Introducing History-Enriched Security Context Transfer to Enhance the Security of Subsequent Handover

Ulrike Meyer*
Darmstadt University of Technology
umeyer@cdc.informatik.tu-darmstadt.de

Susanne Wetzel
Stevens Institute of Technology
swetzel@cs.stevens.edu

Abstract

Many solutions for securing inter-provider handover proposed to date make use of the concept of security context transfer. However, none of these solutions addresses problems arising from subsequent handover. In this paper, we provide a formal model for subsequent security context transfer and define a set of security requirements. We furthermore present a new solution that meets all but one of these requirements. In particular, we combine the concept of a history-enriched security context transfer with a policy-based handover decision process.

1. Introduction

One of the main challenges in securing inter-provider handover is to allow for active mobile devices (MDs) to securely access a destination network (DEST) without causing perceivable interruptions of on-going connections. Handover generally includes some form of mutual authentication and authorization between MD and DEST as well as establishing cryptographic keys and negotiating security mechanisms to be used after handover. The most promising approach to meet the efficiency requirements imposed by handover procedures to date is to derive the cryptographic keys that will be used after handover from the ones used before handover and transfer these to MD's new point of network attachment. This kind of a security context transfer also includes the sending of the network's authorization for this particular handover.

The concept of Security Context Transfer (SCT) has been suggested previously (e.g., [9, 8, 1, 3, 2, 6]) and is widely used for handover in mobile communication standards like GSM and UMTS. However, none of the previous work explicitly addresses the problems arising from subsequent inter-provider handover. In particular, subsequent context transfers make the protection of the connection after handover depend on the protection of each connection between MD and a previously serving network. Moreover, previous work fails to meet the interest of previously serving networks in the negotiation of secure mechanisms to be used after handover. Instead, it is implicitly assumed that all network providers offer the same level of protection and that MD and DEST will try to maintain the protection level upon handover. This, however, is an unrealistic assumption as different providers will typically support different security mechanisms and—depending on the type of handover agreement—different providers will take different risks if weak mechanisms are used after handover.

In this paper we provide a formal model for subsequent context transfers and define a new set of security requirements. We furthermore present the new concept of a history-enriched SCT that meets all but one of these requirements. The context history includes information on all security mechanisms used prior to the handover in question as well as information on how the transfered cryptographic keys were derived. MDs and networks determine their handover policies dependent on this context history. Consequently, the new procedure not only protects all participants in the handover from attacks arising from the use of any sort of weak mechanisms before this handover but also enables the inter-operation of providers supporting different security levels. Moreover, our procedure explicitly specifies a method to negotiate the security mechanisms to use after handover that not only takes MD's and DEST's policies but also policies of previously serving networks into account. This helps to protect against attacks arising from the use of weak mechanisms after handover.

Outline: In Section 2 we provide the model for an inter-provider handover procedure and introduce the security architecture of the wireless access technology in question. In Section 3 we define the new requirements for SCT followed by the new history-enriched policy-based approach in Section 4. We close this paper by summarizing previous work on SCT (Section 5).

*This work was completed while the author was a visiting scholar at Stevens Institute of Technology.

Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)
0-7695-2520-2/06 $20.00 © 2006 IEEE

2. Model

Each mobile device is pre-registered with a dedicated network, its Home Network (HN). In a pre-registration process, MD and HN exchange credentials (e.g., a shared secret key, or public key certificates, or combinations of user names and passwords). HN usually has roaming agreements with a number of Foreign Networks (FN). Based on these roaming agreements and the credentials exchanged with HN, MD and a network (i.e., HN or any FN that has a roaming agreement with HN) can authenticate each other by means of an authentication protocol a and agree upon a master session key by means of a key agreement protocol ka.

We use the term Anchor Network (AN) to denote the network to which MD authenticated itself first. AN can either be HN, or any FN to which MD has roamed to. Upon roaming to AN, MD and AN first negotiate the authentication and key agreement protocols a_0 and ka_0 to be used as well as the encryption and integrity protection mechanisms em_0 and im_0 to be used after successful authentication. By means of the master session key K_0 and a key establishment process ke, MD and AN establish data protection keys EK_0 and IK_0 to be used for encryption and integrity protection. Upon successful authentication, AN and MD share the security context $S_0 = (K_0, \underbrace{a_0, ka_0, ke_0, em_0, im_0}_{ss_0})$. We refer to S_0 as the *initial* security context.

An inter-provider handover procedure consists of three phases. In the first phase, the reason for handover is detected (e.g., a decrease in the reception level of the currently serving network access point) by MD (mobile-initiated handover), the currently serving network (network-initiated handover) or by the currently serving network with help of measurement data provided by MD (mobile-assisted handover). In a second phase, a candidate destination network is selected and in the third phase the handover itself is carried out. The second and third phase can either be controlled by the network (network-controlled handover) or by the mobile device (mobile-controlled handover). Due to the page limitations, we focus our discussion on mobile-assisted, network-controlled handover only.

A *first order* handover is a handover from AN as source network SRC_1 to a destination network DEST_1. A *second order* (k-th *order*) handover is a handover from $\mathrm{DEST}_1 = \mathrm{SRC}_2$ ($\mathrm{DEST}_{k-1} = \mathrm{SRC}_k$) to a destination network DEST_2 (DEST_k). In the following we refer to handover of second or higher order as subsequent handover.[1] A k-th order handover includes the transfer of a security context S_k to DEST_k. The components of the security context are detailed in Section 4.

Our model of the anchor network, the initial security context, and the subsequently serving networks leads to three new control types for subsequent handover reflecting different types of handover agreements between access networks: SRC-*controlled* handover, AN-*controlled* handover, and HN-*controlled* handover.[2]

SRC-controlled handover: In SRC-controlled procedures, the source network SRC_k of a k-th order handover determines the handover reason, selects the candidate destination network and initiates as well as authorizes the actual handover. A handover of MD from SRC_k to DEST_k can take place if SRC_k and DEST_k have a handover agreement. The HN of MD is only involved in the k-th order handover procedure if it is the source or destination network. Consequently, HN delegates the control of a first order handover to AN, AN delegates the control of a second order handover to $\mathrm{DEST}_1 = \mathrm{SRC}_2$ and so on. MD is assured of SRC_k's authorization of the handover as soon as it receives a handover command message from SRC_k. Similarly, SRC_k's authorization of the handover is assured to DEST_k in form of a handover request. In case of commercial network access providers, SRC_k's authorization includes its guarantee to reimburse DEST_k for service provisioning. For a mobile device, a SRC-controlled handover implies transitive trust in the network providers: MD trusts HN's authorization by means of the initial authentication between MD and AN. Subsequently, MD trusts handover commands received from $\mathrm{AN} = \mathrm{SRC}_1$, $\mathrm{DEST}_1 = \mathrm{SRC}_2$ and so on.

AN-controlled handover: In an AN-controlled k-th order handover, AN determines the handover reason (with the help of SRC_k), selects the candidate destination network and initiates as well as authorizes the actual handover. MD accepts handover commands originating from AN only. A handover of MD from SRC_k to DEST_k can take place if DEST_k and AN have a handover agreement. It is AN that assures MD of DEST_k's authorization to offer service after handover. Similarly, AN assures DEST_k that MD is authorized to be handed over to DEST_k. In case of commercial network access providers, AN's authorization of the handover provides DEST_k with the guarantee that AN will reimburse DEST_k for its service provisioning.

HN-controlled handover: In an HN-controlled han-

[1] Previous work on SCT-based inter-provider handover ([2, 3, 6, 7, 8, 9]) model only the currently serving source network and the destination network of a handover. Consequently, previous solutions can be interpreted as addressing first order handover only or as ignoring the impact of previous handover.

[2] We assume that within one technology only one type of handover procedures will be standardized, such that all subsequent handover are of the same control type.

dover procedure, SRC_k determines that a handover reason occurred and informs HN. HN selects the candidate destination network and initiates as well as authorizes the actual handover. MD accepts handover commands that originate from its HN only. A handover to DEST_k can take place if HN and DEST_k have a handover agreement. It is HN that assures DEST_k that MD is authorized to be handed over to DEST_k. In case of commercial providers, HN's authorization provides DEST_k with the guarantee that HN will reimburse DEST_k for its service provisioning.

Figure 1 describes a network-initiated handover procedure. HCN stands for the Handover Controlling Network. Depending on the type of handover control, HCN is to be replaced by HN, AN or SRC_k.

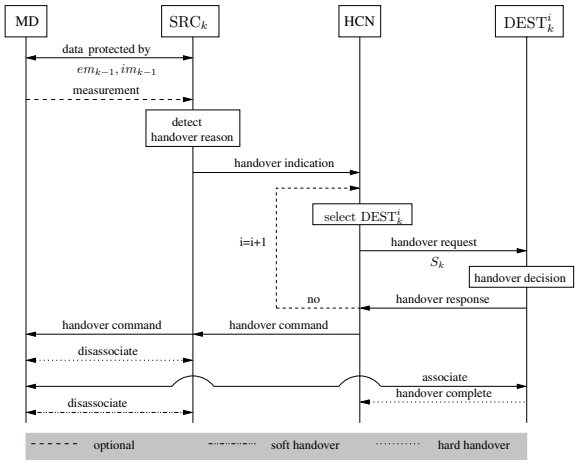

Figure 1. Handover procedure

Before a k-th order handover, MD is connected to SRC_k. MD and SRC_k use some encryption mechanism em_{k-1} and some integrity protection mechanism im_{k-1} to protect data and signaling traffic with some keys EK_{k-1} and IK_{k-1} established by means of some key establishment process ke_{k-1}. MD may support SRC_k in detecting a handover reason. Upon detecting a reason for an inter-provider handover, SRC_k sends a handover indication message to HCN. This includes MD's identity as well as measurement data regarding the quality of MD's reception of candidate networks. HCN uses the measurement data and possibly other information on the candidate networks to determine an ordered list of candidate destinations $L = \{\text{DEST}_k^1, \ldots \text{DEST}_k^n\}$. Starting with $i = 1$, HCN sends a handover request to DEST_k^i. The handover request includes the identities of MD, DEST_k^i, SRC_k, and HCN as well as a security context S_k. Upon receipt of a handover request, DEST_k^i decides whether or not to accept the handover request. Based on its decision, DEST_k^i sends a positive or negative response.
In case of a negative response, HCN proceeds with the next candidate in L. Otherwise, HCN sends a handover command to MD including DEST_k^i's identity. MD either disassociates from SRC_k before associating with DEST_k^i (hard handover) or first associates with DEST_k^i and then disassociates from SRC_k (soft handover).

3. Security Requirements

In this section we define requirements to secure subsequent inter-provider handover. Requirements R-1 to R-3 address security problems in a k-th order handover that arise from previous handover. Requirements R-4 to R-8 address security problems that are independent of any previous handover, but arise from the newly-specified, different types of handover control.[3]

R-1 *$DEST_k$ shall base its decision on whether to accept or refuse a given handover request on (1) the so-called cipher suites $cs_0 = (ke_0,\ em_0,\ im_0), \ldots, cs_{k-1} = (ke_{k-1},\ em_{k-1},\ im_{k-1})$ previously used between MD and any of the previously serving networks; (2) the initial authentication protocol a_0; (3) the initial key agreement protocol ka_0; and (4) the method used to derive the master key K_k (to be used after handover).*

For example, this allows DEST_k to reject a handover request if 'no integrity protection' and 'no encryption' was used between SRC_k and MD and thus, e.g., protects DEST_k against accepting the handover request for an attacker that managed to impersonate a victim MD to SRC_k.

R-2 *Knowledge of a master session key K_k (used by MD and $DEST_k$ after handover) shall not reveal any information on any previously used master session key K_j $(0 \leq j \leq k-1)$.*

E.g., this protects SRC_j $(1 \leq j \leq k)$, MD, and HCN from attacks due to the use of weak security mechanisms after the k-th order handover and malicious destination networks that exploit their knowledge of K_k.

R-3 *Knowledge of a previously used master session key K_j $(0 \leq j \leq k-1)$ shall not reveal any information on K_k to anyone except MD and $DEST_k$.*

This protects, for example, DEST_k and MD from attacks arising from weak security mechanisms used between MD and any previously serving network. It also protects DEST_k from any malicious SRC_j $(1 \leq j \leq k)$ that exploits its knowledge of K_j.

R-4 *The negotiation of the cipher suite $cs_k = (em_k,\ im_k,\ ke_k)$ to be used after the k-th order handover shall enforce compliance with policies set by HCN, MD, and $DEST_k$.*

It is crucial for HCN to influence the selection of mechanisms to be used after handover as otherwise it may suffer from, e.g., impersonation attacks due to the use of a weak cipher suite cs_k.

R-5 *The negotiation of the cipher suite cs_k shall be protected against bidding down attacks.*

R-6 *The security context shall include information on the lifetime of the initial master session key K_0.*

This, for example, allows DEST_k to have considerations on the key lifetime be part of its handover decision. It furthermore allows HCN to pass on its restrictions on key lifetimes to DEST_k.

[3] For the general case, it is impossible to prove that this constitutes a complete set of requirements.

R-7 *The security context transfer between HCN and $DEST_k$ on a k-th order handover shall be integrity protected (including replay protection) and shall provide a proof of its origin (HCN) to $DEST_k$. Furthermore, the security context transfer shall be encrypted in case it includes any confidential information.*

R-8 *The handover command message shall always be integrity protected (including replay protection) and provide a proof of its origin (HCN) to MD.*

4. History-Enriched SCT

In the following we detail a concept that will meet the requirements R-1, R-2 and R-5 to R-8 for subsequent handover. While our solution for the SCR-controlled case does not meet R-3, our solutions for the HN-controlled and the AN-controlled cases meet R-3 in part. The new components of our solutions are a context history, key derivation methods for each control type, a specification of policies and handover agreements, as well as the handover procedure itself, including a new negotiation method for cs_k.

Context History: In order to allow for DEST_k to base its handover decision on the security mechanisms used before handover (R-1), we add the following security suite history ssh_{k-1} to the context transfered during a k-th order handover $ssh_{k-1} = (ss_0, cs_1, \ldots cs_{k-1})$. We also add T_{k-1} to the context to provide information on the total lifetime of the initial security context at the time of initiation of the k-th order handover.[4]

Key Derivation: The key derivation depends on who controls the handover. In an SRC-controlled handover, MD, SRC_{k-1} and SRC_k negotiate a key derivation function kd_{k-1} as part of the security mechanism negotiation on the $(k-1)$-st order handover. kd_{k-1} is included in the security context S_k which DEST_k receives from SRC_k during the k-th order handover. We require kd_{k-1} to be a pre-image resistant hash function (see [4] for a precise definition) that takes K_{k-1}, DEST_k's identity, and a fresh random number r as input and derives master session key K_k. The pre-image resistance guarantees that knowledge of K_k does not reveal any information on any previously used master session key K_j $(0 \leq j \leq k-1)$ (R-2). However, knowledge of any previous key and the identities of the previously serving networks implies knowledge of K_k. Consequently, R-3 can not be met by this key derivation method. In particular, SRC_k and any previously serving network SRC_j $(1 \leq j \leq k)$ gain knowledge of K_k. In an AN-controlled handover, MD and AN negotiate a pre-image resistant hash function as key derivation function kd_0 during the negotiation of the initial security suite ss_0. This key derivation function is included in any subsequent security context transfer S_k. Again, kd_0 takes K_0, DEST_k's identity, and a fresh random number r as input and derives a master session key K_k. The pre-image resistance again guarantees that R-2 it met. The fact that AN (and MD) derive K_k from K_0 additionally guarantees that knowledge of K_j $(1 \leq j \leq k-1)$ does not reveal any information on K_k. However, it is important to note that knowledge of K_0 reveals all subsequently used K_j $(1 \leq j \leq k)$. Consequently, this key derivation method meets R-3 in part. In an HN-controlled handover, MD and HN negotiate a key derivation function kd_0 as part of the pre-registration process. HN either obtains knowledge of the initial master session key K_0 during the (roaming) authentication between AN and MD, or AN transfers K_0 to HN as part of the handover indication message. K_k is derived from K_0 in the same was as in the AN-controlled case. Again, this key derivation method meets R-2 and guarantees that the knowledge of K_j $(1 \leq j \leq k-1)$ does not reveal any information on K_k. However, AN as well as HN may gain knowledge of all of the subsequently used master session keys. Consequently, R-3 is again met in part.

Policies: MD, HCN and any destination network DEST each have policies w.r.t. which cipher suites they allow to be used after a k-th order handover, $(0 \leq k \leq h)$, $h \in \mathbb{N}$, given any particular security suite history ssh_{k-1}. They express these policies by pre-defining sets $\mathscr{H}_{MD}|_{ssh_{k-1}}$, $\mathscr{H}_{HCN}|_{ssh_{k-1}}$, and $\mathscr{H}_{DEST}|_{ssh_{k-1}}$ of cipher suites $cs = (ke, em, im)$ for any possible security suite history ssh_{k-1}. MD, HCN and DEST set $\mathscr{H}_{MD}|_{ssh_{k-1}}$, $\mathscr{H}_{HCN}|_{ssh_{k-1}}$ or $\mathscr{H}_{DEST}|_{ssh_{k-1}}$ to be empty iff they do not allow for handover for a particular security suite history ssh_{k-1} at all. Furthermore, MD, HCN and DEST each have a policy setting an upper bound on how long an initial security context may be used. To express this policy, MD, HCN and DEST each define a threshold Tr_{MD}, Tr_{DEST}, respectively Tr_{HCN}.

Handover Agreement: HCN and DEST enter a handover agreement that regulates terms and conditions (including, e.g., accounting issues) for HCN-controlled k-th order handover to DEST $(1 \leq k \leq h)$. The handover agreement includes an exchange of credentials that allow HCN and DEST to establish an authenticated and encrypted channel.

As part of the handover agreement, DEST commits to its handover policies by providing HCN with super-

[4] We do not further specify, how T_{k-1} is determined. It should, however, be a measure of the lifetime of the initial security context measured in both time units as well as the amount of data that was so far encrypted and integrity protected with keys derived from the initial master key.

sets $\mathscr{H}^*_{DEST}|_{ssh_{k-1}} \supset \mathscr{H}_{DEST}|_{ssh_{k-1}}$, $(1 \leq k \leq h)$.

Committing to a superset rather than $\mathscr{H}_{DEST}|_{ssh_{k-1}}$ for each $1 \leq k \leq h$ provides DEST with some degree of flexibility. It allows DEST to further restrict its handover policy over time but yet explicitly exclude specific security suite histories after which it will not allow handover at all, already at the time of entering the handover agreement. In turn, DEST's commitment allows HCN to pre-select candidate networks and consequently, e.g., avoid requesting handover to a destination network in vain.

Similarly, DEST commits to an upper boundary u_{DEST} on Tr_{DEST}. This again helps HCN to pre-select candidate destination networks and allows DEST to lower its threshold without prior notice to HCN.

Handover Procedure: Upon detecting a handover reason, HCN determines an ordered list L of candidate destination networks it has a handover agreement with.

HCN picks the first network DEST^1_k in its list and checks whether $u_{\text{DEST}^1_k} \leq T_{k-1}$. HCN then checks whether $\mathscr{H}^*_{\text{DEST}^1_k}|_{ssh_{k-1}}$ is not empty. If either of these checks fail, HCN restarts the selection process with DEST^2_k. If both checks are successful, HCN sends a handover request to DEST^1_k.

The handover request is authenticated and encrypted by means of keys agreed upon based on the credentials established on entering the handover agreement (R-7). The handover request includes the security context $S_k = \{K_k, kd_0, [kd_1, \ldots kd_{k-1}], ssh_{k-1}, T_{k-1}, Tr_{\text{HCN}}\}$ as well as the identities of MD, HCN and DEST^i_k.

Upon receipt of the handover response, DEST^i_k checks whether it received the handover request from an HCN it indeed has a handover agreement with. DEST^i_k then checks whether $Tr_{\text{DEST}^i_k} < T_{k-1}$ and whether $\mathscr{H}_{\text{DEST}^i_k}|_{ssh_{k-1}}$ is not empty. If either of these checks fail, DEST^i_k sends a negative handover response to HCN.[5] If all of theses checks are successful, DEST^i_k sends a positive handover response to HCN.

Upon receipt of a positive handover request from DEST^i_k, HCN starts to negotiate a cipher suite cs_k with DEST^i_k as described below. If the negotiation is successful, HCN selects DEST^i_k as the next destination network DEST_k and sends a handover command to MD including the identity of DEST_k and the fresh random number r used by HCN as input to the key-derivation function. This handover command is integrity protected with key material derived from K_0 (AN-controlled and HN-controlled case) or K_{k-1} (SRC-controlled case) in order to meet R-8. Upon receipt of the handover command MD checks whether $T_{k-1} \leq Tr_{MD}$. In the positive case, MD and DEST_k use the negotiated ke_k to establish keys EK_k and IK_k for encryption and integrity protection.

We suggest the following method to negotiate the cipher suite cs_k to be used after handover: MD sends $\mathscr{H}_{MD}|_{ssh_{k-1}}$ to SRC_k, e.g., during connection establishment. Upon detecting a handover reason, SRC_k forwards $\mathscr{H}_{MD}|_{ssh_{k-1}}$ to HCN. HCN computes the intersection of $\mathscr{H}_{HCN}|_{ssh_{k-1}}$ and $\mathscr{H}_{MD}|_{ssh_{k-1}}$ and includes the result in the handover request to DEST_k. DEST_k in turn computes the intersection of the received set of cipher suites with $\mathscr{H}_{\text{DEST}_k}|_{ssh_{k-1}}$ and selects a cipher suite cs_k from this intersection. DEST_k informs HCN of its choice as part of the handover response. HCN checks whether cs_k complies with its policy, i.e., whether $cs_k \in \mathscr{H}_{\text{DEST}_k}|_{ssh_{k-1}}$. If so, it includes cs_k in the handover command message sent to MD.[6] MD checks whether cs_k complies with its policy. In not, MD drops the connection to SRC_k.

Obviously, the above negotiation methods takes the policies of DEST_k, SRC_k as well as MD into account. However, it leaves the final choice of the cipher suite to DEST_k, such that DEST_k can choose the cipher suite it prefers most. DEST_k may easily ignore preferences indicated by MD or HCN. Depending on the application scenario, leaving the final choice to HN or MD may seem more suitable. We will provide details on alternative negotiation methods in the extended version of this paper.

In order to protect the negotiation against bidding down attacks, HCN and MD have to ensure that the transfer of $\mathscr{H}_{MD}|_{ssh_{k-1}}$ from MD to HCN over SRC_k is integrity protected. In case HCN is AN, MD can protect the transfer with IK_0, im_0. In case HCN is SRC_k, MD can protect the transfer with IK_{k-1} and im_{k-1}. Finally, in case HCN is HN, MD again uses IK_0 (derived from K_0 which is known to HN). In addition, HCN and DEST_k have to ensure that the handover request message as well as the handover response message exchanged between them are integrity protected by means of the keys derived from the credentials they share in order to meet R-5.

Handling Long Histories With every subsequent handover, the security suite history is extended by one cipher suite *cs* and in the SRC-controlled case also by one key derivation function *kd*. The actual number of occurring subsequent handover depends on the size of each of the subsequently serving networks, the speed of MD, the duration of an ongoing connection as well as the path MD takes through each of the networks.

[5] DEST^i_k may also reject a handover request for other reasons than security reasons, e.g., if it does not have the free capacity.

[6] HCN gains knowledge of cs_k at this point, such that it can include cs_k in subsequent security contexts.

The easiest way to reduce the length of the key history is to omit the order and frequency of appearance of a cipher suite. The length of the key history is thus reduced to the number of cipher suites specified for the wireless technology in question. It is, however, important to note that omitting the order and frequency reduces the flexibility of MD, SRC_k and DEST_k in setting their policies and may thus reduce their ability to secure themselves against attacks arising from the subsequent use of certain cipher suites. However, whether or not this is a serious restriction, depends on the technology in question and can not be further resolved with the level of abstraction applied in this paper.

Other ways to limit the length of key histories are to restrict the number of allowed subsequent handover to a certain value set by HCN, or to transfer some notion of security levels of the so far used cipher suites rather then the actual suites. Details on these methods will be discussed in the extended version of this paper.

5. Related Work

SCT has recently been discussed in the context of first order inter-provider handover [1, 2, 3, 6, 7, 8, 9]. This previous work defines security requirements equivalent to R-2, R-8 and R-7 but restricted to first order handover only. Similarly, [7, 8] consider R-3 for first order handover only. In contrast, the requirements R-1, R-5, R-4 or R-6 are neither defined nor addressed in previous work. The authors of [1, 2, 3, 7], concentrate on how SCT potentially allows for faster inter-provider handover but address only R-8 and R-7. It is only in [6, 8, 9] that the first order variant of R-2 is addressed and only Wang et al. [8] also address R-3 in the first order variant. In [6] Soltwisch et al. suggest to derive K_0 from K_1 by adding a random number r_1 to K_0. Upon transfer to MD, r_1 is integrity protected but yet sent in the clear. By construction, SRC (respectively DEST) as well as any attacker that obtained knowledge of K_0 (respectively K_1) can thus easily obtain K_1 (respectively K_0) by intercepting r_1. Consequently, this key derivation method neither meets R-2 nor R-3. Zhang et al. [9] suggest to derive K_1 by means of a pseudo-random function with K_0 and a random number r_1 as input. As in [6], r_1 is transfered to MD in the clear. However, the use of the pseudo-random function as key derivation function guarantees that R-2 is partly met. R-3 is not adressed. Wang et al. [8] suggest to derive K_1 in the same way as in [6]. However, the currently serving network transfers K_1 to MD encrypted with an encryption key EK_0 shared between SRC and MD. This key derivation method meets R-2. However, an attacker that gained knowledge of EK_0 can intercept and decrypt the key transfer and thus obtain K_1. It is, in our opinion, not a good solution to transfer the future master key to MD protected by the old data protection keys. Furthermore, by construction, the source network in their handover procedure gains knowledge of K_1. In order to meet R-3, Wang et al. suggest to use the transfered master key K_1 to authenticate a Diffie-Hellman key-exchange between MD and DEST after handover and derive data protection keys for use after handover from the exchanged key. However, the authors fail to notice that by knowledge of K_0, the source network (or any attacker with knowledge of K_0) can mount a man-in-the-middle attack against the key-exchange. Consequently, R-3 cannot be met by this method.

6. Future Work

In this paper we assume that all subsequent handover are of the same control type. Changing control types on subsequent handover leads to a number of interesting questions with respect to the trust relationships between the networks that require further investigation. Furthermore, future work includes exploring the possible use of the key-splitting approach for inter-provider roaming suggested in [5]. The goal is to develop a key agreement method for HN-controlled handover that enables MD and DEST_k to derive a master session key in a way such that the remaining requirement is met as well. That is, the knowledge of K_j $(0 \leq j \leq k-1)$ does not reveal any information on the master session key K_k.

References

[1] M. Georgiades, N. Akhtar, C. Politis, and R. Tafazolli. AAA context transfer for seamless and secure multimedia services over all-IP infrastructures. In *EW'04*, 2004.

[2] M. Georgiades, H. Wang, and R. Tafazolli. Security of context transfer in future wireless communications. In *Wireless World Research Forum*, July 2003.

[3] J. Loughney, M. Nakhjiri, C. Perkins, and R. Koodil. Context tranfer protocol. Internet-Draft, August 2004.

[4] A. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996.

[5] U. Meyer, J. Cordasco, and S. Wetzel. An approach to enhance inter-provider roaming through secret-sharing and its application to WLANs. In *WMASH'05*, 2005.

[6] R. Soltwisch, X. Fu, and D. Hogrefe. A method for authentication and key exchange for seamless interdomain handover. In *ICON'04*, 2004.

[7] H. Wang and A. Prasad. Security context transfer in vertical handover. In *PIMRC'03*, 2003.

[8] H. Wang and A. R. Prasad. Fast authentication for inter-domain handover. In *ICT'04*, 2004.

[9] W. Zhang. Interworking security in heterogeneous wireless IP networks. In *ICN'04*, 2004.