Extending Port Knocking Authorization with Deception Mechanisms

ISSM-581: Research Project

Spring 2021

Sairam Kumar Bitla (sbitla@student.concordia.ab.ca)
Sai Pramukha Siripuram (ssiripur@student.concordia.ab.ca)
RamaChandini Devarapalli(rdevarap@student.concordia.ab.ca)
DivyaNelakurthi (dnelakur@student.concordia.ab.ca)

Research Project

Submitted to the Faculty of Graduate Studies
Concordia University of Edmonton
Edmonton, Alberta

In Partial Fulfilment of the
Requirements of ISSM-581 course

Advisor: Dr Sergey Butakov (sergey.butakov@concordia.ab.ca)
Department of Information Systems Security Management
Concordia University of Edmonton,
Edmonton T5B 4E4, Alberta, Canada

Extending Port Knocking Authorization with Deception Mechanisms

Sairam Kumar Bitla (sbitla@student.concordia.ab.ca)
Sai Pramukha Siripuram (ssiripur@student.concordia.ab.ca)
RamaChandini Devarapalli(rdevarap@student.concordia.ab.ca)
Divya Nelakurthi (dnelakur@student.concordia.ab.ca)

Approved:

Sergey Butakov [Original Approval on File]

Sergey Butakov Date: June 23, 2021

Primary Supervisor

Patrick Kamau [Original Approval on File]

Patrick Kamau, PhD, MCIC, PChem. Date: June 23, 2021

Dean, Faculty of Graduate Studies

Table of Contents

List of Figures

List of Tables

Extending Port Knocking Authorization with Deception Mechanisms

Sairam Kumar Bitla (sbitla@student.concordia.ab.ca)
Sai Pramukha Siripuram (ssiripur@student.concordia.ab.ca)
Rama Chandini Devarapalli (rdevarap@student.concordia.ab.ca)
Divya Nelakurthi (dnelakur@student.concordia.ab.ca)
Advisor: Sergey Butakov (sergey.butakov@concordia.ab.ca)

***Abstract* – Authentication /Authorization mechanisms are always a potential target for an attacker. Port Knocking and its variations such as Single Packet Authentication (SPA) allow administrators to put additional shield on network interfaces with sensitive services. The existing SPA solutions are prone to various attacks, such as key leaks on the client side or due to man-in-the-middle attacks. To mitigate this vulnerability problem, decoy keys can be used in the storage and transmission operations. These keys are encoded using public-private key encryption to protect confidentiality of the transmitted data. This additional layer of security decreases the chance of single point of failure through key leakages in generic port-knocking and single packet authorization schemes. To measure the potentially impact of new additions on the usability of an SPA system the experiment was conducted that allows to check the extra time required for authentication. The experiment indicated there is no noticeable negative impact on the timing which allows utilizing decoy keys in SPA systems.**

Keywords - Network Security, Port Knocking, Single Packet Authorization, Deception Mechanisms, Decoying, Honey keys, Honeywords, Honey Pot, Man-in-the-middle.

I. INTRODUCTION

Security of network services has become one of the primary concerns for any connected enterprise. Due to the rapid growth in technology and data sharing, people are connected now more than ever. Creating a balance between protection, privacy and usability of the system is one of the challenges that come with this exponential development. Because of the increasing dependency on the cyber systems, securing the critical infrastructure is vital in ensuring the access to the crucial services. If these key services are hacked, it could pose a serious threat to the economies and communities of the industry. For example, according to an incident reported on May 11, 2021, where a cyber-attack knocked an essential U.S gasoline pipeline - Colonial Pipeline offline in demand of ransom indicated the threat to the cybersecurity is rapidly spreading across various industries and sectors. These attacks are becoming more numerous and widespread as millions of individuals around the world, in some cases exposing back doors to networks that lack organizational or institutional security protection[1]. Because of the exponential advances in technology, IT and the security demands are rising in lockstep, necessitating the addition of an extra layer of security to networks to reduce the risk of cyber attacks. This layer of protection can be accomplished by placing a shield over the server's administrative entry and briefly lifting the shield when reached by a legitimate user using a hidden non-repeatable function.

Most web protection schemes are two-step in nature. The first stage is authentication, which is about verifying the user's identity, and the second stage is permission, which grants access to the user to various resources depending on their identity. To support program deployment and maintenance, modern operating systems depend on well-designed authorization processes.

Authentication is a crucial factor in the security of network services. It allows the control system to identify each other involved in the communication before being part of any activity. Authentication is the “access point” to the communication and hence is a potential pick out to the attackers. This implies that the entire authentication process must be safeguarded by protocols and rules that justify the mechanism of authentication. Authorization is a protection method for deciding the level of access or user/client rights relevant to device resources, including databases, utilities, computer systems, data, and application functions. This is the method of granting or refusing access to a network resource that gives the user access to a range of services depending on the user’s identity [2].

Confidentiality, Integrity, and Authentication [CIA] factors have become a major concern for sensitive traffic during communication. Without protocols/ mechanisms, the flow of information sent over the network is unsure and that could be vulnerable to unauthorized disclosure and modification [3]. The conventional method of proving the identity is based on “something that principal knows (password or passphrase), something the principal is (biometric values) or something the principal has (identity card or smart card)” [4].

Port-knocking (also known as Spread Spectrum TCP) is an authorization technique that relies on something the principal knows [4]. In this scenario, it is not the traditional password sequence the principal has, but it is the port sequence which is the characteristic trait of such a mechanism.

II. LITERATURE REVIEW

A. Port Knocking

Port Knocking (PK) is the method of relaying the information across the closed ports in the network to authorize users before allowing them to access the protected service. The server which is willing to communicate keeps all its ports closed (using a firewall) until they are knocked with a predefined set of sequence in correct order to establish a connection by opening the desired communication port. The server is set to default DROP stance which when receives packet drops them silently. However, the server logs connection attempts made against the closed port. An external process (daemon) computes these logs and checks for the correct sequence. Once it recognizes the valid knock, the process updates the connection policies of the firewall and allows the connection from the client to a specific required service. To perform this the client IP address and the port to be opened must be encoded in the port sequence.

Port knocking is a firewall installed security mechanism that adds another layer of authentication and helps to reduce the data available from malicious scans. Port knocking is used to deter an intruder from inspecting a device for exploits by performing a port search and the ports tend to be closed before the intruder delivers the correct knock chain. Using this technique to increase security has several advantages. However, there are certain significant draw backs that need to be resolved [5].

1) Advantages of Port Knocking:

The attacker needs to identify proactively even before the port knocking device is already functioning, to begin attacking the system. In most situations, it becomes difficult for the attacker because one of the objectives of port knocking is concealment.

Port knocking provides additional layer of security to the network. As part of the defense in depth technique, port knocking is used. If port access is successfully achieved by the intruder, there are other port protection systems that are already in operation, along with the service authentication mechanism assigned to open ports.

It takes a large-scale brute force attack to defeat port knock protection to accomplish even a simple sequence. In the 1-65535 port range, an attacker will have to test each three-port combination to launch an attack against a three-knock TCP series. The stateful behavior of the port knocking helps multiple clients to authenticate IP addresses from various sources. Simultaneously, allowing the legitimate user along the firewall when the firewall itself in the center of the port attack from various IP addresses. The firewall ports would still appear to be closed to any attacking IP address [2].

The basic implementation of PK has some fundamental flaws that anyone who monitors the network traffic between the client and the server will be able to deduce the port sequence and can use this sequence to replay a connection attempt to the server and its services. It also has the problem of out of order delivery, usage of Network Address Translators (NAT), an association of authentication and connection, failure of Daemon, and the ability to transmit the data.

2) Limitations of Port Knocking

Delivery of Out-of-Order Packets: To read the sequence correctly, the port knock sequence must be in the right order. According to the shortest path calculation, the packets will take different paths in the network to receive the destination. Some packets may take longer to reach destination due to traffic. Due to that the order of receiving the packets might change in the knock sequence. If a knock is received out of order, the sequence is interrupted and no operation is performed by the server, results in denial of service.

If an attacker wants to launch a denial-of-service attack on a single network, they will use a spoofing attack by using the client's IP address as the source address and sending one packet at a time to a random port on the internet. Since the server is unable to distinguish between attacker and client packets, the chain will be disrupted, and access to the legitimate user will be denied [4].

Network Address Translators (NATs): If Network Address Translation is used to pass the traffic from client to server, the public IP address is shared between the private network of computers and every system on the local network occur on the internet with the public IP address. This is a problem since port knocking is necessary to open the requested port for the IP address. The server receives the client's public IP address. If the port opened to the public IP address means the port will be opened to all clients on the network sharing the same public address[4].

Authentication-Connection Association: It is necessary to point out that after a port is opened, there is no conceptual association between the authentication sequence and the client who is trying to communicate. Thus, a port that successfully opened will then be targeted by an intruder [3].

Failure of Daemon: A well-tested reliable code must be used as the server implementation allows for automatically changing firewall rules. If a daemon fails or does not work

correctly, the system could be unreachable or can be compromised easily.

These downsides of traditional PK are addressed by extending it to Single Packet Authorization (SPA).

B. Single Packet Authorization

SPA is a variation of standard PK in which the knock is referred to as an Approved Packet (AP), which is encoded within a single packet [3]. This mechanism comprises both Authentication and Authorization. A single packet authenticates the user to the server for simple remote administration; this technique not only addresses the challenge of out of order delivery but also by encoding the information into a single packet, it simplifies the operation.

SPA is a protocol was introduced as a next-generation passive authentication technology. SPA and port knocking consist similar architecture, but different processes, and SPA eliminates the limitations of port knocking such as out-of-order delivery, replay attack, and spoofing attacks. The first Single packet authorization available was called fwknop in 2005. It was created as the first port to knock by combining OS fingerprinting and port knocking. It provides a combination of authentication and authorization services.

In this approach, the client sends a SPA packet. When the packet is received by the server, it validates the credentials of the encoded packet and opens the door for connection. The encoded packet can be the combination of timestamp, client IP address and password.

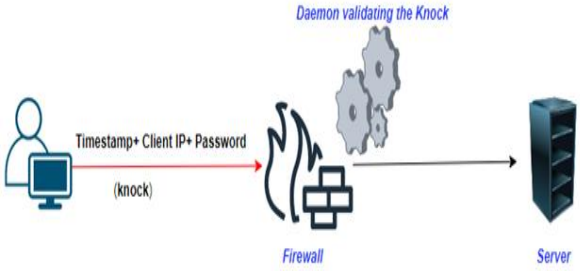

Figure 1: SPA Mechanism

When an attacker attempts to exploit a vulnerability in server software, the first step is to locate the target. With the help of tools like Nmap, it is simple to build a list of targeted systems that may be ready to exploit. However, SPA uses a default-drop packet filter stance that provides services only to the IP addresses that can prove their identities by a passive process. To authenticate remote IP addresses through this passive means, no TCP/IP stack access is required. Nmap does not even able to identify that server is running if it is protected in this process, and even though the attacker has a zero-day exploit, it does not matter [6].

SPA commonly used for SSH access to servers where it limits potential attacks by hiding the fact that SSH service even exists on the protected system. The SPA process occurs before the TLS connection, which helps to reduce attacks like DDoS (Distributed Denial of service) targeted at the TLS ports.

There is a common architecture for single packet authorization and port knocking, but different distribution mechanisms. In SPA, the data is encoded in a single packet, typically UDP or ICMP. This ensures that instead of being able to transmit two bytes of data per packet, the SPA can send up to 1500 bytes on an ethernet network between client and the server in one packet. This is a huge advantage over traditional port knocking, which leads to eliminating out-of- order packet delivery problem. The encoded information in these packets is timestamp, client IP address and, password combination which helps to protect from Replay attack [6]. On another hand if authorizing packet is intercepted, the attacker may try to use brute force to decrypt it and get the secret keys.

Single Packet Authorization resolves most of the limitations of traditional port knocking, but one problem remains open in SPA that is a compromise of keys by an external intruder or insider leads to the single point of failure. If the authorization key appears in the attacker's hands the SPA security layer will be defeated. This action cannot be suspected because the authorization is carried out based on the client secret key which leads to the expose of the database. This issue along with the issue of the potential decryption of the intercepted key can be addressed with the help of deception mechanisms.

C. Deception

Deception is a planned action performed to deceive the attackers and after making them perform specific actions that could help computer security defense [5].

Deception techniques will be used in our life all around the time. For example, in sports, deception techniques are used to deceive the other team making them believe that they are following a particular strategy in the game. As well, in cybersecurity deception and decoy-based techniques have been used in security for so long time in technologies such as honeypots and honey-tokens [7].

In the information security community, deception-based approaches are rapidly gaining attention. The main goals of this security control to achieve:

- lead the attackers away from the truth(real)
- Apply to the data collected by the attacker with risk and uncertainty.
- Add deception mechanism decoys to our security systems to detect data leakage and intrusion.

Lead the attackers away from the actual resource:

If attackers penetrate the system and effectively bypass the identification and deterioration methods, the protection system should not only be able to obscure our data, but also lead the attackers away by manipulating them into false data. Furthermore, it is also an effective defensive approach to frustrate the intruder by placing false keys using endless files. Such files appear small in organization servers but, when the attacker started downloading the files it will exhaust the bandwidth of the attacker and raise alarms [7].

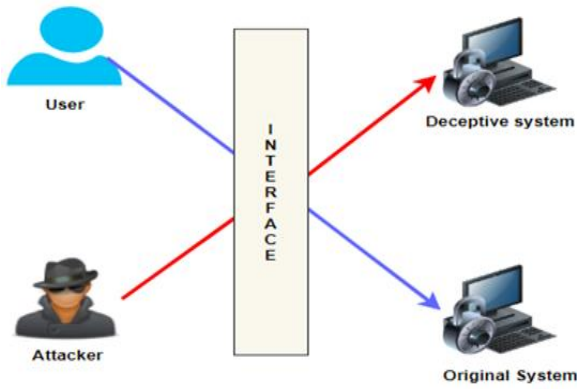

Figure 2: Deceptive Interface

Apply to the data collected by the attacker with risk and uncertainty:

Even the attacker obtained the sensitive information from the organization when the false information is injected into it can lead the attacker into a confusing state. The introduction of false information can debase or devalue the correct data obtained by the attacker.

Add deception mechanism decoys to the security systems to detect data leakage and intrusion:

Mechanisms based on deceit are an efficient means of enticing attackers to reveal themselves. To detect the accessing items and to do odd activities other IDS tools available for identification, but the advantage of deception tools is that between a typical user operation and an irregular one there is a straightforward idea. This disparity significantly enhances the efficiency of deception-based security measures and decreases the number of false positives and the scale of the log files [7].

1) *Deception Techniques*

There are still three simple stages of deception. Dissimulation, obfuscation, and emulation i.e., showing the false. Dissimulation can be accomplished in three ways. Masking, repackaging, and dazzling are some of the techniques used. Masking helps to deceive the attacker by hiding the real. However, if the masking is done completely, it appears as non-existing and can be a challenging task. This can be achieved by repackaging. In repackaging, the truth is hidden by making it look like something else. If this also turns to be difficult, the dazzling technique is used, which is used to confuse the targeted objects with other objects making it difficult to distinguish the truth from the deceit.

When it comes to simulating, the truth can be presented as false. When it is difficult to achieve, the second option needs to be considered. Finally, the last technique decoy, the target of this technique is to divert the attacker's attention from the most important components to the less important components [8].

2) *Deception Mechanisms*

Deception mechanisms are used in various information security applications. Taxonomy of such mechanisms includes honeypots, fake keys, fake accounts, endless files, DNS redirections, anti-forensics, and honey tokens in different forms. To deceive the attacker, decoy passwords are inserted into the database. The authentication system employs an additional hardened server known as a honey checker to determine whether the matching password is genuine or a forgery. If a decoy password is discovered, the honey checker raises an alarm and directs a potential attacker to the honeypot. The use of decoy passwords has been also proposed as part of the SAuth authentication scheme [7].

a) *Honeypot:* A Honeypot is a specially designed piece of software that mimics another system, normally with vulnerable services to attract the attention of an attacker who may sneak through the network.

b) *Decoy Mechanisms*: To see if anyone is attempting to log into them, decoy accounts are built. When an attempt is made, security experts will look at the attackers' techniques and tactics without being identified or compromising any data[9].

c) *Honeywords:* Honeywords is a password-protection service. False passwords are stored in the authentication server's password file to confuse attackers. Honeywords tend to be standard passwords chosen by the user. As a result, an attacker stealing a honeyword password file would have a hard time telling the difference between honeywords and true user passwords.

A real user password, for example, is one of the following. The rest are a honey phrase (generated using a simple "chaffing-with-a-password" algorithm in this project) which becomes difficult to identify the right password [10].

- KW3rcJ2rdIEgqBx0zEsq

- kLwWXwfVulITzL0ROykE
- dcRohVMZ95Bt5hsUfrDX
- GiHEpOb0d2qzQ9moKvRZ
- Lba5QWKYyBO4G7xa3hZz

d) *A mechanism for honeywords generation:* The honeywords are formed by tweaking the characters in the specific positions: each character in a specific position is replaced by a randomly chosen character of the same type, for example, digits are replaced by digits, letters are replaced by letters, and special characters (anything other than a letter of digits and numbers) are replaced by special characters. With the aid of this generation of Decoy passwords, two advantages can be achieved [10].

- Firstly, the intruder is in a frustrating situation because he must pick one of the passwords.
- The second function is that if the incorrect index is sent to the honey checker, it will detect and alert the password file that has been broken.

3) *Applications of Deception mechanisms*

In computer security, the use of some forms of deception (to defend) against cyber-attacks is a common practice [11]. For instance, Honeypot is a trap designed to entice the attackers by masquerading as an actual user database resource. Encryption is another security practice that attempts to deceive by concealing the true information by replacing it with a random confusing set of strings. Applications of these mechanisms include:

- Traditional cyber defence technologies, such as firewalls and endpoint security systems, detect decoy users on networks by employing deception techniques.
- Emulation-based traps (decoys) can also imitate medical devices, automated teller machines (ATMs), retail point-of-sale systems, switches, and routers.
- SDN Controller applications for secure network protection and IDS so on
- Robotics

4) Introduction to WebSPA

Through submitting a single HTTP/S request to the web server, the OWASP WebSPA Project authorizes the execution of an Operating System (OS) premeditated order. It includes a cryptographically secure open mechanism on the web application layer, like well-known port-knocking methodologies [12].

By providing a jar file that "tails" an existing web server's access log, this project introduces the concept of web knocking. A user submits a specially designed URL, which causes a predefined O/S order to be executed. There will be no new ports or infrastructure built. Similarly, to traditional network port-knocking schemes, the goal of the OWASP WebSPA Project is to create a secret contact channel for Operating System (O/S) commands over the web application layer. This channel is not bidirectional in any way: only the client may send commands to the server. In the current version, the reverse, i.e., the server which issue commands to the client, is not an alternative [12].

If port knocking is "a type of host-to-host communication in which information flows through closed ports," network knocking is "a type of host-to-host communication in which information flows through erroneous URLs." Finally, to replicate the features of Single Packet Authorization (SPA), a user's entire action is sent with a single GET request [12].

Problems with OWASP WebSPA Tool

- Storing of plain keys into system instead of cryptographically keys will make attacker prone.
- Key Leakage – No mechanism for hiding the transport layer.
- No mechanism of IDS/IPS detection and log file pollution.
- IP bases Authentication and weak Cryptography.
- No password rotations v/s Anonymity

Solutions and Improvements to overcome these problems.

- Introducing private-public key cryptography for keys in the system
- Adding fake entries to the database with keys.
- Sending decoy knocks with original knock and re-ordering it.
- Storing decoy knocks into database to mitigate such attacks as MIM, Brute force, etc.
- Introducing an intermediate server (Honey Checker) to detect IDS related problems.

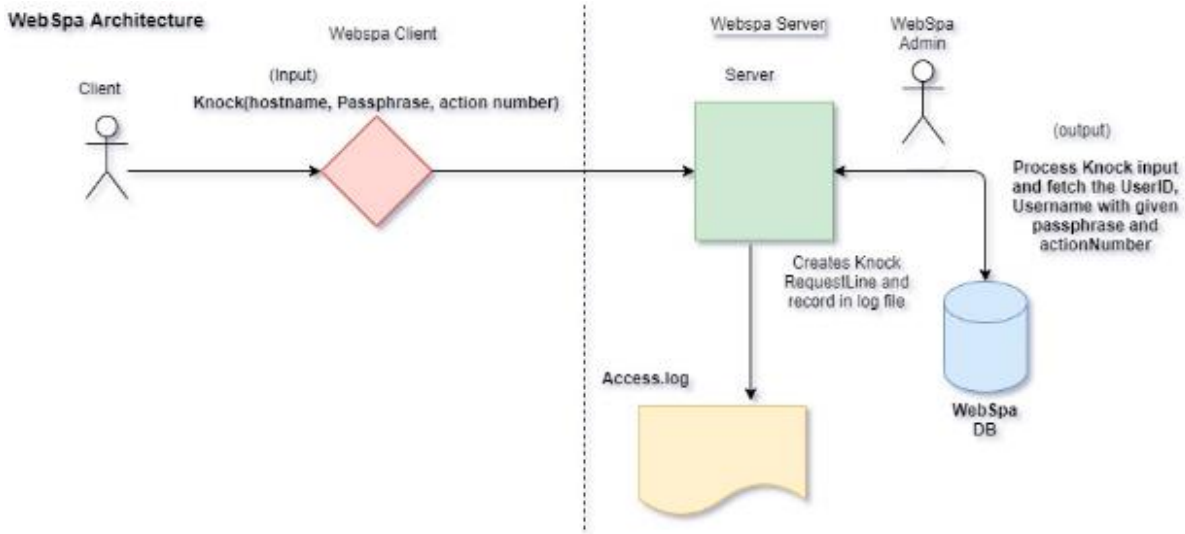

Figure 3: OWASP WebSPA Architecture[12]

III. ADDING DECEPTION TO WEBSPA

Research - Problem Identification

- Literature review and research objectives
- Problem identification and statement

Methodological Approach

- WebSPA system code base setup locally
- Configuring WebSPA client, server, and database
- Adding new feature into WebSPA system
- Development and Optimization
- Testing of newly added features
- Performing experiments to capture the metrics
- Data collection and outcomes

New Features development

- Private and public key implementation to WebSPA pass-phrase transitions
- Sending multiple decoy knocks with original knock request and saving in new table
- Introducing new table(DECOY_USERS) in WebSPA schema
- Storing of multiple decoy knocks in database table
- Re-ordering the decoy knock request to server

Actions

Step 1 - Perform the complete analysis and code review WebSPA codebase.
Step 2 - Complete triaging on security-related issues segregation of port knocking and implementation of deception as an additional security layer.
Step-3: Adding improvements to WebSPA code base.

A. WebSPA code to be modified to generate multiple knocks (actual knock + decoy knocks) for WebSPA logins.
B. A new feature would be added to recognize the fake knocks and the actual knock on the server-side.
C. The code would be modified to include a Public and Private key encryption mechanism for implementing the deception technique.

Step 4 - Perform regression testing and unit testing for developed features in the WebSPA system.
Step-5: The functionality is verified against various other metrics like response time and number of knocks.

Outcomes

Step 1 - The complete understanding workflow of WebSPA code base and how HTTP knocking works.
Step 2 - Understanding drawbacks in WebSPA system and network protection techniques.
Step 3 - Able to generate multiple knocks with a single request.

A. Implement asymmetric cryptography to WebSPA user passphrases.
B. Able to recognize fake knocks and detect Man-in-the-Middle attack.
C. Re-ordering of sending decoy knocks.

Step 4 - Regression testing and performance measurements time (ms).
Step 5- Experiments on extra security mechanisms introduced into WebSPA.

A. *Updates on WebSPA*

Although SPA eliminates most of the difficulties associated with classic port knocking, one issue remains unsolved: the compromise of keys by an intruder. The Single Packet Authorization security layer is bypassed if the authorization key is in the hands of an attacker. To deceive and to detect the attempts made by the intruder, this experiment has introduced a deception mechanism in the application.

Deception on the stored keys can be achieved by introducing decoy keys (fake keys) into the system. These words resemble the actual keys which confuses the potential attacker. Fake keys can be distinguished from the actual passwords using Fake password and Password table respectively. During transmission of these keys, deception can be implemented by sending a burst of password knocks initially in which the actual and the fake passwords knock is embedded. The fake and the actual passwords are distinguished at the server level and are stored in Fake password table and Password table, respectively. If this transmission of password burst is eavesdropped by an attacker, it was noticed to achieve deception by identifying the knock in these tables.

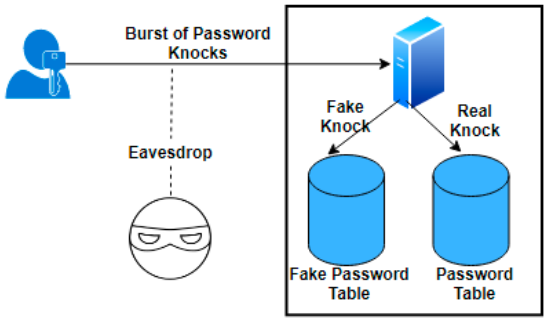

Figure 4: Deception during Key transmission

When the attacker tries to replay the knock, the server verifies the knock and decodes the password. If the password matches with the entries of the Fake Password table, the actual system falls in a failed closed state and the system warns of a Man-in-the-Middle attack or else server grant the access as indicated in the below figure.

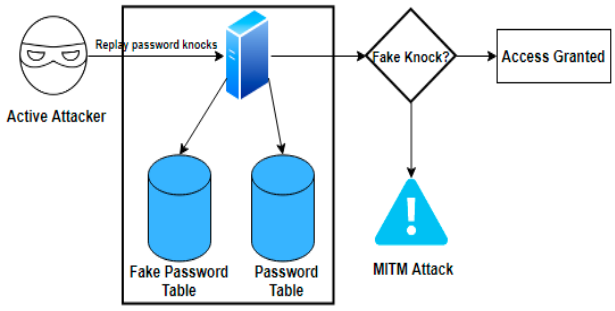

Figure 5: Detecting Man-in-the-Middle Attack

IV. EXPERIMENTAL RESULTS

The aim of this experiment is to ascertain that there is no huge negative impact of the extra security mechanism added on the usability of the application. To analyze the performance of the system, an experiment is conducted where the passphrase knocks are re-ordered, and the time of the transmission is measured for various number of knocks with and without asymmetric encryption.

Experiment was done using the following steps:

i. Assuming 5 re-order decoy knocks with every single knock request sent to server
ii. Use one correct passphrase (which exists in WebSPA database) and measure time between the client sending the request and the server doing some action on the server side.
iii. Repeat this 30 times to accumulate stats.
iv. Use one incorrect passphrase (which doesn't exist in WebSPA database) and measure time between the client sending the request and the server doing some action on the server side.
v. Repeat this 30 times to accumulate some statistics.

repeat steps 1-5 using 10, 50, 100, 250, 500, 1000 passphrases.

Average extra auth time WebSPA users	Without Asymmetric Encryption	With Asymmetric Encryption
10 passphrases	4.75861	4.448276
50 passphrases	4.103448	3.275862
100 passphrases	3.724168	4
250 passphrases	3.724138	3.551724
500 passphrases	3.551724	3.517241
1000 passphrases	4.655172	3.103448

Table 4. 1 Extra auth time with and without asymmetric encryption

As observed, the response time is in 3-5ms range which is acceptable for any interactive application, there is no

significant change and, the added security mechanism does not hugely impact the system performance. Asymmetric encryption was implemented to address the inherent problem of having to share the key in symmetric encryption models by utilizing a pair of public-private keys to eliminate the requirement to exchange the key. Compared to symmetric encryption, asymmetric encryption takes more time in general.

In our experiments, it has been noticed that there is a slight decrease (Approx. of 0.30 milliseconds) in total roundtrip duration for Authentication of users in WebSPA and it is because of various factors in implementation of Algorithms-specific metrics such as.

- Key pairs are not generated every time for every user, however keys are generated at initial Startup of server and stored into local disk like static key pairs, and which will be cached every time of authentication performed.
- Usage of shorter key length, say key size is 1024 bits which is small as such, 128-byte key length.
- In Asymmetric encryption RSA 1024 bits is quite fast, but as the length of the key becomes longer, the time required to operate on private keys rises rapidly.
- This was implemented in Java7 programming language, which provides powerful built-in methods under package javax.crypto. They were initialized when class loads perform quicker operation with encypt() and decrypt() [13].
- One of the major differences is that system do not store passphrases in plain text, rather it saves them in binary format which will also make a huge performance improvements and time complexity by reducing the memory overhead on JVM.

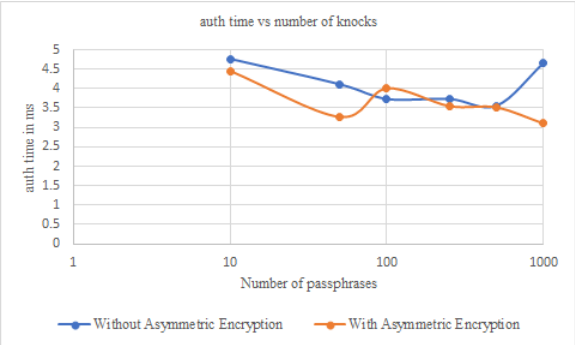

Figure 6: Graph of auth time v/s number of knocks

V. DISCUSSION

The updated WebSPA system is expected to handle the following attack scenarios of Man-in-the-Middle and Stolen password DB

A. Man-in-the-middle attack:

A man-in-the-middle attack is a sort of cyber attack in which an unauthorized third party enters an online communication between two hosts and stays undetected by the two parties. The information that was just realized by the two users can be monitored and/or changed in this scenario.

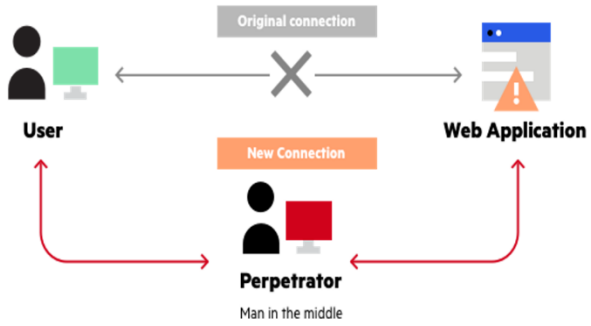

Figure 7: Man-in-the-Middle-attack [14]

Assume a case of standard Port knocking, where an attacker would be able to steal the secret knock details by eavesdropping on communication between client and the server.

However, in the implementation of this deception technique, to deceive the attacker decoy keywords that are randomly created and are sent together with the original password in random order. These knocks are initially stored in different databases after they are received by the server.

Having eavesdropped on the earlier communication, attacker would be unsure of the actual password among those burst of password knocks. If the attacker tries to replay the knocks, the improved system will check the knocks with existing fake knocks in the database and then alerts the IT security about a potential MIM attack. Thus, this additional mechanism provides foresight of an attack.

B. Stolen database of passwords.

Database contains confidential and sensitive information, so it is the favourite target for the attackers. The data theft gives attackers access to sensitive information, allowing them to read and change the liable information. The attackers are breaking into database using simple attack methods such as exploiting weak passwords and SQL injections and take advantage of unpatched vulnerabilities.

Additional security mechanism to protect the database:

Even though system uses different tables for Actual and the fake passwords, there is chance that database will be hacked. This is a serious issue because an attacker can gain the access to the password table and compromise the existing functionality. However, according to the modified WebSPA, as the keys/knocks are encrypted using asymmetric encryption, it would be difficult to exploit the confidentiality of the data and hence the updated system provides protection against the Stolen database of passwords. These updates which are used to achieve deception during transmission allow us to use asymmetric encryption.

The code, updates and the documentation are stored in Github:
https://github.com/sbitla1/ExtendingPortKnockingRM3_eTrunk

VI. CONCLUSION

This paper discussed two ways of adding deception mechanisms to Single Packet Authorization. Although SPA extends the port knocking by verifying the encrypted packet(single knock), it is still vulnerable to key leaks. If the key is leaked, it effectively leads to compromise of entire protection. Security of the WebSPA HTTP port-knocking is improved by (1) generating decoy knocks (fake knocks) for the original secret key: and by (2) implementation of the asymmetric cryptography to WebSPA user's passphrases. It would be difficult for the attacker to find the original key from the multiple encrypted fake keys. The multiple decoys knock along with original key, were saved in the two different tables, password table and fake password table. When the attacker tries to replay the knock, the server verifies and decodes the key, and if it matches the fake password table entries, the actual table closes, and the server warns the man in the middle attack. An experiment was conducted to test the performance and delay of the system and it is observed that there was no noticeable increase of time required for authentication. Therefore, one can conclude that the additional security mechanism has no negative impact on system performance. Further, the system can be enhanced by including additional features like a honeychecker and honeypot. However, this security enhancement may have a mild impact on usability, including difficulty in configuring and using it. This research demonstrates that the additional security feature helps the network team to detect various snooping attacks on the system without affecting its performance.

VII. REFERENCES

[1] R. McMillan, D. Volz, D. Hobbs and Tawnell, "Beyond Colonial Pipeline, Ransomware Cyberattacks Are a Growing Threat," 2021.

[2] M.Krzywinski, "Port Knocking," Linux Journal, 16 june 2003. [Online]. Available: https://www.linuxjournal.com/article/6811. [Accessed 01 2021].

[3] R. deGraaf, J. Aycock and M. Jacobson, "Improved Port Knocking with Strong Authentication," *21st Annual Computer Security Applications Conference (ACSAC'05),* , p. 11, 2005.

[4] S. Jeanquier, "An Analysis of Port Knocking and Single Packet Authorization Msc Thesis," p. 76, 2006.

[5] Z.A.Khan, N.Javaid, M. Arshad, A.Bibi and B.Qasim, "Performance Evaluation of Widely Used Portknoking Algorithms," *2012 IEEE 14th International Conference on High Performance Computing and Communication & 2012 IEEE 9th International Conference on Embedded Software and Systems,* p. 5, 2012.

[6] M. Rash, "Single Packet Authorization," Linux Journal, 01 April 2007. [Online]. Available: https://www.linuxjournal.com/article/9565. [Accessed 01 2021].

[7] M. Almeshekah, E. H. Spafford and M. J. Atallah, "Improving Security Using Deception," p. 18, 2013.

[8] E. Spafford and M. Almeshekah, "Planning and Integrating Deception into Computer Security Defenses," *CERIAS,* p. 12, 2014.

[9] M. Pascucci, "How to Use Decoy Deception for Network Protection," algosec, 2010. [Online]. Available: https://www.algosec.com/blog/author/matthew-pascucci/. [Accessed 01 2021].

[10] J. Ari and R. L.Rivest, "Honeywords:Making Password-Cracking Detectable," *MIT,* vol. 2, p. 19, 2013.

[11] J. Yuill, D. Denning and F. Feer, "Using Deception to Hide Things from Hackers: Processes, Principles, and Techniques," *Journal of Information Warfare,* p. 14, 2006.

[12] OWSP, "OWASP webspa," OWASP, 2011. [Online]. Available: https://owasp.org/www-project-webspa/migrated_content. [Accessed 2021].

[13] JavaTeam, "API specification for the Java™ Platform," java platform standard Ed. 7, [Online]. Available: https://docs.oracle.com/javase/7/docs/api/javax/crypto/package-summary.html.

[14] Imperva, "Man in the middle (MITM) attack," Imperva, [Online]. Available: https://www.imperva.com/learn/application-security/man-in-the-middle-attack-mitm/. [Accessed 04 06 2021].

[15] A. Mallik, "MAN-IN-THE-MIDDLE-ATTACK: UNDERSTANDING IN SIMPLE," *Cyberspace: Jurnal Pendidikan Teknologi Informasi,* vol. 2, p. 26, 2018.