Comparative Analysis of Different Cryptosystems for Hierarchical Mobile IPv6-based Wireless Mesh Network

Ramanarayana Kandikattu and Lillykutty Jacob
(Corresponding author: Ramanarayana Kandikattu)
Department of Electronics and Communication Engineering, National Institute of Technology Calicut
NIT Campus P.O., Calicut-673601, India
(Email: k_ramnarayan@rediffmail.com, lilly@nitc.ac.in)
(Received June 5, 2008; revised and accepted Oct. 2, 2008)

Abstract

Wireless Mesh Network (WMN) is advocated as the major supporting technology for the next generation wireless Internet satisfying the needs of anywhere-anytime broadband Internet access. In order to support secure ubiquitous communications for mobile users, WMN must have an efficient key setup procedure to secure control packets as well as data packets. In this paper we apply four different cryptosystems, namely: (1) RSA; (2) Elliptic Curve Digital Signature Algorithm (ECDSA) (3) Identity-Based Cryptography (IBC); and, (4) Elliptic Curve Cryptography-Based Public Key Cryptosystem (ECCSCPKC), to secure Hierarchical Mobile IPv6 (HMIPv6) based WMN. We present detailed cost analysis and numerical results to compare these systems for their suitability to secure HMIPv6 based WMN.

Keywords: HMIPv6, mobility management, security, wireless mesh network

1 Introduction

The proliferation of wireless devices such as laptops, PDAs, bluetooth devices etc., is increasing enormously all over the world. As a consequence of that, the demand for ubiquitous and broadband wireless access is increasing rapidly. Many research efforts are going on to meet these ever increasing demands. But there are many open issues yet to be resolved and the real deployment of broadband wireless Internet is still in infancy. Wireless Mesh Network (WMN) [2] appears to be the most viable technology supporting broadband connectivity for mobile clients. It has emerged to supplement the existing wired network providing cheap wireless network coverage and access. WMNs have wide range of applications ranging from civilian wireless Internet applications to tactical and emergency response applications. The ease of deployment, flexibility, self configuration, multihop connectivity, etc., are some the attractive features of WMN.

WMN comprises Access (or mesh) Routers (ARs) and gateway routers. They form backbone network with the help of their point-to-point radio links. In order to provide ubiquitous wireless network access to the Mesh Clients (MCs) lying in a large geographic area, WMN requires a large number of ARs, with each one covering a portion of the area and forming a subnet with MCs lying in that area. WMN uses multihop routing protocol for self configuration of routes among the MCs. This scenario gives a hierarchical structure where a MC can access the Internet client through its associated AR and through the gateway router. To provide continual Internet access to the MCs that move across the subnets, WMN needs an efficient mobility management mechanism [8, 30] that consumes minimal bandwidth, and computational resources and minimizes delay for handover process. Though there many proposals available in the literature to manage mobility, Mobile IPv4(MIPv4) [22], Mobile IPv6 (MIPv6) [16] and Hierarchical Mobile IPv6 (HMIPv6) [25] are the most widely accepted protocols by the research community. In this paper HMIPv6 is considered as the candidate protocol because of its hierarchical structure that minimizes communication overhead and handover latency.

Security is a major concern for any network. WMN is prone to various active and passive attacks during the handover and data transfer phases [24]. Therefore, WMN needs to have a security architecture for key setup among the authenticated MCs and should have mechanisms to protect control and data packets. MIPv6 and its enhancements (including HMIPv6) use Internet Key Exchange (IKE) [17] protocol for key distribution among the participating nodes and IPSec [10] for protecting signaling and data packets. The IKE protocol requires four to six packets with two to three turn-around times to create a Security Association (SA) between the pair of participating network entities. The negotiated key is then given to

the IPsec stack which is used to form secure IPsec tunnel between them. Moreover, IPSec protocol stack is defined for different cryptosystems and supports various cryptographic algorithms. The selection of proper cryptosystem is vital, because of the processing delay and computational costs associated with the systems.

Shared key cryptosystems are not suitable for ubiquitous applications, because they require every communicating node to have a pair-wise shared key with every other communicating node in the network, which is difficult to meet. Certificate-based cryptosystems (CBC) such as RSA, ECC are also not suitable for ubiquitous applications because, they require every node to piggyback a long certificate (typically 1 Kbyte) and to affix its signature with every signaling packet that it originates. This increases communication overhead.

IKE-based key setup is difficult to achieve in ad hoc wireless environment with dynamic connections. Moreover, IPv6 and its extensions do not address the use of public key infrastructure in a very large network with dynamic communication channels.

Therefore, the investigation of key setup and applicability of different cryptosystems to HMIPv6-based WMN and their impact on the performance of the network is significant.

In this paper we consider a generic Secure Wireless Mesh Network (SWMN) architecture that adopts the hierarchical structure proposed in HMIPv6. Figure 1 depicts a single domain of SWMN. Various entities in SWMN are defined as follows:

- Operator (O): An entity that operates the wireless mesh network. The wireless mesh network may contain single domain or multiple domains of different scales, either physically adjacent or non-adjacent.
- Mobility Anchor Point (MAP): An entity that controls and manages a regional domain or simply a domain. Every domain contains many subnets, covering the geographic area of interest. MAP is the gateway router to/from the domain.
- Access Router (AR): An entity that manages a single subnet. An operator which has multiple regional domains has multiple MAPs, one per domain and multiple ARs one per subnet.
- Mesh Client (MC): An entity that wants to communicate with another MC or wants to access the Internet.

The concepts of on-Link Care-of Address (LCoA), Regional Care-of Address (RCoA) and Home Address (HoA) are same as that of HMIPv6 [25].

We apply four different public key cryptosystems namely: (1) RSA [19]; (2) Elliptic Curve Digital Signature Algorithm (ECDSA) [14]; (3) Identity-Based Cryptography (IBC) [5, 6, 11, 23]; and (4) Elliptic Curve Cryptography-based Self Certified Public Key Cryptosystem (ECCSCPKC) [26] to protect SWMN. The corresponding systems are denoted respectively as: $SWMN_S$, $SWMN_D$, $SWMN_I$ and $SWMN_E$.

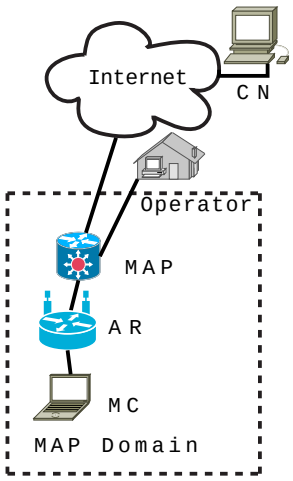

Figure 1: A MAP domain in SWMN

The rest of the paper is organized as follows. Section 2 gives the preliminaries of IBC and its application to SWMN. Section 3 presents ECCSCPKC and its operations in SWMN context. Section 4 presents an outline of related work. Section 5 presents a generic secure location update process that adopts any of the four systems, namely RSA, ECDSA, IBC and ECCSCPKC. Section 6 gives an analytical model for the cost estimation. Section 7 presents detailed numerical analysis and comparison of SWMN systems with HMIPv6. Finally, Section 8 gives the conclusions and future work.

2 Identity-based Cryptography

Conventional certificate-based cryptography requires a lengthy (typically 1K Byte) certificate to distribute the public key among the participating nodes. In the ad hoc network scenario, the certificate is piggy-backed on the control packets to distribute public key. This method incurs heavy communication overhead, consumes network bandwidth and computational resources.

Identity-Based Cryptography (IBC) eliminates the need for certificates because it enables to extract the public key of an authorized participating node from the identity of that node. Moreover, IBC allows any pair of authenticated clients to generate pair-wise shared key if their identities are known to each other. Shamir introduced the concept of IBC [23] in 1984. Later, Boneh et al. proposed a basic identity-based signature scheme [6] and presented identity- based encryption scheme using pairing technique [5]. A good survey on pairing-based cryptographic protocols is provided by [11]. The following gives an overview of the basics of pairing technique.

2.1 Bilinear Pairing

Let G_1 be an additive group and G_2 be a multiplicative group of the same prime order q. Let P be an arbitrary generator of G_1. Assume that the discrete logarithm problem is hard in both G_1 and G_2. A mapping $F : G_1 \times G_1 \rightarrow G_2$ satisfying the following properties is called a cryptographic bilinear map as defined by Boneh et al. [6].

- Bilinearity: $F(\alpha P,\beta\ Q) = F(P,Q)^{\alpha}\ \beta = F(\alpha P,Q)^{\beta} = F(P,\beta Q)^{\alpha}$ for all $P, Q \in G_1$ and $\alpha, \beta \in Z_q^*$, where $Z_q^* = \{1, 2, \ldots q-1\}$.
- Non-degeneracy: If P is a generator of G_1, then $F(P,P)$ is the generator of G_2; in other words $F(P,P) \neq 1$.
- Computability: There exists an efficient algorithm to compute $F(P,Q)$ for all $P, Q \in G_1$.

Modified Weil and Tate pairings on an elliptic curve over a finite field are examples of cryptographic bilinear maps.

2.2 Domain Parameter Setup

IBC requires a trusted third party called *Public Key Generator* (PKG) to generate the public-private key pair corresponding to each node's identity using pairing based mechanisms. In $SWMN_I$, operator does the role of trusted third party. It performs the following domain-parameter initialization:

- Generates the pairing parameters (q, G_1,G_2,F,P, H_1).
- Picks a random $s \in Z_q^*$ as domain secret and computes domain-public key as $P_{pub} = s.P$.

We define the *domain-parameters* as:(q, G_1, G_2, F, P, H_1, P_{pub}) and the *domain certificate* as: (domain-parameters, $s.H_1$(domain-parameters). The operator must keep 's' confidential, while making domain-certificate publicly known. All the entities under an operator use the same domain parameters. The legitimacy of the domain parameters can be checked by validating the domain certificate as follows: $F(P,\ s.H_1(\text{domain-parameters}))=F(s.P,\ H_1(\text{domain-parameters})) = F(P_{pub}, H_1(\text{domain-parameters}))$.

2.3 Public-private Key Pair Extraction

$SWMN_I$ requires every participating entity e.g., MAP, AR, MC to obtain its identity, and public-private key pair from the operator before entering into the network. Table 1 gives the identity structure used for different entities of $SWMN_I$ and Table 2 gives the notations used. Note that, the freshness of identity is decided by the expiry time.

Operator generates public key from the ID of an entity by applying domain hash on it, and computes the corresponding private key by multiplying public key with domain secret s. The public-private key pair for MAP, AR, MC are generated as follows:

$$\begin{aligned} K_{MAP_{jk}} &= H_1^{O_k}(ID_{MAP_{jk}}) \\ K_{MAP_{jk}}^{-1} &= s^{O_k}.H_1^{O_k}(ID_{MAP_{jk}}) \\ K_{AR_{ijk}} &= H_1^{O_k}(ID_{AR_{ijk}}) \\ K_{AR_{ijk}}^{-1} &= s^{O_k}.H_1^{O_k}(ID_{AR_{ijk}}) \\ K_{MC_{ijk}} &= H_1^{O_k}(ID_{MC_{ijk}}) \\ K_{MC_{ijk}}^{-1} &= s^{O_k}H_1^{O_k}(ID_{MC_{ijk}}). \end{aligned}$$

Note that, the superscript O_k is used to indicate that the domain hash H, and domain secret s are operator specific.

2.4 Pair-wise Shared Key Setup

Once registered entities (e.g., $MC_{1,1,1}$ and $MC_{2,1,1}$) in an administrative domain are equipped with their ID, domain parameters, and public-private key pair, then they can establish pair-wise shared key with each other using bilinearity as given in Equation (1).

$$\begin{aligned} & K_{MC_{1,1,1},MC_{2,1,1}} \\ &= F^{O_1}(K_{MC_{1,1,1}}^{-1}, H_1^{O_1}(ID_{MC_{2,1,1}})) \\ &= F^{O_1}(s^{O_1}.H_1^{O_1}(ID_{MC_{1,1,1}}), H_1^{O_1}(ID_{MC_{2,1,1}})) \\ &= F^{O_1}(H_1^{O_1}(ID_{MC_{1,1,1}}), s^{O_1}.H_1^{O_1}(ID_{MC_{2,1,1}})) \\ &= F^{O_1}(H_1^{O_1}(ID_{MC_{1,1,1}}), K_{MC_{2,1,1}}^{-1}) \\ &= K_{MC_{2,1,1},MC_{1,1,1}} \end{aligned} \quad (1)$$

3 Elliptic Curve Cryptography Based Self Certified Public Key Cryptosystem (ECCSCPKC)

Though IBC solves the problem of key setup, it has the following drawbacks: (1) PKG knows the private-public key pair of every participating entity. Therefore if PKG is compromised the entire security of WMN is under threat. (2) IBC basic operations such as pairing computation, signature generation, signature verification are quite expensive than the counterpart operations in other cryptosystems such as RSA or ECC. To eliminate these drawbacks Tsaur proposed a hybrid system of ECC and IBC called ECCSCPKC in [26]. It has the advantages of light-weight computations of ECC and simple public key distribution without certificates as in IBC. ECCSCPKC eliminates the threat of private key leakage, because in this system PKG is not aware of the private key of the entity for which it generated the keys. In this subsection we present the important functions of ECCSCPKC in the SWMN context, in similar lines as given in [26].

3.1 Domain Parameter Setup

ECCSCPKC requires a trusted third party called *Public Key Generator* (PKG) for user registration. PKG and

Table 1: Identity structure of different entities in $SWMN_I$ and $SWMN_E$

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
$ID_{MAP_{jk}}$	MAP's network prefix								reserved								Expiry time			
$ID_{AR_{ijk}}$	AR's network prefix								reserved								Expiry time			
$ID_{MC_{ijk}}$	MC's Home address																Expiry time			

user use Zero Knowledge Proof (ZKP) to generate public-private key pair corresponding to each node's identity. ZKP allows the user to generate private (secret key) on its own, therefore PKG does not know the private key of user. In the SWMN context, operator does the role of trusted third party. ECCSCPKC requires each operator to perform the following domain-parameter initialization:

- Pick the ECC parameters (E, B, p), where p is the field size, typically a large prime or a power of 2 of about 160 bits, E is the elliptic curve defined over field F_p, B is the base point of order n (a large prime of typically 160 bits) over $E(F_p)$.
- Pick a random $s^{O_k} \in [2, n-2]$ as operator's secret key and compute operator's public key as $P^{O_k} = s^{O_k}B$.
- Select a one way hash function h that maps an arbitrary bit string to a fixed length bit string r, where $r \in [2, n-2]$.

Here the superscript O_k indicates operator $k's$ parameters. We define *domain-parameters* as (E, B, p, n, P^{O_k}, h) The operator must keep 's^{O_k}' confidential, while making domain-parameters publicly known. All the entities under the operator use the same domain parameters.

3.2 User Registration and Key Setup

In $SWMN_E$, any entity that wants to register with operator has to obtain its identity and key pair from operator using the following sequence of operations. Again Table 1 gives the identity structure used in $SWMN_E$ and Table 2 gives the notations. Figure 2 illustrates the steps involved in user registration process. Let MC_i be an example entity.

- MC_i who wants to register in a region served by MAP_{jk} under the operator O_k sends user registration request to the operator O_k.
- Operator O_k generates identity $ID_{MC_{ijk}}$ and sends it to the requested MC_i, which now can be denoted as MC_{ijk}.
- MC_{ijk} selects randomly an integer $x_{MC_{ijk}} \in [2, n-2]$ as master key and computes $V_{MC_{ijk}} = h(x_{MC_{ijk}}, ID_{MC_{ijk}})B$ and sends $(ID_{MC_{ijk}}, V_{MC_{ijk}})$ to the operator.
- Operator selects a random integer $l^{O_k} \in [2, n-2]$ and computes the following:

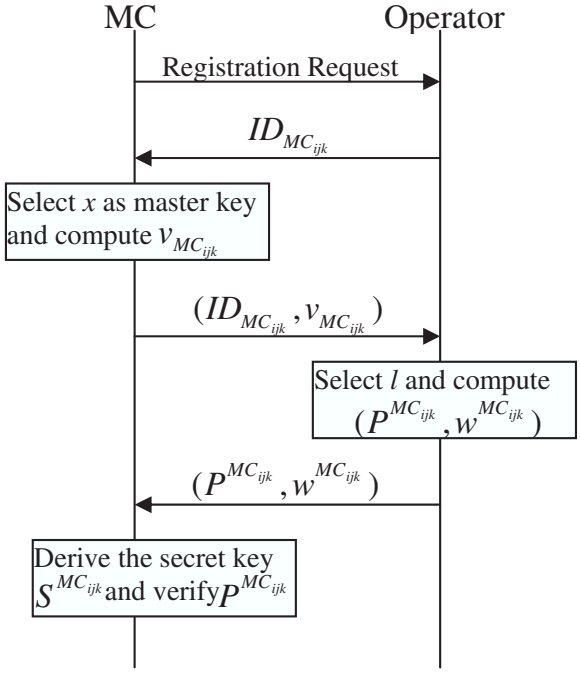

Figure 2: User registration process

  - A public key:

$$\begin{aligned} P^{MC_{ijk}} &= V_{MC_{ijk}} + (l^{O_k} - h(ID_{MC_{ijk}}))B \\ &= (P_x^{MC_{ijk}}, P_y^{MC_{ijk}}). \end{aligned}$$

  - A witness:

$$\begin{aligned} & w^{MC_{ijk}} \\ = \; & l^{O_k} + s^{O_j}(P_x^{MC_{ijk}} + h(ID_{MC_{ijk}})) \bmod n, \end{aligned}$$

    and responds MC_{ijk} with $(P^{MC_{ijk}}, w^{MC_{ijk}})$.

- MC_{ijk} then does the following operations:
  - derives the secret key $s^{MC_{ijk}}$ as:

$$s^{MC_{ijk}} = w^{MC_{ijk}} + h(x_{MC_{ijk}}, ID_{MC_{ijk}}) \bmod n.$$

  - verifies the authenticity of $P^{MC_{ijk}}$ by checking if $s^{MC_{ijk}}B$ is equal to $P^{MC_{ijk}} + h(ID_{MC_{ijk}})B + [(P_x^{MC_{ijk}} + h(ID_{MC_{ijk}})) \bmod n]P^{O_k}$.

MC_{ijk} accepts $(s^{MC_{ijk}}, P^{MC_{ijk}})$ as the private-public key pair and $ID_{MC_{ijk}}$ as its identity, if the

above verification is valid, otherwise discards them. Note that, the operator can not see the private key of MC_{ijk} in this process. Therefore the threat due to private key leakage is avoided.

3.3 Session Key Exchange

Let the two registered entities MC_{ijk} and $MC_{i'jk}$ want to exchange their session keys. Figure 3illustrates the procedure and is explained as follows:

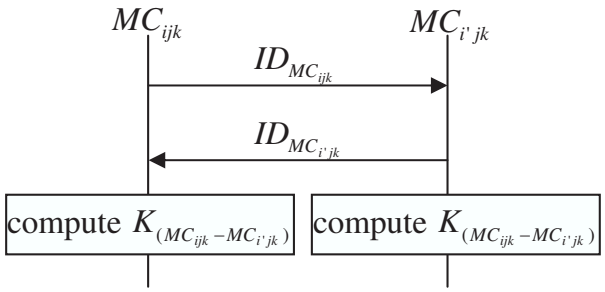

Figure 3: Session key exchange

- MC_{ijk} communicates its identity and public key information to $MC_{i'jk}$ and vice versa.
- MC_{ijk} computes the session key $K_{(MC_{ijk}-MC_{i'jk})}$ as follows:

$$\begin{aligned} V_{MC_{i'jk}} &= P_{MC_{i'jk}} + h(ID_{MC_{i'jk}}).B \\ &\quad +[(P_x^{MC_{i'jk}} + h(ID^{MC_{i'jk}})) \\ &\quad \bmod n].P^{O_k}. \\ K_{(MC_{ijk}-MC_{i'jk})} &= s^{MC_{ijk}}.V^{MC_{i'jk}} \\ &= (s^{MC_{ijk}} s^{MC_{i'jk}} \bmod n).B. \end{aligned}$$

- Similarly $MC_{i'jk}$ computes the session key $K_{(MC_{i'jk}-MC_{ijk})}$. Note that $K_{(MC_{ijk}-MC_{i'jk})} = K_{(MC_{i'jk}-MC_{ijk})}$.

3.4 The Digital Signature Scheme

Let MC_{ijk} be the signer and $MC_{i'jk}$ be the verifier and m be the message to be signed.

1) Signature generation
   - MC_{ijk} randomly chooses a time variant integer $k \in [2, n-2]$, and computes $k.B = (X_a, Y_a)$.
   - MC_{ijk} computes $r = X_a \bmod p$ and $s = k + s^{MC_{ijk}}.h(m,r) \bmod n$.
   - MC_{ijk} transmits the signature (r,s) and m to $MC_{i'jk}$.

2) Signature verification:
   $MC_{i'jk}$ computes

$$\begin{aligned} & V^{MC_{ijk}} \\ = \; & P^{MC_{ijk}} + h(ID^{MC_{ijk}}).B + [(P_x^{MC_{ijk}} \\ & +h(ID_{MC_{ijk}})) \bmod n].P^{O_k} \end{aligned}$$

and

$$\begin{aligned} & s.B - V^{MC_{ijk}}.h(m,r) \\ = \; & k.B + (s^{MC_{ijk}} h(m,r) \bmod n).B \\ & -(s^{MC_{ijk}}.B).h(m,r) \\ = \; & kB = (x_1, y_1). \end{aligned}$$

If $x_1 = r \bmod p$) holds, then the signature is valid.

The signature generation and verification process is illustrated in Figure 4.

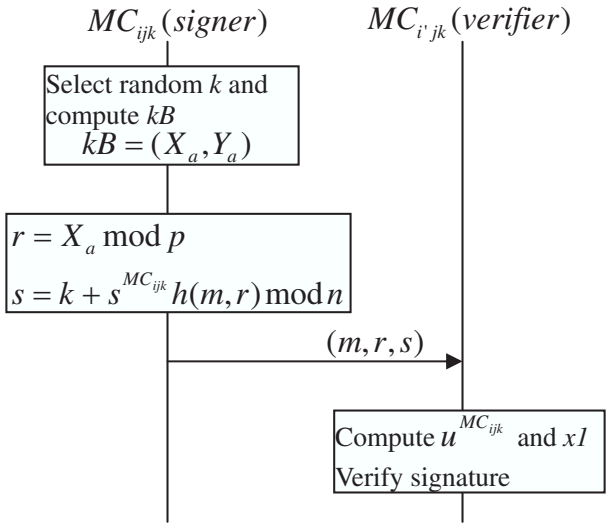

Figure 4: Signature generation and verification

4 Related Work: Security Issues in MIPv6 and Its Enhancements

HMIPv6sec [12] is a security extension to HMIPv6 protocol. This is based on a mechanism called cryptographically generated addresses (CGA) [4]. CGA is a technique whereby an interface part of IPv6 address of a node is cryptographically associated with node's public key and some other parameters. But CGAs themselves are not certified. Therefore, a malicious node can generate CGA using its public key. This protocol also allows nodes to use its self generated public-private key pair and does not require trusted third party. Even though the IP address and public key are cryptographically associated, if the public key is not certified by any trusted authority then the association between public key and node cannot be verified.

Table 2: Notations used in SWMN

Symbol	Meaning
O_k	Operator k
$MAP_{j,k}$	MAP j controlled by operator k
$AR_{i,j,k}$	AR i under MAP j controlled by operator k
$MC_{i,j,k}$	MC i under MAP j controlled by operator k
ID	Identity
m, n	bitwise concatenation of m and n
$ID_{MAP_{j,k}}$	identity of $MAP_{j,k}$
$ID_{AR_{i,j,k}}$	identity of $AR_{i,j,k}$
$ID_{MC_{i,j,k}}$	identity of $MC_{i,j,k}$
h	Domain hash function
s^{O_k}	Domain secret key of operator k
P^{O_k}	Domain public key of operator k
$s^{MAP_{j,k}}$	Private key of $MAP_{j,k}$
$P^{MAP_{j,k}}$	Public key of $MAP_{j,k}$
$s^{AR_{i,j,k}}$	Private key of $AR_{i,j,k}$
$P^{AR_{i,j,k}}$	Public key of $AR_{i,j,k}$
$s^{MC_{i,j,k}}$	Private key of $MC_{i,j,k}$
$P^{MC_{i,j,k}}$	Public key of $MC_{i,j,k}$
$A \rightarrow *$: m	Entity A broadcasts message m
$A \rightarrow B$: m	Entity A unicasts message m to entity B
$m, Sign_{s^A}()$	Concatenation of message m and signature of entity A over m
seq_X	Sequence number of node X
K_{X-Y}	Shared key between node X and node Y
MIC_{X-Y}	Hash(K_{X-Y}, message)

Malicious node can generate its own public-private key pair and can enter the network and then access the resources illegally.

5 Location Update

The binding update process in SWMN exactly follows HMIPv6 protocol except that the originator of a control packet appends security payload to it in order to protect the packet from replay, modification, and fabrication attacks. The security payload denoted by *sec_payload* for different cryptosystems is as follows:

- RSA: {sequence number, digital certificate, signature}
- ECDSA: {sequence number, digital certificate, signature}
- IBC: {sequence number, identity of source node, signature/MIC}
- ECCSCPKC: {sequence number, identity of source node, public key of source node, signature/MIC}

Originator of a control packet increments the sequence number by one for each new control packet generated, to protect the message from replay attacks. Originator appends its digital certificate/identity/(identity and public key) to enable the recipient node to extract public key from it. The recipient node uses the extracted originator's public key to verify the signature in the control packet. IBC/ECCSCPKC allows the recipient node to generate shared key between itself and sender node once the identity of the sender node is known. In $SWMN_I$ and $SWMN_E$ systems the first control packet exchange between a pair of nodes carry signature to protect the messages. After learning the identities each other, the pair of nodes generate the shared key and use Message Integrity check Code (MIC) for subsequent control packet exchanges. Since MIC is a hash value of shared key and message, it helps to verify message integrity and authenticate originator of the message as well, with relatively inexpensive hash operation. In all four versions of SWMN, each node validates the received message as follows:

1) Checks whether the message is fresh with the help of sequence number;
2) Ensures that certificate/identity is valid;
3) Verifies the signature/MIC; If all these checks are satisfied, then the receiver node authenticates the sender node and accepts the message; otherwise, receiver node discards the message.

Router advertisement: Access Router AR_{ijk} broadcasts (link local multicast) router advertisement through all its interfaces as in Equation 2. A mobile client, say $MC_{1,1,1}$, which is at one hop distance from AR_{ijk} can receive the advertisement. $MC_{1,1,1}$ identifies its present location it belongs to, i.e., home region or foreign region, with the help of router advertisement.

$$AR_{ijk} \rightarrow * : RouterAdv, sec\_payload \quad (2)$$

Then two cases exist:

1) if MC is in home region, it does not need to do RCoA update, but it has to register its LCoA with its associated MAP, i.e., HA.
2) If MC is in foreign region, then it has to do both, LCoA update with MAP and RCoA update with HA. The sequence of steps for LCoA and RCoA updates are explained next.

5.1 LCoA Update

When a MC is switched on first time, then it registers its LCoA with the associated MAP as follows:

$$FMC_{1,1,1} \rightarrow AR_{ijk} : BU(HoA, LCoA), sec\_payload \quad (3)$$

$$AR_{ijk} \rightarrow MAP : BU(HoA, LCoA), sec\_payload \quad (4)$$

$$MAP \rightarrow AR_{ijk} : BA(HoA, LCoA), sec\_payload \quad (5)$$

$$AR_{ijk} \rightarrow MC_{1,1,1} : BA(HoA, LCoA), sec\_payload \quad (6)$$

$MC_{1,1,1}$ configures LCoA and sends binding update (BU) as in Equation (3). Upon reception of the registration request, AR_{ijk} validates the message. If all the verifications are satisfactory then AR forwards this update as in Equation (4) to its associated MAP through secure channel between them, otherwise it drops the request. It is assumed that all the MAPs and ARs under an operator establish pair-wise shared keys among themselves and establish secure channel between each pair as soon as WMN is formed. AR/MAP uses keyed message integrity check code (MIC) to protect integrity of control packet. After validating, MAP creates a binding entry in its binding cache and records the association between MC's HoA and LCoA along with expiry time. Then MAP responds with Binding Acknowledgment (BA) to $MC_{1,1,1}$ via AR_{ijk} as in Equations (5) and (6).

Each MC registered under a MAP should update its location information as and when the MC moves to another link in the same MAP or before the lapse of expiry time, otherwise the entry will be deleted from the MAP's cache. MAP sends all the packets meant for a MC with the help of binding information. Note that, the AR does the BU signature verifications and responds with BA on behalf of MAP to reduce the computational overhead on MAP. Since AR and MAP have mutual trust relations, MAP trusts the verifications done by AR. This distributed mechanism reduces computational load on MAP.

5.2 RCOA Update

When a MC moves away from its home region, it registers LCoA with MAP similar to the process explained in the previous subsection. In addition to that, it registers RCoA with HA. The messages used for RCoA update are given in the following equations and the process is self explanatory.

$$MC_{1,1,1} \rightarrow AR_{ijk} : BU(HoA, RCoA), sec\_payload$$
$$AR_{ijk} \rightarrow MAP : BU(HoA, RCoA), sec\_payload$$
$$MAP \rightarrow HA : BU(HoA, RCoA), sec\_payload$$
$$HA \rightarrow MAP : BA(HoA, RCoA), sec\_payload$$
$$MAP \rightarrow AR_{ijk} : BA(HoA, RCoA), sec\_payload$$
$$AR_{ijk} \rightarrow MC_{1,1,1} : BA(HoA, RCoA), sec\_payload.$$

LCoA and RCoA update processes not only update the location information but also ensure pair-wise mutual authentication among MC_{ijk}, foreign MAP and HA. It is assumed that every MAP pre-computes the pair-wise shared key with every neighboring MAP using IBC/ECCSCPKC shared key setup procedure and maintains the key list in a lookup table. MAP uses MIC computed with the shared key for authentication and data integrity check instead of signatures to save delay and computational overhead. This process sets up secure channels among MC, AR, HA and foreign MAP.

5.3 Route Optimization

SWMN uses Return Routability (RR) test [16] for secure Route Optimization (RO) as illustrated in Figure 5. and assumes no security association between (Correspondent Node) CN and MC. The messages used are listed in Table 2 and the definitions are as given in [16].

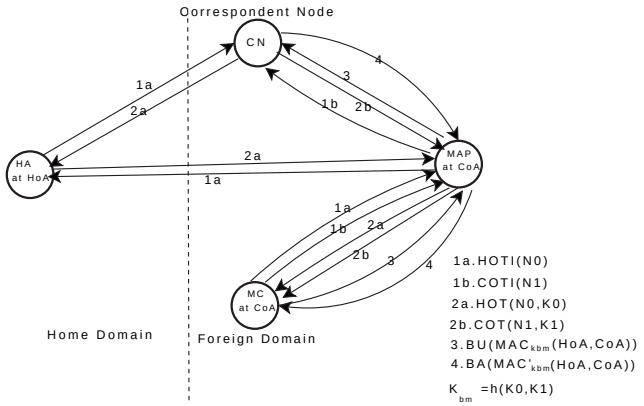

Figure 5: Message flow for RR test

The RR test secures the network against most common MIPv6 attacks such as connection hijacking attack, bombing attack, state storage exhaustion attack, CPU exhaustion attack, reflection and amplification attack.

6 Estimation of Cost

6.1 Analytical Model

In this subsection we describe an analytical model based on a 2-D cellular Configuration [1] for the WMN and random walk model for mobility. We make the following assumptions:

1) Each subnet that is managed by an AR is in the form of hexagonal cell;

2) Each regional domain that is managed by a MAP contains hexagonal cells, with the structure as shown in the Figure 6.

3) All the regional domains are of same size.

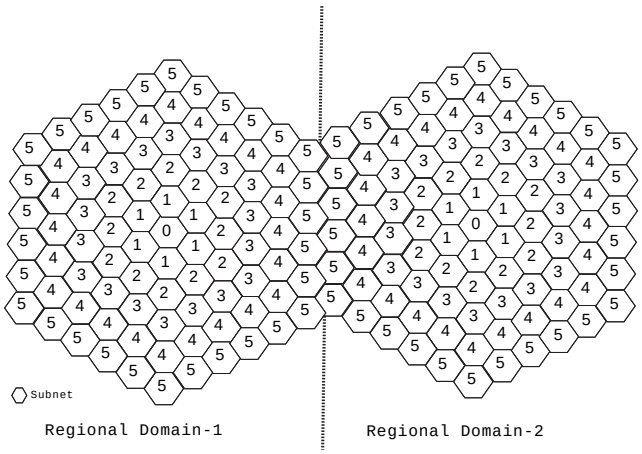

Figure 6: Cellular representation of SWMN

The inner most cell is labelled with '0' and the cell labelled with '1' forms the first ring around cell '0' and so on.

Let q be the probability that a MC stays in the current cell, then using random walk mobility model [1], the probability that movement of the MC will result in increasing distance r, from cell '0' denoted by $(p^+(r))$ or decreasing distance $(p^-(r))$ with respect to cell '0' are given by:

$$p^+(r) = \frac{1}{3} + \frac{1}{6r} \quad \text{and} \quad p^-(r) = \frac{1}{3} - \frac{1}{6r}$$

The movement of the MC with respect to cell '0' can be represented as an Markovian chain. Let $\alpha_{r,r+1}$ represents the transition probability that the movement will result in increasing distance from cell '0' and $\beta_{r,r-1}$ represents the transition probability that the movement will result in decreasing distance from cell '0'. Assuming that a MAP domain of R rings, the transition probabilities are given by:

$$\begin{aligned} \alpha_{r,r+1} &= \begin{cases} (1-q) & \text{if } r = 0 \\ (1-q)(\frac{1}{3} + \frac{1}{6r}) & \text{if } 1 \le r \le R \end{cases} \\ \beta_{r,r-1} &= (1-q)(\frac{1}{3} - \frac{1}{6r}) \qquad \text{if } 1 \le r \le R. \end{aligned}$$

Using the above transition probabilities, the steady state probability of state r, π_r, can be expressed as:

$$\pi_r = \pi_0 \prod_{i=0}^{r-1} \frac{\alpha_{i,i+1}}{\beta_{i+1,i}} \qquad \text{for } 1 \le r \le R$$

with the requirement $\sum_{r=0}^{R} = 1$, and π_0 can be expressed as

$$\pi_0 = \frac{1}{1 + \sum_{r=1}^{R} \prod_{i=0}^{r-1} \frac{\alpha_{i,i+1}}{\beta_{i+1,i}}}$$

Let C^x_{Ur}, C^x_{Uh} and C^x_{RO} represent costs for regional location update (local binding update), home location update (global binding update) and route optimization respectively. Here, the superscript x represents either HMIPv6 or SWMN. According to the mobility model presented [9, 18, 20], the average location update cost per unit time can be expressed as

$$C^x_{LU} = \frac{\pi_R.\alpha_{R,R+1}.(C^x_{Uh} + C^x_{RO}) + (1 - \pi_R \alpha_{R,R+1}) C^x_{Ur}}{T}$$

where T represents the average cell residence time that MC stays in a cell.

6.2 Location Update Cost

In this subsection we compute the costs for $HMIPv6$, $SWMN_S$, $SWMN_D$, $SWMN_I$, $SWMN_E$ systems. Table 3 depicts the notations and symbols used in this cost analysis.

6.2.1 HMIPv6

According to the message flow given in (3)-(7) and as per [28], the cost for location update with HA (termed as home registration cost) and the cost for location update with MAP (termed as regional registration cost) for each location update in HMIPv6 are given as follows:

$$\begin{aligned} C^{HMIPv6}_{Uh} &= 2a_l + 2a_p + a_h + 2(\rho + d_{pl} + d_{hp})\delta_U \\ C^{HMIPv6}_{Ur} &= 2a_l + a_p + 2(\rho + d_{pl})\delta_U. \end{aligned}$$

Since SWMN uses default RR test proposed in MIPv6 without any changes, the cost for RO in both HMIPv6 and SWMN is same. As per the message flow in Figure 5 and as per [16] the signaling cost for RO in HMIPv6/SWMN can be expressed as:

$$\begin{aligned} C^{HMIPv6}_{RO} &= 3a_m + 6a_p + 6a_l + 2a_h + 3a_c + (6\rho \\ &\quad +6d_{lp} + 2d_{ph} + 2d_{hc} + 4d_{pc})\delta_U). \end{aligned}$$

6.2.2 $SWMN_S$

Assume that the average HMIPv6 control packet size in location update is 128 bytes. RSA requires about 745 bytes additional overhead to carry certificate and signature [15]. Then the average control packet size in $SWMN_S$ becomes 873 bytes, which means the transmission cost to carry the $SWMN_S$ control packet is seven times that of HMIPv6 control packet. Therefore the proportionality constant for transmission cost of control packet in $SWMN_S$ is set to $7\delta_U$.

RSA uses digital signatures to protect the packets. It requires five signature verification and four signature generation operations during regional registration. It requires ten signature verifications and seven signature generation operations during home registration. Therefore the costs for home registration and regional registration for each location update in $SWMN_S$ are given as follows:

$$\begin{aligned} C^{SWMN_S}_{Uh} &= 2a_l + 2a_p + a_h + 2(\rho + d_{pl} + d_{hp})(7\delta_U) \\ &\quad +10\epsilon_S + 7\psi_S \\ C^{SWMN_S}_{Ur} &= 2a_l + a_p + 2(\rho + d_{pl})(7\delta_U) + 5\epsilon_S + 4\psi_S. \end{aligned}$$

Table 3: Notations used in cost analysis

Entity	Short representation		Cryptosystem	Short representation
MAP	p		RSA	S
AR	l		ECDSA	D
MC	m		IBC	I
HA	h		ECCSCPKC	E
CN	c			

Parameter	Meaning	Units
d_{xy}	Distance (in hops) between entity x and entity y	–
a_x	Processing cost of control packet at entity x	msec
δ_U	Transmission cost for control packet delivery	msec
δ_D	Transmission cost for data packet delivery	msec
n_y^x	Average number of entity y in entity x's coverage area	–
v_x	Data packet processing cost at entity x	msec
λ_a	Data packet arrival rate at MAP	packets/sec
λ_b	Arrival rate of first packet in a session at HA	packets/sec
η	Data Packet processing cost at HA	msec
ϵ_z	Cost for signature verification in cryptosystem z	msec
ψ_z	Cost for signature generation in cryptosystem z	msec
ρ	Proportionality constant for MC-AR wireless link	–
γ	Cost for MIC computation	msec
τ_z	Cost for shared key computation in cryptosystem z	msec
C_{Uh}^x	Home registration cost in system x	msec
C_{Ur}^x	Regional registration cost in system x	msec
C_{RO}^x	Route optimization cost in system x	msec
C_LU^x	Location update cost per unit time in system x	–
C_PD^x	Packet delivery cost per unit time in system x	–
C_T^x	Total signaling cost per unit time in system x	–

6.2.3 $SWMN_D$

ECDSA requires about 636 bytes additional overhead to carry certificate and signature [19], making the average control packet size in $SWMN_D$ 764 bytes. Therefore the proportionality constant for transmission cost of control packet is set to $6\delta_U$. ECDSA requires five signature verification and four signature generation operations during regional registration. It requires ten signature verifications and seven signature generation operations during home registration. Therefore the cost for home registration and regional registration for each location update in $SWMN_D$ are given as follows:

$$\begin{aligned} C_{Uh}^{SWMN_D} &= 2a_l + 2a_p + a_h + 2(\rho + d_{pl} + d_{hp})(6\delta_U) \\ &\quad +10\epsilon_D + 7\psi_D \\ C_{Ur}^{SWMN_D} &= 2a_l + a_p + 2(\rho + d_{pl})(6\delta_U) + 5\epsilon_D + 4\psi_D. \end{aligned}$$

6.2.4 $SWMN_I$

IBC requires about 64 bytes additional overhead to carry sequence number, Identity, and signature/MIC, making the average control packet size in $SWMN_I$ 192 bytes. Therefore the proportionality constant for transmission cost of control packet is set to $1.5\delta_U$. IBC requires two signature verifications, one signature generation, and two pairing computations to compute shared keys; and six MIC operations during regional registration. We assume that the shared key between MC and HA is pre-computed. It requires three signature verifications, one signature generation, and twelve MIC operations during home registration. Therefore the home registration cost and regional registration cost for each location update in $SWMN_I$ are given as follows:

$$\begin{aligned} C_{Uh}^{SWMN_I} &= 2a_l + 2a_p + a_h + 2(\rho + d_{pl} + d_{hp})(1.5\delta_U) \\ &\quad +3\epsilon_I + \psi_I + 2\tau_I + 12\gamma \\ C_{Ur}^{SWMN_I} &= 2a_l + a_p + 2(\rho + d_{pl})(1.5\delta_U) \\ &\quad +2\epsilon_I + \psi_I + 2\tau_I + 6\gamma. \end{aligned}$$

6.2.5 $SWMN_E$

ECCSCPKC requires about 84 bytes additional overhead to carry sequence number, Identity, public key and signature/MIC, making the average control packet size in $SWMN_E$ 212 bytes. Therefore the proportionality constant for transmission cost of control packet is set to $1.75\delta_U$.

ECCSCPKC requires two signature verifications, one signature generation, and two pairing computations to compute shared keys; and six MIC operations during regional registration. We assume that the shared key between MC and HA is precomputed. It requires three signature verifications, one signature generation and twelve MIC operations during home registration. Therefore the

home registration cost and regional registration cost for each location update in $SWMN_E$ are given as follows:

$$\begin{aligned} C_{Uh}^{SWMN_E} &= 2a_l + 2a_p + a_h + 2(\rho + d_{pl} + d_{hp})(1.75\delta_U) \\ &\quad +3\epsilon_E + \psi_E + 2\tau_E + 12\gamma \\ C_{Ur}^{SWMN_E} &= 2a_l + a_p + 2(\rho + d_{pl})(1.75\delta_U) + 2\epsilon_E \\ &\quad +\psi_E + 2\tau_E + 6\gamma. \end{aligned}$$

The communication costs for different cyprosystems used in SWMN are summarized in Table 4, and the computational cost for home registration and the regional registration for each location update are summarized in Table 5.

6.3 Packet Delivery Cost

Let n_m^l be average number of MC's in a AR's coverage area, n_m^p be the average number of MC's in a MAP's coverage area, and n_l^p be the number of ARs in a MAP's coverage area. Then they are related by:

$$n_m^p = n_l^p n_m^l.$$

The packet delivery cost comprises following cost components:

1) The packet processing cost at MAP;
2) The packet processing cost and, at HA;
3) The packet transmission cost from CN to MC, t_{CN-MC}. The packet processing cost per unit time in HMIPv6 can be expressed as:

$$C\_PD^{HMIPv6} = v_p + v_h + t_{CN-MC}. \quad (7)$$

The packet processing cost at MAP, *i.e.*, v_p has the following components: (1) Cost for lookup into table for mapping of RCoA into LCoA; and (2) Cost for lookup into the routing table for routing the packet to the concerned AR. Route optimization process allows CN to send the packets directly to MC without passing through HA. But the first packet from CN should tunnel through HA. Then the packet processing cost at MAP includes de-capsulation and en-capsulation costs of the tunnelled packet from HA. These costs are neglected for the sake of simplicity of analysis. The cost for lookup into (RCoA, LCoA) mapping table is proportional to the size of mapping table. The size of mapping table is proportional to the number of MCs in the MAP domain. The cost for lookup into routing table is proportional to the logarithm of the length of the routing table [28, 29] which is equal to the number of AR's in the MAP's domain. Let λ_a be the packet arrival rate at MAP, the packet processing cost at MAP can thus be expressed as:

$$v_p = \lambda_a(\alpha n_m^p + \beta log(n_l^p)), \quad (8)$$

where α and β are the proportionality constants for binding-table lookup and routing table-lookup, respectively.

He packet processing at HA is proportional to the arrival rate of first packet in the session and is given by:

$$v_h = \lambda_b \eta, \quad (9)$$

where λ_b is the session arrival rate and η is the unit packet processing cost at HA. Assuming that the average session size is σ packets, λ_b is $\frac{\lambda_a}{\sigma}$. That means λ_b packet/sec travel from CN to MC via HA and (λ_a-λ_b) packets/sec travel directly via MAP without passing through HA. Therefore the packet transmission cost has two components: i) the transmission cost of those packets which come via HA; and, ii) the transmission cost of those packets which come directly. Therefore, the packet transmission cost per unit time is given by:

$$\begin{aligned} t_{CN-MC} &= ((d_{cp} + d_{pl} + \rho)(\lambda_a - \lambda_b) + (d_{ch} + d_{hp} \\ &\quad +d_{pl} + \rho)\lambda_b)\delta_D. \end{aligned} \quad (10)$$

Substituting Equations (8), (9), and (10) in Equation (7), the packet delivery cost per unit time is given by

$$\begin{aligned} C\_PD^{HMIPv6} &= \lambda_a(\alpha n_m^p + \beta log(n_l^p)) + \lambda_b \eta \\ &\quad +((d_{cp} + d_{pl} + \rho)(\lambda_a - \lambda_b) \\ &\quad +(d_{ch} + d_{hp} + d_{pl} + \rho)\lambda_b)\delta_D. \end{aligned}$$

We assume that the data packets do not not use any cryptography, therefore the packet delivery cost per unit time in other four systems are also the same:

$$\begin{aligned} C\_PD^x &= \lambda_a(\alpha n_m^p + \beta log(n_l^p)) + \lambda_b \eta + ((d_{cp} + d_{pl} + \rho) \\ &\quad \cdot(\lambda_a - \lambda_b) + (d_{ch} + d_{hp} + d_{pl} + \rho)\lambda_b)\delta_D, \end{aligned}$$

where the superscript 'x' stands for any one of the four systems. The total signaling cost per unit time for HMIPv6 is given by:

$$C\_T^{HMIPv6} = C\_LU^{HMIPv6} + C\_PD^{HMIPv6}.$$

Similarly, the total signaling cost per unit time for SWMN systems is given by:

$$C\_T^x = C\_LU^x + C\_PD^x,$$

where the superscript 'x' stands for any one of the four systems.

7 Numerical Results and Discussion

The computation times on a node with 1GHz, Pentium-III processor are considered for numerical values. We use the following values from [21]: RSA signature generation and verification times of 7.9ms and 0.4ms, respectively; ECDSA signature generation and verification times of 5.77ms and 7.15ms, respectively; IBC Bilinear Signature (BLS) generation and verification times of 2.22ms and 45.8ms, respectively; and pairing computation for shared key generation of 20ms. AES encryption/decryption time for 128 Byte data is about 8.4μs

Table 4: Communication overhead (in Bytes) for different cryptosystems

	$\|Seq.No\|$	$\|Certificate\|$	$\|Identity\|$	$\|Signature\|$	$\|MIC\|$	Total	Cost factor
CBC-RSA	4	613	-	128	-	745	$7\delta_U$
CBC-ECDSA	4	592	-	40	-	636	$6\delta_U$
IBC	4	-	20	40	40	64	$1.5\delta_U$
ECCSCPKC	4	-	20+20*	40	40	84	$1.75\delta_U$

*Additional overhead due to public key

Table 5: Computational overhead (in msec) for different cryptosystems

	Regional Registration		Home Registration	
	cost expression	cost	cost expression	cost
RSA	$5\epsilon_S + 4\psi_S$	33.6	$10\epsilon_S + 7\psi_S$	59.3
ECDSA	$5\epsilon_D + 4\psi_D$	58.83	$10\epsilon_D + 7\psi_D$	111.89
IBC	$2\epsilon_I + \psi_I + 2\tau_I + 6\gamma$	90.27	$3\epsilon_I + \psi_I + 2\tau_I + 12\gamma$	92.53
ECCSCPKC	$2\epsilon_E + \psi_E + 2\tau_E + 6\gamma$	1.1	$3\epsilon_E + \psi_E + 2\tau_E + 12\gamma$	1.46

and SHA-1 takes 5.73μs [7]. ECCSCPKC signature generation and verification times are approximated in [26] as $30T_{MM}$+T_h and $166.36T_{MM}$+$2T_h$, respectively; where T_{MM} and T_h are computation times for modular multiplication and hash, respectively. Modular multiplication time defined over Galois field $F_{2^{163}}$ by a node with 1GHz, P-III processor is 1.9μs [27]. Accordingly, the ECCSCPKC signature generation and verification times are approximated as 0.062ms and 0.328ms, respectively. The cost of ECCSCPKC shared key computation is approximated as $87.24T_{MM} + T_h$ [26], which is approximately 0.172ms.

A 100Kbps average data rate is assumed over the wireless link between MC and AR, and 1Mbps average data rate over wireless link between AR and MAP and between MAPs. The control packet includes the IPv6 basic header (40 bytes) and some of optional IPv6 extension headers. The length of BU, BA, HOTI, HOT, COTI, COT are, respectively, 72, 64, 56, 64, 56, and 64 Bytes [3]. Therefore, for the sake of simplicity of numerical evaluation the average control packet size is taken as 128 bytes. The average additional control packet overhead due to security payload varies for different cryptosystems. The details are tabulated in Table 4. With 128 byte control packet, the average transmission delay over MC-AR wireless link is 10.24ms, and over AR-MAP and MAP-MAP wireless link it is 1ms. Assuming the average data packet size of 1KByte, the transmission cost for data packet delivery δ_D is 8ms. The parameter values are given in Table 6, and the corresponding numerical results are tabulated in Table 7.

From the numerical results presented in Table 7, the percentage additional cost overhead for home registration in $SWMN_E$, $SWMN_I$, $SWMN_D$ and $SWMN_S$ systems, over HMIPv6, are respectively, 100%, 168%, 610% and 652%. Similarly, the percentage additional cost overhead for regional registration in $SWMN_E$, $SWMN_I$, $SWMN_D$ and $SWMN_S$ systems, over HMIPv6, are respectively, 41%, 190%, 378% and 387%. From these values we conclude that, $SWMN_E$ has the minimal per location update cost overheads than other cryptosystems.

Next, we compare the performance of HMIPv6 and different SWMN versions on the basis of total signaling cost at varying: (1) Packet-to-Mobility Ratio (PMR); (2) average data packet arrival rate; and, (3) Average cell residence time, T. PMR is defined as the ratio of packet arrival rate to mobility rate, i.e., PMR is $\lambda_a T$.

7.1 The Impact of Packet-to-Mobility Ratio

Figure 7 shows the total signaling cost as a function of PMR for HMIPv6 and four SWMN systems. we have taken the region domain size of R=4, λ_a=[1-40] packets/sec, T=[0.1-20]. From the figure we can observe that, the bottom most solid line (lowest cost) represents HMIPv6 without security. The top most solid line (highest cost) represents $SWMN_S$. In between, the systems with increasing cost order are $SWMN_E$,$SWMN_I$ and $SWMN_D$. It can be observed that cost of all the systems are high at lower PMR values and approaching to a minimal value at medium PMR values and then increasing with increasing PMR values. This is because of the fact that at lower PMR values (either cell residence time is low or packet arrival rate is low) then the location update cost per unit time dominates packet delivery cost and when PMR is high (either user residence time in a cell (T) is large or packet arrival rate is high) then packet delivery cost dominates location update cost for all the systems. At medium PMR values the performance of the system is observed as optimum. From the figure we can infer that, both RSA and ECDSA incurs heavy cost on the system, ECCSCPKC has the minimal cost overhead of the SWMN systems and IBC has the moderate cost overhead.

Table 6: Cost parameters

parameter	a_l	a_p	a_h	ρ	d_{pl}	d_{hp}	ϵ_S	ψ_S	ϵ_D	a_m
Value	10	5	5	10	5	32	0.4	7.9	5.17	5
parameter	ψ_D	ϵ_I	ψ_I	τ_I	ϵ_E	ψ_E	τ_E	γ	δ_U	
Value	7.15	2.22	45.8	20	0.328	0.062	0.172	0.00573	1	
parameter	a_c	d_{hc}	d_{pc}	δ_D	α	β	n_m^p	η	λ_b	
Value	5	32	32	8	0.3	0.7	15	10	$\frac{\lambda_a}{\sigma}$	

Table 7: Numerical results

C_{Uh}^{HMIPv6}	$C_{Uh}^{SWMN_S}$	$C_{Uh}^{SWMN_D}$	$C_{Uh}^{SWMN_I}$	$C_{Uh}^{SWMN_E}$	C_{RO}^{HMIPv6}
99	752.3	710.9	268.53	200.96	476
C_{Ur}^{HMIPv6}	$C_{Ur}^{SWMN_S}$	$C_{Ur}^{SWMN_D}$	$C_{Ur}^{SWMN_I}$	$C_{Ur}^{SWMN_E}$	C_{RO}^{SWMN}
55	268.6	263.83	160.27	78.6	476

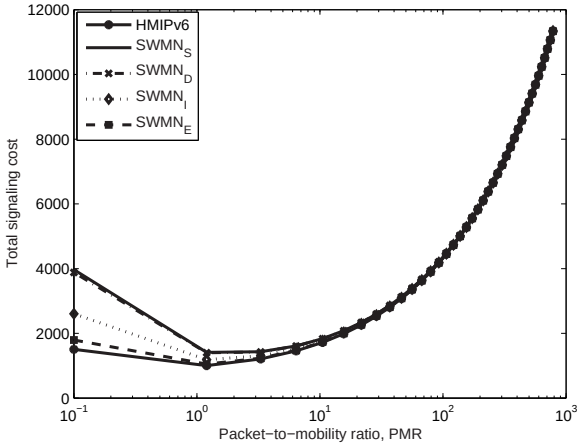

Figure 7: Effect of packet-to-mobility ratio (PMR) on total signaling cost

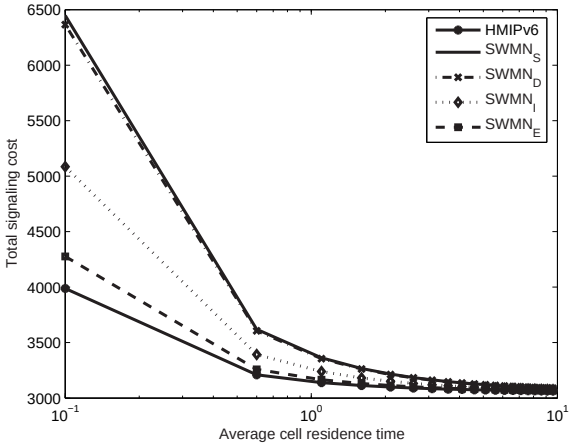

Figure 8: Effect of cell residence time (T)

7.2 The Impact of Average Cell Residence Time

We investigate the impact of user-variant mobility. Let packet arrival rate λ_a=10 packets/sec and regional network size R=4. Figure 8 shows the effect of cell residence time (T) on total signaling cost for various systems. From the figure we observe that, HMIPv6 has the lowest total signaling cost followed by $SWMN_E$, $SWMN_I$, $SWMN_D$ and $SWMN_S$. The total signaling cost of all the systems is converging to some minimal value as the cell residence time increases. As mentioned previously, this is because of the fact that, as the cell residence time increases the location update cost per unit time decreases. As a result the packet delivery cost dominates location update cost. From the figure we can conclude that, ECCSCPKC offers lowest cost of all SWMN systems.

7.3 The Impact of Average Packet Arrival Rate

The cell residence time T=1. The packet arrival rate is varied from 0.1 packets/sec to 10 packets/sec. Figure 9 shows the impact of packet arrival rate on various cryptosystems. Again, $SWMN_E$ offers minimal cost after HMIPv6, followed by $SWMN_I$, $SWMN_D$ and $SWMN_S$. Since residence time is fixed, at lower packet arrival rates the location update cost dominates the total cost. At higher packet arrival rates, all the curves are converging to the same cost. But the total costs in all cases are increasing monotonically with increase in packet arrival rate.

From the performance analysis, we can infer that, $SWMN_E$ is the best choice to achieve both security and performance.

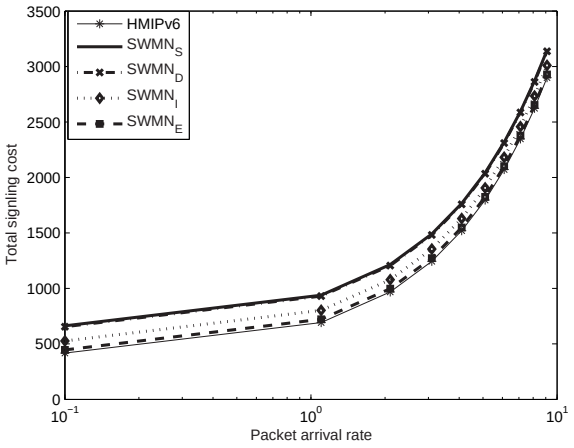

Figure 9: Effect of packet arrival rate

8 Conclusions

This paper considers four cryptosystems, namely, RSA, ECDSA, IBC and ECCSCPKC, for their suitability for protecting HMIPv6 based WMN. We found that the ECCSCPKC outperforms the other systems in offering the desired security while keeping the overhead minimum. Though the per location update costs are comparatively high than HMIPv6 costs, we observed that, the total costs of various SWMN systems are approaching to HMIPv6 total cost with increase in packet arrival rate or with increase in cell residence time, or both. This tendency is due to the fact that, packet processing cost dominates location update cost at higher packet arrival rates or cell residence time or both. Though IBC is comparable to ECCSCPKC, it has the problem of private key leakage at operator. Therefore, in IBC based systems the entire security of the system is compromised if operator is compromised. Moreover, ECCSCPKC offers a secure mechanism for user registration and public-private key setup, while the other systems are lacking this feature. The secure handover mechanism presented in this paper ensures mutual authentication among MC, AR, foreign MAP and HA and forms secure tunnels among them. SWMN is discussed for single operator case but it can be extended to multiple operator case as well.

References

[1] I. F. Akyildiz, and W. Wang, "A dynamic location management scheme for next generation multitier PCS systems," *IEEE Transactions on Wireless Communications*, vol. 1, no. 1, pp. 178-189, 2002.

[2] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 2, pp. 445-487, 2005.

[3] J. Arkko, and C. Vogt, *Credit-Based Authorization for Binding Lifetime Extension*, Internet draft, May 2004. (http://tools.ietf.org/html/draft-arkko-mipv6-binding-lifetime-extension-00)

[4] T. Aura, *Cryptographically Generated Addresses (CGA)*, IETF, RFC 3972.

[5] D. Boneh, and M.K. Franklin, "Identity-based encryption from the weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586-615, 2003.

[6] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," *Journal of Cryptology*, vol. 17, no. 4, pp. 297-319, 2004.

[7] Botan-a BSD licensed crypto library-benchmarks. (http://botan.randombit.net/bmarks.html)

[8] L. H. Chang, C. L. Lo, J. J. Lo, W. T. Liu, and C. C. Yang, "Mobility management with distributed AAA architecture," *International Journal of Network Security*, vol. 4, no. 3, pp. 241-247, 2007.

[9] C. W. Chen, M. C. Chuang, and C. S. Tsai, "An efficient authentication scheme between MANET and WLAN on IPv6 based Internet," *International Journal of Network Security*, vol. 1, no. 1, pp. 14-23, 2005.

[10] N. Doraswamy, and D. Harkins, *IPSec: The New Security Standard for the Internet, Intranets, and Virtual Private Networks*, Prentice Hall PTR, 2003.

[11] R. Dutta, R. Barua, and P. Sarkar, "Pairing-based cryptographic protocols: a Survey," *Cryptology ePrint Archive*, 2004.

[12] W. Haddad, S. Krishnan, and H. Soliman, *Using Cryptographically Generated Addresses (CGA) to secure HMIPv6 Protocol (HMIPv6sec)*, Internet draft, Aug. 2006. (http://tools.ietf.org/html/draft-haddad-mipshop-hmipv6-security-06)

[13] D. Hankerson, J. L. Hernandez, and A. Menezes, "Software implementation of elliptic curve cryptography over binary fields," *Proceedings of the CHES 2000*, LNCS 1965, pp. 1-25, 2001.

[14] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer publishers, 2004.

[15] R. Housley, W. Polk, W. Ford, and D. Solo, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, RFC 3280, 2002.

[16] D. Johnson, C. Perkins, and J. Arkko, *Mobility Support in IPv6*, IETF, RFC 3775.

[17] C. Kaufman, *Internet Key Exchange (IKEv2) Protocol*, IETF, RFC 4306.

[18] J. Li, P. Zhang, and S. Sampalli, "Improved security mechanism for mobile IPv6," *International Journal of Network Security*, vol. 6, no. 3, pp. 291-300, 2008.

[19] A. J. Menezes, S. A. Vanstone, and P. C. V. Oorschot, *Handbook of Applied Cryptography*, CRC press, 1996.

[20] S. Pack, and Y. Choi, "A Study on Performance of Hierarchical Mobile IPv6 in IP-Based Cellular Networks," *IEICE Transactions on Communications*, vol. E87-B, no. 3, pp. 462-469, 2004.

[21] S. L. Paulo, M. Barreto et al., "Efficient implementation of pairing-based cryptosystems," *Journal of Cryptoplogy*, pp. 321-334, 2004.

[22] C. Perkins, *IP Mobility Support for IPv4*, IETF, RFC 3220.

[23] A. Shamir, "Identity-based cryptosystems and signature schemes," *Advances in Cryptography- Crypto' 84*, pp. 47-53, 1985.

[24] M. S. Siddiqui, and C. S. Hong, "Security issues in wireless mesh networks," *Proceedings of MUE' 07*, pp. 717-722, 2007.

[25] H. Soliman, C. Castelluccia, K. Malki, and L. Bellier, *Hierarchical Mobile IPv6 mobility management (HMIPv6)*, IETF, RFC 4140.

[26] W. J. Tsaur, "Several security schemes constructed using ECC-based self-certified public key cryptosystems," *Applied Mathematics and Computation*, vol. 168, no. 1, pp. 447-464, 2005.

[27] A. Weimerskirch, D. Stebila, and S.C. Shantz, "Generic GF(2^m) arithmetic in software and its application to ECC," *proceedings of The Eighth Australasian Conference on Information Security and Privacy (ACISP)*, pp. 79-92, 2003.

[28] J. Xie and I. F. Akyildiz, "A distributed dynamic regional location management scheme for mobile IP," *Proceedings of IEEE INFOCOM*, vol. 2, pp. 1069-1078, 2002.

[29] C. Y. Yang and C. Y. Shiu, "A secure mobile IP registration protocol," *International Journal of Network Security*, vol. 1, no. 1, pp. 38-45, 2005.

[30] C. K. Yeh and W. B. Lee, "An overall cost-effective authentication technique for the global mobility network," *International Journal of Network Security*, vol. 9, no. 3, pp. 227-232, 2009.

Appendix A: Abbreviations Used in SWMN

WMN	Wireless Mesh Network
RSA	Ron Rivest, Adi Shamir and Leonard Adleman algorithm
ECDSA	Elliptic Curve Digital Signature Algorithm
IBC	Identity Based Cryptography
ECCSCPKC	Elliptic Curve Cryptography-based Self Certified Public key Cryptosystem
MIPv6	Mobile IP version 6
HMIPv6	Hierarchical Mobile IPv6
AR	Access Router
AP	Access Point
MC	Mesh Client
MIPv4	Mobile IP version 4
IKE	Internet Key Exchange Protocol
IPSec	Internet Protocol Security
MAP	Mobility Anchor Point
LCoA	On-Link Care-of-Address
RCoA	Regional Care-of-Address
HoA	Home Address
SWMN	Secure Wireless Mesh Network
PKG	Private Key Generator
ZKP	Zero Knowledge Proof
MIC	Message Integrity Check code
MAC	Message Authentication Code
HA	Home Agent
BU	Binding Update
BA	Binding Acknowledgment
HOTI	Home test Init
COTI	Care-of Test Init
HOT	Home Test
COT	care-of Test
RO	Route Optimization
RR test	Return Routability test
CPU	Central Processing Unit
SHA	Secure Hash Algorithm
AES	Advanced Encryption Standard
BLS	Bilinear Signature
PMR	Packet-to Mobility Ratio
CGA	Cryptographically Generated Addresses

Ramanarayana Kandikattu received the Bachelor of Engineering Degree in Electronics and Communication Engineering from Andhra University, India, in 1992 and the Master of Engineering Degree in Communication Systems from PSG College of Technology, Coimbatore, India, in 2003. He is currently a PhD student in the Department of Electronics and Communication Engineering, National Institute of Technology Calicut, India. His research interests include security in ad hoc networks and secure mobility management in wireless mesh networks. He is a student member of IEEE.

Lillykutty Jacob obtained the Bachelor of Science (Engineering) Degree in Electronics and Communication Engineering from Kerala University, India in 1983, the Master of Technology Degree in Electrical Engineering (Communication) from Indian Institute of Technology Madras, India in 1985, and the Doctor of Philosophy Degree in Electrical Communication Engineering from Indian Institute of Science Bangalore, India in 1993. She was with the Department of Computer Science, KAIST, South Korea, during 1996 1997, for Post Doctoral Research, and with the Department of Computer Science, National University of Singapore, during 1998 2003, as a visiting faculty. Since 1985, she has been with National Institute of Technology Calicut, India, where she is currently a Professor and Head of Electronics and Communication Engineering Department. She is a senior member of IEEE and has about 80 publications in various International Journals and International conferences.