Secure and Seamless Payment for Wireless Mesh Networks

Albert Levi, Serhat Can Leloglu

Sabanci University, Turkey

levi@sabanciuniv.edu

canleloglu@sabanciuniv.edu

Abstract— Wireless Mesh Network (WMN) technology is a multi-hop high-speed networking technology for broadband network access. WMNs are easy to deploy and cheap comparing to base stations. It is highly desirable to build and use a secure payment system over WMNs to provide network access to mobile or stationary clients. In this paper a secure and seamless way of pre-payment for internet access is proposed and networks simulations for this system are shown.

Keywords— **Wireless** Mesh Networks, Cryptography, Payment Systems, Security, Network Simulation

1. Introduction

Wireless Mesh Networks [1] are very suitable systems for service providing; moreover a system built by using WMNs should support user identification, authentication as well as authorization and accounting.

In common payment systems service providers do not fully trust clients, but in reality service providers –intentionally or not- may over charge the clients or charge for services that they did not provide. Using native cryptographic algorithms it is proven that every action could have an undeniable cryptographic proof so that the system achieves mutual trust.

The designed secure and seamless pre-payment system has the properties such as wide-coverage, seamless mobility and roaming, anonymity, mutual authentication, two-way honesty, preventing double spending and unlinkability. 10 protocols are designed for system entity actions in the system, and these protocols are tested using network simulator 3 [1]. The designed system had successful results in unit tests and these results are presented in this paper too.

1. Cryptographic Notes

Using some cryptographic primitives such as public key cryptosystems and hash functions forms up the designed protocol. 2048-bit RSA [2] is used for public key encryption-decryption and signature purposes. AES-128 [3] is used for symmetric key cryptography and SHA-256 [3, 4] is used as a hash algorithm in the system. HMAC [3, 4] algorithm is used for challenge-response protocols.

1. General Overview of Proposed Scheme and System Entities

The proposed system is a secure pre-payment infrastructure for WMNs that also considers users' privacy and fairness. In this infrastructure there are users as well as tools that are used for service providing. Table 1 gives a list of system entities that function in the proposed system.

TABLE I  
System Entities

|  |  |
| --- | --- |
| Description: D:\My Documents\albert\tt proje\phone.png | Mobile user (client) |
| Description: D:\My Documents\albert\tt proje\accessPoint.png | Access Point (AP) with mesh routing capability. From now on in this document, it is called as AP, but please note that it also has routing capability. |
| Description: D:\My Documents\albert\tt proje\cloudWithoutDots.png | Mesh backbone of the operator |
| Description: D:\My Documents\albert\tt proje\gateway.png | Gateway (GW) that connects the mesh backbone to outer world and also to the operator's server |
| Description: D:\My Documents\albert\tt proje\operator.png | Operator's server (OP). Keeps necessary logs and user info. |
| Description: D:\My Documents\albert\tt proje\trustedThirdParty.png | Trusted Third Party (TTP). Payment related logs are mostly to be generated by the TTP. |

Figure 1 shows the topology of the network and connections between entities.



Figure 1. Network Topology

Connection between serving access points is wireless, and they use 802.11s protocol [5]. This mesh backbone is like a cloud from the mobile user’s perspective. It is a black box; which receives packets from mobile user and delivers them to the gateway in a multi-hop manner. Mesh backbone uses HWMP (Hybrid Wireless Mesh Protocol) protocol [6], which is a hybrid routing protocol, which uses routing tables.

Connection medium between mesh backbone and gateway (GW) is wireless. Gateways and operators communicate through wired connection. The connection between an operator and TTP is wired also. Wired connections use 802.3 (Ethernet protocol).

1. Connection Card Structure

*Connection Card* is the main deed that clients buy from operators and use to get Internet service. Connection cards include credits as tokens. Hash tokens are generated using hash chains as discussed below. Connection cards also have unique Serial Numbers (), which are to be used for alias computation later.

Tokens for getting Internet service are basically links in a hash chain. For each set of tokens, the operator picks on a random IV (Initialization Vector) and takes hashes of it many times. The number of hash operations is actually the number of tokens in a set.

…

is the first token to use. Then we use the token in the increasing order of token index. In this way, we exploit one-way property of hash algorithms such that an attacker cannot learn the next token even if he knows the previous ones.

1. Alias Computation

Aliases are temporary identifiers for clients. Aliases change frequently using a secure protocol. By changing the aliases frequently, anonymity is achieved in the system to some extent.

Serial number (SN) of the connection card, which is bought from an operator, will be used as a base for client’s aliases. An alias will be computed by performing following operations:

1. Client will pick a random 128-bit unsigned number and call it his nonce .
2. Perform XOR operation with and his nonce,
3. Client will use this alias whenever his identity is required.

One may argue that this kind of alias computation would have a risk of causing same alias for several users. Making TTP to check proposed alias to be a unique alias at a point of time solves this problem.

1. Protocols

There are 10 protocols to make this system work. These protocols define packet transfers and routes. Cryptographic primitives and the way they are used are also explained in the protocol designs.

Some protocols show similarity in between. *Initial Authorization* and *Reuse of a Connection Card*. The only difference between these two protocols is hash token number. *Initial Authorization* uses the very first hash token on the other hand *Reuse of a Connection* Card uses the other hash tokens on the hash chain. This kind of similar protocols will be explained simultaneously.

* 1. End-to-End Two-Way Protocols

End-to-End two-way protocols are main and the most common protocols in the system.

The protocols classified as end-to-end two way are *Initial Authorization, Reuse of a Connection Card, Disconnection, Change Alias* protocols. These protocols transmit same sized packets from client to TTP. TTP does same cryptographic operations on the packet and forwards the packet to the client. In this protocols client performs an encryption over a 384-bit packet using RSA-2048 sends it to the TTP. TTP decrypts this cipher using RSA-2048 private key and signs 256-bit data using RSA-2048 private key. TTP sends this signed data to client through mesh backbone. Every station receiving the encrypted and signed packet; verifies the signature and forwards the original packet until it reaches the destination.

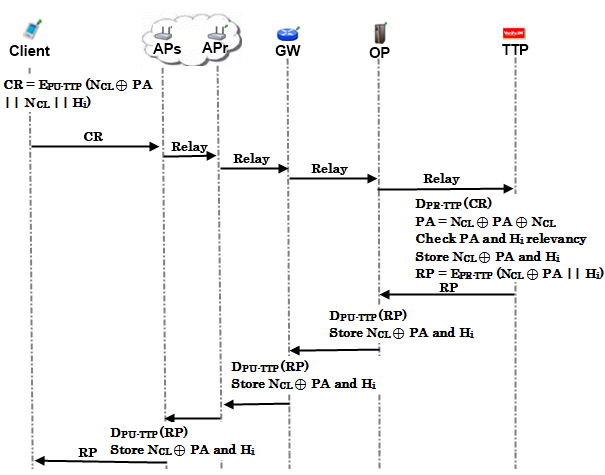


Figure 2. End-to-End Two Way Protocol Flow

Initial Authorization is the first protocol that a user uses in the system in order to get authorized. It is used only once by a particular user.

Reuse of a Connection Card protocol is used when a user does not finish the tokens in a connection card and would like to use the remaining tokens at a later time. Initial Authorization and Reuse of a Connection Card protocols only differ in packet content and even that content is very similar.

The beginning time of a session for a user is stored when a user performs one of the previously told two protocols. Disconnection protocol yields the ending time of the session. In this way, the TTP learns the amount of time that the user got served. This information is used for settlement purposes.

One of the privacy preserving features of the proposed system is that access points ask every user to change their aliases from time to time. When received such a command from the access point, clients compute a new alias and send it to the TTP for signature. The overall process is called Change Alias protocol.

* 1. Access Point Authentication



Figure 3. Access Point Authentication

Access Point Authentication, which is shown in Figure 3, takes place between a mobile client and an access point. It is a challenge-response type of protocol to authenticate the AP to the client.

Access Point Authentication starts with an access point sending a request to the client. Client sends a 128-bit challenge to the access point. Access Point performs a HMAC on this challenge using the last hash token as a key. Client performs the same operation and compares two results. If they match, access point is verified as authenticated.

* 1. Packet Transfer

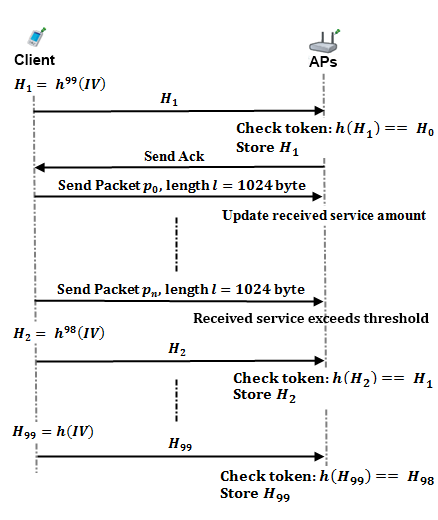


Figure 4. Packet Transfer

Packet transfer, shown in Figure 4, protocol is the mostly used and simplest protocol among the other ones. It is the main service access protocol using tokens one by one. One token of the hash chain is sent from client to AP and the client starts to use broadband access service. Usage is charged in time basis. Every 5 minutes client sends a new hash token to continue to get Internet service. When a user sends a hash token it means that he/she already paid for the service and if disconnection protocol is called after e.g. 2 minutes, user could not get a refund for remaining 3 minutes.

* 1. Update Packets

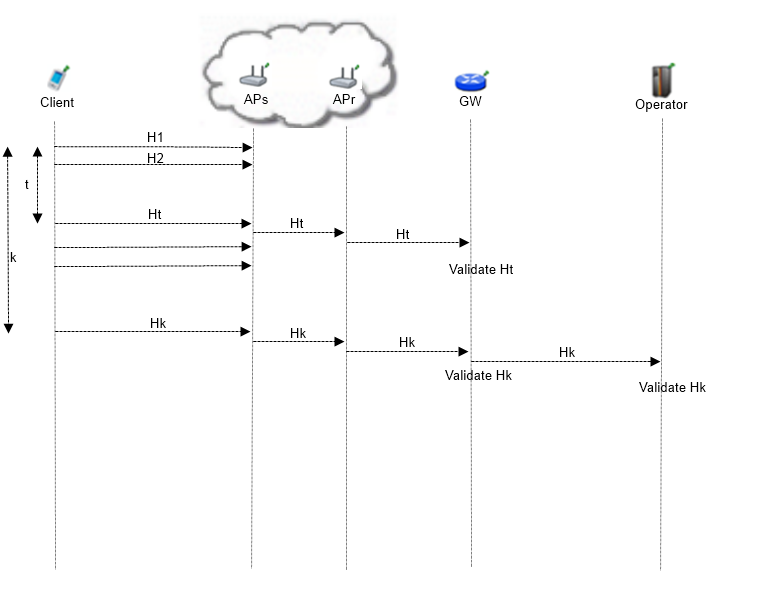


Figure 5. Update Packets

Update Packets protocol, shown in Figure 5, is used in case of an unexpected behaviour in network. If a client drops out of the network, operators and TTP need to know that this client is not active anymore. In order to handle this unexpected behaviour, access points periodically update operators using Update Packets protocol.

In this protocol, client sends concatenation of 128-bit alias and 128-bit hash token to the operator. Operators update TTP in case of a drop. This protocol is a one way end-to-end protocol.

* 1. Seamless Mobility and Roaming (Payment Related)



Figure 6. Seamless Mobility and Roaming

Seamless Mobility and Roaming protocols, shown in Figure 6, is run whenever client changes serving access point. The running protocol is called Seamless Mobility if the new access point belongs to the same operator as previous access point. If the operators differ then the protocol is called Seamless Roaming.

In this kind of protocols client sends a 384-bit request packet to old access point. Old access point receives this packet and performs an encryption on it using RSA-2048, than signs this cipher text using RSA-2048 private key. Old access point sends this packet to client and client relays it to the new access point. New access point decrypts the package using RSA-2048 private key and verifies the signature using RSA-2048 public key.

Finally new access point and the client run a Challenge-Response protocol to authenticate new access point.

If the running protocol is Seamless Roaming, then after receiving break-off request from the client, old access point sends a disconnection request to the TTP. This part of the protocol is not important for the unit tests because it runs in the background.

* 1. Distributing Access Point Public Keys



Figure 7. Distributing Access Point Public Keys

Achieving seamless mobility in home operator and also to support seamless roaming, a public key distribution mechanism is placed within the system.

In Figure 7, a generic model for public key distribution is shown. This protocol has two parts; one is certificate generation for access point public keys, the other one is distribution of the public keys. The part between operator and the TTP is offline. This part of the protocol runs during set-up, before the deployment of the access points in the field.

1. Unit Test Results

Unit tests cover protocol behaviours under low pressure. In these tests there is only one user, and this user performs the same protocol every minute. These tests are done to ensure that a module of the system, in this case protocols, are fit for use.

As discussed earlier some protocols show similarity considering packet sizes and packet routes. Since there would be no difference between unit tests of protocols that are in the same group, there is one result chart for a particular group of protocols.

* 1. Results for End-to-End Two-Way Protocols

As discussed earlier end-to-end two-way protocols deliver same sized packets from clients to TTP and vice-versa. The cryptographic operations and their turns are the same between these protocols.

Unit tests for this kind of protocols consist of a user, running the same protocol every minute. Charts present the average delay of the packet delivery over time. In this simulation the user sends the packet to a serving access point and the packet hops 2 times in the mesh backbone until it reaches the gateway. Gateway forwards the packet to operator and operator transmits the packet to TTP. TTP processes this packet and sends it back to the client through the same route.

Figure 8 gives the result for unit test of end-to-end two-way protocols.

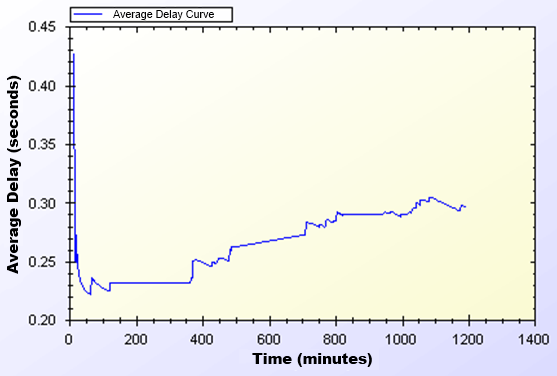


Figure 8. End-to-End Two-Way Protocol Unit Test Result

As shown in Figure 8, there is a delay that shows variation between 0.43 and 0.25 second. This unstable behaviour is caused by different initial packet delays. System needs some packets to set up paths between mesh nodes. The performance stabilizes in time. Steady state situation scores a delay of 0.3 second for end-to-end communication for above-mentioned protocols.

* 1. Results for Access Point Authentication

Access Point Authentication protocol, consists of a challenge-response protocol. It contains two HMAC operations.

Unit test for this protocol contains a user, trying to run access point authentication protocol with a serving access point every minute. The resulting chart, presented on Figure 9, shows the average delay of the protocol versus time.



Figure 9. Access Point Authentication Unit Test Result

As shown in Figure 9, average delay of access point authentication converges to 0.05 second in the steady state. The initial delay values are higher than the later ones, because nodes need some time to establish and see who is around. At the time of initial deployment, wireless nodes send and receive beacons and perform operations using them.

* 1. Results for Seamless Mobility and Roaming

*Seamless Mobility* and *Seamless Roaming* protocols have the same behaviour since client sends and receives same length of packets. Thus, they are grouped together for unit tests.

Unit test for Seamless Mobility and Seamless Roaming protocols consists of a client changes serving access point every minute. Client is located in between two access points and these access points are both eligible for service. Since these protocols must be seamless to the user it is important to get reasonable delays for these protocols.

Figure 10 presents the unit test result for Seamless Mobility and Roaming protocols.



Figure 10. Seamless Mobility and Roaming Unit Test Result

In unit test for these protocols, a 0.15 second of network delay for access point change is observed. Similar to other protocols, there is a transitive period at the beginning of the simulations, however it reaches steady state in time and gains balance.

* 1. Results for Packet Transfer

Packet transfer is the mostly used protocol in the system. It is crucial to have small amount of network delay for this protocol because of it’s often use. Packet transfer unit test scenario is that a client sends a 512-byte packet every minute.

Figure 11 shows the unit test result for Packet Transfer protocol.



Figure 11. Packet Transfer Unit Test Result

Unit test gave a higher average delay value at the early parts of the simulation but expectedly it reaches a balance through time. As seen on Figure 11, at steady state, packets are received in a very short amount of time, which is around 0.0002 second.

1. Conclusion

In unit tests, standalone performances of the protocols under trivial usage scenarios are analysed. Unit tests set an example for how the system will behave in empty hours. In this way, the first proof-of-concept implementation of the system is provided and showed that the designed protocols reach steady state and reasonable performance in time.

Results are very important since the actual usage of the system is a combination of these protocols. Unit tests show that the proposed system is a considerable and an effective pre-payment system.

References

1. Akyildiz, I. F., Wang, X., and Wang, W. (2005) Wireless mesh networks: a survey, *Computer Networks and ISDN Systems*, 47(4): 445-487.
2. Network Simulator 3 Official Web Site: <http://www.nsnam.org/>
3. Rivest, R., Shamir, A., and Adleman, L. (1978) A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, 21(2): 120–126
4. FIPS PUB 197 (2001) Announcing the Advanced Encryption Standard (AES), <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>
5. Trappe, W., and Washington, L. (2006) Introduction to cryptography with coding theory, *Person Education, Inc*
6. Joseph D. Camp and Edward W. Knightly The IEEE 802.11s Extended Service Set Mesh Networking Standard
7. Kai Yang, Jian-feng Ma, Zi-hui Miao (2009) Hybrid routing protocol for wireless mesh network, Computational Intelligence and Security – CIS ’09