Open Access Library Journal
2022, Volume 9, e9099
ISSN Online: 2333-9721
ISSN Print: 2333-9705

Discussion on "Zero-Trust" or "Evidence-Only" Architecture

Xianghao Nan

CPK Laboratory, Beijing, China
Email: nanxianghao@bochtec.com

How to cite this paper: Nan, X.H. (2022) Discussion on "Zero-Trust" or "Evidence-Only" Architecture. *Open Access Library Journal*, **9**: e9099.
https://doi.org/10.4236/oalib.1109099

Received: July 13, 2022
Accepted: August 22, 2022
Published: August 25, 2022

Copyright © 2022 by author(s) and Open Access Library Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).
http://creativecommons.org/licenses/by/4.0/

Open Access

Abstract

Recently, the U.S. Federal government and the Department of Defense announced the "Zero Trust" reigniting the debate on the logic of trust. Subject authentication is the core technology of cyber security. The traditional system is based on the reasoning logic of trust. The trust logic is the product of the situation that the authenticity of the subject cannot be proved, and the authenticity of the subject is remedied by a third party's certificates. However, the authenticity of certificates still cannot be proved, and can't be used as evidence after the fact, so a complete signature protocol cannot be constructed. Trust provides a basis for face to face transaction, but not as evidence afterwards. Trust-based reasoning logic adopts a decentralized key generation system, and the decentralized system has too strong exclusivity, which is easy to be used by criminal groups. Therefore, there is a new requirement to construct new authentication logic, which is "Evidence Only Architecture". The evidence based authentication logic proves the authenticity of the subject through the one-to-one mapping between the identifier and the key. However, we have to admit that it is difficult to establish such a mapping. As long as the mapping is established, the real digital signature can be constituted and can be used as evidence after the fact.

Subject Areas

Information and Communication: Security, Privacy, and Trust

Keywords

PKI, CPK, Authentication, Trust, Evidence, Logic, Cyber

1. Introduction

Subject authentication is the core technology of cyber security, but it has been a

difficult problem. Recently, the "zero trust architecture" project [1] put forward by the U.S. DoD and the federal government has once again put the issue of trust on the crest of a wave. In 2005, PITAC proposed "mutual suspicion" as a security principle for the first time in its report on "Cyber Security" [2]. This is an epoch-making conclusion. This was a watershed in the development of authentication theory, but after 20 years, it still remains in the age of trust logic and has not made any progress. Both the U.S. DoD and the federal government recently proposed a "zero-trust architecture" [3], saying, "Never trust, always verify". In the proof of subject authenticity, it is right to take zero trust as the starting point, and it is also right to always verify, but it does not indicate what to verify. In this "zero trust architecture", the concept of "identifier" is put forward for the first time to distinguish identifier from identity, which is the only way to the subject authentication, although the authenticity of identifier has not been solved yet.

There are two kinds of technologies subject authentication: one is the PKI certification system based on trust, and another is the CPK authentication system based on evidence. Through these two systems, the difference between trust mechanism and zero-trust mechanism in the authentication system is studied, and the discussion is further deepened, so that our theoretical research on cyber security is on the right track. In fact, the proposal of "mutual suspicion" and "zero trust" has sounded the end of the era of trust logic, while the rise of evidence-based authentication logic is lighting the fire of the development of new logic.

2. Management Mode

2.1. PKI Decentralized Mode

The system in which the key pair is generated by individual is called decentralized system. With the emergence of the Internet, the concept of decentralized system brought about a new problem under the new situation of public network, which has broken the boundary of private network. In the past, private networks were used by the military and related government departments, but now they are used by all Internet users. At this time, asymmetric public key system appeared, which can realize the closure of arbitrary communicating two sides in public network. The National Security Agency (NSA) realized that its plans to migrate classified managing method of closed LAN security to the open Internet would not work and had to explore a new way. The reason is very simple, the technology has developed to close the two communicating sides, and there is no need for classified closure. The policy of civil-military integration adopted later was a major change brought about by the emergence of new technologies. In such environment, decentralized key management is put forward by PGP, the key is generated by individuals, and the public key is published. PKI distributes public keys on the basis of decentralized PGP in the form of certificates in which the public-key and identifier is bound. The Certificate form was originated on the network of the U.S. Department of Defense. The key was generated by the Key Management Center and distributed in the form of certificates, which was called

the Certificate Agency (CA). PKI borrowed the form of the certificate, and called it Certificate Authentication (CA) at first, but is changed to the Certificate Authority (CA).

The system of generating private-keys by individuals is excessively exclusive, and it is also exclusive to regulators. Such exclusivity, if used by the underworld and drug cartels, will cause great difficulties in solving the case, which is obviously detrimental to safeguarding national interests. Now that the CA is effectively central, what exactly is the benefit of generating private-keys by individuals? Some people say that CA crime can be prevented, but in fact, the main security threat is not CA crime, but is the criminal's crime using the binding function of CA to commit fake certificate.

2.2. CPK Centralized Mode

CPK centralized mode is a traditional key management mode. Whether it can meet the needs of large-scale, individual and open network is the only criterion to measure the rationality of key management. Centralization and decentralization are just different management methods, without principle differences. PKI can also be centrally managed. China Customs introduced PKI, and creatively adopted central key distribution according to the needs of its own business. The practice proves that it is feasible to transform PKI into a centralized system, and it should be the user's right to select what kind of mechanism. But in China, because the centralized system for distributing keys did not comply with China's digital signature law, the competent department refused to approve it. A specific technical method is determined in legal form, reflecting the backwardness and confusion in the regulations and management of information security in China. European electronic signature law stipulates that as long as it is approved by both parties, the signature has legal effect. It's straightforward and consistent with the law of contract respecting users' rights.

CPK implements centralized mode, because of the solution to large scaled key management, one step of horizontal management to the whole network can be achieved with a few KB matrix space which can represent infinite public-keys, and the public matrix is published, so that anyone can calculate anyone's public-key, ensuring the personalized needs allowing to be supervised. More importantly, CPK is a public key system that establishes one-to-one mapping between identifier and key, which is recognized as the core technology, because only one-to-one mapping can solve the authenticity of identifier and ontology, and then prove the authenticity of the subject. This kind of system solves a pair of contradictions between decentralized application and centralized management, which can ensure the information security system to run more effectively.

3. Authentication Logic

3.1. PKI Trust Logic

PKI certification system is the product of trust logic, and the proof by third party

is a last resort in the case that one cannot prove his own authenticity. In the book of IATF (Information Assurance Technical Framework), the trust transfer of PKI is described as follows: If a trust relationship is established between two CAs, the employees of the two CAs have the same trust relationship. Schnaier has said that if this logic holds, it could lead to the joke that UCLA graduates can go to MIT to get their diplomas. In fact, the international standard CC described that trust transfer causes trust dilution, so the transfer should not be more than four times. The initial PKI attempts to increase the number of CA by trust transfer to solve the problem of large-scale key management. Obviously, IATF's understanding is wrong.

We have to see that the current key management mechanism, whether centralized KDC or decentralized CA mechanism, still follows the principle of trust, and the existing trust logic based on behavior and belief logic based on model reasoning are still not free from the bondage of trust relationship. Trust as a sociological term, plays an important role, but the authentication system is a proof system, proofs needs evidence, not trust, proof has nothing to do with trust. There is no need for any additional provisions in the proof, because artificial provisions fall under the category of trust. Trust is a basis for transaction, but it can not be evidence afterwards. Therefore, it only applies to situations where evidence is no longer required afterwards. According to the experience of U.S. military cyber warfare [3], the most effective means of network attack is to obtain login through password and take over system rights by trust transfer, thus trust transfer has become a hidden danger of security.

3.2. CPK Truth Logic

CPK authentication system executes truth logic. In truth logic, identifier and identity have long been distinguished, and identity is defined as the unity of identifier and ontology. The authenticity of identity can be solved only when the authenticity of identifier is solved. In 2005, PITAC declared in its report on "Cyber Security" that "Cyber security is so complicated, there is no silver bullet". However, in Chinese folk QNS studio in 2006, put forward that identity is composed of identifier and ontology, subject authentication can only be solved by identifier authentication, and thus constructed one to one mapping between identifier and keys, and the authenticity of identifier is achieved by key paring. The authenticity of identifier further proves the authenticity of subject [4]. CPK found the "silver bullet" and solved the subject authenticity through the identifier authentication. As long as the authenticity of the subject is solved, other security proofs become as simple as stacking wood, so the authentication of the subject becomes the core technology of cyber security.

Truth logic is an authentication logic based on evidence, consists of evidence-showing system and evidence-verifying system, in which what evidence is shown, what evidence is verified. Without evidence there is nothing to be verified. From the perspective of theoretical research, authentication logic only stays

on trust logic, the theory is unable to move forward. The establishment of trust relationship is not the ultimate goal to be achieved by the authentication system, but is to prove the authenticity of the subject to achieve information assurance, because the truth logic solved the problem of the authenticity of the identifier claimed by the subject. The key management center can prove the scope and authenticity of the used key, hence the key distribution breaks away from trust logic and opens up a new key distribution method based on proof relation. In terms of CPK system, the relation between users and the KDC is a proof relation, where the authenticity of the center can be proved, and the authenticity of public matrix can also be proved, Independent of trust, the proof system becomes more and more objective.

4. Authentication Methods

4.1. Certification Method of PKI

The certification method of PKI is carried out by using digital signature standard DSS [5], and DSS is a mathematical formula to prove whether the public and private keys are paired. The proof is to use the private-key and random number to calculate a check code c and proof code s, and its verification is to use the public-key and proof code s to calculate the check code c'. If $c = c'$, the pair of the public and private keys is proved. A trust relationship can be established between the prover and the verifier, but it is not yet a signature, because only the pair of public and private keys is proved, and the subject authenticity proof was not given yet. Therefore, DSS itself does not have the function of signature. In order to enable DSS to have digital signature function, the CA center of PKI binds the identity and public-key in the certificate, so that DSS signature and CA certificate are combined to form digital signature, but this can only be true under the assumption that the CA has been verified to be true. However, CA's authenticity cannot be proved before the problem of subject's authenticity proof is solved, so it has to go back to the trust logic and cannot achieve "zero trust" or "never trust".

In addition, a digital signature should have the same lifetime as the signed document, the key cannot be replaced during the validity period of the document. However, in the individualized decentralized system, the key can be changed, so the validity of the certificate is needed to be verified.

4.2. The Authentication Method of CPK

The authentication method of CPK is carried out under CPK public key system. CPK is realized by elliptic curve ECC. The curve is defined with $y^2 = x^3 + ax + b$ and parameter T = (a, b, G, p, n), where G is the generator, p is the module, and n is the order. In CPK, there is a one-to-one mapping between the subject's identifier and the key. So the authenticity proof of identifier is simple, first use random number k to generate check code c: $k\mathrm{G} = (x, y) \to c$, then use random number k and private-key sk to generate proof code s: $k^{-1}sk \bmod n = s$, its veri-

fication is to use proof code s and public-key PK to calculate the check code c': $s^{-1}PK = k\mathrm{G} \to c'$. If $c = c'$, it is proved that sk and PK are a key-pair, thus the authenticity of the key is proved. The key is directly generated by the identifier and a one-to-one mapping is formed between the idetifier and the key. Therefore, the authenticity of the key directly proves the authenticity of the identifier. After the authenticity of the identifier is proved, the authenticity proof of the subject, slave and object can be realized by the combination principle of elliptic curve. For example, the authenticity of the subject is proved by the sum of the private-keys of identifier and ontology, and verified by the sum of the public-keys of identifier and ontology. The slave authenticity is proved by the sum of the private-keys of identifier and the slave, and verified by the sum of the public-keys of identifier and the slave. The object authenticity is proved by the sum of the public-keys of identifier and the object, and verified by the sum of the public-keys of identifier and the object. The compound authentication proves simultaneously the authenticity of the subject, slave and object, and provides proof of traceability, attribution and responsibility. The public key matrix of CPK is published with the signature of key management center, so the authenticity of the subject (key management center) and the the public matrix (object) can be verified by everyone, the scope and the proof relationship are clear, which is the biggest difference from CA.

5. Authentication Range

The following takes network communication as an example to compare the authentication range of PKI and CPK. Communication events are divided into two events, sending events occur at the sending end, receiving events occur at the receiving end, sending events and receiving events constitute a virtual internet of event (IoE). In the IoE, it is up to the sender to send information, including malware like viruses, at will. But the actual control is in the hands of the receiving side, which has the right to decide whether to accept or reject, and whether to process or not. As the authentication system is the unification of the proof system and the verification system, the proof of authenticity should be provided in the sending event to ensure that the verification can be passed in the receiving event.

5.1. CPK Communication Event

The sender provides the authenticity evidence of the subject, slave, and object. For example, Alfa sends data X to Beta, then Alfa is the subject, Beta is the slave, and data is the object.

There are two cases of sending event. One is the case where the receiver needs to separate “proof before event” and “proof after event”, such as online communication with a large volume of business; The second is the case that does not need to be handled separately, such as offline communication like E-mail. Among them, proof before event is carried out before data transmission, while

proof after event is carried out after data transmission. Proof of before event and proof after event are independent of each other.

5.1.1. CPK Sending Event

Evidence for proof before event includes proofs of the authenticity of the sender Alfa, receiver Beta, data X is provided separately. The object authenticity proof can be given separately. Evidence for proof before event, there are three types:

The first type is identifier authentication. Identifier authentication is a signature to identifier by key. It is called "identifier signature". Identifier includes static and dynamic. Static identifier authenticity code SIC is to replace the traditional "password certification", but does not prevent replication attacks:

$$k_1\mathrm{G} = (x_1, y_1); (x_1 + y_1)^2 \bmod 2^{16} = c_1 \tag{1}$$

$$\left(k_1^{-1} sk_{\mathrm{Alfa}}\right) \bmod n = s_1$$

$$\mathrm{SIC} = (s_1, c_1)$$

where, k is a random number, G is the generator, sk_{Alfa} is the private key of $\mathrm{IP}_{\mathrm{Alfa}}$, c is the check code, s is the signature code, and (s, c) constitutes the signature.

The dynamic identifier authenticity code DIC is to replace the traditional dynamic password. The private-key is added to time to prevent copy and DOS attacks:

$$k_2\mathrm{G} = (x_2, y_2); (x_2 + y_2)^2 \bmod 2^{16} = c_2 \tag{2}$$

$$\left(k_2^{-1} \left(sk_{\mathrm{Alfa}} + \mathrm{time}\right)\right) \bmod n = s_2$$

$$\mathrm{DIC} = (s_2, c_2)$$

The second type is ontology authentication, marked with Ont. Ontology authentication is signature to ontology by identifier. It is called "ontology signature" or "identity authentication"

$$k_3\mathrm{G} = (x_3, y_3); (x_3 + y_3)^2 \bmod 2^{16} = c_3 \tag{3}$$

$$k_3^{-1} \left(\mathrm{ontology}_{\mathrm{alfa}} + sk_{\mathrm{Alfa}}\right) \bmod n = s_3$$

$$\mathrm{Ont} = (s_3, c_3)$$

The third type is slave or object authentication, marked with Obj: slave and object authentication is the signature of the subject to the slave or object. Among them, slave authentication can be carried out before data transmission, therefore, identifier, ontology and slave authentication are called "proof before event", and object authentication is called "proof after event".

$$k_4\mathrm{G} = (x_4, y_4); (x_4 + y_4)^2 \bmod 2^{16} = c_4 \tag{4}$$

$$k_4^{-1} \left(\mathrm{data} + sk_{\mathrm{Alfa}}\right) \bmod n = s_4$$

$$\mathrm{Obj} = (s_4, c_4)$$

A signature simultaneously certifies the authenticity of subject, slave and object.

The sender can encrypt data using CPK key encryption protocol. First computes the public-key PK_{Beta} of the receiver (IP_{Beta}):

$$\text{Hash}\left(\text{IP}_{\text{Beta}}\right) = v_i; \sigma \sum \text{R}_{vi} = PK_{\text{Beta}}$$

Encrypts the data encryption key with the other party's public key PK_{Beta}

$$k\text{G} \to \text{key}; \text{E}_{\text{key}}\left(\text{data}\right) = \text{code}; k * PK_{\text{Beta}} = \lambda$$

Sends (code, λ) to Beta.

5.1.2. CPK Receiving Event

The receiving event mainly verifies the sender's evidence, including IP_{Alfa}, IP_{Beta} and data authenticity. The verification Implements CPK protocol and GAP one-step protocol [6].

When verifying, first calculates the signer's public-key PK_{Alfa} with the identifier:

$$\text{Hash}\left(\text{IP}_{\text{Alfa}}\right) = v_i, \sum \text{R}_{[v_i]} \to PK_{\text{Alfa}}$$

The verification before event is as follows:

The first type: to verify static identifier authenticity code, directly proves the authenticity of the subject:

$$s_1^{-1} PK_{\text{Alfa}} = k_1\text{G} \to c_1'$$

The second type: to verify dynamic identifier authenticity code: directly proves the authenticity of the subject:

$$s_2^{-1}\left(PK_{\text{Alfa}} + \text{timeG}\right) = k_2\text{G} \to c_2'$$

The third type: to verify the authenticity of the slave: to prove the authenticity of the subject and the slave simultaniously;

$$s_3^{-1}\left(\text{IP}_{\text{Beta}}\text{G} + PK_{\text{Alfa}}\right) = k_3\text{G} \to c_3'$$

The verification after event is carried out after the data is received, to prove the authenticity of the subject and objectsimultaniously;

$$s_4^{-1}\left(\text{dataG} + PK_{\text{Alfa}}\right) = k_4\text{G} \to c_4'$$

If the data is encrypted, first decrypts data before authentication. Beta uses its own private-key to decrypt the data encryption key:

$$sk_{\text{Beta}}^{-1} * \lambda = \text{key}$$

Decrypt data with data encryption key:

$$\text{D}_{\text{key}}\left(\text{code}\right) = \text{data}$$

5.2. PKI Communication Event

5.2.1. PKI Sending Event

Evidence for proof before event: Traditional symmetric password.

Evidence for proof after event: executes the DSS signature protocol.

$$k_1\text{G} = \left(x_0, y_0\right); x_0 \bmod n \to c_1$$

$$k_1^{-1}(\text{data} + c * sk) \bmod n = s_1$$

The authenticity proof of public-key *PK* bounded to identity by certificate, such as:

$$\text{Hash}(\text{IP}_{\text{alfa}} + PK) = h$$

$$k_2^{-1}(h + sk_{\text{CA}}) \bmod n = s_2;$$

$\text{sign}_1 = (s_1, c_1)$ and $\text{sign}_2 = (s_2, c_2)$ are combined to form a complete signature. But the authenticity of certificate has not been provided

When encrypting, PKI first ask for the other party's public-key certificate, after verification of the certificate, the data encrypting key can be encrypted with the public-key.

5.2.2. PKI Receiving Event

Verification before event: passwords can be compared but do not prove the authenticity of the subject.

Verification after event: implements DSS protocol and SSL protocol with 6-steps 13-sentences.

First use the public key *PK* provided by the sender's certificate to verify the signature to the data:

$$s_1^{-1}(\text{data} * \text{G} + c_1 * PK) \to c_1'$$

If $c_1 = c_1'$, it is proved that the private-key *sk* used for signature and the public-key *PK* used for verification are a pair of keys, so the data is true. But there's no proof of whose signature. Since the sender's identity and public-key are bound by the certificate, the certificate is also verified:

$$\text{Hash}(\text{IP}_{\text{Alfa}} + PK) = h$$

$$s_2^{-1}(h * \text{G} + c_2 * PK_{\text{CA}}) \to c_2';$$

If $c_2 = c_2'$, it proves that this is the signature of the sender Alfa, but the authenticity of CA has not been proved.

6. Function and Performance

The Function comparison between PKI and CPK of above example is summarized in the following table.

Function	Matrix	Proofs Before Event	Key Encryption	Signature	Verify	Subject Evidence	Verify
PKI	CA	No	1. Ask for certificate 2. Verify certificate, 2*n*G 3. Key encryption, 2*n*G	224B + identifier	4*n*G	4nB	2nG
CPK	8 × 8	Yes	Key encryption, 2*n*G	1*n* + 4B = 36B	2*n*G	1nB	1nG
	4 × 4	Yes	Key encryption, 2*n*G	10B..20B	2*n*G	1nB	1nG
Compared		No: Yes	3:1	7:1	4:2	4: 1	2:1

Note: n * G in the table represents an elliptic curve operation.

The performance comparison between zero trust and evidence only architectures is simply summarized in the following table.

Performance	Zero Trust Arcitectire	Evidence Only Architecture
Security Principle	Mutual Suspicion → Zero Trust	Evidence sowing and verification
Authentication Logic	BAN Logic	Truth Logic
Authentication protocol	SSL (13 steps)	GAP (one step)
Authentication Method	Trust Transfer	A thing a proof
Authentication Object	To recognize foe	To recognize friend
Network Application	Back to Civil-military separation	Can realize Civil-military integration
Identifier Formation	Identifier was formed in 2021	Identifier was formed and solved in 2006
Private-key generation	Decentralized Generation, no supervision	Centralized generation, allowing supervision
Public-key generation	Generated from private-key	Computed by a Public matrix
Identifier Authentication	X	Identifier is authenticated by key
Subject Authentication	Strong Password + easy CA Certificate	Subject is proven by identifier
Dynamic Password	None	Authenticated Identifier with time
Identity Authentication	None	Ontology is proven by identifier
Slave Authentication	None	Slave is authenticated by Subject
Object Authentication	Proved by CA Certificate	Object is proved by subject
DSS Signature Standard	Only establishes trust relation	Identifier is Mapped into key
Digital Seal	X	For individual or organization
Recognizer		Friend or foe identification, Anti-counterfeit label
Access Authentication	None	Subject verification
Adopt Authentication	By certificate	Object verification
Data Encryption	DES: fixed block encryption	BLK: dynamic block encryption
Key Encryption	X	Computes other party's public key to encrypt
Domain of keys	Unable to define	Defined clearly
Software trade mark	X	√
Software 1st class authorization	Single authorization trusted computing	Software Authorized by manufacturer
Software 2nd class authorization	None	Software Authorized by clients
Software 3rd class authorization	None	Software Authorized by individual
Digital currency issuance	Issued by central bank	Opened by account
Currency template	None	Issued by commercial bank
Authorization Letter	None	Authorization letter of Central bank
Currency flow	None	Payer and payee are indicated
Currency attribution	None	No vaults or purses needed
Crime of duplication	Difficult to find	Easy to find
Function	For payment	For payment and settlement

6. Summary

The difference is mainly reflected in whether the authenticity of the subject can be proved and whether the function of "proof before event" can be realized, and these are the most critical elements to achieve the goal of information assurance. For example, in communication, the authenticity of the subject can be verified before data transmission, and in transaction, the authenticity of currency can be verified before payment. PKI uses certificates to recover its subject authentication function, but the authenticity of CA still relies on trust relationship and the use of certificates increases the burden of certificate verification and increases the amount of information, causing a big performance difference. Especially when the system is extended, the relationship between different CAs can only be trusted.

In the field of communication and areas of the economy, 5G networks, satellite networks, remote control, sensor networks, and digital economy booming, the demand for the subject authentication is becoming more and more urgent. In this case, an in-depth discussion of zero-trust architecture or evidence-only architecture is necessary.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] Department of Defense (2021) Zero Trust Reference Architecture, Version 1.0.

[2] President's Information Technology Advisory Committee (2005) Cyber Security: A Crisis of Prioritization.

[3] CRS Report for Congress (2004) Clay Wilson, Information Warfare and Cyber War: Capabilities and Related Policy Issues.

[4] Nan, X.H. (2006) CPK on Identifier Authentication. Publishing House of Defense Industry, Beijing.

[5] National Institute of Standards and Technology, INST PUB 186, Digital Signature Standards, U.S. Department of Commerce 1994.

[6] Nan, X.H. (2020) GAP One-Step Protocol, Communication Technology. *Communication Technology*, **53**, 3030-3033.