Adaptive Authentication with AI-Based Risk Scoring Project Report

Bermal Aratoğlu

B201202039

Buğra Çelebi

B201202005

Project Objective

The primary objective of this project is to dynamically activate or deactivate multi-factor authentication (MFA) systems based on a risk score generated by a deep learning model. By analyzing user login details and login information, a risk score is produced. Based on this score, the system determines whether MFA should be triggered. This approach enhances user security while preventing unnecessary MFA activation, thereby improving the user experience.

Introduction

Multi-factor authentication (MFA) is a widely used method to enhance account security. However, MFA is often perceived as time-consuming by users and is seen as an unnecessary burden, particularly during routine, daily logins. This perception can negatively impact user experience and reduce engagement with the system.

On the other hand, security measures such as MFA are critical for ensuring data security. Given the increasing prevalence of cyber threats and phishing attacks, neglecting security can result in significant risks.

The deep learning-based system developed in this project aims to balance security and user experience. By analyzing user login details and producing a risk score, the system determines whether MFA should be triggered. This ensures that

security measures are intensified in risky situations while eliminating unnecessary MFA activation during secure logins, thereby enhancing efficiency.

Project Scope

In this project, a deep learning model has been developed to enable MFA systems to operate dynamically based on a risk-based score and tested on a web application. The project includes the following steps:

1. **Dataset:**
    - The Risk-Based Authentication (RBA) dataset was used, which contains user login details and behavioral information.
2. **Model Development:**
    - A **Feedforward Neural Network (FNN)** model was developed using TensorFlow.
    - The model was trained to analyze user login information and produce a risk score.
3. **Model Deployment:**
    - The developed deep learning model was saved in the **SavedModel** format and converted to **TensorFlow.js (TFJS)** format to make it operable in a web environment.
    - The model was integrated into a **Next.js**based web application, where its functionality was tested.
4. **User Authentication and Data Management:**
    - User authentication processes and login records were managed using **Firebase Authentication**.
    - User data was securely stored using **Firestore Database**.

Through this scope, the developed system ensures security while providing an efficient authentication experience with minimal time loss for users.

Dataset

The dataset used in the project was prepared for risk-based authentication (RBA) and underwent preprocessing and feature engineering steps. The dataset consists of the following features:

1. **Basic User Information:**
    - **User ID:** Unique identifier for the user.
    - **IP Address, Country, ASN:** User's IP information, country, and Autonomous System Number (ASN).
    - **OS Name and Version, Device Type:** User's device and operating system details.
2. **Time-Based Features:**
    - **Day, Hour, Weekday:** Detailed information about login time.
    - **Peak Hours:** Indicates whether the login occurred during peak hours.
3. **Behavior-Based Features:**
    - **City Change, Region Change:** Binary (0/1) indicators for city or region changes compared to previous logins.
    - **City Frequency, Region Frequency:** Frequency of logins from specific cities/regions.
    - **Unusual Device:** Indicates whether the login was made from a previously unused device.
4. **Performance and Risk Features:**
    - **Round-Trip Time (RTT) [ms]:** Network latency during login.
    - **RTT Normalization:** Normalized RTT value.
    - **Login Success Ratio:** User's successful login ratio.
    - **Failed Attempts (10min):** Number of failed login attempts within the last 10 minutes.
    - **Time Delta:** Time difference between two consecutive logins.
    - **Is Attack IP, Is Account Takeover:** Binary indicators for attack IPs or account takeovers.

Modeling

In the modeling phase of the project, a **Feedforward Neural Network (FNN)** was developed to analyze user login details and compute a risk score. The creation, training, and evaluation of the model involved the following steps:

1. **Data Preparation:**
   - **Features (X):** All columns except `Login Successful`.
   - **Target (y):** `Login Successful` (binary classification target).
   - **Data Splitting:**
     - Data was split into training (70%), validation (15%), and test (15%) sets.
2. **Model Architecture:**
   The model was built using TensorFlow as a **Sequential** neural network with the following structure:
   - **Input Layer:** Input size with 24 features.
   - **Hidden Layers:**
     - **Dense (128 neurons, ReLU activation, L2 Regularization - 0.01):** Regularization to prevent overfitting.
     - **Dropout (0.2):** Randomly deactivates neurons to improve generalization.
     - **Batch Normalization:** Stabilizes weights and accelerates learning.
     - The same structure was repeated for hidden layers with **64** and **32** neurons.
   - **Output Layer:**
     - **Dense (1 neuron, sigmoid activation):** Used for binary classification.
3. **Model Compilation:**
   - **Loss Function:** `binary_crossentropy`
   - **Optimizer:** Adam optimizer (learning rate = 0.001)
   - **Evaluation Metrics:** `accuracy` and `AUC`

4. **Model Training:**
   The model was trained with the following hyperparameters:
   - **Epochs:** 100
   - **Batch Size:** 128
   - **Validation Data:** Used to monitor performance during training.
5. **Model Evaluation:**
   The model was evaluated on test data with the following results:
   - **Test Accuracy:** 0.8079
   - **Test AUC:** 0.8301

These results demonstrate that the model is reliable in both accurate classification and generating dependable risk scores.

Model Conversion

The trained FNN model was saved in the **SavedModel** format and converted to **TensorFlow.js** using the TensorFlow.js Converter:

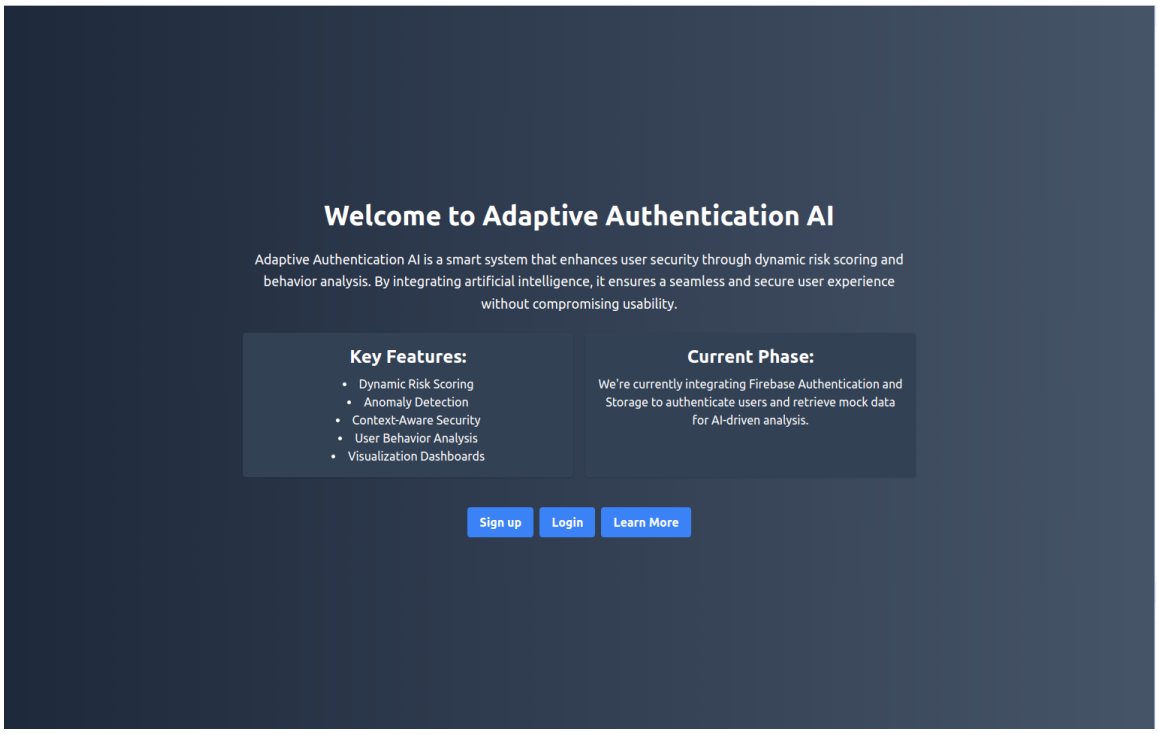
```
tensorflowjs_converter --input_format=tf_saved_model \
                       saved_model_directory \
                       web_model_directory
```

Model Integration with the Website

The developed **Feedforward Neural Network (FNN)** model analyzes user login details to produce a risk score, determining whether MFA should be triggered. The model was converted to the **TensorFlow.js (TFJS)** format and integrated into a **Next.js**-based web application.

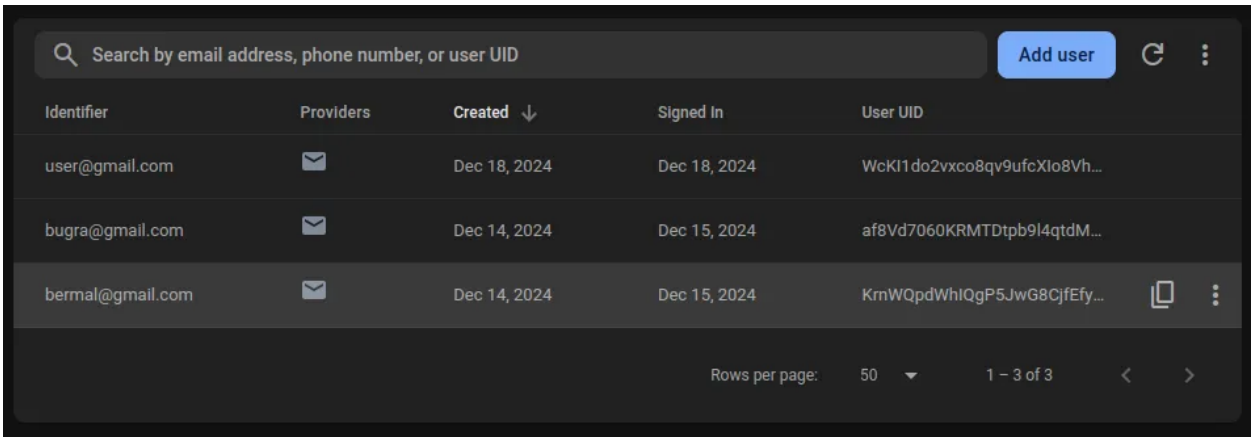
Home Page of the Adaptive Authentication Website

Below is the homepage of the web application where the **Adaptive Authentication AI** system is integrated. The webpage highlights the project's key features and the current development phase:



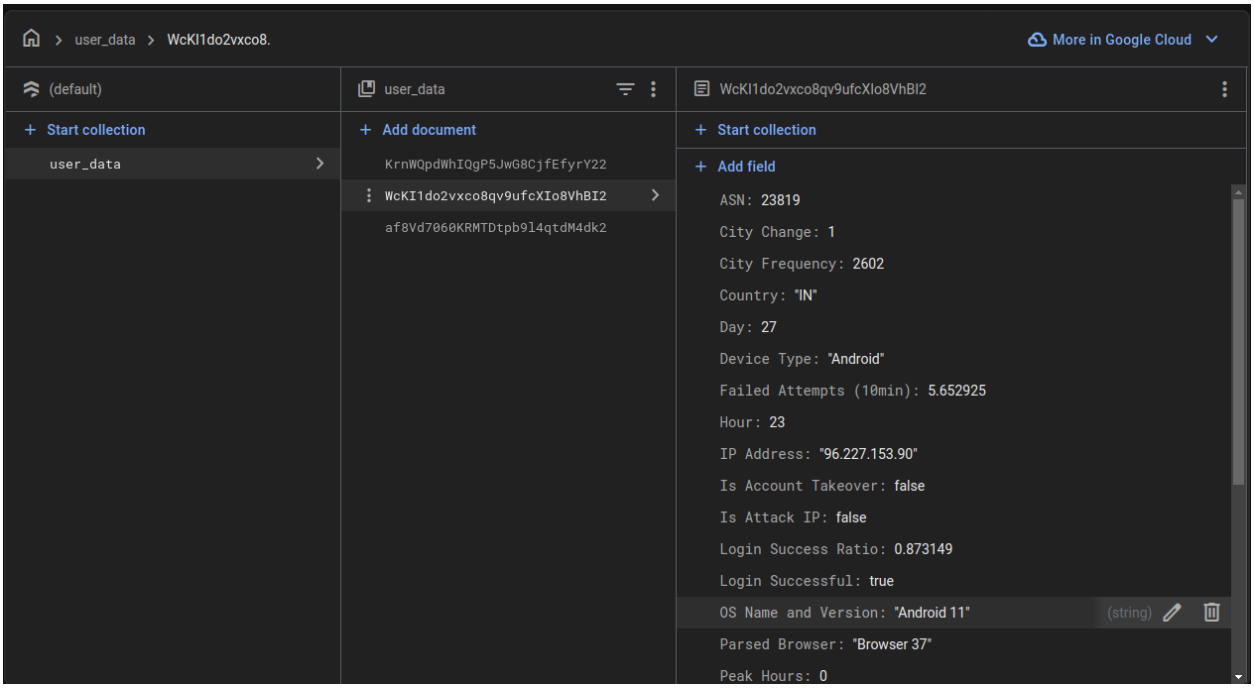
Sign Up and Firebase Authentication

The sign-up page allows users to register with their email and password, ensuring a secure authentication system. Below is an example of a user being successfully registered in Firebase Authentication:



Firestore Database for Additional User Information

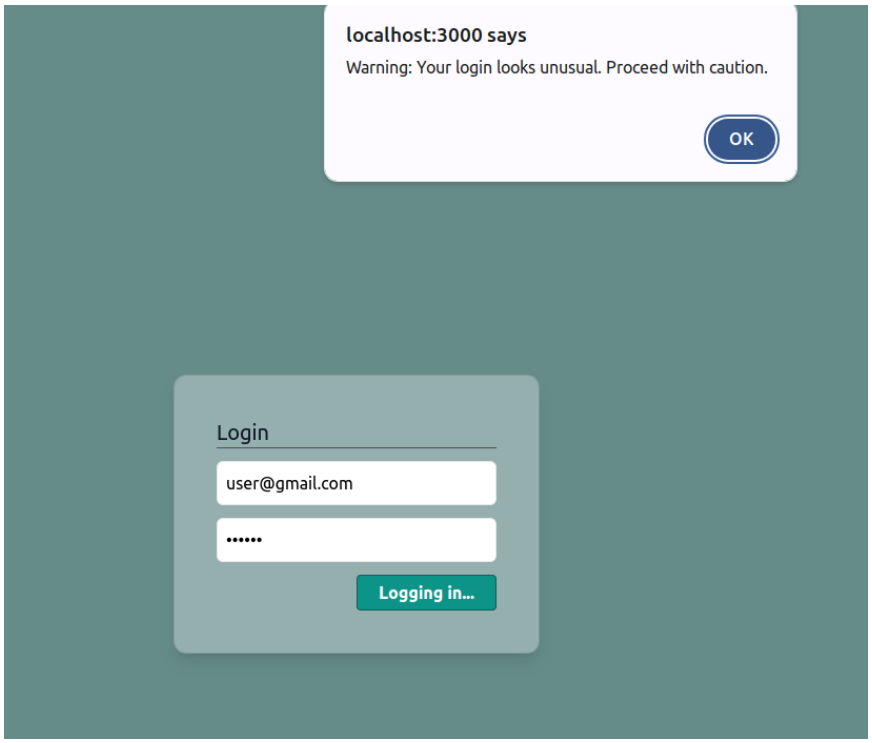
Once the user successfully registers, additional user data (such as login details, region changes, and session activity) is stored in Firestore Database for further analysis and risk scoring. Below is an example of the data saved in Firestore:



Login Page and Predicted Output

The login page allows users to enter their credentials. The system processes the login details, generates a risk score, and determines whether access is granted or denied based on the predicted output:

```
Prediction Result:                    LoginCard.jsx:20
0.4203548548221588
>
```



But if the access is granted, the user is directed to a dashboard:

```
Prediction Result:                    LoginCard.jsx:20
0.7281025457382202
Login Successful: Navigating          LoginCard.jsx:24
to dashboard
>
```

Dynamic MFA Activation In the Project

The risk score dynamically determines the MFA activation:

- **Low Risk:** MFA is bypassed.
- **Medium Risk:** User is alerted, however login is accepted.
- **High Risk:** User is alerted and login is denied.

Analysis Results

The deep learning model developed in this project has shown promising results with the provided dataset, achieving **80% accuracy** and an **AUC of 0.83**. However, some key observations include:

1. **Feature Complexity and Human-Generated Data:**
    - While the model performs well on the dataset, the large number of features makes it challenging to validate performance with manually generated data.
    - This highlights the need for further testing and evaluation using additional datasets.

2. **Future Potential of AI Systems:**
    - AI methods are increasingly critical in cryptology and security systems.
    - Such dynamic risk assessment systems can significantly enhance MFA methods, providing reliable and adaptive security measures.

Dynamic Secondary Authentication in MFA Systems

The model's ability to analyze risk scores enables the selection of **dynamic secondary authentication methods** in MFA systems, making them more flexible, dynamic, and secure.

Examples:

- **Low Risk:** Skip MFA or use light verification (e.g., email notification).
- **Medium Risk:** Use alternative methods such as push notifications or biometric verification (e.g., fingerprint or face recognition).
- **High Risk:** Apply stronger methods such as hardware security keys or additional identity verification.

Advantages of Dynamic MFA

1. **User Experience:**
    - Reduces unnecessary security steps for trusted logins.
2. **Security:**
    - Stronger methods are triggered in high-risk situations, mitigating security breaches.
3. **Flexibility:**
    - Dynamically adapts to risk levels using factors like device type, location, and network data.

Conclusion

The developed model has demonstrated strong performance on the existing dataset but requires further testing with real-world scenarios and new datasets. AI

methods are playing an increasingly critical role in security systems, and this model has the potential to enhance MFA systems by introducing new layers of dynamic risk-based decision-making, leading to more secure and reliable authentication mechanisms.