SAP: Seamless Authentication Protocol for Vertical Handoff in Heterogeneous Wireless Networks

Scott C.-H. Huang[1], Hao Zhu[2], and Wensheng Zhang[3]

[1] City University of Hong Kong, Email: shuang@cityu.edu.hk
[2] Florida International University, Email: hao.zhu@fiu.edu
[3] Iowa State University, Email: wzhang@cs.iastate.edu

Abstract

802.11 standards support high data rates for a low price and thus provides an economical way for WLANs. On the other hand, 3G standards offer a much wider area of coverage that enables ubiquitous connectivity. The integration of them takes advantages from both sides and offers the possibility of achieving anywhere, anytime cost-efficient Internet access. To facilitate such integration, seamless vertical handoff is one of the major challenges because it needs to make physical movement transparent to mobile users and preserves application-level connectivity. Previous works did not consider the impact of authentication mechanisms on the performance of vertical handoff, especially on its delay. In a 3G-WLAN integration environment, since 3G and WLAN may use different authentication servers, when a mobile terminal hands over across them, certain authentication procedure needs to be performed. According to the literature, such authentication delay may be as high as hundreds of milliseconds, which is intolerable for delay-sensitive applications. We present seamless authentication protocols (SAPs) for vertical handoff in wireless heterogeneous networks, to reduce this delay. Simulation results show that SAP significantly reduces the delay caused by authentication procedures in vertical handoff.

1 Introduction

Technological development of wireless networks have brought a deep change in our lifestyle. In addition to traditional voice services, many data services (e.g. WWW, IP multimedia) have been carried over the wireless terrestrial networks as a last-mile access of today's Internet. The wide deployment of wireless infrastructures facilitates the accessibility to the IP-based data and therefore moves one step forward toward making it available anywhere anytime.

The 802.11 standards allow the realization of economic Wireless LANs that support data rates anywhere from 1Mbps to 54 Mbps based on the distance to the access point. However, 802.11 access points can cover areas of only a few thousand square meters, making them suitable for enterprise networks and public hot-spots such as hotels, coffee saps, and airports. On the contrary, wireless cellular networks, using the 3G standards, offer a much wider area of coverage that enables ubiquitous connectivity. However, 3G cellular networks require significant capital investments and support limited peak rates that range from 64 Kbps to nearly 2Mbps. The two techniques offer characteristics that complement each other perfectly. Thus, the combination of 3G and WLAN techniques offers the possibility of achieving anywhere, anytime cost-efficient Internet access, bringing benefits to both end users and service providers.

In such heterogeneous wireless systems, one of the major challenges is seamless vertical handoff. Seamless handoff is involved in the availability of the mobile terminal to successively attach to different access points or base stations in heterogeneous wireless networks. There have been several works on vertical handoff in the literature. In [22], a roaming scheme that considered the relative bandwidth of WLAN and GPRS was proposed. In [18] a detailed vertical handoff signaling procedure was presented. [25] proposed a mobility management system that integrates a connection manager to maintain a connection without additional network infrastructure support. [14] provided a quantitative analysis of a mobile IPv4-based WLAN-GPRS handoff prototype and identified a number of side effects related to the link layer and routing mechanisms. However, these works did not consider the impact of authentication mechanisms on the performance of vertical handoff, especially on its delay. Wherever a vertical handoff takes place, for the purpose of security, an authentication process should be performed to verify the identity of the mobile terminal. Since different access networks may use different authentication servers and protocols, when a mobile terminal hands over from one access network to another (e.g. from 3G to WLAN), an authentication procedure (e.g. 802.11i [6]) needs to be performed. As shown in [9], the delay of such procedure may go up to hundreds of milliseconds, which is intolerable for delay-sensitive applications such as voice-over-IP (VoIP) or streaming applications.

In this paper, we focus on designing seamless authentication protocols and schemes for vertical handoff in heterogeneous wireless networks. Similar to [4, 5, 8, 10, 21], our design is based on the following principle: the trust established in the previous access network can be used by the target access network to verify the legitimacy of the mobile terminal. The basic ideas are as follows. Without loss of autonomy, we have the two different authentication servers share a common secret, which is to be used later to generate a temporary handoff key. During a vertical handoff (e.g. the mobile terminal switches from 3G to WLAN), the target access network uses this temporary handoff key obtained from its authentication server to admit the mobile terminal for a short while (e.g. up to 10 seconds). Meanwhile, the full authentication is being performed simultaneously. Based on the result of the full authentication, the access point decides whether or not to permanently admit the mobile terminal. Since authentication servers take no part in the temporary authentication, the delay of temporary authentication is significantly less than that of full authentication. As a result, the impact of full authentication on vertical handoff delay is greatly mitigated. On the other hand, since full authentication is being performed on the background, the security loophole of temporary authentication is quite limited. Following these ideas, we design three fast authentication protocols, SAP v1,v2,v3, and their corresponding key management schemes. In particular, SAP v1 is the most secure while v2 has the least key management overhead. SAP v3 is designed to balance the advantage and disadvantage of SAP v1 and v2. SAP v1, v2 and v3 form a complete spectrum of tradeoffs between security and performance. We also provide a detailed security analysis of the proposed protocols and compare their performance through extensive simulations. Simulation results show that, compared with full authentication, SAP significantly reduces authentication delay during vertical handoff.

The rest of this paper is organized as follows: Section 2 is regarding the background and motivation, on which our protocols are based. In Section 3 we present our protocols, namely the full authentication and SAP v1,v2,v3, in vertical handoff. In Section 4 we do security analysis. The results of performance evaluation are shown in Section 5, Section 6 lists related work, and Section 7 concludes our work.

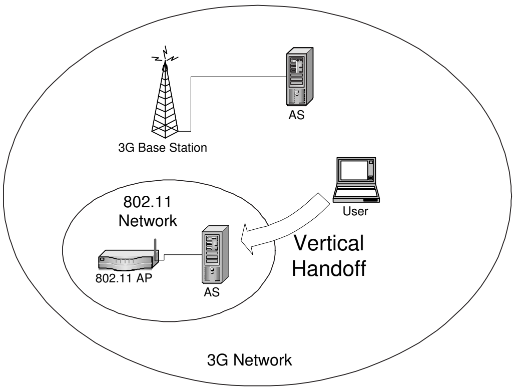


Figure 1. Vertical Handoff in Heterogeneous Wireless Networks

2 Background and Motivation

2.1 Background

In this section, we start with introducing the system model of our core scenario. Then we review the handoff procedures in both 3G and 802.11i networks with particular emphasis on security considerations.

2.1.1 System Model

As explained in [12], we consider the loosely-coupled interworking since its complexity and cost of deployment are lower than the tightly-coupled one. Specifically, 3G cellular networks and WLANs are integrated through the Internet, and there is no direct link from 802.11 elements (e.g. Access Points) to 3G network ones (e.g. Packet Data Serving Nodes or 3G core network switches). From a security point of view, this means different access networks connect different authentication servers (ASs) through the Internet. The users that access services of the 802.11 gateway include users having locally signed on or mobile users visiting from other networks. This model is quite light-weight, flexible, and practical [12]. A vertical handoff takes place when a mobile terminal switches between different access networks with different wireless techniques or service providers.

2.1.2 Authentication in 3G

3G authentication lies in the *Routing Area Update* process. Take *Intra-SGSN (Serving GPRS Support Node) routing area update* for example, the mobile terminal sends the routing area update request to the target RNC(Radio Network Controller), and the RNC forwards it to the target SGSN. The target SGSN then needs to authenticate the mobile terminal to determine whether or not the request can be accepted. *Inter-SGSN routing area update* has no exception either. Authentication needs to be performed when the routing area update request reaches the target SGSN.

The network access security specified in 3GPP has three building blocks: *authentication and key agreement (AKA), UMTS encryption algorithm (UEA), and UMTS integrity algorithm (UIA)*. We only focus on the authentication process of AKA. This process provides mutual authentication for both the users and the network. Two keys are generated in 3GPP AKA: CK for encryption and IK for integrity. There is a secret key K, shared by the user and the network and available only to the *authentication center* (AuC) in the user's *home environment* (HE) and the USIM (Universal mobile telecommunication system Subscriber Identity Module) on the mobile terminal.

Upon receiving an authentication request from a visitor location register(VLR) or SGSN, HE/HLR distributes to SGSN/VLR a set of authentication vectors (AVs), ordered based on the sequence number. Each AV is good for one AKA between VLR/SGSN and the USIM. To authenticate a user, VLR/SGSN retrieves the next available AV. Based on the secret material in the AV, the mobile terminal can authenticate the network. Then, the mobile terminal generates the response and sends it back. VLR/SGSN then authenticates the user by comparing the received response with the expected one.

2.1.3 Authentication in 802.11 WLANs

An AP and associated MTs form a *basic service set* (BSS). A collection of APs connected by a wired network can extend a BSS into an *extended service set* (ESS). If a MT wants to join an ESS network, it must be authenticated by showing its credentials to AP. AP then passes these credentials to a fixed *authentication server* (AS) to verify these credentials. Upon receipt of AS's decision, AP will either associate or reject the aspirant MT.

A typical authentication procedure in 802.11i usually involves EAP/TLS authentication and RADIUS back-end protocol. In fact, concrete EAP authentication methods and back-end protocols are beyond the scope of 802.11i, but EAP/TLS is the *de facto* authentication protocol and RADIUS is the *de facto* back-end transport protocol for EAP over IP networks. There will be certain latency involved in the whole handoff process, which can be divided into two phases:

Probe phase: The mobile terminal (MT) scans through all possible channels to find access points (APs) of good signal strength. It is called the passive scan. These beacon messages are usually sent periodically by APs at an interval of 10*ms*. Also, MT can actively send probe requests in the first place to get responses from APs. It is called active scan. The delay caused by scanning is called the *probe delay*, which is sometimes of magnitude 100*ms*. In our system, by letting 3G BSs share the channel assignment of each 802.11 AP, the probe delay can be significantly reduced.

Reassociation phase: After finding the preferred AP, MT tries to associate with it and performs *context transfer*. The new AP will first try to contact the old AP and get the *security context* (such as encryption key, etc.). This can be done only if there is a trust relation between these two APs. Specifically, if the two APs are within the same ESS, then the old AP can pass the entire security context to the new AP to reduce the authentication latency. Otherwise, it means the two APs have different ASs associated with them. The new AP should then ask for its AS to authenticate MT, and performs a full authentication. Considering the transmission delay between AP and AS plus the processing delay at AS, the authentication delay is usually of magnitude of several hundred milliseconds or even several seconds [9].

2.2 Motivation

Since 3G networks and WLANs are loosely integrated through the Internet and they may not share the same AS, this heterogeneity makes context caching [21] impractical. For instance, existing protocols such as IAPP [4, 5] or Seamoby [8] all assumed the homogeneity of the networks to perform Layer 2 context (e.g. MAC address of the old and new APs, encryption keys) transfer during a handoff. Other works such as Bargh *et al* [10] assumed the homogeneity of access networks to perform proactive key distribution. In the example of Figure 1, none of these protocols work due to the lack of a shared AS (Note that different ASs have different security contexts). Therefore, an full authentication cannot be avoided and the resulting delay becomes the bottleneck of the handoff process. For example, the typical latency of a full authentication in WLAN can be as high as 800*ms* (Arbaugh [9]), which is much greater than the maximal tolerable latency for delay-sensitive applications (e.g. the end-to-end delay of 150*ms* for VoIP).

Some people may think about making use of cellular networks to conceal the full authentication delay. That is, suppose WLANs are covered by a 3G cellular network and a mobile terminal hands over from 3G to a WLAN, the mobile terminal can keep the 3G connection until the WLAN authentication finishes. However, this idea has two severe drawbacks. First, it is not always guaranteed that each WLAN is covered by a cellular network[1]; Second, it is not applicable to vertical handoff between WLANs using different ASs. Therefore, it is necessary to find a sound solution

[1] This can happen at the edge of the 3G cell or in concrete buildings where 3G's signal strength is very weak.

for seamless authentication in integrated wireless networks. These observations motivate us to design seamless authentication protocols to reduce the impact of authentication delay on Quality of Service (QoS) when a mobile terminal roams (1) from 3G network to WLAN, (2) from WLAN to 3G, or (3) between WLANs using different ASs,

3 The Seamless Authentication Protocols

For brevity and clarity, we only describe the protocols as the solution for vertical handoff from 3G to WLAN. The solution can be easily extended and applied to other types of handoff between access networks associating with different ASs.

3.1 Sharing Secrets Between ASs

To facilitate fast authentication in vertical handoff, one approach is to have the new AP/BS utilize the security context of the mobile terminal from the old AP/BS. Since the SAP-related operations performed in an AP are the same as those in a BS, we use AP to stand for AP/BS in the rest of the paper. To preserve autonomy, we have different authentication servers share a "SAP master key" SAP_M instead of the whole credential database. This key is updated periodically with a relatively large interval (such as every 30 minutes), and then ASs distribute it to their APs/BSs. Upon receiving new SAP_M, each AP computes the SAP temporary key sap_i and distributes it to every mobile terminal it is currently associated with. sap_i is periodically updated by AP and then distributed to the mobile terminals. This update should be frequent enough (e.g. every several minutes) to discourage attackers and provide better security.

3.2 Notation and Key Hierarchy

We list the notations and key hierarchy here. The cryptographic keys involved are shown in Table 1. We use *pseudo-random functions* [15] to generate cryptographic keys. We use the following notation $h_i(X)$ to represent $F_X(i)$ for $0 \leq i \leq 6$, where F is a pseudo-random function constructed using techniques in [15]. For example, $h_3(X)$ means $F_X(3)$.

3.3 The Full Authentication Protocol

As the building block, we proposed the full authentication protocol, which is a 4-way handshake protocol applying the *challenge and response* principle. Similar to existing full authentication protocols [6, 16], the full authentication protocol provides mutual authentication and establishes the security context (PMK) between MTs and APs. The key hierarchy of the full authentication is shown in Figure 2. The description of the full authentication protocol is shown in Table 2.

abbreviation	full name & description
MK	master key, shared by MT and AS only
PMK	pairwise master key, shared by MT,AP,AS
SAP_M	SAP master key, shared by AS & AP
sap_i	SAP temporary key, shared by MT & AP
K_{MT}	SAP session key, shared by MT and AP
N_X^i	The i-th nonce generated by entity X
SAK	subgroup assignment key, shared by ASs and APs
$k_0^{AS}, k_1^{AS}, \ldots$	subgroup keys generated by ASs and shared by ASs and APs
$k_0^{MT}, k_1^{MT}, \ldots$	subgroup keys shared by APs and MTs

Table 1. Notation

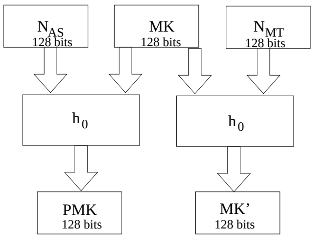


Figure 2. Key hierarchy of full authentication

Notation:"A ⟶ B: C" or "B ⟵ A: C " represents the sending of message C from A to B.
1. MT ⟶ AP ⟶ AS: $\langle ID_{MT}, N_{MT}^1, FAR\rangle$
//FAR stands for "Full Authentication Request"
2. MT ⟵ AP ⟵ AS: $\langle E_{MK'}(N_{MT}^1, N_{AS}^1), FAC\rangle$
//FAC stands for "Full Authentication Challenge"
//$MK' = h_0(MK \| N_{MT}^1)$ is computed at AS
3. MT computes MK', decrypts the message, gets N_{AS}^1, and computes $PMK = h_0(MK \| N_{AS}^1)$
4. MT ⟶ AP ⟶ AS:
$\langle ID_{MT}, N_{MT}^2, HMAC_{PMK}(ID_{MT} \| N_{AS}^1 \| N_{MT}^2), FAC\rangle$
//FAC stands for "Full Authentication Response"
5. AS verifies HMAC. If it succeeds, go to the next step. Otherwise, abort.
6. AP ⟵ AS: $\langle Auth\ Succ, PMK\rangle$
//The message is sent via a secure channel
7. MT ⟵ AP: $\langle Auth\ Succ\rangle$.
MT and AP change association state.

Table 2. The full authentication

As shown in the table, steps 3 and 5 guarantee the authenticity of AP and MT respectively, thus mutual authentication is provided. At step 3, MT computes PMK by applying h_0 to MK and N_{AS}^1. Also, at step 5, AS needs to verify HMAC so PMK will be computed too. Hence the security context PMK (shared pairwise key) is established between MT and AP. The actual encryption key used subsequently is not necessarily PMK. It is better to use its derived keys in order to avoid direct exposure of the shared secret and discourage cryptanalysis.

3.4 SAP v1, v2 and v3

With the full authentication, we proposed three seamless authentication protocols called SAP v1, v2 and v3 respectively. The main difference between these SAPs lies in the key distribution schemes. In particular, when the material of temporary handoff key is updated, which is similar to the procedure depicted in Figure 3 (a), SAP v1 lets APs distribute temporary handoff keys to MTs using unicast. As an opposite approach, APs in SAP v2 broadcast temporary handoff keys to each associated MT. Due to the space limitation, we do not give the details of SAP v1 and SAP v2, and use SAP v3 to represent our solution.

The demand of exclusively sharing authentication keys between the AS and every mobile terminal inherently requires the key update operation to be pair-wise. Such key updates, as in SAP v1, incur the communication cost that is linear to the number of mobile terminals, and thus limits the scalability of the system. Under this framework, some group-based key update approach can be adopted to reduce the communication overhead and thus improve scalability. SAP v2 is an extreme example of this approach since it allows the AS shares the same key with all mobile terminals in the service set as a group. Obviously, SAP v2 exposes security holes for internal attacks when an innocent mobile terminal is compromised. SAP v3 is a tradeoff between SAP v1 and SAP v2. SAP v1 being the most secure and SAP v2 being the most efficient, SAP v3 tries to balance between security and performance using a tunable randomness.

3.4.1 SAP v3 key management

Key management at AS As shown in Figure 3 (a), each AS generates a *Subgroup Assignment Key* SAK and applies an r-bit pseudo-random function h_3 to group MTs into 2^r subgroups (h_3 takes an arbitrary length input and outputs r bits). The subgroup ID, j, is generated by AP following:

$$j = h_3(ID_{MT} \| SAK)$$

In addition to SAP_M and SAK, each AS also generates a pair of subgroup keys K_j^{AS} and K_j^{MT} for each subgroup j, where $j = 0, \cdots, 2^r - 1$. Here, each K_j^{AS} is a secret shared between AS' and APs, while each K_j^{MT} is shared between APs and the MTs in group j. Note that each K_j^{MT} must have the same length as each K_j^{AS}, but they should be different. Otherwise, K_j^{AS} would be disclosed to the MTs in group j. The SAK, K_j^{AS} and K_j^{MT} are distrbuted to APs via secure wired channel, and will be updated periodically (e.g., every 30 minutes). Each mobile terminal is given a subgroup ID j when it first gets associated with an AP. Meanwhile, the full authentication is performed and, if successful, the AP preloads each MT in group j with an initial k_j^{MT}.

Key management at AP and MT Each AP uses a pseudo-random function h_4 to generate a nuance n_j for each subgroup j. h_4 is shared by APs and ASs, and outputs bit strings of the same length as that of k_j^{AS}. h_5, known to everybody, is the the pseudo-random function used by each MT to update its group key k_j^{MT}. The key management procedures are as follows:

- AP computes $n_j = h_4(SAP_M \| j \| k_j^{AS})$, $j = 0, \cdots, 2^r - 1$.
- MT $\longleftarrow$ AP: $\{E_{k_j^{MT}}(n_j, v(n_j)) | j = 0, \cdots, 2^r - 1\}$, where $v(n_j)$ is the version number of n_j
- Each MT decrypts his own piece of message with its k_j^{MT} by checking the correctness of $v(n_j)$. With n_j, it updates its group key k_j^{MT} by $k_j^{MT} \leftarrow h_5(k_j^{MT} \oplus n_j)$
- AP updates $\{k_j^{MT}\}$ by $k_j^{MT} \leftarrow h_5(k_j^{MT} \oplus n_j)$, $j = 0, \cdots, 2^r - 1$

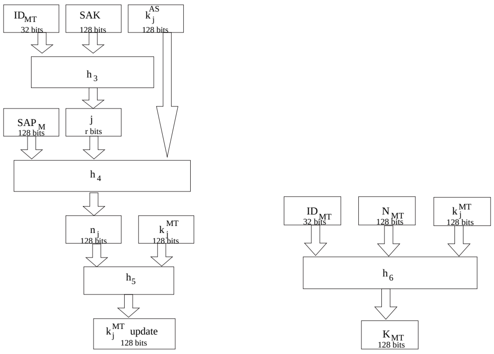
With respect to each subgroup, the AP broadcasts the encrypted message to every MT it is currently associated with. Since each MT only possesses its own subgroup key, it can only decrypt his own piece of message. Since APs have Internet connection, they can be synchronized with existing time synchronization protocols (e.g. Network Time Protocol [19]) and perform subgroup key updates in a synchronized way.

3.4.2 SAP v3 authentication

The protocol of SAP v3 is shown in Table 3. This quick 2-way SAP authentication (steps 1-5) temporarily associates MT with the new AP. In the meantime, AP also tunnels ID_{MT}, N_{MT}^1 to AS to perform full authentication (step 6). We only allow SAP association up to a predefined time (e.g. 10sec). If full authentication fails or SAP association expires, AP will simply de-associate with MT (step 7).

4 Security Analysis

Similar to the analysis in [13], we can prove that the full authentication protocol has the the following seven merits: (1) Robust method of proving identity that cannot be spoofed; (2) Method of preserving identity over subsequent transactions that cannot be transferred; (3) Mutual authentication; (4) Authentication keys that are independent of encryption keys; (5) Security against impersonating attack; (6) Security against cryptanalysis attack; and (7) Security



(a) Group key update (b) Key hierarchy

Figure 3. SAP v3 group key update and key hierarchy

against replay attack. Consequently, the system can be securely guarded with the full authentication protocol. We also analyzed the security of SAP v1, v2 and v3 in the following subsections.

1. MT $\longrightarrow$ AP:
$\langle ID_{MT}, N^1_{MT}, HMAC_{k_j^{MT}}(ID_{MT} \| N^1_{MT}), SAPReq\rangle$
//$SAPReq$ stands for “SAP Authentication Request”
2. AP computes its subgroup ID $\big($by plugging in $h_3(ID_{MT} \| SAK)\big)$ and locates its corresponding k_i^{MT}. AP then uses this k_i^{MT} to verify HMAC. If it succeeds, AP computes $K_{MT} = h_6(ID_{MT}, N^1_{MT}, k_j^{MT})$ and temporarily associates with MT using K_{MT}.
3. MT $\longleftarrow$ AP:
$\langle N^1_{AP}, HMAC_{K_{MT}}(N^1_{AP} \| ID_{MT}), SAPSucc\rangle$
//$SAPSucc$ stands for “SAP Authentication Success”
4. MT verifies HMAC and both MT and AP change association MTtus to SAP association.
5. Initiate the full authentication procedure.
6. If full authentication succeeds, MT and AP change association status to permanent association and use full authentication key PMK for encryption henceforth.
7. If full authentication fails or SAP association expires, AP de-associates with MT.

Table 3. SAP v3 authentication protocol

4.1 SAP Key Management and Authentication

4.1.1 Impersonating Attack

SAP v1 If an attacker impersonates an innocent MT, then it has to forge the following data: ID_{MT},N^1_{MT}, and $HMAC_{K_{MT}}(ID_{MT})$. However, the $HMAC$ key K_{MT} cannot be faked because it equals to $h_2(sap_{i,MT} \| N^1_{MT})$ and the attacker has no knowledge about $sap_{i,MT}$. On the other hand, if the attacker impersonates AP, it has to send the following data: N^1_{AP} and $HMAC_{K_{MT}}(N^1_{AP} \| ID_{MT})$. This cannot be faked, as the attacker cannot generate a nonce ID_{MT} along with its corresponding $HMAC$ without knowing the MAC key K_{MT}. Therefore, SAP v1 is robust against the impersonating attack.

SAP v2 The strength of SAP v1 is originated from the attacker’s lack of knowledge about $sap_{i,MT}$, but this comes at a price. Each AP has to unicast $sap_{i,MT}$ to every MT. This may cause significant communication overhead when the number of MTs is large. SAP v2 trades in the security on this aspect for more efficiency by broadcasting a shared root key sap_i to every MT in the service set. As a result, everybody can impersonate others (even the AP) as long as they are in the same service set. More specifically, Therefore, SAP v2 is vulnerable to the impersonating attack.

SAP v3 SAP v3 strikes a balance between SAP v1 and v2. Both the security level and the key management cost are between SAP v1 and v2. In SAP v3, the SAP root key has been broken into 2^r pieces $\{k_1, k_2, \ldots\}$, so the probability that an attacker can successfully impersonate a node or an AP equals $1/2^r$ and its overhead increases correspondingly too. If h_6 is secure enough, to impersonate a node or an AP, the attacker needs to be able to compute the SAP key K_{MT}, which is derived from the root key k_i. The security of SAP v3 thus solely depends on the knowledge of K_{MT} and the security of h_6, which can even be kept secret to increase security (though we didn’t assume that and it is not absolutely necessary.) We do not consider the case of launching a cryptanalysis attack on these hash functions because it is beyond the scope of our discussion. The probability that the attacker knows the same k_i as the victim node is $1/2^r$, so the probability of successfully impersonating a node is $1/2^r$ too.

The security of SAP v3 comes from the hiding of subgroup information. Hiding the subgroup assignment key SAK prevents every node from knowing its group ID. Also, the SAP root key k_i's will never be exposed directly in the messages, resulting in the inability of determining whether two nodes belong to the same group or not. These properties are carefully designed so as to discourage attackers. If these data were not hidden, an attacker can simply listens to the communication, wait for the node that belongs to the same group, and launch the impersonating (or other) attacks. In SAP v3, it is not easy. If the pseudorandom function h_3 is chosen properly and SAK is kept secret, the attacker cannot do anything other than guessing the grouping information. This helps discourage attackers while preserving most of the security.

4.1.2 Cryptanalysis attack

SAP v1 is robust against cryptanalysis attack. SAP v1's robustness is due to the use of HMAC and the secrecy of the MAC key K_{MT}. Since attacker has no knowledge about K_{MT}, the best attack he can launch is the direct *birthday attack* on the underlying hash function. However, if this underlying hash function is chosen carefully (such as MD5), the practical usefulness is negligible [11, 17]. Different from v1, SAP v2 is vulnerable to cryptanalysis attack since K_{MT} only depends on ID_{MT}, N^1_{MT}, sap_i, in which I_{MT} is public, N^1_{MT} can be generated, and sap_i is known to all MTs associated with the current AP. SAP v2 is vulnerable to the known plaintext attack because the adversary can record all communication messages between an MT of the same group and get many 3-tuples $ID_{MT}, N^1_{MT}, HMAC_{KMT}(ID_{MT})$ for cryptanalysis. On top of that, the adversary can even fake the communication message and record AP's response message $N^1_{AP}, HMAC_{K_{MT}}(N^1_{AP} \| ID_{MT})$ for cryptanalysis, too. However, the adversary cannot perform the chosen plaintext attack, as he has no control over the nonce N^1_{AP}, nor can he perform the replay attack, as the nonce is used throughout the scheme to provide freshness.

SAP v3 needs to be examined and analyzed in a probabilistic fashion. Similar to earlier discussion on impersonating attack, in SAP v3, the adversary has knowledge about K_{MT} with probability $1/2^r$. Therefore, with this probability the adversary can perform the known plaintext attack. SAP v3 is robust against chosen plaintext and replay attack for the same reasons as v2.

As a result, we conclude that SAP v1 is robust against cryptanalysis, v2 is vulnerable to known plaintext attack, and v3 is vulnerable to know plaintext attack with probability $1/2^r$. Between v1, v2, and v3, the least secure case is v2. However, the worst thing that may happen is the misuse of the short temporary association, which might not cause big problems since full authentication is still being processed. SAP v1, v2, v3 altogether provide a complete spectrum of tradeoffs between security and efficiency.

5 Simulation Results

In this section, we evaluate the performance of the full authentication and SAP v1, v2 and v3. Our simulation is based on ns-2 [1]. APs and BS have different authentication servers (ASs). Mobile users are assumed to have subscribed both WLAN and 3G services. We assume vertical handoff from 3G to WLAN follows a Poisson process, with a mean handoff interval. We assume the service latency of each authentication request is exponentially distributed with a mean service time. Each AP connects to AS through the Internet, and the average end-to-end bandwidth and delay of the link is 10 Mbps and 10 ms respectively. For simplicity, the channel capacity of each WLAN is assumed to be 2 Mbps, and all data packets in WLANs are served with 802.11 DCF [2].

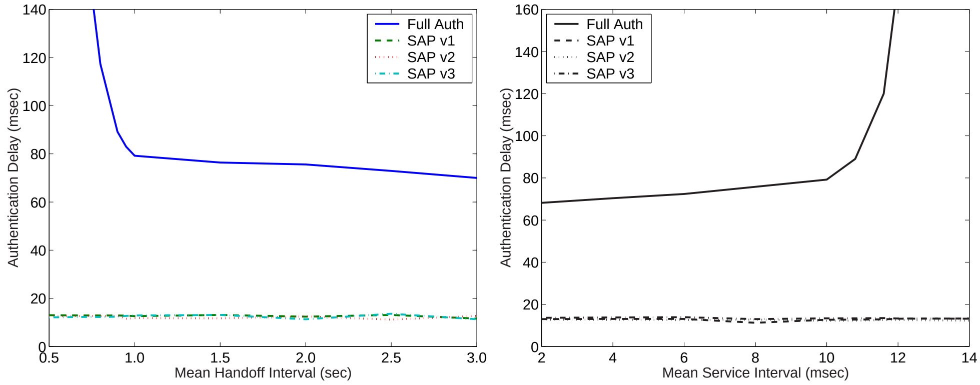
We first evaluate the performance of SAP by comparing it with the full authentication protocol. The performance matric is the authentication delay, which is equal to the time interval from AP sending the authentication request to the time the mobile terminal being admitted by AP. We also study the impact of mean handoff interval and mean service time. Then we evaluate the overhead of key distribution schemes used by SAP v1, v2 and v3.

5.1 The Authentication Delay

We compare the authentication delay of the full authentication and SAP v1, v2 and v3. We first fix the mean service time of AS to 10 ms and evaluate the authentication delay as the function of the mean handoff interval. We change the aggregated mean handoff interval[2] from 0.5 second to 3.0 seconds. As shown in Figure 4 (a), the authentication delay of the full authentication is much longer than that of SAP v1, v2 and v3. Specifically, when mean handoff interval is quite small (say less than 1.0 second), the authentication delay increases a lot. This long delay mainly comes from the queuing delay at AS due to limited processing capacity. When the mean handoff interval increases, the queuing delay at AS becomes almost zero. However, due to the service latency at AS plus the transmission and propagation delays between the APs and the BS, the authentication delay of full authentication is still much longer than that of SAP v1, v2, and v3.

According to Figure 4(a), compared to full authentication, SAP v1, v2 and v3 are not sensitive to mean handoff

[2] It is because there are multiple APs sharing the same AS.



(a)

(b)

Figure 4. The authentication delay of the full authentication and SAP v1, v2 and v3

interval at all. The reason is that each authentication is locally processed by APs, while the full authentication takes place on the background. Since SAP v1, v2 and v3 differ only in key distribution, we can see that their delays are very close to each other.

We then evaluate the delay of full authentication and SAP v1, v2 and v3 as a function of the mean service time of the AS. We fix mean handoff interval to 1.0 second. The mean service time is changed from 2 ms to 14 ms. The results are shown in Figure 4 (b). For full authentication, we can see that the delay increases rapidly if the mean service time is greater than 10 ms. This can also be explained by the queuing delay at AS. Similar to Figure 4 (a), the delays of SAP v1, v2 and v3 are not sensitive to the processing delay of AS. It is also because that AP does not need to contact with the AS to perform the temporary authentication.

5.2 The Overhead of Key Distribution

Compared to the full authentication, SAP v1, v2, and v3 require that each AP should periodically update the temporary handoff key, and distribute distribute the updated key to each associated mobile terminal. In this section, we evaluate the communication overhead of key distribution in SAP v1, v2 and v3. The overhead (in bps) is equal to the total amount of traffic (in bits) used for key distributions divided by the simulation time[3]. The key update interval is assumed to be 2 seconds. We evaluate the overhead as a function of the number of nodes. The overhead of SAP v3 is further studied as the function of the number of groups.

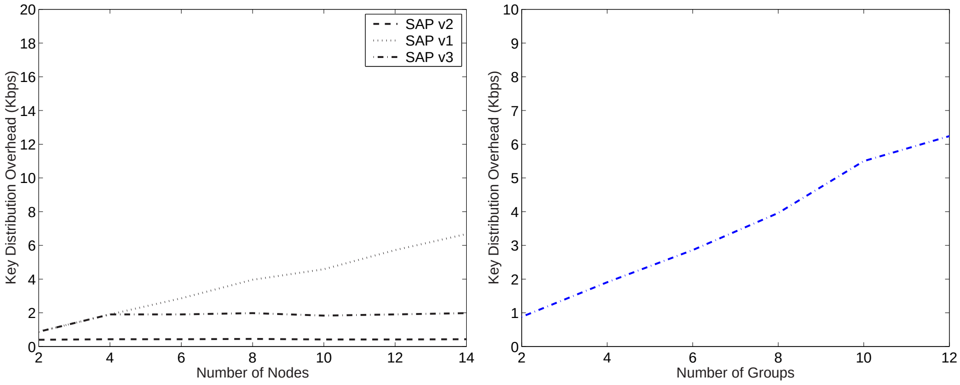
We first study the overhead under different number of nodes. For SAP v3, the number of group is fixed to be 4. As shown in Figure 5 (a), SAP v1 has the least overhead while the SAP v2 has the most. Since SAP v1 distributes this key to each mobile terminal one-by-one through unicast, the overhead increases as the number of mobile terminals increases. On the other hand, since SAP v2 distributes the temporary handoff key to each mobile terminal using broadcast, the overhead is minimized. However, as stated in Section 4, SAP v2 has the least security. Fortunately, since the size of each key distribution message is quite small (less than 30 bytes), the overhead of SAP v1 is not very large provided that the number of nodes is moderate. SAP v3 balances the tradeoff between key distribution overhead and security. As shown in Figure 5 (a), as the number of nodes increases, the overhead of SAP v3 is much smaller than that of SAP v1. In particular, as shown in Figure 5 (b), the overhead of SAP v3 is linearly proportional to the number of groups.

6 Related Work

In 802.11 Wireless LANs, 802.11i [6] defined the *Robust Security Network* (RSN), The authentication scheme of 802.11i was based on IEEE 802.1X [3], which employed *Extensible Authentication Protocol* (EAP), allowing different authentication mechanisms to establish layer 2 session key dynamically. IEEE 802.1X does not define the way that EAP messages are passed between the authenticator and the authentication server. Remote Authentication Dial-In User Service (RADIUS) [23] (RFC 2865 – 2869, RFC 3162, RFC 2548) is the most common back-end protocol. IEEE standardized in IEEE 802.11f [7]. It was designed for the enforcement of unique association throughout a ESS (Extended Service Set) and for secure exchange of station's security context between current access point(AP) and new AP during handoff period.

Stemm and Katz [24] defined the idea of vertical handoff. Mishra *et al* [20] analyzed the handoff process at the link layer and found that the factors that influence the hand-

[3] Note that the amount of traffic includes MAC layer frames (e.g. RTS, CTS, ACK...[2]



(a)

(b)

Figure 5. The overhead of key distribution in SAP v1, v2 and v3

off latency are probe, authentication, and reassociation delays. They showed that the probe delay is the primary contributor to the overall handoff latency and it is significant enough to affect the quality of service for many applications. Arbaugh *et al* [9] did empirical studies on handoff latencies, supporting our motivation of reducing the reassociation delay in a vertical handoff. They also used neighbor graphs and proactive caching algorithms to reduce the reassociation delay.

The IETF Seamoby group [8] defined different protocols for seamless IP-level handoff by reducing network discovery and reconfiguration delays. Compared with our scheme, they all assumed a *shared AS* between different networks. Bargh *et al* [10] found that IAPP and Seamoby results are not directly applicable for inter-domain seamless mobility and extending them for inter-domain mobility requires enhancements to the security infrastructure.

7 Conclusion

We presented three protocols, SAP v1,v2,v3, in conjunction with the full authentication protocol to deal with vertical handoff across heterogeneous wireless networks integrated in the fashion of loosely-coupled interworking. Our protocols were designed to avoid performing full authentication prior to the handoff. Instead, we used a temporary handoff key (which are K_{MT} in v1,v2, and k_j in v3) to temporarily associate MT with AP while performing the full authentication simultaneously. This facilitates the seamless handoff process because the temporary handoff key bridges the gap caused by full authentication. Simulation results showed that our protocols are efficient and do not have high communication overhead. The security of SAP v1 originated from using different temporary keys for SAP handoff, but this causes more overhead of key management. The efficiency of SAP v2 originated from using a flat key for temporary handoff, but this causes security flaws. SAP v3 was designed by combining v1 and v2 in a randomized fashion, thus providing a complete spectrum of tradeoffs between security and efficiency. We believe that, with the parameter r properly chosen, SAP v3 can provide good security for this temporary handoff process (say 10 seconds), and after that the full authentication results can take over. The randomization was designed in such a way to discourage attackers even for this short period of time. For this we strongly believe that SAP v3 is efficient, secure, and practical in providing seamless authentication for vertical handoff. Our future work will cover the following issues. First, we will study more efficient and secure key management schemes such as polynomial-based key distribution techniques to secure temporary handoff keys. Second, because some malicious MTs may utilize temporary handoff keys to obtain temporary admission, and then launch attacks before the full authentication finishes, we will study necessary denial of service countermeasures and resource access control policies to deal with such threats.

References

[1] VINT Group, UCB/LBNL/VINT Network Simulator–ns (Version 2). `http://mash.cs.berkeley.edu/ns`.

[2] IEEE. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Nov. 1997. *IEEE Standard 802.11*.

[3] IEEE. Standard for Local and Metropolitan Area Networks – Port-Based Network Access Control. `http://standards.ieee.org/getieee802/download/802.1X-2001.pdf`, 2001. *IEEE Draft P802.1X/D11*.

[4] IEEE. Draft 4 Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation, July 2002. *IEEE Draft 802.1f/D4*.

[5] IEEE. Draft 5 Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation, Jan. 2003. *IEEE Draft 802.1f/D5*.

[6] IEEE. Medium Access Control (MAC) Security Enhancements, May 2003. *IEEE Standard 802.11i/D4.0*.

[7] IEEE. Trial-Use Recommended Practice for Multi-Vendor Access Point Interoperability via an Inter-Access Point Protocol Across Distribution Systems Supporting IEEE 802.11 Operation, July 2003. *IEEE Standard 802.11f*.

[8] IETF. Context Transfer, Handoff Candidate Discovery, and Dormant Mode Host Alerting (Seamoby). `http://www.ietf.org/html.charters/OLD/seamoby-charter.html`, June 8 2004.

[9] W. A. Arbaugh, A. Mishra, M. Shin, N. Petroni, T. C. Clancy, I. Lee, and K. Jang. Using Neighbor Graphs in Support of Fast and Secure WLAN Mobility. `http://www.umiacs.umd.edu/partnerships/ltsdocs/LTS-talk-04-1.pdf`, Feb. 4 2004.

[10] 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 *ACM WMASH'04*, pages 50–60, 2004.

[11] M. Bellare, R. Canetti, and H. Krawczyk. Message Authentication Using Hash Functions: The HMAC Construction. *RSA Lab's CryptoBytes*, 2(1), 1996.

[12] M. Buddhikot, G. Chandranmenon, S.-J. Han, Y.-W. Lee, S. Miller, and L. Salgarelli. Integration of 802.11 and Third-Generation Wireless Data Networks. In *IEEE INFOCOM'03*, San Francisco, USA, Mar. 30 – Apr. 3 2003.

[13] J. Edney and W. A. Arbaugh. *Real 802.11 Security: Wi-Fi Protected Access and 802.11i*. Addison-Wesley Professional, July 15 2003.

[14] J. W. Floroiu, R. Ruppelt, D. Sisalem, F. Focus, and J. V. Stephanopoli. Seamless Handover in Terrestrial Radio Access Networks: A Case Study. *IEEE Communication Magazine*, 41(11):110–114, Nov. 2003.

[15] O. Goldreich, S. Goldwasser, and S. Micali. How to Construct Random Functions. *Journal of the ACM*, 33(4):792–807, 1986.

[16] H. Haverinen and J. Salowey. Extensible Authentication Protocol Method for GSM Subscriber Identity Modules (EAP-SIM). `http://ietfreport.isoc.org/all-ids/draft-haverinen-pppext-eap-sim-16.txt`, Dec. 21 2004.

[17] H. Krawczyk, M. Bellare, and R. Canetti. Keyed-Hashing for Message Authentication. `http://www.ietf.org/rfc/rfc2104.txt`, Feb. 1997. *IETF RFC 2104*.

[18] J. McNair, I. F. Akyildiz, and M. D. Bender. An Inter-System Handoff Technique for the IMT-2000 System. In *IEEE INFOCOM'00*, volume 1, pages 208–216, Tel Aviv, Isreal, Mar. 2000.

[19] D. Mills. Internet time synchronization: the network time protocol. *IEEE Transactions on Communications*, 39:1482–1493, 1991.

[20] A. Mishra, M. Shin, and W. Arbaugh. An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. Technical Report UMIACS-TR-2002-75, University of Maryland – College Park, 2002.

[21] A. Mishra, M. Shin, and W. Arbaugh. Context Caching using Neighbor Graphs for Fast Handoffs in a Wireless Network. Technical Report UMIACS-TR-2003-46, University of Maryland – College Park, 2003.

[22] K. Pahlavan, P. Krishnamurthy, A. Hatami, M. Wliant-tila, J. P. Makela, R. Pichna, and J. Vallstron. Handoff in Hybrid Mobile Data Networks. *IEEE Personal Communications*, 7(2):34–47, Apr. 2000.

[23] C. Rigney, S. Willens, A. Rubens, and W. Simpson. Remote Authentication Dial-In User Service (RADIUS). `http://www.ietf.org/rfc/rfc2865.txt`, 2000. *IETF RFC 2865*.

[24] M. Stemm and R. H. Katz. Vertical Handoffs in Wireless Overlay Networks. *ACM MONET*, 3(4):225–350, 1998.

[25] Q. Zhang, C. Guo, Z. Guo, and W. Zhu. Efficient Mobility Management for Vertical Handoff between WWAN and WLAN. *IEEE Communication Magazine*, 41(11):102–108, Nov. 2003.