Carleton University

Local Authentication in WiMAX

COMP 4905: Honours Project

Jonathan Désilets
4/20/2008

Faculty Advisor: Michel Barbeau, School of Computer Science

I. ABSTRACT

The IEEE 802.16 standard Privacy and Key Management (PKM) protocol suffers from a number of performance and security concerns which make it unsuitable for use in a WiMAX network operating in Mesh Mode. Reliance on centralized base stations to perform node authentication creates a bottleneck that could be avoided completely through the addition of local authentication mechanisms. After careful review and study of a number of existing local authentication schemes, proposed modifications to the PKM protocol are made which incorporates a Threshold RSA authentication scheme to achieve local authentication as well as enhanced security and robustness. The proposed scheme is shown to be of comparable performance to the original PKM protocol in the worst case and substantially faster as the distance between candidate nodes and base stations increase. Furthermore, the proposed protocol is demonstrated to be considerably more secure and resistant to attack than the original PKM protocol.

II. ACKNOWLEDGEMENTS

I would like to offer my sincere thanks to my faculty advisor, Dr. Michel Barbeau, for all of his help and support throughout the completion of my honours project. His aid was invaluable and greatly appreciated.

III. TABLE OF CONTENTS

IV. LIST OF TABLES

1 INTRODUCTION

WiMAX is an emerging telecommunications technology based off the IEEE 802.16 standard which offers enticing long-distance wireless capabilities and the practical establishment of Metropolitan Area Networks (MANs). WiMAX remains, however, a relatively new development in wireless technologies and the resource base of research on the subject is still relatively small, though it continues to grow steadily. One area where the amount of existing research is particularly small includes alternate means of authentication within the IEEE 802.16 standard protocol, specifically modifying the authentication protocols for a WiMAX network operating in Mesh mode to perform local authentication. The default behaviour of a WiMAX network operating in the Mesh mode is for new candidate nodes (CN) to be authenticated by one of a few centralized Base Stations (BS). This authentication is performed by connecting to a sponsor node (SN) and passing messages through other Subscriber Stations (SS) already within the network should the candidate node not have a direct communication link to a base station. Not only does this behaviour result in a number of security concerns, it also consumes a substantial amount of traffic and message passing that may prove to be unnecessary. If subscriber stations could locally authenticate candidate nodes wishing to connect to the network without having to exchange a lengthy series of messages with base stations both the network usage and authentication times could be substantially cut down.

This paper is structured as follows: Section 2 examines WiMAX in more detail, focusing specifically on the existing authentication protocols described in the IEEE 802.16 standard and the resulting performance problems and security concerns. Section 3 moves on to take a closer look at existing authentication schemes which support local authentication and were considered for adaption into the IEEE 802.16 standard. Section 4 provides a more in-depth view of the threshold authentication scheme which was selected as the candidate to be adapted for use with WiMAX. Section 5 culminates into a detailed description of the proposed authentication protocol which supports local authentication and enhanced security over the standard authentication scheme detailed in the IEEE 802.16 standard. Finally, Section 6 concludes the report and offers a summary of the results as well as direction for further research on the topic.

2 WIMAX

2.1 OVERVIEW

WiMAX, otherwise known as the Worldwide Interoperability for Microwave Access, is a wireless telecommunications technology with great potential to enhance and support the wireless needs of today's ever changing society. Based entirely on the IEEE 802.16 standard, WiMAX's protocol is completely standardized and fully detailed. For the purposes of this report, however, the majority of the 802.16 standard is out of scope: the focus will remain exclusively on the portion of the standard detailing authentication, specifically authentication in Mesh mode. As has been touched on indirectly elsewhere, WiMAX supports a number of modes of operations including, but not limited to Point to Multipoint (PMP) and Mesh mode. Mesh mode is of particular interest because of its sheer versatility and ability to most accurately handle perhaps the most desirable type of wireless network: the wireless mesh. With a wireless mesh in place, the entire network can be abstracted to a wireless "cloud" of nodes, complete with redundancy and support for node failures. Despite its dynamic and demanding natures, the existing authentication protocols detailed in the IEEE 802.16 standard for use in Mesh mode are sorely lacking in the areas of both performance and security.

WiMAX's default authentication protocol is part of the IEEE 802.16 standard's Privacy and Key Management (PKM) protocol [IEEE 802.16 Working Group on Broadband Wireless Access, 2004]. When a new Subscriber Station (SS) wishes to join the network it is required to authenticate itself to one of the Base Stations (BS) of the wireless mesh. In order to accomplish this messages are passed between the new SS and the BS using other SSs of the network as intermediaries. Furthermore, WiMAX capable wireless devices are all in possession of a manufacturer issued X.509 digital certificate and these certificates are used throughout the protocol to ensure a measure of security. The actual authentication portion of the PKM protocol is not particularly complex or difficult; in fact, the majority of the complexity results from the network having to forward messages back and forth between the new SS and one of the BSs.

Section 2.2 provides an outline of all relevant messages used throughout the PKM authentication protocol, while Section 2.3 details an example authentication exchange between a new SS and an existing WiMAX mesh.

Finally, Section 2.4 offers an analysis of the authentication portion of the PKM protocol with specific focus on performance and security concerns.

2.2 PKM AUTHENTICATION MESSAGES

2.2.1 AUTHENTICATION INFORMATION MESSAGE

The Auth-Info message (see Table 1 below) contains the SS manufacturer's X.509 certificate, issued by the manufacturer itself or by an external authority.

Table 1 - Authentication Information Message Attributes (PKM)

Attribute	Contents
CA-Certificate	Certificate of manufacturer CA that issued SS certificate.

2.2.2 AUTHENTICATION REQUEST MESSAGE

The Auth-Req message (see Table 2 below) contains:

- A manufacturer-issued X.509 certificate.
- A description of the cryptographic algorithms the requesting SS supports.
- The SS's Basic Connection Identifier (CID).

Table 2 - Authentication Request Attributes (PKM)

Attribute	Contents
SS-Certificate	Contains the SS's X.509 user certificate
Security-Capabilities	Describes requesting SS's security capabilities.
SAID	SS's primary SAID equal to the Basic CID.

2.2.3 AUTHENTICATION REPLY MESSAGE

The Auth-Reply message (see Table 3 below) contains:

- An Authorization Key (AK) encrypted with the SS's public key.
- A 4-bit key sequence number, used to distinguish between successive generations of AKs.
- A key lifetime for the AK.
- The identities (i.e. SAIDs) and properties of the single primary and zero or more static Security Associations (SA) the SS is authorized to obtain keying information from.

- An operator shared secret key known to all nodes in the network.
- A 4-bit key sequence number, used to distinguish between successive generations of operator shared secret keys.
- A key lifetime for the operator shared secret key.

Table 3 - Authentication Reply Attributes (PKM)

Attribute	Contents
AUTH-Key	Authorization (AUTH) Key, encrypted with the target SS's public key.
Key-Lifetime	AK's active lifetime.
Key-Sequence Number	AK sequence number.
(one or more) SA Descriptor(s).	Each compound SA-Descriptor attribute specifies an SAID and additional properties of the SA.
PKM Configuration Settings (optional)	PKM time values.
Operator Shared Secret	Mesh Mode Only: Key known to all.
Key-Sequence Number	Mesh Mode Only: Sequence number of the Operator Shared Secret.
Key-Lifetime	Mesh Mode Only: Lifetime of the Operator Shared Secret.

2.2.4 AUTHENTICATION REJECT MESSAGE

The Auth-Reject message (see Table 4 below) contains:

- An error code detailing the reason for the rejection of the authorization request.
- Optionally, a string detailing the error in plain English.

Table 4 - Authentication Reject Message (PKM)

Attribute	Contents
Error-Code	Error code identifying reason for rejection of authorization request.
Display-String (Optional)	Display String providing reason for rejection of authorization request.

2.3 EXAMPLE

Suppose a subscriber station, *A*, wishes to join an established network. *A* will henceforth be known as the Candidate Node (CN). *A* is not in range of the base station *B*, but it *is* in range of another subscriber station already in the network: *C*. Since *A* cannot talk directly to *B* it will instead talk to *C*, trusting *C* to forward the messages on to *B*. At this point, *C* has become *A*'s Sponsor Node (SN). The first message *A* will send is the Auth-Info message, which contains the X.509 certificate of *A*'s manufacturer which also happens to be the Certificate Authority (CA) for *A*'s own X.509 certificate. Before waiting for a reply, *A* will then immediately send an Auth-Req message as well. This is because the Auth-Info message is completely informative and base stations are permitted to completely

ignore it if desired; since there will never be a response to the Auth-Info message, *A* is free to send the Auth-Req message immediately. As soon as *C* receives the Auth-Info and Auth-Req messages *C* will use its own knowledge of the network to forward the messages in such a way that they will eventually reach the base station *B*. If *C* has a direct link to *B* then the forwarding is straightforward and simple; if *C* does not have a direct link to *B* it will simply forward the messages through other intermediary subscriber stations until they finally reach *B*.

When *B* receives the Auth-Req message from *A* it will either accept or reject the authentication request. Should *A*'s authentication be accepted *B* will construct and return an Auth-Reply message, using *A*'s public key from its X.509 certificate to encrypt the sensitive portions of the data. Alternatively, *B* will return an Auth-Reject message with the appropriate error code. In either case, this message will trace the same route back to *C* that the Auth-Req message took to reach *B* only in reverse. Once *C* receives the reply it will forward the message back to *A*, resulting in *A* either being fully authenticated or rejected and forced to attempt authentication again. In either case, the authentication portion of the PKM protocol will have completed.

2.4 ANALYSIS

Despite the fact that the authentication scheme in the PKM protocol is both simple to understand and relatively easy to implement there are a number of very real concerns with the protocol, the first of which is performance. As it was demonstrated in the example above, the requirement for all candidate nodes wishing to join the network to authenticate themselves directly to a base station can result in a great deal of superfluous messages. Whenever authentication messages have to be routed through the network in order to open communication between a candidate node and a base station needless work is being performed and network bandwidth is being wasted. This observation is the primary motivation to incorporate local authentication within the IEEE 802.16 standard. If a candidate node can simply talk to and authenticate itself locally with its sponsor node there will never be a need to route a lengthy series of messages back and forth to the base station. Not only does this improve the performance of the network, but it also enhances the network's ability to handle faults and hardware failures. Centralized control areas such as base stations are prone to becoming bottlenecks as well as liabilities should they fail for any reason; decentralized approaches such as local authentication help dramatically to alleviate these occurrences, resulting in a more robust network.

In addition to the performance concerns intrinsic to the PKM protocol's authentication scheme, there are also a number of very relevant security concerns. One of the first things to notice about WiMAX's default authentication scheme is that there is only one-way authentication: the candidate node authenticates itself to the network's base station, but there is no authentication whatsoever in the opposite direction. Consider the case where a malicious node is masquerading as a sponsor node or base station in an effort to subvert candidate nodes; using the PKM protocol as described, there is no way for candidate nodes to verify the identity of its sponsor node, nor even determine if the sponsor node is truly a member of the network. This can lead to a number of cases where an innocent candidate node is manipulated and prevented from engaging in normal network activities should it become the target of a malicious sponsor node.

Additionally, the PKM protocol's authentication scheme is also vulnerable to message replay attacks. Careful examination of the authentication messages will quite clearly reveal that there is no transient information anywhere in the exchange; there is no way for either the candidate node or the sponsor node to determine the liveness of the message exchange. Any of the messages or replies could just as easily be replays of old requests and replies. This vulnerability can quite easily result in candidate nodes being manipulated into believing they have completed authentication when they have really only been sent a replay of a past Auth-Reply message or of base stations unwittingly authenticating malicious nodes who have simply used a replay of a past Auth-Req message.

Perhaps the most serious vulnerability in the PKM protocol's authentication scheme is its susceptibility to man-in-the-middle attacks as a result of the lack of sponsor node authentication and enforced message integrity. As has been outlined earlier, in WiMAX's default operation it is not difficult at all to take advantage of a candidate node, but this vulnerability can be extended further to full-scale man-in-the-middle-attacks since message integrity is not enforced at all. Any and all of the authentication messages used throughout the PKM protocol are completely susceptible to being modified maliciously in transit, there are no mechanisms within the protocol to ensure message integrity. The full scope of man-in-the-middle attacks are beyond the scope of this report, but attacks of this nature are both serious and practical and can escalate further into such things as Denial of Service attacks.

3 LOCAL AUTHENTICATION SCHEMES

Given the clear advantages that result from a decentralized approach to networking it should come as no surprise that there exist a great variety of local authentication schemes. During the course of attempting to incorporate local authentication into the IEEE 802.16 standard a number of these schemes were examined and reviewed in order to gauge their suitability for use in WiMAX. The following subsections will examine a few of the more prominent implementations of local authentication developed for use in wireless networks and present the conclusions that were regarding their suitability.

3.1 PRE-SHARED KEY

Likely the most basic method of local authentication available, the use of a pre-shared key is both simple and straightforward. Every subscriber station that is a valid member of the network is in possession of a pre-shared and secret key; authentication occurs by each party involved simply proving to the other that they have knowledge of the secret key, which can easily be performed locally. Dellutri et al. proposed this technique as a means to allow Bluetooth enabled devices to operate as a Personal Trusted Device for purposes of locking and unlocking a laptop [Dellutri, Me, & Strangio, 2005]. The local authentication mechanism involved was in creating a secure channel between a Bluetooth device and the laptop to allow for proper communication. The proposed scheme used a Diffie Hellmann cryptosystem to construct a shared and secure private session key between the two devices. In order to ensure a man-in-the-middle attack could not succeed, a pre-shared key was deemed necessary. This requirement was added to ensure a man-in-the-middle attack would be unable to substitute the public key data used during the key-construction process with arbitrary values of its own choosing, thus compromising the security of the session key.

As an alternative to a pre-shared key, Dellutri et al. also proposed using a midlet on all the devices which would be downloaded by a trusted computer that acts as a Midlet Distribution Center, with the key used to authenticate to this trusted computer firmly embedded into the code itself. The only real benefit that the use of a midlet would provide is user convenience: the user would no longer be prompted to enter the pre-shared key during authentication procedures as the midlet would handle it. This solution, of course, requires a great deal of additional systems and resources and it is unlikely that these costs are worth simple ease of use. Overall, the use of a pre-shared

key was deemed too simple and well known to incorporate into the IEEE 802.16 standard. The concept is not particularly innovative and even with a pre-shared key there are a number of security concerns that remain.

3.2 AUTHENTICATION, AUTHORIZATION, ACCOUNTING (AAA) ARCHITECTURE

Given the prevalence of the Authentication, Authorization, Accounting (AAA) architecture in today's wireless environment it should come as no surprise that many local authentication schemes build heavily upon the underlying architecture that are already in existence. Liang & Wang proposed a local authentication scheme based off of the AAA architecture that concerned itself primarily with supporting mobile users that leave their home domain and enter foreign networks [Liang & Wang, 2004]. In the proposed scheme, whenever mobile users entered a new network the local (foreign) authentication server would first check for any pre-existing security associations for the mobile user and use them when available. Failing this, the local authentication server would estimate the total time that the mobile user would be likely to remain in the network, and if it was greater than some pre-define threshold, it would contact the mobile user's home authentication server, authenticate the mobile user, and then set up a security association between the mobile user and itself to be used in the future. In this way, once a mobile user had entered the network once and been remotely authenticated in order to set up a security association, it could then be locally authenticated in the future should it leave the network and later return. Liang & Wang's scheme was deemed to be essentially a form of caching of mobiles users and their resulting security associations and the concept was not found to be particularly interesting to work with.

3.3 AUGMENTED PRIVACY AND KEY MANAGEMENT PROTOCOL

Hamid & Khan proposed a series of modifications to the Privacy and Key Management v2 protocol of the IEEE 802.16e standard designed to enhance the security and robustness of the system [Hamid & Khan, 2006]. These modifications were designed primarily around safeguarding the authentication protocol from a number of attacks, among them sponsor node impersonation, replay attacks and man-in-the-middle attacks, but also managed to achieve local authentication while doing it. In order to accomplish this, the proposed protocol added timestamp information to every message within the PKMv2 protocol, as well as requiring sponsor nodes to digitally sign one of their messages in order to authenticate themselves to candidate nodes. With these modifications in place, a candidate node could connect to the network by contacting any sponsor node and authenticating itself (and see the sponsor

node do the same) locally with the sponsor node to get the shared secret key currently in use by the network. After authentication the candidate node could then begin contacting additional neighbours and use the shared secret key to build up its connectivity. While Hamid & Khan's modifications were primarily motivated to improve the security of the PKMv2 protocol they did managed to incorporate local authentication into the equation as well. Given that these changes were already well documented and explained and did not offer substantial room for improvement at all, the Augmented PKMv2 protocol was passed over to allow for more variety on the subject.

3.4 THRESHOLD AUTHENTICATION

Secret sharing techniques have long been both an interesting and innovative area of research and their application can often be extended to other areas of computer science, as is the case with the concept of threshold authentication outlined by Desmedt & Frankel. The primary motivation of their technique was to combine secret sharing schemes into the authentication process to provide unconditionally secure authentication schemes [Desmedt & Frankel, 1992]. In the proposed model, instead of mutually authenticating to some sort of centralized base station, or even a single node, candidate nodes wishing to join the network authenticate against a subset of designated authentication nodes (the total number of which must be equal to the system threshold) within the network. Each authentication node has the capabilities of partially signing a message in such a way as to be able to construct a fully signed message when enough partial results have been gathered and combined (the number of partial results needed depends on the system threshold). Naturally, this process ensures that after it has received partial results from enough sponsor nodes to meet the threshold, there is no way for the candidate node to be impersonated or to impersonate someone else in the future. Desmedt & Frankel proposed that this scheme could be used to generate RSA signatures as well as a number of other mechanisms to allow for authentication to occur. In the end, this scheme was selected for adaptation into the IEEE 802.16 standard, as it was both interesting, practical and possessed of a number of desirable and helpful characteristics.

4 THRESHOLD AUTHENTICATION SCHEME

As was briefly mentioned in Section 3, the threshold authentication scheme is possessed of a number of interesting properties that make it particularly attractive to be incorporated into the IEEE 802.16 standard. In particular, it provides a means of local authentication while also offering a significant increase to the security of the authentication process. The properties of threshold authentication completely eliminate the problem of malicious sponsor nodes, as there is no way for a single node to sign any messages on behalf of the network as each node has only a partial share of the key. The only situation where malicious sponsor nodes could manipulate a candidate node would be one in which there are a number of malicious nodes all working in tandem so that they could meet the system threshold and be able to produce a fully signed message between them. This also assumes that the malicious nodes themselves have a partial share of the private key, which is by no means guaranteed. Furthermore, this technique is inherently distributed and makes absolutely no assumptions as to the layout of the network and could thus be deployed on essentially any topology. Finally, threshold schemes also enhance security against intrusion in the network: consider the case where a malicious user compromises the security of one of the authentication nodes. What has the attacker gained? A partial share of the private key. In order to get a real gain, an attacker would have to compromise as many nodes as the system threshold value. This may not seem like a huge improvement, but consider that in a non threshold scheme, an attacker would most certainly have recovered the entire private key with only a single node takeover.

Incorporating the threshold scheme into authentication turned out to be a relatively simple process. Desmedt & Frankel demonstrated that it is possible to generate an RSA public and private key pair using a threshold scheme in a way that the private key is never stored as anything aside from a number of distributed partial shares across the network [Desmedt & Frankel, 1992]. Although their original technique was rather heavy on math and quite complex, it did not take long for more efficient techniques for threshold authentication to be discovered. Currently, Shoup's protocol [Shoup, 2000] is the most efficient means of working with a threshold RSA scheme, though there has been much work put into furthering and expanding upon Shoup's original protocol, such as the work of Gennaro et al. For the purposes of this report, Shoup's algorithm will be adapted to function within the IEEE 802.16 standard in an effort to reduce complexity. With Shoup's algorithm functioning in a WiMAX

environment, the modifications of Gennaro et al. can be made relatively simply should the additional properties be desired.

The following subsections will provide an in-depth explanation of Shoup's Threshold RSA protocol, as well as an overview of the additions that Gennaro et al. made to Shoup's protocol in an effort to make is more robust in a dynamic wireless environment.

4.1 SHOUP'S THRESHOLD RSA PROTOCOL

There are two primary phases of Shoup's algorithm, namely the Key Sharing Phase and the Signature Computation Phase [Shoup, 2000]. The key sharing phase concerns itself with constructing the public RSA key as well as the partial private RSA key shares for the network, whereas the signature computation phase describes how the network uses its partial shares to compute a signed RSA signature for a given message. The key sharing phase requires a single trusted node within the network to generate the partial key shares; Shoup labels this node as the "dealer" in his protocol. To begin the key sharing phase, the dealer chooses two random and large primes of equal length, p and q, where $p = 2p' + 1$, $q = 2q' + 1$, where p' and q' are also prime. The RSA modulus is then computed to be $N = pq$ and Shoup computes the value $m = p'q'$. The dealer must then chose an RSA public exponent, e, such that $e > l$, where l is the total number of nodes within the network. The public key for the network is now the value $PK = (N, e)$. The dealer then computes the secret key d as $de = 1 \bmod m$ (which only the dealer can compute, since it has the factorization of n). The dealer then chooses t random values $a_1, a_2, \ldots, a_t$ from $\mathbb{Z}_m$ (where t is the system threshold value) and defines the function $f(z) = a_t z^t + a_{t-1} z^{t-1} + \cdots + a_1 z + d$. Shoup's protocol requires that the maximum number of partial shareholders, n, be fixed and public. Shoup also sets the value $\Delta = n!$. That said, the partial shareholder i is given the share $d_i = f(i) \bmod m$.

The signature computation phase occurs when a message M needs to be signed by the network. Each partial shareholder i computes the value $y = \mathrm{H}(M) \in \mathbb{Z}_N^*$ and then the signature fragment $\sigma_i = y^{2\Delta \cdot d_i} \bmod N$ and publishes the signature fragment. As soon as t+1 of these signature fragments are gathered, the value $\sigma' = \prod_{j=1}^{t+1} \sigma_{i_j}^{2\Delta \cdot L_s(0, i_j)} \bmod N$ can be computed (where $L_s(0, i_j)$ is the appropriate Langrangian coefficients). This

value can be reduced to $\sigma' = y^{4\Delta^2 \cdot d} \mathrm{mod} N$ through the use of algebra. The standard Euclidean algorithm can then be used to show that $\sigma = y^{1/e}$, given σ', which allows the full signature ($y^d \bmod N$) to be computed.

Despite the efficiency of Shoup's algorithm, there are a number of limitations, namely the fact that the maximum number of partial shareholders is fixed and must be known in advance. Given the highly dynamic nature of wireless networks this restriction is quite possible unreasonable. Fortunately, there has been a great deal of work done on working with Shoup's original algorithm and modifying it to make it less restrictive, such as that done by Gennaro et al.

4.2 GENNARO ET AL.'S ADDITIONS

The primary motivation of Gennaro et al.'s work was to modify Shoup's threshold RSA protocol to be less restrictive and more adaptable to dynamic ad-hoc networks [Gennaro, Halevi, Krawczyk, & Rabin, 2008]. Among other modifications, Gennaro et al. were able to modify Shoup's protocol to function correctly without requiring a maximum number of partial shareholders in advance. Furthermore, they devised a way to allow new shareholders to be given partial shares *after* the key distribution phase had already occurred in a manner similar to generating message signatures (*t* nodes must collaborate to generate the new partial share). This allows the network to accept new nodes as partial shareholders without having to generate an entirely new RSA keys. Although these properties greatly improve the capabilities of the threshold RSA scheme, the mathematics involved quickly becomes even more complex than it already was. For the sake of keeping complexity to bearable levels, the specifics of incorporating Gennaro et al.'s modifications are left purposefully vague. The important consideration is that these modifications are possible in the first place.

4.3 LIMITATIONS

Despite the numerous benefits of incorporating a threshold RSA scheme into the WiMAX authentication process there remain a number of potential problems and areas for concern. First of all, a threshold scheme's functionality falls apart in cases where there are insufficient nodes within the network capable of being authorization nodes, such as when a network is newly formed. In cases such as these, a threshold scheme is unable to meet the system threshold and thus no signing can occur at all. While it remains possible for a new RSA key-pair to be generated and the system threshold value lowered so as to be functional, it should be noted that such operations

would be fairly intensive and a new public key would need be released and used every time – the old key would need to be revoked. As it so happens, certificate revocation is another area where threshold schemes suffer from a number of potential problems. The primary concern stems from the fact that there is no way for an authorization node to have its signing rights revoked without the entire network revoking the current key and generating a new RSA key while ensuring the desired node is not included in the key distribution phase. This is quite obviously inefficient, as it requires the entire network revoking its current RSA certificate and generating a new one merely to deal with a single authorization node removal.

5 PROPOSED AUTHENTICATION SCHEME

5.1 OUTLINE OF CHANGES

The following authentication scheme is the culmination of all of the research presented thus far. Its primary purpose is to incorporate local authentication mechanisms into the IEEE 802.16 standard for use with a WiMAX network operating in Mesh mode. Secondary goals include enhancing the security of the authentication process above and beyond what the IEEE 802.16 standard's PKM protocol provides. In order to realize these goals, the PKM protocol was modified in the following ways:

- A Threshold RSA scheme has been incorporated into the IEEE 802.16 standard. WiMAX networks will use threshold RSA to generate an RSA key-pair to digitally sign messages in order to authenticate the network to candidate nodes.
- Nonce values have been added to all authentication messages as a form of transient information in order to prevent message replay attacks.
- HMAC has been incorporated into the Authentication Request message in order to provide message integrity and prevent man-in-the-middle attacks.
- Sponsor nodes create a message digest of the Authentication Reply message and have the network digitally sign it using threshold RSA and send both the Authentication Reply and Message Digest to the candidate node. This results in both message integrity as well as mutual authentication.

Section 5.2 provides an outline of all relevant messages used throughout the proposed authentication protocol, while Section 5.3 outlines an example of the new protocol in action. Section 5.2 analyzes the newly proposed authentication scheme in comparison to the IEEE 802.16 standard's default PKM protocol.

5.2 AUTHENTICATION MESSAGES

5.2.1 AUTHENTICATION INFORMATION MESSAGE

The Auth-Info message (see Table 5 below) contains the SS manufacturer's X.509 certificate, issued by the manufacturer itself or by an external authority.

Table 5 - Authentication Information Message Attributes

Attribute	Contents
CA-Certificate	Certificate of manufacturer CA that issued SS certificate.

5.2.2 AUTHENTICATION REQUEST MESSAGE

The Auth-Req message (see Table 6 below) contains:

- A randomly generate nonce value.
- A manufacturer-issued X.509 certificate.
- A description of the cryptographic algorithms the requesting SS supports.
- The SS's Basic CID.
- An HMAC of the entire request message to ensure message integrity.

Table 6 - Authentication Request Attributes

Attribute	Contents
SS-Nonce	A randomly generate nonce value.
SS-Certificate	Contains the SS's X.509 user certificate
Security-Capabilities	Describes requesting SS's security capabilities.
SAID	SS's primary SAID equal to the Basic CID.
HMAC	An HMAC for the entire message to ensure message integrity.

5.2.3 AUTHENTICATION REPLY MESSAGE

The Auth-Reply-Wrapper message wrapper (see Table 7 below) contains:

- The authentication reply message, encrypted with the target SS's public key.
- The signed message digest of the authentication reply, encrypted with the target SS's public key.

Table 7 - Authentication Reply Wrapper Attributes

Attribute	Contents
Auth-Reply	The authentication reply message, encrypted with the target SS's public key.
Auth-Reply-Digest	The signed message digest of the authentication reply.

The Auth-Reply message (see Table 8 below) contains:

- The SS-Nonce value from the Authorization Request message.
- An AK (Authorization Key).
- A 4-bit key sequence number, used to distinguish between successive generations of AKs.

- A key lifetime.
- The identities (i.e. SAIDs) and properties of the single primary and zero or more static SA's the SS is authorized to obtain keying information from.
- An operator shared secret key known to all nodes in the network.
- A 4-bit key sequence number, used to distinguish between successive generations of operator shared secret keys.
- A key lifetime for the operator shared secret key.

Table 8 - Authentication Reply Attributes

Attribute	Contents
SS-Nonce	The nonce value from the Auth Request message.
AUTH-Key	Authorization (AUTH) Key.
Key-Lifetime	AK's active lifetime.
Key-Sequence Number	AK sequence number.
(one or more) SA Descriptor(s).	Each compound SA-Descriptor attribute specifies an SAID and additional properties of the SA.
PKM Configuration Settings (optional)	PKM time values.
Operator Shared Secret	Mesh Mode Only: Key known to all.
Key-Sequence Number	Mesh Mode Only: Sequence number of the Operator Shared Secret.
Key-Lifetime	Mesh Mode Only: Lifetime of the Operator Shared Secret.

The Auth-Reply-Digest message (see Table 9 below) contains:

- The public threshold RSA key for the network as a whole.
- A hash of the Authentication Reply message, encrypted with the network's private RSA key by using the threshold RSA protocol.

Table 9 - Authentication Reply Digest

Attribute	Contents
Network-Public-Key	The public threshold RSA key for the network as a whole.
Message-Digest	The hashed value of the Authentication Reply message, encrypted with the network's private RSA key by using the threshold RSA protocol.

5.2.4 AUTHENTICATION REJECT MESSAGE

The Auth-Reject message (see Table 10 below) contains:

- An error code detailing the reason for the rejection of the authorization request.

- Optionally, a string detailing the error in plain English.

Table 10 - Authentication Reject Message

Attribute	Contents
Error-Code	Error code identifying reason for rejection of authorization request.
Display-String (Optional)	Display String providing reason for rejection of authorization request.

5.2.5 SIGNATURE REQUEST MESSAGE

The Sig-Req message (see Table 11 below) contains:

- The hashed value of the Authentication Reply message that needs to be signed.

Table 11 - Signature Request Message

Attribute	Contents
Message-Digest	The hashed value of the Authentication Reply message that needs to be signed.

5.2.6 SIGNATURE REPLY MESSAGE

The Sig-Reply message contains:

- A partial signature of the Authentication Reply message digest.

Table 12 - Signature Reply Message

Attribute	Contents
Partial-Sig	A partial signature of the Authentication Reply message digest.

5.3 EXAMPLE

In order to provide a meaningful comparison to the IEEE 802.16 standard's original PKM protocol, the example from Section 2.3 will be repeated here, using the newly proposed modifications rather than the default protocol. Suppose a subscriber station, *A*, wishes to join an established network. *A* will henceforth be known as the Candidate Node (CN). *A* does not care whether or not it is in range of the base station *B* since it will be authenticated locally. *A* will simply select any node that *is* in range, in this case node *C*, and communicate with that node. At this point *C* has become *A*'s Sponsor Node (SN). The first message *A* will send is the Auth-Info message, which contains the X.509 certificate of *A*'s manufacturer which also happens to be the Certificate Authority (CA) for *A*'s own X.509 certificate. Before waiting for a reply, *A* will compute a random nonce value, construct an Auth-Req message,

calculate an HMAC for the message and then immediately send the Auth-Req message to *C*. Once more, this is because the Auth-Info message is completely informative and sponsor nodes are permitted to completely ignore it if desired; since there will never be a response to the Auth-Info message, *A* is free to send the Auth-Req message immediately. Once *C* has received the Auth-Info and Auth-Req messages *C* will use the HMAC of the Auth-Req message to ensure the message has not been modified in transit. Should the Auth-Req message show signs of tampering *C* will reply with an Auth-Reject message. If the Auth-Req message is genuine *C* will instead accept or reject the authentication request. Should *A*'s authentication be accepted *C* will construct an Auth-Reply message, using *A*'s public key from its X.509 certificate to encrypt the data before computing a message digest. *C* will then use its own knowledge of the network to select *t* other trusted nodes in the network and will proceed to send a Sig-Req message to each of them. While *C* awaits the Sig-Reply messages, it will compute its own partial signature of the message digest. Once all *t* nodes have replied with a Sig-Reply message, *C* will gather all of the partial signatures and combine them into a fully signed message digest for the Auth-Reply message. *C* will then construct the Auth-Reply-Wrapper message and send it to *A*. Alternatively, *C* will return an Auth-Reject message with the appropriate error code if *A*'s authentication request should be rejected.

Once *A* receives the Auth-Reply-Wrapper from *C*, it will decrypt the Auth-Reply message and Auth-Reply-Digest using its own private key and the network's public key respectively. *A* will then verify the nonce value found within the Auth-Reply message to ensure it is the correct value and thus guarantee that the message was not part of a message replay attack. Following this, *A* will compute a message digest for the Auth-Reply message and compare it to the message digest signed and sent by *C*. Should the two digests be equal, *A* knows that the message has not been tampered with and that the entire message is genuine and not forged. At this point *A* knows that it has successfully completed authentication with the network, or alternatively was rejected if *A* received an Auth-Reject message. In either case, the newly proposed modifications to the PKM protocol will have completed.

5.4 ANALYSIS

Although the specific details of threshold RSA are quite complex, the high level view of the proposed authentication scheme is much simpler and remains just as easily understood as the original PKM protocol. With the incorporation of local authentication the problem of a long string of superfluous messages back and forth between candidate nodes and base stations has been eliminated. Unfortunately, the threshold RSA scheme introduced a great

deal more messages in order to request and collect partial signatures. The number of new messages introduced by threshold RSA is, however, controllable through modification of the system threshold variable *t* should fine tuning be needed. Furthermore it should be noted that although the change in number of messages may not have changed, the proposed authentication scheme can potentially operate much more quickly than the original PKM protocol. Rather than constantly forwarding messages back and forth from node to node in sequence, the proposed protocol distributes the message digest to be signed and the nodes in the network calculate partial signatures in parallel. Under the assumptions that processing time is negligible in comparison to communication time between nodes and that it requires 1 time unit to send a message along a channel, the proposed authentication scheme will always require only 5 time units to complete the authentication process (1 for Auth-Info, 1 for Auth-Req, 1 for Sig-Req, 1 for Sig-Reply and 1 for Auth-Reply-Wrapper) assuming there are *t* trusted nodes within one hop of the sponsor node. The original PKM protocol, on the other hand, will require at minimum 3 time units (1 for Auth-Info, 1 for Auth-Req and 1 for Auth-Reply), but for every intermediary node between the candidate node and the base station the PKM protocol will require an additional 3 time units in order to forward the messages to the base station.

Additionally, the proposed authentication scheme is more secure than the IEEE 802.16 standard's PKM protocol. The proposed scheme is not susceptible to sponsor node impersonation as a result of utilizing mutual authentication along with threshold RSA; the only exception being if $t+1$ malicious nodes within the network collaborate together and are in possession of valid partial shares, which is highly unlikely. The addition of the HMAC to the Auth-Req message and the signed message digest to the Auth-Reply message also ensures that message modification attacks cannot occur, which furthermore prevents man-in-the-middle attacks from being a threat. Finally, the network's private RSA key is also heavily secured. An attacker would need to infiltrate and steal the partial shares of $t+1$ nodes before being able to recover the private RSA key, as opposed to the single infiltration most commonly required.

6 RESULTS

The benefits of incorporating local authentication mechanisms into the IEEE 802.16 standard should, at this point, be clear and tangible. Even without the performance gains, WiMAX's default PKM protocol is simply too unsecure to be utilized in any environment where security and robustness is even a small concern. The proposed authentication protocol manages to not only integrate local authentication into WiMAX with similar if not better performance than the default PKM, but it also secures the authentication process against a number of serious and practical attacks against the network. Despite its complexity, Threshold RSA is a viable implementation of local authentication that can be integrated into the IEEE 802.16 standard, offering a number of unique security guarantees. Additionally, threshold RSA can be modified as proposed by Gennaro et al. (2008) in order to be made highly tolerant and support an incredibly dynamic network setup.

While the results of this project show promise, the complexity of threshold RSA remains an undisputable concern. Hamid & Khan's augmented PKMv2 protocol accomplishes many of the same goals as the proposed authentication scheme while at the same time remaining incredibly simple and straightforward. A threshold RSA scheme may provide a few stronger security guarantees, but it is unclear if the complexity and cost of implementing threshold RSA is worth the slight enhancement to security. Regardless of its practicality of implementation, threshold RSA remains a very interesting and intriguing concept and area of research and in environments that lend themselves naturally to the threshold model or demand more stringent security guarantees it could prove to be incredibly effective.

7 REFERENCES

Dellutri, F., Me, G., & Strangio, M. A. (2005). Local Authentication with Bluetooth enabled Mobile Devices. *Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services, 2005. ICAS-ICNS 2005.* (pp. 72-77). Papeete, Tahiti: IEEE Computer Society.

Desmedt, Y., & Frankel, Y. (1992). Shared Generation of Authenticators and Signatures (Extended Abstract). In *Advances in Cryptology - CRYPTO '91* (pp. 457-469). Berlin / Heidelberg: Springer-Verlag.

Gennaro, R., Halevi, S., Krawczyk, H., & Rabin, T. (2008). Threshold RSA for Dynamic and Ad-Hoc Groups. *Advances in Cryptogrophy - EUROCRYPT '08.* Istanbul: Springer.

Hamid, Z., & Khan, S. A. (2006). An Augmented Security Protocol for WirelessMAN Mesh Networks. *International Symposium on Communications and Information Technologies, 2006. ISCIT '06.* (pp. 861-865). Bangkok: IEEE Computer Society.

IEEE 802.16 Working Group on Broadband Wireless Access. (2004, October 1). IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems. New York, New York, United States of America.

Liang, W., & Wang, W. (2004). A Local Authentication Control Scheme Based on AAA Architecture in Wireless Networks. *Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th* (pp. 5276-5280). IEEE Computer Society.

Shoup, V. (2000). Practical Threshold Signatures. *Lecture Notes in Computer Science , 1807*, 207-220.