Azure PaaS Core reference

Threat Model

Prepared by

Simone Curzi

Preparation Date

16 May 2021

Contributors

* John Doe

Table of Contents

[Table of Contents 2](#_Toc72058296)

[Introduction 6](#_Toc72058297)

[Executive Overview 8](#_Toc72058298)

[The Model 11](#_Toc72058299)

[Assumptions 11](#_Toc72058300)

[The Severities 11](#_Toc72058301)

[Diagrams 11](#_Toc72058302)

[Reference 12](#_Toc72058303)

[External Interactors 12](#_Toc72058304)

[User 12](#_Toc72058305)

[Processes 12](#_Toc72058306)

[API Management 12](#_Toc72058307)

[Application Gateway 13](#_Toc72058308)

[Business Logic Web API 13](#_Toc72058309)

[Data processing Azure Function 13](#_Toc72058310)

[Static Content Web App 13](#_Toc72058311)

[Data Store 14](#_Toc72058312)

[Azure Key Vault 14](#_Toc72058313)

[Azure SQL 14](#_Toc72058314)

[Azure Storage 15](#_Toc72058315)

[Cosmos DB 15](#_Toc72058316)

[Flows 15](#_Toc72058317)

[Execute request 15](#_Toc72058318)

[Forward request 15](#_Toc72058319)

[Forward request to BL 15](#_Toc72058320)

[Get processed data 15](#_Toc72058321)

[Get secrets 16](#_Toc72058322)

[Get user-provided file 16](#_Toc72058323)

[Get/set user data 16](#_Toc72058324)

[Save processed data 16](#_Toc72058325)

[Send request 16](#_Toc72058326)

[Store a file to Azure Storage 16](#_Toc72058327)

[Trust Boundaries 16](#_Toc72058328)

[Azure Trust Boundary 16](#_Toc72058329)

[Threat Types 17](#_Toc72058330)

[Credentials for the resource may be stolen and used to access it 18](#_Toc72058331)

[Sensitive data may be disclosed due to lack of encryption at rest 19](#_Toc72058332)

[High-privileged accounts can be exploited to perform malicious actions 19](#_Toc72058333)

[Sensitive data stored may be disclosed due to weak authorization 20](#_Toc72058334)

[Sensitive data may be disclosed in transit 21](#_Toc72058335)

[Secrets may be disclosed due to weak protection 22](#_Toc72058336)

[Misconfigurations may allow compromise 23](#_Toc72058337)

[Malicious content may be sent to the solution 24](#_Toc72058338)

[Attacks may go undetected due to lack of monitoring 24](#_Toc72058339)

[An internal resource may be accessed by external actors 26](#_Toc72058340)

[A malicious actor may compromise the content of the storage 27](#_Toc72058341)

[Solution may be tampered due to weak protection 28](#_Toc72058342)

[A malicious actor may cause a full-scale outage 28](#_Toc72058343)

[Mitigations 30](#_Toc72058344)

[Adopt a Geographically Redundant configuration 31](#_Toc72058345)

[Adopt active geo-replication 31](#_Toc72058346)

[Adopt Azure Application Gateway with WAF 31](#_Toc72058347)

[Adopt Resource Tokens to get access to data in Cosmos DB 32](#_Toc72058348)

[Apply a transparent encryption 32](#_Toc72058349)

[Apply Client-Side Encryption for Azure Storage 33](#_Toc72058350)

[Apply column-level encryption 33](#_Toc72058351)

[Apply SQL Database dynamic data masking 33](#_Toc72058352)

[Apply strong channel encryption 34](#_Toc72058353)

[Apply the least privilege principle 35](#_Toc72058354)

[Azure Private Link for Azure SQL Database and Azure Synapse Analytics 36](#_Toc72058355)

[Azure SQL Database and Azure Synapse Analytics network access controls 36](#_Toc72058356)

[Configure access to Azure Cosmos DB from virtual networks (VNet) 36](#_Toc72058357)

[Configure Always Encrypted by using Azure Key Vault 37](#_Toc72058358)

[Configure Azure Defender for App Service 37](#_Toc72058359)

[Configure Azure Defender for Key Vault 38](#_Toc72058360)

[Configure Azure Defender for SQL 38](#_Toc72058361)

[Configure Azure Defender for Storage 39](#_Toc72058362)

[Configure Azure Private Link for an Azure Cosmos account 39](#_Toc72058363)

[Configure customer-managed keys for your Azure Cosmos account with Azure Key Vault 40](#_Toc72058364)

[Configure IP firewall in Azure Cosmos DB 40](#_Toc72058365)

[Configure the allowed CipherSuites 41](#_Toc72058366)

[Configure the minimum required version of Transport Layer Security (TLS) 41](#_Toc72058367)

[Create custom events to detect solution-specific security attacks 42](#_Toc72058368)

[Define and enforce actions for each Alert 42](#_Toc72058369)

[Enable all logging features in Azure Services 43](#_Toc72058370)

[Enable soft delete for AKV 44](#_Toc72058371)

[Enable soft delete for blobs 44](#_Toc72058372)

[Enable the Secure transfer required option on all of your storage accounts 44](#_Toc72058373)

[Ensure Azure Key Vault availability and redundancy 45](#_Toc72058374)

[Have a revocation plan in place 45](#_Toc72058375)

[Implement high-availability for Azure SQL 45](#_Toc72058376)

[Implement multi-region high-availability for Cosmos DB 46](#_Toc72058377)

[Limit duration of Service SAS Tokens not associated with a stored access policy 47](#_Toc72058378)

[Limit exposure to inbound network traffic 47](#_Toc72058379)

[Limit network exposure for Azure Key Vault 48](#_Toc72058380)

[Limit shared access signature (SAS) tokens to HTTPS connections only 48](#_Toc72058381)

[Lock resources to prevent unexpected changes 49](#_Toc72058382)

[Raise alerts when significant events occur 49](#_Toc72058383)

[Recover the Cosmos DB 50](#_Toc72058384)

[Recover the database using Point in Time Restore 50](#_Toc72058385)

[Redirect HTTP to HTTPS 51](#_Toc72058386)

[Regenerate your account keys periodically 51](#_Toc72058387)

[Reject requests not protected by TLS 51](#_Toc72058388)

[Restrict service access to data operations in Azure Cosmos DB 52](#_Toc72058389)

[Secure application configuration data using Environment Variables 52](#_Toc72058390)

[Secure Azure Cosmos keys using Azure Key Vault 53](#_Toc72058391)

[Secure your account access keys with Azure Key Vault 53](#_Toc72058392)

[Store business-critical blob data with immutable storage 53](#_Toc72058393)

[Store secrets in Azure Key Vault 54](#_Toc72058394)

[Use Azure AD Service Accounts 55](#_Toc72058395)

[Use FTPS 55](#_Toc72058396)

[Use Managed Identities 56](#_Toc72058397)

[Use separate Key Vaults 57](#_Toc72058398)

[Use system-assigned managed identities to access Azure Cosmos DB data 58](#_Toc72058399)

[Use the isolated pricing tier 58](#_Toc72058400)

[Use user delegation SAS Tokens 58](#_Toc72058401)

[Use virtual network service endpoints and rules for servers in Azure SQL Database 59](#_Toc72058402)

[Validate all input before processing it 59](#_Toc72058403)

[The Roadmap 60](#_Toc72058404)

# Introduction

The present document describes a security assessment performed on the design of a solution called Azure PaaS Core reference, using the Threat Modeling approach.

OWASP[[1]](#footnote-1) defines Threat Models as “structured representations of all the information that affects the security of an application. In essence, [a Threat Model] is a view of the application and its environment through security glasses”[[2]](#footnote-2).

Continuing with its description, OWASP defines “Threat Modeling [as the] process for capturing, organizing, and analyzing all of this information. Threat Modeling enables informed decision-making about application security risk. In addition to producing a model, typical threat modeling efforts also produce a prioritized list of security improvements to the concept, requirements, design, or implementation”.

The Azure PaaS Core reference system is a simple PaaS solution used to test the Azure PaaS Core template and demonstrate its usage.

The reference scenario represents a simple Web Application on Azure, having the intent of collecting documents to be processed by the solution. You may think to a request submission system, where the user provides part of the information by filling some web pages, and part by sending files that need to be processed. The received data is stored in part into Cosmos DB and in part into Azure SQL, Cosmos DB contains the data related to the interaction of the user with the system and to her requests. The Azure SQL instead receives the data extracted from the uploaded documents.

The example does include only the Data Ingress processes plus some processing, and remains at an high level. Moreover, it does not analyze the Administration scenario nor the CI/CD processes.

All the mitigations identified here are from https://docs.microsoft.com.

This document includes the following sections:

* The **Executive Overview**, which represents a summary view of the solution and its risks.
* **The Model**, which describes how the Solution in scope has been understood.
* The **Threat Types**, describing the identified risks.
* The **Mitigations**, which lists the activities that could address the identified risks.
* **The Roadmap**, which describes a potential plan for improving the security of the Solution.

For a non-technical view, it is recommended to focus on the **Executive Overview**. Architects, Security Experts and other technical roles may find value in the whole Report.

This document assumes a good understanding of the main characteristics of the Solution in scope.

# Executive Overview

The Threat Model describes the main risks identified for the solution in scope and what can be done to address them. The analysis has identified a total of 71 potential issues, called Threat Events, which can be categorized under 13 categories, called Threat Types. The following pie chart shows the distribution of the Threat Types by Severity. It is notable that the analysis has identified 0 Critical and 9 High Severity Threat Types.

Figure 1 - The Threat Types by Severity chart.

To address those Threat Types, the analysis has identified a total of 59 Mitigations. The following pie chart shows the distribution of the Mitigations by Status.

Figure 2 - The Mitigations by Status chart.

The meaning of the potential States is:

* **Existing**, which indicates a mitigation that already exist, when the Threat Model is performed.
* **Proposed**, used to indicates new Mitigations you are proposing, as a Threat Modeler.
* **Implemented**, which is used for existing Mitigations which have been introduced as a consequence of your Threat Model.
* **Approved**, to indicate a Mitigation that is known (for example it is tracked in the Backlog), but it has not been planned, yet.
* **Planned** is to be used for Mitigations that are known and have been planned for implementation.

The Threat Types identified during the Threat Modeling exercise do not necessarily correspond to vulnerabilities which will necessarily be exploited by attackers. This means that the Threat Model and this report do not describe a prescriptive set of activities that must be necessarily done, but a guidance to understand the risk represented by the analyzed solution and a starting point to decide how to address it. Thus, the recommended approach is to start a conversation to define what to do for each Threat Type. In fact, the main Stakeholders of this initiative have various options:

* Simply accept the risk, track it and do not implement any action.
* Select one or more Mitigations, picking from the proposed list or identifying additional options not included in this document, to mitigate the risk.
* Avoid the risk, by completely removing the functionality that is causing it.
* Transfer the risk to someone else, for example by getting an Insurance.

Ultimately, it is important to understand that the goal of Security is not to nullify the Risk, but to make it acceptable.

# The Model

The present Chapter represents the understanding related to the analyzed Solution, through a representative diagram.

## Assumptions

Assumptions are used to indicate conditions or information that you are using as part of the Threat Model. They are particularly important, because the Threat Model relies on them to be true. In other words, if they are false, then the Threat Model would be in whole or in part be false.

The Threat Model has been created under the following Assumptions.

* Cloud PaaS Services are assumed to be secure. As every software, it is certain that they have security vulnerabilities, but they are owned by the Cloud Provider. The configuration of the Cloud Services is instead a responsibility of the Organization owning the solution, and therefore it is analyzed by the Threat Model.

## The Severities

Threat Type and Events have been assigned different Severities, to represent their importance, in accordance with the following table.

|  |  |
| --- | --- |
| Severity | Description |
| Critical | Critical Severity. It is related to the most important issues, which could cause a catastrophic failure of the solution and critical damage to the involved counterparts. |
| High | High Severity. It is related to important issues that may cause major disruption and thus should be seriously considered. |
| Medium | Medium Severity. It is related to issues that should represent some concern and thus should not be overlooked. |
| Low | Low Severity. It is related to minor issues. |
| Info | Informational only. It is related to mitigated issues or to topics that are included for completeness. |
| Unknown | Unknown or unassigned severity. |

Table 1 - The definition of the various Severity levels.

## Diagrams

The analysis has produced the following diagrams, representative of the Solution in scope.

### Reference

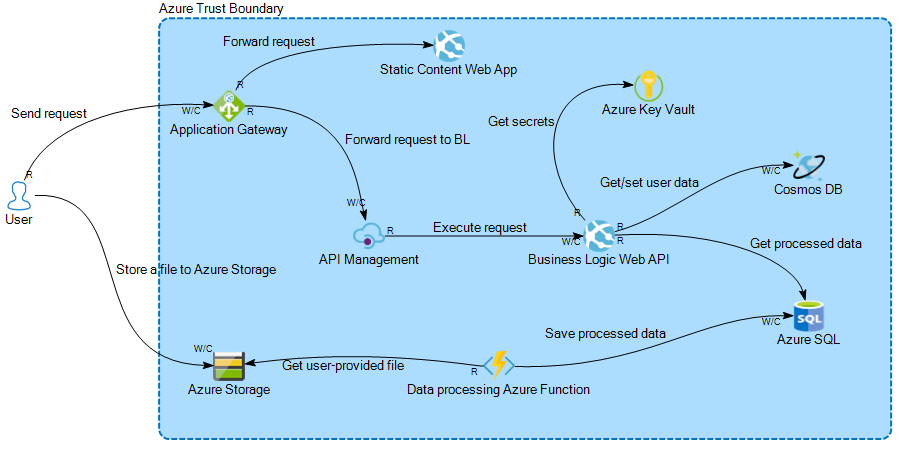


Figure 3 - The 'Reference' scenario.

The main diagram for the reference scenario.

## External Interactors

The External Interactors identified by the analysis are represented here.

### User

|  |  |
| --- | --- |
| High Privileges | False |
| Promiscuous use | True |
| Shared device | False |

## Processes

The Processes identified by the analysis are represented here.

### API Management

|  |  |
| --- | --- |
| Description | API Management (APIM) is a way to create consistent and modern API gateways for existing back-end services.  API Management helps organizations publish APIs to external, partner, and internal developers to unlock the potential of their data and services. Businesses everywhere are looking to extend their operations as a digital platform, creating new channels, finding new customers and driving deeper engagement with existing ones. API Management provides the core competencies to ensure a successful API program through developer engagement, business insights, analytics, security, and protection. You can use Azure API Management to take any backend and launch a full-fledged API program based on it.  For more information, please refer to https://docs.microsoft.com/en-us/azure/api-management/api-management-key-concepts. |

### Application Gateway

|  |  |
| --- | --- |
| Description | Azure Application Gateway is a web traffic load balancer that enables you to manage traffic to your web applications. Traditional load balancers operate at the transport layer (OSI layer 4 - TCP and UDP) and route traffic based on source IP address and port, to a destination IP address and port.  For more information, please refer to https://docs.microsoft.com/en-us/azure/application-gateway/overview. |

### Business Logic Web API

|  |  |
| --- | --- |
| Description | A Web API is an application programming interface for either a web server or a web browser.  Source: https://en.wikipedia.org/wiki/Web\_API.  Web APIs are typically hosted in Azure App Service, which is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. You can develop in your favorite language, be it .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python. Applications run and scale with ease on both Windows and Linux-based environments.  App Service not only adds the power of Microsoft Azure to your application, such as security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from Azure DevOps, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and TLS/SSL certificates.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/overview. |

### Data processing Azure Function

|  |  |
| --- | --- |
| Description | Azure Functions allows you to run small pieces of code (called "functions") without worrying about application infrastructure. With Azure Functions, the cloud infrastructure provides all the up-to-date servers you need to keep your application running at scale.  A function is "triggered" by a specific type of event. Supported triggers include responding to changes in data, responding to messages, running on a schedule, or as the result of an HTTP request.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-functions/functions-overview. |

### Static Content Web App

|  |  |
| --- | --- |
| Description | Application software that runs on a web server, unlike computer-based software programs that are stored locally on the Operating System (OS) of the device. Web applications are accessed by the user through a web browser with an active internet connection. These applications are programmed using a client–server modeled structure—the user ("client") is provided services through an off-site server that is hosted by a third-party.  Source: https://en.wikipedia.org/wiki/Web\_application.  Web Applications are typically hosted in Azure App Service, which is an HTTP-based service for hosting web applications, REST APIs, and mobile back ends. You can develop in your favorite language, be it .NET, .NET Core, Java, Ruby, Node.js, PHP, or Python. Applications run and scale with ease on both Windows and Linux-based environments.  App Service not only adds the power of Microsoft Azure to your application, such as security, load balancing, autoscaling, and automated management. You can also take advantage of its DevOps capabilities, such as continuous deployment from Azure DevOps, GitHub, Docker Hub, and other sources, package management, staging environments, custom domain, and TLS/SSL certificates.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/overview. |

## Data Store

The Data Stores identified by the analysis are represented here.

### Azure Key Vault

|  |  |
| --- | --- |
| Description | Azure Key Vault helps solve the following problems:  Secrets Management - Azure Key Vault can be used to Securely store and tightly control access to tokens, passwords, certificates, API keys, and other secrets  Key Management - Azure Key Vault can also be used as a Key Management solution. Azure Key Vault makes it easy to create and control the encryption keys used to encrypt your data.  Certificate Management - Azure Key Vault is also a service that lets you easily provision, manage, and deploy public and private Transport Layer Security/Secure Sockets Layer (TLS/SSL) certificates for use with Azure and your internal connected resources.  Azure Key Vault has two service tiers: Standard, which encrypts with a software key, and a Premium tier, which includes HSM-protected keys.  Azure Key Vault are considered Data Stores, because their prevalent role is storage of secrets and keys. |

### Azure SQL

|  |  |
| --- | --- |
| Description | Azure SQL Database is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user involvement. Azure SQL Database is always running on the latest stable version of the SQL Server database engine and patched OS with 99.99% availability. PaaS capabilities that are built into Azure SQL Database enable you to focus on the domain-specific database administration and optimization activities that are critical for your business.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/sql-database-paas-overview. |

### Azure Storage

|  |  |
| --- | --- |
| Description | The Azure Storage platform is Microsoft's cloud storage solution for modern data storage scenarios. Core storage services offer a massively scalable object store for data objects, disk storage for Azure virtual machines (VMs), a file system service for the cloud, a messaging store for reliable messaging, and a NoSQL store.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction. |

### Cosmos DB

|  |  |
| --- | --- |
| Description | Azure Cosmos DB is a fully managed NoSQL database for modern app development. Single-digit millisecond response times, and automatic and instant scalability, guarantee speed at any scale. Business continuity is assured with SLA-backed availability and enterprise-grade security. App development is faster and more productive thanks to turnkey multi-master data distribution anywhere in the world, open source APIs and SDKs for popular languages. As a fully managed service, Azure Cosmos DB takes database administration off your hands with automatic management, updates and patching. It also handles capacity management with cost-effective serverless and automatic scaling options that respond to application needs to match capacity with demand.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/introduction. |

## Flows

The Flows identified by the analysis are represented here.

### Execute request

|  |  |
| --- | --- |
| Source | API Management |
| Target | Business Logic Web API |
| Flow Type | Read-Write/Command |

### Forward request

|  |  |
| --- | --- |
| Description | Forward the request to get static content. |
| Source | Application Gateway |
| Target | Static Content Web App |
| Flow Type | Read only |

### Forward request to BL

|  |  |
| --- | --- |
| Source | Application Gateway |
| Target | API Management |
| Flow Type | Read-Write/Command |

### Get processed data

|  |  |
| --- | --- |
| Source | Business Logic Web API |
| Target | Azure SQL |
| Flow Type | Read only |

### Get secrets

|  |  |
| --- | --- |
| Source | Business Logic Web API |
| Target | Azure Key Vault |
| Flow Type | Read only |

### Get user-provided file

|  |  |
| --- | --- |
| Source | Data processing Azure Function |
| Target | Azure Storage |
| Flow Type | Read only |

### Get/set user data

|  |  |
| --- | --- |
| Source | Business Logic Web API |
| Target | Cosmos DB |
| Flow Type | Read-Write/Command |

### Save processed data

|  |  |
| --- | --- |
| Source | Data processing Azure Function |
| Target | Azure SQL |
| Flow Type | Write/Command only |

### Send request

|  |  |
| --- | --- |
| Description | Send a request to get static content from the system. |
| Source | User |
| Target | Application Gateway |
| Flow Type | Read-Write/Command |

### Store a file to Azure Storage

|  |  |
| --- | --- |
| Description | Store a file in Azure Storage, using a URL provided by the Business Logic Web API and containing a SAS Token allowing creating a specific object. |
| Source | User |
| Target | Azure Storage |
| Flow Type | Write/Command only |

## Trust Boundaries

The Trust Boundary identified by the analysis are represented here.

### Azure Trust Boundary

# Threat Types

This section represents the various issues identified as part of the Threat Modeling analysis.

For each Threat Type, you have represented here various details, including:

* The **Name** of the Threat Type, as title.
* The **Severity** of the finding, calculated as maximum of the related Threat Events. The meaning of the various levels is described in the table below.

|  |  |
| --- | --- |
| Severity | Description |
| Critical | Critical Severity. It is related to the most important issues, which could cause a catastrophic failure of the solution and critical damage to the involved counterparts. |
| High | High Severity. It is related to important issues that may cause major disruption and thus should be seriously considered. |
| Medium | Medium Severity. It is related to issues that should represent some concern and thus should not be overlooked. |
| Low | Low Severity. It is related to minor issues. |
| Info | Informational only. It is related to mitigated issues or to topics that are included for completeness. |
| Unknown | Unknown or unassigned severity. |

Table 2 - Definition of the Severities.

* The **Description** of the Threat Type.
* The list of **Affected Objects**, which are the Entities, Flows or the Threat Model itself. For each of them, the related type is expressed through the short form[[3]](#footnote-3) as a prefix between square brackets, and the related severity for the associated Threat Event.
* A list of **Mitigations** split by Status. This list is expressed by a table
  + The **Object** to which the mitigation applies.
  + The Name of the **Mitigation**.
  + The specific **Severity** for the object.
  + The **Strength** of the Mitigation, in accordance with the following table.[[4]](#footnote-4)

|  |  |
| --- | --- |
| Strength | Description |
| Maximum | Strength is maximum: it may be enough to mitigate the Threat. |
| Strong | Strength is important, but it may not be enough to address the Threat alone. |
| Average | Strength is average. It significantly mitigates the Threat, but probably not enough. |
| Weak | Low strength. It has some importance for mitigating the Threat. |
| Negligible | Negligible strength. It has no measurable impact on mitigating the Threat. |

Table 3 - Definition of possible Strengths for the Mitigations.

Threat Types are sorted by Severity.

## Credentials for the resource may be stolen and used to access it

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | If credentials are not protected adequately, they may be stolen and then used to get unwarranted access to the resource itself. |
| Affected Objects | * [F] Get user-provided file (High) * [F] Store a file to Azure Storage (High) * [F] Get/set user data (High) * [F] Save processed data (High) * [F] Get processed data (High) * [F] Get secrets (High) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Get processed data | Use Managed Identities | High | Strong | | [F] Get processed data | Use Azure AD Service Accounts | High | Average | | [F] Get secrets | Use Managed Identities | High | Strong | | [F] Get secrets | Use Azure AD Service Accounts | High | Average | | [F] Get secrets | Use separate Key Vaults | High | Average | | [F] Get user-provided file | Use Managed Identities | High | Strong | | [F] Get user-provided file | Secure your account access keys with Azure Key Vault | High | Average | | [F] Get user-provided file | Use Azure AD Service Accounts | High | Average | | [F] Get user-provided file | Use user delegation SAS Tokens | High | Average | | [F] Get user-provided file | Regenerate your account keys periodically | High | Average | | [F] Get user-provided file | Have a revocation plan in place | High | Average | | [F] Get user-provided file | Limit duration of Service SAS Tokens not associated with a stored access policy | High | Average | | [F] Get/set user data | Secure Azure Cosmos keys using Azure Key Vault | High | Average | | [F] Get/set user data | Use system-assigned managed identities to access Azure Cosmos DB data | High | Average | | [F] Save processed data | Use Managed Identities | High | Strong | | [F] Save processed data | Use Azure AD Service Accounts | High | Average | | [F] Store a file to Azure Storage | Use Managed Identities | High | Strong | | [F] Store a file to Azure Storage | Secure your account access keys with Azure Key Vault | High | Average | | [F] Store a file to Azure Storage | Use Azure AD Service Accounts | High | Average | | [F] Store a file to Azure Storage | Use user delegation SAS Tokens | High | Average | | [F] Store a file to Azure Storage | Regenerate your account keys periodically | High | Average | | [F] Store a file to Azure Storage | Have a revocation plan in place | High | Average | | [F] Store a file to Azure Storage | Limit duration of Service SAS Tokens not associated with a stored access policy | High | Average | |

## Sensitive data may be disclosed due to lack of encryption at rest

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | Data stored in the storage may be disclosed because it is not encrypted. |
| Affected Objects | * [D] Azure Storage (High) * [D] Cosmos DB (High) * [D] Azure SQL (High) |
| Existing Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [D] Azure SQL | Apply a transparent encryption | High | Weak | | [D] Azure Storage | Apply a transparent encryption | High | Weak | | [D] Cosmos DB | Apply a transparent encryption | High | Weak | |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [D] Azure SQL | Configure Always Encrypted by using Azure Key Vault | High | Maximum | | [D] Azure SQL | Apply column-level encryption | High | Weak | | [D] Azure Storage | Apply Client-Side Encryption for Azure Storage | High | Maximum | | [D] Cosmos DB | Configure customer-managed keys for your Azure Cosmos account with Azure Key Vault | High | Weak | |

## High-privileged accounts can be exploited to perform malicious actions

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | The assignment of excessive rights to accounts may allow a malicious actor who gets access to their credentials to compromise the system. |
| Affected Objects | * [F] Get user-provided file (High) * [F] Store a file to Azure Storage (High) * [F] Get/set user data (High) * [F] Save processed data (High) * [F] Get processed data (High) * [F] Get secrets (High) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Get processed data | Apply the least privilege principle | High | Strong | | [F] Get secrets | Apply the least privilege principle | High | Strong | | [F] Get user-provided file | Apply the least privilege principle | High | Strong | | [F] Get/set user data | Adopt Resource Tokens to get access to data in Cosmos DB | High | Strong | | [F] Get/set user data | Apply the least privilege principle | High | Strong | | [F] Get/set user data | Restrict service access to data operations in Azure Cosmos DB | High | Average | | [F] Save processed data | Apply the least privilege principle | High | Strong | | [F] Store a file to Azure Storage | Apply the least privilege principle | High | Strong | |

## Sensitive data stored may be disclosed due to weak authorization

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | Sensitive data stored in the Data Store may be disclosed as a result of an unprotected storage. An example may be related to the need to expose a table containing sensitive data because of some other content, when the protection of the sensitive data is left to the caller. An attacker may be able to disclose the sensitive data, by changing the query performed by the database client, for example through a SQL Injection. |
| Affected Objects | * [F] Get user-provided file (High) * [F] Store a file to Azure Storage (High) * [F] Get/set user data (High) * [F] Save processed data (High) * [F] Get processed data (High) * [F] Get secrets (High) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Get processed data | Apply SQL Database dynamic data masking | High | Strong | | [F] Get processed data | Apply the least privilege principle | High | Strong | | [F] Get secrets | Apply the least privilege principle | High | Strong | | [F] Get user-provided file | Apply the least privilege principle | High | Strong | | [F] Get/set user data | Apply the least privilege principle | High | Strong | | [F] Save processed data | Apply the least privilege principle | High | Strong | | [F] Store a file to Azure Storage | Apply the least privilege principle | High | Strong | |

## Sensitive data may be disclosed in transit

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | Sensitive data may be transferred and intercepted by a man-in-the-middle.  It is important to note that internal Flows between Azure services stay in Azure. Nevertheless, since the connection goes through the shared networking in Azure, you should always encrypt all communication.  For more information, please refer to https://owasp.org/www-community/attacks/Man-in-the-middle\_attack. |
| Affected Objects | * [F] Send request (High) * [F] Forward request (High) * [F] Get user-provided file (High) * [F] Store a file to Azure Storage (High) * [F] Forward request to BL (High) * [F] Execute request (High) * [F] Get/set user data (High) * [F] Save processed data (High) * [F] Get processed data (High) * [F] Get secrets (High) |
| Existing Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Execute request | Apply strong channel encryption | High | Strong | | [F] Forward request | Apply strong channel encryption | High | Strong | | [F] Forward request to BL | Apply strong channel encryption | High | Strong | | [F] Get processed data | Apply strong channel encryption | High | Strong | | [F] Get secrets | Apply strong channel encryption | High | Strong | | [F] Get user-provided file | Apply strong channel encryption | High | Strong | | [F] Get/set user data | Apply strong channel encryption | High | Strong | | [F] Save processed data | Apply strong channel encryption | High | Strong | | [F] Send request | Apply strong channel encryption | High | Strong | | [F] Store a file to Azure Storage | Apply strong channel encryption | High | Strong | |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Execute request | Configure the minimum required version of Transport Layer Security (TLS) | High | Average | | [F] Execute request | Reject requests not protected by TLS | High | Average | | [F] Forward request | Configure the minimum required version of Transport Layer Security (TLS) | High | Average | | [F] Forward request | Redirect HTTP to HTTPS | High | Average | | [F] Forward request to BL | Configure the minimum required version of Transport Layer Security (TLS) | High | Average | | [F] Forward request to BL | Configure the allowed CipherSuites | High | Average | | [F] Get user-provided file | Limit shared access signature (SAS) tokens to HTTPS connections only | High | Average | | [F] Get user-provided file | Configure the minimum required version of Transport Layer Security (TLS) | High | Average | | [F] Get user-provided file | Enable the Secure transfer required option on all of your storage accounts | High | Average | | [F] Store a file to Azure Storage | Limit shared access signature (SAS) tokens to HTTPS connections only | High | Average | | [F] Store a file to Azure Storage | Configure the minimum required version of Transport Layer Security (TLS) | High | Average | | [F] Store a file to Azure Storage | Enable the Secure transfer required option on all of your storage accounts | High | Average | |

## Secrets may be disclosed due to weak protection

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | The deployed solution configuration and secrets may be disclosed or tampered by a malicious actor, because they are not adequately protected. |
| Affected Objects | * [P] Static Content Web App (High) * [P] Business Logic Web API (High) * [P] Application Gateway (High) * [P] Data processing Azure Function (High) * [P] API Management (High) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [P] API Management | Use Managed Identities | High | Maximum | | [P] API Management | Store secrets in Azure Key Vault | High | Strong | | [P] API Management | Use separate Key Vaults | High | Average | | [P] Application Gateway | Store secrets in Azure Key Vault | High | Strong | | [P] Application Gateway | Use separate Key Vaults | High | Average | | [P] Business Logic Web API | Use Managed Identities | High | Maximum | | [P] Business Logic Web API | Store secrets in Azure Key Vault | High | Strong | | [P] Business Logic Web API | Secure application configuration data using Environment Variables | High | Average | | [P] Business Logic Web API | Use separate Key Vaults | High | Average | | [P] Data processing Azure Function | Use Managed Identities | High | Maximum | | [P] Data processing Azure Function | Store secrets in Azure Key Vault | High | Strong | | [P] Data processing Azure Function | Use separate Key Vaults | High | Average | | [P] Static Content Web App | Use Managed Identities | High | Maximum | | [P] Static Content Web App | Store secrets in Azure Key Vault | High | Strong | | [P] Static Content Web App | Secure application configuration data using Environment Variables | High | Average | | [P] Static Content Web App | Use separate Key Vaults | High | Average | |

## Misconfigurations may allow compromise

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | Mistakes in the design, implementation or configuration of the solution may be exploited by a malicious actor to compromise it. |
| Affected Objects | * [P] Static Content Web App (High) * [P] Business Logic Web API (High) * [D] Azure Storage (High) * [P] Data processing Azure Function (High) * [D] Azure SQL (High) * [D] Azure Key Vault (High) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [D] Azure Key Vault | Configure Azure Defender for Key Vault | High | Strong | | [D] Azure SQL | Configure Azure Defender for SQL | High | Strong | | [D] Azure Storage | Configure Azure Defender for Storage | High | Strong | | [P] Business Logic Web API | Configure Azure Defender for App Service | High | Average | | [P] Data processing Azure Function | Configure Azure Defender for App Service | High | Average | | [P] Static Content Web App | Configure Azure Defender for App Service | High | Average | |

## Malicious content may be sent to the solution

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | Malicious content may be sent to the solution and then it could be executed or stored to be sent back to other users. Typical examples of those types of attacks are Stored XSS (see https://owasp.org/www-community/Types\_of\_Cross-Site\_Scripting) or Code Injection/Remote Code Execution (see https://owasp.org/www-community/attacks/Code\_Injection). |
| Affected Objects | * [F] Send request (High) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Send request | Validate all input before processing it | High | Strong | | [F] Send request | Raise alerts when significant events occur | High | Average | | [F] Send request | Define and enforce actions for each Alert | High | Average | |

## Attacks may go undetected due to lack of monitoring

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **High** | |
| Description | Monitoring is a key practice to detect attacks while they occur. If it is weak, malicious actors may be able to compromise it without being detected.  Many Azure Services provide infrastructural monitoring capabilities out of the box, but this may not be sufficient from a security point of view, because the standard monitoring does not understand the solution, the data it manages or the Business context. |
| Affected Objects | * [F] Send request (High) * [F] Forward request (High) * [F] Get user-provided file (High) * [F] Store a file to Azure Storage (High) * [F] Forward request to BL (High) * [F] Execute request (High) * [F] Get/set user data (High) * [F] Save processed data (High) * [F] Get processed data (High) * [F] Get secrets (High) |
| Existing Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Execute request | Adopt Azure Application Gateway with WAF | High | Average | |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [F] Execute request | Configure Azure Defender for App Service | High | Average | | [F] Execute request | Enable all logging features in Azure Services | High | Average | | [F] Execute request | Define and enforce actions for each Alert | High | Average | | [F] Execute request | Create custom events to detect solution-specific security attacks | High | Average | | [F] Execute request | Raise alerts when significant events occur | High | Average | | [F] Forward request | Configure Azure Defender for App Service | High | Average | | [F] Forward request | Enable all logging features in Azure Services | High | Average | | [F] Forward request | Define and enforce actions for each Alert | High | Average | | [F] Forward request | Create custom events to detect solution-specific security attacks | High | Average | | [F] Forward request | Raise alerts when significant events occur | High | Average | | [F] Forward request to BL | Enable all logging features in Azure Services | High | Average | | [F] Forward request to BL | Define and enforce actions for each Alert | High | Average | | [F] Forward request to BL | Raise alerts when significant events occur | High | Average | | [F] Get processed data | Configure Azure Defender for SQL | High | Strong | | [F] Get processed data | Enable all logging features in Azure Services | High | Average | | [F] Get processed data | Define and enforce actions for each Alert | High | Average | | [F] Get processed data | Raise alerts when significant events occur | High | Average | | [F] Get secrets | Configure Azure Defender for Key Vault | High | Strong | | [F] Get secrets | Enable all logging features in Azure Services | High | Average | | [F] Get secrets | Define and enforce actions for each Alert | High | Average | | [F] Get secrets | Raise alerts when significant events occur | High | Average | | [F] Get user-provided file | Configure Azure Defender for Storage | High | Strong | | [F] Get user-provided file | Enable all logging features in Azure Services | High | Average | | [F] Get user-provided file | Define and enforce actions for each Alert | High | Average | | [F] Get user-provided file | Raise alerts when significant events occur | High | Average | | [F] Get/set user data | Enable all logging features in Azure Services | High | Average | | [F] Get/set user data | Define and enforce actions for each Alert | High | Average | | [F] Get/set user data | Raise alerts when significant events occur | High | Average | | [F] Save processed data | Configure Azure Defender for SQL | High | Strong | | [F] Save processed data | Enable all logging features in Azure Services | High | Average | | [F] Save processed data | Define and enforce actions for each Alert | High | Average | | [F] Save processed data | Raise alerts when significant events occur | High | Average | | [F] Send request | Enable all logging features in Azure Services | High | Average | | [F] Send request | Define and enforce actions for each Alert | High | Average | | [F] Send request | Raise alerts when significant events occur | High | Average | | [F] Store a file to Azure Storage | Configure Azure Defender for Storage | High | Strong | | [F] Store a file to Azure Storage | Enable all logging features in Azure Services | High | Average | | [F] Store a file to Azure Storage | Define and enforce actions for each Alert | High | Average | | [F] Store a file to Azure Storage | Raise alerts when significant events occur | High | Average | |

## An internal resource may be accessed by external actors

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **Medium** | |
| Description | Internal resources are typically not protected as the external ones, because they are assumed to be protected by the security controls in place at the perimeter. If an attacker manages to bypass the perimeter, then it may achieve undue control of the target systems. |
| Affected Objects | * [P] Static Content Web App (Medium) * [P] Business Logic Web API (Medium) * [P] Data processing Azure Function (Medium) * [P] API Management (Medium) * [D] Cosmos DB (Medium) * [D] Azure SQL (Medium) * [D] Azure Key Vault (Medium) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [P] API Management | Use Managed Identities | Medium | Strong | | [D] Azure Key Vault | Limit network exposure for Azure Key Vault | Medium | Average | | [D] Azure SQL | Use virtual network service endpoints and rules for servers in Azure SQL Database | Medium | Average | | [D] Azure SQL | Azure Private Link for Azure SQL Database and Azure Synapse Analytics | Medium | Average | | [D] Azure SQL | Azure SQL Database and Azure Synapse Analytics network access controls | Medium | Average | | [D] Azure SQL | Configure Azure Defender for SQL | Medium | Average | | [P] Business Logic Web API | Use Managed Identities | Medium | Strong | | [P] Business Logic Web API | Use the isolated pricing tier | Medium | Strong | | [P] Business Logic Web API | Limit exposure to inbound network traffic | Medium | Average | | [D] Cosmos DB | Configure IP firewall in Azure Cosmos DB | Medium | Average | | [D] Cosmos DB | Configure access to Azure Cosmos DB from virtual networks (VNet) | Medium | Average | | [D] Cosmos DB | Configure Azure Private Link for an Azure Cosmos account | Medium | Average | | [P] Data processing Azure Function | Use Managed Identities | Medium | Strong | | [P] Data processing Azure Function | Use the isolated pricing tier | Medium | Strong | | [P] Static Content Web App | Use Managed Identities | Medium | Strong | | [P] Static Content Web App | Use the isolated pricing tier | Medium | Strong | | [P] Static Content Web App | Limit exposure to inbound network traffic | Medium | Average | |

## A malicious actor may compromise the content of the storage

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **Medium** | |
| Description | The storage may be compromised and as a consequence all data may be lost. This may for example be associated to a ransowmware attack. |
| Affected Objects | * [D] Azure Storage (Medium) * [D] Cosmos DB (Low) * [D] Azure SQL (Medium) * [D] Azure Key Vault (Medium) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [D] Azure Key Vault | Enable soft delete for AKV | Medium | Average | | [D] Azure Key Vault | Configure Azure Defender for Key Vault | Medium | Average | | [D] Azure SQL | Recover the database using Point in Time Restore | Medium | Average | | [D] Azure SQL | Configure Azure Defender for SQL | Medium | Average | | [D] Azure Storage | Configure Azure Defender for Storage | Medium | Average | | [D] Azure Storage | Enable soft delete for blobs | Medium | Average | | [D] Azure Storage | Lock resources to prevent unexpected changes | Medium | Average | | [D] Azure Storage | Store business-critical blob data with immutable storage | Medium | Average | | [D] Cosmos DB | Recover the Cosmos DB | Low | Average | |

## Solution may be tampered due to weak protection

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **Medium** | |
| Description | The deployed solution code tampered by a malicious actor, because it is not adequately protected. |
| Affected Objects | * [P] Static Content Web App (Medium) * [P] Business Logic Web API (Medium) * [P] Data processing Azure Function (Medium) |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [P] Business Logic Web API | Use FTPS | Medium | Average | | [P] Data processing Azure Function | Use FTPS | Medium | Average | | [P] Static Content Web App | Use FTPS | Medium | Average | |

## A malicious actor may cause a full-scale outage

|  |  |  |
| --- | --- | --- |
| Severity | |  | | --- | | **Low** | |
| Description | A full-scale outage may cause the solution to be unavailable for its users. This may be caused by natural events but also by a malicious actor. |
| Affected Objects | * [D] Azure Storage (Low) * [D] Cosmos DB (Low) * [D] Azure SQL (Low) * [D] Azure Key Vault (Info) |
| Existing Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [D] Azure Key Vault | Ensure Azure Key Vault availability and redundancy | Info | Strong | |
| Proposed Mitigations | |  |  |  |  | | --- | --- | --- | --- | | Object | Mitigation | Severity | Strength | | [D] Azure SQL | Adopt active geo-replication | Low | Strong | | [D] Azure SQL | Implement high-availability for Azure SQL | Low | Strong | | [D] Azure Storage | Adopt a Geographically Redundant configuration | Low | Strong | | [D] Cosmos DB | Implement multi-region high-availability for Cosmos DB | Low | Strong | |

# Mitigations

This section describes the possible actions identified as already included as part of the Solution, or that have been proposed as additional activities as possible improvements.

For each Mitigation, you have represented here various details, including:

* The **Name** of the Mitigation, as title.
* The **Control Type** for the Mitigation, as defined by the following table.

|  |  |
| --- | --- |
| Control Type | Description |
| Unknown | Unknown or undefined control type. |
| Preventive | Preventive control, which reduces the probability or impact of the threat event, effectively gaining time to the defenders and forcing the attacker to make more noise as a result of the various attempts to overcome the security control. |
| Detective | Detective control, which allows detecting an attack while it is in progress. |
| Corrective | Corrective control, which allows responding to the attack. |
| Recovery | Recovery control, which allows to recover from the damages occurred from an attack. |
| Deterrent | Deterrent controls, which are mostly there to convince the potential attacker that the cost for her may be higher than acceptable, compared with the potential gain. |
| Other | Other types of Security Controls. |

Table 4 - Definition of the Control Types adopted by the document.

* A **Description** of the Mitigation.
* The list of **Affected Findings**, that contains the list of the Threat Events associated to the Mitigation, described with a table including the following information:
  + The associated **Object** to which the specific Threat Event refers.
  + The **Threat Type** associated to the Threat Event.
  + The **Strength** of the Mitigation, in accordance with the following table.[[5]](#footnote-5)

|  |  |
| --- | --- |
| Strength | Description |
| Maximum | Strength is maximum: it may be enough to mitigate the Threat. |
| Strong | Strength is important, but it may not be enough to address the Threat alone. |
| Average | Strength is average. It significantly mitigates the Threat, but probably not enough. |
| Weak | Low strength. It has some importance for mitigating the Threat. |
| Negligible | Negligible strength. It has no measurable impact on mitigating the Threat. |

Table 5 - Definition of possible Strengths for the Mitigations.

* The **Status** of the Mitigation, in accordance with the following table.

|  |  |
| --- | --- |
| Status Name | Description |
| Existing | The mitigation already exists. |
| Proposed | The mitigation has been proposed. |

Table 6 - Definition of the Status of the Mitigation.

* The **Directives**, which contain information related on how the mitigation should be applied to specific scenarios.

Mitigations are sorted alphabetically.

## Adopt a Geographically Redundant configuration

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | Geographically Redundant configurations of Azure Storage, like Geo-redundant storage and Geo-zone-redundant storage, guarantee availability also in case of region-wide outages.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Storage | A malicious actor may cause a full-scale outage | Strong | Proposed | |

## Adopt active geo-replication

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | Active geo-replication is an Azure SQL Database feature that allows you to create readable secondary databases of individual databases on a server in the same or different data center (region).  Active geo-replication is designed as a business continuity solution that allows the application to perform quick disaster recovery of individual databases in case of a regional disaster or large scale outage. If geo-replication is enabled, the application can initiate failover to a secondary database in a different Azure region. Up to four secondaries are supported in the same or different regions, and the secondaries can also be used for read-only access queries. The failover must be initiated manually by the application or the user. After failover, the new primary has a different connection end point.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/active-geo-replication-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | A malicious actor may cause a full-scale outage | Strong | Proposed | |

## Adopt Azure Application Gateway with WAF

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Azure Web Application Firewall (WAF) on Azure Application Gateway provides centralized protection of your web applications from common exploits and vulnerabilities. Web applications are increasingly targeted by malicious attacks that exploit commonly known vulnerabilities. SQL injection and cross-site scripting are among the most common attacks.  For more information, please refer to https://docs.microsoft.com/en-us/azure/web-application-firewall/ag/ag-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Attacks may go undetected due to lack of monitoring | Average | Existing | |

## Adopt Resource Tokens to get access to data in Cosmos DB

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Resource Tokens allow to provide granular access rights to resources stored in Cosmos DB. They are the recommended approach when you want to provide access to resources in your Cosmos DB account to a client that cannot be trusted with the primary key.  Cosmos DB resource tokens provide a safe alternative that enables clients to read, write, and delete resources in your Cosmos DB account according to the permissions you've granted, and without need for either a primary or read only key.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/secure-access-to-data#resource-tokens. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get/set user data | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | |

## Apply a transparent encryption

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Transparent encryption like TDE for Azure SQL or SSE for Azure Storage encrypt the data at rest automatically.  While this provides some form of protection, the Strength is typically Weak for Azure services, because it mostly addresses a risk that is mostly mitigated: the risk of insiders to the Cloud Provider, getting access to some customer's data. In fact, Cloud Providers typically apply very strict security controls, including physical controls, to prevent malicious insiders to get access to user data. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | Sensitive data may be disclosed due to lack of encryption at rest | Weak | Existing | | [D] Azure Storage | Sensitive data may be disclosed due to lack of encryption at rest | Weak | Existing | | [D] Cosmos DB | Sensitive data may be disclosed due to lack of encryption at rest | Weak | Existing | |

## Apply Client-Side Encryption for Azure Storage

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Various client libraries for Azure Storage, including the Azure Storage for .NET, for Java and for Python, support encrypting data within client applications before uploading to Azure Storage, and decrypting data while downloading to the client. The .NET and Java libraries also support integration with Azure Key Vault for storage account key management.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-client-side-encryption for .NET, https://docs.microsoft.com/en-us/azure/storage/common/storage-client-side-encryption-java for Java and https://docs.microsoft.com/en-us/azure/storage/common/storage-client-side-encryption-python for Python. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Storage | Sensitive data may be disclosed due to lack of encryption at rest | Maximum | Proposed | |

## Apply column-level encryption

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Azure SQL allows to encrypt single columns using symmetric encryption.  While this provides some form of protection, the Strength is typically Weak for Azure SQL, because it mostly addresses a risk that is mostly mitigated: the risk of insiders to the Cloud Provider, getting access to some customer's data. In fact, Cloud Providers typically apply very strict security controls, including physical controls, to prevent malicious insiders to get access to user data. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | Sensitive data may be disclosed due to lack of encryption at rest | Weak | Proposed | |

## Apply SQL Database dynamic data masking

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Azure SQL Database, Azure SQL Managed Instance, and Azure Synapse Analytics support dynamic data masking. Dynamic data masking limits sensitive data exposure by masking it to non-privileged users.  Dynamic data masking helps prevent unauthorized access to sensitive data by enabling customers to designate how much of the sensitive data to reveal with minimal impact on the application layer. It’s a policy-based security feature that hides the sensitive data in the result set of a query over designated database fields, while the data in the database is not changed.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/dynamic-data-masking-overview.  An example of how to configure this is in https://docs.microsoft.com/en-us/azure/azure-sql/database/dynamic-data-masking-configure-portal. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get processed data | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | |

## Apply strong channel encryption

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Channel encryption allows to protect data in transit so that it is harder to access and disclose sensitive data.  It is important to ensure that the encryption is strong enough for the purposes of the solution, because it may not be enough to simply enable it. For more information about how to get strong channel encryption, please refer to https://cheatsheetseries.owasp.org/cheatsheets/Transport\_Layer\_Protection\_Cheat\_Sheet.html.  To validate the strength of the channel encryption, it is possible to use tools like Qualys SSLLabs's SSLTest, which is freely available from https://www.ssllabs.com/ssltest/,  Managed Cloud Services typically have strong channel encryption enforced, but it is less strong than it could be, because the need for Security is mediated by the need to support reasonably old devices. For this reason, most services allow to configure the minimum protocol allowed. The recommendation is to use the strongest available, which at the time of writing is TLS 1.2. When available, TLS 1.3 would become the preferred protocol, provided that it is supported by the targeted clients. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Forward request | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Forward request to BL | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Get processed data | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Get secrets | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Get user-provided file | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Get/set user data | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Save processed data | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Send request | Sensitive data may be disclosed in transit | Strong | Existing | | [F] Store a file to Azure Storage | Sensitive data may be disclosed in transit | Strong | Existing | |

## Apply the least privilege principle

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | In information security, computer science, and other fields, the principle of least privilege (PoLP), also known as the principle of minimal privilege or the principle of least authority, requires that in a particular abstraction layer of a computing environment, every module (such as a process, a user, or a program, depending on the subject) must be able to access only the information and resources that are necessary for its legitimate purpose.  Source: https://en.wikipedia.org/wiki/Principle\_of\_least\_privilege. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get processed data | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | | [F] Get processed data | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | | [F] Get secrets | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | | [F] Get secrets | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | | [F] Get user-provided file | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | | [F] Get user-provided file | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | | [F] Get/set user data | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | | [F] Get/set user data | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | | [F] Save processed data | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | | [F] Save processed data | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | | [F] Store a file to Azure Storage | High-privileged accounts can be exploited to perform malicious actions | Strong | Proposed | | [F] Store a file to Azure Storage | Sensitive data stored may be disclosed due to weak authorization | Strong | Proposed | |

## Azure Private Link for Azure SQL Database and Azure Synapse Analytics

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Private Link allows you to connect to various PaaS services in Azure via a private endpoint. For a list of PaaS services that support Private Link functionality, go to the Private Link Documentation page. A private endpoint is a private IP address within a specific VNet and subnet.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/private-endpoint-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | An internal resource may be accessed by external actors | Average | Proposed | |

## Azure SQL Database and Azure Synapse Analytics network access controls

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | When you create a logical SQL server from the Azure portal for Azure SQL Database and Azure Synapse Analytics, the result is a public endpoint in the format, yourservername.database.windows.net.  You can use the following network access controls to selectively allow access to a database via the public endpoint:  Allow Azure Services: When set to ON, other resources within the Azure boundary, for example an Azure Virtual Machine, can access SQL Database  IP firewall rules: Use this feature to explicitly allow connections from a specific IP address, for example from on-premises machines.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/network-access-controls-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | An internal resource may be accessed by external actors | Average | Proposed | |

## Configure access to Azure Cosmos DB from virtual networks (VNet)

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | You can configure the Azure Cosmos account to allow access only from a specific subnet of virtual network (VNet). By enabling Service endpoint to access Azure Cosmos DB on the subnet within a virtual network, the traffic from that subnet is sent to Azure Cosmos DB with the identity of the subnet and Virtual Network. Once the Azure Cosmos DB service endpoint is enabled, you can limit access to the subnet by adding it to your Azure Cosmos account.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-configure-vnet-service-endpoint. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Cosmos DB | An internal resource may be accessed by external actors | Average | Proposed | |

## Configure Always Encrypted by using Azure Key Vault

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Always Encrypted is a data encryption technology that helps protect sensitive data at rest on the server, during movement between client and server, and while the data is in use. Always Encrypted ensures that sensitive data never appears as plaintext inside the database system. After you configure data encryption, only client applications or app servers that have access to the keys can access plaintext data.  For more information, please refer to https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | Sensitive data may be disclosed due to lack of encryption at rest | Maximum | Proposed | |

## Configure Azure Defender for App Service

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | Azure Defender for App Service uses the scale of the cloud to identify attacks targeting applications running over App Service. Attackers probe web applications to find and exploit weaknesses. Before being routed to specific environments, requests to applications running in Azure go through several gateways, where they're inspected and logged. This data is then used to identify exploits and attackers, and to learn new patterns that will be used later.  By using the visibility that Azure has as a cloud provider, Security Center analyzes App Service internal logs to identify attack methodology on multiple targets. For example, methodology includes widespread scanning and distributed attacks. This type of attack typically comes from a small subset of IPs, and shows patterns of crawling to similar endpoints on multiple hosts. The attacks are searching for a vulnerable page or plugin, and can't be identified from the standpoint of a single host.  For more information, please refer to https://docs.microsoft.com/en-us/azure/security-center/defender-for-app-service-introduction. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] Business Logic Web API | Misconfigurations may allow compromise | Average | Proposed | | [P] Data processing Azure Function | Misconfigurations may allow compromise | Average | Proposed | | [F] Execute request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [P] Static Content Web App | Misconfigurations may allow compromise | Average | Proposed | |

## Configure Azure Defender for Key Vault

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | Azure Key Vault is a cloud service that safeguards encryption keys and secrets like certificates, connection strings, and passwords.  Azure Defender detects unusual and potentially harmful attempts to access or exploit Key Vault accounts. This layer of protection allows you to address threats without being a security expert, and without the need to manage third-party security monitoring systems.  When anomalous activities occur, Azure Defender shows alerts and optionally sends them via email to relevant members of your organization. These alerts include the details of the suspicious activity and recommendations on how to investigate and remediate threats.  For more information, please refer to https://docs.microsoft.com/en-us/azure/security-center/defender-for-key-vault-introduction. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Key Vault | A malicious actor may compromise the content of the storage | Average | Proposed | | [D] Azure Key Vault | Misconfigurations may allow compromise | Strong | Proposed | | [F] Get secrets | Attacks may go undetected due to lack of monitoring | Strong | Proposed | |

## Configure Azure Defender for SQL

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | Azure Defender for SQL include functionality for identifying and mitigating potential database vulnerabilities and detecting anomalous activities that could indicate threats to your databases:  - Vulnerability assessment - The scanning service to discover, track, and help you remediate potential database vulnerabilities. Assessment scans provide an overview of your SQL machines' security state, and details of any security findings.  - Advanced threat protection - The detection service that continuously monitors your SQL servers for threats such as SQL injection, brute-force attacks, and privilege abuse. This service provides action-oriented security alerts in Azure Security Center with details of the suspicious activity, guidance on how to mitigate to the threats, and options for continuing your investigations with Azure Sentinel.  For more information, please refer to https://docs.microsoft.com/en-us/azure/security-center/defender-for-sql-introduction. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | An internal resource may be accessed by external actors | Average | Proposed | | [D] Azure SQL | A malicious actor may compromise the content of the storage | Average | Proposed | | [D] Azure SQL | Misconfigurations may allow compromise | Strong | Proposed | | [F] Get processed data | Attacks may go undetected due to lack of monitoring | Strong | Proposed | | [F] Save processed data | Attacks may go undetected due to lack of monitoring | Strong | Proposed | |

## Configure Azure Defender for Storage

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | Azure Defender for Storage provides an additional layer of security intelligence that detects unusual and potentially harmful attempts to access or exploit storage accounts. This layer of protection allows you to address threats without being a security expert or managing security monitoring systems.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/azure-defender-storage-configure. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Storage | A malicious actor may compromise the content of the storage | Average | Proposed | | [D] Azure Storage | Misconfigurations may allow compromise | Strong | Proposed | | [F] Get user-provided file | Attacks may go undetected due to lack of monitoring | Strong | Proposed | | [F] Store a file to Azure Storage | Attacks may go undetected due to lack of monitoring | Strong | Proposed | |

## Configure Azure Private Link for an Azure Cosmos account

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | By using Azure Private Link, you can connect to an Azure Cosmos account via a private endpoint. The private endpoint is a set of private IP addresses in a subnet within your virtual network. You can then limit access to an Azure Cosmos account over private IP addresses. When Private Link is combined with restricted NSG policies, it helps reduce the risk of data exfiltration. To learn more about private endpoints, see the Azure Private Link article.  An example about how to achieve this is in https://docs.microsoft.com/en-us/azure/private-link/tutorial-private-endpoint-cosmosdb-portal.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-configure-private-endpoints. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Cosmos DB | An internal resource may be accessed by external actors | Average | Proposed | |

## Configure customer-managed keys for your Azure Cosmos account with Azure Key Vault

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Data stored in your Azure Cosmos account is automatically and seamlessly encrypted with keys managed by Microsoft (service-managed keys). Optionally, you can choose to add a second layer of encryption with keys you manage (customer-managed keys).  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-setup-cmk. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Cosmos DB | Sensitive data may be disclosed due to lack of encryption at rest | Weak | Proposed | |

## Configure IP firewall in Azure Cosmos DB

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | To secure the data stored in your account, Azure Cosmos DB supports a secret based authorization model that utilizes a strong Hash-based Message Authentication Code (HMAC). Additionally, Azure Cosmos DB supports IP-based access controls for inbound firewall support. This model is similar to the firewall rules of a traditional database system and provides an additional level of security to your account. With firewalls, you can configure your Azure Cosmos account to be accessible only from an approved set of machines and/or cloud services. Access to data stored in your Azure Cosmos database from these approved sets of machines and services will still require the caller to present a valid authorization token.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-configure-firewall. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Cosmos DB | An internal resource may be accessed by external actors | Average | Proposed | |

## Configure the allowed CipherSuites

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Require that clients use a more secure set of algorithms for the channel encryption by configuring the SSL Policy in Azure Application Gateway.  For more information, please refer to https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-configure-ssl-policy-powershell.  Use Qualys SSLLabs SSLTest (https://www.ssllabs.com/ssltest/) to identify the ciphersuites that should be removed, and validate the configuration after it has been applied. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Forward request to BL | Sensitive data may be disclosed in transit | Average | Proposed | |

## Configure the minimum required version of Transport Layer Security (TLS)

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Require that clients use a more secure version of the channel encryption protocol by configuring the minimum version of TLS for that account. At the time of writing, it is recommended to allow only TLS 1.2 or higher.  For more information related to Azure Storage, please refer to https://docs.microsoft.com/en-us/azure/storage/common/transport-layer-security-configure-minimum-version.  For more information related to App Services (Web Apps, Web Apis and Azure Functions), please refer to https://docs.microsoft.com/en-us/azure/app-service/configure-ssl-bindings#enforce-tls-versions. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Sensitive data may be disclosed in transit | Average | Proposed | | [F] Forward request | Sensitive data may be disclosed in transit | Average | Proposed | | [F] Forward request to BL | Sensitive data may be disclosed in transit | Average | Proposed | | [F] Get user-provided file | Sensitive data may be disclosed in transit | Average | Proposed | | [F] Store a file to Azure Storage | Sensitive data may be disclosed in transit | Average | Proposed | |

## Create custom events to detect solution-specific security attacks

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | Identify situations that may be at risk and, for each one of them, identify potential events to be raised to detect potential attacks.  Those events should be created in Azure Monitor, because it represents a unified view of the events on Azure. You may use Azure App Insights to raise such custom events.  For more information on App Insights, please refer to https://docs.microsoft.com/en-us/azure/azure-monitor/app/app-insights-overview.  For more information on how to raise custom events, please refer to https://docs.microsoft.com/en-us/azure/azure-monitor/app/custom-operations-tracking. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request | Attacks may go undetected due to lack of monitoring | Average | Proposed | |

## Define and enforce actions for each Alert

|  |  |
| --- | --- |
| Control Type | Corrective |
| Description | Who operates the solution needs to know exactly what to do when an Alert is raised, because security events may require a prompt and precise answer to avoid making things worse.  Those procedures must be enforced and tried with regular simulations. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request to BL | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get processed data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get secrets | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get user-provided file | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get/set user data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Save processed data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Send request | Malicious content may be sent to the solution | Average | Proposed | | [F] Send request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Store a file to Azure Storage | Attacks may go undetected due to lack of monitoring | Average | Proposed | |

## Enable all logging features in Azure Services

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | All Azure Services provide logging features out of the box. Enable all of them. If there is the need to selectively enable logging for any reason, ensure that security events are still raised. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request to BL | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get processed data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get secrets | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get user-provided file | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get/set user data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Save processed data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Send request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Store a file to Azure Storage | Attacks may go undetected due to lack of monitoring | Average | Proposed | |

## Enable soft delete for AKV

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Soft-delete allows you to recover deleted vaults and vault objects.  For more information, please refer to https://docs.microsoft.com/en-us/azure/key-vault/general/soft-delete-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Key Vault | A malicious actor may compromise the content of the storage | Average | Proposed | |

## Enable soft delete for blobs

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Soft delete for blobs protects your data from being accidentally or erroneously modified or deleted. When soft delete for blobs is enabled for a storage account, blobs, blob versions, and snapshots in that storage account may be recovered after they are deleted, within a retention period that you specify.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/blobs/soft-delete-blob-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Storage | A malicious actor may compromise the content of the storage | Average | Proposed | |

## Enable the Secure transfer required option on all of your storage accounts

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | When you enable the Secure transfer required option, all requests made against the storage account must take place over secure connections. Any requests made over HTTP will fail.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-require-secure-transfer. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Sensitive data may be disclosed in transit | Average | Proposed | | [F] Store a file to Azure Storage | Sensitive data may be disclosed in transit | Average | Proposed | |

## Ensure Azure Key Vault availability and redundancy

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | Azure Key Vault features multiple layers of redundancy to make sure that your keys and secrets remain available to your application even if individual components of the service fail.  The contents of your key vault are replicated within the region and to a secondary region at least 150 miles away, but within the same geography to maintain high durability of your keys and secrets. For details about specific region pairs, see Azure paired regions. The exception to the paired regions model is Brazil South, which allows only the option to keep data resident within Brazil South. Brazil South uses locally redundant storage (LRS) to replicate your data three times within the single location/region.  For more information, please refer to https://docs.microsoft.com/en-us/azure/key-vault/general/disaster-recovery-guidance. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Key Vault | A malicious actor may cause a full-scale outage | Strong | Existing | |

## Have a revocation plan in place

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | If a SAS is compromised, you will want to revoke that SAS as soon as possible. To revoke a user delegation SAS, revoke the user delegation key to quickly invalidate all signatures associated with that key. To revoke a service SAS that is associated with a stored access policy, you can delete the stored access policy, rename the policy, or change its expiry time to a time that is in the past.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Implement high-availability for Azure SQL

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | The goal of the high availability architecture in Azure SQL Database and SQL Managed Instance is to guarantee that your database is up and running minimum of 99.99% of time (For more information regarding specific SLA for different tiers, Please refer SLA for Azure SQL Database and SQL Managed Instance), without worrying about the impact of maintenance operations and outages. Azure automatically handles critical servicing tasks, such as patching, backups, Windows and Azure SQL upgrades, as well as unplanned events such as underlying hardware, software, or network failures. When the underlying database in Azure SQL Database is patched or fails over, the downtime is not noticeable if you employ retry logic in your app. SQL Database and SQL Managed Instance can quickly recover even in the most critical circumstances ensuring that your data is always available.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/high-availability-sla.  A possible approach is to implement the auto-failover groups feature, which allows you to manage the replication and failover of a group of databases on a server or all databases in a managed instance to another region. It is a declarative abstraction on top of the existing active geo-replication feature, designed to simplify deployment and management of geo-replicated databases at scale. You can initiate failover manually or you can delegate it to the Azure service based on a user-defined policy. The latter option allows you to automatically recover multiple related databases in a secondary region after a catastrophic failure or other unplanned event that results in full or partial loss of the SQL Database or SQL Managed Instance availability in the primary region. A failover group can include one or multiple databases, typically used by the same application. Additionally, you can use the readable secondary databases to offload read-only query workloads. Because auto-failover groups involve multiple databases, these databases must be configured on the primary server. Auto-failover groups support replication of all databases in the group to only one secondary server or instance in a different region.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/auto-failover-group-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | A malicious actor may cause a full-scale outage | Strong | Proposed | |

## Implement multi-region high-availability for Cosmos DB

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | Azure Cosmos DB provides high availability in two primary ways. First, Azure Cosmos DB replicates data across regions configured within a Cosmos account. Second, Azure Cosmos DB maintains 4 replicas of data within a region.  Azure Cosmos DB is a globally distributed database service and is a foundational service in Azure. By default, is available in all regions where Azure is available. You can associate any number of Azure regions with your Azure Cosmos account and your data is automatically and transparently replicated.  As a globally distributed database, Azure Cosmos DB provides comprehensive SLAs that encompass throughput, latency at the 99th percentile, consistency, and high availability. The table below shows the guarantees for high availability provided by Azure Cosmos DB for single and multi-region accounts.  Consider adopting multi-region writes with Availability zones.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/high-availability. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Cosmos DB | A malicious actor may cause a full-scale outage | Strong | Proposed | |

## Limit duration of Service SAS Tokens not associated with a stored access policy

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | A service SAS that is not associated with a stored access policy cannot be revoked. For this reason, limiting the expiry time so that the SAS is valid for one hour or less is recommended.  Source: https://docs.microsoft.com/en-us/azure/storage/blobs/security-recommendations. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Limit exposure to inbound network traffic

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | It is possible to use various approaches to control access to App Service and Azure Functions.  Azure App Service on Windows lets you define a list of IP addresses that are allowed to access your app. The allowed list can include individual IP addresses or a range of IP addresses defined by a subnet mask.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/app-service-ip-restrictions.  App Service Environment uses Network Security Groups (NSGs) to restrict network access and control the number of exposed endpoints.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/environment/app-service-app-service-environment-control-inbound-traffic.  For Azure Functions, you can use Inbound Access Restrictions to define a priority-ordered list of IP addresses that are allowed or denied access to your app. The list can include IPv4 and IPv6 addresses, or specific virtual network subnets using service endpoints. When there are one or more entries, an implicit "deny all" exists at the end of the list. IP restrictions work with all function-hosting options.  Access restrictions are available in the Premium, Consumption, and App Service.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/app-service-ip-restrictions. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] Business Logic Web API | An internal resource may be accessed by external actors | Average | Proposed | | [P] Static Content Web App | An internal resource may be accessed by external actors | Average | Proposed | |

## Limit network exposure for Azure Key Vault

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | The virtual network service endpoints for Azure Key Vault allow you to restrict access to a specified virtual network. The endpoints also allow you to restrict access to a list of IPv4 (internet protocol version 4) address ranges. Any user connecting to your key vault from outside those sources is denied access.  There is one important exception to this restriction. If a user has opted-in to allow trusted Microsoft services, connections from those services are let through the firewall. For example, these services include Office 365 Exchange Online, Office 365 SharePoint Online, Azure compute, Azure Resource Manager, and Azure Backup. Such users still need to present a valid Azure Active Directory token, and must have permissions (configured as access policies) to perform the requested operation. For more information, see Virtual network service endpoints.  For more information, please refer to https://docs.microsoft.com/en-us/azure/key-vault/general/overview-vnet-service-endpoints and https://docs.microsoft.com/en-us/azure/key-vault/general/network-security. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Key Vault | An internal resource may be accessed by external actors | Average | Proposed | |

## Limit shared access signature (SAS) tokens to HTTPS connections only

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Requiring HTTPS when a client uses a SAS token to access blob data helps to minimize the risk of eavesdropping.  For more information, refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Sensitive data may be disclosed in transit | Average | Proposed | | [F] Store a file to Azure Storage | Sensitive data may be disclosed in transit | Average | Proposed | |

## Lock resources to prevent unexpected changes

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | As an administrator, you may need to lock a subscription, resource group, or resource to prevent other users in your organization from accidentally deleting or modifying critical resources. You can set the lock level to CanNotDelete or ReadOnly. In the portal, the locks are called Delete and Read-only respectively.  CanNotDelete means authorized users can still read and modify a resource, but they can't delete the resource.  ReadOnly means authorized users can read a resource, but they can't delete or update the resource. Applying this lock is similar to restricting all authorized users to the permissions granted by the Reader role.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-resource-manager/management/lock-resources. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Storage | A malicious actor may compromise the content of the storage | Average | Proposed | |

## Raise alerts when significant events occur

|  |  |
| --- | --- |
| Control Type | Detective |
| Description | Raise alerts in Azure Monitor when the abnormal event is verified more than once in a short time span, or when a single event occurs, if it is particularly significant.  For more information about Alerts in Azure Monitor, please refer to https://docs.microsoft.com/en-us/azure/azure-monitor/platform/alerts-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Forward request to BL | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get processed data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get secrets | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get user-provided file | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Get/set user data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Save processed data | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Send request | Malicious content may be sent to the solution | Average | Proposed | | [F] Send request | Attacks may go undetected due to lack of monitoring | Average | Proposed | | [F] Store a file to Azure Storage | Attacks may go undetected due to lack of monitoring | Average | Proposed | |

## Recover the Cosmos DB

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | Azure Cosmos DB automatically takes backups of your data at regular intervals. The automatic backups are taken without affecting the performance or availability of the database operations. All the backups are stored separately in a storage service, and those backups are globally replicated for resiliency against regional disasters. The automatic backups are helpful in scenarios when you accidentally delete or update your Azure Cosmos account, database, or container and later require the data recovery.  Consider changing the backup interval and retention period as required by your Organization, as discussed in https://docs.microsoft.com/en-us/azure/cosmos-db/online-backup-and-restore#configure-backup-interval-retention.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/online-backup-and-restore. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Cosmos DB | A malicious actor may compromise the content of the storage | Average | Proposed | |

## Recover the database using Point in Time Restore

|  |  |
| --- | --- |
| Control Type | Recovery |
| Description | You can restore a standalone, pooled, or instance database to an earlier point in time by using the Azure portal, PowerShell, or the REST API. The request can specify any service tier or compute size for the restored database. Ensure that you have sufficient resources on the server to which you are restoring the database.  When complete, the restore creates a new database on the same server as the original database.  You generally restore a database to an earlier point for recovery purposes. You can treat the restored database as a replacement for the original database or use it as a data source to update the original database.  For more information, please refert to https://docs.microsoft.com/en-us/azure/azure-sql/database/recovery-using-backups#point-in-time-restore. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | A malicious actor may compromise the content of the storage | Average | Proposed | |

## Redirect HTTP to HTTPS

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | By default, clients can connect to web apps by using both HTTP or HTTPS. We recommend redirecting HTTP to HTTPs because HTTPS uses the SSL/TLS protocol to provide a secure connection, which is both encrypted and authenticated.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/configure-ssl-bindings#enforce-https. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Forward request | Sensitive data may be disclosed in transit | Average | Proposed | |

## Regenerate your account keys periodically

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Rotating the account keys periodically reduces the risk of exposing your data to malicious actors. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Reject requests not protected by TLS

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Considering that all clients are supposed to use TLS, requests that are not encrypted are most probably sent by a malicious actor and thus must be rejected. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Execute request | Sensitive data may be disclosed in transit | Average | Proposed | |

## Restrict service access to data operations in Azure Cosmos DB

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | In Azure Cosmos DB, there are two ways to authenticate interactions with the database service:  - using your Azure Active Directory identity when interacting with the Azure portal,  - using Azure Cosmos DB keys or resource tokens when issuing calls from APIs and SDKs.  Each authentication method gives access to different sets of operations, with some overlap:  - Azure AD identities can be used for Account Management and Resource Management.  - Azure Cosmos DB keys and resource tokens can be used for Resource Management and Data Operations.  Services which need to perform Data Operations are therefore granted also Resource Management rights, typically unnecessarily. Consider to restrict the right by disallowing the execution of non-data operations with keys. You can achieve this by restricting these operations to Azure Resource Manager calls only.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-restrict-user-data. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get/set user data | High-privileged accounts can be exploited to perform malicious actions | Average | Proposed | |

## Secure application configuration data using Environment Variables

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Don't store application secrets, such as database credentials, API tokens, or private keys in your code or configuration files. The commonly accepted approach is to access them as environment variables using the standard pattern in your language of choice. In Azure App Service, you can define environment variables through app settings and connection strings. App settings and connection strings are stored encrypted in Azure. The app settings are decrypted only before being injected into your app's process memory when the app starts. The encryption keys are rotated regularly.  For more information, please refer to https://wikipedia.org/wiki/Environment\_variable and https://docs.microsoft.com/en-us/azure/app-service/configure-common. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] Business Logic Web API | Secrets may be disclosed due to weak protection | Average | Proposed | | [P] Static Content Web App | Secrets may be disclosed due to weak protection | Average | Proposed | |

## Secure Azure Cosmos keys using Azure Key Vault

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Azure Cosmos key allows read or read/write access. If they are sent in query string or if they are stored unsafely, then they may be disclosed and used to access the data.  To prevent this issue, it is possible to use Azure Key Vault. The idea is to store the credentials in Key Vault and then using a Managed Identity to access them.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/access-secrets-from-keyvault. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get/set user data | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Secure your account access keys with Azure Key Vault

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Microsoft recommends using Azure AD to authorize requests to Azure Storage. However, if you must use Shared Key authorization, then secure your account keys with Azure Key Vault. You can retrieve the keys from the key vault at runtime, instead of saving them with your application.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/blobs/security-recommendations#identity-and-access-management. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Store business-critical blob data with immutable storage

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Immutable storage for Azure Blob storage enables users to store business-critical data objects in a WORM (Write Once, Read Many) state. This state makes the data non-erasable and non-modifiable for a user-specified interval. For the duration of the retention interval, blobs can be created and read, but cannot be modified or deleted. Immutable storage is available for general-purpose v1, general-purpose v2, BlobStorage, and BlockBlobStorage accounts in all Azure regions.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-immutable-storage. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure Storage | A malicious actor may compromise the content of the storage | Average | Proposed | |

## Store secrets in Azure Key Vault

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Don't store application secrets, such as database credentials, API tokens, or private keys in your code or configuration files.  A good approach in many situations is to use Azure Key Vault for advanced secrets management, provided that the Key Vault is accessed with a Managed Identity. This approach allows more granular control over the secrets and stronger protection than other approaches, like the Environment Variables. In fact, in most situations accessing Key Vault without Managed Identities represents a risk and shall be avoided. Consider that in the worst case, when you have a single Key Vault shared by the various subsystems of your solution and all those subsystems use credentials stored unsafely to access it, the introduction of Key Vault would actually negatively impact the security of the solution, instead of improving it. This is because any of those subsystems could be compromise and as a result it all the secrets managed by the solution would be compromised.  In any case, the adoption of Managed Identities to have direct access to the resources is the preferred approach, because it removes the need to store secrets anywhere. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] API Management | Secrets may be disclosed due to weak protection | Strong | Proposed | | [P] Application Gateway | Secrets may be disclosed due to weak protection | Strong | Proposed | | [P] Business Logic Web API | Secrets may be disclosed due to weak protection | Strong | Proposed | | [P] Data processing Azure Function | Secrets may be disclosed due to weak protection | Strong | Proposed | | [P] Static Content Web App | Secrets may be disclosed due to weak protection | Strong | Proposed | |

## Use Azure AD Service Accounts

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Azure AD Service Accounts are normal Azure AD Accounts assigned to services hosted on Azure, to access other resources. Those accounts are not managed, thus their password must be regularly changed by the owner of the service. Still, they should be preferred over mechanisms like SAS Tokens.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-auth-aad. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get processed data | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Get secrets | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Save processed data | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Use FTPS

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | App Service supports both FTP and FTPS for deploying your files. Use FTPS instead of FTP when possible. When one or both of these protocols are not in use, you should disable them.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/deploy-ftp#enforce-ftps. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] Business Logic Web API | Solution may be tampered due to weak protection | Average | Proposed | | [P] Data processing Azure Function | Solution may be tampered due to weak protection | Average | Proposed | | [P] Static Content Web App | Solution may be tampered due to weak protection | Average | Proposed | |

## Use Managed Identities

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Managed identities eliminate the need for developers having to manage credentials by providing an identity for the Azure resource in Azure AD and using it to obtain Azure Active Directory (Azure AD) tokens. This also helps accessing Azure Key Vault where developers can store credentials in a secure manner. Managed identities for Azure resources solves this problem by providing Azure services with an automatically managed identity in Azure AD.  For more information, please refer to https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/overview.  The list of Azure services supporting Managed Identities is available in https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/services-support-managed-identities. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] API Management | An internal resource may be accessed by external actors | Strong | Proposed | | [P] API Management | Secrets may be disclosed due to weak protection | Maximum | Proposed | | [P] Business Logic Web API | An internal resource may be accessed by external actors | Strong | Proposed | | [P] Business Logic Web API | Secrets may be disclosed due to weak protection | Maximum | Proposed | | [P] Data processing Azure Function | An internal resource may be accessed by external actors | Strong | Proposed | | [P] Data processing Azure Function | Secrets may be disclosed due to weak protection | Maximum | Proposed | | [F] Get processed data | Credentials for the resource may be stolen and used to access it | Strong | Proposed | | [F] Get secrets | Credentials for the resource may be stolen and used to access it | Strong | Proposed | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Strong | Proposed | | [F] Save processed data | Credentials for the resource may be stolen and used to access it | Strong | Proposed | | [P] Static Content Web App | An internal resource may be accessed by external actors | Strong | Proposed | | [P] Static Content Web App | Secrets may be disclosed due to weak protection | Maximum | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Strong | Proposed | |

## Use separate Key Vaults

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | The recommended approach is to use a vault per application per environment (Development, Pre-Production and Production). This helps you not share secrets across environments and also reduces the threat in case of a breach. By extent, if the solution is complex and has a significant sensitivity, it may be best to consider introducing multiple Key Vaults, for instance one for the Front-End and one for the Back-End components.  For more information, please refer to https://docs.microsoft.com/en-us/azure/key-vault/general/best-practices#use-separate-key-vault. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] API Management | Secrets may be disclosed due to weak protection | Average | Proposed | | [P] Application Gateway | Secrets may be disclosed due to weak protection | Average | Proposed | | [P] Business Logic Web API | Secrets may be disclosed due to weak protection | Average | Proposed | | [P] Data processing Azure Function | Secrets may be disclosed due to weak protection | Average | Proposed | | [F] Get secrets | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [P] Static Content Web App | Secrets may be disclosed due to weak protection | Average | Proposed | |

## Use system-assigned managed identities to access Azure Cosmos DB data

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Implement a robust, key rotation agnostic solution to access Azure Cosmos DB keys by using Managed Identities.  For more information, please refer to https://docs.microsoft.com/en-us/azure/cosmos-db/managed-identity-based-authentication. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get/set user data | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Use the isolated pricing tier

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Except for the isolated pricing tier, all tiers run your apps on the shared network infrastructure in Azure App Service. The isolated tier gives you complete network isolation by running your apps inside a dedicated App Service environment. An App Service environment runs in your own instance of Azure Virtual Network.  For more information, please refer to https://docs.microsoft.com/en-us/azure/app-service/environment/intro. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [P] Business Logic Web API | An internal resource may be accessed by external actors | Strong | Proposed | | [P] Data processing Azure Function | An internal resource may be accessed by external actors | Strong | Proposed | | [P] Static Content Web App | An internal resource may be accessed by external actors | Strong | Proposed | |

## Use user delegation SAS Tokens

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | A user delegation SAS is secured with Azure Active Directory (Azure AD) credentials and also by the permissions specified for the SAS. A user delegation SAS is analogous to a service SAS in terms of its scope and function, but offers security benefits over the service SAS.  For more information, please refer to https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Get user-provided file | Credentials for the resource may be stolen and used to access it | Average | Proposed | | [F] Store a file to Azure Storage | Credentials for the resource may be stolen and used to access it | Average | Proposed | |

## Use virtual network service endpoints and rules for servers in Azure SQL Database

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Virtual network rules are one firewall security feature that controls whether the server for your databases and elastic pools in Azure SQL Database or for your databases in Azure Synapse accepts communications that are sent from particular subnets in virtual networks.  For more information, please refer to https://docs.microsoft.com/en-us/azure/azure-sql/database/vnet-service-endpoint-rule-overview. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [D] Azure SQL | An internal resource may be accessed by external actors | Average | Proposed | |

## Validate all input before processing it

|  |  |
| --- | --- |
| Control Type | Preventive |
| Description | Input must be verified before being processed, independently from its source. In fact, in accordance with the Defense in Depth principle, even trusted sources may be spoofed or compromised and thus may send malicious content.  For this reason, all input must be verified based on strict compliancy rules and rejected if not found aligned with the known good.  For more information and guidance, please refer to https://cheatsheetseries.owasp.org/cheatsheets/Input\_Validation\_Cheat\_Sheet.html. |
| Affected Threats | |  |  |  |  | | --- | --- | --- | --- | | Object | Threat | Strength | Status | | [F] Send request | Malicious content may be sent to the solution | Strong | Proposed | |

# The Roadmap

The Threat Model has split the recommended Mitigations to be implemented in three phases:

* Short Term, which represent activities that could be done immediately.
* Mid Term, which represent activities that could be done at a later stage.
* Long Term, which includes activities that are more complex or less urgent than the previous categories, and for this reason may be implemented even later.

This list should not be considered as a prescriptive guidance, but just as a starting point.

Based on this list, it is possible to evaluate the effect of each phase on the Residual Risk. The following chart represents this evaluation.

Figure 4 - The effects of the Roadmap on the Residual Risk.

The previous chart applies different colors to the bars depending on the fact that the Residual Risk may be considered acceptable or not. This is achieved considering an Acceptable Risk corresponding to 0 Critical or High Severity, 10 Medium Severity, 20 Low Severity and 20 Information level Threat Events.

The following table describes the Roadmap by listing the Mitigations selected for each phase.

|  |  |
| --- | --- |
| Mitigation | Related Threats |
| **Short Term** | |
| Azure SQL Database and Azure Synapse Analytics network access controls | Proposed   * [D] Azure SQL: An internal resource may be accessed by external actors |
| Configure IP firewall in Azure Cosmos DB | Proposed   * [D] Cosmos DB: An internal resource may be accessed by external actors |
| Configure the allowed CipherSuites | Proposed   * [F] Forward request to BL: Sensitive data may be disclosed in transit |
| Configure the minimum required version of Transport Layer Security (TLS) | Proposed   * [F] Forward request: Sensitive data may be disclosed in transit * [F] Get user-provided file: Sensitive data may be disclosed in transit * [F] Store a file to Azure Storage: Sensitive data may be disclosed in transit * [F] Forward request to BL: Sensitive data may be disclosed in transit * [F] Execute request: Sensitive data may be disclosed in transit |
| Enable all logging features in Azure Services | Proposed   * [F] Send request: Attacks may go undetected due to lack of monitoring * [F] Forward request: Attacks may go undetected due to lack of monitoring * [F] Get user-provided file: Attacks may go undetected due to lack of monitoring * [F] Store a file to Azure Storage: Attacks may go undetected due to lack of monitoring * [F] Forward request to BL: Attacks may go undetected due to lack of monitoring * [F] Execute request: Attacks may go undetected due to lack of monitoring * [F] Get/set user data: Attacks may go undetected due to lack of monitoring * [F] Save processed data: Attacks may go undetected due to lack of monitoring * [F] Get processed data: Attacks may go undetected due to lack of monitoring * [F] Get secrets: Attacks may go undetected due to lack of monitoring |
| Enable soft delete for AKV | Proposed   * [D] Azure Key Vault: A malicious actor may compromise the content of the storage |
| Enable soft delete for blobs | Proposed   * [D] Azure Storage: A malicious actor may compromise the content of the storage |
| Enable the Secure transfer required option on all of your storage accounts | Proposed   * [F] Get user-provided file: Sensitive data may be disclosed in transit * [F] Store a file to Azure Storage: Sensitive data may be disclosed in transit |
| Limit exposure to inbound network traffic | Proposed   * [P] Static Content Web App: An internal resource may be accessed by external actors * [P] Business Logic Web API: An internal resource may be accessed by external actors |
| Limit shared access signature (SAS) tokens to HTTPS connections only | Proposed   * [F] Get user-provided file: Sensitive data may be disclosed in transit * [F] Store a file to Azure Storage: Sensitive data may be disclosed in transit |
| Lock resources to prevent unexpected changes | Proposed   * [D] Azure Storage: A malicious actor may compromise the content of the storage |
| Redirect HTTP to HTTPS | Proposed   * [F] Forward request: Sensitive data may be disclosed in transit |
| Reject requests not protected by TLS | Proposed   * [F] Execute request: Sensitive data may be disclosed in transit |
| Restrict service access to data operations in Azure Cosmos DB | Proposed   * [F] Get/set user data: High-privileged accounts can be exploited to perform malicious actions |
| Use FTPS | Proposed   * [P] Static Content Web App: Solution may be tampered due to weak protection * [P] Business Logic Web API: Solution may be tampered due to weak protection * [P] Data processing Azure Function: Solution may be tampered due to weak protection |
| Use separate Key Vaults | Proposed   * [P] Static Content Web App: Secrets may be disclosed due to weak protection * [P] Business Logic Web API: Secrets may be disclosed due to weak protection * [P] Application Gateway: Secrets may be disclosed due to weak protection * [P] Data processing Azure Function: Secrets may be disclosed due to weak protection * [P] API Management: Secrets may be disclosed due to weak protection * [F] Get secrets: Credentials for the resource may be stolen and used to access it |
| **Mid Term** | |
| Adopt a Geographically Redundant configuration | Proposed   * [D] Azure Storage: A malicious actor may cause a full-scale outage |
| Adopt active geo-replication | Proposed   * [D] Azure SQL: A malicious actor may cause a full-scale outage |
| Adopt Resource Tokens to get access to data in Cosmos DB | Proposed   * [F] Get/set user data: High-privileged accounts can be exploited to perform malicious actions |
| Apply the least privilege principle | Proposed   * [F] Get user-provided file: High-privileged accounts can be exploited to perform malicious actions * [F] Get user-provided file: Sensitive data stored may be disclosed due to weak authorization * [F] Store a file to Azure Storage: High-privileged accounts can be exploited to perform malicious actions * [F] Store a file to Azure Storage: Sensitive data stored may be disclosed due to weak authorization * [F] Get/set user data: High-privileged accounts can be exploited to perform malicious actions * [F] Get/set user data: Sensitive data stored may be disclosed due to weak authorization * [F] Save processed data: High-privileged accounts can be exploited to perform malicious actions * [F] Save processed data: Sensitive data stored may be disclosed due to weak authorization * [F] Get processed data: High-privileged accounts can be exploited to perform malicious actions * [F] Get processed data: Sensitive data stored may be disclosed due to weak authorization * [F] Get secrets: High-privileged accounts can be exploited to perform malicious actions * [F] Get secrets: Sensitive data stored may be disclosed due to weak authorization |
| Azure Private Link for Azure SQL Database and Azure Synapse Analytics | Proposed   * [D] Azure SQL: An internal resource may be accessed by external actors |
| Configure access to Azure Cosmos DB from virtual networks (VNet) | Proposed   * [D] Cosmos DB: An internal resource may be accessed by external actors |
| Configure Azure Defender for App Service | Proposed   * [P] Static Content Web App: Misconfigurations may allow compromise * [P] Business Logic Web API: Misconfigurations may allow compromise * [P] Data processing Azure Function: Misconfigurations may allow compromise * [F] Forward request: Attacks may go undetected due to lack of monitoring * [F] Execute request: Attacks may go undetected due to lack of monitoring |
| Configure Azure Defender for Key Vault | Proposed   * [D] Azure Key Vault: A malicious actor may compromise the content of the storage * [D] Azure Key Vault: Misconfigurations may allow compromise * [F] Get secrets: Attacks may go undetected due to lack of monitoring |
| Configure Azure Defender for SQL | Proposed   * [D] Azure SQL: An internal resource may be accessed by external actors * [D] Azure SQL: A malicious actor may compromise the content of the storage * [D] Azure SQL: Misconfigurations may allow compromise * [F] Save processed data: Attacks may go undetected due to lack of monitoring * [F] Get processed data: Attacks may go undetected due to lack of monitoring |
| Configure Azure Defender for Storage | Proposed   * [D] Azure Storage: A malicious actor may compromise the content of the storage * [D] Azure Storage: Misconfigurations may allow compromise * [F] Get user-provided file: Attacks may go undetected due to lack of monitoring * [F] Store a file to Azure Storage: Attacks may go undetected due to lack of monitoring |
| Configure Azure Private Link for an Azure Cosmos account | Proposed   * [D] Cosmos DB: An internal resource may be accessed by external actors |
| Configure customer-managed keys for your Azure Cosmos account with Azure Key Vault | Proposed   * [D] Cosmos DB: Sensitive data may be disclosed due to lack of encryption at rest |
| Create custom events to detect solution-specific security attacks | Proposed   * [F] Forward request: Attacks may go undetected due to lack of monitoring * [F] Execute request: Attacks may go undetected due to lack of monitoring |
| Define and enforce actions for each Alert | Proposed   * [F] Send request: Malicious content may be sent to the solution * [F] Send request: Attacks may go undetected due to lack of monitoring * [F] Forward request: Attacks may go undetected due to lack of monitoring * [F] Get user-provided file: Attacks may go undetected due to lack of monitoring * [F] Store a file to Azure Storage: Attacks may go undetected due to lack of monitoring * [F] Forward request to BL: Attacks may go undetected due to lack of monitoring * [F] Execute request: Attacks may go undetected due to lack of monitoring * [F] Get/set user data: Attacks may go undetected due to lack of monitoring * [F] Save processed data: Attacks may go undetected due to lack of monitoring * [F] Get processed data: Attacks may go undetected due to lack of monitoring * [F] Get secrets: Attacks may go undetected due to lack of monitoring |
| Have a revocation plan in place | Proposed   * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it |
| Implement high-availability for Azure SQL | Proposed   * [D] Azure SQL: A malicious actor may cause a full-scale outage |
| Implement multi-region high-availability for Cosmos DB | Proposed   * [D] Cosmos DB: A malicious actor may cause a full-scale outage |
| Limit network exposure for Azure Key Vault | Proposed   * [D] Azure Key Vault: An internal resource may be accessed by external actors |
| Raise alerts when significant events occur | Proposed   * [F] Send request: Malicious content may be sent to the solution * [F] Send request: Attacks may go undetected due to lack of monitoring * [F] Forward request: Attacks may go undetected due to lack of monitoring * [F] Get user-provided file: Attacks may go undetected due to lack of monitoring * [F] Store a file to Azure Storage: Attacks may go undetected due to lack of monitoring * [F] Forward request to BL: Attacks may go undetected due to lack of monitoring * [F] Execute request: Attacks may go undetected due to lack of monitoring * [F] Get/set user data: Attacks may go undetected due to lack of monitoring * [F] Save processed data: Attacks may go undetected due to lack of monitoring * [F] Get processed data: Attacks may go undetected due to lack of monitoring * [F] Get secrets: Attacks may go undetected due to lack of monitoring |
| Recover the database using Point in Time Restore | Proposed   * [D] Azure SQL: A malicious actor may compromise the content of the storage |
| Secure application configuration data using Environment Variables | Proposed   * [P] Static Content Web App: Secrets may be disclosed due to weak protection * [P] Business Logic Web API: Secrets may be disclosed due to weak protection |
| Secure Azure Cosmos keys using Azure Key Vault | Proposed   * [F] Get/set user data: Credentials for the resource may be stolen and used to access it |
| Secure your account access keys with Azure Key Vault | Proposed   * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it |
| Store business-critical blob data with immutable storage | Proposed   * [D] Azure Storage: A malicious actor may compromise the content of the storage |
| Store secrets in Azure Key Vault | Proposed   * [P] Static Content Web App: Secrets may be disclosed due to weak protection * [P] Business Logic Web API: Secrets may be disclosed due to weak protection * [P] Application Gateway: Secrets may be disclosed due to weak protection * [P] Data processing Azure Function: Secrets may be disclosed due to weak protection * [P] API Management: Secrets may be disclosed due to weak protection |
| Use Azure AD Service Accounts | Proposed   * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it * [F] Save processed data: Credentials for the resource may be stolen and used to access it * [F] Get processed data: Credentials for the resource may be stolen and used to access it * [F] Get secrets: Credentials for the resource may be stolen and used to access it |
| Use Managed Identities | Proposed   * [P] Static Content Web App: An internal resource may be accessed by external actors * [P] Static Content Web App: Secrets may be disclosed due to weak protection * [P] Business Logic Web API: An internal resource may be accessed by external actors * [P] Business Logic Web API: Secrets may be disclosed due to weak protection * [P] Data processing Azure Function: An internal resource may be accessed by external actors * [P] Data processing Azure Function: Secrets may be disclosed due to weak protection * [P] API Management: An internal resource may be accessed by external actors * [P] API Management: Secrets may be disclosed due to weak protection * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it * [F] Save processed data: Credentials for the resource may be stolen and used to access it * [F] Get processed data: Credentials for the resource may be stolen and used to access it * [F] Get secrets: Credentials for the resource may be stolen and used to access it |
| Use system-assigned managed identities to access Azure Cosmos DB data | Proposed   * [F] Get/set user data: Credentials for the resource may be stolen and used to access it |
| Use user delegation SAS Tokens | Proposed   * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it |
| Use virtual network service endpoints and rules for servers in Azure SQL Database | Proposed   * [D] Azure SQL: An internal resource may be accessed by external actors |
| Validate all input before processing it | Proposed   * [F] Send request: Malicious content may be sent to the solution |
| **Long Term** | |
| Apply Client-Side Encryption for Azure Storage | Proposed   * [D] Azure Storage: Sensitive data may be disclosed due to lack of encryption at rest |
| Apply SQL Database dynamic data masking | Proposed   * [F] Get processed data: Sensitive data stored may be disclosed due to weak authorization |
| Configure Always Encrypted by using Azure Key Vault | Proposed   * [D] Azure SQL: Sensitive data may be disclosed due to lack of encryption at rest |
| Limit duration of Service SAS Tokens not associated with a stored access policy | Proposed   * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it |
| Regenerate your account keys periodically | Proposed   * [F] Get user-provided file: Credentials for the resource may be stolen and used to access it * [F] Store a file to Azure Storage: Credentials for the resource may be stolen and used to access it |
| Use the isolated pricing tier | Proposed   * [P] Static Content Web App: An internal resource may be accessed by external actors * [P] Business Logic Web API: An internal resource may be accessed by external actors * [P] Data processing Azure Function: An internal resource may be accessed by external actors |

1. OWASP is a not-profit organization and “open community dedicated to enabling organizations to conceive, develop, acquire, operate, and maintain applications that can be trusted”. It has been founded in the United States in December 2001 and has rapidly asserted itself as one of the most important players in the Application Security field. [↑](#footnote-ref-1)
2. Ssee.: <https://www.owasp.org/index.php/Threat_modeling>. [↑](#footnote-ref-2)
3. Please refer to <https://threatsmanager.com/training/intro/basic/short-form/> for a description of the various values. [↑](#footnote-ref-3)
4. The Mitigation Strength is relative to the Threat: in other words, the same Mitigation may have different strengths for different Threats. [↑](#footnote-ref-4)
5. The Mitigation Strength is relative to the Threat: in other words, the same Mitigation may have different strengths for different Threats. [↑](#footnote-ref-5)