Husky AI

Owner:
Reviewer:
Contributors: Imported from TM-BOM
Date Generated: Tue Oct 07 2025

Executive Summary

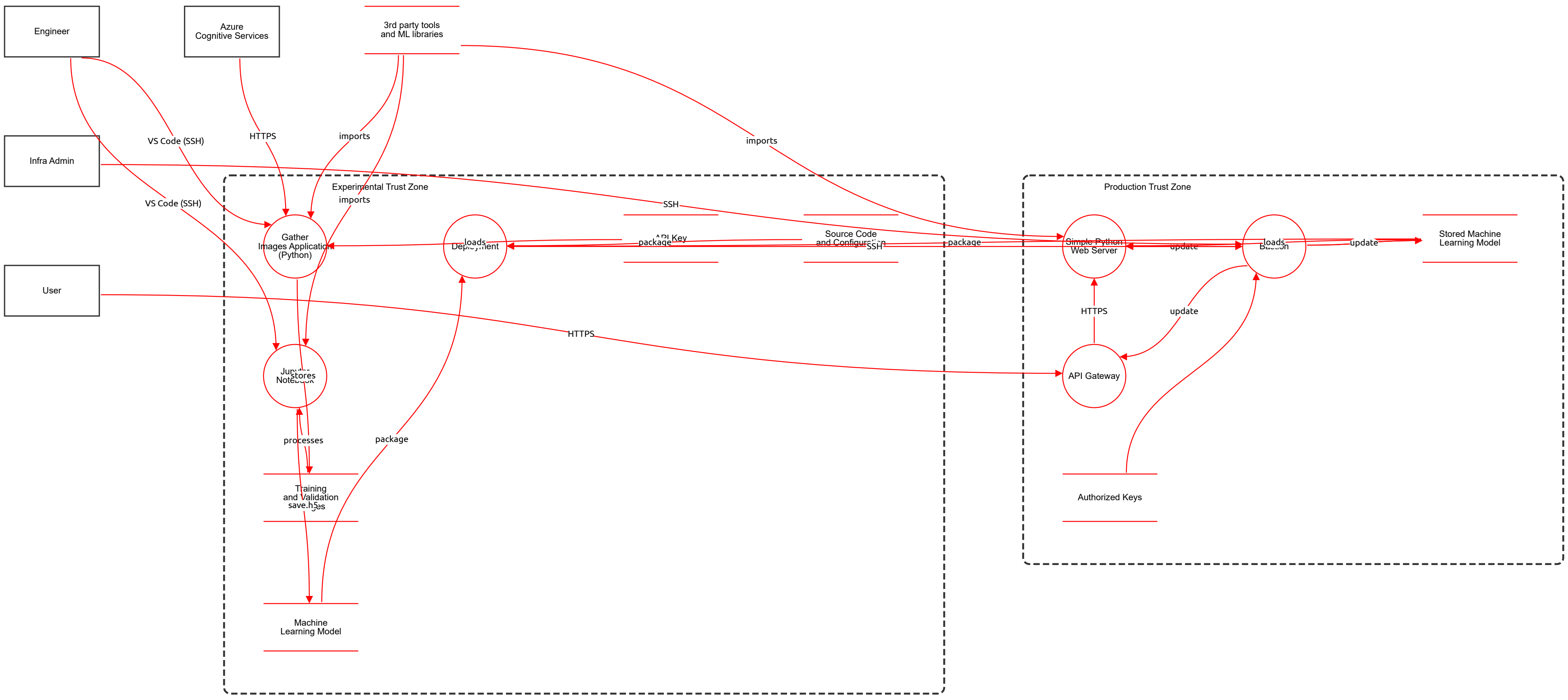
High level system description

A machine learning system to classify Huskies vs dogs. HuskyAI is a machine learning system designed to classify images and distinguish between huskies and non-huskies. It integrates secure data handling practices with a robust convolutional neural network (CNN) for image recognition. Secure Image Retrieval: HuskyAI uses TLS to securely fetch images from Azure Cognitive Services, ensuring encryption during data transmission and validating the server's authenticity to prevent man-in-the-middle attacks. Data Storage and Access Controls: Azure Blob Storage is used to store datasets, with public access fully blocked. Access is controlled using Role-Based Access Control (rbac) and Attribute-Based Access Control (ABAC) to enforce granular, identity-based permissions. Jupyter Notebooks, which host model development and experimentation, are also secured with rbac and ABAC, preventing unauthorized public access. Developer Authentication: Developers access the system through SSH keys protected by passphrases. This adds an additional layer of security, reducing the likelihood of unauthorized access even if keys are exposed. Model and Dataset Dataset Composition: The dataset comprises approximately 1,300 husky images and 3,000 non-husky images sourced via Bing's image search. Data undergoes manual cleansing and is split into training and validation sets to enhance model performance. Model Design: HuskyAI employs a CNN with: Convolutional layers for feature extraction. Max-pooling layers for dimensionality reduction. Dropout layers to prevent overfitting. Dense layers for final classification. The model is trained with the Adam optimizer and a learning rate of 0.0005, optimized for accuracy and computational efficiency. Security Considerations rbac and ABAC controls across storage and development environments ensure sensitive data and configurations are protected. TLS ensures secure communication channels, preventing eavesdropping or data interception during image retrieval. Applications HuskyAI is tailored for accurate image classification and can be adapted for other domains requiring precise visual differentiation, with a focus on maintaining strong security postures. HuskyAI combines state-of-the-art machine learning techniques with stringent security controls, including secure communications, robust access management, and encrypted developer authentication, to deliver a reliable and secure image classification system.

Summary

Total Threats	54
Total Mitigated	0
Total Open	54
Open / Critical Severity	0
Open / High Severity	35
Open / Medium Severity	17
Open / Low Severity	2

Husky AI


Engineer
Azure Cognitive Services
3rd party tools and ML libraries
Infra Admin
User
VS Code (SSH)
HTTPS
imports
imports
VS Code (SSH)
Experimental Trust Zone
imports
SSH
Gather Images Application (Python)
Deployment
loads
API Key
package
Source Code and Configuration
SSH
package
Jupyter Notebook
stores
HTTPS
processes
package
Training and Validation Images
save.h5
Machine Learning Model
Production Trust Zone
Simple Python Web Server
update
Bastion
loads
update
Stored Machine Learning Model
HTTPS
update
API Gateway
Authorized Keys

Husky AI

Engineer (Actor)

Description: A Data Engineer responsible for building, training, and deploying machine learning models.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing of Engineer Identity	Spoofing	High	Open		The Engineer actor, positioned outside trust boundaries, interacts with internal processes like Gather Images Application and Jupyter Notebook via SSH, potentially allowing an attacker to impersonate the engineer and gain unauthorized access to the Experimental Trust Zone.	Implement multi-factor authentication and use certificate-based SSH authentication with regular key rotation.

Infra Admin (Actor)

Description: Administrator responsible for securing and maintaining production infrastructure.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing of Infrastructure Admin	Spoofing	High	Open		The Infrastructure Admin actor, located outside all trust zones, accesses the Bastion in the Production Trust Zone via SSH, which could be spoofed to gain administrative privileges.	Enforce SSH with passphrase-protected keys and implement just-in-time access controls.

Azure Cognitive Services (Actor)

Description: External service providing resources for machine learning experimentation.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing of Azure Cognitive Services	Spoofing	Medium	Open		As an external actor outside trust boundaries, Azure Cognitive Services provides images to the internal Gather Images Application via HTTPS, risking spoofing attacks that could inject malicious data into the Experimental Trust Zone.	Validate server certificates and use mutual TLS for all communications with external services.

User (Actor)

Description: External user interacting with the HuskyAI system via the API Gateway.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing of User Identity	Spoofing	Medium	Open		The User actor, external to trust zones, sends requests to the API Gateway in the Production Trust Zone via HTTPS, potentially allowing spoofed identities to access the system.	Require API authentication tokens and implement rate limiting to prevent abuse.

3rd party tools and ML libraries (Store)

Description: External third party tools for the services

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Third-Party Libraries	Tampering	High	Open		The 3rd party tools and ML libraries store, positioned outside trust boundaries, supplies code to multiple internal processes across zones, risking tampering that could introduce vulnerabilities or backdoors.	Use integrity checks like hashing and source from trusted repositories with regular updates and vulnerability scanning.
	Information Disclosure from Libraries	Information Disclosure	Medium	Open		External third-party libraries could leak sensitive data if compromised, given their adjacency to external actors and integration into internal processes.	Conduct code reviews and use libraries with minimal permissions and logging.

Gather Images Application (Python) (Process)

Description: This is a Python-based application responsible for gathering images from external sources, specifically Azure Cognitive Services, and storing them in the designated Training and Validation Images storage.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Data Poisoning via Tampering	Tampering	High	Open		The Gather Images Application process, inside the Experimental Trust Zone, receives images from external Azure Cognitive Services crossing the trust boundary, allowing potential tampering or poisoning of training data.	Implement data validation, checksums, and anomaly detection on incoming images.
	Denial of Service on Image Gathering	Denial of Service	Medium	Open		Exposed to external flows crossing into the zone, this process could be flooded with requests, disrupting image collection.	Apply rate limiting and resource quotas on the process.
	Elevation via Unauthorized Code Execution	Elevation of Privilege	High	Open		As a Python process handling external inputs, it risks code injection leading to privilege escalation within the Experimental Trust Zone.	Run in a sandboxed environment with least privilege principles.

Jupyter Notebook (Process)

Description: A Jupyter Notebook environment that processes the images stored in Training and Validation Images, executes code using external ML libraries, and provides a UI for engineers to interact with and manipulate data, allowing for iterative model development. It can save trained machine learning models to Machine Learning Model storage.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Model Training	Tampering	High	Open		The Jupyter Notebook process, within the Experimental Trust Zone, processes training images and could be tampered with via malicious code imports or inputs, affecting model integrity.	Use version control and signed notebooks, with input sanitization.
	Information Disclosure of Training Data	Information Disclosure	Medium	Open		Handles sensitive training data inside the zone, risking disclosure if the notebook UI is compromised.	Encrypt data in transit and at rest, and restrict access via RBAC.
	Repudiation of Changes in Notebook	Repudiation	Low	Open		Actions in the notebook may not be properly logged, allowing denial of changes to models or data.	Enable comprehensive auditing and logging of all notebook executions.

Deployment (Process)

Description: Handles the deployment of the machine learning model by packaging the model and all necessary source code and configuration stored in Source Code and Configuration. It receives the final model from Jupyter Notebook and prepares it for deployment to the production environment.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Deployment Artifacts	Tampering	High	Open		The Deployment process, in the Experimental Trust Zone, packages models and code for production, risking tampering that could deploy malicious artifacts across zones.	Use digital signatures and integrity checks on all deployment packages.
	Elevation through Deployment Privileges	Elevation of Privilege	High	Open		Has access to sensitive stores and flows to production, potentially allowing escalation if compromised.	Implement strict RBAC and monitor deployment activities.

Training and Validation Images (Store)

Description: Contains images used for training and validation of machine learning models.
Data set: Training and Validation Images
Contains images used for training and validation of machine learning models.
Record count maximum of 100000 with data sensitivity of biz and access control methods of rbac

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Training Images	Tampering	High	Open		This store inside the Experimental Trust Zone holds business-sensitive images, accessible via internal flows, risking tampering that poisons ML models.	Enable versioning and immutability features in storage, with access logging.
	Information Disclosure of Images	Information Disclosure	Medium	Open		Encrypted store but adjacent to processes with external interactions, potentially leaking data if access controls fail.	Enforce ABAC and encrypt data at rest.

API Key (Store)

Description: Stores API keys for secure access to external services.
Data set: API Keys
Stores API keys for secure access to external services.
Record count maximum of 20 with data sensitivity of cred and access control methods of rbac

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Information Disclosure of API Keys	Information Disclosure	High	Open		Stores credentials inside Experimental Trust Zone, with flows to gathering process, risking exposure of keys to external services.	Use secret management tools like Azure Key Vault with rotation policies.
	Tampering with Stored Keys	Tampering	High	Open		Keys could be altered, affecting authentication to external services.	Implement write-once policies and audit trails.

Machine Learning Model (Store)

Description: Contains the machine learning models in serialized format.
Data set: Bastion Logs
Contains trained machine learning models in serialized format for production use.
Record count maximum of 5000 with data sensitivity of biz and access control methods of acl

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Model Theft via Information Disclosure	Information Disclosure	High	Open		Stores ML models in Experimental Trust Zone, with flows to deployment, risking model extraction or reverse engineering.	Encrypt models and use access controls with monitoring.
	Tampering with Model Files	Tampering	High	Open		Models could be modified, leading to inaccurate or malicious behavior in production.	Use hashing and signatures for model integrity.

Source Code and Configuration (Store)

Description: Stores source code and configuration files for deployment and production setup.
Data set: Source Code and Configuration
Stores source code and configuration files for deployment and production setup.
Record count maximum of 200 with data sensitivity of biz and access control methods of rbac

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Source Code	Tampering	High	Open		Stores code and configs in Experimental Trust Zone, flowed to deployment, risking injection of backdoors.	Employ code signing and repository protections.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Repudiation of Code Changes	Repudiation	Medium	Open		Changes to code may not be attributable, allowing denial of malicious modifications.	Integrate with version control systems that log commits immutably.

Simple Python Web Server (Process)

Description: Serves as simple web server

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Denial of Service on Web Server	Denial of Service	Medium	Open		Simple Python Web Server in Production Trust Zone receives ingress from API Gateway, vulnerable to request flooding.	Implement auto-scaling and DDoS protection.
	Elevation via Model Inference	Elevation of Privilege	High	Open		Loads models and processes user inputs, potentially allowing inference attacks to extract privileged information.	Apply input validation and differential privacy techniques.

API Gateway (Process)

Description: Serves as the entry point for external users to interact with the production environment via HTTPS. It routes user requests to the Simple Python Web Server and ensures secure communication. The API Gateway enforces request validation and manages APIs exposed to the public while ensuring access control to internal services.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing at API Gateway	Spoofing	High	Open		API Gateway in Production Trust Zone handles ingress from external User over HTTPS crossing the boundary, risking spoofed requests.	Enforce token-based authentication and validate all incoming requests.
	Information Disclosure through Gateway	Information Disclosure	Medium	Open		Routes sensitive traffic, potentially leaking data if not properly secured.	Ensure end-to-end encryption and logging without sensitive data.

Bastion (Process)

Description: A secure access management component for administrative functions. It provides controlled SSH access for the Infrastructure Admin to internal production resources, such as the Stored Machine Learning Model and Simple Python Web Server.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Elevation via Bastion Access	Elevation of Privilege	High	Open		Bastion in Production Trust Zone provides SSH access from external admin, crossing boundary, allowing potential escalation to production resources.	Use session recording and least privilege for bastion users.
	Repudiation of Admin Actions	Repudiation	Medium	Open		Administrative actions through bastion may lack proper auditing.	Enable detailed logging and non-repudiation mechanisms.

Authorized Keys (Store)

Description: Contains SSH keys used for securing administrative access.
Data set: Authorized Keys
Contains SSH keys used for securing administrative access.
Record count maximum of 100 with data sensitivity of cred and access control methods of rbac

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Information Disclosure of SSH Keys	Information Disclosure	High	Open		Stores credentials in Production Trust Zone, used by bastion, risking key exposure.	Store in encrypted vaults with access monitoring.

Stored Machine Learning Model (Store)

Description: Contains storage for machine learning models in serialized format.
Data set: Stored Machine Learning Models
Contains trained machine learning models in serialized format for production use.
Record count maximum of 10 with data sensitivity of biz and access control methods of rbac

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Production Models	Tampering	High	Open		Stores models in Production Trust Zone, updated via bastion, risking tampering for malicious inference.	Implement immutability and verification checks.

HTTPS (Data Flow)

Description: Transfer data from Azure Cognitive Services to Gather Images Application in Python.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing on Image Retrieval Flow	Spoofing	Medium	Open		The HTTPS flow from external Azure Cognitive Services to Gather Images Application crosses into Experimental Trust Zone, potentially allowing spoofed sources to inject poisoned data.	Validate TLS certificates and pin public keys.
	Information Disclosure in Transit	Information Disclosure	Low	Open		Although encrypted, the flow crosses boundaries; weak TLS could lead to disclosure.	Enforce TLS 1.3 with strong ciphers.

imports (Data Flow)

Description: Transfer data from Third Party tools and ML libraries to Gather Images Application in Python.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering in Library Import Flow	Tampering	High	Open		The imports flow from external third-party tools to internal Gather Images Application crosses into Experimental Trust Zone, risking tampered libraries.	Verify library integrity with signatures before import.

imports (Data Flow)

Description: Transfer data from Third Party tools and ML libraries to Jupyter Notebook.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with ML Library Imports	Tampering	High	Open		Imports flow from external libraries to Jupyter Notebook inside Experimental Trust Zone, potentially introducing tampered code.	Use locked dependencies and scan for vulnerabilities.

VS Code (SSH) (Data Flow)

Description: Transfer data from Engineer to Gather Images Application in Python.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing on SSH Flow	Spoofing	High	Open		VS Code (SSH) flow from external Engineer to Gather Images Application crosses into Experimental Trust Zone, risking spoofed access.	Require client certificates for SSH.

VS Code (SSH) (Data Flow)

Description: Transfer code and ML models from Engineer locally to Jupyter Notebook.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Elevation via SSH Access	Elevation of Privilege	High	Open		SSH flow from external Engineer to Jupyter Notebook allows potential code execution with elevated privileges inside the zone.	Restrict SSH commands and use bastion-like controls.

stores (Data Flow)

Description: Transfer images from Gather Images Application to Training and Validation Images.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Stored Images	Tampering	Medium	Open		Stores flow internal to Experimental Trust Zone, but if process compromised, could tamper with images en route to storage.	Use signed data transfers and validation.

loads (Data Flow)

Description: API Key Storage to Gather Images Application in Python.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Information Disclosure of Keys in Flow	Information Disclosure	High	Open		Loads flow internal, but exposes credentials from storage to process, risking interception if not secured.	Use secure in-memory handling and avoid logging keys.

processes (Data Flow)

Description: Load from Training and Validation Images to Jupyter Notebook.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering During Image Processing	Tampering	Medium	Open		Processes flow internal, potential for tampering if notebook is compromised.	Implement data integrity checks during loading.

package (Data Flow)

Description: Transfer data from Machine Learning Model to Deployment.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Model Package	Tampering	High	Open		Package flow internal, but critical for deployment; tampering could affect production.	Sign models before transfer.

save.h5 (Data Flow)

Description: Transfer final model from Jupyter Notebook to Machine Learning Model.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Information Disclosure of Model Save	Information Disclosure	Medium	Open		Save.h5 flow internal, risking model leak if storage access is breached.	Encrypt flow and storage.

package (Data Flow)

Description: Transfer from Machine Learning Model Blob to Deployment Service.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering in Model Transfer	Tampering	High	Open		Package flow from production model to deployment, though id suggests prod, but layout in experimental? Wait, id is ml-model-deployment-service, but source is ml-models-blob (prod) to deployment-service (experimental), so crosses zones! High risk for tampering across boundaries.	Use secure channels and verification across zones.

package (Data Flow)

Description: Transfer data from Source Code and Configuration to Deployment.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Code Flow	Tampering	High	Open		Package flow internal to experimental, risking code injection.	Integrity checks on code.

HTTPS (Data Flow)

Description: Transfer from User to API Gateway.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Denial of Service on Ingress Flow	Denial of Service	Medium	Open		HTTPS flow from external User crosses into Production Trust Zone, vulnerable to DoS attacks.	Deploy WAF and rate limiting.

update (Data Flow)

Description: Transfer data from Bastion to API Gateway.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Elevation on Update Flow	Elevation of Privilege	High	Open		Update flow internal to prod, but from bastion, potential for unauthorized updates.	Authorize updates with RBAC.

HTTPS (Data Flow)

Description: Transfer data from API Gateway to Simple Python Web Server.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing Internal Requests	Spoofing	Medium	Open		HTTPS flow internal, but if gateway compromised, could spoof to server.	Internal mTLS.

update (Data Flow)

Description: Transfer data from Bastion to Simple Python Web Server.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering via Update	Tampering	High	Open		Update flow from bastion to web server, risking tampered configurations.	Signed updates.

loads (Data Flow)

Description: Transfer sensitive data from Stored Machine Learning Model to Simple Python Web Server.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Information Disclosure of Model Load	Information Disclosure	Medium	Open		Loads flow internal, potential leak during loading.	Secure loading mechanisms.

SSH (Data Flow)

Description: Transfer sensitive data from Deployment Service to Bastion

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing on SSH Deployment	Spoofing	High	Open		SSH flow from experimental deployment to prod bastion crosses zones, high risk for spoofing.	Cross-zone authentication 強化.

update (Data Flow)

Description: Transfer sensitive data from Bastion to Stored Machine Learning Model.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering Across Zones	Tampering	High	Open		Update flow from bastion (prod) to ml-models-blob, but id bastion-ml-model, potentially crossing? Description to Stored ML Model.	Verify updates.

SSH (Data Flow)

Description: Transfer data from Infrastructure Admin to Bastion.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Spoofing Admin SSH	Spoofing	High	Open		SSH flow from external admin crosses into prod zone.	MFA for admin access.

update (Data Flow)

Description: Transfer sensitive data from Bastion to Stored Machine Learning Model.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Elevation on Model Update	Elevation of Privilege	High	Open		Update flow internal, potential for unauthorized model changes.	RBAC on updates.

(Data Flow)

Description: Transfer sensitive data from Authorized Keys Storage to Bastion.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Information Disclosure of Keys Flow	Information Disclosure	High	Open		Flow from authorized keys to bastion, risking key exposure.	Encrypted storage and transfer.

imports (Data Flow)

Description: Transfer data from Third Party tools and ML libraries to Simple Python Web Server.

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Tampering with Prod Imports	Tampering	High	Open		Imports flow from external tools crosses into prod zone to web server.	Integrity verification.