Port-Knocking Method For Enhancing Network Security

Gulomov Sherzod Rajaboevich
Head of Information Security department, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan
sherhisor30@gmail.com

Mirzaeva Malika Bakhadirovna
Ph.D., of Hardware and Software of Management Systems in Telecommunication department, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan
malikamirzaeva01@gmail.com

Iminov Abdurasul Abdulatipovich
Ph.D., head of the Information Technologies department, Academy of Internal Affairs, Tashkent, Uzbekistan
iminovabdurasul1970@gmail.com

2022 International Conference on Information Science and Communications Technologies (ICISCT) | 978-1-6654-7229-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICISCT55600.2022.10146918

Abstract. **This paper considers the issues and the functionality mechanism of Port-knocking in computer networks. The scheme of a dynamic Port-knocking based on authentication and buffer information is proposed. Based on the proposed dynamic port knocking scheme, a flowchart of dynamic Port-knocking algorithm is proposed, which improves the detection of ports by using a sequence of knocks of various lengths.**

Keywords— Port-knocking, length pool, dynamic length, authentication decision making, dynamic length knock sequence, firewall, packet monitoring.

I. Introduction

Forty-five years after the birth of the Internet, it is well known that the Internet is a hostile place. Any host connected to the Internet must be protected from unauthorized intrusion and other attacks. Unfortunately, the only secure system is one that is completely inaccessible, but to be useful many hosts must service available to other hosts. While some services must be available to anyone from any location, others must be accessible only to a limited number of people or from a limited set of locations. The most obvious way to restrict access is to require users to authenticate before granting them access. Traditionally, this is up to the services themselves: before giving users access to anything important, they must first verify their identity using any of a variety of methods. While this is effective, it's not a perfect solution. Many network services are large and complex systems. It is not uncommon for some of these services to have flaws in the authentication mechanisms that can allow attackers to gain unauthorized access. In addition, some services are inherently insecure: users cannot authenticate themselves.

A common method of restricting sources of network connections is to use a firewall. A firewall works by selectively accepting or rejecting network packets based on their source addresses or other characteristics. Unfortunately, the source addresses in packets say little about the person who sent the packets; determined attackers are quite capable of hiding the source of the packets they send. Once a port on the firewall is open to a single host, any attacker can potentially (not) exploit that opening. Also, not all authorized users have predictable IP addresses, so granting them access through a firewall requires opening a firewall for much or all of the Internet.

II. Problem Statement

Port-knocking technology performs a sequence of attempts to connect to closed ports. Even though all ports are closed, you can track all connection attempts in the firewall log files. The server, most often, does not respond to these connections in any way, but it reads and processes them. But if a series of connections was previously designated by the user, then a certain action will be performed. As an example, connecting to an SSH service on port 22. Port-knocking allows you to perform more than just this action. the trigger allows you to perform other actions (say, turning off the power, rebooting the system, etc.). Port-knocking is a network defense mechanism that operates on the following principle: a network port is closed by default until it receives a predetermined sequence of data packets that “forces” the port to open [1]. For example, it can make the SSH port "invisible" to the outside world, and open only to those who know the desired sequence. Figure 1 shows the functionality mechanism Port-knocking.

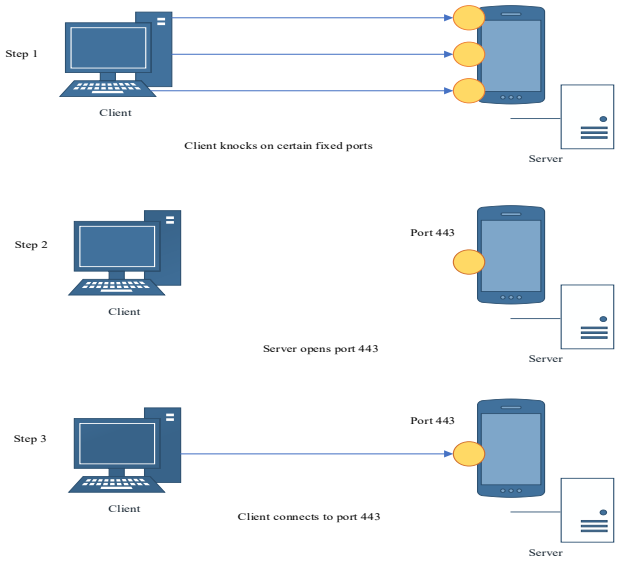

Fig.1. The functionality mechanism Port-knocking

III. The Scheme of a Dynamic Port-Knocking

Figure 2 shows the scheme of a dynamic Port-knocking where the client wants to connect to the SSHD server after being authenticated. The client starts the process as a port switch, sending a UDP packet to the server.

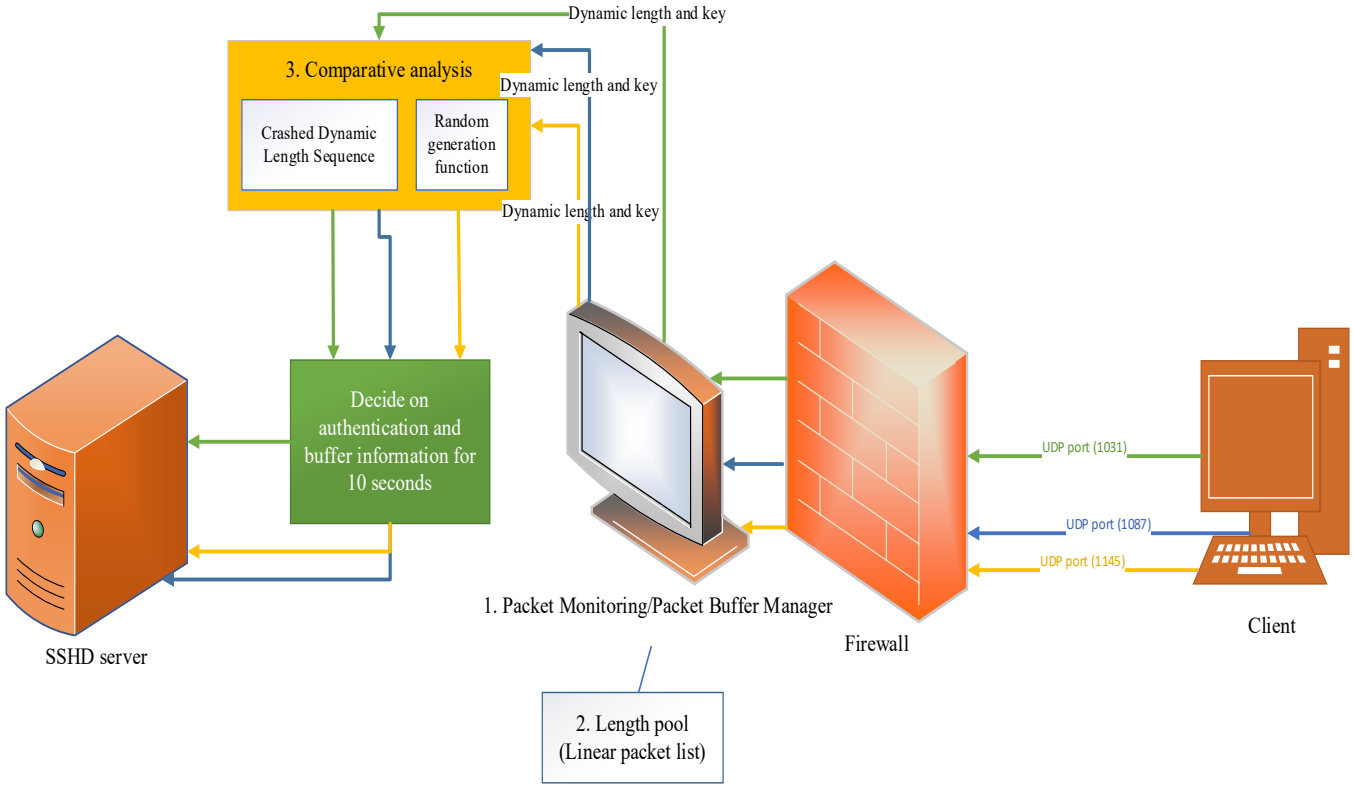

Fig.2.Scheme of dynamic port-knocking

A UDP packet contains an Ethernet header, an IP header, a UDP header, a data text passphrase, and an Ethernet trailer. This mechanism uses UDP because it does not require an ACK from the server. Without a response to the packet, the network is less vulnerable.

1. Packet Monitoring/Packet buffer manager

The Buffering Monitor/Manager module is responsible for monitoring the ports that are involved in the Port-Knocking process.

2. Length pool

The length pool (LP) is maintained in the form of a linear list that stores the various lengths that are allowed to use the Port-Knock [2-3]. The dynamic length (DL) is selected among the members of the LP.

A knocked-down sequence of dynamic length (SDL) is a two-dimensional array that stores a sequence of knocks according to the length of the sequence. The function of randomly generating knocking sequences is used by the Dynamic Length Knock Sequence (DLKS) method, which takes port ranges and DL as input and generates a knock sequence.

3. Comparative analysis

The packet monitor / packet buffer manager receives the DL from the first packet and buffers the received packets. When the authentication is completed, the received knock sequence and the selected DL are sent to the comparison stage. The comparison module selects a predetermined sequence of knocks in SDL based on the received DD and compares it with the received one. The comparison result is sent to the authentication decision module. In DLKS, the comparison module has a random generator function to generate a random knock sequence based on the received DL and a random key that are sent from the client side.

Authentication decision making

The true result of the comparison indicates a legitimate user, and hence the server sends an acknowledgment packet and the post-authentication phase begins to establish a secure connection between the port and the server. Otherwise, the buffers occupied by this Port-Knocking are flushed [4-5]. The DL structure shows how the dynamic concept is used in port-knocking to have different knock sequence lengths. It also talks about the impact of using DL on both security and performance.

IV. Flowchart of Dynamic Port-Knocking Algorithm

Figure 3 proposes a Flowchart of dynamic Port-knocking algorithm. The proposed algorithm improves basic port detection by using a sequence of knocks of varying length on each authentication attempt. This enhancement provides a high level of security and reduces the chance of a server being hacked, especially in the case of a large number of simultaneous authentications. Analysis shows that kicking out dynamic length ports solves both security and performance issues. The security of the base knock of the port is evaluated based on the probability of breaking the knock sequence with two parameters, which are the number of users simultaneously requesting authentication, A, and the length of the Port-knocking, I_{PK}. Thus, the way to calculate the probability of a hacking sequence is different from the basic port-knocking.

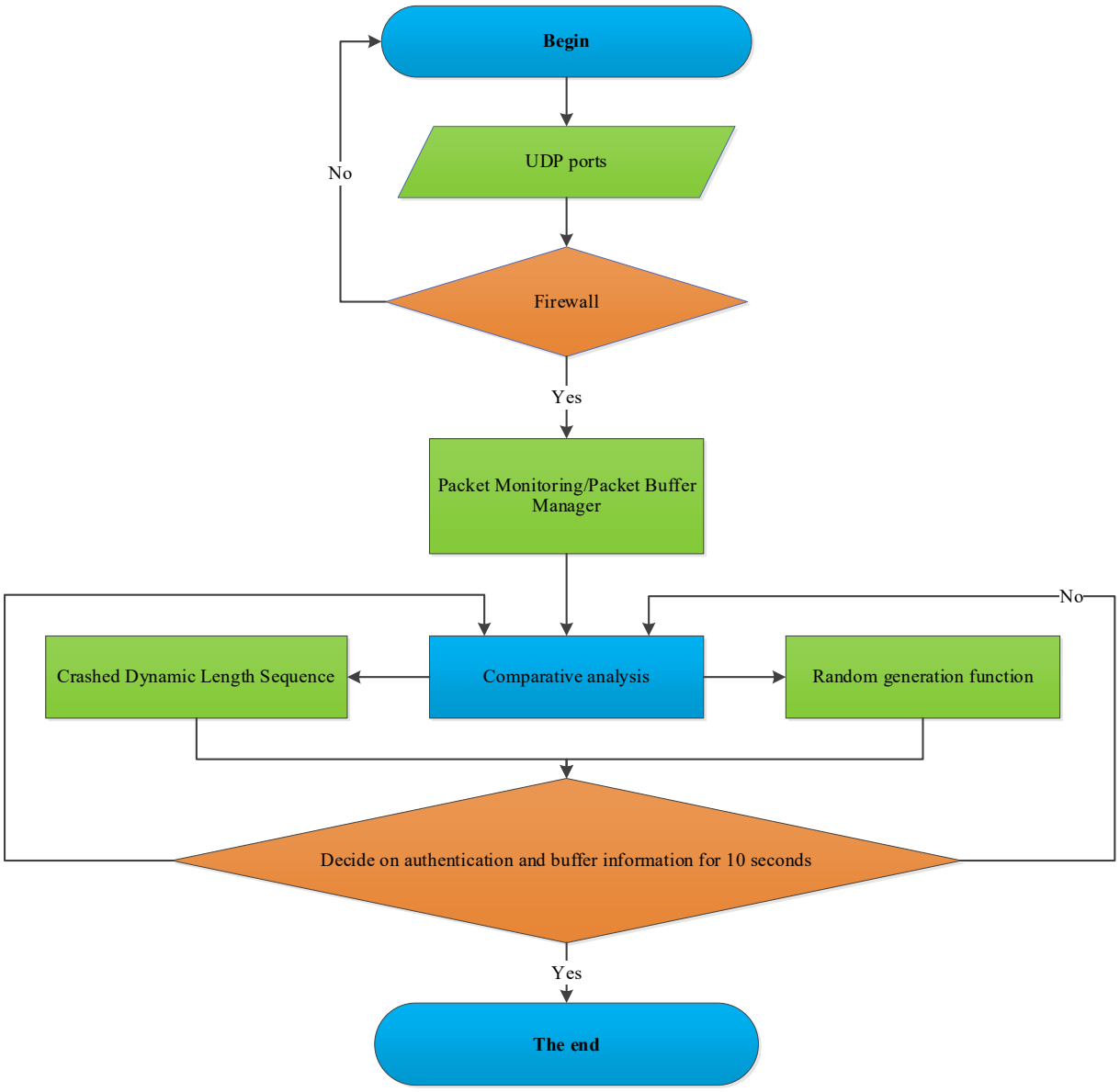

Fig.3. Flowchart of dynamic Port-knocking algorithm

The base port knocking protection is estimated based on the number of users using the same A sequence, the length of the I_{PK} knock sequence [6-7], and the probability of using the correct order of the $I_{PK}!$ knock sequence, as shown in the following (1) equation.

$$R(BL) = \frac{A \times 2^{(-1)\times I_{PK}}}{I_{PK}!} \quad (1)$$

According to equation (1), an increase in A leads to a higher probability of breaking the knock sequence. Meanwhile, an increase in I_{PK} leads to a decrease in probability, since it is generally more difficult for an attacker to get the correct sequence of ports [8-9]. A dynamic length port knock is expected to consider both the best and worst case based on the sequence knock length used. Thus, it is assumed that the number of users who applied for authentication with the knocking I_{PK_i} is equal to each other. Therefore, the sequence breaking probability of a dynamic length port knock is calculated by the following equation.

$$\begin{aligned} &R(BL)_{\text{dynamic length}} \\ &= (R(BL))_{I_{PK_1}} \times (R(BL))_{I_{PK_2}} \times \dots \\ &\times \left(R(BL)\right)_{I_{PK}\,\text{Number of hourly pool lengthsc}} \\ &\quad \text{Number of hourly pool lengths} \\ &= \prod_{i=1} (R(NB))_{I_{xb_i}} \end{aligned} \quad (2)$$

$R(BL)$ is calculated as follows.

$$R(BL) = \frac{Y \times 2^{(-1)\times I_{PK}}}{I_{PK}! \times \text{length pool}} \quad (3)$$

The following expression represents the dynamic length $R(BL)$ for 20 simultaneous authentications:

$$\begin{aligned} R(BL)_{\text{dynamic length}} &= \left(\frac{20 \times 2^2}{3! \times 3} \times \frac{20 \times 2^5}{5! \times 3} \times \frac{20 \times 2^7}{7! \times 3}\right) \\ &= \frac{20^3}{3^3} \times \left(\frac{2^2}{3!} \times \frac{2^5}{5!} \times \frac{2^7}{7!}\right) \end{aligned} \quad (4)$$

The probability of sequence cracking by dynamic length in this method is close to "0" when using the round function. And so, the performance of dynamic length port interrupt authentication is evaluated using the following two parameters:

1. Port Knock Authentication Time (AT) which starts timing when the port blocker sends an authentication request at the end of the authentication process.

2. Buffering Load (BL) shows the load that is placed on the cloud gateway when monitoring ports and creating dynamic address lists to buffer port breaker trigger packets. BL is estimated by the amount of memory that is used to buffer received knock packets during the authentication process [10]. The dynamic port knocking method introduces a LP that stores a sequence of knocks of various lengths. The LP is used in the port breaker script in the implementation section. After that, the BL is calculated based on the number of members in the LP.

Assume that the number of elements in the LP is m. Therefore, the "input" rules that are defined for dynamic port knocking are calculated by the following equation.

$$\text{Number of input rules} = DL_{PK_1} + DL_{PK_2} + \cdots + DL_{PK_m} = \sum_{i=1}^{M} DL_{PK_i} \qquad (5)$$

For each input rule, a dynamic list of addresses is created. Indeed, with a single authentication, the BL is equal to $I \times 32$ bits. Typically, for one dynamic port tapping attempt, the minimum space required is $I_{min} \times 32$ bits and the maximum space required is $I_{min} \times 32$ bits. When multiple users request authentication at the same time, the minimum and maximum occupied spaces are $A \times I_{min} \times 32$ and $\times I_{max} \times 32$ bits.

V. Conclusion

In conclusion, it should be noted that proposed Port-Knock method for enhancing network security is allowed to improve basic port discovery by using a sequence of knocks of varying length on each authentication attempt. This enhancement provides a high level of security and reduces the chance of a server being hacked, especially in the case of a large number of simultaneous authentications.

References

[1] 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.

[2] V. Srivastava, A. K. Keshri, A. D. Roy, V. K. Chaurasiya, and R. Gupta, "Advanced port knocking authentication scheme with QRC using AES," in 2011 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), pp. 159-163, IEEE, Apr. 2011.

[3] Gulomov Sherzod Rajaboevich, Nasrullaev Nurbek Baxtiyorovich, Toshev Sanjar Komilovich. A model for preventing malicious traffic in DNS servers using machine learning. 2021 International Conference on Information Science and Communications Technologies (ICISCT). 03-05 November 2021, Tashkent, Uzbekistan. DOI: 10.1109/ICISCT52966.2021.9670269

[4] H. Al-Bahadili and A. Hadi, "Network Security Using Hybrid Port Knocking," IJCSNS,vol. 10, no. 8, p. 8, 2010.

[5] Karimov Madjit Malikovich, Gulomov Sherzod Rajaboevich, Tashmatova Shaxnoza Sobirovna, Elmuradov Temurmalik. Differentiated Services Code Point (DSCP) Traffic Filtering Method to Prevent Attacks. 2021 International Conference on Information Science and Communications Technologies (ICISCT). 03-05 November 2021, Tashkent, Uzbekistan. DOI: 10.1109/ICISCT52966.2021.9670203

[6] Gulomov Sherzod Rajaboevich, Malikova Nodira Turgunovna, Qurbonova Kabira Erkinovna, Arzieva Jamila Tileubaevna. A method to improve the quality of service and overcome the loss of network packets. 2021 International Conference on Information Science and Communications Technologies (ICISCT). 03-05 November 2021, Tashkent, Uzbekistan. DOI: 10.1109/ICISCT52966.2021.9670120

[7] L. Boroumand, M. Shiraz, A. Gani, and R. H. Khokhar, "Impact of Port Knocking Authentication on Security and Performance: A Mobile Cloud Computing Perspective, in KSII Cloud Computing Symposium, The 5th International Conference on Internet (ICONI), 2013

[8] Kusuma, A. P. (2016). Implementasi Simple Port Knocking Pada Dynamic Routing (OSPF) Menggunakan Simulasi GNS3. Jurnal Manajemen Informatika Volume 5 Nomor 2, 7-17.

[9] Yusupov Sabirjan Yusupdjanovich, Gulomov Sherzod Rajaboevich. Improvement the schemes and models of detecting network traffic anomalies on computer systems. 2020 IEEE 14th International Conference on Application of Information and Communication Technologies (AICT), Tashkent, Uzbekistan, 2020. -4p. doi: 10.1109/AICT50176.2020.9368781

[10] M. M. Bakhadirovna, S. M. Azatovich and B.M.Ulug'bek O'tkir Ugli, "Study of Neural Networks in Telecommunication Systems," *2021 International Conference on Information Science and Communications Technologies (ICISCT)*, 2021, pp. 1-4, doi: 10.1109/ICISCT52966.2021.9670198.