ISSN 2229-5518

A Result Analysis of Modified Hybrid Port Knocking (MHPK) with Strong Authentication

Ms. Priyanka Sahu, Ms. Megha Singh, Prof. Deepak kulhare

Abstract— It is sometimes desirable to allow access to open ports on a firewall only to authorized external users and present closed ports to all others. We examine ways to construct an authentication service to achieve this goal, and then examine one such method, Modified Hybrid port knocking. Implementations by presenting a novel port knocking architecture that provides strong authentication while addressing the weaknesses of existing port knocking systems. This paper are developing and evaluating the performance of a new proposed modified hybrid port knocking (MHPK) technique with proposed encryption/decryption technique. Prime concerned of the proposed work is to prevent different –different type of port attack and fulfill the entire security requirement for network. Proposed technique is the combination of four concepts, these are port knocking (PK), Symmetric key encryption/decryption, steganography and mutual authentication. In conclusion, port knocking deserves future consideration and can be a valuable layer in defense-in-depth. The performance evaluation and analysis of the proposed technique is calculating by measuring of the some parameter like security, portability, and average authentication time, which is showing the superiority of the proposed technique as compare existing technique.

Index Terms— Authentication, Attack's, Cryptography, Firewall, Port Knocking, Portability, Security, Steganography.

1 INTRODUCTION

The Internet can be seen as a huge network of nodes connected together to provide different services. The question that comes is how to access the servers in the network. This issue has been addressed by following approaches: [1]

- One can be by using a firewall, which can control thetraffic based on IP addresses.
- Another can be by using more customized devices like Intrusion Detection and Prevention Systems.

But still, even after putting such efforts towards securing the network, cases of network-attacks are often reported. In general, the first step of any attacker is information gathering, in which the attacker tries to find the complete details of the victim system or network like the services running, ports opened, version number of some particular software's etc. In the second step, attackers try to find out both the existing and zero-day vulnerabilities in the version of services, and may exploit those vulnerabilities to cause breach of confidentiality, integrity or availability issues. The first line of defense against any attack is system's firewall. The firewall is used to limit the resources of any network connections. A firewall works on predefined set of rules according to which it accepts or rejects any packet. These rules may be based upon the IP addresses or some other characteristics. Since very little information is revealed by the source address of a packet, the attacker may easily disguise the origin of the packet by modifying its contents or inserting its own packets, thereby making it bypass the firewall for potential use. [2] Port Knocking [3] is appropriate for users who require access to servers that are not publicly available. Port knocking refers to a method of communication between two computers, usually a client and a server, in which the information is encoded and possibly encrypted in to a sequence of port numbers. This sequence is termed as knock sequence. The server can keep all its ports closed but open it on demand if users have authenticated themselves by providing a specific knock sequence (a sort of password).

Initially all the ports on the server are closed for public communication and the server is monitoring all the attempts to connect to the services. The clients initiate a connection by sending SYN packets to somespecific ports on the server which are specified in the knock sequence. During the knocking process by the client, server offers no response and it just monitors the knocks silently on the specified ports. When the server detects a valid knock sequence, it triggers a server side process and opens the port for communication with the client. This is a form of the IP communication over closed ports. The definition of valid knock sequence and the server side process is completely user dependent and can result in modification of firewall rules and other administrative system events .This is an authentication mechanism, in which closed ports on the server system are knocked in a predefined sequence to provide services to the legitimate user after successful authentication. It provides an additional layer of security to the server by having additional advantage of stealth also. In other words, port knocking is a mechanism which is used to hide the services running on a hardened server which have users that require continual access to services and data from remote locations and that are not running any common public services. [3][4].

Some advantages of using the port knocking technique are as follow: [1]

- Almost impossible to determine whether port knockingis implemented on the server machine or not.
- Detection by sniffing is practically difficult.
- It is a firewall based method for user authentication for non-common services.
- Establishes connections to the hosts with no open ports by the subversive use of closed ports.
- Benefits from access control provided by IDS andfirewalls.

- *Ms. Priyanka Sahu, M. Tech Scholar , CSE Dept. , CIIT INDORE* ***sahupriyanka20@gmail.com***
- *Ms. Megha Singh, Assistant Professor, CSE Dept., CIIT INDORE* ***maggii.megha@gmail.com***
- *Prof. Deepak Kulhare, Associate Professor, CSE Dept., CIIT INDORE* ***dkulhare@gmail.com***

The simplest implementation of port knocking uses a log file to interface with the firewall software. This simple approach makes port knocking highly accessible for home users who would like to harden their NIX systems. One of the strong advantages of port knocking is that the protected services do not require any modification. Port knocking is easy to set up and presents no performance issues when dealing with a modest number of incoming connections. Cryptography has the ability to provide a number of services which aid us in protecting our information in various ways as it is sent across networks or stored on physical media. Confidentiality is a crucial element of network communications when private information is being stored or transmitted. The use of encryption has allowed us to protect such information and prevent it being disclosed to unauthorized parties. Similarly, data integrity ensures that our information is not modified in transit, and that we can trust that the information received is as intended. The slightly more contemporary field of public key cryptography has the added ability of providing non-repudiation whereby it can be proven that an individual did indeed send a particular piece of information. Cryptography has proven to be extremely important in authentication protocols, as it is necessary for certain pieces of information to be protected from unauthorized modification in order to result in successful authentication.

2 PROBLEMS IN EXISTING HYBRID PORT KNOCKING TECHNIQUE

The existing model of hybrid port knocking mechanism.

2.1 Hybrid port knocking (HPK)

The HPK technique consists of seven main steps. In what follows, is the description of the seven steps [5].

1. Traffic monitoring

PK server is installed behind the network firewall, monitoring and checking traffic arrived to firewall (gateway).

2. Traffic capturing and analyzing

The PK server captures only the traffic holding a payload (image) for further processing.

3. Image processing

In this step, the PK server extracts the payload (image) from the received packet. The payload is supposed to hide some information using Steganography that can be used to prove the knockers identity and request. If the payload, contains encrypted information, which is demand to encryption/decryption algorithm to access intended information.

4. Client authenticating

After the PK server makes sure that the payload was carrying an encrypted request, it needs to make sure that it is communicating with the correct client, so it takes a random number and encrypts it using the clients GnuPG public key and sends it as a payload to the client.

5. Server authentications

The client now receives the packet carrying the encrypted payload, extracts it and decrypts it using the servers GnuPG public key. Then the client sends the random number as a payload back to the PK server to ensure its identity.

6. Proving the identity of the client

The PK server is still in the monitoring/sniffing state and receives the reply from the client to its random number check. The server extracts the payload and checks if the received message holds the same number as the one randomly generated and sent to the client.

7. Port closing

Finally, in this step, after the task is completed, either the client informs the PK server to close the port, or the PK server decides to close the opened port.

There are certain problems encountered with the existing model of hybrid port knocking mechanism. An attacker can discover the actual sequence of packets in port knocking process and launch attacks in some of the following ways:

- A sequence replay attack in which a particular set of packets can be sent to the victim again and again.
- Getting the sequence number by sniffing the packets.
- Observing the pattern of knock sequences in promiscuous mode and running port scans.
- Detection and interpretation of simultaneous knock sequences.
- Load caused on the network and individual systems.
- Packets are delivered and received out of order due to network latency and other factors, thereby exposing the knock sequence in most of the cases.
- Single packet authentication can lead to the disclosure of complete data if the packet is captured by the attacker in between the client and the server machine.
- Knock sequences being influenced by the use of spoofed packets.
- Connection failure if client behind a Network Address Translators (NATs).
- Poor mapping between authentication and connection.
- Problem in authentication using cryptography.
- Data extraction from intruder packets.

3 PROPOSED SOLUTION

Main goal of proposed research is the design Port-knocking (PK) model and develop Port knocking system with high security. The proposed research will analyze all types of port attacks and will find those reasons which are the cause of big problem in the network. Proposed model will be the combination of two different techniques like port-knocking (PK), and Cryptography. Proposed concept will be highly secured and efficient so this proposed technique will known as “Modified Hybrid Port-Knocking (MHPK)” technique. We designed our own port knocking system using a cryptographically-secure challenge response authentication system that accounts for out-of order packet delivery and partially addresses the complications caused by NATs.

3.1 Proposed Architecture: The MHPK technique defines in consists of seven main steps. Here proposed MHPK technique consists of only four main steps which are define in figure 1.

- Packet Capturing and Packet De-multiplexing
- Authentication
- Confidentiality
- Port Closing

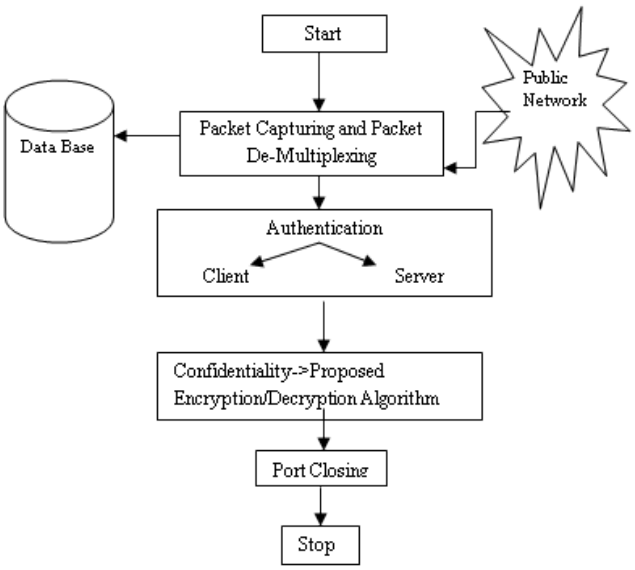


Figure 1: Block Diagram of Proposed Technique

3.2 Quality Attributes of Proposed Architecture:

- ***Adaptability and Availability:*** The proposedarchitecture will be efficiently implementable in hardware as well as on general purpose large, medium and small sized processors (for e.g. microprocessors, microcontrollers and smart cards respectively).
- ***Portability:*** The proposed software will be in Javaprogramming language (and will execute on any system with JVM compiler has been ported to). The portability of other software implementations will depend on portability of choice of programming language.
- ***Performance:*** To be able to implement proposedarchitecture special purpose hardware with the goal to decreasing memory requirements and execution time, special consideration should be given to possible (and alternative) simplifications of proposed architecture.

4 PROPOSED TECHNIQUE

The MHPK technique consists of nine main steps in what follows is the description of the nine steps.

1. Traffic monitoring

In this step, a PK server is installed behind the network firewall, as shown in fig. 2, monitoring and checking traffic arrived to firewall (gateway).

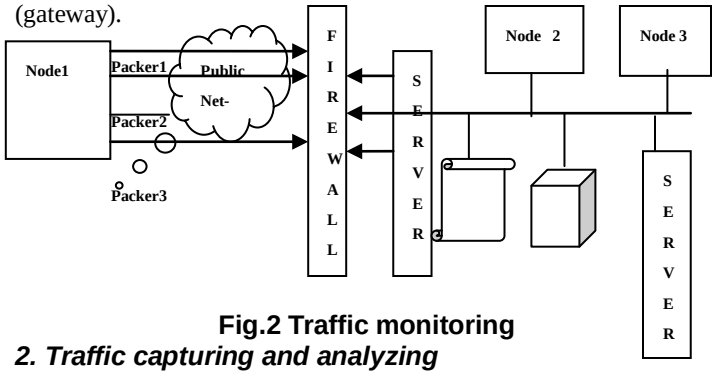


Fig.2 Traffic monitoring

2. Traffic capturing and analyzing

In this step, the PK server captures only the traffic holding a payload (image) for further processing, as shown in fig. 3. In this figure,

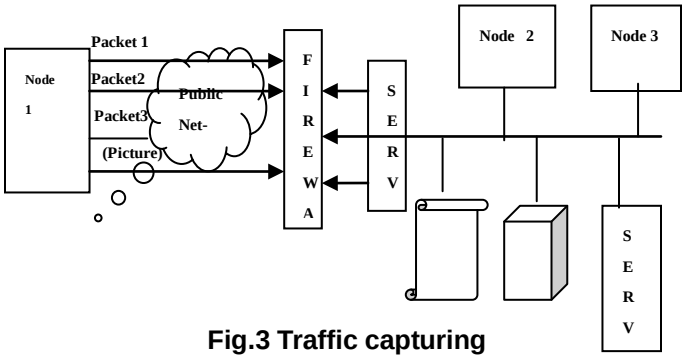


Fig.3 Traffic capturing

for example, only Traffic #3 is captured for further processing because it contains an image.

3. Image processes and Cryptography Functioning

In this step, the PK server extracts the payload (image) from the received packet. The payload is supposed to hide some information using Steganography that can be used to prove the knockers identity and request. If the payload, contains encrypted information, which is demand to encryption/decryption algorithm to access intended information, see figure (4).

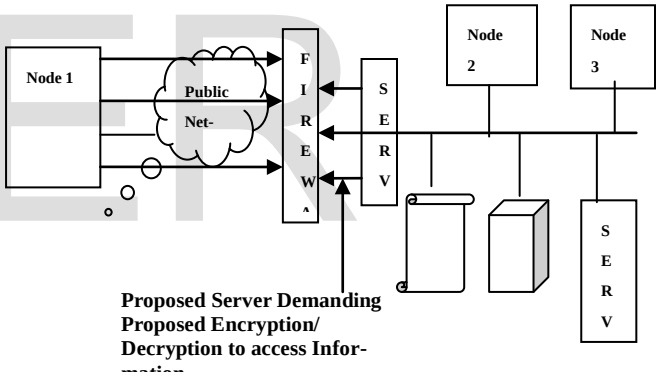


Fig.4 PK server demanding Encryption/Decryption Algorithm to Access Information

4. Client authenticating

After the PK server makes sure that the payload was carrying an encrypted request, it needs to make sure that it is communicating with the correct client, so it takes a random number and encrypts it using the clients GnuPG public key and sends it as a payload to the client.

5. Server authentications

The client now receives the packet carrying the encrypted payload, extracts it and decrypts it using the servers GnuPG public key. Then the client sends the random number as a payload back to the PK server to ensure its identity.

6. Proving the identity of the client

The PK server is still in the monitoring/sniffing state and receives the reply from the client to its random number check.The server extracts the payload and checks if the received message holds the same number as the one randomly generated and sent to the client.

7. Key Exchanging

After checking authenticity between client and server key exchange step will follow. In this step Client and Server exchanged symmetric key through GNUPG public key technique.

8. Proposed Encryption/Decryption

After key exchanging between client and server. Server called encryption/decryption process to access proper encrypted information from payload of the image. If the message is identified then the PK server executes the opening/closing of the requested port on the firewall, or executes the remote command based on the client's request. If the payload, contains intended information, which is either to demand the firewall to open/close a port for the client as shown in Fig. 5, or execute a command remotely on the appropriate server as shown in Fig. 6.Otherwise, if the result of the image processing fails to reveal valid authentication parameters, the PK server blocks the IP address of the source that sent the knocks and the payload (image). No port open/close or remote command execution is done in this step, only ensuring that the received payload holds a request which needs further authentication.

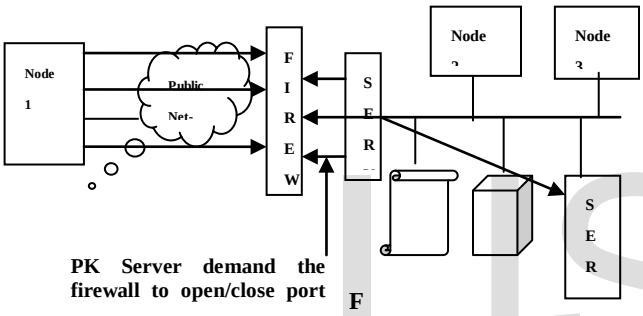


Fig.5 PK server demanding firewall to open/close port

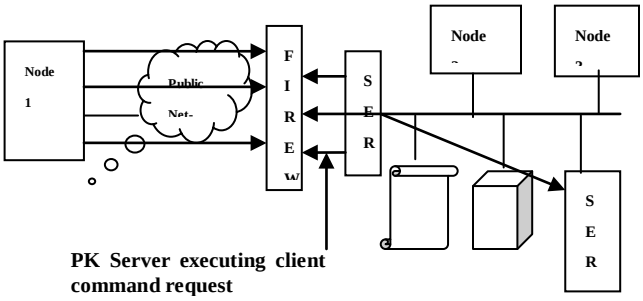


Fig.6 PK server executing clients command request

In any of these two cases, the PK server demands firewall to close the open port. In this case, if the client wants to access the system again, it needs to initiate new access or authentication request, i.e., start from phase #1.

9. Port closing

Finally, in this step, after the task is completed, either the client informs the PK server to close the port, or the PK server decides to close the opened port after specified silent period on that open port as shown in fig. 7.

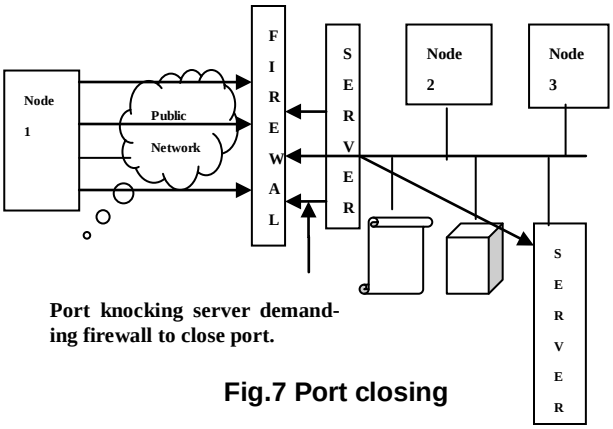


Fig.7 Port closing

5 PROPOSED ENCRYPTION/ DECRYPTION TECHNIQUE

Proposed encryption decryption technique is basedon the block cipher code (CBC) mode. Proposed encryption algorithm is using logical operations like XOR, shift (left or right circular) to mix key value and text value with each other. In this 128 bits long text can be encrypt or decrypt at a time with 128 bits key value. Initially 160 bits are passing as a input key value, then this input key values are dividing into two part, first part of the input key is 128 bits long and this will treat as an actual key and second part of the input key is 32 bits long. Now second part again will divide into two equal parts of 16 bits each. First 16 bits will be treat as an initialization vector one (IV-1) and second part will be treat as an initialization vector two (IV-2). figure 8 and figure 9 is showing architecture of the proposed encryption and decryption.

5.1 Architecture of proposed encryption

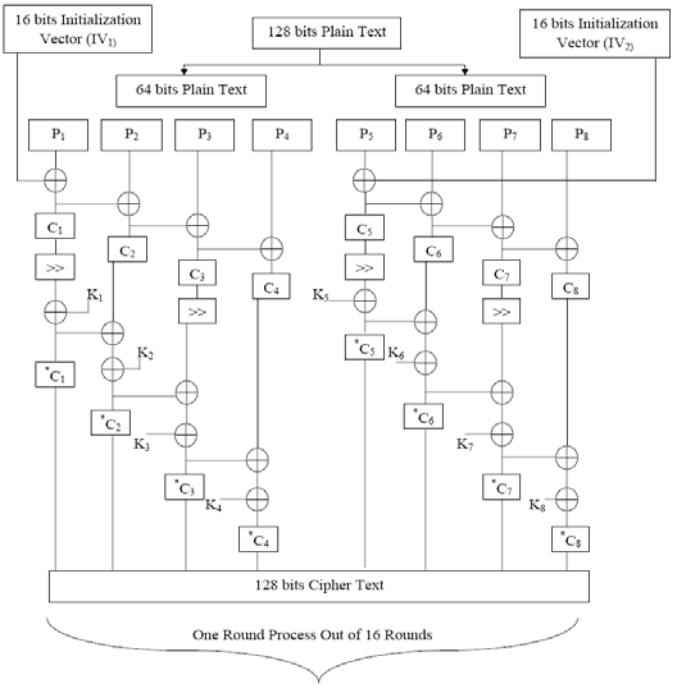


Fig. 8 Architecture of Proposed Encryption

5.1.1 Proposed Encryption Algorithm Step

i) Select plain text of 128 bits to be encrypted.
ii) Divide 128 bits plain text into two (Left, Right) blocks of 64 bits each.
iii) Now again divide these blocks (left and right) into 8 sub blocks of 16 bits each. 4 sub-blocks (P_1, P_2, P_3, P_4,) for left and 4 sub-blocks (P_5, P_6, P_7, P_8,) and for the right.
iv) Select 160 bits randomly as a key value in which first 32 bits will become initialization vectors (IV_1, IV_2) of 16 bits each. Remaining 128 bits key value will be divided into 8 sub-key values (K_1 to K_8) of 16 bits each.
v) Perform XOR operation between initialization vector (IV_1) and P1. Output of this will become C1.
vi) Perform XOR between C_1 and P_2. Output of this will become C_2.
vii) Apply 2 bits right circular shift on C_1 and perform XOR with K_1. Output of this will become $^*C_1$.
viii) Perform XOR between C_2 and $^*C_1$ and resultant will XOR with K_2. Output of this will become $^*C_2$.
ix) Perform XOR between C_2 and P_3. Output of this will become C_3.
x) Apply 2 bits right circular shift on C_3 and perform XOR with $^*C_2$. Resultant of this wills XOR with K_3. Output of this will become $^*C_3$.
xi) Perform XOR between C_3 and C_4. Output of this will become C_4.
xii) Perform XOR between C_4 and $^*C_3$. Resultant of this wills XOR with K_4. Output of this will become $^*C_4$.
xiii) Repeat process 5 to 12 for right sub block of the plain text with initialization vector (IV_2), K_5 to K_8 in place of K_1 to K_4 and P_5 to P_8 in place of P_1 to P_4.
xiv) Repeat Process 1 to 13 till 16 round.
xv) Now Combine all sub block into single cipher block.
xvi) Exit.

5.2 Architecture of proposed decryption:

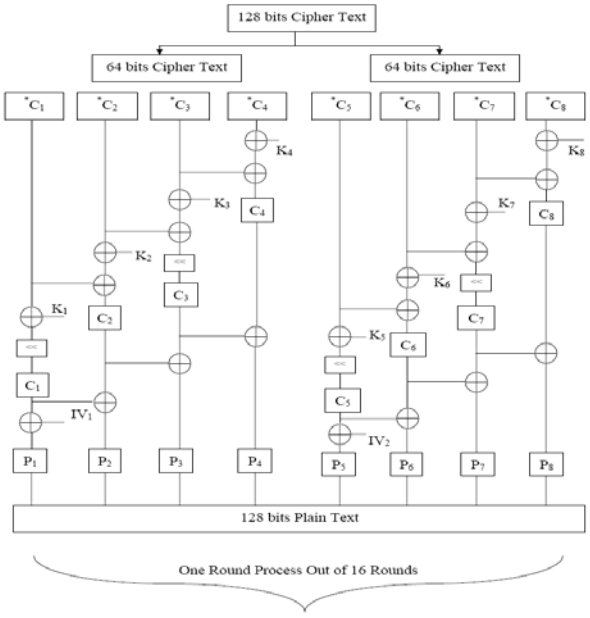


Fig. 9 Architecture of Proposed Decryption

i) Select Cipher text of 128 bits to be decrypted.
ii) Divide 128 bits Cipher text into two (Left, Right) blocks of 64 bits each.
iii) Now again divide these blocks (left and right) into 8 sub blocks of 16 bits each. 4 sub-blocks ($^*C_1$, $^*C_2$, $^*C_3$, $^*C_4$,) for left and 4 sub-blocks ($^*C_5$, $^*C_6$, $^*C_7$, $^*C_8$,) and for the right.
iv) Select 160 bits randomly as a key value in which first 32 bits will become initialization vectors (IV_1, IV_2) of 16 bits each. Remaining 128 bits key value will be divided into 8 sub-key values (K_1 to K_8) of 16 bits each.
v) Perform XOR between K_1 and $^*C_1$. And apply bits right circular shift in reverse out put this will become C_1.
vi) Perform XOR between C_1 and Initialization Vector (IV_2). Output of this will become P_1.
vii) Perform XOR between $^*C_2$ and K_2, resultant value will XOR with $^*C_1$. Output of this will become C_2.
viii) Perform XOR between C_2 and C_1. Output of this will become P_2.
ix) Perform XOR between $^*C_3$ and K_3, resultant value will XOR with $^*C_2$ and apply 2 bits right circular shift in reverse. Output of this will become C_3.
x) Perform XOR between C_3 and C_2. Output of this will become P_3.
xi) Perform XOR between $^*C_4$ and K_4, resultant value will XOR with $^*C_3$. Outputs of this will C_4.
xii) Perform XOR between C_4 and C_3. Output of this will become P_4.
xiii) Repeat process 5 to 12 for right sub block of the cipher text with initialization vector (IV_2), K_5 to K_8 in place of K_1 to K_4 and ($^*C_5$, $^*C_6$, $^*C_7$, $^*C_8$,) in place of ($^*C_1$, $^*C_2$, $^*C_3$, $^*C_4$,).
xiv) Repeat Process 1 to 13 till 16 round.
xv) Exit.

There is a big difference between existing and proposed port knocking technique is the encryption/decryption algorithm used by proposed technique to provide strong authorization. Proposed modified hybrid port knocking technique is just like existing port knocking technique but number of steps has shuffled up and down and added two more steps. These two steps are key exchanging and proposed encryption/decryption which is described above. Due to this encryption/decryption step proposed port knocking technique produced more good results.
The MHPK technique is found has a built-in detection capability that can be adjusted to countermeasure after a specific number of failure attempts. For brute force attack, proposed encryption and decryption will required 2^{128} time to crack actual key which is impossible

6 COMPARISONS OF MHPK AND HPK TECHNIQUE

Proposed encryption and decryption technique is creating differences between proposed modified hybrid port knocking (MHPK) system and other existing hybrid port knocking system. There are various parameters to evaluate existing system and proposed system but we have selected three parameters which are following:-

- Security
- Portability
- Symmetric Key Concept

Each parameters results shown in the below table -.

Security:

Existing System	Proposed System
Security	
Low	High

Table 1: Security Comparison

Portability:

Existing System	Proposed System
Portability	
No	Yes

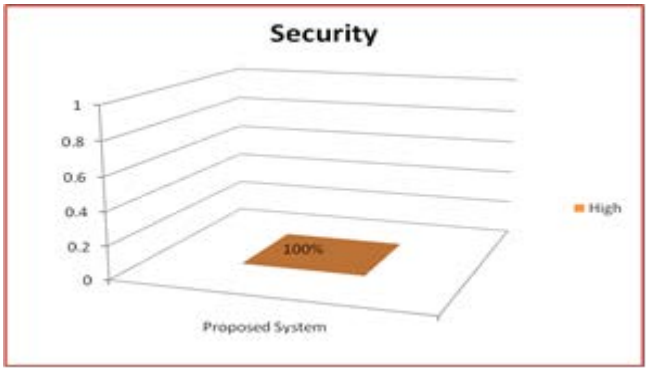
Table 2: Portability Comparison

Symmetric Key Concept:

Existing System	Proposed System
Symmetric Key	
No	Yes

Table 3: Key Comparison

From the table 1- 3 it is very clearly that proposed system is producing good results as compare existing system. Table 1 is showing the security concept between existing and proposed system. Security of the proposed system is very high as compare existing due to symmetric key cryptography concept used. Table 2 is showing the portability concept between proposed and existing system. Proposed system is portable because it is implementation language (JDK1.7). Finally table 3 is showing symmetric key cryptography concept. Existing system does not using symmetric key concept where proposed system is using this concept. In this I am using 128 bits key value which is also causes of high security.
Graph 1 is showing the security percentage of the proposed system. Which is 100% due to adding symmetric key value which 128 bits in length and this key value is not breakable?



Graph 1: Security evolution of the Proposed System

6.1 Cryptanalysis

Even if a symmetric cipher is currently unbreakable by exploiting structural weaknesses in its algorithm, it is possible to run through the entire space of keys in what is known as a brute force attack. Since longer symmetric keys require exponentially more work to brute force search, a sufficiently long symmetric key makes this line of attack impractical. With a key of length n bits, there are 2^n possible keys. This number grows very rapidly as n increases. Moore's law suggests that computing power doubles roughly every 18 to 24 months, but even this doubling effect leaves the larger symmetric key lengths currently considered acceptable well out of reach. Security level is the relative strength of an algorithm. An algorithm with a security level of x bits is stronger than one of y bits if x > y. If an algorithm has a security level of x bits, the relative effort it would take to "beat" the algorithm is of the same magnitude of breaking a secure x-bit symmetric key algorithm (without reduction or other attacks). The 128-bit security level is for sensitive information, and the 192, 256-bit level is for information of higher importance [2]. Here proposed algorithm having 128 bits key length so there are 2^{128} possible keys. The larger number of operation (2^{128}) required to try all possible 128-bit keys is widely considered to be out of reach for conventional digital computing techniques for the future.
There is a big difference between existing and proposed port knocking technique is the encryption/decryption algorithm used by proposed technique to provide strong authorization. Proposed modified hybrid port knocking technique is just like existing port knocking technique but number of steps has shuffled up and down and added two more steps. These two steps are key exchanging and proposed encryption/decryption which is described above. Due to this encryption/decryption step proposed port knocking technique produced more good results.

7 PERFORMANCE EVALUATIONS

To evaluate performance of the MHPK technique I have study analyzed several hacking scenario with existing port knocking techniques. The scenarios performed were the following:

- **U2R:** unauthorized access to local super-user (root) privileges, e.g., various ``buffer overflow'' attacks;
- **R2L:** unauthorized access from a remote machine, e.g. guessing password;
- **IP spoofing:** An attacker may fake their IP address so the receiver thinks it is sent from a location that it is not actually from.
- **DNS poisoning:** The attacker will send incorrect DNS information which can cause traffic to be diverted. The DNS information can be falsified since name servers do not verify the source of a DNS reply
- **Smurf :**An attack where a ping request is sent to a broadcast network address with the sending address spoofed so many ping replies will come back to the victim and overload the ability of the victim to process the replies
- **Ping broadcast:** A ping request packet is sent to a broadcast network address where there are many hosts. The source address is shown in the packet to be the IP address of the computer to be attacked
- **Man in the middle attack:** An attacker may watch a session open on a network. Once authentication is complete, they may attack the client computer to disable it, and use IP spoofing to claim to be the client who was just authenticated and steal the session.
- **Probing:** surveillance and other probing, e.g., port scanning.
- **Brute force attack:** brute force attack is a particular strategy used to break our lovingly crafted key. This is the most widely used method of cracking key and it involves running through all the possible permutations of keys until the correct key is found. For example, if our key is 2 characters long and consists of letters and numbers – and is case

sensitive, then a brute force attack would see a potential 3,844 different "guesses" at our key. This is because:
First character: lower case letters (26) + upper case letters (26) + numbers (10) = 62
Second character: same = 62
Total permutations = 62*62 = 3,844

8 RESULTS ANALYSIS

When investigating the existing HPK technique, it found that it is vulnerable to most of the attack, because the technique doesn't have any detection capability and is by default vulnerable to several attack. The existing technique is also found to be vulnerable to some attack, because the technique can only detect limited attack. But, the MHPK technique is found to be immune to above mentioned attack, because the technique has a built-in detection capability that can be adjusted to countermeasure after a specific number of failure attempts. For brute force attack, proposed encryption and decryption will required 2^{128} time to crack actual key which is impossible.

9 Conclusion

During my analysis of port knocking as a network security mechanism it has been becomes clearer how it must be viewed. Many of the criticisms about port knocking come from those who are looking for the be-all end-all of network authentication mechanisms no wonder they're disappointed. When looking at port knocking schemes it becomes clear that certain issues crop up time and time again. Overall conclusions of this proposed work are as follows:-

- The MHPK technique can be easily implemented.
- The MHPK technique is prevent different type of attack, because it uses cryptography to authorization and integrity, and Steganography for confidential and reduce packet capturing overhead, and mutual authentication to authenticate.
- The MHPK technique is much more secure than the traditional PK and the single packet authentication techniques, because solved problems that others failed in.
- The communication protocol used is a simple secure encryption scheme that uses GnuPG keys with Cryptography and Steganography constructions.

10 Further Research

The concealment aspect of port knocking is a feature which may be of interest to malware writers, especially when a newly opened port may be indicative that a host has been compromised and is running a 'listener' to allow the attacker to connect to the machine. Similarly, an open port may allow other attackers to discover the already compromised machine and claim it as their own. Worms spread automatically, sometimes by scanning the local network (or the Internet) and connecting to vulnerable services running on discovered machines in order to exploit them (exactly one threat that port knocking aims to protect against), and it is in the worm owner's interest to maintain control over his network of compromised computers. By implementing a form of port knocking into their worms, the authors can maintain control over machines that become compromised.

11 REFERENCES

[1]Vikas Srivastava, Alok Kumar Keshri, Abhishek Dutta Roy, Vijay Kumar Chaurasiya, Rahul Gupta,"Advanced Port Knocking Authentication Scheme with QRC using AES " [2011 IEEE].

[2] RenniedeGraaf, John Aycock and Michael Jacobson, Jr., Department of Computer Science, University of Calgary, Alberta, Canada, "Improved Port Knocking with Strong Authentication", 21st Annual Computer Security Applications Conference (ACSAC 2005), http://www.acsac.org/2005/papers/156.pdf

[3] Port Knocking http://www.portknocking.org/

[4] M. Krzywinski, "portknocking.org", "Port Knocking from the inside out", [Online]Available: http://www.portknocking.org/docs/portknocking_an_introduction.pdf, accessed Jan 2011, pp. 6-7.

[5] Dr. Hussein Al-Bahadili and Dr. Ali H. Hadi "Network Security Using Hybrid Port Knocking" IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.8, August 2010.

[6] Doyle M., "Implementing a Port Knocking System in C". Department of Physics, University of Arkansas, 2004.