell](#).

```
PS C:\> Set-ExecutionPolicy -ExecutionPolicy RemoteSigned
```

- a. When invited, press A

```
PS C:\> Install-Module -Name Az -AllowClobber -Scope CurrentUser
```

- a. When invited, press Y
- b. Then press A

3. If Azure PowerShell is already installed, check that you are using the latest version of AzureRM:

```
PS C:\> Update-Module -Name Az
```

4. Install the Azure AD Module. See [Azure Active Directory PowerShell for Graph](#).

```
PS C:\> Install-Module AzureAD
```

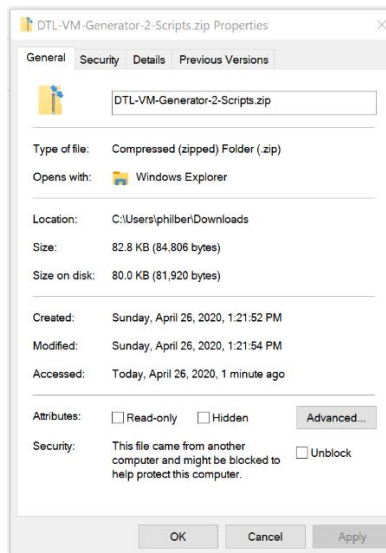
- a. When invited, press A

5. Download on your local machine the scripts:
<https://github.com/petehauge/DTL-VM-Generator/tree/V2-Scripts>. See Appendix A: Utility scripts.

[Find file](#)[Clone or download](#)

Select **Download ZIP**.

- Unblock the `DTL-VM-Generator-2-Scripts.zip`. Right-click on the file, check **Unblock**, and then click **OK**.



- Create a `scripts` folder at the root of your C: drive on your local machine.
- Unzip the content of the `DTL-VM-Generator-2-Scripts` folder within the `DTL-VM-Generator-2-Scripts.zip` archive file into the `C:\scripts\` folder.

Create the configuration file

- Still in the `C:\scripts\` folder, create the file `config.csv` by copying the `demoConfig.csv` file

Note .csv files have a header row. This can be misleading when using a text editor. This header row shouldn't be deleted.

- Change the content of **ResourceGroupName** and use the following value: **COMPASS-HOL-RG**
- Change the content of **StorageAccountName** and use the name of the storage account you choose, i.e. the value of `<DestStorageAccount>`
- Change the content of **IpConfig** and use the following value: **Public**
- Change the content of **StorageContainerName** and use the following value: **vhds**
- Change the content of **StorageKey** and use value of the storage key you got, i.e. the value of `<DestStorageKey>`
- Change the content of **ShutDownTime**, by default VM stopped at 7PM.
- Change the content of **LabOwners** and **LabUsers** fields. Must be empty or contains a list of emails accounts declared in the Azure AD subscription separated by commas.
- Save your `config.csv` file.

For example, to create only one DevTest Labs environment named Lab1, the `config.csv` file would look like below (opened by [Visual Studio Code Excel](#) file handler):

Dev Test Lab Name	Resource Group Name	Ip Config	Storage Account Name	Storage Container Name	
Lab1	COMPASS-HOL-RG	Public	compasshols	vhds-2020-04	
Storage Account Key	Shut Down Time	Timezone Id	Lab Region	Lab Owners	Lab Users
<key>	1900	"W. Europe Standard Time"	westeurope		

Create the DevTest Labs for the hands-on labs

The provided scripts are memory intensive and need uninterrupted Internet connectivity.

The scripts will take approximatively two hours to create the DevTest Labs.

For reliability purpose, it's recommended to deploy the script from inside an Azure Virtual Machine (VMs).

The main scripts take as input a .csv file (`config.csv`) for creating multiple DevTest Labs from an Azure Blob storage containing VHDs and JSON descriptors.

The script creates the labs, add the VHDs as custom images and creates one claimable VM for each custom image.

Additionally, you can set owners and users for such labs together with additional data like shutdown time, region, and such. The users added to the lab have no permissions to create new VMs, so that the content of the lab is fixed.

The script creates the labs in parallel. All the output (even from multiple jobs) goes to the console.

To generate log files from them, just redirect the error stream or call Start/StopTranscript in the console. You can set whatever Debug output you desire in your console and the script should behave appropriately.

- As a rule of thumb, you need to have 100 MB free of RAM for creating one hands-on lab. So if you plan to deploy in parallel 10 DevTest Labs, you need to have 1 GB free RAM for the workstation you are using to execute the scripts.
- If you plan to deploy more than 30 DevTest Labs in a single batch, it's a best practice to deploy from an 8 GB virtual machine and 4 CPUs. Otherwise you may encounter "out of memory exception" or "credential errors".

Run the main scripts (in order of logical execution)

1. Open a PowerShell console, go to the `C:\Scripts` folder.
2. Sign in to your Azure subscription:

```
PS C:\> .\Login-AzSub.ps1
```

3. Type or paste your subscription ID. **Please note that, to create the DevTest Labs, the Azure account you are using needs to be owner of the subscription.**
 - a. To get your Subscription ID, go to your **COMPASS-HOL-RG** resource group and copy the subscription ID.
 - b. Connect to your Azure AD tenant.

```
PS C:\> Connect-AzureAd
```

4. In the PowerShell console, now run the following scripts:

Note By default, the logs are displayed on the screen, if you want to generate log file, you can run in the current PowerShell console the following command before executing the script: `Start-Transcript -path "C:\Log.txt" -append`.

When you've finished, simply type `Stop-Transcript` to stop the recording into the log file.

- a. Adds a custom role to the subscription.

```
PS C:\> .\New-CustomRole.ps1
```

The **No VM Creation User** role doesn't have permissions to create new resources in Azure and can only claim, unclaim, start, stop and restart VMs.

If you already deployed the **No VM Creation User** role, please update the definition:

```
PS C:\> .\Update-NoCreationVMRoleDefinition.ps1
```

- b. Reads the `Config.csv` file and creates an empty DTL Labs ready to be filled with VMs later on.
(Be sure to execute the `New-EmptyLabs.ps1` file NOT `New-EmptyLab.ps1`)

```
PS C:\> .\New-EmptyLabs.ps1 -ConfigFile .\Config.csv -CustomRole "No VM Creation User"
```

Note To include guest Azure AD users: you must have the [AzureAD PowerShell cmdlets module](#) installed AND ran the [Connect-AzureAD](#) cmdlet before running this script/

- c. Add specific tags to resources created later with the DevTest Labs.
It can help to understand and track the cost. For that, you can define your tags on the command line when creating labs like this:

```
PS C:\> .\Add-TagsToResources.ps1 -ConfigFile .\Config.csv -tags  
@{'Course'='MicrosoftDataInUseCompass' ; 'BillingCode'='12345'} -tagLabsResourceGroup $true
```

- d. Fill a lab with VMs. You can pass a series of patterns that match image names to create and specify what to do in case there are already existing VMs in the lab with the same name. It performs the following operations:
- Get the json descriptors for the VMs and select the ones matching the patterns
 - Copy VHDs from blob storage to DTL lab storage
 - Create custom images from the VHDs
 - Create one VM for each custom image
 - Create the network topology described in the json descriptors
 - Delete the custom images from the lab

For example, create all the VMs in the labs:

```
PS C:\> .\Create-Vms.ps1 -IfExist "Error"
```

Or create only the VMs for the exercises of the specific Confidential Computing (CC) track:

```
PS C:\> .\Create-Vms.ps1 -ImagePattern "CC*" -IfExist "Error"
```

Or alternatively the ones for the exercises of the specific Homomorphic (HE) track:

```
PS C:\> .\Create-Vms.ps1 -ImagePattern "HE*" -IfExist "Error"
```

The creation of the different DevTest Labs will take approximately one hour.

As already noticed, one should stress that this is highly recommended to the scripts from inside an Azure VM to minimize network delays and avoid network outage.

By default, there is a timeout (5 hours) after which the script will finish.

If one or more VMs failed to deploy inside one or more Labs, the best corrective action consist in creating a new config file with the name of the failed labs and then redeploy them using the following command:

```
PS C:\> .\Create-Vms.ps1 -IfExist "Delete"
```

Use the DevTest Labs for the hands-on labs

Access your DevTest Lab

1. When the DevTest Lab is provisioned, launch a browser and navigate to <https://portal.azure.com>. Once prompted, login with your Microsoft Azure credentials. If prompted, choose whether your account is an organizational account or just a Microsoft Account. Enter the related credentials to sign in.
2. In the [Azure Portal](#), open the resource group named **COMPASS-HOL-RG**.
3. Select the right subscription if the resource group is not displayed.
4. Click **DevTest Lab**.

Start a VM

1. under **Claimable virtual machines**, right-click the intended VM row, and then select **Claim machine**.
2. When the VM is started, it will be displayed in the **My Virtual Machines** pane.
After one minute, the status will be **Running**.

Connect to a VM

Connect to a Windows VM

1. Under **My Virtual Machines**, select the intended lab VM row.
2. At the end of line click ..., and then select **Connect**.

The screenshot shows the 'Connect with RDP' section in the Azure Portal. At the top, there are tabs for 'RDP', 'SSH', and 'BASTION', with 'RDP' being the active tab. Below the tabs, the text 'Connect with RDP' is followed by instructions: 'To connect to your virtual machine via RDP, select an IP address, optionally change the port number, and download the RDP file.' There are two input fields: 'IP address *' with a dropdown menu showing 'Public IP address (52.166.88.167)' and a 'Port number *' field with the value '3389'. A blue button labeled 'Download RDP File' is positioned below these fields. At the bottom, there is a 'Can't connect?' section with two links: 'Test your connection' and 'Troubleshoot RDP connectivity issues'.

3. Select **Download RDP file**, then open the downloaded RDP file.
4. Click **Connect** on the Remote Desktop Connection dialog.
5. When invited, specify the username and password as detailed in section § Windows credentials in the Appendix B: Hands-on labs' resources. (Do not use your organizational account or your Microsoft Account one.)

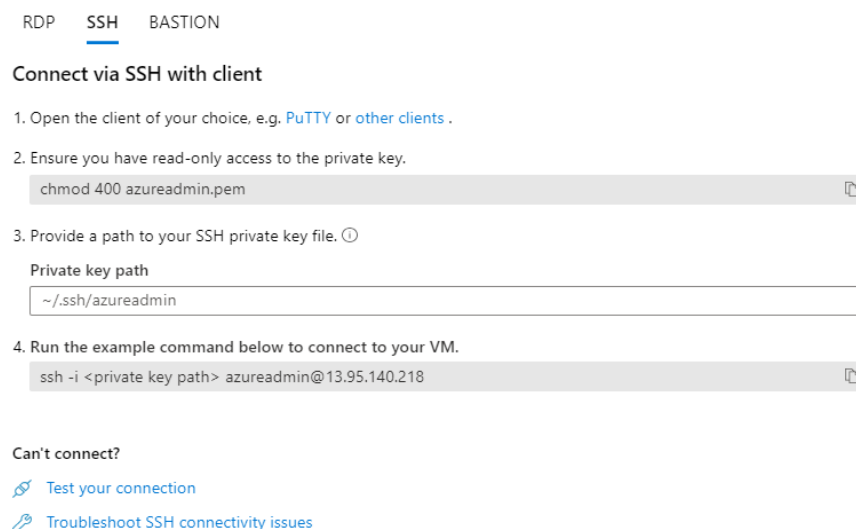
- Click **Yes** to connect, if prompted that the identity of the remote computer cannot be verified.

Copy the provided SSH key pair

Copy the provided SSH key pair as indicated in section § Linux credentials for SSH connections in the Appendix B: Hands-on labs' resources, i.e. the `id_rsa.pub` file for the RSA public key and the `id_rsa` file for the corresponding private key – to the `%USERPROFILE%\.ssh` folder, on your local machine.

Connect to a Linux VM

- Under **My Virtual Machines**, select the intended VM row.
- At the end of line click **...**, and then select **Connect**. A SSH connection string will be displayed.



The screenshot shows the 'SSH' tab selected in the 'Connect via' section. It provides a step-by-step guide: 1. Open a client like PuTTY. 2. Ensure read-only access to the private key, with a terminal snippet `chmod 400 azureadmin.pem`. 3. Provide a path to the private key file, with a text input field containing `~/ssh/azureadmin`. 4. Run the example command `ssh -i <private key path> azureadmin@13.95.140.218`. Below this, there is a 'Can't connect?' section with links to 'Test your connection' and 'Troubleshoot SSH connectivity issues'.

- Now open on your local machine, a prompt command line and enter the provided SSH connection string.

```
C:\> ssh azureadmin@13.95.140.218
```

- When prompted, type "yes". Optionally specify your passphrase if any for your private key to unlock it.

At the stage, you should be connected on the Linux lab VM with a Bash shell.

Stop a VM

- Under the **My Virtual Machines** pane, select the VM row.
- Right-click it, and then select **Unclaim**.

Remove your environment for the hands-on labs

You will deprovision any Azure resources that were created in support of the lab.

Delete the Resource group in which you placed your Azure resources

1. From the Portal, navigate to the blade of your **COMPASS-HOLS-RG** resource group and select **Delete** in the command bar at the top.
2. Confirm the deletion by re-typing the resource group name and selecting **Delete**.

You should follow all steps provided *after* attending the hands-on labs.

If so, this concludes the hands-on labs setup instructions for the Data in use Protection Compass.

Appendix A: Utility scripts

The repo <https://github.com/lucabol/DTLCustomImagesLab> contains a collection of scripts, which might have value on their own to build slightly different solutions and are, therefore, lightly documented below. Refer to the code for full description of arguments.

- `Login-AzSub.ps1` to log into Azure with a specific subscription Id. You need to use this script the first time to connect and select the right Azure Subscription.
- `Add-TagsToResources.ps1` to add specific tags to resources created later with DevTest Labs. It can help to understand and track the cost. To do so, you can define your tags on the command line when creating labs like this:

```
PS C:\> .\Add-TagsToResources.ps1 -ConfigFile .\Config.csv -tags  
@{'Course'='MicrosoftDataInUseProtectionCompass' ;'BillingCode'='12345'} -tagLabsResourceGroup $true
```

- `Create-Vms.ps1` to create VM(s) in labs. For example, if you'd like to delete and recreate all the VMs for the specific Confidential Computing (CC) Track by using the latest VHDS:

```
PS C:\> .\Create-Vms.ps1 -ImagePattern "CC*" -IfExist "Delete"
```

Or likewise, to delete and recreate all the VMs for the specific Homomorphic Encryption (HE) Track by using the latest VHDS

```
PS C:\> .\Create-Vms.ps1 -ImagePattern "HE*" -IfExist "Delete"
```

- `Get-LabsInfo.ps1` to display the name of the VMs running in all the labs described in the configuration csv file.
- `Get-LabInfo.ps1` to display the name of the VMs running in a specific lab.
- `Get-VmStatus.ps1` to display the status, power state and artifacts of the VMs running in all labs described in the configuration csv file.
- `Remove-Vms.ps1` to remove all the VMs in all the labs described in the configuration csv file matching certain patterns in the Notes field. For example, if you'd like to remove and delete all the VMs for the specific Confidential Computing (CC) Track, you will use:

```
PS C:\> .\Remove-Vms.ps1 -MatchBy "Note" -Pattern "CC-HOL"
```

Likewise, if you'd like to remove and delete all the VMs for the specific Homomorphic Encryption (HE) track, you will use instead:

```
PS C:\> .\Remove-Vms.ps1 -MatchBy "Note" -Pattern "HE-HOL"
```

- `Remove-Vm.ps1` to remove a specific VM in a specific lab matching certain patterns
- `Remove-Labs.ps1` to remove all the labs described in the configuration csv file.
- `Remove-Lab.ps1` to remove one specific lab.
- `Set-AccessControl.ps1` to add permissions into all the labs according to the content of LabOwners and LabUsers described in the configuration csv file.
- `Stop-LabVms.ps1` to stop all the VMs running in a specific lab.

Appendix B: Hands-on labs' resources

List of labs

The comprises the following two hands-on labs:

1. Confidential Computing (CC) lab.
2. Homomorphic Encryption (HE) lab.

Confidential Computing (CC) lab

This lab requires the following VMs:

Name of the VM	Name of VHD	Type of VM	Original Network IP	DNS server
CC-HOL-WDEV-01	CC-HOL-WDEV-01.vhd	Standard D2 v3	10.0.0.20 255.0.0.0	Inherit from VNet
CC-HOL-LTEST-01	CC-HOL-LTEST-01.vhd	Standard DC2s v2		Inherit from VNet
CC-HOL-LTEST-02	CC-HOL-LTEST-02.vhd	Standard DC2s v2		Inherit from VNet

Homomorphic Encryption (HE) lab

This lab requires the following VMs:

Name of the VM	Name of VHD	Type of VM	Original Network IP	DNS server
HE-HOL-WDEV-01	HE-HOL-WDEV-01.vhd	Standard D2 v3	10.0.0.20 255.0.0.0	Inherit from VNet
HE-HOL-LDEV-01	HE-HOL-LDEV-01.vhd	Standard D2 v3		Inherit from VNet

Virtual machine (VM) type and disk type

VM type

Size	vCPU	Memory GiB	Temp storage (SSD) GiB
Standard D2 v3	2	8	40
Standard DC2s v2	2	8	100

Managed HDD Disk Type (Standard)

Size	Size HDD GB
S10	128
S4	32

Size of VHDS

Below are the details of the disks for each virtual machine (VM).

Name of VM	Size of Disk	Disk Space Used	Disk Space Free
CC-HOL-WDEV-01	127GB	60 GB	66.5 GB
CC-HOL-LTEST-01	30GB	1.5 GB	27.5GB
CC-HOL-LTEST-02	30GB	1.5 GB	27.5 GB
HE-HOL-WDEV-01	127 GB	60 GB	65 GB
HE-HOL-LDEV-01	30 GB	2 GB	26 GB

Azure DevTest Labs JSON files

To get a list of available size of VMs in Azure UK South, type the following command:

```
PS C:\> Get-AzureRmVMSize -Location uksouth
```

Description	osType	imageName	vhdFileName	Size	storageType	dnsServer
CC hands-on lab	Windows	CC-HOL-WDEV-01	CC-HOL-WDEV-01.vhd	Standard D2 v3		
CC hands-on lab	Linux	CC-HOL-LTEST-01	CC-HOL-LTEST-01.vhd	Standard DC2s v2		
CC hands-on lab	Linux	CC-HOL-LTEST-02	CC-HOL-LTEST-02.vhd	Standard DC2s v2		
HE hands-on lab	Windows	HE-HOL-WDEV-01	HE-HOL-WDEV-01.vhd	Standard D2 v3		
HE hands-on lab	Linux	HE-HOL-LDEV-01	HE-HOL-LDEV-01.vhd	Standard D2 v3		

List of credentials

Windows credentials

Below is the list of username and password of administrator accounts on each Windows VM:

Name of VM	Username	Password
CC-HOL-WDEV-01	azuredemo	Password.1!!
HE-HOL-WDEV-01	azuredemo	Password.1!!

Linux credentials for SSH connections

Below is the list of RSA SSH key sets for use on each Linux VM:

Name of VM	Username	Public key file	Private key file
CC-HOL-LTEST-01	azureadmin	id_rsa.pub	id_rsa
CC-HOL-LTEST-02	azureadmin	id_rsa.pub	id_rsa
HE-HOL-LDEV-01	azureadmin	id_rsa.pub	id_rsa

Appendix C: Download the VHD files to run them on-premises

If you want to download all the files at once, please use the following steps:

Download and Install AzCopy 8.1 available for Windows, Linux and Mac: <https://aka.ms/downloadazcopypr>

1. Navigate to the following folder `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Microsoft Azure\`
2. Open a PowerShell console.
3. Download all the VHDs and JSON files:

```
PS C:\> AzCopy /Source:https://compasshols.blob.core.windows.net/vhds-2020-04  
/Dest:%userprofile%\Vhds /S
```

If you want to download only specific files, you can refer to the following list of filenames:

	JSON Filename	VHD Filename	Track
1	<code>CC-HOL-WDEV-01.json</code>	<code>CC-HOL-WDEV-01.vhd</code>	Confidential Computing (CC)
2	<code>CC-HOL-LTEST-01.json</code>	<code>CC-HOL-LTEST-01.vhd</code>	Confidential Computing (CC)
3	<code>CC-HOL-LTEST-02.json</code>	<code>CC-HOL-LTEST-02.vhd</code>	Confidential Computing (CC)
4	<code>HE-HOL-WDEV-01.json</code>	<code>HE-HOL-WDEV-01.vhd</code>	Homomorphic Encryption (HE)
5	<code>HE-HOL-LDEV-01.json</code>	<code>HE-HOL-LDEV-01.vhd</code>	Homomorphic Encryption (HE)

For example, if you want to download only the `CC-HOL-WDEV-01.vhd` file, run the following command:

```
PS C:\>AzCopy /Source:https://compass.blob.core.windows.net/vhds-2020-04/ CC-HOL-WDEV-01.vhd  
/Dest:%userprofile%\vhds\ CC-HOL-WDEV-01.vhd
```


Appendix D: Add/Remove VMs according to a planning

If you want, to save cost, you can add or remove VM according to a schedule.

Below are some (series of) commands that can be run on purpose:

Commands #1:

```
PS C:\>. \New-CustomRole.ps1
PS C:\>. \New-EmptyLabs.ps1
PS C:\>. \Create-Vms.ps1 -ImagePattern "CC*" -IfExist "Error"
```

Command #2:

```
PS C:\>. \Remove-Vms.ps1 -MatchBy "Name" -Pattern "CC-HOL"
```

Commands #3:

```
PS C:\>. \Remove-Vms.ps1 -MatchBy "Note" -Pattern "CC"
PS C:\>. \Create-Vms.ps1 -ImagePattern "CC*" -IfExist "Error"
```

Commands #4:

```
PS C:\>. \Remove-Vms.ps1 -MatchBy "Name" -Pattern "CC-HOL-WDEV-01"
PS C:\>. \Remove-Vms.ps1 -MatchBy "Name" -Pattern "CC-HOL-LTEST-01"
PS C:\>. \Remove-Vms.ps1 -MatchBy "Name" -Pattern "CC-HOL-LTEST-02"
```

Commands #5:

```
PS C:\>. \Create-Vms.ps1 -ImagePattern "CC-HOL" -IfExist "Error"
```

Commands #6:

```
PS C:\>. \Remove-Labs.ps1
```