Husky AI

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

Executive Summary

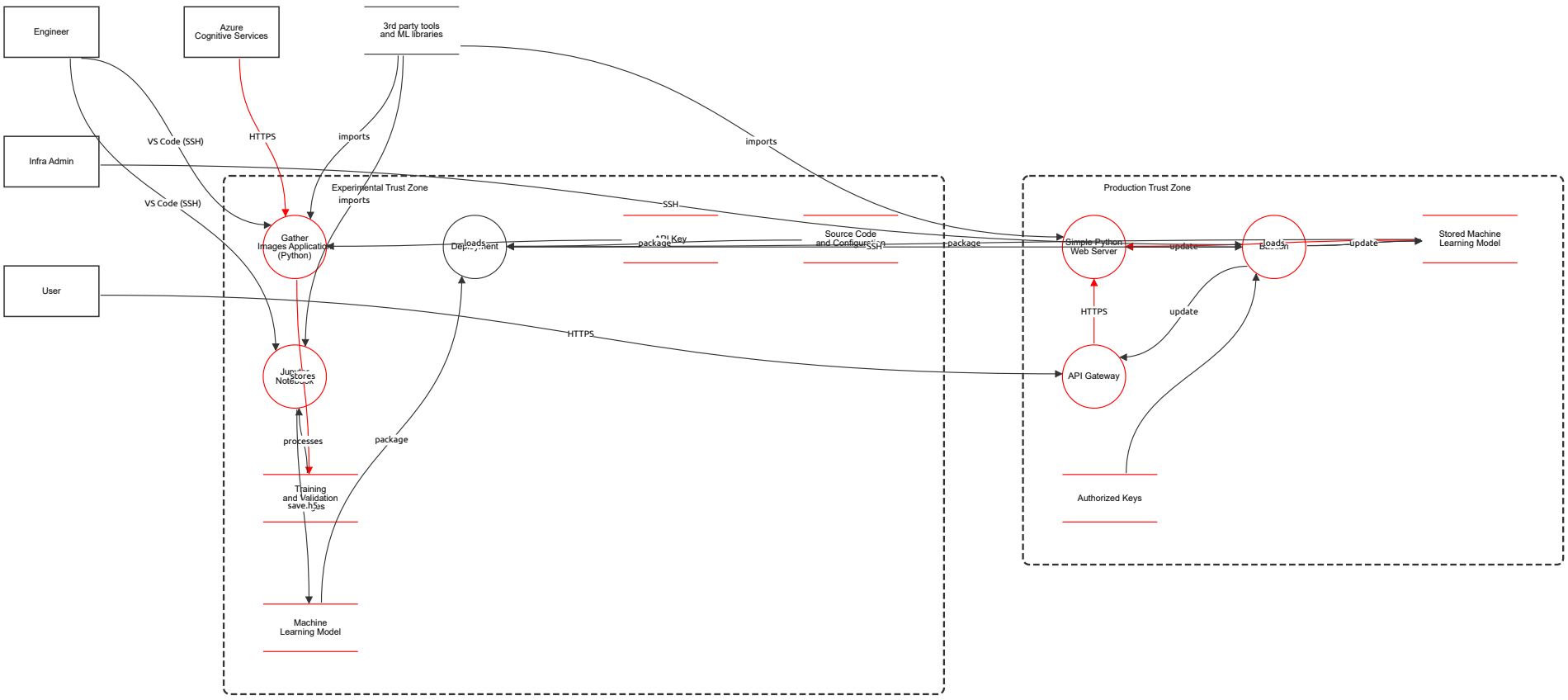
High level system description

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	16
Total Mitigated	0
Total Open	16
Open / Critical Severity	0
Open / High Severity	11
Open / Medium Severity	5
Open / Low Severity	0

Husky AI


Engineer
Azure Cognitive Services
3rd party tools and ML libraries
Infra Admin
User
VS Code (SSH)
HTTPS
imports
SSH
Experimental Trust Zone
Gather Images Application (Python)
Deployment
API Key
package
Source Code and Configuration
Production Trust Zone
Simple Python Web Server
loads
update
Stored Machine Learning Model
API Gateway
Authorized Keys
stores
processes
Training and Validation save.h5as
Machine Learning Model

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

Infra Admin (Actor)

Description: Administrator responsible for securing and maintaining production infrastructure.

Number	Title	Type	Severity	Status	Score	Description	Mitigations

Azure Cognitive Services (Actor)

Description: External service providing resources for machine learning experimentation.

Number	Title	Type	Severity	Status	Score	Description	Mitigations

User (Actor)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

3rd party tools and ML libraries (Store)

Description: External third party tools for the services

Number	Title	Type	Severity	Status	Score	Description	Mitigations

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
	Unencrypted internal data flow	Information Disclosure	Medium	Open		The flow from Gather Images Application to Training and Validation Images is not encrypted, potentially exposing sensitive training data within the Experimental Trust Zone.	Encrypt data at rest and in transit using AES-256 for storage and TLS 1.3 for internal communications.

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
	Model manipulation in experimental environment	Tampering	High	Open		Jupyter Notebook process resides in the Experimental Trust Zone and handles sensitive ML models, making it vulnerable to tampering that could affect model integrity.	Implement code signing for all notebooks and models, enforce RBAC with least privilege, and enable audit logging for all model modifications.

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

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
	Unencrypted storage of sensitive training data	Information Disclosure	High	Open		Training and Validation Images store contains sensitive business data with maximum record count of 100000 but is not encrypted at rest, risking data exposure if storage is compromised.	Enable encryption at rest using Azure Storage Service Encryption and enforce access controls with RBAC and ABAC policies.

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
	Credential exposure in storage	Information Disclosure	High	Open		API Key storage contains highly sensitive credentials with data sensitivity level 'cred' but lacks proper encryption mechanisms, risking credential theft.	Use Azure Key Vault for secure credential management and implement automatic key rotation with strict access policies.

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 intellectual property exposure	Information Disclosure	High	Open		Machine Learning Model store contains business-sensitive trained models with ACL access control but lacks encryption, risking IP theft if accessed without authorization.	Encrypt models at rest using AES-256 and implement model watermarking to track unauthorized usage.

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
	Source code exposure risk	Information Disclosure	Medium	Open		Source Code and Configuration storage contains business-sensitive deployment configurations but lacks encryption, potentially exposing system architecture details.	Encrypt storage at rest and implement CI/CD security scanning to detect hardcoded secrets in configurations.

Simple Python Web Server (Process)

Description: Serves as simple web server

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Unencrypted communication to web server	Information Disclosure	High	Open		Flow from API Gateway to Simple Python Web Server is not encrypted, exposing data in transit within the Production Trust Zone.	Enforce TLS 1.3 encryption for all internal service communications and implement mutual TLS authentication.
	Web server DoS vulnerability	Denial of Service	Medium	Open		Simple Python Web Server is directly accessible from API Gateway without rate limiting, making it susceptible to DoS attacks.	Implement rate limiting at the API Gateway and deploy a Web Application Firewall (WAF) to filter malicious traffic.

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
	API Gateway boundary crossing exposure	Information Disclosure	High	Open		API Gateway serves as entry point from external User to internal Production Trust Zone without sufficient request validation, risking data leakage.	Implement comprehensive input validation, API rate limiting, and OAuth 2.0 authentication at the gateway level.

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
	Bastion host privilege escalation	Elevation of Privilege	High	Open		Bastion process provides SSH access to internal production resources and connects to multiple sensitive stores, creating a high-value target for privilege escalation.	Implement just-in-time access controls, enforce multi-factor authentication, and restrict SSH access with certificate-based authentication only.

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
	SSH key storage vulnerability	Information Disclosure	High	Open		Authorized Keys storage contains sensitive SSH credentials with data sensitivity 'cred' but relies only on RBAC without encryption, risking unauthorized administrative access.	Store SSH keys in Azure Key Vault with hardware security module (HSM) protection and implement automatic key rotation.

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
	Unencrypted production model storage	Information Disclosure	High	Open		Stored Machine Learning Model contains business-sensitive production models but is not encrypted, risking exposure of intellectual property.	Enable encryption at rest for blob storage and implement access logging with real-time monitoring for unauthorized access attempts.

HTTPS (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Data interception during external transfer	Information Disclosure	Medium	Open		Flow from Azure Cognitive Services to Gather Images Application uses HTTPS but traverses between external service and internal Experimental Trust Zone, risking man-in-the-middle attacks.	Implement certificate pinning and validate server certificates to prevent TLS interception attacks.

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

imports (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

VS Code (SSH) (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

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

stores (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Unencrypted internal data transfer	Information Disclosure	Medium	Open		Flow transferring images from Gather Images Application to Training and Validation Images is not encrypted, exposing data during internal transfer within Experimental Trust Zone.	Encrypt data in transit using TLS 1.3 and implement data loss prevention (DLP) policies to monitor sensitive data movement.

loads (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

processes (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

package (Data Flow)

Description: Transfer data from Machine Learning Model to Deployment.

Number	Title	Type	Severity	Status	Score	Description	Mitigations

save.h5 (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

package (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

package (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

HTTPS (Data Flow)

Description: Transfer from User to API Gateway.

Number	Title	Type	Severity	Status	Score	Description	Mitigations

update (Data Flow)

Description: Transfer data from Bastion to API Gateway.

Number	Title	Type	Severity	Status	Score	Description	Mitigations

HTTPS (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations
	Unencrypted production data flow	Information Disclosure	High	Open		Flow from API Gateway to Simple Python Web Server is not encrypted, exposing sensitive data in transit within the Production Trust Zone.	Enforce end-to-end TLS encryption and implement service mesh with mutual TLS authentication for all internal communications.

update (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

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
	Unencrypted model loading flow	Information Disclosure	High	Open		Flow loading sensitive data from Stored Machine Learning Model to Simple Python Web Server is not encrypted, risking model exposure during transfer.	Encrypt data in transit using TLS 1.3 and implement secure model loading mechanisms with integrity verification.

SSH (Data Flow)

Description: Transfer sensitive data from Deployment Service to Bastion

Number	Title	Type	Severity	Status	Score	Description	Mitigations

update (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

SSH (Data Flow)

Description: Transfer data from Infrastructure Admin to Bastion.

Number	Title	Type	Severity	Status	Score	Description	Mitigations

update (Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

(Data Flow)

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

Number	Title	Type	Severity	Status	Score	Description	Mitigations

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