Ultimate security by combining Cloudflare's and Akamai's APIs and machine learning.

Abdelrahman Abdelwahab, STEM high school for boys- 6th of October.

Ahmed Osama, WE for applied technology.

Abstract

As of 2023, 30,000 websites are hacked daily, and 64% of companies worldwide have experienced at least one form of cyber-attack. *As hacking increases, the need to implement a secure security system increases. This paper discusses the implementation of a security system that combines Cloudflare's API, Akamai's API, and machine learning (ML) algorithm. Machine learning and deep learning algorithms were used to determine which one gets the best accuracy. However, the XG Boost classifier determines the highest accuracy since it can deal efficiently with large datasets and its ensemble learning. Also, the XG Boost model was recompiled to interact with other APIs. This project can be used like any other API, but this project provides the features of the two used APIs, their security layer, and the ML algorithm. The secondary research, e.g., research papers and datasets, method is used to get all data used in the paper or for the implementation of the project. Qualitative data plays a crucial role in elucidating the characteristics and functionality of APIs- especially Cloudflare's and Akamai's APIs-. It is employed to articulate the purpose and mechanics of APIs, delineating how they function and their intended usage. The IPs were collected to be used as training data for the ML model. The data is filtered (removing the uncompleted IPs) and examined randomly to ensure its quality. The result was significant as the accuracy of this project was 97.6%. Therefore, the faults and bugs in Cloudflare's API and Akamai's API were fixed, enhancing the security of many datasets.*

I. Introduction

Due to the existence of the great growth in technologies, the need to implement a secured network is increasing. The gross usage of computerized systems has raised critical threats with hacking [1].

Hacking is a way to find the weak points of a system or network and use these points to access, edit, or gain data without legal authentication. Instead, the hackers may break down the system [2]. By 2025, cybercrime will cost the world $10.5 trillion yearly [3]. This amount is greater than the half Gross Domestic Product of Europe. In 2023, 30,000 websites are hacked daily, and 64% of companies worldwide have experienced at least one form of cyber-attack [3].

API (application programming interface) is a set of rules and protocols that allows different software applications to communicate and interact with each other. Cloudflare's API is an enormous server that increases security and reliability. "It does that by serving as a reverse proxy for the user web traffic" [3]. Also, Akamai does the same with some differences such as features. In terms of performance, Akamai cannot match the speed of Cloudflare. In addition, Cloudflare has a free plan for teams under 50, while Akamai does not offer any free plan. However, they have some common properties for example both offer a whole range of CDN services and enterprise security and content delivery solutions [8], [9]. However, both have been hacked.

Therefore, the implementation of a new security system is necessary to provide security for communication, networks, and datasets. This paper discusses the implementation of a security system made from the combination of Cloudflare's and Akamai's APIs and ML- to detect whether the IP is safe or suspicious. So, by combining them, not only the security will increase but also their features will be merged, making many varieties for the user.

For the part of IP detection, ML and deep learning algorithms, such as logistic regression, support vector machine, RNN, random forest classifier, AdaBoost classifier, and decision tree classifier, were used to determine the highest accuracy. However, the XGBC classifier (ML model) determines the highest accuracy.

After applying the XGBC classifier, the project could detect and prevent suspicious IPs. XG Boost is an accumulated learning method. And also provides more dependable explications than other machine learning algorithms. XG Boost is more durable than other ensemble classifiers and confers more high-grade performance on a variety of ML data sets. In addition, it has high performance with great accuracy [10]. It can deal with imbalanced data [10] - some classes (target labels) have significantly more examples than other classes in the training data. This appeared in the collected data as the blacklist IPs were much greater than the safe IPs.

II. Abbreviation Table

Words	Abbreviations
Machine learning	ML
Deep Learning	DL
Internet Protocol	IP
Content Delivery Network	CDN
Application programming interface	API
Distributed denial of service	DDOs
Domain Name system	DNS
Logistic Regression	LR
Recurrent Neural Network	RNN
Extreme Gradient Boosting	XGBoost
Secure Sockets layer	SSL
Transport Layer Security	TLS
Structured Query Language	SQL
cross-site scripting	XSS
web application firewall	WAF

III. Application programming interfaces (APIs)

The research question of this paper is "How to implement an impenetrable API by combining Cloudflare's and Akamai's APIs and machine learning model." Implementing such a project will increase the security of the API and provide the features of the two APIs.

1. API

API is a set of rules and protocols that allows different software applications to communicate and interact with each other as Figure (1) illustrates. It defines the methods and data structures that developers can use to build and integrate various software components without needing to understand the inner workings of each component. APIs enable developers to leverage the functionality of other software systems, services, or platforms, making it

easier to create complex applications by using pre-built building blocks. [7]

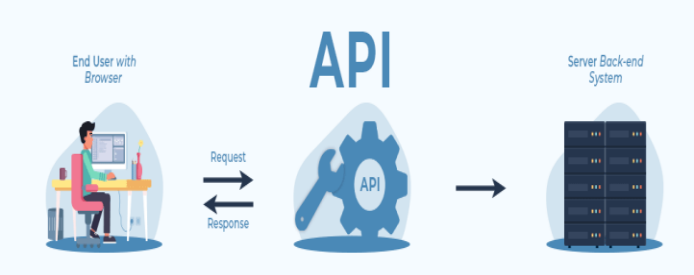


Figure (1) illustrates the mechanism of API.

APIs play a critical role in modern software development by enabling developers to access services, retrieve data, perform actions, and integrate with external systems seamlessly. They can be used for various purposes, such as retrieving data from databases, interacting with web services, controlling hardware devices, and more. APIs can be designed for different levels of abstraction, from low-level system APIs that interact with hardware components to high-level APIs that provide specific functionalities like payment processing, social media integration, or cloud services. [7]

2. Cloudflare

Cloudflare is a content delivery network (CDN) and internet security company that offers services such as content delivery, Distributed denial of service (DDoS) protection, security enhancements, and optimization tools. It operates by routing website traffic through its globally distributed network of servers as Figure (2) shows [8].

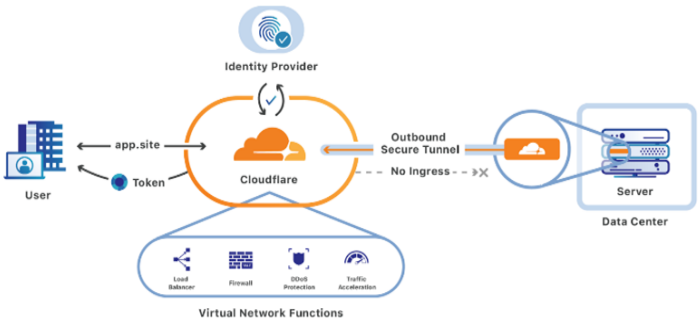


Figure (2) demonstrates the mechanism of Cloudflare [12].

3. DNS

DNS Configuration: Update your domain's DNS records to point to Cloudflare's DNS servers. Cloudflare will then manage your domain's traffic. The Domain Name System (DNS) is the phonebook of the Internet. Humans access information online through domain names, such as nytimes.com or espn.com. Web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources [15].

Each device connected to the Internet has a unique IP address which other machines use to find the device [15]. DNS servers eliminate the need for humans to memorize IP addresses such as 192.168.1.1 (in IPv4), or more complex newer alphanumeric IP addresses such as 2400:cb00:2048:1::c629:d7a2 (in IPv6).

4. Content Delivery

Cloudflare, a leading content delivery network (CDN) provider, leverages its extensive global network infrastructure to optimize content delivery and enhance the performance of websites and web applications. By strategically distributing cached content across its network of data centers worldwide, Cloudflare reduces latency and accelerates the delivery of static and dynamic web content to end-users. Through its edge server architecture, Cloudflare efficiently caches and serves web assets, including HTML pages, images, videos, and other multimedia content, ensuring fast and reliable access regardless of the user's geographical location. Additionally, Cloudflare's content delivery capabilities include intelligent routing algorithms that dynamically route traffic through the fastest and most reliable network paths, further improving response times and minimizing packet loss. With its robust content delivery network, Cloudflare empowers organizations to deliver a seamless and responsive user experience, optimize web performance, and scale their online presence to meet growing demands effectively [9].

5. Security and load balancing

Cloudflare offers robust load-balancing solutions designed to efficiently distribute network traffic across multiple servers or data centers. With customizable routing policies and advanced traffic management features, Cloudflare ensures high availability, scalability, and reliability for web applications and services. Leveraging its global anycast network, Cloudflare intelligently directs incoming requests to the nearest and most optimal server location, minimizing latency and delivering a fast user experience. Additionally, Cloudflare's load balancing features include health checks, failover mechanisms, and traffic shaping rules, enabling proactive monitoring of server health, automatic traffic rerouting during failures, and prioritization of critical traffic during peak demand. On the cybersecurity front, Cloudflare provides a robust suite of security solutions aimed at protecting websites and web applications from various cyber threats. Leveraging its global network infrastructure, Cloudflare offers distributed denial-of-service (DDoS) protection, shielding against large-scale attacks that aim to disrupt online services. Additionally, Cloudflare offers a web application firewall (WAF) that helps filter and block malicious traffic, safeguarding against common web application vulnerabilities such as SQL injection and cross-site scripting (XSS) attacks. Cloudflare's security offerings also include bot management tools to identify and mitigate automated threats, ensuring legitimate users can access online resources without interference. As illustrated in Figure (3). As Bumanglag and Kettani stated, "Moreover, Cloudflare's SSL/TLS encryption capabilities help secure data transmission between clients and servers, protecting sensitive information from interception and unauthorized access. With its comprehensive suite of load balancing and security features, backed by a global network infrastructure, Cloudflare empowers organizations to fortify their online presence, maintain operational resilience, and safeguard their digital assets against evolving cybersecurity threats while optimizing web performance and user experience." [9]

6. Akamai:

Akamai stands as a leading content delivery network (CDN) provider. Their reputation is built upon a comprehensive suite of services meticulously designed to refine the entire internet experience. At the core of their offerings lies a powerful combination of optimized content delivery and application acceleration. This translates to a user experience characterized by lightning-fast loading times, smooth web interactions, and robust security measures across the web. To achieve these results, Akamai incorporates a range of performance optimization features. These include asset compression, image optimization, and resource minification, all working in concert to dramatically reduce load times for websites and applications. The tangible benefit? A seamless and responsive user experience – a critical factor in driving user engagement and satisfaction. But Akamai's expertise extends beyond content delivery. They are specialists in application acceleration as well. This goes beyond simply delivering content quickly. By employing advanced techniques like caching, compression, and route optimization, Akamai meticulously minimizes latency and enhances the responsiveness of web applications and APIs. The results are undeniable: a significant improvement in user experience, a surge in user engagement, and ultimately, business growth for organizations that leverage the power of Akamai's services [9].

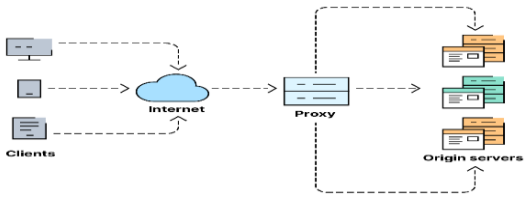


Figure (3) illustrates how Cloudflare balances the load [13].

7. Edge Server

Akamai's edge server configuration stands as a cornerstone of its global network, meticulously designed to facilitate efficient content delivery across the digital landscape. With thousands of strategically positioned edge servers dispersed throughout data centers worldwide, Akamai ensures that content is seamlessly caught and served to end-users with exceptional reliability and performance. These edge servers are meticulously configured to optimize content delivery, leveraging caching mechanisms to store frequently accessed content locally. By doing so, Akamai minimizes latency and accelerates content delivery, guaranteeing swift access to resources regardless of users' geographical locations. This strategic approach not only enhances user experiences but also bolsters overall reliability, as Akamai's edge servers are adeptly prepared to handle peak traffic periods and unforeseen surges in demand. Through the utilization of Akamai's edge server infrastructure, organizations can confidently deliver content and applications with minimal latency and maximum availability, thereby establishing a robust digital presence capable of meeting the dynamic needs of modern users [9].

8. Security Solutions

Security Solutions: Akamai offers security solutions like DDoS mitigation and bot protection. Integrate these solutions to enhance security alongside Cloudflare's offerings [9] as Figure (4) shows.

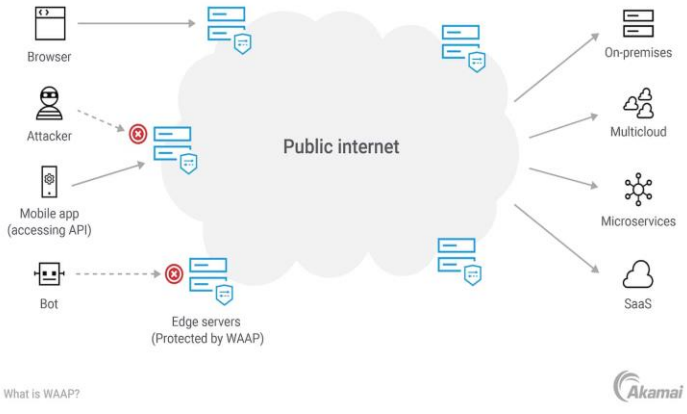


Figure (4) provides what Akamai's API protect

IV. XGBoost:

XGBoost stands for Extreme Gradient Boosting. Some optimizations used include regularized model formalization to prevent overfitting and tree pruning to reduce model complexity. Due to its efficient tree-boosting algorithm and regularized Model technique, XGBoost models often achieve better accuracy than other machine learning algorithms. The models can handle complexity through hyperparameters like learning rate and number of boosting iterations. The most important factor behind the success of XGBoost is its scalability in all scenarios. The system runs more than ten times faster than existing popular solutions on a single machine [16]. For example, testing the SVM (support vector machine) model on the data took more than an hour to run, but XGBoost ran in 41 seconds. XGBoost model scales billions of examples in distributed or memory-limited settings. The scalability of XGBoost is due to several systems and algorithmic optimizations. These innovations include the following: a novel tree learning algorithm for handling sparse data; and a theoretically justified weighted quantile sketch procedure that enables handling instance weights in approximate tree learning. Parallel and distributed computing makes learning faster, enabling quicker model exploration [16]. Furthermore, it is an ensemble learning method; in other words, XGBoost combines the predictions of multiple weak models to produce a stronger one. All the above make the XGBoost robust and improve its accuracy.

V. Methods

i. Integration Between Akamai's and Cloudflare's API:

Combining Akamai, Cloudflare, and AI can create a powerful solution that enhances the performance, security, and intelligence of your web applications. To create this new complex security, the application needs, traffic patterns, and potential

AI Cases must be understood in the network. The code snippet is a Python script that integrates data from Akamai and Cloudflare, two content delivery network (CDN) providers. The script fetches data from Akamai using its API, applies AI-generated insights to this data, and then is expected to apply these insights to Cloudflare using its API.

Importing Libraries: The script imports several Python libraries, including requests for making HTTP requests, pandas to load the data in data type pandas data frame, sklearn for machine learning tasks, XGBoost for gradient boosting, and imblearn for imbalanced data handling as shown in Figure (5).

```
import requests
import openai
import pandas as pd
from sklearn.model_selection import train_test_split,
GridSearchCV, StratifiedKFold
from sklearn.datasets import make_classification
from sklearn.pipeline import Pipeline
from xgboost import XGBClassifier
from sklearn.feature_selection import SelectKBest, chi2

from imblearn.over_sampling import SMOTE
```

Figure (5) shows the Imported libraries.

From Figure (6): Akamai and Cloudflare API Configuration: Configuration parameters for Akamai and Cloudflare API endpoints and API keys are defined at the beginning of the script.

```
# Akamai API configuration
akamai_api_url =
"https://api.example.com/akamai/data_endpoint"
akamai_api_key = "your_akamai_api_key"

# Cloudflare API configuration
cloudflare_api_url =
"https://api.cloudflare.com/client/v4/"
cloudflare_api_key = "your_cloudflare_api_key"
```

Figure (6) shows the code for the APIs

In Figure (7) get_akamai_data Function: This function sends an HTTP GET request to the Akamai API endpoint using the provided API key for authorization. It expects a JSON response and returns the fetched data.

```
# Function to retrieve data from Akamai API
def get_akamai_data():
    headers = {"Authorization": f"Bearer
{akamai_api_key}"}
    response = requests.get(akamai_api_url,
headers=headers)
    data = response.json()
    return data
```

Apply_insights_to_cloudflare Function as shown in Figure (8): This function is a placeholder and lacks implementation. It is intended to apply insights generated by AI to Cloudflare. The specific logic for interacting with the Cloudflare API and applying insights needs to be implemented within this function.

```
# Function to apply insights to Cloudflare
def apply_insights_to_cloudflare(insights):
    # Implement Cloudflare API requests based on insights
    Pass
```

generate_ai_insights Function: This function is also a placeholder and lacks implementation. Its purpose is to generate AI-driven insights based on the data obtained from Akamai. The specific AI

Figure (8) illustrates the code to apply insights to Cloudflare's API.

logic for generating insights is missing in the code, and it needs to be implemented. Figure (9)

```
# Function to generate AI insights (you need to
implement this)
def generate_ai_insights(data):
# Use AI (GPT-3 or your choice) for insights
# Implement your AI logic here and return insights
    Pass
```

Main Function: The main function is the entry point of the script. It calls get_akamai_data to retrieve data from Akamai and stores it in the akamai_data variable. It then calls generate_ai_insights to generate AI-driven insights based on the Akamai data. However, the implementation of this function is incomplete, so it does not currently generate any insights. Finally, it calls apply_insights_to_cloudflare to apply these insights to Cloudflare, although this function is also incomplete and does not perform any actual actions on Cloudflare. Figure (10).

Figure (9) ai meta code to connect AI script with the main script

```
# main function

def main():
    # Retrieve data from Akamai API
    akamai_data = get_akamai_data()

    # Generate AI insights
    ai_insights = generate_ai_insights(akamai_data)

    # Apply AI insights to Cloudflare
    apply_insights_to_cloudflare(ai_insights)

if __name__ == "__main__":
    main()
```

This code serves as a framework for integrating data from Akamai, applying AI-generated insights, and potentially making changes to Cloudflare based on these insights. However, significant implementation work is required for the generate_ai_insights and apply_insights_to_cloudflare functions to make the script functional. Additionally, any AI model or logic used for generating insights needs to be incorporated into the code.

ii. Machine learning implementation:

Data Collection and Preparation: The data is collected from a programmer in a cybersecurity company from Git Hub. The collected data was in the form which is in Table (1); there is an ML rule that states that when the number of features increases, the accuracy is enhanced.

IP	Case (safe or suspicious)
18.148.223.130	Safe
75.39.229.204	Safe
154.41.195.168	Safe

Table (1) illustrates the format of the collected data.

XG boost model requires one data type as an input. For example, the collected data has IPs that have "*", which the model considers a string (data type in Python). This obstacle appeared after the features had been increased as shown in Table (2). The data that will estimate the highest accuracy should be integers- to make the model determine relations. So, it was an obstacle. The code in Figure (11) shows how those two obstacles were defeated.

```
for i in range(len(df)):

    if "*" not in str(df.iloc[i,0]):
        x = str(df.iloc[i,0]).split(".")
        b=[]
        if len(x)==4:
            for i in range(len(x)):
                b.append(x[i])

            out_csv.append(b)
            b.append(df.iloc[i,1])
```

Figure (11) illustrates the code to overcome the mentioned obstacles.

The code iterated on all the IPs. It initially checked that there was no "*" in the IP. Then, it used the string method "split", which splits the IP into 4 numbers- meaning four features. The non-existence of the "." won't affect the accuracy since it is constant in all IPs, and the same will happen to any IP that the model will detect. leading, the collected data to be in the form illustrated in Table (2). Also, the cases -safe and

Figure (10) import the startup code to the whole

suspicious- were replaced by 1 and 0 respectively as that enhanced the model.

IP1	IP2	IP3	IP4	Case
18	148	223	130	1
3	60	243	123	0
127	147	158	152	0

Using machine learning technology by giving some data "blocked data - safe data" - so the AI detects the blocked data and prevents it from accessing the application or the network also the AI will predict if the blocked data changed or got a new shape [5]. In that case, those data were gathered to check Internet protocols instead of predicting them

Table (2) illustrates the final form of the collected data after editing. Where 1 means safe and 0 means suspicious

[6]. The dataset of suspicious and safe IPs was collected (qualitative data). The number of IPs is greater than 30 thousand. The data is analyzed by using content analysis methods. The code in Figure (12) the “pandas” library to read the dataset. and “df.isnull().sum()” is used to check that there are no null cells in the data frame. “df.info()” is used to check that the type of data in the columns is integers and how many entries.

```
import pandas as pd
df = pd.read_csv('save.csv')
x = df.drop("case", axis=1)
y = df["case"]
print(df.isnull().sum())

print(df.info())
```

Figure (12) illustrates the code that helps to examine the data, check its preparation for the model, and split the data into dependent and independent.

Also, “df.drop("case", axis=1)” and “df["case"]” divided the data into two categories X (The four parts of the IPs- independent variables) and Y (safe or suspicious- dependent variables). Then, an XG boost model- a type of ML- was implemented. The model trained on that data. Finally, the ML algorithm was recompiled to interact with other APIs.

The code in Figure (13) does the following: handles the class imbalance using SMOTE to oversample the minority class and balance the classes, creates a classification pipeline with XG Boost Classifier as the model, Grid Search CV will evaluate all combinations of hyperparameters -max-depth and n-estimators- defined and return the one with the best validation score, and The final model is selected based on hyperparameters that perform best on the held-out validation data during grid search.

```
# Handle class imbalance with SMOTE
rus = SMOTE(sampling_strategy={0:8974 , 1:8974 })
X_res, y_res = rus.fit_resample(x, y)

# Split data
X_train, X_test, y_train, y_test = train_test_split(X_res,
y_res, test_size=0.1, random_state=1000)

# Model selection pipeline
clf = Pipeline([('feature_selection', fea_sel),
          ('classification',
XGBClassifier(objective='binary:logistic',
                        n_estimators=500, max_depth=8))])
# Hyperparameter tuning
parameters = { 'feature_selection__k': list(range(1,4)),

          'classification__max_depth': [3,5,7],
          'classification__n_estimators': [100,300,500]}

cv = StratifiedKFold(n_splits=5)
grid = GridSearchCV(clf, parameters, cv=cv, n_jobs=-1,
verbose=1)

# Fit the model
grid.fit(X_train, y_train)
```

Figure (13) illustrates the model code and some methods to increase the accuracy.

Integrate Cloudflare, Akamai, and machine learning. will involve leveraging APIS " AKAMAI API - CLOUDFLARE API ". and hooks provided by these platforms to incorporate AI-driven decisions.

iii. Monitoring and Analytics:

Using Akamai and Cloudflare's monitoring tools to gain insights into applications - network performance, user behavior, and security threats.

Integrate analytics tools to track user engagement, conversion rates, and other relevant metrics.

iv. Redundancy:

Configuring failover mechanisms using both Akamai and Cloudflare's load balancing and traffic management features. Ensure that if one service experiences downtime, traffic seamlessly shifts to the other without major disruptions.

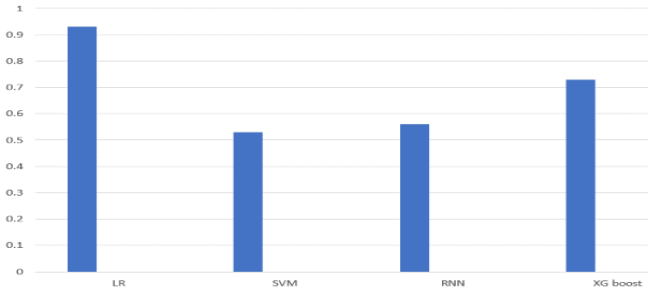
v. Optimization:

Recompilation of XGBoost -ML- model based on feedback and performance data and added it to the integration of APIs. Finally, testing it on different strategies using Cloudflare's API tester, which allows to test the project on a real website.

VI. Results

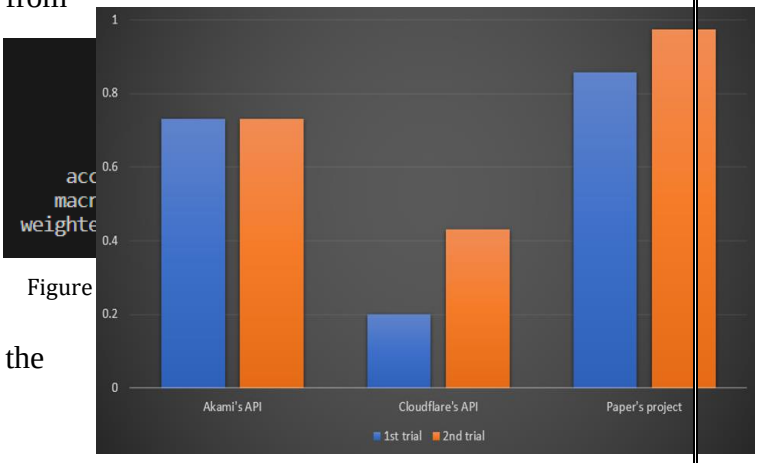
i. Negative results:

One of the greatest obstacles that faced the project was achieving a high accuracy of the ML model. For example, the data was imbalanced, so many of the tested ML algorithms had biases in their prediction. Graph (1) shows the accuracy of each model. So, it may be eccentric that although logistic



Graph (1) shows the accuracy of different ML models.

regression (LR) has the highest accuracy, it was not used. That is because the data had blacklist IPs much greater than safe IPs, and LR can't deal with imbalanced data, so there were biases in the data. For instance, when the RL printed its predictions, none of them was safe. On the other hand, by examining the accuracy report of the XG boost classifier as shown in Figure (14), it can be easily determined from the precision that the model predicts both categories according to what it learned from



Figure

the

Graph (2) illustrates the accuracy of different trials for APS.

training data. in other words, no bias exists.

ii. Positive results:

The three APIs were tested using the Cloudflare tester to test it on a real website. Firstly, Each API is tested individually. Cloudflare -individually- started with 20% accuracy and ended with 43%. Akamai's API's highest accuracy was 73% even though it was running for 2 hours- usually when the runtime increases, the accuracy increases. When it comes to the paper's project, the first attempt was 85.7%. However, it ended with 97.5%. Graph (2) includes all results, showing how much the paper's project's accuracy is greater than other APIs.

VII. Discussion

From the result section, it can be inferred that the project addressed the research question. Furthermore, it provides more features to the users. The project is highly significant for organizations that look for protection of their digital assets and data, enhance their online presence, and mitigate various cyber threats. Also, leverage both Cloudflare and Akamai security features to create a multi-layered security approach. Implement Distributed Denial-of-Service Attack " DDoS " mitigation, bot protection, and OWASP " The OWASP Foundation " top 10 security measures through both platforms. Making them work together will be more secure than each of them alone as they

filter out malicious traffic and ensure that the web servers remain available and responsive during an attack. In addition to the two APIs that can detect anything anomalous, the ML model also supports them to do that effectively. The project system can analyze login attempts and detect patterns that suggest cyberattacks, helping to prevent unauthorized access to systems and accounts.

Limitations:

The training data has IPv4. Now, there is IPv6, so when an IPv6 sends requests, machine learning won't be practical in this case since the ML trained on data has IPv4. However, that won't have strong negative impacts as IPv6 isn't familiar nowadays. Also, that is the reason why the chosen training data is IPv4.

As the system is complicated, it requires experts to be ready for any unexpected errors. Although the system has high accuracy, the user should always be ready for anything.

Recommendations:

Although the familiar IPs are IPv4, the security system should be prepared for all cases. So, it is recommended to train the machine learning model on training data of IPv6. The main cause that this paper didn't use IPv6 in the training data is there is no huge dataset of IPv6.

VIII. Conclusion

Nowadays, hacking is increasing. Also, many security systems are hacked, leading to losing data and money. Therefore, the significance of this paper appears since it discusses the implementation of a new security system. It is the combination of Cloudflare's and Akamai's APIs and machine learning. This system has the security layers and features of the two APIs. The two APIs were combined. Then, the recompiled machine learning algorithm was added to them. The XG boost model -machine learning- role is to determine whether the IP is safe or suspicious. The collected data of the ML model is greater than 30 thousand IPs. Although the collected data is imbalanced and large, XG boost deals with it effectively. In addition, it is an ensemble learning method, which increases the accuracy. Then, the project was tested on a real website to simulate its real implementation. The project proved its competence as its accuracy is 97.6%. This accuracy is too high when compared with the accuracy of Cloudflare's API or Akamai's API alone. Finally, the project provides the features of both APIs, their security layer, and the ML to ensure security and enhance the APIs' industry.

IX. References

[1] R. Nair, C. P. Kasula, S. Vankayala, and N. Chakraborty, "IP network anomaly detection using machine learning," *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)*, 2019. doi:10.1109/i2ct45611.2019.9033545

[2] A. Gupta and A. Anand, "Ethical hacking and hacking attacks," *International Journal Of Engineering And Computer Science*, 2017. doi:10.18535/ijecs/v6i4.42

[3] R. Vardhman, "How many cyber attacks happen per day in 2023?," Techjury, https://techjury.net/blog/how-many-cyber-attacks-per-day/#:~:text=Globally%2C%2030%2C000%20websites%20are%20hacked,ransomware%20cases%20grew%20by%2092.7%25. (accessed Aug. 22, 2023).

[4] E. Nygren, R. K. Sitaraman, and J. Wein, "Networked Systems Research at akamai," ACM SIGOPS Operating Systems Review, vol. 44, no. 3, pp. 1–1, 2010. doi:10.1145/1842733.1842735

[5] S. K. Jauhar et al., "How to use no-code artificial intelligence to predict and minimize the inventory distortions for Resilient Supply Chains," International Journal of Production Research, pp. 1–25, 2023. doi:10.1080/00207543.2023.2166139

[6] J. Ofoeda, R. Boateng, and J. Effah, "Application programming interface (API) research," International Journal of Enterprise Information Systems, vol. 15, no. 3, pp. 76–95, 2019. doi:10.4018/ijeis.2019070105

[7] M. Nadeem et al., "Preventing cloud network from spamming attacks using Cloudflare and KNN," Computers, Materials & Continua, vol. 74, no. 2, pp. 2641–2659, 2023. doi:10.32604/cmc.2023.028796

[8] E. Nygren, R. K. Sitaraman, and J. Wein, "Networked Systems Research at akamai," ACM SIGOPS Operating Systems Review, vol. 44, no. 3, pp. 1–1, 2010. doi:10.1145/1842733.1842735

[9] A. Garlapati, D. R. Krishna, K. Garlapati, and G. Narayanan, "Predicting employees under stress for pre-emptive remediation using machine learning algorithm," *2020 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, 2020. doi:10.1109/rteict49044.2020.9315726

[10] S. Sanghi, "Unraveling the magic of apis: A beginner's guide to integration superpowers," DEV Community, https://dev.to/ainasanghi/unraveling-the-magic-of-apis-a-beginners-guide-to-integration-superpowers-c39 (accessed Sep. 15, 2023).

[11] Cloudflare, "Self-hosted applications · Cloudflare Zero Trust Docs," Self-hosted applications · Cloudflare Zero Trust docs, https://developers.cloudflare.com/cloudflare-one/applications/configure-apps/self-hosted-apps/ (accessed Sep. 17, 2023).

[12] "What is load balancing?," NetScaler, https://www.netscaler.com/articles/what-is-load-balancing (accessed Sep. 17, 2023).

[13] Akamai, What is application security? | Akamai, https://www.akamai.com/glossary/what-is-application-security (accessed Sep. 17, 2023).

[14] K. Bumanglag and H. Kettani, "On the Impact of DNS Over HTTPS Paradigm on Cyber Systems," *2020 3rd International Conference on Information and Computer Technologies (ICICT)*, 2020. doi:10.1109/icict50521.2020.00085

[15] T. Chen and C. Guestrin, "XGBoost," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016. doi:10.1145/2939672.2939785

X. Appendix

A. Training dataset

[1] Miroslav Stampar, "IPs collection," 2024. [Online]. Available: https://github.com/stamparm/ipsum/tree/master/levels