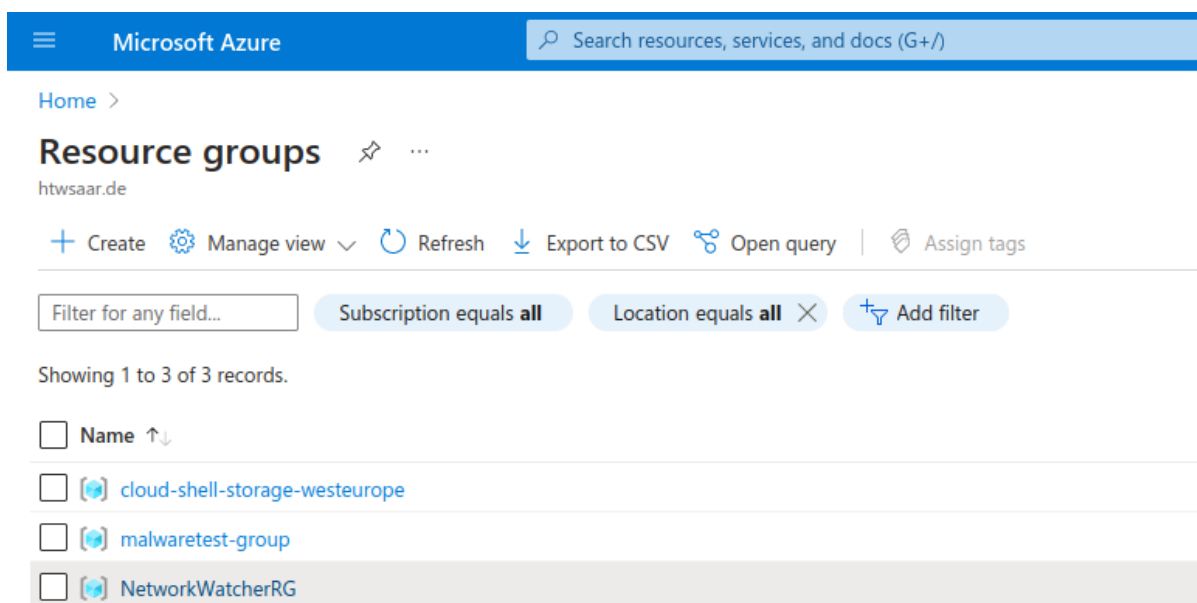


malware test

this code will help you generate a azure cloud architecture with 2 machines one containing the malware library and the other the agent with a pre installed share and also the agent you just need to install it setup the policy to aggressive and try it

prerequisite

- having a azure license
- having azure cloud on your machine and connected to your azure account
- rename the terraform.tfvars.template to terraform.tfvars
- add your IP to the **ip-whitelist** inside the *terraform.tfvars* file
- copy your agent in **ansible/roles/win10/files** and rename it edrinstaller.msi | edinstaller.exe
- go to **terraform/04 ansible** and comment out the line with **--skip-tags "install_msi"** if you have an exe or **--skip-tags "install_exe"** if you have an msi.
- having a resource group named *malwaretest-group*

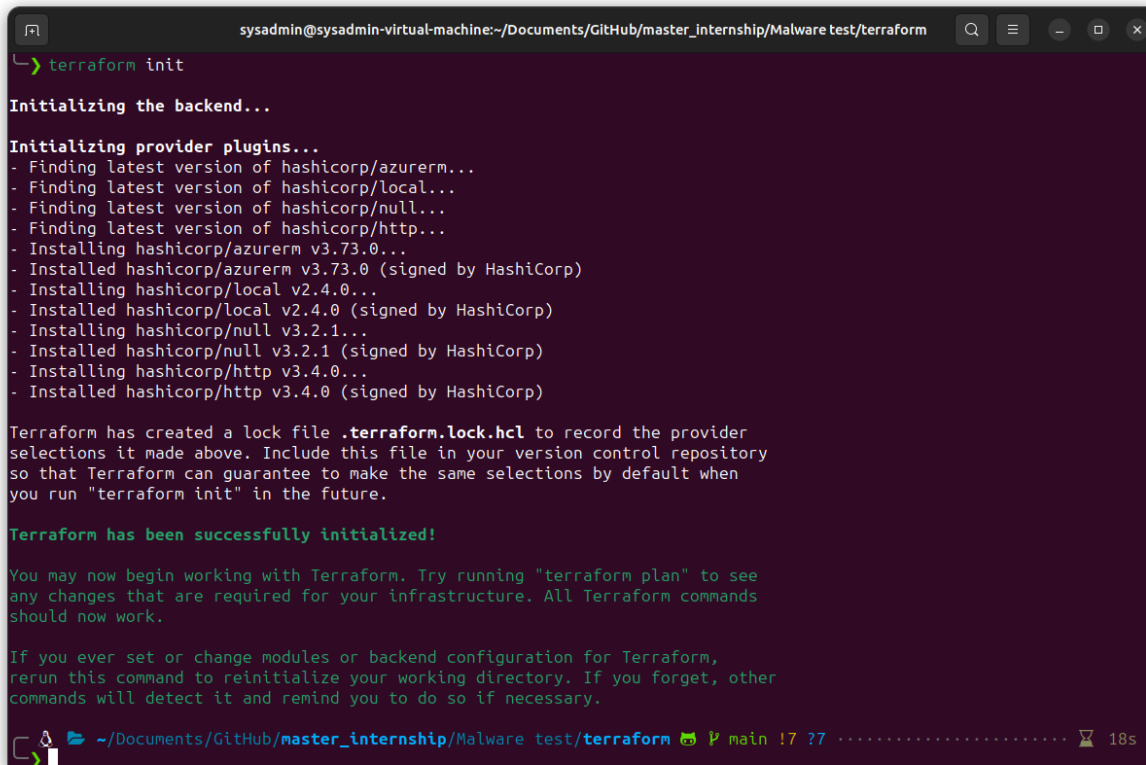


creating the environment

for executing the terraform code you need to move to the folder *terraform*

execute the following commandes:

- **terraform init** this will install the necessary dependencies.



```

sysadmin@sysadmin-virtual-machine:~/Documents/GitHub/master_internship/Malware test/terraform
$ terraform init

Initializing the backend...

Initializing provider plugins...
- Finding latest version of hashicorp/azurerm...
- Finding latest version of hashicorp/local...
- Finding latest version of hashicorp/null...
- Finding latest version of hashicorp/http...
- Installing hashicorp/azurerm v3.73.0...
- Installed hashicorp/azurerm v3.73.0 (signed by HashiCorp)
- Installing hashicorp/local v2.4.0...
- Installed hashicorp/local v2.4.0 (signed by HashiCorp)
- Installing hashicorp/null v3.2.1...
- Installed hashicorp/null v3.2.1 (signed by HashiCorp)
- Installing hashicorp/http v3.4.0...
- Installed hashicorp/http v3.4.0 (signed by HashiCorp)

Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

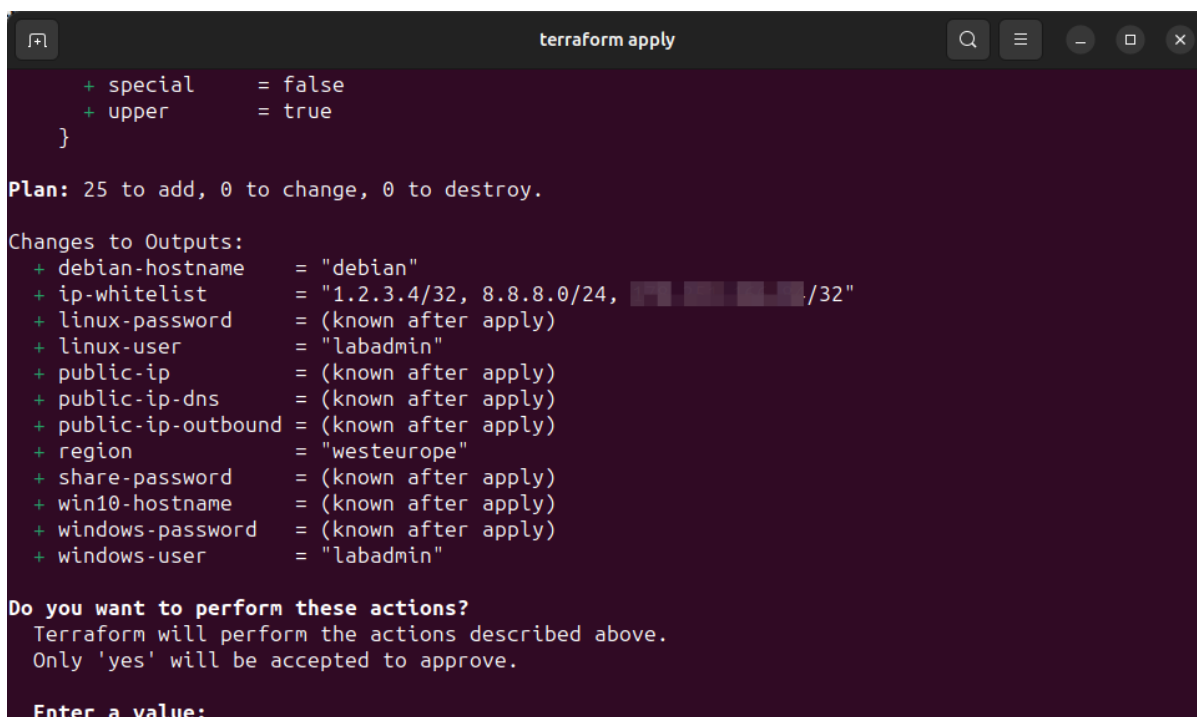
Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

```

- **terraform plan** this will plan the resources
- **terraform apply** this will execute the plan, here you need to type **yes** when asked



```

terraform apply

+ special      = false
+ upper        = true
}

Plan: 25 to add, 0 to change, 0 to destroy.

Changes to Outputs:
+ debian-hostname = "debian"
+ ip-whitelist    = "1.2.3.4/32, 8.8.8.0/24, [REDACTED]/32"
+ linux-password  = (known after apply)
+ linux-user      = "labadmin"
+ public-ip       = (known after apply)
+ public-ip-dns   = (known after apply)
+ public-ip-outbound = (known after apply)
+ region          = "westeurope"
+ share-password  = (known after apply)
+ win10-hostname  = (known after apply)
+ windows-password = (known after apply)
+ windows-user    = "labadmin"

Do you want to perform these actions?
Terraform will perform the actions described above.
Only 'yes' will be accepted to approve.

Enter a value:

```

after it was succesful it will prompt you the username of both machines and there passwords in case you forgote the password you can use **terraform output** to display it again

```

sysadmin@sysadmin-virtual-machine:~/Documents/GitHub/master_internship/Malware test/ter...
6  unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
null_resource.ansible-provisioning (remote-exec): 10.13.37.200 : ok=15  changed=
11  unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

null_resource.ansible-provisioning: Creation complete after 13m6s [id=4992478808861520211]

Apply complete! Resources: 25 added, 0 changed, 0 destroyed.

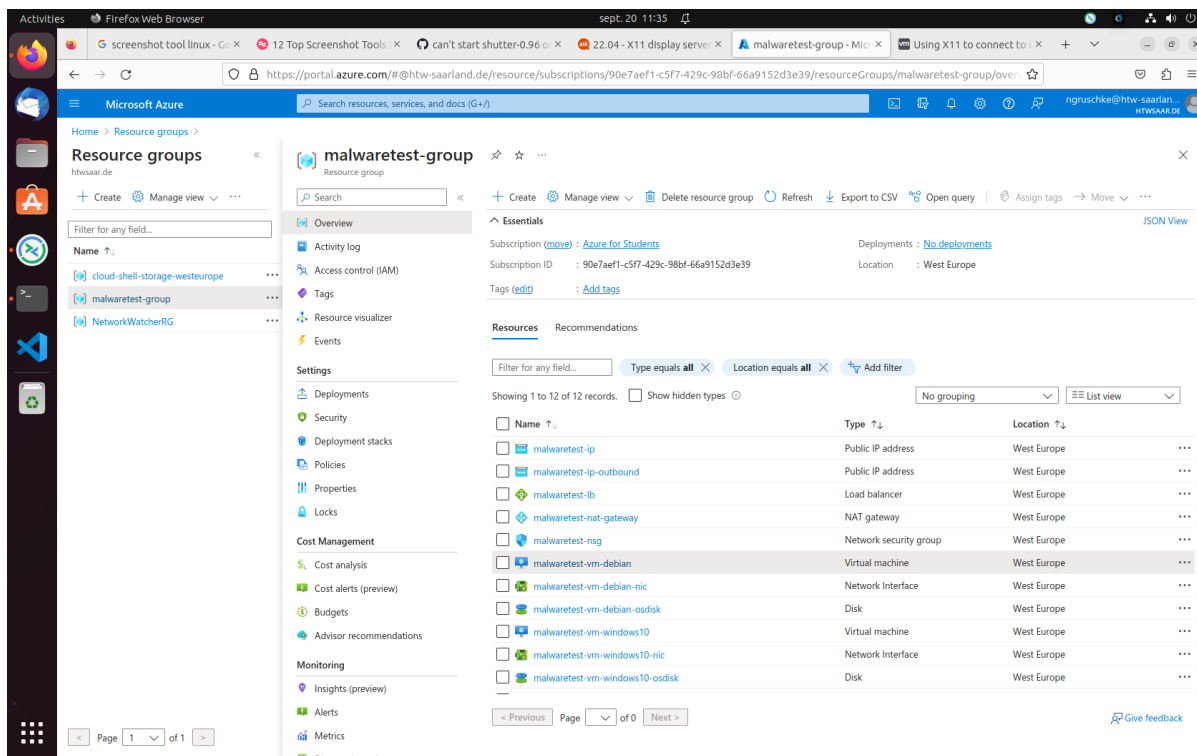
Outputs:

debian-hostname = "debian"
ip-whitelist = "1.2.3.4/32, 8.8.8.0/24, [REDACTED]/32"
linux-password = "Kixi3N3l7P8YRZsW"
linux-user = "labadmin"
public-ip = "4.180.70.178"
public-ip-dns = "malwaretest.westeurope.cloudapp.azure.com"
public-ip-outbound = "20.160.166.248"
region = "westeurope"
share-password = "K2xbrZ"
win10-hostname = "win10-iwvWA"
windows-password = "hRvOpVv7ahnUXIl3"
windows-user = "labadmin"

~/Doc/GitHub/master_internship/Malware test/terraform main !8 ?10 ... 19m 10s

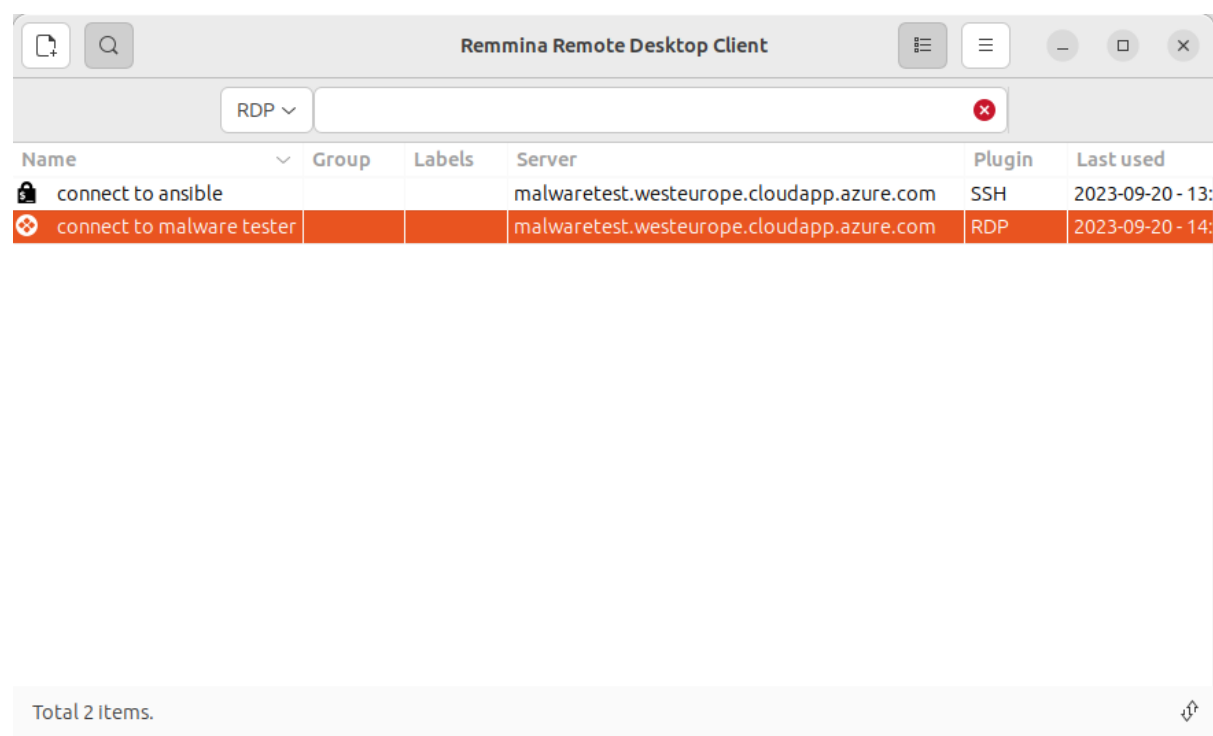
```

If you go to the azure page you will see the following



executing the test

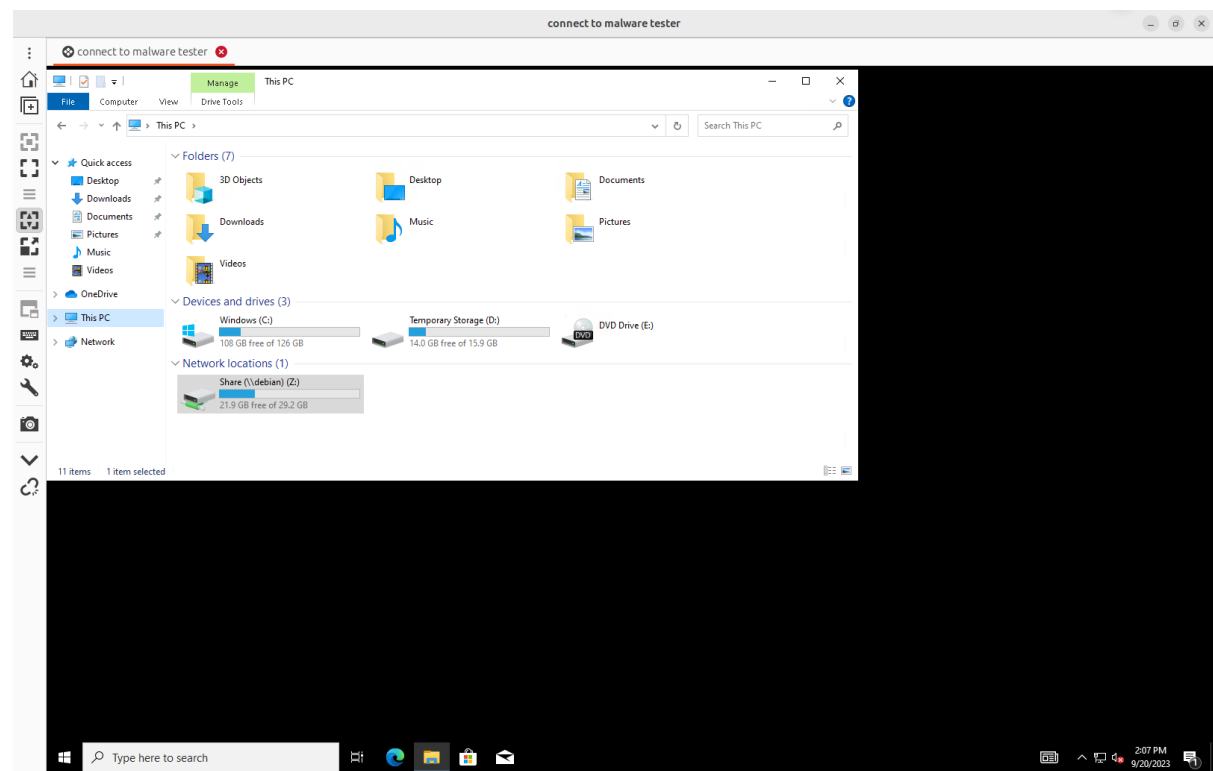
after the apply run successfully, you can copy the public-ip-dns value, this will be useful if you want to connect to the debian machine or/and to the windows machine.



the port that are available are the port 22 to ssh to the debian machine and the port 3389 to RDP to the windows machine. Since you added your public IP to the whitelist it can only be done from a machine in your network.

if you ssh to debian machine, after inserting the password you have to `sudo su` and then to `cd /usr/share/theZoo` to see all the malware which will be copied.

when you connect via RDP to the windows machine you will see the share in your explorer.



Here it can happen that the password is missing or the share is not available, to fix this issue with the password you only have to reenter it (the password can be fount in the terraform output) with the user

"theZoo". In case you cannot access it you just need to delete it and redo a mapping to the remote "`\debian\Share`".

since the agent is install by default you juste need to check if it is correctly installed(some times you need to disable microsoft defender), and then to copy the content of the share to the windows PC and count how many files are remaining.

destroy the environment

After you don't need the environment anymore like test finish or changing the installer you can run the command `terraform destroy --auto-approve` inside the Azure cloud console and all resources generated by terraform will be destroy. and it will look like this .

Home > malwaretest-group

Search

Create Manage view Delete resource group Refresh Export to CSV Open query Assign tags Move Delete Export template

Overview

Activity log

Access control (IAM)

Tags

Resource visualizer

Events

Settings

Deployments

Security

Deployment stacks

Policies

Properties

Locks

Cost Management

Cost analysis

Cost alerts (preview)

Budgets

Advisor recommendations

Monitoring

Insights (preview)

Alerts

Monitoring

Essentials

Subscription (move): Azure for Students

Deployments: No deployments

Subscription ID: 90e7aef1-c5f7-429c-98bf-66a9152d3e39

Location: West Europe

Tags (edit): Add tags

Resources Recommendations

Filter for any field... Type equals all Location equals all Add filter

Showing 0 to 0 of 0 records. Show hidden types

No grouping

Name Type Location

No resources match your filters

Try changing or clearing your filters.

Create resources Clear filters

Learn more