# Configuring Data Security Policies in Microsoft Azure:

**Course Overview:**

* Many data regulations instruct companies to keep classified customer data in the same country it was collected. We can use Microsoft Azure Data security offerings to enforce this in cloud platform.
* Imp Topics covered in the course:
  + Enforcing data sovereignty in Microsoft Azure
  + Implementing data classification
  + Configuring data retention policies
  + Azure Policy service.

# Configuring Data Classification in Microsoft Azure

**Understanding Security Requirements:**

* 1st step is setting up security policies is to understand the security requirements and what we are trying to protect. To achieve better ROI on security investment, the Org needs to first understand its security requirements and priorities. Understanding Security requirements is imp before going ahead and implementing security for the org.
* There are 3 main questions that needs to be answered. These questions are around 3 categories:  
  + **Governance**: How to monitor, audit, and report the organization's security. We need to have answers to questions like below:

Is there any mandatory reporting that you need to provide to the upper management concerning the security? Are there requirements for you to audit the compliance procedures you have implemented for your company?

* + **Risks**: What kind of risks the org faces while trying to protect info.  
    We need to recognize what type of risks you face while trying to protect identifiable information. We need to know what kind of data are we saving Ex: IP,PII, Client Financial information.

Do we have any disaster recovery and business continuity plan in case of security breach?

* + **Compliance**: Any specific industry, government or regulatory requirements that dictate or provide recommendations on orgs security controls. Ex: if you are accepting credit card payments there are specific regulatory requirements that need to be followed.

**Understanding Data classification:**

After we have identified security priorities, we are ready to apply security Rules. These rules are applied to group of resources or data. This means to apply security rules we need to classify your data and resources first.

Following is the in general classification of data in any org:

1. **Public**: Data can be shared with the public and introduces no threat to the company.   
   Ex: company contact information or the portfolio list.
2. **Internal Data**: Data shouldn't be shared with the public, and is only meant to be used by the company employees. Ex: list of policies and procedures in the company. Sharing this information with the public is harmful, but it might not impose any critical threat to the company.
3. **Confidential Data:** This info should only be shared with a select group of company employees or customers. Ex: list of development environments. If this information is leaked to non-authorized company employees or the public, it can cause serious threat to the company and its customers
4. **Top Secret Data:** This Info should only be shared with a small group of company employees and customers. If this information is stolen, it can cause serious threat to the company and its customers. Ex: client's credit card numbers, name, email address, or social security numbers.

What is Data Classification?

Data classification is the process of associating a metadata to a digital asset which identifies the type of data associated with that asset. For example, you can label a group of data as public, and the next group as confidential.

Data classification is a common starting point for governance. It categorizes data by sensitivity and business impact.

As an example of data classification, we can have a look at Microsofts data classification.  
Microsoft's data is classified into five categories:

* Non-Business: data from the customer's life that doesn't belong to Microsoft.
* Public: business data that is freely available and approved for public consumption.
* General Data: business data that is not meant for public audience, and it should only be accessible to Microsoft employees.
* Confidential Data: business data that could cause harm to Microsoft if overshared.
* Highly Confidential Data: business data that would cause extensive harm to Microsoft if overshared.

There is no exact answer to how to classify data in our org. We know the data and industry, and using this knowledge we can come up with data classification.

**Data Classification in Microsoft Azure:**

* Microsoft suggests that any asset in the cloud should have documented metadata. We can apply any no of metadata, but following 3 are quite imp:
* Data Classification Ex: Public, internal etc
* Business Criticality: Ex: Non-critical, critical
* Billing responsibility: specifies which dept or branch is resp to pay for this specific resource.
* 3 technologies in Azure that can be used to classify data:

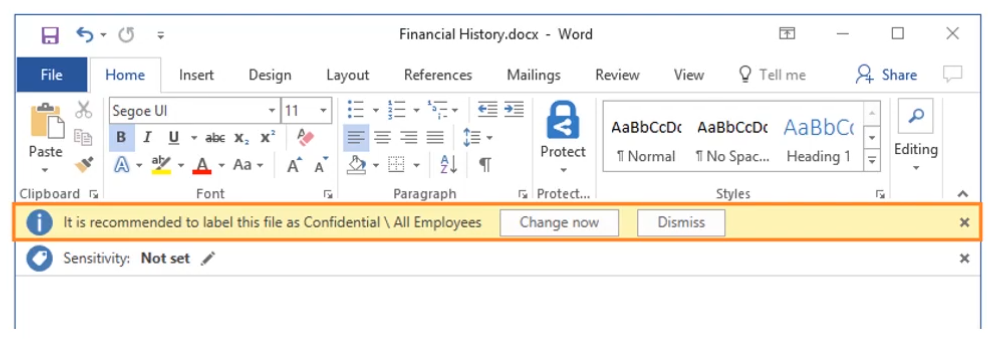
1. Azure RM Tags
2. Advanced Data Security for MS Azure DB.
3. Azure Information Protection Labels for MS office documents and emails.

* ARM tags:
  + Suggested approach for metadata storage. These tags can be used to apply data classification information to deployed resources.
  + Valuable tool for managing resources and applying policies.
  + Azure tags can be managed in the portal or programmatically.
  + We can apply resource manager tags to your Azure resources to logically organize them into taxonomies.
  + Each tag consists of a name and a value, for example, a tag name could be department, and its value could be IT.
  + After we apply tags, we can retrieve all the resources in subscription with that tag name and value. Tags enable us to retrieve related resources from different resource groups.
  + Tags can be applied to resources manually or automatically.   
    We can use an Azure policy to enforce tagging rules and conventions.   
    We can also create an Azure policy that automatically applies tags during resource deployments.
  + We can assign up to 50 tags to a specific resource.
  + Tags cannot be applied to classic resources such as Cloud Services, they can only be applied to resources created with ARM.
  + A tag name can be up to 512 characters, and a tag value can be up to 256 characters. The tag name can be up to 128 characters for storage accounts.
  + Resource group tags are not inherited by the children within the resource group.
  + Tag names cannot contain a few characters, for example, angle brackets, percentage, ampersand, question mark, and forward and back slashes.
  + Generalized VMs don't support tags, however, the VMs we create from these generalized VMs do support tags, and we can assign up to 50 tags to them.

**Azure Information protection:**

* + Technology which can be used to classify data and it targets Microsoft office documents.
  + Cloud-based solution that helps an organization to classify and protect its documents and emails by applying **labels**.
  + Labels can be applied automatically by administrators who define rules and conditions.
  + Labels can also be manually applied by users, or a combination of manual and automatic where users are given recommendations when they are creating a new document.

In below SC, the administrator has set up a rule which is going to scan this document and determine if this document has confidential information. If it does, it is going to give a recommendation to the user creating the document so this user can go ahead and add a new label named confidential to this document. The user can apply the label or dismiss the recommendation.



* + Labels can also include visual markings, for example, headers, footers, or watermarks, and these visual markings can show up in documents and emails on the client's machine.
  + Steps an Admin should take to provision and use Azure Information Protection:
    - First, this service should be provisioned in the Azure portal  
      In the Azure portal, we can search for Azure Information Protection, click on the service, and click on the Create button. After that, if we have the correct licensing we can go ahead and provision the service.

We must have either of the following licenses to be able to provision a new instance of Azure Information Protection.

* + - * Azure Information Protection Premium P1, which is included within Enterprise Mobility and Security E3, or
      * Azure Information Protection Premium P2, which is included within Enterprise Mobility and Security E5.
      * Office 365 subscription that includes Azure Rights Management.
    - Administrator should install the Azure Information Protection client on user's machines.  
      After we provision the server-side component, we need to go ahead and download the client for Azure Information Protection and install it on our client's machine

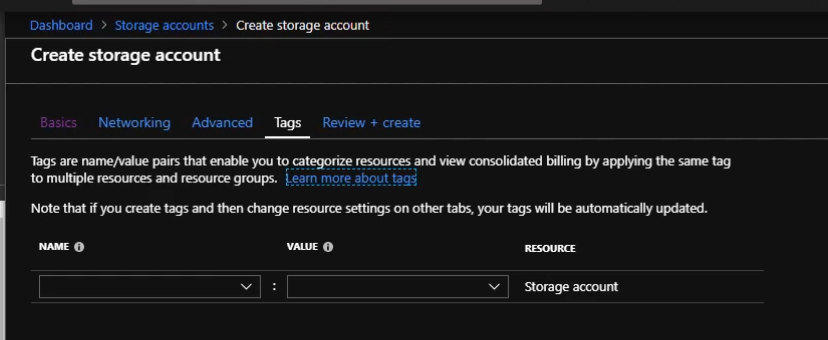
**Data classification for Azure SQL Databases:**

* + While creating DBs we can assign RM Tags. However with tags we are tagging the database, not the data within the database.
  + Data discovery and classification provides advanced capabilities built into Azure SQL Databases. It provides discovering, classifying, labeling, and protecting the sensitive data in our Azure SQL Database and data warehouse. This service can search our data within the database and automatically detect business, financial, healthcare, personally identifiable data, or PII, and other types of confidential data.
  + Data discovery and classification is part of the Advanced Data Security (ADS) offering.
  + On our Database, we have the tab for “Advanced Data security”. We can enabler ADS for our server. This gives us few extra security services like Data Discovery and Classification, Vulnerability Assessment, and Advanced Threat Protection.

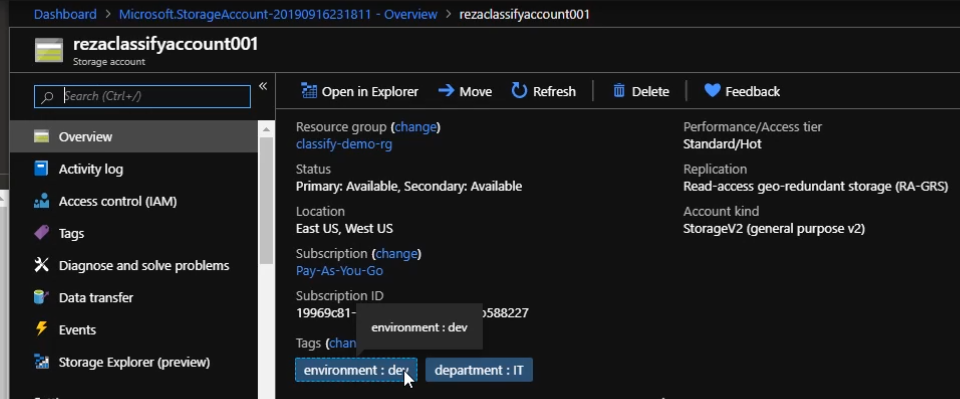
**Demos:**

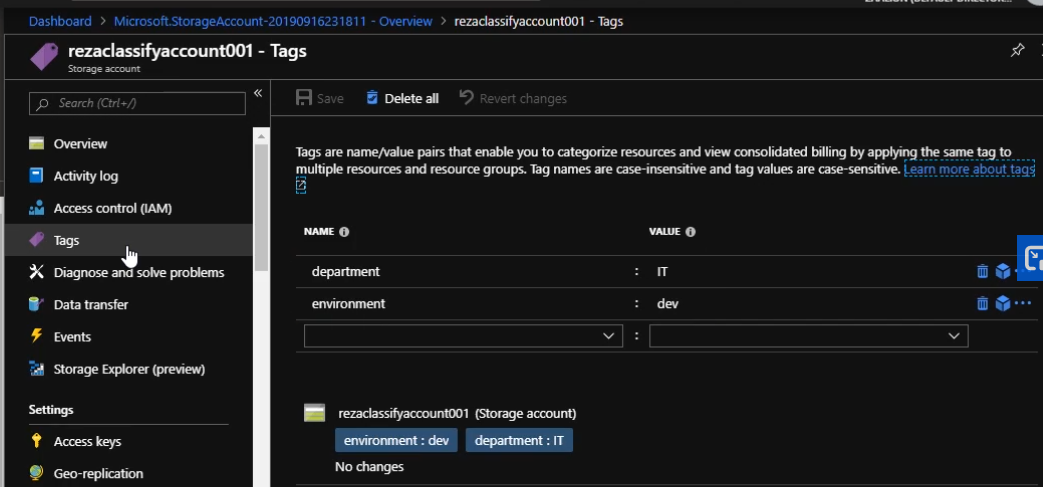
* **DEMO: Using Azure Policy to enforce tags.**

1. Create a storage account.
2. In the process of creating resource we have the option of assigning tags:

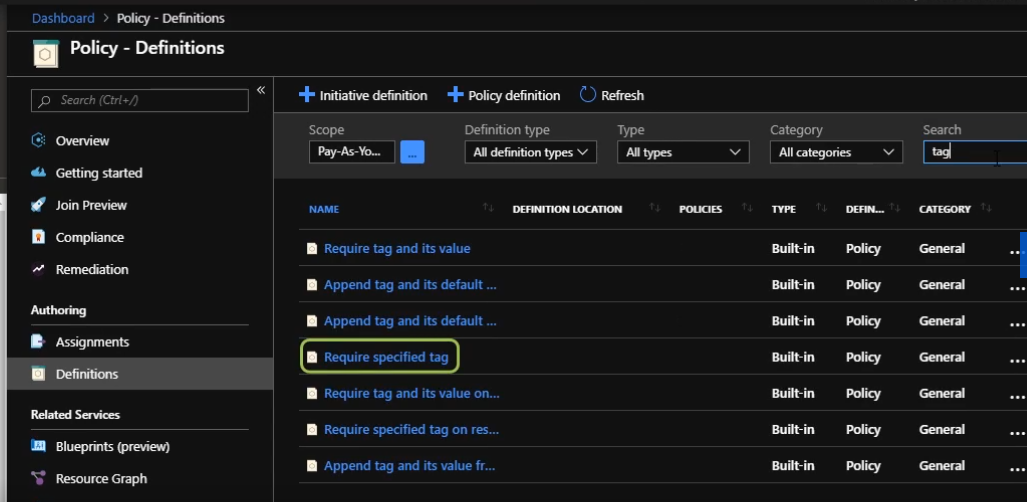


Below SC shows the assigned tags:

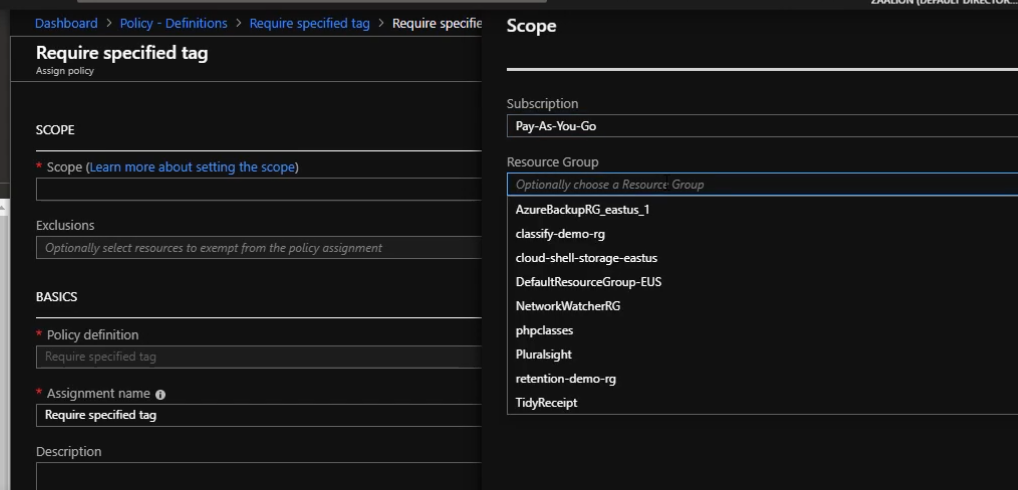




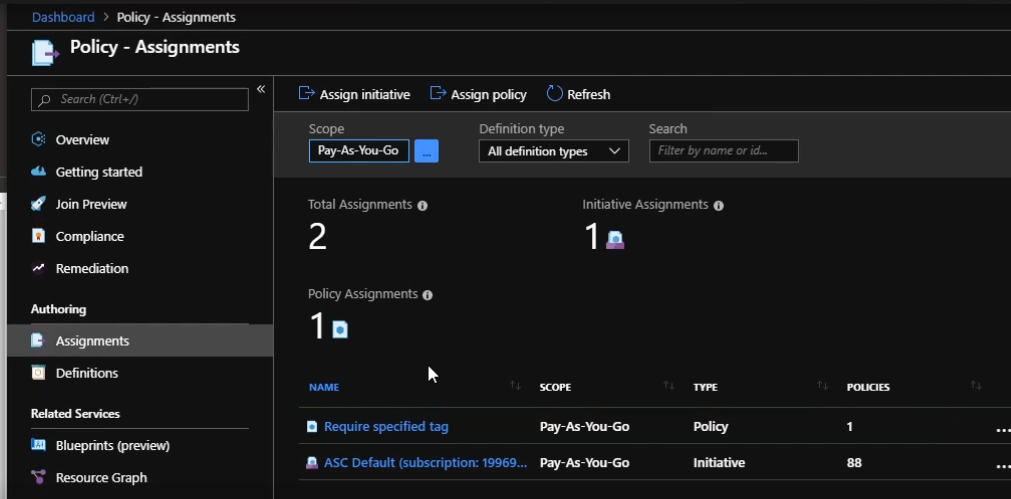
1. We can use Azure policies to enforce that specific tags should be added during resource creation.
2. Following shows the predefined policy (Require Specified tag) that can be used to enforce tags. Note that we can also create custom policies. First search for Azure policy service, go to Definitions and search for ‘tag’.



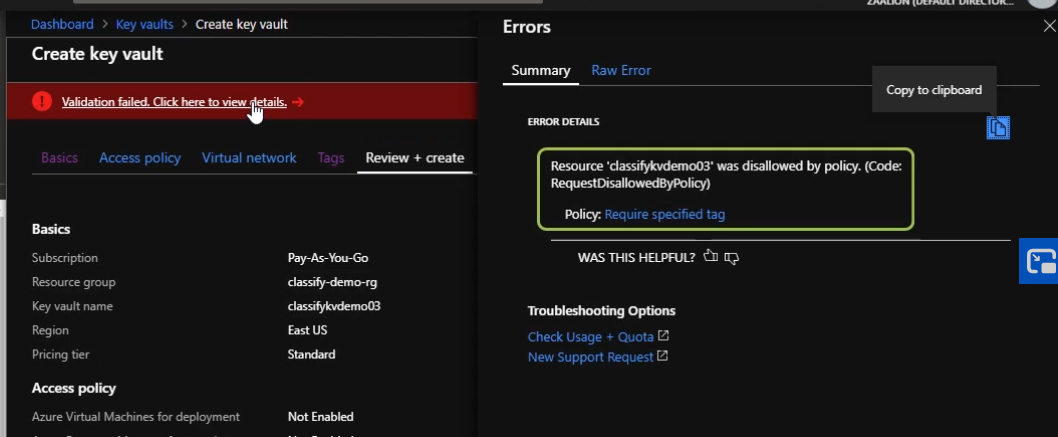
1. To be able to enforce the policy, we need to assign it first. On the below page we select the scope – Management group, subscription or resource group. If we choose a resource group from this drop-down, the policy will be enforced at the resource group level. However, if we leave the resource group empty, the policy will be enforced at the subscription level. We provide assignment name, description and finally Assign.



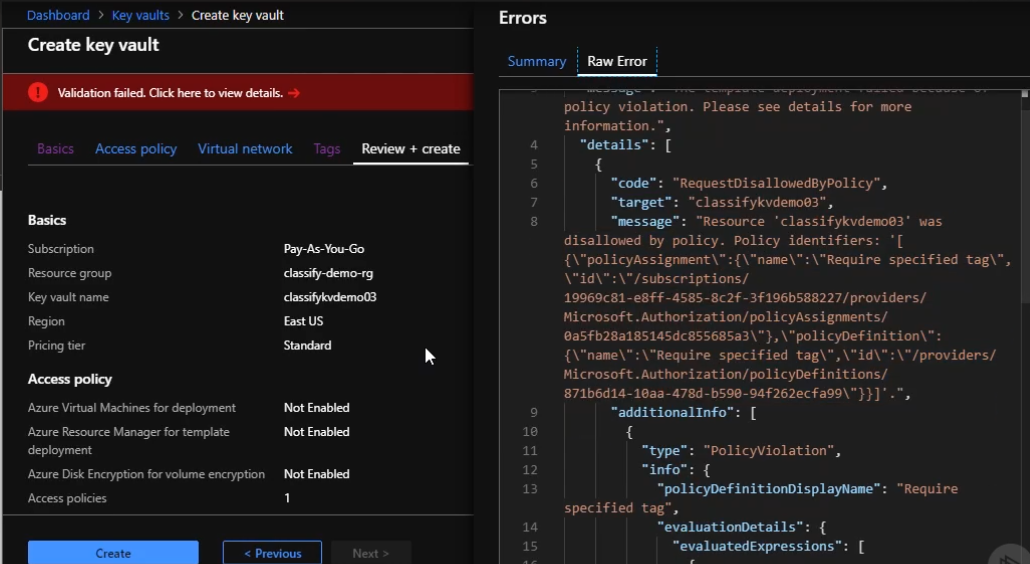
1. Once assigned, we can see the policy under assignments.



1. Now if we try to create a resource without specified tags, we would get an error:

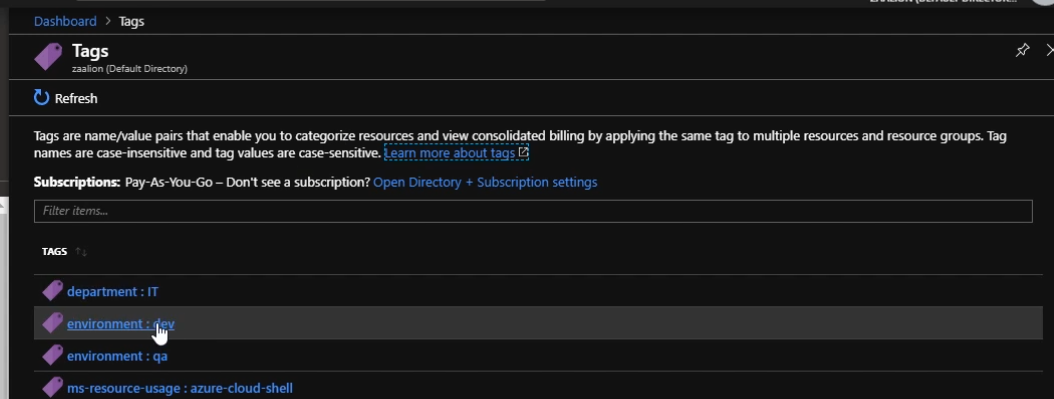


Clicking on raw error we get more details:

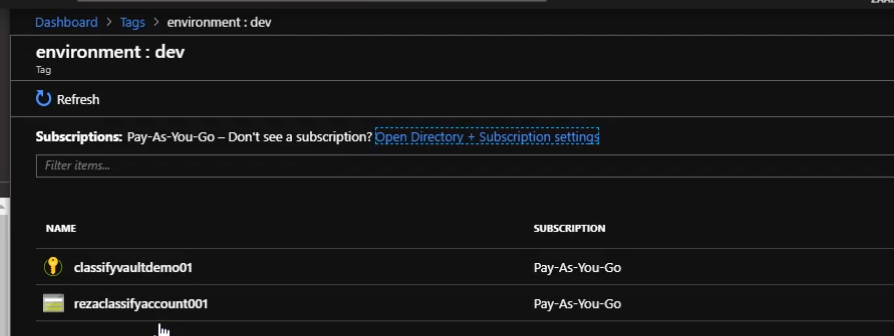


* **DEMO: Finding resources by tags:**

1. Search for Tags in all resources. Here we would see all tags within our subscription.



Click on any of the tags to see all resources with that tag:



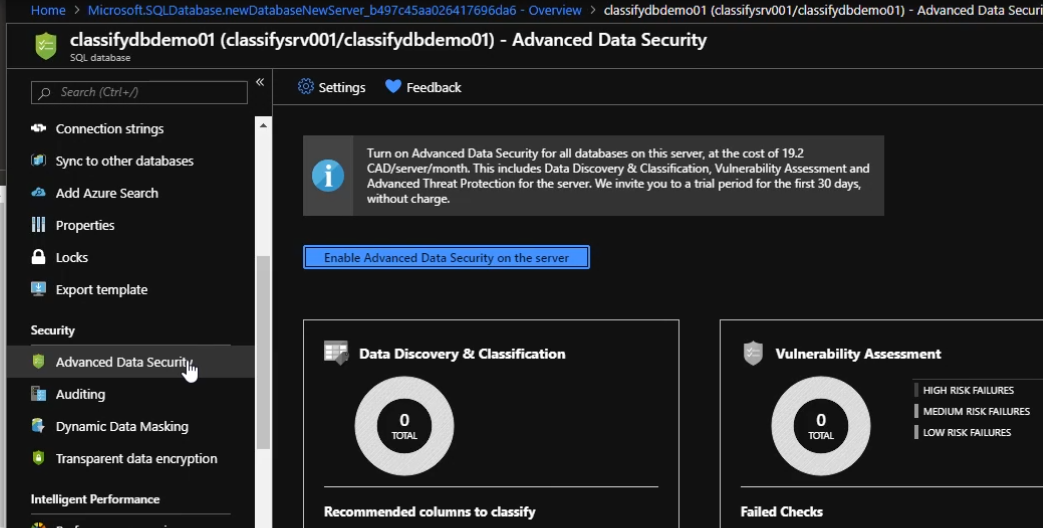
We could do the same programmatically using Azure CLI or Powershell.

* **DEMO: Azure SQL Database Data Discovery and Classification**

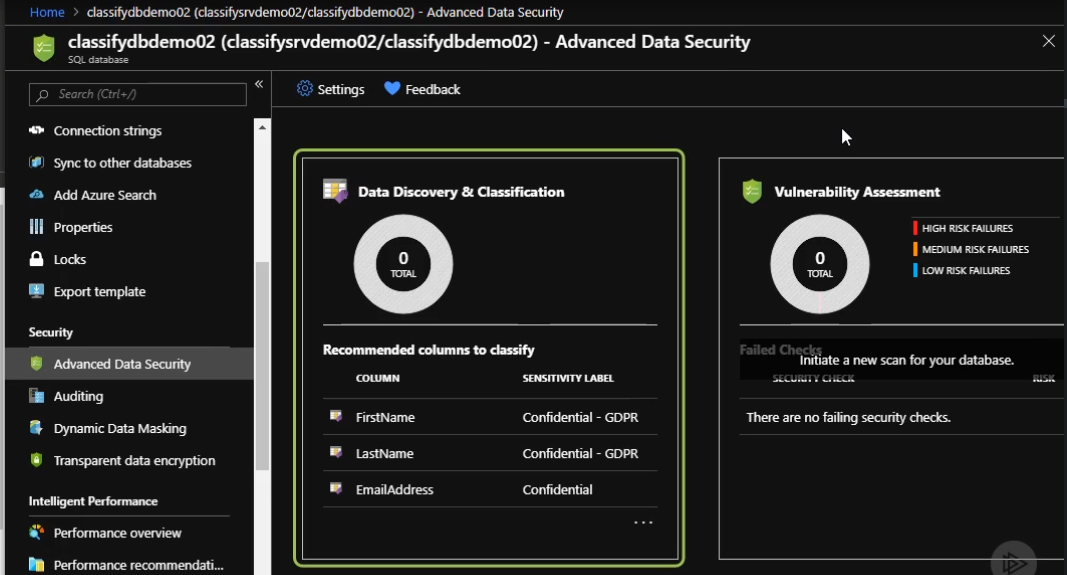
To be able to protect data and also comply with regulations, for example, GDPR, we need to know what kind of data we are storing in my database. we can go ahead and manually find those, or we can use Advanced Data Security features, which is provided by Microsoft.

This service automatically classifies our data and tell us what kind of information we are saving in your database. It looks for FirstName, LastName, email addresses, phone numbers, or PIIs, personally identifiable information. This service can also classify financial information such as credit card numbers and Social Security numbers.

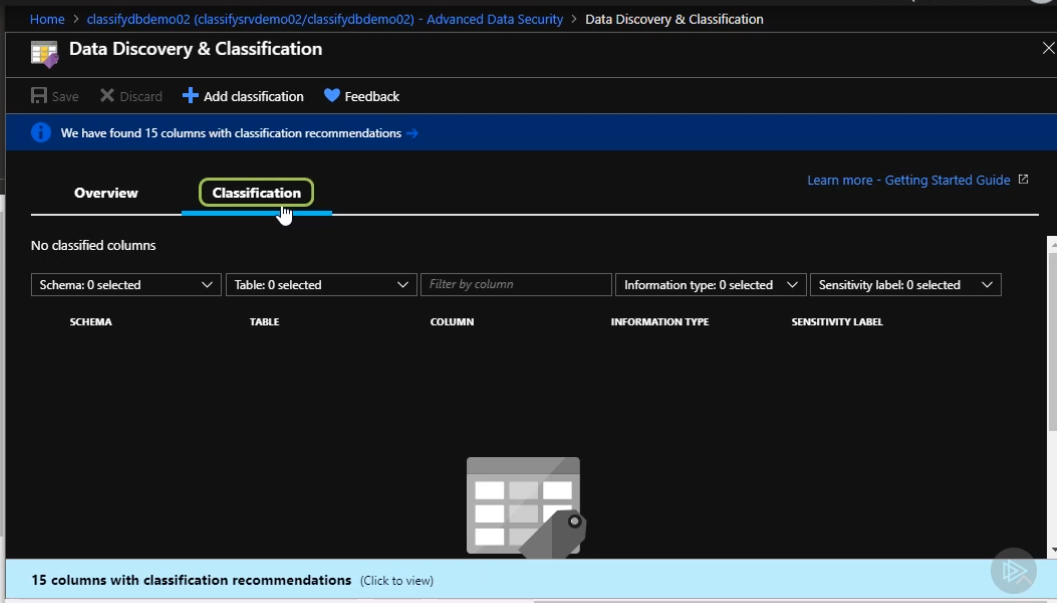
For a selected DB, when we go to the “Advanced Data Security “we see below option to enable ADS



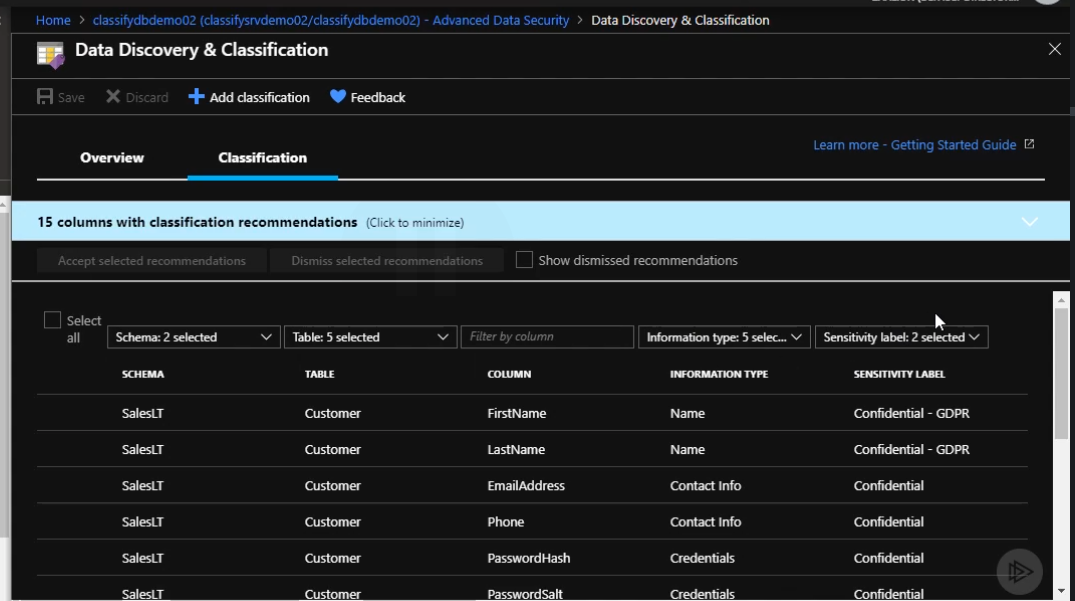
Note that enabling ADS costs money. One of the advanced security feature is “Data Discovery and Classification”.



Clicking on this we can see below screen. Note that we already have some classification recommendations. We can also specify our custom classifications.

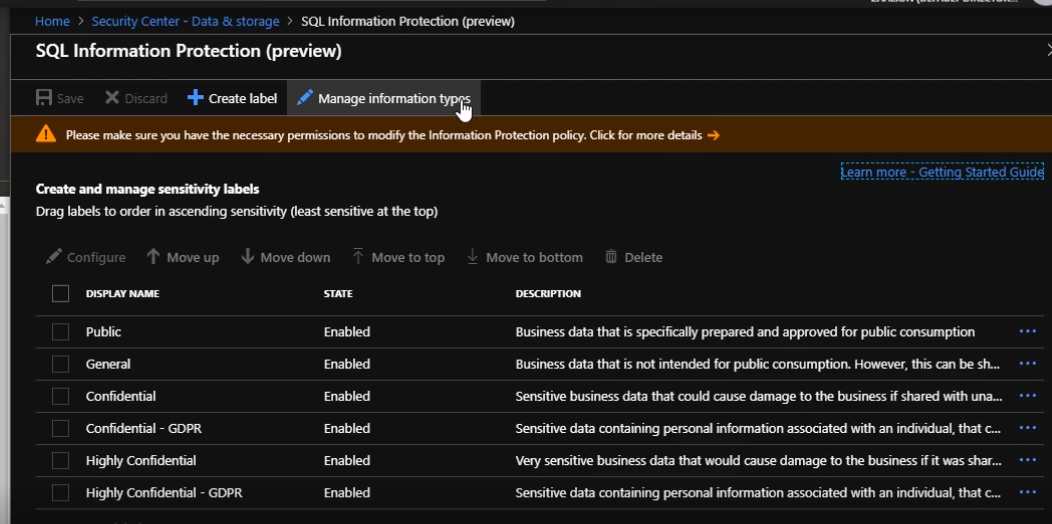


Clicking on the recommendations we get following. We can accept or dismiss the recommendations.

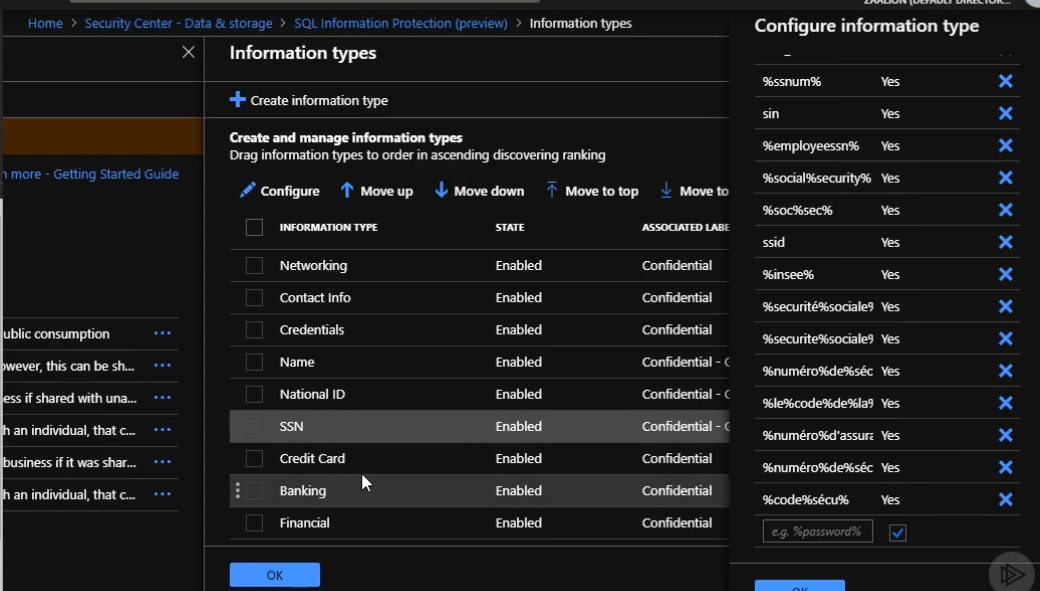


Once we create/modify the classifications, these will show up in SQL database audits. We can check who all have accessed the classified information.

We can create new labels from Security Center. Here we can also see existing labels.



Click on “Manage Information types” to add new labels



# Configuring Data Retention in Microsoft Azure:

**Overview:**

Key Topics:

* Data Retention Policies
* Significance of data retention in today’s world.
* Configuring data retention for key Azure resources like: Storage accounts, Azure SQL Database backups, Azure virtual machine backups, and Azure Monitor Logs (These are the same logs being generated by Application Insights and being used by Log Analytics)

**Introduction to Data Retention Policies:**

Why would we want to manage and govern our information?

1. Industry regulations and internal policies might require us to retain data for a minimum period. Ex: Company might have an internal policy which requires us to keep customer information for 7 years.
2. We are reducing the risk of a data breach by permanently deleting old content that we are no longer required to keep. For example, we might put a policy in place that makes sure the old customer data will get permanently deleted.
3. Deleting old data helps your organization to be more agile by ensuring that your users work only with content that's current and relevant to them.

Retention policy can help us achieve all these above goals.

A retention policy has 2 pillars:

1. 1st makes sure you are retaining the content safely so it can't be permanently deleted before the end of the retention period.
2. 2nd makes sure the content is permanently deleted at the end of the retention period.

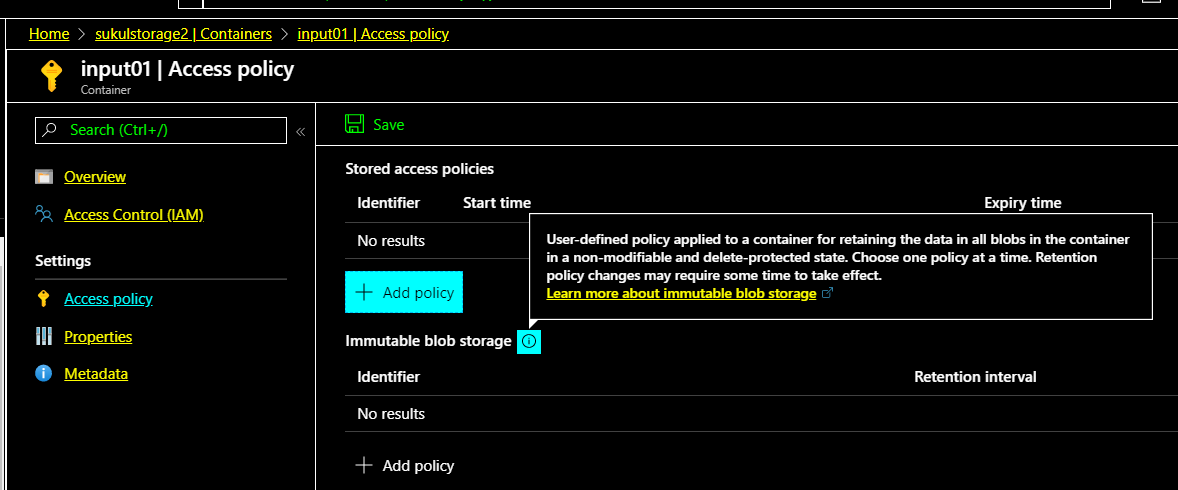
**Azure Blob Storage retention – Immutable Storage:**

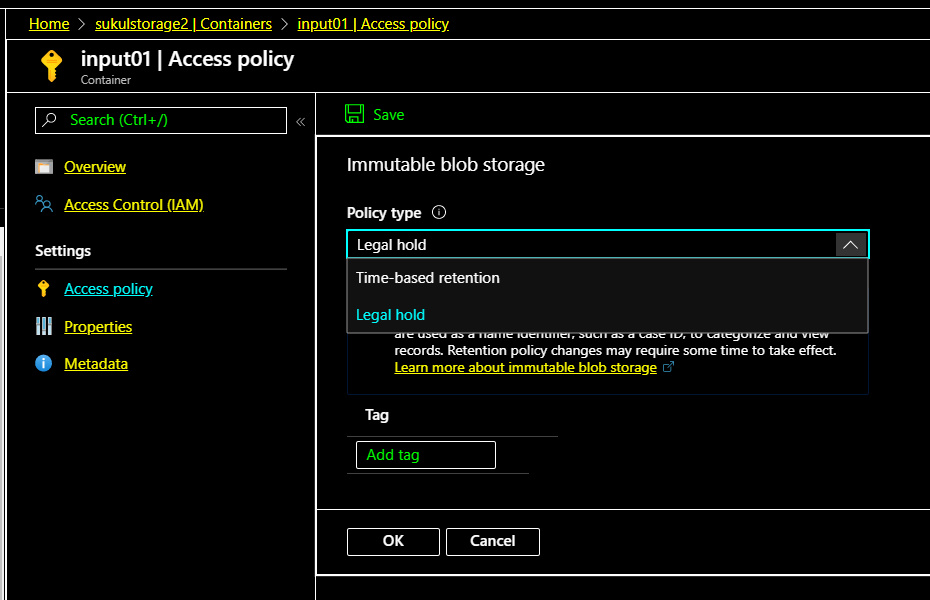
Azure offers 2 services for Azure Blob Storage retention:

1. immutable storage for Azure Blob storage,
2. Azure blob storage lifecycle

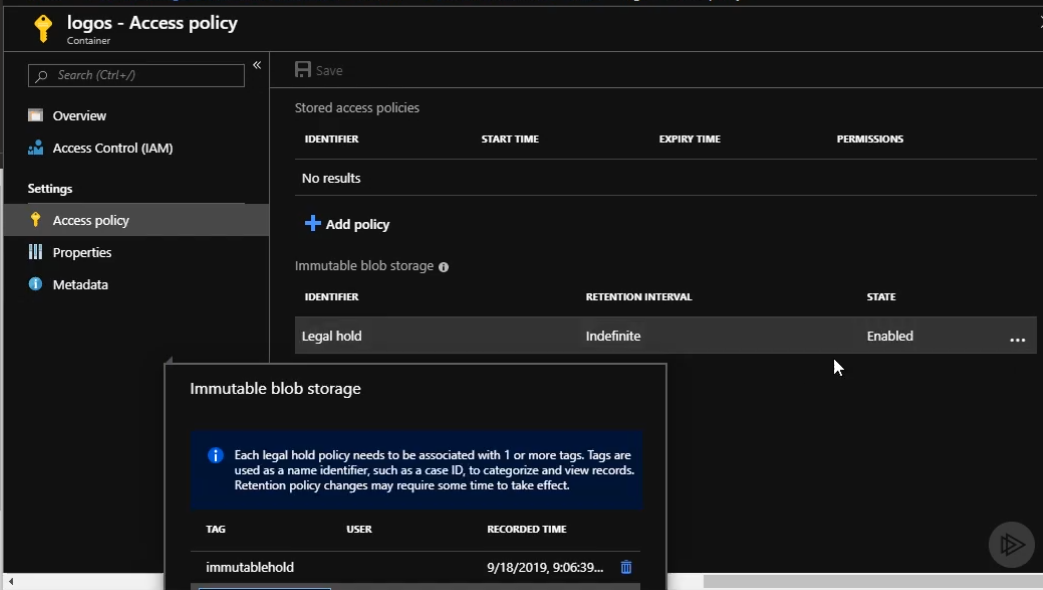
Immutable Storage for Azure blob Storage:

* Allows Org to store business-critical data in a Write Once Read Many (WORM) state.
* Blobs inside these containers can still be created and read, but not modified or deleted
* Immutable storage is enabled for General Purpose v2 and Blob Storage accounts in all Azure regions.
* This service can be used in any scenario to protect critical data against modification or deletion.
* Immutable storage for Azure Blob storage helps organizations comply with several regulations.(SEC 17a-4(f), CFTC 1.31(d), FINRA regulations)
* Ensures that data can’t be deleted or modified by any user including users with account admin privileges.
* Enables users to store critical information in tamper proof state for desired duration or until the hold is removed.
* There are two types of storage holds:
  + **Time based:** blobs can be created or read, but not modified or deleted until the retention period is expired
  + **Legal Hold:** if the retention interval is not known, users can set legal holds to store data immutably until the legal hold is cleared.
* Note that immutable storage is configured at the container level, and it will affect all the blobs inside the container.
* We can remove legal holds in portal or programmatically at any time. We can have multiple legal holds assigned to one container. We can also have a combination of time based retention and legal holds assigned to containers.
* After a time-based or legal hold is applied on a container, all child blobs move into an immutable WORM state in less than 30 seconds. All new blobs that are uploaded to that container will also move into the immutable state.
* All blobs in that container stay in the immutable state until all legal holds are cleared, even if their effective retention period has expired. And the opposite is true too. A blob stays in an immutable state until the effective retention period expires, even though all legal holds have been cleared.
* Trying to delete a blob inside the immutable container gives error.
* When creating Legal holds, we need to associate at least one tag to the legal hold. This can be used to work with the legal hold later, especially through PowerShell or the REST API.
* Once we add an immutable storage policy, we need to wait for 30 seconds to make sure the blobs inside this container are moved to the immutable status
* Certain Limitations include :
  + Up to 1, 000 time-based immutable policies assigned to a container.
  + Minimum retention interval is one day and the maximum is about 400 years.

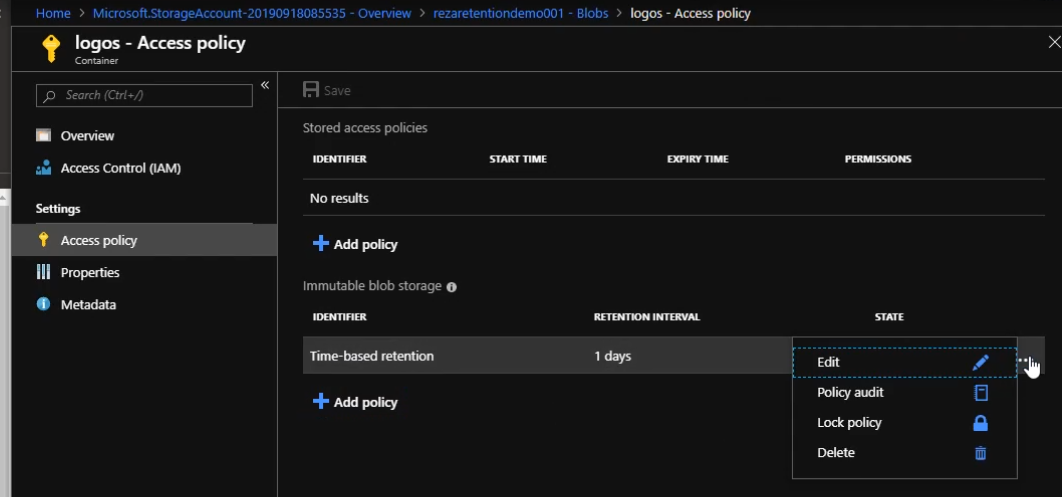




* We can delete the legal hold policy as well. We click on the three dots, Edit, and let's remove all the tags associated with this legal hold and click on the OK button. This shall delete the legal hold policy. Remember we can multiple legal holds if we want.



* Note that in case of time based retention, once we lock the policy, we won’t be able to delete any file inside the container until the retention period is over. Note that when we create a time based policy, the state will be unlocked. In this state we can edit (increase decrease the no of days), delete the policy. Here we can also lock the policy. Note that we don’t need to lock the policy for retention to come to play. Any unlocked policy can be deleted.

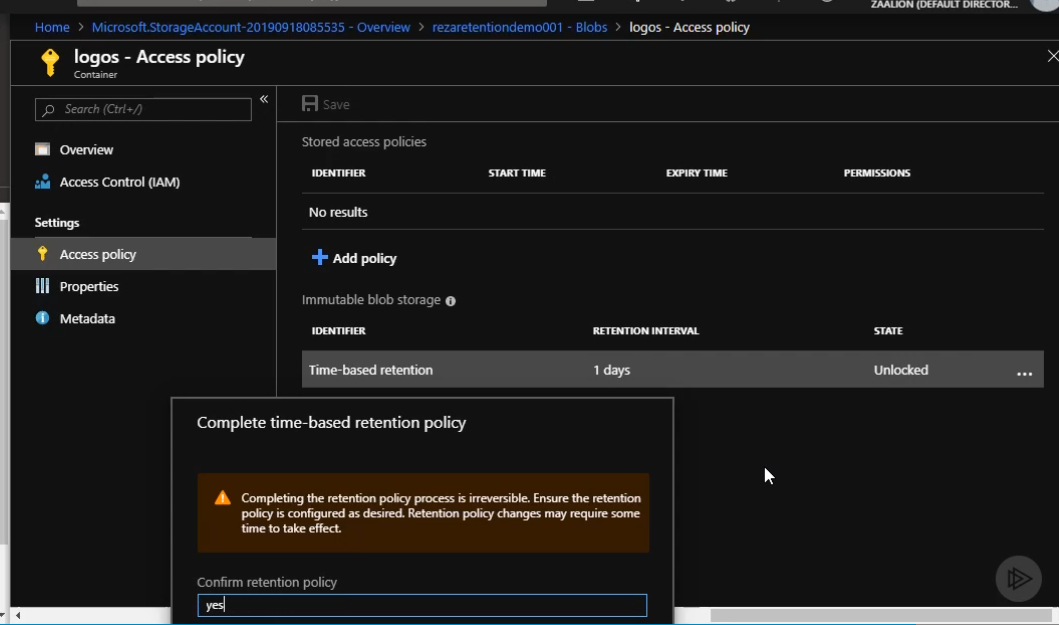


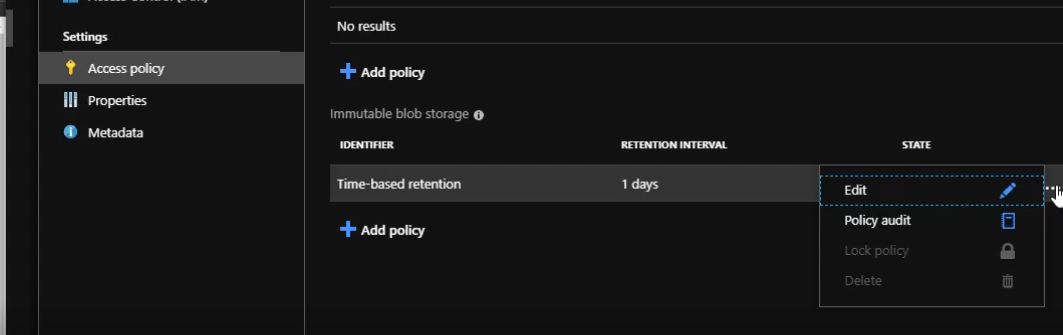
* Once the time based policy is locked, we cannot make changes to the policy till the time specified is over. We cannot delete the files or even the container till the time specified has passed.

Below shows that once it’s locked, we cannot delete the policy. We have to wait for expiry.

We can still upload new files to the container, read existing blobs, but not delete the container, blobs or the policy.

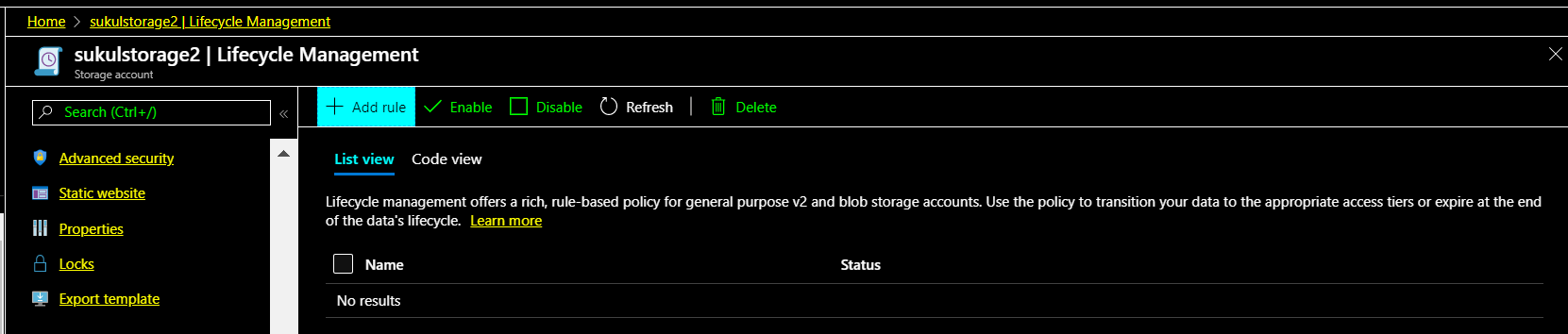
So remember the impacts of locking in a time based policy.

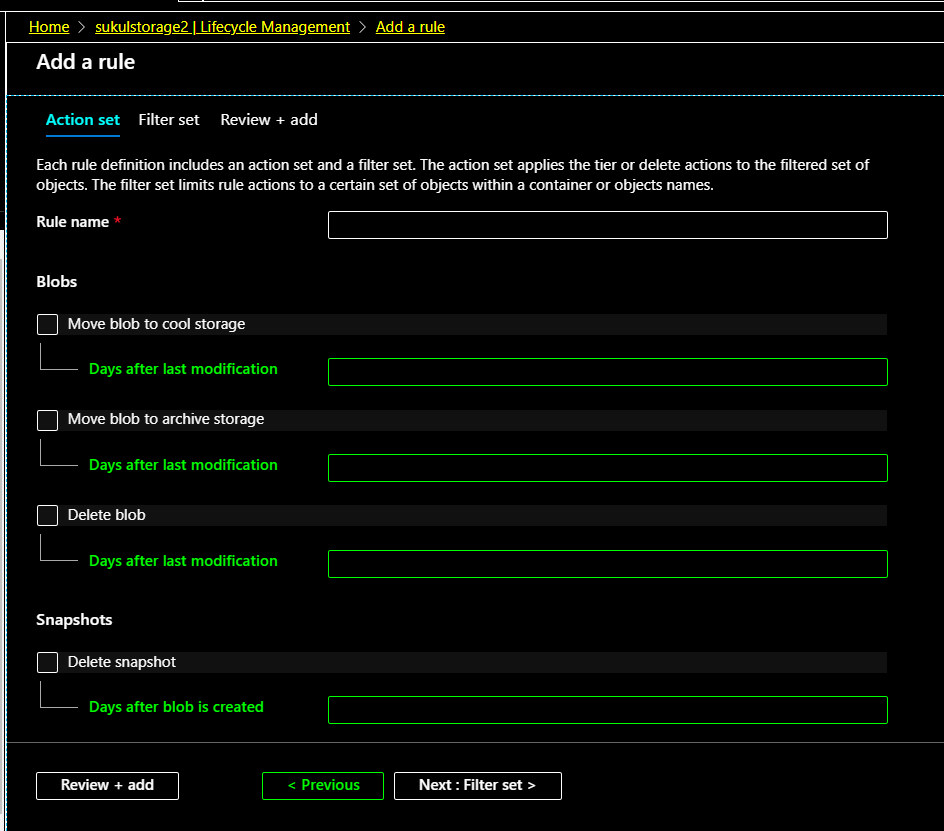


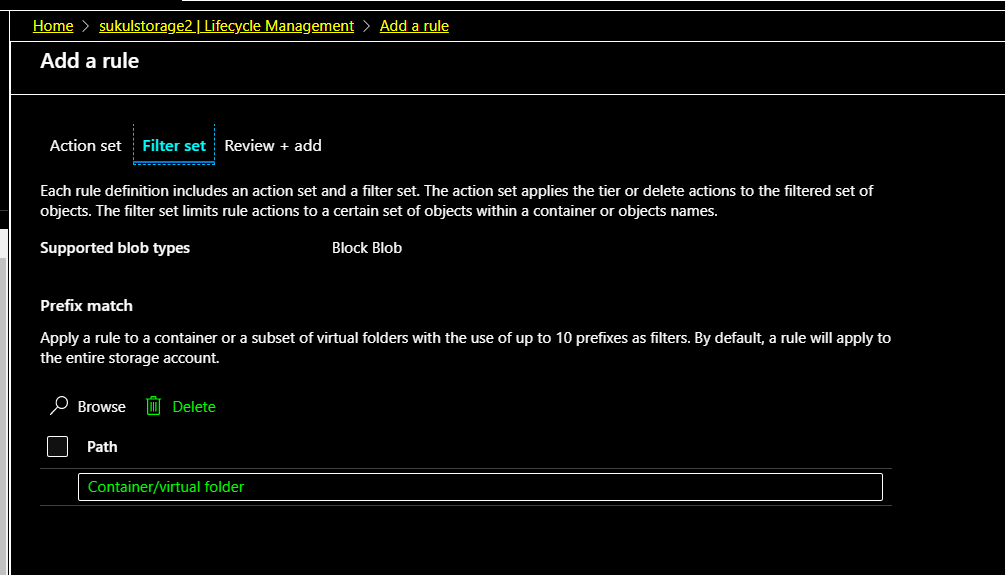


**Blob Lifecycle:**

* We know there are 2 pillars of Data retention - First one was about retaining the data before the retention period is expired, and the second pillar was about deleting it after the expiry time.
  + Immutable storage is addressing the first pillar. This makes sure data is untouchable and not deleted before the retention time is expired.
  + The data lifecycle is addressing the second pillar.
* Azure Blob storage lifecycle management offers a rich rule-based policy for General Purpose v2 and Blob storage accounts.
* We can use the policy to transition your data to the appropriate access tiers, or expire at the end of data's lifecycles.
* Azure lob access tiers:
  + Hot Tier: The pricing model of the hot access tier is optimized for frequent access.
  + Cool Tier: he pricing model for the cool access tier is optimized for storage. This means you pay more for frequent access comparing to the hot tier.
  + Archive tier
* Azure Blob Storage lifecycle management:
  + helps us transition blobs to a cooler storage tier to optimize for performance and cost.
  + allows deletion of blobs at the end of their lifecycle.
  + We can define rules to be run once per day at the storage account level
  + We can apply these rules to containers or a subset of blobs.(We can identify these blobs using prefixes as filters.)







* We define lifecycle rules at the storage account level. Note that we need to specify a name for the rule. We can define this using Azure portal. There are 3 main sections defined in the rule.
  + The first one moves the blobs to the cool storage after a specific number of days.
  + The second one moves the blob to the archive storage, and
  + The third one deletes the blob after a specific amount of days, and these days are calculated from the last modification time of the blob object.
  + We also have the option to delete the blob snapshots after a specific number of days.

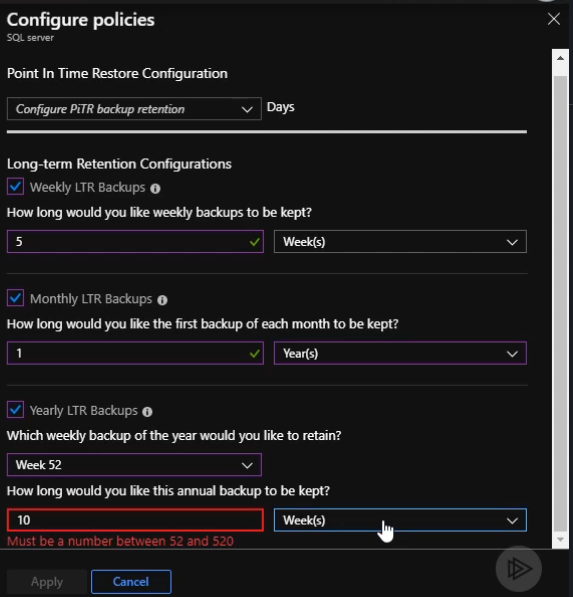
We can apply these rules to containers or to a subset of blobs.

* Similar to other Azure resources, we can work with lifecycle rules using Azure portal, PowerShell, Azure CLI, or the REST APIs.
* Note that when we create rules, the numbers of days that we specify are from the last modification date, not from the time we moved the blob to the cool storage or archive.
* Once the lifecycle management rule is created, we can click on the rule and enabler/disable it or remove conditions from it or even delete it.

**Azure SQL Database Backup Retention (LTR, PITR)**

* Azure SQL Database uses SQL Server technology to create database backups.
* Full database backups are created every week, differential backups every 12 hours, and transaction log backups every 5-10 minutes.
* The backups are stored in storage blobs, and are replicated through different regions (RA-GRS).
* As part of Organizations data management, it is important to make sure these Azure SQL Database backups are available before the retention time is over, and are securely deleted after the retention time is expired.
* Azure SQL Database has 2 backup types:
  + PITR Backup (Point in Time retention):
    - Taken automatically
    - Default retention time of these backups is 7 days. We can configure this option of 7, 14, 21, 28, and 35 days. The maximum amount of days we can set for a point-in-time backup retention is 35 days, and the backups will be deleted after that.
  + LTR Backup (Long Term Backup):
    - Taken manually.
    - They are not enabled by default.
    - Can configure the retention time of up to 10 years. The backups will be deleted after the retention time is expired.
    - We can change the backup retention period using the Azure portal, PowerShell, or the REST API.
    - There are 3 types of long term backups - weekly backups, monthly backups, and yearly backups, and you have the freedom to independently configure the retention time for each of these backups.

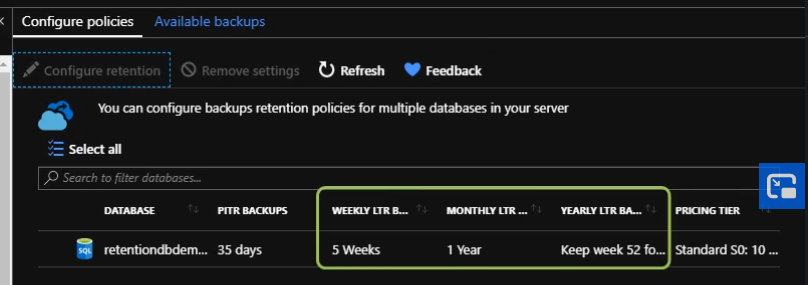
When we configure LTR we also specify how long will the backups be retained. Once of the weekly backup is used for yearly backup.



* We need to configure the backup retention at the server level. So in our Azure SQL Database dashboard, click on the server name, and we will be redirected to the SQL Server dashboard. Under Settings, click on Manage Backups. Here we can see that we already have the default retention of 7 days for automated point-in-time backup. Once for each DB on the server. Click on the “Available Backups” to see if there are any long term backups.

Click on the DB and go to Configure policies to manage PITR and LTR.

For PTR we have the options of 7, 14, 21, 28, and 35 days

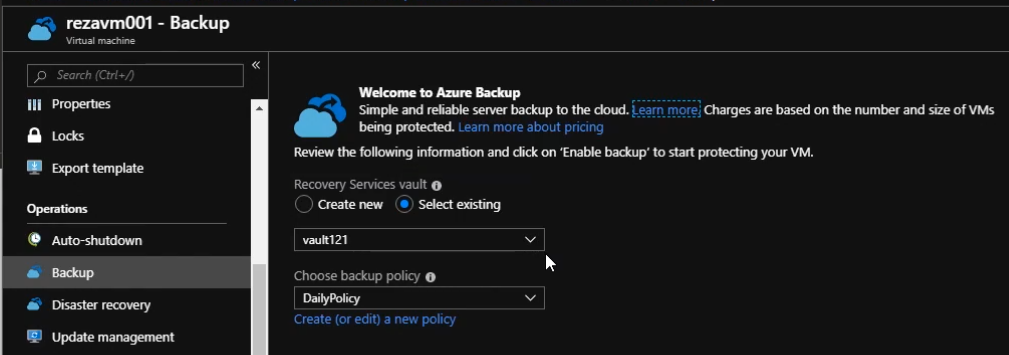


* Note that These backups will be kept in Azure Blob storage, and you need to pay for storage

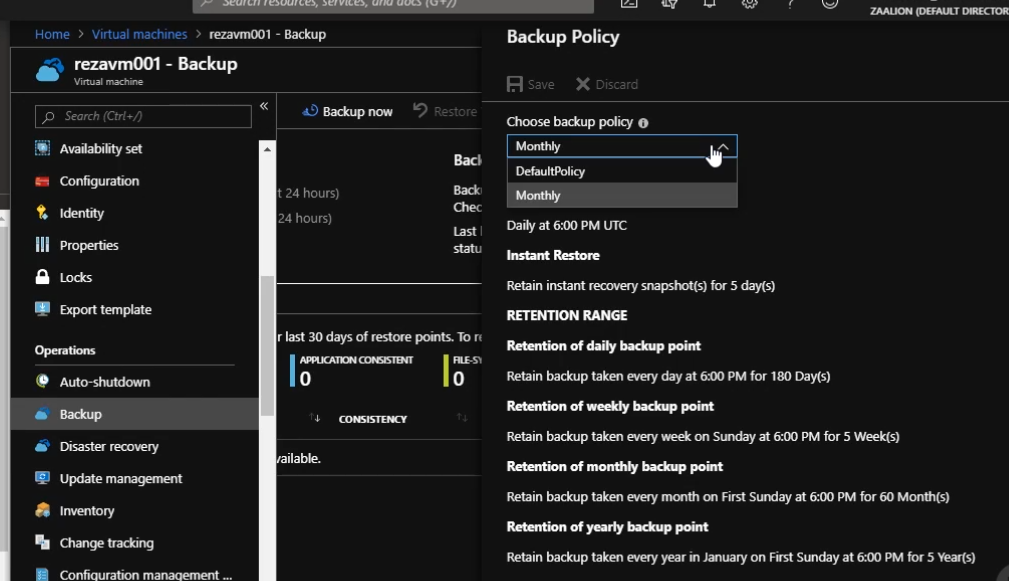
**Azure VM Backup Retention:**

* We can backup Azure virtual machines using the Azure Backup service.
* The virtual machine is protected once a backup copy of the data has been created in the Azure Backup vault.
* Azure Backup can keep up to 9, 999 recovery points, also known as backup copies or snapshots per protected virtual machine.
* The virtual machine backup frequency and retention period can be adjusted via a backup policy.
* We can easily go ahead and create a new policy and assign it to our virtual machine.
* To do so, we need to navigate to **the Recovery Services vault.(**1st we need to create a recovery services vault resource**)** Under Protected items, click on Backup items and you can see the list of all the existing resource backups. We can go ahead and create a new backup policy for our virtual machine. After we create this policy, we can go ahead and assign it to your virtual machine backup
* To reiterate virtual machine backups are stored inside Recovery Services vault.

Following shows when configuring backup for the VM, we need to select a recovery services vault and the correct backup policy. These policies are defined in the recovery services vault. The defaultPolicy provide 180 days of retention of backups.



* When we create a new backup policy, we need to decide on the backup schedule. We also need to decide how long I want to retain the instant recovery snapshots. We also need to decide on the retention on weekly backups, monthly backups, and yearly backups.



**Azure Monitor Log Retention – Application insights and Log Analytics Workspace:**

* Azure Monitor collects, analyzes, and acts on telemetry data collected from your applications and workloads.
* Azure Monitor collects following types of data:
  + Application Data- This is the data about performance and functionality of the code you have written. For example, you can use Application Insights plugin to collect performance data from your application and send them to Azure Monitor.
  + Guest OS Data- data about the operating system on which your application is running.
  + Azure resource data - Data about the operation and health of an Azure resource.
  + Subscription level data - data about the operation and management of an Azure subscription.
  + Azure Tenant Data - logs related to operation of tenant-level Azure services such as Azure Active Directory.
* These logs might have some audit information as well, which we might want to keep to comply with a few government or internal regulations. So it’s very important for us to have control over the retention period of these logs.
* Azure Monitor Logs are immutable (Once created, Activity Log entries cannot be modified or deleted by the system) we also can't change the logs in the interface or programmatically.
* Activity Log events are stored for 90 days, but the retention time can be set between 30 and 730 days. If you need to keep the logs for more than 730 days, export the logs to Azure Blob storage or Azure Event Hubs.
* To Change Azure monitor log retention, we go to Log Analytics workspace or Application inSights (These are different things) and simply click on usage and estimated costs(Both Log Analytics workspace and Application insights have this option). After that, click on data retention. You'll be faced with a slider which you can use to adjust the data retention period. The minimum retention period is 30 days, and the maximum is 730 days. After adjusting the data retention, simply click on OK.

# Configuring Data Sovereignty in Microsoft Azure:

**Microsoft Data Privacy Policies:**

1. No advertising use: Microsoft doesn't use customer data for purposes unrelated to the cloud services. This means your data won't be used for advertising purposes.
2. Third Party requests: Microsoft doesn't disclose your data, except as directed by you, or required by law.
3. Data Deletion: if you decide to delete your data in Microsoft Azure, or you terminate your contract, Azure will ensure safe deletion of your data so the next person using the same storage doesn't have access to the data belonging to you.
4. Data Location: Sometimes it is very important for companies to know in which jurisdiction and region their data is stored. You can transparently see where your data is stored in the Trust Center website.
5. Data transfer: Microsoft takes steps to ensure the data transfer is compliant with applicable laws.
6. Data ownership: Microsoft doesn't claim data ownership over the customer's information in Microsoft Azure. Microsoft doesn't inspect, approve, or monitor applications that customers deploy to Azure. Microsoft doesn't know what kind of data customers choose to store in Azure.

Microsoft trust Center website: <https://www.microsoft.com/en-us/trust-center>.  
This is a centralized place if you are looking for security, privacy, and compliance information related to Microsoft products.

<https://azuredatacentermap.azurewebsites.net/> to check where Microsoft data centers are available.

Most Azure services enable customers to specify the Region where their customer data will be stored. Microsoft may replicate to other Regions for data resiliency but Microsoft will not replicate or move customer data outside the Geo (see below for details). Customers and their end users may move, copy, or access their customer data from any location globally.

**Controlling Location of the data in Microsoft:**

Data sovereignty is the idea that data is subject to the laws of the nation it is collected from, so for the organizations to comply with data sovereignty regulations they need to be able to control the region where their data is stored.

2 methods of controlling location of data in Azure:

1. Azure Policies to restrict region deployments: Using Azure policies, you can put some governance around it so your employees can only go ahead and provision resources in the prescribed regions.
2. Control Data replication: To make sure replicated data ends up in the region that we want to. We can control data replication for storage accounts, virtual machines, and Azure SQL Database

Azure Policy:

* Azure policies can be used to restrict resource deployment to specific regions.
* Azure Policy is a service in Azure that you can use to define, assign, and manage policies.
* Using policies, we can enforce different rules over your resources so these resources stay compliant with corporate standards and service level agreements. Earlier we had created a policy to enforce company standards and proper resource tagging on our resources.
* Azure Policy evaluates our resources for non-compliance with assigned policies, and provides a report. We can go ahead and act on that report. Ex: imagine we have a few existing virtual machines which are provisioned in North America. After that, we go ahead and add a policy which limits all the subscription deployments to the European Union region. After that, in the Azure dashboard, we are going to get a report with a list of resources provisioned outside European Union so we can go ahead and fix those.
* There are many Azure policies out-of-the-box that you can go and use. For example, you can use an Azure policy to only allow provisioning of small virtual machine sizes in a subscription.
* Policies can be assigned at multiple levels. These levels are management group, subscription, and resource group. Management group is a level above subscription, so you can use that to manage multiple subscriptions under one umbrella.
* You can use pre-defined policies, or define yours if needed.
* To use a policy, first we need to define a policy. A policy is a JSON document. We can create a policy definition in the portal or programmatically. In case or predefined policies, definition is already created and ready to use. The predefined policies are called as “Built-in”
* The policy definition should be assigned to take place within a specific scope.
* Policies have parameters, so we can use them to generalize the policy definitions. For example, you can have a generalized policy which can be used to restrict deployment to regions, and then pass the list of prescribed regions as the policy parameter.
* The effect of the policy can be Deny. Deny means that if the condition of the policy is not met, the action will be denied. There are other types of effects as well.
* When we define a policy, we also have options to specify Exclusions. This allows some resources to be exempt from policy restrictions.

Control Data Replication Regions:

Here we cover 1) Azure Storage Redundancy 2) Virtual Machine disaster recovery 3) Azure SQL Database geo-replication

Azure Storage redundancy:

* The data in your Microsoft Azure storage account is always replicated to ensure durability and high availability. This protects your data from hardware failures, network or power outages, and massive natural disasters affecting Azure data centers.
* You have the option to replicate your data within one data center, across zonal data centers within the same region, or across geographically separated regions.
* There are six Azure storage redundancy options that you can choose from:
  + Locally Redundant Storage (LRS): Cheapest option. Replicates data within the same Data Center.
  + Zone redundant Storage (ZRS): Data will be replicated among different data centers in the same region.
  + Geo-redundant storage
  + Read-access geo-redundant storage
  + Geo-zone-redundant storage
  + Read-access geo-zone-redundant storage.

The Last 4 options replicate data amongst regions. With these we have the flexibility to choose to which target region your storage account data will be replicated. The Target region should be is in compliance with our data sovereignty regulations. This can be configured Azure portal and programmatically.

* To see which region the storage account is replicated, we can go to the Storage account and click on “Geo Replication”. Here we can see which region the data is replicated to. By default the paired region is used.
* Note that we can change the geo replication option by going under “Configuration” and changing the geo replication option. If we change to LRS or ZRS, we don’t see any replication info under the “Geo replication” tab.

Virtual Machine Disaster Recovery:

* **Azure Site Recovery** manages and orchestrates disaster recovery of on-premises machines and Azure virtual machines. This includes replication, failover, and recovery.
* You can set up disaster recovery for an Azure virtual machine by replicating it to a different Azure region. The target region is configurable programmatically or in Azure portal.
* The target region we choose should be in compliance with the Data sovereignty regulations.
* To configure, we go to the VM main page and click on Disaster recovery. Here we can see the default region for replication. Default is the paired region. We can use the dropdown and choose another region. When we select a new region, a new RG, Vnet would be created in the target region. After starting the replication , it might take some time for replication to complete.

Azure SQL DB Active Geo Replication:

* Active geo-replication is an Azure SQL Database feature that allows you to create readable secondary databases of your database.
* These secondaries can be created in the same or different data center or region, and you can define up to four secondaries in the same or different regions.
* You can configure these secondary regions in the Azure portal, and same as other resources, you need to make sure the target regions you choose to replicate your data to are in compliance with your data sovereignty regulations.
* To Configure go to the DB main page > Geo Replication. Here we can see the primary region for our DB. However we can go and select upto 4 secondary regions. When we choose a secondary we need to go through the entire process of provisioning a new DB on a server in the secondary region.