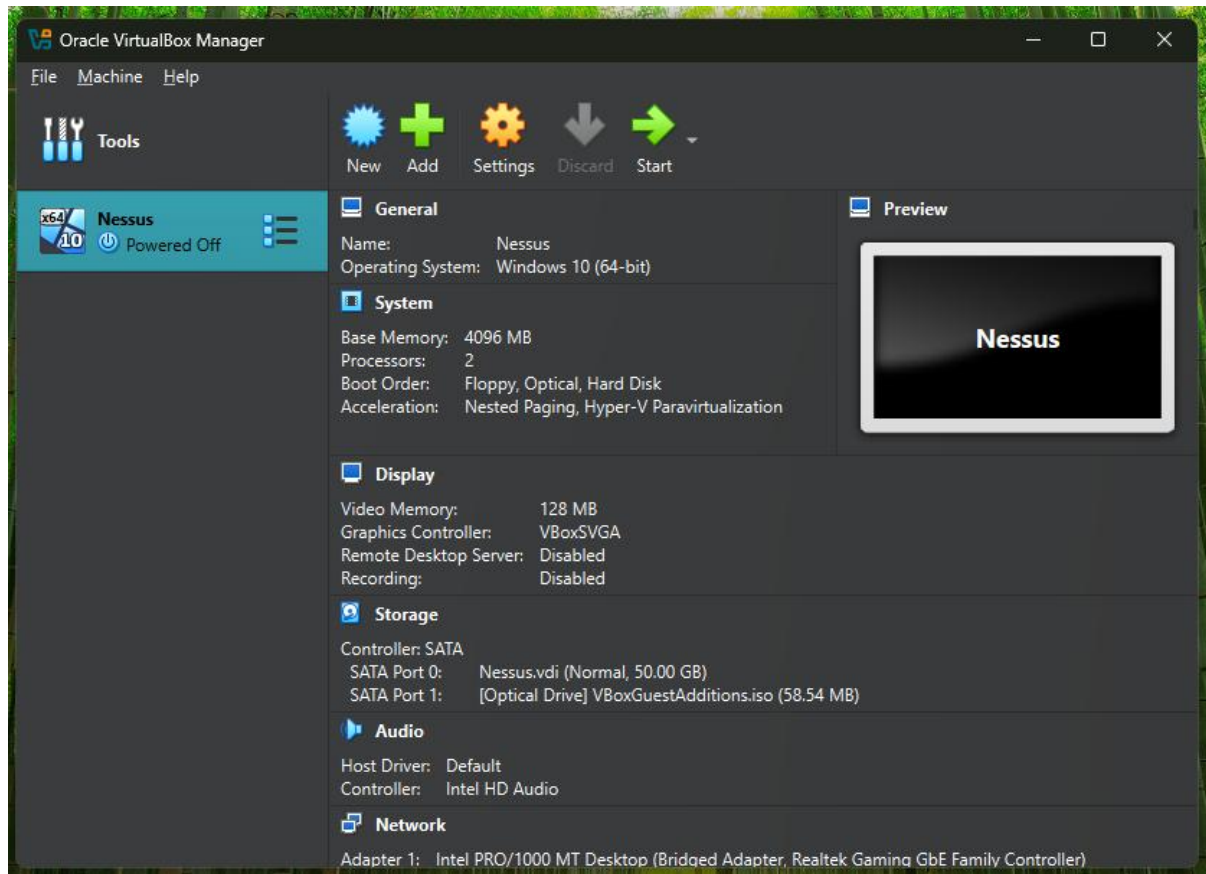


Steps to Perform(using PowerShell)

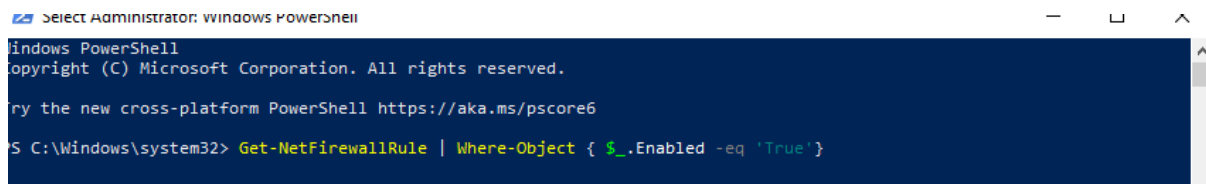
Step 1: Initialize VMware

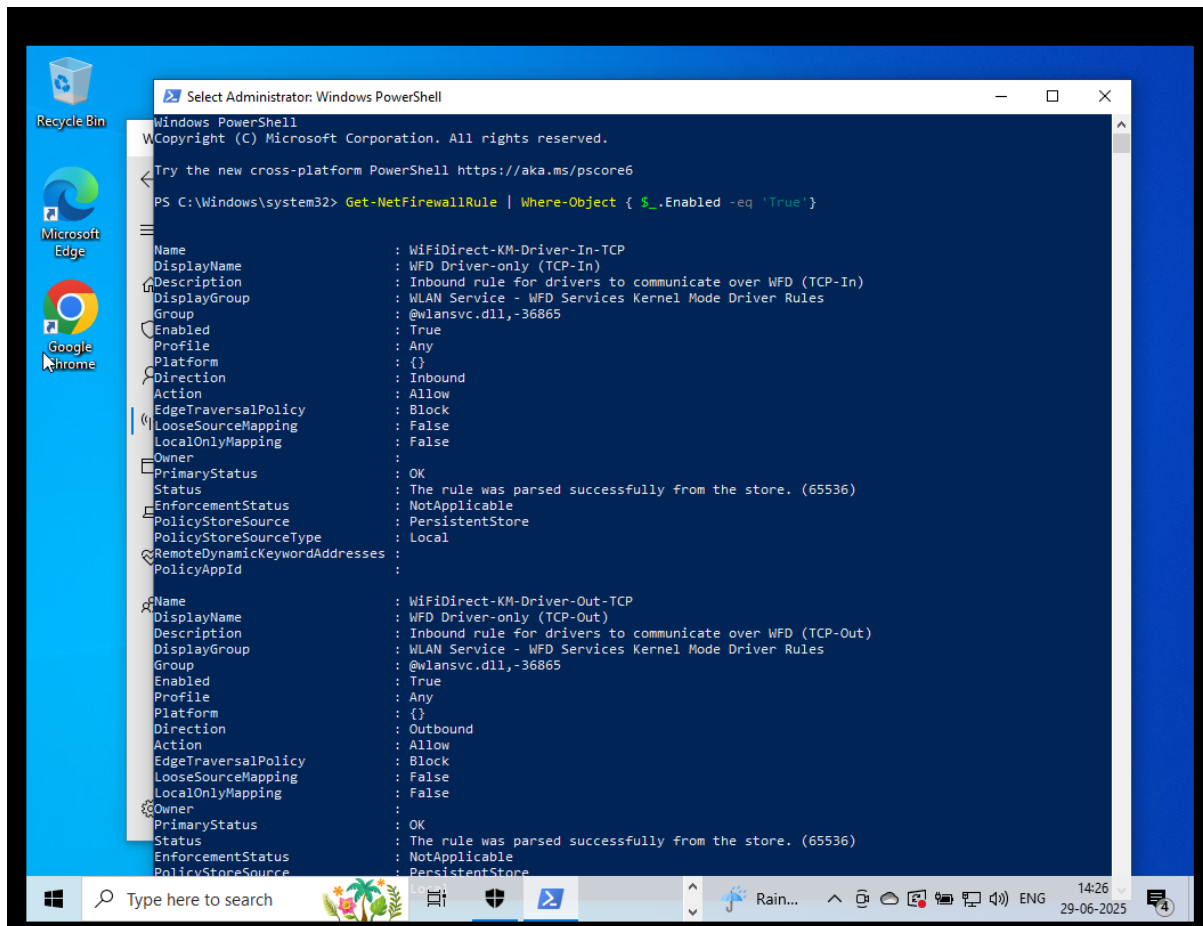
- Setup vmware and open windows on vm



Step 2: List Existing Firewall Rules

```
Get-NetFirewallRule | Where-object { $_.Enable -eq 'True' }
```

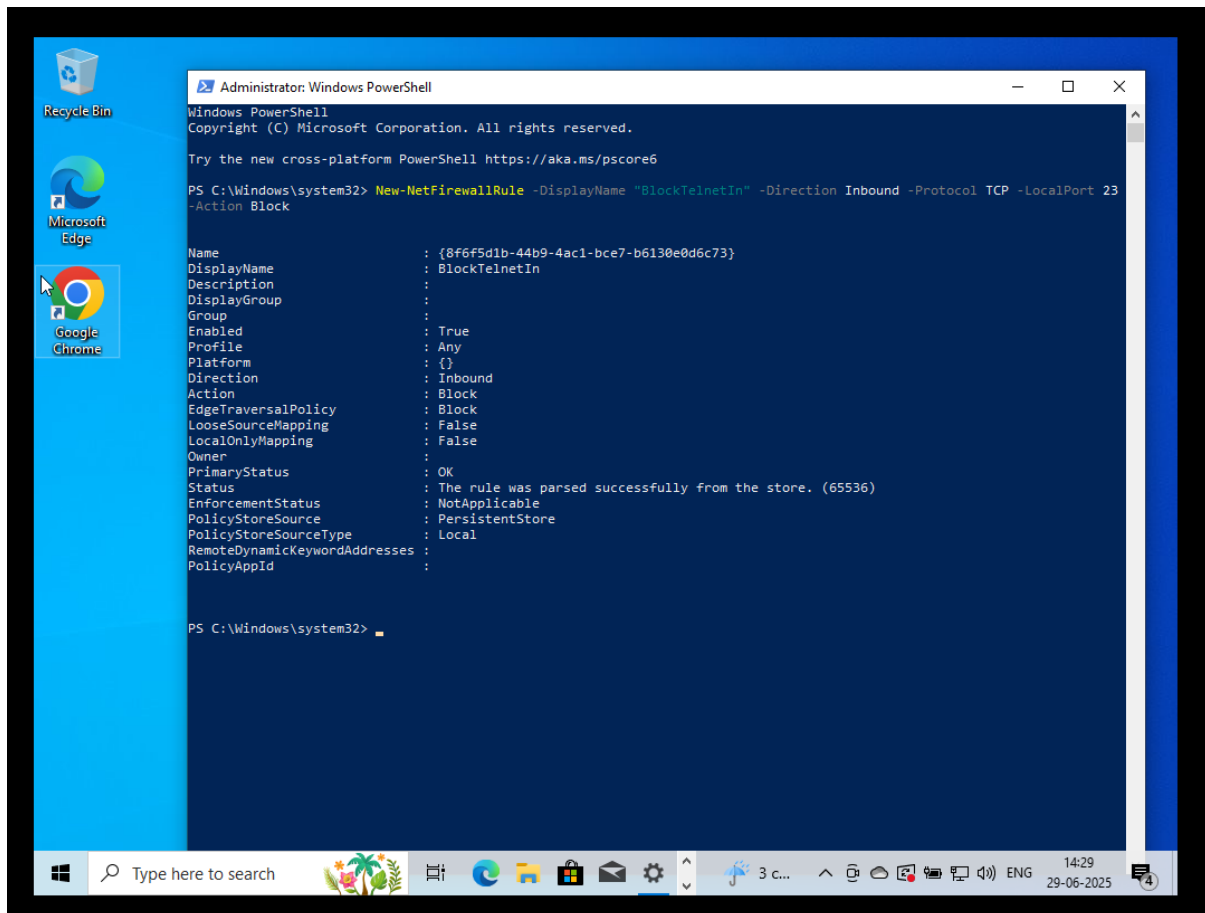




Step 3: Block Inbound Traffic on Port 23 (Telnet):

```
New-NetFirewallRule -DisplayName "BlockTelnetIn" -
Direction Inbound -Protocol TCP -LocalPort 23 -Action
Block
```

```
PS C:\Windows\system32> New-NetFirewallRule -DisplayName "BlockTelnetIn" -Direction Inbound -Protocol TCP -LocalPort 23
-Action Block
```



Step 4: Test the Blocked Port:

- Open PowerShell and try:

```
Test-NetFirewallRule -ComputerName 127.0.0.1 -port 23
```

```
PS C:\Windows\system32> Test-NetFirewallRule -ComputerName 127.0.0.1 -port 23
```

```
PS C:\Windows\system32> Test-NetConnection -ComputerName 127.0.0.1 -port 23  
WARNING: TCP connect to (127.0.0.1 : 23) failed
```

```
ComputerName      : 127.0.0.1  
RemoteAddress     : 127.0.0.1  
RemotePort        : 23  
InterfaceAlias    : Loopback Pseudo-Interface 1  
SourceAddress     : 127.0.0.1  
PingSucceeded     : True  
PingReplyDetails (RTT) : 0 ms  
TcpTestSucceeded  : False
```

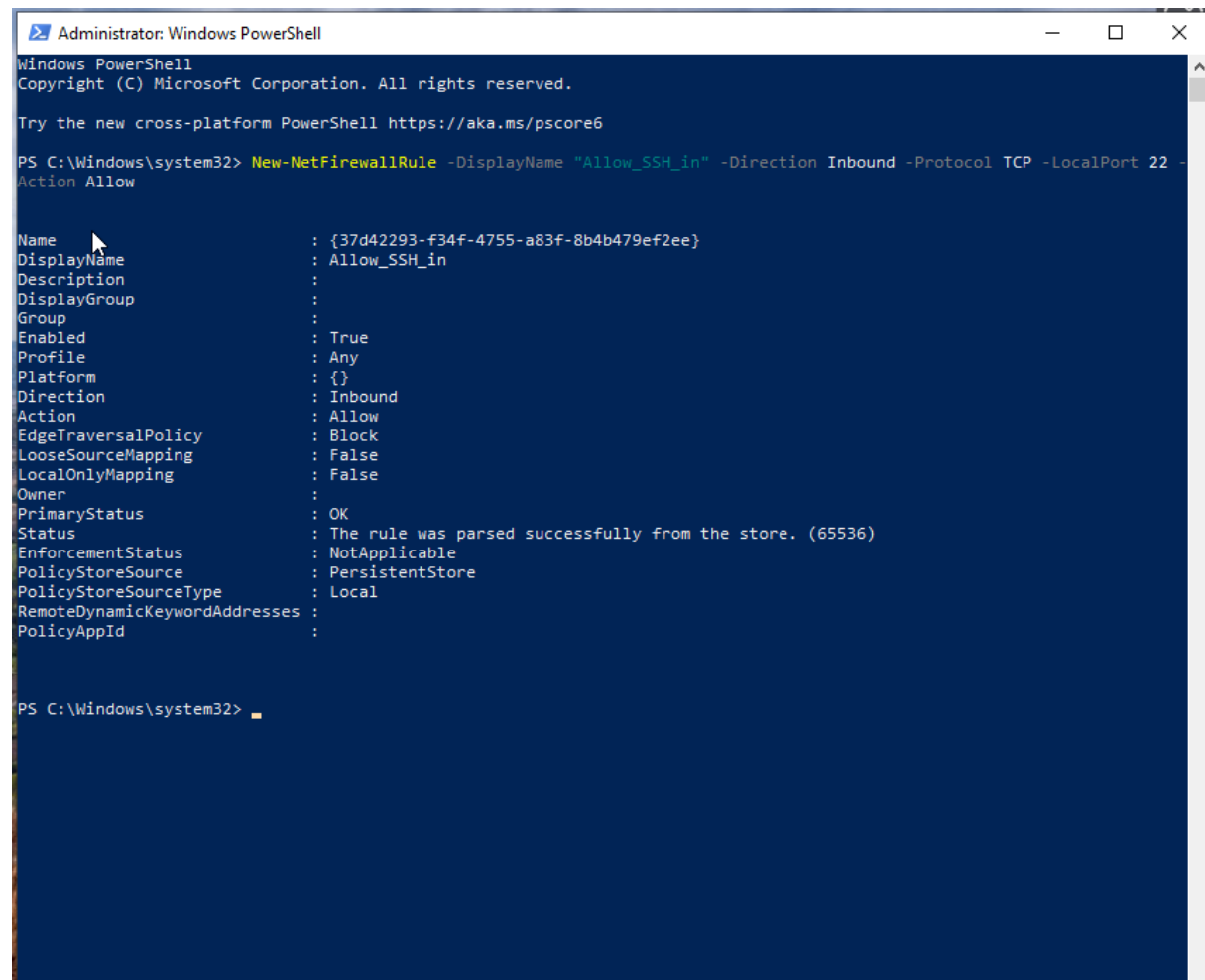
- Failed TCP test confirms the port is blocked

Step 5: Allow SSH Traffic on Port 22:

- If using WSL or SSH server:

```
New-NetFirewallRule -DisplayName "Allow_SSH_In" -
Direction Inbound -Protocol TCP -LocalPort 22 -Action
Allow
```

```
PS C:\Windows\system32> New-NetFirewallRule -DisplayName "Allow_SSH_in" -Direction Inbound -Protocol TCP -LocalPort 22 -
Action Allow
```



The screenshot shows a Windows PowerShell window titled "Administrator: Windows PowerShell". The prompt is "PS C:\Windows\system32>". The command entered is `New-NetFirewallRule -DisplayName "Allow_SSH_in" -Direction Inbound -Protocol TCP -LocalPort 22 -Action Allow`. The output displays the properties of the newly created firewall rule:

```
Name : {37d42293-f34f-4755-a83f-8b4b479ef2ee}
DisplayName : Allow_SSH_in
Description :
DisplayGroup :
Group :
Enabled : True
Profile : Any
Platform : {}
Direction : Inbound
Action : Allow
EdgeTraversalPolicy : Block
LooseSourceMapping : False
LocalOnlyMapping : False
Owner :
PrimaryStatus : OK
Status : The rule was parsed successfully from the store. (65536)
EnforcementStatus : NotApplicable
PolicyStoreSource : PersistentStore
PolicyStoreSourceType : Local
RemoteDynamicKeywordAddresses :
PolicyAppId :
```

The prompt returns to `PS C:\Windows\system32>`.

Step 6: Remove the Block Telnet Rule:

```
Remove-NetFirewallRule -DisplayName "BlockTelnetIn"
```

```
PS C:\Windows\system32> Remove-NetFirewallRule -DisplayName "BlockTelnetIn"
```

- To confirm rule is deleted try:

```
Get-NetFirewallRule -DisplayName "BlockTelnetIn"
```

```
PS C:\Windows\system32> Remove-NetFirewallRule -DisplayName "BlockTelnetIn"
PS C:\Windows\system32> Get-NetFirewallRule -DisplayName "BlockTelnetIn"
Get-NetFirewallRule : No MSFT_NetFirewallRule objects found with property 'DisplayName' equal to 'BlockTelnetIn'.
Verify the value of the property and retry.
At line:1 char:1
+ Get-NetFirewallRule -DisplayName "BlockTelnetIn"
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (BlockTelnetIn:String) [Get-NetFirewallRule], CimJobException
+ FullyQualifiedErrorId : CmdletizationQuery_NotFound_DisplayName,Get-NetFirewallRule

PS C:\Windows\system32> █
```