Using a Variation of Elgamal Signature to Support Fast and Lazy Authenticating Origin Autonomous Systems

Bo Yang

1. School of Information Technology, Jiangxi University of Finance and Economics, Nanchang 330013, China
2. High Level Engineering Research Center of Electronic-Commerce, Jiangxi Provincial Colleges and Universities, Nanchang 330013, China
Email: B.Yang@acm.org

Abstract—About the schemes for authenticating origin autonomous systems, related documents think it is appropriate to authenticate address attestations and related public keys only using the information of update messages. However, this approach is infeasible in existing schemes because update messages are limited in length to 4096 bytes and thus are too small to carry the necessary public key certificates. For realizing this approach, in this paper, we make full use of specific characteristics of a variation of ElGamal signature and thus present a method for fast and lazy authenticating origin ASs. In the process of hierarchical issuing keys of proposed method, an efficient formula of computing public keys is achieved. For authenticating public keys, it is not necessary for a verifier to check a series of certificates existing in certification path from the owner of advertised prefix to the IANA (Internet Assigned Numbers Authority). Thus the lazy authentication of origin ASs is achieved because a verifier can authenticate address attestations and related public keys only using the information of update messages. The path validation from IANA to the owner of advertised prefix, where signatures can only be verified separately in existing asymmetric cryptography based solutions of IP prefix hijacking, can be replaced with efficient computing of only a single formula, which can bring about fast authentication of origin ASs. To the best of our knowledge, it is the first work for authenticating origin autonomous systems by using ElGamal signature to achieve a fast scheme that can authenticate address attestations and related public keys only using the information of update messages.

Index Terms—Origin Autonomous Systems; Asymmetric Cryptography Based Authenticating; Fast and Lazy; ElGamal Signature

I. INTRODUCTION

The Internet is a network of networks because it consists of about 43,000 Autonomous Systems (ASs) [1]. Border Gateway Protocol (BGP) is the de-facto standard routing protocol for exchanging and maintaining routing information among these ASs. To some extent, BGP can be regarded as an adhesive of the Internet. However, BGP can not authenticate origin ASs when update messages broadcast among ASs [2] [3] [27]. By advertising IP prefixes which are not assigned or belong to other ASs, malicious or misconfigured ASs will be mistakenly assumed to be origin ASs of these prefixes, and thus packets which aim at these IP prefixes are redirected, to these wrong origin ASs. This kind of attack is called IP prefix hijacking [2].

There are two categories of IP prefix hijacking, which is illustrated in Figure 1 [13].

The first category is the full prefix hijack, where the same prefix already announced by the victim. Other ASs will select one such route to adopt. In this case, the Internet is partially polluted. For example, in the left part of Figure 1, both the attacker and the victim announce the same prefix 10.1.0.0/16. Consequently, AS2 and AS3 may prefer the attacker’s route because of the shorter AS path [13].

The second category is the Sub-prefix hijack, where the attacker announces only a subnet of a prefix announced by the victim and this subnet is not announced previously. Unless filtered, this new sub-prefix is injected into the forwarding table regardless of the route of the existing prefix. Due to the longest prefix matching policy, traffic destined to this subnet follows the false route of the sub-prefix. Therefore, most of the Internet is likely polluted. In the right part of Figure 1, the attacker announces the sub-prefix 10.1.1.0/24. This route is likely accepted by all ASs as a new forwarding table entry. Hence, traffic destined to IPs such as 10.1.1.1 is misled to the attacker [13].

IP prefix hijacking is a real and serious threat, which is showed by many incidents. For instance, in February 2008 the misoperation of Pakistan Telecom (AS17557) resulted in the hijacking of YouTube traffic on global scale [5]. Some of earlier incidents of that kind are Cogent (AS174) and a major disruption of the Internet on April 1997 caused by AS7007 [3]. One of recent incident is the interception induced by China’s misconfiguration [6] [26], and so forth.

Recently, the initial deployment of the Resource Public Key Infrastructure (RPKI) [8], which is the necessary component of asymmetric cryptography based solutions, is being taken up by the IETF Secure Inter-Domain Routing (SIDR) working group [4] [7] [8].
There is much concern around the topic of IP prefix hijacking [3] [7] [24] [26]. Many solutions, mainly including the cryptography-based [9-14] and the detection-based [2] [15-18], have been proposed. The security of asymmetric cryptography based solutions, especially the S-BGP [9], has received wide acceptance.

Typical asymmetric cryptography based solutions of IP prefix hijacking are S-BGP [9], SoBGP [10], psBGP [11], and OA [12], most of whose computation burdens of authenticating origin ASs are too heavy [20]. No existing asymmetric cryptography based work can authenticate address attestations and related public keys only using the information of update messages. The proposed method can to some extent overcome the two shortages inherent in these existing asymmetric cryptography based solutions.

About the schemes for authenticating origin autonomous systems, related documents [9] think it is appropriate to authenticate address attestations and related public keys only using the information of update messages. This would ensure that each receiving BGP speaker would receive all the data needed to validate the attestations in an update message, and it would be easy for each BGP speaker to include its own certificate as part of the forwarding process. However, this approach is infeasible in existing schemes because update messages are limited in length to 4096 bytes and thus are too small to carry the necessary public key certificates.

For realizing this approach, in this paper, we present a method based on a variation [21] of ElGamal signature to authenticate origin ASs. By using the proposed method, autonomous systems can fastly and lazily authenticate origin ASs when update messages broadcast among ASs. The “fast authenticating” is achieved by an efficient formula for computing public keys, whose computing can substitute for separate verifications of path validation from IANA (Internet Assigned Numbers Authority) to the owner of advertised prefix. The “lazy authenticating” is feasible because almost all the information needed to authenticate origin ASs can be received in update messages, and thus the searching and downloading address attestations and public keys are not necessary. To the best of our knowledge, it is the first work for authenticating origin autonomous systems by using ELGamal signature to achieve a fast scheme that can authenticate address attestations and related public keys only using the information of update messages.

II. PREREQUISITE KNOWLEDGE

The prerequisite knowledge introduced in this section includes the authenticating framework in existing asymmetric cryptography solutions of IP prefix hijacking, and the variation of Elgamal signature used in proposed method.

A. The Authenticating Framework in Existing Asymmetric Cryptography Based Solutions of IP Prefix Hijacking

In existing methods for preventing IP prefix hijacking, asymmetric cryptography based solutions have a great effect, some typical schemes of which are S-BGP [9], SoBGP [10], psBGP [11], OA [12], HCBGP [13], and so on. The initial deployment of the Resource Public Key Infrastructure (RPKI) [8] taken up by the IETF SIDR working group [7] shows this kind of methods, especially the S-BGP, is to some extent accepted [26]. To a great extent, RPKI can be regarded as the reiterating and refining for the part of authenticating origin ASs of S-BGP. Thus, the authenticating framework described in this section is mainly about the S-BGP.

In asymmetric cryptography based solutions of IP prefix hijacking, address attestations are important elements. An address attestation is a signature signed by the advertised prefix owner’s private key, by choosing an origin AS (AS0) and signing the AS0 # (AS0 #, the number of AS0). A certified address attestation can prevent malicious attackers from hijacking advertised prefix by securing the binding between the advertised prefix and AS0.

The process of certifying an address attestation can be divided into two necessary parts. The first part is that verifying the address attestation by using the advertised prefix owner’s public key. The second part is that authenticating the owner’s public key. There is a path along which the owner’s public/private key pair is issued from IANA to the owner of advertised prefix. The second part, which is the public key authentication, is always the path validation from IANA to the owner of advertised prefix.

In existing asymmetric cryptography based methods for authenticating origin ASs, creation and distribution of address attestations and related certificates is not described explicitly [9-11]. For convenient reading, in this subsection, we describe it as following.

In Figure 2, public key certificates are used to bind a public key to an organization and to a set of prefixes. In a certificate, the subject is the DNS name of the organization, and the set of prefixes owned by it are put into a private extension of certificate. These certificates are arranged into a rooted tree parallelly the existing IP address allocation. The IANA is the root. RIRs (Regional Internet Registries) are the nodes of the second tier. Every node in this tree is ISP, DSP or an organization. One of the most important fields of a node’s certificate is the signature of DNS name of current node signed by the private key of parent node.

![Two categories of IP prefix hijacking](image)
Figure 2. Creation and distribution of address attestations and related certificates in existing asymmetric cryptography based solutions of IP prefix hijacking

For a prefix, its address attestation and related certificates existing in certification path from signers to IANA are generated independently by different parties. Every signature in address attestation and related certificates have to be verified respectively and independently. In the point of a verifier, verifying these independent signatures is a burden.

Based on Figure 2, the authenticating framework can be divided into four parts [9-11], which are issuing public/private key pairs, creating address attestations, distribution of address attestations and related certificates, verifying address attestations and related public keys (see Figure 3).

Issuing public/private key pairs: The owner of the prefix (note that unless otherwise noted, all of the words “prefix”, which appear in the rest of this paper, denote the advertised prefixes in update messages), is issued public/private key pairs by its key issuers. Note that the issuing process is hierarchical. The issuing chain is from IANA to the owner of advertised prefix, which is showed in figure 2. Every public/private key pair is corresponding to a public key certificate. Thus, a series of certificates will be brought about by this issuing process. There are 3~7 certificates in a single issuing chain which is from IANA to the owner of advertised prefix [12]. A public key certificate is always an X.509 certificate. The size of X.509 certificate is about 500~1000 bytes long [9].

Creating address attestations: The owner of advertised prefixes creates an address attestation for prefix, which is a signature signed by the owner’s private key, by choosing an origin AS (AS0) and signing the AS0 #.

Distribution of address attestations and related certificates: Address attestations and related certificates are mainly distributed to all of ASs out of band rather than in update messages. Every owner of advertised prefix uploads address attestations and related certificates to the repositories. In most cases, all address attestations and related certificates should be downloaded from the repositories to all of ASs before the prefix is advertised in update messages (see figure 2). Related documents think it is appropriate to authenticate address attestations and related public keys only using the information of update messages. This would ensure that each receiving BGP speaker would receive all the data needed to validate the attestations in an update message, and it would be easy for each BGP speaker to include its own certificate as part of the forwarding process. However, this approach is infeasible because update messages are limited in length to 4096 bytes and thus are too small to carry the necessary public key certificates (an X.509 certificate is about 500~1000 bytes long) [9-11].

Verifying address attestations and related public keys: All of ASs can verify the address attestation using the public key of owner of prefix i. The public key is gotten from the corresponding certificate, and should be also authenticated before using. Under most conditions, the distribution and verification of address attestations are not in step with broadcast of update messages.

Algorithm 0: The variation of Elgamal signature [21] used in proposed method

Key generation
1. Generate a large random prime p and a generator g of the multiplicative group $Z_p$, where p and g are all 128 bytes. $Z_p$ is the set of positive integers which are less than the p.
2. Select a prime q, which is 160 bits long and $q \equiv 1 \pmod{p}$. The q is a divisor of (p-1).
3. Select a random integer $s_0$, $0 < s_0 < q$
4. Compute $pk = g^{s_0} \pmod{p}$
5. The public key is (p, g, pk); the private key is $s_0$

Signature generation
1. Select a random secret integer k, $0 < k < q$
2. Compute $r = g^k \pmod{p}$
3. Compute $s = (h(m) + r \cdot s_0) \pmod{q}$ ($m$ is the message which is signed)
4. The signature for m is the pair (r, s).

Verification
1. Verify that $0 < r < p$ and $0 < s < q$; if not, then reject the signature.
2. Accept the signature if and only if $g^s = r^{k \cdot s_0} \cdot pk^{r \cdot s_0} \pmod{p}$
B. The Variation of Elgamal Signature used in Proposed Method

The theory lays cryptographic foundation of our method is ELGamal signature, which is a digital signature scheme based on the difficulty of computing discrete logarithms. The ELGamal signature requires a hash function h: \{0, 1\}^* \rightarrow \mathbb{Z}_p, where p is a large prime number. The signing equation of the ELGamal is that s = k^m \cdot (h(m) \cdot s_0 \cdot r) \mod p where m denotes the signed message, s denotes the signature. A different k must be selected for each message signed; otherwise, the private key can be determined with high probability.

There are several variations of ELGamal signature [21], one of which is used by us to design the proposed method. Most of these variations alter signing equations. The variation used by us in this paper alters signing equation into \(s = (k \cdot h(m) + r \cdot s_0) \mod q\), where q is a divisor of (p-1). The q is 160 bits long and \(1 = g^s \mod p\).

It requires a hash function h: \{0, 1\}^* \rightarrow \mathbb{Z}_q

The variation of Elgamal signature used in proposed method, can be seen in Algorithm 0.

III. THE PROPOSED METHOD

We now present our method. This method can be divided into several parts. These parts are authenticating framework, issuing public/private key pairs, creating address attestations, distribution of address attestations and the public-key-related information, and verifying address attestations and related public keys.

A. The Authenticating Framework

Figure 4 is about the creation and distribution of address attestations and public-key-related information in proposed method. Comparing Figure 4 with Figure 2, there are two main differences. The first difference is that in Fig. 4, repositories are eliminated, which is owing to address attestations and public-key-related information are all distributed within update messages. The second difference is that in Figure 4, there is no public key certificate. Public key certificates in Figure 2 are replaced with the public-key-related information in Figure 4. As public key certificates, the public-key-related information is also used to authenticate the public keys.

Above framework which is described in Figure 2 is followed by us except two modifications. The two modifications are as follows.

Firstly, all of address attestations are distributed in update messages rather than out of band. For advertising the prefix, an AS adds current address attestation into update messages and sends these update messages to its neighbors.

Secondly, all of address attestations are also verified in step with broadcast of update messages. Every AS in AS-PATH verifies current address attestation by using the public key corresponding prefix, when receiving the update messages about the advertised prefix. Therefore, in the proposed method, the distribution and verification of address attestations are all in step with broadcast of update messages.

Next, we describe the proposed method in detail according to above authenticating framework. The proposed method can be divided into following parts that are issuing public/private key pairs, creating address attestations, distribution of address attestations and the public-key-related information, and verifying address attestations and related public keys.

B. Issuing Public/Private Key Pairs

In the proposed method, a prefix owner has to get credible keys from key issuers so as to create address attestations. IANA, which is regarded as trust root in almost all of existing asymmetric key based solutions, acts as the first tier prefix owner who initially owns all possible prefixes in the key issuing course. The IANA issues some large prefixes to RIRs (Regional Internet Registries) and make them owners of these prefixes. These issued prefixes are not owned by IANA any more.

In the second tier, RIRs get credible keys from IANA as follows. For issuing keys to these prefix owners, the IANA signs these prefix#s (numbers of prefixes) using the variation of ELGamal signature. The signatures are sent to RIRs and act as their private keys whose corresponding public keys can be computed by some formulas.

Similarly, in the third tier, prefix owners are issued their prefixes and corresponding public/private key pairs by a RIR in the second tier. Note that these issued prefixes should be contained by the prefixes of the RIR, and the corresponding private keys should be the signatures of corresponding prefix#s signed by private keys of their issuers. This course of issuing prefixes and keys can be applied to the fourth tier, the fifth tier, and so forth. The algorithms based on a variation of ELGamal signature for issuing keys, which includes initiation, hierarchical issuing of key pairs, validation of signatures, are as follows. (1) Initiation: This initiation is executed by IANA, Using the key generation for ELGamal signature [21], IANA creates itself public/private key pair and the parameters of the whole signature/verification system, whose steps are as follows as well as the initiation specifies a hash function h: \{0, 1\}^* \rightarrow \mathbb{Z}_q.

Algorithm 1: Initiation
1. Select a large random prime p
2. Select a generator g of the unique cyclic group \(\mathbb{Z}_p\). Select a prime q, which is 160 bits long and \(1 = g^s \mod p\). The q is a divisor of (p-1).
3. Select a random integer \( s_0 \) such that \( 0 < s_0 < q \)
4. Compute \( pk = g^{s_0} \mod p \)
5. IANA’s public key is \( (p, g, q, PK_0) \), which is publish-ed to all of entities of the current system; IANA’s private key is \( s_0 \)

(2) Hierarchical issuing of key pairs: Some level entity (the level of IANA is assumed to be 0), who is the owner of \( \text{prefix}_{i-0} \) is issued public/private key pair for \( \text{prefix}_{i-0} \), by the owner of \( \text{prefix}_{i-1} \) where \( \text{prefix}_{i-1} \) is contained by \( \text{prefix}_{i-1} \). This key pair issuing is actually to create a variation of ELGamal signature of \( \text{prefix}_{i-1} \) signed by the owner of \( \text{prefix}_{i-1} \). The owner of \( \text{prefix}_{i-1} \), whose public/private key pair corresponding \( \text{prefix}_{i-1} \) is \( pk_{i-1}/sk_{i-1} \), issues a public/private key pair for the owner of \( \text{prefix}_{i} \) as following algorithm.

Algorithm 2: Hierarchical issuing of key pairs
1. Select a random secret integer \( k_i \), \( 0 < k_i < q \).
2. Compute \( r_i = g^{k_i} \mod p \)
3. Compute \( s_i = (k_i h(\text{prefix}_{i-1}) + r_i \cdot s_{i-1}) \mod q \)
4. The signature for \( \text{prefix}_{i-1} \) signed by the owner of \( \text{prefix}_{i-1} \) is \( (r_i, s_i) \)
5. The owner of \( \text{prefix}_{i-1} \) send the pair \( (r_i, s_i) \) and \( r_i, r_2, ..., r_{i-1} \) to the owner of \( \text{prefix}_{i} \) (the level of IANA is assumed to be 0), who is the owner of \( \text{prefix}_{i-1} \)

By algorithm 2, the owner of \( \text{prefix}_{i} \) is issued the pair \( (r_i, s_i) \), which corresponds to \( \text{prefix}_{i} \). The private key is the signature \( s_i \); the public key is a value of function of \( pk_{i-1} \) and \( r_i \) where the function will be derived and explained later. It is no matter that a private key is known by its issuer under the assumption that the issuer has no incentives to hijack its customers’ addresses as customer traffic will always traverse its network [13].

(3) Validation of signatures: By Algorithm 2, the owner of \( \text{prefix}_{i} \) is issued the pair \( (r_i, s_i) \), which corresponds to \( \text{prefix}_{i} \). The owner of \( \text{prefix}_{i} \) should validate this signature including private key \( s_{i-1} \) the algorithm used to do which is as Algorithm 3.

Algorithm 3: Validation of signatures
1. Obtain authentic public key corresponding \( \text{prefix}_{i-1} \), which is \( pk_{i-1} \)
2. Verify that \( 0 < r_i < p \) and \( 0 < s_i < q \); if not, then reject the signature.
3. Accept the signature \( (r_i, s_i) \) if and only if \( g^{s_i} = (r_i h(\text{prefix}_{i}) + (pk_{i-1})^{s_{i-1}}) \mod p \)

C. Creating Address Attestations

In the proposed method, an address attestation is a variation of ELGamal signature of \( \text{AS}_0 \) signed by the private key corresponding to \( \text{prefix}_{i} \). This private key is owned by the owner of \( \text{prefix}_{i} \). It is issued by the owner of \( \text{prefix}_{i-1} \). Using this private key, which is denoted by \( s_i \), the owner of \( \text{prefix}_{i} \) can create an address attestation of “\( \text{prefix}_{i}, \text{AS}_0 \)” as following algorithm.

Algorithm 4: Creating an address attestation of “\( \text{prefix}_{i}, \text{AS}_0 \)”
1. Select a random secret integer \( k \), \( 0 < k < q \).
2. Select an origin autonomous system of \( \text{prefix}_{i} \), whose number denoted by \( \text{AS}_0 \)
3. Compute \( r = g^{k} \mod p \)
4. Compute \( s = (k h(\text{AS}_0) + r \cdot s_{i}) \mod q \)
5. The signature for \( \text{AS}_0 \) signed by the owner of \( \text{prefix}_{i} \) is the pair \( (r, s) \)
6. The owner of \( \text{prefix}_{i} \) sends the pair \( (r, s) \) and \( r_i, r_2, ..., r_{i-1} \) to the origin autonomous system of \( \text{prefix}_{i} \)

After executing Algorithm 4, the owner of advertised prefix gets the address attestation and public-key-related information. The address attestation is the signature \( (r, s) \); the public-key-related information is the \( r_1, r_2, ..., r_{i-1} \). Next, the address attestation and public-key-related information will be transported as part of update messages, which is described as following.

D. Distribution of Address Attestations and the Public-Key-Related Information

After creating address attestation and the public-key-related information, the owner of \( \text{prefix}_{i} \) sends them to the origin \( \text{AS} \) of \( \text{prefix}_{i} \). Any other \( \text{AS} \) in the \( \text{AS-PATH} \) can use them to authenticate whether the \( \text{AS}_0 \) is the origin \( \text{AS} \) of prefix, or not.

Before verifying, this address attestation and the public-key-related information should be distributed to all of the \( \text{AS} \)s in the \( \text{AS-PATH} \). As what is described before, in proposed method, any of address attestations and the public-key-related information are distributed in update messages rather than out of band. The process of distribution is as follows.

After receiving address attestation and public-key-related information from the owner of \( \text{prefix}_{i} \), \( \text{AS}_0 \) puts it into update messages of \( \text{prefix}_{i} \) and advertises this prefix by sending update messages to some neighbors. These neighbors verify them, and send update messages to other their neighbors after concatenating themselves numbers of \( \text{AS} \) (\( \text{AS} \)s) to the \( \text{AS-PATH} \) attributes. The rest distributing of the address attestation can be done in the same manner. So, by broadcasting within update messages, this address attestation and the public-key-related information can be distributed to all of \( \text{AS} \)s in the \( \text{AS-PATH} \).

E. Verifying Address Attestations and Related Public Keys

Algorithm 5: Verifying Address Attestations and Related Public Keys
1. From update message obtain the address attestation which is the pair \( (r, s) \), and the public-key-related information which is \( r_1, r_2, ..., r_{i-1} \)
2. Verify that \( 0 < r, r_1, r_2, ..., r_{i-1} < p \) and \( 0 < s < q \); if not, then reject the signature.
3. Compute public key corresponding \( \text{prefix}_{i} \), which is \( pk_{i} \), according to Formula (3)
4. Accept the address attestation, which is the pair \( (r, s) \), if and only if \( g^{s} = (r h(\text{prefix}_{i}) + (pk_{i})^{s_{i}}) \mod p \)

By broadcasting with update messages, the address attestation and public-key-related information can be received by every \( \text{AS} \) in the \( \text{AS-PATH} \). A receiver can take the address attestation from update messages of \( \text{prefix}_{i} \) and verify it using public key corresponding to \( \text{prefix}_{i} \). The algorithm 5 is used to verify the address attestation gotten from Algorithm 4.

In the variation of ELGamal signature, a public key is a function of corresponding private key. If we denote public key by \( pk_{i} \) and denote private key by \( s_{i} \), then the function mapping a private key to corresponding public
key is that \( pk = g^x \mod p \), where \( g \) and \( p \) are parameters published to all of entities of the current system in the initiation phase. According to Algorithm 3, we can see that for validation of signatures, the following equation should be checked.

\[
g^x = (r_0)^{(h_{\text{prefix}} \cdot pk_{i-1}) \mod p} \mod p \quad (1)
\]

The signature \((r_i, s_i)\) is correct if this equation is satisfied otherwise false. Because \( pk = g^x \mod p \) holds in the variation of ElGamal signature, we can know that

\[
pk = (r_0)^{(h_{\text{prefix}} \cdot pk_{i-1}) \mod p} \mod p \quad (2)
\]

Next, we recursively apply the formula (2) as follows.

\[
pk_1 = (r_1)^{(h_{\text{prefix}} \cdot pk_{i-1}) \mod p} \mod p \quad (3)
\]

\[
pk_2 = (r_1)^{(h_{\text{prefix}} \cdot r_2) \mod p} \mod p \quad (4)
\]

\[
pk_{i-2} = (r_1)^{(h_{\text{prefix}} \cdot r_{i-2}) \mod p} \mod p \quad (5)
\]

\[
pk_i = (r_1)^{(h_{\text{prefix}} \cdot pk_{i-1}) \mod p} \mod p \quad (6)
\]

Therefore,

\[
pk_i = (r_0)^{(h_{\text{prefix}} \cdot pk_{i-1}) \mod p} \mod p
\]

\[
= (r_0)^{(h_{\text{prefix}} \cdot (r_1)^{h_{\text{prefix}} \cdot pk_{i-2}) \mod p} \mod p) \mod p
\]

\[
= (r_0)^{(h_{\text{prefix}} \cdot ((r_1)^{h_{\text{prefix}} \cdot pk_{i-3}) \mod p) \mod p} \mod p
\]

\[
= \cdots = \cdots \cdots
\]

After the process of iteration from \( pk_1 \) to \( pk_{i-1} \), the following equation can be achieved.

\[
pk_i = \prod_{j=1}^{i} r_j \mod p \mod p \quad (7)
\]

The public key of prefix, \( s_i \) which is denoted by \( pk \), and can be computed using formula (7), can be used to verify address attestations of “prefix/AS0” by all of ASs of AS-PATH. The validation algorithm is as Algorithm 5. If the validation of an address attestation is succeed using the \( pk \), computed from formula (7), then following statements can be authenticated.

1. The address attestation is the variation of ElGamal signature of AS0 signed by the private key (denoted by \( s_i \)) corresponding the \( pk \).
2. The public/private key pair of prefix, \( s_i \), is issued to the owner of prefix, definitely by the hierarchical issuing process. In other words, The public key \( pk \) is authenticated.

Formula (7) shows that every \( r_j \) (\( j = 1, 2, \ldots, i \)) should also be distributed in step with address attestations, so as to make verifiers compute \( pk \).

IV. Evaluation

In this section, for proposed method, we evaluate security, update message size, and overhead of computation.

A. Secure Evaluation

In proposed method, an address attestation about the advertised prefix is created by signing the AS0 with the advertised prefix owner’s private key. A receiver verifies this address attestation using public key. The public key is authenticated at the same time. Obviously, the binding “prefix/AS0” is secure after executing this authentication process.

This process is same as that of S-BGP [9]. The security of S-BGP has received wide acceptance. Thus, like S-BGP, proposed method can defend against full prefix hijack and sub-prefix hijack.

B. Update Message Size

According to size of related information used to defend against IP prefix hijacking, we compare the proposed method with the S-BGP.

\( Size_1 \approx (500-1000) \times 7 + 128 + 20 = 3648-7148 \) bytes

\( Size_2 = 128 \times 7 + 128 + 20 = 1044 \) bytes

The \( Size_1 \) denotes the size of related information in S-BGP. This information includes public key certificates and an address attestation. A public key certificate is about 500~1000 bytes long. There are 3~7 certificates in a single issuing chain which is from IANA to the owner of advertised prefix [12]. We assume there are 7 certificates when computing this size. An address attestation is always a DSA (Digital Standard Algorithm) signature shaped like \((r, s)\). Therefore, the address attestation is about 128+20 bytes. Thus, an address attestation is about 128+20 bytes.

The \( Size_2 \) denotes the size of related information in the proposed method. This information includes the public-key-related information, and an address attestation. The public-key-related information is \( r_1, r_2, \ldots, r_i \), where each \( r_j \) (\( j = 1, \ldots, i-1 \)) is 128 bytes long. The number of these \( r_j \) is about 3~7 [12]. We assume the number is 7 when computing this size. An address attestation is always a variation of Elgamal signature shaped like \((r, s)\). Therefore, the address attestation is about 128+20 bytes.

By comparing \( Size_1 \) with \( Size_2 \), we can see that in S-BGP, because update messages are limited in length to 4096 bytes and thus are too small to carry the necessary related information; however, in the proposed method, the limitation of 4096 bytes can not prevent the related information from being part of update messages, and thus a verifier can authenticate address attestations and public keys only according to the information within update messages.

Therefore, in proposed method, it is not necessary for a verifier to check a series of certificates existing in certification path from the owner of advertised prefix to the IANA. Thus the lazy authentication of origin ASs is achieved.
C. Overhead of Computation

In proposed method, for authenticating an origin AS of a prefix, a verifier’s computing burden is mainly that computing public key of owner of prefix using formula (7), and using this public key to verify the address attestation which is a variation of ELGamal signature. Formula (7) is a multi-exponentiation. The computing of Formula (7) can be much more efficient than computing every single exponentiation separately and then multiplying them using some trick algorithms [25].

The path validation from IANA to the owner of advertised prefix, where these signatures can only be verified separately in current typical schemes (such as S-BGP, soBGP, pS-BGP, OA and so forth) of authentication of origin AS, can be replaced with efficient computing of only a single formula (7).

We use \( C_1 \) to denote the computation overhead of the separate path validation from IANA to the owner of advertised prefix in S-BGP; use \( C_2 \) to denote the overhead of computing Formula (7). Assume the signature scheme used in S-BGP is DSA. Because the exponent of exponentiation is 160 bits long, the computation overhead of verification of a single DSA signature is roughly \( 2.5 \times 160 \) multiplications [25]. Assume that the path from IANA to the owner of advertised prefix is 7 long. Thus, the Formula (7) includes 7 exponentiations. The overhead of computing a multi-exponentiation, which includes 7 exponentiations, is roughly \( (7 \times 0.5 + 1) \times 160 \) multiplications [25]. Next, we compare \( C_1 \) with \( C_2 \) as following.

\[
C_1 \approx 2 \times 2.5 \times 160 = 2800 \text{ multiplications}
\]

\[
C_2 \approx (7 \times 0.5 + 1) \times 160 = 720 \text{ multiplications}
\]

By comparing \( C_1 \) with \( C_2 \) we can see that the overhead of efficient computing formula (7) is roughly 720 multiplications, which is far less than the 2800 multiplications roughly required by the path validation from IANA to the owner of advertised prefix in S-BGP. Therefore the proposed method can bring about fast authentication of origin ASs.

V. CONCLUSION AND FUTURE WORK

It is appropriate to authenticate address attestations and related public keys only using the information of update messages. Yet, this approach is infeasible in existing schemes because update messages are limited in length to 4096 bytes and thus are too small to carry the necessary public key certificates. For realizing this approach, in this paper we present a method based on a variation of ELGamal signature to achieve fast and lazy authentication of origin AS. This method includes several parts. These parts are authenticating framework, issuing public/private key pairs, creating address attestations, distribution of address attestations and the public-key-related information, and verifying address attestations and related public keys.

By making full use of specific characteristics of the variation of ELGamal signature, the proposed method can at least enjoy two following advantages: (1) It is not necessary for a verifier to check certificates out of update messages. Almost all of data, which include address attestations and the public-key-related information, can be gotten directly from update messages. (2) Only a multi-exponentiation formula needs to be efficiently computed for completing path validation from IANA to the owner of advertised prefix. Therefore, the proposed method for authenticating origin ASs is not only secure but also fast and lazy. To the best of our knowledge, it is the first work for authenticating origin autonomous systems by using ELGamal signature to achieve a fast scheme that can authenticate address attestations and related public keys only using the information of update messages.

For future work, more extensive studies will be followed for real application, such as key revocation. For key revocation, we may add an expiration time into the signed messages so as to keys are invalid as soon as the expiration time expires.

ACKNOWLEDGMENT

The authors are grateful to the anonymous referees for their valuable comments and suggestions to improve the presentation of this paper. This work was supported in part by the Foundation of High Level Engineering Research Center of Electronic-Commerce of Jiangxi Provincial Colleges and Universities, Science and Technology Foundation of the Education Department of Jiangxi Province (No. GJJ12735), National Nature Science Foundation of China (No. 10961013/A010206), and Natural Science Foundation of Jiangxi (No. 20132BAB201036).

REFERENCES


Bo Yang received the Bachelor’s degree from the Huazhong University of Science and Technology, and the M.S. and Ph.D. degrees in computer science from the University of Nanchang, China. He is currently a faculty at the Department of Computer Science and Technology, School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, China. He has worked at the Institute of Software, Chinese Academy of Sciences. His research interests include Internet routing and network security.