O'REILLY®

Compliments of turbonomic

Microsoft Azure IaaS Solutions

Deploying and Managing the Azure IaaS Platform

Eric Wright

turbonomic

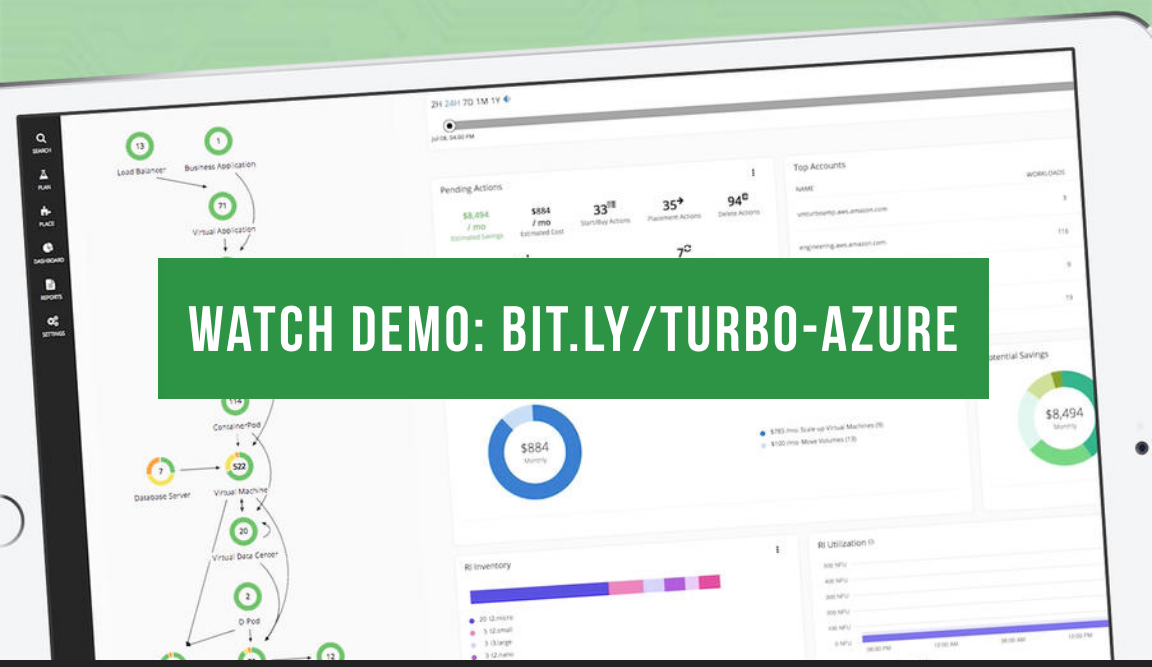
A Microsoft Preferred Partner to
Assess, Migrate and Optimize Cloud Deployments

7 DAYS TO AZURE

30% BETTER PERFORMANCE

30% LOWER COST

30-MINUTE INSTALLATION



WATCH DEMO: BIT.LY/TURBO-AZURE

“Turbonomic and Azure play a critical role in our dynamic developer infrastructure. We would not be able to deliver a genuinely elastic environment otherwise.”

Technical Manager, Cloud Services
ICF

Microsoft Azure IaaS Solutions

Deploying and Managing the Azure IaaS Platform

Eric Wright

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

Microsoft Azure IaaS Solutions
by Eric Wright

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com/safari*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Acquisitions Editor: Nikki McDonald
Development Editor: Virginia Wilson
Production Editor: Justin Billing
Copyeditor: Octal Publishing, LLC
Proofreader: Chris Edwards
Interior Designer: David Futato
Cover Designer: Karen Montgomery
Illustrator: Rebecca Demarest

September 2018: First Edition

Revision History for the First Edition
2018-09-18: First Release

978-1-492-04512-0

[LSI]

Table of Contents

Foreword

Every generation has its defining industries. For our generation, that defining industry is IT. We are creating opportunities and innovations in ways that are changing the rules and limits we once thought were fixed. Let's take, for example, Moore's law. We knew it was happening. There was no doubt about that. The cost of compute continued to decline precipitously. But what would that mean for the experiences that we could deliver? The ramifications of that progress over five or six years, or a decade, really stretches the imagination.

Today, the ability to create and deliver entire solutions in minutes, with fully scalable global infrastructure as a standard, has empowered a new generation of content creators and innovators. Anyone with a few dollars and a brilliant idea now has access to worldwide cutting-edge data platforms and compute arrays. We find ourselves at the precipice of a new wave of innovation, powered by the abstraction of infrastructure, and a new generation at the helm. Due to the very nature of the technology, the pace of change is faster than past technology revolutions—and we must capitalize quickly or be left behind, patching servers one at a time.

The public cloud has opened up incredible possibilities to accelerate growth and innovation in ways that have never been available up to this point, and the possibilities continue to grow. Hybrid and public cloud are now a core part of many organizations' strategies. The true capability and power of the hybrid cloud is finally being realized with workloads running in multiple clouds, on and off premises, and this is just the beginning of the next wave of innovation.

It's my pleasure to work with Eric at Turbonomic as we lead this change and bring the industry and our community into the Azure and hybrid cloud generation.

— Bill Veghte
Executive chairman, Turbonomic
Former COO, Hewlett-Packard
Former senior vice president,
Windows @ Microsoft

Preface

Welcome to the *Microsoft Azure IaaS Solutions* guide. The goal of this guide is to introduce systems administrators, systems architects, and newcomers to Microsoft Azure to some powerful core offerings on the Microsoft public cloud platform.

You will learn common terms, design patterns, and some specific examples of how to deploy IaaS solutions for compute, network, and storage on Azure using both the Azure command-line interface (CLI) and the Azure portal interface. By the end, you will be able to launch and manage Azure IaaS solutions including virtual machines and storage, understand the implications and requirements for security, and identity and access management on Microsoft Azure.

Additional resources are provided throughout the guide for you to explore some of the services and technical examples further. Resources, code samples, and additional reading links for this guide are available online at *https://discopos.se/DeployingAzureSolutions*.

Thanks go out to the entire Azure technical community, the O'Reilly team, and my family for the help and guidance in creating this guide.

— Eric Wright (@DiscoPosse)
August 2018

CHAPTER 1

Introduction to Microsoft Azure

Microsoft Azure is a public cloud platform featuring powerful on-demand infrastructure and solutions for building and deploying applications workloads as well as a wide variety of IT and application services. You can use Azure as a public cloud provider and as a hybrid extension to existing on-premises infrastructure. Organizations that use Microsoft solutions on-premises are able to easily extend their infrastructure and operational processes to Azure.

With the growing popularity of Azure, today's systems administrators need to acquire and strengthen their skills on this fast-growing public cloud platform. In this chapter we explore the Azure public cloud platform with a focus on the Infrastructure-as-a-Service (IaaS) features. We cover general architectural features of the Azure cloud including geographic regions, availability zones, and Service Level Agreements (SLAs) attached to the core Azure IaaS infrastructure.

Check out a full glossary of Azure terms available as a link in the additional resources.

Regions, Availability Zones, Availability Sets, and Uptime SLAs

The Azure cloud environment is segmented logically and physically to provide the following:

Geographic availability
: Low-latency access to geographic locations for more rapid application and service access

Geographic resiliency
: Multiple points of presence for distributing applications, workloads, and services to allow for high availability

Core services are available across the entire infrastructure, including Domain Name System (DNS), security, identity and directory services, and others that are often described as *oxygen services*.

The geographic layout of Azure is divided up into locations grouped into regions, and within each region they are physically separated Availability Zones.

Regions

Azure touts the largest public cloud, and it is growing at the fastest rate by percentage of any public cloud to date with 54 regions as of this writing. *Regions* are defined as an area within a specific geography that does not span across national borders and that contains one or more datacenters.

Regional access is an important consideration for many technical and business reasons. Both deployment considerations and user experience are affected by the availability of multiple regions. You must also weigh advantages against design considerations and complexity when using multiregion architectures.

Using multiple regions in order to support scale-out application and virtual machine deployments provides a way to ensure resiliency and availability. This concept is explored later in this guide in "Design Patterns for Availability Using Azure Virtual Machines" on page 22.

Another use case is ensuring low-latency access to customers within a specific region (e.g., customers in Asia-Pacific geographies would suffer from latency if they were to access a North American region).

There are also specialty regions that are purpose-built to deal with regulatory and governmental boundaries. These include the following:

- US Gov Virginia and US Gov Iowa
- China East and China North
- Germany Central and Germany Northeast

Each specialty region is designed to solve for specific governmental and security regulations that require distinct cloud environments for targeted customers with these requirements (e.g., FedRAMP, DISA).

Regional clouds in China and Germany provide local datacenter operations to be controlled by country-specific providers, which is a requirement for data sovereignty and other regulatory boundaries specific to those regions.

Paired Regions

Another feature within Azure is *Paired Regions*. These regions are in the same geography but are typically at least 300 miles apart and provide the ability to deploy cross-region services and applications while maintaining geographic residency.

Paired Regions also have operational processes that ensure that sequential updates occur and that prioritized regional recovery occurs in the event of an outage. This provides you with better resiliency options for application and systems architects to use when designing your Azure solutions.

Specific Azure services have replication options and will take advantage of the paired region, as shown in Figure 1-1, as the replication target in order to maintain geographic residency for data and application workloads.

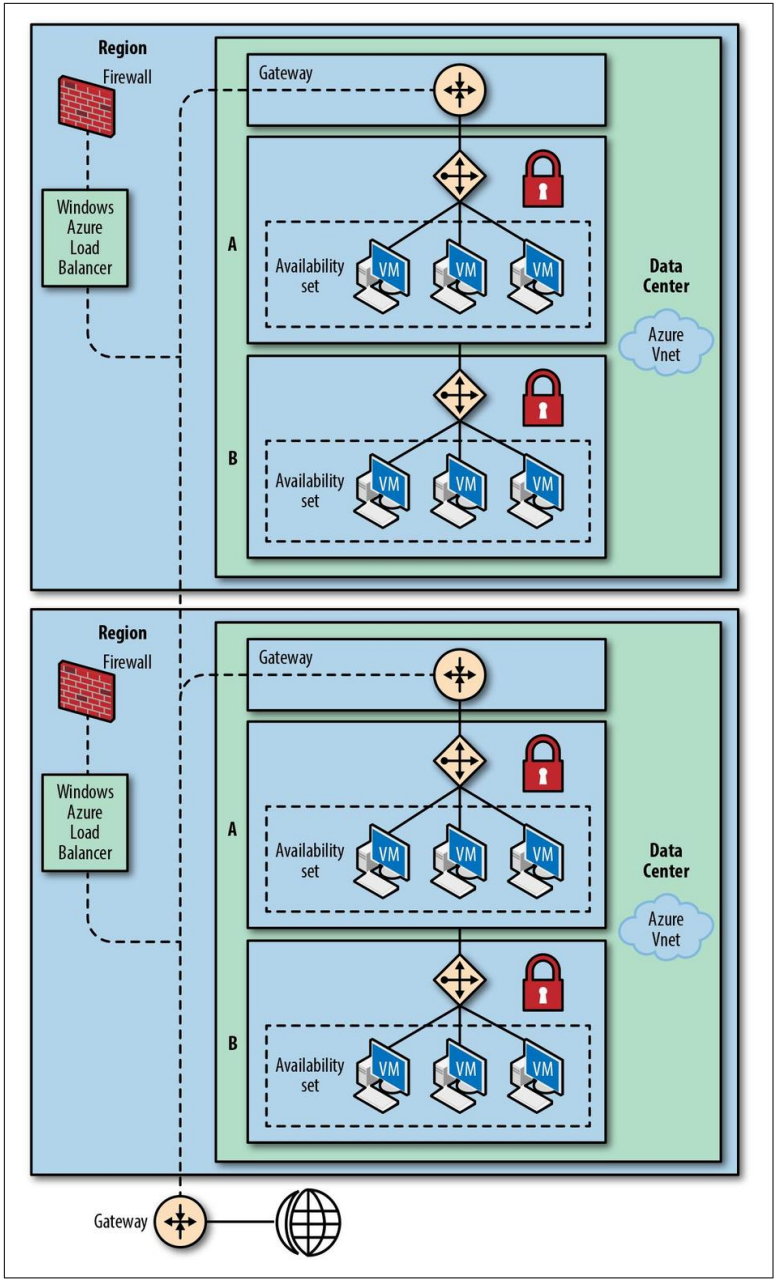


Figure 1-1. Logical design example of a paired region

Using Paired Regions enables deployment patterns that can include applications that might be replicated rather than used in a distributed deployment. This enables active–passive deployment patterns with low-latency access to the second region for rapid recovery in the case of a fault.

Paired Regions services that can be replicated include compute (Azure Virtual Machines), Storage, and Database services. Additional third-party products are available to replicate resources and data outside of the native Azure offerings.

Additional reading and resources for Paired Regions are available online at *http://bit.ly/2Mv9Tlv*.

You can take advantage of the built-in offerings to create or enhance your business continuity and disaster recovery strategy using Azure. This is among one of the many ways to take advantage of the on-demand and built-in capabilities.

Availability Zones

Each region comprises at least one *Availability Zone*, which is defined as a datacenter with independent power, network, and cooling environments. Each Availability Zone is separated by a reasonable distance to ensure protection from a significant disruption (e.g., power grid failure) while also being close enough to maintain low-latency network access to other Availability Zones within the region.

Prior to 2016, Azure abstracted the physical topology within a region from the customer. This has been updated to include specific deployment and visibility of Availability Zones (formerly known as datacenters). There are three supported regions (Central US, France Central, West Europe) and two additional regions that are in preview (East US 2, Southeast Asia) as of this writing.

Availability Sets

Azure provides a powerful resiliency option called *Availability Sets*. This logical construct is made up of multiple VMs that usually make up a distributed application. The Availability Sets option also introduces the concept of a *fault domain*. Availability Sets distribute

across fault domains to ensure greater availability in the case of a localized failure within the Azure infrastructure that could affect application availability on a single VM.

Update domains are also used for Availability Sets, and define the VMs that can be rebooted while still ensuring minimum application access within the Availability Set. This is especially important when designing for operational practices such as patching and software updates.

SLAs on Azure

Each of the Azure services provides SLAs for availability and guidance on how to increase availability through the use of architectural patterns such as using multiple Availability Zones, regions, and other methods to ensure application and service availability.

You calculate availability using the following formula:

> Monthly Uptime % = (Minutes in the Month – Downtime) / Minutes in the Month 100

Azure customers receive a service credit for the Azure services that did not achieve the SLA in the event of a loss of service. Most of the Azure services are credited as follows in single-resource deployments:

> <99.9% Availability = 10% credit
>
> <99% Availability = 25% credit
>
> <95% Availability = 100% credit

Some services vary on SLA options. This can be because of the maturity of the service, the geographic availability, and the criticality of the service. As an example, the Azure DNS service was recently upgraded to a 100% uptime SLA.

Azure Virtual Machines raise the SLA and are credited as follows:

> <99.99% Availability = 10% credit
>
> <99% Availability = 25% credit
>
> <95% Availability = 100% credit

It is important to consider the SLA requirements for each service and to remember that if an SLA is missed, the result is only a credit toward your Azure subscription. Customers are responsible for designing and deploying to meet their own SLA requirements and

to ensure high availability across all layers of the application stack. Each of the Azure services provides service tiers, design patterns, and options to increase availability across the environment.

Now that you have a basic understanding of the Azure environment and architecture, we move on to the IaaS compute platform, and deploy and perform some common operations processes in both the Azure portal and using the Azure CLI.

CHAPTER 2

Azure Virtual Machines

In this chapter, we explore Microsoft Azure Virtual Machines and illustrate how the service compares to on-premises virtualization. You will learn how to deploy an Azure virtual machine (VM) including the various parameters and settings that you can configure. We use the Azure portal (web interface) and Azure CLI, and show how to use the Azure Cloud Shell (web-based CLI) to perform administrative tasks on the Azure VM examples presented here.

Understanding and Deploying on the Azure Compute Platform

The Azure Virtual Machines service is ideal when an organization needs to control more of the cloud workload, including the underlying operating system (OS) and other OS-level dependencies (e.g., applications, libraries, and custom code). On-premises virtualization is a form of IaaS familiar to most systems administrators.

The elasticity of on-demand Azure VMs allows organizations to deploy and scale to meet the demand of developers and customers without the burden of operating the underlying infrastructure. This new on-demand infrastructure model introduces the need for new deployment and design patterns to ensure availability and protection of cloud-based resources on this new on-demand infrastructure model. Organizations must also be aware of the cost of deploying and maintaining resources that are normally treated as sunk costs in a fixed supply, on-premises environment.

VMs are available with a variety of operating systems and many pre-packaged images from the Azure Marketplace, as shown in Figure 2-1.

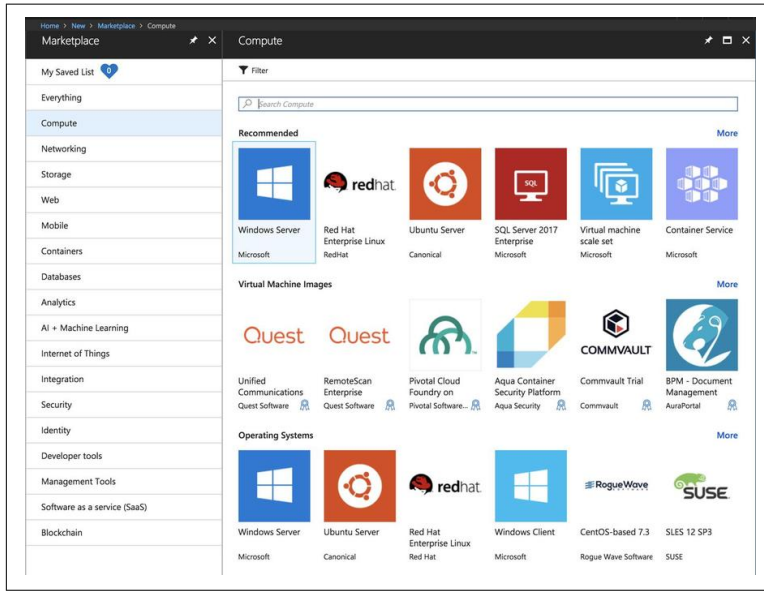


Figure 2-1. A snapshot of the Azure VM catalog

OS choices provide flexibility to meet the needs of your application workloads. When choosing a Microsoft Windows licensed VM, the licensing of the OS is included as a per-hour price and does not require the addition of client access licenses as per the licensing guidelines in Azure.

Some VMs will incur additional hourly or monthly charges based on application licenses, which are often available on-demand, as well. This is an excellent consumption model where applications can be tested without committing to the full cost upfront. Microsoft application licenses (e.g., Microsoft SQL, Microsoft BizTalk, Microsoft System Center) may also take advantage of License Mobility for Software Assurance customers. Additional options are available for Service Provider Licensing and Open Licensing programs.

Many of the VM images and applications support a Bring Your Own License option for organizations with existing Enterprise agreements or licenses that you can apply to your Azure environment.

There are also additional deployment options with prebuilt Managed Images using the HashiCorp Packer that is popular for deploying across hybrid infrastructures using common images.

Understanding and Using Azure Resource Manager

You can define and deploy Azure infrastructure by using Azure Resource Manager. Resource Manager groups services and resources together as a single solution, which simplifies the initial and ongoing management. Prior to the availability of Resource Manager, resources were deployable only in what is called the "classic deployment model," which required configuring and deploying each individual resource or service.

Resource Manager templates allow for the use of declarative descriptions of resources which were formerly entity-level configuration (the only available method using the classic deployment). Resource Manager configurations include the ability to do the following:

- Manage multiple resources using a common configuration
- Repeat deployments using a declarative template and ensure consistency
- Dependency definition to ensure order of operations during deployment
- Tagging, access control, and more, all definable in your Resource Manager configurations

You can use the Azure portal during the creation of resources via the Resource Manager interface, which also outputs the declarative code that can later be used for programmatic deployment and configuration of those resources. Resource templates are also available on the Azure GitHub, which provides practical examples to use and adapt.

Creating and Managing Azure Virtual Machines in the Azure Portal

You can create an Azure VM quickly using the Azure portal in any browser. There are some prerequisites for creating your first VM:

Virtual network
: You need private internal networking configured for Azure.

Secure Shell (SSH) Key for remote access
: You need this for Linux and other operating systems using SSH to administer remotely.

Storage account
: Monitoring, storage, and other resources require a storage account, which you configure when you set up your first Resource Groups.

You can set up each of these prerequisites using the new VM wizard if they are not already set. It is important to understand the needs of the application to ensure it is configured for proper logical isolation as well as access to necessary resources within a resource group.

Further decisions that you need to make during deployment include the following:

VM name
: Assign a unique name to the device.

Storage type
: Choose solid-state drive (SSD) or hard-disk drive (HDD).

Location
: Choose an Azure region.

Size
: Choose a SKU that matches the VM and application needs.

High availability
: Choose whether to use an Availability Set or not.

Resource group
: Use an existing resource group or create a new one.

Storage tier
: Choose whether to use Azure Managed Disks or not.

Networking and public IP address
: Assign appropriate internal and external networking.

Network security group
: Apply firewall and access policies from an existing network security group or create a new group.

Other options
: Set backups, Active Directory membership, and so on.

Some options will incur additional charges, including network addressing and specific storage tiers. There are also extra charges for ingress and egress networking, which is billed on-demand for running VMs.

Example: Deploying a CentOS VM on Azure Compute

Figure 2-2 illustrates a CentOS-based Azure VM deployed using the Azure portal following the Resource Manager model. Using Resource Manager eliminates the need to individually configure your virtual network, storage configuration, and network security groups in many cases.

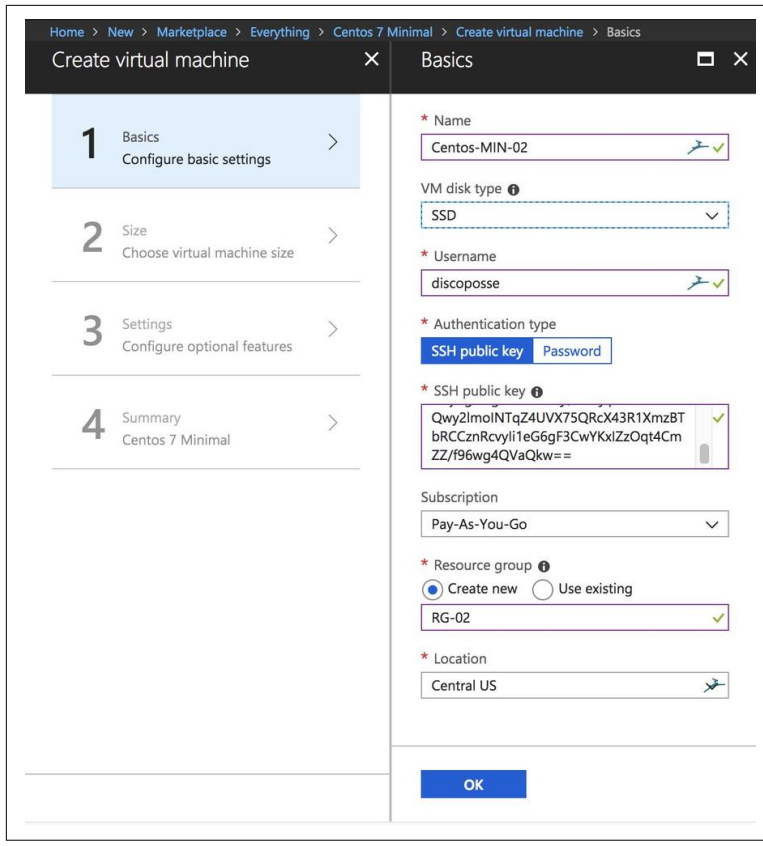


Figure 2-2. Basic options for creating the VM

Choose the SKU for your Azure VM based on the CPU, memory, storage, and performance requirements for your application workload. The righthand column in Figure 2-3 shows the monthly cost of the chosen SKU in the local currency of your Azure subscription.

SKU sizes and availability in regions will vary based on OS and configuration type. Not all SKUs are available in every region. Please consult the online SKU matrix for continuous up-to-date information:

Linux VM SKU Sizes

Windows VM SKU Sizes

In this example, the deployment is being done using a general-purpose B1s SKU, as shown in Figure 2-3, with a single virtual CPU

and 1 GB of virtual memory. This is the lowest cost SKU for this VM, but you can change this as needed by simply modifying the configuration to a new SKU and restarting the VM. Be aware that SKU changes are disruptive because of the need for a restart to apply the update.

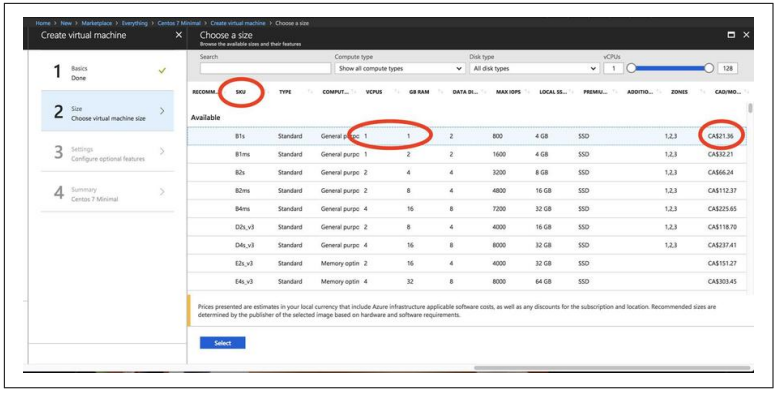


Figure 2-3. Selecting a SKU for your VM

Availability, resiliency, networking, and security options are configured next. Options here include whether to include this VM as part of an Availability Set. This is a standalone machine, which will simply need a single public IP address, and storage will be chosen as a managed disk for ease of administration. The private virtual network is already configured for internal IP addressing.

Network security groups define your security and firewall options. Each network security group is configured for multiple inbound and outbound rules using the source and destination IP address attached to specific IPs, ports, and protocols, as seen in Figure 2-4.

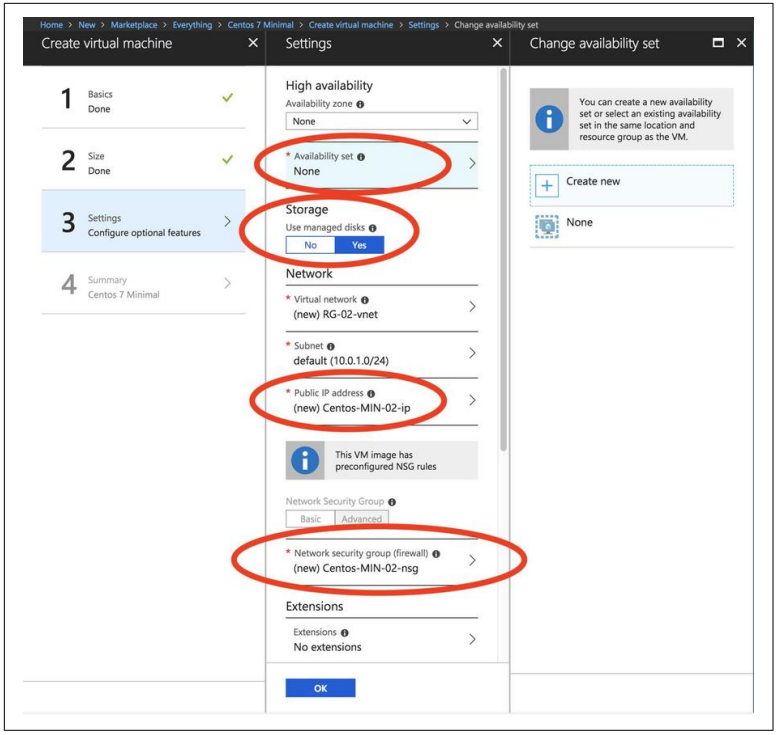


Figure 2-4. Additional VM availability, storage, and networking options

Step 4 brings us to the final part of your deployment, depicted in Figure 2-5. A powerful feature in Azure is that the template output is available as part of every deployment in the Azure portal. Simply click the "Download template and parameters" link before completing the creation process.

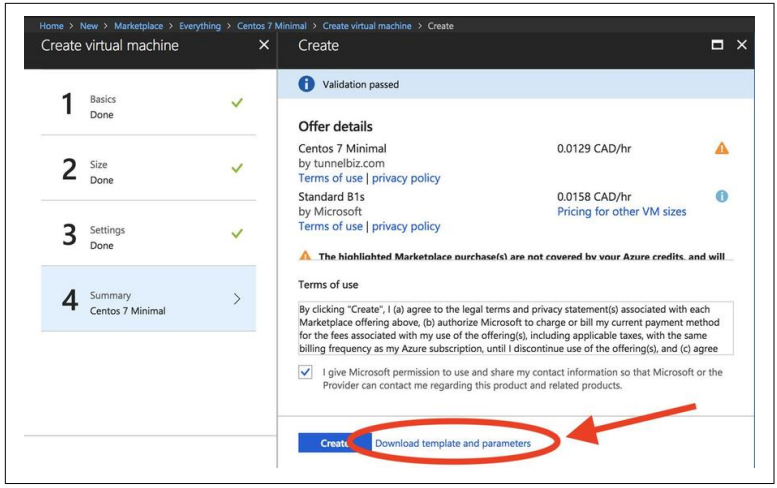


Figure 2-5. Optional ARM template download link

You can use this output to script the creation of similar VMs at a later time. The JSON output shown in Figure 2-6 can be downloaded here, stored in a library, or simply pasted into files for remote CLI use, as well.

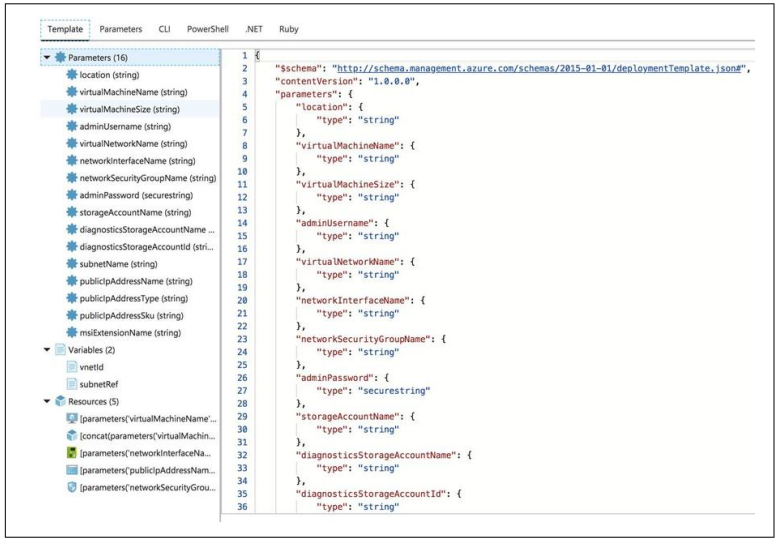


Figure 2-6. JSON example of an Azure VM

Parameters are presented in JSON for use with whichever programmatic deployment tool you choose. Clicking through the various tabs provides access to each language or shell type.

The CLI option is a fully documented Bash script, as shown in Figure 2-7.

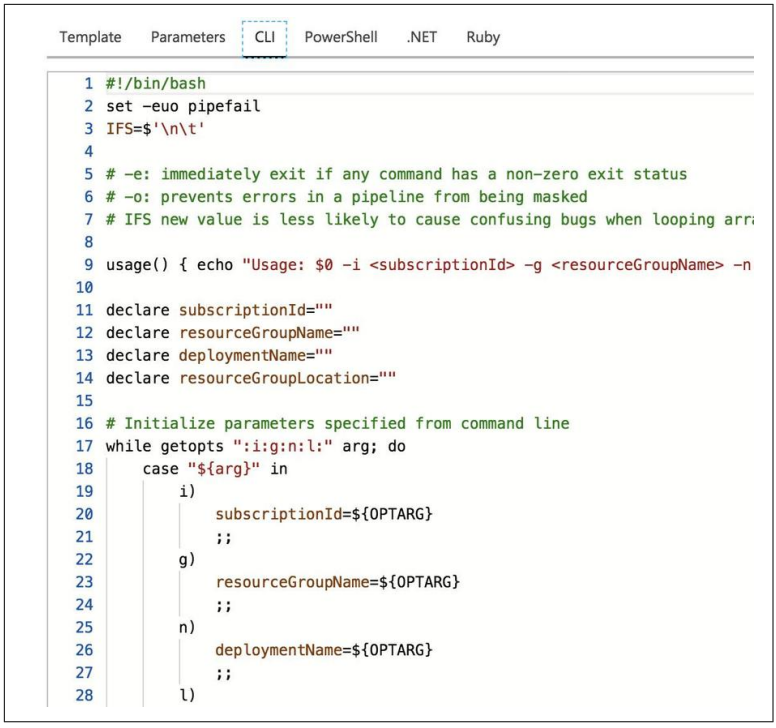


Figure 2-7. Bash script example

The PowerShell tab also offers a fully documented script, as illustrated in Figure 2-8, which you can use along with the parameters file to create and recreate the resource in any PowerShell environment.

You must have PowerShell v5.0 or higher to apply Resource Manager templates. This version comes preinstalled with Windows 10 and can be installed separately on Windows 7.

For more information, go to *http://bit.ly/2xkApsa*.

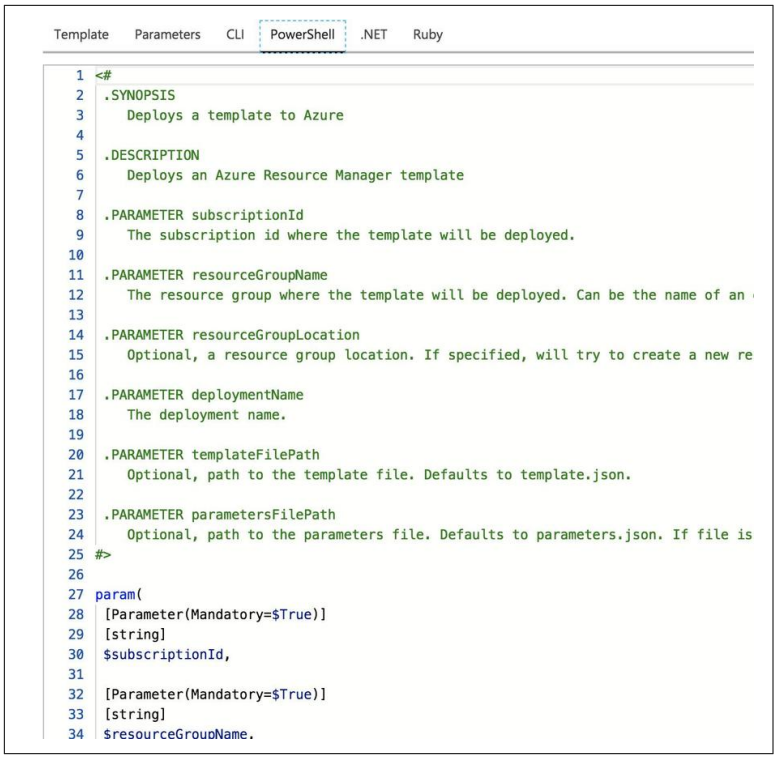


Figure 2-8. PowerShell script example

All that remains is for you to review the overall configuration and billing information, review and accept the licensing and Terms of Use agreement, and complete by clicking the Create button. The VM will become available in the Azure portal.

Managing Azure Virtual Machines in the Azure Cloud Shell

You can manage resources deployed in Azure by using the Azure portal, the Azure CLI (available for multiple operating systems), PowerShell, and the Azure Cloud Shell. Cloud Shell is available in your web browser in the upper-right corner, as shown in Figure 2-9.

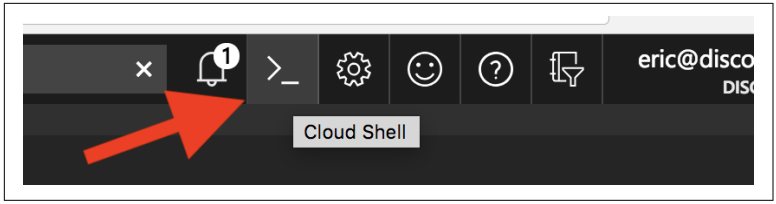


Figure 2-9. The Cloud Shell link in the Azure portal

Options include PowerShell or Bash, which provides flexibility for the environment you are accustomed to. The Cloud Shell examples here are using the `az` command, which is also available to run locally on-premises to remotely manage Azure resources.

You can run the Azure CLI using a Docker container as well. This reduces the need to install and update the CLI on your local workstation or server.

Simply run this command to launch a Docker container and then use the `az` command in the interactive container:

```
docker run -it microsoft/azure-cli
```

For more on Azure CLI in Docker, visit *http://bit.ly/2xfiBPp*.

Resizing an Azure VM in Cloud Shell

You can resize VMs easily with simple one-line commands. The first command used here provides the resize options (SKUs) for a specific VM (Centos-MIN-02) in a resource group (RG-02) and provides the output in a table format (default is JSON).

```
az vm list-vm-resize-options --resource-group RG-02 \
    --name Centos-MIN-02 --output table
```

Figure 2-10 presents the results.

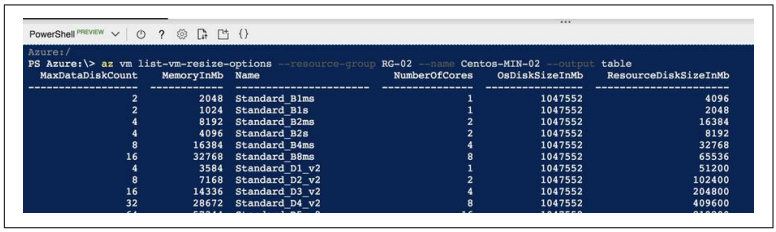


Figure 2-10. Listing VM SKU options using the Azure CLI

Use the following command to apply a new SKU (Standard_B2ms) to the VM that is currently running as a Standard_B1ms. Note that you will not be prompted to confirm the changes, so it is important to understand which commands are disruptive to the VM. Resize commands, for example, will restart the Azure VM with the new SKU:

```
az vm resize --resource-group RG-02 --name Centos-MIN-02 \
    --size Standard_B2ms
```

Each command provides a JSON result indicating changes, parameters, and success/error results, as demonstrated in Figure 2-11.

Figure 2-11. Resizing an Azure VM using the Azure CLI

Rebooting an Azure VM in Azure Cloud Shell

Other commands such as rebooting are also quite simple in the CLI:

```
az vm restart --resource-group RG-02 --name Centos-MIN-02
```

Figure 2-12 displays the results.

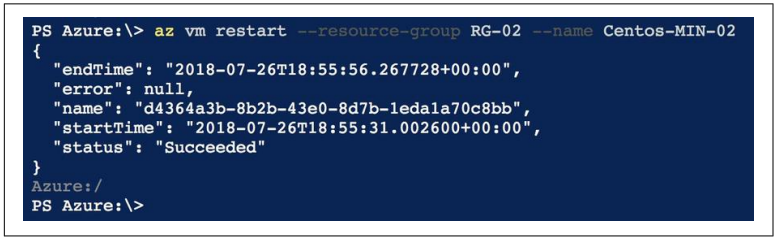


Figure 2-12. Restarting a VM using the Azure CLI

These and many other examples are also available at the Microsoft site, which is updated regularly as new CLI options and PowerShell CmdLets become available.

Design Patterns for Availability Using Azure Virtual Machines

You can ensure availability of your applications using Azure Virtual Machines by taking advantage of existing Azure capabilities. Practices that ensure availability on an Azure environment include, but are not limited to, the following:

Deploy using Availability Sets
: Ensure distribution of resources across failure domains in case of a localized failure.

Use multiple VMs across Availability Zones
: Use distributed VMs to ensure a three 9s (99.9%) SLA and utilize more application-layer options to distribute load, but beware of latency in these designs.

Deploy across regions or using Paired Regions
: Extend beyond just a single region to ensure greater availability even in cases of a localized and significant disruption.

Ensure data and application synchronization across distributed systems
: Application layer and data layers must be kept synchronized or be able to tolerate being distributed across networks and regions.

Use replication services to provide on-demand recovery options
: Keeping asynchronous copies and continuous snapshot copies of resources ensures that you can rollback or recover to an alternate location quickly.

Back up all resources
: Availability and recoverability are two different things. Always back up so that you have point-in-time recovery in case of malware, code or data issues, and other problems that require recovery from an archived version.

Cloud infrastructure does require occasional maintenance, and can cause some resources to lose connectivity without warning. Other unplanned issues can also occur, furthering the need to adopt resiliency in many operational layers. Microsoft will notify you in advance of planned service outages.

However, some of your Azure resources might not need extremely resilient designs. You need to use discretion when designing for the public cloud because resilience comes at a cost, every hour. The balance of resilience and risk versus ongoing cost should be considered in business requirements, application design processes, and documentation.

Reserved VMs are also available for long-running workloads, which locks in a lower price for a specific term (one to three years). The topic of reserved capacity goes beyond our coverage in this guide. Resources for further study are available at *https://discopos.se/DeployingAzureSolutions*.

Now that you understand Azure Virtual Machines infrastructure, let's explore the Azure storage environment as it relates to Azure VMs.

CHAPTER 3

Azure Storage for Virtual Machines

As the Microsoft Azure environment has evolved, so have the storage offerings. Storage is divided by types of storage and performance.

On-premises and cloud storage use different requirements that can make cloud storage appear confusing. The number of choices and flexibility of Azure storage will prove to be an advantage as you become more comfortable with the platform.

Storage types include Storage Accounts or Managed Disks, each of which offers different performance tiers and service offerings around replication, resiliency, and other features.

In this chapter we explore the basics of Azure Storage, including storage tiers, features, resiliency options, and capabilities for storage attached to your Azure Virtual Machines infrastructure. You will also learn important deployment patterns and replication options to increase the availability and recoverability of your Azure Storage workloads.

Storage Accounts

Azure provides storage options for many workloads, from VMs to object storage and everything in between. Azure Storage comes in many types that span legacy compute support and can include other services across the entire Azure family.

You use storage accounts to manage groups of storage objects. These storage options can include different tiers of performance, as well. Deploying an Azure VM with traditional unmanaged disks requires a storage account to hold the virtual hard disks (VHDs). These are stored as Binary Large OBjects (blobs) and must be scaled individually to manage performance and capacity.

The storage types that systems administrators should be aware of include the following:

Azure Blob Storage
: Scalable object stores that hold binary and text data including page blobs, which is how VHD files are stored

Azure File Storage
: File shares without the need for underlying management of file servers

Azure Table Storage
: NoSQL storage for structured data

Azure Queue Storage
: Used for holding message queue data (important for distributed scale-out applications)

Block blob storage is also available as Hot and Cold storage. Hot storage is used for regularly accessed resources kept on fast, flash-based hardware, whereas Cold storage data, designed for infrequent access, is stored on much slower HDD hardware.

Cold storage is ideal for recovery images and backups, but it is not well suited for production-running workloads.

Azure Managed Disks

You can also take advantage of the more resilient option, Azure Managed Disks, to gain more stability and scalability for VM disk storage on Azure. Managed Disks come in different tiers:

Premium Managed Disks
: Offered in seven types, scaled by capacity and by input/output operations per second (IOPs)

Standard Managed Disks
: Offered in seven types, scaled by capacity with consistent but low-performing IOPs

Tables 3-1 and 3-2 illustrate the Azure Storage types across the two tiers.

Table 3-1. Matrix of Premium disks (courtesy of Microsoft; source: http://bit.ly/2xe3fKU)

Premium disk type	P4	P6	P10	P15	P20	P30	P40	P50
Disk size	32 GB	64 GB	128 GB	256 GB	512 GB	1024 GB (1 TB)	2048 GB (2 TB)	4095 GB (4 TB)
IOPS per disk	120	240	500	1100	2300	5000	7500	7500
Throughput per disk	25 MBps	50 MBps	100 MBps	125 MBps	150 MBps	200 MBps	250 MBps	250 MBps

Table 3-2. Matrix of Standard disks (courtesy of Microsoft; source: http://bit.ly/2xe3fKU)

Standard disk type	S4	S6	S10	S15	S20	S30	S40	S50
Disk size	30 GB	64 GB	128 GB	256 GB	512 GB	1024 GB (1 TB)	2048 GB (2 TB)	4095 GB (4 TB)
IOPS per disk	500	500	500	500	500	500	500	500
Throughput per disk	60 MBps	60 MBps	60 MBps	60 MBps	60 MBps	60 MBps	60 MBps	60 MBps

Choosing your storage is a delicate and challenging task. Changing storage types is possible, but doing so disrupts your Azure VMs given that it requires a reboot to detach and reattach the VHD from the new location.

For example, a transactional system (e.g., database workloads) that performs frequent reads and writes to the underlying storage will be better suited to the Premium Disk type to access the scalable IOPS. Other applications that require less frequent or less volatile access to storage (e.g., simple client/server or file repository) might make better candidates for Standard Disk type.

Another consideration for your storage choice is cost. When choosing performance tiers, or managed versus unmanaged storage, you must also consider the direct per-hour cost as well as the long-term administrative overhead of each option.

Profiling your workload performance and consumption needs is very important before choosing your cloud storage option. Storage can be a significant bottleneck and must also match the appropriate SKU for your compute choice.

Storage Replication Options

Storage replication is available on Azure, with multiple models to provide resiliency and availability across different failure domains. There are four storage replication models:

Locally Redundant Storage (LRS)
: Able to withstand partial loss of underlying storage hardware

Zone Redundant Storage (ZRS)
: Able to withstand loss of access to an Availability Zone

Geo-Redundant Storage (GRS)
: Able to withstand the loss of access to a region

Read-Access Geo-Redundant Storage (RA-GRS)
: Able to withstand loss of access to a region with read-only access at a remote region

Each option presents distinct advantages for the chosen resiliency, but you must weigh them along with the cost and administrative requirements to manage them on an ongoing basis. Cloud storage differs greatly from on-premises storage in both operational practices and in day-to-day costs.

Design Patterns for Availability Using Azure Storage

Using Managed Disks and the built-in storage capabilities also ensures greater resiliency. Design patterns to consider for increasing your storage availability include, but are not limited to, the following:

Use storage replication options
: Make use of built-in replication and availability in the underlying storage architecture.

Back up Azure VMs and use snapshots
: Always use a backup process to ensure application-consistent backups and use snapshots where appropriate to store safe copies of your Azure VM disks.

Use third-party storage options
: Many storage companies provide distributed storage using Azure appliances and can often provide a proxy to on-premises for seamless management across the hybrid estate.

The most important part of your resiliency strategy is matching to business and workload requirements including any constraints on budget and application-level ability.

Extending your storage across the hybrid environment also introduces additional latency, which must be continuously monitored and accounted for because it can affect applications and overall performance.

With this understanding of your storage options, we turn next to identity and access management and how to assign and restrict access to your Azure IaaS resources.

CHAPTER 4

Identity and Access Management

In this chapter, you will learn how Microsoft Azure handles identity and access management. We cover both how and why you can provide or prevent access to resources. We also explore Azure Active Directory (Azure AD) and how it relates to your existing Microsoft Active Directory with a look at different support tiers and the associated features available for your organization as a result.

Access Control and Authorization

There are two critical security functions in any IT environment:

Authentication
: Who are you?

Authorization
: Are you permitted to perform a specific task?

Granting and restricting access to your resources within the Azure environment is a critical operational process. The use of authentication and authorization will affect who has access to resources and how they access them. Identity and access management are different than the network security groups and application security groups discussed in the next chapter.

Microsoft uses Microsoft Active Directory for identity and access management within Azure. This makes adapting to the identity management on Azure much easier for those familiar with the concept of Active Directory on-premises.

Azure AD domain services and on-premises Active Directory are different technical platforms, despite having shared technology roots. Certain features are not available in Azure AD and some features require more effort to design and maintain using Active Directory.

For more information on the key differences and features, go to *http://bit.ly/2MyDiv2*.

Deploying Active Directory on Microsoft Azure

Active Directory is a multitenant, geographically distributed directory services platform that debuted with Microsoft Windows 2000 Server edition to authenticate and authorize services such as users, computers, file shares, and more.

Azure AD comes in multiple tiers of service depending on the features needed to support your Azure or other Microsoft Active Directory–integrated services. Every version of Azure AD is deployed for resiliency and availability and is accessible in every region.

Azure AD Tiers

There are four tiers for Azure AD, with different features and prices. As with many Azure services, you can use Enterprise and Open licenses to extend to your Azure accounts.

Free and Basic

The free edition is a lightweight directory to provide access control for cloud-only or cloud-first organizations on Azure. There is no SLA for the Azure AD Free edition; however, the service provides the needed functionality and availability for many individuals or organizations getting started with the basics of Azure.

The basic tier introduces group-based access management, branded login options, self-service password management, and the Application Proxy feature. Organizations usually begin with the Basic tier primarily to allow for group permissions and ease-of-use with self-service password resets.

The SLA for Basic and higher editions is 99.9% and is the first of three tiers that requires a pay-per-user-per-month model.

Premium P1 and Premium P2

Premium P1 adds advanced group features, multifactor authentication, third-party integration support, and much more. Premium tiers also introduce mobile device management options. This tier is required for Azure AD Connect. Single Sign-On is limited to 10 applications per user.

Premium P2 includes identity protection and very detailed privileged identity management as well as access reviews, which might be needed with more advanced or larger implementations of Azure AD.

Federating to an Existing Active Directory Environment

You can integrate Azure AD into your existing Microsoft Active Directory environment by using Azure AD Connect. This allows you to use on-premises Active Directory credentials to authenticate and authorize access to Azure resources.

Bidirectional synchronization ensures up-to-date information at all times in both the on-premises and the Azure AD environments. This is similar to the way that cross-forest trusts work between disparate Active Directory environments.

It is important for you to understand the limitations and supported topologies for deploying Azure AD Connect. For the full details and latest information about the supported and unsupported topologies, go to the Microsoft Azure website.

As noted earlier, you must be running Premium P1 or Premium P2 Azure AD to enable federation to an on-premises Active Directory.

With identity and access management covered, let's move on to networking and security on the Azure environment to see how specific object access is managed.

CHAPTER 5

Networking and Security on Microsoft Azure

In this chapter, let's explore how networking on the Microsoft Azure public cloud platform enables connectivity and security throughout the variety of services and across all regions and Availability Zones. This includes the products and methods to secure your services on Azure and the ability to access Azure resources in Open Systems Interconnection (OSI) Layers 3 through 7.

Ensuring network and application access groups for your resources is particularly important in order to maintain infrastructure and application protection. The networking and security features discussed in this chapter are available throughout the entire Azure infrastructure, which ensures consistency and a simplified approach to defining your Azure deployment structure.

Core Networking and Security on Azure

The core features in the Azure networking environment we cover here include the following:

Virtual networks
: A virtual private cloud within the Azure cloud environment that is given private subnets and external access to other networks (including internet) using a gateway

Application security groups (ASGs)
: Role-based access control (RBAC) to allow granular access to applications or groups of applications

Network security groups (NSGs)
: Network-layer firewall to filter inbound and outbound traffic by network, port, and protocol

These three features come together to make up the isolated and highly secure environment for your virtual cloud within Azure.

Each resource group may have a shared set of ASGs and NSGs, but each VM resource can have only one NSG or ASG applied to it. There are many choices to make when creating and maintaining these security groups, so administrators are encouraged to work with their security and networking teams to ensure that consistency and secure practices are used at all times.

Network Security Group Basics

The Azure VM creation process includes steps to create a new or attach an existing NSG. Figure 5-1 illustrates this process.

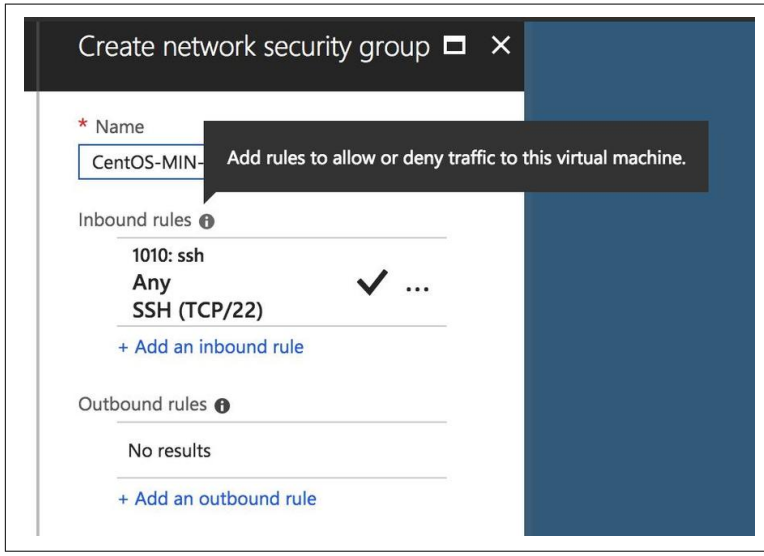


Figure 5-1. NSG port configuration

Selecting inbound rules will bring up a policy form that defines your flow rule as source, source port range, destination, destination port range, and protocol, as shown in Figure 5-2.

Each flow rule is also chosen as either Allow or Deny and given a Priority. Multiple rules are processed in order, based on reverse numbering. Lower numbered rules are processed last, which also means you should provide gaps between rule numbers (e.g., 400, 300, and 200) in case there is a need to apply additional rules between existing ones.

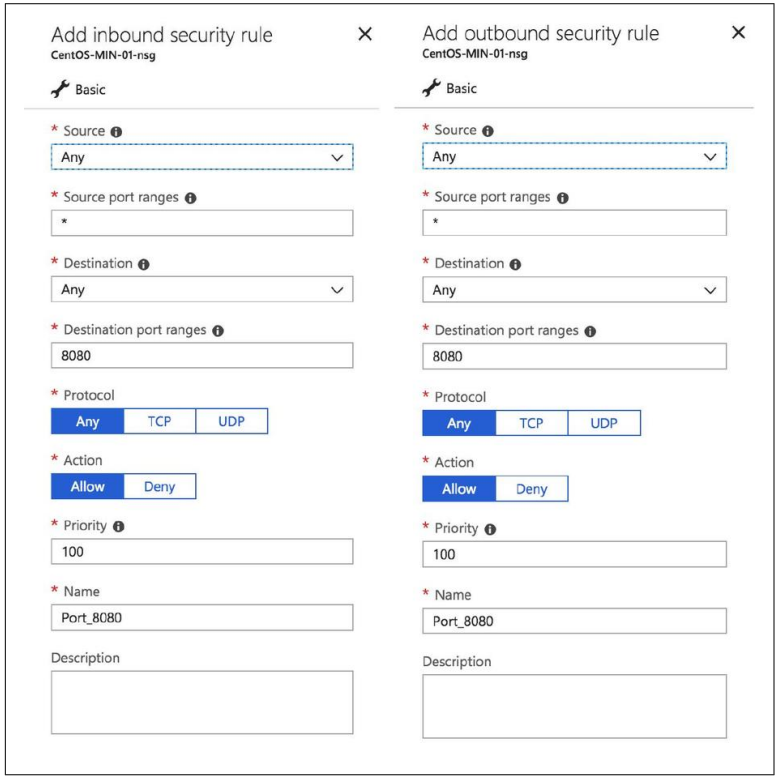


Figure 5-2. Creating an inbound and outbound security rule

The same process applies for outbound flow rules with the same criteria needed to enforce the policy.

Flow and filtering of traffic is stateful, so when an inbound or an outbound rule is created, there is no need to create the alternate rule in the other direction. When a rule for inbound port 80 (HTTP) is

created, for example, the outbound traffic on the same port/protocol is automatically allowed.

NSG flow rules are enforced immediately when they are created or modified. There is no need to restart the VM or perform any other steps to enact the new policies.

Azure ExpressRoute Networking

Organizations that are committed to a hybrid cloud model will often add Azure ExpressRoute networking as a high-speed and low-latency connection to their on-premises infrastructure. Using ExpressRoute, an SLA-bound, resilient, and redundant connection is made directly to the Azure service. Direct connections are delivered to the customer premises by a service provider or partner.

Features and advantages of ExpressRoute include the following:

Layer 3 (L3) connectivity
: Full IP-routed network access to your geographic Azure region

Direct route access to additional Azure infrastructure
: Additional Azure region access available as an add-on service

Redundant connectivity
: Partner network delivers two independent connections

BGP dynamic routing
: Connectivity SLA is dependent on creating two BGP connections to each of the Microsoft Enterprise Edge routers

Additional service access
: ExpressRoute also gives direct, low-latency access to Office 365 and Dynamics 365 services

Continuous and secure L3 connectivity enables greater flexibility for services, data, and applications to reside either on-premises or in the Azure cloud.

Alternative methods to extend your network into the Azure cloud are available using third-party virtual private network appliances and services. These alternate methods will not be given an SLA and must be designed and deployed to ensure continuous, redundant connectivity. ExpressRoute is also preferred because of the additional advantage of lower-latency access to the Azure network using physical fiber connectivity.

Design Patterns for IaaS Networking and Security Services

Certain networking and security practices should be included in your day-to-day operations on your Azure environment, including the following:

Auditing your Azure resources
: Audit practices must be extended to include all Azure resources, which may require some adaptation.

Using RBAC and advanced options
: Using Premium P1/P2 Azure AD ensures more granularity with granting access and logging resource usage.

Logging all activity
: Ensure that existing logging and monitoring solutions are actively managing your Azure resources and infrastructure.

Extending the feedback loop to application, network, and security teams
: Your security and networking teams must be involved in continuous management and administration of Azure resources to maintain consistency and security in the hybrid environment.

Extending IDS/IPS to the Azure platform
: Investigate all options for extending your other security processes into the Azure environment.

Security on Microsoft Azure is among the strongest of any IT environment, and includes advanced certifications for regulatory and governmental agencies, which are continuously audited, updated, and maintained. Microsoft has a vested interest in the security of the underlying infrastructure on Azure; however, within your Azure subscription the responsibility falls clearly within your IT organization.

Ensuring that security practices are extended to the Azure cloud must be a continuous process.

Next Steps in Your Azure Journey

This guide has been created to give some specific examples of core IaaS use on Microsoft Azure and an overview compute, storage, and

networking for Azure IaaS features. This is only the beginning of your journey to learning Azure. The next steps are to define what your use-cases and goals are for Azure for both personal and work purposes.

If would like to access additional learning tools, resources, detailed code examples, and technical certification, visit *https://discopos.se/DeployingAzureSolutions*.

About the Author

Eric Wright is a technology evangelist at Turbonomic, a blogger at *DiscoPosse.com*, and he runs the GC On-Demand podcast. With a long history in the industry as a systems architect and technologist, Eric is also deeply involved in technology communities including Microsoft, VMware, OpenStack, Kubernetes, DevOps, and many others. Eric is also the cofounder of Virtual Design Master and RapidMatter, both of which were founded on the power of people and community in technology.