# AZ-104T00A – Administer Azure Virtual Machines

Good day everyone. In the last session we discussed Storage in Azure with different services available for storing our data. Today we are going to discuss Virtual Machines.

## Configure Virtual Machines

Azure Virtual Machines enables you to create on-demand, scalable computing resources. Azure Architects commonly use virtual machines to gain greater control over the computing environment. An Azure virtual machine is an on-demand, scalable computer resource that is available in [Azure](https://cloudacademy.com/library/azure/). Virtual machines are generally used to host applications when the customer requires more control over the computing environment than what is offered by other [compute resources](https://cloudacademy.com/course/getting-started-with-azure-virtual-machines-988/related-resouces/). When you leverage a virtual machine to host your application, you get the flexibility of virtualization without the need to buy or maintain any underlying physical hardware. That said, you will have to manage the typical tasks that are associated with any other server, including configuration, patch management, and software installation. Because they are quick and easy to set up, Azure virtual machines are often used to deploy development and test environments. Organizations will also use Azure virtual machines to host their applications in Microsoft Azure, due to their pay-as-you-go nature, which allows you to only pay for VM’s when you need them. Azure virtual machines are also often used to extend on-prem data centers to Microsoft Azure, because VM’s that are attached to a virtual network can communicate with on-prem environments over a site-to-site VPN. When designing an application infrastructure that includes Azure VMs, there are important considerations to think about. You should think about virtual machine naming conventions as well as where your VM’s will be deployed. Generally speaking, virtual machines in Azure should be deployed in locations that are closest to those who will be accessing those VM’s. Other important considerations include VM sizing requirements and the number of VM’s that will be needed, especially since Microsoft Azure imposes CPU and virtual machine quotas that you may or may not need to have lifted. You should also think about what operating system your VM’s will need to run and what the configuration of your VM’s should look like once they start.

Review Cloud Service Responsibilities

The responsibilities associated with configuring and maintaining virtual machines is shared between Microsoft and the customer. The following chart shows how the responsibilities are handled across the IaaS (virtual machines), PaaS, SaaS, and on-premises offerings. To better explain the concept of shared responsibility, a common example given in courses that talk about the cloud is pizza. How many of you love pizza? I imagine quite a few. So, pizza. I could make it at home and if I make my pizza at home, I'm responsible for getting the pizza dough and the sauce and the cheese and the toppings. I have to have power, fire and oven. Any electricity or gas? I want some drinks. I need a table somewhere to eat it at. I need cutlery, I need plates. I need everything. I have complete control of all of these elements. The exact contents of the dough, the exact type of cheese, the exact ingredients in my sauce, the exact type of oven. But I have all the responsibility. I have to think about all of those things. Now if I start shifting to a take and bake, I go somewhere and they have premade the pizza. All I have to do is take the pizza so I'm not focused anymore about the pizza dough or the cheese, it's provided for me. All I focus on is taking that pizza, but I have to provide my own power, my own fire, my own oven, my own table and cutlery and drinks. Then, I can get pizza delivery. So now I'm shifting the responsibility even more. I don't have to think about cooking it anymore. The power, the electricity, the oven. But I still need drinks. I still need a place to eat it. Probably not cutlery. I use my hands. So, I'm responsible the less and less. But it also means I have less control about these things. Maybe I don't care. That's probably the case when a pizza gets delivered to me. Do I care about the oven they used? Do I care about where they got their electricity from? No. So does it taste good? That's all I care about. What could be complete dining out. I'm not responsible for anything other than consuming and eating my pizza. So, we can apply this same idea to computers, thankfully. If I think about a stack, there's networking, there's storage, there's the physical compute servers. There's typically a hypervisor, Hyper-V, VMware. As an operating system, Windows, Linux, middleware systems, those abstractions. Maybe it's MQ or some kind of bus technology, a runtime to run my application, like .net or Java Enterprise Edition J2E. And then I get to my application and my data. This is what I care about. This is what provides the business value. But on premises my organization is responsible for everything. But I also have all the control. I can pick every aspect of this. As we start moving to cloud services, we start off with infrastructure as a service and at basic level it can be thought as virtual machines. So here the vendor, Azure in this case, is responsible for the hypervisor, for the physical fabric, the compute, the storage, the networking. The consumer of infrastructure for the service gets a virtual machine. They're responsible for everything inside that virtual machine. The operating system, installing middleware, installing runtimes, maintaining them, which means patching, firewall, policy, backup, disaster recovery, the list goes on and on. So, I have ordered the flexibility still within that VM, but I have all the responsibility. I have to think about all those different elements. Now, when we talk about Azure and we talk about a VM, there are extensions, there are services that help me do all of things I just said, backup DR, Policy, firewall, anti-malware, you name it. There are things that help me, but I'm still responsible for thinking about that tooling. That we have platform as a service. So now the vendor is even taking care of the operating system, the middleware, the runtime. They're taking care of the patching, of the backup. They just provide a place for me to deploy my custom app with my data. My focus on my line of business application is just my app. I don't care about all of those other things and we'll talk about PaaS. Next level is Software as a service. Now, this isn't Azure. This would be more something like Microsoft 365, Dynamics 365. It provides the business application, the end service I need. Whereas PaaS is not really providing the business application, it's providing that foundation I can build my business application on, but it's not doing the business logic. Whereas a SaaS is actually doing that business logic. It's providing that capability for me. So, I get different levels of responsibility. I have to do less and less as I move up to SaaS model, but I might get a bit less configuration, a bit less flexibility. I hope this helped to understand this model.

Plan Virtual Machines

Before you create an Azure virtual machine, it's helpful to make a plan for the machine configuration. Certain resources are required in order to deploy an [Azure virtual machine](https://cloudacademy.com/course/getting-started-with-azure-virtual-machines-988/course-introduction/). Generally speaking, if these resources do not exist prior to VM creation, they will be created when the VM is created. For example, before creating a virtual machine, a resource group must exist. This is because a new virtual machine must be contained within a resource group. A virtual network is also required before deploying a virtual machine, because a virtual machine must be connected to a virtual network. This also means that a virtual NIC is also necessary. If you are deploying a virtual machine that uses unmanaged disks, a storage account will be required to hold the virtual hard disks for the virtual machine. If the virtual machine you are deploying uses only managed disks, a storage account is not required. A VM that will be remotely accessed will require a public IP address. If it will only be accessed internally, a public IP address is not required. Although they are not required, data disks can be attached to a virtual machine to expand its storage capabilities. It’s usually a good idea to attach at least one data disc to a VM if you plan to host an application on that VM. Usually, we start with the creation of the resource group if not already present, then we proceed with the network part. As we discussed in the network session, Virtual networks are used in Azure to provide private connectivity between Azure Virtual Machines and other Azure services. Virtual machines and services that are part of the same virtual network can access one another. By default, services outside the virtual network can't connect to services within the virtual network. You can, however, configure the network to allow access to the external service, including your on-premises servers. Then we have to choose the name of the VM reflecting our naming convention. We can specify a name with up to 15 characters on a Windows virtual machine and 64 characters on a Linux virtual machine. Then we have to consider where to put this VM. So, the location. Each virtual machine is in a region where you want the resources like CPU and storage to be allocated. The regional location lets you place your virtual machines as close as possible to your users. The location of the machine can improve performance and ensure you meet any legal, compliance, or tax requirements. After that, we have to choose the Virtual machine size. Azure offers different memory and storage options for different [virtual machine sizes](https://learn.microsoft.com/en-us/azure/virtual-machines/sizes). The best way to determine the appropriate machine size is to consider the type of workload your machine needs to run. Based on the workload, you can choose from a subset of available virtual machine sizes. Another important element to be considered is the Storage. [Azure Managed Disks](https://learn.microsoft.com/en-us/azure/virtual-machines/managed-disks-overview) handle Azure storage account creation and management in the background for you. You specify the disk size and the performance tier (Standard or Premium). Azure creates and manages the disk. As you add disks or scale the virtual machine up and down, you don't have to worry about the storage being used. We have to consider also costs in this scenario. A subscription is billed two separate costs for every virtual machine: compute and storage. By separating these costs, you can scale them independently and only pay for what you need. For the compute expenses, we have 2 options. **Consumption-based**, in which we pay for compute capacity by the second. We are able to increase or decrease compute capacity on demand and start or stop at any time. Use consumption-based pricing if you run applications with short-term or unpredictable workloads that can't be interrupted. The other option is **Reserved Virtual Machine Instances. It’s** an advance purchase of a virtual machine for one or three years in a specified region. The commitment is made up front, and in return, you get up to 72% price savings compared to pay-as-you-go pricing. RIs are flexible and can easily be exchanged or returned for an early termination fee. Use this option if the virtual machine has to run continuously, or you need budget predictability, and you can commit to using the virtual machine for at least a year. Then we have to consider the Operating System. Azure provides various operating system images that you can install into the virtual machine, including several versions of Windows and flavours of Linux. Azure bundles the cost of the operating system license into the price. You have the option to use your own license, getting price savings. If you're looking for more than just base operating system images, you can search [Azure Marketplace](https://azuremarketplace.microsoft.com/marketplace/apps/category/compute). There are various install images that include not only the operating system but popular software tools, such as WordPress. The image stack consists of a Linux server, Apache web server, a MySQL database, and PHP. Instead of setting up and configuring each component, you can install an Azure Marketplace image and get the entire stack all at once.

Determine Virtual Machines Sizing

Rather than specify processing power, memory, and storage capacity independently, Azure provides different virtual machine sizes that offer variations of these elements in different size configurations. Azure provides a wide range of virtual machine size options that allow you to select the appropriate mix of compute, memory, and storage for your needs. **General-purpose** virtual machines offer a balanced CPU to memory ratio. These machines are perfect for development and testing environments, small databases, and low traffic Web servers. **Compute-optimized** virtual machines offer high CPU to memory ratios. These virtual machines are ideal for medium traffic Web servers and network appliances. They are also good for batch processing and can function as application servers. **Memory-optimized** virtual machines offer high memory to core ratios. Organizations typically use memory-optimized virtual machines for relational database servers, large caches, and for performing in-memory analytics. **Storage optimized** virtual machines offer high disk throughput and IO. They are perfect for big data, SQL, and NoSQL databases. **GPU** virtual machines are specialized for heavy graphic rendering and video editing. These VMs come with single GPUs and multiple GPUs. **High performance compute** virtual machines are the fastest and most powerful CPU virtual machines available. These virtual machines are designed to handle high performance compute workloads, including things like molecular modeling, genomic research, and financial risk modeling. Some high-performance compute virtual machines come with optional high throughput network interfaces. The type of virtual machine that you deploy will be largely dependent on the workload you plan to host on the VM. Azure allows you to change the virtual machine size when the existing size no longer meets your needs. You can resize a virtual machine if your current hardware configuration is allowed in the new size. This option provides a fully agile and elastic approach to virtual machine management.

Determine Virtual Machines Storage

All Azure virtual machines have at least two disks: an operating system disk and a temporary disk. Virtual machines can also have one or more data disks. All disks are stored as virtual hard disks (VHDs). A VHD is like a physical disk in an on-premises server but, virtualized. Every virtual machine has one attached operating system disk. The OS disk has a pre-installed operating system, which is selected when the virtual machine is created. The OS disk is registered as a SATA drive (Serial Advanced Technology Attachment) and labeled as the C: drive by default (in Windows). Every VM has also a Temporary Disk attached. Data on a temporary disk might be lost during a maintenance event or when you redeploy a virtual machine. During a standard reboot of the virtual machine, the data on the temporary drive should persist. However, there are cases where the data might not persist, such as moving to a new host. Therefore, any data on the temporary drive shouldn't be data that's critical to the system. On Windows virtual machines, the temporary disk is labeled as the D: drive by default. This drive is used for storing the **pagefile.sys** file. On Linux virtual machines, the temporary disk is typically /dev/sdb. This disk is formatted and mounted to /mnt by the Azure Linux Agent. Then we can attach also data disks. A data disk is a managed disk that's attached to a virtual machine to store application data, or other data you need to keep. Data disks are registered as SCSI drives and are labeled with a letter you choose. The size of a virtual machine determines how many data disks you can attach and the type of storage you can use to host the data disks.

Demonstration Creating a VM in the Portal

Create a new Windows VM and install IIS Feature.

Connect to Virtual Machine

You generally connect to remote machines with either RDP or SSH. To do so, you either need to assign a public IP address (with the RDP/SSH port exposed) to the VM to which you are trying to connect, or you need to provision an additional jump server, assign a public IP address to that jump server, and then connect to the other virtual machines using private IP addresses internally. You can also try implementing Network Security Groups (NSGs) to restrict the source IP addresses and ports allowed for your network traffic. Still, you are exposing RDP/SSH ports to the source servers over the Internet, which could be a potential security threat. To overcome this issue, Microsoft has created a managed PaaS service called Azure Bastion to provide secure connections to Azure Virtual Machines using the SSL channel through a browser directly without using any external client. This service helps you to limit threats like port scanning and other malware. The Azure Bastion service is provisioned within a VNet within a separate subnet called **AzureBastionSubnet.** If you have multiple VNets in your environment, you will need to deploy Azure Bastion for each VNet separately.

Connect to Windows Virtual Machine

To connect to a Windows-based virtual machine hosted on Azure, use the Microsoft Remote Desktop application with the remote desktop protocol (RDP). The Remote Desktop app provides a graphical user interface (GUI) session to an Azure virtual machine that runs any supported version of Windows. The following image shows how to use the RDP protocol to connect to a Windows-based virtual machine in the Azure portal. For Windows-based virtual machines, usernames can be a maximum of 20 characters in length and cannot end in a period (“.”). Many common usernames are blocked during the creation process. Examples of blocked account names include: 1, 123, a, admin, administrator, john, and several other easily guessable names. Passwords must be between 12 and 123 characters in length, and they must meet several complexity requirements.

Connect to Linux Virtual Machine

To connect to a Linux-based virtual machine, you can use a secure shell protocol (SSH) client. SSH is an encrypted connection protocol that allows secure sign-ins over unsecured connections. Depending on your organization's security policies, you can reuse a single public-private key pair to access multiple Azure virtual machines and services. You don't need a separate pair of keys for each virtual machine or service you wish to access. The **public key** is placed on your Linux virtual machine, or any other service that you wish to use with public-key cryptography. The **private key** remains on your local system. For Linux-based virtual machines, you can specify an existing SSH public key or a password when creating a Linux VM. Linux usernames must be between 1 and 32 characters in length, and passwords must be between 6 and 72 characters. Like Windows, certain easily guessable usernames and passwords are automatically blocked when creating through the Azure portal.

## Configure Virtual Machine Availability

Resiliency is a critical part of any application architecture. Azure provides several features and capabilities to make it easier to design resilient solutions. The platform helps you to avoid a single point of failure at the physical hardware level and provides techniques to avoid downtime during host updates. Using features such as availability zones, availability sets, and load balancers provides you the capabilities to build highly resilient and available systems.

Plan for Maintenance Downtime

Azure Administrators must be prepared for planned and unplanned failures. Let's explore three scenarios that can lead to your Azure virtual machine being impacted.

An **unplanned hardware maintenance** event occurs when the Azure platform predicts that the hardware or any platform component associated to a physical machine is about to fail. When the platform predicts a failure, it issues an unplanned hardware maintenance event. Azure uses Live Migration technology to migrate your virtual machines from the failing hardware to a healthy physical machine. Live Migration is a virtual machine preserving operation that only pauses the virtual machine for a short time, but performance might be reduced before or after the event.

**Unexpected downtime** occurs when the hardware or the physical infrastructure for your virtual machine fails unexpectedly. Unexpected downtime can include local network failures, local disk failures, or other rack level failures. When detected, the Azure platform automatically migrates (heals) your virtual machine to a healthy physical machine in the same datacenter. During the healing procedure, virtual machines experience downtime (reboot) and in some cases loss of the temporary drive.

**Planned maintenance** events are periodic updates made by Microsoft to the underlying Azure platform to improve overall reliability, performance, and security of the platform infrastructure that your virtual machines run on. Most of these updates are performed without any impact to your virtual machines or Cloud Services.

Setup Availability Sets

An availability set provides redundancy and availability for [virtual machines](https://cloudacademy.com/course/getting-started-with-azure-virtual-machines-988/course-introduction/). Deploying at least two virtual machines into an availability set ensures that at least one of them remains available whenever planned or unplanned maintenance occurs. When you deploy at least two virtual machines into an availability set, you qualify for a VM SLA of 99.95% uptime. Even if you deploy a single virtual machine, by itself, into an availability set, you can qualify for a 99.9% VM SLA, provided you use premium SSD or ultra-disk for all operating system disks and all data disks that are attached to the VM. Deploying VMs in an availability set will not protect an application from failures associated with the operating system of the VMs or from failures that are application-specific. However, placing virtual machines in an availability set will provide protection against network outages, physical hardware failures, and power interruptions within an Azure datacenter.

Review Update and Fault Domains

Every VM that is deployed into an availability set is assigned an update domain and a fault domain by [Microsoft Azure](https://cloudacademy.com/library/azure/). An availability set contains five update domains by default, although this can be increased to 20 update domains in resource manager deployments. An update domain is a group of virtual machines and underlying physical hardware that can be rebooted at the same time. When planned maintenance is performed on the Azure platform, only one update domain is rebooted at a time. This ensures that all VMs and associated hardware are not taken down at the same time. A fault domain is a group of virtual machines that shares a common power source and a common network switch. When virtual machines are added to an availability set, they are distributed across up to three different fault domains in resource manager deployments, or across two fault domains in classic deployments.

Review Availability Zones

Availability zones are similar in concept to availability sets. However, there is a distinct difference. While availability sets are used to protect applications from hardware failures within an Azure datacenter, availability zones, protect applications from complete Azure datacenter failures. An availability zone is a unique physical location that exists within an Azure region. Every availability zone contains at least one datacenter within the region. Each of these datacenters has its own power, its own networking, and its own cooling. To ensure resiliency, every enabled region in Azure consists of at least three separate zones that are physically separated. It is this physical separation that protects applications from datacenter failures. I should note here that when you deploy virtual machines in an availability zone, they will be covered by a 99.99% VM uptime SLA. Like availability sets, availability zones consist of fault domains and update domains. However, unlike availability sets, an availability zone consists of a single fault domain and a single update domain. Deploying three or more VMs across three availability zones within an Azure region will cause Azure to split those VMs across three different fault domains and three different update domains to ensure that the virtual machines in different zones are never updated at the same time. An availability set protects your azure resources from failures within datacenters whereas an availability zone protects from entire datacenter failures. From the Service Level Agreement (SLA) standpoint, like uptime and connectivity, with Availability Set Azure offers 99.95% SLA whereas with Availability Zone we have 99.99% SLA.

Compare Vertical to Horizontal Scaling

A robust virtual machine configuration includes support for scalability. Scalability allows throughput for a virtual machine in proportion to the availability of the associated hardware resources. A scalable virtual machine can handle increases in requests without adversely affecting response time and throughput. For most scaling operations, there are two different ways of scaling. You can scale up and down, or you can scale in and out. Scaling up and down, which is also referred to as vertical scaling, is the process of upgrading an existing [virtual machine](https://cloudacademy.com/course/advanced-vm-management-in-azure-1020/introduction/) to a more powerful virtual machine, or downgrading to a smaller, less powerful virtual machine. In [Azure](https://cloudacademy.com/library/azure/), this means resizing the existing VM to a larger size with more RAM and more CPU, or to a smaller size with less RAM and less CPU. Because vertical scaling involves the re-size of virtual machines, this kind of scaling results in VM restarts. Scaling out is a way to handle an increased load on an application or website. Scaling out is commonly referred to as horizontal scaling. When you scale out, you keep the same VM size, but you add more VM instances to the scale set. Instead of growing the size of the VMs in the scale set, you increase the number of VMs in the scale set. When you scale back in, you reduce the number of VMs in the scale set. You might want to use vertical scaling to scale down an application's server over the weekend, when the application is not in use. By scaling down over the weekend, you can save on compute costs. A common use for scaling in and out, or horizontal scaling, is when you have a public-facing website or web application. When supporting this type of application, it usually makes sense to add instances to support increased workloads, while removing instances when that load drops. This allows you to keep up with demand without causing an outage.

Create Scale Sets

An [Azure virtual machine scale set](https://cloudacademy.com/course/advanced-vm-management-in-azure-1020/vertical-scaling-vs-horizontal-scaling/) is essentially a group of identical, load-balanced VMs. You can configure an [Azure VM](https://cloudacademy.com/course/advanced-vm-management-in-azure-1020/introduction/) scale set to automatically increase the number of VM instances or decrease the number of VM instances based on demand or on a predefined schedule. Organizations use scale sets to ensure high availability for their applications and so they can centrally manage, configure, and update all VMs within the scale set. VM scale sets are often used to build large-scale compute solutions, to service big data solutions, and to host large container workloads. Because applications that run on a virtual machine scale set are distributed across several instances within the scale set, such applications benefit from redundancy and increased performance. An application that's hosted on a virtual machine scale set is usually accessed through a load balancer. The load balancer serves as a single point of access and then distributes application requests to any one of the instances within the scale set itself. A VM scale set allows you to perform maintenance and even updates on an application instance, without the need for any downtime. This is because requests are automatically distributed to other available instances within the scale set while you are performing your maintenance. Virtual machine scale sets allow you to easily create and manage multiple VMs that have the same configuration, because, in a scale set, all underlying VM instances are created from the same base OS image and configuration. That being the case, it becomes fairly easy to manage dozens or even hundreds of virtual machines without needing to worry about performing configuration tasks or even about network management.

VM scale sets can be used with Azure load balancers for basic layer four traffic distribution, or if you need more advanced layer seven distribution or SSL termination, you can use a scale set with Azure Application Gateway. Because scale sets sit behind load balancers, they provide high-availability and application resiliency. You can even use availability zones with scale sets to ensure that your VM instances within the scale sets are spread out within a single [Azure](https://cloudacademy.com/library/azure/) datacenter or even across multiple Azure data centers.

If you host an application on a manual group of virtual machines instead of a VM scale set, the process for adding additional instances to support the application becomes a manual process. While a VM scale set can automatically create new instances from a central config, using a manual group of VMs requires you to create, configure, and then ensure compliance of all of the additional VM instances that you manually spin up. Also load-balancing a group of VMs is a manual process, whereas a VM scale set handles the load-balancing for you automatically. VM scale sets automate the distribution of VM instances across availability sets or availability zones. Doing things, the hard way with a manual group of VMs requires you to manually perform these tasks.

Configure Autoscale

The key benefit of a VM scale set is its ability to automatically scale as resource demand dictates. As demand for an application increase, a VM scale set can automatically increase the number of VM instances to support the application load. The VM scale set can also automatically reduce the number of VM instances as demand for the application decreases. This is known as autoscaling. This autoscaling feature helps control costs because it minimizes the number of VM instances that are running. VM scale sets can support as many as 1000 virtual machine instances if you use Azure-supplied VM images. If you choose to use your own custom VM images, the limit is 600 VM instances. So, to repeat, Autoscaling allows you to dynamically scale your configuration to meet changing workload demands. Autoscaling minimizes the number of unnecessary virtual machine instances that run your application when demand is low. Your customers continue to receive an acceptable level of performance as demand grows and more virtual machine instances are automatically added. When you create an Azure Virtual Machine Scale Sets implementation in the Azure portal, you can enable autoscaling. For optimal performance, you should define a minimum, maximum, and default number of virtual machine instances to use during the autoscale process. Deploying a scale set incurs no additional costs outside of the underlying VM instance costs, load balancer costs, and any managed disk storage costs. There are also no costs associated with the cloud management and autoscaling features, outside of the charges for the VMs that are spun up.

Demonstration – Virtual Machine Autoscaling

Configure Virtual Machine Extensions

In Azure, there are a lot of Configuration Management Tools available for Virtual Machines. A configuration management tool is a software or a programming language used to configure Operating systems. We can use Ansible, Chef and Puppet for example, or Cloud-INIT for Linux machines or Azure custom script extensions for both Windows and Linux VMs. There is also PowerShell Desired State Configuration.

Let's start with Ansible. Ansible is used for VM creation, application deployment, and for configuration management. It's an automation engine that uses an agentless model to manage target machines. To configure VMs with Ansible, you define specific configuration tasks in what are known as playbooks. Ansible modules are then used to perform specific configuration tasks.

Chef is another tool that can be used to manage VM configuration. You can use Chef to configure, deploy, and manage your infrastructure. Using Chef to manage the configuration of virtual machines requires you to install the Chef client on them. The Chef client communicates with one or more centralized Chef servers that hold and manage the configurations for your virtual machines.

Like Chef, Puppet is another enterprise-ready, automation platform. It's used to automate the deployment process as well as application delivery. To use Puppet, you install agents on your target VMs. This allows Puppet Master to define your desired configuration. You can even integrate Puppet with solutions like Jenkins and GitHub. This is typically done to improve the DevOps workflow for the organization.

When it comes to automating the configuration of a Linux VM, you can use Cloud-INIT. Cloud-INIT is a popular way of customizing a Linux VM during its initial boot. Cloud-INIT can install packages and even write files. It can also be used to configure users and security. Since it runs during the initial boot process, Cloud-INIT doesn't require any agents nor does it require any additional steps. When you use Cloud-INIT to customize a Linux VM, you don't use apt-get install, or YUM to install packages. What you do is define a list of packages that you want to install. Cloud-INIT will then use the package management tool for your specific distro to install your defined packages.

Azure VM extensions are used to perform post-deployment configuration and automation tasks on [Azure VMs](https://cloudacademy.com/course/advanced-vm-management-in-azure-1020/introduction/). Azure Architects use various mechanisms to complete post-deployment configuration tasks and automation operations for their Azure virtual machines. When a virtual machine requires software installation or anti-virus protection, or the machine needs to run a script, you can use virtual machine extensions to manage these processes. Virtual machine extensions are often used to deploy things like antivirus agents, monitoring agents, and even agents that are used by Chef and Puppet. You can use a virtual machine extension during the initial deployment of a new VM, or you can run extensions against a VM after it's been deployed.

Implement Custom Script Extensions

In the Azure VM extensions, we have Azure custom script extension, that can be used to download and execute scripts on both Linux VMs and Windows VMs. While the custom script extension is often used during the creation of a VM, it can also be used even after the VM has been created. Scripts that are used with the Azure custom script extension can be downloaded from Azure storage or from other public locations. GitHub repositories are a common source for script downloads. A common example of the Azure custom script extension in action, that you'll often find online, is where it is used to automatically install IIS on a Windows VM. We will actually work through a demonstration of this later on.

Implement Desired State Configuration

Last, but not least, PowerShell Desired State Configuration, or DSC, is a management platform that you can use to specify the configuration of target machines. You can use it to configure Windows machines natively, or you can use it on Linux through the open management infrastructure server offering from the open group. PowerShell DSC is typically used to ensure a VM remains configured as required. You use a DSC configuration to specify what software should be installed on a VM. You can also use a DSC configuration to define how a VM should be configured. To use PowerShell DSC, you must deploy a Local Configuration Manager engine on each target VM. This local engine, or LCM, configures the VM based on configurations that are pushed down to it. These configurations are stored on a central pull server, which communicates with the local LCM engine on each target VM so that it can provide the necessary configurations while also reporting on compliance.

Demonstration – Custom Script Extension

Install Network Watcher Agent to a VM called demo-vm01.

Install IIS through a custom script extension.