A Novel Protocol for Securing Network Slice Component Association and Slice Isolation in 5G Networks

Vipin N Sathi, Manikantan Srinivasan, Prabhu K Thiruvasagam, Siva Ram Murthy Chebiyyam
Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai, India, 600036
vipinns@cse.iitm.ac.in,mani@cse.iitm.ac.in,prabhut@cse.iitm.ac.in,murthy@iitm.ac.in

ABSTRACT

Slicing of a 5G network by creating virtualized instances of network functions facilitates the support of different service types with varying requirements. The management and orchestration layer identifies the components in the virtualization infrastructure to form an end-to-end slice for an intended service type. The key security challenges for the softwarized 5G networks are, (i) ensuring availability of a centralized controller/orchestrator, (ii) association between legitimate network slice components, and (iii) network slice isolation. To address these challenges, in this paper, we propose a novel implicit mutual authentication and key establishment with group anonymity protocol using proxy re-encryption on elliptic curve. The protocol provides (i) controller independent distributed association between components of a network slice, (ii) implicit authentication between network slice components to allow secure association, (iii) secure key establishment between component pairs for secure slice isolation, and (iv) service group anonymity. The proposed protocol's robustness is validated with necessary security analysis. The computation and bandwidth overheads of the proposed protocol are compared with that of the certificate based protocol, and our proposed protocol has 9.52% less computation overhead and 13.64% less bandwidth overhead for Type A1 pairing.

CCS CONCEPTS

• **Security and privacy → Key management**; **Access control**; **Security protocols**; **Mobile and wireless security**; • **Networks → Network management**; **Mobile networks**;

KEYWORDS

5G; Telco Cloud; Network Functions Virtualization; Network Slicing; Secure Network Slice Association; Secure Network Slice Isolation; Proxy Re-encryption

ACM Reference Format:
Vipin N Sathi, Manikantan Srinivasan, Prabhu K Thiruvasagam, Siva Ram Murthy Chebiyyam. 2018. A Novel Protocol for Securing Network Slice Component Association and Slice Isolation in 5G Networks. In *21st ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM '18), October 28-November 2, 2018, Montreal, QC, Canada.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3242102.3242135

MSWIM '18, October 28-November 2, 2018, Montreal, QC, Canada

ACM ISBN 978-1-4503-5960-3/18/10…$15.00
https://doi.org/10.1145/3242102.3242135

1 INTRODUCTION

Fifth generation (5G) networks are primed to support new softwarized services. The services to be supported in 5G networks can be broadly grouped under the categories of enhanced mobile broadband (eMMB), mission critical communication, ultra-reliable and low-latency communications (uRLLC), and massive machine-to-machine communications. The system support required for aspects such as latency, mobility, traffic and connection density, spectrum and power efficiency, and data rates significantly vary for the services, which can be provided only by dedicated networks [5]. However, establishing dedicated networks is not an optimal choice due to high capital and operational costs. The ideal solution is to utilize a single physical infrastructure to provide the necessary support with sufficient demarcation. Network softwarization using network function virtualization (NFV) and software defined networking gives the necessary tools to slice the physical network infrastructure into several logical networks (network slicing [3]) such that different service requirements are efficiently supported [2].

The maturity of cloud computing paradigm and network softwarization drives the 5G network deployments to be realized as Telco Clouds. Telco Clouds utilize virtualized and distributed architecture, resulting in a shift from network of entities to a network of capabilities, and network for connectivity to network for services. The 5G services are realized as suitable network slices enabled as a chain of network slice components (NSCs). An NSC is a an abstraction of a network function or capability designed for providing the functional requirements of an end-to-end network slice [11]. Examples of NSCs include 5G network function components such as access and mobility management function, session management function, policy control function, network slice selection function, unified data management, and authentication server function [8].

NFV architectural framework consists of (1) NFV infrastructure (NFVI), (2) virtual network functions (VNFs), and (3) management and orchestration (MANO) layer. Enabling network slices require interactions between service layer, network function layer and infrastructure layer. The interactions between the layers are achieved with well defined APIs, which are co-ordinated typically by an efficient MANO layer. A centralized controller, say an orchestrator (*OR*), part of the MANO layer, is responsible for network service (NS) life cycle management (including instantiation, scale-in/out, performance measurements, event correlation, termination), global resource management, validation and authorization of NFVI resource requests and policy management for NS instances. The *OR* uses the network service templates defined by the service provider to form the service function chain (SFC) or VNF forwarding graph (VNF-FG) to provide service to the user. Establishment of SFC/VNF-FG is performed by the *OR*. There are scenarios where distributed association between NSCs may be required, for instance, in the case

of uRRLC, new identical NSCs are instantiated to support horizontal scaling by the localized VNF manager without contacting *OR* to reduce the latency, and the newly instantiated NSCs are instructed to associate with peer NSC for creating end-to-end network slice.

Some of the key 5G network security aspects identified by the industry and standard bodies include; (1) holistic security orchestration and management with suitable identification and authentication mechanisms, (2) robust security support at all levels for a network slice, (3) ensuring privacy protection, (4) increased robustness against cyber attacks, and (5) security assurance for higher degree of security automation [1, 9]. Major security risks associated with network softwarization are ensuring controller/orchestrator availability [10], isolation failure, malicious insider, compromised NFV instances, and insecure data access [6]. Attackers may perform DDoS kind of attacks to delay the communication between entities (e.g., with and within the MANO layer) or disrupt the life cycle management of the network services and NSCs [10].

Distributed Association between NSCs: If the NSCs can get associated in a distributed way to form end-to-end network slices, then unavailability of the *OR* will not cause disruption of the network services. The association should happen only between the NSCs (NSCs of an SFC/VNF-FG) of the same service provider (SP) (e.g., an IoT SP) to avoid the inclusion of a malicious (attacker's) NSC in the end-to-end network slice. This can be avoided by verifying the authenticity of the NSCs by themselves during the association.

Certificates based Secure Distributed Association:
When using certificates the *OR* has to sign and issue certificates to all the NSCs while they are created, using the private key which the *OR* has reserved for the SP to which the NSCs belong to (*OR* needs to generate public-private key pair corresponding to all SPs). But the problem with certificate based verification is that it reveals information about the SP to which an NSC belongs. This allows the attacker to identify and target attacks on NSCs which belong to a certain SP. Hence there is a requirement for service group anonymity while the NSCs get associated in a distributed manner.

Pre-configured Key based Secure Distributed Association:
When using pre-configured keys and NSC associations dynamically change (e.g., greening strategies requiring dynamic NSC association), it is difficult to find a common key between the NSCs. Also, the keys pre-configured for a group of NSCs may have to be reconfigured if at least one of the NSCs has moved out of the group.

Proposed Protocol for Secure Distributed Association:
We propose a novel protocol based on proxy re-encryption scheme using bilinear pairing on an elliptic curve to provide secure service group anonymous association between NSCs of an SP. According to our proposed protocol, information about the SP of an NSC is revealed only between the NSCs which belong to the same SP, and the NSCs which do not belong to the same SP cannot determine the SP's identity. Also, our proposed protocol provides network slice isolation by protecting the communication between legitimate NSCs using separate encryption keys for every NSC pair.

The rest of the paper is organized as follows. Related work is discussed in section 2. Section 3 provides the system model, and section 4 defines the research problem. The proposed protocol is detailed in section 5 followed by security analysis in section 6. Section 7 provides aspects on performance analysis in terms of computation overhead followed by conclusion in section 8.

2 RELATED WORK

Public Key Infrastructure (PKI): In PKI, authenticity is ensured using certificates. Key issues are certificate management, storage, computational cost, and distribution by the certificate authority.

Proxy Re-encryption: Proxy re-encryption allows a proxy to convert a ciphertext of one entity to another entity's ciphertext. While doing this conversion the actual content of the message and the private key of the entity to whom the actual ciphertext is converted is not revealed to the proxy. The proxy uses re-encryption keys to do the ciphertext conversion. No information about the secret keys of the participating entities is revealed to the proxy from the re-encryption key. Thangam and Chandrasekaran [12] propose a proxy re-encryption scheme based on elliptic curve. According to their scheme, multiple parties can decrypt a ciphertext which is encrypted with the public key of the party who issues re-encryption keys to the entities. This may be suitable for a secure broadcast communication in a group, but not suitable for securing peer-to-peer communication in a group.

Security and Network Slicing: The end-to-end slicing approach in 5G networks makes management of slices and inter-slice access complex. Management of access to slices, protecting access to slices, secure mutual access between radio access network and core network resources, and secure attachment of mobile equipment to slice instances are major concerns of a secure end-to-end slice management. Targetable components for an attacker in a softwarized network are VNFs, hypervisor, communication with MANO, *OR*, etc., [10]. Ni *et al.* [8] propose a secure mechanism to allow fog nodes (controller) to select proper network slices by hiding the accessing service type of users, and anonymously authenticate users to IoT servers. The authors of [8] have assumed that the NSCs are already securely associated to form an SFC to provide network services to users (only selection of proper network slices are considered). In this paper, we address how NSCs of an SP are securely associated to form an SFC in a distributed manner with service group anonymity to provide required services to the users. Liu *et al.* [7] propose two heterogeneous signcryption schemes (PKI scheme and certificate-less scheme) for mutual communication between different network NSCs having heterogeneous cryptosystems in a network sliced 5G network. A signcryption scheme performs digital signature and encryption together. Their schemes are based on the difficulty of discrete logarithm problem (DLP) on elliptic curve.

Our proposed protocol based on proxy re-encryption and bilinear pairing on elliptic curve, restricts decryption of a ciphertext only by an intended entity unlike [12]. Proxy re-encryption based approach is opted to address the issues of the certificate based scheme. The proposed protocol

(1) Ensures mutual trust between participating entities without using explicit signatures.
(2) Secures distributed association between NSCs to form a secure network slice even if the *OR* is unavailable (under DDoS attack).
(3) Provides service group anonymity, and secure network slice isolation by protecting inter NSC communication.

To the best knowledge of the authors, the proposed protocol is the first solution for securing distributed association between NSCs of a network slice in 5G networks with service group anonymity without the involvement of *OR*.

3 SYSTEM MODEL

In softwarized 5G networks, service providers offer different services using end-to-end network slices. NSCs that are part of a network slice are enabled as virtualized functions in the NFVI. Every NSC has to get securely associated with its legitimate peer NSC, which belongs to the same network slice. A network slice is configured to enable a new service; (1) when a user / device requests a new service type, (2) for bringing operational efficiencies such as greening strategies. Figure 1 shows a sample Telco Cloud based NFV deployment of three service providers sp_1, sp_2, and sp_3. The service types st_1, st_2, and st_3 are realized as an end-to-end network slice using the NSCs in the Telco Cloud (RAN cloud, Edge cloud and Core cloud). In Figure 1, shape of an NSC differentiates its functionality and it's fill pattern indicates to which cloud it belongs.

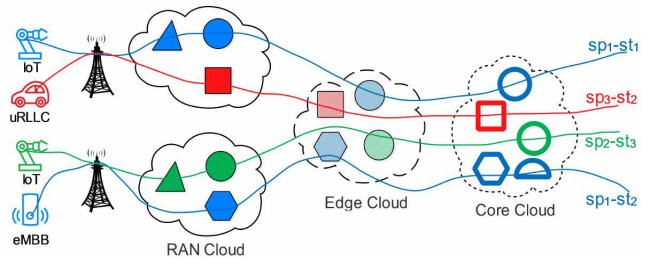

Figure 1: A sample network slice enabled 5G Telco Cloud supporting multiple service providers and service types.

In these deployments, the orchestrator (*OR*) creates a pool of NSCs (VNF instances) with the capabilities to support the required services. Let $\mathcal{S} = \{s_1,\ldots,s_i,\ldots,s_z\}$ be the set of all NSCs created by the *OR*. The number of service providers in the system is N and the number of service types at most served by a service provider is M. A service provider sp_n $(1 \leq n \leq N)$ supports st_m $(1 \leq m \leq M)$ service types. *OR* is assumed to be a trusted entity; but it can passively participate in attacks. Some of the NSCs may become malicious. There can be other network entities which are malicious and capable of poisoning some of the NSCs created by the *OR*.

4 PROBLEM DEFINITION

NSCs of a service type belonging to a service provider are identified using certain meta data provided by the *OR* at the time of NSC creation. Network slicing involves association between legitimate NSCs of an end-to-end network slice, that meets the specifications of a service type supported by a service provider. A centralized controller such as *OR* has the responsibility to perform secure NSC association for enabling a network slice. This requires the availability of the *OR* at all times, which is a major security requirement for an NFV MANO. A DDoS attack on *OR* can make *OR* unavailable to perform NSC association. Controller (*OR*) availability dependency can be resolved by a distributed association mechanism which ensures secure association between NSC pairs of a network slice. Also, there are scenarios where distributed association between NSCs may be required as in the case of uRRLC, for which horizontal scaling by the localized VNF manager without contacting the *OR* can be performed to reduce the end-to-end service latency.

5 IMAKE-GA PROTOCOL

In this paper, we propose a protocol for secure association between legitimate NSCs of a network slice. The proposed protocol ensures implicit mutual authentication and key establishment with service group anonymity (IMAKE-GA) between NSCs. An NSC supporting the proposed IMAKE-GA protocol does not reveal the identity of the SP to which the NSC belongs, as it is not using certificates to authenticate the peer NSC. Hence the information about the SP is not revealed to attackers who try to associate with the legitimate NSC (In the certificate based authentication, identity of the authority who issued the certificate is revealed). The proposed IMAKE-GA protocol is based on proxy re-encryption scheme using bilinear pairing on an elliptic curve. Cryptographic modifications to the existing elliptic curve based proxy re-encryption scheme [12] enables the proposed IMAKE-GA protocol to restrict decryption of a ciphertext only by an intended receiver. To the best knowledge of the authors, the proposed IMAKE-GA protocol is the first solution for service group anonymous secure distributed association between NSCs of a network slice in 5G networks, and to enable secure slice isolation for the communication between NSCs without the involvement of *OR*. The key benefits of the proposed IMAKE-GA protocol are enabling,

(1) Secure distributed association between NSCs of an SP (without the assistance of *OR*).
(2) Implicit mutual authentication between NSCs of an SP.
(3) Secure key establishment between NSCs of an SP to have slice isolation for inter NSC communication.
(4) Service group anonymity (information as to which SP an NSC belongs is not revealed).

Public Cryptographic Parameters (CP): Let E be an elliptic curve of order n over a finite field F_q (q is a large prime number). G is the base point on the curve E, defined by the *OR*. Let G_1 be an additive group of points of E, and G_2 be a multiplicatively-written group of order n. Let there be a bilinear map $e : G_1 \times G_1 \longrightarrow G_2$ and $z = e(G, G)$. Also, let there be functions *map* and *rmap* to convert a message to a point on the curve E and back. The public cryptographic parameters are $CP = (E, q, n, e, G, G_1, G_2, z, map, rmap, kG)$. The parameter kG is computed using a very large random integer $k \in Z_n^*$.
Key Pair Generation: The *OR* maintains a distinct private-public key pair for each service type offered by a service provider. Let $Pr_{OR} = \zeta$, (ζ randomly chosen from Z_n^*), and $Pu_{OR} = \zeta G$, represent the private and public key, respectively for a service type st_m of a service provider sp_n. Every NSC $s_i \in \mathcal{S}$ after the initial mutual authentication with the *OR*, generates a private key $Pr_{s_i} = \delta_i$, (δ_i randomly chosen from Z_n^*) and a public key $Pu_{s_i} = \delta_i G$.
Re-encryption Key Generation: *OR* issues the re-encryption key ($Rek_{OR \to s_i}$) to NSC s_i during the instantiation of NSC s_i. *OR* computes the re-encryption key $Rek_{OR \to s_i}$ using the inverse of its private key $\zeta^{-1} \in Z_n^*$ corresponding to the service type st_m of the service provider sp_n* and the public key $\delta_i G$ of NSC s_i as follows,

$$\begin{aligned} Rek_{OR \to s_i} &= \zeta^{-1} Pu_{s_i} + \zeta^{-1} kG \\ &= \zeta^{-1}(\delta_i + k)G \end{aligned} \quad (1)$$

The parameter kG is included in the re-encryption key generation to ensure revocation (invalidation) of the credentials of NSC s_i if it turns malicious.

Encryption of a Message by s_i for s_j: A per message secret $r \in Z_n^*$ is generated by NSC s_i which wishes to send a message P_m to NSC s_j (encryption using the public keys of *OR* and NSC s_j). The secret value r prevents replay attacks by an attacker. The ciphertext $C_{OR_{s_j}}$ of message P_m is calculated using $\delta_j G$ (public key of NSC s_j), supplied by NSC s_j. The ciphertext $C_{OR_{s_j}}$ is expressed in terms of two components A_j and B_j.

$$C_{OR_{s_j}} = (A_j,\ B_j) = (rPu_{OR}, [e(rPu_{s_j}, (Pu_{s_j}\ +\ kG))\ G\ +\ P_m]) = (r\zeta G,\ [z^{r\delta_j(\delta_j+k)}G\ +\ P_m]) \quad (2)$$

The output component A_j of the encryption operation can be pre-computed by the NSC s_i to reduce the effective computation time. Component A_j helps the legitimate NSCs which have their re-encryption key to extract the message P_m from the component B_j.

Re-encryption of Ciphertext $C_{OR_{s_j}}$ by s_j: C_{s_j} is the re-encrypted ciphertext of $C_{OR_{s_j}}$, computed by NSC s_j using its re-encryption key $Rek_{OR\longrightarrow s_j}$ and its private key Pr_{s_j}.

$$C_{s_j} = (A'_j,\ B'_j) = ([e(A_j,\ Pr_{s_j}Rek_{OR\rightarrow s_j})],\ B_j) = ([z^{r\delta_j(\delta_j+k)}],\ [z^{r\delta_j(\delta_j+k)}G\ +\ P_m]) \quad (3)$$

The parameter $Pr_{s_j}Rek_{OR\rightarrow s_j}$ can be pre-computed by the NSC s_j to reduce the effective computation time.

Decryption of C_{s_j} by s_j using its private key: NSC s_j decrypts the ciphertext C_{s_j} as follows,

$$P_m = B'_j - A'_jG = z^{r\delta_j(\delta_j+k)}G\ +\ P_m - z^{r\delta_j(\delta_j+k)}G \quad (4)$$

IMAKE-GA Protocol for NSC Association: Any two legitimate NSCs s_i and s_j can use IMAKE-GA protocol to perform implicit mutual authentication and secure key establishment by securely exchanging their Diffie-Hellman (DH) parameters. NSCs s_i and s_j exchange Pu_{s_i} and Pu_{s_j} to get the DH parameters aG and bG in encrypted form $C_{OR_{s_j}}$ and $C_{OR_{s_i}}$, respectively. NSCs s_i and s_j compute $C_{OR_{s_j}}$ and $C_{OR_{s_i}}$, respectively, using equation 2 as follows,

$$C_{OR_{s_j}} = (A_j,\ B_j) = (r_1\zeta G,\ [z^{r_1\delta_j(\delta_j+k)}G\ +\ aG]) \quad (5)$$

$$C_{OR_{s_i}} = (A_i,\ B_i) = (r_2\zeta G,\ [z^{r_2\delta_i(\delta_i+k)}G\ +\ bG]) \quad (6)$$

The re-encrypted ciphertext C_{s_j} constructed from $C_{OR_{s_j}}$ using equation 3 by NSC s_j is as follows,

$$C_{s_j} = (A'_j,\ B'_j) = ([z^{r_1\delta_j(\delta_j+k)}], [\ z^{r_1\delta_j(\delta_j+k)}G\ +\ aG]) \quad (7)$$

Similarly, the re-encrypted ciphertext C_{s_i} constructed from $C_{OR_{s_i}}$ by NSC s_i is as follows,

$$C_{s_i} = (A'_i,\ B'_i) = ([z^{r_2\delta_i(\delta_i+k)}], [\ z^{r_2\delta_i(\delta_i+k)}G\ +\ bG]) \quad (8)$$

Then NSCs s_j and s_i perform decryption of their re-encrypted ciphertext C_{s_j} and C_{s_i}, respectively, using equation 4 as follows,

$$aG = B'_j - A'_jG = [z^{r_1\delta_j(\delta_j+k)}G\ +\ aG] - [z^{r_1\delta_j(\delta_j+k)}G] \quad (9)$$

$$bG = B'_i - A'_iG = [z^{r_2\delta_i(\delta_i+k)}G\ +\ bG] - [z^{r_2\delta_i(\delta_i+k)}G] \quad (10)$$

NSCs s_i and s_j compute abG as a secure key to provide slice isolation for the inter communication between them.

6 SECURITY ANALYSIS

1. IMAKE-GA Protocol Allows Decryption of $C_{OR_{s_j}}$ only by NSC s_j: If another NSC, say s_l tries to decrypt C_{OR_j} then it will compute its re-encryption cipher as follows,

$$C_{s_l} = (A'_j,\ B'_j) = ([e(A_j,\ Rek_{OR\rightarrow s_l})],\ B_j) = ([z^{r(\delta_l+k)}],\ [z^{r\delta_j(\delta_j+k)}G\ +\ P_m]) \quad (11)$$

The value of k is only known to *OR*, and the value of δ_j is only known to NSC s_j. Hence any other NSC s_l other than NSC s_j cannot decrypt C_{OR_j}, since finding k from kG, and δ_j from $\delta_j G$ are infeasible because of the hardness of DLP on elliptic curve.

2. IMAKE-GA Protocol Enables NSC Revocations: An NSC can be revoked by *OR*, if the NSC behaves maliciously. Revocation can be done using a revocation list based on the public key of the NSC which is identified as malicious. We assume, that the malicious behavior of an NSC can be identified by a peer NSC or by the *OR*. When *OR* identifies an NSC as malicious, it provides a revocation list which includes the public key of the malicious NSC to all other NSCs that belong to the network slice of the malicious NSC. This ensures the malicious NSC can no more communicate with any other NSCs of the network slice. In case the malicious NSC s_j is aware that it is has been revoked, it can attempt to establish a new association with NSC s_i using a new public key $\beta_j\delta_j G$ instead of $\delta_j G$. NSC s_i now computes $C_{OR_{s_j}}$ as follows,

$$C_{OR_{s_j}} = (A_j,\ B_j) = (rPu_{OR},\ [e(r\beta_j\delta_j G, (\beta_j\delta_j G\ +\ kG))\ G\ +\ P_m]) = (r\zeta G,\ [z^{r\beta_j\delta_j(\beta_j\delta_j+k)}G\ +\ P_m]) \quad (12)$$

The re-encrypted ciphertext of s_j, computed by NSC s_j would be,

$$C_{s_j} = (A'_j,\ B'_j) = ([e(A_j,\ Rek_{OR\rightarrow s_j})],\ B_j) = ([z^{r(\delta_j+k)}],\ [z^{r\beta_j\delta_j(\beta_j\delta_j+k)}G\ +\ P_m]) \quad (13)$$

Now, for NSC s_j to extract P_m from B'_j, it requires r and k. NSC s_j cannot find r from A'_j, though it can find $z^{(\delta_j+k)}$ from kG and $\delta_j G$ using the bilinear paring e; $e((\delta_j G + kG), G) = z^{(\delta_j+k)} = y$. Finding r from y^r is infeasible, even when y is known and $A'_j = y^r$, because of the hardness of DLP on multiplicative cyclic group of integers modulo prime number. Similarly, NSC s_j cannot find k from kG because of the hardness of DLP on elliptic curve.

3. IMAKE-GA Protocol Ensures Service Group Anonymous Association: Information about the SP to which an NSC belongs to is not revealed to an attacker as the IMAKE-GA protocol is not using certificate based authentication.

4. IMAKE-GA Protocol Provides Implicit Authentication, Key Establishment, and Slice Isolation: The proposed IMAKE-GA protocol uses the public key of the *OR* and the receiver NSC to send the DH parameters between the NSCs. The NSCs can decrypt the DH parameters only if they have obtained their re-encryption key from *OR*. By utilizing unique private keys for the service types supported by every service provider, *OR* enables implicit authentication and association only between those legitimate NSCs which can be part of same network slice. The secure key established between the NSCs can protect inter NSC communication to provide slice isolation.

7 PERFORMANCE ANALYSIS

The computational and bandwidth overheads of the IMAKE-GA protocol is compared with certificate based authentication and key establishment (CBAKE) protocol. In CBAKE protocol, an NSC verifies the peer NSC's certificate, generates and signs a random symmetric key. It then sends to the peer NSC, the symmetric key encrypted with the public key of the peer along with the signature of the symmetric key. CBAKE protocol does not support group anonymity and has explicit signature overhead. The protocols were implemented and executed on an Intel®Core™ i7-3770 3.40GHz processor for a Type A, and Type A1 pairing with the help of JPBC library [4]. CBAKE protocol implementation uses elliptic curve based encryption, decryption, and signature algorithm. Number of bits required to represent the order (number of points on the curve) of Type A and Type A1 pairing elliptic curves are 160 and 1022, respectively. The discrete logarithm security of Type A and Type A1 pairing elliptic curves are 1024 bits and 2048 bits, respectively.

The average measured computation overhead of CBAKE and IMAKE-GA protocols for Type A and Type A1 pairing are shown in Table 1. For Type A pairing, the CBAKE protocol has less computation overhead than the proposed IMAKE-GA protocol. The proposed IMAKE-GA protocol's computation overhead is 9.52% less than the CBAKE protocol's computation overhead for Type A1 pairing (high security). For an end-to-end distributed association between NSCs of a network slice, the total computation overhead required is at least $2 \times T_{IMAKE-GA}$.

Table 1: Computation Overhead of CBAKE and IMAKE-GA

Cryptographic Operation	Avg. Execution Time: Type A Pairing (ms)	Avg. Execution Time: Type A1 Pairing (ms)
Point addition (t_{ad})	0.0855	0.1330
Point subtraction (t_{sb})	0.0875	0.1430
Scalar multiplication with a random value (t_{sc1})	17.3781	160.9119
Scalar multiplication with the result of pairing (t_{sc2})	54.3414	158.7116
Bilinear pairing (t_{pr})	10.2125	113.8323
Elliptic curve digital signature generation (t_{sig})	0.0398	0.2370
Elliptic curve digital signature verification (t_{ver})	34.7932	320.7780
Elliptic curve encryption (t_{ec-enc})	17.4319	159.0673
Elliptic curve decryption (t_{ec-dec})	17.3274	157.8101
IMAKE-GA encryption ($t_{enc} = t_{pr} + t_{sc1} + t_{sc2} + 2 \times t_{ad}$)	82.1030	433.7218
IMAKE-GA re-encryption ($t_{rek} = t_{pr}$)	10.2125	113.8323
IMAKE-GA decryption ($t_{dec} = t_{sc2} + t_{sb}$)	54.4289	158.8546
IMAKE-GA key generation ($t_{key} = t_{sc1}$)	17.3781	160.9119
CBAKE ($T_{CBAKE} = 2 \times t_{ver} + t_{sig} + t_{ec-enc} + t_{ec-dec}$)	**104.3858**	**958.6707**
IMAKE-GA ($T_{IMAKE-GA} = t_{enc} + t_{rek} + t_{dec} + t_{key}$)	**164.1225**	**867.3206**

For CBAKE protocol, initial certificate exchange results in a bandwidth overhead of twice the size of a certificate (s_{cert}), twice the size of a point (s_{pt}) of an elliptic curve for the encrypted symmetric key, and twice the size of order (s_{ord}) on the elliptic curve for the signature. The minimum information required in the certificate are the public key of the NSC, and signature of the authority. The minimum size of a certificate is $s_{pt} + 2 \times s_{ord}$. The total bandwidth overhead of the CBAKE protocol is $2 \times (s_{cert} + s_{pt} + s_{ord}) = 2 \times (2 \times s_{pt} + 3 \times s_{ord}) = (4 \times s_{pt} + 6 \times s_{ord})$. Bandwidth overhead of the IMAKE-GA protocol includes twice the size of a point of an elliptic curve point for the initial public key exchange between the NSCs, four times the size of a point of an elliptic curve for exchanging the encrypted values. So the total bandwidth overhead of IMAKE-GA protocol is $6 \times s_{pt}$. Table 2 shows the bandwidth overheads of the CBAKE and IMAKE-GA protocols for Type A and Type A1 pairing. The IMAKE-GA protocol has 13.64% lower bandwidth overhead than CBAKE protocol for Type A1 pairing.

Table 2: Bandwidth Overhead of CBAKE and IMAKE-GA

Protocol	Type A Pairing (s_{pt} = 1024, s_{ord} = 160) (bits)	Type A1 Pairing (s_{pt} = 2080, s_{ord} = 1022) (bits)
CBAKE ($4 \times s_{pt} + 6 \times s_{ord}$)	5056	14452
IMAKE-GA ($6 \times s_{pt}$)	6144	12480

8 CONCLUSION

We have proposed IMAKE-GA, a novel secure association protocol for enabling secure network slices in 5G networks. The proposed IMAKE-GA protocol ensures distributed secure association between NSC pairs of a network slice with service group anonymity. It is based on proxy re-encryption scheme using bilinear pairing on an elliptic curve. The protocol modifies the cryptographic operations of the existing elliptic curve based proxy re-encryption scheme [12] to restrict decryption of the ciphertext only by an intended receiver. Service group anonymity property of the IMAKE-GA protocol prevents targeted DDoS attacks on NSCs of a certain SP while the NSCs get associated in a distributed way. We found that the proposed IMAKE-GA protocol has 9.52% and 13.64% less computational and bandwidth overhead, respectively, compared to the CBAKE protocol for Type A1 pairing.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Chester Rebeiro for his insightful discussions. This research work was supported by the Department of Science and Technology (DST), New Delhi, India.

REFERENCES

[1] ETSI GS NFV-SEC 001. 2014. Network Functions Virtualisation (NFV); NFV Security; Problem Statement. http://www.etsi.org/deliver/etsi_gs/NFV-SEC/001_099/001/01.01.01_60/gs_NFV-SEC001v010101p.pdf
[2] 5GPP Architecture Working Group. 2017. View on 5G Architecture Version 2.0. *5G Architecture White Paper* (Dec 2017).
[3] NGMN Alliance. 2016. Description of Network Slicing Concept, NGMN 5G P1. https://www.ngmn.org/fileadmin/user_upload/161010_NGMN_Network_Slicing_framework_v1.0.8.pdf
[4] Angelo De Caro and Vincenzo Iovino. 2011. jPBC: Java pairing based cryptography. In *IEEE Symposium on Computers and Communications (ISCC)*. 850–855.
[5] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina. 2017. Network Slicing in 5G: Survey and Challenges. *IEEE Communications Magazine* 55, 5 (May 2017), 94–100.
[6] S. Lal, T. Taleb, and A. Dutta. 2017. NFV: Security Threats and Best Practices. *IEEE Communications Magazine* 55, 8 (Aug 2017), 211–217.
[7] J. Liu, L. Zhang, R. Sun, X. Du, and M. Guizani. 2018. Mutual Heterogeneous Signcryption Schemes for 5G Network Slicings. *IEEE Access* 6, 1 (Jan 2018), 7854–7863.
[8] J. Ni, X. Lin, and X. S. Shen. 2018. Efficient and Secure Service-Oriented Authentication Supporting Network Slicing for 5G-Enabled IoT. *IEEE Journal on Selected Areas in Communications* 36, 3 (March 2018), 644–657.
[9] NOKIA. 2017. Security challenges and opportunities for 5G mobile networks. https://onestore.nokia.com/asset/201049/Nokia_5G_Security_White_Paper_EN.pdf
[10] F. Reynaud, F. X. Aguessy, O. Bettan, M. Bouet, and V. Conan. 2016. Attacks against Network Functions Virtualization and Software-Defined Networking: State-of-the-art. In *IEEE NetSoft Conference and Workshops (NetSoft)*. 471–476.
[11] S. Sharma, R. Miller, and A. Francini. 2017. A Cloud-Native Approach to 5G Network Slicing. *IEEE Communications Magazine* 55, 8 (Aug 2017), 120–127.
[12] V. Thangam and K. Chandrasekaran. 2016. Elliptic Curve Based Proxy Re-Encryption. In *ACM Conference on Information and Communication Technology for Competitive Strategies (ICTCS)*. 121:1–121:6.