# Password Spray Attack Detection And Simulation Using Python

Nagalakshmi. R ,Ms.V.Yogashri

*Department of Computer Science, Rathinam College of Arts and Science (Autonomous), Coimbatore, Tamil Nadu, India.*

***Abstract*** *- Password spray attacks are a common form of cyberattack in which attackers attempt to gain unauthorized access by using a single commonly used password across multiple user accounts. Unlike traditional brute-force attacks, this method avoids account lockouts and remains difficult to detect using conventional security mechanisms. This project focuses on the detection and simulation of password spray attacks using Python. The system simulates login attempts across multiple user accounts and analyses authentication logs to identify suspicious patterns, such as repeated use of the same password for different usernames within a short period of time. By applying threshold-based detection techniques, the system can effectively flag potential password spraying activities. The proposed solution collects and processes login data including username, password, IP address, and timestamp. It then applies grouping and analysis algorithms to detect anomalies and generate alerts when predefined thresholds are exceeded. The simulation helps in understanding attacker behaviour and evaluating the effectiveness of detection strategies. This project enhances cybersecurity by providing a simple yet effective method to identify password spray attacks in real-time environments. It can be extended with additional features such as IP tracking, time-based analysis, and automated alert systems for improved security monitoring.*

***Keywords*** *-Password Spray Attack, Cyber Security, Python Simulation, Intrusion Detection, Authentication Security.*

## 1.Introduction

In the modern digital world, user authentication systems play a crucial role in protecting sensitive information from unauthorized access. With the rapid growth of online platforms, cyber threats targeting login systems have also increased significantly. Among these threats, password-based attacks remain one of the most common and effective methods used by attackers. One such attack is the password spray attack, where an attacker attempts to access multiple user accounts using a single commonly used password. Unlike brute-force attacks, which target one account with many passwords, password spraying spreads attempts across many accounts, making it difficult to detect using traditional security systems. This low-frequency attack method helps attackers avoid account lockouts and detection mechanisms. Existing security systems mainly rely on rule-based or signature-based detection techniques, which are often ineffective against new or unknown attack patterns. As a result, password spray attacks can go unnoticed and lead to serious security breaches, including unauthorized access, data theft, and system compromise. To address this issue, this project proposes a Password Spray Attack Detection and Simulation System using Python. The system is designed to simulate password spray attacks and monitor login attempts in real time. By analysing patterns such as repeated password usage across multiple accounts and abnormal login behaviour, the system can detect potential attacks and generate alerts. This approach not only helps in understanding how password spray attacks work but

also provides an effective method to detect and prevent them. The implementation using Python ensures simplicity, flexibility, and ease of integration with existing systems. Overall, the project aims to enhance cybersecurity by providing a proactive solution for detecting and mitigating password spray attacks.

## 2.Literature Review

Recent studies in cybersecurity highlight the increasing threat of password-based attacks, particularly password spray attacks, which attempt a small number of common passwords across multiple accounts to avoid detection. Traditional security mechanisms such as account lockout policies and basic intrusion detection systems often fail to effectively identify these attacks due to their low-frequency nature. Several researchers have proposed log-based analysis techniques to detect abnormal login behaviour by monitoring failed login attempts, IP address patterns, and time intervals. Machine learning approaches have also been explored to improve detection accuracy by classifying login activities into normal and malicious categories. Techniques such as anomaly detection, clustering, and supervised learning models have shown promising results in identifying hidden attack patterns.

In addition, security frameworks have incorporated real-time monitoring systems that analyse authentication logs and generate alerts when suspicious activity is detected. Simulation-based approaches are also used to evaluate system performance by generating synthetic attack data. Despite these advancements, challenges remain in reducing false positives and improving detection efficiency. This study builds upon existing research by implementing a Python-based detection system that combines log analysis, feature extraction, and simulation of password spray attacks to enhance detection accuracy and system performance. The proposed approach aims to provide a simple, efficient, and scalable solution for identifying password spray attacks in real-time environments.

## *3.Methodology*

(i)**Data Collection:** Login data is collected from system logs or generated using a simulation environment, including attributes such as username, IP address, timestamp, and login status.

(ii)**Data Pré-processing**: The collected data is cleaned by removing duplicates and inconsistencies. Relevant fields are selected, and the data is transformed into a structured format. Normalization and noise reduction techniques are applied to improve data quality.

(iii)**Feature Extraction**: Important features such as number of failed login attempts per IP address, login frequency, and time intervals between attempts are extracted to identify attack patterns.

(iv)**Attack Simulation**: A simulation module is implemented to mimic password spray attacks by trying common passwords across multiple user accounts to test the system.

(v)**Detection Mechanism**: The system analyses login patterns using anomaly detection and pattern recognition techniques to identify suspicious activities.

(vi)**Classification**: Based on the analysis, the system classifies activities into two categories: normal login and password spray attack.

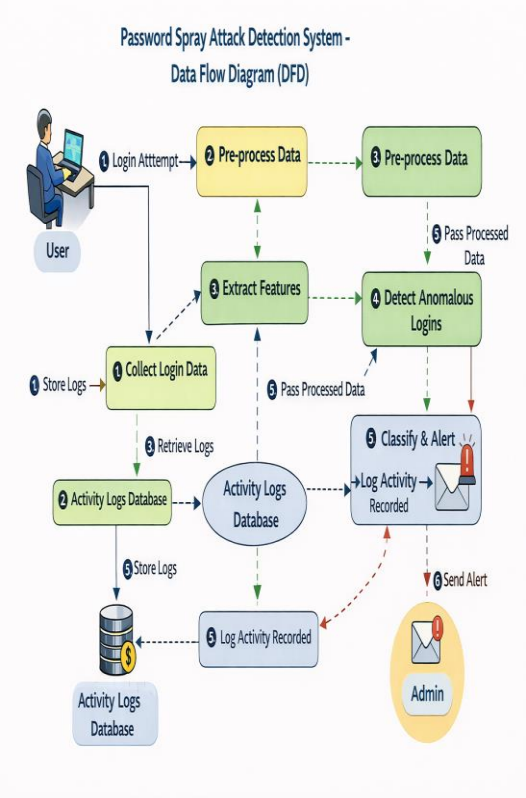
(vii)**Alert Generation**: If a potential attack is detected, the system generates alerts and notifies the administrator for immediate action.

(viii)**Result Visualization**The final results are displayed on an admin dashboard for real-time monitoring and analysis.

## *4.System Architecture*

The system architecture defines the overall structure and interaction between different modules of the password spray attack detection system. It follows a layered approach where login data is collected and passed through preprocessing and feature extraction stages. The processed data is then analysed by the detection module to identify abnormal login patterns. A simulation module is used to generate attack scenarios for testing. Based on the analysis, the system classifies activities as normal or malicious and generates alerts accordingly. All results are stored in a database and displayed through an admin dashboard for real-time monitoring and control.

1. **Input Layer:** User login attempts, system log data
2. **Data Collection Layer:** Collects login details (username, IP, timestamp, status)
3. **Pre-processing Layer:** Data cleaning, Filtering, Normalization
4. **Feature Extraction Layer:** Extracts patterns like failed attempts, time intervals
5. **Simulation Layer**: Enervates password spray attack scenarios
6. **Detection Layer:** Identifies abnormal login behaviour
7. **Classification Layer:** Classifies as normal or attack
8. **Alert Layer:** Sends alerts to admin
9. **Storage Layer:** Stores logs and results in database
10. **Presentation Layer:** Admin dashboard for monitoring




## 5.Implementation

The implementation of the Password Spray Attack Detection and Simulation system is carried out using Python with a modular approach. Initially, login data is collected from system logs or generated through a simulation environment, containing details such as username, IP address, timestamp, and login status. The collected data is pre-processed by removing duplicate and incomplete records, followed by organizing it into a structured format for analysis. Feature extraction techniques are then applied to derive important attributes such as the number of failed login attempts per IP address, login frequency, and time intervals between attempts.

A simulation module is developed to replicate password spray attacks by attempting common passwords across multiple user accounts. The detection mechanism is implemented using rule-based and anomaly detection methods to identify suspicious login behaviour. If abnormal patterns are detected, the system classifies the activity as either normal or malicious. Finally, alerts are generated and displayed on the admin dashboard, enabling real-time monitoring and quick response to potential threats. This implementation ensures efficient and accurate detection of password spray attacks.

**1. Language Used**

- **Python**
  Chosen for its simplicity and powerful support for data analysis and cybersecurity applications.

**2. Tools Used**

- **Jupiter Notebook / VS Code / PyCharm** – For coding and testing
- **CSV / Log Files** – For storing login data
- **Operating System Logs** – For real-time data (optional)

**3. Libraries Used**

- **pandas** – Data handling and preprocessing
- **NumPy** – Numerical operations
- **matplotlib / seaborn** – Data visualization
- **datetime** – Time-based analysis
- **scikit-learn (optional)** – For anomaly detection or classification

**4. Basic Logic**

- Read login data (CSV/log file)
- Count failed login attempts per IP
- Check multiple account attempts from same IP
- Analyse time interval between attempts
- If threshold exceeded → mark as attack
- Else → normal login

## 6.Appendix

The appendix section provides additional supporting information related to the implementation of the password spray attack detection and simulation system using Python. It includes sample source code that demonstrates how multiple login attempts are simulated using a list of common passwords across several user accounts, along with a basic detection mechanism based on a predefined threshold for failed attempts. A sample dataset is also presented to illustrate login attempts and their corresponding outcomes, such as success or failure. Furthermore, example output results are included to show how the system identifies and alerts potential password spray attacks.

In addition, the appendix outlines the tools and technologies used in the project, including Python programming language and libraries such as Pandas, NumPy, and Matplotlib for data handling and visualization. The system requirements are also specified, covering both hardware and software needs necessary for executing the project efficiently. Finally, additional notes are provided to explain that the detection logic is based on repeated failed login attempts and can be further enhanced using advanced techniques such as machine learning for improved accuracy and real-time threat detection.

## 7.Referances


1. *International Journal of Advanced Computer Science and Applications*, 10(4), 499–508.
2. Behl, A., Behl, K., & Behl, K. (2017). *Cybersecurity and Cyberwar: What Everyone Needs to Know*. Oxford University Press.
3. Stallings, W. (2018). *Network Security Essentials: Applications and Standards* (6th ed.). Pearson Education.
4. Scarfone, K., & Mell, P. (2007). Guide to intrusion detection and prevention systems (IDPS). *National Institute of Standards and Technology (NIST)* Special Publication 800-94.
5. MITRE Corporation. (2023). *Credential Access Techniques – Password Spraying (T1110.003)*. Retrieved from https://attack.mitre.org
6. OWASP Foundation. (2022). *Authentication Cheat Sheet*. Retrieved from https://owasp.org
7. Sommer, R., & Paxson, V. (2010). Outside the closed world: On using machine learning for network intrusion detection. *IEEE Symposium on Security and Privacy*, 305–316.
8. Ahmed, M., Mahmood, A. N., & Hu, J. (2016). A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60, 19–31.
9. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, 41(3), 1–58.