International Journal of Informatics and Communication Technology (IJ-ICT)
Vol. 4, No. 1, April 2015, pp. 13~22
ISSN: 2252-8776

Investigational Security Structural Design Using Client and Server Ambitious Protocols for WMN

Dr. S. Selvakani*, N. Suresh Kumar**
* Professor and Head, Departement of Computer Applications, Francis Xavier Engineering College, Tirunelveli, Tamilnadu, India
** Departement of Computer Applications, Francis Xavier Engineering College, Tirunelveli, Tamilnadu, India

Article Info

Article history:

Received Nov 12, 2014
Revised Feb 20, 2015
Accepted Mar 26, 2015

Keyword:

Authentication
Extensible
Protocol
Security protocol
WMN's authentication

ABSTRACT

Wireless mesh network plays an vital role in our day to day life, Networks are mainly used to deliver voice, video and data but using WMN's we can communicate in outdoor environments without wired. A wireless mesh can deliver network capacity, reliability and security along with the flexibility. Nowadays the WMN's are used by municipalities, public safety agencies, port authorities, and industrial organizations to connect to their workers and constituents. However, many technical issues still exist in this field. In order to provide a better understanding of the research challenges of WMNs, In this paper we propose MobiSEC, a complete security architecture protocols and algorithms for WMNs that provides both access control for mesh users and routers as well as a key distribution schemes that supports layer-2 encryption to ensure security and data confidentiality of all communications that occur in the WMN.

Corresponding Author:

Dr. S. Selvakani,
Professor and Head,
Departement of Computer Applications,
Francis Xavier Engineering College, Tirunelveli, Tamilnadu, India.
Email: sselvakani@hotmail.com

1. INTRODUCTION

A mesh is a multi-path, multi-hop wireless local area network (WLAN) and wide area network (WAN) that is ideal for outdoor deployment. With the help of wireless network one can communicate anywhere without the cost and disruption of running cabling or fiber.

The organizations can combine separate voice video, data networks onto a single network which is simpler to manage and operate. Only few devices are required to maintain and so the network cost is less expensive.

A mesh is quickly recoverable and low maintenance. It automatically finds the best route path through the network and operates smoothly even if a mesh link goes down or a node fails. Because the network is self-forming and self-healing, administration and maintenance costs are lower. In addition, a wireless mesh overcomes the line-of-sight issues that may occur when a space is crowded with the buildings or industrial equipment.

Wireless mesh networks (WMNs) have come out into view recently as a technology for next-generation wireless networking [1, 2]. The network nodes in WMNs, named mesh routers, provides access to mobile users, like access points in wireless local area networks, and they relay information hop by hop, like routers, using the wireless medium. Mesh routers are usually fixed and do not have energy constraints. WMNs, like wired networks, are characterized by infrequent topology changes and rare node failures.

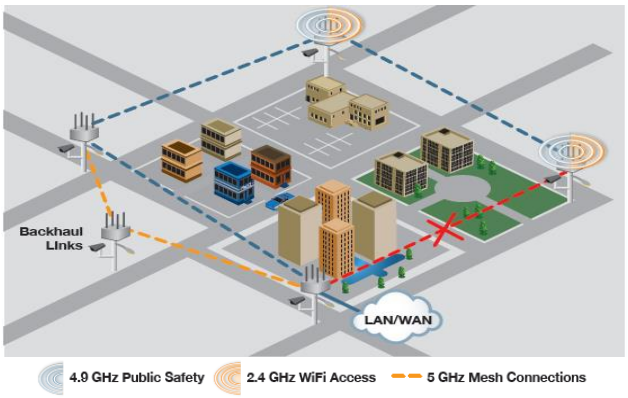

Figure 1. Wireless Mesh Network

The two different security areas are: one related to the access of users terminals (user authentication and data encryption), and the other related to network devices in the backbone of the WMN (mutual authentication of network devices, and secure exchange of data and control messages).

In this paper we propose MobiSEC, a novel security architecture for wireless mesh networks that provides a complete security framework for both the access and backbone areas of the WMN; that is, access control for end-users and mesh routers as well as security and integrity of all data communications that occur in the WMN. This is achieved with layer-2 encryption that uses a shared key whose delivery is assured by two key distribution protocols.

MobiSEC extends the IEEE 802.11i [14] standard to the WMN scenario, exploiting the routing capabilities of wireless mesh routers. A two-step approach is adopted: in the first step new nodes perform the authentication process with the nearest mesh router, according to the 802.11i protocol, like generic wireless clients. In the second step, these nodes can upgrade their role in the network, becoming mesh routers, by further authenticating to a central server, obtaining a temporary key with which all traffic is encrypted. We propose two key distribution protocols tailored for WMNs, named Server and Client Driven. In the Server Driven protocol, all mesh routers periodically send a request to a central server (the Key Server) to obtain a new key list, whereas in the Client Driven protocol the mesh routers obtain from the server a seed and a hash function type to generate the cryptographic keys with a scheme similar to the hash-chain method. Both protocols require a mutual authentication based on certificate exchanges between the mesh router and the server.

The WMN's important feature is the wireless technology used by nodes to form backbone architecture. Furthermore, Mobi-SEC allows seamless mobility of both mesh clients and routers. Client mobility is allowed by the 802.11i implementation, to which our solution is compliant, whereas mesh routers can roam freely around the backbone network after getting the key material from the Key Server, since all other mesh routers create the temporary key using the same information. The proposed solution has been implemented and integrated in MobiMESH [15], a WMN experimental platform that provides a complete framework for analyzing, studying and testing the behavior of a mesh network in a real-life environment. Furthermore, we extended the Network Simulator (ns v.2) [16] implementing the MobiSEC architecture, and performing extensive simulations in large-scale network scenarios to test the behavior of our architecture also in the presence of a large number of nodes and traffic flows.

We measured the performance of MobiSEC in several realistic network scenarios and we compared it both with a static approach that consists in using a fixed key to protect the WMN, as well as with an end-to-end solution that consists in establishing an encrypted IPSec tunnel. The first approach provides an upper bound in terms of achievable throughput, delay and packet losses, while the latter is useful to gauge the performance gap between our proposed architecture and existing end-to-end security solutions. Numerical results show that MobiSEC considerably increases the wireless mesh network security, with a negligible impact on the network performance, thus representing an effective solution for wireless mesh networking.

The main contributions of this paper are as follows: The proposition of MobiSEC, a complete security architecture for both the access and backbone areas of a WMN; The integration of the proposed solution in the experimental platform MobiMESH;A thorough evaluation of the proposed architecture in several realistic network scenarios.

2. RELATED WORKS

There are two main security areas in WMN they are, first related to the access of client terminals and the second is related to mesh backbone, using standard techniques [14,17,18] like MAC, 802.1x, 802.11 the client can authenticated which guarantee a higher flexibility and transparency. The mesh user can connect to the routers without any change to their device and application software. To secure information in mesh network cryptographic techniques are used while transferring data through a wireless network. In [21] the author use PANA (Protocol for carrying Authentication for network access), to authenticate the client user to exchange information through a wireless network. However this causes a severe problem for security so the key distribution techniques can be devised [13, 19, 20].

Other approaches have been proposed to authenticate the users in WMNs, maintaining at the same time a low overhead. In [21] security architecture for high integrity multi-hop WMNs is proposed; a heterogeneous set of WMN providers is modeled as a credit-card based system so that each mesh client does not need to be bound to a specific operator, but can achieve ubiquitous network access by first obtaining a universal pass issued by a trusted third broker. Such an approach is suitable for WMNs managed by multiple operators, whereas in this paper we are interested in a scenario where a single operator manages the WMN and is liable for all the authentication procedures.

Our proposed architecture extends the Diffie–Hellman key exchange protocol for negotiating a key that authorizes a user to access the backbone network services provided by a mesh router situated in a different zone.

3. SECURITY ARCHITECTURE

In this section we describe security architecture to provide both client and backbone security in a wireless mesh network. Client security is guaranteed using the standard 802.11i protocol, while backbone security is provided with a two-step approach: each new router that needs to connect to the mesh network first authenticates to the nearest mesh router exactly like a client node, gaining access to the mesh network. Then it performs a second authentication connecting to a Key Server able to provide the credentials to join the mesh backbone. Finally, the Key Server distributes the information needed to create the temporary key that all mesh routers use to encrypt the traffic transmitted over the wireless backbone.

MobiSEC is independent from the underlying cipher technique adopted. In the numerical evaluation, however, we used WEP [14] for two reasons: on the one hand it is the only cipher technique available for commercial wireless cards in ad hoc mode, which allowed us to implement MobiSEC in the MobiMESH test bed; on the other hand the utilization of WEP permits the robustness of the proposed solution to be proved, even in the presence of a weak cryptographic system. We are currently implementing the CCMP algorithm for the IBSS operating mode [14], which is used by several. Mesh implementations to establish the backbone links and form a multi-hop wireless architecture.

3.1. Client Security

To achieve the highest possible level of transparency, the access mechanism to the wireless mesh network is designed to be identical to that of a generic wireless LAN, where mobile devices connect to an access point. Since almost every wireless device currently available on the market implements the security functionalities described in the IEEE 802.11i protocol [14], we propose to configure mesh routers to comply with such standard. This solution allows users to access the mesh network exploiting the authentication and authorization mechanisms without installing additional software.

In Figure 2, A and B exchange data which are connected to wireless mesh routers N1 and N2 .If such wireless network is not protected by any security system, M will be able to drop the communication, since nodes N1 and N2 will forward the traffic on the wireless link on which M is listening. This situation is prevented by MobiSEC, which encrypts all the traffic transmitted on the wireless link with a stream cipher operating at the data link layer.

Figure 2. A and B exchange data through the WNS's

3.2. BackBone Security

The client security solution illustrated above provides confidentiality and integrity of the information transmitted only on the wireless access link. Therefore, we propose an additional system to secure communications that occur over the wireless backbone. A two-step approach is adopted, in which new nodes dynamically join the network as wireless clients and subsequently can upgrade their role, becoming wireless mesh routers by further authenticating to a Key Server.

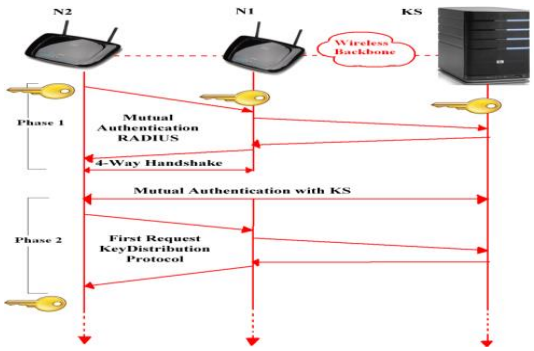

Figure 3. Phases of the connection process performed by a new mesh router (node N2). The depicted keys are used to encrypt backbone traffic

Two major problems arise: on the one hand it is necessary to authenticate new mesh routers that join the network and provide them with the cryptographic material needed to derive keys that make secure data transfer possible. On the other hand, it is important to develop a system with a minimal impact on device mobility. To this end, we designed and implemented a key distribution solution that exploits the existing access network, allowing a new node to connect to a remote server which sends the temporary key used by all mesh routers to encrypt the traffic transmitted over the wireless backbone. Such key represents proof that the new node has the required credentials to become a mesh router.

Figure 3 shows the phases of the connection process performed by a new mesh router (namely, node N2). Note that we illustrate only the most important messages exchanged between the network entities during the authentication process, while the whole procedure is detailed in the following. When N2 wants to connect to the mesh network, it scans all radio channels to detect a mesh router already connected to the wireless backbone, which is therefore able to provide access to all network services (including authentication and key distribution). Let N1 be such router. After connecting to N1; N2 can perform the tasks described by the IEEE 802.11i protocol to complete a mutual authentication with the network and establish a security association with the entity to which it is physically connected through the execution of the 4-Way Handshake protocol (phase 1). In other words, during this phase N2 performs all the activities as a generic wireless client to establish a secure channel with a mesh router (node N1 in our example) that can forward its traffic securely over the wireless backbone. At the end of such phase, N2 obtains the network parameters performing a DHCP request. In phase 2, N2 establishes a secure connection with the Key Server (KS), using the TLS protocol to obtain the necessary information that will be exploited to generate the current key used by all mesh routers to encrypt all the traffic transmitted on the mesh backbone. In particular, the device can connect to the wireless backbone in a secure way and begin executing the routing and access functions.

During phase 2, mesh routers also perform a second authentication, based on the TLS protocol. Only authorized mesh routers that have the necessary credentials can authenticate to the Key Server and obtain the cryptographic material needed to derive the key sequence used to protect the wireless backbone. In our architecture, at the end of the successful authentication, an end-to-end secure channel is established between the Key Server and the mesh routers; the cryptographic material is then exchanged through such channel in a secure way.

To minimize the risks of using the same key for a long time, we propose two key distribution and regeneration protocols, described in Section 6, to create a new key when a pre-determined timeout expires. Both protocols require the synchronization of all mesh routers with a central server.

Figure 4 shows an example network composed of 4 mesh routers connected with 5 wireless links, represented with dashed lines, and the Key Server (KS) Our proposed solution permits an automated and incremental configuration process of the wireless mesh network. At the beginning of the process, only node N1 can connect to the mesh network, since it is the only node that can complete the authentication with the

Key Server and obtain from it the cryptographic material needed to set up an ad hoc and protected wireless link. The neighbors of N1 (N2 and N3) detect a wireless network to which they can connect, and perform the authentication process described by the 802.11i standard as generic wireless clients. Through the wireless network, the mesh routers will be able to authenticate with the Key Server to request the information used by N1 to produce the currently used cryptographic key. After having derived such key, both N2 and N3 will be able to reach each other, as well as node N1, in ad hoc mode. Moreover they will be able to turn on their access interface through which they will provide to node N4 a network connection towards the server.

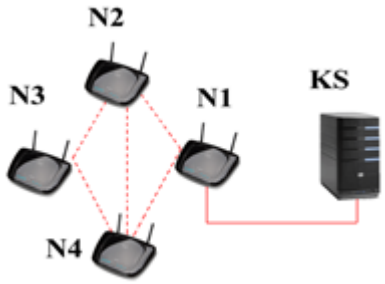

Figure 4. Example of the proposed automated WMN configuration process MobiSEC permits automated and incremental configuration of the wireless mesh network

4. KEY DISTRIBUTION PROTOCOLS

We propose two protocols Server Driven and Client Driven to perform the key delivery and regeneration tasks. In both protocols, time is divided into sessions, whose duration is equal to the product of the number of keys used in a specific session and the key validity time, which is constant for every key of the session.

Consider a single -radio WMN, where all mesh routers are equiped with a single radio interface and communicate with each other using the same wireless channel to exchange data and to perform operations.

4.1. Server Driven Protocol

This protocol provides a reactive method to deliver the keys used by all mesh routers to protect the integrity and confidentiality of the traffic exchanged during a specific interval. In this protocol, each node maintains a list of n keys, which we refer to as the key list. However, we underline that the proposed security architecture is general, and it is designed to manage key lists of arbitrary dimensions.

Fig. 5a shows in detail the message exchanges that occur between the mesh router and the Key Server. The function Ek (•) represents the symmetric cryptographic algorithm established between the two peers after a successful mutual authentication, and it is used to protect the secrecy and the integrity of the successive message exchanges.

idreq and idnode represent respectively the request and the node identifier (i.e. the MAC address of the wireless card on which the request is sent). To improve the robustness of the protocol against reply attacks, all messages can contain further parameters (i.e. a timestamp and a nonce), but for the sake of brevity we did not include them in the figure.

A generic mesh router, after a successful mutual authentication with a central server, sends its first request to obtain the key list used in the current session by the other routers that form the wireless backbone and the time when it was generated, the Key List Timestamp (TSKL). Let us define a session as the maximum validity time of the key list currently used by each node; its duration is the product of the key list cardinality, n, and the maximum validity time of a generic key (the timeout parameter in Fig. 5a). Moreover, the key list validity starts when it is generated, i.e. at TSKL. The node, based on the instant at which it joins the backbone (tnow in Fig. 5a), can identify the key among those in the list currently used by its peers, and its validity time (keyid and T1), according to the following expression:

$$\text{keyed} = \left[\frac{t_{now} - TS_{KL}}{\text{timeout}}\right] + 1 \quad (1)$$

T1= keyid.timeout-(tnow-TSKL).

It is important that each node requests the server the key list that will be used in the next session before the current session expires. This is especially true for nodes that take a long time to receive the

response from the server (due, for example, to slow links or high number of hops from the server). In fact, if the request is sent when the current session is about to expire, the nodes that are connected to the server with the fastest links will receive the response before other nodes; hence they will cut off the others when they enable the new key.

The key index value that triggers the proactive request to the server can be set equal to the difference between the key list cardinality and a correction factor, which can be estimated based on parameters such as the network load, the distance to the server, and the previous time to obtain the response.

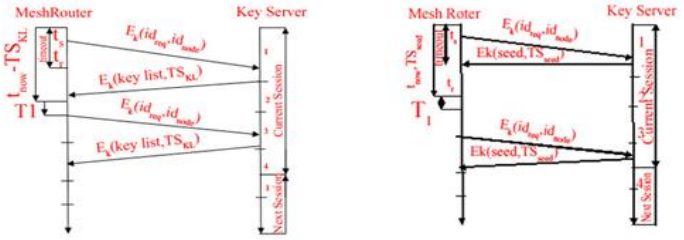

Figure 4. Key distribution protocols: example message exchanges between the mesh router and the Key Server in the (a) Server Driven and (b) Client Driven protocols. Ek ( ) represents the symmetric cryptographic function used to protect the security of the messages, whereas idreq and idnode represent the identifier of the request and of the node, respectively.

In our architecture, such correction factor (c) is computed based on the time necessary to receive the response from the Key Server (Δt), which is estimated according to Eq. (2), where ts is the time when the first or proactive key request was sent, and tr is the time when the corresponding key response was received from the Key Server. So if a node takes a time (Δt in Eq. (2)) greater than timeout to receive the response from the Key Server, it must perform the next proactive request before setting the last key (otherwise, it will not have enough time to obtain the response).

$$\Delta t = t_r - t_s,$$

$$\begin{cases} c = \left\lceil \frac{\Delta t - timeout}{timeout} \right\rceil & if\ \Delta t \geq timeout \\ c = 0 & if\ \Delta t < timeout \end{cases} \quad (2)$$

To illustrate how the correction factor is evaluated, let us refer again to the example message exchange shown in Fig. 5a; the router performs the second request when the third key is set (i.e. the correction factor is equal to 1), so it has enough time to receive the response from the Key Server. In this example, in fact, during the first message exchange it has taken a time greater than timeout to get the response. Note that the first request of the key list sent by the new mesh router to the Key Server will be forwarded by the peer to which it is connected as generic wireless client through the wireless access network, successive requests will be sent directly over the wireless backbone.

4.2. Client Driven Protocol

The Client Driven protocol grants mesh routers more autonomy in the key regeneration process with respect to the Server Driven protocol. In fact, the server provides only a seed and a function type that must be used to compute the sequence of keys used by mesh nodes, with a scheme that resembles a hash-chain method. In our implementation of MobiSEC we use MD5 as hash function, which provides keys with length equal to 128 bit. Note that the proposed framework can easily be modified to use different hash functions and create keys with a different length.

Figure 5b shows the message exchanges performed between the mesh router and the Key Server. As in the previous protocol, a generic mesh router, following a successful mutual authentication with a central server, sends its first request to obtain the seed currently used by the other backbone nodes to create the key sequence, and the time when it was generated, Seed Timestamp (TS_{seed}). Hence, in the Client Driven protocol, a session is defined as the validity time of the current seed, and its duration is the product of the maximum number of keys generated with the same seed and the validity time of a generic key (the timeout parameter).

Eq. (3) illustrates how to compute the number of times the mesh router must apply the hash function to synchronize its first key with that currently used by the other nodes (the r parameter), and its remaining validity time (T_1). The new key is computed as detailed in Equation (4).

$$r = \left\lfloor \frac{t_{now} - TS_{seed}}{timeout} \right\rfloor + 1, \tag{3}$$

T_1=r.timeout (t_{now} TS_{seed}),

$$\begin{cases} key(r, seed) = hash(seed) & if\ r = 1, \\ key(r, seed) = hash(key(r-1, seed)) & if r > 1 \end{cases} \tag{4}$$

To enhance the security of the entire system the following features are added:
1. The argument of the hash function can be obtained by concatenating the seed and the timestamp with a preshared secret known by each node, as proposed for example in
2. A maximum interval for the validity of the seed is set.

The new seed can be obtained by all mesh routers with the same proactive mechanism described above for the Server Driven protocol. Hence, when the mesh router generates one of the last keys that can be computed with the current seed (the one that allows the node to receive the response from the Key Server); it sends a request for a new seed to the server. In Fig. 5b the router performs such a proactive request when the fourth key is generated, since the time spent to get the seed response after sending the first request is less than the key timeout. In this case the correction factor is null, as the timeout value is long enough to obtain the response before the session expiration

5. DESIGN AND IMPLEMENTATION

Figure 6 illustrates the general architecture of the MobiSEC framework. We implemented the key distribution protocols as a client/server application using the OpenSSL library to authenticate and protect the connection that is established when a new node joins the wireless backbone network.

In particular, each communication that takes place between a mesh router and the Key Server uses the TLS protocol both to authenticate the two entities and to protect the key material that is exchanged. The cryptographic material is communicated to the Key Switcher module that performs the tasks defined by our protocols to obtain and install the currently used key. We decided to implement this component as a kernel module to improve its responsiveness, especially under heavy network load conditions. In fact, the routing mechanism operating in kernel space can require a long execution time to manage the soft interrupts generated by the received packets, causing high level of delay in the scheduling of the user space processes.

Therefore, implementing and running the module dedicated to deriving and installing the new key as a user space process may result in unpredictable scheduling delays, sometimes greater than the key validity time. On the other hand, such delay has a negligible effect on the client-side application (the Client Daemon module in Fig. 6, since the correction factor that is used to trigger the proactive request takes into account also this contribution.

5.1. Layer 2 Encryption

In our implementation of MobiSEC we decided to use the encryption techniques provided by the MAC layer, since the most computationally complex operations are performed by the wireless card. Such solution has two main advantages: on the one hand, the network performance is not impaired by executing such procedures; on the other hand a data-link layer encryption reduces the security requirements of the control and routing protocols.

5.2. Multi Radio Extensions

The proposed architecture can easily be applied to a multi-radio WMN, where each node is endowed with several wireless interfaces dedicated to the backbone traffic.

To this end, it is necessary to modify simply the messages format defined by the previous protocols so as to provide the additional information to the other end. In particular, the Key Server generates different cryptographic information for each possible channel, whereas the mesh router requests and obtains the cryptographic information (key list or seed and type of the hash function) that is related only to the wireless channels on which its interfaces are set.

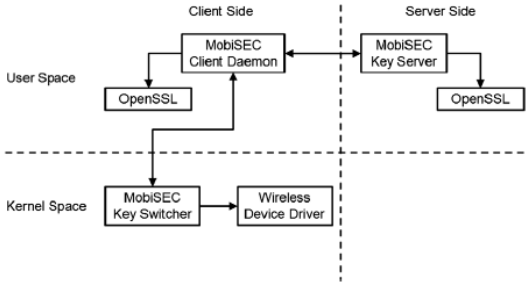

Figure 6. MobiSEC architecture. The client side application is installed on all mesh routers, whereas the server side application is installed executively on the Key Server.

5.3. Synchronization Issues

The synchronization of all mesh routers with the Key Server is a requirement for our architecture. However, in our tests we measured a synchronization difference among all nodes always smaller than a few milliseconds. Therefore, taking ample margins, we introduce a tolerance on the key validity of 2 s. This is obtained using cyclically three of the four hardware registers commonly provided by commercial wireless boards to install the cryptographic keys. The tolerance is realized setting the successive key of the sequence 2 s before the expiration of the current one and maintaining the previous key a further 2 s after its expiration.

Such setting permits the obtaining of a performance that is very close to that achieved with a static key, as we will show in the next section, since both early and late nodes can properly decrypt the received frames.

5.4. Network Partitioning

It may happen that the network is temporarily partitioned in two or more sub networks due to interference or nodes malfunctioning, so that some mesh router can no longer connect to the Key Server. In this case, our architecture allows nodes inside each sub network to continue communicating among them using the current key. Furthermore, they periodically try to contact the Key Server to recover normal operation.

5.5. Detection Techniques and Certificates revocation

Finally, note that the authentication method based on certificate exchanges, used in our architecture, protects against man in the middle attacks, since all the certificates are signed by a trusted certification authority (CA), whose certificate is known by all network devices.

6. EXPERIMENTAL RESULTS

6.1. Full-Mesh Topology

In Figure 7 each router is considered as a node where all nodes belong to the same adhoc wireless cell. Router N3 act as a key server so the other routers send key information and request to N3. In such scenario we measured the throughput of TCP connection established between the nodes over a wireless link protected by Client and Server Driven protocols; Then, we compared such results with those achieved on a radio link protected with a static key and by establishing an encrypted IPSec tunnel between each pair of nodes.

Using D-ITG traffic generator we generated traffics between N1 and N3, We noticed that the maximum throughput is similar for both Client and Server Driven Protocols, since the computation of the key sequence performed by the Client Driven protocol does not impair the achievable throughput.

Furthermore, note that such throughput is very close to the bound provided by the static key technique. On the other hand, the IPSec solution achieves a lower performance, which is mainly due to the fact that layer-2 encryption, used by all the other considered protocols, is directly supported in the wireless card hardware. At the same time, we tested the availability of the Key Server, and we verified that all mesh routers could remain connected even in the presence of a high network load.

In the same scenario we measured the effectiveness of the key tolerance by disabling it and measuring the protocols' performance. The corresponding numerical results are shown in Figure 9, and the performance improvement introduced by implementing the tolerance mechanism is evident (see Figure 8 for a comparison).

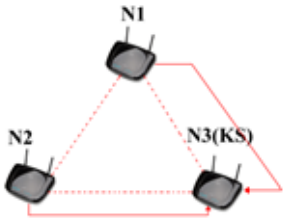

Figure 7. Full-mesh topology. A data transfer is performed between nodes N1 and N3. Although N3 also acts as Key Server, the connection among the three nodes remained available in all the tests we performed

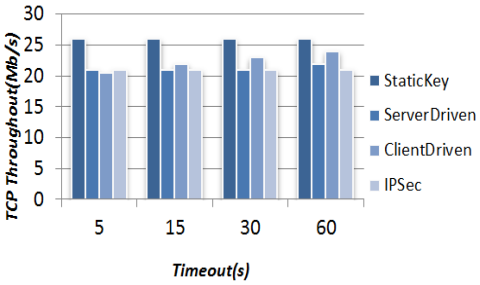

Figure 8. TCP throughput measured in the full-mesh network scenario for different key distribution protocols and key validities, using the tolerance on the key validity time

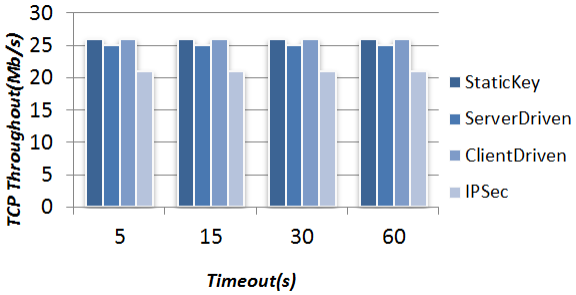

Figure 9. TCP throughput measured in the full-mesh network scenario for different key distribution protocols and key validities, disabling the tolerance on the key validity time

In the same scenario we further measured the packet loss eventually caused by the key renewal procedure, considering a data transfer based on a UDP connection. Packet loss can be critical for real-time multimedia applications, such as VoIP and streaming video. We therefore generated UDP traffic on the wireless link between nodes N2 and N3. The transmission rate was set to 10 Mb/s and several data transfer sessions were performed, each with a duration ranging in the 2–12 min interval. We observe that the choice of the number of keys used in a session, n, has no impact on the packet loss, since we measured a negligible value of such performance Figure in all our experiments.

6.2. Multihop Topology

In Figure 10, we considered the Multi-hop network where solid lines represent wired and dashed lines represents wireless links respectively.

All nodes were physically connected with two wireless interfaces i.e., each mesh router is connected only to the previous and sub-sequent node. All mesh routers runs the client-side application of the Client and Server Driven protocol whereas node KS acted as only Key Server.

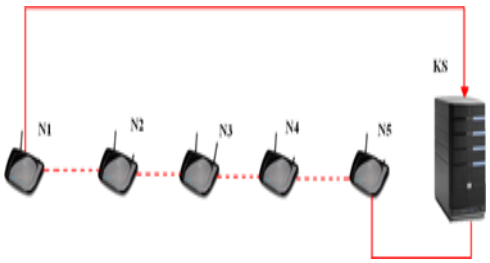

Figure 10. Multi-hop topology. A multi-hop data transfer between nodes N1 and KS is performed to measure the network performance

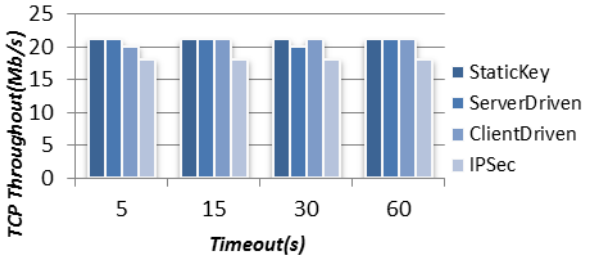

Figure 11. TCP throughput measured in the multi-hop network scenario for different key distribution protocols and key validities, using the tolerance on the key validity time

Then we transfer data between N1 and N2 using DIT-G traffic generator and we noticed that we achieve the maximum throughput of the proposed protocols then we compare the results obtained with the static key solution and establish an encrypted IPSec tunnel between node N1 and KS. The results, reported in Figure 11, confirm the trend of the previous network scenario: the tolerance introduced on the key validity time permits an improvement in the strength of the proposed scheme without reducing consistently the overall throughput. In the same scenario we evaluated the Round Trip Time (RTT), setting the packet size to 1500 bytes. Table 1 shows the results (expressed in milliseconds) that we measured setting the key timeout and the session duration to 30 and 120 s, respectively. The low value of the RTT's standard deviation suggests that our solution guarantees a correct operation even for real-time multimedia applications without introducing perceptible alterations in the transmitted stream. Since all the results we measured show that the IPSec solution performs consistently worse than the proposed protocols, for the sake of brevity in the following we do not report the results obtained with such technique.

Table 1. Round Trip Time Measured in ms for a tcp Connection Establised between nodes N1 and Ks

Parameter	Static Key	Client	Werver
Average RTT	8.2	8.5	8.4
Minimum RTT	7.9	8.3	8.2
Maximum RTT	8.4	8.7	8.6
RTT SD	1.2	1.2	1.2

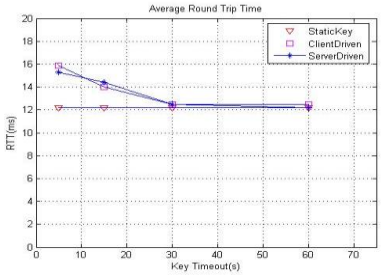

Figure 12. Average Round Trip Time as a function of the key timeoutMeasured in the topology of Figure 10

7. CONCLUSION

In this paper we proposed MobiSEC, a novel security architecture tailored for wireless mesh networks. MobiSEC addresses the security problems of both the access and backbone areas of WMNs, providing an effective and transparent security solution for end-users and mesh nodes. We implemented our proposed security architecture in MobiMESH, a complete wireless mesh network framework, and we tested it in several realistic network scenarios, comparing its performance with that of existing schemes, viz.: static key encryption and end-to-end IPSec tunnel solutions. Furthermore, we simulated the behavior of MobiSEC in large-scale network instances using Network Simulator.

REFERENCES

[1] Kun Zhu; Chenine, M.; Nordstrom, L.; , "ICT Architecture Impact on Wide Area Monitoring and Control Systems' Reliability," *Power Delivery, IEEE Transactions on* , vol.26, no.4, pp. 2801-2808, Oct. 2011

[2] Angelou, G.N.; Economides, A.A.; , "A Decision Analysis Framework for Prioritizing a Portfolio of ICT Infrastructure Projects," *Engineering Management, IEEE Transactions on* , vol.55, no.3, pp.479-495, Aug. 2008

[3] Kum Leng Chin; Chang, E.; , "*A sustainable ICT education ontology*," Digital Ecosystems and Technologies Conference (DEST), 2011 Proceedings of the 5th IEEE International Conference on , vol., no., pp.350-354, May 31 2011-June 3 2011

[4] Chai-Arayalert, S.; Nakata, K.; , "*The Evolution of Green ICT Practice: UK Higher Education Institutions Case Study*," Green Computing and Communications (GreenCom), 2011 IEEE/ACM International Conference on , vol., no., pp.220-225, 4-5 Aug. 2011

[5] Babulak, E.; , "*ICT for Human Development in South Pacific*," Multimedia Information Networking and Security (MINES), 2010 International Conference on , vol., no., pp.621-624, 4-6 Nov. 2010