On blockchain based secure network coding for mobile small cells

Vipindev Adat, Ilias Politis, Stavros Kotsopoulos
Wireless Communications Laboratory
University of Patras, Greece
{vipindev, ipolitis, kotsop}@ece.upatras.gr

Abstract—The next-generation communication technology has to address a large number of devices demanding high quality of services. Network Coding seems to be a promising solution to achieve this in the mobile small cell environment. However, network coding itself comes with specific security challenges which affect the efficiency expected to achieve by employing network coding. Pollution attacks, where malicious intermediate nodes pollute the packets in communication, is one of the major hindrances to the practical implementations of network coding. Even though security schemes are proposed against pollution attacks, secure network coding enabled mobile small cell architecture needs a lot of advancements from the present state of the art. This paper investigates the existing security architectures and proposes a secure network coding architecture that incorporates blockchains to support efficient integrity schemes against pollution attacks.

I. INTRODUCTION

The fifth-generation and upcoming technologies in the communication domain have to address a variety of requirements. With the highly increased number of devices, the future networks are expected to have pico, femto and small cell environments to ensure a better quality of services [1]. Further, these networks are expected to have a much higher data rate with high resiliency and low latency. Network coding enables high data rate communication with resiliency in a cooperative network environment. With device to device communication, it is possible to harvest the maximum available bandwidth in the network and provide every user with a high quality of service. The application of network coding into the small cell environment has been well investigated [2] and looks promising to enhance the user experience in the near future. However, the security challenges arising with the application of network coding needs to be thoroughly addressed before its implementation. With the mixing of packets in the intermediate nodes, network coding is susceptible to specific attacks like pollution attacks which can result in drastic degradation of services. Further, the pollution attacks spread over the network very rapidly which demands the earliest possible detection and prevention of the polluted packets.

Network coding schemes secure against pollution attacks are being investigated for a long time [3]. Most of the existing schemes use homomorphic signatures or message authentication codes to ensure the integrity of the packets in transition. Also, there are information theoretic schemes and hybrid approaches to ensure secure network coding. However, considering the network coding enabled network of small cells, we need to tackle some more issues to ensure the proper security without significantly increasing the complexity of the system. Most of the homomorphic MAC-based schemes follow a shared secret key system and as the number of nodes involved increases, the complexity of the key sharing also increases. Further, when there are inter-cell communications, the system gets more complicated and needs to have keys shared between small cells. In this work we are proposing an optimized architecture for secure network coding enabled small cells (SECRET Small cells) based on some existing literature explained in the next section.

II. BACKGROUND WORK

The integrity schemes for network coding against pollution attacks can be broadly divided into information-theoretic and cryptographic approaches. For the mobile wireless networks employing random linear network coding (RLNC) [4], cryptographic schemes, especially homomorphic signatures or MAC-based schemes, have shown good performances. In this section, we discuss a few important homomorphic MAC and signature-based integrity schemes and two schemes specific to the mobile small cell environments.

In 2009, Agrawal and Boneh introduced an integrity scheme based on Homomorphic Message Authentication Codes (Homomac) to prevent pollution attacks in network coding [5]. This scheme was inspired by the classic MAC of Carter and Wagman [6], but also incorporate the homomorphic property to suit the network coding environment. The homomorphic property ensures that any node having a set of valid vector-tag pairs can create valid tags for a linear combination of vectors using the available tags. This property helps the intermediate nodes to receive different vector tag pairs and mix the vectors and also create valid tags for these encoded vectors. However, a shared secret key is required to do the verification of these tags. Homomac also uses a specific key distribution mechanism similar to [7] to facilitate verification of tags at the intermediate nodes. This proposed version of homomac is secure against pollution attack with a $c$-collison resistance, where $c$ is the number of neighboring nodes to be compromised simultaneously to break the integrity scheme.

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 722424.
Otherwise, security of this MAC-based integrity scheme can be beaten by two different scenarios: if the attacker could forge a valid tag for a non-existing vector or if it can define a second vector which matches with one of the valid tags for a different vector. However, these two conditions are practically difficult to achieve. The first condition is a computationally hard problem while the second possibility has a very less probability to achieve due to the large field size ($2^8$ in most of the cases) of the generation. Further, this probability can be reduced even further by adding more number of tags to the generation, but with an increased computational and communication overheads.

Homomac was followed by many integrity schemes using homomorphic MACs. One of the main challenges in these tag based integrity schemes was to address tag pollution attacks. In the MAC-based schemes, a receiver node will discard packets if they are not having a valid tag associated with this. However, this leads to this new problem of tag pollution attack where the adversary node doesn't corrupt the data packets but instead of sending a valid tag, it sends an invalid tag with the packet. This will lead to the discarding of the genuine packet due to a tag mismatch at the receiver. This reduces the efficiency of the scheme and it is called tag pollution attacks. MacSig [8] addresses this problem by using both homomorphic MACs and signatures. Authors explain the construction of secure homomorphic subspace MACs (HSM) and homomorphic subspace signatures (HSS) using the concept of padding for orthogonality, where the source pads the packet with an extra symbol/tag such that the subspace of this extra symbol is orthogonal to the specific vector of the packet. Further, they also proposed the double random key distribution scheme. This prevents the adversary from selecting any particular set of neighboring nodes to be targeted. In the proposed MacSig protocol, each receiver nodes are distributed with a random subset of keys by the source and the source itself choose random keys to create a number of tags for each generation (double random key distribution). A number of tags are created over the packets and a signature is created over these tags to prevent pollution attacks. While signing, the augmented coefficients are also considered to improve the security of the signature. Each node on the network could verify the signature using the public key of the source and then verify at least one MAC using the subset of keys it holds. This scheme has a better performance against previous cryptographic solutions in terms of computational and communicational overheads as well as prevents tag pollution attacks as well.

A dual MAC-based scheme was proposed by Esfahani et al. [9] in which the HMACs are used to ensure the integrity of data packets and another set of authentication codes called D-MACs are calculated over these HMACs to counteract tag pollution attacks. This proposal also used a c-cover free key distribution algorithm to resist a coalition of adversary nodes to take considerable advantage over the security. Dual MAC ensures that a pollution attack will be detected at a maximum distance of $c - 1$ nodes later unless $c$ neighboring nodes are compromised. However, still there existed a vulnerability of a dual tag pollution attack which the same authors addressed in a continuation of their work incorporating some concepts from MacSig. An efficient HMAC scheme was proposed in [10] where a signature is applied over the D-MACs to ensure the authenticity of the tags. However, this new scheme has reduced number of computations required to verify the tags compared to MacSig due to the c-cover free distribution. On the other hand, it has a probability of some malicious packets traveling some nodes until it finally gets detected. This could also lead to multiple packets getting affected. Further, since only a subset of tags is verified at each node, the security of the protocol is not the maximum that can be achieved with the same number of tags.

However, all these schemes were generic to network coding and didn’t fit properly to the future small cell requirements. The future networks are expected to be a highly dense heterogeneous network of mobile small cells. These small cells have to address the challenges of volatile networks where the devices are highly moving in and out of the network as well as between different small cells. The scalability of the network and making the pre-distribution of keys to be as simple as possible becomes primary concerns for the future small cells. Proceeding further from the state of the art integrity schemes, we are looking at the integrity schemes for network coding enabled SECRET small cells, shown in fig. 1. Since the whole idea of a future network involves low latency high resilient mobile network, the cryptographic approaches look like a better match to the system. Nevertheless, it should also consider the energy efficiency of the schemes. Thus we are trying to extend and modify the existing approaches to be more efficient and secure using the characteristics of the proposed architecture. We propose two initial integrity schemes which suit the proposed system architecture.

A. Secure SDN based SECRET Small Cells

As shown in the fig. 1, the future small cell environment is expected to be a collection of different small cells consisting of mobile users capable of device to device communication
supported by a software-defined mobile core network. Using this as a trusted central unit connected to all the participating users, a secure SDN based integrity scheme was proposed in [11]. This scheme considers the central unit as a trusted storage point and built a secure homomorphic MAC-based integrity scheme for the mobile small environment. This proposed scheme is again based on the mathematical principles proposed in [5] but enhances the security and scalability of the approach. Further, the key distribution is made much simpler as all the nodes could possess all the keys and still successfully secure the system against collision of adversaries. This scheme is explained in four different parts.

1) Setup: The prerequisites of the integrity scheme and the basic details of the architecture are explained in this first part. In RLNC, practically the packets are grouped to generations and sent together. Each packet \( P_i \) can be considered as a vector of \( n \) elements. Homomorphic MACs are created and verified using keys shared between the source and sink nodes. These keys need to be securely shared between all the participating nodes in the network and it can be performed even before the communication starts. A Key Distribution Center (KDC) is responsible for this. It can be the trusted central unit acting as the KDC. This KDC distributes a set of \( L \) keys to every participating node where each key \( K_i \) is of size \( n + 1 \). Further, each node creates its own private public key pair along with KDC and the KDC also shares the public key of every node to the network. The shared symmetric keys are used for creating MACs and the source nodes use their private keys to sign the packets so that the receiving nodes can verify the authenticity using corresponding public key.

2) Tag generation: In the proposed scheme, tags are created only at the source node on native packets. Once these tags are created, the native packet along with its tags is considered for encoding. This reduces the required key size as well. Tags are generated using the same mathematical equation in [5] as:

\[
Tag_{i,l} = \frac{\sum_{j=1}^{P_i,j \times K_l,j}}{K_{l,j+1}}, \quad l \in (1, L) \text{ & } j \in (1, n)
\]  

(1)

These \( L \) tags are attached to each packet. Further, the tags for every generation are also communicated to the central unit directly by the source node along with its signature.

3) Verification: When a generation of coded packets is received at a node through the communication channel, it initiates a two-step verification process. Initially, it verifies whether the tags received corresponds to the packets they are attached with using the shared secret key. For the second step of verification, these nodes will retrieve the tags stored in the central unit corresponding to the received generation and matches them to the tags received via communication channel. The first step of verification checks for data pollution attacks while the second step ensures the tags are not polluted during the transition. The verification algorithm is shown below. If the generation passes both verification process, then the packets are either re-encoded or decoded to get the original message.

4) Re-encoding: Re-encoding in this scheme is very much similar to simple RLNC protocol. The intermediate nodes re-encode verified packets from the generation. Intermediate nodes do not recalculate the tags but instead consider them as a part of the packet and generate the recoded packets with the help of their locally generated coefficients.

\[
Algorithm 1: \text{Verification Algorithm}
\]

| Data: Received packet \( C_i \), \( L \) tags corresponding to \( C_i \) retrieved from the central unit, Key set \( K_s \) |
| Result: 1 if verification is successful and 0 if verification is failed. In case of a failed verification, the type of the attack is also reported. |

Step 1:
Retrieve the coefficient matrix from the received packet

Step 2:
Multiply the tags retrieved from the central unit with the corresponding coefficients

Step 3:
Compare the tags with those appear in the received codeword.

if they don’t match then
  Report Warning and Proceed
else
  Proceed

Step 4:
Create tags for the received packet using MAC algorithm (without considering the coefficient part)

Step 5:
if MAC algorithm output matches with the tags retrieved from central unit then
  1 \( \leftarrow \) Return
else
  0 \( \leftarrow \) Return

However, using the central unit as an integral part of the security scheme introduces the issue of single point of vulnerability. If the central unit is compromised by the adversaries or denied access, then the integrity scheme does not work properly. Further, it also requires every node to have a secure channel with the central nodes to share tags. This again limits the scalability of the system. To overcome these challenges, a distributed blockchain-like ledger is proposed in [12] instead of the central controller.
B. Blockchain-based Integrity Scheme for SECRET Small Cells

In [12], a distributed ledger is used to store the tags instead of a single central controller. This blockchain like architecture ensures that there is no central point of vulnerability. Further, the blockchain is considered highly secure against forging of the registered blocks. This ensures that once the source node makes an entry of the tags attached to a generation to the blockchain, once that block is registered on the chain it is always available for all other nodes and cannot be altered by an adversary. The verification algorithm remains the same as in [11] but instead of the central unit, a blockchain is used to store and share tags with all the participating nodes. The architecture of the blockchain is shown in fig. 3. Here the source node generates tags as per the eq. 1 and the set of tags for each generation will be shared to the blockchain as candidate block, along with the generation number and identification details of the source and destination nodes. The candidate blocks are verified using a Proof of Stake (PoS) algorithm and added to the chain. As per the blockchain property, once a block is verified and being part of the blockchain ledger, it is very hard to alter it. This ensures that the original tags will be available to all nodes connected to the ledger at any time after the verification. When an intermediate node receives a generation of packet via communication channel, it will verify the authenticity of tags by comparing it with the tags stored in the blockchain corresponding to the particular generation.

Considering the situation of an adversary being present in the intermediate nodes, this integrity scheme proves to be secure against a very strong adversary model. This scheme expects the adversary to have access to the complete information and packets received at the compromised node. This also includes the complete set of shared secret keys used to create the tags at the source. However, the adversary will not have access to the private key of the source node. Further, if we consider the compromised node receives enough number of packets to decode them, then the adversary will also have access to the native packets so that it can create completely new tags of the packet. However, this adversary cannot alter an existing block in the chain which ensures that even if the adversary makes new tags which are valid for the packets, still it will be detected as a tag pollution attack in the next genuine node during the comparison of tags received through the communication channel and retrieved from the blockchain. This proposed integrity scheme is comparable with already known and secure schemes like MacSig [8] and Dual HMAC [10] in terms of performance parameters. This new blockchain based scheme has reduced communication and computational overhead compared to previously proposed integrity schemes. Further, the key size reduced compared to the existing approaches. At the same time, this scheme requires a blockchain ledger to be maintained and operated between the complete set of nodes.

However, we realize the attributes of these two schemes needs to be combined and modified to get the optimal system architecture for inter-cell communication of SECRET small cells. Otherwise, if we need to go for a completely blockchain-based architecture, each node in the network will be part of the blockchain which will increase the cost of computations and maintaining the blockchain. On the other hand, a completely centralized network has its own issues as mentioned before. Thus we are proposing this hybrid scheme for secure SECRET small cells.

III. PROPOSED BLOCKCHAIN BASED ARCHITECTURE FOR INTER SMALL CELL COMMUNICATION

Building a completely centralized system for secure sharing of MACs requires all the participating nodes to be connected to the central controller. In case of an inter small cell communication, it needs all the nodes in the network to connect to a single central entity. This leads to the single point of vulnerability as mentioned in the previous section as well as a complex system of communication channels. On the other hand, if we deploy the blockchain structure, it involves all the
individual nodes to be part of the blockchain requires all these nodes to be capable of storing the blockchain and participating in the block verification process. It demands the nodes to meet storage and computational requirements to maintain the blockchain. Considering both situations, having a central unit or having a blockchain involving all nodes to be actively and equally capable of verification and storage of the complete ledger could not be a practical solution. Further, it will also limit the scalability of the network. A hybrid model such that a centralized system inside the small cell and then a blockchain of small cells to share the MACs between the small cells can be used to address these challenges. Thus the end nodes will share the MACs generated by themselves only with their small cell head and the small cell heads form the blockchain to share the MACs in case of inter small cell communications.

This blockchain network of small cells forms as an overlay over the whole network as shown in fig. 4. Each small cell head or hotspot will be the participating node in the blockchain. They are also more powerful in terms of computations and storage compared to the end nodes. Now let us consider a scenario to understand how the MACs are shared during the inter small cell communication. When a node from a particular small cell starts the communication, it generates the packets and its corresponding tags according to eq. 1 and attaches it to the packets. Further, the node communicates this tag to the small cell head as in the centralized system mentioned in [11]. Thus the small cell head receives the information about the newly generated packets and tags. If the destination node is not in the same small cell, then the small cell head creates a new transaction entry including the details of source and destination nodes. It is also possible that more than one node initiates a communication during the same time or in a small time frame in which the candidate block is created. In such cases, more than one transactions will be part of the candidate block. In practical systems, this has a very high probability. The candidate block is verified by a proof of stake algorithm as mentioned in [12] to ensure that no energy is wasted due to the mining process. This blockchain-like distributed ledger is based on bigchainDB [13]. BigchainDB is a distributed database which has the properties of blockchain with high throughput and low latency. It employs a byzantine fault tolerance algorithm which in turn is a PoS approach to achieve consensus. Further, bigchainDB also provides a powerful query service using the metadata associated with the block. This enables even the end nodes to check and access the tags stored in the blockchain whenever required, even without having the blockchain running on them. The end nodes do not have to store the blockchain or perform the verification of blocks. The blockchain is stored over the small cell heads. However, whenever required, the end nodes can access the
TABLE I: Transaction Format

| Field                | Description                                      |
|----------------------|--------------------------------------------------|
| Generation Number    | Number of the generation                         |
| Transaction Hash     | A hash of the transaction                        |
| Source node address  | Address of the source node                       |
| Destination Small Cell | Address of the destination small cell          |
| Keys                 | Keys used by source nodes                        |

required blocks with the query functionality. The generation number and source ID can be used as the metadata to retrieve the particular blocks. Format for the transaction is set as shown in the table I to facilitate this.

IV. CONCLUSIONS

The SECRET small cells ensure high quality of service and resiliency because of the network coding backbone. However, these improvements in QoS could be easily degraded by malicious users with pollution attacks. Securing the network against pollution attacks become a prime concern to harness the best out of the network coding enabled mobile small cell environment. In this work, we propose a blockchain network of small cells to ensure secure network coding in a small cell environment. This blockchain structure supports a homomorphic MAC-based integrity scheme against pollution attacks. Further, to avoid overburdening the end nodes, the blockchain network only comprises the small cell heads and the end nodes connects to the small cell nodes as a central controller to communicate the tags. This architecture helps to reach a balance between the number of communication channels and storage and computational overhead on the end nodes for the integrity schemes and thus ensuring the maximum benefit of SECRET small cells.

REFERENCES

[1] C. Tselios and G. Tsolis, “On QoE-awareness through virtualized probes in 5G networks,” in 2016 IEEE 21st International Workshop on Computer Aided Modelling and Design of Communication Links and Networks (CAMAD), Oct 2016, pp. 159–164.

[2] J. Rodriguez, A. Radwan, C. Barbosa, F. H. Fitzek, R. A. Abd-Allameed, J. Noras, S. M. Jones, I. Politis, P. Galiotos, G. Schulte et al., “Secretsecure network coding for reduced energy next generation mobile small cells: A european training network in wireless communications and networking for 5g,” in 2017 Internet Technologies and Applications (ITA). IEEE, 2017, pp. 329–333.

[3] N. Cai and R. W. Yeung, “Secure network coding,” in Proceedings IEEE International Symposium on Information Theory, 2002, pp. 323–.

[4] T. Ho, M. Mèdard, R. Koetter, D. R. Karger, M. Effros, J. Shi, and B. Leong, “A random linear network coding approach to multicast,” IEEE Transactions on Information Theory, vol. 52, no. 10, pp. 4413–4430, 2006.

[5] S. Agrawal and D. Boneh, “Homomorphic macs: Mac-based integrity for network coding,” in International Conference on Applied Cryptography and Network Security. Springer, 2009, pp. 292–305.

[6] J. L. Carter and M. N. Wegman, “Universal classes of hash functions,” Journal of computer and system sciences, vol. 18, no. 2, pp. 143–154, 1979.

[7] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, “Multicast security; A taxonomy and some efficient constructions,” in IEEE INFOCOM’99. Conference on Computer Communications. Proceedings. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. The Future is Now (Cat. No. 99CH36320), vol. 2. IEEE, 1999, pp. 708–716.