Hindawi Publishing Corporation
Journal of Electrical and Computer Engineering
Volume 2016, Article ID 9859830, 12 pages
http://dx.doi.org/10.1155/2016/9859830

Research Article

Design and Verification of Secure Mutual Authentication Protocols for Mobile Multihop Relay WiMAX Networks against Rogue Base/Relay Stations

Jie Huang and Chin-Tser Huang

Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29201, USA

Correspondence should be addressed to Chin-Tser Huang; huangct@cse.sc.edu

Received 9 March 2016; Accepted 2 June 2016

Academic Editor: Jit S. Mandeep

Mobile multihop relay (MMR) WiMAX networks have attracted lots of interest in the wireless communication industry recently because of its scalable coverage, improved data rates, and relatively low cost. However, security of MMR WiMAX networks is the main challenge to be addressed. In this paper, we first identify several possible attacks on MMR WiMAX networks in which a rogue base station (BS) or relay station (RS) can get authenticated and gain control over the connections and show that the current standard does not address this problem well. We then propose a set of new authentication protocols for protecting MMR WiMAX networks from rogue BS attack, rogue RS attack, and suppress-replay attack. Our protocols can provide centralized authentication by using a trusted authentication server to support mutual authentication between RS and BS, between RS and RS, and between mobile station (MS) and RS. Moreover, our protocols can also provide distributed authentication with a license issued by the trusted server. We use a formal tool called Scyther to analyze and verify the security properties of our protocols. The results show that our protocols can counter rogue BS and RS attacks and suppress-replay attack and are not susceptible to any known attacks.

1. Introduction

In the era of the Internet of Things (IoT) when every device is connected to the Internet, one of the most important technical enablers is the wireless technologies and infrastructure that make connecting different things (devices) possible. Among them, WiMAX (Worldwide Interoperability for Microwave Access) plays a very important role as it delivers lower-cost, longer-range, and high-bandwidth mobile broadband access for mobile clients and devices to connect to the Internet from anywhere at any time.

WiMAX is a broadband wireless access technology designed for the advancement of IEEE 802.16 standard [1]. It is considered to be a replacement for WiFi-based mobile broadband connection, because of its better coverage and faster speed. WiMAX is ideal for high data-rate IP applications such as video conferencing, VoIP, online gaming, and HD video streaming. Another wireless broadband technology, LTE (Long Term Evolution) [1], can also provide high-speed data transmission for mobile phones and data terminals and has been adopted by many cellular service providers. In recent years, some people think that LTE has won over WiMAX in the standard war. However, according to the report from Intel Capital [2], WiMAX does not fade away and is still considered a very good option by many service providers, including Egyptian telecom startup, Orascom Telecom, and Netherlands startup Enertel Holding [2].

In the early days when WiMAX was designed, WiMAX faced the paradox that increasing data rate will reduce reliability, and increasing minimum reliability service will reduce the coverage area [3]. One possible solution is to deploy more base stations (BSs) closely, but the high cost of deploying BSs will give away the original economic competitiveness of WiMAX. Therefore researchers switched to a more viable approach, which is to insert relatively cheaper fixed relay stations into the cell. This kind of networks is called multihop relay networks. In June 2009, the IEEE 802.16 Relay Task

Group (TG) proposed the IEEE 802.16j-2009 amendment, whose main purpose is to expand the previous single-hop 802.16 standard to include multihop capabilities [4], enable the operations of multihop communications based on relay stations (RSs), specify the mobile multihop relay (MMR) deployment, and define two new types of elements, the multihop relay base station (MR-BS) and the relay station (RS). In 2012, 802.16 Working Group announced the latest version of the standard, IEEE 802.16-2012 [5], which incorporated 802.16j along with two other amendments, 802.16h and 802.16m.

Security has been an open challenge in WiMAX since its commencement because of the open-air nature of wireless communications. To overcome the potential attacks to WiMAX, IEEE 802.16 standard specifies a security sublayer in the MAC (Media Access Control) layer. IEEE 802.16-2009 offers an improved authentication and authorization mechanism compared to its previous versions such as IEEE 802.16-2004. IEEE 802.16-2009 provides better encryption methods, more secure key management protocol, and an EAP-based authentication strategy. However, in an MMR WiMAX network, more security issues are exposed since messages have to be transmitted through one or more relay stations, which makes it more difficult to ensure the authenticity of messages and devices involved in the transmission. Therefore the latest standard IEEE 802.16-2012 defines an air interface between an MR-BS and a RS with the following additional security functionalities [5]:

(i) *Trust within a certain cell*, that is, MR-BS and a group of RSs in the MR cell maintain a set of trusted relationships, called Security Zone, in order to satisfy requirements of multihop relay system operations.

(ii) *Centralized security control in MR-BS*, that is, MR-BS is in charge of the generation of the security association materials between RS and MR-BS.

(iii) *Transparency for mobile station (MS) connected to the network through one or more RSs*, that is, any intermediate RS does not try to decrypt the user data or authenticate the MAC management message it receives from the MS but simply relays it to the next node.

(iv) One RS does not have Authorization Key (AK) security context of any other RS.

(v) The intermediate RS authenticates management messages it receives from other RSs using relay-specific shard keys.

(vi) *Protection of nonauthenticated Pairwise Master Key (PKM) messages by MR-BS and the access RS*, that is, any nonauthenticated PKM messages which are transmitted between MS and MR-BS through the access RS will be protected by the HMAC/CMAC based on the shared security associations established between MR-BS and the access RS.

However, even with these additional functionalities, researchers in [6] found that the security mechanism in IEEE 802.16j (part of current IEEE 802.16-2012 standard) is still not adequate in that it has vulnerabilities in its weak protection of some PKM messages and security zone key update and is susceptible to DoS attacks on BS and rogue BS and rogue RS attacks.

In our previous conference paper [7], we focused on addressing the rogue BS and rogue RS attacks in MMR WiMAX networks with centralized authentication schemes and designed a set of protocols to address the aforementioned attacks. However, a formal verification we conducted later using the Scyther tool [8] shows that our previous protocols allow for a new attack called suppress-replay attack, which exploits the asynchronization of time stamps in BS and RS. Moreover, our previous work did not provide a distributed authentication scheme, which is considered to be a more favorable way to conduct authentication nowadays because of the multitude of distributed mobile clients and devices.

In order to address these new issues found in our previous paper, we propose in this paper a complete authentication solution to address the rogue BS and rogue RS attacks in MMR WiMAX networks, the suppress-replay attack found in our previous scheme, and a distributed authentication scheme between RSs. We present three different mutual authentication protocols which utilize a trusted authentication server to support three possible scenarios in which the RS or MS connects to an MMR WiMAX network. Our protocols are conformant to the security requirements of IEEE 802.16-2012 standard. We also verify the correctness of our protocols by utilizing the former verification tool for security protocols called Scyther. According to the verification result, our protocols are able to counter against all the attacks we discussed in our paper.

The remainder of this paper is organized as follows. In Section 2, we give an overview of related works. Section 3 describes the possible rogue BS and rogue RS attacks on access service in MMR networks. In Section 4, we present our new schemes for securing the original authentication protocols for MMR networks against rogue BS and rogue RS. In Section 5, we give a formal analysis and verification of our protocols using the Scyther tool. Finally, we conclude our paper and discuss the future work in Section 6.

2. Related Work

A number of papers have been published regarding the security issues of WiMAX networks since IEEE 802.16 standard was developed. Xu et al. give a detailed analysis on privacy and key management protocols of the standard in [9, 10]. Several other papers addressed the security issues of one-way authentication and rogue base station attack such as [9, 11, 12]. However, these publications considered only the single-hop WiMAX when the MMR WiMAX network had not come to existence.

The authentication issue has been studied in several other types of multihop networks, such as wireless mesh networks [13], cellular networks [14], and sensor networks [15]. However, the MMR WiMAX network is still very recent and has its own unique characteristics that need to be investigated separately.

Distributed and centralized authentication approaches are two major options when it comes to authentication protocol design. In [16], Yang et al. discussed security issues in WiMAX MMR networks and the pros and cons of the two major types of authentication protocol design. In [17], Jin et al. propose an improved mutual authentication scheme in multihop WiMAX networks, in which they improve the X.509 certificate by using ECC algorithm instead of RSA, and modify the flow of mutual authentication to improve the security in multihop WiMAX networks. In [18, 19], Khan et al. proposed a modified PKM protocol using distributed authentication and localized key management scheme. In [3], Tie and Yi proposed a multihop ticket based handover authentication which adopted the idea from Kerberos and used a ticket to allow MS, RS, and BS to mutually authenticate each other. However, the authors in the aforementioned papers did not take rogue access node attack into consideration.

In order to solve the problems like security zone key update, DDoS attack, and rogue RS attack, in [20], the authors propose a design of hybrid authentication and key distribution scheme to support the IEEE 802.16j (part of current IEEE 802.16-2012 standard) MMR requirements. Although the authors claim that this hybrid design is robust enough to prevent rogue node attack, they only consider the case when a rogue RS tries to join the network at initial phase, and they do not take rogue BS attack into account. The latter case will cause more severe damage to the network since a rogue BS can take control of the whole area within its communication range if it successfully joins the network as a legitimate BS. In another paper [21], the authors present a distributed scheme using decode and forward relays with localized authentication, which helps to authenticate MS and RS at initial network entry. However, this scheme still cannot solve the problem of rogue BS attack. In [22], the authors proposed a self-testing approach to defend against rogue BS attack of intelligent terminal. However, their work did not focus on MMR networks and thus cannot address the other security issues in MMR networks we discussed in Section 1. In [23], the authors discussed detection of rogue BS attack in WiMAX networks; however, their discussion did not address the security issues existing specifically in MMR networks.

3. Rogue BS and Rogue RS Attacks in MMR WiMAX Networks

Rogue stations have been one of the most common threats in wireless networks [23–25]. In order to design a secure wireless authentication protocol in MMR networks, rogue stations threat must be considered and addressed.

Denial of Service (DoS) jamming is a type of the attacks that can involve a rogue station. In MMR, a rogue station can be a rogue BS or a rogue RS. In a DoS jamming attack, by jamming a legitimate BS, connectivity between client and a legitimate BS can be interrupted, which makes it possible for a rogue BS to stand in and impersonate the legitimate BS with fake credentials, trying to convince a joining client or RS to connect with it, so as to cause DoS or even redirect the traffic and hijack the communications.

Another form of attack that might involve a rogue station, especially a rogue RS, is man-in-the-middle (MITM) attack [25]. When a client MS or RS initiates a connection, the rogue RS will intercept the connection, and then complete the connection to the intended legitimate RS and proxy all communications to the intended legitimate RS. The rogue RS is now in a position to inject data, modify messages and communications, or eavesdrop on a session that would normally be difficult to decode, such as encrypted sessions.

One form of MITM attack involves asynchronization. If the clocks of the client and the legitimate station are not synchronized, it is possible for the rogue station to launch a suppress-replay attack [26]. In a suppress-replay attack, the rogue station can intercept messages that carry a timestamp corresponding to a future time due to an unsynchronized clock and extract from them the component containing the future timestamp. Then, the rogue station can combine the extracted timestamp component with valid components from other messages to create a fake message and replay it later when the timestamp in the fake message becomes current with respect to the clock of the legitimate recipient station.

An example to demonstrate the suppress-replay attack is given as follows. Suppose we have a client MS whose clock is 2 minutes ahead of the one in the legitimate station BS. At client side time 1:10 pm and time 1:11 pm (which corresponds to BS time 1:08 pm and time 1:09 pm), client MS sends two messages, message 110 and message 111, to the legitimate BS trying to initiate a connection; each message contains its current MS timestamp and component with necessary authentication credentials to prove MS's identity; we call this component *Authentication Component (AC)* here. When an attacker intercepts these two messages, the attacker can extract the AC from message 110 and the 1:11 pm timestamp from message 111 and then combine these two parts together to create a new message $111'$. The attacker will then send out this newly created fake message $111'$ when the time on the legitimate station BS becomes 1:11 pm. IEEE 802.16-2012 uses PKMv2 to counter possible rogue BS attack by using mutual authentication. However, there is an implicit assumption in PKMv2 that BS is always trustworthy; thus PKMv2 does not provide any protection measure to detect and counter the attack from a compromised BS.

Moreover, the distributed security mode in MMR WiMAX networks also makes rogue RS attack more possible. This is because the authentication procedure between RS nodes is not performed by a centralized server but is based on the trust between nodes. If one node is compromised, its trust with other nodes is also compromised.

Another issue with PKMv2 is that the preassumed trust of BS cannot prevent suppress-replay attack mentioned above. To address all of the aforementioned attacks in MMR, careful protocol design is required in order to achieve secure authentication in MMR networks.

4. Proposed Secure Authentication Protocol

We have introduced the related work on security issues in MMR WiMAX networks and have shown that current standards are not sufficient for addressing the rogue BS and rogue RS attacks. In this section, we present a set of new secure protocols to provide robust authentication in MMR WiMAX networks. Specifically, our protocols can defend against rogue BS and RS attack by using a trusted authentication server to provide dual authentication and security zone key. Our protocols support three scenarios of network access: RS connects to an MMR network through BS; RS connects to an MMR network through other RS; and MS connects to an MMR network through RS.

4.1. Assumptions. The proposed protocols are based on the following assumptions:

(a) Before the initial access authentication process, each RS, BS, and mobile user (MS) is preregistered with the Authentication Server (AS) by providing their MAC addresses and other necessary credentials. Each RS, BS, and MS shares its own public key (K_{RS}, K_{BS}, K_{MS}) with AS, and each RS, BS, and MS also gets AS's public key from AS. These keys were obtained during the preregistration phase.

(b) AS is trusted by all the nodes in the MMR WiMAX network. It is believed by the nodes in this network that AS maintains a correct database of all legitimate registered nodes' MAC addresses, each node's corresponding public key, and other credentials. It is easier to ensure the physical security of AS because AS can always be indoor.

4.2. Notations. Before we describe the details of our protocols, we specify the simplified notation for each element used in the protocol:

MAC_X: X'S MAC address (X can be either RS, BS, or MS),

$\mathrm{Seq}_{\_X}$: a sequence number generated by station X,

$\mathrm{Nnc}_{\_X}$: a nonce generated by station X,

K_X: X's secret key shared with AS,

K_{X_PUB}: X's public key stored in X's certificate,

K_{X_PRV}: X's private key corresponding to the public key stored in X's certificate,

PMK: Pairwise Master Key,

SZK: a group key used in the security zone among BS and many RSs,

AK: Authorization Key,

$\mathrm{MD}[M]$: message digest of message M,

CERT_X: X's digital certificate with X's public key included,

$E_{K_X}[M]$: M encrypted with X's shared secret key,

$E_{K_{SZK}}[M]$: M encrypted with security zone key,

$E_{K_{X\_\mathrm{PUB}}}[M]$: M encrypted with X's public key stored in X's certificate,

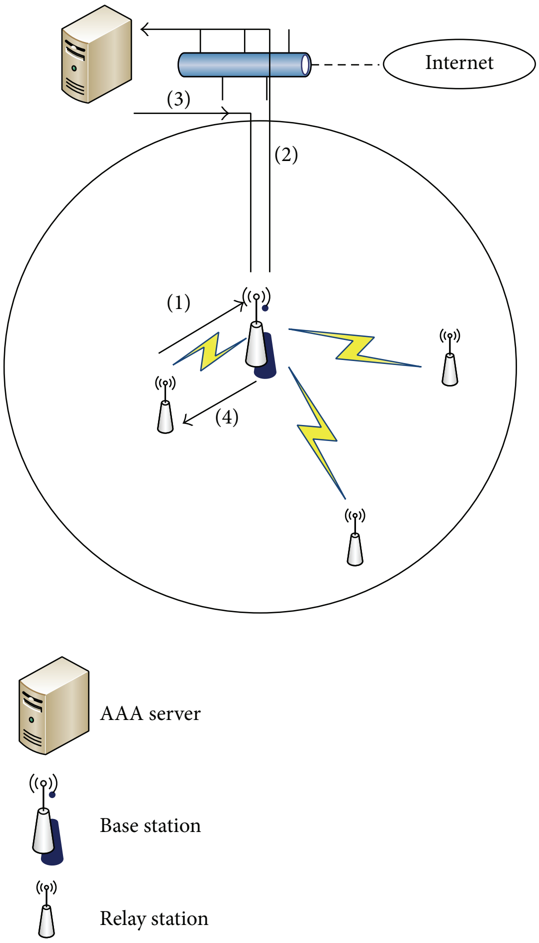


Figure 1: RS connects to MMR networks via BS.

$E_{K_{X\_\mathrm{PRV}}}[M]$: M encrypted with X's private key corresponding to the public key stored in X's certificate,

$\mathrm{License}_{\_X}$: a signature issued to legitimate BS/RS/MS; the signature is generated by AS using AS's private key,

EXPR_X: expiration time for $\mathrm{License}_{\_X}$.

Here the format of $\mathrm{License}_{\_X}$ is $\{E_{K_{AS\_PRV}}[\mathrm{MAC}_X \parallel \mathrm{CERT}_X \parallel \mathrm{EXPR}_X]\}$.

4.3. Scenario 1: RS Connects to the Network through BS. The first scenario in which a RS needs to connect to an MMR network via BS is shown in Figure 1.

A RS broadcasts the AUTH-REQ message when it wants to connect to an MMR WiMAX network. Normally the BS which is the closest to this RS will handle this message and send it to AS. AS will then perform the authentication and send back an AUTH-REPLY message. The detailed message format is specified as follows:

(1) RS → BS:

$\mathrm{MAC}_{RS} \parallel \mathrm{Seq}_{\_RS} \parallel \mathrm{MD}[\mathrm{MAC}_{RS} \parallel \mathrm{Seq}_{\_RS} \parallel K_{RS}] \parallel \mathrm{CERT}_{RS}$

(2) BS → AS:

$MAC_{RS} \| Seq_{\_RS} \| MD[MAC_{RS} \| Seq_{\_RS} \| K_{RS}] \| CERT_{RS} \| MAC_{BS} \| Seq_{\_BS} \| MD[MAC_{BS} \| Seq_{\_BS} \| MAC_{RS} \| K_{BS}] \| CERT_{BS}$

(3) AS → BS:

$E_{K_{BS}}\{MAC_{AS} \| MAC_{RS} \| Seq_{\_BS} \| PMK \| License_{\_BS} \| SZK \| MD[PMK \| SZK \| K_{BS}]\} \| E_{K_{RS}}\{MAC_{AS} \| MAC_{BS} \| Seq_{\_RS} \| PMK \| License_{\_RS} \| SZK \| MD[PMK \| SZK \| K_{RS}]\}$

(4) BS → RS:

$E_{K_{RS}}\{MAC_{AS} \| MAC_{BS} \| Seq_{\_RS} \| PMK \| License_{\_RS} \| SZK \| MD[PMK \| SZK \| K_{RS}]\} \| MD[PMK \| AK \| SZK]$

As we mentioned before, each RS is preregistered with AS. Therefore AS can match the MAC address in the message with the corresponding shared secret key in the database. When AS receives message (2), AS can use MAC address, sequence number, and the shared secret key to calculate the corresponding message digest in order to verify the authenticity of the sender (both BS and RS), since only AS and the corresponding legitimate node know the shared secret key. After successful authentication of BS and RS, AS will use the shared keys of BS and RS to encrypt the authentication reply message and send it back to BS and the requesting RS. If both BS and RS are legitimate, in message (3) AS will generate a PMK and a SZK and include them in the reply message. After BS receives and decrypts the AUTH-REPLY message from AS, it can be sure about the authenticity of the requesting RS. Thus BS can decide whether to grant RS the access to the network or not. BS also gets PMK from AS and uses it to generate AK. PMK is encrypted with BS's shared key with AS. AS also sends the message digest of PMK in this message to BS, which is meant to allow BS to verify the integrity of this PMK it receives. In message (3), BS receives its $License_{\_BS}$ from AS, which works as a unique digital signature to prove its identity to other nodes before the license expires. This signature is generated by encrypting the concatenation of BS's MAC address, BS's certificate (e.g., X.509 certificate), and the expiration time of this license, using AS's private key. As what has been described in our assumptions, every legitimate node in this network has the AS's public key, so when BS gets its license, it can use AS's public key to verify this license's authenticity. Similar license also appears in message (4) in which it provides legitimate RS with a license to prove its identity to other nodes in the future. The usage of signature here in messages (3) and (4) can protect the network from malicious node with a fake license, since it should be computationally infeasible for a party who does not possess the private key to generate a valid signature.

After successful authentication with BS/RS, AS assigns a security zone key, denoted as SZK, for nodes in the security zone to secure future communications between them. At AS, for each AUTH-REQ from RS, AS will check with BS's MAC address in its received message to decide which security zone this requesting RS belongs to. After authentication is successfully completed, a zone key SZK which corresponds to the right BS will be assigned to the RS.

In message (4), along with PMK, BS also sends the message digest of AK which it generated from PMK; the purpose is to let RS verify the integrity of the PMK it receives. If RS can use the PMK it received to generate the same AK which is consistent with the message digest MD[AK] included in message (4), it can believe that the message is not compromised and the AK is good to use.

We analyze why our protocol can protect the MMR WiMAX network against rogue BS or rogue RS attacks. In the case of rogue BS attack, since BS does not have RS's secret key shared with AS, it cannot decrypt or modify the message destined to RS. Hence if RS gets the message from AS which notifies RS of this illegitimate BS, RS will not try to get access through this rogue BS again. Therefore the rogue BS will not have the opportunity to take control of RS in the network.

In the case of rogue RS attack, when an access RS becomes a rogue one, it would not receive the AUTH-REPLY message with a license included. In the case of message hijacking or man-in-the-middle attack, even if an attacker can get the message which was supposed to be transmitted to the legitimate BS or RS, the attacker still cannot decrypt the message to get the license since he does not have the secret key K_{BS} or K_{RS}. Therefore the attacker cannot obtain a valid license without getting authenticated by AS, which it cannot pass.

4.4. Scenario 2: RS Connects to the Network through Other RS. The second scenario in which a RS requests to connect to an MMR network through an edge RS is shown in Figure 2. In this scenario, the requesting RS wants to set up connection with the BS via one or more intermediate RSs. Our protocol for this scenario contains three authentication phases: (i) initial verification of edge RS; (ii) dual authentication of BS and the requesting RS by AS; and (iii) distributed authentication when requesting RS holds a valid license.

In authentication phase (i), when the requesting RS_A tries to get access to the network through another RS_B (edge RS), RS_A needs to verify the authenticity of this intermediate RS_B first. The message format of phase (i) is specified as follows:

(1) Requesting RS_A → Edge RS_B:

$MAC_{RS(A)} \| CERT_{RS(A)} \| Seq_{\_RS(A)} \| MD[MAC_{RS(A)} \| Seq_{\_RS(A)}]$

(2) Edge RS_B → Requesting RS_A:

$MAC_{RS(B)} \| CERT_{RS(B)} \| License_{\_RS(B)} \| MD[MAC_{RS(B)} \| Seq_{\_RS(A)}] \| E_{K_{RS(B)\_PRV}}[E_{K_{RS(A)\_PUB}}[Seq_{\_RS(A)} \| Nnc_{\_RS(B)}]]$

(3) Requesting RS_A → Edge RS_B:

$E_{K_{RS(A)\_PRV}}[E_{K_{RS(B)\_PUB}}[Nnc_{\_RS(B)} \| Nnc_{\_RS(A)}]]$

(4) Edge RS_B → Requesting RS_A:

$E_{K_{RS(B)\_PRV}}[E_{K_{RS(A)\_PUB}}[Nnc_{\_RS(A)}]]$

In messages (1) and (2), the requesting relay station RS_A and the edge relay station RS_B exchange their certificate, such that they know each other's public key. In message (2), based on our assumption that each registered RS has the AS's public key, the requesting RS_A can use it to verify whether the edge

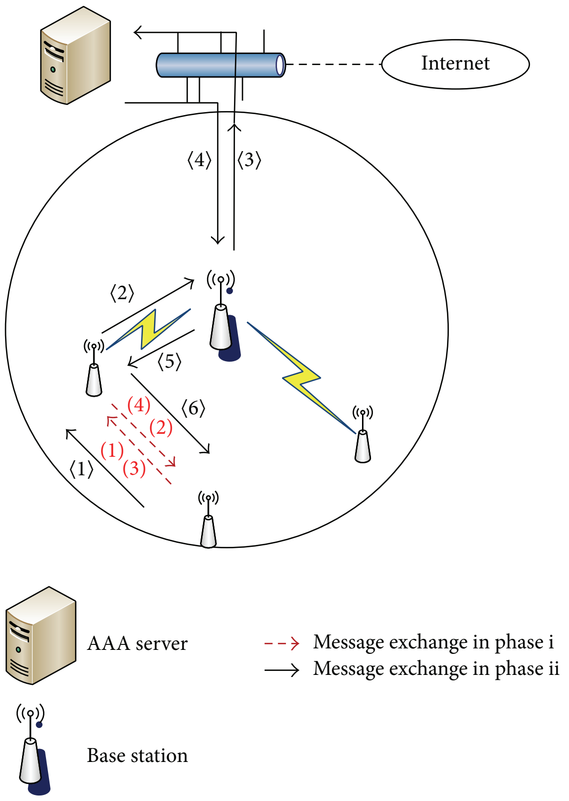


Figure 2: RS connects to MMR networks via an edge RS.

RS_B is a legitimate relay station by checking that the license is currently valid. However, if the authentication phase (i) stops at message (2), then it will be susceptible to a *replay attack* described as follows: a malicious relay station RS_E listening in the middle makes a copy of message (1) sent by the requesting RS_A, makes a separate run of this protocol by forwarding the copy of message (1) to a legitimate edge relay station RS_B (i.e., pretending RS_E itself is the requesting RS_A), and replays the message (2) received from the RS_B to RS_A. When RS_A receives the message, it will be convinced that the malicious RS_E is a legitimate edge relay station. To address this problem of replay attack, we add messages (3) and (4), which is in challenge-response style. In message (3), RS_A concatenates $Nnc_{\_RS(B)}$ (derived from decrypting message (2)) with a nonce of its choice $Nnc_{\_RS(A)}$, encrypts it using the public key of RS_B and the private key of RS_A, and sends it to RS_B. RS_B decrypts the received message (3) and extracts $Nnc_{\_RS(A)}$, encrypts it using the public key of RS_A and the private key of RS_B, and sends it to RS_A in message (4). When RS_A verifies that the $Nnc_{\_RS(A)}$ received in message (4) is the same as the nonce it chooses in message (3), RS_A can believe that RS_B is really the authentic edge relay station, rather than a malicious relay station of a replay attack.

After edge RS_B's authenticity has been verified in phase (i), the requesting RS_A starts phase (ii), in which both BS and RS_A need to be authenticated by AS. The message format of phase (ii) is specified as follows:

⟨1⟩ Requesting $RS_A \to$ Edge RS_B:

$MAC_{RS(A)} \| Seq_{\_RS(A)} \| CERT_{RS(A)} \| MD[MAC_{RS(A)} \| Seq_{\_RS(A)} \| K_{RS(A)}]$

⟨2⟩ Edge $RS_B \to$ (possibly other RS in between) $\to$ BS:

$E_{K_{SZK}}\{MAC_{RS(A)} \| Seq_{\_RS(A)} \| CERT_{RS(A)} \| MD[MAC_{RS(A)} \| Seq_{\_RS(A)} \| K_{RS(A)}]\}$

⟨3⟩ BS $\to$ AS:

$MAC_{RS(A)} \| Seq_{\_RS(A)} \| CERT_{RS(A)} \| MD[MAC_{RS(A)} \| Seq_{\_RS(A)} \| K_{RS(A)}] \| MAC_{BS} \| Seq_{\_BS} \| MD[MAC_{BS} \| MAC_{RS(A)} \| Seq_{\_BS} \| K_{BS}] \| CERT_{BS}$

⟨4⟩ AS $\to$ BS:

$E_{K_{BS}}\{MAC_{AS} \| MAC_{RS(A)} \| Seq_{\_BS} \| PMK \| License_{\_BS} \| SZK \| MD[PMK \| SZK \| K_{BS}]\} \| E_{K_{RS(A)}}\{MAC_{AS} \| MAC_{BS} \| Seq_{\_RS(A)} \| PMK \| License_{\_RS(A)} \| SZK \| MD[PMK \| SZK \| K_{RS(A)}]\}$

⟨5⟩ BS $\to$ Edge RS_B:

$E_{K_{SZK}}\{E_{K_{RS(A)}}\{MAC_{AS} \| MAC_{BS} \| Seq_{\_RS(A)} \| PMK \| License_{\_RS(A)} \| SZK \| MD[PMK \| SZK \| K_{RS(A)}]\} \| MD[PMK \| AK \| SZK]\}$

⟨6⟩ Edge $RS_B \to$ Requesting RS_A:

$E_{K_{RS(A)}}\{MAC_{AS} \| MAC_{BS} \| Seq_{\_RS(A)} \| PMK \| License_{\_RS(A)} \| SZK \| MD[PMK \| SZK \| K_{RS(A)}]\} \| MD[PMK \| AK \| SZK]$

If the authentication between RS_A and AS is successful and RS_A is regarded to be legitimate, then RS_A will be assigned with the security zone key SZK which RS_A and RS_B can use to secure future communications between them.

Comparing the message format in phase (ii) with the message format in the first scenario in which RS connects to the network directly through BS, we can see that the major contents in the messages are very similar except that the security zone key SZK is being used here to encrypt all the messages that are transmitted in this security zone (messages ⟨2⟩ and ⟨5⟩). This design is in accordance with the aforementioned security requirement of IEEE 802.16-2012 standard, which requires that trust is maintained within the security zone in order to support multihop relay system operations. At edge RS_B, RS_B decrypts the message it received with SZK and then forwards the decrypted messages to the requesting RS_A. In this case the message is still secured because it is still encrypted by AS with RS_A's public key.

If the requesting RS_A has been authenticated by AS within a valid period of time before it tries to connect to the current network, RS_A can skip phases (i) and (ii) and directly enter an optional phase (iii) in which a more efficient distributed authentication will be performed. The message format is described as follows:

(1) Requesting $RS_A \to$ All neighborhood nodes:

$MAC_{RS(A)} \| Seq_{\_RS(A)} \| CERT_{RS(A)} \| License_{\_RS(A)} \| MD[MAC_{RS(A)} \| Seq_{\_RS(A)} \| CERT_{RS(A)} \| License_{\_RS(A)}]$

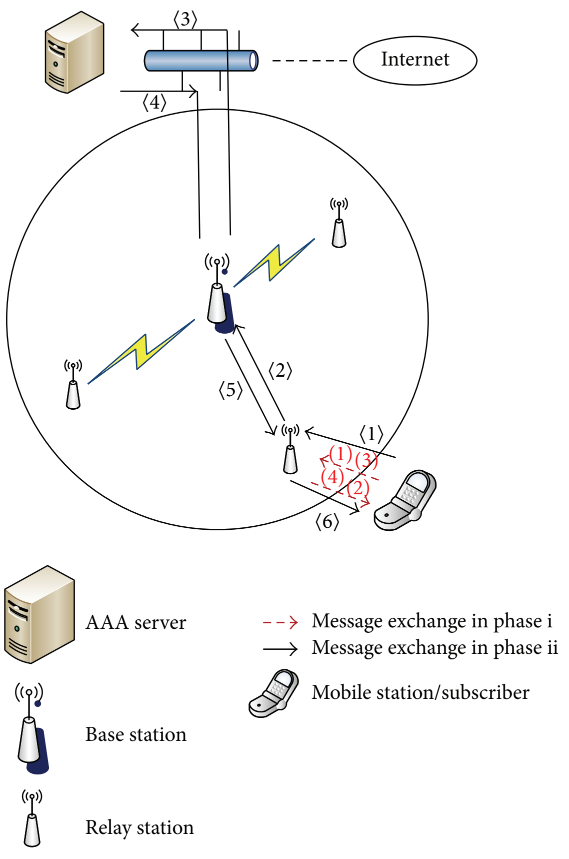

⟨3⟩
Internet
⟨4⟩
⟨2⟩
⟨5⟩
⟨1⟩
(1)
(3)
(4)
(2)
⟨6⟩
AAA server
Message exchange in phase i
Message exchange in phase ii
Mobile station/subscriber
Base station
Relay station

Figure 3: MS connects to MMR networks via an edge RS.

(2) Edge RS_{B} → Requesting RS_{A}:

$\mathrm{MAC}_{\mathrm{RS(B)}} \parallel \mathrm{CERT}_{\mathrm{RS(B)}} \parallel \mathrm{License}_{\_\mathrm{RS(B)}} \parallel \mathrm{MAC}_{\mathrm{BS}} \parallel E_{K_{\mathrm{RS_A\_PUB}}}[\mathrm{SZK}] \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{RS(B)}} \parallel \mathrm{Seq}_{\_\mathrm{RS(A)}} \parallel \mathrm{CERT}_{\mathrm{RS(B)}} \parallel \mathrm{License}_{\_\mathrm{RS(B)}} \parallel \mathrm{MAC}_{\mathrm{BS}} \parallel \mathrm{SZK}]$

In phase (iii), RS_{A} first broadcasts to all its neighborhood nodes message (1), which contains RS_{A}'s MAC address, sequence number, digital certificate, its license, and the message digest. Once a legitimate edge RS_{B} receives this message, it can use AS's public key to verify the authenticity of RS_{A}. If RS_{A} is verified to be authentic, then edge RS_{B} can send RS_{A} message (2) which includes RS_{B}'s license, certificate, security zone key, and the MAC address of BS which is in charge of this security zone. When RS_{A} gets message (2) and has the SZK and BS's MAC address, it can establish secure communications with BS to generate and exchange AK and TEK.

4.5. Scenario 3: MS Connects to the Network through RS. The third scenario in which a mobile user (MS) needs to connect to an MMR network through an edge RS is shown in Figure 3. In this scenario we have two cases to consider: the first one is when MS connects to the network for the first time through one or more RS; the second one is when MS has been authenticated by AS within a valid period of time before it tries to connect to the current network. For both cases, the message formats are similar to the message formats from the second scenario when a RS tries to get access to the network via other RS.

For MS which joins the network for the first time through an edge RS, there are two phases whose message format is specified as follows:

Phase (i) (initial verification of the edge RS)

(1) MS → Edge RS:

$\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{CERT}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}}]$

(2) Edge RS → MS:

$\mathrm{MAC}_{\mathrm{RS}} \parallel \mathrm{CERT}_{\mathrm{RS}} \parallel \mathrm{License}_{\_\mathrm{RS}} \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{RS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}}] \parallel E_{K_{\mathrm{RS\_PRV}}}[E_{K_{\mathrm{MS\_PUB}}}[\mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{Nnc}_{\_\mathrm{RS}}]]$

(3) MS → Edge RS:

$E_{K_{\mathrm{MS\_PRV}}}[E_{K_{\mathrm{RS\_PUB}}}[\mathrm{Nnc}_{\_\mathrm{RS}} \parallel \mathrm{Nnc}_{\_\mathrm{MS}}]]$

(4) Edge RS → MS:

$E_{K_{\mathrm{RS\_PRV}}}[E_{K_{\mathrm{MS\_PUB}}}[\mathrm{Nnc}_{\_\mathrm{MS}}]]$

Note that the four messages in phase (i) are similar to the four messages in phase (i) of Scenario 2 presented in the previous Section 4.4, because we need to prevent the possible replay attack launched by a malicious relay station. We will not repeat the explanation of the replay attack here because it has been explained in detail in the previous subsection.

Phase (ii) (dual authentication of MS and BS)

⟨1⟩ MS → Edge RS:

$\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{CERT}_{\mathrm{MS}} \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel K_{\mathrm{MS}}]$

According to IEEE 802.16-2012 standard, here relay stations do not try to decrypt the user date or authenticate the MAC management message they receive from mobile stations but simply relay it. And RS does not have any key information associate with the MS:

⟨2⟩ Edge RS → (possibly other RS in between) → BS:

$E_{K_{\mathrm{SZK}}}\{\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{CERT}_{\mathrm{MS}} \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel K_{\mathrm{MS}}]\}$

⟨3⟩ BS → AS:

$\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{CERT}_{\mathrm{MS}} \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel K_{\mathrm{MS}}] \parallel \mathrm{MAC}_{\mathrm{BS}} \parallel \mathrm{Seq}_{\_\mathrm{BS}} \parallel \mathrm{MD}[\mathrm{MAC}_{\mathrm{BS}} \parallel \mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{BS}} \parallel K_{\mathrm{BS}}] \parallel \mathrm{CERT}_{\mathrm{BS}}$

⟨4⟩ AS → BS:

$E_{K_{\mathrm{BS}}}\{\mathrm{MAC}_{\mathrm{AS}} \parallel \mathrm{MAC}_{\mathrm{MS}} \parallel \mathrm{Seq}_{\_\mathrm{BS}} \parallel \mathrm{PMK} \parallel \mathrm{License}_{\_\mathrm{BS}} \parallel \mathrm{SZK} \parallel \mathrm{MD}[\mathrm{PMK} \parallel \mathrm{SZK} \parallel K_{\mathrm{BS}}]\} \parallel E_{K_{\mathrm{MS}}}\{\mathrm{MAC}_{\mathrm{AS}} \parallel \mathrm{MAC}_{\mathrm{BS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{PMK} \parallel \mathrm{License}_{\_\mathrm{MS}} \parallel \mathrm{SZK} \parallel \mathrm{MD}[\mathrm{PMK} \parallel \mathrm{SZK} \parallel K_{\mathrm{MS}}]\}$

⟨5⟩ BS → (possibly other RS in between) → Edge RS:

$E_{K_{\mathrm{SZK}}}\{E_{K_{\mathrm{MS}}}\{\mathrm{MAC}_{\mathrm{AS}} \parallel \mathrm{MAC}_{\mathrm{BS}} \parallel \mathrm{Seq}_{\_\mathrm{MS}} \parallel \mathrm{PMK} \parallel \mathrm{License}_{\_\mathrm{MS}} \parallel \mathrm{SZK} \parallel \mathrm{MD}[\mathrm{PMK} \parallel \mathrm{SZK} \parallel K_{\mathrm{MS}}]\} \parallel \mathrm{MD}[\mathrm{PMK} \parallel \mathrm{AK} \parallel \mathrm{SZK}]\}$

⟨6⟩ Edge RS → MS:

$E_{K_{MS}}\{MAC_{AS} \parallel MAC_{BS} \parallel Seq_{\_MS} \parallel PMK \parallel License_{\_MS} \parallel SZK \parallel MD[PMK \parallel SZK \parallel K_{MS}]\} \parallel MD[PMK \parallel AK \parallel SZK]$

In message ⟨6⟩, after MS gets PMK and generates related AK and TEK, MS can verify its AK with the AK in the received message to check their consistency. TEK will be used to secure future communications between BS and MS.

For MS which has been authenticated by AS within a valid period of time before it tries to connect to the current network, a distributed authentication will be performed. The message format is specified as follows:

(1) MS → All neighborhood nodes:

$MAC_{MS} \parallel Seq_{\_MS} \parallel CERT_{MS} \parallel License_{\_MS} \parallel MD[MAC_{MS} \parallel Seq_{\_MS} \parallel CERT_{MS} \parallel License_{\_MS}]$

(2) Edge RS → MS:

$MAC_{RS} \parallel CERT_{RS} \parallel License_{\_RS} \parallel MAC_{BS} \parallel E_{K_{MS\_PUB}}[SZK] \parallel MD[MAC_{RS} \parallel Seq_{\_MS} \parallel CERT_{RS} \parallel License_{\_RS} \parallel MAC_{BS} \parallel SZK]$

MS broadcasts message (1) to all its neighborhood nodes. Message (1) contains MS's MAC address, sequence number, digital certificate, its license, and the message digest. Once a legitimate edge RS receives this message, it can use the AS's public key to verify the authenticity of MS. If MS is verified to be authentic, then edge RS can send MS message (2) which includes RS's license, certificate, security zone key between RS and MS, and the MAC address of BS which is in charge of this security zone. When MS gets message (2) and has the SZK and BS's MAC address, with SZK it can establish a secure communication with BS before it has TEK to encrypt its messages.

5. Formal Analysis of Proposed Protocols

Next, we present the formal analysis and verification of the proposed protocols using a tool called Scyther [8]. Scyther is an automated tool for verification, falsification, and analysis of security protocols [8]. Its effectiveness and correctness have been proved and its operational semantics can be found in [27] for interested readers. Scyther can verify protocols with unbounded number of sessions and roles, if computational resources allow [8]. It is also currently the only existing tool capable of verifying synchronization [27]. Synchronization is an important property in mutual authentication protocols. It indicates that all the messages are transmitted exactly in the order as described by the protocol, which can be used to detect suppress-replay attacks [8]. Hence we include it as one of the properties we need to verify in our proposed protocols, and we choose Scyther since it is the only tool which can do such verification [8]. To have the most accurate verification results, we choose the latest stable version Scyther v1.1.3 released in 2014.

5.1. Model Description. In our model, we describe the behavior of the protocol entities in terms of their roles, that is, an initiator or a responder (receiver) or both. For example, for protocol 1, we have three agents, RS, BS, and AS. All of them play two roles: message initiator and message responder. In Scyther, a *run* is a unique execution of a role that is performed by an agent; that is, each agent executes its runs to implement the protocol and preserve the secrecy of the credentials (e.g., keys, licenses, and sequence number) it claims to achieve. Each agent has several variable definitions for the credentials used in the messages this agent will send or receive when playing different roles. Each agent also has a sequence of events that entails what messages this agent will send or receive, along with its claims. In Scyther, a claim is defined as *claim (A, c, P)*, which means that agent *A* expects goal *c* to hold with parameter *P*. If an attack exists, then the claim the agent wants to achieve will not hold. Such a claim is called a falsified claim [8].

A claim has the locality property, so once agents get the messages, they will be able to view the state of the system from a local perspective. Therefore the protocol needs to make sure that the agent is able to have the knowledge of some properties of global state of the system from the local perspective; for example, the agent is able to know that something is beyond the intruder's knowledge, or that a specific agent is active. However, for the same protocol, the claim on the same secret credential based on different agents might not always hold; that is, it is possible that for agent *A* as the message initiator and agent *B* as the message responder *claim (A, c, P)* holds while *claim (B, c, P)* does not hold [8].

In order to verify a security protocol using Scyther, we need to specify an attacker and a set of agents executing several runs. Scyther will trace all possible attacks that the attacker might launch and determine whether a security claim the agent holds is true or attacks exist. In the verification of our proposed protocols, we focus on two security properties: *secrecy of keys* and *authentication*. To verify these two properties, we use two types of claims in our model: secrecy claim and authentication claim [8].

(i) Secrecy Claim. A secrecy claim is defined as *claim (A, Secret, P)*. It is a statement that the credentials *P* included in this claim by *A* will not be obtained or spoofed by the attacker. Secrecy means that the information in question is not to the knowledge of an attacker, even if it is transmitted over an untrusted network.

When agents are communicating data (public or secret credentials) with untrusted agents, the transmitted data is also open to the attacker. Although transmitted data is public in the air now, this does not imply that our protocol is broken. Rather, what we need is a secrecy claim saying that if an agent only speaks with trusted agents, then the data being transmitted or shared is kept in secret [8].

(ii) Authentication Claim. In Scyther, authentication focuses on the verification that when a role in a protocol is executed, we can guarantee that in the current network there exists at least one entity that is communicating with this role. However, only knowing that some entity is communicating with the role is not enough to guarantee the correctness and robustness of an authentication protocol; we want to use

Scyther to verify and show a stronger guarantee that when the protocol is being executed, the intended entity is aware of the communication, and the messages exchanged between the entity and the role follow the protocol description.

An authentication claim is defined as *claim (A, Nisynch)*. It means agent A sends and receives messages in the order which is exactly the same as what has been described in the protocol [8].

5.2. Security Properties to Be Verified. We aim to verify that our proposed protocols have the following three security properties: *information confidentiality*, *no theft of service possible* [28], and *message sequence synchronization*. Successful verification of these three properties can prove the robustness of our proposed protocols against the possible rogue RS and rogue BS attack described in Section 3. In our verification, we include an additional restriction that only claims concerning sessions between trusted agents are evaluated.

To illustrate our claims in the verification, we define a term *KeyFields*. Each data message exchanged between agents is composed of a set of key elements, for instance, pmk, ak, and szk. *KeyFields* is denoted as the set of elements. The three properties are explained in detail as follows.

5.2.1. Information Confidentiality. This property is satisfied if the access nodes, that is, RS and MS, can make sure that all exchanged keys are kept in secret. This property requires that each individual element α in *KeyFields* should be kept in secret.

The general formalization of the information confidentiality property is given below:

$$\forall\alpha \in \mathit{KeyFields}\colon \mathit{claim}\ (\mathrm{MS/RS}, \mathit{Secret}, \alpha)\ \text{holds}.$$

This formalization can be expanded into the following six claims:

Claim 1: *claim* (MS, *Secret*, pmk) holds.

Claim 2: *claim* (MS, *Secret*, ak) holds.

Claim 3: *claim* (MS, *Secret*, szk) holds.

Claim 4: *claim* (RS, *Secret*, pmk) holds.

Claim 5: *claim* (RS, *Secret*, ak) holds.

Claim 6: *claim* (RS, *Secret*, szk) holds.

5.2.2. No Theft of Service Possible. This property is satisfied if (i) AS can be ensured that neither an unauthorized BS nor an unauthorized RS can be able to impersonate a legitimate one and get access to the network, and (ii) BS has the guarantee that an unauthenticated RS cannot gain access to the services provided, nor could it impersonate a legitimate user. A service should always be bound to an authenticated user. This property is similar to the information confidentiality property but involves different agents of the protocol. Its formal definition is given as follows:

$$\forall\alpha \in \mathit{KeyFields}\colon \mathit{claim}\ (\mathrm{AS/BS}, \mathit{Secret}, \alpha)\ \text{holds}.$$

This formalization is expanded into claims 7–13:

Table 1

Key element	Description
PMK	Pairwise Master Key
AK	Authorization Key
SZK	A group key used in the security zone among BS and many RSs
k(R, A)	K_{RS}, RS's secret key shared with AS
k(B, A)	K_{BS}, BS's secret key shared with AS

Claim 7: *claim* (AS, *Secret*, pmk) holds.

Claim 8: *claim* (AS, *Secret*, szk) holds.

Claim 9: *claim* (AS, *Secret*, k(R, A)) holds.

Claim 10: *claim* (AS, *Secret*, k(B, A)) holds.

Claim 11: *claim* (BS, *Secret*, pmk) holds.

Claim 12: *claim* (BS, *Secret*, ak) holds.

Claim 13: *claim* (BS, *Secret*, szk) holds.

5.2.3. Message Sequence Synchronization. This property is satisfied if all of RS, BS, and AS can be ensured that the corresponding send and receive messages are executed in the order exactly the same as what has been specified in the protocol. Its formal definition is given as follows:

Claim 14: *claim* (RS, *Nisynch*) holds.

Claim 15: *claim* (BS, *Nisynch*) holds.

Claim 16: *claim* (AS, *Nisynch*) holds.

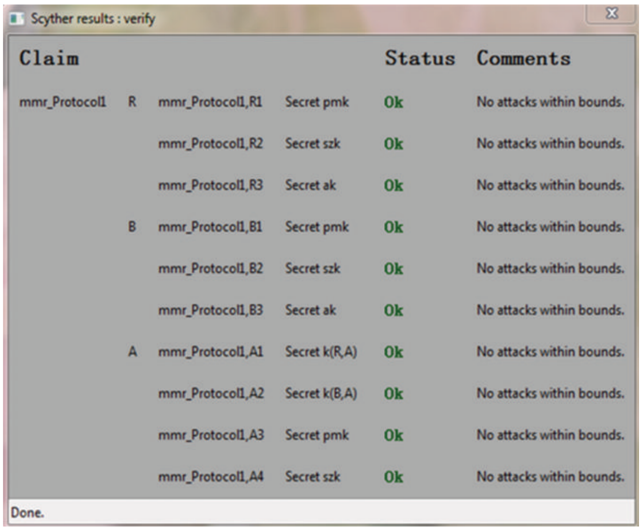
Claim 17: *claim* (MS, *Nisynch*) holds.

5.3. Formal Verifications. Table 1 shows the message elements that are used in our formal model.

Recall that, according to our protocol description in Section 4, several keys will be transmitted. Hence we need to verify our protocols to see whether they can satisfy our secrecy claim and authentication claim.

Below are the specific verification results from Scyther. There are three parts in each of the following figures of verification results. The first part is "Claim" which contains several columns indicating which element α in KeyFields should be kept in secret in a specific node. For example, "R1, Secret pmk" represents key pmk in node R1 should be kept secret. The second part is "Status" which indicates the status of verification with a variety of possible attacks. "Ok" means the specific claim holds (i.e., passes the verification). "Failed" means possible attack(s) exists in this claim, indicating that the design of the authorization protocol might have some potential flaws. The third part is "Comments"; this part provides more specific information regarding the status. The expected results of our protocols are that all claims should hold; that is, we expect to see all status as "Ok" and no attacks are found.

5.3.1. Protocol 1: RS Connects to the Network through BS. From the verification results in Figure 4, we can see that all of our

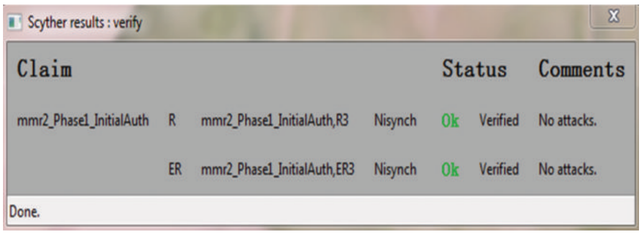


Scyther results : verify

Claim				Status	Comments
mmr_Protocol1	R	mmr_Protocol1,R1	Secret pmk	Ok	No attacks within bounds.
		mmr_Protocol1,R2	Secret szk	Ok	No attacks within bounds.
		mmr_Protocol1,R3	Secret ak	Ok	No attacks within bounds.
	B	mmr_Protocol1,B1	Secret pmk	Ok	No attacks within bounds.
		mmr_Protocol1,B2	Secret szk	Ok	No attacks within bounds.
		mmr_Protocol1,B3	Secret ak	Ok	No attacks within bounds.
	A	mmr_Protocol1,A1	Secret k(R,A)	Ok	No attacks within bounds.
		mmr_Protocol1,A2	Secret k(B,A)	Ok	No attacks within bounds.
		mmr_Protocol1,A3	Secret pmk	Ok	No attacks within bounds.
		mmr_Protocol1,A4	Secret szk	Ok	No attacks within bounds.

Done.

Figure 4: Verification result for Protocol 1.



Scyther results : verify

Claim				Status		Comments
mmr2_Phase1_InitialAuth	R	mmr2_Phase1_InitialAuth,R3	Nisynch	Ok	Verified	No attacks.
	ER	mmr2_Phase1_InitialAuth,ER3	Nisynch	Ok	Verified	No attacks.

Done.

Figure 5: Verification result for Protocol 2-Phase 1, initial authentication.

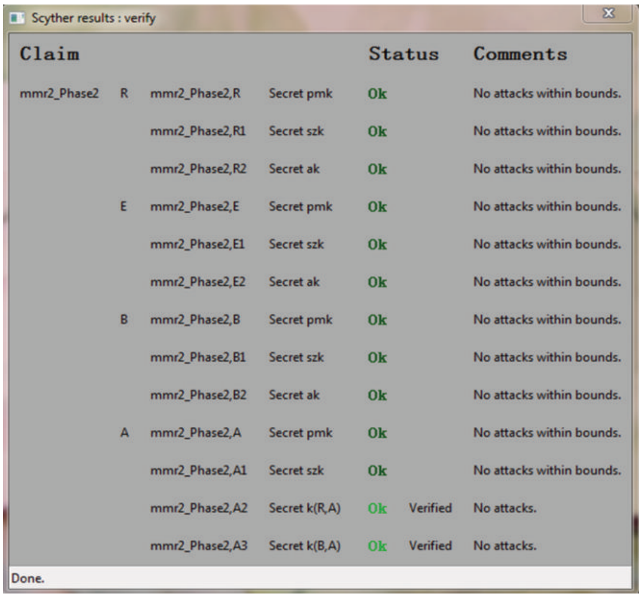
properties from Claim 4 to Claim 13 in Section 5.2 regarding key elements hold and no possible attacks are detected. This proves that our goals of *information confidentiality* and *no theft of service possible* are satisfied. Therefore, all the key information can be transmitted and exchanged safely.

5.3.2. Protocol 2: RS Connects to the Network through Other RS(s)

(i) Verification Results for Phase 1. From the verification results in Figure 5, we can see that Claim 14 holds and no possible attacks are detected. This proves that our goal of message sequence synchronization is satisfied, and the requesting RS can successfully authenticate a legitimate edge RS.

(ii) Verification Results for Phase 2. The verification results for phase 2 are shown in Figure 6, from which we can see that all of our claims from Claim 4 to Claim 13 regarding key element are held and no possible attacks are detected. This proves that our goals of *information confidentiality* and *no theft of service possible* are satisfied. Therefore all the key information can be transmitted and exchanged safely.

(iii) Verification Results for Phase 3. The verification results for phase 3 are shown in Figure 7, from which we can see that Claim 6 regarding the confidentiality of SZK holds and no possible attacks are detected. This proves that our goals of information confidentiality and no theft of service possible are satisfied.

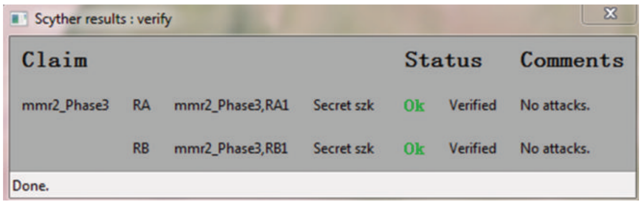


Scyther results : verify

Claim				Status		Comments
mmr2_Phase2	R	mmr2_Phase2,R	Secret pmk	Ok		No attacks within bounds.
		mmr2_Phase2,R1	Secret szk	Ok		No attacks within bounds.
		mmr2_Phase2,R2	Secret ak	Ok		No attacks within bounds.
	E	mmr2_Phase2,E	Secret pmk	Ok		No attacks within bounds.
		mmr2_Phase2,E1	Secret szk	Ok		No attacks within bounds.
		mmr2_Phase2,E2	Secret ak	Ok		No attacks within bounds.
	B	mmr2_Phase2,B	Secret pmk	Ok		No attacks within bounds.
		mmr2_Phase2,B1	Secret szk	Ok		No attacks within bounds.
		mmr2_Phase2,B2	Secret ak	Ok		No attacks within bounds.
	A	mmr2_Phase2,A	Secret pmk	Ok		No attacks within bounds.
		mmr2_Phase2,A1	Secret szk	Ok		No attacks within bounds.
		mmr2_Phase2,A2	Secret k(R,A)	Ok	Verified	No attacks.
		mmr2_Phase2,A3	Secret k(B,A)	Ok	Verified	No attacks.

Done.

Figure 6: Verification result for Protocol 2-Phase 2, mutual authentication.



Scyther results : verify

Claim				Status		Comments
mmr2_Phase3	RA	mmr2_Phase3,RA1	Secret szk	Ok	Verified	No attacks.
	RB	mmr2_Phase3,RB1	Secret szk	Ok	Verified	No attacks.

Done.

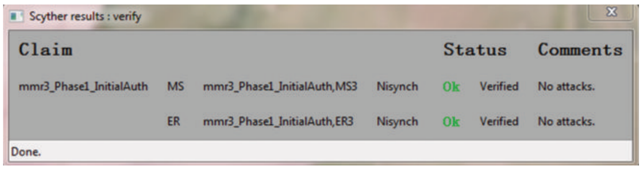
Figure 7: Verification result for Protocol 2-Phase 3, distributed authentication.

5.3.3. Protocol 3: MS Connects to the Network through RS(s)

(i) Verification Results for Phase 1. The verification results for phase 1 are shown in Figure 8, from which we can see that Claims 17 and 14 hold and no possible attacks are detected. This proves that our goal of message sequence synchronization is satisfied, and the MS can successfully authenticate a legitimate edge RS.

(ii) Verification Results for Phase 2. From Figure 9, we can see that all of our claims from Claim 1 to Claim 13 regarding key element are held and no possible attacks are detected. This proves that our goals of *information confidentiality* and *no theft of service possible* are satisfied. Therefore all the key information can be transmitted and exchanged safely.

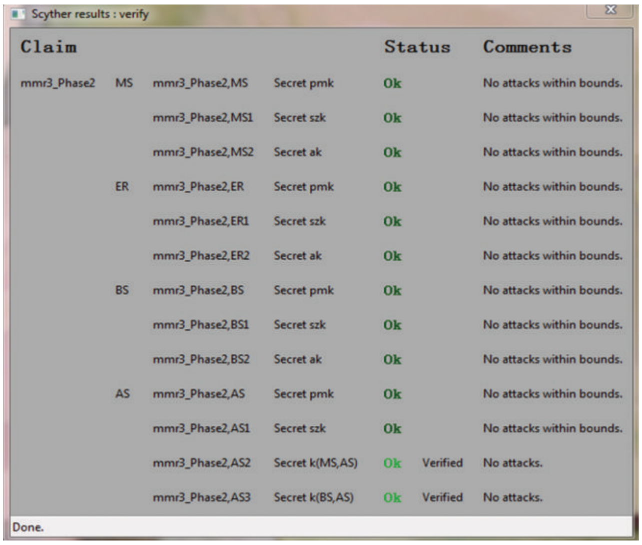
(iii) Verification Results for Phase 3. The verification results for phase 3 are shown in Figure 10, from which we can see that Claims 6 and 3 regarding the confidentiality of SZK hold and no possible attacks are detected. This proves that our goals of



Claim				Status		Comments
mmr3_Phase1_InitialAuth	MS	mmr3_Phase1_InitialAuth,MS3	Nisynch	Ok	Verified	No attacks.
	ER	mmr3_Phase1_InitialAuth,ER3	Nisynch	Ok	Verified	No attacks.

Done.

Figure 8: Verification result for Protocol 3-Phase 1, initial authentication.



Claim				Status		Comments
mmr3_Phase2	MS	mmr3_Phase2,MS	Secret pmk	Ok		No attacks within bounds.
		mmr3_Phase2,MS1	Secret szk	Ok		No attacks within bounds.
		mmr3_Phase2,MS2	Secret ak	Ok		No attacks within bounds.
	ER	mmr3_Phase2,ER	Secret pmk	Ok		No attacks within bounds.
		mmr3_Phase2,ER1	Secret szk	Ok		No attacks within bounds.
		mmr3_Phase2,ER2	Secret ak	Ok		No attacks within bounds.
	BS	mmr3_Phase2,BS	Secret pmk	Ok		No attacks within bounds.
		mmr3_Phase2,BS1	Secret szk	Ok		No attacks within bounds.
		mmr3_Phase2,BS2	Secret ak	Ok		No attacks within bounds.
	AS	mmr3_Phase2,AS	Secret pmk	Ok		No attacks within bounds.
		mmr3_Phase2,AS1	Secret szk	Ok		No attacks within bounds.
		mmr3_Phase2,AS2	Secret k(MS,AS)	Ok	Verified	No attacks.
		mmr3_Phase2,AS3	Secret k(BS,AS)	Ok	Verified	No attacks.

Done.

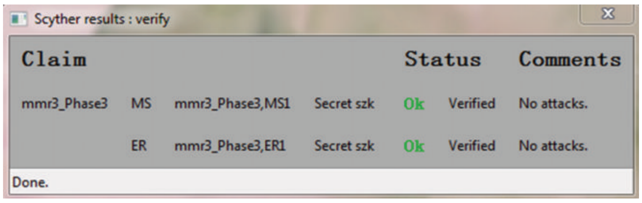
Figure 9: Verification result for Protocol 3-Phase 2, initial authentication.

information confidentiality and *no theft of service possible* are satisfied.

As seen in the above formal analysis, our claims for the secrecy and uniqueness of the exchanged key elements, the no theft of service possible, and message sequence synchronization are satisfied in all of the three protocols.

6. Concluding Remarks

In this paper, we present a set of new secure authentication protocols to address the attack of rogue BS/RS attack in the MMR WiMAX networks. First, relay station and mobile user authentication provide access control. Second, in order to protect the MMR network from rogue BS attack, an authentication server is used to conduct mutual authentication for both BS and RS/MS in the network. In this way, a rogue station can be detected at early stage and its harm to this network can be minimized. Third, a security zone key is generated and securely delivered to an authenticated BS by AS. BS can then distribute this SZK to legitimate RS within the network; in this way the communications within the network can be secured using SZK. Fourth, in order to reduce the overhead introduced by mutual authentications between nodes, a license and a distributed authentication are used as a pass ticket for nodes which have been authenticated within a valid period of time. Such a node with a valid license can get access to the network without having to go through the complete mutual authentication in AS; and a node with a valid license is trustworthy to give away SZK or provide authentication message forwarding. This method is useful for reducing the overhead especially when fast handoff happens frequently. To verify the security properties of our protocols, we apply the Scyther tool to conduct a formal verification. The verification does not find any attack with our protocols, which proves that our protocols satisfy three desirable security goals, namely, *information confidentiality*, *no theft of service possible*, and *message sequence synchronization*.



Claim				Status		Comments
mmr3_Phase3	MS	mmr3_Phase3,MS1	Secret szk	Ok	Verified	No attacks.
	ER	mmr3_Phase3,ER1	Secret szk	Ok	Verified	No attacks.

Done.

Figure 10: Verification result for Protocol 3-Phase 3, distributed authentication.

In the future work, we will perform a numerical analysis on the authentication performance of our protocols in terms of key processing time, response time, and total overhead of provisioning the dual authentication, the security zone key, and the license.

Competing Interests

The authors declare that they have no competing interests.

References

[1] 3GPP.org, "LTE (Long Term Evolution)," http://www.3gpp.org/technologies/keywords-acronyms/98-lte.

[2] Kyle, "Intel Capital: WiMAX Is Not Dead," http://www.nibletz.com/international/intel-capital-wimax-is-not-dead/.

[3] L. Tie and Y. Yi, "Extended security analysis of multi-hop ticket based handover authentication protocol in the 802.16j network," in *Proceedings of the 8th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '12)*, pp. 1–10, Shanghai, China, September 2012.

[4] S. W. Peters and R. W. Heath, "The future of WiMAX: multihop relaying with IEEE 802.16j," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 104–111, 2009.

[5] IEEE Standard for Air Interface for Broadband Wireless Access Systems, "IEEE Std 802.16-2012 (Revision of IEEE Std 802.16-2009)," pp. 1–2542, August 17, 2012.

[6] X. Dai and X. Xie, "Analysis and research of security mechanism in IEEE 802.16j," in *Proceedings of the International Conference on Anti-Counterfeiting, Security and Identification (ASID '10)*, pp. 33–36, IEEE, Chengdu, China, July 2010.

[7] J. Huang and C.-T. Huang, "Secure mutual authentication protocols for mobile multi-hop relay WiMAX networks against Rogue base/relay stations," in *Proceedings of the IEEE International Conference on Communications (ICC '11)*, pp. 1–5, IEEE, Kyoto, Japan, June 2011.

[8] C. Cremers, "The scyther tool: verification, falsification, and analysis of security protocols," in *Computer Aided Verification:*

20th International Conference, CAV 2008 Princeton, NJ, USA, July 7–14, 2008 Proceedings, vol. 5123 of *Lecture Notes in Computer Science*, pp. 414–418, Springer, Berlin, Germany, 2008.

[9] S. Xu, M. Matthews, and C.-T. Huang, "Security issues in privacy and key management protocols of IEEE 802.16," in *Proceedings of the 44th ACM Southeast Regional*, pp. 113–118, Melbourne, Fla, USA, March 2006.

[10] S. Xu and C.-T. Huang, "Attacks on PKM protocols of IEEE 802.16 and its later versions," in *Proceedings of the 3rd International Symposium on Wireless Communication Systems (ISWCS '06)*, pp. 185–189, IEEE, Valencia, Spain, September 2006.

[11] D. Johnston and J. Walker, "Overview of IEEE 802.16 security," *IEEE Security and Privacy*, vol. 2, no. 3, pp. 40–48, 2004.

[12] F. Yang, H. Zhou, L. Zhang, and J. Feng, "An improved security scheme in WMAN based on IEEE standard 802.16," in *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 1191–1194, September 2005.

[13] K. Khan and M. Akbar, "Authentication in multi-hop wireless mesh networks," *Transactions on Science, Engineering and Technology*, vol. 16, pp. 178–183, 2006.

[14] M. E. Mahmoud and X. Shen, "Anonymous and authenticated routing in multi-hop cellular networks," in *Proceedings of the IEEE International Conference on Communications (ICC '09)*, pp. 1–6, IEEE, Dresden, Germany, June 2009.

[15] S. Zhu, S. Setia, S. Jajodia, and P. Ning, "An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 259–271, May 2004.

[16] Y. Fan and Q. Yali, "Two different schemes of authentication in IEEE 802.16j multi-hop relay network," in *Proceedings of the 8th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '12)*, pp. 1–4, Shanghai, China, September 2012.

[17] H. X. Jin, L. Tu, G. Yang, and Y. Yang, "An improved mutual authentication scheme in multi-hop WiMax network," in *Proceedings of the International Conference on Computer and Electrical Engineering (ICCEE '08)*, pp. 296–299, Phuket, Thailand, December 2008.

[18] A. S. Khan, N. Fisal, Z. A. Bakar et al., "Secure authentication and key management protocols for mobile multihop WiMAX networks," *Indian Journal of Science and Technology*, vol. 7, no. 3, pp. 282–295, 2014.

[19] A. S. Khan, H. Lenando, and J. Abdullah, "Lightweight message authentication protocol for mobile multihop relay networks," *International Review on Computers and Software*, vol. 9, no. 10, pp. 1720–1730, 2014.

[20] Y. Lee, H. K. Lee, G. Y. Lee, H. J. Kim, and C. K. Jeong, "Design of hybrid authentication scheme and key distribution for mobile multi-hop relay in IEEE 802.16j," in *Proceedings of the Euro American Conference on Telematics And Information Systems: New Opportunities to Increase Digital Citizenship (EATIS '09)*, Prague, Czech Republic, June 2009.

[21] A. S. Khan, N. Fisal, S. Kamilah, and M. Abbas, "Efficient distributed authentication key scheme for multi-hop relay in IEEE 802.16j network," *International Journal of Engineering Science and Technology*, vol. 2, pp. 2192–2199, 2010.

[22] D. Zhu, N. Pang, and Z. Fan, "A self-testing approach defending against rogue base station hijacking of intelligent terminal," in *Proceedings of the International Conference on Applied Science and Engineering Innovation*, Zhengzhou, China, May 2015.

[23] M. Barbeau and J.-M. Robert, "Rogue-base station detection in WiMax/802.16 wireless access networks," *Annals of Telecommunications*, vol. 61, no. 11-12, pp. 1300–1313, 2006.

[24] T. Shon and W. Choi, "An analysis of mobile WiMAX security: vulnerabilities and solutions," in *Proceedings of the 1st International Conference on Network-Based Information Systems*, Regensburg, Germany, September 2007.

[25] M. Maxim and D. Pollino, *WiMAX Security*, RSA Press, McGraw-Hill/Osborne, Berkeley, Calif, USA, 2002.

[26] M. Khosrowpour, "Managing information technology resources in organizations in the next millennium," in *Proceedings of the Information Resources Management Association International Conferances*, May 1999.

[27] C. J. Cremers, S. Mauw, and E. P. de Vink, "Injective synchronisation: an extension of the authentication hierarchy," *Theoretical Computer Science*, vol. 367, no. 1-2, pp. 139–161, 2006.

[28] E. Kaasenbrood, *WiMAX Security—a formal and informal analysis [M.S. thesis]*, Eindhoven University of Technology, Department of Mathematics and Computer Science, Groningen, The Netherlands, 2006.

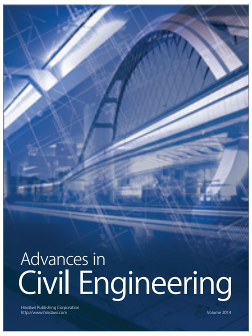
Journal of
Engineering

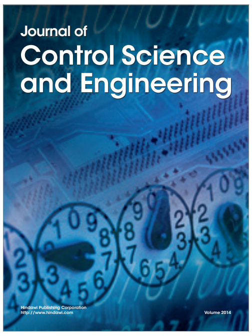
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

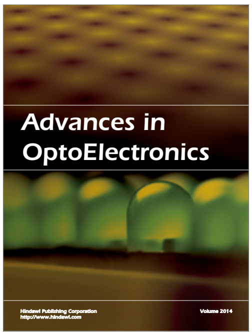
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

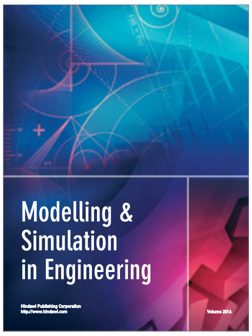
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

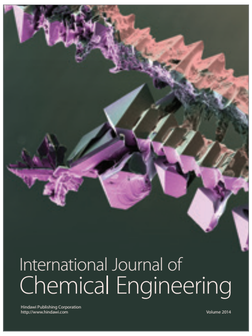

Advances in
OptoElectronics

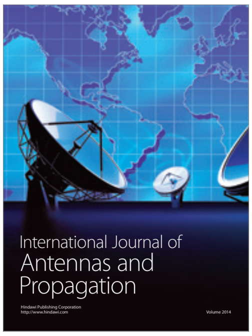
VLSI Design

International Journal of
Navigation and
Observation

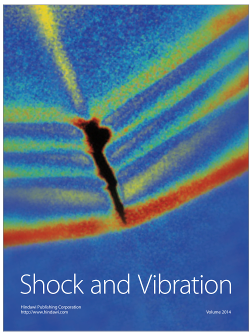

Modelling &
Simulation
in Engineering

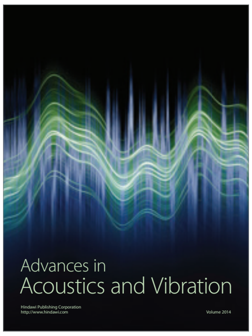
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration