

Enhancing the Security of Single-Packet Authentication Protocol Using Multi-Dimensional Factors

Xiaopeng Yang*, Ziyi Jia, Qiang Wang

National Key Laboratory of Advanced Communication Networks, The 54th Research Institute of China Electronics Technology Group Corporation, Shijiazhuang, China

* Corresponding author: Xiaopeng Yang

Abstract: This paper proposes a design method for a stealth gateway that supports multi-factor authentication, centered around software-defined perimeter (SDP) technology. The gateway dynamically authorizes user access requests initiated from authenticated devices and blocks such requests from unauthenticated devices. After legitimate users' access requests are authenticated through multiple factors including passwords, one-time passcodes, IP addresses, etc., they are forwarded by the stealth gateway. Compared to traditional Virtual Private Networks (VPNs), firewalls, and other security measures, the stealth gateway effectively reduces the attack surface of the network, achieves multi-factor authentication and perimeter invisibility, thereby enhancing overall data security.

Keywords: Stealth Gateway; Multi-factor Authentication; Software-defined Perimeter.

1. Introduction

Traditional network security devices, such as firewalls and Virtual Private Networks (VPNs), struggle to manage user permissions and enforce fine-grained access controls. These devices have a large exposure surface, which means that attackers can easily penetrate the network by scanning exposed ports, IP addresses, and other information, leading to the compromise of critical data assets.

Moreover, traditional network security devices are ineffective in addressing internal security threats; attackers might use viruses, phishing emails, or other tactics to lurk within the network, posing significant security challenges.

Therefore, designing a stealth gateway capable of perimeter invisibility, continuous identity verification, and dynamic access control is crucial for addressing these challenges.

In response to the limitations of current technologies, this paper presents a design method for a stealth gateway supporting multi-factor authentication. The proposed method aims to develop a stealth gateway with features such as perimeter invisibility [1], SSL encrypted tunnels [2], multi-factor authentication [3], and dynamic access control [4]. This approach addresses issues such as large exposure surfaces, lack of fine-grained access control, single authentication methods, and inadequate defense against escalating internal

and external threats. By making targets invisible within cyberspace, thus preventing attacks [5], the method ensures comprehensive protection of resources for enterprises or service providers.

The innovations of this paper are summarized as follows:

(1) Real-Time Endpoint Monitoring with Zero Trust Clients: The zero trust client can continuously monitor the security environment of the endpoint, ensuring the trustworthiness and safety of user terminals. Devices without a secure zero trust client cannot initiate valid access requests.

(2) Network "Stealth": The design maximally reduces the network exposure surface. The stealth gateway supports dynamic encryption mechanisms based on session-specific and device-specific keys, ensuring that even if the session key is compromised, it cannot be used for decryption.

(3) Diverse Authentication Methods: The stealth gateway employs a rich variety of authentication methods, including passwords, digital certificates, one-time passcodes, time-based factors, and IP addresses—ensuring at least five dimensions of authentication to maximize system security.

The rest of this paper is organized as follows: proposed scheme is detailed in Section II. Section III presents terminal scoring algorithm. Section IV details experimental analysis. Section V provides the security analysis. Section VI draws the conclusion.

2. Proposed Scheme

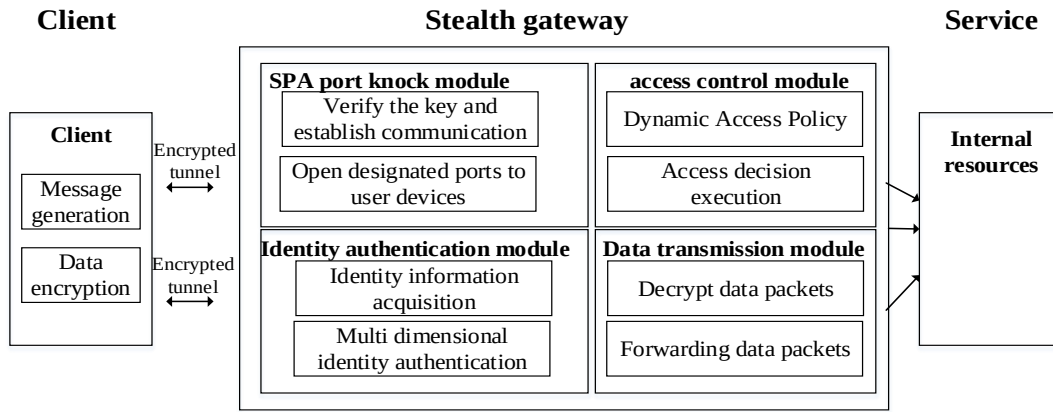


Figure 1. System Structure Diagram of Stealth Gateway

The system architecture diagram for the stealth gateway is shown in Fig. 1, comprising a zero-trust client, stealth gateway, and business system. The zero-trust client acts as the message sender, generating dynamically encrypted single-packet authentication messages. The stealth gateway serves as the receiver, decrypting the messages and extracting the necessary identity information for multi-factor authentication. Based on the authentication results, the stealth gateway decides whether to allow access to the business system. The business system stores the critical information required by users, which can be accessed by legitimate users after authorization by the stealth gateway. To achieve the objectives outlined in Chapter 1, the following technical solutions are adopted:

(1) Endpoint Sensing and Firewall Configuration:

Install the zero-trust client on user endpoints within the local area network (LAN). The zero-trust client continuously monitors and scores the security state of the user endpoint. Only when the score exceeds a predefined threshold is the user endpoint considered a trusted and legally authenticated terminal. Enable the firewall on the stealth gateway by default and configure the iptables firewall rules to deny all incoming traffic by default, discarding all incoming packets.

(2) Initiating Single-Packet Authentication Service:

Start the single-packet authentication service on the stealth gateway to monitor authentication results in real-time. On trusted user endpoints with the zero-trust client installed, start the zero-trust client to encrypt individual UDP packets and send them to the designated port of the stealth gateway for port knocking services.

(3) Authentication Process:

The server-side port receives the packet and decrypts it, verifying the decrypted data. Perform multi-factor authentication on the decrypted packet, including user identity information, IP address, and geographical location, to determine the legitimacy of the user. Upon successful validation of a legitimate user, grant access authorization and configure iptables firewall rules to allow the legitimate user's traffic through, specifying the scope of business access for the user. At this point, the single-packet authentication service running on the stealth gateway detects a successful single-packet authentication result. Set the access interval in the stealth gateway configuration file; if the user exceeds the access time range, their access privileges are revoked. Continuously monitor user access behavior; if abnormal behavior is detected, the stealth gateway will re-

intercept the user, preventing further access.

By implementing these technical solutions, the stealth gateway effectively manages access control and enhances the security of the business system, ensuring that only verified and authorized users can access critical information.

2.1. Terminal Perception and Firewall Configuration

Firstly, endpoint perception scoring is conducted. A zero-trust client is installed on user endpoints within the local area network (LAN). This zero-trust client performs real-time sensing of the security state of the user endpoint, encompassing fundamental security awareness, system security awareness, and compliance awareness for application software, and then aggregates these metrics into a comprehensive score. Only when this score exceeds a predefined threshold T , is the user endpoint considered to be a trusted entity authenticated by default; any endpoint without the zero-trust client installed is deemed untrusted, and all access requests initiated from untrusted endpoints are blocked.

Subsequently, the firewall configuration process is executed. By default, all incoming packets are discarded, thereby preventing external internet users from accessing any port within the LAN under normal circumstances, while not interrupting established communication sessions. Under these conditions, the firewall will, in the absence of user interaction, default to discarding all incoming User Datagram Protocol (UDP) packets, thus making direct access to backend resources impossible and presenting closed ports from an external perspective.

2.2. Single package authentication service startup

Furthermore, initiate the Single Packet Authentication (SPA) service. Upon activation, continuously monitor the SPA results; an empty monitoring result indicates that no SPA result was captured, whereas a non-empty result confirms successful detection of the SPA outcome. The specific process for Step 4 is outlined as follows: The Zero Trust client generates a client message, which includes information such as a 16-byte random number, device fingerprint, timestamp, version number, hardware characteristic code, message type, identity information, and geographical location. This message information is then concatenated according to predefined rules and encoded before a digest calculation is performed to form the plaintext information intended for encryption. The plaintext information is encrypted using a key, with the

sequence of operations being encryption followed by authentication to avoid issues related to cryptographic analysis. Finally, the Zero Trust client software is launched, initiating a port-knocking action to transmit the message to the designated port of the stealth gateway server.

2.3. Authentication

The final step is the authentication process. Prior to deploying a firewall with a default drop policy, transmit the key generated on the client to the server using SSH or another secure tool; this key is stored in the `/etc/fwknop/access.conf` file and includes `KEY_BASE64` and `HMAC_KEY_BASE64`. The Ethernet detector of the stealth gateway captures UDP packets in real-time from the specified port using PCAP packet capture methodology. The stealth gateway decrypts these packets using the key, and subsequently processes the captured UDP packets.

The gateway decodes the plaintext using BASE64 decoding. Once decoded, it extracts a 16-byte random number, timestamp, hardware characteristic code, message type, identity information, and geographical location according to defined rules. By parsing the plaintext, it acquires detailed information about the terminal device and user. It separates the plaintext from the digest according to the rules, calculates the digest of the plaintext, and compares it with the included digest. If they do not match, it suggests possible tampering with the inner data, leading to the packet's discard; otherwise, it proceeds to the next step.

When users request access, their passwords and one-time passcodes are verified. If authenticated successfully, the system checks whether the user's access time and location fall within predetermined parameters. Users whose time and location are normal are deemed legitimate; otherwise, they are considered unauthorized. Once a user's identity is confirmed as legitimate, authorization is granted to define the scope of services accessible to the user.

The first rule in the INPUT chain of the iptables in the stealth gateway is set to the FWKNOP_INPUT chain, which logs temporary admission rules. As this chain is at the top of the INPUT chain, packets from legitimate users will match the temporary rules in the FWKNOP_INPUT chain and be allowed through. At this point, the single packet authentication service of the stealth gateway detects a successful authentication result. After more than 30 seconds, the temporary rules are automatically deleted, and the stealth gateway returns to its "closed" state for all external ports. Simultaneously, the system continuously monitors the visitor's access behavior, implementing a secondary interception and blocking access if any anomalies are detected, such as real-time scores falling below a threshold.

3. Terminal Scoring Algorithm

3.1. Attributes and utility functions definition

Firstly, define the attributes related to endpoint security and establish a utility function for each attribute. These attributes encompass the following items:

- (1) Endpoint Basic Security Awareness Score, which includes: awareness score regarding the status of an enabled Windows firewall; awareness score concerning the installation of antivirus software.
- (2) System Security Awareness Score, comprising: awareness score on prevention of local identity theft; awareness score on password complexity; awareness score on

account lockout thresholds; awareness score on account audit management.

(3) Application Software Compliance Awareness Score, including: awareness score regarding the status of Bluetooth activation; awareness score concerning blacklisted services.

(4) User Behavior Analysis Score, evaluated based on: the normality of user login and operational behaviors.

(5) Authentication Mechanism Awareness Score, assessed by: the use of multi-factor authentication on the endpoint.

For each attribute, define a utility function $s_i(x)$ that maps the measurement x of that attribute to a utility value. The utility value is typically a numerical value between 0 and 1, where 0 represents the worst case, and 1 represents the optimal case.

3.2. Determination of attribute weights

The importance of each attribute w_i is then determined, $\sum_{i=1}^n w_i = 1$, often through expert evaluation or similar methods. The weights assigned to the attributes should satisfy the condition that the sum of all attribute weights equals 1.

3.3. Calculation of comprehensive utility value

With the utility functions and weights established, it is then possible to calculate the composite utility value s for each endpoint:

$$s = \sum_{i=1}^n w_i \cdot s_i(x_i) \quad (1)$$

Here, x_i represents the actual measurement value of the i attribute, which includes the awareness score $s1$ for the status of the Windows firewall, score $s2$ for antivirus software installation, score $s3$ for prevention of local identity theft, score $s4$ for password complexity, score $s5$ for account lockout thresholds, score $s6$ for account audit management, score $s7$ for status of Bluetooth activation, score $s8$ for blacklisted services, score $s9$ for user behavior analysis, and score $s10$ for authentication mechanism.

3.4. Standardization of utility value

Through a simple linear transformation, the utility values are normalized to a specific range, such as 0 to 100.

$$s' = k \cdot s + c \quad (2)$$

Coefficients k and c are used to ensure that s' falls within a desired range.

Set a score threshold T ; if the composite score result is greater than this threshold, $s' > T$, the endpoint is considered a trusted endpoint; otherwise, it is considered untrusted.

4. experimental analysis

The specific experimental process is as follows:

(1) Install the Zero Trust client on the host with IP 172.16.1.22. The Zero Trust client will provide real-time awareness of the security state of the user's computer terminal, including basic endpoint security awareness, system security awareness, and application software compliance awareness, and perform a comprehensive scoring. Only when the score exceeds a predefined threshold (the default threshold is 80) is the user's terminal presumed to be a legitimately authenticated and trusted endpoint. Terminals without the Zero Trust client installed are considered untrusted, and any access requests initiated from untrusted terminals are intercepted.

(2) On the stealth gateway with IP 172.16.1.3, the firewall is enabled by default. Configure the iptables firewall with a default deny access policy, discarding all incoming packets.

By default, the incoming rules of the server's firewall consist of only one rule: discard all incoming packets. This means that external users cannot access any internal port by default, and port scanning tools will indicate that all ports are closed. Additionally, a new chain named FWKNOP_INPUT will be separately added to the firewall for the subsequent dynamic addition and deletion of admission rules. The specific command is as follows:

```
iptables -A INPUT -p udp --dport 22 -m conntrack --ctstate ESTABLISHED,RELATED -j ACCEPT
```

```
iptables -A INPUT -p udp --dport 22 -j DROP
```

At this point, the firewall will discard all incoming UDP packets when there is no user activity, making direct access to the backend resources impossible, which means that the ports appear as closed to the external network.

(3) Input the single packet authentication script into the stealth gateway to initiate the single packet authentication service, and monitor the authentication results in real-time. First, navigate to the /app/script directory and execute the script to start the single packet authentication service. Use the monitoring script tail -f./nohup.out to perform real-time monitoring of the single packet authentication outcomes.

(4) On the trusted user terminals where the zero trust client is installed, start the zero trust client to encrypt individual UDP packets. These packets are then sent to the designated port on the server (port number 62235) for port knocking service. The zero trust client generates the client packets. The packet information is assembled according to rules, encoded, and then a digest calculation is performed to form the plaintext information ready for encryption. The plaintext information is encrypted using a key. Subsequently, the zero trust client program is launched to trigger the port knocking action, sending the packets to the specified port on the stealth gateway server. The specific operational commands are as follows:

```
fwknop -A udp/22 -a clientip -D serverip -key-gen -use-
```

```
hmac --save-rc-stanza
```

(5) The server's port 62235 receives the packets, decrypts them, and validates the decrypted packets. Before deploying a firewall with a default drop policy, use SSH or another secure method to transfer the keys generated by the client to the server's /etc/fwknop/access.conf file, primarily including KEY_BASE64 and HMAC_KEY_BASE64. The Ethernet detector of the stealth gateway captures UDP packets in real-time from the designated port using the PCAP packet capture method. Subsequently, the stealth gateway decrypts these packets using the key, and proceeds to identify and process the captured UDP packets.

(6) Upon decryption, the packet content undergoes multi-dimensional authentication, including verification of user identity information, IP address, geographical location, etc., to determine the legitimacy of the user and establish their security level. The packet is decoded from BASE64; upon completion of decoding, a 16-byte random number, timestamp, hardware signature, message type, identity information, and geographic location are extracted according to established rules. This process reveals the plaintext and retrieves detailed information about the terminal device and user. A hash algorithm is applied to the plaintext, and the computed hash is compared with the one carried within the packet; if they do not match, it suggests possible tampering with the inner data, leading to the packet being discarded; if they match, multi-factor authentication ensues to verify the user's legitimacy and determine their security level based on their identity information.

(7) Once the legitimate user has been verified, access authorization is granted, and iptables firewall policies are configured to allow legitimate users through, specifying the scope of business access for designated users. At this point, the single packet authentication script running on the stealth gateway can detect the successful outcome of single packet authentication, as illustrated in Fig. 2.

Figure 2. Monitoring Results

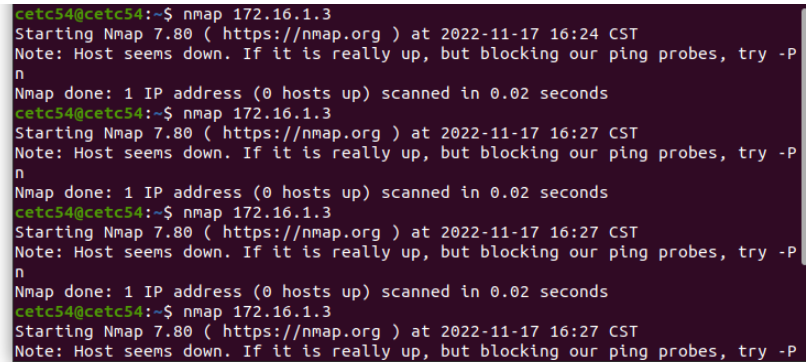
(8) Configure the access time intervals within the stealth gateway settings; users who exceed the designated access period will lose their access privileges. Continuously monitor user access behavior, and if any anomalies are detected, the stealth gateway will perform a secondary interception to prevent further access by the user. Specifically, temporary

rules are automatically deleted after a default period of 30 seconds. Following a successful single packet authentication, the stealth gateway resumes its "closed" state towards all external ports after 30 seconds, preventing continued access to the system. Real-time monitoring of visitor access behavior will trigger a secondary interception and block the access

process if the real-time score drops below a threshold (default 80) indicating anomalous behavior.

(9) During the access process, attempt to scan all ports of the stealth gateway using an unauthorized user terminal, such as a Kali Linux virtual machine, without installing a zero-trust

network client. Open the terminal within the Kali VM and enter the port scanning command `nmap 172.16.1.3`, which yields no results, as detailed in Fig. 3. This procedure is designed to test the "stealth" effect of the network configuration.



```
cetc54@cetc54:~$ nmap 172.16.1.3
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-17 16:24 CST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.02 seconds
cetc54@cetc54:~$ nmap 172.16.1.3
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-17 16:27 CST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.02 seconds
cetc54@cetc54:~$ nmap 172.16.1.3
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-17 16:27 CST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.02 seconds
cetc54@cetc54:~$ nmap 172.16.1.3
Starting Nmap 7.80 ( https://nmap.org ) at 2022-11-17 16:27 CST
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 0.02 seconds
```

Figure 3. Result of Nmap Scan

5. Security Analysis

G1 (Confidentiality). In the methodology proposed herein, only legitimate user devices are authenticated through the stealth gateway and allowed to proceed with subsequent communications. The multi-factor authentication of clients includes at least five dimensions such as passwords, digital certificates, one-time passwords, timestamps, and IP addresses, thereby enhancing the security and confidentiality.

G2 (Integrity). This protocol ensures message integrity through hash operations. The authentication gateway computes a hash and compares it with the message received from the user device to ensure it has not been tampered with. User devices verify the integrity of messages by computing their own hashes and comparing them with the information received from the authentication gateway.

G3 (Anonymity and Non-traceability). During the authentication process, all authentication-related messages are encrypted. Unauthorized users and devices cannot decrypt these messages, preventing unauthorized access to the content.

G4 (Protection against Replay Attacks). Due to the inclusion of the HMAC value of the entire data packet in the knock package sent by the client, the risk of replay attacks is effectively mitigated.

G5 (Resistance to DDoS Attacks and Scanning Attacks). When clients request a connection, the stealth gateway first performs a security check on the client; any client that does not have a zero-trust client installed is considered untrusted and its access requests are blocked. Clients that have a zero-trust client installed but fail the knock package authentication receive no response from the stealth gateway. Since the stealth gateway hides its ports, it remains invisible to scanning attempts, thus preventing resource exhaustion attacks like DDoS and probing attacks like scanning.

G6 (Prevention of Insider Attacks). Once authenticated, users' activities within the system are continuously monitored by the stealth gateway. If abnormal behavior is detected and the real-time scoring falls below a threshold, the user is subjected to secondary interception, blocking further access, which effectively prevents insider attacks.

6. Summary

In summary, this paper centers on Software-Defined Perimeter (SDP) technology, dynamically authorizing access initiated by authenticated devices while blocking access requests from unauthenticated devices. After legitimate users are authenticated via multiple factors including passwords, one-time tokens, IP addresses, and more, their access requests are forwarded through a stealth gateway. Compared to traditional Virtual Private Networks (VPNs) and firewalls, the stealth gateway effectively reduces the attack surface of the network, implements multi-factor authentication, and achieves perimeter invisibility, thereby enhancing the security of data across the network.

Under the condition that the zero-trust client is installed on the device, the stealth gateway can continuously monitor the security environment of the endpoint in real time, ensuring the trustworthiness of the device. The stealth gateway supports SSL/TLS encrypted tunnels and offers a rich variety of authentication methods, including at least five dimensions of multi-factor authentication such as passwords, digital certificates, one-time passwords, timestamps, and IP addresses, ensuring the maximum security of the system. Compared to conventional network protection methods, the proposed solution in this paper significantly diminishes the exposed network surface, achieving comprehensive, multi-factor authentication and access control.

Conflicts of interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

- [1] Qiu, R., Fu, Y., Le, J., Zheng, F., Qi, G., Peng, C., & Li, Y. (2022). A software-defined security framework for power IoT cloud-edge environment. *International Journal of Network Security*, 24(6), 1031-1041. <https://doi.org/10.1109/ICCASIT55263.2022.9986934>
- [2] Chen, J., Miao, F., & Wang, Q. (2007, April). SSL/TLS-based secure tunnel gateway system design and implementation. In

- 2007 International Workshop on Anti-Counterfeiting, Security and Identification (ASID) (pp. 258-261). IEEE. <https://doi.org/10.1109/IWASID.2007.373739>
- [3] Al Sibahee, M. A., Nyangaresi, V. O., Abduljabbar, Z. A., Luo, C., Zhang, J., & Ma, J. (2023). Two-Factor Privacy Preserving Protocol for Efficient Authentication in Internet of Vehicles Networks. IEEE Internet of Things Journal. <https://doi.org/10.1109/JIOT.2023.3340259>
- [4] Zhang, P., Tian, C., Shang, T., Liu, L., Li, L., Wang, W., & Zhao, Y. (2021, May). Dynamic access control technology based on zero-trust light verification network model. In 2021 International Conference on Communications, Information System and Computer Engineering (CISCE) (pp. 712-715). IEEE. <https://doi.org/10.1109/CISCE52179.2021.9445896>
- [5] Moubayed, A., Refaey, A., & Shami, A. (2019). Software-defined perimeter (sdp): State of the art secure solution for modern networks. IEEE network, 33(5), 226-233. <https://doi.org/10.1109/MNET.2019.1800324>