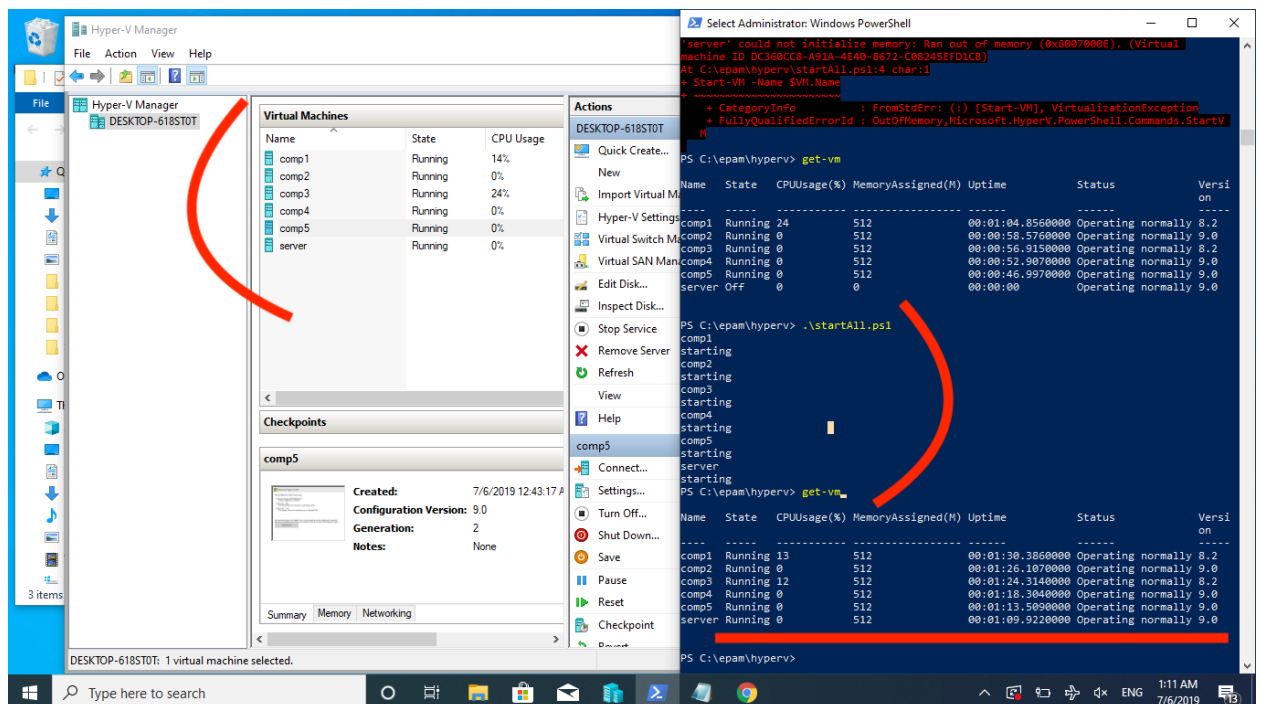


## Dennis Tikhomirov. DevOps Spring'19.

### Task 3. Hyper-V

- Using the GUI, create the following VM without a network connection:
  - Windows 7 (comp1)
  - Windows Server 2016 (server)
  - Ubuntu 18.04 (comp2)
- Using the PowerShell, create the following VM without a network connection:
  - Windows 7 (comp3)
  - Windows 10 (comp4)
  - Ubuntu 19.04 (comp5)

Screenshot #1. Hyper-V manager, deployed VMs



I have created bunch of scripts for managing VMs on Hyper-V host.

#1. List of VMs <https://github.com/dennis00010011b/epam-devops-training/blob/master/Task3HyperV/VMs.xml>

```
<?xml version="1.0"?>
<VMs>
  <VM>
    <Name>comp1</Name>
    <SourcePath>C:\ISOs\IE11.Win7.HyperV\Virtual Machines\106A06B0-0DE9-4997-A87C-3760FFBEC837.vmcx</SourcePath>
    <IsImport>true</IsImport>
    <Memory>536870912</Memory>
    <VHDPATH>./</VHDPATH>
    <OS>Wni7</OS>
  </VM>

  <VM>
    <Name>comp2</Name>
    <SourcePath>C:\ISOs\ubuntu-18.04.2-desktop-amd64.iso</SourcePath>
    <Memory>536870912</Memory>
    <VHDPATH>./</VHDPATH>
    <OS>Ubuntu18.04</OS>
  </VM>

  <VM>
    <Name>comp3</Name>
    <SourcePath>C:\ISOs\IE11.Win7.HyperV\Virtual Machines\106A06B0-0DE9-4997-A87C-3760FFBEC837.vmcx</SourcePath>
    <IsImport>true</IsImport>
    <Memory>536870912</Memory>
    <VHDPATH>./</VHDPATH>
    <OS>Wni7</OS>
  </VM>

  <VM>
    <Name>comp4</Name>
    <SourcePath>C:\ISOs\Win10_18090ct_v2_English_x64.iso</SourcePath>
    <Memory>536870912</Memory>
    <VHDPATH>./</VHDPATH>
    <OS>Wni10</OS>
  </VM>

  <VM>
    <Name>comp5</Name>
    <SourcePath>C:\ISOs\ubuntu-19.04-desktop-amd64.iso</SourcePath>
    <Memory>536870912</Memory>
    <VHDPATH>./</VHDPATH>
    <OS>Ubuntu19.04</OS>
  </VM>

  <VM>
    <Name>server</Name>
    <SourcePath>C:\ISOs\Windows_Server_2016_Datacenter_EVAL_en-us_14393_refresh.ISO</SourcePath>
    <Memory>1036870912</Memory>
    <VHDPATH>./</VHDPATH>
    <OS>Ubuntu19.04</OS>
  </VM>
</VMs>
```

#2. PS script for creating VMs <https://github.com/dennis00010011b/epam-devops-training/blob/master/Task3HyperV/createVM.ps1>

```
$VMName = $args[0]
$ISO = $args[1]
$Memory = $args[2]
$VHDPATH = $args[3]
$VM = @{
    Name = $VMName
    MemoryStartupBytes = $Memory
    Generation = 2
    NewVHDPATH = "$VHDPATH$VMName.vhdx"
    NewVHDSIZEBytes = 53687091200
}
echo Creating $VMName $ISO $Memory $VHDPATH
New-VM @VM

Add-VMdvdDrive -VMName $VMName -Path $ISO

$firmw = Get-VMFirmware $VMName
Set-VMFirmware -VMName $VMName -BootOrder $firmw.BootOrder[2]
```

#3. PS script for deploying VMs <https://github.com/dennis00010011b/epam-devops-training/blob/master/Task3HyperV/deploy.ps1>

```
[xml]$XmlDoc = Get-Content VMs.xml

foreach ($VM in $XmlDoc.VMs.VM) {
    if ($VM.isImport) {
        Import-VM -Path $VM.SourcePath -VhdDestinationPath $VM.VHDPATH -Copy -
        GenerateNewId
    }
    else{
        .\createVM.ps1 $VM.Name $VM.SourcePath $VM.Memory $VM.VHDPATH
    }
}
```

#4. PS script for removing VMs <https://github.com/dennis00010011b/epam-devops-training/blob/master/Task3HyperV/removeAll.ps1>

```
[xml]$XmlDoc = Get-Content VMs.xml
foreach ($VM in $XmlDoc.VMs.VM) {
    remove-vm $VM.Name -force
    if (-Not($VM.isImport)) {
        remove-item "$(Join-Path $VM.VHDPATH $VM.Name).vhdx"
    }
}
```

### 3. Using the PowerShell for computers “comp1”, “comp4”, “comp5” configure NAT and Internet access

```
# Create internal vSwitch
new-vmswitch -SwitchName "IntSwitch" -SwitchType Internal

# Create IPGateway
new-netipaddressstIPAddress 192.168.2.1 -PrefixLength 24 -InterfaceAlias "vEthernet
(IntSwitch)"

#Create NAT network
new-netnat -name "NAT55" -InternalIPInterfaceAddressPrefix 192.168.2.0/24

#Connect VMs to vSwitch
connect-vmnetworkadapter -VMName comp1,comp4,comp5 -SwitchName IntSwitch

# set manually IPAdress on each VM
# for comp4
new-netipaddress 192.168.2.2 -InterfaceAlias "Ethernet"

# for comp#1 (Windows7)
$adapter = Get-WmiObject win32_networkadapterconfiguration -filter "ipenabled =
'true'"
$adapter.EnableStatic("192.168.2.4", "255.255.255.0")

#for comp#5 (Ubuntu 19.04)
sudo ifconfig eth0 add 192.168.2.10 netmask 255.255.255.0
```

### 4. Check your settings from the command line (terminal)

Screenshot #2. Connection between comp1,comp2,host via internal vSwitch

The screenshot displays a virtual machine environment with three main components: Comp1 (Windows 7), Comp4 (Windows 7), and Host (Windows 10). The Host's PowerShell console shows the configuration of a NAT network named 'NAT55' and the connection of the three VMs to the internal vSwitch 'IntSwitch'. The IP addresses are set as follows: Host (192.168.2.1), Comp1 (192.168.2.4), Comp4 (192.168.2.2), and another VM (192.168.2.10). The screenshot also shows the results of ping tests between the VMs and the Host, confirming connectivity. The Host's network configuration is shown in the bottom right, including the NAT55 network and the connection of the three VMs to the internal vSwitch.

Host PowerShell console output:

```
PS C:\Windows\system32> get-netnat
Name : NAT55
InternalIPInterfaceAddressPrefix : 192.168.2.0/24
ExternalIPInterfaceAddressPrefix : 192.168.2.0/24
ICMPQueryTimeout : 30
TcpEstablishedConnectionTimeout : 1800
TcpTransientConnectionTimeout : 120
TcpFilteringBehavior : AddressDependentFiltering
UdpFilteringBehavior : AddressDependentFiltering
UdpIdleSessionTimeout : 120
UdpInboundRefresh : False
Store : Local
Active : True

PS C:\Windows\system32> ping 192.168.2.2
Pinging 192.168.2.2 with 32 bytes of data:
Reply from 192.168.2.2: bytes=32 time<1ms TTL=128
Reply from 192.168.2.2: bytes=32 time<1ms TTL=128
Reply from 192.168.2.2: bytes=32 time<1ms TTL=128
Reply from 192.168.2.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Windows\system32> ping 192.168.2.4
Pinging 192.168.2.4 with 32 bytes of data:
Reply from 192.168.2.4: bytes=32 time<1ms TTL=128
Reply from 192.168.2.4: bytes=32 time<1ms TTL=128
Reply from 192.168.2.4: bytes=32 time<1ms TTL=128
Reply from 192.168.2.4: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Windows\system32>
```

### Screenshot #3. Connection between comp5 ,host via internal vSwitch

The screenshot shows a Hyper-V virtual machine connection window. On the left, a terminal window titled 'Comp5' shows a Linux shell prompt 'dennis@dennis-Virtual-Machine:~'. The terminal output displays the results of a ping command from 192.168.2.1 to 192.168.2.10, showing successful connectivity with 0% packet loss. On the right, a Windows PowerShell window titled 'Administrator: Windows PowerShell' shows the output of a 'ping 192.168.2.10' command, which also shows successful connectivity with 0% packet loss. The PowerShell window also displays the configuration for the 'vEthernet' network adapter, showing it is connected to the 'IntSwitch2' internal switch.

5. Using the PowerShell, add the remaining VM into a network by creating a new vSwitch  
New internal vSwitch is created, 192.168.2.101

```
# One more vSwitch
new-vm switch -SwitchName "IntSwitch2" -SwitchType Internal
new-netipaddress -IPaddress 192.168.2.101 -PrefixLength 24 -InterfaceAlias "vEthernet
(IntSwitch2)"
connect-vmnetworkadapter -VMName comp2,comp3,server -SwitchName IntSwitch2

# set manually IPAddress on each VM
# for 'server' (Windows Server 2016)
new-netipaddress 192.168.2.102 -PrefixLength 24 -InterfaceAlias "Ethernet"

#for comp#2 (Ubuntu 18.04)
sudo ifconfig eth0 add 192.168.2.104 netmask 255.255.255.0

# for comp#3 (Windows7)
$adapter = Get-WmiObject win32_networkadapterconfiguration -filter "ipenabled =
'true'"
$adapter.EnableStatic("192.168.2.103", "255.255.255.0")
```

Screenshot #4. Connection between comp3,server, host via internal vSwitch

The screenshot displays three overlapping Windows command prompt windows, each titled with a host name and IP address. The windows show network configuration details and successful ping tests between the three hosts.

**Server 192.168.2.102**

```
Administrator: C:\Windows\system32\cmd.exe - powershell
Link-local IPv6 Address . . . . . : fe80::700b:4a34:d1fa:3f2c%4
IPv4 Address. . . . . : 192.168.2.3
Subnet Mask . . . . . : 255.255.255.0
IPv4 Address. . . . . : 192.168.2.102
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :

Tunnel adapter isatap.{0BAC1508-6518-4645-9B70-4BE3249F0695}:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

PS C:\Users\Administrator> ping 192.168.2.101

Pinging 192.168.2.101 with 32 bytes of data:
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Users\Administrator> ping 192.168.2.103

Pinging 192.168.2.103 with 32 bytes of data:
Reply from 192.168.2.103: bytes=32 time=1ms TTL=128
Reply from 192.168.2.103: bytes=32 time=2ms TTL=128
Reply from 192.168.2.103: bytes=32 time<1ms TTL=128
Reply from 192.168.2.103: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 2ms, Average = 0ms
PS C:\Users\Administrator>
```

**Comp3 192.168.2.103**

```
Select Administrator: Windows PowerShell
CLASS : _PARAMETERS
SUPERCLASS : _PARAMETERS
CLASS : _PARAMETERS
ELPATH :
PROPERTY_COUNT : 1
EVALUATION : <>
ERROR :
NAMESPACE :
PATH :
ReturnValue : 0

C:\Users\IEUser> ipconfig

Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::11f1:9f49:93a5:
IPv4 Address. . . . . : 192.168.2.103
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :

Tunnel adapter isatap.{0748E33C-9018-4D20-AF60-198B5F882C1D}:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :
C:\Users\IEUser> ping 192.168.2.101

Pinging 192.168.2.101 with 32 bytes of data:
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128
Reply from 192.168.2.101: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\IEUser>
```

**Host 192.168.2.101**

```
PS C:\Windows\system32> ping 192.168.2.102

Pinging 192.168.2.102 with 32 bytes of data:
Reply from 192.168.2.102: bytes=32 time<1ms TTL=128
Reply from 192.168.2.102: bytes=32 time<1ms TTL=128
Reply from 192.168.2.102: bytes=32 time<1ms TTL=128
Reply from 192.168.2.102: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.102:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Windows\system32> ping 192.168.2.103

Pinging 192.168.2.103 with 32 bytes of data:
Reply from 192.168.2.103: bytes=32 time<1ms TTL=128
Reply from 192.168.2.103: bytes=32 time<1ms TTL=128
Reply from 192.168.2.103: bytes=32 time<1ms TTL=128
Reply from 192.168.2.103: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.2.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Windows\system32> ping 192.168.2.104

Pinging 192.168.2.104 with 32 bytes of data:
Reply from 192.168.2.104: bytes=32 time<1ms TTL=64
Reply from 192.168.2.104: bytes=32 time<1ms TTL=64
Reply from 192.168.2.104: bytes=32 time<1ms TTL=64
Reply from 192.168.2.104: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.2.104:
    Packets: Sent = 4, Received = 4, Lost = 0 (0%
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Windows\system32>
```

## 6. Configure DHCP on “server”

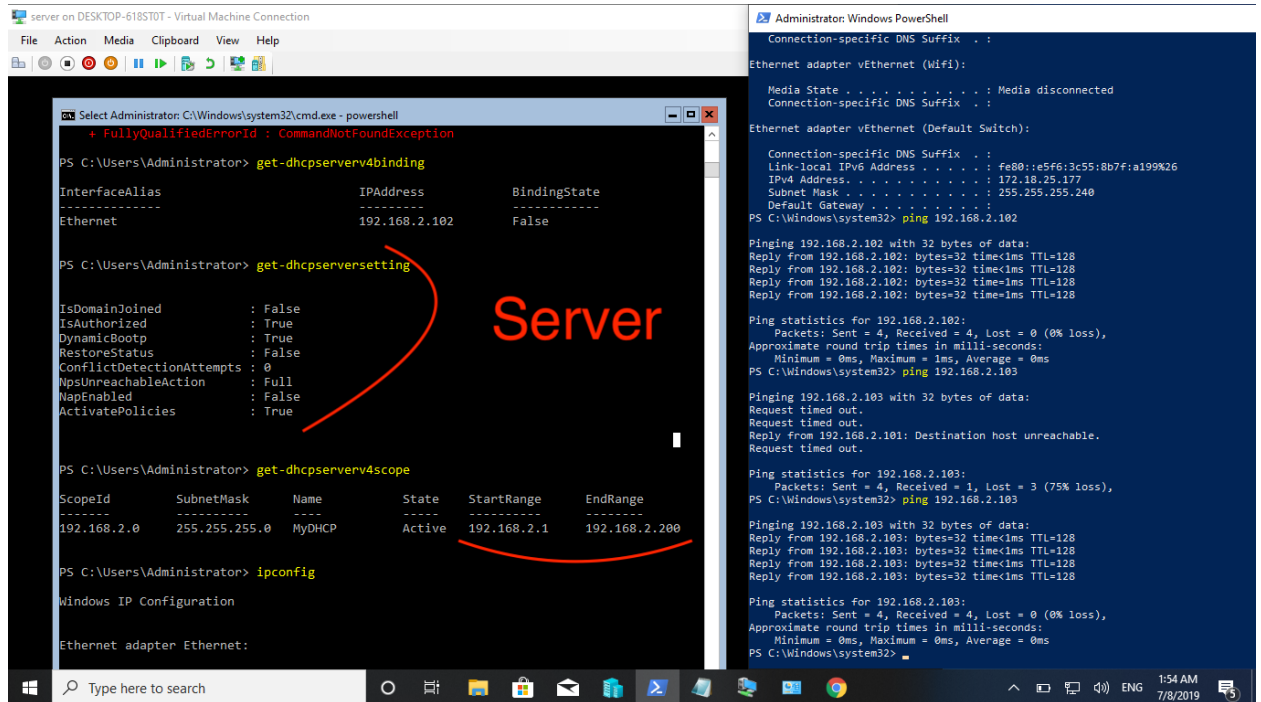
### Screenshot #5. DHCP server configuration

```
# Install DHCP
```

```
Import-Module ServerManager  
Add-WindowsFeature -name DHCP -IncludeManagementTools
```

```
#Add Scope
```

```
Add-DHCPServerV4Scope -Name Pool1 -StartRange 192.168.2.1 -EndRange 192.168.2.200 -  
SubnetMask 255.255.255.0 -State Active
```



## 7. Configure the LAN and Internet access

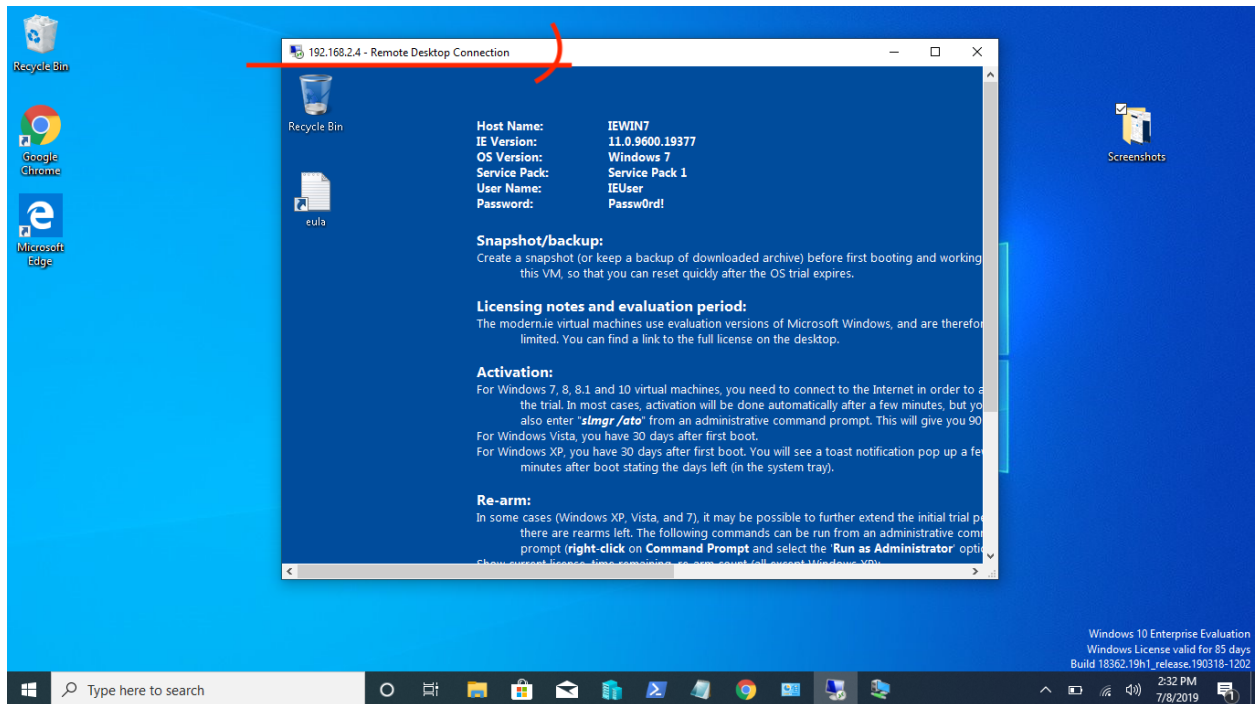
## 8. Check your settings from the command line (terminal)

See screenshots above

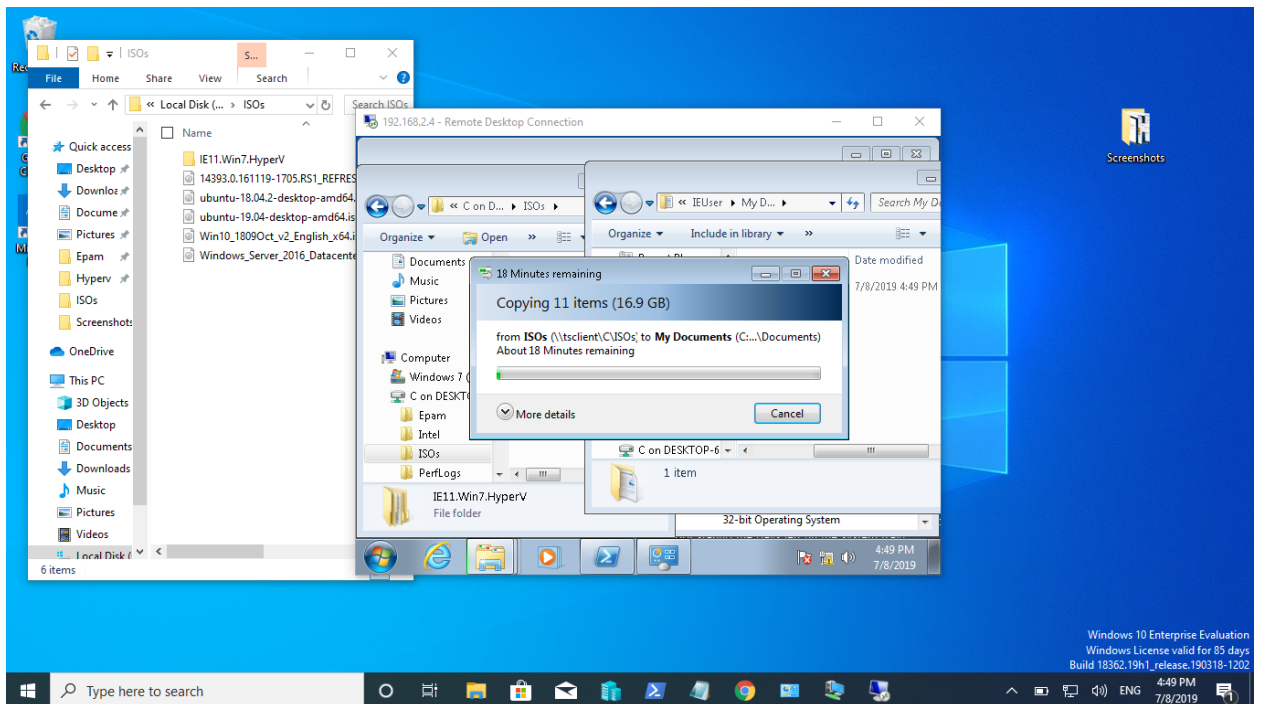


## 9. Configure remote desktop connection to each VM

#Screenshot 6. Remote desktop connection to comp1 (Windows 7)

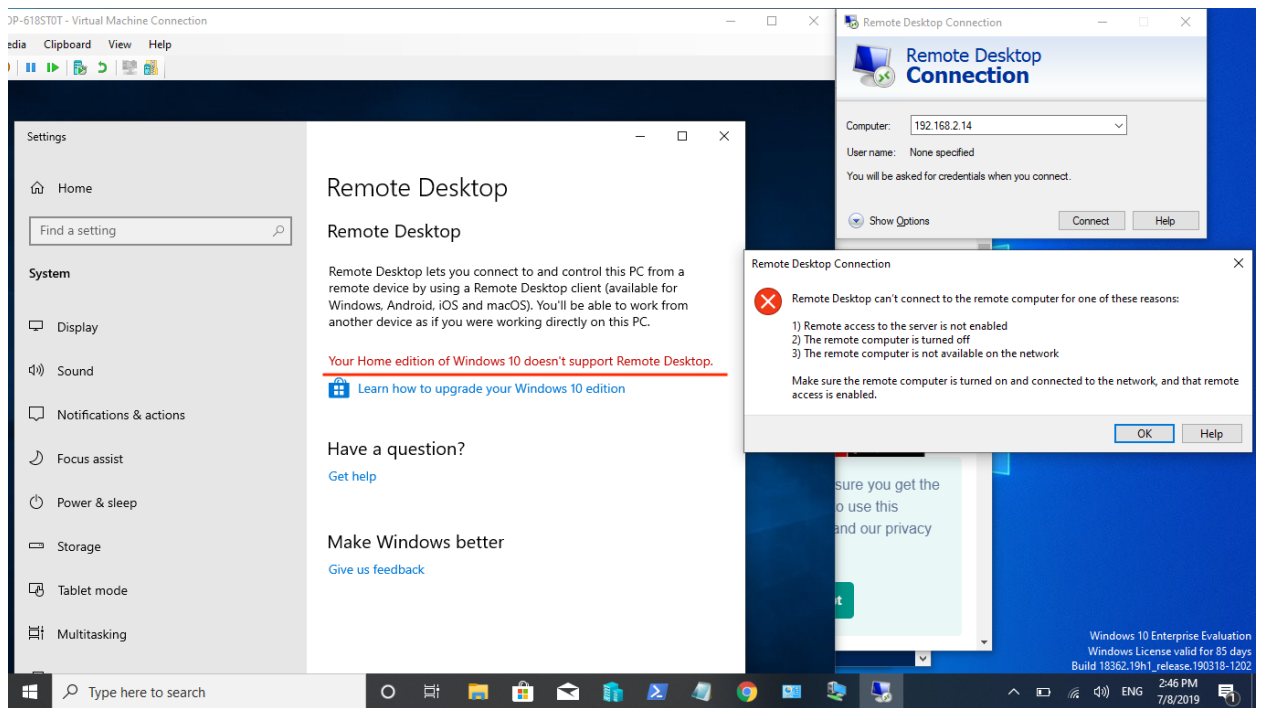


#Screenshot 7. File transfer via RDP between shared host's drive C and guest comp 1 (Windows 7)



#Screenshot 8. RDP is not supported for Windows10 Home(comp4)





10. On the VM "server" install Hyper-V, which install 2 VM with Windows 7 ("comp6" and "comp7")

I have no enough recourses for that

11. Demonstrate file transfer and editing from Host to Guest and back

See screenshot #7 above

12. Create a report with screenshots and attach script files that demonstrate the solution of tasks

See report above