Journal of Computers Vol. 34 No. 5, October 2023, pp. 45-57
doi: 10.53106/199115992023103405004

Research on the Security Protection of Secondary Distribution System Based on Software Defined Perimeters

Erxia Li, Yuling Li, Zilong Han*, Chaoqun Kang, Li Wang

Digital Grid Technology Center State Grid Shanghai Energy Interconnection Research Institute, Shanghai 201210, China

hzlepri@126.com

Received 10 August 2022; Revised 19 January 2023; Accepted 13 March 2023

Abstract. With the new technologies including big data, cloud computing, the Internet of Things (IoT), mobile Internet, and Artificial Intelligence (AI) coming into widespread in the secondary distribution system, it makes the network boundary more fuzzy, and the network security risk points and exposed surfaces significantly increase. Therefore, the traditional boundary-based security protection model has been unable to meet the protection needs. As one of the most popular security concepts at present, the zero trust mechanism can achieve the dynamic protection of information intranets. Based on the concept of zero trust security and software defined perimeter (SDP) technology, this paper designs and implements a security scheme suitable for the secondary distribution system and proposes a novel identity authentication model which can solve the problems of port exposure that existed in the traditional authentication scheme. In addition, the model applies the SM9 identification algorithm to reduce the computational cost of the encryption and decryption in the proposed scheme. Finally, the performance analysis demonstrates that the proposed scheme is effective and suitable for the secondary distribution system which can effectively resist multiple types of network attacks.

Keywords: secondary distribution system, software defined perimeter, secure access, identity authentication

1 Introduction

As the secondary distribution system [1] (SDS) becomes more critical for the electric energy supply system, how to build a structured network and improve its efficiency has become an urgent problem. The main function of the SDS is to realize human contact with the primary system to monitor, control, and enable the primary system to operate safely and economically.

At present, the current security protection scheme of the SDS is based on the principle of "security partition, dedicated network, horizontal isolation and vertical authentication" [2], and the security boundary is established to resist attacks. However, some potential security problems in the SDS cannot be solved by the traditional boundary-based security protection model [3], such as malicious control of distribution terminals, eavesdropping of communication protocols, and tampering of control commands.

As the first line of defense for SDS against external attacks, secure access protocol establishes a secure channel by completing authentication and session keys agreement for terminals. Conventional public-key-infrastructure-based protocols are clearly not suitable for SDS, due to the resource-constrained nature of distribution terminals [4]. In addition, most of the current schemes expose intranet ports, which allows DDoS attacks to easily destroy intranet resources and cause service disruptions [5]. Therefore, a secure authenticated key agreement protocol is needed to tackle security and privacy issues in the SDS.

In the view of the above problems, a security protection scheme for the SDS based on Software-defined Perimeter [6] (SDP) is proposed in this paper. Unlike the existing security scheme, we take the zero trust security mechanism as the guide and combined with the real demands of the SDS, like the security demand of the massive terminal access. The proposed scheme designs a terminal security access model based on the SDP framework, and completes authentication and access control based on SM9 algorithm and terminal identity by setting SDP controller in the security access area of SDS. In addition, the SDP controller first completes the access authentication of the terminal, and then the gateway and the terminal complete the session key agreement. The authentication is based on the SM9 algorithm and the identity information of the terminal. Since SM9 is certificateless, there is no need to worry about the certificate management problems caused by the significant in-

* Corresponding Author

crease of terminals. The proposed scheme separates the authentication process of the SDP controller and the data transmission between the gateway and the terminal, realizes the resource hiding behind the gateway, and ensures the security of the SDS service system. Meanwhile, the scheme can cope well with the mainstream attacks of the SDS and has a good performance on dealing with the unknown threats, which solves the security problem in the field of SDS's terminal access.

This paper is organized as follows. Section II and III presents the background and related work of the zero trust mechanism and SDP. Section IV proposes the security protection scheme of the SDS based on SDP. Section V and section VI analyses the security and performance of the proposed scheme. Finally, Section VII presents the conclusion and determine the work in the next research.

2 Related Work

With the increasing scale of SDS and the trend of cloud deployment [7, 8], the traditional boundary-based protection scheme become inefficiency to cope with the gradually diversifying malicious attack. Therefore, it is necessary to introduce a new security protection mechanism. The zero trust mechanism was introduced in 2010 [9, 10], which is an endogenous security mechanism for dealing with various threats in a borderless network environment. The zero trust security mechanism can be autonomous, adaptive and self-growing. It addresses threats arising from traditional boundary-based security schemes [11]. The core principle of the zero trust mechanism is that the participants in the network should not be trusted and any access to system resources should be considered to be a potential threat. Therefore, the participant and access should be checked and verified [12]. Currently, various mainstream zero trust security architectures have been proposed by the security research institutes such as Google, Qi'anxin, Tencent, and the National Institute of Standards and Technology (NIST) [13-15].

In the field of the power grid, many scholars have already studied how to realize zero trust security mechanisms in smart grid. Xiao, Z. et al. [16] studied the "zero trust" typical business scenario of the power Internet of Things with "Continuous Identity Authentication and Dynamic Access Control" as the core, and designs the power IoT security protection architecture based on zero trust. But their continuous identity authentication and access control will cost lots of computing resources. Mir, A. W. and K. R. Ram Kumar proposed a zero trust user access and identity security model that can be implemented in a smart grid-based SCADA system in [17]. Alagappan, A. et al. [18] studied the zero trust model in the virtual power plants, which effectively prevent the single compromised terminals from spreading laterally and infecting the whole network. However, the scheme is mainly aim at the virtual power plant and the distributed generators, which doesn't fit the security demands of the cloud deployment of the SDS.

The cloud-deploying SDS becomes more virtual, so if the SDS can't hide the service system, it will cause the illegal visit between different service systems. The attackers will invade the system and finally cause the malicious control [19]. Currently, the zero trust protection scheme can't secure the cloud-deploying SDS.

Software-defined Perimeter (SDP), as an implementation of zero trust mechanism, is rising attention due to its ability of hiding resource and secure accessing [20]. Scholars has introduce the SDP structure in power IoT field. R.-X. Qiu, et al. [21] design a software defined security framework for power IoT. The framework uses SDP to protect the security of the cloud and the inner layer of system by rejecting all unauthorized edge traffic. Y.-C. Palmo, et al. [22] investigated and evaluated several federation methods to embed IoT devices into SDP and found that the identity provider (IdP) is the most effective methods. However, the IdP will costs lots of computing and storage resources to store the identity of the terminal. It still need to be improved to apply into the SDS.

Aiming at the above problems exist in the existing schemes, this paper proposes a security protection scheme for the SDS based on SDP, which takes the zero trust security mechanism as the guide and combined with the real demands of the SDS.

The main contributions of this paper are as follows:

(1) Our scheme apply Single Packet Authorization (SPA) technology to hide the resource of the SDS cloud main station and improve the security of the system.

(2) Our scheme authorized terminals' identity and manage the system resource based on the SDP controller.

(3) Our scheme define subject attributes, environment attributes and object attributes in the identity authorization, and use SM9 algorithm to signature and verify, reducing the computing and storage costs and guarantee the integrity and non-repudiation of the data.

3 Preliminaries

3.1 Zero Trust Security Mechanism

The zero trust security mechanism is different from the traditional network boundary-based security mechanism [23]. According to the concept of the zero trust, any internal or external device, application and user in the network cannot be trusted by default. The zero trust security mechanism has three layers including control plane, data plane, and identity management infrastructure, which make it achieve end-to-end security control from subject to object, and the overall security architecture is shown in Fig. 1.

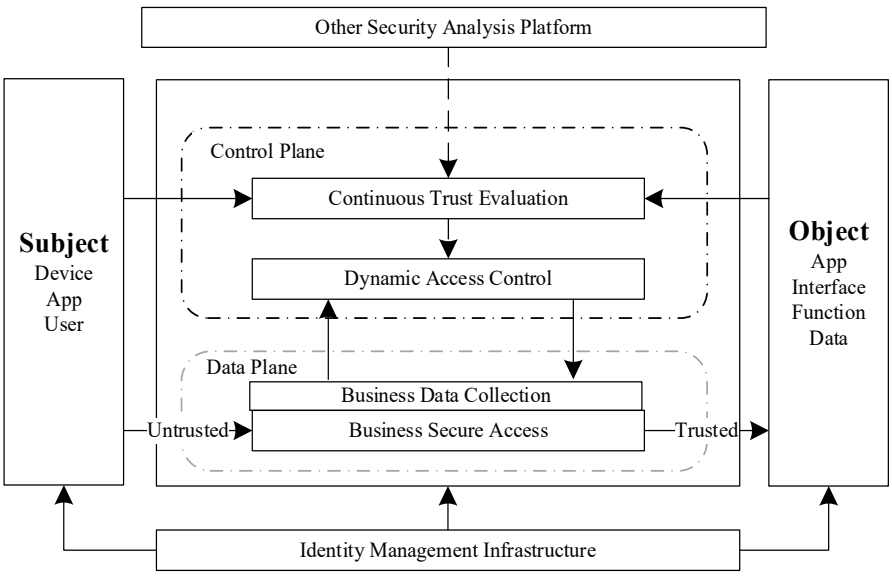

Fig. 1. Zero trust architecture

The control plane is responsible for configuring and managing the access rights and access policies of the subjects. After multidimensional authentications, the control plane and the identity management will generate dynamic access control policies, which can configure the data plane dynamically. By collecting and analyzing the network traffic, the continuous trust evaluation module will conduct a trust level, which stand for the reliability of the corresponding subject. According to the trust level, when a severely untrustworthy trust level is detected, the dynamic access control module will determine whether the privileges should be changed and further decide whether re-authentication or access blocking. Besides the control plane, there's a data plane, including service systems, proxy servers and network devices, which provides hardware and data support for the control plane.

In a nutshell, the zero trust security mechanism is identity-centric. Without distinguishing between internal and external networks but hiding the network resource, the zero trust security mechanism ensures that only authenticated and legitimate terminals or users can access. SDP technology is designed to build virtual boundaries for enterprises through software, so SDP is a proper implementation technique to achieve the zero trust security mechanism in the field of the SDS.

3.2 SDP & SPA

The SDP concept originated from the U.S. Defense Information Systems Agency and has been formally endorsed and popularized by the Cloud Security Alliance over the past decades. SDP embodies the principle of zero trust at the network level by introducing trust mechanisms to control and grant access requests from terminals, which is the highest-level implementation of zero trust security mechanisms. The Deny-All firewall of SDP can be used in the SDS to achieve resource hiding of service system. Any request to access resources must first complete a

single packet authorization at the control plane before it is allowed to establish a trusted connection at the data plane, and any unauthorized packet will be discarded.

The SDP architecture is shown in Fig. 2. SDP consists of three main components: the SDP connection initiating host (IH), the SDP controller and the SDP connection accepting host (AH). The SDP controller can determine which SDP IH can connect with SDP AH, and it can also forward authentication information to external authentication servers, such as authentication servers and geolocation authentication servers. The IH should communicate with the SDP controller firstly to request a list of AH's port and the SDP controller can request hardware or software information from the IH to confirm the identify of IH. The AH should reject by default for all communications from all hosts except the SDP controller, and the AH accepts connections from the IH only after receiving instructions from the SDP controller. Thus, the SDP architecture achieves separation of the control plane from the data plane. All components can be multiple instances for ease of expansion and proper usage.

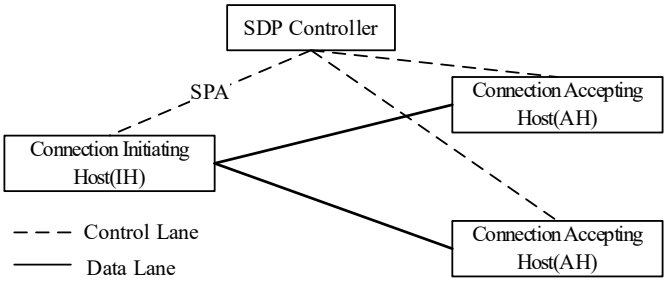

Fig. 2. SDP Architecture

Based on single packet authorization (SPA) technology, SDP implements the access strategy of first authentication and then connection, which can avoid the port exposure risk caused by the traditional authentication protocol. SPA is a lightweight security protocol. When the terminal or user wants to access the SDP controller or gateway or other related system components, the terminal or user must be authenticated through SPA first, so various applications and web servers can be hidden behind the firewall. The firewall can discard all unauthenticated packets by default. So the ports cannot be obtained by the attacker using scanning tools. An important principle of SPA is that packets must be authenticated and encrypted and the server must receive and process packets without replying or sending any acknowledgment message.

3.3 SM9 Signature Algorithm

The SM9 signature algorithm, as an identification cryptographic algorithm, has been strongly supported by the Chinese government. Since 2006, the State Cryptography Administration has organized scholars to carry out the development of the standard specification of China's identification cryptographic algorithm and issued the commercial cryptographic algorithm model SM9 in 2008. The improvement and modification of the standard algorithm was completed in 2014, and the algorithm was officially announced by the State Cryptography Administration in March 2016 with the standard number GM/T 0044- 2016.

The SM9 algorithm, like other identity-based cryptographic algorithms, is secure and efficient based on the elliptic curve bilinear mapping. The SM9-based cipher is identity-based and both communicating participants can calculate each other's public keys based on their identities, thus reducing the complexity of key exchange and key management. Therefore, the security protection scheme using an identity-based cryptographic algorithms can meet the authentication security requirements of SDS. What's more, it also reduces the complexity of certificate management and the bandwidth burden of network communication, which is suitable for the SDS with massive terminals' access. In the encryption and decryption algorithm based on the SM9 identification cipher, the public key of the user comes from its identity information, and the private key is generated by Key Generation Center(KGC). As long as the identity information of user A is obtained, user B can get the public key of user A to encrypt a message and make it securely transmitted to user A in the form of cipher text over the network, and user A can decrypt the message after getting his private key from the SM9 key center. The process is shown in Fig. 3.

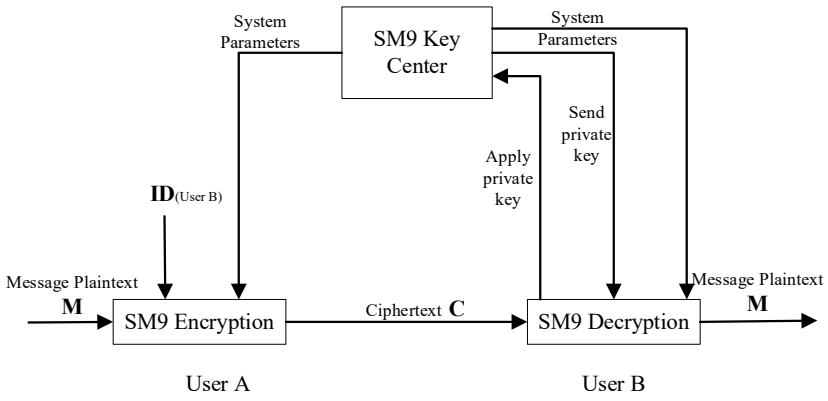

Fig. 3. SM9 encryption & decryption

SM9 digital signature is implemented based on identity cryptographic algorithm. In the traditional certificate-based cryptographic algorithm, if user A wants to verify user B's digital signature, he must first obtain user B's certificate, and verify user B's identity through the existing signature in the certificate. He also has to verify the validity of the signature through user B's public key. However, in the SM9 identity-based cryptographic algorithm, user A can verify user B's signature by directly obtaining user B's identity information ID, as shown in Fig. 4.

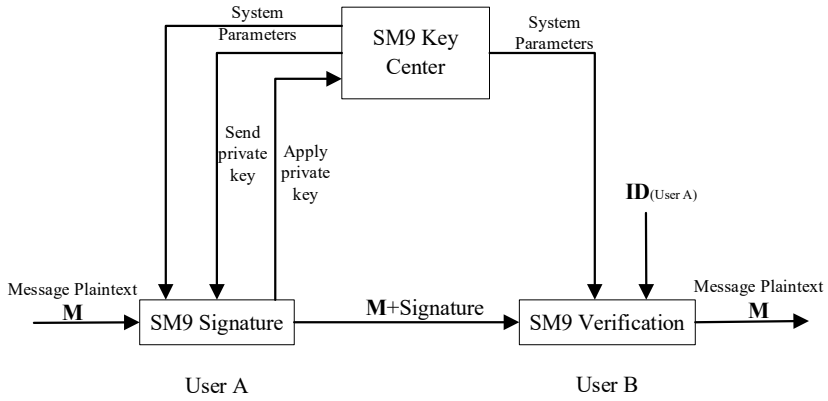

Fig. 4. SM9 signature & verification

The SM9 identification cryptographic algorithm takes all unique identifiers, such as the device fingerprint at the terminal layer of the SDS, as public keys. Without the need for digital certificates, SM9 algorithm can securely distribute exclusive private keys. The authentication process without user name and password transmission eliminates security problems such as weak passwords, brute force cracking, and collision attacks. Because the identity is the public key, there is no certificate exchange authentication process under the premise of ensuring security, and it also takes the ease of usage into account.

4 Proposed Scheme of SDS Based on SDP

As the plentiful distribution terminals accessing and the service system cloud-deploying, the SDS is facing wider attack surface. Attackers may attack the SDS in a roundabout way, such as misreporting fault information through distribution terminals, thus causing a wider security threat. Therefore, securing the access of the terminals has become an urgent problem of the SDS. In this paper, we make full use of the security infrastructure already built

in the power grid by setting up and deploying SDP controllers and gateways in the security access area to make the SDS servers hidden behind the firewall. All unverified packets received are discarded by default to ensure that the SDS servers do not respond to unauthorized connection requests, so attackers cannot know whether the requested ports are being listened to, thus achieving the hiding of ports and ensuring the network security of the SDS. The security protection scheme of the SDS based on SDP is shown in Fig. 5 as follows.

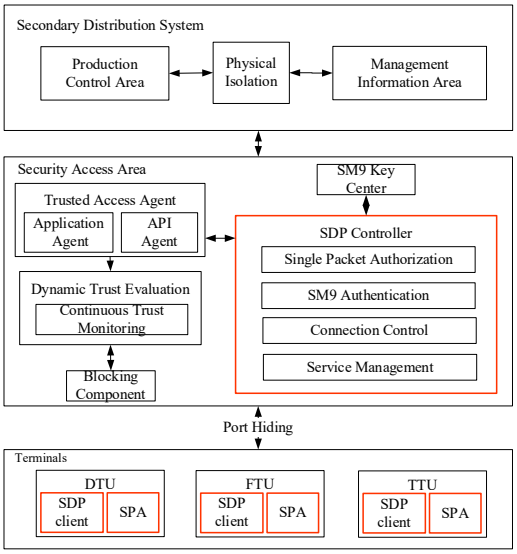

Fig. 5. The security protection scheme of the secondary distribution system based on SDP

The SDS security protection scheme achieves the security protection mechanism of first authentication and then connection through the SDP controller, which is responsible for SPA of distribution terminals and dynamic opening of service communication ports. It can also reduce the exposure of the system and ensure that the SDS does not respond to unauthorized connection requests. In addition, the protection scheme uses the SM9 algorithm to achieve identity-based encryption and signature, which effectively reduces cost of distribution terminal communication. After the terminals authorized by a single packet, the SDP controller needs to continuously evaluate the trust level of terminals based on the zero trust security mechanism. And for the distribution terminals with abnormal behavior, it will promptly notify the SDP controller to block access to avoid malicious attacks or damage to the SDS servers. The process of the security protection scheme designed in this paper is shown in Fig. 6.

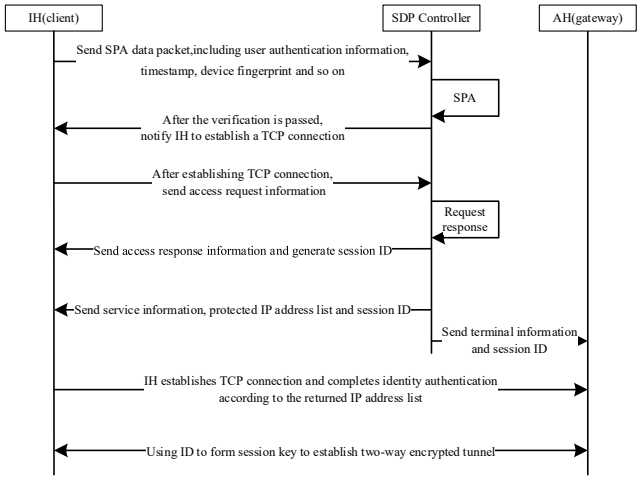

Fig. 6. Interaction diagram of the security protection scheme based on SDP

A summary of the notations used in this paper is presented in Table 1.

Table 1. Summary of notations

Symbol	Description
ID_{IH}	The device fingerprint
ID_{SDP}	The SDP controller identity
ID_S	The session identity
SK_{SDP}	The SDP controller's private key
DK_1	The session key calculates by the terminal side
DK_2	The session key calculates by the gateway side
Enc(*)	Encrypting the data packets
Sign(*)	Signing the data packets
Hash(*)	Performing hash operations
Port	The service port
Timestamp	Current timestamps of the terminals
type	Type of key exchange
subtype	Subtype of key exchange
length	Length of the data packet
$\oplus$	Bitwise exclusive (XOR) operations

Step 1: When the distribution terminal online, it first sends SPA packets to the SDP controller, which include the ciphertext of the timestamp and device fingerprint, and uses the SM9 algorithm for signature. The device fingerprint ID_{IH} consists of subject attributes (MAC address, operating system, port, protocol, service, vendor), environment attributes (online time, IP, access location, service traffic data size) and object attributes (belonging department, manager, authorization time, authorization level). IH uses SM9 asymmetric encryption on the random number R_1 and timestamp Time Stamp by using the preset SDP controller identity ID_{SDP} to obtain:

$$C_1 = \text{Enc}\ (R_1 \,||\, \text{Timestamp}, ID_{SDP})\ . \tag{1}$$

and SM9 digital signature on C_1 and the hash value of the message header to obtain:

$$S_1 = \text{Sign}\ (\text{Hash}\ (\text{type} \,||\, \text{subtype} \,||\, ID_{IH} \,||\, C_1), SK_{IH})\ . \tag{2}$$

where the hash algorithm uses the SM3 algorithm and finally obtains the complete SPA packet (type || subtype || ID_{IH} || C_1 || S_1) and sends it to the SDP controller based on the UDP protocol.

Step 2: The SDP controller performs the signature verification process for the received SPA packets and then uses its own private key SK_{SDP} to decrypt the packets with SM9 to obtain the random number R_1 and timestamp and judge the data freshness according to the timestamp information in the SPA packets. Then, it directly discards the timeout packets. The SDP controller will dynamically open the firewall port according to the management policy and use ICMP to send a message (type || subtype ||| C_2 || S_2) to the terminal to inform it to establish TCP connection, where C_2 and S_2 are:

$$C_2 = \text{Enc}\ (\text{Port} \,||\, \text{Timestamp}, ID_{IH})\ . \tag{3}$$

$$S_2 = \text{Sign}\ (\text{Hash}\ (\text{type} \,||\, \text{subtype} \,|||\, C_2), SK_{SDP})\ . \tag{4}$$

Step 3: After receiving the notification message from the SDP, the terminal initiates a TCP connection and sends an access request message according to the port in the message, indicating that the IH is ready and wants to join the trust management list of the SDP.

Step 4: After receiving the terminal's access request, the SDP controller determines whether the terminal has completed SPA authorization. The terminal that has completed authorization uses the random number R1 in its SPA packet to generate the session identification ID_S to uniquely identify the terminal's access behavior, and the SDP controller sends the access permission packet (type || subtype || length ||C_3 || S_3) to the terminal, where the access response packet is:

$$C_3 = \text{Enc (Service List} \,||\, ID_S \,||\, \text{Timestamp}, ID_{IH}) . \quad (5)$$

The Service List is a list of available services, usually in the form of an array of services in JSON format, including ports, IP addresses, service names, etc., and the terminal calculates:

$$S_3 = \text{Sign (Hash (type} \,||\, \text{subtype} \,||\, \text{length} \,||\, C_3), SK_{SDP}) . \quad (6)$$

At the same time, the SDP controller sends the identity information ID_{IH} of the access terminal and the session identification IDS of these access requests to the gateway side for session key generation and notifies the gateway that this terminal has completed the SPA operation.

Step 5: After receiving the access response information from the SDP controller, the terminal establishes a TCP connection with the port based on the service information Service List in the access response packet, synthesizes the session key $DK_1 = ID_{IH} \oplus ID_S$ based on its own device ID and session ID, uses the SM3 algorithm to hash the session key DK_1 to obtain $M_1 = SM3(DK_1)$ and sends (type || subtype || length || ID_{IH} || Timestamp || M_1) to the gateway side.

Step 6: After receiving the terminal access pass notification from the SDP controller, the gateway side synthesizes the session key $DK_2 = ID_{IH} \oplus ID_S$ using the identity information ID_{IH} of the accessing terminal and the session identifier ID_S of the access request. The SM3 algorithm is used to hash the session key DK_2 to obtain $M_2 = SM3(DK_2)$. If $M_1 = M_2$, the final session key $DK_1 = DK_2$; otherwise, the gateway reports to the SDP controller and closes the TCP connection.

To achieve a continuous evaluation of terminal trust level and session security, the terminal needs to send the heartbeat of SPA to the SDP controller periodically. If the SDP controller does not receive SPA heartbeat from the corresponding terminal within a certain period of time, the SDP controller will determine that the terminal's session is invalid and notify the AH gateway to disconnect the TCP connection with the terminal. When the terminal applies to the SDP controller and the gateway to establish a TCP connection again, SPA is required again. To ensure the security of the session key, the SDP controller can force the IH to perform SPA again according to the service instructions and at the same time update the session ID and notify the gateway side to regenerate the session key. The gateway side and the terminal side will negotiate the key again to update the session key so that the session key of encrypted data is always in a dynamic update state, which enhances the security of the two-way encryption tunnel. Since there is a source network address translation (SNAT) when the distribution terminals access, this paper binds the SPA packet with the hardware device fingerprint by adding the device fingerprint in the SPA packet, which can prevent external malicious attackers from tampering with the source IP by SNAT and thus recognize the disguised identity. In addition, in the actual scenario of distribution, the number of terminals deployed is extremely large, causing problems such as a large number of certificates and difficulties in management. Using a device fingerprint to replace the traditional digital certificate method to realize the authentication of terminals can effectively reduce the communication and computing pressure of terminals.

The designed security protection scheme in this paper makes use of SPA technology to realize the zero trust mechanism and achieve resource hiding so that the attacker cannot know the communication port of the SDS system without completing single-packet authorization, which greatly reduces the success rate of the attack. In addition, the scheme in this paper can still use the established basic cryptographic facilities of the SDS to complete authentication and data encryption communication. After the distribution terminal completes SPA, it can simplify the authentication process between it and the gateway or the SDS servers, which greatly reduces the computation and communication pressure of the gateway and terminals.

5 Security Analysis

The implement of the zero trust security mechanism and SDP framework for information network security protection in the SDS can effectively reduce the attack surface, hide and protect core resources, and monitor terminals through continuous trust evaluation to block unauthorized or abnormal accesses. The security analysis is as follows.

(1) Replay attack

Replay attack, which refers to an attacker's aggression against the host system by sending a packet that has already been received by the host, can occur in two phases: SPA and service interaction in the SDS. The SPA belongs to the control plane in the zero trust mechanism structure, and the SDP controller uses the timestamp to judge the freshness of the message. The data plane interaction takes place only after completing the authentication at the control plane, so the service system is not affected. If a replay attack occurs during the service interaction, the SDP controller can also use timestamp for security defense due to continuous monitoring at the control level and in the service interaction messages.

(2) Man-in-the-middle attack

Man-in-the-middle attack (MITM) is an attack method in which an attacker uses various technical means to virtually arrange a computer between two communicating parties to disrupt communication by intercepting normal network communication data, data tampering, and sniffing. In the SDS, the attacker uses the man-in-the-middle attack to obtain SPA packets, and due to the use of the SM9 algorithm for encryption and signature, the attacker lacks the private key of IH, and the data message cannot be decrypted and tampered with, so it cannot complete the SPA. If the man-in-the-middle attack means is used in the service interaction stage, the same attacker cannot obtain the session key of IH and AH, and the communication message adopts a timestamp to ensure the freshness of the session, which can effectively resist the man-in-the-middle attack.

(3) DDoS attack

Distributed denial of service attack (DDoS attack) is a network attack that uses certain defects of network protocols and uses a disguised approach. A DDoS attack makes the server receive a large number of messages requesting replies in a short period of time, occupying network resources and affecting the normal provision of network services by the server. In the SDS discussed in this paper, SDP uses UDP to complete single packet authorization, which consumes fewer resources compared with the TCP protocol and improves the availability of servers to process and discard invalid packets on a large scale. SPA makes SDP controllers and gateways more resilient against DDoS attacks. Although SPA cannot stop DDOS attacks, it can mitigate the impact of the server's computational consumption due to DDOS attacks and improve server availability.

(4) TCP SYN flooding attack

TCP SYN flooding attack sends a large number of TCP first handshake SYN packets to the target port to occupy system resources, resulting in system denial of service. In the proposed scheme, SPA uses UDP communication and does not expose the TCP port of the service, so malicious attackers cannot know the TCP port, and the SDP controller discards all data from malicious clients and only allows legitimate clients to access the service system safely.

(5) Insider threat

In the security protection principle of the power grid, insiders are reliable, but with the escalation of attack methods, insiders bring security threats that have a greater impact and damaging effect than external security threats. The security scheme designed in this paper can configure the access rights of users so that they can only access the data resources within their defined responsibilities, and other resources are hidden to avoid the security risks caused by the lateral internal movement.

(6) System and application vulnerabilities

The security protection scheme proposed in this paper can effectively reduce the attack surface, hide system and application vulnerabilities, and be invisible to unauthorized users to avoid malicious users by exploiting vulnerabilities.

A comparison of the security properties between the proposed scheme and scheme [26] and scheme [27] is provided in Table 2. We use "√" and "×" to represent whether the scheme satisfies respective security property or not.

Table 2. Security comparison

Security properties	Scheme [26]	Scheme [27]	Ours
Replay attack	√	√	√
MITM	√	√	√
DDoS attack	×	×	√
TCP SYN flooding attack	×	×	√
Insider threat	√	×	√
System and application vulnerabilities	×	×	√

Scheme [26] and scheme [27] use SM9 algorithm and SM2 with digital certificate respectively to achieve a secure and efficient access. However, their system structure will expose the TCP port to potential threats. Attacks like DDos attack and TCP SYN flooding attack will be hard to cope with by this situation. Scheme [27] also will suffer the key disclosure problem due to its PKI infrastructure. What's more, these kinds of scheme can't reach anonymity and untracebility.

In our scheme, the control plane and the data plane are separated so the access and data transmission will be separated, too. By introducing a SDP controller in the system structure, any suspicious access requirement will be discard by default, so the TCP port won't be detected. The scheme makes the system be able to cope with the DDos attack and TCP SYN flooding attack because the TCP port is hided and the attackers won't know whether the service system is exist or not. Therefore, this solution realizes the security protection of the secondary system of power distribution based on SDP and SPA technology, which is more advantageous in terms of communication consumption and computation consumption, i.e. It realizes the authentication and security access of terminals efficiently and further improves the information security protection level.

6 Performance Analysis

We evaluate the performance of the security solution designed in this paper in terms of computational costs and communicational costs. The test environment includes a simulated terminal server and an access gateway server with a Centos7 X64 operating system and Intel(R) Xeon CPU E3-1230 V3@3.40 GHz, written in C language. The simulated terminal server runs the SPA and IH Customer End-Program, and the access gateway server runs the SDP controller and AH. Linux C is used for both client and server. The test structure is shown in the following Fig. 7.

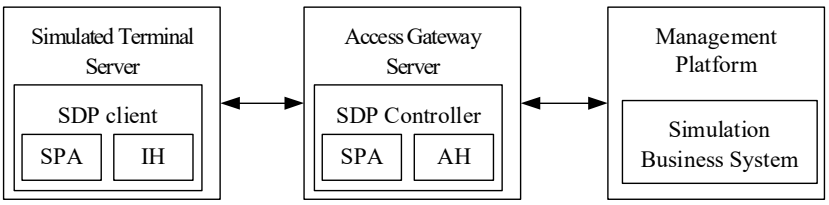

Fig. 7. Test structure

In addition, the performance comparison is divided into two parts, comparing the SPA protocol proposed in this paper with the currently popular FWKNOP [24] and OPENSPA [25] protocols and comparing the security access protection scheme proposed in this paper with the scheme [26] and the information security access protocol of the smart grid [27], by simulating the time required to complete the single packet authorization and security access of the access gateway under the different numbers of terminals. The comparison results are shown below. The results of the comparison are shown in the following Fig. 8 and Fig. 9.

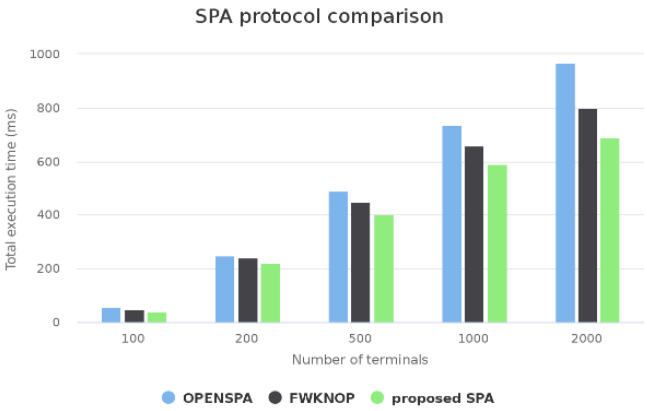

Fig. 8. Comparison diagram of SPA test

Fig. 9. Comparison diagram of overall scheme test

Fig. 8 shows the comparison of the three SPA protocols. As seen from the figure, with the increase in the number of terminal simulations, the performance of the SPA protocol designed in this paper is significantly better than the other two, and it can be seen from the analysis that the size of single packet authorization of OPENSPA reaches 1230 bytes and the response packet size is also 1230 bytes at maximum; the size of single packet authorization of FWKNOP is 225 bytes and the response packet size is also 225 bytes, but FWKNOP is more complicated to configure and requires the advance generation of access keys for devices, and OPENSPA and FWKNOP use standard RSA and SHA algorithms, which consume large computational resources and have shortcomings in convenience and security. The single packet authorization protocol designed in this paper is based on the SM9 protocol to realize encryption and signature, which effectively reduces the communication packet size under the premise of ensuring security, with a single packet authorization packet size of 186 bytes and a response packet size of 172 bytes.

Fig. 9 shows the comparative effect of the schemes for secure terminal access. With the increase in the number of terminal accesses, the scheme proposed in this paper greatly exceeds the scheme [27] in terms of performance and has certain advantages compared with the scheme [26]. In the scheme [27], the SM2 algorithm and digital certificate are used to complete authentication, whose communication consumes approximately 422 bytes of packets, and the gateway needs to communicate with the CA center to verify the legitimacy of the terminal digital certificate, which greatly increases the complexity of the system.

In contrast, scheme [26] uses the SM9 algorithm to achieve secure terminal access, which has greatly improved the performance and consumes only approximately 500 bytes of communication, but these two schemes do not distinguish between the control plane and data plane. The TCP port set used for authentication makes

the leakage surface of the service system larger. In contrast, the scheme designed in this paper separates access authentication from the TCP port. It requires approximately 600 bytes for a complete terminal access authentication, although the overall communication cost is larger than that of the scheme [26], the SPA accounts for 358 bytes, and the terminal and gateway authentication accounts for only 242 bytes, the gateway does not need to care about the SPA process and only needs to receive the authorization from the SDP controller. Therefore, authentication process is more concise and efficient than that in [26].

7 Conclusion

This paper designs and implements a security protection scheme based on the SDP architecture for the SDS. Our scheme builds identity for the terminal based on subject attribute, environment attribute and object attribute, and completes identity authentication based on SM9 algorithm, reducing the cost of certificate management. Our scheme realizes terminal access management and server maintenance through SDP controller, which ensures that unauthorized terminals cannot connect to the server or even know the ports of the server. The scheme effectively solves the security access problem after the deployment of SDS cloud, and provides new ideas and methods for security protection of the SDS.

The proposed scheme improves the existing infrastructure without changing the existing security protection framework of the SDS; The proposed scheme can cope well with the mainstream attacks of the SDS, including replay attacks, MITM attacks, DDOS attacks, and TCP SYN flooding attacks, and has a good performance in defense against internal threats and system vulnerabilities; Through comparative experiments, it is proved that our scheme is superior to other schemes in terms of security and computing cost, and can effectively and safely realize the protection of SDS.

However, SM9 algorithm is based on bilinear pair operation. The computation cost of the bilinear pairing is high compared to other cryptographic operations. Therefore, in future, we will continue to extend our protocol to provide more properties and further reduce computational cost and communicational cost.

8 Acknowledgement

This work is funded by the State Grid Headquarters Science and Technology Project (5400-202155408A-0-0-00) [Key technologies research and development for business security protection of distribution secondary system towards IoT]. Thanks to every member of the team for their contributions.

References

[1] T. Gonen, Electric Power Distribution Engineering, third ed., CRC Press, Boca Raton, 2015.
[2] National Energy Administration, Regulations on Safety Protection of Electric Power Monitoring System, Beijing, <https://zfxxgk.ndrc.gov.cn/web/iteminfo.jsp?id=18500>, 2014 (accessed 01.08.2014).
[3] L. Chen, Z. Dai, M. Chen, N. Li, Research on the Security Protection Framework of Power Mobile Internet Services Based on Zero Trust, in: Proc. 2021 6th International Conference on Smart Grid and Electrical Automation (ICSGEA), 2021.
[4] J. Wang, L. Wu, K.-K. R. Choo, D.-B. He, Blockchain-Based Anonymous Authentication With Key Management for Smart Grid Edge Computing Infrastructure, IEEE Transactions on Industrial Informatics 16(3)(2020) 1984-1992.
[5] A. Moubayed, A. Refaey, A. Shami, Software-defined perimeter (SDP): State of the art secure solution for modern networks, IEEE Network 33(5)(2019) 226-233.
[6] S.M. Kerner, Cloud Security Alliance Defends Cloud With Software Defined Perimeter, Eweek. <https://www.eweek.com/cloud/cloud-security-alliance-defends-cloud-with-software-defined-perimeter/>, 2013 (accessed 15.11.2013).
[7] S. Poudel, G.D. Black, E.G. Stephan, A.P. Reiman, Admittance Matrix Validation for Power Distribution System Models Using a Networked Equipment Model Framework, IEEE Access 10(2022) 9108-9123.
[8] C.-L. Sun, X.-M. Wen, Z.-M. Lu, W.-P. Jing, M. Zorzi, Eco-friendly powering and delay-aware task scheduling in geo-distributed edge-cloud system: a two-timescale framework, IEEE Access 8(2020) 96468-96486.
[9] A. Kerman, O. Borchert, S. Rose, E. Division, A. Tan, Implementing a Zero Trust Architecture, National Institute of Standards and Technology, 2020, 1-17.
[10] J. Garbis, J.-W. Chapman, Zero Trust Security: An Enterprise Guide, Apress, Berkeley, 2021.

[11] Google, Fundamentals of the BeyondCorp 'Zero-Trust' Security Framework. <https://dzone.com/articles/fundamentals-of-the-beyondcorp-zero-trust-security/>, 2017 (accessed 25.01.2017).
[12] C. Buck, C. Olenberger, A. Schweizer, F. Völter, T. Eymann, Never trust, always verify A multivocal literature review on current knowledge and research gaps of zero-trust, Computers & Security 110(2021) 102436.
[13] Gartner, QI Anxin Joint white paper on zero trust architecture and solutions. <http://www.qianxin.com/threat/ report-de-tail?report_id=98/>, 2020 (accessed 10.04.2020).
[14] A. Kerman, O. Borchert, S. Rose, E. Division, A. Tan, Implementing a Zero Trust Architecture Project Description Final, National Cybersecurity Center of Excellence. <https://www.nccoe.nist.gov/publications/project-description/implementing-zero-trust-architecture-project-description-final/>, 2020 (accessed 01.10.2020).
[15] M. Shore, S. Zeadally, A. Keshariya, Zero Trust: The What, How, Why, and When, Computer 54(11)(2021) 26-35.
[16] X.-J. Zhang, L.-D. Chen, J. Fan, X.-Q. Wang, Q. Wang, Power IoT security protection architecture based on zero trust framework, in: Proc. 2021 IEEE 5th International Conference on Cryptography, Security and Privacy (CSP), 2021.
[17] M. Abdul, Wahid, K.R.R. Kumar, Zero Trust User Access and Identity Security in Smart Grid Based SCADA Systems, in: Proc. 12th International Conference on Soft Computing and Pattern Recognition, 2020.
[18] A. Alagappan, S.K. Venkatachary, L.J.B. Andrews, Augmenting Zero Trust Network Architecture to enhance security in virtual power plants, Energy Reports 8(2022) 1309-1320.
[19] G.M. Gilbert, S. Naiman, H. Kimaro, N. Mvungi, A Cloud-Fog Based System Architecture for Enhancing Fault Detection in Electrical Secondary Distribution Network, in: Proc. of the International Conference on Computer Networks, Big Data and IoT (ICCBI - 2019), 2019.
[20] J. Singh, A. Refaey, A. Shami, Multilevel Security Framework for NFV Based on Software Defined Perimeter, IEEE Network 34(5)(2020) 114-119.
[21] R.-X. Qiu, Y. Fu, J. Le, F.-Y. Zheng, G. Qi, C. Peng, Y.-C. Li, A Software-Defined Security Framework for Power IoT Cloud-Edge Environment, International Journal of Network Security 24(6)(2022) 1031-1041.
[22] Y.-C. Palmo, S. Tanimoto, H. Sato, H. Sato, Optimal Federation Method for Embedding Internet of Things in Software-Defined Perimeter, IEEE Consumer Electronics Magazine 12(5)(2023) 68-75.
[23] J.-Y. Feng, T.-T. Yu, Z.-Y. Wang, W.-B. Zhang, G. Han, W.-H. Huang, An Edge Zero-Trust Model Against Compromised Terminals Threats in Power IoT Environments, Jisuanji Yanjiu yu Fazhan/Journal of Computer Research and Development 59(5)(2022) 1120-1132.
[24] fwknop, Single Packet Authorization. <https://github.com/mrash/fwknop>, 2020 (accessed 14.01.2023).
[25] OpenSPA, An open and extensible Single Packet Authorization (SPA) implementation of the OpenSPA Protocol. <https://github.com/greenstatic/openspa>, 2021 (accessed 25.01.2023).
[26] K.-H. Wu, R. Cheng, B.-H. Zheng, W.-C. Cui, Research on Security Communication Protocol of Power Internet of Things, Netinfo Security 21(9)(2021) 8-15.
[27] W. Li, R. Li, K.-H. Wu, R. Cheng, L.-P. Su, W.-C. Cui, Design and Implementation of an SM2-Based Security Authentication Scheme With the Key Agreement for Smart Grid Communications, IEEE Access 6(2018) 71194-71207.