Secure and Efficient Handover Protocols for WMNs

André Egners
Research Group IT Security
RWTH Aachen University, Germany
egners@itsec.rwth-aachen.de

Patrick Herrmann
Research Group IT Security
RWTH Aachen University, Germany
patrick.herrmann@rwth-aachen.de

Ulrike Meyer
Research Group IT Security
RWTH Aachen University, Germany
meyer@itsec.rwth-aachen.de

***Abstract*—Wireless Mesh Networks (WMN) consist of a wireless infrastructure of mesh routers which are connected to the Internet via mesh gateways. Mesh clients on the other hand connect to these routers. To make full use of the connectivity and services offered by a WMN, users should be able to securely *hand over* from one router to the next. In particular, keying material has to be supplied to the new router. Handover protocols designed for infrastructure WLAN cannot be directly applied here as they have clearly been designed with a trusted backbone in mind. In this paper we propose three complementary secure, efficient, and practical proactive handover protocols, which are able to cope with the unique characteristics of WMNs such as the wireless infrastructure and untrusted intermediaries. We have also implemented and evaluated our protocols using our WMN testbed and thus show the feasibility of our solution in time critical contexts.**

***Index Terms*—Handover, Wireless Mesh Networks, EAP, Key Management, Security, IPsec, RADIUS, Diameter.**

I. Introduction

Wireless Mesh Networks (WMNs) consist of a wireless infrastructure of Mesh Routers (MRs) which are connected to the Internet via Mesh Gateways (MGs). Some (or all) MRs act as Network Access Server (NAS) to Mesh Clients (MCs) and other MRs. MCs connected to a WMN can communicate with other MCs on the same WMN or any other node on the Internet. Additionally, MCs may also act as MRs.

MCs in a WMN are typically mobile devices and as such can move from the coverage area of one MR to the next during an ongoing connection like a VoIP call. A handover procedure ensures that the MC can move from some Current Mesh Router (CMR) to some Target Mesh Router (TMR) without any disruption of its ongoing connections. The security challenge of a handover procedure is to ensure that the connection between MC and TMR can be adequately secured while keeping the delay constraints. In particular, the keying material required to protect the connection has to be efficiently established. Running a full authentication via the TMR during handover for this purpose is typically too time consuming.

While there is no prior work on handover security targeted for WMNs, this topic has been extensively studied in infrastructure WLAN [1], [2], [3], [4]. However, these approaches share a major shortcoming that make them hard to deploy to WMNs. They typically transfer keying material over the wired backbone to the TMR. In a WMN where all connections are wireless such an approach would obviously leak the transferred keying material.

978-1-4673-5828-6/13/$31.00

In this paper, we propose three complementary secure and efficient handover protocols for WMNs. The first of these protocols, allows to proactively supply candidate TMRs with keying material as part of the initial EAP authentication of an MC joining the WMN. The other two protocols can be run at any point in time after successful authentication. The first of these protocols allows several TMRs to be proactively supplied with keying material but the MC cannot be sure that these TMRs have already received the keying material at the end of the protocol run. The second one has the advantage that an MC initiating the protocol can be sure that the one TMR supplied with keying material has received it when the protocol terminates. We implemented and evaluated the three newly proposed protocols in our live WMN testbed and integrated them in the de-facto standard Wireless LAN (WLAN) software hostapd and wpa_supplicant. The results of our extensive performance evaluation are presented as part of this paper.

Our paper is structured as follows: In Section II we briefly discuss related work on handover security in infrastructure WLAN and WMNs. Next, in Section III we shortly introduce the framework which we use as a basis to securely interface our handover protocols with. We detail the new handover protocols in Section IV, evaluating their performance in Section V and their security in Section VI. We conclude our paper with a discussion in Section VII.

II. Related Work

CAPWAP [5] was designed to simplify deployment and management of enterprise WLAN infrastructures. The functionality of an AP is split into two components, i.e., Wireless Termination Point (WTP) and the Access Controller (AC). A WTP implements the PHY layer and lower portions of the MAC layer functionalities. Note, that this includes that a Station (STA) is able to secure the communication with a WTP on the link layer using standard IEEE 802.11i [6] mechanisms. The second component is the AC which implements the upper portions of the Medium Access Control (MAC) layer, including authentication and access control features. When a STA moves to another WTP it executes the 4-way handshake [6] with the AC instead of a full Extensible Authentication Protocol (EAP) authentication. Subsequently the required keys are sent from the AC to the WTP reducing the communication between the STA and the Authentication, Authorization and Accounting (AAA) server which is typically

further away. Therefore, the ability to perform fast handover in Configuration And Provisioning for Wireless Access Points (CAPWAP) is a side effect of moving the authentication process to a central component nearby. However, splitting the Access Point (AP) functionality makes it hard to deploy to standardized devices. Additionally, the secure key transport between WTPs, ACs and AAA server is not considered for WMNs where untrusted intermediaries may be present. In CAPWAP several ACs may exist which control a set of WTPs each. If a STA moves to a WTP belonging to different ACs, a full EAP authentication is still required. Therefore, CAPWAP does not necessarily prevent long handover delays.

The EAP re-Authentication Protocol (ERP) [7] is a proposed standard of the IETF. Its purpose is to avoid a full EAP authentication when a STA re-authenticates. Depending on the EAP-method, multiple round-trips between the AAA server and the peer may be required. ERP aims to reduce the handover disconnection time when a STA roams to another domain and after its first authentication in that domain. However, if deployed directly in a WMN, the transport of the re-authentication keys is not sufficiently protected against intermediaries as it is sent in a Remote Dial-in User Service (RADIUS) message which is only protected by MD5 [8]. Therefore, it is possible for intermediate, untrustworthy nodes in a WMN to compromise the handover keys.

The IEEE 802.11r standard [9] delivers a key to the AP to which the STA has first connected. On handover this AP derives further keys and distributes them to the target AP to which the client is moving. This approach is not usable in WMNs since each APs may easily be compromised. In addition, the standard assumes pairwise keys between the APs to securely transport the handover keys without specifying how these keys are to be established.

In [10] Mi-Ho et al. propose a proactive key distribution to reduce the re-authentication delay. Keying material is distributed to neighboring APs of the serving AP. It is assumed that if the STA moves to another AP, it is likely to be one of the neighboring APs of the serving AP. Note that the AAA server is responsible for deriving and delivering the keys to the handover destinations. In a WMN the path from the AAA server to the TMR may contain untrustworthy nodes such that again this approach cannot be applied to WMNs directly. The authors do not specify how the AAA server gains knowledge of the movement of the STA from one AP to another.

The above shows that most handover protocols proposed for infrastructure WLAN are of reactive nature and are thus only able to achieve a hard handover which tends to break ongoing sessions. In addition, secure key transport to the handover destinations is either not addressed or not completely solved in prior approaches. We address these shortcomings by introducing secure, efficient and practical proactive handover protocols achieving a soft handover.

III. Preliminaries

Our handover protocols are built on top of Framework for establishing Security Associations for Sequentially Deployed WMN (FSASD) [11] as it is the only security framework for WMNs which explicitly addresses insider attacks, is compatible to the IEEE 802.11s standard, and can be implemented using OTS hardware supporting the IEEE standards 802.1X and 802.11i. In FSASD each device is authenticated using a key generating EAP method.

The Extended Master Session Key (EMSK) generated during EAP authentication is used as root in a hierarchy of keys. From the EMSK an IPsec security association (containing a Traffic Encryption Key (TEK) and a Traffic Integrity Key (TIK)) is derived. If later on the node N_1 that joined the network acts as NAS, these keys are used to protect the authentication traffic between N_1 and the AAA server with Internet Protocol Security (IPsec). The two remaining keys Peer Authentication Key (PAK) and Key Derive Key (KDK) in the key hierarchy are used for authentication and key derivation during bootstrapping of the required security associations.

FSASD also allows to bootstrap pairwise security associations between any two authenticated nodes in the WMN by using the 3-Party Handshake Protocol for Sequential Deployment (3PHSD) which interfaces with FSASD. In particular, 3PHSD can be used to set up the 802.11i link layer security association between a moving MC (or MR) and its new NAS during handover.

IV. Handover Protocols

The goal of the protocols is to establish a key, i.e., Pairwise Master Keys (PMKs) as known from IEEE 802.11i to MC and the handover destination TMRs. Once the MC decides to associate to one of the TMRs, the two can simply use the established PMK to carry out the 4-way handshake instead of running a full EAP authentication. Thus, the re-association delay (or handover delay) when moving from one MR to another can be greatly reduced.

A. Assumptions

1) Network Assumptions : We assume the network to be operated by a single operator. WMN devices have been deployed according to FSASD, thus sharing the key hierarchy with the AAA server (cf. Section III). As a result, each MR has an IPsec connection to the AAA server of the network operator which represents a confidential, integrity protected and authentic channel. As also the MCs share the key hierarchy with AAA server, a secure channel can be ensured.

2) Key Distribution : In coherence with the the EAP security model, all handover keys are generated by the AAA server and the MC. In particular, the EMSK never leaves the AAA server. In our handover procedures confidential, integrity protected, and authentic key transport from AAA to the TMRs is ensured using IPsec. The required security association is established when TMR joins the WMN. Note that transporting key material from a AAA to NAS is covered by the standard use of EAP over RADIUS. However, proactively delivering handover keys to TMRs is not yet addressed by any standard. We therefore designed and implemented a key transport protocol between AAA and TMRs, which is used in our new

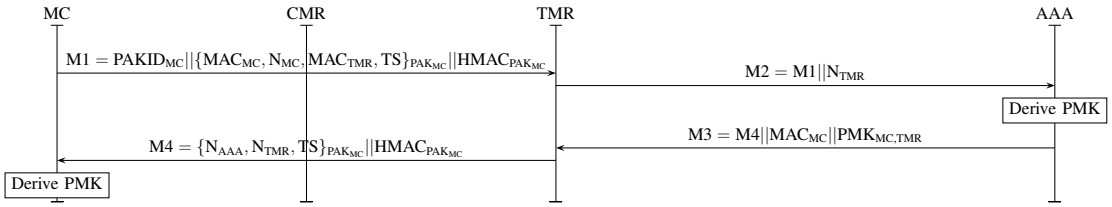


Fig. 1: 3PHSH Protocol

protocols. In this protocol, the TMRs listen for incoming key deliveries from the AAA server. The IPsec SA which was created upon the initial authentication of the TMR ensures that only authenticated and integrity protected key deliveries can be received on this connection. Confidential, integrity protected and authentic delivery of the parameters necessary to generate the handover keys at the MC is ensured by using the PAK of the FSASD key hierarchy.

3) 802.11i PMKSA-Cache : The PMKSA (Pairwise Master Key Security Association) describes a security association between STA and AP. The PMK is used between STA and AP to carry out the 802.11i 4-way handshake which generates keys that are subsequently used to secure the link layer traffic between both parties. The PMK is stored with context information such as the AP MAC address, the lifetime of the PMK and a unique ID called. If a STA and AP share a PMKSA, e.g., because the STA was connected to the AP before, both can use the cached PMK in the 4-way handshake directly [6] instead of running EAP to establish a PMK. For this purpose the STA retrieves the MAC address of the AP from its beacons and sends a (Re)Association Request to the STA including the PMKID in the RSN Information Element of the request. If the AP successfully determined that it has cached the respective PMK for the PMKID it directly starts the 4-way handshake with the STA. If no PMKSA is cached the STA has to run a full EAP authentication which takes significantly longer.

4) PAKID : To preserve the identity privacy of MCs during handover we introduce the PAKID which is generated as follows: PAKID = HMAC-SHA128(PAK, 'PAK Name' ‖ User ‖ 'roaming'). It is used by the AAA, to map a specific PAK to the respective MC. This mapping is necessary since the AAA needs to select the correct PAK to authenticate and en-/decrypt messages of 3-Party Handshake for Handover (3PHSH) and Neighborhood Pre-Authentication (NPA).

The generation of the PAKID is similar to the generation of the PMKID [6]. The PAK shared between MC and AAA is used to key the Keyed-Hash Message Authentication Code (HMAC) based on SHA1. Additionally, a label is supplied along with the long term user name of the MC and the purpose of the ID. As such, the PAKID serves as a temporary Unique Identifier (UID) during the protocol runs of 3PHSH and NPA which is sent in plain from MC to AAA. Intermediate nodes cannot deduce long term identities of MCs from overhearing the PAKID.

B. 3-Party Handshake for Handover (3PHSH)

3PHSH is an extension of the three party protocol proposed in [11]. In context of handover, we adapted the original proposal such that the three parties are (1) an MC, (2) an MR as the handover target, and (3) the AAA which is involved in the handover key generation (cf. Figure 1). We assume that both, the MC and the TMR have been authenticated using EAP according to [11], i.e., they share a specific set of cryptographic keys with the AAA server which are essential to securing the key derivation and the key transport. For implementation purpose we extended the 802.11 wireless beacon using a vendor specific Information Element (IE) containing the IP address of the TMR. The MC can obtain the IP address from the beacon to initiate communication with the TMR on the network layer.

A MC can initiate the protocol with a specific TMR. 3PHSH consists of the messages shown in Figure 1. Message M1 contains the PAKID which is used by the AAA server to map a PAK to a specific MC, as it needs to be able to decrypt some parts of the message and to verify its the integrity and authenticity using the HMAC. The content is encrypted using the PAK shared between the MC and the AAA. Once the TMR receives M1, it appends a nonce (N_{TMR}) to the message and relays it to the AAA server via a secure channel, i.e., the IPsec connection established between TMR and AAA, as message M2. After receiving M2, the AAA generates the PMK to be shared between the MC and the TMR as: $PMK = KDF(KDK_{MC}, label \parallel N_{TMR} \parallel N_{AAA} \parallel N_{MC} \parallel MAC_{MC} \parallel MAC_{TMR})$. The key derivation function is keyed with the KDK_{MC} shared between MC and AAA server according to the FSASD key hierarchy. Additionally, a key label is required, as was well as random nonces of the three parties, and the MAC addresses of the MC and TMR.

Now the AAA delivers the encrypted contents needed by the MC for generating the PMK to the TMR as Message M3. It also appends the PMK and the MAC address of the MC. The MAC address is used by the TMR as an input to generate the PMKID as: $PMKID = HMAC\text{-}SHA1\text{-}128(PMK, label \parallel MAC_{TMR} \parallel MAC_{MC})$. Finally, the TMR only forwards the encrypted parameters to the MC as Message M4. Now the MC is able to generate and insert the PMK into its PMKSA-cache [6]. Based on the PMK and the corresponding PMKID, the MC can now initiate the 4-way handshake with the TMR.

C. Neighborhood Pre-Authentication (NPA)

The so-called Neighborhood Pre-Authentication (NPA) protocol is similar to 3PHSH and can be triggered anytime by a

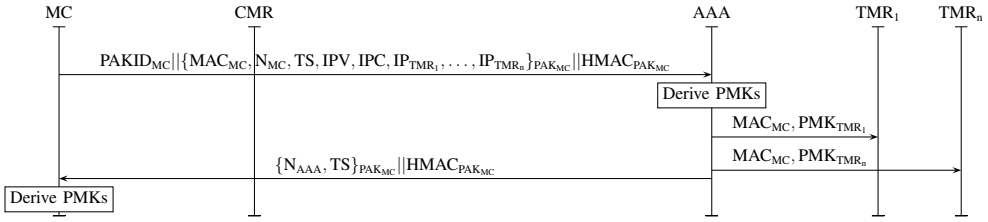


Fig. 2: NPA Protocol

MC after its authentication. NPA is able to initialize multiple handover TMRs in a single protocol run. Additionally, the message overhead is reduced by two messages by communicating directly with the AAA instead of the TMR as an intermediary. Figure 2 depicts a protocol run of NPA. In this example the MC's goal is to establish PMKs with TMR_1 and TMR_n which can potentially be used for handover.

In Message M1 the MC requests PMKs to be generated by the AAA for a set of potential handover candidate TMRs. It includes the relevant parameters to generate the PMKs, i.e., the MC's MAC address, a random nonce and a timestamp of the MC, and most importantly, the list of IP addresses identifying the TMRs. Those are necessary as the AAA needs to be able to address the TMRs. IP version number (four/six) and an IP address count are also included. The mentioned parameters are encrypted using the PAK_{MC} which is shared with the AAA. The HMAC keyed with the PAK_{MC} allows the AAA to ensure data integrity and authenticity. The PAKID is used to map the MC's identity at the AAA.

Once the MC has received Message M2 from the AAA, it can derive the PMK using the provided inputs, as well as the IP address of the respective TMR: $PMK_{MC,TMRi} = KDF(KDK_{MC}, \text{label} \parallel MAC_{MC} \parallel N_{MC} \parallel N_{AAA} \parallel IP_{TMRi})$. Again, as in 3PHSH, the IP addresses of possible handover TMRs are obtained using the vendor specific IE of the 802.11 beacon.

The AAA also sends a message containing the MC's MAC address and the individually generated PMK to each TMR requested by the MC. Each TMR can now generate the PMKID used to map the PMK and insert it into its PMKSA-cache. The MC is now able to use the established PMK during handover to an initialized TMR by sending the respective PMKID in an association request. If a mapping is found both can directly start the 4-way handshake.

D. EAP-TTLS Neighborhood Pre-Authentication (ENPA)

We propose another mechanism to initialize multiple TMRs for handover directly during the initial authentication. ENPA is currently realized as an extension of the EAP-TTLS [12] authentication method. However, it can easily be applied to any other EAP method that allows the transport of Diameter Attribute Value Pairs (AVPs) [13]. The AVPs used by EAP-TTLS and Diameter are syntactically equivalent. As in the protocols described in the previous sections, the MC itself is responsible to specify for which TMRs handover keys should be established.

When an MC associates to the network it scans its surrounding and acquires a number of available TMRs. It also retrieves their IP addresses from the IEEE 802.11 beacon. The MC embeds the IP addresses of the TMRs it chose to prepare for a potential handover along with its MAC address. The MC's MAC address must be used, as it is required by TMRs to generate the PMKID to map the handover PMK to an associating MC.

The PMK is generated as: $PMK_{MC,TMRi} = KDF(KDK_{MC}, \text{label} \parallel IP_{TMRi} \parallel \text{tls\_client\_random} \parallel \text{tls\_server\_random})$. The AAA generates distinct PMKs for each of the embedded IP addresses received from the MC in the AVP. The key derivation uses $KDK_{MC,AAA}$ of the FSASD key hierarchy, a key label, and the individual IP address of the respective TMR. Additionally, tls_client_random and tls_server_random of the EAP-TTLS session are used as salt values similar to the nonces in 3PHSH and NPA.

Once the AAA has generated the PMKs it sends the key, along with the MC's MAC address to TMRs. Note that this particular key transport is encrypted and integrity protected by an IPsec security association between the AAA and the TMR according to FSASD. The AAA sends the necessary key derivation parameters to the MC in an AVP of the RADIUS-Access-Accept message which marks the end of a successful EAP authentication. The MC can now derive the PMKs, generate PMKIDs and insert them into its PMKSA-cache.

Once a handover becomes necessary, the MC simply selects a corresponding PMK and queries the TMR with the according PMKID used in the association request. MC and TMR can then carry out the 4-way handshake based on the PMK.

V. Performance Evaluation

This section presents the performance evaluation of our proactive handover protocols 3PHSH, NPA and ENPA using our live WMN testbed.

A. WMN Testbed Setup

Our WMN testbed uses PC Engines ALIX system boards. All devices run on *Voyage-Linux*, which is a Debian Squeeze based embedded Linux distribution. Each device has a 500 MHz AMD Geode CPU, 256 Megabytes of RAM and two Atheros AR5008 wireless controllers supporting 802.11/a/b/g/n modes. The testbed uses the *batman-adv* (v2012.0.0) routing protocol which automatically adapts to new network topologies. The first wireless card allows MRs to connect other MRs of the WMN, while the second card can be used to distribute connectivity, i.e., act as NAS to other

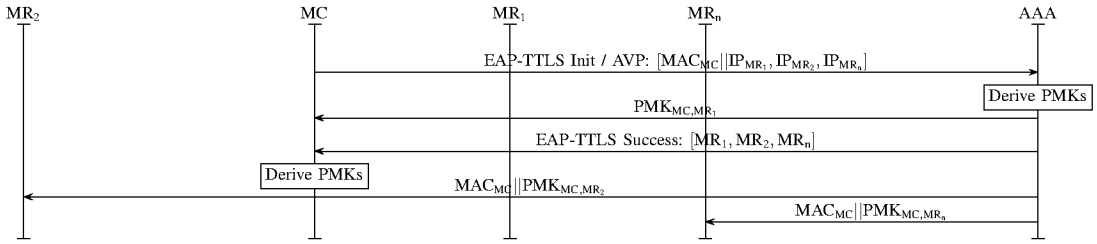


Fig. 3: ENPA Protocol

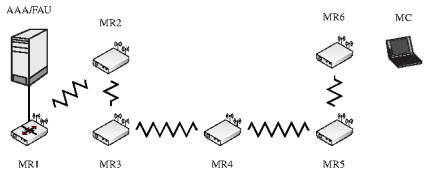


Fig. 4: 3PHSH/ENPA/NPA Evaluation Topology

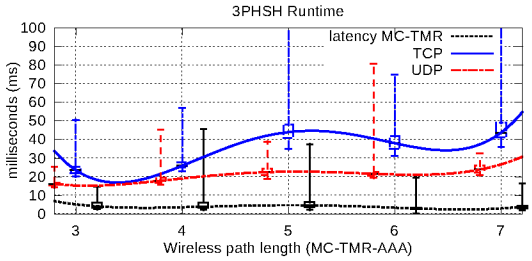


Fig. 5: 3PHSH Results

MRs or MCs. The AAA server of the testbed is located on a virtual machine running on a server box. It is based on Debian Linux 6.0.2 using an Intel(R) Xeon(R) E5450@3.00 GHz with 1024 Megabytes RAM.

B. Methodology

All measurements have been repeated 100 times in each step. The depicted results represent median values, their respective quartiles, and standard deviations.

C. 3PHSH Evaluation

Figure 4 shows the topology that has been used to evaluate the performance of 3PHSH. The purpose of this specific network topology is to increase the number of wireless hops between the MC requesting handover and the AAA generating and distributing the handover keys by one in each test run. In the i-th test run, the MC is connected to MRi (CMR) and will request a handover key for MR$(i+1)$ (TMR).

During evaluation, the distance from MC to the AAA server proved to be most relevant for the protocol's runtime. Figure 5 shows the runtime of 3PHSH using TCP and UDP. In order to make a qualitative statement about the link quality of communication path, we additionally included the latency between MC and TMR. UDP is roughly 50% faster (average) than TCP which can be accounted for by the absence of a handshaking mechanism. However, both are obviously influenced by the

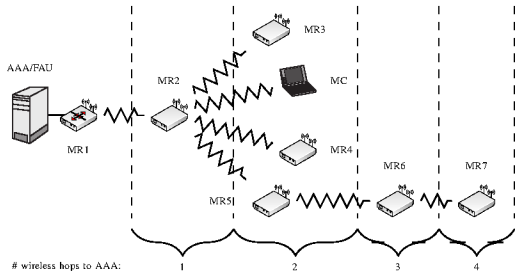


Fig. 7: NPA Evaluation Topology

current link quality. The maximum message size of 3PHSH is 107 byte, which is far less than the Maximum Transmission Unit (MTU) of the testbed (1528 byte). Thus, this result supports the conclusion that the runtime of the protocol is directly proportional to the cumulative link latencies between MC and AAA, as well as between TMR and AAA.

D. NPA Evaluation

NPA was evaluated using the different topologies. Additionally, we analyzed how the number of TMRs included in the MC's request influences the protocols' runtime.

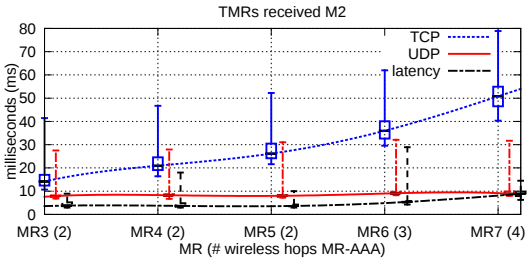
At first the topology shown in Figure 7 was used. The CMR to which the MC is connected remains MR2 in each run. We iteratively increased the maximum distance between TMRs and AAA by one hop. Additionally, we increased the amount of requested TMRs by one. In the last run the MC requests PMKs for MR3, MR4, MR5, MR6, and MR7.

To determine the relevant runtime we computed two values; Δt_{MC} which is the time when the MC receives the final message form the AAA enabling it to derive the PMKs. And Δt_{TMR_i} which marks the time that the TMRs receive the respective PMK from the AAA.

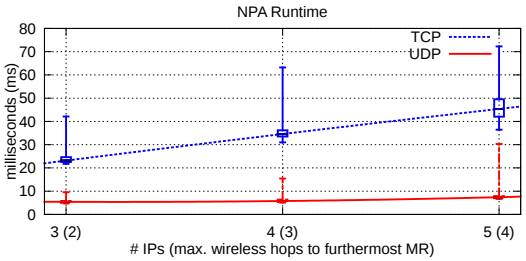
Figure 6(a) shows the result of the last run, i.e., PMKs for all TMRs are requested. The values on the x-axis denote the specific TMRs. The respective y-axis shows the time it took till M2 is received at the TMR. As expected, sequentially delivering the PMKs using TCP has a linear effect on the runtime. Delivering a total of five PMKs to MR3-MR7 takes approximately 50 ms using TCP, and 10 ms using UDP.

Figure 6(b) shows the duration of the TMRs receiving the PMKs for both TCP and UDP. Again, UDP is approximately proportional to the latency between the AAA, TMR, and MC.

Additional measurements using a star topology revealed that the distance to the AAA is the decisive factor rather than



(a) TMR key reception Δt_{TMR_i}



(b) MC key reception Δt_{MC}

Fig. 6: NPA Results

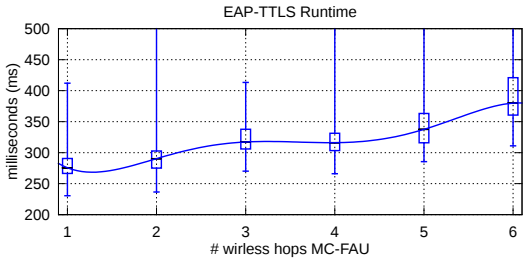


Fig. 8: ENPA Results

Requirement (1-11)	3PHSH	NPA	ENPA
1. Crypto-algorithm independent	yes (no negotiation)		
2. Strong fresh session keys	yes	yes	yes
3. Limit key scope	yes	yes	yes
4. Replay detection mechanism	yes	yes	yes
5. Authenticate all parties	yes	yes	yes
6. Peer and authenticator authorization	indirect		
7. Keying material and confidentiality and integrity	yes	yes	yes
8. Confirm cipher suite selection	no (no negotiation)		
9. Uniquely names keys	yes	yes	yes
10. Prevent the Domino effect	yes	yes	yes
11. Bind key to its context	yes	yes	yes

TABLE I: Security Analysis Summary

the number for requested TMRs. When using TCP to deliver the PMKs to the TMRs, its handshake plays into the overall runtime. Parallelization of PMK delivery is likely reduce to the runtime down to approximately the time it takes to deliver a PMK to a single TMR in the best case.

E. ENPA Evaluation

By considering the analysis results of 3PHSH and NPA, and also prior results shown in [11], it has become evident that the number of wireless hops (MC and TMRs) to the key distribution component, i.e., the AAA server, is the decisive factor for the protocols' runtime. Thus, we used the topology shown in Figure 4 in order to evaluate both vanilla EAP-TTLS and our extension, ENPA. We varied the number of wireless hops between the MC and the AAA, from 1-6. Figure 8 shows the runtime of ENPA and confirms the results obtained in [11].

Because the results for EAP-TTLS already revealed the effects of varying distance between MC and AAA, we fixed the number of wireless hops between MC and AAA to *one*. Instead we varied the number of IPs included in the EAP-TTLS AVP as this is the only difference in communication between MC and AAA. The runtime for EAP-TTLS increases roughly about 20 ms per wireless hop, whereas adding an IPv4 address increases the duration about 7 ms. Considering that ENPA is envisioned to be used during initial authentication, and possibly when the EAP session times out, this increase is negligible.

VI. Security Considerations

We chose to base our security analysis on RFC4962 [14] "Guidance for AAA Key Management" as we rely on an AAA infrastructure. The RFC belongs to the "Best Current Practices" category and describes conditions that a AAA protocol or a collection of protocols from which one of them is a AAA protocol should satisfy. All our proposed protocols exploit the security properties provided of an FSASD deployed WMN. Especially the fact that each MR has an IPsec channel to the AAA server. Table I shows an overview of the results of security analysis. Each of our protocols provides the same security properties required in RFC4962.

(1) Our proposals use as specific instantiation of cryptographic algorithms which can easily be replaced by alternatives without affecting the protocols. However, actual algorithm negotiation and confirmation (8) between the communicating nodes is not explicitly supported. (2) Each protocol execution produces strong fresh session keys, i.e., the PMKs of 256 bits in length. The keys are generated using PRF+ which is recommended and the current best practice [15]. (3) The keys used to generate PMKs, as well as the handover keys themselves have a clearly defined scope. PAKs and KDKs shared between a node and the AAA are used to secure the protocol messages and generate handover keys. The PMKs on the other hand are only used during the 4-way handshake. (4) Replay protection related to PMK delivery from AAA to the TMRs is achieved by their mutual IPsec SA. Protocol messages of 3PHSH and NPA between the MC and AAA rely on time stamps to detect message replay. Loose time synchronization is required.

(5) All parties are authenticated during network deployment by the AAA based on their individual credentials. Message authenticity is assured by IPsec between AAA and the TMRs, and HMACs using the PAK between MCs, AAA, and TMRs.

	3PHSH	NPA	ENPA
Protocol initiation	anytime	anytime	EAP auth.
# TMRs per run	1	$n \geq 1$	$n \geq 1$
PMK reception	guaranteed	opportunistic	opportunistic
IP connectivity	yes	yes	no
# Messages (n=TMRs m=EAP messages)	4	2 + n	n + m
Standard compliant	√	√	√

TABLE II: Protocol Properties Summary

(6) The involved parties implicitly demonstrated possession of relevant keys in each protocol. If a party does not possess the respective key, it is unable to successfully send and receive messages that will be processed by the other parties. This is either enforced by IPsec, or HMACs on the message content. (7) Confidentiality of keying material is either ensured by IPsec between AAA and the TRMs, or by encryption based on the PAK. Keying material transported from and to MCs is secured by the PAK. (9) All keys are uniquely named by using a key label which is also strongly related to the keys' usage. (10) Authenticators, i.e., CMRs and TMRs only hold a limited amount of key material with as specific lifetime. Compromise only allows to access current and new keying material associated with this specific authenticator; others are not directly affected. (11) Key context is explicitly established during key generation using the key label.

VII. Discussion

The proposed protocols 3PHSH, NPA, and ENPA are each proactive and can be used at different epochs of a MC's network connection. Each is able to proactively, i.e., before it is actually necessary, establish fresh PMKs with TMRs that enable the MC a fast and efficient association and authentication based on the 802.11 4-way handshake. ENPA is used at the very beginning of a connection, and can be used anytime the EAP session is refreshed. 3PHSH and NPA are both *post-authentication* protocols, i.e., they are to be used after an initial network association and EAP-authentication. NPA being more efficient in terms of the communication overhead allows to prepare multiple TMRs for fast handover of MCs, whereas each 3PHSH protocol run only bootstraps a single handover destination while offering TMR consent.

Considering the performance evaluation and security analysis of the protocols, it boils down to the performance and their point of execution as the protocols. The message overhead of 3PHSH is $4 \times n$, and $2 + n$ respectively for NPA where n is the number of TMRs for which PMKs are requested. Using ENPA is not considered to be time critical, as MCs will be associate to the CMR after EAP authentication rather than directly handover. Its message overhead is $n + m$ where m is the number of messages of the EAP method. Using EAP-TTLS results in $m \geq 4$. Thus, the protocols can be used alongside each other; ENPA whenever a full EAP-authentication becomes necessary, and 3PHSH or NPA as the MC moves through the WMN.

VIII. Conclusion

In this paper we proposed novel proactive handover protocols for WMNs which are *secure*, *efficient* and *practical*. Table II summarizes their key properties. Contrasting to the highlighted proposals for infrastructure WLAN, our solutions do not suffer from a bootstrapping problem. Using our protocols alongside the FSASD architecture enables us to meet a comprehensive set of security requirements for protocols in the context of AAA key management. 3PHSH, NPA, and ENPA are envisioned to be used in an interplay allowing to proactively instantiate handover candidates as the MC strides through different epochs of its network session. The practical evaluation using a live WMN testbed allowed us to determine the performance of the protocols, and additionally profile the related wireless properties of the 4-way handshake and scanning the spectrum. Altogether, the resulting performance highlights the applicability of our protocols in a time critical context, without negatively impacting ongoing sessions. Correctly deciding the point in time a handover would be beneficial, as well as designing a service to announce and discover handover candidates or key distribution components near to MCs leaves room for further research.

References

[1] T. C. Clancy, "Secure Handover in Enterprise WLANs: CAPWAP, HOKEY, and IEEE 802.11r," *Wireless Communications, IEEE*, vol. 15, pp. 80–85, October 2008.

[2] T. Clancy, M. Nakhjiri, V. Narayanan, and L. Dondeti, "Handover Key Management and Re-Authentication Problem Statement," mar 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5169.txt

[3] K. Hoeper, M. Nakhjiri, and Y. Ohba, "Distribution of EAP-Based Keys for Handover and Re-Authentication," mar 2010. [Online]. Available: http://www.ietf.org/rfc/rfc5749.txt

[4] R. Marin-Lopez, Y. Ohba, F. Pereniguez, and A. F. Gomez, "Analysis of Handover Key Management schemes under IETF perspective," *Comput. Stand. Interfaces*, vol. 32, no. 5-6, pp. 266–273, Oct. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.csi.2009.10.001

[5] B. O'Hara, P. Calhoun, and J. Kempf, "Configuration and Provisioning for Wireless Access Points (CAPWAP) Problem Statement," February 2005. [Online]. Available: http://www.ietf.org/rfc/rfc3990.txt

[6] IEEE, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 6: Medium Access Control (MAC) Security Enhancements (IEEE Std 802.11i-2004)," Jul. 2004.

[7] V. Narayanan and L. Dondeti, "EAP Extensions for EAP Re-authentication Protocol (ERP)," August 2008.

[8] C. Rigney, S. Willens, A. Rubens, and W. Simpson, "Remote Authentication Dial In User Service (RADIUS)," June 2000, updated by RFCs 2868, 3575, 5080.

[9] IEEE, *Amendment 2: Fast Basic Service Set (BSS) Transition*, jul 2008.

[10] A. M. Min-Ho, M.-h. Shin, and W. A. Arbaugh, "Pro-active Key Distribution using Neighbor Graphs," *IEEE Wireless Communications*, vol. 11, pp. 26–36, 2004.

[11] A. Egners, H. Fabelje, and U. Meyer, "FSASD: A Framework for Establishing Security Associations for Sequentially Deployed WMN," in *IEEE WoWMoM*, June 2012.

[12] S. B.-W. P. Funk, "Extensible Authentication Protocol Tunneled Transport Layer Security Authenticated Protocol Version 0 (EAP-TTLSv0)," August 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5281.txt

[13] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, "Diameter Base Protocol," Internet Engineering Task Force, RFC 3588, Sep. 2003. [Online]. Available: http:www.rfc-editor.org/rfc/rfc3588.txt

[14] R. Housley and B. Aboba, "Guidance for Authentication, Authorization, and Accounting (AAA) Key Management," July 2007.

[15] C. Kaufman, P. Hoffman, Y. Nir, and P. Eronen, "Internet Key Exchange Protocol Version 2 (IKEv2)," RFC 5996 (Proposed Standard), IETF.