Google Cloud

Best Practices for Securely Deploying AI on Google Cloud

Anton Chuvakin, Luis Urena

Office of the CISO, Google Cloud

Google Cloud

Google Cloud

Introduction

The rising adoption of cloud platforms including Google Cloud facilitates rapid innovation in artificial intelligence (AI). In addition to the canonical threats that any cloud app needs to protect against, AI systems must be secured against the unique threats they face.

This paper explores the shared responsibility model in securing AI workloads on Google Cloud, emphasizing what Google provides and the crucial steps customers must take. We review essential security domains, address model-specific concerns like prompt injection, and outline proactive measures for making AI systems more resilient against current and future risks. This practical approach is rooted in Google's Secure AI Framework (SAIF), a conceptual framework for securing AI systems, and our understanding of differences and similarities between security AI and traditional systems.

When you are deploying AI workloads on Google Cloud or another modern cloud environment, certain security best practices are derived from traditional cloud security capabilities. For example, we encourage customers to focus on identity management, network controls, and access management, which are similar to what you would deploy in other environments.

However, there are also other controls–including data and network security, security of training data, and filtering inputs and outputs (also relevant for safety, privacy and acceptable use)—that are different and use cloud-native technologies, such as VPC Service Controls (VPC SC) and Sensitive Data Protection (SDP). With these tools, you can more securely deploy AI workloads to the cloud.

Scenario

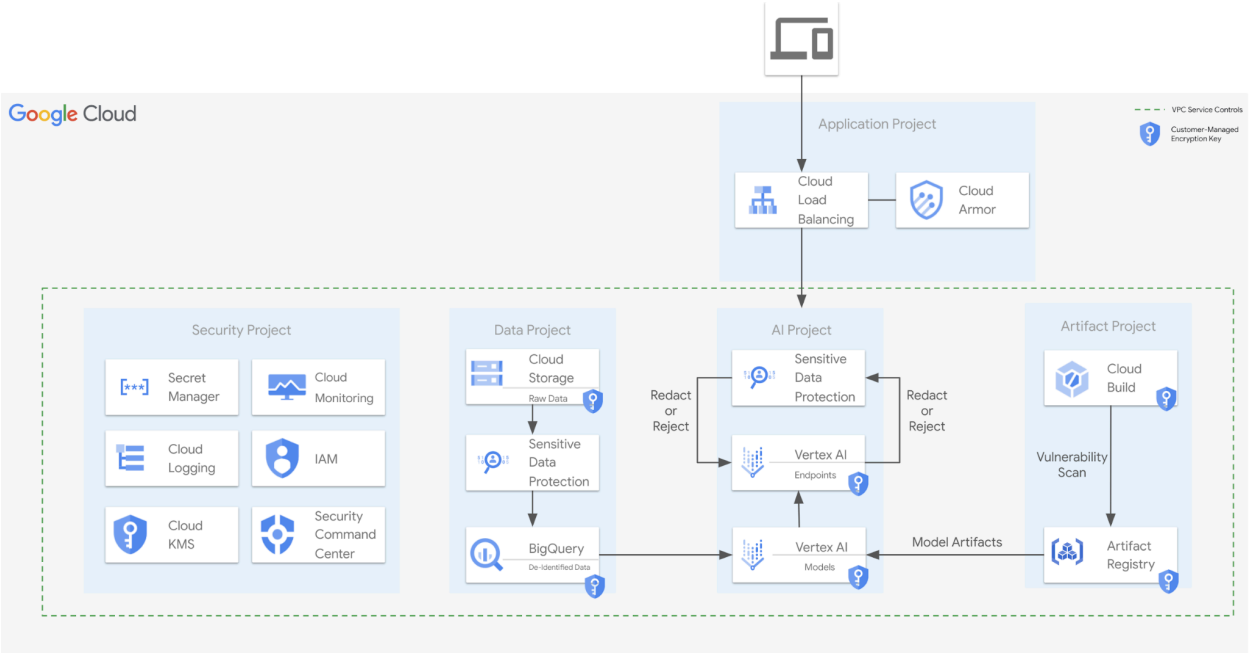
For every organization, the decision to leverage the power of AI hinges on a myriad of questions, including:

- How can I use AI to improve, scale, and accelerate my business?
- What tools and processes are most effective for my data science team?
- How much (and what type of) data does the team need to deliver a solution? Where does the data come from?
- What kind of risks or threats does AI introduce to my organization?
- **How can I help my organization harness the power of AI while minimizing those risks?**

This whitepaper seeks to address the final question, with a focus on helping your organization harness the power of AI while minimizing the security risks and related threats that might potentially impact business operations.

For the purposes of this paper, we assume your organization has an AI/ML strategy, as well as a team of data scientists ready to build and train models on Vertex AI, Google Cloud's unified AI platform. To augment built-in security features, customers can deploy other services like Security Command Center, Google Cloud's multicloud security and risk management solution, which works with organization policies to provide near real-time detection of changes to policies and to AI resource configurations.

To get started, we propose the following architecture to represent a sample application. This includes a minimum set of security controls necessary to allow your data scientists to gain immediate benefit from Vertex AI while ensuring your security team can feel confident about maintaining a strong security posture.



Specifically, this architecture helps you:

- Govern and manage external traffic with Cloud Armor.
- Leverage SDP to identify sensitive data and decide whether to accept or reject input and output.
- Encrypt sensitive data and container images using Cloud Key Management Service (KMS).
- Protect your backend infrastructure using VPC Service Controls.
- Maintain strong identity and access management with Cloud IAM and Organization Policies.
- Support strong incident response with logging, monitoring, and alerting.

Each of these security controls act as different layers of defense to help address threats relevant to your AI workload.

AI Threats

While AI systems inherit traditional cybersecurity risks, they also face a unique spectrum of threats. The specific impact and mitigations depend on how the business is using AI, and the sensitivity of the data involved. Importantly, risk management can vary depending on who is developing the AI application—you or a third party. Broadly, AI threats typically center on four categories: model, application, infrastructure, and data. We've summarized some common security risks and relevant threats to most AI deployments across these categories, grouped by their primary impact:

Model Manipulation and Evasion

- **Prompt Injection:** Crafting malicious inputs to language models that exploit vulnerabilities in the model or lead it to generate harmful content or inaccurate responses.
- **Model Evasion**: Adversarial attacks involving subtle perturbations of input data (e.g., pixel changes in an image) designed to cause misclassifications or unexpected behavior.
- **Insecure Model Output**: If output filtering or monitoring are inadequate, AI models could generate misleading, offensive, or unsafe content.

Application Compromise

- **Denial of ML Service:** Attacks aimed at overloading the AI system or manipulating its resources can render the service unavailable, disrupting operations that rely on AI outputs.

- **Insecure Integrated System**: Vulnerabilities in software interacting with AI models can be leveraged by attackers to gain access, introduce malicious code, or compromise system operations.
- **Model Reverse Engineering**: Attempts to extract intellectual property or proprietary algorithms by analyzing a model's inputs, outputs, or behavior.
- **Unauthorized Model Actions**: Gaining control of an AI model to take actions beyond its intended use, potentially with harmful consequences.

Infrastructure for AI

- **Model Backdooring**: Injecting a hidden trigger during model training that allows an attacker to manipulate the model's behavior with specific inputs later.
- **Model Exfiltration**: Unauthorized appropriation of an AI model, for replicating functionality or to extract intellectual property.
- **Model Serving Compromise**: Attacks manipulating the serving infrastructure (where the model makes predictions) to insert unauthorized models or corrupt responses.

Data Exposure

- **Sensitive Data Leakage:** Exposure of confidential training data, or data extracted from the model through clever use of inputs and observing responses.
- **Inferred Sensitive Data:** AI models may unintentionally reveal sensitive information not overtly contained in training data, derived through patterns or correlations discovered by the model or introduced with frameworks like retrieval-augmented generation (RAG) that enrich fresh or proprietary data into LLM prompts to deliver relevant and accurate information.
- **Data and Model Poisoning**: Injecting malicious data during training or retraining to degrade model performance, skew results towards a specific outcome, or create hidden backdoors (won't apply for frozen models such as those used by Vertex AI).
- **Excessive Data Retention**: Unnecessary storage of data used to train AI models, creating both ethical and legal risks.

Threat landscapes evolve rapidly within the AI domain. Staying vigilant and informed about emerging threats is crucial for robust AI security. You can learn more about the latest cyber threats from Mandiant and other sources.

Google Cloud

AI Security and Shared Responsibility

At Google Cloud, we are committed to helping enterprises develop effective AI risk management strategies to be able to use the full potential of generative AI (gen AI). We believe that shared responsibility is a core component of that effort and a critical concept in securing AI workloads on Google Cloud. This means that both Google and the customer play essential roles in safeguarding AI systems. In the use cases where organizations develop their own applications (as opposed to using AI applications developed by a third party,) the core responsibilities between Google and customer include the following:

Google's Responsibilities include:

- **Infrastructure Security:** Google is responsible for the physical and logical security of the underlying Google Cloud infrastructure, including data centers, networks, and operating systems.
- **Platform Security:** Google Cloud is built with security in mind, offering features like encryption, vulnerability management, and identity and access management (IAM) controls.
- **Platform Compliance:** Google adheres to rigorous compliance standards, such as FedRAMP, HIPAA, and PCI DSS, to ensure data privacy and security.

Customer Responsibilities include:

- **Identity and Access Management (IAM):** Customers are responsible for implementing IAM best practices to control access to AI resources and data. This includes using least privilege, service accounts, and regularly reviewing and updating IAM policies.
- **Network Security:** Customers should configure network security controls to protect AI workloads, such as VPC segmentation, firewall rules, web application firewall protection, and API management.
- **Application Security:** Securing the applications that interact with AI models is crucial. This involves input validation, secure coding practices, and regular security testing.
- **Data Security and Governance:** Customers must implement robust data security measures to protect sensitive data used in AI training and operation. This includes encryption, data loss prevention (DLP), and proper data lifecycle management.
- **Logging and Monitoring:** Continuous monitoring of AI workloads is essential for detecting and responding to security incidents. Customers should leverage Google Cloud logging, monitoring, and SIEM tools to gain visibility into their AI systems.
- **Incident Response:** Having a well-defined incident response plan is crucial for minimizing the impact of security breaches. This plan should outline steps for identifying, containing, and remediating security incidents.

By understanding and fulfilling their respective responsibilities, Google and customers can work together to build secure and trustworthy AI workloads on Google Cloud.

Google's Secure Home for AI

Over the years, Google has adopted an open and collaborative approach to cybersecurity. SAIF is designed to help mitigate specific AI system risks like data poisoning, prompt injection, model stealing, and sensitive data leakage. Gen AI represents a shift in both what and how Google Cloud customers build innovative technologies. Google has unique strengths in security and data protection to ensure gen AI's security.

Our approach includes combining frontline intelligence, expertise, and innovation with a commitment to share threat information with others to help respond to—and prevent—cyber attacks. Due to its decades of investment, Google also has unique strengths in security and data protection: platform and container security and posture management, content abuse, security intelligence and operations, developer, and both AI and human expert assistance.

Customer Best Practices

We encourage you to review and implement Google Cloud Security best practices as outlined in the Enterprise Foundations Blueprint. The Enterprise Foundations Blueprint is a well-designed foundation that enables consistent governance, security controls, scale, visibility, and access to shared services across all workloads in your Google Cloud environment. After you deploy the controls and governance described in that document, you should apply the set of AI Security best practices outlined below. This set of best practices is focused on mitigating threats that apply specifically to AI systems.

Consistent with the four broad categories described above, we focus on securing model, application, infrastructure, and data.

Model Security

While securing applications and infrastructure is a vital component of AI security, prioritizing the security of the AI models themselves must not be overlooked. AI models are the core decision-making engines of the system, making them prime targets for sophisticated attacks.

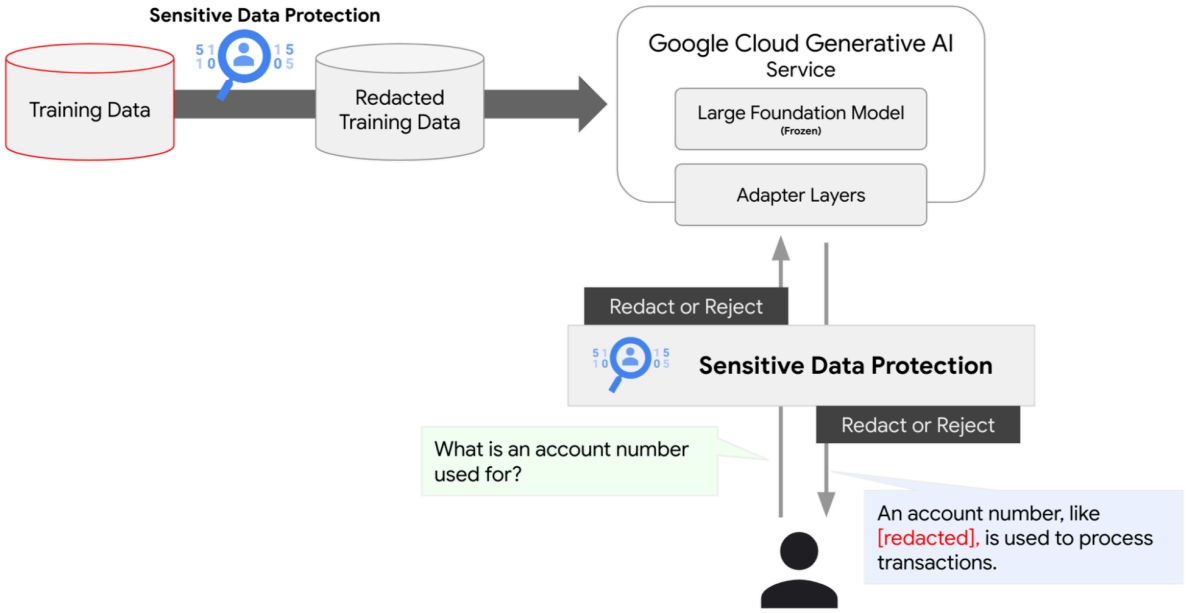
In the scenario we describe, the customer is not training or refining the model (see Scenario section) hence model security is reduced to matching the model to the business problem to be solved and protecting the model from unauthorized access, modification, and disclosure.

Application Security

Protection In-line Prompt and Response

Protecting training pipelines is important, but it is only part of the defense offered in SDP. Since gen AI models take unstructured prompts from users and generate new, possibly unseen responses, you may also want to protect sensitive data in-line. Many known prompt-injection attacks have been seen in the wild. The main goal of these attacks is to manipulate the model into sharing unintended information.

While there are multiple ways to protect against prompt injection, SDP can provide a data-centric security control on data going to and from gen AI foundation models by scanning the input prompt and generated response to ensure that sensitive elements are identified or removed.



While many attackers may attempt prompt injection techniques in an attempt to exfiltrate sensitive data, others may want to manipulate your models to generate content that may be offensive, misleading, or dangerous. Content processed through the AI is assessed against a

list of safety attributes, which include "harmful categories" and topics that can be considered sensitive. By default, these APIs block unsafe content based on a list of safety attributes and their configured blocking thresholds. You may choose to enforce a different threshold for each safety attribute, allowing you to take control over the type of content your AI application accepts or generates.

You should also consider using text embeddings as the third-layer of defense to protect against model manipulation and evasion. A text embedding is a vector representation of text, and they are used in many ways to find similar items. When you create text embeddings, you get vector representations of natural text as arrays of floating point numbers—all of your input text is assigned a numerical representation. By comparing the numerical distance between the vector representations of two pieces of text, an application can determine the similarity between the text or the objects represented by the text.

This becomes quite useful in the security context: although your team has taken great effort and care to consider the ways in which an attacker can craft a prompt to manipulate your model, they can't identify every possible attempt. Therefore, text embeddings can determine that a new prompt is similar to a known malicious prompt—and defenders can use this information to enforce your security guardrails.

Infrastructure Security

A fundamental pillar of AI security lies in safeguarding the underlying infrastructure upon which these systems function. Compromising a system's infrastructure—the hardware, networks, and software it relies on—can expose sensitive AI models, training data, and the overall system to harmful attacks and manipulation. Therefore, the AI infrastructure should ensure that unauthorized users are unable to establish unauthorized access to the model, are prohibited from unauthorized appropriation of the model, and cannot insert unauthorized models or corrupt responses.

Software Supply Chain

In the example architecture, we are securing the software supply chain with Cloud Build and Artifact Registry to create and find vulnerabilities in custom containers. Custom containers allow you to run your training job while using ML frameworks, non-ML dependencies, libraries, and binaries that are otherwise not supported by Vertex AI.

When you build an image with Cloud Build, the image's build provenance is automatically recorded. Build provenance is a collection of verifiable data and includes details such as the digests of the built images, the input source locations, the build arguments, and the build duration. You can leverage this build provenance to confirm that build artifacts are being generated from trusted sources and builders and ensure that provenance metadata describing your build process is complete and authentic.

In fact, you can modify your build configuration file such that the build fails if Cloud Build does not generate this provenance metadata. You may also choose to encrypt the build-time persistent disk with a unique ephemeral Customer-Managed Encryption Key (CMEK) that is generated for each build. Once a build starts, the key is accessible only to the build processes requiring it for up to 24 hours. Then, the key is wiped from memory and destroyed.

If the custom image passes the build, it will be pushed to Artifact Registry, a service that allows you to centrally store artifacts and build dependencies as part of an integrated Google Cloud experience. We recommend that you enable the Container Analysis API before pushing any images to Artifact Registry. The Container Analysis API initiates an automatic vulnerability scan when images are pushed to Artifact Registry. The vulnerability information is continuously updated when new vulnerabilities are discovered and are available in Security Command Center.

Serving Infrastructure

Google Cloud's web application firewall service, Cloud Armor, provides WAF and anti-DDoS capabilities, protecting applications against layer 3, 4, and layer 7 attacks, the Open Web Application Security Project (OWASP) Top 10, and sophisticated application exploits.

A Google Cloud Application Load Balancer is the first entry point for traffic attempting to reach your application's API. As the traffic reaches the External Application Load Balancer, Cloud Armor is immediately assessing the traffic to detect and mitigate network attacks. This assessment is based on preconfigured and custom security policies that you can apply to allow, deny, rate-limit, or redirect requests before traffic reaches your API and backend services. If Cloud Armor determines that the traffic is legitimate, the user will be able to access the application.

We recommend you contain the backend infrastructure within a VPC Service Controls Perimeter, to help you mitigate exfiltration risks by isolating multi-tenant services. The VPC Service Controls perimeter denies access to restricted Google Cloud services from traffic that originates outside the perimeter, which includes the console, developer workstations, and the foundation pipeline used to deploy resources. Before the perimeter is created, you must

design ingress and egress rules exceptions to the perimeter that allow the access paths that you intend. You should also:

- use dry run mode to identify API access violations without interruption to the applications before enforcing the perimeter.
- design a process to consistently add new projects to the perimeter.
- design a process to design exceptions when developers have a new use case that is denied by your current perimeter configuration.

This layered approach, protecting the backend infrastructure with VPC Service Controls and combining a secure software supply chain with threat detection and prevention, allows you to detect and remediate vulnerabilities that may lead to model backdooring, model exfiltration, or model serving compromise attacks.

Data Security

Building an AI/ML system requires a large corpus of data to appropriately train models and oftentimes the data may be considered sensitive. Securing that data appropriately becomes of paramount importance, and we can protect against data leakage risks with encryption and anonymization techniques.

Encryption at Rest

All data stored within Google Cloud is encrypted at rest using the same hardened key management systems that Google uses for our own encrypted data. These key management systems provide strict key access controls and auditing, and encrypt user data at rest using AES-256 encryption standards. No setup, configuration, or management is required. Default encryption is the best choice if your organization doesn't have specific requirements related to compliance or locality of cryptographic material.

However, you may also use CMEK in the Cloud Key Management Service (Cloud KMS). Cloud KMS is a service that lets you manage encryption keys. It can generate, use, rotate, and destroy AES-256, RSA 2048, RSA 3072, RSA 4096, EC P256, and EC P384 encryption keys.

Google Cloud

Customer-managed encryption keys allow you to have more control around key generation, key rotation frequency, and key location. With added control comes added responsibility; therefore, we recommend you appropriately manage and control your keys by:

- Applying fine-grained access controls with predefined roles such as Cloud KMS Admin, CryptoKey Decrypter and Decrypter, and KMS Viewer.
- Setting Organization Policies to ensure:
    - CMEK is required for all resources at the folder, project, or organization-level.
    - Only certain keys can be used to encrypt your AI workloads.
    - Key versions are disabled before they are scheduled for destruction.
    - Keys are protected from accidental deletion by setting a minimum "scheduled for destruction" duration.
- Hosting your encryption keys in a centralized Google Cloud Project.

Because of these added responsibilities, we recommend that you evaluate whether the default encryption is sufficient, or whether you have a compliance requirement that you must use Cloud KMS to manage keys yourself. For more information, see how to meet compliance requirements for encryption at rest.

Sensitive Data Protection

Sensitive Data Protection includes more than 150 built-in infoTypes to help quickly identify sensitive data elements like names, personal identifiers, financial data, medical context, or demographic data. You can identify these elements to choose which records to remove from pipelines or leverage inline transformation to obscure only the sensitive elements while retaining the surrounding context. This enables you to reduce risk while preserving the utility of your data. Inline transformation can be used when preparing training or tuning data for AI models and additionally can protect AI generated prompts and responses in real-time.

Protection for Data Preparation

Customers frequently use their own data to create datasets to train custom AI models, such as when they deploy an AI model on prediction endpoints. In another common example, customers use their own data to fine-tune a LLM to enhance model outputs and better advance relevant business priorities.

The tuning process described above uses customer-specific datasets and creates parameters that are then used at inference time; the parameters reside in front of the "frozen" foundation

model, inside the user's project. To ensure that these datasets do not include sensitive data, we recommend that your organization use the Sensitive Data Protection service to scan the data that was used to create the datasets. Similarly, this method can be used for Vertex AI Search to ensure uploaded data does not include sensitive information.

Logging, Detection, and Response

Logging and Monitoring

Logging on Google Cloud is achieved with Cloud Logging, a real-time log-management system with storage, search, analysis, and monitoring support. By default, Cloud Logging captures Admin Activity logs, System Event, and Policy Denied logs, providing you with insight on resource and metadata configuration modifications and unsuccessful attempts to access resources. Alongside these default logs, we recommend enabling Access Transparency so that you can capture logs that allow you to verify that Google personnel haven't made an error while carrying out your instructions and track compliance with legal or regulatory obligations.

It's important to note that Google Cloud does not log your end-user's interactions with your AI applications. Logging these interactions must be done at the application layer. We recommend that you use the Cloud Logging client library to log the end-user's prompts and the AI application's responses.

You should aggregate and store logs relevant for security and auditing into a centralized project for long-term retention, analysis, and export to external systems. This will require that you create a log bucket, and the log bucket will live in the centralized project. It's important to note that these log buckets only store logs for 30 days unless you increase the retention period. We recommend adjusting the retention period to align with your internal and external compliance requirements.

Detection for Vertex AI

The security posture feature of Security Command Center includes predefined postures that help you secure Vertex AI workloads. Such policies include, but are not limited to, restricting public IP addresses and disabling file downloads, root access, and the Vertex AI Workbench terminal. These postures include detective controls using Security Health Analytics that will, for example, identify when CMEK are disabled across Vertex AI models, datasets, endpoints, training pipelines, and data labeling and custom jobs.

Google Cloud

Detection for AI Workloads on Google Kubernetes Engine

If you're running your AI workload on Google Kubernetes Engine (GKE), you may have a broader attack surface due to the security responsibilities associated with running Infrastructure as a Service (IaaS) services. Nevertheless, Container Threat Detection, a Security Command Center feature, continuously monitors the state of Container-Optimized OS node images. The service evaluates changes and remote access attempts to detect runtime attacks in near-real time. These attacks include, but are not limited to: execution of malicious binaries, libraries, URLs, and scripts; reverse shells; and unexpected child shells.

Augmenting Detection

With multiple security and privacy controls in place, organizations need a centralized location where they can prevent, detect, and respond to threats. Security Command Center provides security posture controls and threat alerts that allow you to define, evaluate, and manage the security status of your AI workloads.

As you rely on Cloud Sensitive Data Protection to prepare data and protect in-line prompt and response, you may want to understand the type of data you're processing. We recommend you send your Cloud Sensitive Data Protection findings to Security Command Center to enrich your findings. When this awareness is fed into Security Command Center, your team can more easily prioritize the findings that are driving greater security and compliance risk, and help make informed decisions to better address those specific issues.

Likewise, we recommend you leverage the predefined postures for VPC Service Controls, given its importance as a preventative measure for data exfiltration. These predefined postures can help you ensure that network-based misconfigurations and threats, such as VM nested virtualization, disabled logs, and network routing changes, are proactively addressed.

Alongside Container Threat Detection, these predefined and custom security postures, can help you detect and mitigate any drift from your defined benchmark. Security Command Center reports instances of drift as findings that you can review, filter, and respond to.

Google Cloud

Building the Future Together with AI

As the world focuses on the potential of AI—and governments and industry work on a regulatory approach to ensure AI is safe and secure—we believe that AI represents an inflection point for digital security. To harness this potential, we encourage your organization to implement the best practices outlined in this whitepaper to securely deploy AI capabilities on Google Cloud. We also recommend reviewing the following resources for additional information:

- Introducing Google's Secure AI Framework
- Securing AI: Similar or Different?
- Gen AI governance: 10 tips to level up your AI program
- Why Red Teams Play a Central Role in Helping Organizations Secure AI Systems
- How Sensitive Data Protection can help secure generative AI workloads

As we develop and deploy new capabilities, we'll continue to share research and explore methods that help to utilize AI in a secure way. This is an important part of our commitment to make Google part of your security team and to ensure that this profoundly helpful technology works for everyone.

Google Cloud

Appendix A: Best Practices Checklist

Building secure AI technology is similar to building other secure software products. Both should follow best practices designed to help developers achieve their goals faster, more efficiently, and with fewer mistakes that can turn into risks down the development chain.

Please refer to the linked best practices for each subject category. These include best practices for model development, application security, infrastructure, and data management.

Model

- [x] Define the specific business problem AI will solve and assess the risks associated with that problem
- [x] Protect models from unauthorized access, modification, and disclosure
- [x] Use secure model development practices based on security software development and software supply chain security

Application

- [x] Implement input and output filtering and validation, designed to detect adversarial examples and prevent malicious manipulations
- [x] Enforce content safety policies to mitigate the risk of AI-generated harmful content
- [x] Run red team tests for security and for AI
- [x] Create an incident response plan for AI-specific issues, covering potential failures or malicious uses

Infrastructure

- [x] Harmonize platform level controls to ensure consistent security across the organization including AI workloads and training environments
- [x] Align with strong identity and access management (IAM) practices, including principles of least privilege
- [x] Collect and retain logs from the above components
- [x] Develop detection content for attacks and abuses specific to AI

Data

- [x] Include AI in data governance and management processes
- [x] Use strong encryption for stored data and data in transit
- [x] Expand data governance processes for AI datasets, including data labeling, and provenance tracking
- [x] Assess business risk to data used for training, tuning, and other context around the system