Project Design Phase-II
Technology Stack (Architecture & Stack)

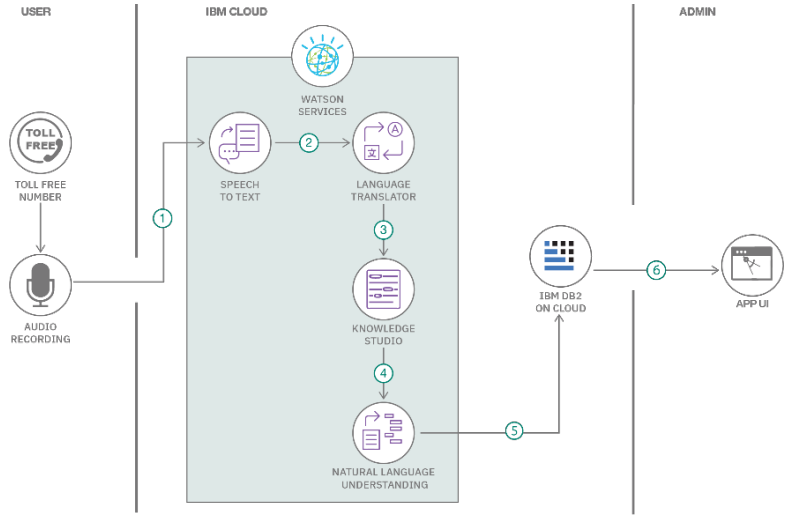
Date	28 October 2023
Team ID	PNT2022TMIDxxxxxx
Project Name	AI-Based Threat Intelligence Platform
Maximum Marks	4 Marks

Technical Architecture:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2

Example: Order processing during pandemics for offline mode

Reference: AI-Driven Cybersecurity: An Overview, Security Intelligence Modeling and Research Directions.



Guidelines:

1. Include all the processes (As an application logic / Technology Block)
2. Provide infrastructural demarcation (Local / Cloud)
3. Indicate external interfaces (third party API's etc.)
4. Indicate Data Storage components / services
5. Indicate interface to machine learning models (if applicable)

Table-1 : Components & Technologies:

Component	Description	Technology
User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
Application Logic-1	Logic for collecting and ingesting threat data from various sources	Python / Apache NiFi / Kafka
Application Logic-2	Logic for analyzing and enriching threat data using natural language processing and machine learning techniques	Python / TensorFlow / spaCy / IBM Watson NLU
Application Logic-3	Logic for generating and delivering threat intelligence reports and alerts using natural language generation and visualization tools	Python / GPT-3 / Plotly / IBM Watson Assistant
Database	Data Type, Configurations etc.	MongoDB, Elasticsearch, etc.
Cloud Database	Database Service on Cloud	AWS S3, AWS DynamoDB, etc.
File Storage	File storage requirements	AWS EBS or Other Storage Service or Local Filesystem
External API-1	Purpose of External API used in the application	VirusTotal API, Shodan API, etc.
External API-2	Purpose of External API used in the application	Slack API, Twilio API, etc.

Component	Description	Technology
Machine Learning Model	Purpose of Machine Learning Model	Threat Detection Model, Threat Classification Model, etc.
Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	AWS EC2, AWS Lambda, Docker, Kubernetes, etc.

Some of the application characteristics are:

Characteristics	Description	Technology
Open-Source Frameworks	List the open-source frameworks used	Python, Apache NiFi, Kafka, TensorFlow, spaCy, GPT-3, Plotly, etc.
Security Implementations	List all the security / access controls implemented, use of firewalls etc. e.g. SHA-256, Encryptions, IAM Controls, OWASP etc.	HTTPS, SSL/TLS, JWT, OAuth 2.0, AWS IAM, AWS KMS, AWS WAF, etc.
Scalable Architecture	Justify the scalability of architecture (3 – tier, Micro-services)	Microservices architecture using Docker and Kubernetes to enable horizontal scaling and load balancing of the application services
Availability	Justify the availability of application (e.g. use of load balancers, distributed servers etc.)	Use of AWS services such as EC2 Auto Scaling Groups, Elastic Load Balancers, S3 Buckets, DynamoDB Tables to ensure high availability and fault tolerance of the application components

Characteristics	Description	Technology
Performance	Design consideration for the performance of the application (number of requests per sec, use of Cache, use of CDN's) etc. give it in this way	Use of Elasticsearch for fast and efficient search and retrieval of threat data; Use of Redis for caching frequently accessed data; Use of AWS CloudFront for content delivery network; Use of GPT-3 for natural language generation; Use of TensorFlow Serving for machine learning model deployment and inference; Use of Apache NiFi for data flow management and orchestration; Use of Kafka for data streaming and messaging; Use of Plotly for interactive and responsive data visualization; Use of Slack and Twilio for real-time communication and notification; Use of VirusTotal and Shodan for external threat data sources; Use of IBM Watson NLU and IBM Watson Assistant for natural language understanding and chatbot interface; Use of AWS Lambda for serverless computing and event-driven execution; Use of AWS EBS for persistent block storage; Use of AWS S3 for object storage; Use of AWS DynamoDB for NoSQL database service; Use of AWS EC2 for virtual server instances; Use of Docker for containerization; Use of Kubernetes for container orchestration; Use of AWS WAF for web application firewall; Use of HTTPS, SSL/TLS, JWT, OAuth 2.0, AWS IAM, AWS KMS for security and encryption; Use of RESTful APIs or GraphQL for web service communication; Use of HTML, CSS, JavaScript / Angular Js / React Js for web-based user interface design.

References:

Introducing AI-powered insights in Threat Intelligence

Best Threat Intelligence Platforms - 2023 Reviews & Comparison

Top 7 Threat Intelligence Platforms & Tools for 2023

9 Best Threat Intelligence Platforms (TIPs)