

How to Run Linux on Hyper-V?

Hyper-V is a type 1 hypervisor developed by Microsoft that can be enabled on Windows machines or installed as a standalone operating system (Hyper-V Server 2016, for example). Hyper-V can run not only Windows but also Linux virtual machines. You can run an unlimited number of Linux VMs on your Hyper-V Server because the majority of Linux distributions are free and open source. Installing Linux on a [Hyper-V](#) VM has some features which compare to installing Windows. Today's blog post explores how to install Linux on Hyper-V using an example of Ubuntu Linux 18 LTS (Long Term Support) and consists of the following sections:

- Reasons to Run Linux on Hyper-V

- Before you begin

- Creating a new Hyper-V VM

- Setting the static MAC Address

- Installing Ubuntu Linux on the VM

- Configuring Linux

- Checking the network settings

- Changing the network settings manually

- Changing a hostname in Ubuntu

- Configuring the SSH server on Ubuntu Linux

- Transferring files between [Linux VM](#) and Windows host machines

- Installing Hyper-V Linux Integration Services

- Tuning Linux Disk Optimizer

[Download a Basic Edition](#) of NAKIVO Backup & Replication right now and get to protect VMware, Hyper-V, or AWS EC2 environments at a special price of \$99/socket. In under one minute, the Basic Edition can be installed on Windows or Linux or deployed as a preconfigured VMware VA, Nutanix AHV VA, or AWS AML. What's more, you can seamlessly upgrade to other editions once your business needs change and your infrastructure grows.

Reasons to Run Linux on Hyper-V

The popularity of Linux is growing and this operating system is time-tested. Let's have a look at the most common reasons that might force you to install Linux on Hyper-V:

You need to test a particular Linux distribution in your Hyper-V virtual environment. For example, a new Linux version has recently been released and you want to test this version before installing on physical servers.

You want to run containers. Linux has excellent container support which may be a reason to install Linux on Hyper-V as a virtual machine.

Testing multi-platform software or software that provides similar functionality. This is a great option when developers make their applications able to run on multiple operating systems, such as Windows, Linux, and Solaris (for example, Oracle Database). You can install the appropriate software on Linux to test it. If the experiment is successful, you can install Linux and supported software in your production environment.

Licensing. This point is related to the previous one. You have to pay for Windows while

Linux is free. If you don't have a Datacenter Edition license for your Windows Server, you must pay for each installed on a Hyper-V VM that exceeds the number of [Windows licenses](#) included in your edition of the host Windows system.

Linux requires fewer hardware resources than Windows. Windows is a resourceintensive operating system whereas Linux is light-weight.

Linux is secure. Users should be aware that Windows systems are vulnerable to a variety of harmful viruses while Linux users may feel safe, especially if their system is configured correctly. Even if you do download a virus, you need to go to great lengths to actually execute the virus on Linux. You need to configure file permissions, for example.

Your privacy is respected if you use Linux.

Linux shell such as bash is essential for automating tasks. You can use regular expressions, scripts and other tricks to manage Linux. You manage Linux, Linux doesn't manage you. The community of developers is amazing – developers like Linux. If any vulnerabilities are found, they are always fixed quickly.

You want to learn Linux and migrate from Windows to Linux.

If you want to convert your [physical servers to Hyper-V VMs](#), use converters. Converting a physical Linux machine to a VM in the Hyper-V format can be performed in two steps. Use [VMware vCenter Converter](#) for converting a physical machine to a VMware VM, then use [Microsoft Virtual Machine Converter](#) for converting a VMware VM to a Hyper-V VM.

Before You Begin

Installing Ubuntu Linux is considered a focal point of today's blog post. If you don't need the GUI (graphical user interface), you can install Ubuntu Server. The installation principle is similar. The number of Linux distributions is high and you can select any distribution you like and install it by using the explained logics. One of the main differences between different Linux families is different package managers you're required to use. Package managers are intended to install, configure, upgrade and delete programs on Linux systems.

Package managers help to resolve such tasks as:

Downloading packages. The packages are automatically downloaded from the most suitable software repository when you enter the command to install a particular needed package or application. Packages include metadata such as version of software, architecture the package is built for, checksums, other needed packages to work properly etc.

Resolving dependencies. When you install software, that software may require its own appropriate packages including libraries. Finding, downloading and installing all this manually consumes a lot of time and efforts. Package managers tell you that there are dependencies and suggest you to install automatically the needed depending packages.

Quality control. Software packages that are stored in authorized software repositories are tested to provide high quality, stability and the absence of bugs.

The standard of file locations is respected. Linux contains a high number of files and directories. Sometimes, different Linux distributions use different paths for storing files as well as using different configuration file names. Package managers allow you to follow the standards and store files in the required locations for each Linux distribution.

Before selecting a Linux distribution to install on Hyper-V, you can familiarize yourself with the information about Linux package managers. Some of the most popular Linux distributions and package managers are listed in the table below.

Linux distributions	Package managers used
Debian, Ubuntu, Kubuntu, Xubuntu, Lubuntu, Mint	APT (Advanced Packaging Tool), DPKG (Debian Package Management System)
OpenSUSE, SLES (SUSE Linux Enterprise Server)	Zypper, YaST
Red Hat, CentOS, Fedora	RPM (Red Hat Package Manager), YUM (Yellowdog Updater Modified)
Gentoo	Portage

As mentioned in the very beginning of this blog post, the installation of Linux on Hyper-V will be demonstrated using an example of Ubuntu Linux. Download the ISO image of Ubuntu 18 LTS from the [official site](#).

Creating a New Hyper-V VM

First, make sure that the Hyper-V role (feature) is enabled on your Windows machine (go to **Control Panel > Programs and Features > Turn Windows features on or off** and verify whether the check box near the Hyper-V string is ticked). In order to [create a new virtual machine](#), open Hyper-V Manager and click **Action > New > Virtual Machine**. The New Virtual Machine Wizard should appear.

 Creating a new VM in Hyper-V Manager.

Before You Begin. There is nothing to configure at this step.

Specify Name and Location. Enter your VM name, for example, Ubuntu18 and select a folder where you will store a new VM. Try to use volumes that are different from your system volume C:. In this case a directory on disk D: is used (*D:\Virtual\Hyper-V*). You can enter the path manually or select a location by clicking the **Browse** button.

The screenshot shows the 'Specify Name and Location' step of the 'New Virtual Machine Wizard'. The left sidebar contains a list of steps: 'Before You Begin', 'Specify Name and Location' (highlighted), 'Specify Generation', 'Assign Memory', 'Configure Networking', 'Connect Virtual Hard Disk', 'Installation Options', and 'Summary'. The main area contains the following text: 'Choose a name and location for this virtual machine. The name is displayed in Hyper-V Manager. We recommend that you use a name that helps you easily identify this virtual machine, such as the name of the guest operating system or workload.' Below this, there is a 'Name:' text box with 'Ubuntu18' entered. Another paragraph states: 'You can create a folder or use an existing folder to store the virtual machine. If you don't select a folder, the virtual machine is stored in the default folder configured for this server.' Below this is a checked checkbox labeled 'Store the virtual machine in a different location'. A 'Location:' text box contains 'D:\Virtual\Hyper-V\' and a 'Browse...' button is to its right. A warning icon and text at the bottom of the main area state: 'If you plan to take checkpoints of this virtual machine, select a location that has enough free space. Checkpoints include virtual machine data and may require a large amount of space.' At the bottom of the wizard are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'.

Specify Generation. Choose the generation of this virtual machine – [Generation 1](#) or [Generation 2](#). *Generation 1* is selected in this example due to better compatibility.

 Selecting the generation for a Hyper-V VM.

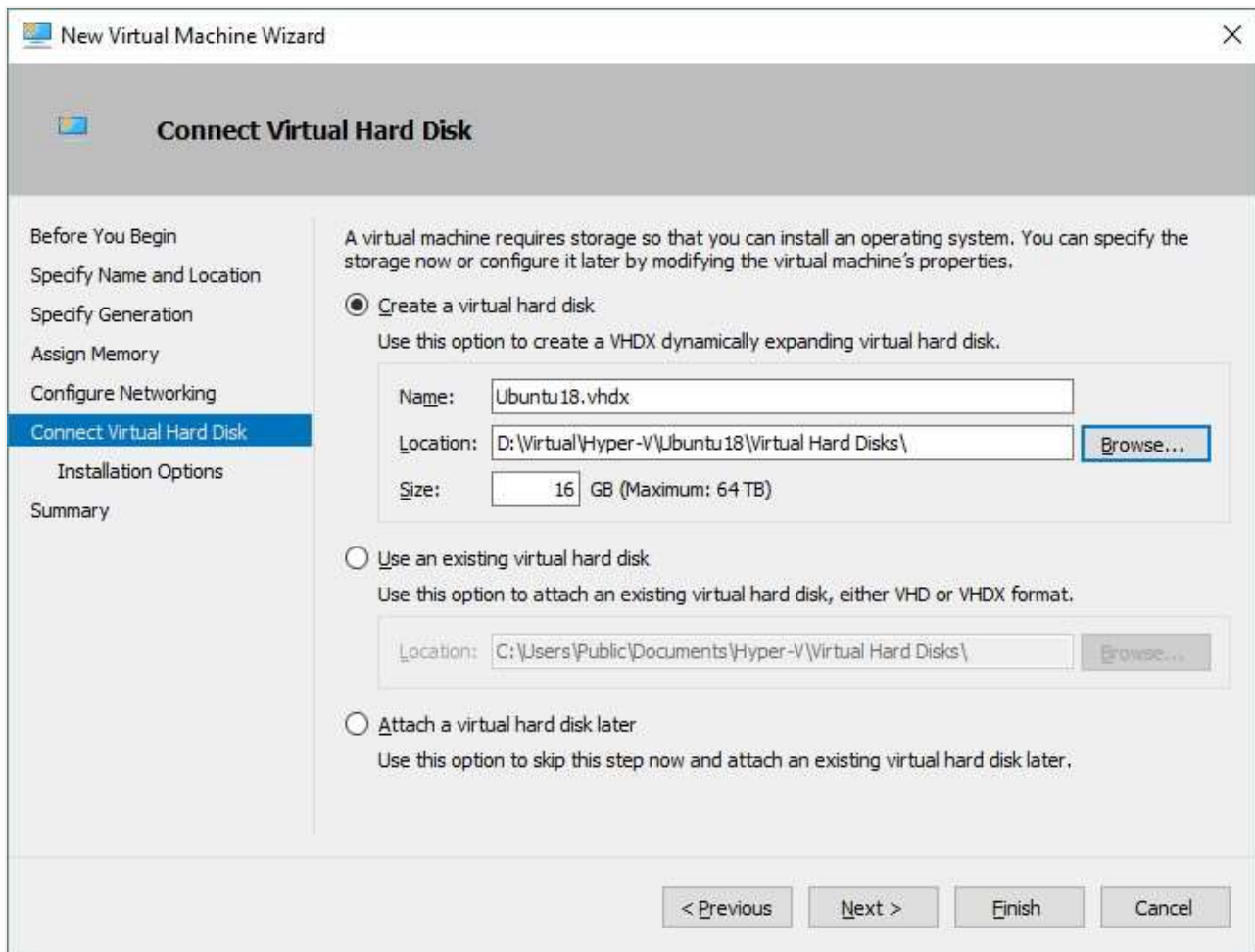
Assign Memory. Specify at least 1 GB of memory for your Ubuntu Linux machine with GUI. You can edit memory settings after VM creation. Tick the *Use Dynamic Memory for this virtual machine* checkbox to use your hardware memory resources more rationally.

 Specifying the amount of memory.

Configure Networking. Select a [virtual switch](#) to which the virtual network adapter of your VM will be connected. The Default Switch is used in this case. Later you can create and configure other virtual switches and connect your VM to the appropriate networks.

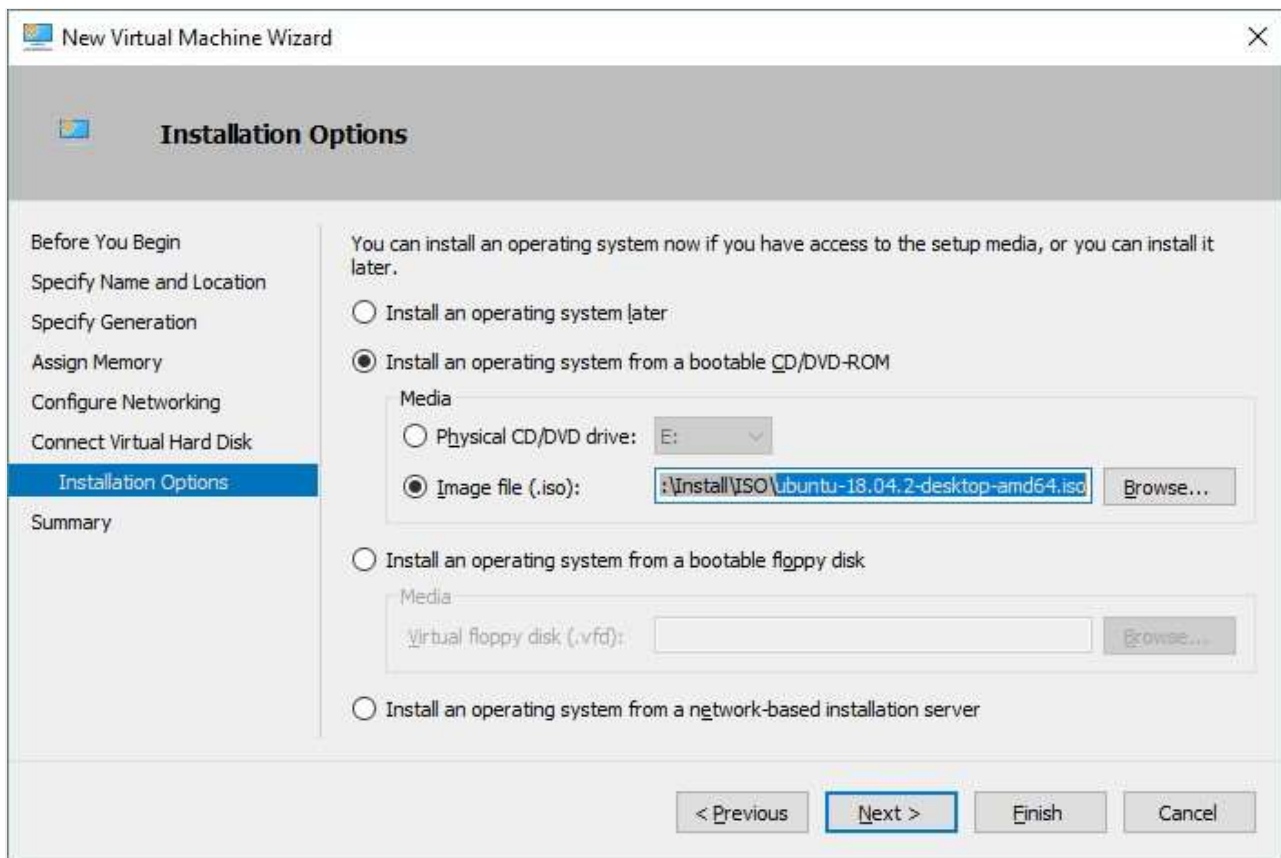
The screenshot shows the 'Configure Networking' step of the 'New Virtual Machine Wizard'. The left sidebar contains a list of steps: 'Before You Begin', 'Specify Name and Location', 'Specify Generation', 'Assign Memory', 'Configure Networking' (highlighted), 'Connect Virtual Hard Disk', 'Installation Options', and 'Summary'. The main area contains the following text: 'Each new virtual machine includes a network adapter. You can configure the network adapter to use a virtual switch, or it can remain disconnected.' Below this, there is a 'Connection:' label followed by a dropdown menu showing 'Default Switch'. At the bottom of the wizard are four buttons: '< Previous', 'Next >' (highlighted), 'Finish', and 'Cancel'.

Connect Virtual Hard Disk. Select the **Create a virtual hard disk** option. A dynamic VHDX virtual disk is created by default. Enter the virtual disk name (for example, *Ubuntu18.vhdx*), select the location and set the size of the virtual disk (at least 16 GB).



The screenshot shows the 'New Virtual Machine Wizard' window, specifically the 'Connect Virtual Hard Disk' step. The left sidebar contains a list of steps: 'Before You Begin', 'Specify Name and Location', 'Specify Generation', 'Assign Memory', 'Configure Networking', 'Connect Virtual Hard Disk' (which is highlighted in blue), 'Installation Options', and 'Summary'. The main area of the wizard has a title bar 'New Virtual Machine Wizard' and a close button. Below the title bar is a section titled 'Connect Virtual Hard Disk'. It contains a paragraph: 'A virtual machine requires storage so that you can install an operating system. You can specify the storage now or configure it later by modifying the virtual machine's properties.' There are three radio button options: 1. 'Create a virtual hard disk' (selected): 'Use this option to create a VHDX dynamically expanding virtual hard disk.' Below this are three input fields: 'Name' (containing 'Ubuntu18.vhdx'), 'Location' (containing 'D:\Virtual\Hyper-V\Ubuntu18\Virtual Hard Disks\'), and 'Size' (containing '16 GB (Maximum: 64 TB)'). There is a 'Browse...' button next to the Location field. 2. 'Use an existing virtual hard disk': 'Use this option to attach an existing virtual hard disk, either VHD or VHDX format.' Below this is a 'Location' input field (containing 'C:\Users\Public\Documents\Hyper-V\Virtual Hard Disks\') and a 'Browse...' button. 3. 'Attach a virtual hard disk later': 'Use this option to skip this step now and attach an existing virtual hard disk later.' At the bottom of the wizard are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'.

Installation options. Select the *Install an operating system from a bootable CD/DVD-ROM*. Select *Image file (.iso)* and browse the ISO installation image of Ubuntu18 that has been downloaded before.



Summary. Check your settings in the installation summary and click **Finish** to create a new virtual machine.

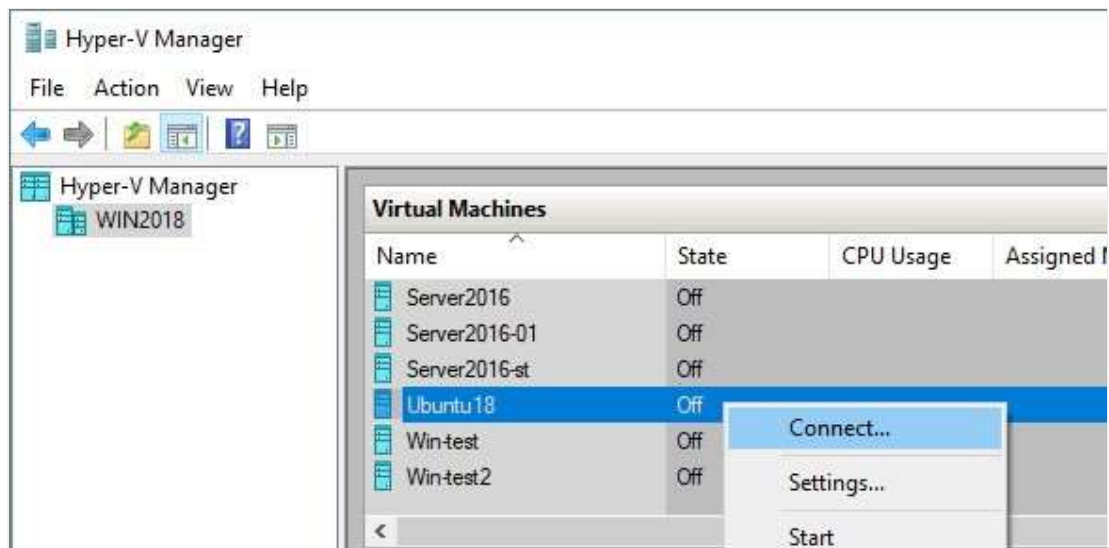
Set the Static MAC Address

Now your new empty VM has been created and you can edit the different VM settings, such as MAC address. It is better to use a static MAC address for Linux running on Hyper-V especially if your Linux VM needs to be migrated between Hyper-V hosts. Right click your VM name and click **Settings**. In the Settings window, select your network adapter, and expand the network adapter options by clicking the + icon. Click **Advanced Features** and in the right section of the window select a static MAC address. Click **OK** to save changes.

 Defining a static MAC address for the virtual network adapter of the Ubuntu VM in HyperV Manager.

Installing Ubuntu Linux on the VM

In Hyper-V Manager, select your Ubuntu virtual machine, right click the VM name and in the context menu, select **Connect** to open VMConnect. As an alternative you can click **Action > Connect** in the menu bar of the Hyper-V Manager Window after selecting the VM.



In the VMConnect window, click the Start button to power on the VM. The virtual machine should be loaded from the ISO image. Then you can see the graphical user interface of the Ubuntu installer. The installer provides multiple configuration steps in the wizard mode.

Welcome. On this screen, select your desired language (English) and click **Install Ubuntu**.

Keyboard layout. Select your keyboard layout (for example, English US) and click **Continue**.

Updates and other software. You can choose between *Normal installation* and *Minimal installation*. Normal installation includes a wider set of installed applications.

Other options allow you to download updates while installing Ubuntu as well as installing third-party software for graphics, Wi-Fi hardware and additional media formats.

 Selecting which components to install for Ubuntu.

Installation type. Now you should prepare a disk for installing Ubuntu. You should create a new partition scheme and format the partitions.

Erase disk and install Ubuntu. This option can be used for partitioning disks automatically. If you are installing Ubuntu Linux for the first time, you can select this option.

Encrypt the new Ubuntu installation for security. This option will not be enabled in the current example.

Use LVM with the new Ubuntu installation. LVM is the Logical Volume Manager that allows you to use an abstraction layer and create a volume group by using two logical or physical disks (or two virtual disks in the case of running a Linux on Hyper-V as a VM). Moreover, if you have no free space available on your volume, you can add one more disk to the volume group and extend your volume group size. This approach adds much more flexibility that can be useful for file servers, database servers etc. LVM allows you to perform resizing of volume groups and the majority of other operations on the fly without powering off your Linux machine.

Something else. By selecting this option, you can manually create partitions of the necessary size and mount them to the appropriate directories. This option is widely utilized by skilled users.

In Linux, disk partitions are not named as disk letters such as *C:* or *D:* as they would be in Windows. In Linux, disks are named as */dev/sda*, */dev/sdb*, */dev/sdc* etc. If the disk is named as */dev/sda* and has three partitions, the names of partitions should be */dev/sda1*, */dev/sda2* and */dev/sda3*.

sda means **SATA Disk**. A long time ago when IDE (parallel ATA) disks were used, **hard disk** devices were named */dev/hda*, */dev/hdb* etc. and partitions were named */dev/hda1*, */dev/hda2*, */dev/hdb1* accordingly.

Disk partitions are mounted to directories that are called mount points in this case. The main directory of the Linux file system is the root directory that is named by the */* symbol. The recommendations for creating the partitioning scheme manually for Linux are displayed in the table. You can use a similar partitions scheme when you install Linux on Hyper-V.

Mount point	Disk partition	Recommended size	Comment
<i>/boot</i>	<i>/dev/sda1</i>	500 MB - 1 GB	Contains the boot loader such as GRUB
<i>/</i>	<i>/dev/sda2</i>	at least 12-20 GB	The most important partition that is required
<i>/home</i>	<i>/dev/sda3</i>		Is used to separate user files and system files
swap	<i>/dev/sda4</i>	4-8 GB, depending on the amount of RAM	The swap partition is used for the same purpose as the Windows swap file
<i>/var/log/</i>	<i>/dev/sdb1</i>		Creating a separate partition for storing logs is not necessary, but is recommended for servers. If you have applications that intensively write log files, your disk may become full. The operating system and applications can crash as a result.

Note that you can create only 4 partitions if the MBR (Master Boot Record) partitioning scheme is used. GPT (GUID Partition Table) allows you to create more than 4 partitions.

Click **Install Now** to continue.

 Selecting the installation type for Ubuntu.

Write the changes to disk? A warning message is displayed. Click **Continue** to apply a new disk partitioning scheme.

Where are you? Select your geographical location and time zone. Software repositories located in your region should be used for better availability and downloading speed.



Who are you? Enter your name, your computer's name (a hostname for your VM in this case), pick a username and define the password. You can select the appropriate option to login automatically if needed. A password is required to log in by default – this is the most secure option, and the one that is recommended. Click **Continue** to start the installation process and copying files.

 Configuring computer's name, username and password.

Wait until the installation process is complete. You can read useful tips and OS explanations while files are being copied.

Installation Complete. You need to restart the computer in order to use the new installation. Click **Restart Now**.

After restarting the Hyper-V Linux virtual machine, you can see the login screen. Select your user name and enter your password to log in. See the window that demonstrates new features of the new version of Ubuntu Linux you have already installed on Hyper-V.

Configuring Linux

You have installed Ubuntu on Hyper-V VM, but Ubuntu, as with any other Linux, must be configured for further comfortable usage. The majority of settings can be configured in the Linux console (Terminal). In order to open a Terminal, click the nine dots in the left bottom corner (Show Applications) and in the menu that appears, click the Terminal icon. Another way to open Terminal is to right click desktop and click **Open Terminal** in the context menu or pressing the **Ctrl+Alt+T** hotkey combination.

Now you see an exquisite terminal window of Ubuntu Linux. Be aware that commands and file names in Linux are case sensitive. *ReadMe.txt* and *readme.txt* are different file names in Linux while in Windows there is no difference between these file names.

 The graphical user interface of Ubuntu Linux 18.

Checking the network settings

Install the net-tools to become able to check network configuration by using the command:

sudo apt-get install net-tools

Where **sudo** allows you to execute the command as a root user (*substitute* user do).

Type **ifconfig** to check the network configuration of your Ubuntu running on a Hyper-V VM.

 The Terminal window of Ubuntu 18.

A default virtual DHCP server connected to the default virtual switch has distributed the IP configuration for the virtual network adapter of the current Ubuntu Linux VM running on Hyper-V. The name of the virtual Ethernet adapter in Linux is *eth0*. The obtained IP address is 172.25.151.20, netmask is 255.255.255.240 and the default gateway is 172.25.151.31 in this case.

Changing the network settings manually

Obtaining your IP address via DHCP is convenient, but if you are planning to use your Linux running on Hyper-V as a server, you should configure network settings manually.

Moreover, if you wish to clone your Linux VM used as a server, changing the IP address and machine name is essential. You may also have to change IP addresses on your Linux VM after reconfiguring your Hyper-V virtual switch or after connecting a virtual network adapter to another virtual switch. Let's find out how to configure network settings for your Linux VM manually.

Changing system settings such as network settings requires root privileges. Type **sudo -i** to have root rights in the console (you should enter your password) during the session (press **Ctrl+D** to exit the root mode when needed). You may notice that the \$ character is changed to the # character in the command line prompt. Enter a new IP address, then reload the network interface: **ifconfig eth0 172.25.151.21 netmask 255.255.255.240 ifconfig eth0 down ifconfig eth0 up**

The settings are applied until you restart the machine.

For setting up the static IP address permanently, use the **netplan** configuration that is used to configure networks in Ubuntu Linux since the version 17 of Ubuntu. In earlier versions of Ubuntu and in some other Linux distributions you can edit your IP address configuration by editing the */etc/network/interfaces* file. Let's edit the *.yaml* file with **vim** by using the command:

vim /etc/netplan/01-network-manager-all.yaml

The file contains the following configuration in this case:

```
network:
  version: 2
  renderer:
    NetworkManager
```

```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: NetworkManager
~
~
```

You can change the IP address, net mask, gateway, and DNS servers by editing this file. You should preserve the file structure including the number of space characters.

Note: A network mask can be written as the number of bits.

255.255.255.0 is equal to 11111111.11111111.11111111.00000000 and is a 24bit mask (24 ones in the decimal format, as you can see).

255.255.255.240 is equal to 11111111.11111111.11111111.11110000 and is a 28bit mask (28 ones in the decimal format are present).

An IP address and net mask consist of 4 octets. Each octet (8bits) can be transformed to the decimal format (see the table below).

1	1	1	1	1	1	1	1	1+1+1+1+1+1+1+1=8
128	64	32	16	8	4	2	1	128+64+32+16+8+4+2+1=255

You can use the table below for converting network masks from one format to another for networks that can be classified as class C networks. The maximum number of hosts that can be connected to the network is shown in the right column.

Bits	Net mask	Hosts
/30	255.255.255.252	2
/29	255.255.255.248	6
/28	255.255.255.240	14
/27	255.255.255.224	30
/26	255.255.255.192	62
/25	255.255.255.128	126
/24	255.255.255.0	254

By using the above explained logics, you can similarly transform any mask from one format to another format manually. If you want to calculate the maximum number of hosts that can be connected to a network, use the formula:

$$2^{(32-m)} - 2$$

Where **m** is the number of bits used for a net mask. For example, you have a /29 network and the maximum number of hosts for this network is:

$$2^{(32-29)} - 2 = 2^3 - 2 = 8 - 2 = 6$$

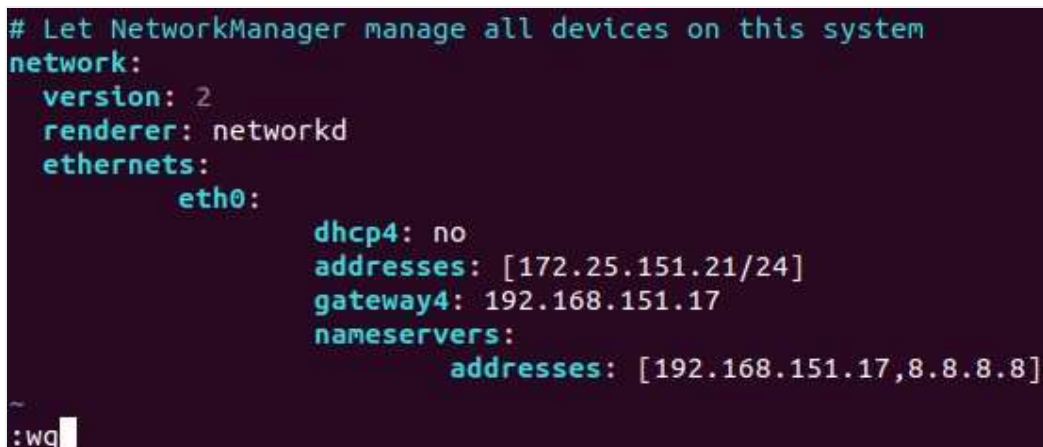
–2 shows that you must exclude the network address and the broadcast address. (For example, for 192.168.1.0/24 network you must exclude 192.168.1.0 and 192.168.1.255 addresses).

Don't forget that at least one IP address must be used as a default gateway to enable a connection to other networks. You can also divide your network into subnets, but for now, let's turn back to configuring network for Ubuntu running on a Hyper-V VM.

Change the renderer from **NetworkManager** to **networkd** and edit your Ubuntu network configuration similarly as shown below. Be attentive when adding space characters to preserve the configuration format.

```
network: version: 2
renderer: networkd
ethernets: eth0:
  dhcp4: no
  addresses: [172.25.151.21/24]
  gateway4: 192.168.151.17
  nameservers:
    addresses: [192.168.151.17,8.8.8.8]
```

Press **Esc** to exit from the editing mode to the navigation mode in vim. Type **:wq** and press **Enter** to write changes and quit the text editor.



```
# Let NetworkManager manage all devices on this system
network:
  version: 2
  renderer: networkd
  ethernets:
    eth0:
      dhcp4: no
      addresses: [172.25.151.21/24]
      gateway4: 192.168.151.17
      nameservers:
        addresses: [192.168.151.17,8.8.8.8]
:wq
```

Note. If you have more than one network adapter (for example, *eth0* and *eth1*) add the similar *eth1*: block to your network configuration *.yaml* file under the *eth0*: block. Check whether your configuration file format is correct after editing with the command:

netplan try

If everything is correct, press **Enter** to keep these settings.

 Checking network configuration in Ubuntu with netplan.

How to change a hostname in Ubuntu?

Let's check the current hostname before changing it.

hostnamectl

As you can see, on the screenshot below, the current hostname of the Linux machine is *Ubuntu-VM*. Set the new *VM-Ubuntu* hostname: **hostnamectl set-hostname VM-Ubuntu**

 Checking a hostname in Ubuntu.

Make sure that the hostname is now changed in the **/etc/hostname** file.

less /etc/hostname

You should also edit the hostname in the */etc/hosts* file.

vim /etc/hosts

vim is a powerful console text editor that has no analogs in Windows PowerShell. By default, you are in the navigation mode. You can navigate by pressing **h** (left), **L**(right), **j** (down), **k** (up) or arrow keys. Press **a** to enter the editing mode to *append* any characters after a cursor. Press **i** if you want to *insert* a character instead of an existing character highlighted by a cursor. Change the *Ubuntu-VM* to *VM-Ubuntu*, press **Esc** to exit from an editing mode to a navigation mode.

Type **:wq** to write changes to the file and quit.

Use the **init 6** command to restart your Linux machine in order to ensure that changes are applied.

Configuring the SSH server on Ubuntu Linux

For greater convenience, you should enable remote access via SSH. The SSH Server is used to access the Linux console remotely via encrypted SSH (secure shell) protocol. For doing this, you should install the SSH server on your Ubuntu VM running on Hyper-V and the SSH client on your Windows machine used as a Hyper-V host.

Open Terminal on your Ubuntu machine and run the command as the root user:

apt-get install -y openssh-server

The configuration of the SSH server is stored in the *sshd_config* file. Edit this file to enable the SSH connection with needed settings.

vim /etc/ssh/sshd_config

Uncomment this string (delete the **#** character)

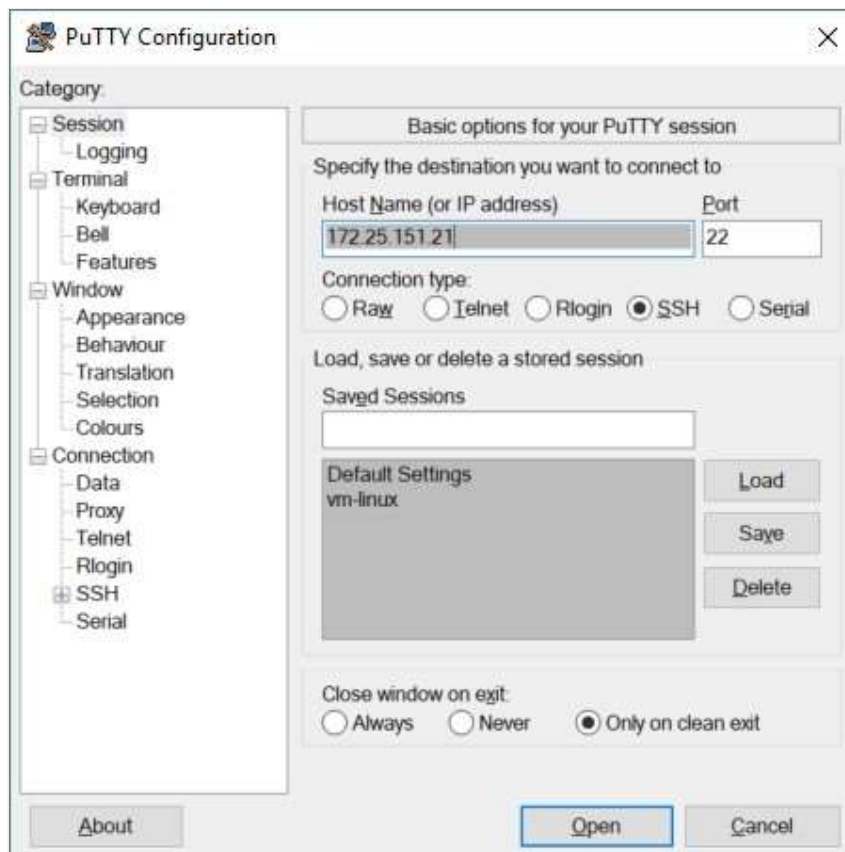
PasswordAuthentication yes

Save the file and exit.

Restart the SSH server daemon (service).

/etc/init.d ssh stop /etc/init.d ssh start

Download the SSH client (for example, [PuTTY](#)) for your Windows machine. Open PuTTY on your Windows machine, enter the IP address of your Ubuntu VM running on Hyper-V and click **Open**.



Enter your username and password. Now you get the Ubuntu console remotely.

Transferring files between Linux VM and Windows host machines

Unlike VMware, you cannot drag and drop files between the Windows host and Linux guest in Hyper-V. As you have configured the SSH server, you can transfer files from your Ubuntu VM to your Hyper-V host and do the inverse by using an SFTP (Secure File Transfer Protocol) client such as [WinSCP](#). WinSCP uses SSH for secure connection and can also use SCP (Secure Copy Protocol) for file transferring.

Run WinSCP on your Windows machine. Select SFTP as a file protocol, enter your host name or IP address, user name and password.

 WinSCP can be used for transferring files from a Linux VM to a Hyper-V host.

When you have logged in, you can see a window with two panels – the first panel displays files on the Windows machine and the second panel displays files on the Linux machine. Go to the designated directories on source and destination machines, select files and directories. Drag and drop the selected objects between panels to copy them.


 The main window of WinSCP when a user is logged in to a remote Linux machine.

How to install Hyper-V Integration Services on Linux?

Hyper-V Linux Integration Services have been built in Ubuntu since version 12.04. If your Linux distribution doesn't include Hyper-V Integration Services, you can download them from the [Microsoft's site](#).

You can check which Integration Services are enabled for your Ubuntu VM running on Hyper-V in PowerShell. Type the following command in PowerShell on the Hyper-V host:
get-vmintegrationservice -VMName "Ubuntu18"

You can also select the services you want Hyper-V to offer to this VM in GUI. Open Hyper-V Manager, right click your VM name, and click **Settings**. In the Settings window, select *Integration Services* in the left pane and tick the check boxes near the services that are needed (*Guest services* must be enabled in this example).

 Selecting which Integration Services must be enabled on a Hyper-V VM running Ubuntu. Click **OK** to save changes.

You can now re-check the enabled services in PowerShell (see the screenshot below). **get-vmintegrationservice -VMName "Ubuntu18"**

 Checking which Hyper-V Linux Integration Services are enabled for the Ubuntu 18 VM.

Once Hyper-V Integration services are installed on your Ubuntu Linux, they must become activated. The `/etc/initramfs-tools/modules` file contains a list of modules that you want to be included in your initramfs.

The *initramfs* is the initial RAM file system intended to mount the root filesystem (*/*). As a result, the initial root file system and init daemon (that is the first process that must run in Linux) can reside in kernel memory cache (not on a ramdisk as for *initrd* filesystem that acts as a block device of a fixed size and consumes more memory for caching). *initrd* was replaced by *initramfs* after Linux kernel 2.6 due to its advantages such as flexibility, efficiency and simplicity. The *initramfs* is a complete set of directories that you can see on your root filesystem and is gzipped into a cpio archive.

Linux kernel can be monolithic and modular. In most cases the module kernel is used. Let's enable loading of the necessary modules.

vim /etc/initramfs-tools/modules

Edit the file and add these strings to enable loading of the appropriate modules.

```
hv_vmbus
hv_storvsc
hv_blkvsc
hv_netvsc
```

Update initramfs: **update-**

initramfs -u

Restart your Hyper-V VM running Ubuntu:

init 6

Verify that Hyper-V related modules for your Linux kernel are loaded.

lsmod | grep hv

```
root@VM-Ubuntu:~# lsmod | grep hv
hv_balloon          24576  0
hv_netvsc           73728  0
hv_utils            28672  0
hv_storvsc          20480  1
scsi_transport_fc   57344  1 hv_storvsc
hv_vmbus            90112  7 hv_balloon,hv_utils,hv_netvsc,hid_hyperv,hv_stor
vsc,hyperv_keyboard,hyperv_fb
root@VM-Ubuntu:~#
```

Tuning Linux Disk Optimizer

Linux has an I/O scheduler mode that helps to optimize disk I/O (input/output) by default that is especially useful for magnetic (spinning) hard disk drives. Hyper-V has its own mechanism for optimizing virtual disks I/O performance. Using two disk optimizers is not necessary and you can disable the built-in Linux disk optimizer. Linux kernel allows you to use different I/O schedulers, each of them using different algorithms. In this case, the NOOP scheduler must be used. NOOP is the simplest I/O scheduler that uses the First In First Out (FIFO) concept for the queue. This is what you need to run Ubuntu on Hyper-V.

In order to edit the I/O scheduler used by Linux kernel, edit the grub configuration file with the command:

vim /etc/default/grub

Find the GRUB_CMDLINE_LINUX_DEFAULT string. In this case this string has the following view:

GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"

Edit this string to get the following view:

GRUB_CMDLINE_LINUX_DEFAULT="elevator=noop"

Save changes to this GRUB configuration file.

Update the GRUB bootloader configuration.

update-grub2

Reboot your Ubuntu Hyper-V VM.

init 6

Conclusion

Today's blog post has explored how to install Ubuntu on Hyper-V. If you have never used Linux, installing Ubuntu on Hyper-V can be the first step to opening the gate to the world of Linux. Creating a new virtual machine is straightforward, but some manual configurations should be made after installation is finished. Running Linux VMs on Hyper-V hosts allows you to clone and migrate VMs easily. Two machines must not have the same host name and IP address, and for this reason, basic network configuration was explained. Hyper-V Linux Integration Services can improve the performance of VMs. While the operating system is running on a VM, both Windows and Windows virtual machines should be backed up to protect precious data. [NAKIVO Backup & Replication](#) can [back up your Hyper-V VMs](#) in the best way possible.