Wireless Handoff Optimization: A Comparison of IEEE 802.11r and HOKEY

Kashif Nizam Khan, Jinat Rehana

Wireless Handoff Optimization: A Comparison of IEEE 802.11r and HOKEY

Kashif Nizam Khan[1,2] and Jinat Rehana[1,2]

[1] Norwegian University of Science and Technology, NTNU
[2] Helsinki University of Technology, TKK
kashifni@stud.ntnu.no, rehana@stud.ntnu.no

Abstract. IEEE 802.11 or Wi-Fi has long been the most widely deployed technology for wireless broadband Internet access, yet it is increasingly facing competition from other technologies such as packet-switched cellular data. End user expectations and demands have grown towards a more mobile and agile network. At one end, users demand more and more mobility and on the other end, they expect a good QoS which is sufficient to meet the needs of VoIP and streaming video. However, as the 4G technologies start knocking at doors, 802.11 is being questioned for its mobility and QoS (Quality of Service). Unnecessary handoffs and re-authentication during handoffs result in higher latencies. Recent research shows that if the handoff latency is high, services like VoIP experience excessive jitter. Bulk of the handoff latency is caused by security mechanisms, such as the 4-way handshake and, in particular, EAP authentication to a remote authentication server. IEEE 802.11r and HandOver KEY (HOKEY) are protocol enhancements that have been introduced to mitigate these challenges and to manage fast and secure handoffs in a seamless manner. 802.11r extends the 802.11 base specification to support fast handoff in the MAC protocol. On the other hand, HOKEY is a suite of protocols standardized by IETF to support fast handoffs. This paper analyzes the applicability of 802.11r and HOKEY solutions to enable fast authentication and fast handoffs. It also presents an overview of the fast handoff solutions proposed in some recent research.

1 Introduction

The last decade in the communications industry has been dominated by mobile services. These services have changed the way we communicate daily. Users are now able to enjoy nonstop mobile services anywhere, anytime and anyplace. And for this reason, user expectations of the mobile services are higher than ever before. Seamless data connectivity, streaming video and Voice over Internet Protocol (VoIP) are now considered as the core applications that will drive the construction of next generation mobile networks [2].

If we now look at the wireless access technologies, we will find that IEEE 802.11 or Wi-Fi has been the prevailing option for wireless access to the internet. Over the years, 802.11 has ruled the wireless world with its ever growing mobility

and security amendments. However, as the options are changing, so are the demands. Recent research has pointed out that 802.11 suffers from a lack of QoS, specifically in case of the real time applications.

Handoff latency is considered as the main reason behind this technical problem. Current advancements in technology leave no space for latencies and unnecessary variation in delays (jitter) while a Mobile Node (MN) moves around the network and the connection jumps from one Access point (AP) to another. At present, the security protocols in wireless environments are designed in such a way that a MN needs to be authenticated at each AP when it moves around [2]. 802.11 essentially employs the EAP (Extensible Authentication Protocol) as a generic authentication protocol which supports multiple authentication methods [13].

Wireless handoff comprising APs is not a simple process, rather it includes a complex set of operations involving multiple layers of protocol execution. One crucial part of this handoff process is EAP re-authentication, which indeed is a major contributor to the overall handoff latency [13]. IEEE 802.11r [1] is an amendment to the original standard, which thankfully targets re-authentication as one of the key areas to lower the handoff latencies. It performs this by keeping the re-authentication procedure as local as possible, thus reducing the round trips. By contrast, Hand Over KEY (HOKEY) essentially targets the fact that in the existing networks, re-authentication to a new AP requires a full EAP exchange with the initial server which authenticated it previously. This phenomenon introduces extra processing delay and data transit delay in the network, which in turn increases the latency which can be 100 to 300 ms (milliseconds) per round trip [13]. The problem is that real-time applications like VoIP allow a maximum end-to-end delay of 50 ms. This goes as the basic motivation behind the HOKEY protocol. [11].

Both 802.11r and HOKEY target to solve the same problem and that is to achieve fast handoffs by reducing latencies. However, they perform modification at different layers of protocol stack. 802.11r modifies the MAC layer whereas HOKEY modifies the EAP layer. This is because, IEEE has standardized 802.11 MAC layer while IETF has standardized EAP. Interestingly, none of them tries to figure out what is the best way to reduce the handoff latencies.

Handoff latency has also attracted much research which focuses on optimized handoff procedures for 802.11. This paper presents an overview of the standards and proposals that have evolved to optimize 802.11 handoffs. Sec. 2 describes the 802.11 handoff procedure. Sec. 3 includes a brief description of 802.11r and HOKEY in terms of improvements and changes for fast handoffs. Sec. 4 summarizes a performance overview of 802.11r and HOKEY from the existing literature. Sec. 5 presents some research ideas that are not standardized yet [10, 4, 12, 9, 5, 8, 3, 6] for 802.11 handoff improvements by reducing the latency. Finally, Sec. 6 concludes the paper

2 802.11 Handoffs

This section briefly presents the key operations of 802.11 handoffs. During the mobility period, when a mobile node moves to the periphery of the current Access Point's (AP) coverage, it is necessary for the mobile node to attach itself to another AP with better coverage to continue the communication. This process of communication handover is simply known as handoff. It essentially incorporates a number of message exchanges among the mobile node, the old AP and the new AP which results in physical connection and state transfer from the old AP to the new AP [4]. During this period of handoff from the old AP to the new AP, the mobile station is potentially unable to communicate any type of data traffic. This delay is known as 'latency'. The higher the latency, the worse the QoS is experienced by the end user.

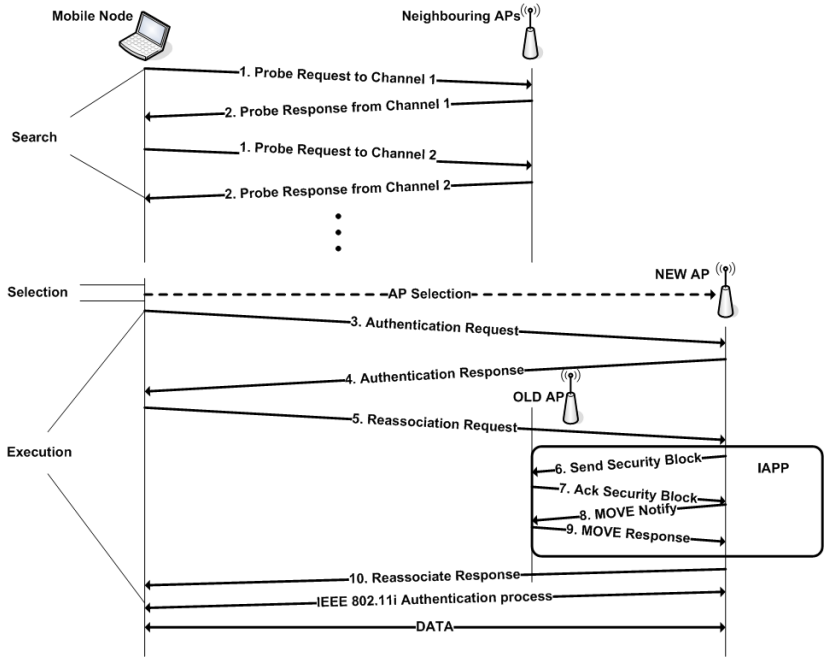

Fig. 1. 802.11 Handoff Example

The handoff process is performed in three basic stages: detection, search and execution [4]. As its name implies, 'detection' phase is the phase when the MN detects a need for handoff. It can be determined by different ways and it typically depends on specific network deployment. For example, the MN can decide to execute a handoff if it experiences a lower signal-to-noise ratio (SNR) or if it sees a number of packet losses consecutively. The term 'lower' in the former

case may be determined by comparing the SNR with a predefined threshold value. Nevertheless, the handoff can be initiated if a degraded performance is experienced by a MN from the current AP or the neighboring APs are offering better signals.

After the detection phase, the MN starts searching for the new AP among the candidate neighboring APs. In this phase, the MN scans through all 802.11 channels. Fig. 1 shows the search phase in which the MN sends a broadcast probe-request and waits for the probe-response from each of the channel. In the Fig. 1, these messages are numbered as 1 and 2 respectively. Upon receiving the responses, the MN chooses the best candidate AP for the handoff process. The selection is based on the SNR associated with each AP's probe response [4]. It is easy to assume that the SNR of the selected AP should be at least above the SNR of the current AP, to avoid unnecessary handoffs.

The execution phase is performed in two stages. At first, the new AP authenticates the MN using open authentication and then re-associates with the MN. Although, IEEE 802.11i extends the protocol to provide secure authentication. In Fig. 1, messages 3 and 4 show the simplest authentication handshake between the MN and the new AP. This authentication is followed by a re-association between the MN and the new AP. Re-association is performed to finalize the communication transfer. The re-association request includes important practical information such as the MAC address of the MN and the old AP and the Extended Service Set (ESS) identifier.

After the re-association the security state is transferred between the old AP and the new AP using the Internet Access Point Protocol (IAPP). This is a 2 way handshake as shown by the messages 6 and 7 in the Fig. 1. Once the security block is exchanged, the old AP needs to transfer all relevant information and communication states associated with the MN to the new AP. New AP issues a MOVE notify request to old AP asking for this information . The information exchange between old AP and new AP is encrypted and authenticated for security of the context information. Messages 8 and 9 in Fig. 1 show the two way MOVE handshake. Finally, the new AP answers the re-association request of the MN (Message 10) and the handoff process is finalized at this point. The MN can now transfer communication data through new AP.

IEEE 802.11i prescribes the use of EAP/IEEE 802.1x mechanism to overcome the vulnerabilities of snooping and intrusion by 3rd parties. It introduces new methods of key distribution to overcome the weaknesses in the earlier methods. A Pairwise Master Key(PMK) is generated between the MN and Authentication Server (AS) through earlier EAP exchange. The PMK is used to derive Pairwise Transient Key (PTK) via 4-way handshake between the MN and the AP. The 4-way messages are sent as Extensible Authentication Protocol Over LAN (EAPOL) frames. This 4-way handshake ensures the integrity of the MN and the AP and demonstrates that there is no man-in-the-middle. It also synchronizes the PTKs.

One of the major drawbacks of wireless handoffs is the slow transfer of connections from old AP to new AP while moving around. Even when a MN jumps

from the coverage of one AP to another in the same mobile domain, a connection needs to be established with the remote AS which may be distant in location. In addition, the number of roundtrips to the remote AS and the number of messaged exchanged between MN and AP also contribute to the delay because of roundtrips and channel acquisition time.

Preauthentication was introduced in 802.1X to mitigate these delays while providing secured authentication at the same time. Preauthentication performs this reduction in latency by caching some of the keying material derived during the authentication step in the neighboring potential new APs to which the MN may roam. In Pair-wise Master Key (PMK) caching, the associated AP and the MN perform the 802.1X authentication and cache the PMK hoping that the MN will associate with this AP in future. In such situations, only a 4-way handshake is needed between MN and AP to create new session keys from the PMK after re-association. The problem of choosing the correct PMK has been solved by using key identifiers in the reassociation request. We will not cover the pre-authentication detail as support for 802.11 preauthentication has been dropped from the 802.11r draft. This is because, although pre-authentication lessens the latency, it is still not sufficient for real time services like voice calls.

3 New Standard Solutions for Fast Handoff

This section presents two of the most powerful protocols that emerged to support fast handoff in mobile WLANs: IEEE 802.11r and HOKEY. As stated earlier, 802.11r is an amendment to IEEE 802.11 and thus, it has been standardized by IEEE. HOKEY is standardized by IETF in the form of RFCs. The following subsections illustrate the basic functions of these two protocol suits in more detail.

3.1 IEEE 802.11r

IEEE 802.11 experienced a major glitch with the specification of Wired Equivalence Protocol (WEP). The level of security provided by WEP was significantly deficient for Wireless applications. IEEE 802.11e is introduced in the process to provide better and enhanced security for real time applications (especially voice). IEEE 802.11i also emerged to mitigate the shortcomings of WEP and to provide enterprise level security. IEEE 802.11i adds stronger encryption and authentication methods for higher data security in Wireless Applications. However, these security amendments achieve stronger security at the expense of large delays which are unacceptable for applications like voice. Mechanisms like re-authentication and QoS re-negotiations during roaming introduce unnecessary and unacceptable large delays resulting in degraded voice quality. As a result, a new IEEE Task group, Task Group r, was formed to address these roaming issues in the 802.11 enhanced security amendments. This group has produced a new Fast Basic Service Set (BSS) Transition standard which is able to minimize

unnecessary delays during roaming while still providing the same amount of security as promised by IEEE 802.11i and 802.11e. Let's look at this 802.11r in a bit more detail.

802.11r performs the fast BSS transition in three major ways: integrating the four-way handshake into the 802.11 authentication/association exchange, pre-allocation of QoS resources and most importantly through efficient key distribution. 802.11r also introduces some new fields in the protocol messages such as: Mobility Domain Information Element (MDIE) and Fast Transition Information Element (FTIE). As the name implies, MDIE is concerned with current mobility domain information and FTIE manages resource reservation and security policy information. FTIE also includes some of the EAPOL-Key messages.

The method in which 802.11r reduces the 802.11i BSS transition time is by piggy-backing the EAPOL-Key messages at the four-way key exchange on top of four existing frames. It is performed by adding security-related information to 802.11 authentication and association request and response. For example, the MDIE and FTIE are contained in beacons, probe responses, association requests and association responses. This piggybacking significantly reduces the overall handoff latency, as PTK derivation step can be overlaid on open authentication and re-association steps. It essentially omits the extra round trips frames for PTK derivation. Details about PTK and 802.11r key hierarchy are discussed in Sec. 3.1.1.

The other optimization technique used by 802.11r is pre-reservation. In this procedure, the MN is allowed to perform the QoS admission control with the new AP before open authentication or re-association. It can be performed in two ways: Over the Distribution Service (OTD) and Over the Air (OTA) [7]. With OTD, the MN communicates with the new target AP via the currently associated AP's distribution services. OTD traffic flow is similar to 802.11i pre-authentication traffic flow and OTD is preferred in 802.11r as it provides pre-reservation capabilities without the interruption of the current traffic flow. QoS provisioning is an additional mechanism used in 802.11r standard to allow fast BSS transitions. In this approach, the pre-reservation is delayed until the association- request/response. It is sometimes appropriate when MN detects that target AP is lightly loaded and the reservation fail [7]. OTA also helps to reduce the latency of full BSS transition if QoS provisioning is included in the delay measurement.

The main aim of 802.11r fast BSS transition is to reduce the security overhead. The most obvious benefit is that bulk of the authentication process is performed before the actual handoff occurs. Once a MN associates itself with an AP residing in a particular subnet, the PMK can be distributed to all the APs that are associated with the subnet or rather we should call it the mobility domain. Hence, when a MN moves across the mobility domain, the PMK is assumed to be present in all the APs and so the time needed to reauthenticate the MN is significantly reduced. This is simply because the latency to communicate with the remote AS is omitted in this case. This prederivation of keys is sup-

ported by a new key management system in 802.11r. The following subsection discusses the new key hierarchy in more detail.

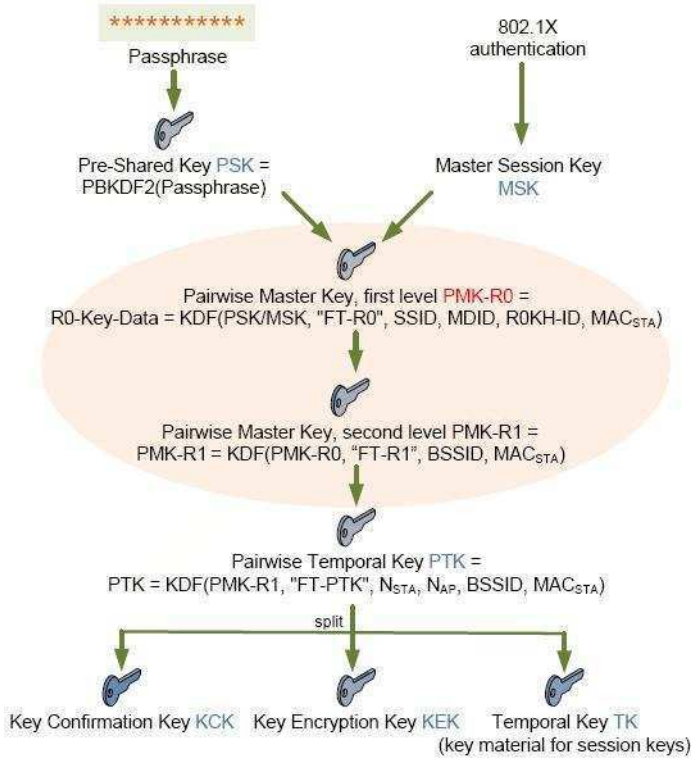

Fig. 2. IEEE 802.11r Key Hierarchy [16]

3.1.1 IEEE 802.11r Key Hierarchy Fig. 2 illustrates the 802.11r key management hierarchy. 802.11r key hierarchy consists of two levels of key holders. When a MN performs initial association and full authentication with an AP, the logical entity R0 Key Holder (R0-KH) derives the top level keys PMK-R0, which is in turn derived from the Master Session Key (MSK) and Pre-Shared Key (PSK) from the AS [1]. There is one R0-KH in each security mobility domain and one PMK-R0 per MN in each security mobility domain. The next level of keys, denoted as PMK-R1 is derived for associating the MN with the new AP and it will be different for different AP-MN pair. The AP also stores the PTK (Pair wise Temporal Key). The Key holders, which were previously known as wireless switches, may be located within an AP or may be a separate physical device.

802.11r key management system specifies two key domains: Security Domain (SD) and the Security Mobility Domain (SMD)[1]. SD comprises of all the entities shown in the figure namely R0-KH, all associated R1-KH and APs. SMD is a set of SD, in which a R0-KH can derive PMK-R1 for any R1KH. SMD essentially sets the boundary within which a MN is allowed to perform fast BSS transitions.

As shown in the Fig. 2, R0-KHID and R1-KHID are derived initially when a MN is associated with an AP residing in a SMD, for the first time. Then PMK-R1 keys are generated using the R0-KHID and R1-KHID. At this moment, the R0-KH is able to distribute the PMK-R1 to all other R1-KH [1]. Alternatively, it may distribute the keys on demand. Now, if a MN moves to another AP in the SMD, the R1-KH associated with the new AP already possesses PMK-R1 and so no IEEE 802.1X authentication is necessary. One interesting thing to notice here is that 802.11r does not specify any protocol for key distribution amongst the key holders and the APs. However, essentially it is assumed that a secure connection exists between APs and key holders.

3.2 HOKEY

IETF is on the way of standardizing another group of effective handover keying protocols known as HOKEY. The aim of HOKEY is twofold- firstly, support handovers from AP to AP and secondly support roaming between different operators. HOKEY has enhanced the EAP protocol to achieve low latency handoffs and method-independent fast re-authentication. To match the terminology of IETF EAP specifications, this section will denote the MN as 'peer'.

The EAP keying hierarchy requires two keys to be derived by all key generating EAP methods: the Master Session Key (MSK) and the Extended MSK (EMSK). In common scenarios, an EAP peer and an EAP server authenticate each other through an EAP authenticator. Successful authentication results in a derivation of a Transient Session Key (TSK) by the EAP peer using the MSK [15]. To avoid unnecessary delays and roundtrips, it is desirable to avoid full EAP authentication when a peer moves from one authenticator to another. Although some EAP methods utilize state information from initial authentication to optimize the re-authentication, most of the method specific re-authentications cost 2 round trips at minimum, with the original EAP server [13].

EAP Re-authentication Protocol (ERP) is designed to provide method independent re-authentications with lower handover latencies. The main idea behind ERP is to permit a peer and the server to verify the possession of keying material which has been previously obtained from an EAP method [15]. And more importantly, this EAP is a single-round trip exchange between peer and server. EAP exchange is independent of lower layer.

The design of EAP is pretty simple. A full EAP exchange is performed whenever a peer tries to connect to any network for the first time. As a result, MSK is available to the EAP authenticator at this point and the peer and server also derive EMSK. EMSK or DSRK (Domain Specific Root key) is used to derive re-authentication Root Key (rRK) which is available both at the peer and the

server[15]. Furthermore, an additional key: re-authentication Integrity Key (rIK) is derived from rRK which is used to prove the possession of keying material during ERP exchange.

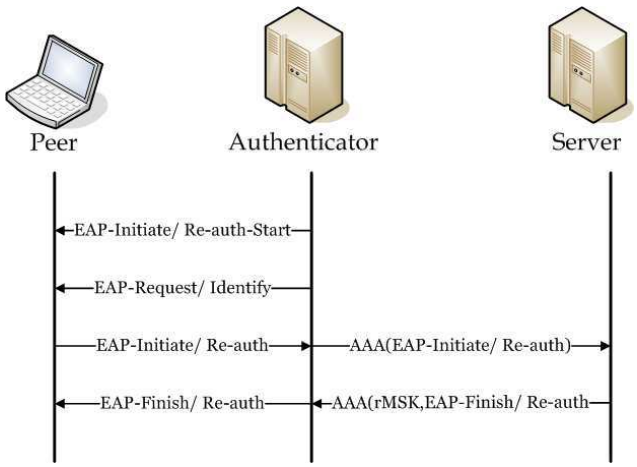

Fig. 3. ERP Exchange

Fig. 3 shows a typical ERP exchange. As the authenticator cannot be sure whether the peer is able to perform ERP beforehand, at first it sends an EAP-Initiate/Re-auth-Start message. It also serves the purpose of advertising the capability of the authenticator to support ERP. If the peer is capable of performing ERP exchange, it replies back with an EAP-Initiate/Re-auth message. If the peer does not know anything about the EAP-Initiate/Re-auth-Start message, it will not respond to it. So, after a certain period, of time the authenticator will initiate EAP by sending EAP-Request/Identity message. Similarly, an authenticator can also fall back to EAP if it does not support ERP.

For the sake of discussion, we assume here that the peer supports ERP and responds with an EAP-Initiate/Re-auth message. This message contains two important pieces of information: keyName-NAI (Network Access identifier) and rIK. The authenticator uses the keyName-NAI field to send the message to the appropriate server. Server uses the keyName-NAI to look up the rIK [15]. After successful verification of rIK, server derives rMSK (re-authentication MSK) from the rRK and a sequence number supplied by the peer with the previous response. This rMSK is then transported along with the EAP-Finish/Re-auth message by the server. rIK is used to protect the integrity of all these messages. Upon receiving the response, the peer verifies the integrity of the message using rIK and generates rMSK.

The above discussion also gives a picture of the key hierarchy used in ERP. At each re-authentication an rMSK is established between a peer and the au-

thenticator which serves the same purpose of MSK. To prove the possession of rRK, rIK is used which is derived from rRK. rRK is derived from EMSK or DSRK. While using ERP, HOKEY is also able to support roaming [6]. Roaming can be performed when the new network has a roaming relationship with MN's home network. One important thing to notice here is that, HOKEY makes use of the EMSK rather than reusing any existing key materials of 802.11i keying hierarchy.

4 Performance of 802.11r and HOKEY

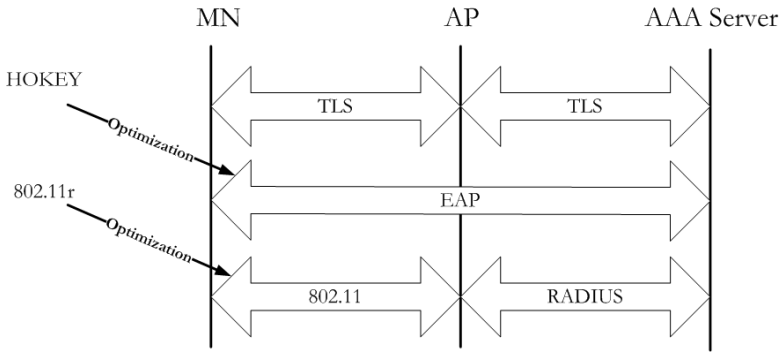

Fig. 4. HOKEY and IEEE 802.11r Optimization

802.11r and HOKEY both target more or less the same goal: keeping the authentication during the handoff local to the access network and thus reducing the handoff latency. However, they operate at different layers of the WiFi protocol stack. As depicted in Fig. 4, 802.11r tries to optimize the 802.11 protocols at the MAC layer whereas, HOKEY tries to optimize EAP which operates at link layer. HOKEY extends the AAA (Authentication, Authorization and Accounting) architecture to distribute security credentials without performing a full EAP authentication. By contrast, 802.11r focuses on passing security credentials directly between APs as a MN transitions from one AP to other.

Both 802.11r and HOKEY change 802.11i key hierarchy in different ways as depicted in Fig. 5. As a result, the relative complexity of the hierarchies also varies. 802.11r hierarchy does not support domain-level keys but HOKEY supports domain-level keys and as such HOKEY supports inter-operator roaming. Traffic keys are generated via the original four-way handshake in HOKEY and via a modified MAC protocol in 802.11r [6].

Key hierarchies play a crucial role in overall system security. In HOKEY, keys are essentially distributed along the AAA hierarchy. So if a AAA entity is compromised, all keys underneath will be compromised. In 802.11r, keys are

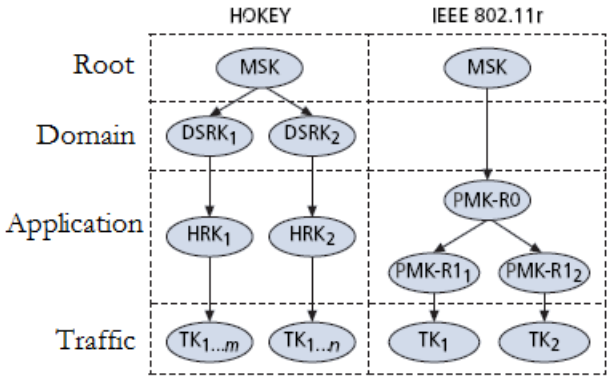

Fig. 5. Key Hierarchies for HOKEY and IEEE 802.11r [6]

handled centrally in R0KH. As discussed earlier, R0KH can be an AP or a separate physical device. In [6] it is argued that, R0KH will be a simpler target for key compromise than an AAA entity. The reason behind this can be that 802.11r centralizes all the keys to the edge of the network. So, 802.11r can be considered as weaker in security model than HOKEY.

[6] also presents a nice comparison of the handover performance of HOKEY and 802.11r. Let,

N = Number of roundtrips required to perform a particular EAP method

Tw = Transmission latency between the MN and AP

Tc = Latency between two close devices in wired LAN such as AP to AP communication and

Ta = Latency between various components and AAA server.

Now, let's look back to initial 802.11i handover: We have already mentioned that most of the EAP method specific authentications cost 2 round trips at minimum, with the original EAP server. So, the time to complete the EAP authentication should be 2N(Tw + Ta). Then, it will take Ta time to distribute the MSK to the AP and finally the four way handshake between MN and AP will take 4Tw time. So, the total 802.11i handover time sums up to 2N(Tw + Ta) + Ta +4Tw.

HOKEY handover time includes the single roundtrip execution of ERP which automatically delivers Handover Root Key (HRK) . Thus HOKEY omits the time to distribute MSK to the AP from 802.11i initial handover. So, for HOKEY the handover time will be 2(Tw + Ta) + 4Tw. 802.11r handover comprises of handover request from MN to AP, key distribution to the new AP and final key handshake at the new AP. It sums up to 2Ta + 2Tc + 2Tw.

The typical values for Tw ,Ta and Tc obtained from several network deployments are: Tw =15μs,Ta=20μs and Tc=5μs [6]. So, HOKEY handover time results in 130μs and IEEE 802.11r handover time results in 70μs. These han-

dover times are far less than a full IEEE 802.11i authentication using an EAP method. From the figures, we can easily compare that 802.11r requires less time for an intra-domain handoff than HOKEY. However, only HOKEY improves the performance of cross-domain handovers.

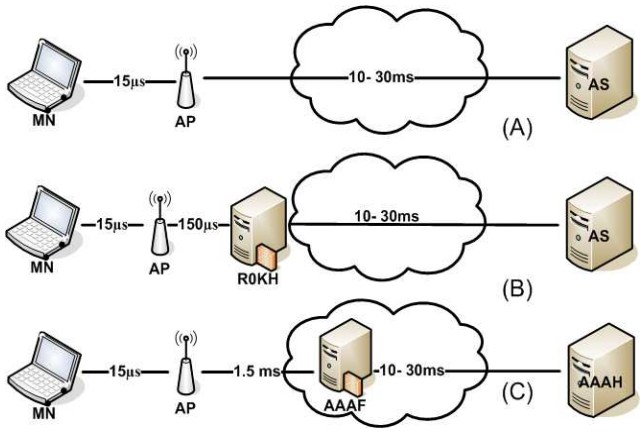

Fig. 6. Experiment of Handover Times in Different Cases

The analysis results found in [6] seems optimistic. To get a rough idea about the handover performance of 802.11r and HOKEY, we performed a simple experiment. As shown in the Fig. 6 (A), in case of 802.11i, the AP obtains the keying materials from the AS over the internet and the rough time is 10-30 ms depending on the location of the AS. In case of 802.11r , the keying materials are fetched and stored in R0KH which is located within a host in the same domain or LAN(Fig. 6 (B)). In case of HOKEY, the location of the proxy server is not mentioned explicitly but it is some where in the inernet but close to the AP (Fig. 6 (C)). We denoted the servers as AAA Home (AAAH) and AAA Foreign (AAAF) servers in the figure. We collected ping time from a host in Computer Science and Engineering Department, TKK to www.helsinki.fi, to get an idea about the 802.11i handovers. The average round trip time was 9.164 ms. For HOKEY, we pinged www.cs.hut.fi which was close to our department and the average round trip time was 1.625 ms. Lastly, to get an idea about 802.11r handovers, we pinged a host on the same LAN and it took .157 ms on an average. So, we can get a rough idea about the optimization of 802.11r and HOKEY from this simple experiment.

HOKEY and 802.11r both offer roughly same performance. However, their infrastructure requirements are different and as a result usage scenarios are also different. HOKEY will be more effective for service provider networks whereas IEEE 802.11r suits well for WLAN environments having low latency constraints. Both the standards solve the problem of handover latency but they implement

it in different ways and at different protocol layers. Choice of either one will strictly depend on particular network specifications and other constraints.

5 Other Proposed Improvements

Much research efforts have been devoted towards reducing the authentication delay and thus reducing latency during handoffs in wireless network. This section tries to present some of the better ideas from those research which can be a good contribution towards reducing the handoff latency in wireless network.

Tuomas Aura et. al. have proposed a new protocol for re-authentication of a MN to different APs or different wireless networks in [2]. This protocol passes credential information to a MN which has recently been attached in a successful association with an AP. The MN uses these credentials as a proof of its previous good behavior and when the MN tries to attach with a new AP it presents these credentials to it. One interesting property of this protocol is that the computations are mainly based on keyed one-way functions. So, it can result in very low computation overhead. Although, this approach takes the risk of passing credentials of one MN to other APs, it has the potential to result in a good solution with low complexity computations.

In [10] the authors have performed an explorative analysis to find the reasons of unnecessary handoffs. They collected data from both 802.11a and 802.11g networks over a period of five days. Their analysis reveals interesting results. They showed that most of the handoffs are triggered based on packet-loss information and this packet-loss information can be adverse in densely populated networks. Although, we are not always dealing with densely populated networks, but even then, handoff mechanisms should be adaptive to congestion losses.

Minho shin et.al. [11] have focused on reducing the probing time by reducing the number of probed channels. They have used neighbor graphs to improve the AP discovery process. However, this approach scales poorly as it requires each AP to store its neighbor graphs. [12] introduces a new selected scanning algorithm combined with caching mechanism. They have used a dynamic channel mask to reduce the probing to a subset of channels rather than all the channels. Finally, [14] shows that handoff latency can be significantly reduced by using shorter beacon intervals and active scanning. All of the above mentioned ideas can be very useful. However, many of them have been implemented and tested in constrained environment. It will be interesting to see how they perform in real networks.

6 Conclusion

Handoff latency is a critical mobility issue for real-time applications such as VoIP over WLAN. It is very important to maintain an acceptable QoS during the handoff process while also supporting advanced security standards. This paper presents a brief overview of the two effective suites of protocols which have been standardized to reduce the handoff latency during wireless handover:

IEEE 802.11r and HOKEY. Although both of the standards have evolved to solve more or less the same problem, they optimize different parts of authentication properties and use cases. As a result, they operate differently. However, as far as effectiveness is concerned, both of the protocols promise to lessen handoff latency while maintaining the same security level as that of IEEE 802.11i. This paper also discusses some other potential research proposals that can achieve better results if those proposals are combined with 802.11r or HOKEY. To conclude, we hope that, HOKEY and 802.11r will be a potential and significant step forward towards efficient wireless networks for real-time applications.

References

1. I. S. 802.11r TM. IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirement. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification. Amendment 3: Specifications for Operation in Additional Regulatory Domains , July 2008.
2. T. Aura and M. Roe. Reducing Reauthentication Delay in Wireless Networks. In *SECURECOMM '05: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks*, pages 139–148, Washington, DC, USA, 2005. IEEE Computer Society.
3. S. Bangolae, C. Bell, and E. Qi. Performance Study of Fast BSS Transition using IEEE 802.11r. In *IWCMC '06: Proceedings of the 2006 international conference on Wireless communications and mobile computing*, pages 737–742, New York, NY, USA, 2006. ACM.
4. M. S. Bargh, R. J. Hulsebosch, E. H. Eertink, A. Prasad, H. Wang, and P. Schoo. Fast Authentication Methods for Handovers between IEEE 802.11 Wireless Lans. In *WMASH '04: Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots*, pages 51–60, New York, NY, USA, 2004. ACM.
5. G. Bianchi. Performance Analysis of the IEEE 802.11 Distributed Coordination Function. *Selected Areas in Communications, IEEE Journal on*, 18(3):535–547, Mar 2000.
6. T. Clancy. Secure Handover in Enterprise WLANs: CAPWAP, HOKEY, and IEEE 802.11r. *Wireless Communications, IEEE*, 15(5):80–85, October 2008.
7. P. Goransson and R. Greenlaw. *Secure Roaming In 802.11 Networks*. Newnes, 2007.
8. V. J.Salowey, M.Nakhjiri and L. Dondeti. Specification for the Derivation of Root Keys from an Extended Master Session Key(EMSK). RFC 5295, The Internet Engineering Task Force, August 2008. http://ietf.org/rfc/rfc5295.txt.
9. S. Pack and Y. Choi. Fast Handoff Scheme Based on Mobility Prediction in Public Wireless LAN Systems. *Communications, IEE Proceedings-*, 151(5):489–495, Oct. 2004.
10. R. Raghavendra, E. M. Belding, K. Papagiannaki, and K. C. Almeroth. Understanding Handoffs in Large IEEE 802.11 Wireless Networks. In *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 333–338, New York, NY, USA, 2007. ACM.

11. M. Shin, A. Mishra, and W. A. Arbaugh. Improving the Latency of 802.11 Handoffs using Neighbor Graphs. In *MobiSys '04: Proceedings of the 2nd international conference on Mobile systems, applications, and services*, pages 70–83, New York, NY, USA, 2004. ACM.
12. S. Shin, A. G. Forte, A. S. Rawat, and H. Schulzrinne. Reducing Mac Layer Handoff Latency in IEEE 802.11 Wireless LANs. In *MobiWac '04: Proceedings of the second international workshop on Mobility management & wireless access protocols*, pages 19–26, New York, NY, USA, 2004. ACM.
13. T.Clancy, M.Nakhjiri, V.Narayanan and L. Dondeti. Handover Key Management and Re-Authentication Problem Statement. RFC 5169, The Internet Engineering Task Force, March 2008. http://ietf.org/rfc/rfc5169.txt.
14. H. Velayos and G. Karlsson. Techniques to Reduce the IEEE 802.11b Handoff Time. volume 7, pages 3844–3848 Vol.7, June 2004.
15. V.Narayanan and L. Dondeti. EAP Extensions for EAP Re-authentication Protocol(ERP)). RFC 5296, The Internet Engineering Task Force, August 2008. http://ietf.org/rfc/rfc5296.txt.
16. T. Aura. Lecture Notes on Network Security: WLAN Security. Helsinki University of Technology (TKK) & University College London (UCL), 2008.