Azure Fundamental for Ethical Hackers and Special Ops Team

"Everything and anything
is hackable and vulnerable
in some way."

Table of Contents

Introduction

I started working with Microsoft Azure many years ago, in the beginning, on the development side using specific Azure services like Service Bus, Storage Accounts, Databases, and later I extended my knowledge in all the other areas.

Two years ago I joined a global manufacturing company with the role of Global Azure Lead, quite scary name right?, it scared me as well, my daily job is to support the entire company on Microsoft Azure for any aspect like Azure Governance, strategies, security, technical implementations, best practices, almost everything at 360 level.

In these two years, I learned more than I couldn't imagine, and most importantly, I now have a good comprehension of what Microsoft Azure is, all of its weaknesses and strengths.

It is important for any security professional and company to understand Microsoft Azure in order to provide the best services and results, there are important concepts that must be very clear to any ethical hacker to be able to perform effective attacks and provide countermeasures.

How can we attack or protect something we don't really know, what is Microsoft Azure and how it works?

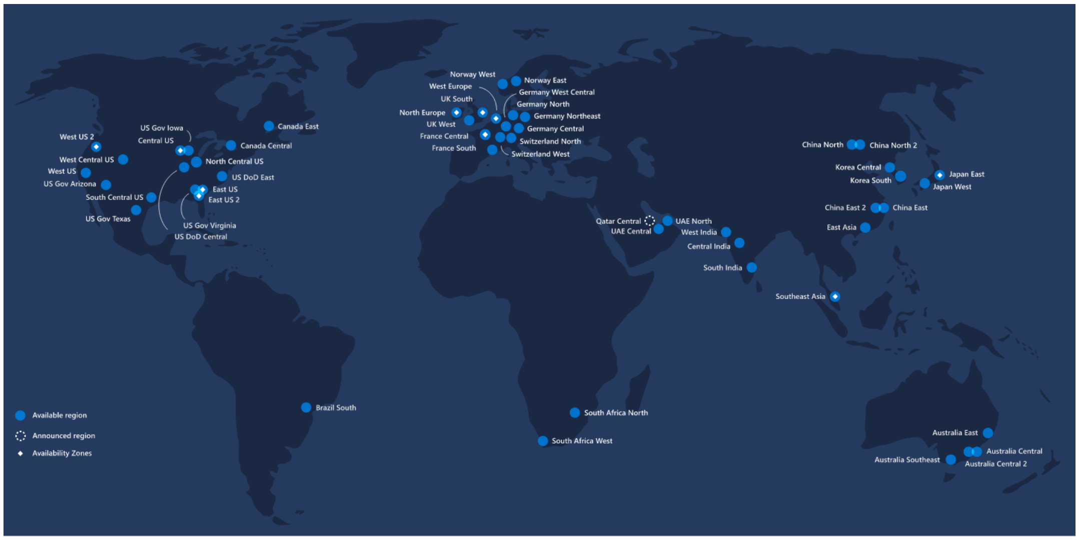
I made this question many times to many people, Azure is huge, and it is almost impossible to be an expert on everything, but we can get a very good understanding of it. In this whitepaper, I will give you the information you need to get a wide comprehension of this topic.

Microsoft Azure Fundamentals

Microsoft Azure, codename Project Red Dog in 2008 and finally Windows Azure in 2010, is a set of technologies and services able to provide a lot of benefits and features out of the box for cloud computing.

The Services

Microsoft Azure provides an enormous number of services like database, virtual machines, high computing, web hosting, services for Artificial Intelligence, I mean almost anything, and these services are hosted in different datacentres around the globe.

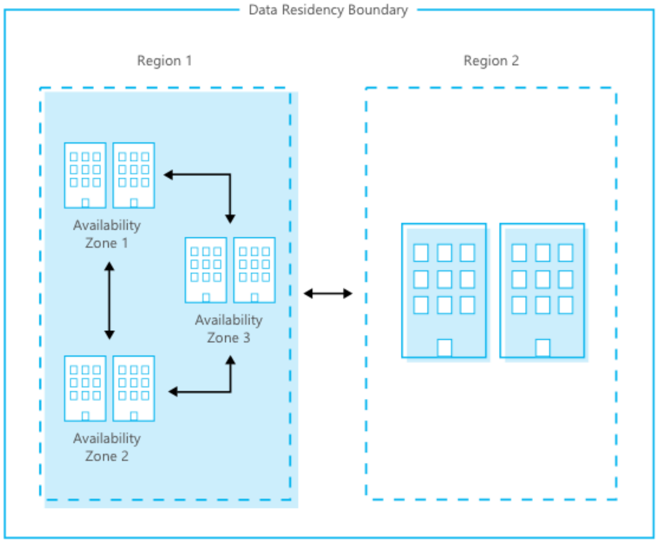


At the moment of this whitepaper, Azure had 55 regions worldwide (two additional Azure Government Secret region locations undisclosed), available in 140 countries that are more global regions than any other cloud provider.

Each **datacentre** is a physical building containing a huge amount of racks and servers. Each datacentre is interconnected with each other through the Microsoft Network, also known as the **Azure Backbone**. This is one of the largest backbone networks in the world spread across hundreds of regions.

● Datacenter
● Edge
Network

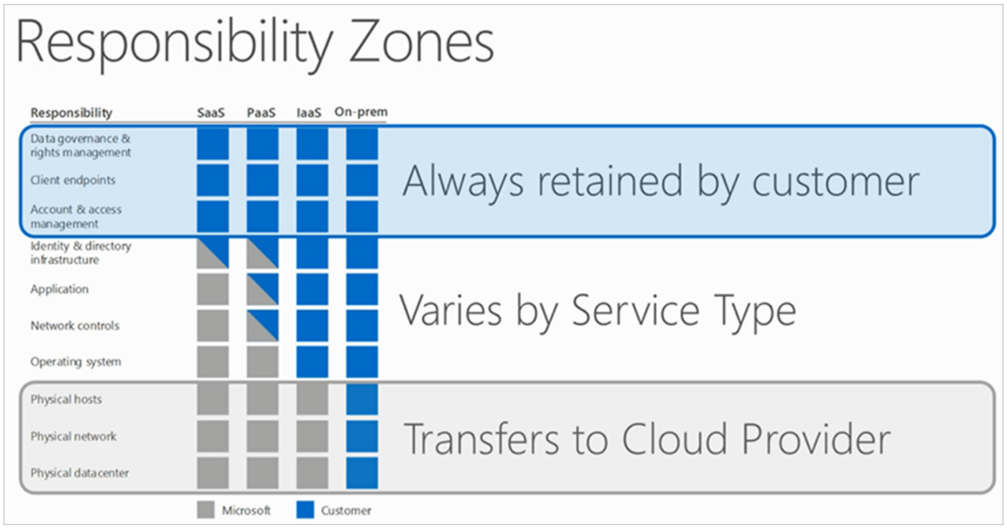
A region is a set of datacenters deployed within a latency-defined perimeter and connected through a dedicated regional low-latency network.



The region is the most important binding in Azure, and because the service is physically hosted in a specific region, it inherits, and it depends on many factors like connectivity with other services in other regions, local law rules and legislation, and more. It is extremely important, for example, a specific penetration test procedure may require a different approach in a country location from another one.

In term of security, there is also another crucial point to consider, and this is the service type, let see more in detail.

Microsoft Azure provides three main categories of software services, **Software as a Service (SaaS)**, **Platform as a Service (PaaS)** and **Infrastructure as a Service (IaaS)**, and the responsibility of managing the security and services depends by which categories we choose, this is called **Responsibility Zones**.



Responsibility Zones

Responsibility	SaaS	PaaS	IaaS	On-prem	
Data governance & rights management	🟦	🟦	🟦	🟦	Always retained by customer
Client endpoints	🟦	🟦	🟦	🟦	
Account & access management	🟦	🟦	🟦	🟦	
Identity & directory infrastructure	◩	◩	🟦	🟦	Varies by Service Type
Application	⬜	◩	🟦	🟦	
Network controls	⬜	◩	🟦	🟦	
Operating system	⬜	⬜	🟦	🟦	
Physical hosts	⬜	⬜	⬜	🟦	Transfers to Cloud Provider
Physical network	⬜	⬜	⬜	🟦	
Physical datacenter	⬜	⬜	⬜	🟦	

⬜ Microsoft 🟦 Customer

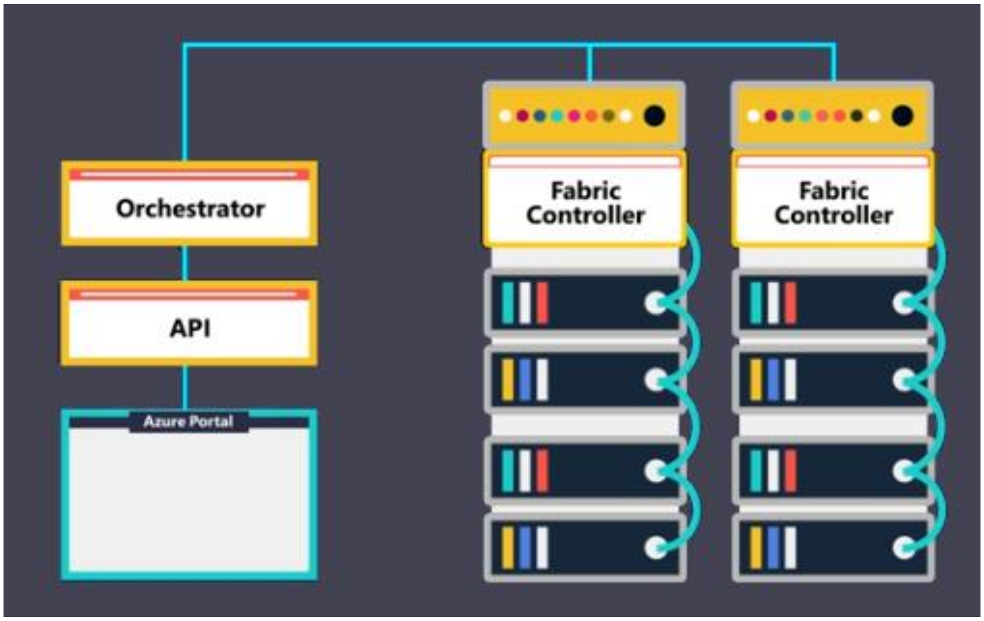
The blue color represents the customer and the grey color represents Microsoft. The On-prem means that all the resources are hosted and managed by the customer so more exposes and subjected to attacks.

Important rule, the attack surface of a cloud environment increases with customer responsibility. It is a good security practice in Azure to use SaaS features and being more Azure as friendly as possible

The Main Components

Let now go deeper into our discussion, how exactly a Microsoft Azure datacentre work, and how it is able to provide such power and reliable service provisioning?

In the picture below, you can see a simple representation of an Azure datacentre, and the explanation may look simple but, believe me, it contains many interesting points of view that even people working in Azure for a long time are still missing.



The Azure datacentre contains a large number of racks hosting many Windows Azure Servers, and each server includes the hypervisor, a virtualization technology to run Virtual Machines, and another very important technology called **Fabric Controller**.

All racks and servers are interconnected with each other through network switches, and all Fabric Controllers are connected to the most important element called the **Orchestrator**.

The Orchestration manages everything that happens in Azure. It organizes users' requests, orchestrates the internal instance in the Fabric Controller, and a lot more. It is the Azure brain.

It will also expose an Azure interface, the **Orchestrator API**. This technology stack containing hundreds of thousands of Web APIs interfaces, and each of these interfaces provide any kind of operation like create a specific service, list one or more services, delete services, create an account, delete them, in other words, any possible Azure operation can be executed from APIs.And to finalize, Microsoft created a fantastic web interface to manage any Azure Service available, the **Microsoft Azure Portal**. The portal uses the Web API to trigger what the user requests and send the command to the Orchestrator.

Let think now from a hacker point of view, if the portal uses the Web API for any operation, it means that we can do anything we want calling these Web API, yes that's correct, and this is why I consider the Azure APIs the most powerful Azure stack, we will see how to use them to perform very advanced scanning and reconnaissance techniques.

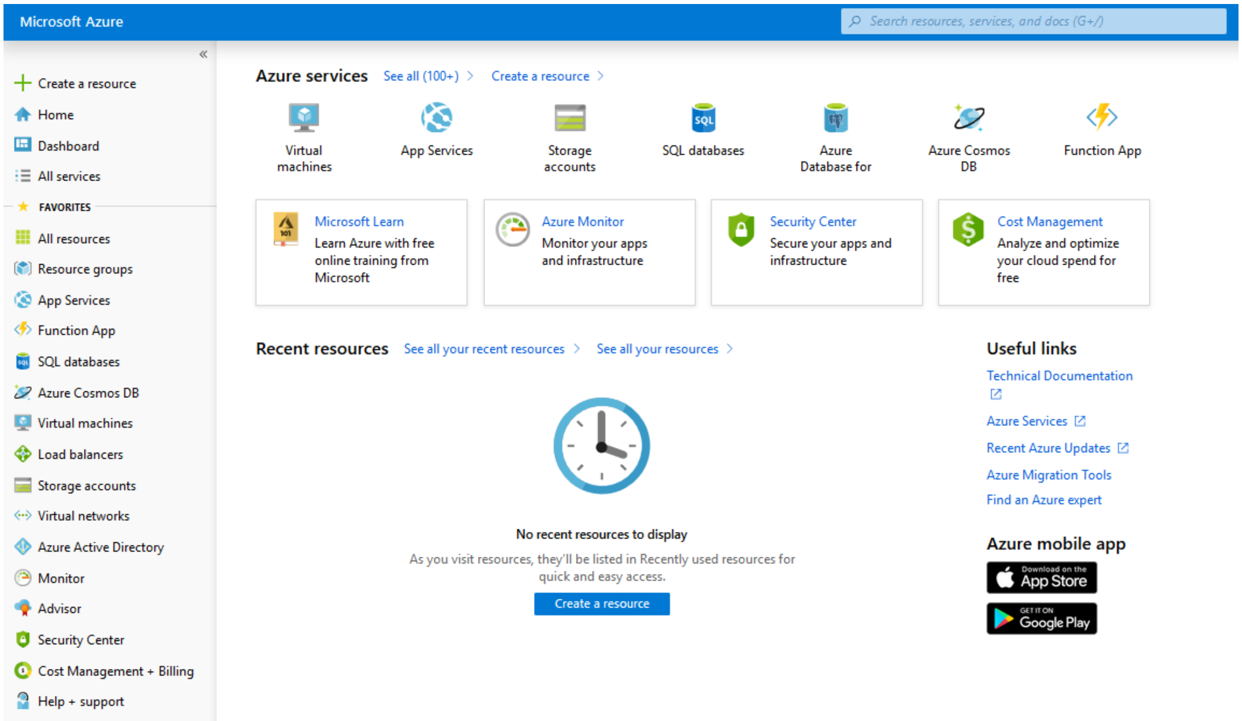
Everything uses Azure APIs, Azure Portal, PowerShell libraries, the Azure .Net SDK, Azure Resource Explorer and more, the point I want to reach here is, if you want to be sure to work at the lowest level with Azure resources, then use the Azure APIs.

You can find all the references and documentation of the Azure APIs in the following link: https://docs.microsoft.com/en-us/rest/api

Microsoft Azure has three main portals, and each of them is used for specific scopes, billing, account, and resources. Let see each one of them.

The Azure Portal

The **Azure Portal** (https://portal.azure.com) is used to manage and operate in the Azure resources, simply speaking, it is a very powerful UI to use the Azure APIs.



Learn Azure with free online training from Microsoft

Monitor your apps and infrastructure

Secure your apps and infrastructure

Analyze and optimize your cloud spend for free

No recent resources to display

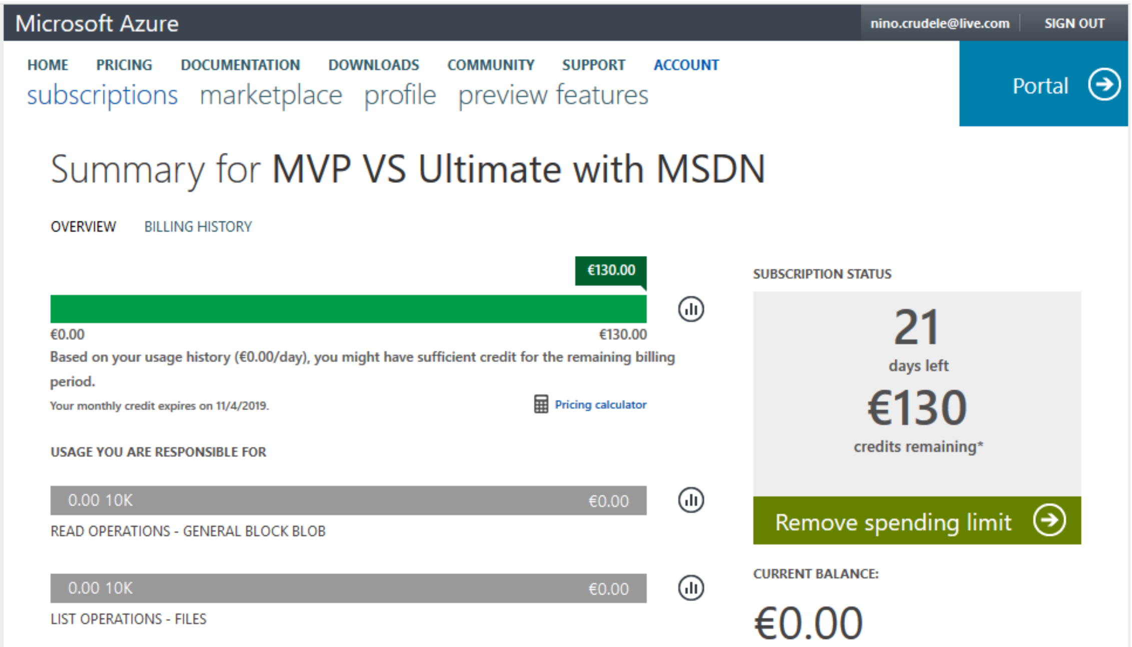
As you visit resources, they'll be listed in Recently used resources for quick and easy access.

Everything we do with the Azure Portal is replicable using the Azure APIs, list resources, managing the resources, even the most complex operations.

We can organize the portal in many ways, and this is almost the most important UI in Microsoft Azure.

The Account Portal

The **Account Portal** is used to manage the billing and some administrative aspects of subscriptions, using this portal we can set the spending limits, assign the service account, change subscription name, we can also do some basic spending reviews, we can transfer a subscription and more.



Summary for **MVP VS Ultimate with MSDN**

Based on your usage history (€0.00/day), you might have sufficient credit for the remaining billing period.

Your monthly credit expires on 11/4/2019.

USAGE YOU ARE RESPONSIBLE FOR

0.00 10K	€0.00
READ OPERATIONS - GENERAL BLOCK BLOB	
0.00 10K	€0.00
LIST OPERATIONS - FILES	

SUBSCRIPTION STATUS

21 days left

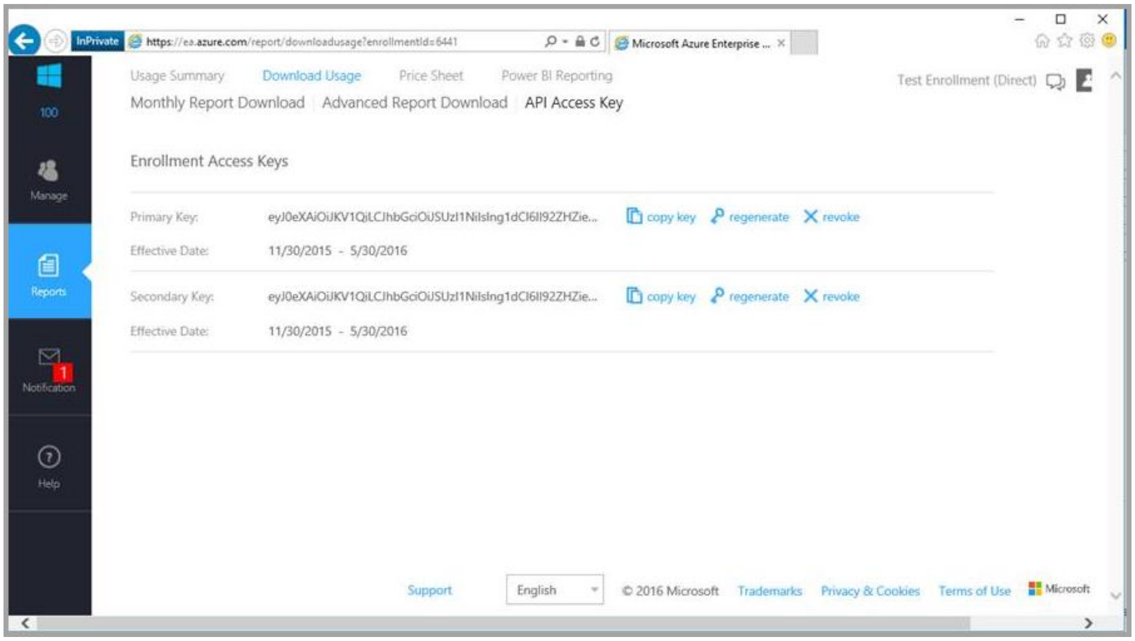
€130 credits remaining*

CURRENT BALANCE:

€0.00

The EA Portal

The **EA Portal** is used by the companies with an Azure Enterprise Agreement, and it is used to manage departments, billing security aspects, it was the first and only portal to use to check and manage the costs, now we can also do that from the Azure Portal. It is still a very used portal because it contains important information regarding the Enterprise Enrolment, for managing department and subscriptions for billing, to download usage and to get the access key for the Azure consumptions APIs.



Enrollment Access Keys

Primary Key:	eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Il92ZHZie...	copy key regenerate revoke
Effective Date:	11/30/2015 - 5/30/2016	
Secondary Key:	eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsIng1dCI6Il92ZHZie...	copy key regenerate revoke
Effective Date:	11/30/2015 - 5/30/2016	

The Enrolment API key is very important, and it must be protected and suddenly regenerated, I wrote an interesting article in my blog related to privilege escalation, I recommend the reading, below the link:

- https://ninocrudele.com/the-three-most-effective-and-dangerous-cyberattacks-to-azure-and-countermeasures-part-3-the-privilege-escalation

The Licensing Model

There are different ways to join Microsoft Azure, but the way Microsoft organizes contract and relationship is always the same, **Tenant**, **Subscriptions**, **Licence type,** and **account**. Microsoft licensing is a jungle. Let us see the most relevant aspects.

The most used Azure licenses are **Azure CSP (Cloud solution Provider Program)**, **Azure EA (Azure Enterprise Agreement)**, **Free Trial** and **MSDN Visual Studio**.

Azure CSP is a contract for resellers, and for companies that want to provide cloud services and hosting services in Microsoft Azure, the Azure EA is a contract for Enterprise companies. In both of these contracts, we may have multiple tenants and subscriptions.

The Free trial, Pay-As-You-Go, and Visual Studio are contract base on a single subscription and tied to a single tenant or the Microsoft tenant. Some of these offers can also be part of the Azure Enterprise Agreement, like the Pay-As-You-Go.

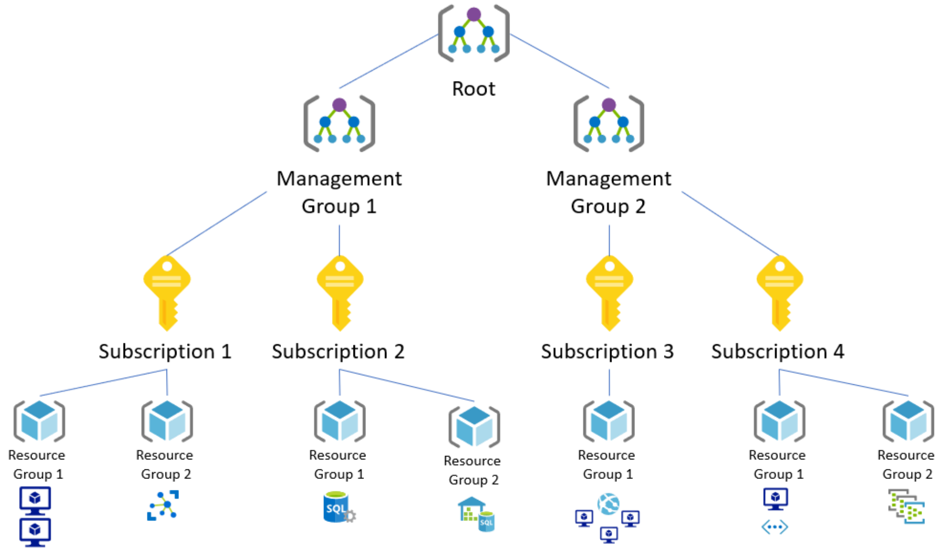
The Enterprise Agreement is the most complex scenario, and there are some key securities areas to look about, for example, the security in the billing areas is much more complex and sensitive than the other contracts, and a privilege escalation can be devastating, especially if implemented in the Global Admin account.

I wrote an interesting article in my blog related to privilege escalation, I recommend the reading, below the link:

- https://ninocrudele.com/the-three-most-effective-and-dangerous-cyberattacks-to-azure-and-countermeasures-part-3-the-privilege-escalation

The Azure Hierarchy and Dependencies

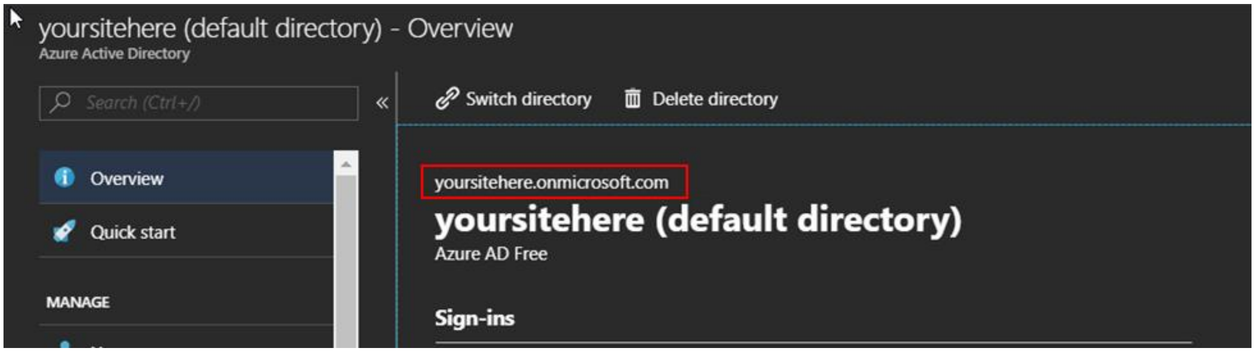
This is one of the most important aspects to know for any ethical hacker who wants to work in Azure, everything in Microsoft Azure is interconnected through networking and configuration, what I mean is, we can reach any Azure resource or by network or by interconnected dependency, and for that reason we need a clear understanding about the relation between the resources and especially about the Azure Hierarchy, below a representation of that.



The Tenant

The **Azure Tenant** is the most important aspect because everything in Azure depends on that. In order to use Microsoft Azure, we first need an account in **Azure Active Directory (Azure AD)**, the account has permission in the Azure AD but not into the Azure resources.

We need to grant access to the Azure resources, and this happens in different ways, when we create a subscription, when we grant access to the Azure AD account into the Azure resources or, in case of the Global Admin, when we enable the Access Management for Azure resources in the Azure AD. We may have a company tenant identified by a public domain name like **companyname.onmicrosoft.com**, or we may have a domain name like **accountname.onmicrosoft.com** for a single account. This is usually for a free trial and Pay-As-you-Go Subscription.



yoursitehere (default directory) - Overview
Azure Active Directory

yoursitehere.onmicrosoft.com
yoursitehere (default directory)
Azure AD Free

Azure AD is the service to manage the accounts and identities in the tenant. We can have one or multiple tenants. Each Azure subscription is tied to a tenant, and the relationship between the tenant and the subscription is very important because it defines the security landscape.

The Management Group

The **Management Groups** provide a logical way to better organize the subscriptions when Microsoft creates a new Azure contract, it creates at least one Tenant, one Root Management Group, and one Subscription.

The Root Management Group is tied to the Tenant, and all the Subscriptions in that group depend on that group. The Root group setting is extremely critical because any security setting applied to that group will affect any internal Subscription and Azure resource.

The Root Management Group can contain other Management Groups, and this is the best practice to manage multiple subscriptions and security. During our scans, we may find more than one.

The Subscription

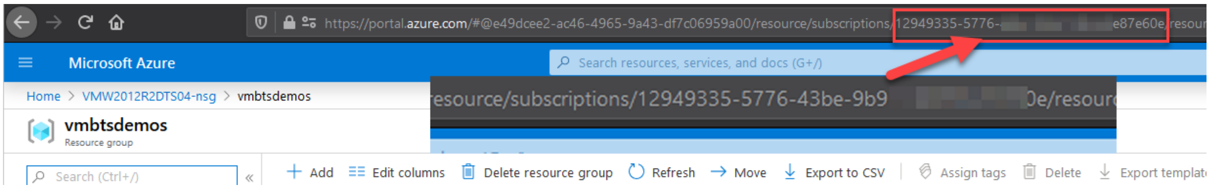
As I mentioned before, the Active Directory Account needs at least one **Azure Subscription** to be able to consume Azure resources, and the Subscription represents the consumption agreement with Microsoft.

We can have multiple Subscriptions under the Root Management Group, each Subscription may use different contract types, which is not really relevant to us, but it is important to know that each Subscription is tied to an Azure AD account named Service Administrator, we will speak later about the accounts.

The Subscription is a logical type of contract container with contains all the resources used in this contract type, resources can be moved between subscriptions, and we can connect networks and resources across subscriptions, keep in mind what I said, the region is only one real physical restriction in Microsoft Azure, anything else is just logical containerization.

Any subscription has a name and a subscription ID, the second one never changes, and it is the unique identification ID, get familiar with that because it is used everywhere, it is in any Azure Url and Azure API call.

In the picture below, you can see the subscription id:



https://portal.azure.com/#@e49dcee2-ac46-4965-9a43-df7c06959a00/resource/subscriptions/12949335-5776-[redacted]e87e60e/resour

esource/subscriptions/12949335-5776-43be-9b9[redacted]0e/resour

Subscription is extremely important, customers use it for many reasons, the subscription is the best way to isolate costs in enterprise scenarios, they use it also to better isolate the security landscape, and another reason is that they need to use a different tenant from the other subscriptions.

The relation between tenant and subscriptions is one to many, in the same Azure EA we can have multiple subscriptions tied to the same tenant, and we can have multiple tenants tied to different subscriptions, we cannot have one subscription tied to multiple tenants but we can grants permission to the same subscription from multiple tenants.

The subscription is still a very high level of organization and management, to achieve a much more granularity, Microsoft introduced the Resource groups.

The Resource Group

Azure contains an endless number of features and resources types, and some infrastructure and solutions can be extremely complex. The **Resource Group** provides a logical way to organize them.

People use Resource Groups to organize the different resources or services in the Subscription, this way, they can manage the security landscape for this specific group of resources. The Resource Group also provides agile deployments and using them, we can achieve much better monitoring, and we can speed up the lifecycle of the resource.

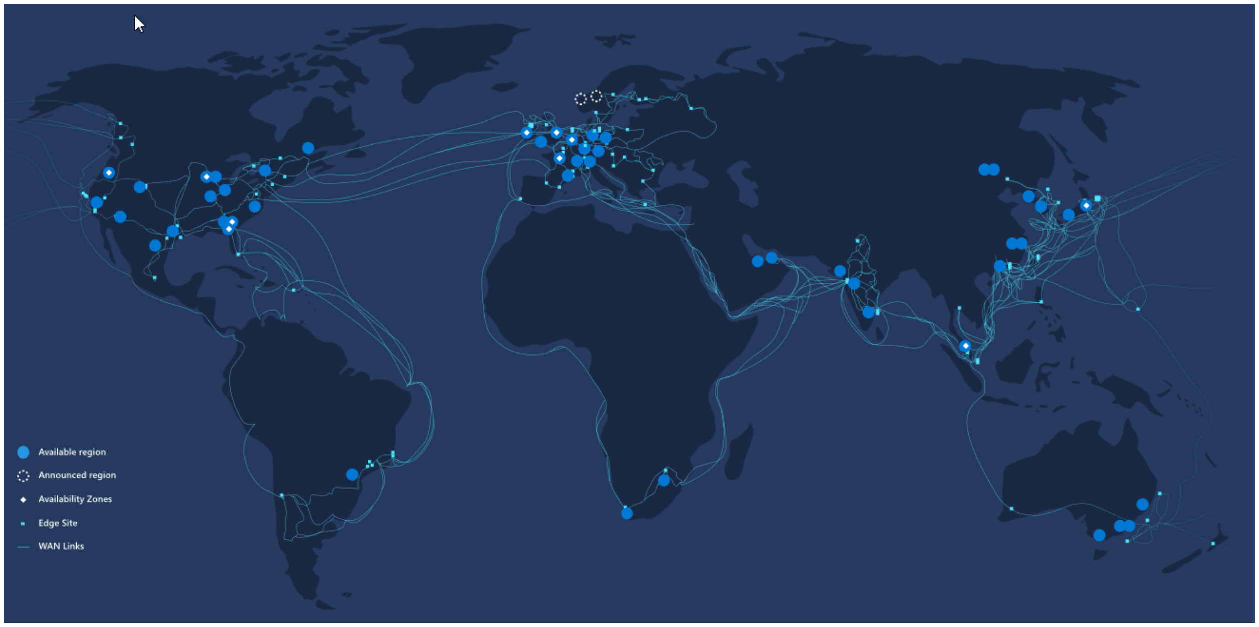
Azure Resource Groups is a very critical asset in Azure and there are many best practices explaining how to use them, especially regarding the naming standard to use, and for that reason, it is important to know that, most of the time, the Resource Group itself can provide us very useful information regarding the infrastructure, we will see more in the naming standards section.

Think about a **namespace where you organize your things**, the resource group is similar to a subscription, but it is not a contract. It is just a logical namespace.

The Network

Everything in Azure is interconnected, the **network** is the core in any Azure infrastructure, and it is required for an ethical hacker to have a very good understanding of that.

The **Azure Network** is global, and it connects up to 150 datacentres across 54 regions around the world. We call this internal network the Azure backbone.



Microsoft is committing to build the most powerful network infrastructure in the world, and it is impressive. The internal connectivity bandwidth is 100 Gbps, and we can maintain that connectivity across the regions using a technology called ExpressRoute, all IP traffic is internal unless we decide to expose it and there are more than 60 CDN (content delivery network) available.

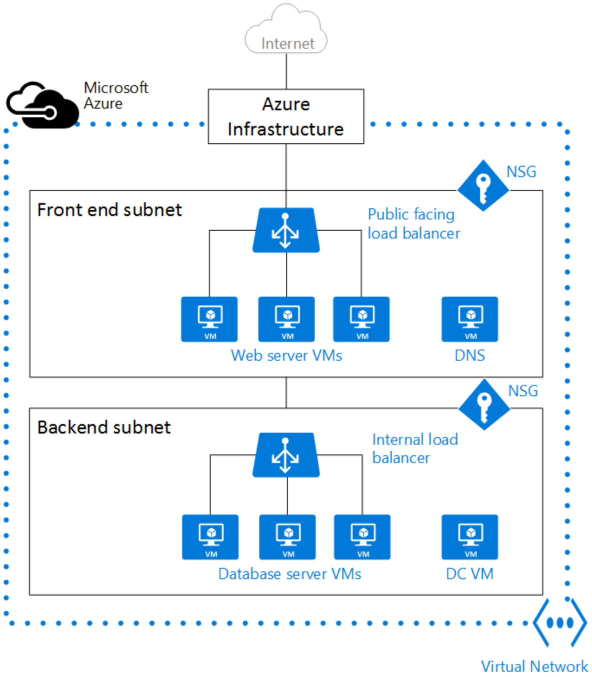
We don't pay for the traffic sent to Microsoft Azure. We pay outgoing traffic only.

The Azure resources are interconnected using **Virtual Networks (VNet)**, thinks about a VNet like the logical representation of your office network, the VNet is an isolated network, and it is required to connect the Azure resources and or if we want to connect to our on-premise resources outside Azure.

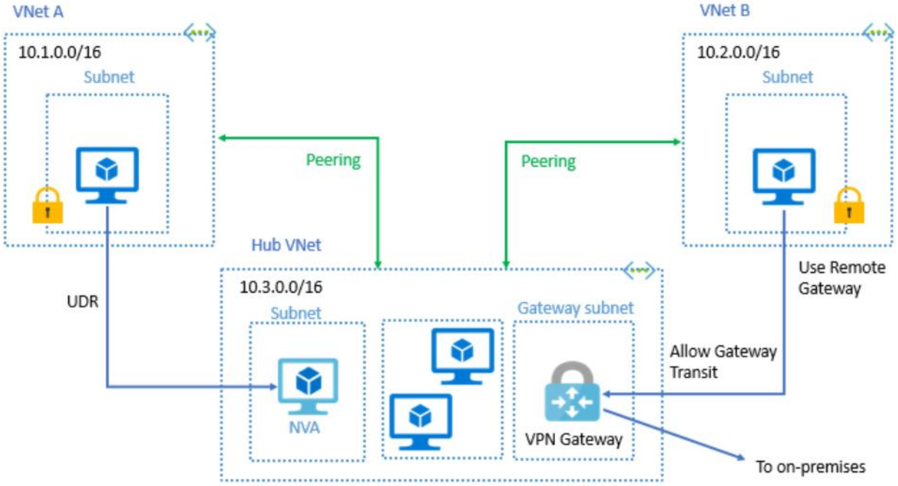
When we create a VNet, we must specify an address space like 10.0.0.0/16, and when we start deploying resources, like a VM, the VNet will assign a private IP like 10.0.0.5.

The VNet is organized in Subnets, and we need at least one subnet to deploy an Azure resource. We use subnets to segment the VNet in different subnetwork areas, for example, to isolate our frontend application layer to the backend.

Each Subnet can be protected by a **Network Security Group (NSG),** which is a basic logical firewall. The NSG can be assigned to a subnet or even to a specific network card of the VM, below an example of a simple two layers scenario.

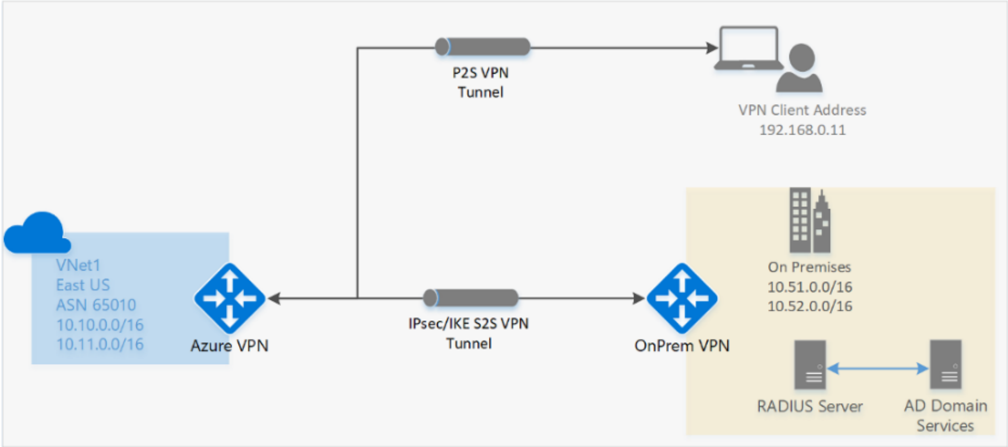


The main scope of a VNet is the region/location and the subscription, and we can connect different VNet together using peering rf a VPN Gateway, we also use VPN Gateway to connect on-premises resources to Azure, below an example.

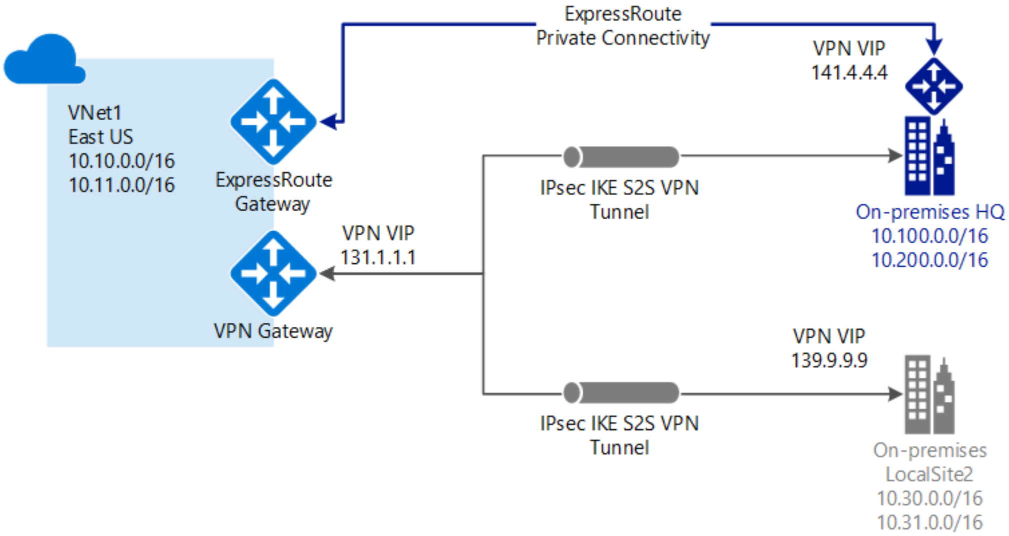


We have three options to connect our on-premise resource to Microsoft Azure:

- **Point-to-site virtual private network (P2S VPN)**: this is a dedicate VPN from a single computer, each computer needs to configure the VPN connectivity in order to connect to the Azure VPN Gateway and through an encrypted channel over the internet.
    - The P2S VPN is used to connect a specific device or a person to Azure without using a more complex connectivity scenario, for example, to connect a developer or a specific computer.
- **Site-to-site VPN (S2S VPN)**: in that case, our on-premise VPN is connected to the Azure VPN Gateway, and all computers in our on-premise VPN will be connected to Azure through an encrypted channel over the internet.



- **Azure ExpressRoute**: this is the first-class connectivity. Essentially, we create a private connection to the Azure backbone, and we become part of the Azure backbone.



We will speak a lot about networking, and we will dig much more in detail in the next publications. This is the most crucial area in Azure, and everything depends on the network design. A bad design means low security and more vulnerabilities.

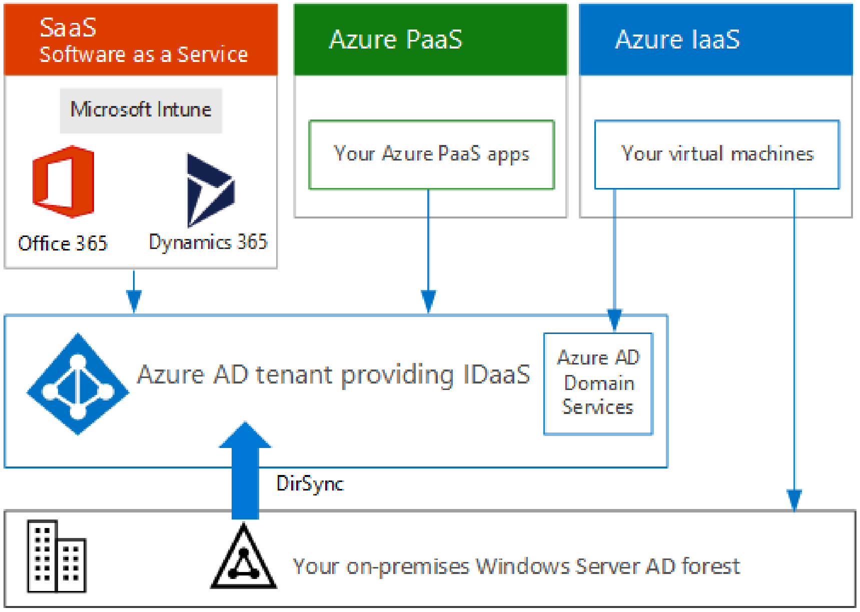
Azure Active Directory

As I mentioned before, in order to start using Microsoft Azure, we need an Account that is provided by **Azure Active Directory** (Azure AD). Azure AD in a multi-tenant service able to offer identity and access capabilities to the Azure services, and we can also connect to our on-premise AD resources.

Using Azure AD, we can implement **Single Sign On (SSO)** to cloud service, host users and group, create and manage domains, and more, the most important aspect related to Azure resources is the account.

The account can be hosted in different ways, and it depends on which type of Azure Service we use, SaaS, PaaS or IaaS. In the case of SaaS, like Office 365 and MSDN consumption, the account will be hosted in Azure AD Tenant for Microsoft 365 Subscription.

In the case of Azure PaaS and Azure IaaS consumption, the account can be hosted in one of the Azure datacentres around the world, and the account can also be synchronized with our on-premise Active Directory, the picture below will show you the relation.



The account in Azure AD is not able to access on any Azure resources until an Azure Subscription is created and bind to the account.

When we open an Azure contract for the first time, Microsoft will create at least one account in the Azure AD and an Azure Subscription tied to that account, this account owns the subscription and everything inside it, this account is Service Administrator and the Owner of the Subscription.

There are two main types of roles, Azure AD roles, and Azure RBAC roles, the first is applied at the Active Directory level, and the second at Management Group Level, it is very important to understand the difference.

Azure AD Roles

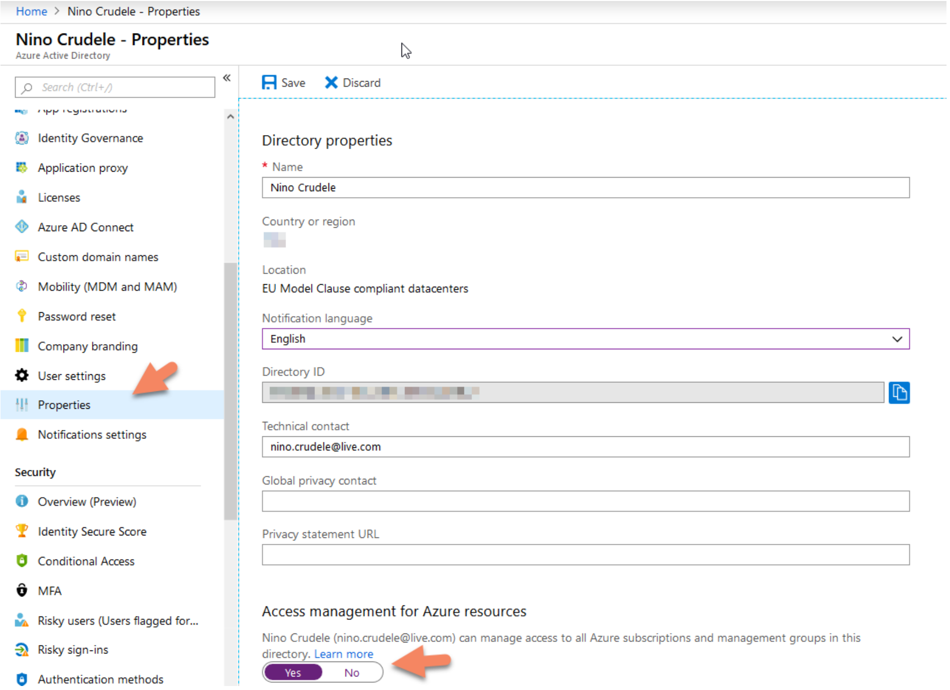
Azure AD roles and RBAC roles are two different worlds in the same universe. Any role applied at the Directory level has an effect on all the Azure resources associated to the specific Tenant, even regardless of the RBAC roles assigned to the Azure account.

The are many roles in Azure AD, but the most important role we need to consider is the Global Admin. He is the most powerful role in Azure AD.

The Global Administrator can read and modify any setting in the Azure AD organization, a privilege escalation to this account can be catastrophic for the company, but there is also a secondary privilege escalation on this role.

The Global Administrator has access to any Azure AD resource but not to any Azure Subscription, however, exists a setting in the Azure Active Directory able to activate full access to any Azure subscriptions tied to the tenant to the account, and activate this feature is very simple:

- Go in the Azure Portal and select Azure AD and Properties.
- At the bottom blade, set the Access management for Azure resources to **Yes**.



Directory properties

* Name: Nino Crudele

Country or region: [redacted]

Location: EU Model Clause compliant datacenters

Notification language: English

Directory ID: [redacted]

Technical contact: nino.crudele@live.com

Global privacy contact:

Privacy statement URL:

Access management for Azure resources

Nino Crudele (nino.crudele@live.com) can manage access to all Azure subscriptions and management groups in this directory. Learn more

[✓ Yes] No

As always, we can also use PowerShell or Azure API. You can find all the references at the link below:

- https://docs.microsoft.com/en-us/azure/role-based-access-control/elevate-access-global-admin

Activating Access management to Azure resources, we give full access of permissions to the account on the entire Azure resource landscape tied to the tenant, this about an account with full access to many Azure EA with hundreds of subscriptions around the world, this is a very dangerous setting to use.

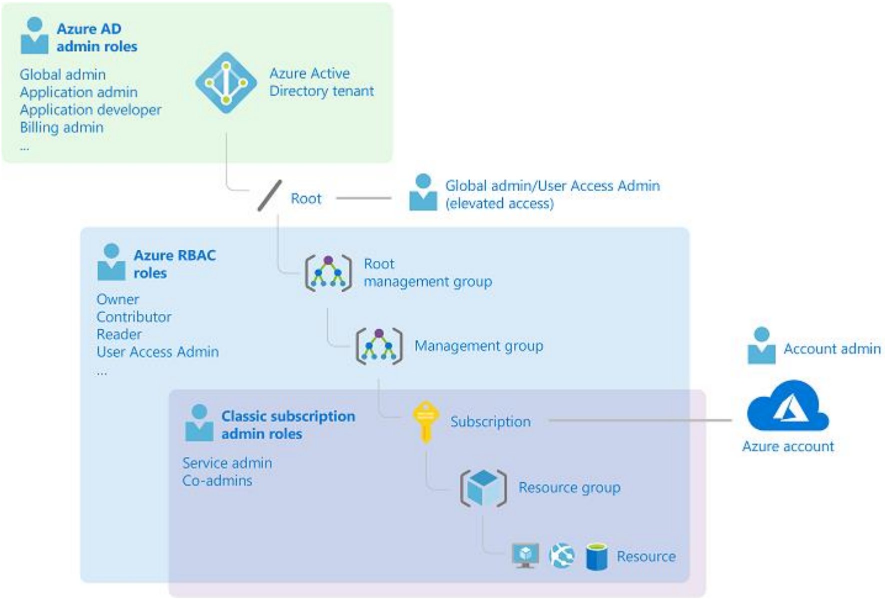
The best practice is to limit the number of people using this account, and we can also activate **Privilege Identity Management** (PIM), using PIM we can control and limit the usage of this account. However, PIM is just a temporary grant to the account, which is not limiting what the account can actually do during that period.

> You can find more information about PIM in the following: https://docs.microsoft.com/en-us/azure/active-directory/privileged-identity-management/pim-configure

Azure Roles

Before June 2018, Microsoft Azure was providing three administrator roles only: **Account Administrator**, **Service Administrator**, and **Co-Administrator**, also called the **Azure Service Manager (ASM)**, later Microsoft introduced Azure Resource Manager and **Role Base Access Control (RBAC)**, and they deprecated the using of ASM.

An Azure role identifies what a user can do into the scope, we first define the role, and after we define the scope, the picture below shows you how it works.

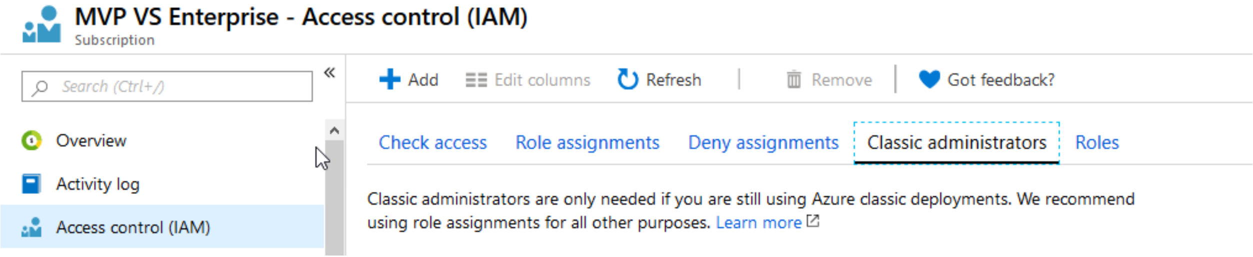


Classic Subscription Roles

Some customers are still using the classic subscription model, so it is important to know the roles and scopes.

- **Account Administrator**: We can define one Account Administrator for each Azure account, he essentially manages the billing of the subscription, but he cannot access the portal.
- **Service Administrator**: We can define one Service Administrator per Azure Subscription, this account is also the Owner of the Subscription, this account has full access in the Azure Portal.
- **Co-Administrator**: We can have up to 200 per Subscription and he has equivalent access who is assigned the Owner role but he cannot, for example, change the association of the subscription to the Azure AD

We can still use these roles, and we can manage them using PowerShell, Azure API, and Azure Portal under the Classic administrator's tab.



Classic administrators are only needed if you are still using Azure classic deployments. We recommend using role assignments for all other purposes. Learn more

Azure RBAC Roles

RBAC has been build on top of Azure Resource Manager (ARM), and it includes more than 70 roles and there are four very important RBAC roles to know:

- **Owner:** This role has full access to any Azure resource in his scope, and he can delegate access to other accounts**.**
- **Contributor:** This role can create and manage any kind of Azure resource in his scope, but it cannot delegate permissions to other accounts.
- **Reader:** This role can view any Azure resource in his scope.
- **User Access Administrator:** This role is important because it can delegate permission to other accounts so, even if not Owner, it can perform privilege escalation**.**

All the other roles have been created to manage specific Azure resources like Virtual Machines or Networking. You can find all roles in detail at the link below:

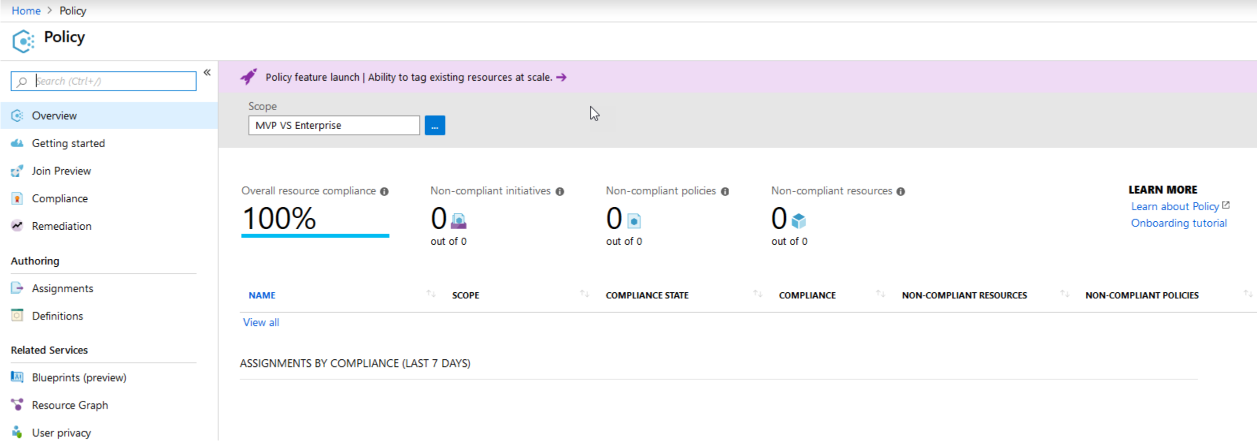
- https://docs.microsoft.com/en-us/azure/role-based-access-control/built-in-roles

Azure Policies

A **policy** is something we define to enforce a rule, and **Azure Policy** is the service to manage the policies, using that service we are able to create, delete, update and assign a policy to a specific scope.

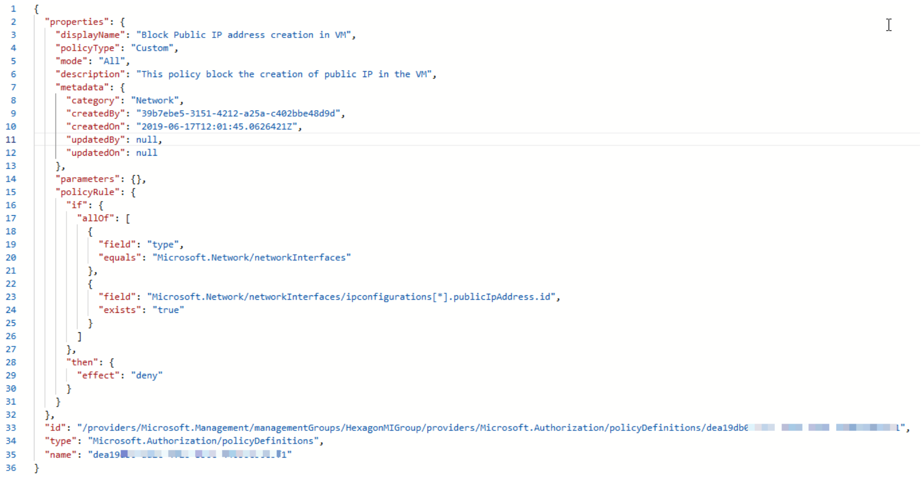
There are many built-in policies in Azure, and they are used by important assets like Security Center, Azure Advisor, Logging and more, any action we do is governed by them, no policy means no control and no governance.

We can manage the Azure Policies in the Azure Portal under Policy.



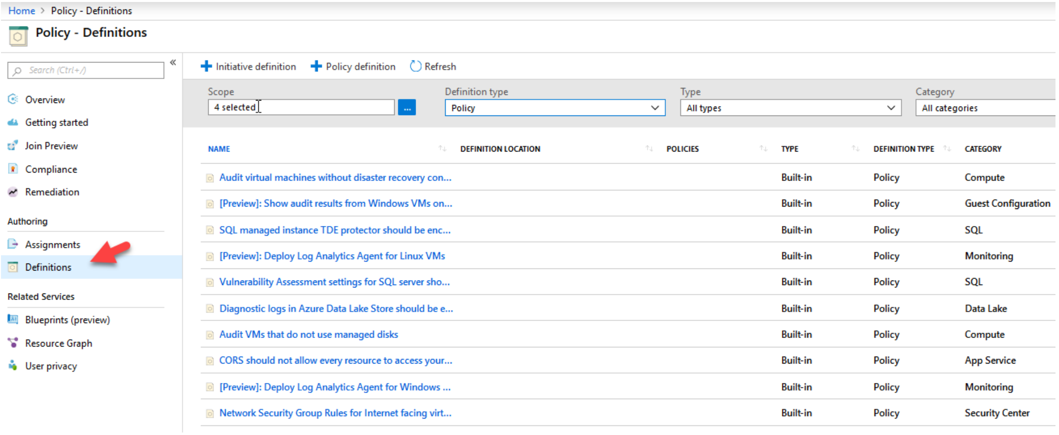
Sometimes people confuse policies with RBAC, a policy is something we assign to apply a standard, for example, a naming convention or to block people on creating public IP addresses, and we use RBAC in conjunction with policies to apply access rules. Policies are the way to apply standards in Azure, and RBAC is the way to manage the authorizations.

We can create custom policies using a **Policy Definition,** and we can implement very complex logic, a policy definition is a JSON structure, below an example of a policy I created to block people on creating public IP addresses.



```
{
  "properties": {
    "displayName": "Block Public IP address creation in VM",
    "policyType": "Custom",
    "mode": "All",
    "description": "This policy block the creation of public IP in the VM",
    "metadata": {
      "category": "Network",
      "createdBy": "39b7ebe5-3151-4212-a25a-c402bbe48d9d",
      "createdOn": "2019-06-17T12:01:45.0626421Z",
      "updatedBy": null,
      "updatedOn": null
    },
    "parameters": {},
    "policyRule": {
      "if": {
        "allOf": [
          {
            "field": "type",
            "equals": "Microsoft.Network/networkInterfaces"
          },
          {
            "field": "Microsoft.Network/networkInterfaces/ipconfigurations[*].publicIpAddress.id",
            "exists": "true"
          }
        ]
      },
      "then": {
        "effect": "deny"
      }
    }
  },
  "id": "/providers/Microsoft.Management/managementGroups/HexagonMIGroup/providers/Microsoft.Authorization/policyDefinitions/dea19db[redacted]",
  "type": "Microsoft.Authorization/policyDefinitions",
  "name": "dea19[redacted]"
}
```

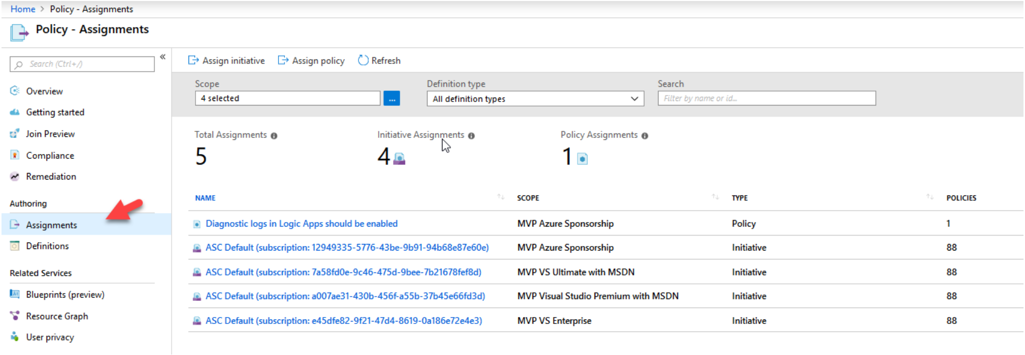
We can see all custom policies definition in Policy and Definitions, and the Category defines which type of service the policy has been created for.



Scope: 4 selected
Definition type: Policy
Type: All types
Category: All categories

NAME	DEFINITION LOCATION	POLICIES	TYPE	DEFINITION TYPE	CATEGORY
Audit virtual machines without disaster recovery con...			Built-in	Policy	Compute
[Preview]: Show audit results from Windows VMs on...			Built-in	Policy	Guest Configuration
SQL managed instance TDE protector should be enc...			Built-in	Policy	SQL
[Preview]: Deploy Log Analytics Agent for Linux VMs			Built-in	Policy	Monitoring
Vulnerability Assessment settings for SQL server sho...			Built-in	Policy	SQL
Diagnostic logs in Azure Data Lake Store should be e...			Built-in	Policy	Data Lake
Audit VMs that do not use managed disks			Built-in	Policy	Compute
CORS should not allow every resource to access your...			Built-in	Policy	App Service
[Preview]: Deploy Log Analytics Agent for Windows ...			Built-in	Policy	Monitoring
Network Security Group Rules for Internet facing virt...			Built-in	Policy	Security Center

We can assign a Policy creating a **Policy Assignment**, and we apply the assignment to a specific scope, which can be from the management group level to the single Azure resource. We can see all policies assignment selecting Policy and Assignments. In the list below, we can see a list of assignments.



Scope: 4 selected
Definition type: All definition types
Search

Total Assignments 5
Initiative Assignments 4
Policy Assignments 1

NAME	SCOPE	TYPE	POLICIES
Diagnostic logs in Logic Apps should be enabled	MVP Azure Sponsorship	Policy	1
ASC Default (subscription: 12949335-5776-43be-9b91-94b68e87e60e)	MVP Azure Sponsorship	Initiative	88
ASC Default (subscription: 7a58fd0e-9c46-475d-9bee-7b21678fef8d)	MVP VS Ultimate with MSDN	Initiative	88
ASC Default (subscription: a007ae31-430b-456f-a55b-37b45e66fd3d)	MVP Visual Studio Premium with MSDN	Initiative	88
ASC Default (subscription: e45dfe82-9f21-47d4-8619-0a186e72e4e3)	MVP VS Enterprise	Initiative	88

An initiative is a collection of policies, for example, we can create and initiative named **Public Internet Protection** and in this initiative, we will collect all the policies related to block any possible exposure on the internet from firewalls, VNet, NSG and more.

Before starting a penetration test, it is important to discuss with the customer about the situation of the Azure Policies and if all resources in the penetration testing scope are compliant, if not, then we need to take note of them.

As good Azure penetration tester must be able to scan the policies in the scope because they provide a lot of important information regarding security issues in the scope, we will speak more about that in Reconnaissance and Scanning whitepaper.

Protect and check Azure Policies is vital, this is one of the first areas attacked by a professional hacker with Azure knowledge, disabling the policies he is able to inhibit the alerting system, nobody will be notified about the attack, a good practice to protect against this attack is to monitor any policy change and create a notification system.

You can find more information about Azure Policies in the following link:
https://docs.microsoft.com/en-us/azure/governance/policy/overview

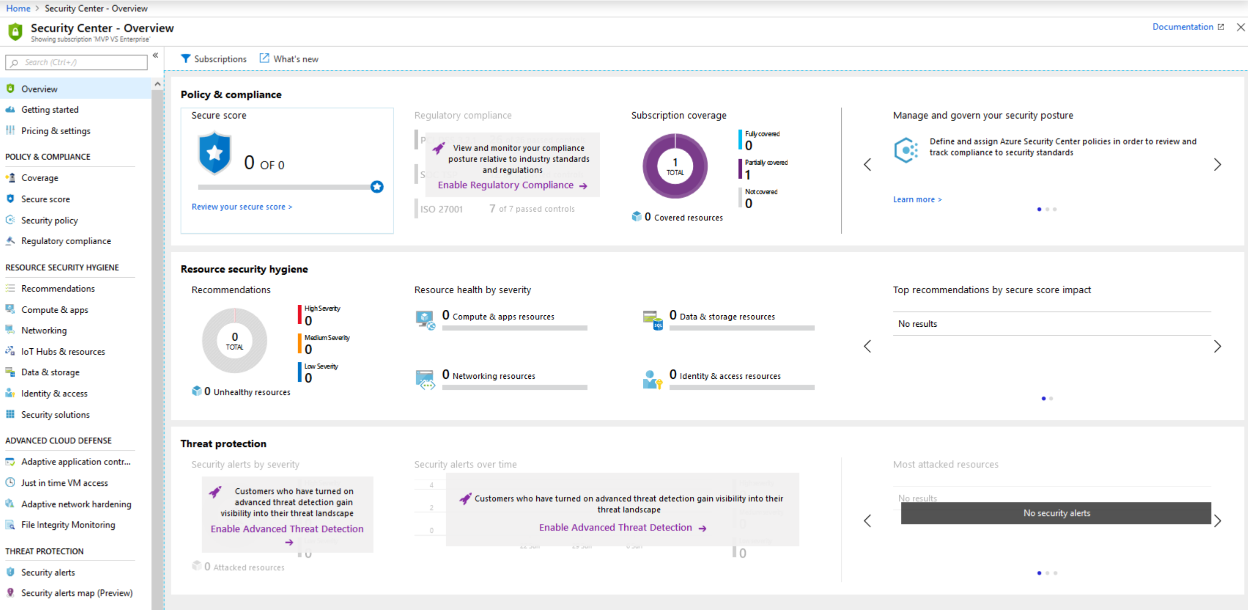
Security Center

Azure Security Center (ASC) is a SaaS service able to provide intrusion detection and intrusion prevention, and it is considered one of the most important assets in Azure security.

ASC uses all data collected from the resources like network, virtual machines and more, to assess the state of the infrastructure security, using it we can collect tons of information about the system and possible vulnerabilities. We also need to be aware if ASC is enabled and used in the system, some customers disable ASC features to speed up operational activities, and they forget to enable these features and policies later, of course, this is not a good practice but, believe me, that happens.

To access to Security Center enter the portal, click on All services and search for Security Center.

As you can see below, the ASC panel provides a lot of information about security, in resource security hygiene, we can quickly check what is not secure in the infrastructure. This is extremely useful for the customers but also for the ethical hacker.


Home > Security Center - Overview
Security Center - Overview
Showing subscription 'MVP VS Enterprise'
Documentation
Search (Ctrl+/)
Overview
Getting started
Pricing & settings
POLICY & COMPLIANCE
Coverage
Secure score
Security policy
Regulatory compliance
RESOURCE SECURITY HYGIENE
Recommendations
Compute & apps
Networking
IoT Hubs & resources
Data & storage
Identity & access
Security solutions
ADVANCED CLOUD DEFENSE
Adaptive application contr...
Just in time VM access
Adaptive network hardening
File Integrity Monitoring
THREAT PROTECTION
Security alerts
Security alerts map (Preview)
Subscriptions
What's new
Policy & compliance
Secure score
0 OF 0
Review your secure score >
Regulatory compliance
View and monitor your compliance posture relative to industry standards and regulations
Enable Regulatory Compliance →
ISO 27001
7 of 7 passed controls
Subscription coverage
1 TOTAL
Fully covered 0
Partially covered 1
Not covered 0
0 Covered resources
Manage and govern your security posture
Define and assign Azure Security Center policies in order to review and track compliance to security standards
Learn more >
Resource security hygiene
Recommendations
0 TOTAL
High Severity 0
Medium Severity 0
Low Severity 0
0 Unhealthy resources
Resource health by severity
0 Compute & apps resources
0 Data & storage resources
0 Networking resources
0 Identity & access resources
Top recommendations by secure score impact
No results
Threat protection
Security alerts by severity
Customers who have turned on advanced threat detection gain visibility into their threat landscape
Enable Advanced Threat Detection →
0 Attacked resources
Security alerts over time
Customers who have turned on advanced threat detection gain visibility into their threat landscape
Enable Advanced Threat Detection →
Most attacked resources
No results
No security alerts

We can retrieve a lot of useful information from ASC using scripting techniques and Azure APIs. We will examine that in Reconnaissance and Scanning whitepaper.

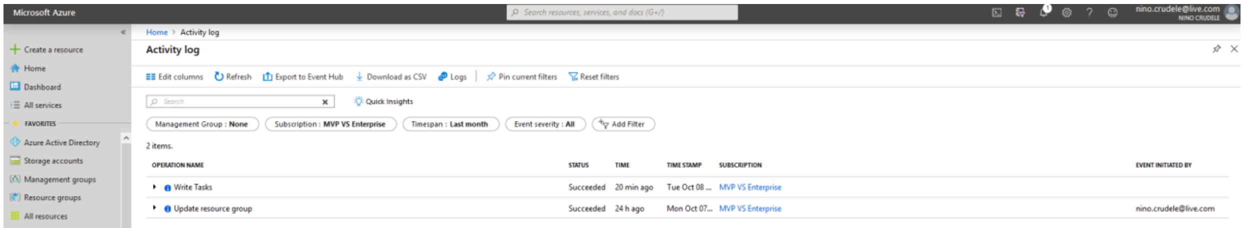
Monitoring and Logs

The amount of data tracked by Azure is massive, and all these logs are saved in Log Analytics Workspaces. These data are used by many Azure assets, Security Center, Azure Advisor, Monitor, Activity Log, and more let investigate about the potentials of that.

Azure Activity Log

Anything that happens into Microsoft Azure is tracked and saved in logs, user access, deployments, any single and minimal operation, and we can use features like Azure Activity Log to queries this information.

To access to **Activity Log**, go into the portal and search for Activity Log, in the picture below, you can see an example.



Activity log

Management Group : **None** | Subscription : **MVP VS Enterprise** | Timespan : **Last month** | Event severity : **All**

2 items.

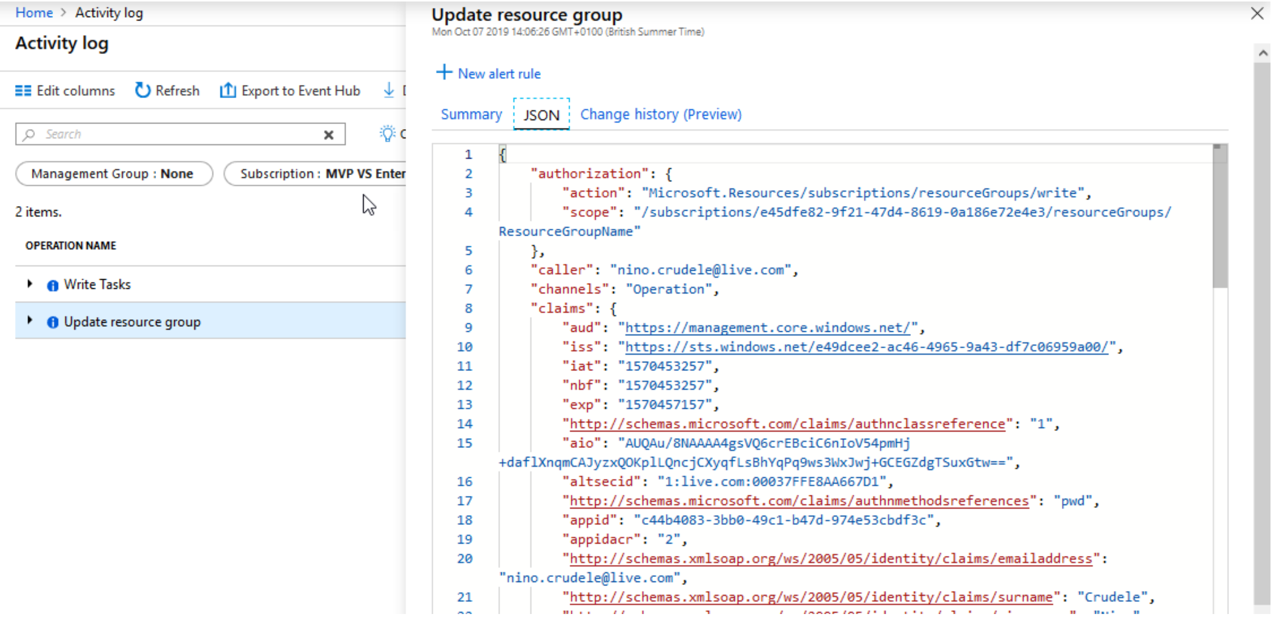
OPERATION NAME	STATUS	TIME	TIME STAMP	SUBSCRIPTION	EVENT INITIATED BY
Write Tasks	Succeeded	20 min ago	Tue Oct 08 ...	MVP VS Enterprise	
Update resource group	Succeeded	24 h ago	Mon Oct 07...	MVP VS Enterprise	nino.crudele@live.com

The default retention is 90 days, but customers may change it for operational reasons.

The number of information we can find in the Activity log is huge, a black hat able to access this single asset and with a good understanding about Microsoft Azure, he is able to understand almost everything about our infrastructure, account names, resources, critical internal issues and much more.

The activity log is also extremely helpful for the ethical hacker and for the forensic activities in order to collect information, investigate, and for very advanced troubleshooting.

Each event is logged in JSON format, and it reports in detail any kind of information.



Activity log

2 items.

OPERATION NAME
Write Tasks
Update resource group

Update resource group

Mon Oct 07 2019 14:06:26 GMT+0100 (British Summer Time)

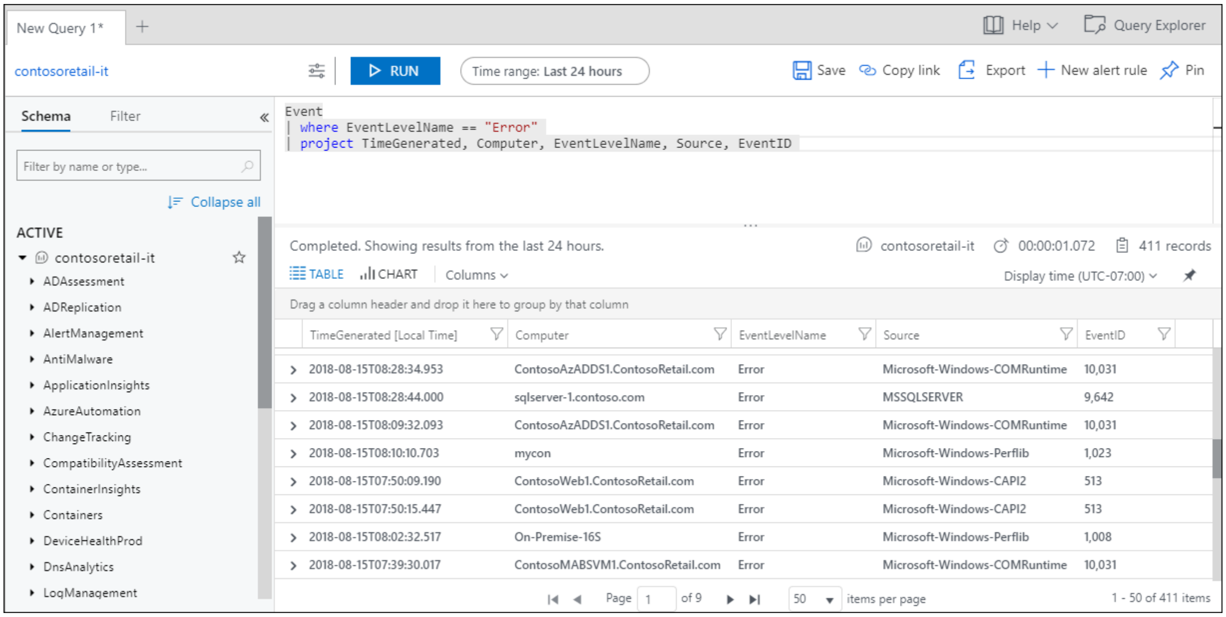
```
{
    "authorization": {
        "action": "Microsoft.Resources/subscriptions/resourceGroups/write",
        "scope": "/subscriptions/e45dfe82-9f21-47d4-8619-0a186e72e4e3/resourceGroups/
ResourceGroupName"
    },
    "caller": "nino.crudele@live.com",
    "channels": "Operation",
    "claims": {
        "aud": "https://management.core.windows.net/",
        "iss": "https://sts.windows.net/e49dcee2-ac46-4965-9a43-df7c06959a00/",
        "iat": "1570453257",
        "nbf": "1570453257",
        "exp": "1570457157",
        "http://schemas.microsoft.com/claims/authnclassreference": "1",
        "aio": "AUQAu/8NAAAA4gsVQ6crEBciC6nIoV54pmHj
+daflXnqmCAJyzxQOKplLQncjCXyqfLsBhYqPq9ws3WxJwj+GCEGZdgTSuxGtw==",
        "altsecid": "1:live.com:00037FFE8AA667D1",
        "http://schemas.microsoft.com/claims/authnmethodsreferences": "pwd",
        "appid": "c44b4083-3bb0-49c1-b47d-974e53cbdf3c",
        "appidacr": "2",
        "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress":
"nino.crudele@live.com",
        "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname": "Crudele",
```

We will see more about Activity Logs in Reconnaissance and Scanning whitepaper.

Azure Monitor

Azure Monitor is the central operation point for logging, from here we can access to other logging areas like Metrics, Alerts, and especially Logs.

In the Logs, we can create queries using a query language named KUSTO, below an example.



```
Event
| where EventLevelName == "Error"
| project TimeGenerated, Computer, EventLevelName, Source, EventID
```

Completed. Showing results from the last 24 hours. contosoretail-it 00:00:01.072 411 records

TABLE
CHART
Columns
Display time (UTC-07:00)
Drag a column header and drop it here to group by that column

TimeGenerated [Local Time]	Computer	EventLevelName	Source	EventID
2018-08-15T08:28:34.953	ContosoAzADDS1.ContosoRetail.com	Error	Microsoft-Windows-COMRuntime	10,031
2018-08-15T08:28:44.000	sqlserver-1.contoso.com	Error	MSSQLSERVER	9,642
2018-08-15T08:09:32.093	ContosoAzADDS1.ContosoRetail.com	Error	Microsoft-Windows-COMRuntime	10,031
2018-08-15T08:10:10.703	mycon	Error	Microsoft-Windows-Perflib	1,023
2018-08-15T07:50:09.190	ContosoWeb1.ContosoRetail.com	Error	Microsoft-Windows-CAPI2	513
2018-08-15T07:50:15.447	ContosoWeb1.ContosoRetail.com	Error	Microsoft-Windows-CAPI2	513
2018-08-15T08:02:32.517	On-Premise-16S	Error	Microsoft-Windows-Perflib	1,008
2018-08-15T07:39:30.017	ContosoMABSVM1.ContosoRetail.com	Error	Microsoft-Windows-COMRuntime	10,031

Page 1 of 9 50 items per page 1 - 50 of 411 items

We can execute KUSTO queries using Azure Data Explorer API, which support communication endpoints, like REST API and MS-TDS, which is a subset of the Microsoft Tabular Data Stream (TDS) protocol, used by the Microsoft SQL Server products, we will see more detail in Reconnaissance and Scanning whitepaper with scanning.

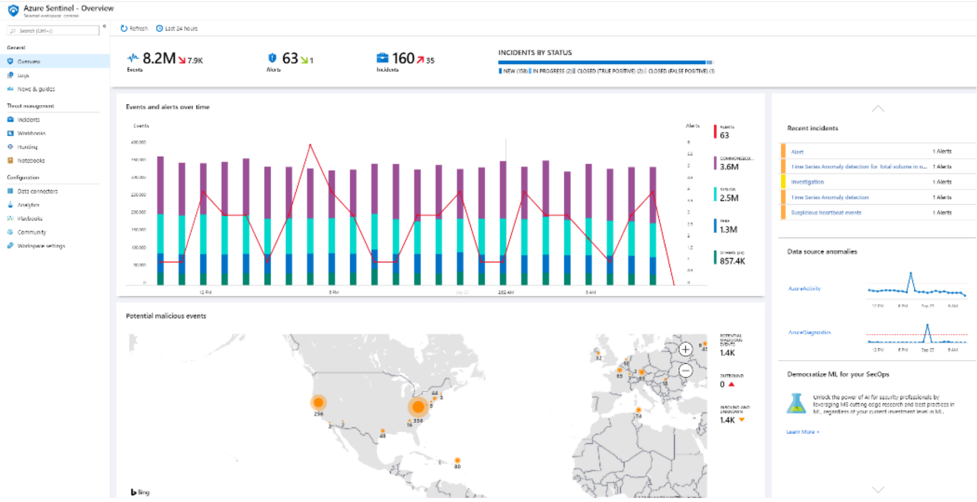
Learning KUSTO is not a must, but it is a great plus to have for any ethical hacker working in Azure, you can find more detail at the link below:

- https://docs.microsoft.com/en-us/azure/kusto/concepts/

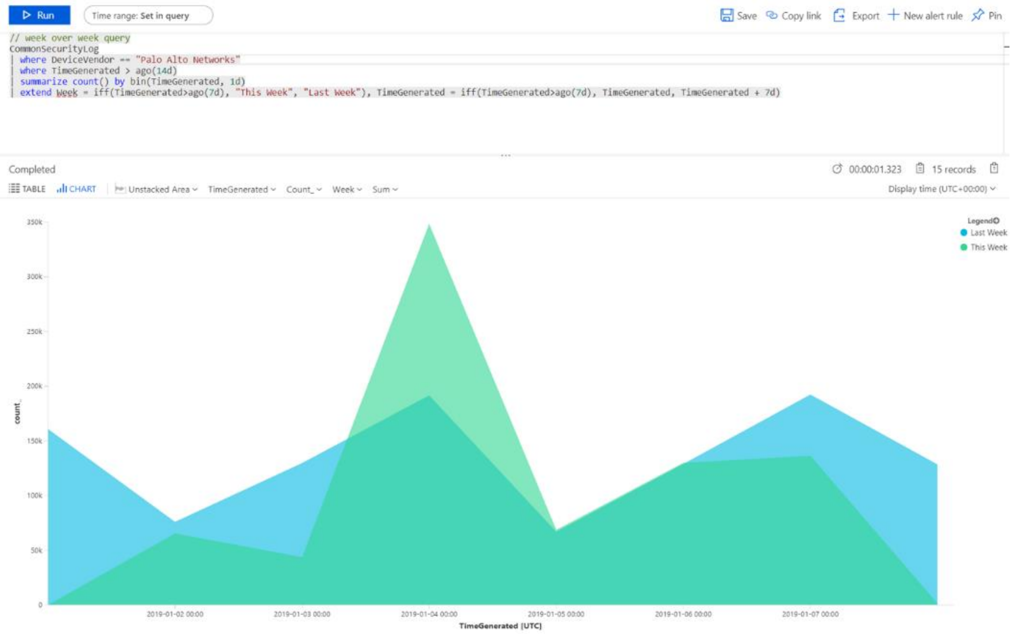
Azure Sentinel

I'd like to spend some word about this new entry in Microsoft Azure, **Azure Sentinel** is the internal **SIEM (Security Information and Event Management)** system.

Azure Sentinel uses **AI (Artificial Intelligence)** to elaborate the data collected to detect the attacks, it also provides amazing capabilities to track and investigate the attack and respond very quickly using tasks and automation, this is very useful for ethical hacking investigation and forensic.



We can also use KUSTO queries for advanced troubleshooting.



```
// week over week query
CommonSecurityLog
| where DeviceVendor == "Palo Alto Networks"
| where TimeGenerated > ago(14d)
| summarize count() by bin(TimeGenerated, 1d)
| extend Week = iff(TimeGenerated>ago(7d), "This Week", "Last Week"), TimeGenerated = iff(TimeGenerated>ago(7d), TimeGenerated, TimeGenerated + 7d)
```

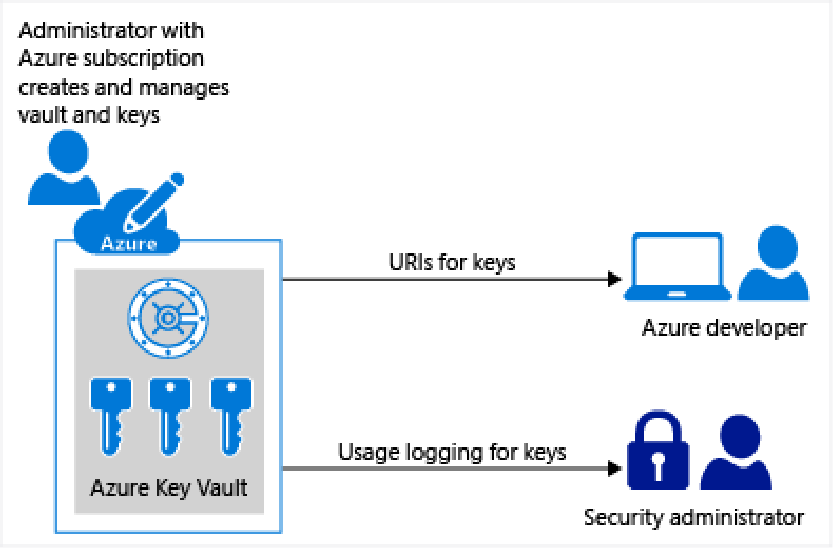
Completed

00:00:01.323 15 records

You can find more information about Azure Sentinel at the following link:
https://docs.microsoft.com/en-us/azure/sentinel/quickstart-get-visibility

Azure Key Vault

Azure Key Vault (Azure KV) is the place to centralize and control the storage of any application secret, in Azure KV developers save passwords and security settings, and people save certificates and any other sensitive information.



All data are encrypted using industry-standard algorithms, key lengths, and hardware security modules (HSMs).

The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. It is immediately understood that this is another important target for any hacker. For that reason, it is important to activate the activity monitor for any vault after we can send the logs to Azure monitor and we can easily create KUSTO queries to create any alert and notification.

The Penetration Testing

Delivering a proper penetration test is one of the most challenging tasks, especially against a complex infrastructure like Microsoft Azure. I see people mentioning **OWASP (Open Web Application Security Project)** top 10 vulnerabilities, or Fuzzy testing, website scanning, or port scanning, but this is a very small part of a pentesting in Azure.

The world is changing, and the Security IT with it, with the evolution of IT and the cloud, many things have changed, no more boundaries or hardware, everything is virtual, and our data are everywhere, in data centers, mobile devices, computers, also hackers evolved.

We need to look at penetration testing in a new way because pentesting evolved as well. We cannot use classic techniques anymore. We need to think and operate widely, with creativity and intuition, and using the best tools for the appropriate scope.

Cloud means enterprise, scaling, complexity, services abstraction, we may need to conduct a pentest to a single resource, a website or an entire Azure Enterprise Agreement with multiple subscriptions and tenants, this is definitely challenging, and it requires good Azure experience and knowledge. If you aspire to become a good Azure pentester, then I recommend you to study Microsoft Azure, starting from the infrastructure first, the networking, and then the security and governance. You must understand Azure and the logic behind it.

When I engaged Azure for the first time, I was working in the integration technology area, playing with services like Service Bus, BizTalk Server, and later using many other technologies like Event Hubs or Logic app. I was confident about my expertise and knowledge on Microsoft Azure, until I joined my actual company, leading an Azure infrastructure in a global company is extremely challenging and I realized how much I didn't know about Azure, the complexity of the topics like governance, strategies, infrastructure, security and the number of features is just unbelievable.

One of the most important security best practices is to always engage a third-party company for penetration testing and not just using the internal security department. I had the opportunity to meet many security companies and ethical hackers, and during this time I realized that one of the common problems is the lack of knowledge and experience in Microsoft Azure, many people approach to the pentesting with the same mindset as on-premise technologies and this is a huge mistake.

Microsoft Azure is a cloud service infrastructure, with a very large number of assets and features, it is a completely different concept from the classic on-premise datacentre approach, we don't attack physical hardware, switches, and routers, we attack logical services.

From the Art of War, If you know your enemies and know yourself, you will not be imperiled in a hundred battles, I mean that we need to know Microsoft Azure in order to defeat and protect it, no doubts.

The on-premise physical network we know, in Microsoft Azure, is a Virtual Network, with subnets and we have Network Security Groups instead of Physical Firewalls. I don't want to spend too much explaining what is a pentest, there is a large number of literature about this topic and you probably already have a good idea about that. The scope of this whitepaper is to provide you with my personal experience and understanding of Microsoft Azure from an ethical hacker perspective and give you a wide vision of what you need to know to perform a pentesting in Microsoft Azure, the strategies and phases.

Penetration Testing Strategies

Let start speaking about the three mains canonical pentesting strategies, Black Box, White Box and Grey Box, and pro and cons.

The picture below represents these three main strategy principals.

In the Black Box, we simulate what an average hacker may compromise, hitting the external endpoints. We don't have any internal information about the target. We have the external endpoint only, for example, it can be a REST API or a website URL or external storage accounts endpoint. I discourage the reader from running wide scanning activity on Azure. This is extremely dangerous and illegal, stick with the external endpoints provided by the customer.

A Black Box pentest can be quick, and it may require a lot of manual activity, experimentation, social engineering, and this is actually the only real way to simulate what a hacker could find without any knowledge of the target. My recommendation is to avoid the Black Box pentesting, and for many reasons, the most important one is that Azure is full of endpoints. A blind external scanning and attack may compromise you and another external company with serious legal consequences.

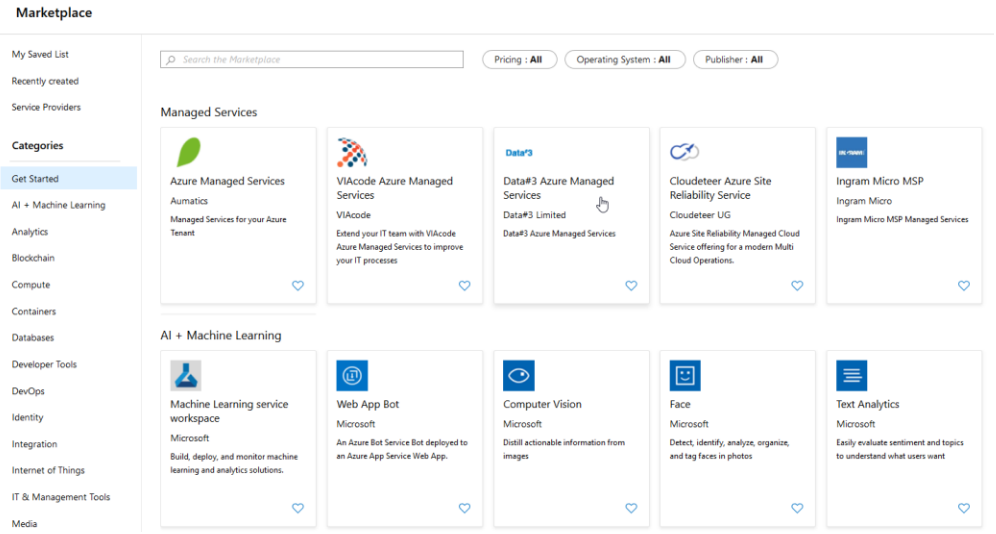
In the Grey Box, we normally have user credentials, and we can potentially do privileged escalation, we also have a good understanding about the networking, this strategy is very effective because we can run internal security functional tests. The advantage is that we have a limited amount of information and we are not completely blind and we can easily perform internal attacks much easier and quicker, and it is a more productive approach compared to the Black Box but, it is a different thing because we are not in a Black Box mindset.

The Gray Box is the best approach to simulate internal attacks, for example, from an internal user.

In the White Box, we have full knowledge of the solution and infrastructure, and we also have full access to all the resources and documentation. The White Box one is the most recommended because we are able to support the customer at 100%, it is the most productive and valuable pentesting strategy for both. The White Box pentesting is a very productive way to collaborate with the customer, and I think that a very good understanding of Microsoft Azure is a big plus, especially in the security and infrastructure areas. I had the opportunity to be part of meetings and calls with other security companies to discuss the engagements with my company, I had the opportunity to meet many good ethical hackers, and the only issue I felt was the lack of knowledge on Microsoft Azure, most of the time that has been a gap and a bottleneck. Many ethical hackers approach to Azure with the on-premise datacentre mindset, asking about endpoints, protocols, authentications and authorizations, they collect all this information and they start the attack to the infrastructure.

The customers do not always have a clear idea about the technologies involved and all the endpoints exposed, and we need to be able to provide guidance and support them on that, I give you some examples.

When you create a virtual machine in Azure, you can also create public IP, and since some months ago, that was default behavior so, a public IP was created every time you were spinning up a new VM. Unfortunately, many people were not aware of that. Another example is the marketplace, many companies provide their product and software through Azure Marketplace, in the marketplace people can easily find a complete set of software and infrastructure ready to use like Elasticsearch cluster and Kubernetes, or FortiGate Firewall and many more.



Managed Services

Azure Managed Services
Aumatics
Managed Services for your Azure Tenant

VIAcode Azure Managed Services
VIAcode
Extend your IT team with VIAcode Azure Managed Services to improve your IT processes

Data#3 Azure Managed Services
Data#3 Limited
Data#3 Azure Managed Services

Cloudeteer Azure Site Reliability Service
Cloudeteer UG
Azure Site Reliability Managed Cloud Service offering for a modern Multi Cloud Operations.

Ingram Micro MSP
Ingram Micro
Ingram Micro MSP Managed Services

AI + Machine Learning

Machine Learning service workspace
Microsoft
Build, deploy, and monitor machine learning and analytics solutions.

Web App Bot
Microsoft
An Azure Bot Service Bot deployed to an Azure App Service Web App.

Computer Vision
Microsoft
Distill actionable information from images

Face
Microsoft
Detect, identify, analyze, organize, and tag faces in photos

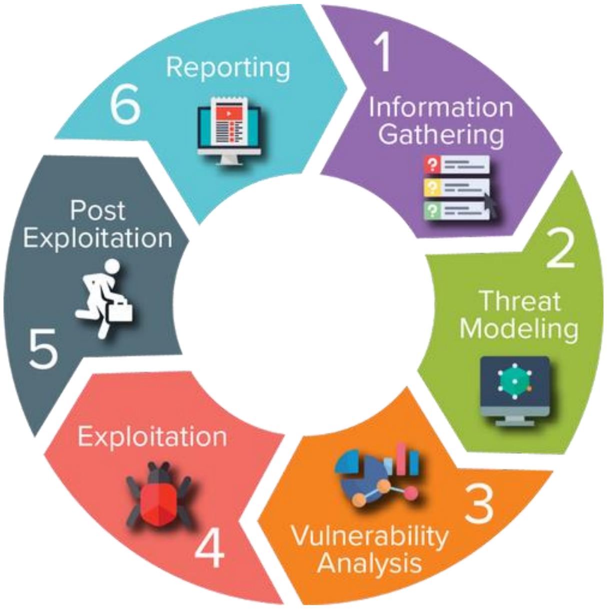
Text Analytics
Microsoft
Easily evaluate sentiment and topics to understand what users want

Some of these appliances are very complex, and people may deploy them without a clear idea about which Azure resources and endpoint will be created, this is absolutely understandable, for that reason we need to be able to support the customer on that, and we can use many tools and techniques, we will see some scanning techniques in the Reconnaissance and Scanning whitepaper.

Another important tip is to speak with people in the customer's team, developers, program managers, about everything and listen to them very carefully, and we can get a lot of useful input and information able to drive our researches and being more incisive in our tests.

The Penetration Testing Phases

We have many different phases and stages in a penetration test, and they can also change depending on specific cases and engagement. In the picture below, you can see a classic penetration testing lifecycle.



I want to recap all these steps in the four most important main phases and collocate them with the related actions we need to take in Microsoft Azure. These main phases are always present in any penetration test in Microsoft Azure.

There are four major phases:

- **Engagement**, during this phase, we agree with the customer about legal aspects, the scope of the engagement, and more.
- **Authentication and authorization**, in the case of White and Grey Box pentesting we need an authentication method in order to access the scope, and we need to define which authorization level we like to use.
- **Reconnaissance and Scanning**, during this phase, we perform a quick assessment, and we get a better vision of the scope.
- **Reporting**, we report our findings.

The Engagement

The first phase in any penetration test is the engagement, and there are two main actors to manage, the customer and Microsoft. The first rule of engagement is to agree and signing a non-disclosure agreement with the customer, this is important in order to protect both, and second, we need to define the scope and type of engagement and penetration test.

We also need to obtain the authorization for the assessment and the penetration test, also named **Get Out of Jail card (GOJ card)**, signed by the **Chief Security Officer (CSO)** and also approved by a second authority like the **Chief Technology Officer (CTO)** or other relevant authorities, the document must include the name of all testers and approvers and signatures, we also need to specify the reason of the test and any other relevant information. Without the GOJ card you are conducting illegal activity, don't trust any other form of internal authorization document, during the test you may incur in very critical situations like finding very sensitive data regarding the company and you will need to be able to explain the reason why you accessed to these data.

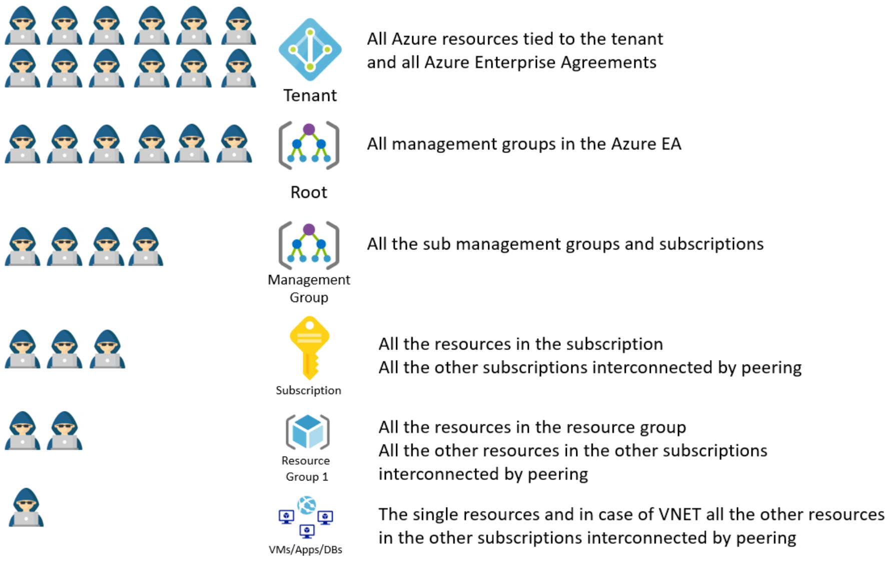
You can find a very good example of the GOJ card at the following link:
http://www.counterhack.net/permission_memo.html

The Scope

It is necessary to define a scope, and in Azure, we have different scope levels, we can use the natural structure of Azure to define these.

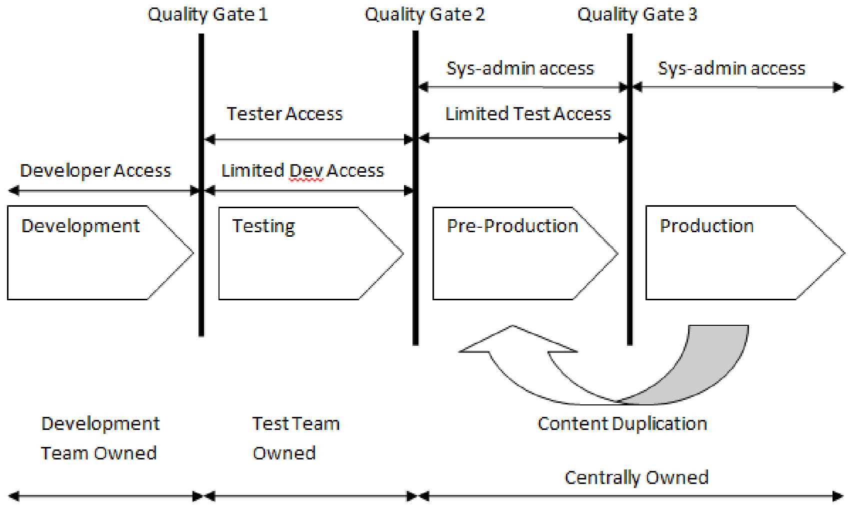
I like to repeat myself here, don't execute scanning in unauthorized scopes and don't execute any kind of attack in any resources if you are not sure that this is in your scope, the Azure Account or the Service Principal is not your scope, this account may have access to external scopes, keep always in mind that, or you can occur in serious security and legal issues.

Below you can see the different scopes with a description, the number of hackers also determines the attack damage.


Tenant
All Azure resources tied to the tenant and all Azure Enterprise Agreements
Root
All management groups in the Azure EA
Management Group
All the sub management groups and subscriptions
Subscription
All the resources in the subscription
All the other subscriptions interconnected by peering
Resource Group 1
All the resources in the resource group
All the other resources in the other subscriptions interconnected by peering
VMs/Apps/DBs
The single resources and in case of VNET all the other resources in the other subscriptions interconnected by peering

As you can see, a privilege escalation at the tenant level can compromise the entire company. We may rarely conduct a pentest at the tenant level, but we may need to provide an assessment at this level before our pentest, so we also need to consider this scope level.

The scope may start from a single endpoint to an entire management group, and the surface can be huge. In any case, it is necessary to identify the environment type and the type of data, below a basic development lifecycle.



The Pre-Production environment is the specular representation of production, and it is the best stage to use to conduct a penetration test, it is recommended to avoid a pentest in the production environment unless specific and critical reasons and in according with the customer.

In production, we may need to conduct forensic activities, especially after an attack, or we may need to conduct a security assessment.

Backup and Disaster Recovery

We may assume the customer has a proper backup and disaster recovery strategy in place, especially in big and critical companies or banks but, believe me, you may be surprised by the answer you get if you ask.

Is a customer responsibility to provide us with these pieces of information?, in my opinion, it is not, it is our responsibility to check if the environment and data we are going to attack are properly backup and if the customer is able to rebuild the environment, especially in case of a very invasive penetration test.

We cannot check if the backup and disaster recovery are well done and working properly, this is not our concern and it requires specific knowledge of the environment and the solution, but we need to ensure that everything is in place and we need to include that in our engagement with a specific paragraph regarding the backup.

The presence of a proper backup strategy is also part of our pentest, and it is related to one of the most dangerous and frequent attacks, the ransomware attack.

The ransomware, from ransom and ware as diminutive of malware, it is a very dangerous class of malware able to compromise the entire company business, and it is usually activated by Trojan horses, the payload may lock the computers or, in the worse scenarios, encrypt any sensitive data in the company.

What can we recommend to the company to avoid this threat?

- Make sure the anti-virus software is updated with the latest signatures in all servers and computer, Microsoft Azure offers great features like the Microsoft Defender Advanced Threat Protection.
- Check that the operating system and application software patches are up to date, we can do that in Azure activating VM agents.
- Install a firewall, both inbound and outbound, on each user's PC and Azure VM.

However, these strategies are not enough, backing up the data is the most effective way to combat ransomware infection. Since the attackers hit their victims by encrypting valuable files and data and leaving them inaccessible, thanks to backups it is possible to restore the files once the infection has been cleaned up.

Customer Engagement

The customer is always right, this is a famous motto popularised by Harry Gordon Selfridge and fellows in 1900, and actually many people use this motto quite frequently, but in my opinion, this is not always true, at least regarding penetration testing.

My first rule is, never trust the customer information, sometimes they don't even have a very good expertise in Azure, and most of the time they don't have a clear idea and vision about what they have in Azure, and we cannot blame the customer for that, this is a classic consequence of the frenetic way to deliver solutions. At this stage, our first mission is to support the customer on understanding the attack surface in respect of all the technologies involved, and the second is to perform a first security assessment.

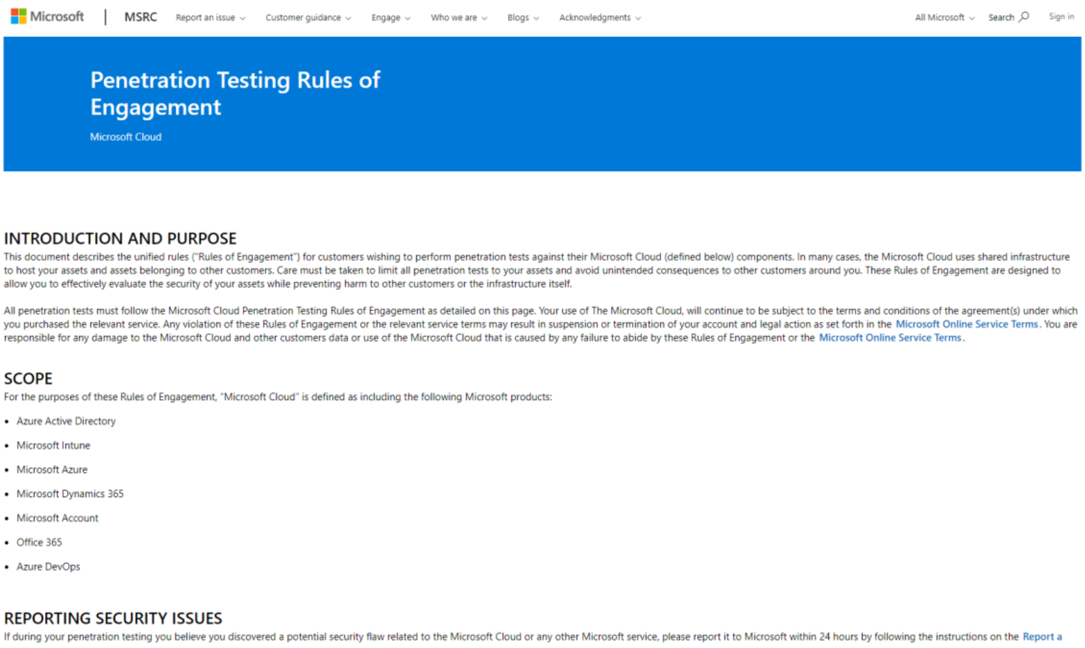
There is not a reason to start a penetration test if the Azure security assets and policies are not properly activated, we take the risk of receiving a lot of false positives, we will speak more about them during the phases.

The Microsoft Engagement

Microsoft Azure is hosting the infrastructure that you are attacking, so theoretically you are not attaching the customer only, but you are also attacking Microsoft. From the Microsoft website, as of June 15, 2017, Microsoft no longer requires pre-approval to conduct a penetration test against Azure resources.

However, there are specific rules of engagement that you need to respect and you can find all of these rules in the **Penetration Testing Rules of Engagement portal**, below the link:

- https://www.microsoft.com/en-us/msrc/pentest-rules-of-engagement?rtc=3



Penetration Testing Rules of Engagement

Microsoft Cloud

INTRODUCTION AND PURPOSE

This document describes the unified rules ("Rules of Engagement") for customers wishing to perform penetration tests against their Microsoft Cloud (defined below) components. In many cases, the Microsoft Cloud uses shared infrastructure to host your assets and assets belonging to other customers. Care must be taken to limit all penetration tests to your assets and avoid unintended consequences to other customers around you. These Rules of Engagement are designed to allow you to effectively evaluate the security of your assets while preventing harm to other customers or the infrastructure itself.

All penetration tests must follow the Microsoft Cloud Penetration Testing Rules of Engagement as detailed on this page. Your use of The Microsoft Cloud, will continue to be subject to the terms and conditions of the agreement(s) under which you purchased the relevant service. Any violation of these Rules of Engagement or the relevant service terms may result in suspension or termination of your account and legal action as set forth in the Microsoft Online Service Terms. You are responsible for any damage to the Microsoft Cloud and other customers data or use of the Microsoft Cloud that is caused by any failure to abide by these Rules of Engagement or the Microsoft Online Service Terms.

SCOPE

For the purposes of these Rules of Engagement, "Microsoft Cloud" is defined as including the following Microsoft products:

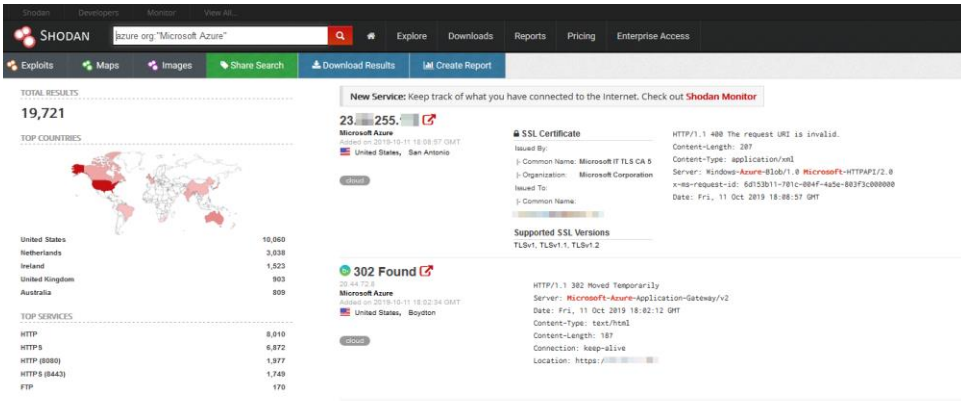
- Azure Active Directory
- Microsoft Intune
- Microsoft Azure
- Microsoft Dynamics 365
- Microsoft Account
- Office 365
- Azure DevOps

REPORTING SECURITY ISSUES

If during your penetration testing you believe you discovered a potential security flaw related to the Microsoft Cloud or any other Microsoft service, please report it to Microsoft within 24 hours by following the instructions on the Report a

The most important are:

- **You cannot execute Denial-of-Service attack (DoS attack)** in any way or form, and some tools may generate a big amount of data during the scanning, be careful about what exactly the tool does and set the proper setting in order to avoid of generating a DoS attack to the cloud.
- **Don't scan widely and uncontrolled**, stick with the scope, and if you see external or unknown DNS or endpoints, then check them very carefully and agree with the customer before scanning anything.
  - For example, you can use shodan.io for particular reconnaissance operations. However, shodan can be very powerful and extremely dangerous, below a simple wide search on Microsoft Azure.



New Service: Keep track of what you have connected to the Internet. Check out **Shodan Monitor**

TOTAL RESULTS

19,721

TOP COUNTRIES

United States	10,060
Netherlands	3,038
Ireland	1,523
United Kingdom	903
Australia	809

TOP SERVICES

HTTP	8,010
HTTPS	6,872
HTTP (8080)	1,977
HTTPS (8443)	1,749
FTP	170

23.[redacted]255.[redacted]
Microsoft Azure
Added on 2019-10-11 18:08:57 GMT
United States, San Antonio

SSL Certificate
Issued By:
|- Common Name: Microsoft IT TLS CA 5
|- Organization: Microsoft Corporation
Issued To:
|- Common Name: [redacted]

Supported SSL Versions
TLSv1, TLSv1.1, TLSv1.2

```
HTTP/1.1 400 The request URI is invalid.
Content-Length: 207
Content-Type: application/xml
Server: Windows-Azure-Blob/1.0 Microsoft-HTTPAPI/2.0
x-ms-request-id: 6d153b11-701c-004f-4a5e-803f3c000000
Date: Fri, 11 Oct 2019 18:08:57 GMT
```

302 Found
20.44.72.8
Microsoft Azure
Added on 2019-10-11 18:02:34 GMT
United States, Boydton

```
HTTP/1.1 302 Moved Temporarily
Server: Microsoft-Azure-Application-Gateway/v2
Date: Fri, 11 Oct 2019 18:02:12 GMT
Content-Type: text/html
Content-Length: 187
Connection: keep-alive
Location: https:/[redacted]
```

You can easily get out of scope results. On the other side, it can be useful for wide scanning techniques. The search above is a broadcasting scanning across the Azure cloud, like AM radio broadcasting. A hacker can easily broadcast the RDP ports 3389 and try to attack the machine. It is quite simple executing the query below:

```
azure org:"Microsoft Azure" port:"3389"
```

RDP 3389 and an SSH 22 port are attacked after the first minute has been published on the internet.

I think that is a given to tell you how powerful this tool can be in good hands and a smart guy, but remember, it is not the tool that makes the hacker a bad one. It is how the tool is used and the scope.

For the last time, the broadcasting scan is not prohibited, scan any endpoint without approval is illegal, you may discover that you are scanning a government or a CIA server hosted in Azure, and I don't think they will be very happy about that.

Another important aspect is the **licensing**. I am referring now about the Azure CSP licensing type. I mentioned bout this license, which is used by Azure resellers, if we are engaged by a reseller, we need to check very carefully what we are going to attack and we need to clarify any possible dependency and permission between the reseller and the owner of the infrastructure hosted.

Authentication and Authorization

In the case of Black box testing, we don't need this information, but in the case of White and Grey box, we do.

There are two phases to care about, the **authentication**, which is the way we are authenticated by the system, and the **authorization,** which is what we are authorized to do in the system after we are authenticated.

There are two different deployment models in Azure, the **Azure Service Manager (ASM),** which was used in the past and now almost deprecated, and the most important one, the **Azure Resource Manager (ARM)** model.

Using ARM, we can authenticate in the system in two options, by AAD Application Registration and the Azure AAD account, and we use Role-Based Access Control (RBAC) to handle the Authorization. The customer should provide us with the account but it is important for us to know how to create them and how they work.

The Application Registration

The **Application Registration** is an identity used by an application, services or automated tools, and it is very useful to authenticate during scripting and to manage a not interactive authentication method.

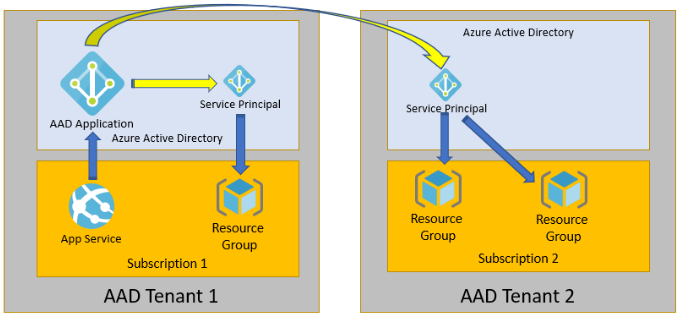
It is the best way to authenticate our automated tools, our code, and our scripts. Simply speaking it is considered as an account for services and applications.

The application registration is composed of two objects, an AAD application, and a Service principal.

- The AAD application resides in Azure Active Directory, and it identifies the application in the tenant.
- The Service Principal is created when the application object gains access to the Azure AD and Azure resources, and it defines the policies and the authentication features, simply speaking, the AAD application represents our application or services and the service principal what we can do in Azure.

The application is global, and the service principal is local for specific tenant and scope, they are both tied, any change in the AAD application will reflect the Service principal.

We use the Service principal to authenticate our code, and to define and manage the authorizations thought RBAC, we can have one AAD application and multiple Service principal, below a representation.



As always, we have many methods to create a Service Principal account, and the most used are by using the Azure Portal, PowerShell, Az CLI, and Azure API.

Azure AAD Account

I already explained about the Azure AD account, and we use this account for interactive authentication, for example, to authenticate into the Azure Portal or a Web application portal.

In the case of White and Grey Box, ask the customer to provide one AD account to use, and with the Read-only role, we don't want to risk to damage any internal resource.

Reconnaissance and Scanning in Azure

We have tons of options and tools, some of them are good for specific scopes others not, and most important thing, we need to know where to scan and why.

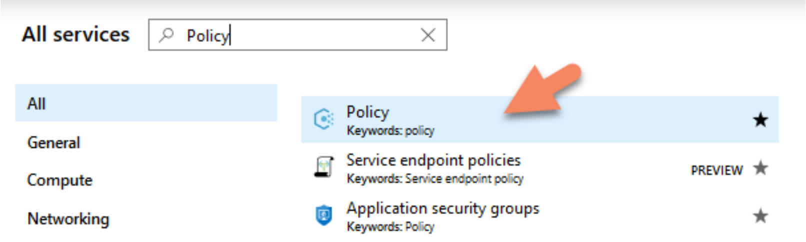
During Reconnaissance and Scanning whitepaper we will examine some of the most creative and important techniques, I said creative because creativity is the top quality to use during this phase, we can execute scanning at any level in Azure, Enterprise Agreement, Management Group, Subscription, Resource and Endpoint level.

We cannot simply trust what the customer communicates to us. I saw a pentesting company executing a test on an Azure solution and scanning the only one public IP address provided by the customer, and a simple scanner was able to show other six public IP exposed in the Resource Group, there is no reason on providing an approximate result unless the customer explicitly require just a pentest on a specific endpoint only.

We need to check all the important security assets and areas and support the customer on solving possible issues, below the most important.

Azure Policies

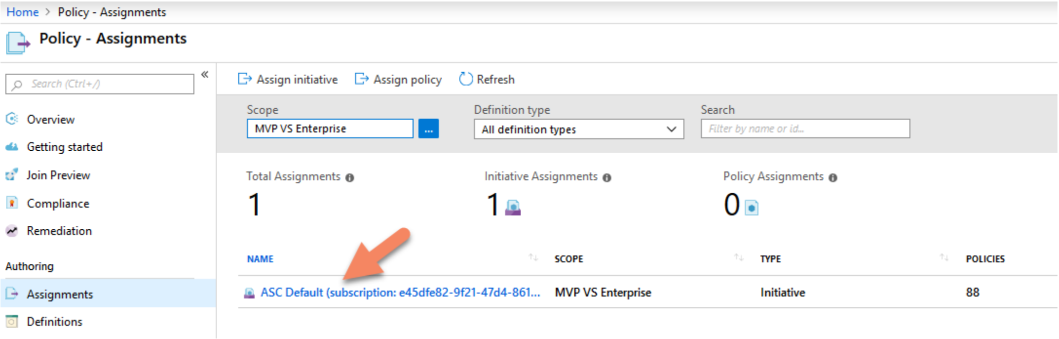
Enter in the Azure Portal and search for Policy.



All services: Policy

- Policy — Keywords: policy
- Service endpoint policies — Keywords: Service endpoint policy
- Application security groups — Keywords: Policy

Click on assignment, and you should see Azure Security Center default initiative and in case any other initiative created by the customer.

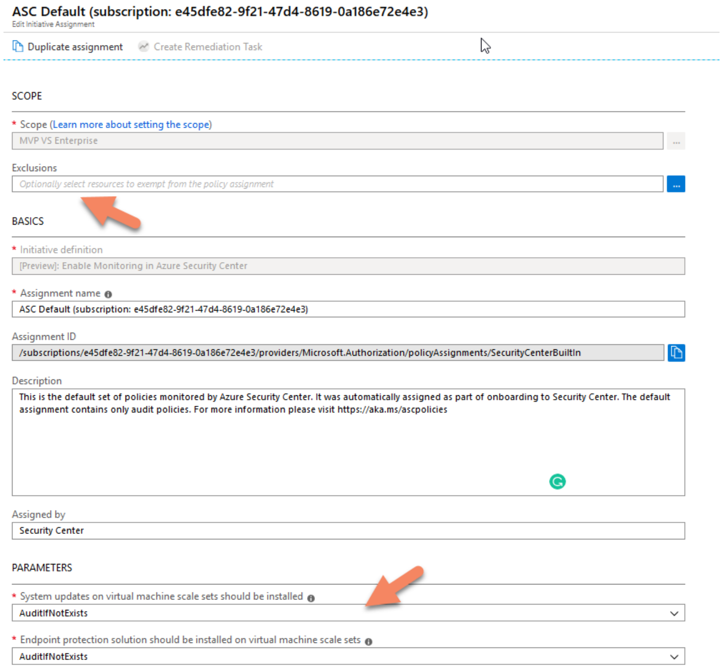


Scope: MVP VS Enterprise
Definition type: All definition types
Search: Filter by name or id...

Total Assignments 1
Initiative Assignments 1
Policy Assignments 0

NAME	SCOPE	TYPE	POLICIES
ASC Default (subscription: e45dfe82-9f21-47d4-861...	MVP VS Enterprise	Initiative	88

Select the initiative and check if something has been excluded, you can also check in the list if some policies have been disabled, if you find exclusions or a policy disabled then check if that may affect the pentesting result.



ASC Default (subscription: e45dfe82-9f21-47d4-8619-0a186e72e4e3)
Edit Initiative Assignment

SCOPE

* Scope (Learn more about setting the scope)
MVP VS Enterprise

Exclusions
Optionally select resources to exempt from the policy assignment

BASICS

* Initiative definition
[Preview]: Enable Monitoring in Azure Security Center

* Assignment name
ASC Default (subscription: e45dfe82-9f21-47d4-8619-0a186e72e4e3)

Assignment ID
/subscriptions/e45dfe82-9f21-47d4-8619-0a186e72e4e3/providers/Microsoft.Authorization/policyAssignments/SecurityCenterBuiltIn

Description
This is the default set of policies monitored by Azure Security Center. It was automatically assigned as part of onboarding to Security Center. The default assignment contains only audit policies. For more information please visit https://aka.ms/ascpolicies

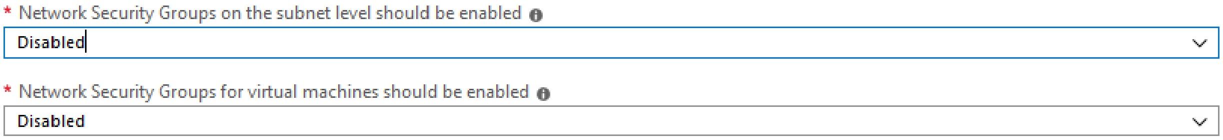
Assigned by
Security Center

PARAMETERS

* System updates on virtual machine scale sets should be installed
AuditIfNotExists

* Endpoint protection solution should be installed on virtual machine scale sets
AuditIfNotExists

For example, the two policies below are regarding the usage of firewall for subnets and virtual machines, and they are disabled, they should be enabled, maybe the customer is using a different firewall appliance or maybe they temporary disabled them, in any case, no sense to attack VMs without any firewall protection, in that case, better have a chat with the customer.

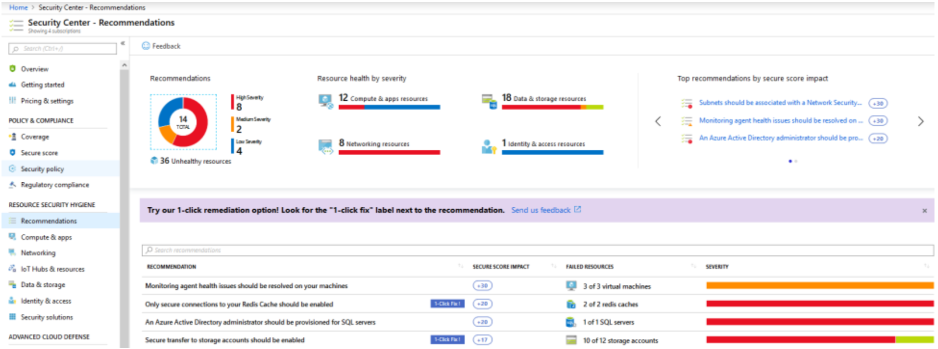


* Network Security Groups on the subnet level should be enabled
Disabled

* Network Security Groups for virtual machines should be enabled
Disabled

Security Center

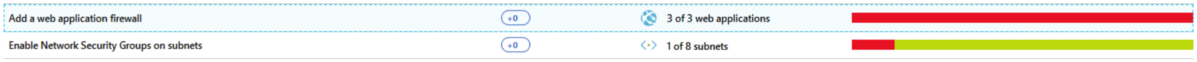
Enter in the Azure Portal, open Security Center and select Recommendations, ASC may provide you very important information like virtual machines not protected or worse, some Security Center features can be disabled, in that case, better have a chat with the customer and avoid a lot of false positives during the pentesting.



Try our 1-click remediation option! Look for the "1-click fix" label next to the recommendation.

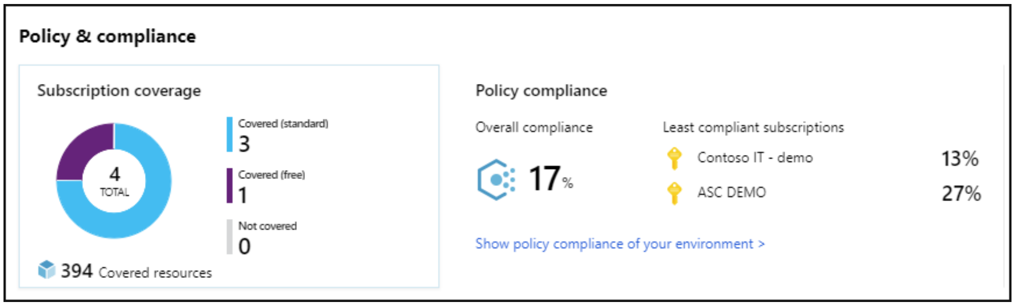
RECOMMENDATION		SECURE SCORE IMPACT	FAILED RESOURCES	SEVERITY
Monitoring agent health issues should be resolved on your machines		+30	3 of 3 virtual machines	
Only secure connections to your Redis Cache should be enabled	1-Click Fix!	+20	2 of 2 redis caches	
An Azure Active Directory administrator should be provisioned for SQL servers		+20	1 of 1 SQL servers	
Secure transfer to storage accounts should be enabled	1-Click Fix!	+17	10 of 12 storage accounts	

For example, the two recommendations below are pretty important, better having a chat with the customer.



Add a web application firewall	+0	3 of 3 web applications	
Enable Network Security Groups on subnets	+0	1 of 8 subnets	

Another important thing to check is the Policy and compliance. Using this feature, we can quickly check the overall compliance across all the subscriptions.



Policy & compliance

Subscription coverage

Covered (standard)
3

Covered (free)
1

Not covered
0

394 Covered resources

Policy compliance

Overall compliance

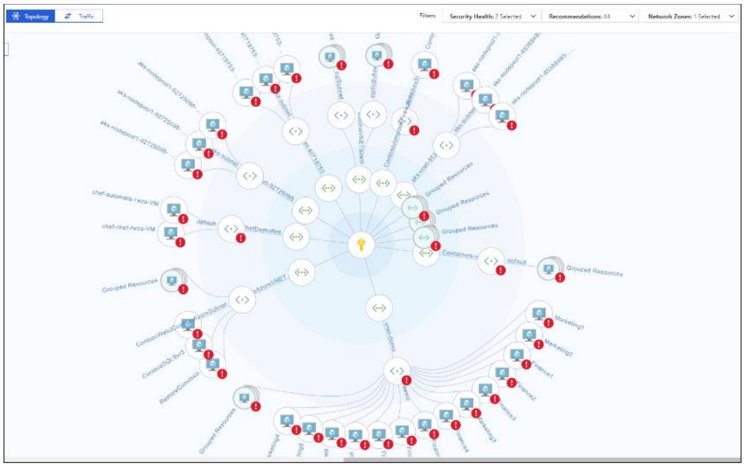
17%

Least compliant subscriptions

Contoso IT - demo	13%
ASC DEMO	27%

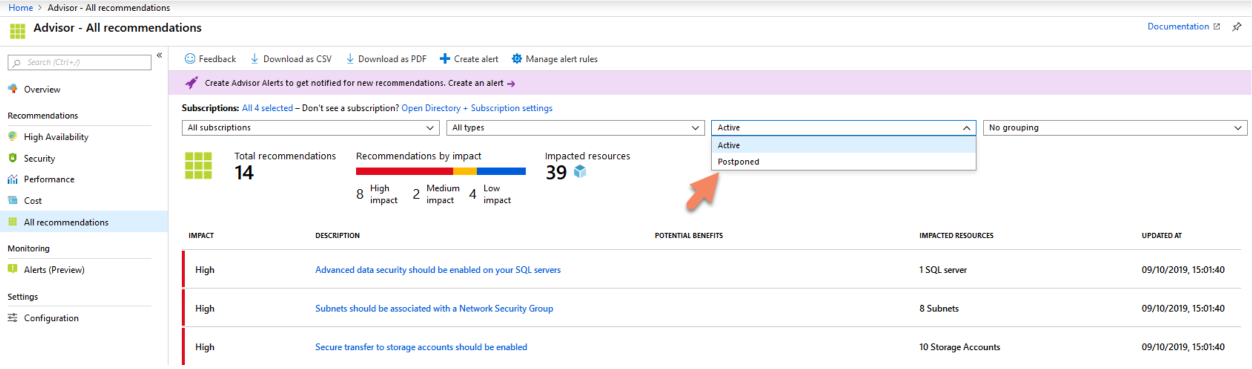
Show policy compliance of your environment >

One of my preferred tools in Security Center is the Network Map, using that we are able to get a great view of the network topology, how many nodes are connected, possible immediate security issues to look at, and more.



Azure Advisor

This is another useful area, enter the portal and search for Advisor, the Azure Advisor is the central point for advisory, and it provides much interesting information like recommendations, alerting and more, and we can also check if any recommendation has been postponed.



Home > Advisor - All recommendations

Advisor - All recommendations

Create Advisor Alerts to get notified for new recommendations. Create an alert →

Subscriptions: All 4 selected – Don't see a subscription? Open Directory + Subscription settings

All subscriptions | All types | Active | No grouping

Active
Postponed

Total recommendations 14

Recommendations by impact: 8 High impact, 2 Medium impact, 4 Low impact

Impacted resources 39

IMPACT	DESCRIPTION	POTENTIAL BENEFITS	IMPACTED RESOURCES	UPDATED AT
High	Advanced data security should be enabled on your SQL servers		1 SQL server	09/10/2019, 15:01:40
High	Subnets should be associated with a Network Security Group		8 Subnets	09/10/2019, 15:01:40
High	Secure transfer to storage accounts should be enabled		10 Storage Accounts	09/10/2019, 15:01:40

These are the most important areas we can check and what we are doing, is not just a quick and useful assessment, but it is also a first pentesting, and the customer will be happy to know that we covered the most important Azure security best practices before starting our penetration test.

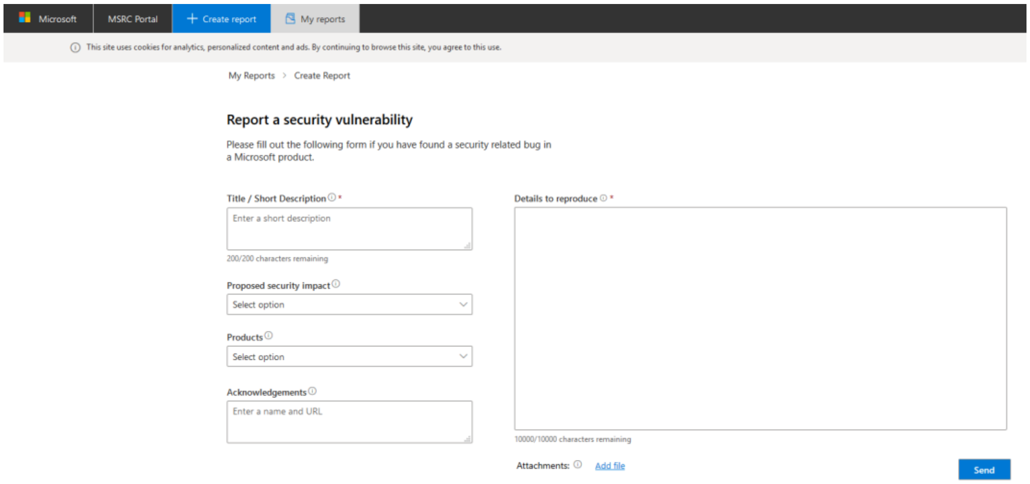
The Attack

This is maybe the funniest part for most of the people, but it is also the most delicate one because during the attack, we may face some particular situations, and we need to handle them properly.

We need to keep in mind that we are not just attaching the solution of the customer, but we are also attacking the host infrastructure, which means Azure, in the next whitepapers we will examine many types of attacks on different resources, and sometimes the attack may also discover a critical issue on Azure.

During a pentest, I discovered a Zero-Day in Azure AD which was also affecting the customer solution, if this happens, we need to immediately report the Zero Day to Microsoft, this is what an ethical hacker does.

To submit a Zero Day, you need to go to the MSRC Researcher Portal, register your account, and insert the security vulnerability.



This site uses cookies for analytics, personalized content and ads. By continuing to browse this site, you agree to this use.

Report a security vulnerability

Please fill out the following form if you have found a security related bug in a Microsoft product.

Title / Short Description *: Enter a short description

200/200 characters remaining

Proposed security impact: Select option

Products: Select option

Acknowledgements: Enter a name and URL

Details to reproduce *:

10000/10000 characters remaining

Attachments: Add file

You can join the Microsoft Bug Bounty Program which is a good opportunity to meet many other security experts and also gain some money, below the portal web site:

- https://www.microsoft.com/en-us/msrc/bounty?rtc=1

The Report

Everybody hate writing documentation, and this is maybe the most hated phase. However, it is very important because we share the finding to the customer.

The report can be a collection of different documents produced by tools, and by us, in the report, we need to share at least the vulnerabilities we found, how to replicate them, and how to protect from any vulnerability.

Hopefully, I will be able to write a whitepaper on this topic, and I will speak in detail about the report but there is one important consideration to do here, we are working in the Microsoft Azure and this is not the same as the on-premise environment. In Microsoft Azure, there are many assets able to protect the infrastructure, network security groups, appliances, internal assets like Security Center and more, and for that reason, it is necessary to build a very good understanding about Azure Security and infrastructure, we can't limit our report on the first two areas only.

About the Author

Write by Nino Crudele [Microsoft Azure MVP and Ethical Hacker]

Nino Crudele is a freelance living in the United Kingdom. He is Global Azure Lead and Cybersecurity expert in Hexagon Manufacturing Intelligence, a global manufacturing company. He is responsible for leading the Microsoft Azure Cloud area, supporting and advising the Company to select the most appropriate cloud strategies and solutions from high-level design to implementation.

He is Microsoft Azure MVP since 2006 and Certified Ethical Hacker, Nino is also an international speaker, author, and a very active community member.

You can contact Nino at mailto:mnino.crudele@live.com (Twitter @ninocrudele)

LinkedIn: https://www.linkedin.com/in/ninocrudele

About the Reviewers

Sandro Pereira [Azure MVP & MCTS BizTalk Server 2010]

Sandro Pereira lives in Portugal and is currently working as an Integrator consultant at DevScope (www.devscope.net). In the past years, he has been working on implementing Integration scenarios both on-premises and cloud for various clients, each with different scenarios from a technical point of view, size, and criticality, using Microsoft Azure (API Management, Logic Apps, Service Bus, Event Hubs, PowerApps, Power Automate, …), Microsoft BizTalk Server and different technologies like AS2, EDI, RosettaNet, SAP, TIBCO and so on

Sandro is very active in the BizTalk community as blogger (https://blog.sandropereira.com), member and moderator on the MSDN BizTalk Server Forums, TechNet Wiki author, Code Gallery and GitHub contributor, member of several online communities, guest author at BizTalk360 and Serveless360, public speaker and technical reviewer of several BizTalk and Azure books and whitepapers, all focused on Integration. He is also the author of the book **BizTalk Mapping Patterns & Best Practices**.

He has been awarded the Microsoft Most Valuable Professional (MVP) since January 2011, for his contributions to the world-wide BizTalk Server community (https://mvp.microsoft.com/en-us/PublicProfile/4030655). He currently holds MCTS: BizTalk Server 2006 and BizTalk Server 2010 certifications.

You can contact Sandro at sandro-pereira@live.com.pt (Twitter: @sandro_asp)