Credits

Date	Contributor	Company
2019-05-01	Matthias Gessenay	Corporate Software
2019-05-12	Stefan Beckmann	Unico Data AG
2019-05-27	Robin Berberat	Corporate Software
2019-07-31	Stefan Beckmann	Unico Data AG
2019-08-06	Stefan Beckmann	Unico Data AG
2019-08-19	Stefan Beckmann	Unico Data AG
Row7		
Row8		
Row9		
Row10		

Introduction to this concept

The following document is a recommended concept in order to set up an Azure Environment. As many aspects need to be kept in mind, the concept is about the whole Azure Environment and how it needs to be built, so that future Azure projects/builds doesn't face foreseeable restrictions.

Azure has grown rapidly since its introduction in 2008. This growth required Microsoft engineering teams to rethink their approach for managing and deploying services. The Azure Resource Manager model was introduced in 2014 and replaces the classic deployment model. Resource Manager enables organizations to more easily deploy, organize, and control Azure resources. Resource Manager includes parallelization when creating resources for faster deployment of complex, interdependent solutions. It also includes granular access control, and the ability to tag resources with metadata. We recommend that all resources are created through the Resource Manager model.

Good Practice: General recommendations for moving to the cloud

Recommendations for Azure enterprise administration	
Limit the number of administrative users	Assign a minimum number of users as Subscription Administrators and/or Co-administrators.

Recommendations for Azure enterprise administration	
Use Role-Based Access	Use Azure Resource Management RBAC whenever possible to control the amount of access that administrators have, and log what changes are made to the environment.

Recommendations for Azure enterprise administration	
Use work accounts	You should sign up for Azure as an organization and use a work or school account to manage resources in Azure. Do not allow the use of existing personal Microsoft Accounts.

Recommendations for Azure enterprise administration	
Define naming conventions	Assign meaningful names to your Azure subscriptions according to defined naming conventions.

Recommendations for Azure enterprise administration	
Use management groups to group your subscriptions	Create your own "root" Management group. Don't modify the built-in management group. Create a structure that reflects your organization, in the view of departments, services and release cycles.

Recommendations for Azure enterprise administration	
Use a managment subscription	Begin by creating a Management Subscription from which you want to perform the automation. Also included in this subscription are services that are to be disconnected from the hub but still centralized, e.g. Workspaces for Log Analytics or Storage

Recommendations for Azure enterprise administration	
Use Tier 0 subscription (Hub)	Use Tier 0 subscription to host shared resources, such as domain controllers and other sensitive roles, and limit the privileges to access it.
Use Tier 1 subscriptions (Spoke)	Use Tier 1 subscription for the rest of the subscriptions.

Recommendations for Azure enterprise administration	
Use separate automation subscription	To automate the initial creation of Azure resources, you'll need to solve the henn and egg problem. For that, we refer to a separate automation subscription. There can be reasons to use a resource group instead.

Recommendations for Azure enterprise administration	
Use project subscriptions	Use decentralized project subscriptions. Delegate management of those subscriptions to the responsible project teams.

Recommendations for Azure enterprise administration	
Separate production from QA	Separate QA environments into distinct subscriptions to allow separation of access and to allow the QA subscription to scale on its own without impacting production.

Create Default RBAC groups in AAD	Create AAD Groups for each Management Group, Subscription and Resource Group one for Read Only, Contributor and Owner. That helps to start quickly with the most rights to delegate and does prevent to do that later much more complex. And you can configure your rolls

Governance

In real life, scaffolding is used to create the basis of the structure. The scaffold guides the general outline and provides anchor points for more permanent systems to be mounted. An enterprise scaffold is the same: a set of flexible controls and Azure capabilities that provide structure to the environment, and anchors for services built on the public cloud. It provides the builders (IT and business groups) a foundation to create and attach new services.

The following image describes the components of the scaffold. The foundation relies on a solid plan for departments, accounts, and subscriptions. The pillars consist of Resource Manager policies and strong naming standards. The rest of the scaffold comes from core Azure capabilities and features that enable a secure and manageable environment. The enterprise scaffold is explicitly designed for the Resource Manager model.

Hierarchy for Enterprise enrollments

The foundation of the scaffold is the Azure Enterprise Enrollment (and the Enterprise Portal). The enterprise enrollment defines the shape and use of Azure services within a company and is the core governance structure. Within the enterprise agreement, customers can further subdivide the environment into departments, accounts and finally, subscriptions. An Azure subscription is the basic unit where all resources are contained. It also defines several limits within Azure, such as number of cores, resources, etc.

Every enterprise is different and the hierarchy in the previous image allows for significant flexibility in how Azure is organized within the company. Before implementing the guidance contained in this document, the hierarchy should be modelled and the impact on billing, resource access, and complexity understood.

Patterns

The three common patterns for Azure enrolments are:

- Functional pattern
- Business unit pattern
- Geographic pattern

The following roles will be defined:

EA = Enterprise Admin, DA = Department Admin, AO = Account Owner, SA = Service Admin

First, the administrators are defined in the portal (ea.azure.com). This could be either a Microsoft or a work or school account. We recommend using a Work-Account.

Secondly, the departments are created. A department can't live without DA. If there is no DA specified, the EA will be the DA of the department:

Once the department is created, accounts can be created on the department. Each department needs to have account owners, those are the only one who can create subscriptions on the account. If a EA wants to create subscriptions on an account, he needs to add himself as the account owner.

At the account level, each account can be named as Dev/Test and therefore be treated separately:

When the account is created, a subscription can be assigned:

This approach is resumed in the CSP Model – whereas the customer doesn't have an Enterprise portal, but multiple subscriptions that are bound to his Active Directory.

Source: https://docs.microsoft.com/en-us/azure/security/governance-in-azure Source: https://www.credera.com/blog/credera-site/azure-governance-part-2-using-subscriptions-resource-groups-building-blocks/

CSP Hierarchy

If Azure Services are provided by a Cloud Service Provider, all the administrative elements above the subscription are eliminated. Within the CSP, the different defined subscriptions are bound directly to an Azure Active Directory (Azure AD).

CSP provided Azure subscriptions can only be managed over the ARM (Azure Resource Manager) Portal. There are no service or co-administrators. To control access, Azure provides roles (see chapter 16.3 Role based access control (RBAC)).

The CSP licence provider is always the owner over all of the ordered subscriptions and defines who else is owner of a subscription. This definition is modelled on the Azure AD tenant of the CSP.

The rights on the Subscription come over a Foreign Principal from the CSP Tenant. You can only choose between the following roles: - Admin Agent - Helpdesk Agent - Sales Agent

The problem is, that this authorization will be applied overall tenants at the same time. And currently, it is not possible to use Privileged Identity Management for clients, from CSP Tenant.

Source: https://docs.microsoft.com/en-us/azure/cloud-solution-provider/customer-management/administration-delegation

Lighthouse

Azure Lighthouse offers service providers a single control plane to view and manage Azure across all their customers with higher automation, scale, and enhanced governance. With Azure Lighthouse, service providers can deliver managed services using comprehensive and robust management tooling built into the Azure platform. This offering can also benefit enterprise IT organizations managing resources across multiple tenants.

This helps to fill the gap from the CSP right delegation. With Azure Lighthouse it's possible to delegate a lot of services, granularly per customer. The following are to the beginning supported services: - Azure Automation - Azure Backup - Azure Monitor - Azure Policy - Azure Resource Graph - Azure Security Center - Azure Service Health - Azure Site Recovery - Azure Virtual Machines - Azure Virtual Network

Under the following link you can find the current list: https://docs.microsoft.com/en-us/azure/lighthouse/concepts/cross-tenant-management-experience#supported-services-and-scenarios

Source: https://docs.microsoft.com/en-us/azure/lighthouse/overview Source: https://docs.microsoft.com/en-us/azure/lighthouse/concepts/cross-tenant-management-experience

Resource Locks

As organizations add core services to the subscription, it becomes increasingly important to ensure that those services are available to avoid business disruption. Resource locks enable to restrict operations on high-value resources where modifying or deleting them would have a significant impact on your applications or cloud infrastructure. You can apply locks on a subscription-, resource group-, or resource-level. Typically, you apply locks to foundational resources such as **virtual networks, gateways, and storage accounts.**

Resource locks currently support two values: **CanNotDelete** and **ReadOnly**. **CanNotDelete** means that users (with the appropriate rights) can still read or modify a resource but cannot delete it. **ReadOnly** means that authorized users can't delete or modify a resource.

To create or delete management locks, you must have access to Microsoft.Authorization/* or Microsoft.Authorization/locks/* actions. Of the built-in roles, only Owner and User Access Administrator are granted those actions.

We recommend to protect core network options with locks. Accidental deletion of a gateway, site-to-site VPN would be disastrous to an Azure subscription. Azure doesn't allow you to delete a virtual network that is in use, but applying more restrictions is a helpful precaution.

- Virtual Network: CanNotDelete
- Network Security Group: CanNotDelete
- Policies: CanNotDelete

Policies are also crucial to the maintenance of appropriate controls. We recommend that you apply a CanNotDelete lock to polices that are in use. Policies can be set via Azure Blueprint.

Source: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-subscription-governance#azure-resource-locks

Azure Policy

IT governance creates clarity between business goals and IT projects. Good IT governance involves planning your initiatives and setting priorities on a strategic level. Does your company experience a significant number of IT issues that never seem to get resolved? Implementing policies helps you better manage and prevent them. Implementing policies is where Azure Policy comes in.

Azure Policy is a service in Azure that you use to create, assign and, manage policy definitions. Policy definitions enforce different rules and actions over your resources, so those resources stay compliant with your corporate standards and service level agreements. Azure Policy runs an evaluation of your resources, scanning for those not compliant with the policy definitions you have. For example, you can have a policy to allow only certain type of virtual machines. **Another requires that all resources have a particular tag**. These policies are then evaluated when creating and updating resources.

Source: https://docs.microsoft.com/en-au/azure/governance/policy/overview

How is Azure Policy different from RBAC?

There are a few key differences between policy and role-based access control (RBAC). RBAC focuses on user actions at different scopes. For example, you might be added to the contributor role for a resource group at the desired scope. The role allows you to make changes to that resource group. Policy focuses on resource properties during deployment and for already existing resources. For example, through policies, you can control the types of resources that can be provisioned. Or, you can restrict the locations in which the resources can be provisioned. Unlike RBAC, policy is a default allow and explicit deny system.

To use policies, you must be authenticated through RBAC. Specifically, your account needs the:

- 'Microsoft.Authorization/policydefinitions/write' permission to define a policy.
- 'Microsoft.Authorization/policyassignments/write' permission to assign a policy.
- 'Microsoft.Authorization/policySetDefinitions/write' permission to define an initiative.
- 'Microsoft.Authorization/policyassignments/write' permission to assign an initiative.

These permissions are not included in the Contributor role.

Source: https://docs.microsoft.com/en-au/azure/governance/policy/overview#how-is-it-different-from-rbac

Policy Definition

Every policy definition has conditions under which it is enforced. Additionally, it has an accompanying action that takes place if the conditions are met.

Azure Policy offers some built-in policies that are available to you by default. For example:

- Require SQL Server 12.0 This policy definition has conditions/rules to ensure that all SQL servers use version 12.0. Its action is to deny all servers that do not meet these criteria.
- Allowed Storage Account SKUs This policy definition has a set of conditions/rules that determine if a storage account that is being deployed is within a set of SKU sizes. Its action is to deny all servers that do not adhere to the set of defined SKU sizes.

- Allowed Resource Type This policy definition has a set of conditions/rules to specify the resource types that your organization can deploy. Its action is to deny all resources that are not part of this defined list.
- Allowed Locations This policy enables you to restrict the locations that your organization can specify when deploying resources. Its action is used to enforce your geo-compliance requirements.
- Allowed Virtual Machine SKUs This policy enables you to specify a set of virtual machine SKUs that your organization can deploy.
- Apply tag and its default value This policy applies a required tag and its default value, if it is not specified by the user.
- Enforce tag and its value This policy enforces a required tag and its value to a resource.
- Not allowed resource types This policy enables you to specify the resource types that your organization cannot deploy.

You can assign any of these policies through the Azure portal, PowerShell, or Azure CLI.

Source: https://docs.microsoft.com/en-au/azure/governance/policy/overview#policy-definition

Naming standards

Introduction

The first pillar of the scaffold is naming standards. Well-designed naming standards enable to identify resources in the portal, on a bill, and within scripts. Most likely, there are already naming standards for on-premises infrastructure. When adding Azure to your environment, those naming standards should be extended to your Azure resources. Naming standard facilitate more efficient management of the environment at all levels.

The choice of a name for any resource in Microsoft Azure is important because:

- It is difficult to change a name later.
- Names must meet the requirements of their specific resource type.

Consistent naming conventions make resources easier to locate. They can also indicate the role of a resource in a solution.

The key to success with naming conventions is establishing and following them across your applications and organizations.

Technical Background

Sources: https://docs.microsoft.com/en-us/azure/architecture/best-practices/naming-conventions, https://blogs.technet.microsoft.com/dsilva/2017/11/10/azure-subscription-governance-resource-group-and-naming-convention-strategies/

Subscriptions

When naming Azure subscriptions, verbose names make understanding the context and purpose of each subscription clear. When working in an environment with many subscriptions, following a shared naming convention can improve clarity.

A generic recommended pattern for naming subscriptions is:

<Company> <Department (optional)> <Product Line (optional)> <Environment>

- Company would usually be the same for each subscription. However, some companies may have child companies within the organizational structure. These companies may be managed by a central IT group. In these cases, they could be differentiated by having both the parent company name and child company name.

- Department is a name within the organization that contains a group of individuals. This item within the namespace is optional. For example: “IT”, “Marketing”, . . .
- Product line is a specific name for a product or function that is performed from within the department. This is generally optional for internal-facing services and applications. However, it is highly recommended to use for public-facing services that require easy separation and identification (such as for clear separation of billing records).
- Environment is the name that describes the deployment lifecycle of the applications or services, such as Dev, Test, or Prod.

Rules and restrictions

Each resource or service type in Azure enforces a set of naming restrictions and scope; any naming convention or pattern must adhere to the requisite naming rules and scope. For example, while the name of a VM maps to a DNS name (and is thus required to be unique across all of Azure), the name of a VNET is scoped to the Resource Group that it is created within.

In general, avoid having any special characters (- or _) as the first or last character in any name. These characters will cause most validation rules to fail.

General naming restrictions

Entity	Scope	Length	Case sensitive	Valid Characters	Suggested Pattern	Example
Resource Group	Subscription	1-90	false	Alphanumeric, underscore, parentheses, hyphen, and period (except at end)	<service short name>-<environment>-rg	profx-prod-rg
Availability Set	Resource Group	1-80	false	Alphanumeric, underscore, and hyphen	<service-short-name>-<context>-as	profx-sql-as
Tag	Associated Entity	512 (name), 256 (value)	false	Alphanumeric	“key” : “value”	“department” : “Central IT”

Compute naming restrictions

Entity	Scope	Length	Case sensitive	Valid Characters	Suggested Pattern	Example
Virtual Machine	Resource Group	**1-15 (Windows), 1-64 (Linux)**	false	Alphanumeric and hyphen	<name>-<role>-vm<number>	profx-sql-vm1
Function App	Global	1-60	false	Alphanumeric and hyphen	<name>-func	calcprofit-func

For all rules and restrictions, please visit https://docs.microsoft.com/en-us/azure/architecture/best-practices/naming-conventions#naming-rules-and-restrictions.

Affixes

When developing a naming convention for a company or project, it is important to select a common set of affixes and their position (suffix or prefix).

While all the information about type, metadata and context is available via API, applying common affixes simplifies visual identification. When incorporating affixes into your naming convention, it is important to clearly specify whether the affix is at the beginning of the name (prefix) or at the end (suffix).

For instance, here are two possible names for a service hosting a calculation engine:

- SvcCalculationEngine (prefix)
- CalculationEngineSvc (suffix)

Affixes can refer to different aspects that describe the particular resources. See in the examples under chapter application.

Good Practices: Naming standards

Affixes

Region

Region	Location	Code
Region Neutral	Location Neutral	**AAAA**
South Africa North	Johannesburg	SANO
South Africa West	Cape Town	SAWE
Central India	Pune	INCE
China East	Shanghai	CHEA
China East 2	Shanghai	CHE2
China North	Beijing	CHNO
China North 2	Beijing	CHN2
East Asia	Hong Kong	ASEA
Japan East	Tokyo, Saitama	JAEA
Japan West	Osaka	JAWE
Korea Central	Seoul	KOCE
Korea South	Busan	KOSO
South India	Chennai	INSO
Southeast Asia	Singapore	ASSO
UAE Central	Abu Dhabi	UACE
UAE North	Dubai	UANO
West India	Mumbai	INWE
Australia Central	Canberra	AUCE
Australia Central 2	Canberra	AUC2
Australia East	New South Wales	AUEA
Australia Southeast	Victoria	AUSO
France Central	Paris	FRCE
France South	Marseille	FRSO
Germany Central	Frankfurt	GECE
Germany North	Germany North	GENO
Germany Northeast	Magdeburg	GENE
Germany West Central	Germany West Central	GEWC
North Europe	Ireland	EUNO
Norway East	Norway	NOEA
Norway West	Norway	NOWE
Switzerland North	Zurich	SCNO
Switzerland West	Geneva	SCWE
UK South	London	UKSO
UK West	Cardiff	UKWE
West Europe	Netherlands	**EUWE**

Region	Location	Code
Canada Central	Toronto	CACE
Canada East	Quebec City	CAEA
Central US	Iowa	USCE
East US	Virginia	USEA
East US 2	Virginia	USE2
North Central US	Illinois	USNC
South Central US	Texas	USSC
US DoD Central	Iowa	USGC
US DoD East	Virginia	USGE
US Gov Arizona	Arizona	USGA
US Gov Iowa	Iowa	USGI
US Gov Texas	Texas	USGT
US Gov Virginia	Virginia	USGV
West Central US	Wyoming	UWCE
West US	California	USW2
West US 2	Washington	USWE
Brazil South	Sao Paulo State	BRSO
Azure Stack	Datacenter	AZBE

Environment

Code	Description
DE	Development
TE	Test
ST	Staging (UAT)
PR	Production
CO	Core
AU	Automation
SB	Sandbox
SP	Special
UN	Undefined

Services

Name	Category	Prefix	Suffix
App Service	App Services	APPS	
App Service Environment	App Services	APSE	
App Service Plan	App Services	ASPL	
Application Insights	App Services	AINS	
Application Security Group		APSG	
Automation Account	Serverless	AUTO	
Availability Set	Compute	AVSE	
Azure Analysis Services	Databases	AASE	
Azure Application Gateway	Security	AAGA	
Azure Automation Hybrid Worker	Hybrid		
Azure Traffic Manager Profile	Networking	ATMP	
Blob	Storage		
Blueprints	Governance	BLPR	
Container	Serverless		
Data Lake Store	Storage		

Name	Category	Prefix	Suffix
Event Grid Domains	Event Hub	EGDO	
Event Grid Subscriptions	Event Hub	EGSU	
Event Hubs	Event Hub	EVHU	
Event Hubs Topics	Event Hub	EHTO	
External Load Balancer	Compute	LBEX	
File	Storage		
Function	Serverless		
Initative	Governance		
Internal Load Balancer	Networking	LBIN	
Key Vault	Other	KEYV	
Load Balancer Networking	Networking	LLBN	
Local Network Gateway	Networking	LNGA	
Log Analytics Workspace	Monitoring	LAWS	
Managed Disk Storage	Storage		
Management Group	Governance	MAGR	
Network Interface	Networking		NIC
Network Security Group	Networking	NSGR	
Network Security Group Rule	Networking		
Policies	Governance		
Public IP Address	Networking		PIP
Public IP Address Networking	Networking	PUBN	
Queue	Serverless		
Recovery Service Vault Storage	Backup	RSVS	
Recovery Services Vault	Backup	RSVA	
Recovery Services Vault – Azure Backup Policy	Backup	ABPO	
Ressource Group	Governance	RSGR	
Route Table	Networking	NRTA	
SQL Database	Database	SQDB	
SQL Datawarehouse	Database	SQDB	
SQL Managed Instance	Database	SQMI	
SQL Server	Database	SQSR	
Storage Account	Storage		
Storage Account Name (data)	Storage		
Storage Account Name (disk)	Storage		
Subnet	Networking	SNET	
Subscription	Governance	SUBS	
Table	Databases		
Tag	Governance		
Virtual Machine	Compute		
Virtual Network (VNet)	Networking	VNET	
VNet Peering	Networking	VNPE	
VPN Gateway	Networking	VPNW	

Naming Conventions

If this naming convention used only for a single-tenant, you can omit the **TenantShort** term. But if you are an MSP/CSP or uses services from it, it recommended that you use this in your notation.

Management Group *Corp Pattern*: `<Prefix>_<CORP|TenantShort>_<Level>`
Corp ID Pattern: `<Prefix>_<ManagementGroupID>_<Level>`

Name Pattern: `<Prefix>_[TenantShort]_<Scope>_<Level>`

ID Pattern: `<Prefix>_<ManagementGroupID>_<Level>`

Examples:

ID	Name
MAG_0001_00	MAG_CORP_00
MAG_0002_01	MAG_Infra_01
MAG_0003_01	MAG_Standard_01
MAG_0004_01	MAG_Special_01
MAG_0005_02	MAG_SuplierA_02
MAG_0006_02	MAG_SuplierB_02

ID	Name
MAG_0007_00	MAG_MYTC_00
MAG_0008_01	MAG_MYTC_Infra_01
MAG_0009_01	MAG_MYTC_Standard_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	MAG = Management Group	
ManagementGroupID	4	Ongoing numbering	
TenantShort	4	MYTC = My Top Company	
Scope	5..30	Infra Standard Special Others	
Level	2	00 = Top Level 01 = Level under Top Level 02 = Level under Level 01	

Subscription *Pattern*: `<Prefix>_[TenantShort]_<Environment>_<SubscriptionID>_<Product|Service|Team>_<Vers`

Examples:

SUB_AU_0001_CentralAutomation_01
SUB_CO_0001_CentralServices_01
SUB_SB_0001_CentralSandbox_01
SUB_PR_1001_BusinesServices_01

SUB_MYTC_AU_0001_CentralAutomation_01
SUB_MYTC_CO_0001_CentralServices_01
SUB_MYTC_SB_0001_CentralSandbox_01
SUB_MYTC_PR_1001_BusinesServices_01
SUB_MYTC_TE_1002_BusinesServices_01
SUB_MYTC_DE_1003_BusinesServices_01
SUB_MYTC_PR_1004_VDIServices_01
SUB_MYTC_SP_2001_ExternalCorpA_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	SUB = Subscription	
TenantShort	4	MYTC = My Top Company	
Environment	2	Described in the chapter Affixes, Environment	

Identifiers	Range	Values/Meaning	Comments
SubscriptionID	4	Ongoing numbering per environment, the first position of the number stands for: 0 = Infrastructure 1 = Standard 2 = Special.	
Product|Service|Team	5..20	CentralAutomation CentralServices BusinesServices VDIServices ExternalCorpA	
VersionNr	2	01..99	

Tag

Blueprints *Pattern*: `<Prefix>_<Description>_<VersionNr>`

Examples: BLP_Automation_01 BLP_Backup_01 BLP_BasicConfig_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	BLP = Blueprints	
Description	4	A description that best describes the purpose or content.	
VersionNr	2	01..99	

Ressource Group *Pattern*: `<Prefix>_[TenantShort]_<Environment>_<Region>_<Service|System>_<VersionNr>`

Examples:

RSG_AU_EUWE_Automation_01
RSG_CO_AAAA_Core_01
RSG_CO_AAAA_Network_01
RSG_PR_AAAA_Network_01
RSG_PR_AAAA_Security_01
RSG_PR_AAAA_Storage_01

RSG_MYTC_AU_EUWE_Automation_01
RSG_MYTC_CO_AAAA_Core_01
RSG_MYTC_CO_AAAA_Network_01
RSG_MYTC_CO_AAAA_Security_01
RSG_MYTC_CO_AAAA_Storage_01
RSG_MYTC_CO_AAAA_Backup_01 RSG_MYTC_CO_EUWE_DomainServices_01 RSG_MYTC_PR_AAAA_Networ
RSG_MYTC_PR_AAAA_Security_01
RSG_MYTC_PR_AAAA_Storage_01
RSG_MYTC_PR_EUWE_ApplicationA_01
RSG_MYTC_TE_AAAA_Network_01
RSG_MYTC_TE_AAAA_Security_01
RSG_MYTC_TE_AAAA_Storage_01
RSG_MYTC_TE_EUWE_ApplicationA_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	RSG = Ressource Group	
TenantShort	4	MYTC = My Top Company	
Environment	2	Described in the chapter Affixes, Environment	
Region	4	Described in the chapter Affixes, Region	
Service|System	5..25	Describes a purpose for which the resource should be used.	

Identifiers	Range	Values/Meaning	Comments
VersionNr	2	01..99	

Declaration:

Resources that are managed from the same team, and where all resources planned to be member of the same resource group, are the best examples for the AAAA Region code.

Virtual Network (VNet) *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>_<Version`

Examples:

VNE_EUWE_CO_0001_01
VNE_EUWE_PR_1001_01
VNE_EUWE_TE_1002_01
VNE_EUWE_DE_1003_01

VNE_MYTC_EUWE_CO_0001_01
VNE_MYTC_EUWE_PR_1001_01
VNE_MYTC_EUWE_TE_1002_01
VNE_MYTC_EUWE_DE_1003_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	VNE = Virtual Network	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID in which subscription the resource will be published.	
VersionNr	2	01..99	

VNet Peering *Pattern*: `<Prefix>_[SourceTenantShort]_<SourceRegion>_<SourceEnvironment>_<SourceSubscripti`

Examples:

VNP_EUWE_CO_0001_01-EUWE_PR_1001_01
VNP_EUWE_PR_1001_01-EUWE_CO_0001_01
VNP_EUWE_CO_0001_01-EUWE_TE_1002_01
VNP_EUWE_TE_1002_01-EUWE_CO_0001_01
VNP_EUWE_CO_0001_01-EUWE_DE_1003_01
VNP_EUWE_DE_1003_01-EUWE_CO_0001_01

VNP_MYTC_EUWE_CO_0001_01-MYTC_EUWE_PR_1001_01
VNP_MYTC_EUWE_PR_1001_01-MYTC_EUWE_CO_0001_01
VNP_MYTC_EUWE_CO_0001_01-MYTC_EUWE_TE_1002_01
VNP_MYTC_EUWE_TE_1002_01-MYTC_EUWE_CO_0001_01
VNP_MYTC_EUWE_CO_0001_01-MYTC_EUWE_DE_1003_01
VNP_MYTC_EUWE_DE_1003_01-MYTC_EUWE_CO_0001_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	VNP = VNet Peering	

Identifiers	Range	Values/Meaning	Comments
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID which is also used for the corresponding VNet.	
VersionNr	2	01..99	

Subnet *Pattern*: `<Prefix>_<Region>_<Environment>_<SubscriptionID>_[CustomerShort]_<Service|System>_<Area`

Examples:

SNE_EUWE_CO_0001_Frontend_FE
SNE_EUWE_CO_0001_Backend_BE
SNE_EUWE_CO_0001_Management_MG
SNE_EUWE_CO_0001_DomainServices_FE
SNE_EUWE_PR_1001_Frontend_FE
SNE_EUWE_PR_1001_Backend_BE
SNE_EUWE_PR_1001_Management_MG
SNE_MYTC_EUWE_PR_1001_DMZ_FE
SNE_MYTC_EUWE_PR_1001_DMZ_BE
SNE_MYTC_EUWE_PR_1001_AppServer_BE

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	VNP = VNet Peering	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID which is also used for the corresponding VNet.	
Service|System	5..25	Describes a purpose for which the resource should be used.	
AreaShort	2	FE = Frontend BE = Backend MG = Management	

Route Table *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>_<VersionNr>`

Examples:

NRT_EUWE_CO_0001_01
NRT_EUWE_PR_1001_01
NRT_EUWE_TE_1002_01
NRT_EUWE_DE_1003_01

NRT_MYTC_EUWE_CO_0001_01
NRT_MYTC_EUWE_PR_1001_01
NRT_MYTC_EUWE_TE_1002_01
NRT_MYTC_EUWE_DE_1003_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	VNE = Virtual Network	
TenantShort	4	MYTC = My Top Company	

Identifiers	Range	Values/Meaning	Comments
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID which is also used for the corresponding VNet.	
VersionNr	2	01..99	

Network Security Group *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>_<Service`

Examples:

NSG_EUWE_CO_0001_Frontend_FE
NSG_EUWE_CO_0001_Backend_BE
NSG_EUWE_CO_0001_Management_MG
NSG_EUWE_CO_0001_DomainServices_FE
NSG_EUWE_PR_1001_Frontend_FE
NSG_EUWE_PR_1001_Backend_BE
NSG_EUWE_PR_1001_Management_MG

NSG_MYTC_EUWE_CO_0001_Frontend_FE
NSG_MYTC_EUWE_CO_0001_Backend_BE
NSG_MYTC_EUWE_CO_0001_Management_MG
NSG_MYTC_EUWE_CO_0001_DomainServices_FE
NSG_MYTC_EUWE_PR_1001_Frontend_FE
NSG_MYTC_EUWE_PR_1001_Backend_BE
NSG_MYTC_EUWE_PR_1001_Management_MG
NSG_MYTC_EUWE_PR_1001_CSTA_DMZ_FE
NSG_MYTC_EUWE_PR_1001_CSTA_DMZ_BE
NSG_MYTC_EUWE_PR_1001_CSTA_AppServer_BE
NSG_MYTC_EUWE_PR_1001_CSTB_AppServer_BE

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	NSG = Network Security Group	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID which is also used for the corresponding VNet.	
Service|System	5..25	Describes a purpose for which the resource should be used.	
AreaShort	2	FE = Frontend BE = Backend MG = Management	

Declaration:

Network Security Groups inherit the name of the Subnet, they are not using a counter as there can't be multiple NSG with the same name.

Network Security Group Rule *Pattern*: `<Prefix>_<Direction>_<Protocol>_<Action>_<FromToWhere|Service|Syst`

Examples:

Inbound:
NSR_in_TCP_allow_JUMPtoVNET-RDP
NSR_in_ANY_allow_CXCCtoVDAIP-WEB

Outbound:
NSR_out_TCP_allow_JUMPtoVNET-RDP
NSR_out_ANY_allow_WAPtoWAP-ANY

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	NSR = Network Security Group Rule	
Direction	2..3	in out	
Protocol	3	ANY TCP UDP	
Action	4..5	allow deny	
FromToWhere|Service|System	5..20	JUMPtoVNET CXCCtoVDAIP ONPREMtoJUMP WAPtoWAP	As a rule, an abbreviation of 4 characters per service is attempted here. But this is not a hard value at the moment.
ShortPortDescription	2..8	RDP HTTP HTTPS ICA DNS WEB DOMAIN	

Application Security Group *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>_<Serv`

Examples: ASG_EUWE_CO_0001_WAP_01
ASG_EUWE_CO_0001_ADDC_01
ASG_EUWE_CO_0001_ADFS_01
ASG_EUWE_CO_0001_ADCA_01
ASG_EUWE_CO_0001_AADC_01

ASG_MYTC_EUWE_CO_0001_WAP_01
ASG_MYTC_EUWE_CO_0001_ADDC_01
ASG_MYTC_EUWE_CO_0001_ADFS_01
ASG_MYTC_EUWE_CO_0001_ADCA_01
ASG_MYTC_EUWE_CO_0001_AADC_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	SUB = Subscription	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID which is also used for the corresponding VNet.	
Service|System	5..25	Describes a purpose for which the resource should be used.	
VersionNr	2	01..99	

VPN Gateway *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>`

Examples:

VPN_EUWE_CO_0001

VPN_MYTC_EUWE_CO_0001

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	VPN = VPN Gateway	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID in which subscription the resource will be published.	

Local Network Gateway *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>`

Examples:

LNG_EUWE_CO_0001

LNG_MYTC_EUWE_CO_0001

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	LNG = Local Network Gateway	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID in which subscription the resource will be published.	

Connection *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<SubscriptionID>_<SiteName>`

Examples:

LNG_EUWE_CO_0001_HQ

LNG_MYTC_EUWE_CO_0001_HQ

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	LNG = Local Network Gateway	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SubscriptionID	4	Same SubscriptionID in which subscription the resource will be published.	
SiteName	2..20	A descriptive name of the remote site.	

Internal Load Balancer *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<LBFunction><Nr>_<VersionN`

Examples:

LBI_EUWE_CO_GENP01_01
LBI_EUWE_CO_CXSF01_01

LBI_MYTC_EUWE_CO_GENP01_01
LBI_MYTC_EUWE_CO_CXSF01_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	LBI = Internal Load Balancer	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
LBFunction	4	GENP = General Purpose or a name that corresponds to the destination service.	
Nr	2	01..99, a number that is oriented towards the target service.	
VersionNr	2	01..99	

Public Load Balancer *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<LB-Function><Nr>_<VersionNr`

Examples:

LBP_EUWE_CO_GENP01_01
LBP_EUWE_CO_WAP001_01

LBP_MYTC_EUWE_CO_GENP01_01
LBP_MYTC_EUWE_CO_WAP001_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	LBP = Public Load Balancer	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
LBFunction	4	GENP = General Purpose or a name that corresponds to the destination service.	
Nr	2	01..99, a number that is oriented towards the target service.	
VersionNr	2	01..99	

Load Balancing Rules *Pattern*: `<Prefix>_<HostnamePart>_<Type>_<Protocol>_<VersionNr>`

Examples:

LBP_WAP00001_FE_01
LBP_WAP00001_BE_01
LBP_WAP00001_HP-HTTPS_01
LBP_WAP00001_LB-HTTPS_01
LBP_WAP00001_IN-HTTPS_01

LBP_WAP00001_FE_01
LBP_WAP00001_BE_01
LBP_WAP00001_HP-HTTPS_01
LBP_WAP00001_LB-HTTPS_01
LBP_WAP00001_IN-HTTPS_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	Same as the Loadbalancer LBI = Internal Load Balancer LBP = Public Load Balancer	

Identifiers	Range	Values/Meaning	Comments
HostnameBase	2	The descriptive part of the hostname, and the number of the first host.	
Type	2	FE = Frontend IP configuration BE = Backend pool HP = Health probe LB = Load balancing rule IN = Inbound NAT rule	
Protocol	2..8	HTTP HTTPS DNS	
VersionNr	2	01..99	

Automation Account *Pattern*: `<Prefix>-[TenantShort]-<Region>-<Environment>-<Name>-<VersionNr>`

Examples:

AAA-EUWE-CO-CentalAutomation-01

AAA-MYTC-EUWE-CO-CentalAutomation-01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	AAA = Azure Automation Account	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
Name	5..20	A descriptive name of the automation account.	
VersionNr	2	01..99	

Log Analytics Workspace *Pattern*: `<Prefix>-[TenantShort]-<Region>-<Environment>-<SecurityLevel>-<Name>-<`

Examples:

LAW-EUWE-AU-N-Automation-01
LAW-EUWE-CO-N-ShortRetention-01
LAW-EUWE-CO-H-LongRetention-01

LAW-MYTC-EUWE-AU-N-Automation-01
LAW-MYTC-EUWE-CO-N-ShortRetention-01
LAW-MYTC-EUWE-CO-H-LongRetention-01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	LAW = Log Analytics Workspace	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
SecurityLevel	1	N = Normal Security H = High Security	
Name	5..20	A descriptive name of the workspace.	
VersionNr	2	01..99	

Recovery Service Vault *Pattern*: `<Prefix>-[TenantShort]-<Region>-<Environment>-<Name>-<VersionNr>`

Examples:

RSV-EUWE-AU-DefaultBackup-01

RSV-EUWE-CO-DefaultBackup-01
RSV-EUWE-PR-DefaultBackup-01
RSV-EUWE-TE-DefaultBackup-01

RSV-MYTC-EUWE-AU-DefaultBackup-01
RSV-MYTC-EUWE-CO-DefaultBackup-01
RSV-MYTC-EUWE-PR-DefaultBackup-01
RSV-MYTC-EUWE-TE-DefaultBackup-01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	RSV = Recovery Service Vault	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
Name	5..20	A descriptive name of the service vault.	
VersionNr	2	01..99	

Azure Backup Policy *Pattern*: `<Prefix>-[TenantShort]-<Region>-<Environment>-<Purpose>-<BackupSchedule>-<`

ABP-EUWE-AU-AVM-D-22-UTCP01-1-7-SO5
ABP-EUWE-CO-AVM-D-22-UTCP01-1-7-SO5
ABP-EUWE-CO-AVM-D-22-UTCP01-1-7-SO5-1stSO12-Jan1stSO10
ABP-EUWE-PR-AVM-D-22-UTCP01-1-7-SO5
ABP-EUWE-PR-AVM-D-22-UTCP01-1-7-SO5-1stSO12-Jan1stSO10
ABP-EUWE-TE-AVM-D-22-UTCP01-1-7-SO5
ABP-EUWE-TE-AVM-D-22-UTCP01-1-7-SO5-1stSO12-Jan1stSO10

Identifiers	Range	Values/Meaning	Comments
Prefix	3	ABP = Azure Backup Policy	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
Purpose	3	AVM = Azure Virtual Machines AFS = Azure File Share SQL = SQL Server in Azure VM	
BackupSchedule	1	D = Daily W = Weekly	
BackupTime	2	Time, only hour	
TimeZone	6	UTCP01 = UTC + 1h UTCM01 = UTC - 1h	
Instant	1	Day: 1..5	Retain instant recovery snapshot(s).
DailyRetention	1..4	Day: 1..9999	Retention of daily backup point.
WeeklyRetention	3..6	Day: MO-SO or SE (Several) for 1..5163 weeks.	Retention of weekly backup point.
MontlyRetention	6..9	On Week Base: 1st,2nd,3rd,4th,LAS = Last On Day Base: 1..28,LA = Last Day: MO-SO or SE (Several) for 1..1188 months.	Retention of monthly backup point.
YearlyRetention	10..15	In Jan, Feb, Mar, Apr, Mai, Jun, Jul, Aug, Sep, Oct, Nov, Dec On Week Base: 1st,2nd,3rd,4th,LAS = Last On Day Base: 1..28|LA = Last Day: MO-SO or SE (Several) for 1..99 years.	Retention of yearly backup point.

Availability Set *Pattern*: `<Prefix>_[TenantShort]_<Region>_<Environment>_<HostnamePart>_<VersionNr>`

Examples:

AVS_EUWE_CO_WAP00001_01
AVS_EUWE_CO_ADDC0001_01
AVS_EUWE_PR_CXSF0001_01

AVS_MYTC_EUWE_CO_WAP00001_01
AVS_MYTC_EUWE_CO_ADDC0001_01
AVS_MYTC_EUWE_PR_CXSF0001_01

Description:

Identifiers	Range	Values/Meaning	Comments
Prefix	3	AVS = Availability Set	
TenantShort	4	MYTC = My Top Company	
Region	4	Described in the chapter Affixes, Region	
Environment	2	Described in the chapter Affixes, Environment	
HostnamePart	6 + 2	The descriptive part of the hostname, and the number of the first host.	
VersionNr	2	01..99	

Tags

The Azure Resource Manager supports tagging entities with arbitrary text strings to identify context and streamline automation. For example, the tag “sqlVersion:”sql2014ee" could identify VMs in a deployment running SQL Server 2014 Enterprise Edition for running an automated script against them. Tags should be used to augment and enhance context alongside of the naming conventions chosen.

Each resource or resource group can have a **maximum of 15 tags**. The **tag name is limited to 512 characters**, and the **tag value is limited to 256 characters**.

Some of the common tagging use cases are:

- Billing; Grouping resources and associating them with billing or charge back codes.
- Service Context Identification; Identify groups of resources across Resource Groups for common operations and grouping
- Access Control and Security Context; Administrative role identification based on portfolio, system, service, app, instance, etc.
- Production environment; Prod, Dev, Test

We recommend: Tag early - tag often. Better to have a baseline tagging scheme in place and adjust over time rather than having to retrofit after the fact. Source: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-using-tags

Blueprints

What it is

Just as a blueprint allows an engineer or an architect to sketch out the design parameters for a project, Azure Blueprints enables cloud architects and central IT to define a repeatable set of Azure resources that implements and adheres to an organization’s standards, patterns, and requirements. Azure Blueprints enables development teams to rapidly provision and stand up new environments knowing that they’re built within

organizational compliance and contain a set of built-in components – such as networking – to speed up development and delivery.

Blueprints are a declarative way to orchestrate the deployment of multiple resource templates and other artifacts such as:

- Role Assignments
- Policy Assignments
- Azure Resource Manager templates
- Resource Groups

Difference to ARM Templates

Blueprints is designed to help with environment setup, which often consists of a set of resource groups, policies, and role assignments, in addition to Resource Manager template deployments. A blueprint is a package to bring each of these artifact types together and allow you to compose and version that package – including through a CI/CD pipeline. Ultimately, each is assigned to a subscription in a single operation that can be audited and tracked.

Nearly everything that you want to include for deployment in Blueprints can be accomplished with a Resource Manager template. However, a Resource Manager template is a document that doesn't exist natively in Azure – each is stored either locally or in source control. The template gets used for deployments of one or more Azure resources, but once those resources are deployed the connection and relationship to the template used is lost.

With Blueprints, the relationship between the blueprint definition (the what should be deployed) and the blueprint assignment (the what was deployed) remains. This connection enables improved tracking and auditing of deployments, the ability to upgrade multiple subscriptions at once that are governed by the same blueprint, and more.

There's no need to choose between a Resource Manager template and a blueprint. Each blueprint can consist of zero or more Resource Manager template artifacts. This means that previous efforts to develop and maintain a library of Resource Manager templates can be leveraged in Blueprints.

Differences to Azure Policy

A blueprint is a package or container for composing focus-specific sets of standards, patterns, and requirements related to the implementation of Azure cloud services, security, and design that can be reused to ensure consistency and compliance.

A policy is a default allow and explicit deny system focused on resource properties during deployment and for already existing resources. It supports IT governance by ensuring that resources **within a subscription** adhere to requirements and standards.

Including a policy in a blueprint enables not only the creation of the right pattern or design during assignment of the blueprint, but ensures that only approved or expected changes can be made to the environment to ensure ongoing compliance to the intent of the blueprint.

A policy can be included as one of many artifacts in a blueprints definition. Blueprints also support using parameters with policies and initiatives.

Definitions of Blueprints

A blueprint is made up of artifacts. Blueprints currently support the following resources as artifacts:

Resource	Hierarchy options	Description
Resource Groups	Subscription	Create a new resource group for use by other artifacts within the blueprint. These placeholder resource groups enable you to organize resources exactly the way you want them structured and provides a scope limiter for included policy and role assignment artifacts as well as Azure Resource Manager templates.
Azure Resource Manager template	Resource Group	These templates can be used to compose complex environments such as a SharePoint farm, Azure Automation State Configuration, or a Log Analytics workspace.
Policy Assignment	Subscription, Resource Group	Allows assignment of a policy or initiative to the management group or subscription the blueprint is assigned to. The policy or initiative must be within the scope of the blueprint (in the blueprint management group or below). If the policy or initiative has parameters, these parameters can be assigned at creation of the blueprint or during blueprint assignment.
Role Assignment	Subscription, Resource Group	Add an existing user or group to a built-in role to ensure the right people always have the right access to your resources. Role assignments can be defined for the entire subscription or nested to a specific resource group included in the blueprint.

Blueprints and management groups

When creating a blueprint definition, you'll define where the blueprint is saved. Currently, blueprints can only be **saved to a management group** that you have Contributor access to. The blueprint will then be available to assign to any child (management group or subscription) of that management group.

Important

If you don't have access to any management groups or any management groups configured, loading the list of blueprint definitions will show that none are available and clicking on Scope will open a window with a warning about retrieving management groups. To resolve this, ensure a subscription you have appropriate access to is part of a management group.

Blueprint parameters

Blueprints can pass parameters to either a policy/initiative or an Azure Resource Manager template. When either artifact is added to a blueprint, the author is able to decide to provide a defined value for each blueprint assignment or to allow each blueprint assignment to provide a value at assignment time. This flexibility provides the option to define a pre-determined value for all uses of the blueprint or to enable that decision to be made at the time of assignment.

Blueprint publishing

When a blueprint is first created, it's considered to be in Draft mode. When it's ready to be assigned, it needs to be Published. Publishing requires defining a Version string (letters, numbers, and hyphens with a maximum length of 20 characters) along with optional Change notes. The Version differentiates it from future changes to the same blueprint and allows each version to be assigned. This also means different Versions of the same blueprint can be assigned to the same subscription. When additional changes are made to the blueprint, the Published Version still exists, in addition to the Unpublished changes. Once the changes are complete, the updated blueprint is Published with a new and unique Version and can now also be assigned.

Assignment of Blueprints

Each Published Version of a blueprint can be assigned to an existing subscription. In the portal, the blueprint will default the Version to the one Published most recently. If there are artifact parameters (or blueprint parameters), then the parameters will be defined during the assignment process.

Management Groups

Overview

If your organization has many subscriptions, you may need a way to efficiently manage access, policies, and compliance for those subscriptions. Azure management groups provide a level of scope above subscriptions. You organize subscriptions into containers called "management groups" and apply your governance conditions to the management groups. All subscriptions within a management group automatically inherit the conditions applied to the management group. Management groups give you enterprise-grade management at a large scale no matter what type of subscriptions you might have.

For example, you can apply policies to a management group that limits the regions available for virtual machine (VM) creation. This policy would be applied to all management groups, subscriptions, and resources under that management group by only allowing VMs to be created in that region.

Hierarchy of management groups and subscriptions

You can build a flexible structure of management groups and subscriptions to organize your resources into a hierarchy for unified policy and access management. The diagram shows an example of creating a hierarchy for governance using management groups.

By creating a hierarchy like this example you can apply a policy, for example, VM locations limited to US West Region on the group "Infrastructure Team management group" to enable internal compliance and security policies. This policy will inherit onto both EA subscriptions under that management group and will apply to all VMs under those subscriptions. As this policy inherits from the management group to the subscriptions, this security policy cannot be altered by the resource or subscription owner allowing for improved governance.

Another scenario where you would use management groups is to provide user access to multiple subscriptions. By moving multiple subscriptions under that management group, you have the ability create one role-based access control (RBAC) assignment on the management group, which will inherit that access to all the subscriptions. Without the need to script RBAC assignments over multiple subscriptions, one assignment on the management group can enable users to have access to everything they need.

Important facts about management groups:

- 10,000 management groups can be supported in a single directory.
- A management group tree can support up to six levels of depth.
- This limit doesn't include the Root level or the subscription level.
- Each management group and subscription can only support one parent.
- Each management group can have multiple children.
- All subscriptions and management groups are contained within a single hierarchy in each directory. See Important facts about the Root management group for exceptions during the Preview.

Management Group Access

Azure management groups support Azure Role-Based Access Control (RBAC) for all resource accesses and role definitions. These permissions are inherited to child resources that exist in the hierarchy. Any built-in

RBAC role can be assigned to a management group that will inherit down the hierarchy to the resources. For example, the RBAC role VM contributor can be assigned to a management group. This role has no action on the management group but will inherit to all VMs under that management group.

The following chart shows the list of roles and the supported actions on management groups.

RBAC Role Name	Create	Rename	Move	Delete	Assign Access	Assign Policy	Read
Owner	X	X	X	X	X	X	X
Contributor	X	X	X	X			X
MG Contributor*	X	X	X	X			X
Reader							X
MG Reader*							X
Resource Policy Contributor						X	
User Access Administrator					X		

*: MG Contributor and MG Reader only allow users to perform those actions on the management group scope.

Sources

Management Groups on docs.microsoft.com

Subscriptions

Azure Subscriptions are implemented in a hub and spoke model. The individual Subscriptions are grouped into three different categories. Depending on the policy and IAM requirement the categories might be modelled as Management Groups.

Subscription Model

Infrastructure Subscriptions

These Subscriptions are used by the infrastructure teams and don't host any business workloads.

Automation Subscription

The question of the question, whether the chicken or the egg was too first, that question we have also with the Subscriptions. Where do we start with deployment, and what will remain safe in the end, even if a kind of reset is performed?

This Subscription provides all the resources needed to deploy the remaining resources. Also in this Subscription are content such as Workspaces for Log Analytics, witch need of persistence, security and compliance.

At the moment it would not be the idea to provide a VNet in this Subscription.

Core Subscription (Hub)

This basically resembles the setup of on-premise data centers, where core infrastructure services (DNS, backup, AD etc.) are operated by a core IT team. These core services are made available to the different environments that host the business-related workloads.

Likewise, there is a Core Subscription that hosts core services and is operated by the core infrastructure team. If possible, there are no business workload operated in this subscription. Some exceptions might have to be made for Azure Resource Types that can be instantiated once per customer, such as Azure Data Catalog.

Sandbox Subscription (Spoke)

The infrastructure team is performing Azure related PoC, development and testing in the Sandbox Subscription. Examples of activities are the operationalization of Azure Resource Types, development of Automation Runbooks, PoC of Resource Types to be introduced. All these activities are executed from an infrastructure point of view. The development teams preform their functional testing in a special Subscription.
Example: - The development team deploys Cosmos DB in a special Subscription to determine if this PaaS could be used in the context of a new business application to be developed. - The infrastructure team will deploy Cosmos DB in the Sandbox to understand the security, financial and operational ramifications.

Standard Subscriptions (Spoke)

In addition to the Core Subscription there are three Standard Subscriptions for Development, Test and Production. These are all operated by the core infrastructure team. Only operationalized Azure Resource Types are deployed into these Subscriptions. From a policy point of view a white-listing approach is used.

Access and deployment methods become ever more restrictive from Subscription to Subscription. In the Development Subscription, the developers are free to deploy Resources as they please, using service requests, Azure portal or the IDE (Integrated Development Environment).

In the Test Subscription the deployment of Resources is in an automated manner only. This can be achieved by triggering the automation from a service requests or a CI/CD framework. This Subscription therefore is also used to test the integrated deployment of Azure resources and business applications. The Production environment is accessibly by the operate teams only. Developers will only gain temporary access in trouble shooting situations.

Special Subscriptions (Spoke)

A differentiation must be made between criteria for placing a workload into a Special Subscriptions (as opposed to a standard Subscription) and creating a new special Subscription (as opposed to using an existing one). The goal is to maintain as few Special Subscriptions as possible. Special Subscriptions that are created for a specific project should be time boxed, to prevent a multiplication of stale Subscriptions. ### Placement Criteria The following criteria must be met for placing a workload into a special Subscription: - One or several of the required Resource Types are not available in the Development Subscription. - Policies in the Development Subscription would have to be altered - but can't for e.g. security reasons. - The resources are used for a PoC or an exploratory project, but not for development.

Creation Criteria

- The resource requirements are such that the Azure Subscription Limits and Quotas might be reached, impacting other workload in the Subscription.
- Subscriptions operated by third parties such as development partners or subsidiaries.

Limits, quotas and constraints

Find the most current version here: https://docs.microsoft.com/en-us/azure/azure-subscription-service-limits

Sources

Source: - Functional Specification document - Felix Bodmer - Subscriptions on docs.microsoft.com - Subscription Service Limits - https://docs.microsoft.com/en-us/azure/architecture/cloud-adoption-guide/adoption-intro/subscription

Resource Groups

After the subscriptions are created, the resource groups should be planned.

In general, all the resources in your group should share the same lifecycle, are the same type (e.g. Storage Accounts, Network) or be used for the same service. You deploy, update, and delete them together. If one resource, such as a database server, needs to exist on a different deployment cycle it should be in another resource group. A reason for resource groups per type is security, a network admin should manage all networks, but nothing else. Same for a storage admin. And sometimes you build a service with different components, and you change in the lifetime of the service some components. But for you, it is more important to you to have the service components together and therefore use a resource group for the service. In many cases, it would be useful to combine these types of resource groups, one for network and one for storage, another for a service and others for different products with the same lifecycle.

General information about resource groups:

1. Each resource can only exist in one resource group.
2. You can add or remove a resource to a resource group at any time.
3. You can move a resource from one resource group to another group (not all resource types can be moved between resource groups)
4. A resource group can contain resources that reside in different regions.
5. A resource group can be used to scope access control for administrative actions.
6. A resource can interact with resources in other resource groups. This interaction is common when the two resources are related but do not share the same lifecycle (for example, web apps connecting to a database).

When creating a resource group, you need to provide a location for that resource group. The resource group stores metadata about the resources. Therefore, when you specify a location for the resource group, you are specifying where that metadata is stored. For compliance reasons, you may need to ensure that your data is stored in a particular region. The region for the resource group is also important if the resources are deployed by templates (see section 15.1 Provisioning with templates).

Each subscription contains a Resource Group per Service (depending on which pattern you **go). Additionally, in each subscription will be one resource group containing all network components to ensure connectivity between the different subscriptions (or regions) (see section 9 Networking)**.

Consider the Hub-Spoke Architecture in **section 9.5** when planning your Resource groups.

Source: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-overview#resource-groups

Managing security, compliance and data privacy

Infrastructure protection

VM Security

VM Authentication and access control

The first step in protecting your VM is to ensure that only authorized users are able to set up new VMs. You can use Azure policies to establish conventions for resources in your organization, create customized policies, and apply these policies to resources, such as resource groups.

VMs that belong to a resource group naturally inherit its policies. Although we recommend this approach to managing VMs, you can also control access to individual VM policies by using role-based access control (RBAC).

When you enable Resource Manager policies and RBAC to control VM access, you help improve overall VM security. We recommend that you consolidate VMs with the same life cycle into the same resource group. By using resource groups, you can deploy, monitor, and roll up billing costs for your resources. To enable users

to access and set up VMs, use a least privilege approach. And when you assign privileges to users, plan to use the following built-in Azure roles:

- **Virtual Machine Contributor:** Can manage VMs, but not the virtual network or storage account to which they are connected.
- **Security Manager:** Can manage security components, security policies, and VMs.
- **DevTest Labs User:** Can view everything and connect, start, restart, and shut down VMs.

Don't share accounts and passwords between administrators, and don't reuse passwords across multiple user accounts or services, particularly passwords for social media or other non-administrative activities. Ideally, you should use Azure Resource Manager templates to set up your VMs securely. By using this approach, you can strengthen your deployment choices and enforce security settings throughout the deployment.

Organizations that do not enforce data-access control by taking advantage of capabilities such as RBAC might be granting their users more privileges than necessary. Inappropriate user access to certain data can directly compromise that data.

Network security

Core networking resources

Access to resources can be either internal (within the corporation's network) or external (through the internet). It is easy for users in your organization to inadvertently put resources in the wrong spot, and potentially open them to malicious access. As with on-premises devices, enterprises must add appropriate controls to ensure that Azure users make the right decisions. For subscription governance, we identify core resources that provide basic control of access. The core resources consist of:

- **Virtual networks** are container objects for subnets. Though not strictly necessary, it is often used when connecting applications to internal corporate resources.
- **Network security groups** are similar to a firewall and provide rules for how a resource can "talk" over the network. They provide granular control over how/if a subnet (or virtual machine) can connect to the Internet or other subnets in the same virtual network.
- Create virtual networks dedicated to external-facing workloads and internal-facing workloads. This approach reduces the chance of inadvertently placing virtual machines that are intended for internal workloads in an external facing space.
- Configure network security groups to limit access. At a minimum, block access to the internet from internal virtual networks, and block access to the corporate network from external virtual networks.

Microsoft Azure enables you to connect virtual machines and appliances to other networked devices by placing them on Azure Virtual Networks. An Azure Virtual Network is a construct that allows you to connect virtual network interface cards to a virtual network to allow TCP/IP-based communications between network enabled devices. Azure Virtual Machines connected to an Azure Virtual Network are able to connect to devices on the same Azure Virtual Network, different Azure Virtual Networks, on the Internet or even on your own on-premises networks.

Azure Virtual Networks are similar to a LAN on your on-premises network. The idea behind an Azure Virtual Network is that you create a single private IP address space-based network on which you can place all your Azure Virtual Machines. The private IP address spaces available are in the Class A (10.0.0.0/8), Class B (172.16.0.0/12), and Class C (192.168.0.0/16) ranges.

Azure Cloud Shell

Storage accounts that you create in Cloud Shell are tagged with `ms-resource-usage:azure-cloud-shell`. If you want to manage the naming, disallow users from creating storage accounts in Cloud Shell or something else, create an Azure resource policy for tags that are triggered by this specific tag.

Source: https://docs.microsoft.com/en-us/azure/cloud-shell/persisting-shell-storage

Operating Azure IaaS Service

We recommend taking advantage of cloud scale infrastructure and increase ease of deployment through a unified "IT Management as a Service," which is called Microsoft Operations Management Suite (OMS). Management capabilities such as monitoring, backup, automation, and so forth are delivered as a service from the cloud that connects all of the servers in all environments (on-premises, Azure, and other clouds such as AWS) and allows IT staff to centrally manage operations. OMS consist of the following 4 modules:

- Log Analytics Gain visibility across your Hybrid Enterprise Cloud
- Automation Orchestrate complex and repetitive operations
- Availability Increase data protection and application availability
- Security Help secure your workloads, servers, and users

Gaining operational insights

Today traditional IT is usually using multiple different tools for platform and application monitoring, network monitoring, and Security Analysis. Extending those tools to the cloud is challenging in various aspects: connectivity, agility, and data volume. Furthermore, it is more and more necessary to combine and analyze information from various sources to gain operational insights. With OMS Log Analytics, organizations can collect, store, and analyze log data from virtually any Windows Server and Linux source and get unparalleled insights across their datacenters and clouds, including Azure and AWS.

Core monitoring

Core monitoring provides fundamental, required monitoring across Azure resources. These services require minimal configuration and collect core telemetry that the premium monitoring services use.

Azure Monitor Azure Monitor enables core monitoring for Azure services by allowing the collection of metrics, activity logs, and diagnostic logs. For example, the activity log tells you when new resources are created or modified.

Metrics are available that provide performance statistics for different resources and even the operating system inside a virtual machine. You can view this data with one of the explorers in the Azure portal and create alerts based on these metrics. Azure Monitor provides the fastest metrics pipeline (5 minute down to 1 minute), so you should use it for time critical alerts and notifications.

You can also send these metrics and logs Azure Log Analytics for trending and detailed analysis, or create additional alert rules to proactively notify you of critical issues as a result of that analysis.

Azure Advisor Azure Advisor constantly monitors your resource configuration and usage telemetry. It then gives you personalized recommendations based on best practices. Following these recommendations helps you improve the performance, security, and availability of the resources that support your applications.

Service Health The health of your application relies on the Azure services that it depends on. Azure Service Health identifies any issues with Azure services that might affect your application. Service Health also helps you plan for scheduled maintenance.

Activity Log Activity Log provides data about the operation of an Azure resource. This information includes:

- Configuration changes to the resource.

- Service health incidents.
- Recommendations on better utilizing the resource.
- Information related to autoscale operations.

You can view logs for a particular resource on its page in the Azure portal. Or you can view logs from multiple resources in Activity Log Explorer.

You can also send activity log entries to Log Analytics. There, you can analyze the logs by using data collected by management solutions, agents on virtual machines, and other sources.

Sources: https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-overview

Log Analytics

To manage access to Log Analytics, you perform various administrative tasks related to workspaces. This article provides best practice advice and procedures to manage workspaces. A workspace is essentially a container that includes account information and simple configuration information for the account. You or other members of your organization might use multiple workspaces to manage different sets of data that is collected from all or portions of your IT infrastructure.

To create a workspace, you need to:

1. Have an Azure subscription.
2. Choose a workspace name.
3. Associate the workspace with your subscription.
4. Choose a geographical location.

Log searches and alerts

We recommend creating log searches and alerts based on all the resources created in Azure.

Securing data

Microsoft is committed to protecting the privacy and securing data, while delivering software and services that help to manage the IT infrastructure of the organization. We recognize that when you entrust your data to others, that trust requires rigorous security. Microsoft adheres to strict compliance and security guidelines—from coding to operating a service.

The OMS service manages cloud-based data securely by using the following methods:

- **Data segregation:** Customer data is kept logically separate on each component throughout the OMS service. All data is tagged per organization. This tagging persists throughout the data lifecycle, and it is enforced at each layer of the service. Each customer has a dedicated Azure blob that houses the long-term data.
- **Data retention:** Aggregated metrics for some of the solutions such as Capacity Management are stored in a SQL Database hosted by Microsoft Azure. This data is stored for 390 days. Indexed log search data is stored and retained according to the pricing plan.
- **Physical security:** The OMS service is manned by Microsoft personnel, and all activities are logged and can be audited. The OMS service runs completely in Azure and complies with the Azure common engineering criteria.
- **Compliance and certifications:** The OMS software development and service team is actively working with the Microsoft Legal and Compliance teams and other industry partners to acquire a variety of certifications.

OMS Log Analytics currently meet the following security standards:

- Windows Common Engineering Criteria
- Microsoft Trustworthy Computing Certification
- ISO/IEC 27001 compliant
- Service Organization Controls (SOC) 1 Type 1 and SOC 2 Type 1 compliant

Backing up and restoring data

Backing up and restoring data are key for any production and most nonproduction workloads. The relevant scenarios are:

- Backup and restore a virtual machine
- Backup and restore files and folders
- Backup and restore application data

We recommend taking advantage of Azure backup.

Azure Backup is the Azure-based service you can use to back up (or protect) and restore your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or off-site backup solution with a cloud-based solution that is reliable, secure, and cost-competitive. Azure Backup offers multiple components that you download and deploy on the appropriate computer, server, or in the cloud. The component, or agent, that you deploy depends on what you want to protect. All Azure Backup components (no matter whether you're protecting data on-premises or in the cloud) can be used to back up data to a Recovery Services vault in Azure. See the Azure Backup components table (later in this article) for information about which component to use to protect specific data, applications, or workloads.

Traditional backup solutions have evolved to treat the cloud as an endpoint, or static storage destination, similar to disks or tape. While this approach is simple, it is limited and doesn't take full advantage of an underlying cloud platform, which translates to an expensive, inefficient solution. Other solutions are expensive because you end up paying for the wrong type of storage, or storage that you don't need. Other solutions are often inefficient because they don't offer you the type or amount of storage you need, or administrative tasks require too much time. In contrast, Azure Backup delivers these key benefits:

- **Automatic storage management** - Hybrid environments often require heterogeneous storage - some on-premises and some in the cloud. With Azure Backup, there is no cost for using on-premises storage devices. Azure Backup automatically allocates and manages backup storage, and it uses a pay-as-you-use model. Pay-as-you-use means that you only pay for the storage that you consume. For more information, see the Azure pricing article (https://azure.microsoft.com/pricing/details/backup).
- **Unlimited scaling** - Azure Backup uses the underlying power and unlimited scale of the Azure cloud to deliver high-availability - with no maintenance or monitoring overhead. You can set up alerts to provide information about events, but you don't need to worry about high-availability for your data in the cloud.
- **Multiple storage options** - An aspect of high-availability is storage replication. Azure Backup offers two types of replication: locally redundant storage and geo-redundant storage. Choose the backup storage option based on need:
  - Locally redundant storage (LRS) replicates your data three times (it creates three copies of your data) in a storage scale unit in a datacenter. All copies of the data exist within the same region. LRS is a low-cost option for protecting your data from local hardware failures.
  - Geo-redundant storage (GRS) is the default and recommended replication option. GRS replicates your data to a secondary region (hundreds of miles away from the primary location of the source data). GRS costs more than LRS, but GRS provides a higher level of durability for your data, even if there is a regional outage.

- **Unlimited data transfer** - Azure Backup does not limit the amount of inbound or outbound data you transfer. Azure Backup also does not charge for the data that is transferred. However, if you use the Azure Import/Export service to import large amounts of data, there is a cost associated with inbound data. For more information about this cost, see Offline-backup workflow in Azure Backup. Outbound data refers to data transferred from a Recovery Services vault during a restore operation.
- **Data encryption** - Data encryption allows for secure transmission and storage of your data in the public cloud. You store the encryption passphrase locally, and it is never transmitted or stored in Azure. If it is necessary to restore any of the data, only you have encryption passphrase, or key.
- **Application-consistent backup** - An application-consistent backup means a recovery point has all required data to restore the backup copy. Azure Backup provides application-consistent backups, which ensure additional fixes are not required to restore the data. Restoring application-consistent data reduces the restoration time, allowing you to quickly return to a running state.
- **Long-term retention** - You can use Recovery Services vaults for short-term and long-term data retention. Azure doesn't limit the length of time data can remain in a Recovery Services vault. You can keep data in a vault for as long as you like. Azure Backup has a limit of 9999 recovery points per protected instance.

Azure virtual Machines

When the Azure Backup service initiates a backup job at the scheduled time, it triggers the backup extension to take a point-in-time snapshot. The Azure Backup service uses the VMSnapshot extension in Windows, and the VMSnapshotLinux extension in Linux. The extension is installed during the first VM backup. To install the extension, the VM must be running. If the VM is not running, the Backup service takes a snapshot of the underlying storage (since no application writes occur while the VM is stopped).

1. During the backup process, Azure Backup doesn't include the temporary disk attached to the virtual machine.
2. Since Azure Backup takes a storage-level snapshot and transfers that snapshot to vault, do not change the storage account keys until the backup job finishes.
3. For premium VMs, we copy the snapshot to storage account. This is to make sure that Azure Backup service gets sufficient IOPS for transferring data to vault. This additional copy of storage is charged as per the VM allocated size.

Microsoft's **best practices** while configuring backups for virtual machines:

- Do not schedule more than 40 VMs to back up at the same time.
- Schedule VM backups during non-peak hours. This way the Backup service uses IOPS for transferring data from the customer storage account to the vault.
- Make sure that a policy is applied on VMs spread across different storage accounts. We suggest no more than 20 total disks from a single storage account be protected by the same backup schedule. If you have greater than 20 disks in a storage account, spread those VMs across multiple policies to get the required IOPS during the transfer phase of the backup process.
- Do not restore a VM running on Premium storage to same storage account. If the restore operation process coincides with the backup operation, it reduces the available IOPS for backup.
- For Premium VM backup, ensure that storage account that hosts premium disks has atleast 50% free space for staging snapshot for a successful backup.
- Make sure that python version on Linux VMs enabled for backup is 2.7

Files and Folders

This backup option is designed to back up files and folders from any Windows machine. The machine can run in Azure, on-premises, or in any other cloud; it can be physical or virtual. You cannot use this option to back up the system state, or to create a Bare-Metal-Restore (BMR) backup. The Recovery Services Vault could be the one that is mentioned in the previous section, or it could be any other Recovery Services Vault.

Azure Backup for files and folders requires the installation of an Azure Backup Agent on the server, which can be downloaded from the Azure Recovery Services Vault. After installing the agent, it is necessary to connect the server to the Recovery Services Vault by downloading the vault credential files from the Recovery Services Vault. The vault credentials file is used only during the registration workflow and expires after 48 hours. Ensure that the vault credential file is available in a location that can be accessed by the setup application.

Sources: https://docs.microsoft.com/en-us/azure/backup/backup-azure-vms-introduction#capacity-planning

Establishing secure remote access

Automating operational procedure

Managing IT services according to ITIL

Recommendations for operation Azure IaaS Services

Use nonpeak hours for backups	Schedule backups during nonpeak hours for VMs so that backup service gets IOPS for transferring data from customer storage account to backup vault.
Spread VMs into different backup schedules	Please make sure that in a policy VMs are spread from different storage accounts. We suggest that if the total number of disks stored in a single storage account from VMs is more than 20, spread the VMs into different backup schedules to get required IOPS during transfer phase of the backup.
Back up critical data to separate location	Ensuring all applications and mission-critical data is backed up at a separate secondary location other than the primary datacenter

Offering management for cloud-based services

Provisioning with templates

With Resource Manager, application designers can create a simple template (in JSON format) that defines deployment and configuration of entire application. This template is known as a Resource Manager template and provides a declarative way to define deployment. By using a template, you can repeatedly deploy the application throughout the app lifecycle and have confidence that resources are deployed in a consistent state.

Within the template, you define the infrastructure for your app, how to configure that infrastructure, and how to publish your app code to that infrastructure. You do not need to worry about the order for deployment because Azure Resource Manager analyzes dependencies to ensure resources are created in the correct order. There is no need to define your entire infrastructure in a single template. Often, it makes sense to divide the deployment requirements into a set of targeted, purpose-specific templates that are linked together.

Templates allow the specification of parameters for customization and flexibility in deployments. Those parameters should fit to the configuration options in the service catalog. The fact that resource manager orchestrates the deployment of multiple components supports a definition of application-specific instead

of infrastructure-specific parameters. Example: For a deployment of a Big Data Service you may ask the customer for parameters such as the amount of master and data nodes, the required performance, and the amount of data that should be handled. All the logic for provisioning multiple virtual machines, load balancers, network settings, application installation, and configuration, etc., could be baked into the template. Source: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-subscription-examples, https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/, https://www.xpirit.com/2017/11/23/best-practices-using-azure-resource-manager-templates/

Networking

Virtual Network (VNet)

An Azure Virtual Network is bound within an Azure subscription and region. It is therefore not possible for multiple subscriptions to use the same Azure Virtual Network. The solution is to create separate Virtual Networks in each of the Azure subscriptions with each using a **different** IP address space.

Resources can only be connected to a virtual network that exists in the same region and subscription the resource is in.

You can connect virtual networks to each other by using:

- **Virtual network peering:** The virtual networks can be both in different tenants and different regions to peer the networks. Bandwidth between resources in peered virtual networks is the same as if the resources were connected to the same virtual network.
- **An Azure VPN Gateway:** The virtual networks can exist in the same, or different Azure regions. Bandwidth between resources in virtual networks connected through a VPN Gateway is limited by the bandwidth of the VPN Gateway.

Source: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-networks-overview

Subnets

Each VNet can contain different subnets.

Each service is placed in a separate subnet.

Network security group (NSG)

For isolation and traffic control, each Subnet is assigned with a network security group, where in- and outbound network connectivity can be controlled. For more specific control, NSG's can be assigned to the network interface card (NIC) of a VM. As services are very separated between subnets, this is only recommended in special cases.

Application security group (ASG)

Instead of NSG on the network interface card, ASG can be used. Application security groups enable you to configure network security as a natural extension of an application's structure, allowing you to group virtual machines and define network security policies based on those groups. You can reuse your security policy at scale without manual maintenance of explicit IP addresses. The platform handles the complexity of explicit IP addresses and multiple rule sets, allowing you to focus on your business logic. With ASG, you can control the traffic without the need to assign an NSG to a NIC, with the full isolation of a NIC.

At the moment it's not possible to use ASG in different VNet to create a NSG rule. It's something that is in planning (Uservoice).

Source: https://docs.microsoft.com/en-us/azure/virtual-network/security-overview#application-security-groups

Hub-Spoke Architecture

The hub is a virtual network (VNet) in Azure that acts as a central point of connectivity to your on-premises network. The spokes are VNets that peer with the hub and can be used to isolate workloads. Traffic flows between the on-premises datacenter and the hub through an ExpressRoute or VPN gateway connection.

The architecture consists of the following components.

- **On-premises network.** A private local-area network running within an organization.
- VPN device. A device or service that provides external connectivity to the on-premises network. The VPN device may be a hardware device, or a software solution such as the Routing and Remote Access Service (RRAS) in Windows Server 2012. For a list of supported VPN appliances and information on configuring selected VPN appliances for connecting to Azure, see About VPN devices for Site-to-Site VPN Gateway connections.
- **VPN virtual network gateway or ExpressRoute gateway.** The virtual network gateway enables the VNet to connect to the VPN device, or ExpressRoute circuit, used for connectivity with your on-premises network. For more information, see Connect an on-premises network to a Microsoft Azure virtual network.
- **Hub VNet.** Azure VNet used as the hub in the hub-spoke topology. The hub is the central point of connectivity to your on-premises network, and a place to host services that can be consumed by the different workloads hosted in the spoke VNets.
- **Gateway subnet.** The virtual network gateways are held in the same subnet.
- **Spoke VNets.** One or more Azure VNets that are used as spokes in the hub-spoke topology. Spokes can be used to isolate workloads in their own VNets, managed separately from other spokes. Each workload might include multiple tiers, with multiple subnets connected through Azure load balancers. For more information about the application infrastructure, see Running Windows VM workloads and Running Linux VM workloads.
- **VNet peering.** Two VNets in the same Azure region, in different regions or even in different tenants can be connected using a peering connection. Peering connections are non-transitive, low latency connections between VNets. Once peered, the VNets exchange traffic by using the Azure backbone, without the need for a router. In a hub-spoke network topology, you use VNet peering to connect the hub to each spoke.

We recommend defining a logical structure of the private address spaces, that can be used in Azure. It is important to keep in mind, that each address space must be flexible for other resources but not overlap with other private address spaces of other VNets or locally used private address spaces. Adress Spaces cannot be changed in peered VNets – the Virtual Networks must be unpeered to add or remove additional address spaces, and then again be peered.

Source: https://docs.microsoft.com/en-us/office365/enterprise/designing-networking-for-microsoft-azure-iaas

Intersite network connectivity – integrate Azure in the Corporate Network – the Hub VNET

You can connect your on-premises network to a virtual network using any combination of the following options:

- **Point-to-site virtual private network (VPN):** Established between a virtual network and a single PC in your network. Each PC that wants to establish connectivity with a virtual network must configure its connection independently. This connection type is great if you're just getting started with Azure, or for developers, because it requires little or no changes to your existing network. The connection uses the SSTP protocol to provide encrypted communication over the Internet between the PC and a virtual network. The latency for a point-to-site VPN is unpredictable, since the traffic traverses the Internet.

- **Site-to-site VPN:** Established between your VPN device and an Azure VPN Gateway deployed in a virtual network. This connection type enables any on-premises resource you authorize to access a virtual network. The connection is an IPSec/IKE VPN that provides encrypted communication over the Internet between your on-premises device and the Azure VPN gateway. The latency for a site-to-site connection is unpredictable, since the traffic traverses the Internet.
- **Azure ExpressRoute:** Established between your network and Azure, through an ExpressRoute partner. This connection is private. Traffic does not traverse the Internet. The latency for an ExpressRoute connection is predictable, since traffic doesn't traverse the Internet.
- **Azure Virtual Network Peering:** Virtual network peering enables you to seamlessly connect Azure virtual networks. Once peered, the virtual networks appear as one, for connectivity purposes. The traffic between virtual machines in the peered virtual networks is routed through the Microsoft backbone infrastructure, much like traffic is routed between virtual machines in the same virtual network, through private IP addresses only.

Source: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-peering-overview

VPN Gateway

A VPN gateway is a type of virtual network gateway that sends encrypted traffic across a public connection to an on-premises location. You can also use VPN gateways to send encrypted traffic between Azure virtual networks over the Microsoft network. To send encrypted network traffic between your Azure virtual network and your on-premises site, you must create a VPN gateway for your virtual network.

Each virtual network can have only one VPN gateway, however, you can create multiple connections to the same VPN gateway. An example of this is a Multi-Site connection configuration. When you create multiple connections to the same VPN gateway, all VPN tunnels, including Point-to-Site VPNs, share the bandwidth that is available for the gateway.

Configuring a VPN Gateway

A VPN gateway connection relies on multiple resources that are configured with specific settings. Most of the resources can be configured separately, although they must be configured in a certain order in some cases.

Gateway types

Each virtual network can only have one virtual network gateway of each type. When you are creating a virtual network gateway, you must make sure that the gateway type is correct for your configuration.

The available values for -GatewayType are:

- Vpn
- ExpressRoute

Planning

	Point-to-Site	**Site-to-Site**	**ExpressRoute**
Azure Supported Services	Cloud Services and Virtual Machines	Cloud Services and Virtual Machines	Services list
Typical Bandwidths	Typically < 100 Mbps aggregate	Typically < 1 Gbps aggregate	50 Mbps, 100 Mbps, 200 Mbps, 500 Mbps, 1 Gbps, 2 Gbps, 5 Gbps, 10 Gbps
Protocols Supported	Secure Sockets Tunneling Protocol (SSTP)	IPsec	Direct connection over VLANs, NSP's VPN technologies (MPLS, VPLS,. . .)

	Point-to-Site	Site-to-Site	ExpressRoute
Routing	RouteBased (dynamic)	We support PolicyBased (static routing) and RouteBased (dynamic routing VPN)	BGP
Connection re-siliency	active-passive	active-passive or active-active	active-active
Typical use case	Prototyping, dev / test / lab scenarios for cloud services and virtual machines	Dev / test / lab scenarios and small scale production workloads for cloud services and virtual machines	Access to all Azure services (validated list), Enterprise-class and mission critical workloads, Backup, Big Data, Azure as a DR site
SLA	SLA	SLA	SLA
Pricing	Pricing	Pricing	Pricing
Technical Docu-mentation	VPN Gateway Documentation	VPN Gateway Documentation	ExpressRoute Documentation
FAQ	VPN Gateway FAQ	VPN Gateway FAQ	ExpressRoute FAQ

For SLA, Pricing and technical documentation see: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways

Gateway SKUs

The new VPN gateway SKUs streamline the feature sets offered on the gateways:

SKU Features *Basic* Route-based VPN: 10 tunnels for S2S/connections; no RADIUS authentication for P2S; no IKEv2 for P2S Policy-based VPN: (IKEv1): 1 S2S/connection tunnel; no P2S

VpnGw1, VpnGw2, and VpnGw3 Route-based VPN: up to 30 tunnels (*), P2S, BGP, active-active, custom IPsec/IKE policy, ExpressRoute/VPN coexistence

The Basic SKU is considered a legacy SKU. The Basic SKU has certain feature limitations. You can't resize a gateway that uses a Basic SKU to one of the new gateway SKUs, you must instead change to a new SKU, which involves deleting and recreating your VPN gateway.

Source: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpn-gateway-settings#gwsku

Site-to-Site and Multi-Site

Site-to-Site

A Site-to-Site (S2S) VPN gateway connection is a connection over IPsec/IKE (IKEv1 or IKEv2) VPN tunnel. S2S connections can be used for cross-premises and hybrid configurations. A S2S connection requires a VPN device located on-premises that has a public IP address assigned to it and is not located behind a NAT.

Multi-Site

This type of connection is a variation of the Site-to-Site connection. You create more than one VPN connection from your virtual network gateway, typically connecting to multiple on-premises sites. When working with multiple connections, you must use a RouteBased VPN type . Because each virtual network can only have one VPN gateway, all connections through the gateway share the available bandwidth. This is often called a "multi-site" connection.

Source: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways#s2smulti

Point-to-Site

A Point-to-Site (P2S) VPN gateway connection lets you create a secure connection to your virtual network from an individual client computer. A P2S connection is established by starting it from the client computer. This solution is useful for telecommuters who want to connect to Azure VNets from a remote location, such as from home or a conference. P2S VPN is also a useful solution to use instead of S2S VPN when you have only a few clients that need to connect to a VNet.

Unlike S2S connections, P2S connections do not require an on-premises public-facing IP address or a VPN device. P2S connections can be used with S2S connections through the same VPN gateway, as long as all the configuration requirements for both connections are compatible.

Source: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways#P2S

Express Route

Microsoft Azure ExpressRoute lets you extend your on-premises networks into the Microsoft cloud over a private connection facilitated by a connectivity provider. With ExpressRoute, you can establish connections to Microsoft cloud services, such as Microsoft Azure, Office 365, and CRM Online.

ExpressRoute connections do not go over the public Internet. This allows ExpressRoute connections to offer more reliability, faster speeds, lower latencies, and higher security than typical connections over the Internet.

An ExpressRoute connection does not use a VPN gateway, although it does use a virtual network gateway as part of its required configuration. In an ExpressRoute connection, the virtual network gateway is configured with the gateway type 'ExpressRoute', rather than 'Vpn'.

What is Azure ExpressRoute?

When talking about ExpressRoute, in most cases an ExpressRoute circuit is referenced. An ExpressRoute circuit represents the logical connection between your on-premises infrastructure and Microsoft cloud services through a connectivity provider. So when thinking about using Azure ExpressRoute, it is important to be aware that there is always a connectivity provider involved. You need to consider this when it comes to overall costs of ExpressRoute.

It is possible to order multiple ExpressRoute circuits. Each circuit can be in the same or different regions, and can be connected to on premises through different connectivity providers. A circuit has a fixed bandwidth and is mapped to exactly one connectivity provider and one peering location (e.g. West Europe).

Common scenarios There obviously exist a variety of reasons to implement ExpressRoute. The following scenarios are probably the most common ones:

- Storage, backup and recovery
- BI and big data (working with big amounts of data in the cloud)
- Hybrid apps (e.g. app in the cloud and database on premise)

Unlike Site-to-Site VPN which only works for IaaS, Azure ExpressRoute can be used with various other public services (e.g. Websites, IoT, Backup, database services).

Encryption of traffic Although ExpressRoute is a private connection, traffic flowing over the network is **not encrypted**. Encryption in transit can be achieved by encrypting traffic flowing over the connection.

If you wish to encrypt your traffic over ExpressRoute, you have three options:

- Application level encryption
- IPsec Site-to-Site VPN over ExpressRoute using Azure VPN Gateways. Check out ExpressRoute documentation for guidance about how to configure a site-to-site VPN over ExpressRoute Microsoft peering: https://docs.microsoft.com/en-gb/azure/expressroute/site-to-site-vpn-over-microsoft-peering

- Third-party appliance that performs encryption of traffic flowing over ExpressRoute. For example by deploying physical and/or virtual encryption devices on both sides (e.g. Fortinet, F5, Steelhead, etc)

Source: Stefan Johner: https://blog.jhnr.ch/2018/05/29/azure-expressroute-overview/ , 2018

Connectivity Models

Co-located at a cloud exchange If you are co-located in a facility with a cloud exchange, you can order virtual cross-connections to the Microsoft cloud through the co-location provider's Ethernet exchange. Co-location providers can offer either Layer 2 cross-connections, or managed Layer 3 cross-connections between your infrastructure in the co-location facility and the Microsoft cloud.

Point-to-point Ethernet connections You can connect your on-premises datacenters/offices to the Microsoft cloud through point-to-point Ethernet links. Point-to-point Ethernet providers can offer Layer 2 connections, or managed Layer 3 connections between your site and the Microsoft cloud.

Any-to-any (IPVPN) networks You can integrate your WAN with the Microsoft cloud. IPVPN providers (typically MPLS VPN) offer any-to-any connectivity between your branch offices and datacenters. The Microsoft cloud can be interconnected to your WAN to make it look just like any other branch office. WAN providers typically offer managed Layer 3 connectivity. ExpressRoute capabilities and features are all identical across all of the above connectivity models.

Source: https://docs.microsoft.com/en-us/azure/expressroute/expressroute-connectivity-models

ExpressRoute Providers

If you're looking for an ExpressRoute service-provider, we recommend visiting the following site, for an up-to-date list of each provider broken down by location: https://docs.microsoft.com/en-us/azure/expressroute/expressroute-locations-providers

Bandwidth Options

You can an up-to-date list of the currently available bandwith options under the folowing site: https://docs.microsoft.com/en-gb/azure/expressroute/expressroute-introduction#bandwidth-options

Recommendations

To connect your on-premise location with your Hub-VNet in Azure, we recommend setting one VPN Connection (Site-2-Site) to Azure in main VNet "Company-VNet", peer all VNets of the subscriptions with the main VNet, so all resources can communicate with each other. Secure traffic with NSGs. Keep the maximum available bandwidth in mind (1.25 GB with the VPN Gateway 3).

https://docs.microsoft.com/en-us/azure/security/azure-security-network-security-best-practices

Recommendations for cloud connectivity	
Optimize intranet connectivity to your edge network	Over the years, many organizations have optimized intranet connectivity and performance to applications running in on-premises datacenters. With productivity and IT workloads running in the Microsoft cloud, additional investment must ensure high-connectivity availability and that traffic performance between your edge network and your intranet users is optimal.

Recommendations for cloud connectivity	
Optimize throughput at your edge network	As more of your day-to-day productivity traffic travels to the cloud, you should closely examine the set of systems at your edge network to ensure that they are current, provide high availability, and have sufficient capacity to meet peak loads.
For a high SLA use ExpressRoute	Although you can utilize your current Internet connection from your edge network, traffic to and from Microsoft cloud services must share the pipe with other intranet traffic going to the Internet. In addition, your traffic to Microsoft cloud services is subject to Internet traffic congestion. For a high SLA and the best performance, use ExpressRoute, a dedicated WAN connection between your network and Azure. ExpressRoute can leverage your existing network provider for a dedicated connection. Resources connected by ExpressRoute appear as if they are on your WAN, even for geographically distributed organizations
Analyse your current network	- Analyse your client computers and optimize for network hardware, software drivers, protocol settings, and Internet browsers. - Analyse your on-premises network for traffic latency and optimal routing to the Internet edge device. - Analyse the capacity and performance of your Internet edge device and optimize for higher levels of traffic. - Analyse the latency between your Internet edge device (such as your external firewall) and the regional locations of the Microsoft cloud service to which you are connecting. - Analyse the capacity and utilization of your current Internet connection and add capacity if needed. Alternately, add an ExpressRoute connection.

Recommendations for cloud connectivity	
Plan and design networking for Azure	- Prepare your intranet for Microsoft cloud services. - Optimize your Internet bandwidth. - Determine the type of VNet (cloud-only or cross-premises). - Determine the address space of the VNet.- Determine the subnets within the VNet and the address spaces assigned to each.- Determine the DNS server configuration and the addresses of the DNS servers to assign to VMs in the VNet.- Determine the load balancing configuration (Internet-facing or internal).- Determine the use of virtual appliances and user-defined routes.- Determine how computers from the Internet will connect to virtual machines.- For multiple VNets, determine the VNet-to-VNet connection topology.- Determine the on-premises connection to the VNet (S2S VPN or ExpressRoute).- Determine the on-premises VPN device or router.- Add routes to make the address space of the VNet reachable.- For ExpressRoute, plan for the new connection with your provider.- Determine the Local Network address space for the Azure gateway. - Configure on-premises DNS servers for DNS replication with DNS servers hosted in Azure. *Determine the use of forced tunneling and user-defined routes.

Azure Network Peering

Overview

Virtual network peering enables you to seamlessly connect Azure virtual networks. Once peered, the virtual networks appear as one, for connectivity purposes. The traffic between virtual machines in the peered virtual networks is routed through the Microsoft backbone infrastructure, much like traffic is routed between virtual machines in the same virtual network, through private IP addresses only. Azure supports:

- VNet peering - connecting VNets within the same Azure region
- Global VNet peering - connecting VNets across Azure regions

The benefits of using virtual network peering, whether local or global, include:

- Network traffic between peered virtual networks is private. Traffic between the virtual networks is kept on the Microsoft backbone network. No public Internet, gateways, or encryption is required in the communication between the virtual networks.
- A low-latency, high-bandwidth connection between resources in different virtual networks.
- The ability for resources in one virtual network to communicate with resources in a different virtual network, once the virtual networks are peered.
- The ability to transfer data across Azure subscriptions, deployment models, and across Azure regions.
- The ability to peer virtual networks created through the Azure Resource Manager.
- No downtime to resources in either virtual network when creating the peering, or after the peering is created.

Requirements and Constraints

- **You can peer virtual networks in the same region, or different regions**. The following constraints do **not** apply when both virtual networks are in the same region, but do apply when the

virtual networks are globally peered:

  - The virtual networks can exist in any Azure public cloud region, but not in Azure national clouds.
  - Resources in one virtual network cannot communicate with the IP address of an Azure internal load balancer in the peered virtual network. The load balancer and the resources that communicate with it must be in the same virtual network.
  - **You cannot use remote gateways or allow gateway transit**. To use remote gateways or allow gateway transit, both virtual networks in the peering must exist in the same region.
  - Communication across globally peered virtual networks through the following VM types is not supported: **High performance compute and GPU**. This includes H, NC, NV, NCv2, NCv3, and ND series VMs.

- The virtual networks can be in **the same, or different subscriptions**. If you don't already have an AD tenant, you can quickly create one. You can use a VPN Gateway to connect two virtual networks that exist in different subscriptions that are associated to different Active Directory tenants.
- The virtual networks you peer must have **non-overlapping IP address spaces**.
- You **can't add address ranges to, or delete address ranges from a virtual network's address space once a virtual network is peered with another virtual network**. To add or remove address ranges, delete the peering, add or remove the address ranges, then re-create the peering. To add address ranges to, or remove address ranges from virtual networks, see Manage virtual networks.
- When peering two virtual networks created through Resource Manager, a peering must be configured for each virtual network in the peering. You see one of the following types for peering status:
  - Initiated: When you create the peering to the second virtual network from the first virtual network, the peering status is Initiated.
  - Connected: When you create the peering from the second virtual network to the first virtual network, its peering status is Connected. If you view the peering status for the first virtual network, you see its status changed from Initiated to Connected. The peering is not successfully established until the peering status for both virtual network peerings is connected.
- A peering is established between two virtual networks. **Peerings are not transitive.** If you create peerings between:
  - VirtualNetwork1 & VirtualNetwork2
  - VirtualNetwork2 & VirtualNetwork3

  There is no peering between VirtualNetwork1 and VirtualNetwork3 through VirtualNetwork2. If you want to create a virtual network peering between VirtualNetwork1 and VirtualNetwork3, you have to create a peering between VirtualNetwork1 and VirtualNetwork3.
- You **can't resolve names in peered virtual networks** using default Azure name resolution. To resolve names in other virtual networks, you **must use Azure DNS** for private domains or a custom DNS server.
- Resources in peered virtual networks in the same region can communicate with each other with the same bandwidth and latency as if they were in the same virtual network. Each virtual machine size has its own maximum network bandwidth however.
- A virtual network can be peered to another virtual network, and also be connected to another virtual network with an Azure virtual network gateway. When virtual networks are connected through both peering and a gateway, traffic between the virtual networks flows through the peering configuration, rather than the gateway.
- There is a nominal charge for ingress and egress traffic that utilizes a virtual network peering.

Source: https://docs.microsoft.com/en-us/azure/virtual-network/virtual-network-manage-peering#requirements-and-constraints

Connectivity

After virtual networks are peered, resources in either virtual network can directly connect with resources in the peered virtual network.

The network latency between virtual machines in peered virtual networks in the same region is the same as the latency within a single virtual network. The network throughput is based on the bandwidth that's allowed for the virtual machine, proportionate to its size. There isn't any additional restriction on bandwidth within the peering.

The traffic between virtual machines in peered virtual networks is routed directly through the Microsoft backbone infrastructure, not through a gateway or over the public Internet.

Service chaining

You can configure user-defined routes that point to virtual machines in peered virtual networks as the next hop IP address, or to virtual network gateways, to enable service chaining. Service chaining enables you to direct traffic from one virtual network to a virtual appliance, or virtual network gateway, in a peered virtual network, through user-defined routes.

You can deploy hub-and-spoke networks, where the hub virtual network can host infrastructure components such as a network virtual appliance or VPN gateway. All the spoke virtual networks can then peer with the hub virtual network. Traffic can flow through network virtual appliances or VPN gateways in the hub virtual network.

Virtual network peering enables the next hop in a user-defined route to be the IP address of a virtual machine in the peered virtual network, or a VPN gateway. You cannot however, route between virtual networks with a user-defined route specifying an ExpressRoute gateway as the next hop type.

Pricing

There is a nominal charge for ingress and egress traffic that utilizes a virtual network peering connection.

Gateway transit is a peering property that enables a virtual network to utilize a VPN/ExpressRoute gateway in a peered virtual network for cross premises or VNet-to-VNet connectivity. Traffic passing through a remote gateway in this scenario is subject to VPN gateway charges or ExpressRoute gateway charges and does not incur VNet peering charges. For example, If VNetA has a VPN gateway for on-premises connectivity and VNetB is peered to VNetA with appropriate properties configured, traffic from VNetB to on-premises is only charged egress per VPN gateway pricing or ExpressRoute pricing. VNet peering charges do not apply.

The price for 100 GB in and/or out is currently (August 2019) $1 within the same region and $3.5 between regions.

Protecting virtual networks

We recommend locking all Network resources to "CanNotDelete".

Network Security Groups

A network security group (NSG) contains a list of security rules that allow or deny network traffic to resources connected to Azure Virtual Networks (VNet). NSGs can be associated to subnets or individual network interfaces (NIC) attached to VMs (Resource Manager). When an NSG is associated to a subnet, the rules apply to all resources connected to the subnet. Traffic can further be restricted by also associating an NSG to a VM or NIC.

NSGs contain two sets of rules: Inbound and outbound. The priority for a rule must be unique within each set. All NSGs contain a set of default rules. The default rules cannot be deleted, but because they are assigned the lowest priority, they can be overridden by the rules that you create.

The default rules allow and disallow traffic as follows:

- **Virtual network:** Traffic originating and ending in a virtual network is allowed both in inbound and outbound directions.
- **Internet:** Outbound traffic is allowed, but inbound traffic is blocked.
- **Load balancer:** Allow Azure Load Balancer to probe the health of your VMs and role instances. If you override this rule, Azure Load Balancer health probes will fail which could cause impact to your service.

Forced Tunneling

Recommended to enable for point-to-site connections.

Application Security Groups (ASG)

Network security micro segmentation

ASGs enable you to define fine-grained network security policies based on workloads, centralized on applications, instead of explicit IP addresses. Provides the capability to group VMs with monikers and secure applications by filtering traffic from trusted segments of your network.

Implementing granular security traffic controls improves isolation of workloads and protects them individually. If a breach occurs, this technique limits the potential impact of lateral exploration of your networks from hackers.

Security definition simplified

With ASGs, filtering traffic based on applications patterns is simplified, using the following steps:

- Define your application groups, provide a moniker descriptive name that fits your architecture. You can use it for applications, workload types, systems, tiers, environments or any role.
- Define a single collection of rules using ASGs and Network Security Groups (NSG), you can apply a single NSG to your entire virtual network on all subnets. A single NSG gives you full visibility on your traffic policies, and a single place for management.
- Scale at your own pace. When you deploy VMs, make them members of the appropriate ASGs. If your VM is running multiple workloads, just assign multiple ASGs. Access is granted based on your workloads. No need to worry about security definition again. More importantly, you can implement a zero-trust model, limiting access to the application flows that are explicitly permitted.

Single network security policy

ASGs introduce the ability to deploy multiple applications within the same subnet, and isolate traffic based on ASGs. With ASGs you can reduce the number of NSGs in your subscription. In some cases, you can use a single NSG for multiple subnets of your virtual network. ASGs enable you to centralize your configuration, providing the following benefits in dynamic environments:

- **Centralized NSG view:** All traffic policies in a single place. It's easy to operate and manage changes. If you need to allow a new port to or from a group of VMs, you can make a change to a single rule.
- **Centralized logging:** In combination with NSG flow logs, a single configuration for logs has multiple advantages for traffic analysis.
- **Enforce policies:** If you need to deny specific traffic, you can add a security rule with high priority and enforce administrative rules.

Filtering east-west traffic

With ASGs, you can isolate multiple workloads and provide additional levels of protection for your virtual network.

In the following illustration, multiple applications are deployed into the same virtual network. Based on the security rules described, workloads are isolated from each other. If a VM from one of the applications is compromised, lateral exploration is limited, minimizing the potential impact of an attacker.

In this example, let's assume one of the web server VMs from application1 is compromised, the rest of the application will continue to be protected, even access to critical workloads like database servers will still be unreachable. This implementation provides multiple extra layers of security to your network, making this intrusion less harmful and easy to react on such events.

EastWest Traffic Example

Filtering north-south traffic

In combination with additional features on NSG, you can also isolate your workloads from on premises and azure services in different scenarios.

In the following illustration, a relatively complex environment is configured for multiple workload types within a virtual network. By describing their security rules, applications have the correct set of policies applied on each VM. Similar to the previous example, if one of your branches is compromised, exploration within the virtual network is limited therefore minimizing the potential impact of an intruder.

In this example, let's assume someone on one of your branches connected using VPN, compromise a workstation and has access to your network. Normally only a subset of your network is required for this branch, by isolating the rest of your network; all other applications will continue to be protected and unreachable. ASGs another layers of security to your entire network.

Another interesting scenario, assuming you have detected a breach on one of your web servers, a good idea would be to isolate the VM for investigation. With ASGs, you can easily assign a special group predefined for quarantine VMs on your first security policy. These VMs lose access providing an additional benefit to help you react and mitigate this treats.

NorthSouth Traffic Example

Summary

Application Security Groups along with the latest improvements in NSGs, have brought multiple benefits on the network security area, such as a single management experience, increased limits on multiple dimensions, a great level of simplification, and a natural integration with your architecture, begin today and experience these capabilities on your virtual networks.

Source: https://azure.microsoft.com/en-in/blog/applicationsecuritygroups/

Virtual network appliances

While Network Security Groups and User Defined Routing can provide a certain measure of network security at the network and transport layers of the OSI model, there are going to be situations where you'll want or need to enable security at high levels of the stack. In such situations, we recommend that you deploy virtual network security appliances provided by Azure partners.

Azure network security appliances can deliver significantly enhanced levels of security over what is provided by network level controls. Some of the network security capabilities provided by virtual network security appliances include:

- Firewalling
- Intrusion detection/Intrusion Prevention

- Vulnerability management
- Application control
- Network-based anomaly detection
- Web filtering
- Antivirus
- Botnet protection
- If you require a higher level of network security than you can obtain with network level access controls, then we recommend that you investigate and deploy Azure virtual network security appliances.

Sources: https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpngateways?toc=%2fazure%2fvirtual-network%2ftoc.json#diagrams, https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-about-vpn-gateway-settings

Identity Management

AD Connect

Azure AD Connect will integrate your on-premises directories with Azure Active Directory. This allows you to provide a common identity for your users for Office 365, Azure, and SaaS applications integrated with Azure AD. This topic will guide you through the planning, deployment, and operation steps. It is a collection of links to the topics related to this area.

We recommend using Azure AD Connect. If Azure Active Directory Sync is in use, we recommend as well to upgrade to AD Connect, as DirSync is deprecated.

- Synchronizing users to Azure AD is a free feature and doesn't require customers to have any paid subscription.
- Synchronized users are not automatically granted any license. Admins still have total control on the license assignment.
- Microsoft's recommendation is for IT admins to synchronize all their users. This not only unblocks the users to access any Azure AD integrated resource but also gives a much broader view for IT admins to see what applications are being accessed by their users.

Azure Active Directory Connect is made up of three primary components: the synchronization services, the optional Active Directory Federation Services component, and the monitoring component named Azure AD Connect Health.

- Synchronization - This component is responsible for creating users, groups, and other objects. It is also responsible for making sure identity information for your on-premises users and groups is matching the cloud.
- AD FS - Federation is an optional part of Azure AD Connect and can be used to configure a hybrid environment using an on-premises AD FS infrastructure. This can be used by organizations to address complex deployments, such as domain join SSO, enforcement of AD sign-in policy, and smart card or 3rd party MFA.
- Health Monitoring - Azure AD Connect Health can provide robust monitoring and provide a central location in the Azure portal to view this activity. For additional information, see Azure Active Directory Connect Health.

Source: https://docs.microsoft.com/en-us/azure/active-directory/hybrid/whatis-azure-ad-connect

User Sign-in options

Azure Active Directory (Azure AD) Connect allows your users to sign in to both cloud and on-premises resources by using the same passwords. This article describes key concepts for each identity model to help you choose the identity that you want to use for signing in to Azure AD.

- Password hash synchronization with Seamless Single Sign-on (SSO)
- Pass-through authentication with Seamless Single Sign-on (SSO)
- Federated SSO (with Active Directory Federation Services (AD FS))

I need to	PHS with SSO	PTA with SSO	AD FS
Sync new user, contact, and group accounts in on-premises Active Directory to the cloud automatically.	x	x	x
Set up my tenant for Office 365 hybrid scenarios.	x	x	x
Enable my users to sign in and access cloud services by using their on-premises password.	x	x	x
Implement single sign-on by using corporate credentials.	x	x	x
Ensure that no passwords are stored in the cloud.		x*	x
Enable on-premises multi-factor authentication solutions.			x

*Through a lightweight agent.

Source: https://docs.microsoft.com/en-us/azure/active-directory/hybrid/plan-connect-user-signin#choosing-the-user-sign-in-method-for-your-organization

Password hash synchronization

With password hash synchronization, hashes of user passwords are synchronized from on-premises Active Directory to Azure AD. When passwords are changed or reset on-premises, the new password hashes are synchronized to Azure AD immediately so that your users can always use the same password for cloud resources and on-premises resources. The passwords are never sent to Azure AD or stored in Azure AD in clear text. You can use password hash synchronization together with password write-back to enable self-service password reset in Azure AD.

In addition, you can enable Seamless SSO for users on domain-joined machines that are on the corporate network. With single sign-on, enabled users only need to enter a username to help them securely access cloud resources.

Source: https://docs.microsoft.com/en-us/azure/active-directory/hybrid/plan-connect-user-signin#password-hash-synchronization

Pass-through authentication

With pass-through authentication, the user's password is validated against the on-premises Active Directory controller. The password doesn't need to be present in Azure AD in any form. This allows for on-premises policies, such as sign-in hour restrictions, to be evaluated during authentication to cloud services.

Pass-through authentication uses a simple agent on a Windows Server 2012 R2 domain-joined machine in the on-premises environment. This agent listens for password validation requests. It doesn't require any inbound ports to be open to the Internet.

In addition, you can also enable single sign-on for users on domain-joined machines that are on the corporate network. With single sign-on, enabled users only need to enter a username to help them securely access cloud resources.

Source: https://docs.microsoft.com/en-us/azure/active-directory/hybrid/plan-connect-user-signin#pass-through-authentication

Federation with a new or existing farm with AD FS

With federated sign-in, your users can sign in to Azure AD-based services with their on-premises passwords. While they're on the corporate network, they don't even have to enter their passwords. By using the federation option with AD FS, you can deploy a new or existing farm with AD FS in Windows Server 2012 R2. If you choose to specify an existing farm, Azure AD Connect configures the trust between your farm and Azure AD so that your users can sign in.

Source: https://docs.microsoft.com/en-us/azure/active-directory/hybrid/plan-connect-user-signin#federation-that-uses-a-new-or-existing-farm-with-ad-fs-in-windows-server-2012-r2

Good practices

Centralize identity management

One important step towards securing your identity is to ensure that IT can manage accounts from one single location regarding where this account was created. While the majority of the enterprises IT organizations will have their primary account directory on-premises, hybrid cloud deployments are on the rise and it is important that you understand how to integrate on-premises and cloud directories and provide a seamless experience to the end user.

To accomplish this hybrid identity scenario we recommend three options:

- Synchronize your on-premises directory with your cloud directory using Azure AD Connect
- Federate your on-premises identity with your cloud directory using Active Directory Federation Services (AD FS)
- Implement Passthrough Authentication (PTA).

Organizations that fail to integrate their on-premises identity with their cloud identity will experience increased administrative overhead in managing accounts, which increases the likelihood of mistakes and security breaches.

Enable Single Sign-On (SSO)

When you have multiple directories to manage, this becomes an administrative problem not only for IT, but also for end users that will have to remember multiple passwords. By using SSO you will provide your users the ability of use the same set of credentials to sign-in and access the resources that they need, regardless where this resource is located on-premises or in the cloud.

Use SSO to enable users to access their SaaS applications based on their organizational account in Azure AD. This is applicable not only for Microsoft SaaS apps, but also other apps, such as Google Apps and Salesforce. Your application can be configured to use Azure AD as a SAML-based identity provider. As a security control, Azure AD will not issue a token allowing them to sign into the application unless they have been granted access using Azure AD. You may grant access directly, or through a group that they are a member of.

The decision to use SSO will impact how you integrate your on-premises directory with your cloud directory. If you want SSO, you will need to use federation, because directory synchronization will only provide same sign-on experience.

Deploy password management

In scenarios where you have multiple tenants, or you want to enable users to reset their own password, it is important that you use appropriate security policies to prevent abuse. In Azure you can leverage the self-service password reset capability and customize the security options to meet your business requirements.

It is particularly important to obtain feedback from these users and learn from their experiences as they try to perform these steps. Based on these experiences, elaborate a plan to mitigate potential issues that may occur during the deployment for a larger group. It is also recommended that you use the Password Reset Registration Activity report to monitor the users that are registering.

Organizations that want to avoid password change support calls but do enable users to reset their own passwords are more susceptible to a higher call volume to the service desk due to password issues. In organizations that have multiple tenants, it is imperative that you implement this type of capability and enable users to perform password reset within security boundaries that were established in the security policy.

Enforce multifactor authentication (MFA)

For organizations that need to be compliant with industry standards, such as PCI DSS version 3.2, multi-factor authentication is a must have capability for authenticate users. Beyond being compliant with industry standards, enforcing MFA to authenticate users can also help organizations to mitigate credential theft type of attack, such as Pass-the-Hash (PtH).

By enabling Azure MFA for your users, you are adding a second layer of security to user sign-ins and transactions. In this case, a transaction might be accessing a document located in a file server or in your SharePoint Online. Azure MFA also helps IT to reduce the likelihood that a compromised credential will have access to organization's data.

User role based access control (RBAC)

See Role based access control (RBAC)

Control locations where resources are created using resource manager

Enabling cloud operators to perform tasks while preventing them from breaking conventions that are needed to manage your organization's resources is very important. Organizations that want to control the locations where resources are created should hard code these locations.

To achieve this, organizations can create security policies that have definitions that describe the actions or resources that are specifically denied. You assign those policy definitions at the desired scope, such as the subscription, resource group, or an individual resource.

You can find more Information regarding Azure Policy in our Chapter on Security

Actively monitor for suspicious activities

According to Verizon 2016 Data Breach report, credential compromise is still in the rise and becoming one of the most profitable businesses for cyber criminals. For this reason, it is important to have an active identity monitor system in place that can quickly detect suspicious behavior activity and trigger an alert for further investigation. Azure AD has two major capabilities that can help organizations monitor their identities: Azure AD Premium anomaly reports and Azure AD identity protection capability.

Make sure to use the anomaly reports to identify attempts to sign in without being traced, brute force attacks against a particular account, attempts to sign in from multiple locations, sign in from infected devices and suspicious IP addresses. Keep in mind that these are reports. In other words, you must have processes and procedures in place for IT admins to run these reports on the daily basis or on demand (usually in an incident response scenario).

In contrast, Azure AD identity protection is an active monitoring system and it will flag the current risks on its own dashboard. Besides that, you will also receive daily summary notifications via email. We recommend that you adjust the risk level according to your business requirements. The risk level for a risk event is an indication (High, Medium, or Low) of the severity of the risk event. The risk level helps Identity Protection users prioritize the actions they must take to reduce the risk to their organization.

Organizations that do not actively monitor their identity systems are at risk of having user credentials compromised. Without knowledge that suspicious activities are taking place using these credentials, organizations won't be able to mitigate this type of threat.

Sources: https://docs.microsoft.com/en-us/azure/active-directory/connect/active-directory-aadconnect

Role based access control (RBAC)

Azure Role-Based Access Control (RBAC) enables fine-grained access management for Azure. Using RBAC, you can grant only the amount of access that users need to perform their jobs. We recommend going with the least privilege security principles.

Within each subscription, you can grant up to 2000 role assignments.

Restricting access based on the **least privilege security principles** is imperative for organizations that want to enforce security policies for data access. Azure Role-Based Access Control (RBAC) can be used to assign permissions to users, groups, and applications at a certain scope. The scope of a role assignment can be a subscription, a resource group, or a single resource.

You can leverage built in RBAC roles in Azure to assign privileges to users. Consider using Storage Account Contributor for cloud operators that need to manage storage accounts. For cloud operators that needs to manage VMs and storage account, consider adding them to Virtual Machine Contributor role.

Organizations that do not enforce data access control by leveraging capabilities such as RBAC may be giving more privileges than necessary to their users. This can lead to data compromise by allow users access to certain types of types of data (e.g., high business impact) that they shouldn't have in the first place.

Security Principal

A security principal is an object that represents a user, group, or service principal that is requesting access to Azure resources.

- **User** - An individual who has a profile in Azure Active Directory. You can also assign roles to users in other tenants. For information about users in other organizations, see Azure Active Directory B2B.
- **Group** - A set of users created in Azure Active Directory. When you assign a role to a group, all users within that group have that role.
- **Service principal** - A security identity used by applications or services to access specific Azure resources. You can think of it as a user identity (username and password or certificate) for an application.

Role definition

A role definition is a collection of permissions. It's sometimes just called a role. A role definition lists the operations that can be performed, such as read, write, and delete. Roles can be high-level, like owner, or specific, like virtual machine reader.

Built-in Roles

- Owner - Has full access to all resources including the right to delegate access to others.
- Contributor - Can create and manage all types of Azure resources but can't grant access to others.
- Reader - Can view existing Azure resources.
- User Access Administrator - Lets you manage user access to Azure resources.

Custom Roles

If the built-in roles don't meet the specific needs of the organization, there can be created custom roles. Just like built-in roles, custom roles can be assigned to users, groups, and service principals at subscription, resource group, and resource scopes. Custom roles are stored in an Azure Active Directory (Azure AD) tenant and can be shared across subscriptions. Each tenant can have up to 2000 custom roles. Custom roles can be created using Azure PowerShell, Azure CLI, or the REST API.

Scope

Scope is the boundary that the access applies to. When you assign a role, you can further limit the actions allowed by defining a scope. This is helpful if you want to make someone a Website Contributor, but only for one resource group.

In Azure, you can specify a scope at multiple levels: subscription, resource group, or resource. Scopes are structured in a parent-child relationship where every child will have only one parent.

Access that you assign at a parent scope is inherited at the child scope. For example:

- If you assign the Reader role to a group at the subscription scope, the members of that group can view every resource group and resource in the subscription.
- If you assign the Contributor role to an application at the resource group scope, it can manage resources of all types in that resource group, but not other resource groups in the subscription.

Azure also includes a scope above subscriptions called management groups (See 6 Management Groups). When you specify scope for RBAC, you can either specify a management group or specify a subscription, resource group, or resource hierarchy.

Assignment

A role assignment is the process of binding a role definition to a user, group, or service principal at a particular scope for the purpose of granting access. Access is granted by creating a role assignment, and access is revoked by removing a role assignment.

The diagram shows an example of a role assignment. In this example, the Marketing group has been assigned the Contributor role for the pharma-sales resource group. This means that users in the Marketing group can create or manage any Azure resource in the pharma-sales resource group. Marketing users do not have access to resources outside the pharma-sales resource group, unless they are part of another role assignment.

Sources: https://docs.microsoft.com/en-us/azure/role-based-access-control/overview

How RBAC determines if a user has access to a resource

The following are the high-level steps that RBAC uses to determine if you have access to a resource on the management plane. This is helpful to understand if you are trying to troubleshoot an access issue.

1. A user (or service principal) acquires a token for Azure Resource Manager.
2. The token includes the user's group memberships (including transitive group memberships).
3. The user makes a REST API call to Azure Resource Manager with the token attached.
4. Azure Resource Manager retrieves all the role assignments and deny assignments that apply to the resource upon which the action is being taken.
5. Azure Resource Manager narrows the role assignments that apply to this user or their group and determines what roles the user has for this resource.
6. Azure Resource Manager determines if the action in the API call is included in the roles the user has for this resource.

7. If the user doesn't have a role with the action at the requested scope, access is not granted. Otherwise, Azure Resource Manager checks if a deny assignment applies.
8. If a deny assignment applies, access is blocked. Otherwise access is granted.

Source: https://docs.microsoft.com/en-us/azure/role-based-access-control/overview#security-principal

Availability in Azure

Planned and unplanned maintenance in Azure

Azure periodically performs updates to improve the reliability, performance, and security of the host infrastructure for virtual machines. These updates range from patching software components in the hosting environment (like operating system, hypervisor, and various agents deployed on the host), upgrading networking components, to hardware decommissioning. The majority of these updates are performed without any impact to the hosted virtual machines. However, there are cases where updates do have an impact:

- If a reboot-less update is possible, Azure uses memory preserving maintenance to pause the VM while the host is updated or the VM is moved to an already updated host altogether.
- If maintenance requires a reboot, you get a notice of when the maintenance is planned. In these cases, you'll also be given a time window where you can start the maintenance yourself, at a time that works for you.

Planned maintenance that requires a reboot, is scheduled in waves. Each wave has different scope (regions).

- A wave starts with a notification to customers. By default, notification is sent to subscription owner and co-owners. You can add more recipients and messaging options like email, SMS, and Webhooks, to the notifications using Azure Activity Log Alerts.
- At the time of the notification, a self-service window is made available. During this window, you can find which of your virtual machines are included in this wave and proactively start maintenance according to your own scheduling needs.
- After the self-service window, a scheduled maintenance window begins. At some point during this window, Azure schedules and applies the required maintenance to your virtual machine.

The goal in having two windows is to give you enough time to start maintenance and reboot your virtual machine while knowing when Azure will automatically start maintenance.

You can use the Azure portal, PowerShell, REST API, and CLI to query for the maintenance windows for your VMs and start self-service maintenance.

Source: https://docs.microsoft.com/en-gb/azure/virtual-machines/windows/maintenance-and-updates

Understand VM Reboots – maintenance vs. downtime

There are three scenarios that can lead to virtual machine in Azure being impacted: unplanned hardware maintenance, unexpected downtime, and planned maintenance.

- **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 will issue an unplanned hardware maintenance event to reduce the impact to the virtual machines hosted on that hardware. **Azure uses Live Migration technology to migrate the Virtual Machines from the failing hardware to a healthy physical machine**. Live Migration is a VM preserving operation that only pauses the Virtual Machine for a short time. Memory, open files, and network connections are maintained, but performance might be reduced before and/or after the event. In cases where Live Migration cannot be used, the VM will experience Unexpected Downtime, as described below.

- **An Unexpected Downtime** is when the hardware or the physical infrastructure for the virtual machine fails unexpectedly. This 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. The attached OS and data disks are always preserved.

Virtual machines can also experience downtime in the unlikely event of an outage or disaster that affects an entire datacenter, or even an entire region. For these scenarios, Azure provides protection options including availability zones and paired regions.

- **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 upon your Virtual Machines or Cloud Services. While the Azure platform attempts to use VM Preserving Maintenance in all possible occasions, there are rare instances when these updates require a reboot of your virtual machine to apply the required updates to the underlying infrastructure. In this case, you can perform Azure Planned Maintenance with Maintenance-Redeploy operation by initiating the maintenance for their VMs in the suitable time window.

Source: https://docs.microsoft.com/en-gb/azure/virtual-machines/linux/manage-availability#understand-vm-reboots—maintenance-vs-downtime

Availability Sets

Overview

An availability set is a logical grouping of VMs within a datacenter that allows Azure to understand how your application is built to provide for redundancy and availability. We recommended that two or more VMs are created within an availability set to provide for a highly available application and to meet the **99.95% Azure SLA**. There is no cost for the Availability Set itself, you only pay for each VM instance that you create. When a single VM is using Azure Premium Storage, the Azure SLA applies for unplanned maintenance events.

An availability set is composed of two additional groupings that protect against hardware failures and allow updates to safely be applied - fault domains (FDs) and update domains (UDs). You can read more about how to manage the availability of Linux VMs or Windows VMs.

Fault domains

A fault domain is a logical group of underlying hardware that share a common power source and network switch, similar to a rack within an on-premises datacenter. As you create VMs within an availability set, the Azure platform automatically distributes your VMs across these fault domains. This approach limits the impact of potential physical hardware failures, network outages, or power interruptions.

Update domains

An update domain is a logical group of underlying hardware that can undergo maintenance or be rebooted at the same time. As you create VMs within an availability set, the Azure platform automatically distributes your VMs across these update domains. This approach ensures that at least one instance of your application always remains running as the Azure platform undergoes periodic maintenance. The order of update domains being rebooted may not proceed sequentially during planned maintenance, but only one update domain is rebooted at a time.

Availability Zones

Overview

Availability Zones is an alternative to an Availability Set a high-availability offering that protects your applications and data from datacenter failures. Availability Zones are unique physical locations within an Azure region. Each zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions. The physical separation of Availability Zones within a region protects applications and data from datacenter failures. Zone-redundant services replicate your applications and data across Availability Zones to protect from single-points-of-failure. **With Availability Zones, Azure offers industry best 99.99% VM uptime SLA**. The full Azure SLA explains the guaranteed availability of Azure as a whole.

An Availability Zone in an Azure region is a combination of a fault domain and an update domain. For example, if you create three or more VMs across three zones in an Azure region, your VMs are effectively distributed across three fault domains and three update domains. The Azure platform recognizes this distribution across update domains to make sure that VMs in different zones are not updated at the same time.

Build high-availability into your application architecture by co-locating your compute, storage, networking, and data resources within a zone and replicating in other zones. Azure services that support Availability Zones fall into two categories:

- Zonal services – you pin the resource to a specific zone (for example, virtual machines, managed disks, IP addresses), or
- Zone-redundant services – platform replicates automatically across zones (for example, zone-redundant storage, SQL Database).

To achieve comprehensive business continuity on Azure, build your application architecture using the combination of Availability Zones with Azure region pairs. You can synchronously replicate your applications and data using Availability Zones within an Azure region for high-availability and asynchronously replicate across Azure regions for disaster recovery protection.

Source: https://docs.microsoft.com/en-us/azure/availability-zones/az-overview

Support of Availability Zones

You can find an up-to-date table with all Services and Regions supporting Availability Zones under the following link: https://docs.microsoft.com/en-us/azure/availability-zones/az-overview#services-support-by-region

Source: https://docs.microsoft.com/en-us/azure/availability-zones/az-overview#pricing

Pricing

There is no additional cost for virtual machines deployed in an Availability Zone. 99.99% VM uptime SLA is offered when two or more VMs are deployed across two or more Availability Zones within an Azure region. There will be additional inter-Availability Zone VM-to-VM data transfer charges.

Source: https://docs.microsoft.com/en-us/azure/availability-zones/az-overview#pricing

Storage

General Purpose v2

General-purpose v2 (GPv2) accounts are storage accounts that support all of the latest features for blobs, files, queues, and tables. GPv2 accounts support all APIs and features supported in GPv1 and Blob storage accounts. They also support the same durability, availability, scalability, and performance features in those

account types. Pricing for GPv2 accounts has been designed to deliver the lowest per gigabyte prices, and industry competitive transaction prices.

For block blobs in a GPv2 storage account, you can choose between hot and cool storage tiers at the account level, or hot, cool, and archive tiers at the blob level based on access patterns. Store frequently, infrequently, and rarely accessed data in the hot, cool, and archive storage tiers respectively to optimize costs.

GPv2 storage accounts expose the Access Tier attribute at the account level, which specifies the default storage account tier as Hot or Cool. The default storage account tier is applied to any blob that does not have an explicit tier set at the blob level. If there is a change in the usage pattern of your data, you can also switch between these storage tiers at any time. The archive tier can only be applied at the blob level.

Recommendations

For applications requiring only block or append blob storage, using GPv2 storage accounts is recommended, to take advantage of the differentiated pricing model of tiered storage. However, you may want to use GPv1 in certain scenarios, such as:

- You still need to use the classic deployment model. Blob storage accounts are only available via the Azure Resource Manager deployment model.
- You use high volumes of transactions or geo-replication bandwidth, both of which cost more in GPv2 and Blob storage accounts than in GPv1, and don't have enough storage that benefits from the lower costs of GB storage.
- You use a version of the Storage Services REST API that is earlier than 2014-02-14 or a client library with a version lower than 4.x and cannot upgrade your application.
- You can apply the tiering individually to each object in the BLOB storage.

Pricing guides: - Hot is up to 80% more expensive than cool storage - Hot is nearly free for read, while cool costs more for read - Both hot and cool are nearly free to write

General Purpose Storage V1 vs V2:

- General Purpose V2 is up to 40% more expensive than V1 storage.
- V1 doesn't offer tiering, which can save a lot of money
- However, General Purpose V1 will be retired in the coming years, so just use V2

Replication

The data in your Microsoft Azure storage account is always replicated to ensure durability and high availability. Replication copies your data so that it is protected from transient hardware failures, preserving your application up-time.

You can choose to replicate your data within the same data center, across data centers within the same region, or across regions. If your data is replicated across multiple data centers or across regions, it's also protected from a catastrophic failure in a single location.

Replication ensures that your storage account meets the Service-Level Agreement (SLA) for Storage even in the face of failures. See the SLA for information about Azure Storage guarantees for durability and availability.

When you create a storage account, you can select one of the following replication options:

- Locally redundant storage (LRS)
- Zone-redundant storage (ZRS)
- Geo-redundant storage (GRS)
- Read-access geo-redundant storage (RA-GRS)

Scenario	LRS	ZRS	GRS	RA-GRS
Node unavailability within a data center	Yes	Yes	Yes	Yes
An entire data center (zonal or non-zonal) becomes unavailable	No	Yes	Yes	Yes
A region-wide outage	No	No	Yes	Yes
Read access to your data (in a remote, geo-replicated region) in the event of region-wide unavailability	No	No	No	Yes
Designed to provide _ durability of objects over a given year	at least 99.999999999% (11 9's)	at least 99.9999999999% (12 9's)	at least 99.99999999999999% (16 9's)	at least 99.99999999999999% (16 9's)
Supported storage account types	GPv1, GPv2, Blob	GPv2	GPv1, GPv2, Blob	GPv1, GPv2, Blob

Storage replication is not intended as backup (deletions will be propagated immediately), but for availability. In this context, GRS ans RA-GRS are defined:

- GRS replicates your data to another data center in a secondary region, but that data is available to be read only if Microsoft initiates a failover from the primary to secondary region.
- Read-access geo-redundant storage (RA-GRS) is based on GRS. RA-GRS replicates your data to another data center in a secondary region, and also provides you with the option to read from the secondary region. With RA-GRS, you can read from the secondary region regardless of whether Microsoft initiates a failover from the primary to secondary region.

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy-grs

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy

Locally redundant storage

Locally redundant storage (LRS) is designed to provide at least 99.999999999% (11 9's) durability of objects over a given year by replicating your data within a storage scale unit, which is hosted in a datacenter in the region in which you created your storage account. A write request returns successfully only once it has been written to all replicas. These replicas each reside in separate fault domains and update domains within one storage scale unit.

A storage scale unit is a collection of racks of storage nodes. A fault domain (FD) is a group of nodes that represent a physical unit of failure and can be considered as nodes belonging to the same physical rack. An upgrade domain (UD) is a group of nodes that are upgraded together during the process of a service upgrade (rollout). The replicas are spread across UDs and FDs within one storage scale unit to ensure that data is available even if hardware failure impacts a single rack or when nodes are upgraded during a rollout.

LRS is the lowest cost option and offers least durability compared to other options. In the event of a datacenter level disaster (fire, flooding etc.) all replicas might be lost or unrecoverable. To mitigate this risk, Geo Redundant Storage (GRS) is recommended for most applications.

Locally redundant storage may still be desirable in certain scenarios:

- Provides highest maximum bandwidth of Azure Storage replication options.
- If your application stores data that can be easily reconstructed, you may opt for LRS.
- Some applications are restricted to replicating data only within a country due to data governance requirements. A paired region could be in another country. For more information on region pairs, see Azure regions.

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy-lrs

Zone redundant storage

Zone redundant storage (ZRS) is designed to simplify the development of highly available applications. ZRS provides durability for storage objects of at least 99.9999999999% (12 9's) over a given year. ZRS replicates your data synchronously across multiple availability zones. Consider ZRS for scenarios like transactional applications where downtime is not acceptable.

ZRS enables customers to read and write data even if a single zone is unavailable or unrecoverable. Inserts and updates to data are made synchronously and are strongly consistent.

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy-zrs

Geo-redundant storage

Geo-redundant storage (GRS) is designed to provide at least 99.99999999999999% (16 9's) durability of objects over a given year by replicating your data to a secondary region that is hundreds of miles away from the primary region. If your storage account has GRS enabled, then your data is durable even in the case of a complete regional outage or a disaster in which the primary region is not recoverable.

For a storage account with GRS enabled, an update is first committed to the primary region. Then the update is replicated asynchronously to the secondary region, where it is also replicated.

With GRS, both the primary and secondary regions manage replicas across separate fault domains and upgrade domains within a storage scale unit as described with LRS.

Considerations:

- Since asynchronous replication involves a delay, in the event of a regional disaster it is possible that changes that have not yet been replicated to the secondary region will be lost if the data cannot be recovered from the primary region.
- The replica is not available unless Microsoft initiates failover to the secondary region. If Microsoft does initiate a failover to the secondary region, you will have read and write access to that data after the failover has completed. For more information, please see Disaster Recovery Guidance.
- If an application wants to read from the secondary region, the user should enable RA-GRS.

When you create a storage account, you select the primary region for the account. The secondary region is determined based on the primary region and cannot be changed. You can find all region pairings, as well as more information regarding paired regions in the following Microsoft Article: https://docs.microsoft.com/en-us/azure/best-practices-availability-paired-regions

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy-grs

Read-access geo-redundant storage

Read-access geo-redundant storage (RA-GRS) maximizes availability for your storage account. RA-GRS provides read-only access to the data in the secondary location, in addition to geo-replication across two regions.

When you enable read-only access to your data in the secondary region, your data is available on a secondary endpoint as well as on the primary endpoint for your storage account. The secondary endpoint is similar to the primary endpoint but appends the suffix –secondary to the account name. For example, if your primary endpoint for the Blob service is myaccount.blob.core.windows.net, then your secondary endpoint is myaccount-secondary.blob.core.windows.net. The access keys for your storage account are the same for both the primary and secondary endpoints.

Some considerations to keep in mind when using RA-GRS:

- Your application has to manage which endpoint it is interacting with when using RA-GRS.

- Since asynchronous replication involves a delay, changes that have not yet been replicated to the secondary region may be lost if data cannot be recovered from the primary region, for example in the event of a regional disaster.
- If Microsoft initiates failover to the secondary region, you will have read and write access to that data after the failover has completed.
- RA-GRS is intended for high-availability purposes.

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy?toc=%2fazure%2fstorage%2fblobs%2ft

VM Disk Storage

Just like any other computer, virtual machines in Azure use disks as a place to store an operating system, applications, and data. All Azure virtual machines have at least two disks – a Windows operating system disk and a temporary disk. The operating system disk is created from an image, and both the operating system disk and the image are virtual hard disks (VHDs) stored in an Azure storage account. Virtual machines also can have one or more data disks, that are also stored as VHDs.

Operating system disk

Every virtual machine has one attached operating system disk. It's registered as a SATA drive and labeled as the C: drive by default. This disk has a maximum capacity of 2048 gigabytes (GB).

Temporary disk

Each VM contains a temporary disk. The temporary disk provides short-term storage for applications and processes and is intended to only store data such as page or swap files. Data on the temporary disk may be lost during a maintenance event or when you redeploy a VM. During a standard reboot of the VM, the data on the temporary drive should persist.

The temporary disk is labeled as the D: drive by default and it used for storing pagefile.sys. The size of the temporary disk varies, based on the size of the virtual machine.

Data disk

A data disk is a VHD 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 that you choose. The size of the virtual machine determines how many data disks you can attach to it and the type of storage you can use to host the disks.

Azure creates an operating system disk when you create a virtual machine from an image. If you use an image that includes data disks, Azure also creates the data disks when it creates the virtual machine. Otherwise, you add data disks after you create the virtual machine.

You can add data disks to a virtual machine at any time, by attaching the disk to the virtual machine. You can use a VHD that you've uploaded or copied to your storage account or use an empty VHD that Azure creates for you. Attaching a data disk associates the VHD file with the VM by placing a 'lease' on the VHD so it can't be deleted from storage while it's still attached.

VHDs

The VHDs used in Azure are .vhd files stored as page blobs in a standard or premium storage account in Azure.

Azure supports the fixed disk VHD format. The fixed format lays the logical disk out linearly within the file, so that disk offset X is stored at blob offset X. A small footer at the end of the blob describes the properties of the VHD. Often, the fixed format wastes space because most disks have large unused ranges in them. However, Azure stores .vhd files in a sparse format, so you receive the benefits of both the fixed and dynamic disks at the same time.

All .vhd files in Azure that you want to use as a source to create disks or images are read-only, except the .vhd files uploaded or copied to Azure storage by the user (which can be either read-write or read-only). When you create a disk or image, Azure makes copies of the source .vhd files. These copies can be read-only or read-and-write, depending on how you use the VHD.

When you create a virtual machine from an image, Azure creates a disk for the virtual machine that is a copy of the source .vhd file. To protect against accidental deletion, Azure places a lease on any source .vhd file that's used to create an image, an operating system disk, or a data disk.

Before you can delete a source .vhd file, you'll need to remove the lease by deleting the disk or image. To delete a .vhd file that is being used by a virtual machine as an operating system disk, you can delete the virtual machine, the operating system disk, and the source .vhd file all at once by deleting the virtual machine and deleting all associated disks. However, deleting a .vhd file that's a source for a data disk requires several steps in a set order. First you detach the disk from the virtual machine, then delete the disk, and then delete the .vhd file.

Don't delete a source .vhd file from storage or the storage account itself – Microsoft can't recover that data for you.

Premium vs. Standard storage

Azure Disks are designed for 99.999% availability. Azure Disks have consistently delivered enterprise-grade durability, with an industry-leading ZERO% Annualized Failure Rate.

There are two performance tiers for storage that you can choose from when creating your disks – Standard Storage and Premium Storage. Also, there are two types of disks – unmanaged and managed – and they can reside in either performance tier.

Standard storage (HDD)

Standard Storage is backed by HDDs and delivers cost-effective storage while still being performant. Standard storage can be replicated locally in one datacenter or be geo-redundant with primary and secondary data centers.

Azure Standard Storage delivers reliable, low-cost disk support for VMs running latency-insensitive workloads. It also supports blobs, tables, queues, and files. With Standard Storage, the data is stored on hard disk drives (HDDs). When working with VMs, you can use standard storage disks for Dev/Test scenarios and less critical workloads, and premium storage disks for mission-critical production applications. Standard Storage is available in all Azure regions.

Standard storage (SSD)

Azure Standard Solid State Drives (SSD) Managed Disks are a **cost-effective storage option** optimized for workloads that need consistent performance at lower IOPS levels. Standard SSD offers a good entry level experience for those who wish to move to the cloud, especially if you experience issues with the variance of workloads running on your HDD solutions on premises. Standard SSDs deliver better availability, consistency, reliability and latency compared to HDD Disks, and are suitable for Web servers, low IOPS application servers, lightly used enterprise applications, and Dev/Test workloads.

Managed Disks: Standard SSDs are only available as Managed Disks. Unmanaged Disks and Page Blobs are not supported on Standard SSD. While creating the Managed Disk, you specify the disk type as Standard SSD and indicate the size of disk you need, and Azure creates and manages the disk for you. Standard SSDs support all service management operations offered by Managed Disks. For example, you can create, copy or snapshot Standard SSD Managed Disks in the same way you do with Managed Disks.

Virtual Machines: Standard SSDs can be used with all Azure VMs, including the VM types that do not support Premium Disks. For example, if you're using an A-series VM, or N-series VM, or DS-series, or any other Azure VM series, you can use Standard SSDs with that VM. With the introduction of Standard SSD,

we are enabling a broad range of workloads that previously used HDD-based disks to transition to SSD-based disks, and experience the consistent performance, higher availability, better latency, and an overall better experience that come with SSDs.

Highly durable and available: Standard SSDs are built on the same Azure Disks platform, which has consistently delivered high availability and durability for disks. Azure Disks are designed for 99.999 percent availability. Like all Managed Disks, Standard SSDs will also offer Local Redundant Storage (LRS). With LRS, the platform maintains multiple replicas of data for every disk and has consistently delivered enterprise-grade durability for IaaS disks, with an industry-leading ZERO percent Annualized Failure Rate.

Snapshots: Like all Managed Disks, Standard SSDs also support creation of Snapshots. Snapshot type can be either Standard (HDD) or Premium (SSD). For cost saving, we recommend Snapshot type of Standard (HDD) for all Azure disk types. This is because when you create a managed disk from a snapshot, you're always able to choose a higher tier such as Standard SSD or Premium SSD.

Source: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/standard-storage

Premium storage (SSD)

Premium Storage is backed by SSDs, and delivers high-performance, low-latency disk support for VMs running I/O-intensive workloads. You can use Premium Storage with DS, DSv2, GS, Ls, or FS series Azure VMs.

Azure Premium Storage delivers high-performance, low-latency disk support for virtual machines (VMs) with input/output (I/O)-intensive workloads. VM disks that use Premium Storage store data on solid-state drives (SSDs). To take advantage of the speed and performance of premium storage disks, you can migrate existing VM disks to Premium Storage.

In Azure, you can attach several premium storage disks to a VM. Using multiple disks gives your applications up to 256 TB of storage per VM. With Premium Storage, your applications can achieve 80,000 I/O operations per second (IOPS) per VM, and a disk throughput of up to 2,000 megabytes per second (MB/s) per VM. Read operations give you very low latencies.

With Premium Storage, Azure offers the ability to truly lift-and-shift demanding enterprise applications like Dynamics AX, Dynamics CRM, Exchange Server, SAP Business Suite, and SharePoint farms to the cloud. You can run performance-intensive database workloads in applications like SQL Server, Oracle, MongoDB, MySQL, and Redis, which require consistent high performance and low latency.

For the best performance for your application, we recommend that you migrate any VM disk that requires high IOPS to Premium Storage. If your disk does not require high IOPS, you can help limit costs by keeping it in standard Azure Storage. In standard storage, VM disk data is stored on hard disk drives (HDDs) instead of on SSDs.

Source: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/premium-storage

Azure ultra SSD Storage

Azure ultra solid state drives (SSD) (preview) offer high throughput, high IOPS, and consistent low latency disk storage for Azure IaaS virtual machines (VMs). This new offering provides top of the line performance at the same availability levels as our existing disks offerings. One major benefit of ultra SSDs is the ability to dynamically change the performance of the SSD along with your workloads without the need to restart your VMs. Ultra SSDs are suited for data-intensive workloads such as SAP HANA, top tier databases, and transaction-heavy workloads.

Currently, ultra SSDs are in preview and you must enroll in the preview in order to access them.

Source: https://docs.microsoft.com/de-de/azure/virtual-machines/windows/disks-enable-ultra-ssd

Unmanaged disks

Unmanaged disks are the traditional type of disks that have been used by VMs. With these, you create your own storage account and specify that storage account when you create the disk. You have to make sure you don't put too many disks in the same storage account, because you could exceed the scalability targets of the storage account (20,000 IOPS, for example), resulting in the VMs being throttled. With unmanaged disks, you have to figure out how to maximize the use of one or more storage accounts to get the best performance out of your VMs.

Managed disks

Managed Disks handles the storage account creation/management in the background for you and ensures that you do not have to worry about the scalability limits of the storage account. You simply specify the disk size and the performance tier (Standard/Premium), and Azure creates and manages the disk for you. Even as you add disks or scale the VM up and down, you don't have to worry about the storage being used.

You can also manage your custom images in one storage account per Azure region and use them to create hundreds of VMs in the same subscription.

Azure Managed Disks simplifies disk management for Azure IaaS VMs by managing the storage accounts associated with the VM disks. You only have to specify the type (Premium or Standard) and the size of disk you need, and Azure creates and manages the disk for you.

Managed Disks handles storage for you behind the scenes. Previously, you had to create storage accounts to hold the disks (VHD files) for your Azure VMs. When scaling up, you had to make sure you created additional storage accounts so you didn't exceed the IOPS limit for storage with any of your disks. With Managed Disks handling storage, you are no longer limited by the storage account limits (such as 20,000 IOPS / account). You also no longer have to copy your custom images (VHD files) to multiple storage accounts. You can manage them in a central location – one storage account per Azure region – and use them to create hundreds of VMs in a subscription.

Managed Disks will allow you to create up to 10,000 VM disks in a subscription, which will enable you to create thousands of VMs in a single subscription. This feature also further increases the scalability of Virtual Machine Scale Sets (VMSS) by allowing you to create up to a thousand VMs in a VMSS using a Marketplace image.

Managed Disks provides better reliability for Availability Sets by ensuring that the disks of VMs in an Availability Set are sufficiently isolated from each other to avoid single points of failure. It does this by automatically placing the disks in different storage scale units (stamps). If a stamp fails due to hardware or software failure, only the VM instances with disks on those stamps fail. For example, let's say you have an application running on five VMs, and the VMs are in an Availability Set. The disks for those VMs won't all be stored in the same stamp, so if one stamp goes down, the other instances of the application continue to run.

Azure Disks are designed for 99.999% availability. Rest easier knowing that you have three replicas of your data that enables high durability. If one or even two replicas experience issues, the remaining replicas help ensure persistence of your data and high tolerance against failures. This architecture has helped Azure consistently deliver enterprise-grade durability for IaaS disks, with an industry-leading ZERO% Annualized Failure Rate.

You can use Azure Role-Based Access Control (RBAC) to assign specific permissions for a managed disk to one or more users. Managed Disks exposes a variety of operations, including read, write (create/update), delete, and retrieving a shared access signature (SAS) URI for the disk. You can grant access to only the operations a person needs to perform his job. For example, if you don't want a person to copy a managed disk to a storage account, you can choose not to grant access to the export action for that managed disk. Similarly, if you don't want a person to use an SAS URI to copy a managed disk, you can choose not to grant that permission to the managed disk.

Use Azure Backup service with Managed Disks to create a backup job with time-based backups, easy VM restoration and backup retention policies. **Managed Disks only support Locally Redundant Storage (LRS) as the replication option**; this means it keeps three copies of the data within a single region. For regional disaster recovery, you must backup your VM disks in a different region using Azure Backup service and a GRS storage account as backup vault. Currently Azure Backup supports data disk sizes up to 1TB for backup. Read more about this at Using Azure Backup service for VMs with Managed Disks.

We recommend that you use Azure Managed Disks for new VMs, and that you convert your previous unmanaged disks to managed disks, to take advantage of the many features available in Managed Disks.

Comparison

You can find an up-to-date comparison between the available storage disks in Azure under the following link: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/disks-types

Available disk sizes

You can find more information for all available managed disks in Azure, as well as tables listing the available disk sizes under the following links:

Premium SSD: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/disks-types#premium-ssd

Standard SSD: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/disks-types#standard-ssd

Standard HDD: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/disks-types#standard-hdd

Storage Firewalls and Virtual Networks

Azure Storage provides a layered security model allowing you to secure your storage accounts to a specific set of allowed networks. When network rules are configured, only applications from allowed networks can access a storage account. When calling from an allowed network, applications continue to require proper authorization (a valid access key or SAS token) to access the storage account.

Turning on Firewall rules for your Storage account will block access to incoming requests for data, including from other Azure services. This includes using the Portal, writing logs, etc. For participating services you can re-enable functionality through the Exceptions section below. To access the Portal you would need to do so from a machine within the trusted boundary (either IP or VNet) that you have set up.

Storage accounts can be configured to deny access to traffic from all networks (including internet traffic) by default. Access can be granted to traffic from specific Azure Virtual networks, allowing you to build a secure network boundary for your applications. Access can also be granted to public internet IP address ranges, enabling connections from specific internet or on-premises clients.

Network rules are enforced on all network protocols to Azure storage, including REST and SMB. Access to your data from tools like the Azure portal, Storage Explorer, and AZCopy require explicit network rules granting access when network rules are in force.

Network rules can be applied to existing Storage accounts, or can be applied during the creation of new Storage accounts.

Once network rules are applied, they are enforced for all requests. SAS tokens that grant access to a specific IP Address service serve to limit the access of the token holder, but do not grant new access beyond configured network rules.

Virtual Machine Disk traffic (including mount and unmount operations, and disk IO) is not affected by network rules. REST access to page blobs is protected by network rules.

Backup and Restore of Virtual Machines using unmanaged disks in storage accounts with network rules applied is not currently supported.

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-network-security

Exceptions

While network rules can enable a secure network configuration for most scenarios, there are some cases where exceptions must be granted to enable full functionality. Storage accounts can be configured with exceptions for Trusted Microsoft services, and for access to Storage analytics data.

Source: https://docs.microsoft.com/en-us/azure/storage/common/storage-network-security#exceptions # Azure Virtual Machines

Overview

Azure Virtual Machines (VM) is one of several types of on-demand, scalable computing resources that Azure offers. Typically, you choose a VM when you need more control over the computing environment than the other choices offer.

An Azure VM gives you the flexibility of virtualization without having to buy and maintain the physical hardware that runs it. However, you still need to maintain the VM by performing tasks, such as configuring, patching, and installing the software that runs on it.

Azure virtual machines can be used in various ways. Some examples are:

- **Development and test** – Azure VMs offer a quick and easy way to create a computer with specific configurations required to code and test an application.
- **Applications in the cloud** – Because demand for your application can fluctuate, it might make economic sense to run it on a VM in Azure. You pay for extra VMs when you need them and shut them down when you don't.
- **Extended datacenter** – Virtual machines in an Azure virtual network can easily be connected to your organization's network.

The number of VMs that your application uses can scale up and out to whatever is required to meet your needs.

Source: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/overview

Azure Virtual Machines

Types and Sizes

Azure VMs are available in several different types and sizes. These sizes define the resources available to the VM, like vCPU and Memory, but also what limitations the machine is subjected to, like the amount of NICs or data disks assignable to the VM.

To cover the broad range of possible applications virtual machines can be used for, Azure offers a wide amount of possible VM-sizes to its consumers. Each size offers a different arrangement of resources and limitations. The following table sorts the available VM-sizes in six categories and provides a short explanation on what the intended application-purpose is for the given category.

Type	Sizes	Description
General purpose	B, Dsv3, Dv3, DSv2, Dv2, Av2, DC	Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	Fsv2	High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	Esv3, Ev3, Mv2, M, DSv2, Dv2	High memory-to-CPU ratio. Great for relational database servers, medium to large caches, and in-memory analytics.

Type	Sizes	Description
Storage optimized	Lsv2	High disk throughput and IO ideal for Big Data, SQL, NoSQL databases, data warehousing and large transactional databases.
GPU	NC, NCv2, NCv3, ND, NDv2 (Preview), NV, NVv3 (Preview)	Specialized virtual machines targeted for heavy graphic rendering and video editing, as well as model training and inferencing (ND) with deep learning. Available with single or multiple GPUs.
High performance compute	HB, HC, H	Our fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA).

Source: https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sizes

Operating Systems

Azure provides a variety of OS images that you can install into the VM, including several versions of Windows and Linux. The choice of OS will influence your hourly compute pricing as Azure bundles the cost of the OS license into the price.

Additionally, if you are looking for more than just base OS images, you can search the Azure Marketplace for more sophisticated install images that include the OS and popular software tools installed for specific scenarios. For example, if you needed a new WordPress site instead of setting up and configuring each component, you can leverage a Marketplace image and install the required stack all at once.

Finally, if you can't find a suitable OS image, you can create your disk image with what you need, upload it to Azure storage, and use it to create an Azure VM. It is to note however, that Azure only supports 64-bit operating systems.

Azure Hybrid Benefit For customers with Software Assurance, Azure Hybrid Benefit for Windows Server allows you to use your on-premises Windows Server licenses and run Windows virtual machines on Azure at a reduced cost.

Source: Microsoft AZ-100.3, Deploying and Managing Virtual Machines

VM Scale Sets

Overview

Virtual machine scale sets are an Azure Compute resource you can use to deploy and manage a set of identical VMs. With all VMs configured the same, VM scale sets are designed to support true auto-scaling and as such makes it easier to build large-scale services targeting big compute, big data, and containerized workloads. So, as demand goes up more virtual machine instances can be added, and as demand goes down virtual machines instances can be removed. The process can be manual or automated or a combination of both.

Scale sets works in a way that provides many benefits.

- All VM instances are created from the same base OS image and configuration. This approach lets you easily manage hundreds of VMs without additional configuration tasks or network management.
- Scale sets support the use of the Azure load balancer for basic layer-4 traffic distribution, and Azure Application Gateway for more advanced layer-7 traffic distribution and SSL termination.
- Scale sets are used to run multiple instances of your application. If one of these VM instances has a problem, customers continue to access your application through one of the other VM instances with minimal interruption.

- Customer demand for your application may change throughout the day or week. To match customer demand, scale sets can automatically increase the number of VM instances as application demand increases, then reduce the number of VM instances as demand decreases.
- Scale sets support up to 1,000 VM instances. If you create and upload your own custom VM images, the limit is 300 VM instances.

Differences between Virtual Machines and Scale Sets

Scale sets are built from virtual machines. With scale sets, the management and automation layers are provided to run and scale your applications. You could instead manually create and manage individual VMs, or integrate existing tools to build a similar level of automation. The following table outlines the benefits of scale sets compared to manually managing multiple VM instances.

Scenario	Manual group of VMs	Virtual machine scale set
Add additional VM instances	Manual process to create, configure, and ensure compliance	Automatically create from central configuration
Traffic balancing and distribution	Manual process to create and configure Azure load balancer or Application Gateway	Can automatically create and integrate with Azure load balancer or Application Gateway
High availability and redundancy	Manually create Availability Set or distribute and track VMs across Availability Zones	Automatic distribution of VM instances across Availability Zones or Availability Sets
Scaling of VMs	Manual monitoring and Azure Automation	Autoscale based on host metrics, in-guest metrics, Application Insights, or schedule

There is no additional cost to scale sets. You only pay for the underlying compute resources such as the VM instances, load balancer, or Managed Disk storage. The management and automation features, such as autoscale and redundancy, incur no additional charges over the use of VMs.

Source: https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/overview?toc=%2fazure%2fvirtual-machines%2fwindows%2ftoc.json

Extensions

Azure virtual machine extensions are small applications that provide post-deployment configuration and automation tasks on Azure VMs. For example, if a virtual machine requires software installation, anti-virus protection, or a configuration script inside, a VM extension can be used. Extensions are all about managing your virtual machines.

Azure VM extensions can be:

- Managed with Azure CLI, PowerShell, Azure Resource Manager templates, and the Azure portal.
- Bundled with a new VM deployment or run against any existing system. For example, they can be part of a larger deployment, configuring applications on VM provision, or run against any supported extension operated systems post deployment.

Azure VM Agent

To handle the extension on the VM, you need the Azure Windows Agent installed. The Azure VM agent manages interactions between an Azure VM and the Azure fabric controller. The VM agent is responsible for many functional aspects of deploying and managing Azure VMs, including running VM extensions. The Azure VM agent is preinstalled on Azure Marketplace images, and can be installed manually on supported operating systems.

Source: https://docs.microsoft.com/en-us/azure/virtual-machines/extensions/overview

Container

Azure Container Instances

Containers are becoming the preferred way to package, deploy, and manage cloud applications. Azure Container Instances offers the fastest and simplest way to run a container in Azure, without having to manage any virtual machines and without having to adopt a higher-level service.

Azure Container Instances are a great solution for small-scale tasks like task automation, build jobs or simple applications. For large-scale operations where you require full container orchestration, automatic scaling or coordinated application upgrades, the Azure Kubernetes Service is the recommended solution. You can find more information on that further below.

Source: https://docs.microsoft.com/en-us/azure/container-instances/container-instances-overview

Features

- **Faster startup times** Azure Container Instances can start containers in Azure in seconds, without the need to provision and manage VMs.
- **Public IP connectivity and DNS Name** Azure Container Instances enables exposing your containers directly to the internet with an IP address and a fully qualified domain name. When you create a container instance, you can specify a custom DNS name label so your application is reachable at *customlabel.azureregion*.azurecontainer.io.
- **Hypervisor-level security** Historically, containers have offered application dependency isolation and resource governance but have not been considered sufficiently hardened for hostile multi-tenant usage. Azure Container Instances guarantees your application is as isolated in a container as it would be in a VM.
- **Custom sizes** Containers are typically optimized to run just a single application, but the exact needs of those applications can differ greatly. Azure Container Instances provides optimum utilization by allowing exact specifications of CPU cores and memory. You pay based on what you need and get billed by the second, so you can fine-tune your spending based on actual need.
- **Persistent storage** To retrieve and persist state with Azure Container Instances, we offer direct mounting of Azure Files shares.
- **Linux and Windows containers** Azure Container Instances can schedule both Windows and Linux containers with the same API. Simply specify the OS type when you create your container groups.

Source: https://docs.microsoft.com/en-us/azure/container-instances/container-instances-overview

Securing your Azure Container Instances

- **Use a private registry** Containers are built from images that are stored in one or more repositories. These repositories can belong to a public registry, like Docker Hub, or to a private registry. An example of a private registry is the Docker Trusted Registry, which can be installed on-premises or in a virtual private cloud. You can also use cloud-based private container registry services, including Azure Container Registry.

  A publicly available container image does not guarantee security. Container images consist of multiple software layers, and each software layer might have vulnerabilities. To help reduce the threat of attacks, you should store and retrieve images from a private registry, such as Azure Container Registry or Docker Trusted Registry. In addition to providing a managed private registry, Azure Container Registry supports service principal-based authentication through Azure Active Directory for basic authentication flows. This authentication includes role-based access for read-only (pull), write (push), and owner permissions.

- **Monitor and scan container images** Security monitoring and scanning solutions such as Twistlock and Aqua Security are available through the Azure Marketplace. You can use them to scan container images in a private registry and identify potential vulnerabilities.
- **Protect credentials** Containers can spread across several clusters and Azure regions. So, you must secure credentials required for logins or API access, such as passwords or tokens. Ensure that only privileged users can access those containers in transit and at rest. Inventory all credential secrets, and then require developers to use emerging secrets-management tools that are designed for container platforms. Make sure that your solution includes encrypted databases, TLS encryption for secrets data in transit, and least-privilege role-based access control. Azure Key Vault is a cloud service that safeguards encryption keys and secrets for containerized applications. Because this data is sensitive and business critical, secure access to your key vaults so that only authorized applications and users can access them.

Source: https://docs.microsoft.com/en-us/azure/container-instances/container-instances-image-security

Azure Container Registry

Azure Container Registry is a managed Docker registry service based on the open-source Docker Registry 2.0. Create and maintain Azure container registries to store and manage your private Docker container images.

Use container registries in Azure with your existing container development and deployment pipelines, or use ACR Tasks to build container images in Azure. Build on demand, or fully automate builds with source code commit and base image update build triggers.

Source: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-intro

Service Tiers

Azure Container Registry (ACR) is available in multiple service tiers, known as SKUs. These SKUs provide predictable pricing and several options for aligning to the capacity and usage patterns of your private Docker registry in Azure.

SKU	Managed	Description
Basic	Yes	A cost-optimized entry point for developers learning about Azure Container Registry. Basic registries have the same programmatic capabilities as Standard and Premium (such as Azure Active Directory authentication integration, image deletion, and web hooks. However, the included storage and image throughput are most appropriate for lower usage scenarios.
Standard	Yes	Standard registries offer the same capabilities as Basic, with increased included storage and image throughput. Standard registries should satisfy the needs of most production scenarios.
Premium	Yes	Premium registries provide the highest amount of included storage and concurrent operations, enabling high-volume scenarios. In addition to higher image throughput, Premium adds features including geo-replication for managing a single registry across multiple regions, content trust for image tag signing, and firewalls and virtual networks (preview) to restrict access to the registry.

The Basic, Standard, and Premium SKUs (collectively called managed registries) all provide the same programmatic capabilities. They also all benefit from image storage managed entirely by Azure. Choosing a higher-level SKU provides more performance and scale. With multiple service tiers, you can get started with Basic, then convert to Standard and Premium as your registry usage increases.

Source: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-skus

Best Practices

Network-close deployment Create your container registry in the same Azure region in which you deploy containers. Placing your registry in a region that is network-close to your container hosts can help lower both latency and cost.

Network-close deployment is one of the primary reasons for using a private container registry. Docker images have an efficient layering construct that allows for incremental deployments. However, new nodes need to pull all layers required for a given image. This initial `docker pull` can quickly add up to multiple gigabytes. Having a private registry close to your deployment minimizes the network latency. Additionally, all public clouds, Azure included, implement network egress fees. Pulling images from one datacenter to another adds network egress fees, in addition to the latency.

Geo-replicate multi-region deployments Use Azure Container Registry's geo-replication feature if you're deploying containers to multiple regions. Whether you're serving global customers from local datacenters or your development team is in different locations, you can simplify registry management and minimize latency by geo-replicating your registry. Geo-replication is available only with Premium registries.

To learn how to use geo-replication, see the three-part tutorial, Geo-replication in Azure Container Registry.

Repository namespaces By leveraging repository namespaces, you can allow sharing a single registry across multiple groups within your organization. Registries can be shared across deployments and teams. Azure Container Registry supports nested namespaces, enabling group isolation.

For example, consider the following container image tags. Images that are used corporate-wide, like `aspnetcore`, are placed in the root namespace, while container images owned by the Production and Marketing groups each use their own namespaces.

Dedicated resource group Because container registries are resources that are used across multiple container hosts, a registry should reside in its own resource group.

Although you might experiment with a specific host type, such as Azure Container Instances, you'll likely want to delete the container instance when you're done. However, you might also want to keep the collection of images you pushed to Azure Container Registry. By placing your registry in its own resource group, you minimize the risk of accidentally deleting the collection of images in the registry when you delete the container instance resource group.

Authentication When authenticating with an Azure container registry, there are two primary scenarios: individual authentication, and service (or "headless") authentication. The following table provides a brief overview of these scenarios, and the recommended method of authentication for each.

Type	Example scenario	Recommended method
Individual identity	A developer pulling images to or pushing images from their development machine.	az acr login
Headless/service identity	Build and deployment pipelines where the user isn't directly involved.	Service principal

For in-depth information about Azure Container Registry authentication, see Authenticate with an Azure container registry.

Manage registry size The storage constraints of each container registry SKU are intended to align with a typical scenario: **Basic** for getting started, **Standard** for the majority of production applications, and **Premium** for hyper-scale performance and geo-replication. Throughout the life of your registry, you should manage its size by periodically deleting unused content.

You can use the Azure CLI command `az acr show-usage` to display the current size of your registry or find the current storage used in the Overview of your registry in the Azure Portal.

Source: https://docs.microsoft.com/en-us/azure/container-registry/container-registry-best-practices

Backup and Recovery

Backing up and restoring data are key for any production and most nonproduction workloads. The relevant scenarios are:

- Backup and restore a virtual machine
- Backup and restore files and folders
- Backup and restore application data

We recommend taking advantage of Azure Backup.

Azure Backup

Azure Backup is the Azure-based service you can use to back up, protect and restore your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or off-site backup solution with a cloud-based solution that is reliable, secure, and cost-competitive. Azure Backup offers multiple components that you download and deploy on the appropriate computer, server, or in the cloud. The component or agent that you deploy depends on what you want to protect. All Azure Backup components (no matter whether you're protecting data on-premises or in the cloud) can be used to back up data to a Recovery Services vault in Azure. See the Azure Backup components table (later in this article) for information about which component to use to protect specific data, applications, or workloads.

Traditional backup solutions have evolved to treat the cloud as an endpoint or static storage destination, similar to disks or tape. While this approach is simple, it is limited and doesn't take full advantage of an underlying cloud platform, which translates to an expensive, inefficient solution. Other solutions are expensive because you end up paying for the wrong type of storage, or storage that you don't need. Other solutions are often inefficient because they don't offer you the type or amount of storage you need, or administrative tasks require too much time. In contrast, Azure Backup delivers these key benefits:

- **Automatic storage management** Hybrid environments often require heterogeneous storage - some on-premises and some in the cloud. With Azure Backup, there is no cost for using on-premises storage devices. Azure Backup automatically allocates and manages backup storage, and it uses a pay-as-you-use model. Pay-as-you-use means that you only pay for the storage that you consume. For more information, see the Azure pricing article (https://azure.microsoft.com/pricing/details/backup).
- **Unlimited scaling** Azure Backup uses the underlying power and unlimited scale of the Azure cloud to deliver high-availability - with no maintenance or monitoring overhead. You can set up alerts to provide information about events, but you don't need to worry about high-availability for your data in the cloud.
- **Multiple storage options** An aspect of high-availability is storage replication. Azure Backup offers two types of replication: locally redundant storage and geo-redundant storage. Choose the backup storage option based on need:
  - Locally redundant storage (LRS) replicates your data three times (it creates three copies of your data) in a storage scale unit in a datacenter. All copies of the data exist within the same region. LRS is a low-cost option for protecting your data from local hardware failures.
  - Geo-redundant storage (GRS) is the default and recommended replication option. GRS replicates your data to a secondary region (hundreds of miles away from the primary location of the source data). GRS costs more than LRS, but GRS provides a higher level of durability for your data, even if there is a regional outage.
- **Unlimited data transfer** Azure Backup does not limit the amount of inbound or outbound data you transfer. Azure Backup also does not charge for the data that is transferred. However, if you use the Azure Import/Export service to import large amounts of data, there is a cost associated with inbound

data. For more information about this cost, see Offline-backup workflow in Azure Backup. Outbound data refers to data transferred from a Recovery Services vault during a restore operation.
- **Data encryption** Data encryption allows for secure transmission and storage of your data in the public cloud. You store the encryption passphrase locally, and it is never transmitted or stored in Azure. If it is necessary to restore any of the data, only you have encryption passphrase, or key.
- **Application-consistent backup** An application-consistent backup means a recovery point has all required data to restore the backup copy. Azure Backup provides application-consistent backups, which ensure additional fixes are not required to restore the data. Restoring application-consistent data reduces the restoration time, allowing you to quickly return to a running state.
- **Long-term retention** You can use Recovery Services vaults for short-term and long-term data retention. Azure doesn't limit the length of time data can remain in a Recovery Services vault. You can keep data in a vault for as long as you like. Azure Backup has a limit of 9999 recovery points per protected instance.

Azure Virtual Machines

When the Azure Backup service initiates a backup job at the scheduled time, it triggers the backup extension to take a point-in-time snapshot. The Azure Backup service uses the VMSnapshot extension in Windows, and the VMSnapshotLinux extension in Linux. The extension is installed during the first VM backup. To install the extension, the VM must be running. If the VM is not running, the Backup service takes a snapshot of the underlying storage (since no application writes occur while the VM is stopped).

- During the backup process, Azure Backup doesn't include the temporary disk attached to the virtual machine.
- Since Azure Backup takes a storage-level snapshot and transfers that snapshot to vault, do not change the storage account keys until the backup job finishes.
- For premium VMs, we copy the snapshot to storage account. This is to make sure that Azure Backup service gets sufficient IOPS for transferring data to vault. This additional copy of storage is charged as per the VM allocated size.

Microsoft's best practices while configuring backups for virtual machines:

- Do not schedule more than 40 VMs to back up at the same time.
- Schedule VM backups during non-peak hours. This way the Backup service uses IOPS for transferring data from the customer storage account to the vault.
- Make sure that a policy is applied on VMs spread across different storage accounts. We suggest no more than 20 total disks from a single storage account be protected by the same backup schedule. If you have greater than 20 disks in a storage account, spread those VMs across multiple policies to get the required IOPS during the transfer phase of the backup process.
- Do not restore a VM running on Premium storage to same storage account. If the restore operation process coincides with the backup operation, it reduces the available IOPS for backup.
- For Premium VM backup, ensure that storage account that hosts premium disks has at least 50% free space for staging snapshot for a successful backup.
- Make sure that python version on Linux VMs enabled for backup is 2.7

Files and Folders

This backup option is designed to back up files and folders from any Windows machine. The machine can run in Azure, on-premises, or in any other cloud; it can be physical or virtual. You cannot use this option to back up the system state, or to create a Bare-Metal-Restore (BMR) backup. The Recovery Services Vault could be the one that is mentioned in the previous section, or it could be any other Recovery Services Vault.

Azure Backup for files and folders requires the installation of an Azure Backup Agent on the server, which can be downloaded from the Azure Recovery Services Vault. After installing the agent, it is necessary to connect the server to the Recovery Services Vault by downloading the vault credential files from the Recovery Services

Vault. The vault credentials file is used only during the registration workflow and expires after 48 hours. Ensure that the vault credential file is available in a location that can be accessed by the setup application.

Azure Site Recovery

What is Azure Site Recovery

Azure Site Recovery (ASR) is a service, that orchestrates and automates replication and failover of your Azure VMs between different Regions, of your on-premises VMs and physical servers to Azure and of your on-premises machines to a secondary datacenter. It's processes assist you in providing meaningful contributions to your business-continuity and disaster recovery strategy. Additionally, Azure Site Recovery allows you to move your IaaS-Solutions to other Regions or into Availability Zones.

How does it work?

To explain how ASR works, We'll be employing a Demo Environment, that looks like the following picture:

We have our Source Environment, containing our Resources, in this case the two VMs, that we would like to secure with Azure Site Recovery. On the other side, we have our Target Environment, where the copy of our Source Environment will be deployed into. To ensure that everything will run smoothly during a Failover, we need to make sure that the Target Environment mirrors our Source Environment.

Now, when we enable replication for our Azure VMs, the following will happen:

1. The Site Recovery Mobility service extension is automatically installed on the VMs.
2. The extension registers the VMs with Site Recovery.
3. Continuous replication begins for the VMs. Disk writes are immediately transferred to the cache storage account in the source location.
4. Site Recovery processes the data in the cache, and sends it to the target storage account, or to the replica managed disks.
5. After the data is processed, crash-consistent recovery points are generated every five minutes. App-consistent recovery points are generated according to the setting specified in the replication policy.

After these steps have been completed, our demo environment should look like this. As you can see, so far only our Data is being transferred over to the Target Environment. Only when the Failover process is initiated, the VMs will be created. Speaking of...

Failover Process

After you've first created the Replication Policy, it is highly recommended that you test the Failover Process, to ensure that everything is running smoothly. During the Failover, our Environment will look like this:

Our Source Environment is not available, be it for either an planned or an unplanned interruption in service. For this reason, a Failover has been initiated. Azure has created two identical VMs, mirroring those usually available in our Source Environment. If everything has been set up correctly, our customers will have experienced only a minor downtime during the moments that the failover was initiated, and now they're continuing their activities on the Target Environment.

Once our Source Environment has been re-established, we can failback to it and then clean up the Target Environment.

Infrastructure protection

VM Security

VM Authentication and access control

The first step in protecting your VM is to ensure that only authorized users are able to set up new VMs. You can use Azure policies to establish conventions for resources in your organization, create customized policies, and apply these policies to resources, such as resource groups.

VMs that belong to a resource group naturally inherit its policies. Although we recommend this approach to managing VMs, you can also control access to individual VM policies by using role-based access control (RBAC).

When you enable Resource Manager policies and RBAC to control VM access, you help improve overall VM security. We recommend that you consolidate VMs with the same life cycle into the same resource group. By using resource groups, you can deploy, monitor, and roll up billing costs for your resources. To enable users to access and set up VMs, use a least privilege approach. And when you assign privileges to users, plan to use the following built-in Azure roles:

- **Virtual Machine Contributor:** Can manage VMs, but not the virtual network or storage account to which they are connected.
- **Security Manager:** Can manage security components, security policies, and VMs.
- **DevTest Labs User:** Can view everything and connect, start, restart, and shut down VMs.

Don't share accounts and passwords between administrators, and don't reuse passwords across multiple user accounts or services, particularly passwords for social media or other non-administrative activities. Ideally, you should use Azure Resource Manager templates to set up your VMs securely. By using this approach, you can strengthen your deployment choices and enforce security settings throughout the deployment.

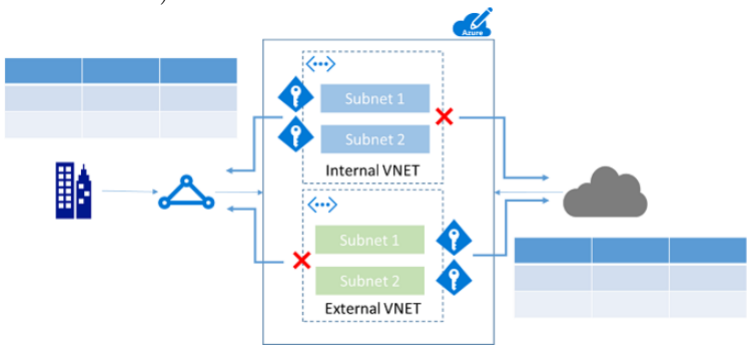
Organizations that do not enforce data-access control by taking advantage of capabilities such as RBAC might be granting their users more privileges than necessary. Inappropriate user access to certain data can directly compromise that data.

Network security

Core networking resources

Access to resources can be either internal (within the corporation's network) or external (through the internet). It is easy for users in your organization to inadvertently put resources in the wrong spot, and potentially open them to malicious access. As with on-premises devices, enterprises must add appropriate controls to ensure that Azure users make the right decisions. For subscription governance, we identify core resources that provide basic control of access. The core resources consist of:

- **Virtual networks** are container objects for subnets. Though not strictly necessary, it is often used when connecting applications to internal corporate resources.
- **Network security groups** are similar to a firewall and provide rules for how a resource can "talk" over the network. They provide granular control over how/if a subnet (or virtual machine) can connect to the Internet or other subnets in the same virtual network.



- Create virtual networks dedicated to external-facing workloads and internal-facing workloads. This approach reduces the chance of inadvertently placing virtual machines that are intended for internal workloads in an external facing space.
- Configure network security groups to limit access. At a minimum, block access to the internet from internal virtual networks, and block access to the corporate network from external virtual networks.

Microsoft Azure enables you to connect virtual machines and appliances to other networked devices by placing them on Azure Virtual Networks. An Azure Virtual Network is a construct that allows you to connect virtual network interface cards to a virtual network to allow TCP/IP-based communications between network enabled devices. Azure Virtual Machines connected to an Azure Virtual Network are able to connect to devices on the same Azure Virtual Network, different Azure Virtual Networks, on the Internet or even on your own on-premises networks.

Azure Virtual Networks are similar to a LAN on your on-premises network. The idea behind an Azure Virtual Network is that you create a single private IP address space-based network on which you can place all your Azure Virtual Machines. The private IP address spaces available are in the Class A (10.0.0.0/8), Class B (172.16.0.0/12), and Class C (192.168.0.0/16) ranges.

Source: https://docs.microsoft.com/en-us/azure/security/security-best-practices-and-patterns

Resource Locks

https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-subscription-governance#azure-resource-locks

As organizations add core services to the subscription, it becomes increasingly important to ensure that those services are available to avoid business disruption. Resource locks enable to restrict operations on high-value resources where modifying or deleting them would have a significant impact on your applications or cloud infrastructure. You can apply locks to a subscription, resource group, or resource. Typically, you apply locks to foundational resources such as **virtual networks, gateways, and storage accounts.**

Resource locks currently support two values: **CanNotDelete** and **ReadOnly**. **CanNotDelete** means that users (with the appropriate rights) can still read or modify a resource but cannot delete it. **ReadOnly** means that authorized users can't delete or modify a resource.

To create or delete management locks, you must have access to Microsoft.Authorization/* or Microsoft.Authorization/locks/* actions. Of the built-in roles, only Owner and User Access Administrator are granted those actions.

We recommend to protect core network options with locks. Accidental deletion of a gateway, site-to-site VPN would be disastrous to an Azure subscription. Azure doesn't allow you to delete a virtual network that is in use, but applying more restrictions is a helpful precaution.

- Virtual Network: CanNotDelete
- Network Security Group: CanNotDelete
- Policies: CanNotDelete

Policies are also crucial to the maintenance of appropriate controls. We recommend that you apply a CanNotDelete lock to polices that are in use. https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-lock-resources

Azure Policy

IT governance creates clarity between business goals and IT projects. Good IT governance involves planning your initiatives and setting priorities on a strategic level. Does your company experience a significant number of IT issues that never seem to get resolved? Implementing policies helps you better manage and prevent them. Implementing policies is where Azure Policy comes in.

Azure Policy is a service in Azure that you use to create, assign and, manage policy definitions. Policy definitions enforce different rules and actions over your resources, so those resources stay compliant with your corporate standards and service level agreements. Azure Policy runs an evaluation of your resources, scanning for those not compliant with the policy definitions you have. For example, you can have a policy to allow only certain type of virtual machines. **Another requires that all resources have a particular tag**. These policies are then evaluated when creating and updating resources.

How is it different from RBAC?

There are a few key differences between policy and role-based access control (RBAC). RBAC focuses on user actions at different scopes. For example, you might be added to the contributor role for a resource group at the desired scope. The role allows you to make changes to that resource group. Policy focuses on resource properties during deployment and for already existing resources. For example, through policies, you can control the types of resources that can be provisioned. Or, you can restrict the locations in which the resources can be provisioned. Unlike RBAC, policy is a default allow and explicit deny system.

To use policies, you must be authenticated through RBAC. Specifically, your account needs the:

- Microsoft.Authorization/policydefinitions/write permission to define a policy.
- Microsoft.Authorization/policyassignments/write permission to assign a policy.
- Microsoft.Authorization/policySetDefinitions/write permission to define an initiative.
- Microsoft.Authorization/policyassignments/write permission to assign an initiative.

These permissions are not included in the Contributor role.

Policy definition

Every policy definition has conditions under which it is enforced. And, it has an accompanying action that takes place if the conditions are met.

In Azure Policy, we offer some built-in policies that are available to you by default. For example:

- Require SQL Server 12.0: This policy definition has conditions/rules to ensure that all SQL servers use version 12.0. Its action is to deny all servers that do not meet these criteria.
- Allowed Storage Account SKUs: This policy definition has a set of conditions/rules that determine if a storage account that is being deployed is within a set of SKU sizes. Its action is to deny all servers that do not adhere to the set of defined SKU sizes.
- Allowed Resource Type: This policy definition has a set of conditions/rules to specify the resource types that your organization can deploy. Its action is to deny all resources that are not part of this defined list.
- Allowed Locations: This policy enables you to restrict the locations that your organization can specify when deploying resources. Its action is used to enforce your geo-compliance requirements.
- Allowed Virtual Machine SKUs: This policy enables you to specify a set of virtual machine SKUs that your organization can deploy.
- Apply tag and its default value: This policy applies a required tag and its default value, if it is not specified by the user.
- Enforce tag and its value: This policy enforces a required tag and its value to a resource.
- Not allowed resource types: This policy enables you to specify the resource types that your organization cannot deploy.
- You can assign any of these policies through the Azure portal, PowerShell, or Azure CLI.

https://docs.microsoft.com/en-us/azure/azure-policy/azure-policy-introduction

Working to keep customer data safe

Security design and operations

Infrastructure protection

Network protection

Data protection (Backup)

Identity and access

https://docs.microsoft.com/en-us/azure/security/azure-security-identity-management-best-practices

Owning and controlling data

Managing compliance and data privacy regulations

Azure Security Center

Cloud security recommendations for enterprise architects

Azure enterprise administration

License

```
 Considerations for the public: By using one of our public
 licenses, a licensor grants the public permission to use the
 licensed material under specified terms and conditions. If
 the licensor's permission is not necessary for any reason--for
 example, because of any applicable exception or limitation to
 copyright--then that use is not regulated by the license. Our
 licenses grant only permissions under copyright and certain
 other rights that a licensor has authority to grant. Use of
 the licensed material may still be restricted for other
 reasons, including because others have copyright or other
 rights in the material. A licensor may make special requests,
 such as asking that all changes be marked or described.
 Although not required by our licenses, you are encouraged to
 respect those requests where reasonable. More_considerations
 for the public:
wiki.creativecommons.org/Considerations_for_licensees
```

==========

Creative Commons Attribution 4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

Section 1 – Definitions.

a. Adapted Material means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synched in timed relation with a moving image.

b. Adapter's License means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.

c. Copyright and Similar Rights means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.

d. Effective Technological Measures means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.

e. Exceptions and Limitations means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.

f. Licensed Material means the artistic or literary work, database, or other material to which the Licensor applied this Public License.

g. Licensed Rights means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licensor has authority to license.

h. Licensor means the individual(s) or entity(ies) granting rights under this Public License.

i. Share means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.

j. Sui Generis Database Rights means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.

k. You means the individual or entity exercising the Licensed Rights under this Public License. Your has a corresponding meaning.

Section 2 – Scope.

a. License grant.

1. Subject to the terms and conditions of this Public License, the Licensor hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:

a. reproduce and Share the Licensed Material, in whole or in part; and

b. produce, reproduce, and Share Adapted Material.

2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.

3. Term. The term of this Public License is specified in Section 6(a).

4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, and to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License, simply making modifications authorized by this Section 2(a)

(4) never produces Adapted Material.

5. Downstream recipients.

a. Offer from the Licensor – Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

b. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, the Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

1. Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.
3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties.

Section 3 – License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.
    1. If You Share the Licensed Material (including in modified form), You must:
        a. retain the following if it is supplied by the Licensor with the Licensed Material:
            i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);
            ii. a copyright notice;
            iii. a notice that refers to this Public License;
            iv. a notice that refers to the disclaimer of warranties;
            v. a URI or hyperlink to the Licensed Material to the extent reasonably practicable;
        b. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and
        c. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or hyperlink to, this Public License.
    2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or hyperlink to a resource that includes the required information.
    3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.
    4. If You Share Adapted Material You produce, the Adapter's License You apply must not prevent recipients of the Adapted Material from complying with this Public License.

Section 4 – Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:

a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database;
b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material; and
c. You must comply with the conditions in Section 3(a) if You Share all or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.

Section 5 – Disclaimer of Warranties and Limitation of Liability.

a. UNLESS OTHERWISE SEPARATELY UNDERTAKEN BY THE LICENSOR, TO THE EXTENT POSSIBLE, THE LICENSOR OFFERS THE LICENSED MATERIAL AS-IS AND AS-AVAILABLE, AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED MATERIAL, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHER. THIS INCLUDES, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OR ABSENCE OF ERRORS, WHETHER OR NOT KNOWN OR DISCOVERABLE. WHERE DISCLAIMERS OF WARRANTIES ARE NOT ALLOWED IN FULL OR IN PART, THIS DISCLAIMER MAY NOT APPLY TO YOU.

b. TO THE EXTENT POSSIBLE, IN NO EVENT WILL THE LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE) OR OTHERWISE FOR ANY DIRECT, SPECIAL, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, EXEMPLARY, OR OTHER LOSSES, COSTS, EXPENSES, OR DAMAGES ARISING OUT OF THIS PUBLIC LICENSE OR USE OF THE LICENSED MATERIAL, EVEN IF THE LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES, COSTS, EXPENSES, OR DAMAGES. WHERE A LIMITATION OF LIABILITY IS NOT ALLOWED IN FULL OR IN PART, THIS LIMITATION MAY NOT APPLY TO YOU.

c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

Section 6 – Term and Termination.

a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.

b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:

   1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or

   2. upon express reinstatement by the Licensor.

   For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.

c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.

d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

Section 7 – Other Terms and Conditions.

a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.

b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

Section 8 – Interpretation.

a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.

b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.

c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.

d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

==========

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the "Licensor." The text of the Creative Commons public licenses is dedicated to the public domain under the CC0 Public Domain Dedication. Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at creativecommons.org/policies, Creative Commons does not authorize the use of the trademark "Creative Commons" or any other trademark or logo of Creative Commons without its prior written consent including, without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.

Creative Commons may be contacted at creativecommons.org.