Karl Ots

Hardening Azure OpenAI Service for Enterprise Usage

Metropolia University of Applied Sciences

Bachelor of Engineering

Degree Programme in Information and Communication Technology

Bachelor’s Thesis

7 January 2024

Abstract

Author:	Karl Ots
Title:	Hardening Azure OpenAI Service for Enterprise Usage
Number of Pages:	38 pages + 2 appendices
Date:	7 January 2024
Degree:	Bachelor of Engineering
Degree Programme:	Degree Programme in Information and Communication Technology
Professional Major:	Information and Communication Technologies
Supervisors:	Janne Salonen

Generative artificial intelligence is here and has had a disruptive impact on the industry. In this thesis the author studies secure adoption of generative artificial intelligence services in an enterprise context. Primarily this thesis focuses on Azure OpenAI service.

The thesis presents an overview of the domain, including defining generative artificial intelligence, and providing a summary of early regulatory implications. Enterprise security requirements towards generative AI applications are discussed at length, by applying established enterprise security architecture methodologies. A detailed breakdown of ChatGPT and Azure OpenAI security capabilities is included to compare how the similar, yet different implementations meet the security requirements.

A reference application architecture is designed in Microsoft Azure and an accompanying threat model produced. Implementation of the reference application and security controls to mitigate the identified threats is reported. Finally, a hardened configuration of the reference application architecture is presented.

Keywords: generative artificial intelligence, security, OpenAI

Tiivistelmä

Tekijä:	Karl Ots
Otsikko:	Azure Open AI -palvelun kovettaminen suuryrityskäyttöön
Sivumäärä:	38 sivua + 2 liitettä
Aika:	7.1.2024
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Tieto- ja viestintätekniikan tutkinto-ohjelma
Ammatillinen pääaine:	Tietojenkäsittely ja tietoliikenne
Ohjaajat:	Janne Salonen

Generatiivinen eli tuottava tekoäly on täällä ja se on tullut jäädäkseen. Tässä insinöörityössä tutkitaan tuottavan tekoälyn palveluiden turvalliseen käyttöön suuryritysten kontekstissa. Erityisesti tässä työssä keskitytään Azure OpenAI -palveluun.

Työssä esitellään tuottavan tekoälyn toimintaympäristö aina määritelmästä viimeaikaisen lainsäädännön esittelyyn. Suuryritysten tietoturvavaatimukset esitellään perusteellisesti soveltamalla vakiintuneita kokonaisarkkitehtuurin ja kokonaisturvallisuuden menetelmiä. Työssä verrataan myös samankaltaisten, mutta toteutukseltaan hyvin erilaisten ChatGPT:n ja Azure OpenAI -palveluiden eroja tietoturvan näkökulmasta.

Työssä suunnitellaan ja toteutetaan tietoturvallinen referenssitoteutus tuottavan tekoälyn käyttöön suuryritysten kontekstissa käyttäen Microsoft Azure -palveluita. Suunnitelma uhkamallinnetaan ja tulokset sisällytetään kovennettuun referenssitoteutukseen.

Avainsanat:	tuottava tekoäly, tietoturva, OpenAI

Contents

Appendices

List of Abbreviations

AI	Artificial intelligence.
CDN	Content distribution network.
CMK	Customer managed keys.
DDOS	Distributed denial of service.
IaaS	Infrastructure as a Service.
LLM	Large language model.
NIST	National Institute of Standards and Technology.
OWASP	Open Worldwide Application Security Project
PaaS	Platform as a Service.
RBAC	Role-based access control.
SaaS	Software as a Service.
WAF	Web application firewall.
WORM	Write once, read many.

1 Introduction

Generative artificial intelligence represents a tectonic shift in adoption of digital services. While several risks remain to be addressed, OpenAI's GPT-4 and its predecessor have become the fastest growing application ever, reaching 100 million monthly active users in record time (Reuters, 2023). Remarkably, generative AI is being embraced by not only consumers and fast-moving startups, but also typically slower-moving enterprises. Furthermore, enterprise organizations have spent the last decade stockpiling data in the hopes of unlocking value from it while following the mantra that "data is the new oil". Now they have the data assets in place. But do they have the data governance, secure development, and cloud security capabilities to reap the benefits of generative AI securely?

This paper discusses secure adoption of generative artificial intelligence services in a real-life enterprise context. Specifically, we are focusing on Azure OpenAI service.

Chapter 2 presents an overview of the domain, including defining generative artificial intelligence, and providing a summary of early regulatory implications. Chapter 3 discusses enterprise security requirements towards generative AI applications. A detailed comparison of ChatGPT and Azure OpenAI security capabilities is also included. Chapter 4 presents a reference application architecture, a threat model and security controls to mitigate the identified threats. Finally, chapter 5 presents a hardened configuration of the reference application architecture.

It is worth noting that the statements and solutions presented in this paper are accurate as of the time of writing (end of 2023). The underlying technology, identified risks, and compliance and regulatory landscape are moving extremely fast in this field. Therefore, any gaps discussed might already be addressed by the time of reading.

2 Generative Artificial Intelligence security

In this chapter, we discuss definition and issues of generative artificial intelligence, provide an overview of applicable regulation, and introduce OWASP Top 10 list for LLM applications.

2.1 Generative Artificial Intelligence

Large language models (LLMs) represent a significant advancement in natural language processing. These statistical language models are trained to predict the next word in a partial sentence, using massive amounts of data. With the addition of multi-modal capabilities – ability to process images in addition to text – these so-called generative artificial intelligence models open a plethora of new use cases, previously reserved for highly specialized, narrow artificial intelligence. For example, generative AI can be used for machine translation, text summarization, virtual assistants and for creating hyper-personalized marketing campaigns at scale.

OpenAI’s GPT-4, a widely popular LLM, is a transformer-style (Vaswani, et al., 2017) model which performs well even on tasks that have typically eluded narrow, task-specific AI models. Successful task categories include abstraction, coding, mathematics, medicine, and law. According to (OpenAI, 2023), GPT-4 performs at “human-level” in a variety of academic benchmarks. While several risks remain to be addressed, the success of GPT-4 and its predecessor is remarkable.

From a critical point of view, generative artificial intelligence has been shown to generate incorrect outputs, sometimes referred to as hallucinations. These hallucinations can include incorrect references, statements of fact, mathematical calculations, and even high-level concepts. This issue is made worse by the manner how hallucinations are presented within the outputs. Hallucinations are not distinguishable from factually correct outputs and are often presented in the same manner of confidence, often in between correct outputs.

Identification of hallucinations is a core question in generative AI. According to (Bubeck, et al., 2023) “unrecognized hallucinations can lead to the propagation of errors into downstream uses and influences – including the future training of LLMs”. The authors highlight a need to develop and share best practices to assure the quality of the outputs, especially when it comes to critical applications in medicine, journalism, and transportation. Closed-domain hallucinations (errors made in the context of explicit reference material, such as summarizing documents), are presented as closer to being addressed by the authors. However, open-domain hallucinations continue to pose a challenge, as verifying them requires extensive research outside of the actual prompt-answer session itself.

Additionally, generative AI can be intentionally used for malicious purposes. As noted in (Bubeck, et al., 2023), the use of generative AI by adversarial users can have significant impact on the scope and magnitude of disinformation campaigns. The authors illustrate this by creating emotionally manipulative messages as part of an anti-vaccine disinformation campaign.

As the models have the capability to generate code, they can also be used by adversaries to generate exploits for recently announced vulnerabilities. Having this capability in the hands of adversarial users will significantly influence how enterprises approach their cyber hygiene and incident response functions.

For the purposes of this paper, we are omitting the malicious use cases for generative AI and focusing on securing approved use cases in a semi-private enterprise setting. This approach opens multiple areas of interest, such as:

- AI usage security (user accountability; training and model data governance)

- AI application security (application design; safety systems)

- AI platform security (cloud platform security; model security)

In this paper we are primarily concentrating on **AI platform security**.

2.2 Regulation

The first public sector entities and nation states have started to address generative AI.

In the Blueprint for an AI Bill of Rights (The White House, 2023), the Biden administration defines five principles to build measures that protect the public against threats from artificial intelligence. The principles were announced together with an Executive Order (White House, 2023), defining a number of upcoming regulations. While most of the principles are still positioned as high-level recommendations rather than regulation, they are likely to be closely followed by the technology industry in the United States. The principles include:

- Safe and effective systems (secure software development applied to AI)
- Algorithmic discrimination protections (algorithmic bias)
- Data privacy (agency over how personal data is used)
- Notice and explanation (transparency)
- Human alternatives, consideration, and fallback (opt-out)

In an executive order (State of California, 2023) California defines the need to address critical issues to society in state legislation in state legislation. In addition to ordering a report on identifying suitable use cases for generative AI, the executive order stresses the importance of performing a thorough risk assessment, covering:

- high-risk use cases, such as “consequential decisions affecting access to essential goods and services”.
- risks from bad actors.
- risks to democratic and legal processes.

The executive order also defines a timeline for a risk analysis on threats caused by generative AI to critical infrastructure in the state. The risk analysis is scheduled to be performed by March 2024. Furthermore, the state is planning to publish guidelines addressing “safety, algorithmic discrimination, data privacy, and notice of when materials are generated by GenAI”.

The European Union AI Act is a legal framework several years in the making (Madiega, 2023). As of December 2023, the European Parliament and Commission reached a provisional agreement on the act. However, it will still take considerable time before the text becomes EU law. As illustrated in Figure 1, the act defines different controls based on the risk introduced by each category of artificial intelligence. The regulation is still evolving, but at the time of writing, large language models are understood to be classified as limited risk AI systems, requiring the vendors to meet several additional transparency requirements, and committing to report any serious incidents to the European Commission.

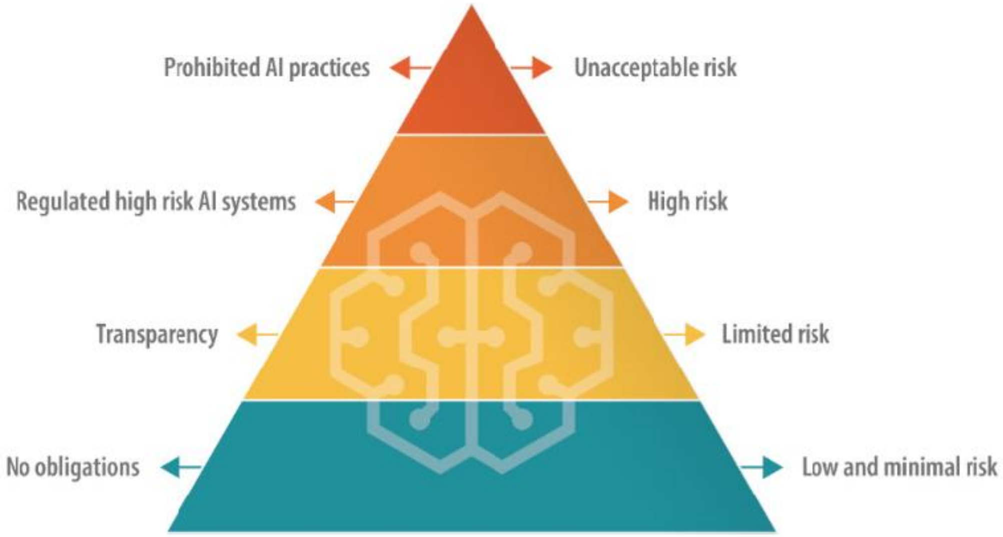


Figure 1: Risk-based approach of AI act

2.3 Ethical principles

There are several safety and ethics principles defined in the responsible AI space. When it comes to implementation and use in the public sector, (Leslie, 2019) defines them as:

- **Fairness**, further defined as fairness of data, design, outcome and implementation. According to this principle, the designers and users of AI systems should pay close attention to mitigating the biases on the outputs and implementations of their models.
- **Accountability**, further defined as answerability (justification of AI-supported decisions should be the responsibility of humans) and auditability (justification of outcomes and demonstration of responsible design).
- **Sustainability**, which calls for sensitivity to the real-world impacts of its use. In practice, this means performing Stakeholder Impact Assessments, templates of which are provided by the author.
- **Safety**, further defined as technical objectives of accuracy, reliability, security, and robustness. Specifically, when listing the risks posed to security and robustness, the author calls out adversarial attacks, such as data poisoning and misdirected reinforcement learning.
- **Transparency**, further defined as a combination of the ability to know understand why the AI system behaved as it did, and the justifiability of the processes that go into the design, implementation, and outcome of the AI system.

Furthermore, the author illustrates potential harm caused by AI systems. In addition to potential for discrimination and privacy violations, they call out:

- Denial of individual autonomy and rights.
- Unjustified, and unreliable outputs of AI systems.
- Reduction in human-to-human connections.

2.4 OWASP TOP 10 for large language model applications

The OWASP Top 10 for large language model applications project (OWASP, 2023) is collecting security guidance to help developers, data scientists and security experts design and build large language model applications and plugins. With version 1.0 released in August 2023 and version 1.1 in October 2023, it’s one of the first publicly available projects that has already produced actionable results in this area. Table 2-1 summarizes the top 10 potential vulnerabilities. The full project documentation complements this by including common examples of each vulnerability, prevention and mitigation strategies, and example attack scenarios. While its primary intention is to support securing AI applications, it can certainly be leveraged in securing AI usage and AI platforms as well.

Table 2-1: Summary of OWASP Top 10 for large language model applications

LLM01: Prompt Injection	Prompt Injection occurs when an adversary manipulates the LLM through specially constructed prompts, causing the model to unknowingly execute arbitrary commands by the attacker.
LLM02: Insecure Output Handling	Insecure Output Handling occurs when a downstream component accepts the LLM output without proper scrutiny, such as passing model output directly to backend systems.
LLM03: Training Data Poisoning	Training Data Poisoning occurs when an adversary manipulates the LLM’s fine-tuning process to introduce vulnerabilities or biases that can compromise the model’s security, effectiveness, or ethical behavior.
LLM04: Model Denial of Service	Model Denial of Service occurs when an adversary interacts with the LLM in a manner that consumes an abnormally high number of resources, resulting in lowered quality of service for all users and higher cloud costs.

LLM05: Supply Chain Vulnerabilities	Supply Chain Vulnerabilities is a category of vulnerabilities that occur when the integrity of either a component of the software supply chain or training data of the LLM is compromised.
LLM06: Sensitive Information Disclosure	Sensitive Information Disclosure occurs when sensitive information such as proprietary algorithms or personally identifiable information of other users of the LLM is revealed to unauthorized parties.
LLM07: Insecure Plugin Design	Insecure Plugin Design is a category of vulnerabilities that occur when adversaries construct malicious requests to the LLM plugin, circumventing the security controls of the model.
LLM08: Excessive Agency	Excessive Agency occurs when damaging actions are performed in response to unexpected outputs from the LLM.
LLM09: Overreliance	Overreliance is a category of vulnerabilities that occur when human decision-making is overly dependent on the LLM.
LLM10: Model Theft	Model Theft occurs when adversaries gain unauthorized access to the LLM and exfiltrate proprietary information of the model.

3 Requirements

In this chapter, we discuss enterprise security requirements and how they apply to cloud services and generative artificial intelligence.

3.1 Enterprise software security requirements

An established enterprise organization is likely to follow an enterprise security architecture methodology, such as Sherwood Applied Business Security Architecture (SABSA). By following such a methodology, the enterprise defines their unique risk appetite, as well as processes for quantifying both risks and effectiveness of controls. Risk appetite is quantified as the level of risk that is still acceptable to the enterprise, in order to meet their business objectives. Within the context of SABSA methodology, these objectives are further defined as a library of **business enablement objectives** and **control objectives**. These objectives make up the core of a security framework in a technologically agnostic manner.

When new software is introduced, the enterprise needs to select **security controls** that are consistent with the existing security architecture and effective in enabling the business objectives. When selecting security controls, an enterprise can assess which controls cause the residual risk rating to cross the desired threshold. Figure 2 illustrates the risk appetite spectrum in control selection.

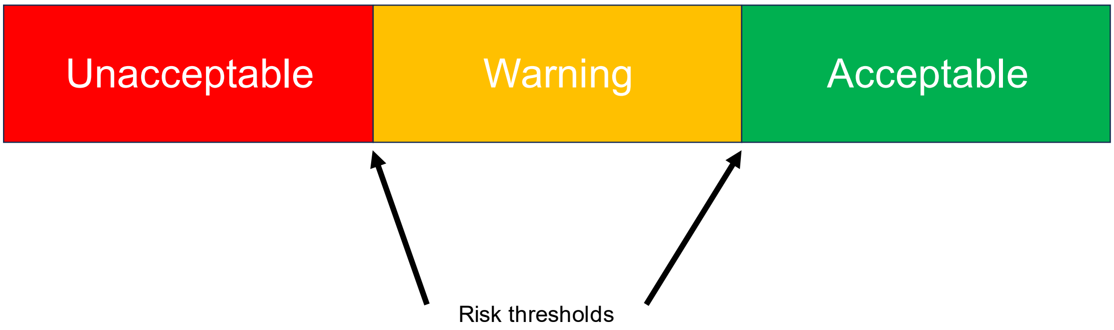


Figure 2: Spectrum of risk appetite

To effectively select controls, enterprises rely on their internal control frameworks. For example, a cloud security control framework defines the policies and controls needed to secure cloud services used by the enterprise. As the internal control framework is derived from the control and enablement objectives of the enterprise security architecture, it is tailored to the risk appetite of the enterprise. Furthermore, this relationship provides internal and external auditors with the traceability and assurances they require.

Defining an internal control framework is an arduous and time-consuming task, so enterprises often rely on industry-standard control frameworks to base their internal control frameworks on, such as Center for Internet Security (CIS) Benchmarks.

To secure emerging services, such as generative AI, no such framework exists. In the case of generative AI, the pressure to adopt these services in a timely manner is significant. In practice, this has led to enterprises altering their regular processes and opting to invest their own efforts in building these frameworks from scratch. As most enterprises adopt generative AI using a hosted service, the baseline controls are very similar to cloud services.

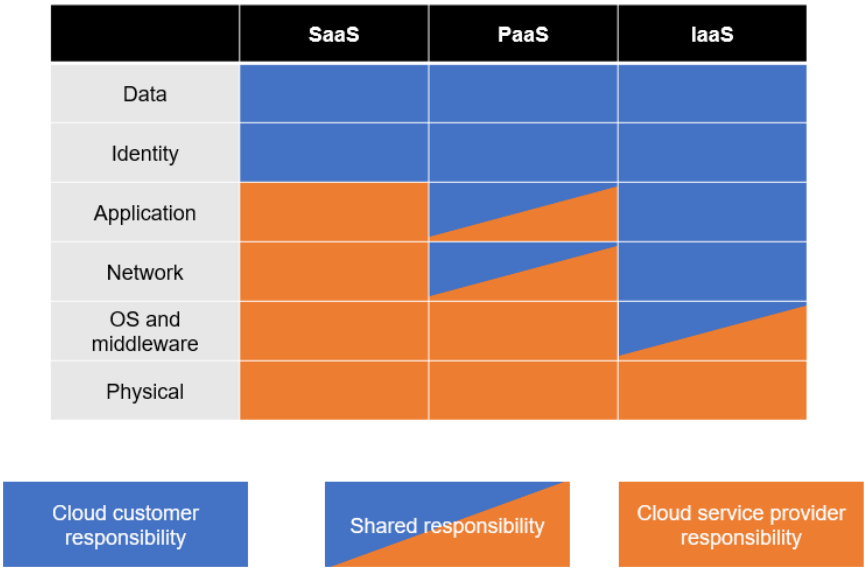
In order to select the controls that are suitable for their risk appetite, enterprises can follow established cloud control frameworks. Some control frameworks include:

- Center for Internet Security’s Microsoft Azure Foundations Benchmark
- Cloud Security Alliance’s Cloud Controls Matrix (CCM)
- Microsoft Cloud Security Benchmark (MCSB).

3.2 Enterprise security requirements for cloud services

3.2.1 Shared responsibility matrix

In the context of cloud services, security is always a shared responsibility. Figure 3 illustrates how the security responsibilities are shared between the enterprise consuming the services and the cloud service providers.



	SaaS	PaaS	IaaS
Data	🟦	🟦	🟦
Identity	🟦	🟦	🟦
Application	🟧	◩	🟦
Network	🟧	◩	🟦
OS and middleware	🟧	🟧	◩
Physical	🟧	🟧	🟧

🟦 Cloud customer responsibility

◩ Shared responsibility

🟧 Cloud service provider responsibility

Figure 3: Cloud shared responsibility matrix

Software as a Service (SaaS) includes a high level of security out of the box. Outside of toggling built-in features on or off, the responsibilities of the enterprise are solely on securing the data and identities used with the SaaS. It’s the responsibility of enterprises to assess whether the security implementation of the cloud provider across application, networking, operating system, middleware, and physical layers meets their requirements. If not, they should select another cloud service model as the availability of compensating controls is limited to Data and Identity layers.

In Platform as a Service, PaaS, enterprises gain more control over the cloud service. Namely, they are now responsible for securely configuring application and network layers of control. Depending on the PaaS service, the breadth of available security controls can be quite exhaustive, ranging from log verbosity configuration to cipher suite selection. This cloud service model is often chosen when security controls in SaaS are too limited, but the enterprise still wishes to benefit from the evergreen and managed nature of cloud services.

In Infrastructure as a Service (IaaS), enterprises are responsible for securing a large number of layers. This cloud service model provides the most control, but least amount of built-in security, meaning that this model both allows and requires the enterprise to take responsibility for more layers compared to other cloud service models. This service model is often chosen when the enterprise already has an established security operations model in the cloud, or due to compatibility issues when migrating existing applications. However, compared to operating in their wholly owned datacenters, enterprises are not able to control host operating systems or physical security. It's the responsibility of enterprises to assess whether the security implementation of the cloud provider meets their requirements.

3.2.2 Microsoft Cloud Security Benchmark

The Microsoft cloud security benchmark (MCSB) is a framework of technical controls for Microsoft Azure (Microsoft, 2023). At the time of writing, MCSB consisted of 85 controls and 116 security baselines, sets of implementation guidance for individual Microsoft Azure services. The controls are categorized under 12 control domains.

3.3 Enterprise security requirements for generative AI

Since its release in November 2022, OpenAI’s ChatGPT enjoyed explosive user adoption. Despite not offering the Plus and Enterprise features for months, it was also widely adopted in enterprises. This comes to show that when the business demand is high enough, any and all established security governance is omitted. This is comparative to the fast adoption of cloud computing during the early Covid-19 pandemic and adoption of remote working, the decision to adopt the new services were made first, and security requirements were considered only after the fact.

Even a year after the launch of ChatGPT, the full set of risks related to generative AI are not yet understood. At the same time, the prospective use cases continue to evolve, making the dual goal posts of control and enablement objectives elude us. The best we can do is to adapt our closest equivalent controls, namely from the cloud computing and data analytics domains. As the generative AI industry continues to mature, we will hopefully see the current situation improving.

In absence of established frameworks and for the purposes of this paper, a subset of common cloud security requirements was selected. Note that while these security requirements represent a subset of typical enterprise needs, each enterprise is different and has a unique business environment and risk appetite. The security requirements are listed in Table 3-1.

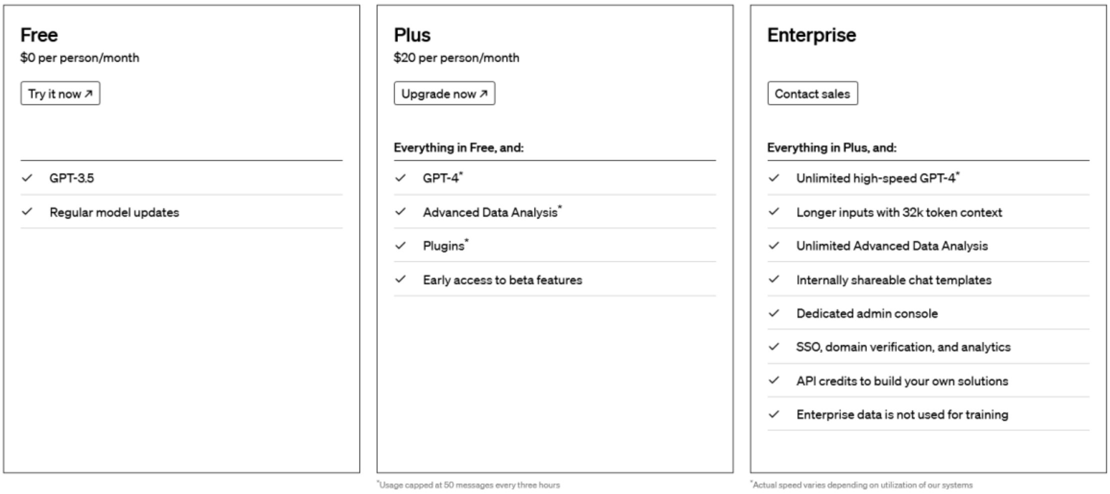
Table 3-1: representative enterprise security requirements for generative AI systems.

Requirement	**Description**
Audit logging	Ability to provide a trusted log for all user and administrative activities.
Data protection	Ability to encrypt data in transit and at rest.
Data residency	Ability to control data location.
Identity and access management	Ability to authenticate and authorize user access.
Network isolation	Ability to control inbound and outbound network traffic.
Privacy and Compliance	Ability to meet regulatory and industry-specific regulation.

3.4 Comparing OpenAI ChatGPT Enterprise and Azure OpenAI

There are two main hosting models for using generative AI services by Open AI: directly from OpenAI as SaaS (ChatGPT), or through Microsoft Azure as PaaS (Azure OpenAI service). While both options share a lot of the code base, hosting infrastructure in the Microsoft Azure cloud, and functionality, they are operated by distinct companies with distinct differences in available security controls.

OpenAI offers three plans for their ChatGPT product. As illustrated by Figure 4, the difference between the plans is mainly focused on available features, performance, and throttling. ChatGPT Enterprise is positioned as an enterprise-ready plan, with some exclusive security controls.



Free
$0 per person/month

- ✓ GPT-3.5
- ✓ Regular model updates

Plus
$20 per person/month

Everything in Free, and:

- ✓ GPT-4*
- ✓ Advanced Data Analysis*
- ✓ Plugins*
- ✓ Early access to beta features

*Usage capped at 50 messages every three hours

Enterprise

Everything in Plus, and:

- ✓ Unlimited high-speed GPT-4*
- ✓ Longer inputs with 32k token context
- ✓ Unlimited Advanced Data Analysis
- ✓ Internally shareable chat templates
- ✓ Dedicated admin console
- ✓ SSO, domain verification, and analytics
- ✓ API credits to build your own solutions
- ✓ Enterprise data is not used for training

*Actual speed varies depending on utilization of our systems

Figure 4: summary of OpenAI ChatGPT plans (OpenAI, 2023)

Security controls of OpenAI’s ChatGPT Enterprise were assessed based on the company’s Trust Center materials (OpenAI, 2023), which include a security whitepaper, Consensus Assessments Initiative Questionnaire (CAIQ), SOC 2 Type 2 report and a redacted penetration testing report from 2022.

According to the assessment, the prompt data for users in Enterprise tier is not used for modelling purposes. Specifically, OpenAI claims that “Data sent via the API or ChatGPT Enterprise are default opt-out and are not used to train our models.” However, at the time of writing, the company presents contradictory information in other parts of their trust documents. When it comes to the description of their model training, they state that they “have a scrubbing process which removes PII from the data sets prior to being processed for training.” This applies to “only for the data that will be used for training. Customer prompt/completion data we do not have the ability to anonymize.” Based on the available documentation, it is unclear whether this lack of an ability to anonymize prompt data applies to their entire architecture or simply the free tier. As far as assurances regarding prompt data privacy go, the current description is incomplete.

Enterprise plan features are defined in (Open AI, 2023). These mainly focus on paid privacy, as enterprise data is not used for training, SSO, domain verification and yet unspecified features of an analytics dashboard. As the security questionnaire includes a reference to missing full audit logging capability, it remains unclear what is the exact nature of the Analytics Dashboard and Usage Insights.

The product page also lists data encryption at rest (using AES 256) and encryption in transit (TLS 1.2) as features, but as these are the default features of the underlying Microsoft Azure components, it remains unclear if these are limited to Enterprise tier or apply to all instances.

Furthermore, there will be another, 4th, pricing plan, which will be positioned between the Enterprise and Free pricing tiers. It is described as an offering for smaller organizations.

ChatGPT data is stored in Microsoft Azure datacenters in the United States, specifically in West US 2, East US, East US 2 and South Central US. End users are not able to influence where the data is stored.

For Azure OpenAI Service, prompt, completion, embeddings, and training data remains in the enterprise control (Microsoft, 2023). The service can be deployed to Australia East, Canada East, West Europe, France Central, Japan East, Qatar East, Sweden Central, Switzerland North, UK South, East US, East US2, North Central US, and South Central US datacenter regions.

It is worth noting that while Azure OpenAI is considered as Generally Available from the perspective of agreement terms and service level agreements, at the time of writing access to the access to the service is not publicly available. Rather it is gated behind an application form. Microsoft states that this is both due to high demand for the service, but also because of “Microsoft’s commitment to responsible AI” (Microsoft, 2023).

Table 3-2: comparison of supported security controls.

Control	**ChatGPT Free & Plus**	**ChatGPT Enterprise**	**Azure OpenAI Service**
Privacy and Compliance: prompt privacy	No	Yes (limited)	Yes
Identity and access management: SSO	No	Yes (limited)	Yes
Data protection: encryption at rest	Unclear	Yes	Yes (including BYOK)
Audit logging	No	No	Yes
Network isolation	No	No	Yes
Data residency	No	No	Yes

Table 3-2 above summarizes the differences between ChatGPT and Azure OpenAI Service from the perspective of security controls. While OpenAI and particularly its Enterprise tier provide the latest features and continue to also catch up on security, privacy and compliance, Azure OpenAI service is likely to be a better fit for enterprise usage at the time of writing this publication. This is mainly due to lack of control for data residency, unclear assurances for prompt data privacy, and missing core technical controls, such as audit logging and network isolation.

4 Design

In this chapter, we present a reference application architecture, its threat model and security controls selected to mitigate the identified threats.

4.1 Reference application architecture

For the purpose of this paper, a high-level reference application architecture was defined, as illustrated in Figure 5. At its core, the architecture follows a familiar 3-tier software architecture:

- **Presentation tier**, consisting of a front-end application allowing the user to prompt questions and review results. At its simplest form, this is a web application providing chatbot functionality. In an enterprise setting, this tier could be integrated in the existing application or workflow, such as customer relationship management (CRM) tool, call center software, or internal communications application.
- **Application tier**, consisting of the large language model service. In this tier, actual models such as GPT-3.5 Turbo, GPT-4 and DALL-E 3 are hosted and exposed to the presentation tier.
- **Data tier**, consisting of fine-tuning and training data. Fine-tuning is a process of customizing the large language model with sample dataset specific to the enterprise. Fine-tuning the model is a necessary step in established enterprise use cases, as this allows for higher quality results, improved model request latency and at times lower model usage costs.

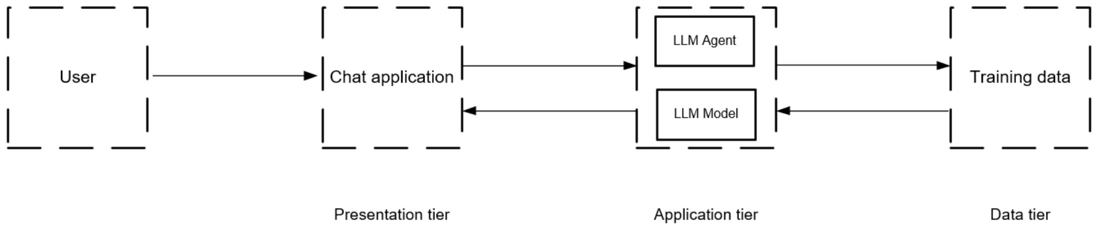


Figure 5: reference application architecture

The architecture is implemented in Microsoft Azure in a manner illustrated in Figure 6.

- **Presentation tier** is implemented as an Azure App Service Web Application, a fully managed PaaS service for hosting web applications. The service was selected as it is one of the most widely used Microsoft Azure services used for hosting front-end applications. In addition to the full web service capabilities, App Service can also be used in a more lightweight fashion as a hosting platform for REST API, serverless, and mobile applications. As such, it is going to likely be used as part of the presentation layer, even if the enterprise would integrate the application into their own frontend.
- **Application tier** is implemented as an Azure OpenAI Service, a fully managed PaaS service for hosting large language models. At the time of writing, the service supports OpenAI GPT-4, GPT-3, Codex, DALL-E, and Whisper models, with close collaboration and co-development of models with OpenAI. Compared to OpenAI, Azure OpenAI offers more control on the deployment, as well as some additional features such as responsible AI content filtering. Azure OpenAI APIs are designed to be closely compatible with those of OpenAI. As a result, developers can use the same Python client libraries for both services with minimal changes (OpenAI, 2023).
- **Data tier** is implemented as Azure Storage Account, a fully managed PaaS storage service. Specifically, we are using Blob Storage, an object storage service designed for storing unstructured data such as binary files.

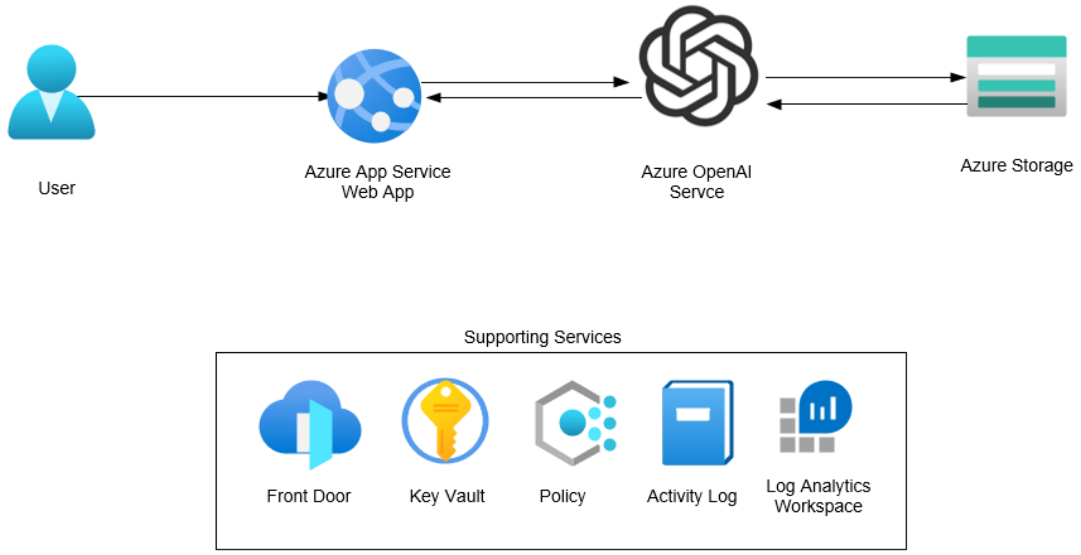


Figure 6: reference application architecture in Microsoft Azure

Additionally, we are using a number of supporting Azure services as part of the solution. As they are not unique to this solution and have been extensively covered by other sources, we are not addressing them in further detail. These services include:

- **Azure Front Door**, a content delivery network (CDN) service with a web application firewall (WAF) functionality. The service provides protection against common web application vulnerabilities and DDoS attacks.
- **Azure Key Vault**, a secure storage and management solution for cryptographic keys, certificates, and secrets.
- **Azure Policy**, a service for enforcing controls across Azure services.
- **Activity Logs**, a type of immutable platform-level logs that contain information such as deletion of resources and changes in access control assignments.
- **Log Analytics Workspace**, for storing audit log information from the rest of the services. The service supports write once, read many (WORM) capabilities, ensuring that the log files are not tampered with.

4.1.1 Reference application architecture limitations

The reference architecture aims to illustrate key components regarding the usage of Azure OpenAI, not to be a comprehensive description of all adjacent services in a full enterprise context. We are particularly focusing on the end user scenario, where internal users are prompting an already trained model through a web interface.

Our primary assumption is that core cloud security and IT service management capabilities are already present and operational.

There are several areas where the reference architecture can be expanded upon. These include:

- **Operational flow (MLOps)**. The current architecture is focusing on the end result, without consideration for model training, fine-tuning or continuous improvement. This represents a simplified view of real-world enterprise scenarios, where models are constantly developed, and even new use cases and data sources added. A similar security approach should be undertaken throughout the whole generative AI supply chain.
- **Embeddings support**. An embedding is mathematical representation of the semantic meaning of a piece of text. Embeddings are often used for searching and anomaly detection. In order to use embeddings, a vector store such as Azure Cognitive Search or Azure CosmosDB could be introduced to the architecture.
- **Additional Azure components**. In addition to the supporting services mentioned above, it is worth considering adding additional Azure components to the architecture. For example, introducing an Azure API Management service would allow us to monitor and manage all API calls from the frontend application to the model service. This would also be beneficial when considering extension support.

4.2 Reference application threat model

A threat modeling exercise was performed following the STRIDE (spoofing, tampering, repudiation, information disclosure, denial of service, elevation of privilege) methodology.

Figure 7 illustrates the threat model diagram of the reference application, including data flows and trust boundaries. The threat model includes 33 threats, 16 of which are considered for mitigation in the AI application level, and 17 of which were mitigated in the AI platform level, as part of our reference architecture. The full threat model report is in Appendix 1.

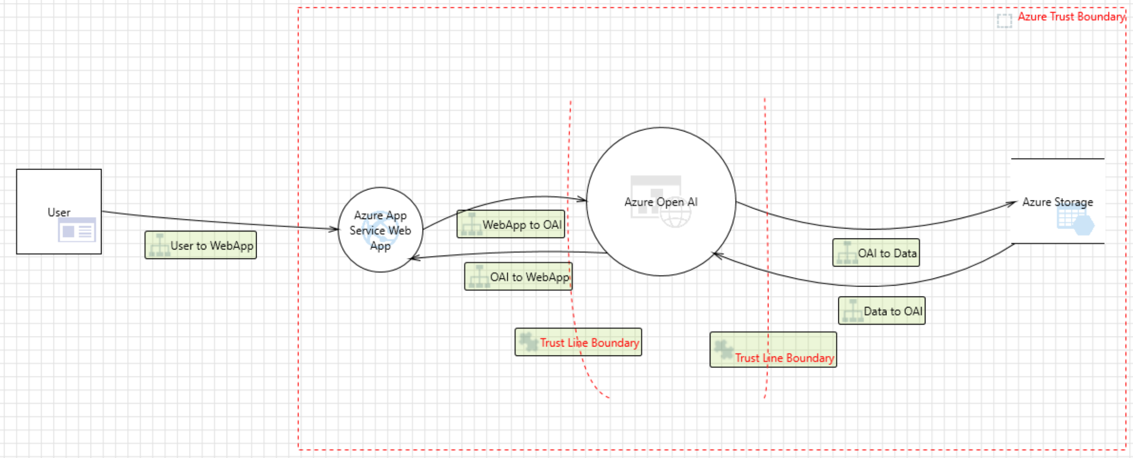


Figure 7: threat model of the reference application

4.3 Microsoft Cloud Security Baseline for Azure OpenAI

Microsoft Cloud Security Baseline for Azure OpenAI was released in October 2023 (Microsoft, 2023). At the time of writing, the baseline is still incomplete, covering 7 recommendations across 6 controls of the complete baseline of 35 controls, enumerated in Table 4-1 below.

Table 4-1: baseline security controls for Azure OpenAI Service

Control Domain	**MCSB control ID**	**Control title**	**Guidance**	**Feature**
Data Protection	DP-2	Monitor anomalies and threats targeting sensitive data	Azure OpenAI services data loss prevention capabilities allow customers to configure the list of outbound URLs their Azure OpenAI services resources are allowed to access.	Data Leakage/Loss Prevention
Data Protection	DP-5	Use customer-managed key option in data at rest encryption when required	Enable and implement data at rest encryption using customer-managed key when required.	Data at Rest Encryption Using CMK
Data Protection	DP-6	Use a secure key management process	Use Azure Key Vault to create and control the life cycle of your encryption keys, including key generation, distribution, and storage.	Key Management in Azure Key Vault
Identity Management	IM-8	Restrict the exposure of credential and secrets	Ensure that secrets and credentials are stored in secure locations such as Azure Key Vault, instead of embedding them into code or configuration files.	Secrets Support Integration and Storage in Azure Key Vault

Logging and threat detection	LT-4	Enable network logging for security investigation	Enable resource logs for the service.	Azure Resource Logs
Network Security	NS-2	Secure cloud services with network controls	Disable public network access either using the service-level IP ACL filtering rule or a toggling switch for public network access.	Disable Public Network Access
Network Security	NS-2	Secure cloud services with network controls	Deploy private endpoints for all Azure resources that support the Private Link feature, to establish a private access point for the resources.	Azure Private Link

Table 4-2 presents the MCSB baseline controls as mapped to industry-standard control frameworks. Even if they would not be familiar with MCSB control framework, this mapping presents enterprises with a familiar and traceable translation layer between their own control framework and the controls selected in this paper.

Table 4-2: Selected controls mapped to CIS and NIST controls.

Control Domain	**MCSB control ID**	**CIS Controls v8 ID(s)**	**NIST SP800-53 r4 ID(s)**
Data Protection	DP-2	3.13 - Deploy a Data Loss Prevention Solution	AC-4: INFORMATION FLOW ENFORCEMENT SI-4: INFORMATION SYSTEM MONITORING

Data Protection	DP-5	3.10 - Encrypt Sensitive Data In Transit	SC-8: TRANSMISSION CONFIDENTIALITY AND INTEGRITY
Data Protection	DP-6	N/A	IA-5: AUTHENTICATOR MANAGEMENT SC-12: CRYPTOGRAPHIC KEY ESTABLISHMENT AND MANAGEMENT SC-28: PROTECTION OF INFORMATION AT REST
Identity Management	IM-8	16.9 - Train Developers in Application Security Concepts and Secure Coding 16.12 - Implement Code-Level Security Checks	IA-5: AUTHENTICATOR MANAGEMENT
Logging and threat detection	LT-4	8.2 - Collect Audit Logs 8.5 - Collect Detailed Audit Logs 8.6 - Collect DNS Query Audit Logs 8.7 - Collect URL Request Audit Logs 13.6 - Collect Network Traffic Flow Logs	AU-3: CONTENT OF AUDIT RECORDS AU-6: AUDIT REVIEW, ANALYSIS, AND REPORTING AU-12: AUDIT GENERATION SI-4: INFORMATION SYSTEM MONITORING
Network Security	NS-2	3.12 - Segment Data Processing and Storage Based on Sensitivity 4.4 - Implement and Manage a Firewall on Servers	AC-4: INFORMATION FLOW ENFORCEMENT SC-2: APPLICATION PARTITIONING SC-7: BOUNDARY PROTECTION

5 Security control implementation

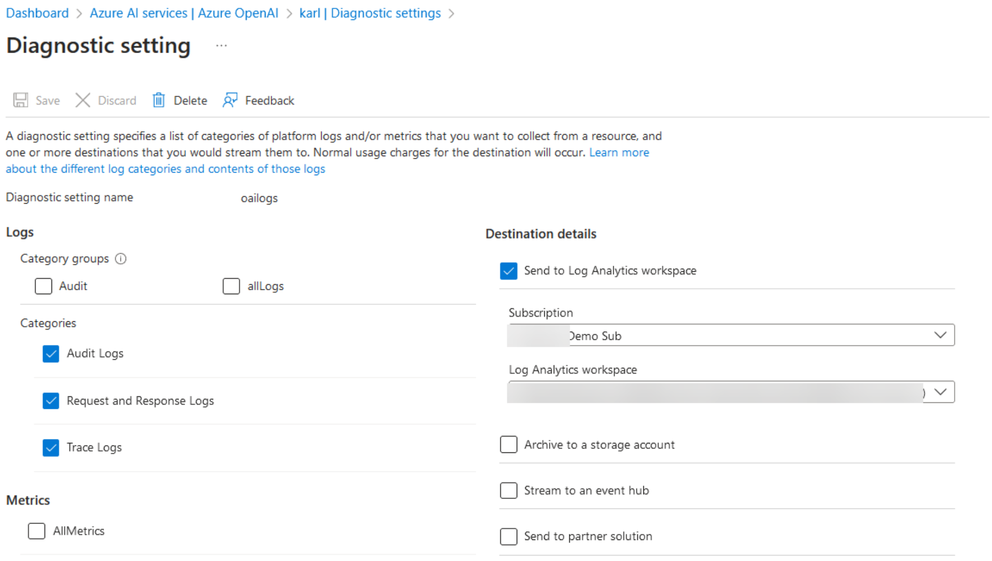
In this chapter, we present a hardened configuration of the reference application architecture.

5.1 Azure OpenAI

5.1.1 Audit logging

In the default configuration, the Azure OpenAI instance does not produce any log data. As such, only Azure activity logs are available. These include cloud control pane level events, such as write and delete operations of entire resources. The operations are logged regardless of whether they were successful or not.

More detailed logs from the data pane of the service need to be enabled explicitly. These logs include events such as chat completions, file uploads, image generations and administrative activity on viewing or editing model configuration. To enable audit log generation of the service itself, a new log export rule needs to be created under Diagnostic setting. Figure 8 illustrates a properly configured setting. The full list of available audit logs is defined in (Microsoft, 2023).



Dashboard > Azure AI services | Azure OpenAI > karl | Diagnostic settings >

Diagnostic setting ···

A diagnostic setting specifies a list of categories of platform logs and/or metrics that you want to collect from a resource, and one or more destinations that you would stream them to. Normal usage charges for the destination will occur. Learn more about the different log categories and contents of those logs

Diagnostic setting name: oailogs

Logs

Category groups
- □ Audit
- □ allLogs

Categories
- ☑ Audit Logs
- ☑ Request and Response Logs
- ☑ Trace Logs

Metrics
- □ AllMetrics

Destination details
- ☑ Send to Log Analytics workspace

Subscription: [redacted] Demo Sub

Log Analytics workspace: [redacted]

- □ Archive to a storage account
- □ Stream to an event hub
- □ Send to partner solution

Figure 8: configuration of audit log exporting for Azure OpenAI

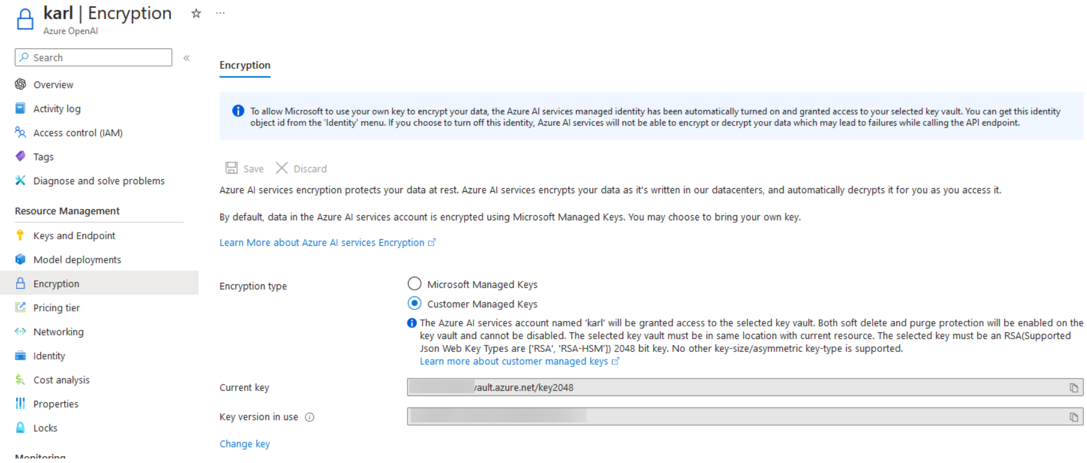
Suitable audit log destinations vary based on each enterprise’s needs. For storing and analysing the logs in the cloud, it is often best to use Log Analytics workspace. If the logs need to be also sent to a centralized cyber hygiene team for monitoring, the same rule can be used to select multiple destinations.

5.1.2 Data protection

Following the standard behavior and in line with OpenAI ChatGPT, the **data at rest** in Azure OpenAI is encrypted using AES-256 and Microsoft-managed keys (MMK) by default. **Data in transit** is similarly encrypted using TLS 1.2.

Data residency can be controlled by selecting the Azure region where the Azure OpenAI instance is deployed to. The service can be deployed to Australia East, Canada East, West Europe, France Central, Japan East, Qatar East, Sweden Central, Switzerland North, UK South, East US, East US2, North Central US, and South Central US datacenter regions.

Encryption keys for data at rest can be controlled by choosing the Customer Managed Keys (CMK) encryption type, as shown in Figure 9. This functionality is sometimes referred to as Bring Your Own Key (BYOK) encryption and is often a required control in regulated industries. This allows for full control of key operations, rotation, and higher encryption strength of 2048-bit RSA.



To allow Microsoft to use your own key to encrypt your data, the Azure AI services managed identity has been automatically turned on and granted access to your selected key vault. You can get this identity object id from the 'Identity' menu. If you choose to turn off this identity, Azure AI services will not be able to encrypt or decrypt your data which may lead to failures while calling the API endpoint.

Azure AI services encryption protects your data at rest. Azure AI services encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

By default, data in the Azure AI services account is encrypted using Microsoft Managed Keys. You may choose to bring your own key.

Learn More about Azure AI services Encryption

Encryption type: ○ Microsoft Managed Keys ● Customer Managed Keys

The Azure AI services account named 'karl' will be granted access to the selected key vault. Both soft delete and purge protection will be enabled on the key vault and cannot be disabled. The selected key vault must be in same location with current resource. The selected key must be an RSA(Supported Json Web Key Types are ['RSA', 'RSA-HSM']) 2048 bit key. No other key-size/asymmetric key-type is supported.
Learn more about customer managed keys

Current key: [redacted]vault.azure.net/key2048

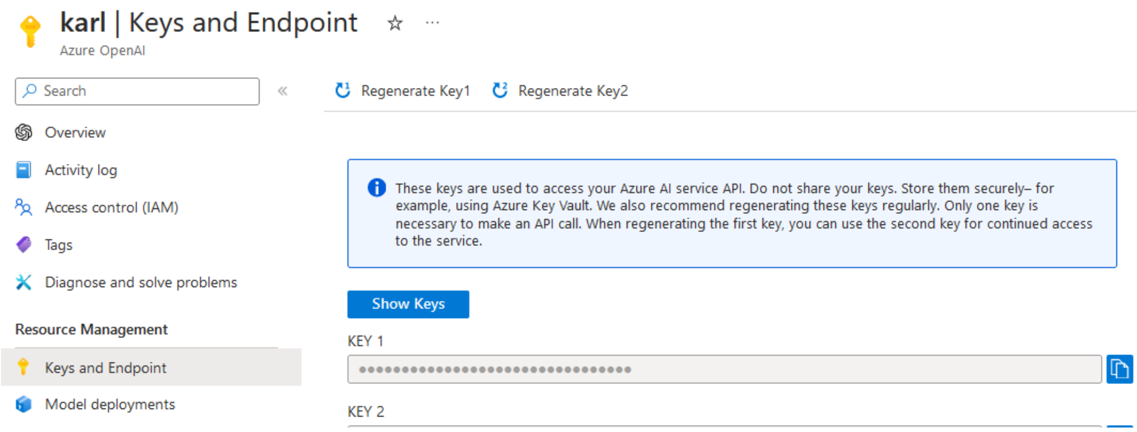
Key version in use: [redacted]

Change key

Figure 9: Azure OpenAI service encrypted using Customer Managed Keys

5.1.3 Identity and Access management

Azure OpenAI supports two access models: centrally managed identity using Entra ID (formerly Azure AD), and local authentication using API keys (see Figure 10). Enterprises should avoid using local authentication whenever possible and always use Entra ID authentication for end users, developers, administrators, and data scientists. Application and other non-interactive access should be granted using Entra ID Managed Identities, to avoid storing any credentials in code.



These keys are used to access your Azure AI service API. Do not share your keys. Store them securely– for example, using Azure Key Vault. We also recommend regenerating these keys regularly. Only one key is necessary to make an API call. When regenerating the first key, you can use the second key for continued access to the service.

KEY 1: ••••••••••••••••••••••••••••••••

Figure 10: local authentication for Azure OpenAI

At the time of writing, there is no user interface or template parameter available for disabling API key-based authentication. The only available method is by editing the RESTful parameters at runtime, as illustrated in Listing 1.

```
az rest -m patch -u /subscriptions/{subscription ID}}/resourceGroups/{resource
group}/providers/Microsoft.CognitiveServices/accounts/{account name}?api-
version=2021-04-30 -b '{"properties": { "DisableLocalAuth": true  }}'
```

Listing 1: Azure command line interface command for disabling API key-based authentication if Azure OpenAI

Note that it might be unfeasible to disable the API key-based authentication in some use cases throughout the development lifecycle, as Azure OpenAI Studio uses the API key authentication.

Enterprises should use built-in role-based access control (RBAC) roles to grant access to centrally managed identities in Entra ID. There are two built-in roles available for Azure OpenAI:

- **Cognitive Services OpenAI User**, which provides prompt completion access, as well as limited access to view model and deployment information. While still quite privileged, this is the standard role the users or applications should be granted.

- **Cognitive Services OpenAI Contributor**, which provides full access including the ability to fine-tune, deploy and generate text. This role should be used for privileged users.

5.1.4 Networking isolation

By default, in Azure OpenAI service, both inbound and outbound network traffic is unrestricted. While inbound traffic gets protected by Azure DDoS Infrastructure Protection, this merely protects against cloud-scale volumetric attacks, and is no substitute for workload-level network protection. Figure 11 illustrates network the network controls of our solution.

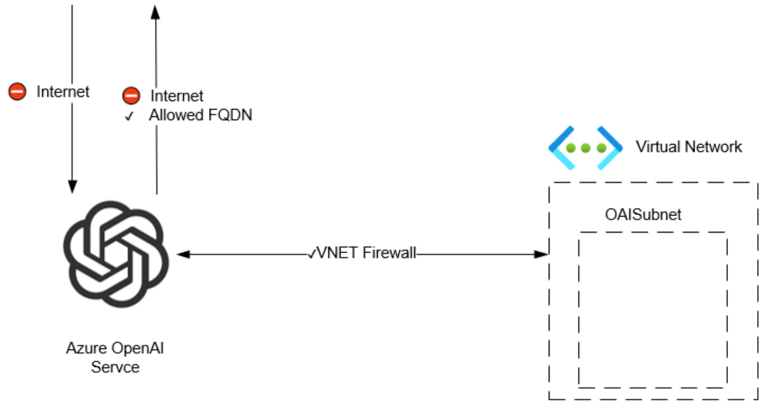
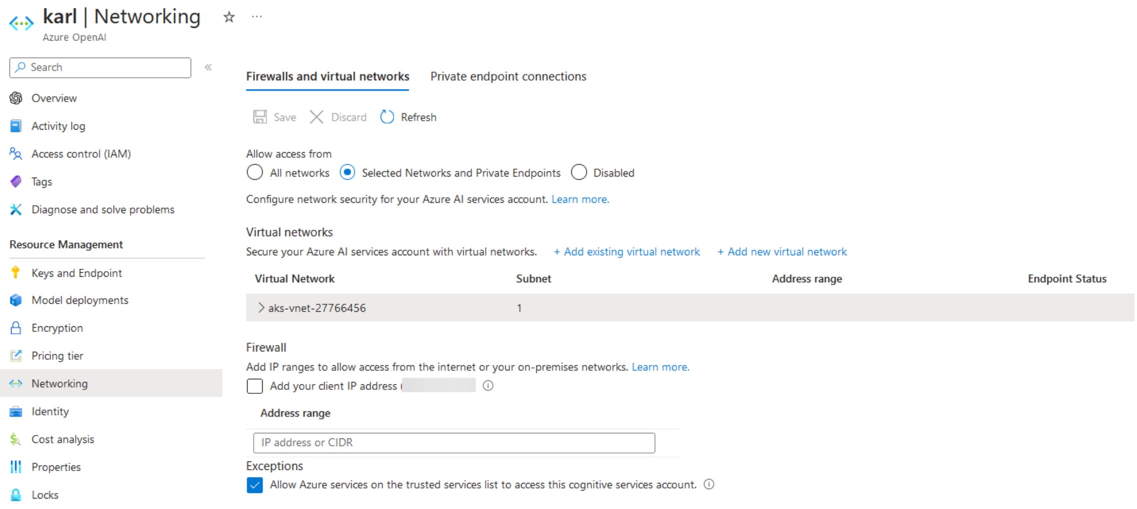


Figure 11: network controls for Azure OpenAI

To control **inbound** network traffic, we need to enable the Selected Networks and Private Endpoints mode under Networking -> Firewall. At least one subnet of an Azure virtual network is required as configuration. Figure 12 illustrates the properly configured service. This setting can also be configured as infrastructure as code, using the NetworkRuleSet property (Microsoft, 2023).



Allow access from
○ All networks ● Selected Networks and Private Endpoints ○ Disabled

Configure network security for your Azure AI services account. Learn more.

Virtual networks

Secure your Azure AI services account with virtual networks. + Add existing virtual network + Add new virtual network

Virtual Network	Subnet	Address range	Endpoint Status
> aks-vnet-27766456	1		

Firewall

Add IP ranges to allow access from the internet or your on-premises networks. Learn more.

□ Add your client IP address ([redacted] ⓘ

Address range

IP address or CIDR

Exceptions

☑ Allow Azure services on the trusted services list to access this cognitive services account. ⓘ

Figure 12: configuration of virtual network firewall for Azure OpenAI

To control **outbound** network traffic, we need to configure the data loss prevention capability of Azure OpenAI. This lets us configure the explicit list of up to 1000 Fully Qualified Domain Names (FQDNs) our Azure OpenAI instance is allowed to access.

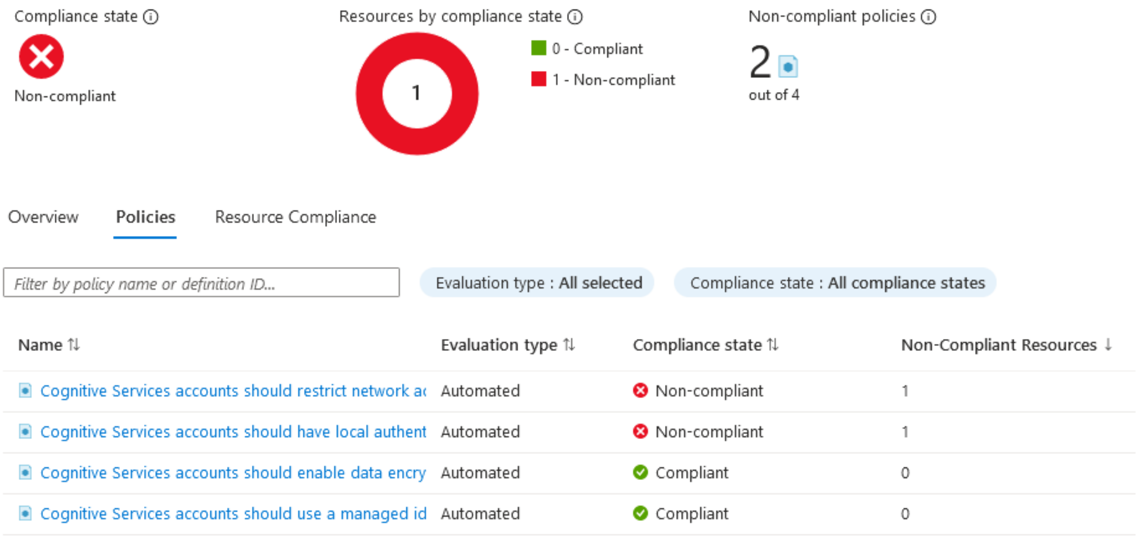
Data loss prevention is configured by enabling the restrictOutboundNetworkAccess property and updating the allowedFqdnList with a list of approved domain names. At the time of writing, there is no user interface or template parameter available to configure this in infrastructure as code. The only available method is by editing the RESTful parameters at runtime, as illustrated in Listing 2.

```
az rest -m patch -u /subscriptions/{subscription ID}}/resourceGroups/{resource
group}/providers/Microsoft.CognitiveServices/accounts/{account name}?api-
version=2021-04-30 -b '{"properties": { "restrictOutboundNetworkAccess": true,
"allowedFqdnList": [ "karlots.com" ] }}'
```

Listing 2: Azure command line interface command for configuring outbound network controls

5.1.5 Policy enforcement

Azure has a built-in feature for enforcing security controls across at scale, named Azure Policy. Policies can be deployed across the entire enterprise cloud footprint, and they can be used to monitor, prevent, and automatically remediate any misconfigurations against the desired secure state. The policy engine is also continuously evaluating the resources' compliance against the policies (see Figure 13), providing the enterprise with a view of their security posture over time.



Compliance state ⓘ

✗ Non-compliant

Resources by compliance state ⓘ

🟩 0 - Compliant

🟥 1 - Non-compliant

Non-compliant policies ⓘ

2 out of 4

Evaluation type : All selected

Compliance state : All compliance states

Name ⇅	Evaluation type ⇅	Compliance state ⇅	Non-Compliant Resources ↓
Cognitive Services accounts should restrict network ac	Automated	✗ Non-compliant	1
Cognitive Services accounts should have local authent	Automated	✗ Non-compliant	1
Cognitive Services accounts should enable data encry	Automated	✓ Compliant	0
Cognitive Services accounts should use a managed id	Automated	✓ Compliant	0

Figure 13: monitoring the security configuration of Azure OpenAI using Azure Policy

There are no built-in policies available for Azure OpenAI yet. However, as the service is under the Microsoft.CognitiveServices resource provider, some existing policies built for other Cognitive Services can be reutilized. These include the following built-in policies, also available as a policy initiative in Appendix 2:

- Cognitive Services accounts should restrict network access
- Cognitive Services accounts should have local authentication methods disabled
- Cognitive Services accounts should enable data encryption with a customer-managed key
- Cognitive Services accounts should use a managed identity

5.2 Supporting Azure services

5.2.1 App Service

To limit the application to the appropriate audience, App Service Web App is configured to exclusively use Entra ID authentication. This forces incoming requests to pass through the authentication module, which evaluates whether the authentication claims are coming from the specific Entra ID tenant. This module validates, stores, and refreshes the authentication tokens in a dedicated token store within the App Service (Ots, 2021). Authorization of the requests can be performed using Entra ID conditional access, which allows for evaluating of Entra ID group memberships and modern risk-based user information, such as device health, network location and multi-factor authentication status.

By default, all inbound traffic to the Web App is allowed. To control inbound traffic, Access Restrictions is configured to specify an allow list of traffic. In the case of this reference implementation, the traffic should be filtered based on the Azure Front Door Instance ID header. When this is configured, all other traffic is denied, as illustrated in Figure 14.

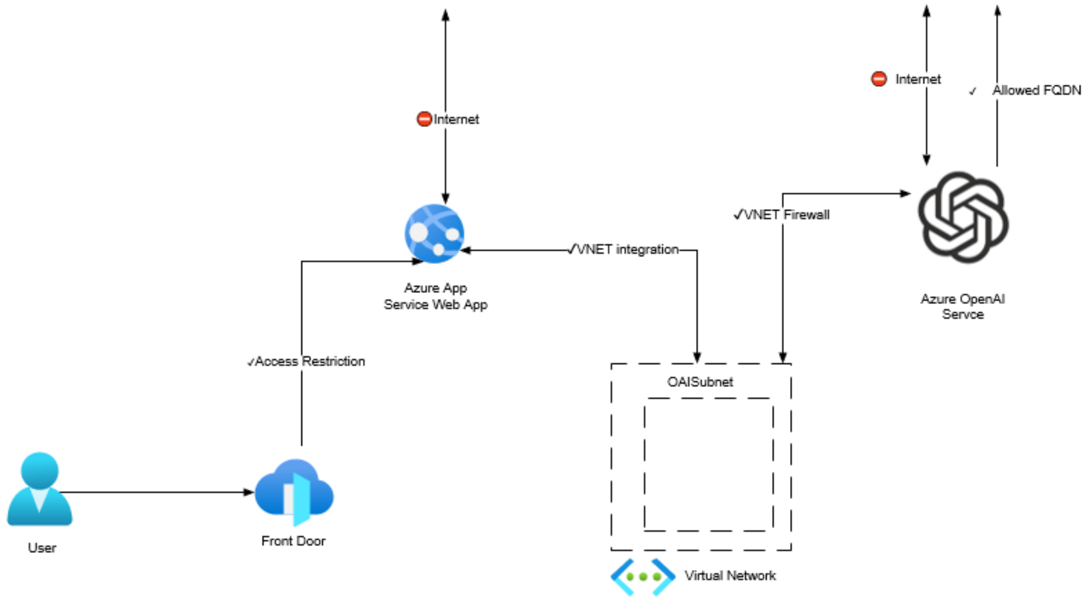


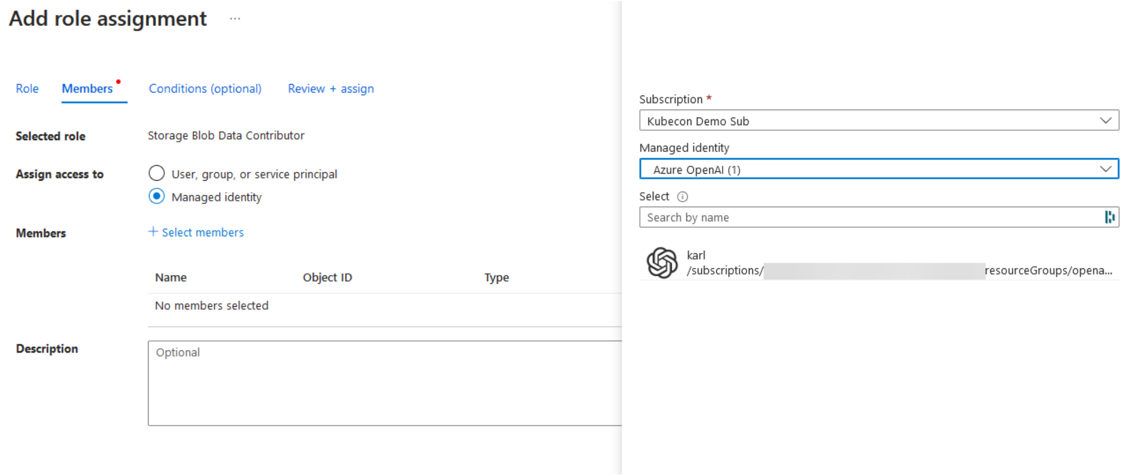
Figure 14: network controls for App Service

Outbound traffic is controlled by configuring virtual network integration with the target virtual network and enabling the WEBSITE_VNET_ROUTE_ALL setting. The virtual network is also configured to allow traffic from this App Service endpoint and deny any outbound traffic to the internet.

Audit logging is enabled by configuring the log export functionality under Diagnostic Settings, as for the same feature in Azure OpenAI. In the case of App Service Web App, the logs include both allowed and denied requests made to the web app (AppServiceIPSecLogs), and web server logs. In the case of Azure Front Door, the logs include any requests that match the rules in the Web Application Firewall of Azure Front Door (FrontDoorWebApplicationFirewallLog).

5.2.2 Storage Account

Storage Account supports two access models: centrally managed identity using Entra ID, and local authentication using shared access keys. Enterprises should avoid using local authentication whenever possible and always use Entra ID authentication. In our reference implementation, the system-assigned managed identity of the Azure OpenAI instance should be granted access to the Storage Account, as illustrated in Figure 15.



Add role assignment ...

Selected role: Storage Blob Data Contributor

Assign access to: ○ User, group, or service principal ● Managed identity

Members: + Select members

Name	Object ID	Type
No members selected		

Description: Optional

Subscription *: Kubecon Demo Sub

Managed identity: Azure OpenAI (1)

Select: Search by name

karl
/subscriptions/[redacted]resourceGroups/opena...

Figure 15: Granting a managed identity access to the Storage Account

Storage Account supports disabling the local authentication in a more mature way. Local authentication is disabled in the portal UI under Settings - Allow storage account key access.

To control inbound network traffic, we need to enable the Selected Networks and Private Endpoints mode under Networking -> Firewall. At least one subnet of an Azure virtual network is required as configuration. This feature behaves the same way as that of the Azure OpenAI Service. Figure 16 illustrates the compounded effects of the network controls.

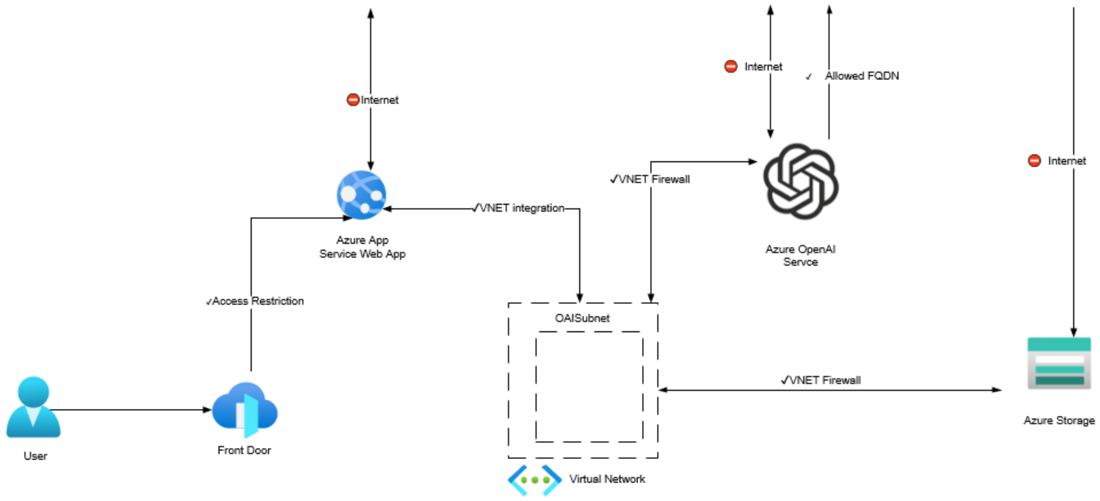


Figure 16: Network controls for Azure Storage Account

Audit logging for Storage Account is enabled by configuring the log export functionality under Diagnostic Settings, as for the same feature in Azure OpenAI and App Service Web App. These include logs for administrative activities, such as disabling or tampering with the network controls. Enabling Microsoft Defender for Cloud for the Storage Account will additionally monitor and alert against suspicious activity, anonymous scans and potential malware being uploaded.

Encryption keys for data at rest can be controlled by choosing the Customer Managed Keys (CMK) encryption type, as for the same feature in Azure OpenAI.

6 Conclusion

This study started off as research into a fairly limited scope, hardening of Azure OpenAI service. Draft findings of this study and hardening steps were presented at the ESPC23 conference in Amsterdam in November 2023. Based on the feedback from other industry practitioners, it became evident that the research question warranted expansion.

As the author found out, there is a distinct lack of any publicly available benchmarks, frameworks, and hardening guides for the service. Even more, there seems to be a lack of consistent methodology of evaluating any emerging digital services that lack a clear reference guide. While not originally planned, the threat model and the more expanded chapter 3 were added to address this gap.

The main finding of this study is that the security maturity of Azure OpenAI service is still extremely low compared to established cloud services. Where it not for extreme pressure from the business, it would not be considered ready for production. On a more positive note, even during the limited timespan of a few months of working on this study, the situation changed as features and security functionality were added. While the starting point of security functionality for Azure OpenAI was almost non-existent, it has gradually improved over the last quarter of 2023. Partial Azure Security Baseline is now available, and other security features such as built-in policy support are likely to follow soon. Some features that were added under the hood, will likely be addressed with proper tooling and UI support in the future.

References

Bubeck, S. et al., 2023. *Sparks of artificial general intelligence: Early experiments with gpt-4,* s.l.: arXiv preprint arXiv:2303.12712.

Leslie, D., 2019. *Understanding artificial intelligence ethics and safety: A guide for the responsible design and implementation of AI systems in the public sector.,* s.l.: The Alan Turing Institute.

Madiega, T., 2023. *Briefing EU Legislation in Progress: Artificial Intelligence Act.* [Online]
Available at: https://www.europarl.europa.eu/news/en/headlines/society/20230601STO93804/eu-ai-act-first-regulation-on-artificial-intelligence
[Accessed 7 January 2024].

Microsoft, 2023. *Data, privacy, and security for Azure OpenAI Service.* [Online]
Available at: https://learn.microsoft.com/en-us/legal/cognitive-services/openai/data-privacy
[Accessed 2 November 2023].

Microsoft, 2023. *GitHub.* [Online]
Available at: https://github.com/MicrosoftDocs/SecurityBenchmarks/commits/master/Azure%20Offer%20Security%20Baselines/3.0/azure-openai-azure-security-benchmark-v3-latest-security-baseline.xlsx
[Accessed 15 December 2023].

Microsoft, 2023. *Microsoft Learn, How do I get access to Azure OpenAI.* [Online]
Available at: https://learn.microsoft.com/en-us/azure/ai-services/openai/overview#how-do-i-get-access-to-azure-openai
[Accessed 7 January 2024].

Microsoft, 2023. *Microsoft Learn, Microsoft.CognitiveServices accounts resource definition.* [Online]
Available at: https://learn.microsoft.com/en-us/azure/templates/microsoft.cognitiveservices/accounts?pivots=deployment-

language-bicep#networkruleset
[Accessed 7 January 2024].

Microsoft, 2023. *Microsoft Learn, Overview of Microsoft cloud security benchmark.* [Online]
Available at: https://learn.microsoft.com/en-us/security/benchmark/azure/overview
[Accessed 6 January 2024].

Microsoft, 2023. *Microsoft Learn, Resource provider operations, Microsoft.CognitiveServices.* [Online]
Available at: https://learn.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations#microsoftcognitiveservices
[Accessed 7 January 2024].

Open AI, 2023. [Online]
Available at: https://openai.com/enterprise

OpenAI, 2023. *ChatGPT Enterprise: Compare ChatGPT plans.* [Online]
Available at: https://openai.com/enterprise
[Accessed 6 November 2023].

OpenAI, 2023. *GitHub, openai-python.* [Online]
Available at: https://github.com/openai/openai-python
[Accessed 7 January 2024].

OpenAI, 2023. *GPT-4 Technical report,* s.l.: arXiv:2303.08774v4.

OpenAI, 2023. *Trust Center.* [Online]
Available at: https://trust.openai.com/
[Accessed 29 October 2023].

Ots, K., 2021. *Azure Security Handbook.* 1 ed. Zürich: Apress.

OWASP, 2023. *OWASP Top 10 for Large Language Model Applications.* [Online]
Available at: https://owasp.org/www-project-top-10-for-large-language-model-applications/
[Accessed 30 November 2023].

Reuters, 2023. *ChatGPT sets record for fastest-growing user base - analyst note.* [Online]
Available at: https://www.reuters.com/technology/chatgpt-sets-record-fastest-

growing-user-base-analyst-note-2023-02-01/
[Accessed 7 January 2024].

State of California, E. D., 2023. *Executive Order N-12-23.* Sacramento, CA: State of California Executive Department.

The White House, 2023. *What is the Blueprint for an AI Bill of Rights?.* [Online] Available at: https://www.whitehouse.gov/ostp/ai-bill-of-rights/
[Accessed 7 January 2024].

Vaswani, A. et al., 2017. Attention is All you Need. *Advances in Neural Information Processing Systems,* Issue 30.

White House, 2023. *Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence.* [Online] Available at: https://www.whitehouse.gov/briefing-room/presidential-actions/2023/10/30/executive-order-on-the-safe-secure-and-trustworthy-development-and-use-of-artificial-intelligence/
[Accessed 7 January 2024].

Appendix 1: Threat Modeling Report

Threat Model Name: Reference threat model for Azure Open AI

Owner: Karl Ots

Description: This is a reference threat model for Azure Open AI (OAI) usage in the enterprise. This model aims to illustrate key threats regarding the usage of OAI, not to be a comprehensive description of all adjacent services. Note that this model looks at the end user scenario, where internal users are prompting an already trained model through a web interface. If the mitigation is marked as Needs Investigation, the mitigation requires you to analyze the threat and assign a proper mitigation based on your own risk appetite.

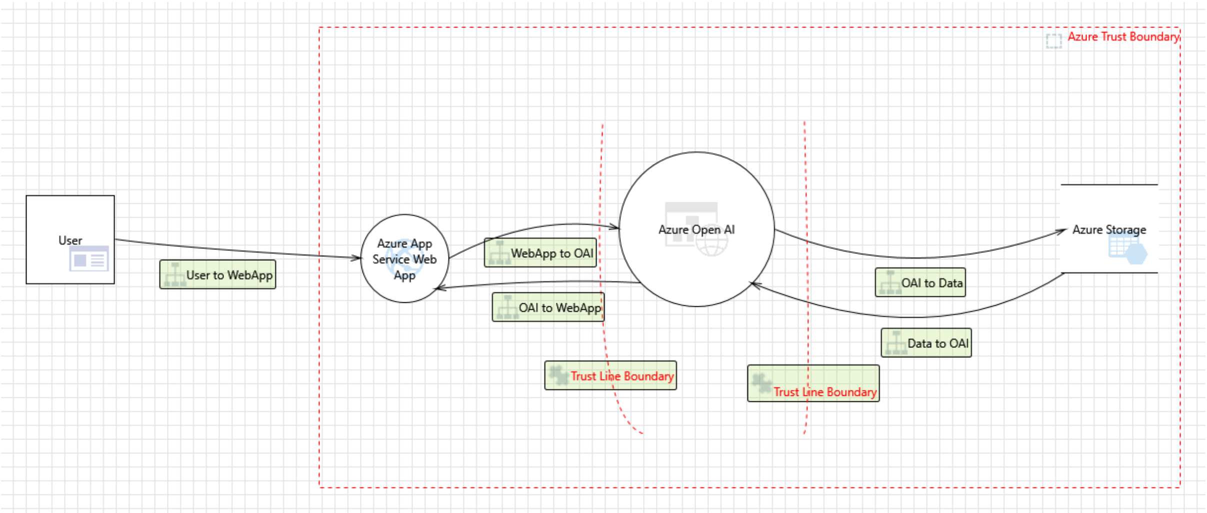
Assumptions: The main assumption is that core enterprise IT capabilities exists in the organization. Specifically, this applies to the cloud landing zone, as well as to enterprise security architecture. No industry-specific threats or requirements are considered. No specific assumptions on the sensitivity of the data or models used is considered.

External Dependencies: The following items are assumed to be implemented and operated according to best practices: Identity and Access management Logging architecture Change management Incident response.

Threat Model Summary:

Not Started	0
Not Applicable	0
Needs Investigation	16
Mitigation Implemented	17
Total	33
Total Migrated	0

Diagram: Reference Application



Reference Application Diagram Summary:

Not Started	0
Not Applicable	0
Needs Investigation	16
Mitigation Implemented	17
Total	33
Total Migrated	0

Threat(s) Not Associated With an Interaction:

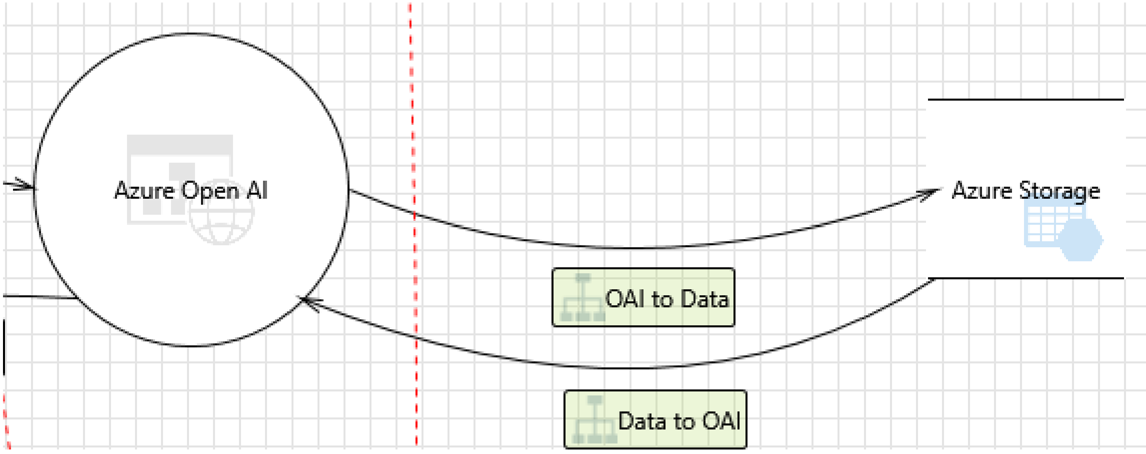
1. An adversary can gain unauthorized access to resources in an Azure subscription [State: Mitigation Implemented] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary can gain unauthorized access to resources in Azure subscription. The adversary can be either a disgruntled internal user, or someone who has stolen the credentials of an Azure subscription.
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal Identity and Access Management process.
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Enable fine-grained access management to Azure Subscription using RBAC.
SDL Phase:	Design

2. An adversary may spoof an Azure administrator and gain access to Azure subscription portal [State: Mitigation Implemented] [Priority: High]

Category:	Spoofing
Description:	An adversary may spoof an Azure administrator and gain access to Azure subscription portal if the administrator's credentials are compromised.
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal Identity and Access Management process.
Short Description:	Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address
Possible Mitigation(s):	Enable fine-grained access management to Azure Subscription using RBAC. Enable Azure Multi-Factor Authentication for Azure Administrators.
SDL Phase:	Design

Interaction: Data to OAI



3. An adversary can reverse weakly encrypted or hashed content [State: Needs Investigation] [Priority: Medium]

Category:	Information Disclosure
Description:	An adversary can reverse weakly encrypted or hashed content
Justification:	<no mitigation provided>
Short Description:	Information disclosure happens when the information can be read by an unauthorized party
Possible Mitigation(s):	Do not expose security details in error messages. Implement Default error handling page. Set Deployment Method to Retail in IIS. Use only approved symmetric block ciphers and key lengths. Use approved block cipher modes and initialization vectors for symmetric ciphers. Use approved asymmetric algorithms, key lengths, and padding. Use approved random number generators. Do not use symmetric stream ciphers. Use approved MAC/HMAC/keyed hash algorithms. Use only

	approved cryptographic hash functions. Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections.
SDL Phase:	Implementation

4. An adversary may gain access to sensitive data from log files [State: Mitigation Implemented] [Priority: High]

Category:	Information Disclosure
Description:	An adversary may gain access to sensitive data from log files
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal IR and monitoring process.
Short Description:	Information disclosure happens when the information can be read by an unauthorized party
Possible Mitigation(s):	Ensure that the application does not log sensitive user data. Ensure that Audit and Log Files have Restricted Access.
SDL Phase:	Implementation

5. Attacker can deny the malicious act and remove the attack foot prints leading to repudiation issues [State: Mitigation Implemented] [Priority: Medium]

Category:	Repudiation
Description:	Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal IR and monitoring process.
Short Description:	Repudiation threats involve an adversary denying that something happened
Possible Mitigation(s):	Ensure that auditing and logging is enforced on the application. Ensure that log rotation and separation are in place. Ensure that Audit and Log Files have Restricted Access. Ensure that User Management Events are Logged.
SDL Phase:	Implementation

6. An adversary can steal sensitive data like user credentials [State: Needs Investigation] [Priority: High]

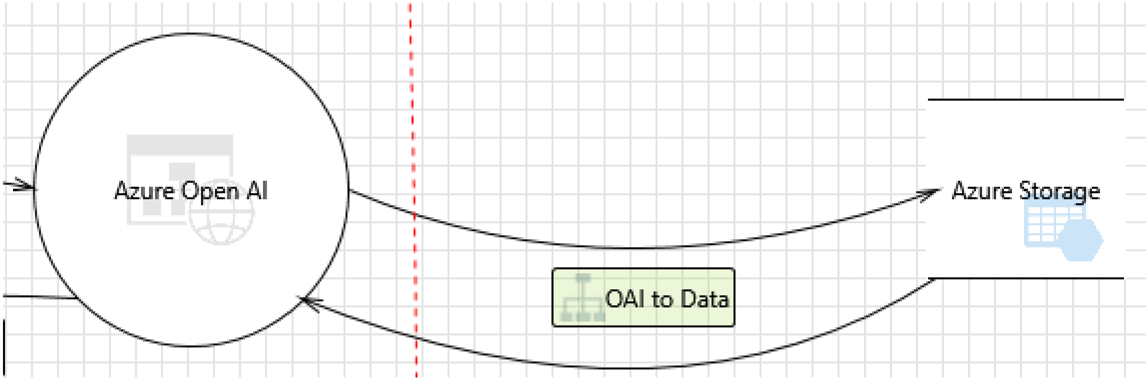
Category:	Spoofing
Description:	Attackers can exploit weaknesses in system to steal user credentials. Downstream and upstream components are often accessed by using credentials stored in configuration stores.

	Attackers may steal the upstream or downstream component credentials. Attackers may steal credentials if, Credentials are stored and sent in clear text, Weak input validation coupled with dynamic sql queries, Password retrieval mechanism are poor,
Justification:	<no mitigation provided>
Short Description:	Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address
Possible Mitigation(s):	Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs. Perform input validation and filtering on all string type Model properties. Validate all redirects within the application are closed or done safely. Enable step up or adaptive authentication. Implement forgot password functionalities securely. Ensure that password and account policy are implemented.
SDL Phase:	Implementation

7. An adversary may spoof Azure Storage and gain access to Web Application [State: Mitigation Implemented] [Priority: High]

Category:	Spoofing
Description:	If proper authentication is not in place, an adversary can spoof a source process or external entity and gain unauthorized access to the Web Application
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal Identity and Access Management process.
Short Description:	Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address
Possible Mitigation(s):	Consider using a standard authentication mechanism to authenticate to Web Application.
SDL Phase:	Design

Interaction: OAI to Data



8. An adversary can gain unauthorized access to Azure Storage due to weak access control restrictions [State: Mitigation Implemented] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary can gain unauthorized access to Azure Storage due to weak access control restrictions
Justification:	Control access from OAI to Storage using a system assigned managed identity.
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Refer to https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/use-your-data-securely
SDL Phase:	Implementation

9. An adversary may gain unauthorized access to Azure Storage account in a subscription [State: Mitigation Implemented] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary may gain unauthorized access to Azure Storage account in a subscription
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal Identity and Access Management process.
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Assign the appropriate Role-Based Access Control (RBAC) role to users, groups and applications at the right scope for the Azure Storage instance.
SDL Phase:	Implementation

10. An adversary can abuse poorly managed Azure Storage account access keys [State: Mitigation Implemented] [Priority: High]

Category:	Elevation of Privileges

Description:	An adversary can abuse poorly managed Azure Storage account access keys and gain unauthorized access to storage.
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal Identity and Access Management process.
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Ensure secure management and storage of Azure storage access keys. It is recommended to rotate storage access keys regularly, in accordance with organizational policies.
SDL Phase:	Implementation

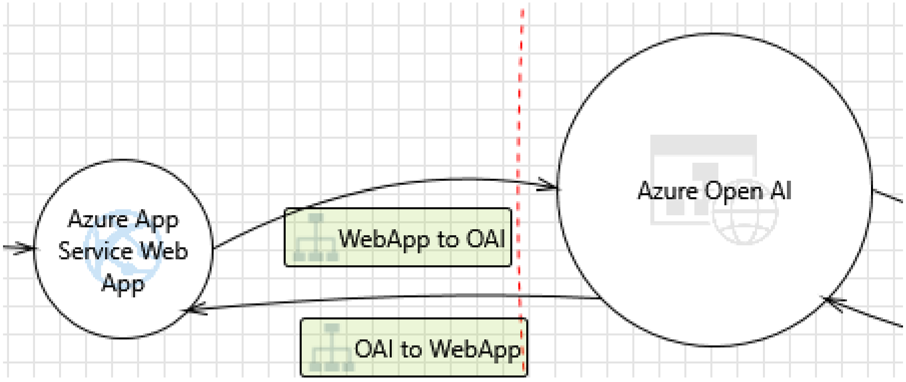
11. An adversary can deny actions on Azure Storage due to lack of auditing [State: Mitigation Implemented] [Priority: Medium]

Category:	Repudiation
Description:	Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system.
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an established internal IR and monitoring process.
Short Description:	Repudiation threats involve an adversary denying that something happened
Possible Mitigation(s):	Use Azure Storage Analytics to audit access of Azure Storage. If possible, audit the calls to the Azure Storage instance at the source of the call.
SDL Phase:	Implementation

12. An adversary can gain unauthorized access to Azure Storage due to weak CORS configuration [State: Needs Investigation] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary can gain unauthorized access to Azure Storage due to weak CORS configuration
Justification:	<no mitigation provided>
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Ensure that only specific, trusted origins are allowed.
SDL Phase:	Implementation

Interaction: OAI to WebApp



13. An adversary may block access to the application or API hosted on Azure App Service Web App through a denial of service attack [State: Needs Investigation] [Priority: High]

Category:	Denial of Service
Description:	An adversary may block access to the application or API hosted on Azure App Service Web App through a denial of service attack
Justification:	<no mitigation provided>
Short Description:	Denial of Service happens when the process or a datastore is not able to service incoming requests or perform up to spec
Possible Mitigation(s):	Leverage Azure API Management for managing and protecting APIs.
SDL Phase:	Implementation

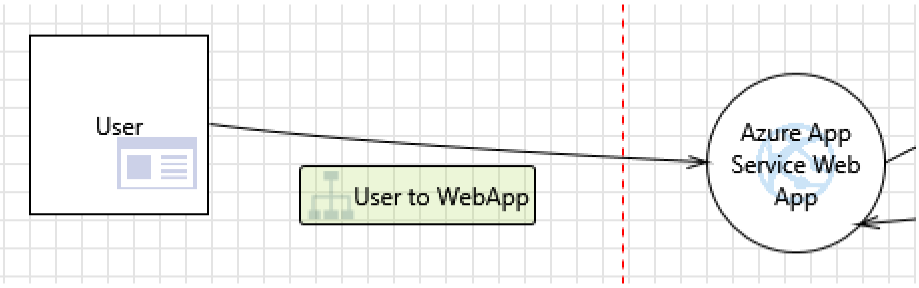
14. An adversary may gain long term persistent access to related resources through the compromise of an application identity [State: Needs Investigation] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary may gain long term persistent access to related resources through the compromise of an application identity
Justification:	<no mitigation provided>
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Store secrets in secret storage solutions where possible, and rotate secrets on a regular cadence. Use Managed Service Identity to create a managed app identity on Azure Active Directory and use it to access AAD-protected resources.
SDL Phase:	Implementation

15. An adversary may perform action(s) on behalf of another user due to lack of controls against cross domain requests [State: Needs Investigation] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary may perform action(s) on behalf of another user due to lack of controls against cross domain requests
Justification:	<no mitigation provided>
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Ensure that only trusted origins are allowed if CORS is being used.
SDL Phase:	Implementation

Interaction: User to WebApp



16. An adversary may block access to the application or API hosted on Azure App Service Web App through a denial of service attack [State: Needs Investigation] [Priority: High]

Category:	Denial of Service
Description:	An adversary may block access to the application or API hosted on Azure App Service Web App through a denial of service attack
Justification:	<no mitigation provided>
Short Description:	Denial of Service happens when the process or a datastore is not able to service incoming requests or perform up to spec
Possible Mitigation(s):	Network level denial of service mitigations are automatically enabled as part of the Azure platform (Basic Azure DDoS Protection).Implement application level throttling (e.g. per-user, per-session, per-API) to maintain service availability and protect against DoS attacks. Leverage Azure API Management for managing and protecting APIs.
SDL Phase:	Implementation

17. An adversary may gain long term persistent access to related resources through the compromise of an application identity [State: Needs Investigation] [Priority: High]

Category:	Elevation of Privileges
Description:	An adversary may gain long term persistent access to related resources through the compromise of an application identity
Justification:	<no mitigation provided>
Short Description:	A user subject gains increased capability or privilege by taking advantage of an implementation bug
Possible Mitigation(s):	Store secrets in secret storage solutions where possible, and rotate secrets on a regular cadence. Use Managed Service Identity to create a managed app identity on Azure Active Directory and use it to access AAD-protected resources.
SDL Phase:	Implementation

18. Attacker can steal user session cookies due to insecure cookie attributes [State: Needs Investigation] [Priority: High]

Category:	Information Disclosure
Description:	Attacker can steal user session cookies due to insecure cookie attributes
Justification:	<no mitigation provided>
Short Description:	Information disclosure happens when the information can be read by an unauthorized party
Possible Mitigation(s):	Applications available over HTTPS must use secure cookies.. All HTTP based applications should specify http only for cookie definition.
SDL Phase:	Implementation

19. An adversary can deny performing actions against Azure App Service Web App due to lack of auditing, leading to repudiation issues [State: Mitigation Implemented] [Priority: Medium]

Category:	Repudiation
Description:	An adversary can deny performing actions against Azure App Service Web App due to lack of auditing, leading to repudiation issues
Justification:	Deploy Azure Front Door with a web application firewalll and enforce it for the Web App using Access Restrictions.
Short Description:	Repudiation threats involve an adversary denying that something happened
Possible Mitigation(s):	Implement application level auditing and logging, especially for sensitive operations, like accessing secrets from secrets storage solutions. Other examples include user management

	events like successful and failed user logins, password resets, password changes, account lockouts and user registrations.
SDL Phase:	Implementation

20. An adversary can fingerprint an Azure web application or API by leveraging server header information [State: Mitigation Implemented] [Priority: Medium]

Category:	Information Disclosure
Description:	An adversary can fingerprint an Azure web application or API by leveraging server header information
Justification:	Follow the mitigation recommendations for your middleware and hosting model.
Short Description:	Information disclosure happens when the information can be read by an unauthorized party
Possible Mitigation(s):	Remove standard server headers to avoid fingerprinting. For example, for IIS applications you need to modify your web.config file.
SDL Phase:	Implementation

21. An adversary can read sensitive data by sniffing or intercepting traffic to Azure App Service Web App [State: Mitigation Implemented] [Priority: High]

Category:	Tampering
Description:	An adversary can read sensitive data by sniffing or intercepting traffic to Azure App Service Web App
Justification:	Follow the policies and procedures of your organization to enforce proper encryption in transit.
Short Description:	Tampering is the act of altering the bits. Tampering with a process involves changing bits in the running process. Similarly, Tampering with a data flow involves changing bits on the wire or between two running processes
Possible Mitigation(s):	Configure SSL certificate for custom domain in Azure App Service. Force all HTTP traffic to the app service to be over HTTPS by enabling the HTTPS only option on the instance. Enable HTTP Strict Transport Security (HSTS).
SDL Phase:	Implementation

22. An adversary may perform action(s) on behalf of another user due to lack of controls against cross domain requests [State: Needs Investigation] [Priority: High]

Category:	Elevation of Privileges

Description: An adversary may perform action(s) on behalf of another user due to lack of controls against cross domain requests

Justification: <no mitigation provided>

Short Description: A user subject gains increased capability or privilege by taking advantage of an implementation bug

Possible Mitigation(s): Ensure that only trusted origins are allowed if CORS is being used.

SDL Phase: Implementation

23. An adversary may be able to perform action(s) on behalf of another user due to lack of controls against cross domain requests [State: Needs Investigation] [Priority: High]

Category: Spoofing

Description: An adversary may be able to perform action(s) on behalf of another user due to lack of controls against cross domain requests

Justification: <no mitigation provided>

Short Description: Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address

Possible Mitigation(s): Ensure that authenticated pages incorporate UI Redressing or clickjacking defences. Mitigate against Cross-Site Request Forgery (CSRF) attacks.

SDL Phase: Implementation

24. An adversary can gain unauthorized access to resources in an Azure subscription [State: Mitigation Implemented] [Priority: High]

Category: Elevation of Privileges

Description: An adversary can gain unauthorized access to resources in Azure subscription. The adversary can be either a disgruntled internal user, or someone who has stolen the credentials of an Azure subscription.

Justification: Mitigated by following best WAF and CAF practices, as well as relying on an establisehd internal Identity and Access Management process.

Short Description: A user subject gains increased capability or privilege by taking advantage of an implementation bug

Possible Mitigation(s): Enable fine-grained access management to Azure Subscription using RBAC.

SDL Phase: Design

25. An adversary may spoof an Azure administrator and gain access to Azure subscription portal [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

Description: An adversary may spoof an Azure administrator and gain access to Azure subscription portal if the administrator's credentials are compromised.

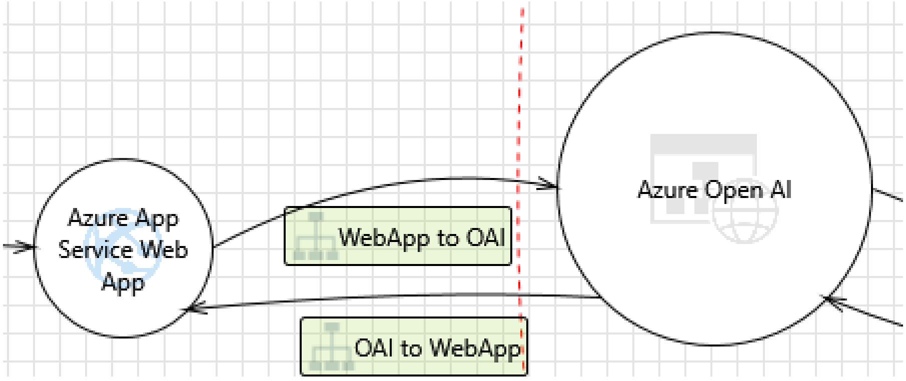
Justification: Mitigated by following best WAF and CAF practices, as well as relying on an establisehd internal Identity and Access Management process.

Short Description: Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address

Possible Mitigation(s): Enable fine-grained access management to Azure Subscription using RBAC. Enable Azure Multi-Factor Authentication for Azure Administrators.

SDL Phase: Design

Interaction: WebApp to OAI



26. An adversary can reverse weakly encrypted or hashed content [State: Needs Investigation] [Priority: Medium]

Category: Information Disclosure

Description: An adversary can reverse weakly encrypted or hashed content

Justification: <no mitigation provided>

Short Description: Information disclosure happens when the information can be read by an unauthorized party

Possible Mitigation(s): Do not expose security details in error messages. Implement Default error handling page. Set Deployment Method to Retail in IIS. Use only approved symmetric block ciphers and key lengths.Use approved block cipher modes and initialization vectors for symmetric ciphers. Use approved asymmetric algorithms, key lengths, and padding. Use approved random number generators. Do not use symmetric stream ciphers. Use approved MAC/HMAC/keyed hash algorithms. Use only

	approved cryptographic hash functions. Verify X.509 certificates used to authenticate SSL, TLS, and DTLS connections.
SDL Phase:	Implementation

27. An adversary may gain access to sensitive information [State: Mitigation Implemented] [Priority: High]

Category:	Information Disclosure
Description:	An adversary may gain access to sensitive data from log files
Justification:	Mitigated by following best WAF and CAF practices, as well as relying on an establisehd internal IR and monitoring process.
Short Description:	Information disclosure happens when the information can be read by an unauthorized party
Possible Mitigation(s):	Ensure that the application does not log sensitive user data. Ensure that Audit and Log Files have Restricted Access.
SDL Phase:	Implementation

28. An adversary can gain access to sensitive information through error messages [State: Needs Investigation] [Priority: High]

Category:	Information Disclosure
Description:	LLM applications have the potential to reveal sensitive information, proprietary algorithms, or other confidential details through their output. This can result in unauthorized access to sensitive data, intellectual property, privacy violations, and other security breaches. It is important for consumers of LLM applications to be aware of how to safely interact with LLMs and identify the risks associated with unintentionally inputting sensitive data that may be returned by the LLM in output elsewhere.
Justification:	<no mitigation provided>
Short Description:	Information disclosure happens when the information can be read by an unauthorized party
Possible Mitigation(s):	Integrate adequate data sanitization and scrubbing techniques to prevent user data from entering the training model data. Implement robust input validation and sanitization methods to identify and filter out potential malicious inputs to prevent the model from being poisoned.
SDL Phase:	Implementation

29. Attacker can deny the malicious act and remove the attack foot prints leading to repudiation issues [State: Needs Investigation] [Priority: Medium]

Category:	Repudiation

Description:	Proper logging of all security events and user actions builds traceability in a system and denies any possible repudiation issues. In the absence of proper auditing and logging controls, it would become impossible to implement any accountability in a system
Justification:	<no mitigation provided>
Short Description:	Repudiation threats involve an adversary denying that something happened
Possible Mitigation(s):	Ensure that auditing and logging is enforced on the application. Ensure that log rotation and separation are in place. Ensure that Audit and Log Files have Restricted Access. Ensure that User Management Events are Logged.
SDL Phase:	Implementation

30. An adversary can steal sensitive data like user credentials [State: Needs Investigation] [Priority: High]

Category:	Spoofing
Description:	Attackers can exploit weaknesses in system to steal user credentials. Downstream and upstream components are often accessed by using credentials stored in configuration stores. Attackers may steal the upstream or downstream component credentials. Attackers may steal credentials if, Credentials are stored and sent in clear text, Weak input validation coupled with dynamic sql queries, Password retrieval mechanism are poor,
Justification:	<no mitigation provided>
Short Description:	Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address
Possible Mitigation(s):	Explicitly disable the autocomplete HTML attribute in sensitive forms and inputs. Perform input validation and filtering on all string type Model properties. Validate all redirects within the application are closed or done safely. Enable step up or adaptive authentication. Implement forgot password functionalities securely. Ensure that password and account policy are implemented. Implement input validation on all string type parameters accepted by Controller methods.
SDL Phase:	Implementation

31. An adversary may spoof Azure App Service Web App and gain access to Web Application [State: Mitigation Implemented] [Priority: High]

Category:	Spoofing
Description:	If proper authentication is not in place, an adversary can spoof a source process or external entity and gain unauthorized access to the Web Application

Justification: Disable local access keys and use Managed Identities.

Short Description: Spoofing is when a process or entity is something other than its claimed identity. Examples include substituting a process, a file, website or a network address

Possible Mitigation(s): Consider using a standard authentication mechanism to authenticate to Web Application.

SDL Phase: Design

32. An adversary can gain access to sensitive data by performing a prompt injection [State: Needs Investigation] [Priority: High]

Category: Tampering

Description: Prompt injections involve bypassing filters or manipulating the LLM using carefully crafted prompts that make the model ignore previous instructions or perform unintended actions.

Justification: <no mitigation provided>

Short Description: Tampering is the act of altering the bits. Tampering with a process involves changing bits in the running process. Similarly, Tampering with a data flow involves changing bits on the wire or between two running processes

Possible Mitigation(s): Implement strict input validation and sanitization for user-provided prompts. Use context-aware filtering and output encoding to prevent prompt manipulation. Regularly update and fine-tune the LLM to improve its understanding of malicious inputs and edge cases. Monitor and log LLM interactions to detect and analyze potential prompt injection attempts.

SDL Phase: Implementation

33. An adversary can gain access to sensitive data stored in Web App's config files [State: Mitigation Implemented] [Priority: High]

Category: Tampering

Description: An adversary can gain access to the config files and if sensitive data is stored in it, it would be compromised. An attacker queries the model API using carefully crafted inputs and prompt injection techniques to collect a sufficient number of outputs to create a shadow model.

Justification: Store sensitive configuration data in Azure Key Vault or Azure App Configuration.

Short Description: Tampering is the act of altering the bits. Tampering with a process involves changing bits in the running process. Similarly, Tampering with a data flow involves changing bits on the wire or between two running processes

Possible Mitigation(s): Store sensitive configuration data in Azure Key Vault or Azure App Configuration.

SDL Phase: Implementation

Appendix 2: Azure Policy initiatives

Two custom Azure policy initiatives were created as part of this paper and published to https://github.com/karlgots/openai.

- Policy initiative definition OAIPolicyInitiative.json, with policies covering Azure OpenAI service.

- Policy initiative definition RefAppPolicyInitiative.json, with policies cocering the Azure services in the reference application.