Security threats in systems supporting IPv6 mobility and state-of-the art security solutions

ANEMONE

Author:

Zoltán Faigl

June, 2007

Table of Contents

Acknowledgment

This study was written as part of the work on the development of a large-scale test-bed offering support of mobile users and devices as well as enhanced services within ANEMONE. The author is grateful to Stefan Lindskog and Anna Brunstrom, from the Privacy and Security Group of Karlstad University (KaU), for their useful comments that they made during the research visit of the first author at KAU, from September to December 2006.

1. Introduction

The attainment, with assurance, of high adequate system reliability, security and integrity is the greatest challenge for computer systems. The entire application environment (rather than just the computer-system) must satisfy simultaneously a variety of critical requirements, which we mean that the failure to satisfy may result in serious consequences [1].

The purpose of this study is to give an overview of the security threats in IPv6 based mobility services. In addition the work involves the research and identification of the protocols relative to IP mobility issues, the understanding of their mechanisms and aims. The security threats relating to mobile IP services can be treated only if we have a deep insight to the related protocol mechanisms and their interdependences. The study also contains a discussion of the state-of-the-art security solutions used in these systems.

The main protocols that we are concerned of are Mobile IPv6 (MIPv6) and all of its descendants, such as Network Mobility (Nemo), Hierarchical Mobile IPv6 (HMIPv6), Fast Mobile IPv6 (FMIPv6) protocols. Aside from this, all the traditional network layer protocols and mechanisms affect the security of IP mobility solution, e.g., Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), Domain Name Service (DNS), and the routing protocol, such as RIPng, OSPFv3, IS-IS, or some multicast routing protocol (PIM-SM). However, the main part of this document deals just with the threats and countermeasures that circumvent the IP mobility related mechanisms. These security solutions suppose that the DNS service, Neighbor Discovery, routing and other fundamental services of the Internet work well.

In the following, in Section 2, we are giving an overview of the main terms for computer-system related risks, then, in Section 3, the common methods for identifying vulnerabilities and specifying threats are discussed. In Section 4, typical vulnerabilities and threats in wireless IP based systems are described. Then, in Section 5, we identify the protocols for vulnerability and threat analysis, and, in Section 6, there are investigated.

2. Definitions

In this section we give some terms related with the analysis of risks in computer-systems. The terms were taken from the famous book of Peter G. Neumann on computer-related risks [1], and also from two of the numerous papers on risk assessment in computer-systems [5][7]. Moreover the taxonomy of Pfleeger and Pfleeger [2] on threats is also used.

Risk management is the total process of identifying, measuring, and minimizing the uncertain events that can affect resources. This definition also implies the process of bringing management (remedial action) and control into the risk analysis. The main ingredients to risk assessment are the identification of vulnerabilities, threats, possible countermeasures in the system, and the calculation of the residual risks.

Vulnerability is a weakness that may lead to undesirable consequences. Vulnerability is a weakness in an information system, system security procedure, internal controls, or implementation that an attacker could exploit. It can be a specification flaw, a design flaw, their combination, or an implementation flaw, e.g., coding bug. Vulnerability, flaw and weakness are often used synonyms [47].

An *attack* occurs when an attacker with a reason to strike takes advantage of a vulnerability to threaten an asset.

Threat is any circumstance or event with the potential to adversely impact an information system. Threat is the danger that a vulnerability can actually lead to undesirable consequences – for example, that it can be exploited intentionally or triggered accidentally. The impact can be, e.g., unauthorized access, destruction, disclosure, modification of data, or denial of service (DoS). Ch. P. Pfleeger and Sl. Pfelleger distinguish between [2]:

- *Interception*: some unauthorized party gains access to an asset, e.g., by eavesdropping, link monitoring, packet capturing, system compromise. Interception can not be avoided, but *confidentiality* services (e.g. end-to-end encryption) present a countermeasure to decrease these threats.
- *Interruption*: an asset of the system becomes lost, unavailable, or unusable, e.g., by destruction of hardware, introduction of noise, removal of routing, denial of service attacks. Interruption can not be prevented, but can be controlled, by redundant links, backup systems, controlling DoS attacks, or by access control. More generally these countermeasures aim to satisfy *availability*, reliability of a system.
- *Modification*: if an unauthorized party not only accesses but tampers with an asset, the threat is a modification. Examples are attacks changing record in a database; modify data in transit; modify software with viruses, Trojan horses, trapdoors. It can be prevented with *integrity* services, such as hashing, use of digital signatures.
- *Fabrication*: an unauthorized party inserts counterfeit (fake) objects into the system, e.g., by inserting new record in a database, injecting new packets or replaying old packets in the communication, spoofing IP address. Typical fabrication attacks are the man-in-the-middle attacks, when the attacker gains control of the communication between two parties. Fabrication can be detected by mechanisms providing *authenticity*, e.g., digital signatures, challenge-response mechanisms, assuring the parties about the source of information. Replay can be prevented using fresh nonces, timestamps.

A *risk* is a potential problem, with causes and effects; to some authors, it is the harm that can result if a threat is actualized; to others it is a measure of the extent of that harm, such as the product of the likelihood and the extent of the consequences. Explicit measures and quantification of risk are themselves risky. Avoiding risks is an exceedingly difficult task that poses a pervasive problem [1]. Risk can be defined as the possibility that a particular threat will adversely impact an information system by exploiting some of its vulnerabilities [5]. A security risk can be quantified as the product of the likelihood of the security breach and the cost of a security breach [7].

Countermeasure is an action, procedure, technique, or other measure that reduces risk to an informational system.

Residual risk is the portion of risk remaining after a countermeasure is applied.

Reliability implies that a system performs functionally as is expected, and does so consistently over time. Reliability is also a measure of how well a system lives up to its expectations over time, under specified environmental conditions.

Security implies freedom from danger, or, more specifically, freedom from undesirable events such us malicious or accidental use. Security is also a measure on how well the system resists penetrations by outsiders and misuses by insiders.

Integrity implies that certain desirable conditions are maintained over time.

3. Methods for identifying vulnerabilities and finding threats

Identifying vulnerabilities

There exist two main formal approaches for the identification of security threats in a system [30].

The first approach is the usage of formal methods, formal logics such as BAN [33], CKT5 [35], GNY [34], AT [36], SvO [37], AUTLOG [38]. A good summary and comparison of formal logics can be found in [39]. In these cases, the vulnerabilities and threats are collected as a result of a formal analysis of the system. For the formal methods community, the main goal is to express security goals, and efficient proof techniques that allow full automation of proofs even for infinite spaces. They work on an abstract level, and work with abstract algebra terms, such as the terms in the Dolev-Yao style model [29]. This presents also the limitation of these techniques. They are typically restricted to passive adversary models, and certain protocol environments, e.g., they are unable to express the partial knowledge of the secret, and thus to model a reactive adversary. The idea of formal proof would stand only if everything that can happen with the participants in the real system could happen at the abstract level. Formal methods are able to show missing requirements and design errors appearing at the abstract level.

The second formal approach is the use of cryptographic methods for proving security protocols. The main goal of the cryptographic community is to define the security of a cryptographic algorithm or a protocol against the most powerful, known attacks, where the adversary has polynomial-time restrictions. Important results of this branch are the linear and differential cryptoanalysis methods, worked out by Matsui, Shamir, and Biham

[40][41][42]. A good summary on linear and differential cryptoanalysis is written by Howard M. Heys [43].

However the cryptographic definitions are at a first glance quite unrelated to the abstractions in formal methods. There are research results which try to combine the advantages of the two approaches. A great step forward was made by M. Backes, B. Pfitzmann and M. Waidner, presenting a universally composable cryptographic library [30]. They proposed a specific Dolev-Yao-style library (which can work on abstract level) with a provably secure real implementation. They introduced the notion of "as secure as", also called as reactive simulatibility, which means that their model is able to prove the security of protocols at an abstract level with an ideal cryptographic library, and subsequently to plug-in the real library, and state that the real protocol is as good as the abstract one. They can deal at abstract level with arbitrary polynomial-time reactive attackers that, e.g., can corrupt machines, modify messages on insecure channels, influence the scheduling, and influence the users themselves.

The vulnerabilities imported at the implementation phase, and the related threats, can not be shown by any of these methods. Formal verification and property-based testing are techniques that are based on the design and the specification of the system, but a computer-system includes policies, procedures, and an operating environment, and these external factors can be difficult to express in a form amenable to formal verification or property-based testing.

There exist other vulnerability analysis methods, which try to capture also the implementation-level vulnerabilities. These are briefly described in the followings.

Penetration testing [3] is a test for evaluating the strength of all security controls on the computer system. It is often called tiger team attack or red team attack. The goal of the study is to violate the site security policy and the normal service. It does not replace the careful design and specification with structured testing. Unlike other testing technologies, it examines procedural and operational controls as well as technical controls. Penetration testing can be simply called "black-box" testing, because here the tester may only know the interfaces of the components.

Structured vulnerability analysis is a "white-box" testing where all the internal functions, the source code of the software, the components in the system are known by the tester. The tester can, e.g., run automatic search tools for coding bugs.

Fault injection is a way for testing the survivability of the system which became faulty in some part. It gives information on the fault resistance of the system. A possible fault injection is, for example, when we modify some bits of a binary executable, and analyze the behavior in such case.

Design and implementation inspection is the traditional way for secure software development. In this case the designers or implementers of the system confront the steps, they are making, with security guidelines. It is useful during specification, design and implementation.

Sean Barnum and Gary McGraw give a good summary in their paper on how to collect and apply the knowledge on security in software development [31]. The main knowledge factors are:

- the general security *principles*,
- the *guidelines* for “things to do and avoid” during software development at the semantic level,
- the *rules* which are similar to guidelines, but they are at the syntactic level, e.g., how to write a good code,
- the *attack patterns*, which describe general attacks, and help to find vulnerabilities already during the design,
- the known *vulnerabilities* of the product, and
- the *historical trends* of the vulnerabilities.

Threat-modeling methods

In secure software development the best common practice in threat identification is threat-modeling. In threat-modeling the threats related to a system are defined by a systematic analysis for given vulnerabilities in the applied mechanisms and their implementation, and by looking for possible ways that the adversary shall make to exploit those vulnerabilities. The vulnerabilities may be identified by any of the above mentioned methods. A complete threat-modeling is almost impossible in complex systems.

Hence, aside from the general systematic methods for threat-modeling, the historical trends for the presence and rate of vulnerabilities in similar systems are also used. Vulnerability trends can lead the search for threads, i.e., determining the parts of the system to focus on. Naturally, to identify threats, we also need to estimate the intents and possibilities of the attackers.

In this study, we identify threats of IP mobility services using threat-modeling approach [4][10], i.e., we analyze systematically a set of possible attacker aims, and collect the relevant threats. In order to get some guidelines for the systematic search of vulnerabilities and the relating threats, related work in threat-modeling methods is discussed. It is followed by the description of typical threats in wireless IP based networks with mobility and multihoming features, in Section 4.

Related Works in threat identification methods

The ideas in [1] illustrate perfectly that the approach we follow is widely used. However, the paper simplifies very much the relation between threats and vulnerabilities, which is far from the reality. It does not give the details of the determination of vulnerabilities and threats. The paper is describing a probabilistic decision-tree approach to quantify the security risks in a system. The roots of trees are the possible vulnerabilities of the system, and the threats are directly originating from the vulnerabilities. However, this direct step from a vulnerability to a threat is not true for real attacks. Typically, an attack would combine several vulnerabilities in multiple steps to represent a real threat. Each threat is branched out to two leaves, one representing the fact when the system applies countermeasure against the threat. The second represents the lack of countermeasure, causing a residual risk in the system. It calculates the total residual risk of the system by calculating the weighted sum of the residual risks, where the weights are computed from the probabilities of the vulnerabilities, the threats and the lack of countermeasures. In a real scenario, it is very difficult to find the precise probability values or distribution of these events, though adversary models, describing the characteristics of the attacker, can help to make assumptions.

S. Hariri et al. [8] mention several existing tools for the analysis of threats, such as fault trees, graph-models, and performance models that analyze vulnerability by checking logs of system software and monitoring performance metrics. Graph-based modeling enables the description of the steps of an attack in time sequence order, the specification of the interdependencies of the attack steps, the description of the preconditions and the postconditions of the attack, the explicit expression of the adversary actions, and the analysis of the capabilities and resources required [17]. Generally used graph-based modeling methods are the fault and attack trees. For a detailed description of attack trees, see the work of B. Schneier [6], and other references, such as [13][12][10]. The main building blocks of an attack tree are the nodes connected by edges, which model the steps of an attacker. The single top node of an attack tree represents the achievement of the attacker's ultimate goal. The interdependencies of the steps are modeled by OR and AND nodes: an OR-node can occur if any of its child events occur, for an AND-node to occur all of its child events are necessary. The tree nodes can be augmented by costs; as a consequence, the most likely attack path can be calculated. Petri-nets [44][45] are also able to illustrate threats similarly to attack trees. They are, however, not restricted to tree structures, e.g., cyclic Petri nets make sense sometimes for attack modeling, when the attacker uses the same strategy to jump from one machine to another inside a network. Colored Petri nets [46], where the tokens can have different colors representing different objects of an attack, with complex transition rules can also be used for very comprehensive attack modeling.

There exist also intentions to collect common attack patterns to share security-related knowledge [17], i.e., describing common sets of exploits in a more abstract form that is applicable across multiple systems [31], such as attacker transitions from preconditions to postconditions, but unfortunately no public forum on attack patterns could be reached by the author.

Secure software development guidelines, such as [9] or [31], and the threat-modeling process at Microsoft software development [10] propose that one of the most common ways of finding threats is to use the STRIDE categories. STRIDE is the abbreviation of the six most common threats to which software systems are exposed. It has the following meaning:

- **S**poofing: attackers pretend to be someone (or something) else

- **T**ampering: attackers change data in transit or rest
- **R**epudiation: attackers perform actions that can not be traced back to them
- **I**nformation disclosure: attackers steal data in transit or rest
- **D**enial of service: attackers interrupt the legitimate operation of a system.
- **E**levation of privilege: attackers perform actions that they are not authorized to perform.

The systematic analysis of the system components, and their related functionalities, driven by the STRIDE categories aids to find the vulnerabilities and to identify the real threats. Secure software development guidelines recommend to deal with security issues in every stage of the secure software development process [9],[31].

Security evaluation may also have recourse to modeling and simulation tools. The usage of emulations enables the combination of real and virtual worlds to study the interaction between malware and systems, and probe for new vulnerabilities and possible threats [11].

The threat-modeling process often includes methodologies to define the likelihood of the threats. The estimation of the probabilities or the rate of the threats is typically based on the attacker model, i.e., the characteristics of the attacker. [7] states that attackers in a remote environment have different behavior than the traditional adversary, who must be physically at the attack location. The number of potential adversaries and the value that the adversary can gain are positively correlated with the rate of security breaches. In contrast, the presence of factors that discourage attacks (capture, incarceration, disclosure of attack tools, damage to reputation, physical harm) and the collective cost of equipment and effort required are negatively correlated with attack frequency. However, the risk of capture or harm for remote attackers is far less deterrent than it is for burglars. The time, effort and other resources become significant factors in the remote adversaries' choice of target. Typical adversary types are insiders and outsiders. Co-opted, disgruntled, or non-malicious employees are typical insiders, while foreign nation states, terrorist organizations, organized crime committers, economic competitors or vandals are typical outsiders [16], [19]. [14] defines a detailed insider attacker model.

To determine the rate of the security breaches, Schechter [7] describes an econometric model, which models the inter arrival time of security breaches with a regression model. The author state that a software system's strength is dominated by the cost to find vulnerabilities, and create exploits. Measuring security strength is difficult because it is a function of an adversary's cost, and not the defender's. The authors define two types of market prices for unreported vulnerabilities. The bid price, highest price offered, and the ask price, lowest price asked for an unreported vulnerability. If the two bounds meet, transaction can happen, and the vulnerability becomes reported. The security strength and the likelihood of threats can be expressed by the probability of the next transition, which comes within a specified time period for a given price. The probabilities could be calculated with a regression model. However, in practice, this method is not applicable, because of the lack of convenient input data.

Besides identifying threats and vulnerabilities by a systematic search, the knowledge of vulnerability trends from the history also represent a helpful tool in secure software development. The trends can be calculated using Common Vulnerabilities and Exposures (CVE) collections. These collections are maintained by secure software development communities, such as Computer Emergency Response Team (CERT, http://www.cert.org/), Australian Computer Emergency Response Team (AusCERT, http://www.auscert.org.au/), Computer Incident Advisory Capability (CIAC, http://ciac.llnl.gov/ciac/index.html), Computer Operations, Audit and Security Technology (COAST, http://www.cerias.purdue.edu/coast/), Forum of Incident Response and Security Teams (FIRST, http://www.first.org/), L. D Stein, The WWW Security FAQ (Stein, http://www.w3.org/Security/Faq/), and W3C Security Resources (W3C, http://www.w3.org/Security/) [16]. These collections are quite diverse, and it is difficult to make advanced search queries in them.

Lekkas and Spinellis [15] recommend a scoreboard to let users to record useful information of the vulnerabilities. Their results indicated that a limited set of classes is enough to classify vulnerabilities based on the titles of CVEs. They classified CVEs based on the impact of the advisories, and the main groups were denial of service, execute code, root privileges, modify-delete, crash-reboot, overwrite, spoof. They also classified CVEs based on their applicability. The outcome of the classification was the following list of groups: Windows OS, Linux OS, HTTP, Perl, CGI, FTP, Internet Explorer, Solaris, Java, DNS, LCMP, Telnet, Netscape, and others.

Jiwnani and Zelkowitz [18] state that threat-modeling is an iterative approach for assessing vulnerabilities in a system, and prioritizing them via risk analysis. However, this approach has several back draws, characterized as follows:

- It is impossible to find all threats using threat-modeling.
- The set of potential threats might be very different to the actual set of threats to the system.
- The number of all possible threats might be too large to allow any future analysis.

They recommend a susceptibility matrix to specify how and where vulnerabilities occur in a system. The analysis indicated that the vulnerabilities are typically concentrated in certain locations with certain causes. Most of the high-level vulnerability areas are common, at least, in Windows and Linux OS. The locations with high vulnerability rates and high impact were in both cases the system initialization, memory management, identification and authentication. The causes of the vulnerabilities were mainly validation, identification/authentication, and exploitable logic. The high-level impacts were DoS, unauthorized access, root access, file manipulation, etc. Their main result was finding out that the initial classification of the flaws (at a given year) was indicative for the future, i.e., where the flaws appeared later. Thus the susceptibility matrix can be used as a vulnerability indicator. Consequently, this approach enables to concentrate software developers to high-vulnerable areas, based on real, implementation-level experiences.

Compared to threat-modeling by systematic search for vulnerabilities and threats, the security analyst may not know which parts of the system should be treated more carefully when this method is used. However, threat-modeling has the advantage of not being limited to the search of threats indicated by the historical trends.

What about future threats? Jungck and Shim [20] identified future security challenges in high-speed networks. To address the threats facing network's today and future demands, we need new security methodologies. The homogeneity of the network, the high-speed connections, and the lack of security measures during the design and implementation of the protocols expose the Internet to global attacks. In fact, the internet network services today could be broken down globally by a well-designed attack, in a very short time period. Today the first sign of a virus in a network is that a part of the network goes down or becomes degraded. Proactively eliminating just the known threats places an impractical burden on existing server and network infrastructures. The security management today can not defeat with future unknown threats and day-zero attacks. The security management requires new real-time solutions. Threat detection systems must be self-adapting and collaborating, analyzing also application level data. They should response immediately if they identify a new threat through behavior monitoring.

4. Typical vulnerabilities and threats in wireless IP based networks

In this section we give some threats common in TCP/IP based wireless networks. We also consider the threats originating from multihoming and mobility. Finally we give some threats relating to the ad-hoc networks, and special applications, i.e., multicast streaming and VoIP.

Overview

Networks can be attacked at multiple layers. An adversary can use or misuse any mechanism of a protocol with a malicious intent, e.g., misuse any field in service data units in order to exploit an exception handling bug in the implementation. Moreover, the attacker does not necessary need to attack the target service to achieve his goal. It is often enough to attack the sub-protocols in the sub-layers on which the service is depending.

Marin [22] presents two vulnerabilities which existed in old IP stack implementations. The first could be exploited by the so-called Land attack, where the adversary generated frames with identical source and destination address, which crashed some older OSs. The second example is the Teardrop attack, where the attacker generated overlapping IP fragments, using the bit flags and the fragment offset field in the IP header, causing error at the destination IP stack. A typical example for the misuse of protocols is the Smurf attack, where the adversary overwhelms the targeted machine IP address with regular ICMP echo response messages by generating ICMP echo requests to different destinations in the name of the target station. Variants of these attacks exist.

Attacks at different layers

Some of the threats in different layers are given by Landwehr and Goldschlag [16].

At the data-link layer, the ARP and RARP protocols represent a weak point. For example, spoofing can be achieved by malicious responses to ARP requests; unsolicited updates to ARP tables can cause denial of service, man-in-the-middle attacks.

At the network layer, the routing and addressing mechanisms can be attacked, e.g., the manipulation of routing tables can cause denial of service. The error handling and control mechanisms can also be used maliciously. ICMP does not provide authentication and integrity, thus, it can be used for the manipulation of routing or other malicious intent. An attacker can inject IP packets, with spoofed origin. The SNMP protocol has weak authentication, systems can be opened for malicious router reconfigurations.

At the transport layer, the ports below 1024 are linked to specific services, so the open ports can give tips for the attackers, which known vulnerabilities to exploit. The traditional TCP and UDP protocols do not provide authentication and strong integrity, which cause the potential for forgery and injection of data. The randomness of TCP sequence numbers is often not achieved, thus the adversary can inject packets.

At the application layer, each application brings new vulnerabilities into the system, e.g, exploitation of coding errors in sendmail made possible to gain root privileges on a server, thus stop and modify audit logs, install malicious software, read, modify, delete user applications and data, and launch new attacks to other machines. Web applications are often criticized for their weak authentication, the attacker can easily bypass authentication in case of HTTP basic and HTTP Digest authentication protocols. Postscript files can alter printer behavior. Java applets can have unlimited access to system resources. JAVA is an interpreted, safe language that is not executed as native code. However, byte-codes can bypass the Java security manager due to bugs, and, thus, hostile applets can be admitted. ActiveX controls are not restricted to access system resources, consequently they should be signed. Many users do not take care of the validity of the certificates, giving the possibility to adversaries to cause harm, close down machines, format hard disk, and install malicious code. Malicious codes, i.e., worms, viruses and Trojan horses, represent the most serious threat for the legitimate functioning of the internet and for secure services. They can install a Trojan horse on the machine, to launch newer attacks such as the distributed denial of service attack (DDoS) [28] causing serious damages for the users and the systems.

Threats based on the position of the attacker

This part discusses generally the possibilities of the attacker at different locations related to the target.

Link type

In case of non-switched link, the attacker can easily wiretap, inject packets, hence interception and fabrication have higher probabilities. Blocking traffic can be either difficult or easy, and if it is easy, then the attacker can easily modify and interrupt (i.e., block or misroute) traffic.

In case of switched link the previous things may be more difficult. But with spoofing ARP or ND messages the adversary may get access to the packet flow, and can also resend it. Hence all the four threats are present.

On-the-path and off-the-path attacker

The attacker is in a good position if it is on-the-path. In case of wired link the attacker can block the link (interrupt), and also to monitor (intercept). In case of wired and wireless link, if the attacker can gain control over a switch or router, then it is easy to block (interrupt) traffic. Injection of packets (fabrication) is also easy for an on-the-path attacker.

An off-the-path attacker can not intercept and modify on the flight. But can inject packets with e.g. spoofed source IP address. However the attacker needs to consider ingress filtering.

Use case for typical vulnerabilities and threats in fixed TCP/IP networks

C. E. Landwehr [16] presents the open-up procedure of a corporate network for Internet services, and the sudden increase of the number of possible threats for the intranet network. The article presents the stages of the intranet network development; from closed local networks to networks interconnected through VPN. The more we depend on public internet networking and other services, the more the number of vulnerabilities and new threats increases. The main threats for the corporation, connecting to the public internet through a firewall are as follows. The threats are given with the *STRIDE* categories.

- Malicious software (virus, worm): *D, I, E, S.*
- Unauthorized external connections by outsiders, because of backdoors by-passing the firewall: *I, E.*
- Eavesdropping mainly by insiders: *I, E.*
- Internet based DoS affecting components outside the firewall: *D.*
- Unauthorized data modification mainly affecting components outside the firewall: *T.*
- Web-address spoofing, i.e., adversary creates similar web addresses, with bogus information: *S,* decrease of the reputation.
- Spoofing order information: *S, D.*
- Misconfigured firewall: *I, R, E,* false sensitivity of security.
- Compromise identity of participants of the business communications through the network: *I.*
- Compromise of internal traffic patterns: *I.*
- Improperly managed cryptography: *I, T.*

In summary, Pfleeger and Pfleeger [2] find the following typical threats in communication networks:

- Interception of data in transit
- Access to programs or data at remote hosts
- Modification of programs or data at remote hosts
- Modification of data in transit
- Insertion of communications impersonating a user
- Insertion of a repeat of a previous communication
- Blocking if selected traffic
- Blocking of all traffic
- Running a program at a remote host

They group these potential threats in the following groups:

- Wiretapping
- Impersonation
- Message confidentiality violations
- Message integrity violations
- Hacking
- Code integrity violations
- Denial of service

Threats in wireless networks

The main security threats in wireless networks are the same as in fixed TCP/IP networks, extended with new ones because of additional vulnerabilities, caused by the difference in the characteristics of the environment [21]. The main differences are:

- Open wireless access medium: augments threats of injecting and eavesdropping, there is no physical barrier.
- Limited bandwidth: more vulnerable to DoS attacks, due to limited bandwidth and in-band signaling
- System complexity: far more complex network mechanisms, random access, mobility support, efficient channel utilization, all components introducing new security vulnerabilities.

The threats introduced by the wireless medium, are channel jamming at RF level and traffic analysis. Both could be prevented only at physical layer level. The risk of unauthorized access, eavesdropping, message forgery, message replay, man-in-the-middle attacks (e.g., by forged ARP messages), and session hijacking are far higher than in wired medium. If no authentication, integrity checking and confidentiality services are provided in the data-link layer.

Threats caused by mobility and multihoming

The effects of multihoming and mobility on the transport-layer on security are presented in [32]. The aim of multihoming is to achieve reliable message transfer. The goal of mobility is to enable continuous communication over address changes, and to provide reachability mechanism. Both, multihoming and mobility, can be considered as dynamic multi-addressing, i.e., each node has a set of IP addresses changing dynamically. The difference to traditional security solutions is, that, firstly, the traditional security solutions suppose static network topology and unchanging addresses during the whole session. Secondly, the mobile signaling can be easily misused, introducing newer potential threats of DoS, session hijacking, spoofing, intercepting data, and man-in-the-middle attacks.

The main new issues caused by dynamic multi-addressing, according to [32], are as follows:

- Dynamic topology: the number of paths increases where eavesdropping is possible. Plain text secrets are more vulnerable.
- Faith of old addresses: when a node leaves an address, it is subsequently allocated to another node. Some packets in flight may end up at the new owner of the address. This can reveal information about the session, and gives possibility for a malicious new address owner to execute attacks.
- Symptoms of failure: In this environment addressing failures are more accepted. However, it is dangerous to trust a peer which has just exhibited symptoms of a failure, because that could be part of an attack.
- We need to consider the possibility of an endpoint making false claims about its addresses, thus, automatically denying service for honest nodes with that addresses.
- Value of spoofing mobile signaling packets and other attacks to the signaling plane increase.
- Importance of error handling: it is very important to forward and process the error handling messages, i.e., ICMP and protocol specific notifications, in the gateways and routers, because it can happen by nature, that some service data are directed to a non-existing address or to the false destination. The only defense possibility of the destination node or the router of the sub-network is to stop the source somehow, otherwise the network will be flooded with useless traffic.

Additionally, some potential attacks in dynamic multi-addressing environment are, e.g., as follows. Their feasibility depends on the specific scenario.

- Address squatting: the malicious multihoming node gives in his list a fake address as secondary address. When the honest node wants to associate to the service, it will suffer a denial of access, because the service automatically avoids close the connection to avoid address conflicts.
- Association hijacking: The old address left by a node is distributed to the attacker node. Attacker receives some packets, which were sent for the previous node. The attacker uses the other party as oracle to discover, e.g., the verifiers for the honest

node’s session, before the honest user could have notified the other party about the address change. Attacker node initiates an address change, to set the old address again as a primary destination address.

- Bombing attack: The multihoming attacker establishes a connection to a server, gives the address of the target as a secondary address. Starts a download, and ignore all subsequent responses. The server automatically redirects messages to the secondary address. Attacker sends spoofed acknowledgments to the server to accelerate traffic, and to cause denial of service at the target node.

Threats in ad-hoc networks

Special services introduce new threats. For ad-hoc environment the threats are the same as for the wireless networks, and extended with some new ones [21], [23]. The cause of that is that nodes are also in charge of providing network services, such as routing and addressing, and the connections have sporadic characteristics, the routing paths change dynamically. The new threats in ad-hoc networks are:

- Control plane attacks: malicious nodes can inject false routing messages, inject erroneous routing information, replay old routing information, distort routing information, introduce excessive traffic load by retransmissions and inefficient routing. They can disrupt the discovery and maintenance of routes between 2 nodes multihop away from each other.
- Data plane attacks: adversaries can simply drop the packets, or replay previously recorded packets, inject forged packets. (The solution of these threats needs already reactive approach: detection and exclusion of mal-behaving nodes).

Threats in multicast services

In case of multicast services the eavesdropping, i.e., tampering of data by external users, can be solved by providing confidentiality in the traditional way, e.g., symmetric encryption. However, multicast source authentication is difficult to achieve, and can be exposed to a special insider attack called collusion attack, i.e., where registered, malicious user group sends forged, tampered data to honest multicast group members [26].

Threat in VoIP and other multimedia services

Voice over IP (VoIP) services or other services generating streaming type traffic are sensitive in specific performance aspects, such as delays, jitters, packet loss, compared to services creating elastic type traffic [24]. Moreover multimedia services need a much more complex signaling procedure than traditional data services, and special treatment at the network perimeters (firewalls and application gateways) to work. They are typically dynamically configured before creating sessions, in order to be adaptable for all environments. These characteristics expose VoIP and similar multimedia services to new threats. Some threats regarding VoIP services are:

- Tampering dynamic network parameter settings leading to other threats, such as session hijacking, DoS, spoofing caller identity, direct victim’s calls through the adversaries call manager etc.
- Get malicious configuration files, malicious software updates, by giving false IP address (e.g., by rogue DHCP) to the client, or direct the client to rogue server (e.g., by ARP spoofing).
- DoS attack need not completely shut down the system, but only delay voice packets.
- Tampering QoS parameters in a multimedia service aware network. Unprivileged use of higher priority service.
- Eavesdropping data has a much higher risk compared to traditional phone services.
- Inject packet with headers causing buffer overflow in the VoIP application because of improper packet header handling. This can lead to DoS, elevation of privilege, information disclosure etc. on the user’s machine.
- Weak access control at the Web interface of the phone service. Threat for privacy.
- Misconfiguration of firewalls, letting open ports to internal machines for outsiders.
- Careless usage of cryptography. Encryption of small packets may cause bottleneck.

5. Identifying protocols and mechanisms for security threat analysis

The purpose of this study is to analyze potential threats in IPv6 based mobility services. This fact leads primarily the choice of which mobility protocols to take into consideration. Moreover, a mobility testbed is implemented in the framework of the project that determines also our choice; because the protocols implemented there need to be examined.

We are interested in the threat-modeling of mobility protocols at the network layer, based on IPv6. The basic protocol in this field is the Mobil IPv6 (MIPv6) protocol that defines a macro-mobility service. All the other mobility protocols in the network layer are the descendants of MIPv6, trying to achieve better variants with less signaling overhead. The planed testbed will contain Nemo (Network Mobility) protocol. Moreover, Fast Mobile IPv6 (FMIPv6) and Hierarchical Mobile IPv6 (HMIPv6) protocols are also candidates to implement later on the mobility testbed. Consequently, we plan to analyze the potential threats in MIPv6, Nemo, HMIPv6 and FMIPv6.

In order to have a broader and entire identification of the interesting protocols, we made a database of all the IP mobility related standards and drafts, from IETF, and made queries related to IP mobility and security in the titles.

The source of the list of standards and drafts was the following site, containing an archive of IP related drafts and standards from 1999:
http://www.join.uni-muenster.de/Dokumente/Alle_Dokumente.php

Additionally, the official IETF site was also taken into consideration when creating the database, mainly the documents from the Internet area:
http://www.ietf.org/html.charters/wg-dir.html

The keywords for the identification of IP mobility related standards and drafts were as follows:
"*mobil*", "*hierarchical*", "*MIP*", "*NEMO*", "*network mobility*", "*fast*", "*seamless*", "*handover*", "*dynamic*", "*host identity*", "*multihoming*”.

Moreover, the keywords for security were the followings:
"*port scanning*", "*protect*", "*identity*", "*hash*", "*authentication*", "*SA*", "*IKE*", "*credit*", "*credential*", "*misbehavior*", "*diameter*", "*radius*", "*PKI*", "*X509*", "*policy*", "*secur*", "*privilege*", "*access*", "*aaa*", "*key*", "*IPsec*", "*authorization*", "*crypto*", "*privacy*", "*vpn*", "*eap*", "*threat*", "*trust*".

Now, we give the resulted lists of the queries. We also highlight with boldface the standards which seem interesting to analyze, at first glance.
The main mobility related RFC standards were the following:

Mobile IPv6 Management Information Base, RFC 4295.

Authentication Protocol for Mobile IPv6, RFC 4285.

Mobile IP Version 6 Route Optimization Security Design Background, RFC 4225.

Mobile Node Identifier Option for Mobile IPv6 (MIPv6), RFC 4283.

Dynamic Host Configuration Protocol for IPv6 (DHCPv6), RFC 4242.

Mobile IPv6 Fast Handovers for 802.11 Networks, RFC 4260.

Things Multihoming in IPv6 (MULTI6) Developers Should Think About, RFC 4219.

Threats Relating to IPv6 Multihoming Solutions, RFC 4218.

Third Generation Partnership Project (3GPP) Networks, RFC 4215.

Hierarchical Mobile IPv6 Mobility Management (HMIPv6), RFC 4140.

Fast Handovers for Mobile IPv6, RFC 4068.

Dynamic Host Configuration Protocol for IPv6 (DHCPv6), RFC 4076.

Fast Reroute Extensions to RSVP-TE for LSP Tunnels, RFC 4090.

Dynamic Host Configuration Protocol for IPv6 (DHCPv6), RFC 3898.

Mobile Nodes and Home Agents, RFC 3776.

Mobility Support in IPv6, RFC 3775.

Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6, RFC 3736.

Dynamic Host Configuration Protocol (DHCP) version 6, RFC 3633.

DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6), RFC 3646.

Goals for IPv6 Site-Multihoming Architectures, RFC 3582.

Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers, RFC 3319.

Dynamic Host Configuration Protocol for IPv6 (DHCPv6), RFC 3315.

IP Mobility Support for IPv4, RFC 3220.

The main security related RFC standards were the following:

Authentication Protocol for Mobile IPv6, RFC 4285.

Common Misbehavior Against DNS Queries for IPv6 Addresses, RFC 4074.
Intermediate System to Intermediate System (IS-IS) Cryptographic Authentication, RFC 3567.
IPv6 Neighbor Discovery (ND) Trust Models and Threats, RFC 3756.
Mobile IP Version 6 Route Optimization Security Design Background, RFC 4225.
Privacy Extensions for Stateless Address Autoconfiguration in IPv6, RFC 3041.
An Internet Attribute Certificate Profile for Authorization, RFC 3281.
Securing Mobile IPv6 Route Optimization Using a Static Shared Key, RFC 4449.
Threats Relating to IPv6 Multihoming Solutions, RFC 4218.
Implementing Company Classification Policy with the S/MIME Security Label, RFC 3114.
Extension to Sockets API for Mobile IPv6, RFC 4584.
Securing Mobile IPv6 Route Optimization Using a Static Shared Key, RFC 4449.
Mobile IPv6 and Firewalls: Problem Statement, RFC 4487.
Problem Statement for Bootstrapping Mobile IPv6 (MIPv6), RFC 4640
IKEv2 Mobility and Multihoming Protocol (MOBIKE), RFC 4555
Design of the IKEv2 Mobility and Multihoming (MOBIKE) Protocol, RFC 4621

It is important to note, that typically the RFC standards do not entirely cover ongoing works, thus, it is expedient to consider also the newest drafts related to mobility and security, such as:

Mobile IPv6 Operation with IKEv2 and the revised IPsec Architecture, draft-ietf-mip6-ikev2-ipsec-07.txt
AAA Goals for Mobile IPv6, draft-ietf-miv6-aaa-ha-goals-03.txt
MIP6-bootstrapping via DHCPv6 for the Integrated Scenario, draft-ietf-mip6-bootstrapping-integrated-dhc-01.txt
Mobile IPv6 bootstrapping in split scenario, draft-ietf-mip6-bootstrapping-split-03.txt
MIPv6 Authorization and Configuration based on EAP, draft-giretta-mip6-authorizazion-eap-04.txt
Using IPsec between Mobile and Correspondent IPv6 Nodes, draft-ietf-mip6-cn-ipsec-04.txt

6. Threats in IPv6 based mobility services

This section investigates the threats in IPv6 based mobility services. We describe briefly the aim and mechanism of each protocol, then, we identify the main threats originating either from the misuse of the protocol mechanisms or from external mechanisms, out-of-the scope of the protocol.

Threats in all IPv6 multihoming solutions

Introduction

Threats related to all IPv6 Multihoming solutions are discussed in RFC 4218 [61]. This is an informational standard issued by the Network Working Group of IETF in October 2005.
We need to consider threats relating to multihoming solutions only if we assume that this is the weakest link in the security of the Internet infrastructure for the multihoming applications. However, it is clear that today, there are other weak links, such as the security of DNS and routing services, and without solving them, the security solutions for multihoming fail. When considering the threats relating multihoming solutions, our assumption is that DNS and routing services function and perform in a by and large trustworthy way.

Threats for the traditional Internet newtorking

Existing attacks for non-multihoming networks are described in this part. Before them, we highlight the assumptions that are not always explicitly discussed.

The assumptions of applications today raise the following problems:

- Place trust in FQDN resolvation to destination IP address (DNS).
- Place trust in routing (routers, routing protocols), packets are routed to the adversary's IP address.
- We generally bind cryptographic keying material and SAs to FQDN's or IP addresses, not to the identity of the peers (interruption, perhaps interception, modification, fabrication)

Threats for non-multihoming networks

Redirection attack

The redirection of traffic to not the intended address is a threat which can be achieved in many ways:

- Routing: The attacker compromises the routing service by injecting fake long prefix routing information into routing tables, causing non-optimized routing of the traffic on the touched part of the network or leading to routing errors, disruption of traffic.
- DNS: the adversary modifies DNS forward lookup (IP) (see RFC 3833, Threat analysis of the DNS) leading to fake IP address resolvation, phishing attacks.
- On-the-path node: an on-the-path attacker can redirect any IP-based traffic, and can intercept, modify and fabricate traffic. To become on-the-path attacker, in case of a public access node, the attacker may inject false Neighbor Discovery or ARP reply messages (ND/ARP spoofing), used to attract all traffic for the legitimate next hop. In this case the attacker was on the same link where the attack happened.
- Not-on-the-path node, but between the host and the DNS server: the adversary may modify DNS reply messages to attract traffic.
- Cause DoS, while not-on-the-path: by false ND or ARP the attacker can cause the honest hosts to believe in a non-existing L2 address. This belief is holded for e.g., one minute, until their ARP cache holds the fake L2 address. This can lead to cause a black-hole for the traffic on a link.

The internet community is working out state-of-the-art solutions for these problems. These are, e.g., Secure DNS, secure BGP, Secure ND.

Packet injection

Another threat in IP-based networks is the fabrication, i.e., packet injection. The problem is caused by the fact that IP addresses are used as identifier in traditional transport-layer protocols, such as TCP and STCP. If no ingress filtering is applied at the perimeters of the networks, then any source address can be used for the packet, in case of ingress filtering the address space of the subnetwork, where the packet is transmitted from, can be used as source address. Hence, there exists a potential injection of malicious packets for transport-layer or above protocols.

The state-of-the-art mitigations for the are making difficult to spoof packets by higher layer mechanisms, e.g., in TCP the attacker has to use the correct sequence number and ports. The lifetime of connection, short window size make hard for an off-path attacker to inject acceptable TCP packet. SCTP uses a 32 bit verification tag which has to be known by the attacker to inject a believable packet. IPSec prevents injections by authentication.

Flooding attacks

Another common threat is the flooding attack, which can also be considered as a redirection attack. Here, the aim of the attacker is to cause DoS, and the attack should not be easily traced back to him. Flooding attacks can be caused in many different ways:

- Reflection without amplification: in this case the attacker induces the resource consumption of other nodes on the network, or the DoS of network services. If the attacker's influence is not amplified by some protocol behaviors, then we speak about a redirection attack without amplification. A TCP Syn attack with spoofed source IP can considered as this type of attack.
- On-the-path attacker: if the attacker is between node A and B, then it can flood A in the following way. Send a TCP Syn to B in the name of A, amplify the requested traffic from B by TCP acknowledgment messages in the name of A, increase the congestion window, and block explicit control messages (Explicit Congestion Notification) from A to B. Any streaming protocol can be used for flooding, if the explicit acknowledgments and feedbacks of the target are forged.
- If attacker is not on the path, then the attack can made only in case of lack of ingress filtering at the perimeters of the network.
- If there is no ingress filtering, the attacker must be on the path at least at the initialization phase of the flooding attack or the attacker must be able to make a blind setup, i.e., guess all the protecting parameters of the participating parties counter fabrication. For example the attacker need to guess the initial TCP sequence number of the server.

Threats for multihoming networks

In multihoming network, the attacker has more possibilities to be on-the-path. The time shift between the movement event (real locator change) and the notification of the communicating peers (binding update) open up new potential threat for the communicating parties (mobile node, peer node), in addition, it arises potential DoS threats for all the Internet infrastructure.

Redirection attack

The attacker can redirect the message flow to:

1. itself: this leads to threats for the confidentiality of the traffic, i.e., interception, or for the integrity of the messages, i.e., modification.
2. to anywhere which is not the destination: these cause threats for the availability, i.e, may cause interruption, DoS for other nodes.

Redirection to the attacker (1.) is always possible for on-the-path attacker. For off-the-path attackers this can be executed in the following ways:

- Once traffic is already flowing: the classic redirection in multihoming can be done. The attacker tries to make a binding update, i.e., make believe for the communicating peer that the location of the attacked node changed. To prevent this attack, the communicating peer should be able to verify, whether the claimed locator really belongs to the claimant.
- Time-shifting attacks: the attacker is firstly on-the-path, then goes away and launches the attack. For example the attacker can leave in the visited network a bogus ARP entry to cause interruption. The attacker can interrupt ongoing services. After eavesdropping the necessary information, the attacker can move away and launch a DoS attack with spoofed messages. For example, it can send TCP Reset after intercepting the good sequence number, port number, etc.
- Premediated redirection: the attacker knows preliminary, that A and B will communicate in the near future. The attacker initiates a connection to B claiming that he is A, at the given location. If the solution to the classic redirection attack is based on "prove you are the same as initially", then A will fail to prove this to B because the attacker initiated the communication. This may cause redirection from A to the attacker, or DoS between A and B. To prevent this attack, the verification of whether a locator belongs to the peer cannot simply be based on the first peer that made contact.
- Replay: While the multihoming problem doesn't inherently imply any topological movement, it is useful to also consider the impact of site renumbering in combination with multihoming. In that case, the set of locators for a host will change each time its site renumbers, and, at some point in time after a renumbering event, the old locator prefix might be reassigned to some other site. This potentially give an attacker the ability to replay whatever protocol mechanism was used to inform a host of a peer's locators so that the host would incorrectly be led to believe that the old locator (set) should be used even long after a renumbering event. This is similar to the risk of replay of Binding Updates in MIPv6, but the time constant is quite different; Mobile IPv6 might see movements every second while site renumbering, followed by reassignment of the site locator prefix, might be a matter of weeks or months. The solution for these

attacks is given by replay protection (fresh nonces), and careful timeout policy for locators.

Redirection to other nodes

Possible attacks to redirect traffic to anywhere on the Internet (2.) are as follows:

- Sending packets to a blackhole: the attacker can use the classic redirection attack to redirect to a non-existent locator or anywhere on the Internet. The solutions counter redirection to the attacker work also for this case.
- Flooding other nodes by basic third party DoS: in this attack the attacker floods any node on the Internet. The attacker can stay in a slow link anywhere in the Internet. B is on a fast link and A is the victim. The attacker could flood A directly but is limited by its low bandwidth. If the can establish communication with B, ask B to send it a high-speed media stream, then the attacker can presumably fake out the "acknowledgements/feedback" needed for B to blast out packets at full speed. So far, this only hurts the path between the attacker and the Internet. If the attacker could also tell B "I'm at A's locator", then the attacker has effectively used this redirection capability in multihoming to amplify its DoS capability, which would be a source of concern.
- Flooding other nodes by on-path help: in this case, the attacker controls an on-the-path node between A and B. The attack is the same as in the previous case, but the on-the-path node injects spoofed acknowledgment messages masquerading as A, and also blocks the trials of A to stop the flooding.
- Privacy related attacks: the use of identifiers make possible defense to some attacks, but also make possible to track the identity. In multihoming solutions, the locators need to be exchanged between the communicating parties. Locators can be wiretapped, eavesdropped, if the multihoming signal control does not provide some privacy protection (e.g., encryption).

Mobile IPv6 (MIPv6)

Introduction

The request for comments for mobility support in IPv6 protocol was specified by the Internet Engineering Task Force (IETF) in June 2004 (RFC 3775) [49]. The related standards and drafts can be found in the repository of the Mobility for IPv6 Working Group (see http://www.ietf.org/html.charters/mip6-charter.html). Already at the specification of the protocol security questions were taken into consideration. A lightweight, but well scalable security solution was included into the route optimization (RO) procedure. The use of IPSec ESP tunnel in transport mode was recommended between the Home Agent (HA) and the Mobile Node (MN).

Problem statement

The idea of the protocol is to support mobility on top of existing IP infrastructure. One of the design goals in Mobile IPv4 was to make mobility possible without requiring modifications to the routers, the applications, or the stationary end hosts. However, in Mobile IPv6 the stationary end hosts may provide support for mobility, i.e., route optimization. The basic support enables CNs to send traffic to the Home Address (HoA) of the Mobile Node (MN), and the Home Agent (HA) takes care of tunneling the traffic to the MN. In the opposite direction, the traffic from the MN to the CN is reverse tunneled to the HA, so the CN will see that the packets cone from the HoA. In route optimization a Correspondent Node (CN), i.e., a peer for MN, learns a binding between the mobile node's stationary home address (HoA) and its current temporary care-of address (CoA). Then, the CN can communicate directly to the MN.

The most important assumptions of Mobile IPv6 are as follows:

- The routing prefixes available to a node are determined by its location, and therefore the node must change its IP address as it moves. Mobile nodes must rely on DHCP or Rputing Advertisement messages.
- The routing infrastructure is assumed to be secure and well functioning, delivering packets to their destinations as identified by destination address. If this assumption does not hold, the base routing service becomes false, and the security services related to MIPv6 can not countermeasure the threats. In order to maintain a trustworthy distributed routing database in the routers, policy rules try to limit the amount of faulty routing table information coming from peers that are in other administrative domains.

Mobile IP (version 4 and 6) tries to solve two problems at the same time. First, it allows transport layer sessions to continue even if the underlying host(s) move and change their IP address. This is due to the fact that traditional transport layer protocols (TCP, UDP) use the IP address as a stationary identifier of the host. Second, Mobile IP allows a node to be accessed through a static IP address, i.e., its home address (HoA). In other words, Mobile IP tries to reserve the identifier nature of IP address. However, IP addresses are locators by their nature.

Idea

A design principle of Mobile IP is to maintain the location state changes in the fewest possible nodes in the internet. Basically, it is enough that the HA act as a proxy for the mobile node, maintain a routing table entry for the mobile node, and tunnel the packets destined for the mobile node to its care-of address. This routing is sometimes called triangular routing since it was originally planned that the packets from the MN to the CN could still traverse directly, for shortened paths and increased performance. But triangular routing is no more used in MIPv6. When packets between MN and CN are tunneled via HA, the solution is called basic mobility support (or basic support).

In addition to the basic support, Mobile IPv6 allows direct communication between the MN and the CN. In this case, correspondent nodes must also support Mobile IPv6 and follow the location state changes of mobile nodes. As a result of a routing optimization procedure initiated by the MN, the CN inserts a binding cache entry (BCE) to its binding cache, i.e., a local routing exception with a short lifetime (e.g. one minute) which binds the home address and the care-of address of the MN. Hence, all the packets, coming from the transport layer of the CN and including the home address of the MN as a destination address, are source routed to the care-of address of the MN. The source address of packets arriving from the care-of address is changed to the home address for the upper layer protocols. Note, that in fact, the binding between the home address and the care-of-address is a binding between an identifier and a locator.

A second design principle in Mobile IP is the trust assumption. It is assumed that the home agent and the mobile node know each other from a prior arrangement, i.e., they could establish a trust relationship, share secrets, and trust in the same root certificate authority. However, the CN and the MN do not need to have any prior arrangement, enabling Mobile IPv6 to function in a scalable manner, without requiring any configuration at the CNs. The security solution in the routing optimization follows this assumption, and thus implements a lightweight, but scalable security solution. The path between the MN and HA can be protected by the use of IPSec tunnel, but between the MN and CN it is not easy to solve the key management, certificate verification, if the CN and MN are not in a business relationship, and if they can appear in any part of the network due to the mobility. Moreover, it is not enough to certify the identity (home address) of a MN, but the CN also needs to verify that the MN is really at the claimed location (care-of address). Traditional certificates do not support the certification of location address; moreover, it is not manageable to create certificates for each care-of addresses of the mobile node. Hence, there are limitations to deploy PKI infrastructure, so to use IPSec between the CN and the MN or HA, however sometimes, the CN and the MN can know preliminary each other, and thus it is possible to use PKI based IPsec solution.

The mechanism of the protocol

In this part, a short description of the main mechanisms of the Mobile IPv6 protocol is given.

Participants

The participants of the protocol are as follows:

- Mobile Node (MN): the MN has a stationary home address, which serves as its identifier for other peers, and also as its locator if it moves in its home network. Additionally, it gets or sets a care-of address when moving in a foreign network. The care-of address can vary, based on the routing prefix of the foreign network and possibly on the DHCP configuration.
- Home Agent (HA): the HA is the proxy which maintains routing information for the MN and tunnels packets to its care-of address, if it is in a foreign network.

- Correspondent Node (CN): the CN is the communicating peer staying in any part of the internet. It may not have any relationship with the MN prior to the communication.

Mechanism of the protocol

Tunneling in Basic Support

Basically MIPv6 functions in the following manner. The HA works as a stationary proxy for the MN. Whenever the MN is away from its home network, the home HA intercepts packets destined to the node's HoA from a CN, and forwards the packets by tunneling them to the node's care-of address (CoA). Tunneling means the encapsulation of the received packets from the CN, i.e., the addition of a new IPv6 header with the CoA as a destination address and the address of the HA as the source address. After reception, the mobile node takes out the original packet from the encapsulated packet, thus, its transport-layer gets packets with the home address in the destination address field. In the reverse direction, the packets are sent back to the CN through the HA. The packets from the MN to the CN are encapsulated and provided with a header containing the CoA as source address and the HoA as destination address. The HA unpack the packets which already have the HoA in the source address field and the address of the CN in the destination address field, and send them to the CN. Consequently the tunneling makes transparent the mobility for the transport layer of the CN and the MN, but generates very long paths. To keep the binding information in the HA up-to-date, the MN sends binding update message to the HA, whenever it changes care-of address. Figure 1 presents the data transfer between the MN and the CN in case of MIPv6 basic support.

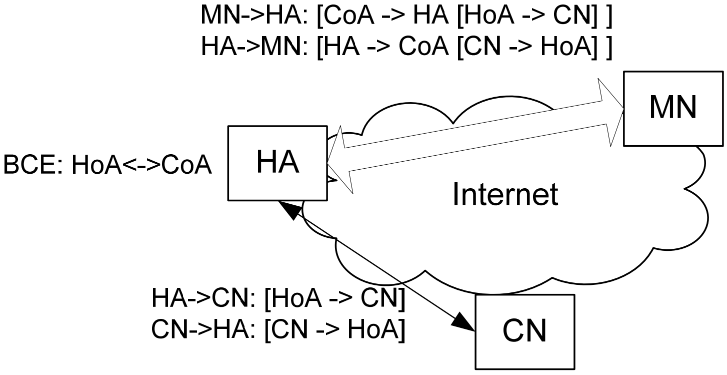

Figure 1. MIPv6 tunneling in case of basic support.

Route Optimization

To alleviate the performance penalty, Mobile IPv6 includes a mode of operation that allow the MN and the CN to exchange packets directly, bypassing the HA completely after the initial setup phase. This mode of operation is called route optimization (RO). When RO is used, the MN sends its current care-of address directly to the CN, using Binding update messages. The CN stores the binding between the HoA and the CoA in its

Binding Cache. The mechanism for route optimization will be described at the security solutions, because it was designed taking into consideration many security concepts. Their understanding is easier after the discussion of threats against MIPv6. Figure 2 illustrates the data transfer in case of route optimization.Figure 1

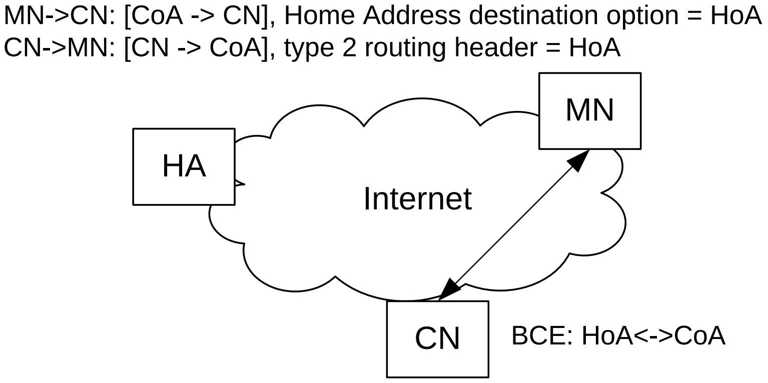
MN->CN: [CoA -> CN], Home Address destination option = HoA
CN->MN: [CN -> CoA], type 2 routing header = HoA
MN
HA
Internet
CN
BCE: HoA<->CoA

Figure 2. MIPv6 data flow in case of route optimization.

Main threats of the protocol

In this part the main threats for MIPv6 are presented. Taking a more abstract angle, IPv6 mobility can be defined as a mechanism for managing local exceptions to routing information in order to direct packets that are sent to the home address to another address, i.e., the care-of address. It is local, since the routing exceptions are valid only at the home agent, and in the correspondent node if RO is used.

The main danger is in the possibility of unauthorized creation of Binding Cache Entries (BCE). Since the CN has no relationship in prior to the communication, it is the most vulnerable for unauthorized Binding Updates.

The dangers of time and space shifting

In a stationary IPv4 network, the attacker must be between the communication nodes at the same time as the nodes communicate. However, in MIPv6 the attacker could attach itself between the HA and the CN for a while, create a BCE at the CN, leave the position and continuously update the correspondent about the mobile node’s whereabouts. This would make the CN to send packets to a false CoA, as long as the BCE remains valid. Another attack is when a large amount of false BCEs for the same home network are inserted, and they expire at the same time, thus redirecting traffic to the HA causing DoS.

Without ingress filtering, i.e., where on the perimeters of the networks routers filter out the packets with fake source address, and without protection if binding updates, the BCEs in CNs could be updated to any false value from any location of the Internet. Ingress filtering limit IP source address spoofing, but does not prevent it.

The main threats

If an attacker can compromise the BCEs, then the following potential threat arise. In this case we are talking about an active attacker. If the attacker directs to itself the communication, then this is a threat for the secrecy and integrity (interception, modification). This means also a threat for DoS for the MN (interruption). The fake BCEs can lead to DoS at other nodes or network parts in the Internet. In case of passive attacker, the main threat is represented by the attacks on privacy. The movements of the MN can be tracked easily, since packets may include potentially sensitive information, e.g., in routing headers or home address options (the CoA and HoA can be eavesdropped).

Firstly, we present the main attacks against the participating nodes. These attacks can also be extended to other nodes in the network.

Basic address stealing

An attacker illegitimately claims to be a given MN at a given CoA and tries to steal the traffic destined to that address. If Binding Updates were not authenticated at all, an attacker could fabricate and send spoofed Binding Updates from anywhere in the Internet. All nodes supporting CN functionality would become unwitting accomplices to this attack. To send a fake Binding Update to the CN, the attacker must known the CoA of the impersonated MN and the address of the CN. The address of a CN might be easily known if the CN is a stationary server. Nodes, that are part of the network infrastructure, such as DNS servers, are particularly interesting targets for attackers, and their address can be easily known. However, ingress filtering limits the IP origin spoofing, consequently, the attacker can only send packets from its own sub-network. Due to this, the attacker may attack a node with a CoA in his own sub-network, or should use the Alternate Care-of Address sub-option in the attack. The Binding Update Authorization mechanism in the route optimization (RO) procedure primarily aims to mitigate this threat. The attacker could send spoofed Binding Update messages to the HA in the name of the MN, when it is in its home network, redirecting its traffic to any CoA or a CoA of the attacker. This would cause DoS for the MN, and interception, modification, fabrication threats. However, in the assumptions we supposed that there is a security association between the HA and MN, thus this attack is prevented by the authentication of Binding Update messages.

Future address stealing

In this attack, the attacker creates a Binding Cache Entry with the home address that it anticipates the target node will use. Then it releases this address. If the BCE had a long expiration time, the target node would acquire the same home address while the BCE is still effective. The attacker could launch successful DoS or MITM attack. This attack needs fairly specific conditions, thus not taken to be a serious threat. It can be limited by short BCE lifetime.

Attacks against secrecy and privacy

By spoofing Binding Updates, the attacker could redirect all packets between two IP nodes to itself, without being an on-the-path attacker. The attacker could also launch a MITM by sending spoofed Binding Updates to both parties, and insert itself in the middle of all connections between them. These attacks represent a threat against secrecy and integrity (interception, interruption, modification, fabrication of data).Strong end-to-end encryption and integrity can prevent these attacks, but a potential DoS threat still exists. The return routability (RR) security mechanism, implemented in route optimization, weakly tries to authenticate the claimed care-of address message, if it relates really to a given home address (identifier).

Basic DoS attacks

By sending spoofed Binding Update messages, the attacker could redirect all packets sent between two IP nodes to a random or nonexistent address or addresses. This can cause an interruption threat for the participating nodes. Moreover any internet node can be targeted by this attack, including nodes belonging to the infrastructure (e.g., DNS servers).

Replaying and blocking binding updates

Any binding update protocol can be attacked by the following mechanism. The attacker blocks binding updates from the mobile at its new location, e.g., by jamming the radio link or by mounting a flooding attack, and takes over the mobile’s connection at the old location. The attacker can impersonate the mobile until the BCE expires at the CN. An attacker may be able to replay recently authenticated Binding Update messages to the correspondent and, consequently, to direct packets to the mobile node’s previous location. This also would represent an interruption, interception threat for the MN. Both of the previous attacks require that the attacker be on the same local network as the MN where it can observe packets and block the MN’s traffic. Therefore these attacks are not as serious as ones that can be mounted from remote locations. The limited lifetime of BCEs, the associated nonces limit the time frame for replay attacks. The security mechanism in routing optimization makes use of MAC and sequence numbers in Binding Updates. The CN must carefully delete expired BCEs from its binding cache.

Attacks against other nodes or whole networks

Basic flooding

By spoofed Binding Update messages, the attacker could redirect traffic to an arbitrary IP address, overloading the target with excessive volume of packets. This attack is called bombing attack, and it can also degrade the service of a whole sub-network. The attack can be very effective. The attacker can know, that there is a heavy data flow between an MN and a CN, and can redirect the traffic to a target CoA. A more sophisticated version of this attack is when the attacker initiates the data stream, redirects this data stream to the target CoA by spoofed Binding Updates, and amplifies the traffic by false acknowledgments. The attacker may also refresh the BCEs by sending periodically Binding Updates. This attack is serious because the target node can be any node or network, and the target may not be able to prevent the attack. For example, a TCP stream can not be reset by the target node, because the flooding data is not processed at the TCP layer of the target node, and no TCP connection Reset is sent. The return routability mechanism in route optimization weakly protects against this attack.

Return-to-home flooding

In this case the attacker’s target is not a false CoA, but the home address or the home network. The attacker claims to be a mobile with the target home address. It claims to be in a visited network, starts to download a heavy data stream from the CN. The attacker, then, stops to refresh the BCEs at the CN, or sends a binding update cancellation. This would cause the CN to redirect the traffic to the home network. The attacker may keep alive even increase the rate of the connection by sending spoofed acknowledgments. This threat is a serious one, and the basic protection mechanism is again the return routability in route optimization. It is hard to fully protect against this threat.

Inducing unnecessary binding updates

Any protocol, which creates states already at the starting phase of the communication, is vulnerable for resource exhausting attacks, where the attacker initiates many instances of useless sessions. Binding update protocols are not an exception, and they need a careful design to remain stateless and consume as less resources as possible, until the parties are authenticated. At least the initiator has to be authenticated by the responder before the responder consumes a big amount of resources. In this case, several resource exhaustion attacks are possible. When the MN receives an IP packet from a CN via the HA, it may initiate the binding update for routing optimization purpose. If the attacker sends spoofed IP packets with different CN address in the source field, the MN may initiate route optimization procedures to many CNs. This attack exhausts the resources of the MN and injects false BCEs in the binding cache of many CNs. A CN can also be attacked by sending spoofed IP packets to a large number of mobiles, with the target CN’s address in the source field. However, this can be limited by setting a limit on the amount of resources that a node uses for processing binding updates, by heuristics, by binding access control lists, by accepting binding updates from or to addresses with which they had meaningful communication. A good measure of meaningful communication is the minimum threshold for per-address packet counting. Moreover the Neighbor Discovery uses a Destination Cache, which can be used for seeing the history of communication.

Forcing non-optimized routing

With a successful DoS attack against the network or the target node, or a successful resource exhaustion of the target node, an attacker can prevent to run the routing optimization successfully, and, thus, force the target to use the less efficient home agent based routing. The target node can mitigate the effects of the attack by reserving more space for binding cache, reverting to non-optimized routing only when it cannot otherwise cope with the DoS attack, by trying aggressively to return to optimized routing, or by favoring mobiles with which it has an established relationship. This attack is not as serious as the ones described earlier, however applications with low latency constraints can suffer drastically from the additional delays caused by triangle routing.

Reflection and amplification

A reflection attack is when the attacker sends data to other nodesm tricking them to send the same number, or more, packets to the target. In the latter case we speak already about amplification. A reflection can hide the address of the attacker. Triangle routing would easily create opportunities for reflection. If a CN accepts packets from a MN with Home Address Option field used, then the CN will reply to the MN via the HA. The packet from the MN could be spoofed, thus the home address or home network could be a target. Reflection and amplification can be prevented by ensuring that the correspondent only responds to the same address from which it received a packet.

Out-of-the-band attacks

ICMP Host unreachable

The attacker can send ICMP Host unreachable messages to the HA or the CN, telling that the CoA is not reachable. If it is sent to the CN, then the attacker can prevent the route optimization, forcing non-optimized routing. If it is sent to the HA, then it can block the HA to send more packets to the CoA of the MN, resulting in a DoS for the MN at that address.

Fake default router

The attacker sends fake Router advertisement messages to the MN, and, thus, the MN sends all the messages to the CN via the attacker. The attacker has to be at the same sub-network as the MN.

Summary of threats in MIPv6

In conclusion, there are DoS, MITM, confidentiality and impersonation threats against the parties involved in sending legitimate Binding Updates, and DoS threats against any other party.

No	Attack name	Target	Severity	Mitigation
	Basic address stealing	MN's CoA, Any node's address	High.	RR

	Future address stealing	MN	Low	RR, BCE lifetime in CN
	Attacks against secrecy and integrity	MN	Low	RR, IPSec
	Basic DoS	Any	Med	RR
	Replaying and blocking Binding Updates	MN	Low	BCE lifetime, seq. number MAC
	Basic flooding	Any	High	RR
	Return-to-home flooding	Any	High	RR
	Inducing unnecessary binding updates	MN, CN	Med	Heuristics,
	Forcing non-optimized routing	MN	Low	Heuristics
	Reflection and amplification	N/A	Med	BU design

Network Mobility (Nemo)

Introduction

The aim of Network Mobility protocol is to enable mobile networks (entire networks which move) to attach to different points of the Internet, allowing session continuity and reachability to every node in the mobile network. The basic support for network mobility was specified in RFC 3963 by the Network Mobility (nemo) Working Group of IETF in January 2005 [50].

Problem statement

The protocol aims to provide connectivity for each node in a moving network in a way that the network prefix remains the same in the mobile network. Thus the nodes may use the normal IPv6 stack without any extension to mobility; moreover, static routers can stay in the mobile network. Moreover, ordinary mobile nodes using MIPv6, or entire mobile networks can attach to the mobile network. This leads to the definition of nested mobility, when mobile networks include one or more mobile networks, until arbitrary depth.

Idea

The idea of Network Mobility Basic Support protocol was to extend the MIPv6 by introducing the Mobile Router (MR), similarly to the default MN. When a MR makes home registration, the HA makes not only a binding entry between the HoA and the CoA of the MR as it is the case in MIPv6 basic support for the MN, but the HA also registers and advertises the network prefixes of the Mobile Network of that MR. Any packet destined to the mobile network thus arrives to the HA, which tunnels the packet to the MR. The MR decapsulates the original packet and routes it to the destination. The Network Mobility Basic Support standard does not define route optimization, so correspondent registration is not discussed in the basic standard. However there are drafts dealing already with route optimization.

An important definition of the standard is “nested mobility”. This means that a Mobile Network can contain arbitrary number of mobile networks, for any depth. Nested mobility can cause high signaling overhead, if no route optimization is used, because all the traffic that intercepts a MR makes a detour to the HA of that MR, which may located far away from the MR.

The mechanism of the protocol

Participants

Mobile Router (MR): extends the definition of MIPv6 MN by adding routing capability between the CoA of the router and the nodes at the mobile network. The available network prefixes at the network of the MR are called Mobile Network Prefixes. One Mobile Network may have multiple MRs.
Home Agent (HA): extends the definition of MIPv6 HA by the fact, that the HA advertises to the infrastructure an aggregation of all Mobile Network Prefixes, which are related with the registered MRs. The HA owns a prefix table, which links the Mobile Network Prefixes to binding entries with MRs.
Correspondent Node (CN): the same as in MIPv6.

The mechanism of the protocol

Dynamic Home Address discovery

The MR has to know which HAs in its home network support NeMo, i.e., which HA will be able to manage the Binding Update from the MR, and advertise the Mobile Network Prefixes. To achieve this, the protocol specifies a new Dynamic Home Agent Discovery Address (DHAAD) Request message sent from the MR, containing the information that the requesting MN was in fact a MR. The DHAAD Reply message contains a flag indicating if the HA supports MRs.

Binding Update

The Binding Update is extended by a flag indicating that BU was sent by a MR. The Binding Update process is for registering the new CoA of the MR at the HA. As a result of a successful BU process, the HA will tunnel the packets coming to an address from the address space of the Mobile Network Prefixes of the MR. The necessary information is stored in form of Binding Cache Entries in the HA, linking the HoA with the CoA, moreover, the HA maintains a Prefix Table for the Mobile Network Prefixes.

To be able to manage and advertise the Mobile Network Prefixes, the HA must know them. The HA can gather this information in the following ways:

- Implicit: In implicit mode, the HA gets this knowledge by any mechanism, e.g., manual configuration, linking Mobile Network Prefixes to the MR’s HoA.
- Explicit: In this case, the Binding Update message sent by the MR also contains the Mobile Network Prefixes. The HA will update the prefix table for that MR based on this information.
- Support for dynamic Routing Protocols: In this case no implicit or explicit Mobile Network Prefix transfer happens. The MR and the HA run an intra-domain routing protocol on their tunnel interfaces (e.g., RIPng or OSPF). If there are changes in the home link prefixes, the HA notifies the MR, or if there are changes in the Mobile Network Prefixes, then the MR notifies the HA about the change. They use routing update messages for this. They must be sure, that they do not propagate information outside the home network and to the visited network.

Bidirectional tunnel between MR and HA

After the Binding Update process a bi-directional tunnel is created between the MR and HA, composed of two unidirectional tunnels. The tunnel from the MR to the HA has the CoA of the MR as the entry point and the address of the HA as the exit point. The tunnel from the HA to the MR has the HA’s address as the entry point and the CoA of the MR as the exit point, respectively. Packets between a node in the Mobile Network and a CN have the addresses of the CN and that of the node (with one of the Mobile Network Prefixes). These packets are encapsulated between the HA and MR, i.e., extended by an IPv6 header which contains the CoA and the address of the HoA. The standard does not deal with the question, why the packets destined to an address having a given Mobile Network Prefix arrive to the HA, however, the HA has to advertise it to the Intrernet infrastructure. If the Mobile Network Prefixes would change often, this could cause problems to maintain. However, Mobile Network Prefixes are not supposed to change often, only the binding of the MR’s HoA to CoAs change due to the mobility. Figure 3 presents the functioning of the bi-directional tunnel between the MR and the HA.

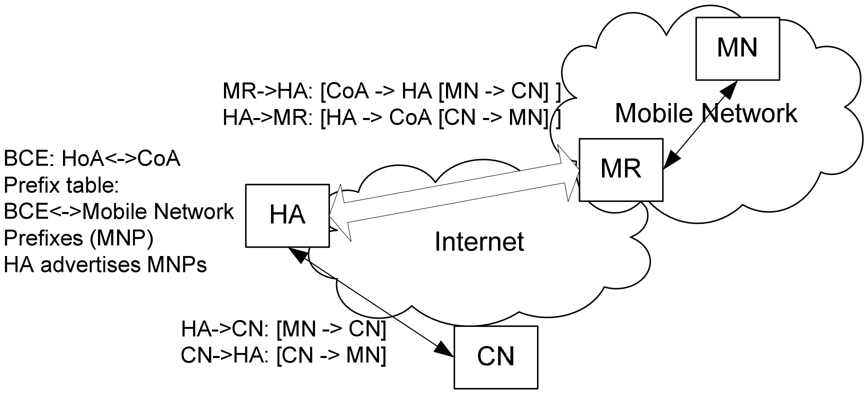

Figure 3. Tunneling in case of NEMO Basic support

Main threats of the protocol

Threat of inconsistent routing updates and routing advertisement in the visited network: the MR must not propagate routing advertisements from the home network to the visited network, because it can cause inconsistencies for the routing of the packets in the visited network.

Threat of fake Mobile Subnet Prefixes: if an attacker could register inconsistent Mobile Network Prefixes, the HA would advertise and attract traffic destined to these networks, then it would tunnel them to the MR. The MR must check, if the packets arriving on the tunnel interface are destined to one of its Mobile Network Prefixes. Otherwise, the Mobile Network would be flooded with false traffic.

The disclosure of Mobile Network Prefixes: the attacker could eavesdrop the Mobile Network Prefixes, if they were sent in explicit mode in the Binding Update messages, and there were no confidentiality protection on the bi-directional tunnel.

Threat regarding fake Binding Cache Entries: The Binding Update process has to be protected, otherwise an attacker could create false bindings, redirecting the traffic of the CN to itself, or to unsuspicious nodes. The same threats are present than for MIPv6, however, here all the nodes in the mobile network and their CNs are threatened.

Hierarchical MIPv6 (HMIPv6)

Introduction

The Hierarchical Mobile IPv6 protocol introduces an extension for MIPv6 and IPv6 Neighbor Discovery to allow local mobility handling. It was specified by the Mobility for

IP: Performance, Signaling and Handoff Optimization (mipshop) Working Group of IETF. The main HMIPv6 standard is RFC 4140, was issued in August 2005 [51].

Problem statement

MIPv6 causes much signaling overhead, and can lead to performance bottlenecks in the Home Networks at the HA. Route Optimization can reduce the burden on HA, however in this case all CNs must be updated at movements of the MN, not only the HA.

Idea

Reduce the number of Binding Updates needed for MIPv6 by introducing mobility regions, and by the use of "local HAs" which tunnels the data to the actual location of the MN. The movement within the region induces Binding Update only once, to the "local HA" called Mobility Anchor Point (MAP). All the CNs and the HA know the Regional Care of Address (RCoA) of the MN which is unique for the MN within the entire region. CNs and the HA exchange data traffic with the MN by using its RCoA as destination address. The MAP tunnels the traffic to the actual location, the on-link CoA (LCoA) of the MN. As a consequence, the radio link is only used by one BU to each MAP for a MN, when it moves within the region and gets a new LCoA. However, the movement between regions induces more signaling overhead than the traditional MIPv6, since the MN may discover available MAPs, select one and register to it, and then, update the HA and possibly the CNs if route optimization is used, with the new RCoA. The HMIPv6 supports also Fast Mobile IPv6 Handovers, to help to achieve seamless mobility for MNs' applications.

The mechanism of the protocol

The participants of the protocol

Mobile Node (MN): MIPv6 mobile node extended with MIPv6-aware functionalities
Correspondent Node (CN): the same as in MIPv6
Home Agent (HA): the same as in MIPv6
Access router (AR): The MN's default router at a given location. It provides the on-link CoA (LCoA), which is the same as the CoA in MIPv6
Mobility Anchor Point (MAP): the MAP is a router located at the visited network, used as a "local HA" by the MN. The MAP manages locally the mobility of the MN in the MAP domain. In a visited network more then one MAPs can be achieved, and they can stay at different level's of the routing hierarchy, thus covering smaller or larger domains, sub-domains. Within a MAP domain the MN has the same Regiona CoA (RCoA). The MN provides this RCoA to the exterior, thus the mobility of the MN within the same MAP domain is seamless for the HA and the CNs. The MAP has no knowledge about the MN's HoA.

Preconditions of the protocol

The ARs must be either manually configured to advertise given MAPs with given distance values (hop counts), or the MAP can be configured to advertise MAP capabilities using Routing Advertisements with MAP option. In the latter case the routers must propagate trustworthily the information in the direction of access routers, moreover they have to increase the distance value.

The mechanism of the protocol

MAP discovery

The Access Routers send Routing Advertisement messages with MAP option, indicating that the advertised node has MAP capability, and also giving the global IP address of MAP. The subnet prefix for the given MAP domain is extracted from this address (first 64 bits). The MAP option is an extension for Neighbor Discovery.

Registration to MAP

If the MN is HMIPv6-aware, then after moving to a new MAP domain, it has to register with the MAP. The MN sends a Binding Update to the MAP, containing its on-link CoA (LCoA) and RCoA as the Home Address. The RCoA is formed by the MN in a stateless manner, by combining the MN’s interface ID and the subnet prefix received in the MAP option of Routing Advertisement. The MAP will act as a local HA, by creating a binding cache entry linking the RCoA to LCoA, and by tunneling all the traffic coming to RCoA to the LCoA. The packets addressed to the RCoA will be intercepted by the MAP because it uses proxy Neighbor Advertisement for the RCoA address.

Every time the MN detects movement, it checks by MAP discovery if it is still in the same MAP domain. If it is in the same domain, it only sends a new Binding Update to the MAP with the new LCoA, distributed by the actual access router (AR) to which subnet it connects.

Update HA and CNs with new RCoA

If the MN changes domain, it should discover, select the new MAP, then registrate to the new MAP. Then it has to send a Binding Update to its the HA and all the CNs, with the RCoA as “care-of-address”. In other words, after changing domain, the MN must execute a home registration and correspondent registrations at the CNs, with Binding Update messages, in which the home address option is set to the HoA and the source address is the RCoA. The specification allows having more then one MAP where the MN is registered. In this case the MN should update the HA and the CNs with each RCoA, or at least with one RCoA.

Tunneling between the MAP and the MN

Following a successful registration at the MAP, a bi-directional tunnel between the MN and the MAP is established. The data flow sent from the MN RCoA to the CN's address is encapsulated with an IP header where the source address is the LCoA and the destination address is the MAP's global IP address. The MAP decapsulates these packets and send them to the CN (see Figure 5). Following the rules of the MIPv6 route optimization, the MN gives the home address destination option (see MIPv6) with the HoA. Thus, if the CN can not reach the MN through the RCoA, after the expiration of the Binding, it will fallback to use "basic support", i.e., send the packets to the HoA of the MN (see Figure 4). In the opposite direction, if a packet arrives from a CN to the MAP, with RCoA as the destination address, the MAP encapsulates the packet and adds an IP header with the MAP global IP address as source address and the LCoA as destination address. Then the MN decapsulates the original packet, where the source address was the CN's address and the destination address was the RCoA.

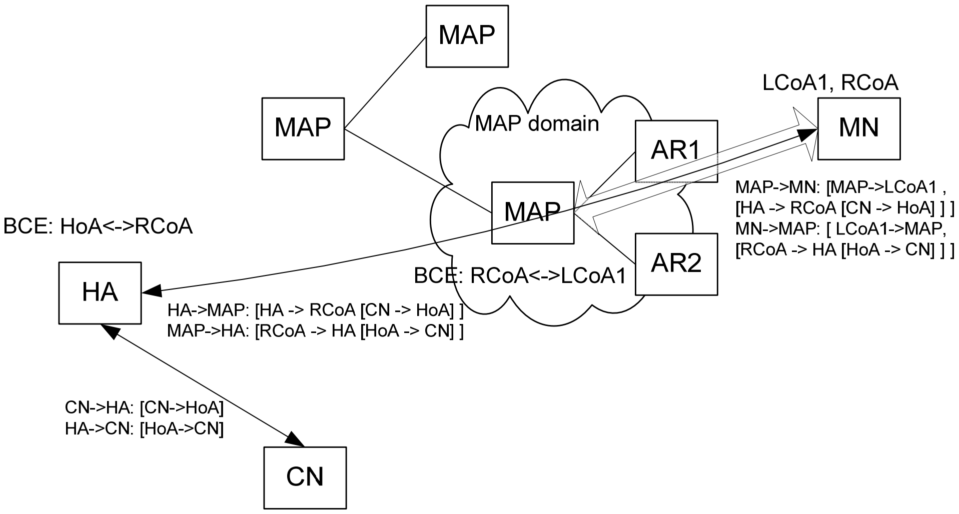

Figure 4. Data transfer in HMIPv6 when it is used together with MIPv6 basic support.

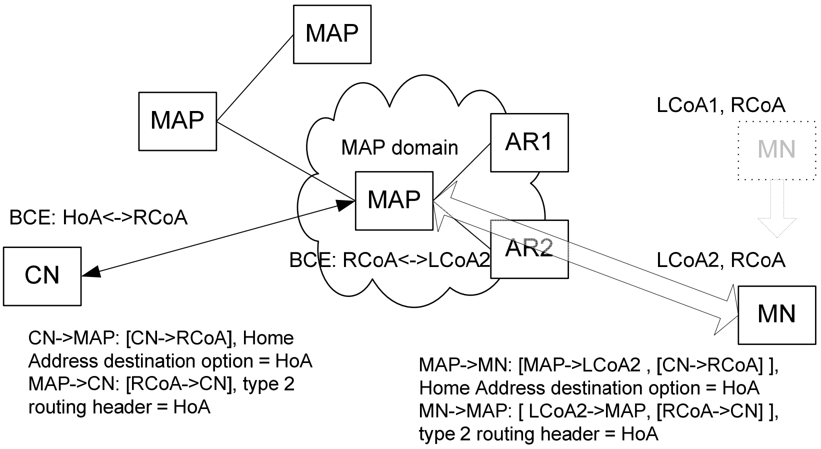
MAP
MAP
MAP domain
AR1
LCoA1, RCoA
MN
MAP
BCE: HoA<->RCoA
AR2
CN
BCE: RCoA<->LCoA2
LCoA2, RCoA
MN
CN->MAP: [CN->RCoA], Home
Address destination option = HoA
MAP->CN: [RCoA->CN], type 2
routing header = HoA
MAP->MN: [MAP->LCoA2 , [CN->RCoA]],
Home Address destination option = HoA
MN->MAP: [LCoA2->MAP, [RCoA->CN]],
type 2 routing header = HoA

Figure 5. Data transfer when HMIPv6 and MIPv6 route optimization are used together.

Local Mobility Management Optimization

Besides the basic bidirectional tunneling of the data flow between the MN and the MAP, the standard enables two communication types with less overhead. The first can be used for short-term communications, during which the MN will not change its LCoA. In this case the MN could leave out the tunneling, and send packets directly to the CN with the LCoA in the source address field. In this case the CN will also respond with the LCoA in the destination field. This is illustrated in Figure 6. The second option is to use the RCoA as a source address without using the Home Address option (as fall-back option to basic support and as identifier for the CN's upper-layer protocols). In this case the CN will believe that the MN is a fixed node at the RCoA. This is presented in Figure 7. The first case functions only until the MN remains at the same LCoA, while the second case enables only local mobility within a MAP domain.

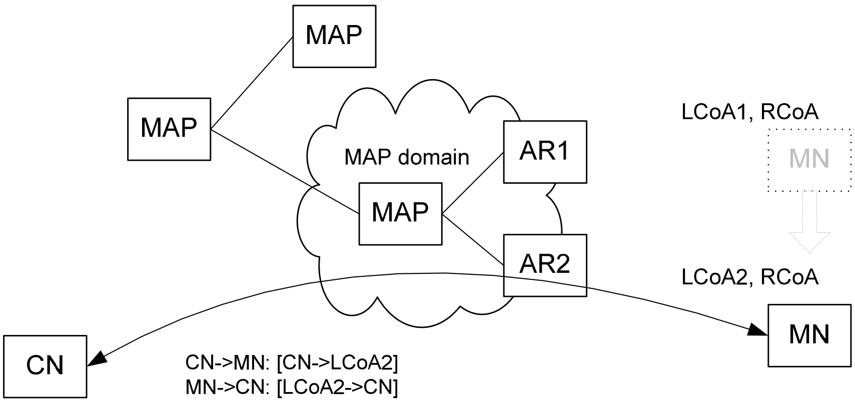

Figure 6. Local mobility management optimization by avoiding HMIPv6 and MIPv6 functionalities.

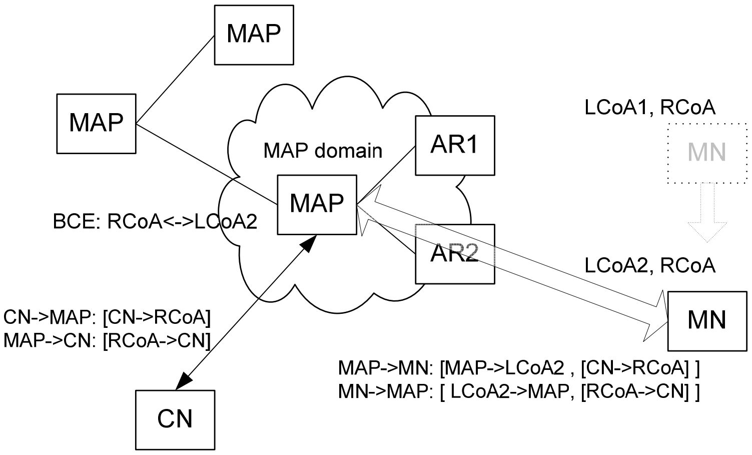

Figure 7. Local mobility management optimization by avoiding MIPv6 basic support and route optimization.

Main threats of the protocol

This specification introduces a new concept to Mobile IPv6, namely, a Mobility Anchor Point that acts as a local Home Agent. It is crucial that the security relationship between the mobile node and the MAP is strong; it must involve mutual authentication, integrity protection, and protection against replay attacks. Confidentiality may be needed for payload traffic, but is not required for binding updates to the MAP. The absence of any of these protections may lead to malicious mobile nodes impersonating other legitimate ones or impersonating a MAP. Any of these attacks will undoubtedly cause undesirable impacts to the mobile node's communication with all correspondent nodes having knowledge of the mobile node's RCoA.

Malicious Binding Update at the MAP: this threat may lead to the redirection of traffic destined to a given RCoA to a malicious LCoA or to an unsuspected network. The Binding Updates must be protected, only authorized MN should send Binding Updates to MAP. However this limits the scalability, because the MN should have trust relationship or preconfigured security associations with the MAPs.

Threats counter the CN: The return routability procedure of MIPv6 can be used also in HMIPv6. In HMIPv6 the return routability procedure entrusts the CN that the MN which has a given HoA owns the RCoA at the moment. Without the return routability procedure, the CN would be posed to the same threats as in MIPv6.

Fast Handovers for Mobile IPv6 (FMIPv6)

Introduction

The Fast Handovers for Mobile IPv6 (FMIPv6) protocol is an extension of MIPv6. It is specialized by the Mobility for IP: Performance, Signaling and Handoff Optimization (mipshop) working group of IETF. The primary standard for FMIPv6 is RFC 4068, and it was issued in July 2005 [52].

Problem statement

During handover there is a period during which the MN is unable to send or receive packets. This handover latency is resulting from three main processes, i.e., movement detection (e.g. by some link-layer change or routing advertisement), new CoA configuration, and the Binding Update procedure of MIPv6. This is latency is often unacceptable to real-time applications, such as VoIP. This specification addresses the following problems:

- How to allow a mobile node to send packets as soon as it detects a new subnet link?

- How to deliver packets to a mobile node as soon as its attachment is detected by the new access router?

Idea

The idea of this solution is that the mobile node gathers the necessary information on the neighboring networks before moving, and tries to auto-configure with a new CoA (NCoA). Moreover the MN does not wait passively for necessary information: it always acts actively, so e.g., in the new network, it sends immediately a Fast Neighbor advertisement to the access router, to signal that the NCoA is active. The solution eliminates the latency caused by router discovery, the new CoA configuration, and that the new access router becomes aware of the active usage of NCoA.

Besides, the solution makes possible continuous data transfer between the MN and CNs before the standard Binding Update process of MIPv6 registers NCoA to the HA and to the CNs in case of route optimization. So the interruption caused by the Binding Update process is eliminated. To achieve this, the MN sends a Fast Binding Update to the previous access router directly, just before moving, or indirectly, after moving, via the new access router. The successful FBU results in the verification of new CoA for duplication and a binding cache entry at the previous access router linking the previous CoA with the new CoA. After a successful Fast Binding Update, the MN changes to the NCoA, the previous access router tunnels all the packets destined to the previous CoA to the new CoA. In the opposite direction, the MN reverse tunnels all the packets through the previous access router. This bi-directional tunnel between the MN and the previous access router exists until the MIPv6 binding update process terminates.

The solution doesn’t eliminate the latency of movement detection. The CN, HA and MN functionalities are the same as in MIPv6, the MN has extended functionalities.

The mechanism of the protocol

The participants of the protocol are the following:
Mobile Node (MN): the same as in MIPv6, with some extended functionalities.
Correspondent Node (CN): the same as in MIPv6.
Home Agent (HA): the same as in MIPv6.
Previous Access Router (PAR): the access router from which the MN moves.
Next Access Router (NAR): the access router to which the MN moves.

Preconditions of the protocol

The PAR should be informed about the neighbor access routers (candidate NARs for the MN). PAR has to know the AP-IDs (access point IDs), link-layer addresses, IP addresses and subnet prefixes of the NARs.

The phases of the protocol for a handover

Neighborhood discovery

This is a message exchange between the MN and the PAR, triggered by a link-specific event, or initiated after route discovery. The MN sends a Router Solicitation for Proxy Advertisement (RtSolPr) message to the PAR. The PAR replies with a Proxy router Advertisement (PrRtAdv). The reply contains (AP-ID, AR-info) tuples, which means that it contains the Access Point-ID (e.g., SSID), L2 address, IP address and the subnet prefixes of the available NARs. The result is that the MN collects information about its possible future visited networks. The MN generates a prospective new CoA for each NAR, based on the advertised subnet prefixes. Neighborhood discovery is illustrated in Figure 8.

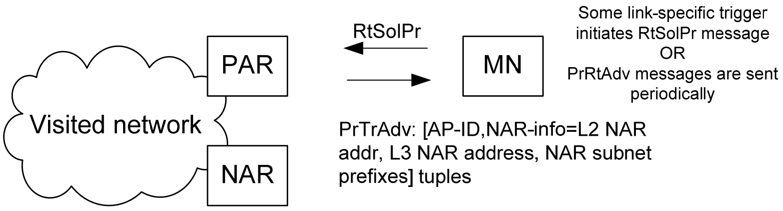
Visited network
PAR
NAR
RtSolPr
MN
Some link-specific trigger initiates RtSolPr message
OR
PrRtAdv messages are sent periodically
PrTrAdv: [AP-ID,NAR-info=L2 NAR addr, L3 NAR address, NAR subnet prefixes] tuples

Figure 8. FMIPv6 neighborhood discovery.

Fast Binding Update

It is triggered by a link-specific handover, just at moving. The MN tries to send a Fast Binding Update (FBU) message to the PAR, and waits for an FBAck from the PAR. If this succeeds, then a predictive handover happened (see later). If the MN does not get the FBAck, or is not able to send the FBU to the PAR because it moved away, then a Reactive handover happens (see later). In this case the FBU message is also sent to the PAR by the MN; but indirectly, through the NAR. When the PAR gets an FBU, firstly it must be persuaded that the NCoA propsed by the MN is unique in the subnet of NAR. In case of predictive handover, it exchanges two messages with the PAR, i.e. HI and HA, to know that the NCoA is unique. In case of reactive handover, the NAR only forwards the FBU to the PAR if the NCoA was unique. As a result if the Fast Binding Update process, the PAR creates a binding cache entry for the MN, linking PCoA to NCoA, and the MN changes its network address from PCoA to NCoA. Figure 9 and Figure 10 illustrate the fast binding update process in case of predictive and reactive handover.

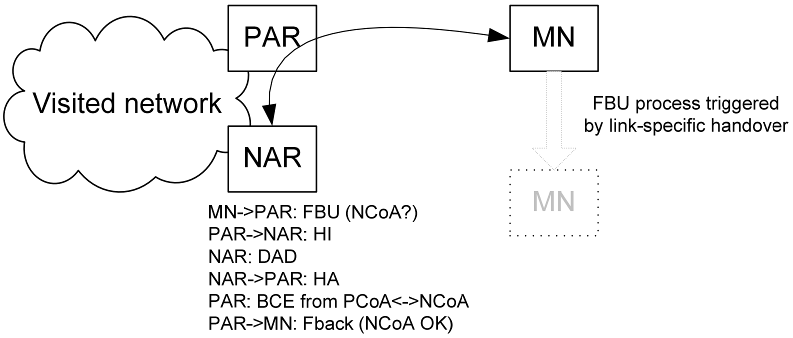

Figure 9. Fast binding update process in case of predictive handover.

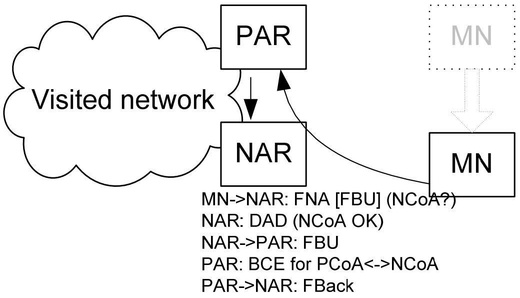

Figure 10. Fast binding update process in case of reactive handover.

Fast Neighbor Advertisement

When a MN moves to a new visited network, it sends immediately a Fast Neighbor Advertisement message to the NAR, in order to notify to it that the NCoA is in use by the MN. As a result the NAR will forward the packets from CNs to the MN's NCoA.

Tunneling

After Fast Binding Update, a tunnel between the MN and the PAR is established, so the dataflow from CNs to the PCoA is cached and tunneled from the PAR to the MN, and the data flow from the MN to CNs is tunneled to the PAR, to seem as the traffic came from the PCoA. This is illustrated in Figure 11. During this phase the MN initiates the MIPv6 Binding Update to the NCoA with its HA and in case of route optimization with all of its CNs. As soon as the Binding Update terminates, the tunneling between MN and PAR lapses.

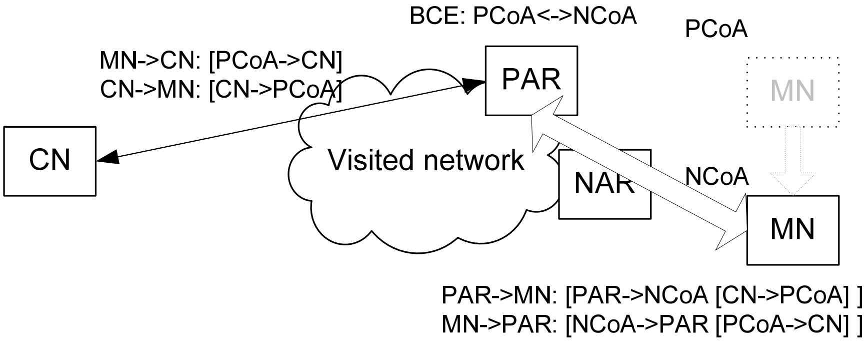

Figure 11. Tunneling of packet flow arriving to the PCoA through the NCoA in FMIPv6.

Predictive Fast Handover

This is the normal case that is primarily happens during handover. In this case the MN sends the Fast Binding Update (FBU) message to the PAR, and receives a Fast Binding Acknowledgment (FBack) from the PAR. The FBU contains the NCoA that the MN proposes to use after handover. In the background, before sending back the FBack, the PAR exchanges a Handover Initiate (HI) and a Handover Acknowledge (HA) message with the NAR. The NAR runs a Duplicate Address Detection (DAD) mechanism to check the uniqueness of NCoA. If the NCoA proposed by the MN collides with an already used address, then the NAR sends back another NCoA in the HA message, and the PAR sends back the new NCoA to the MN in the FBack message. After a succesfull Fast Binding Update the PAR generates a binding cache entry for the PCoA and NCoA pair, and the MN changes to use the NCoA. However, the MN can not use immediately this address. Firstly, the NAR has to know, that the NCoA address is used. This is achieved by the fact, that the MN sends immediately a Fast Neighbor Advertisement (FNA) in the new network, informing the NAR about the active usage of the NCoA. Secondly, the CNs won't know that the data sent from the NCoA is originating from the same node than those from the PCoA. This state holds until a MIPv6 Binding Update procedure will be done. So due to the Fast Binding Update procedure, the packets sent to the PCoA of the MN are tunneled to the NCoA by the PAR. The PAR also caches the packets until the address change happens. And the MN reverse tunnels the packets through the PAR, so the CN will not see the change to NCoA. After the MIPv6 BU, the packets are no more sent to the PCoA and are not tunneled through the PAR. The reactive handover is illustrated in Figure 12.

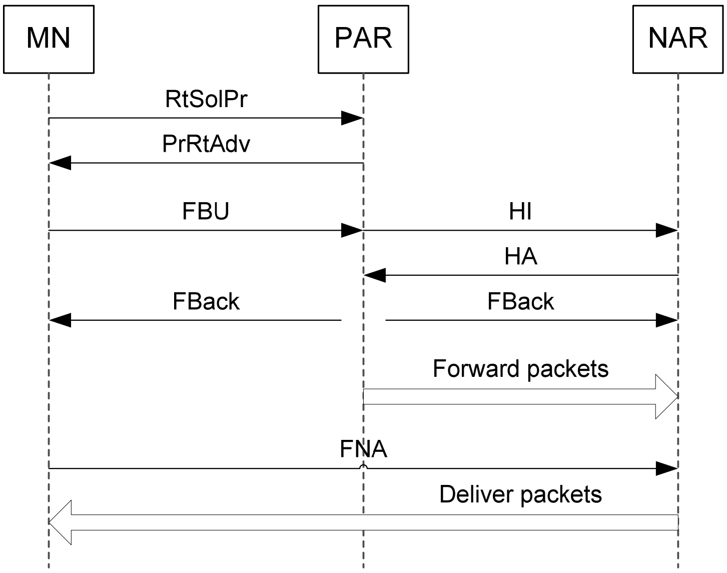

Figure 12. Predictive Handover in FMIPv6.

Reactive Fast Handover

In the Reactive Fast Handover, the MN does not get an FBack for the FBU from the PAR, or even, the MN is not able to send the FBU to the PAR. So the MN moves to the new visited network and picks the NCoA that it calculated for itself, without DAD. The MN sends an FNA to the NAR, to notify it about the use of the new address. Moreover it involves an FBU for the PAR in the FNA message. If the NCoA address is unique, then the NAR sends the FBU to the PAR to run the fast binding update process, and thus, to create a binding cache entry for the MN at the PAR. Then the tunneling of dataflow to the PCoA is used in the way as described before, until the MIPv6 binding Update process terminates. The reactive handover is presented in Figure 13.

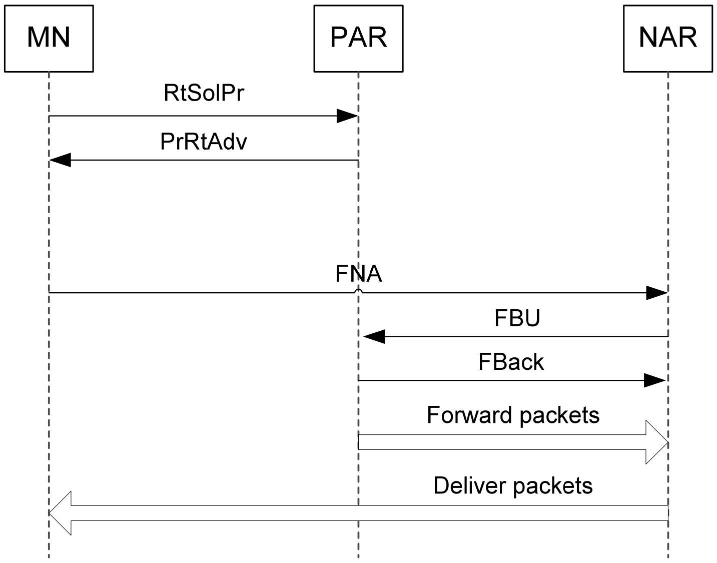

Figure 13. Reactive Handover in FMIPv6.

Main threats of the protocol

Malicious Fast Binding cache entries in PAR: if the FBUs are not protected, the traffic to honest MNs can be redirected to an attacker or to an unsuspecting node or network. All threats which are similar to the MIPv6 Binding update related threats can be imagined, if the FBU is not protected. The standard (RFC 4068) proposes that the origin of the FBUs should be protected by checking at the PAR that the FBU contains a well-formed PCoA. More generally speaking, a mechanism is needed to check that the FBU came from a node that legitimately owns the PCoA. A simple PCoA verification, either it is from the subnet of PAR or not, is not enough for this. The standard also proposes that the FBU may restrict FBUs from L2 addresses, which are in the router's neighbor cache. This would limit the attacker to send FBU messages with spoofed L2 origin addresses.

Malicious selection of NCoA: The attacker may send an FBU which binds the subsequent traffic to an NCoA of an unsuspecting node. However, the PAR and NAR are in trust relationship by the assumptions of the protocol, and they can detect duplicate addresses.

Threats on the communication between PAR and NAR: The messages between PAR and NAR must be protected because this is the way which assures DAD for NCoAs, and which leads to the creation of binding cache entries in PARs. In case of Reactive Fast Handover, the Fast Binding Update message could be fabricated to make false binding cache entries in PARs, thus redirecting traffic.

7. Security solutions for IPv6 based mobile networks

Solutions for all multihoming networks

In general, the aim of securing multihoming solutions is to tie the applied security mechanisms to the identifier of the multihoming node, and not to the locators. The multihoming node should be authenticated based on its identifier. However, this mechanism should not be based on checking previously used locators. Sometimes it is also an aim to check, whether the given identity is really at the claimed location.

Solutions for MIPv6

The security goal defined at the design of Mobile IPv6 was to provide a solution as secure as the non-mobile IPv4 Internet. Traditional IPv4 gives little protection against on-the-path attackers; as a consequence, on-the-path threats, such as disruption, modification, interception remain a residual risk, unless IPSec is used. Still, in case of using IPSec, disruption, denial of service, and redirection of flows are possible.

Problems with plain IPSec solution

Early in the MIPv6 design process, it was assumed that plain IPSec could be the default way to secure Binding Updates with arbitrary correspondent nodes. However, this turned out to be impossible. Plain IPSec relies on an infrastructure for key management, which to be usable with any arbitrary pair of nodes, would need to be global in scope. Such a global PKI does not exist, nor is it expected to come into existence any time soon. More minor issues that also surfaced at the time were: (1) an insufficient filtering granularity for the state of IPSec at the time, (2) the cost to establish security association in terms of CPU and roundtrip times, and (3) expressing the proper authorization for binding updates. In case of issue (3), it is not enough to authenticate just the identity, but also, to bind the identity to the localization (i.e., current care-of address) in a trusted and verifiable way for the CN.

The issues (1) and (3) were addressed between the HA and MN in RFC 3776 [54]. However the lack of global PKI remains unsolved.

One way to provide global key infrastructure for mobile IP could be DNSSEC or Secure Neighbor Discovery. These infrastructures are currently worked out. The idea of these architectures is to provide a public certificate for each IP address and sign the binding update by the node having that IP address. However, in order to be secure, each link in such a system must be secure. There must be a chain of keys and signatures all the way down from the root (or at least the common trust anchor of the MN and the CN) to the given IP address. And each signature should explicitly authorize the lower key to manage the corresponding address below. Checking all the signatures on the tree would place a considerable burden on the CN, making route optimization prohibitive, or justifiable only in very particular circumstances. Consequently the obvious question is whether the costs of deploying the global secure DNS infrastructure is worth the additional protection it affords, as compared to simply using return routability for both home address and care-of address verification.

The return routability mechanism is the current security solution in Mobile IPv6 route optimization. It was designed to mitigate the threats discussed in the previous section. The protection level of return routability is close to that of a static IPv4-based Internet. It produces an acceptable cost in terms of packets, delay, and processing. The aim of return routability mechanism is to check, whether the MN is reachable both by home address and care-of address. The check yields false positives if the routing infrastructure is compromised or if there is an on-the-path attacker between the CN (verifier) and the address to be verified (CoA). With these exceptions, it is assumed that a successful reply indicates that there is indeed a node at the given address, and that the node is willing to reply to the probes sent to it.

The basic return routability mechanism consists of two checks, a Home Address check and a Care-of address check. These checks are running independently and paralelly. The MN initiates the home address test with a Home Test Init message, and the care-of address test with a Care-of Test Init message. The Home Test Init goes through the HA, the Care-of Test Init goes directly to the CN. Then the CN sends two challenges, as a reply for each request, on the same paths where the init messages came. This prevents reflection and amplification attacks. (Note that with fake routing advertisements, IP origin spoofing, these attacks are still possible.) The first challenge is the Home Test, the other challenge is the Care-of Test message. These messages contain a token, which are a

Protecting home registration

The home registration is the binding update process between the MN and the HA. This is needed for basic mobility support. The standard recommends the usage IPSec extension headers and the Encapsulating Security Protocol (ESP) protocol in transport mode to protect the MIPv6 signaling between the MN and the HA. It must use at least a non-null authentication algorithm which provides data origin authentication, connectionless integrity and optional anti-replay protection. The basic mobility support standard directs the reader to RFC 2406 [58], which describes in details the IPsec ESP protocol. Besides ESP, Authentication Header (AH) protocol could also be used to authenticate the messages between the MN and the HA. This solution is described in RFC 2402 [57].

The MN and HA must establish two Security Associations (SAs), one in each direction. The key management for the SAs can be done in the following ways:

- Static key distribution: keys are distributed off-line, for each MN-HA relation. This must be supported. If Internet Key Exchange protocol version 1 (IKEv1) is used with pre-shared authentication key, then it must be used in aggressive mode.

- Internet Key Exchange (IKE): dynamic key management may also be supported, in a way as described in RFC 2409 [59]. IKE phase 1 credentials must be recognized, (by SPD or MIPv6 processing), to be able to create a new SA in phase 2. If phase 1 identity is FQDN, then secure DNS may be used to trustfully resolve the IP address. MIPv6 is carefully designed to not to send BU before IKE exchange (see 11.3.2 in RFC 3775).

The details of protecting signaling between the MN and the HA communication will be described later (based on the standard RFC 3776).

Protecting correspondent registration

MIPv6 supports route optimization (RO) to bypapss the HA-MN tunnel and to make a direct communication between the MN and the CN. To achieve this, the CN takes part in mobility management, i.e., it registers routing exceptions for the MN to source route the packets sent originally to the HoA to the CoA. The routing exceptions are stored in the binding cache of the CN for a short life-time. In order to countermeasure the threats regarding and originating from malicious binding updates, the binding update procedure must be protected. In fact, the MN has to provide correct authorization data, which can be obtained via the Return Routability (RR) procedure. The RR is run before sending the BU to the CN. The RR procedure is part of the RO. The RR results in the generation of binding management key (Kbm) at the MN and CN. The key is then used to generate a Hashed Message Authentication Code (HMAC), i.e., HMAC_SHA1, for the authentication and integrity verification of BUs sent by the MN. The HMAC is sent within the BU, and checked by the CN. After successful verification a Binding Acknowledgment (BAck) is sent back to the MN, and the Binding Cache Entry is created in the CN. The message flow and computations of the route optimization mechanism are illustrated in Figure 14.

Return Routability procedure

The RR procedure gives assurance to the CN that the right MN is sending a BU. The RR checks that the locator (CoA) where the Binding Update comes from is really possessed by the claiming identity (HoA). RR does not protect against on-the-path attackers.

The elements of RR are the following:

- Secret key of the CN (Kcn): it is used internally by the CN to produce nonces for the MN.
- Nonces: fresh nonces are generated in given intervals by the CN. Nonces are stored locally and internally by the CN. The CN also maintains indices for the nonces, and send these indices within the “test” messages. The MN sends back the indexes in the replies. Nonce indices are for indicating cases when the CN refreshes nonce and still gets messages having HMAC signature calculated with previous nonces. Nonces and nonce indices change also when the CN refreshes Kcn.

- Keygen tokens: the CN generate two keygen tokens based on Kcn and the fresh nonce. It uses a HMAC_SHA1 algorithm to compute them. The keygen tokens do not have to be stored locally in the CN, because they can be recalculated when needed. The keygen tokens are sent in two different ways, i.e., directly and indirectly through the HA to the MN. If the MN receives them, it can generate the Binding Management Key (Kbm) used to authenticate the Binding Update.
- Cookies: The home init cookie is sent by the MN to CN in Home Test Init. The care-of init cookie is sent in the Care-of test init message from the MN to the CN. Cookies ensure that parties, who have not seen the requests from the MN, can not spoof replies to the MN.

Aim of RR procedure is that the CN obtain some assurance about that the MN is addressable with its CoA and HoA. It accepts BU only at this case. Instructs the CN to direct it traffic from HA to CN directly.

The RR tests whether the Home Test and Care-of Test messages, addressed to the CoA and HoA are finally routed to the same MN. MN must give the proof to have received the tokens in the two parallel messages. The MN has to combine the two tokens into one Kbm, and use Kbm to sign the BU.

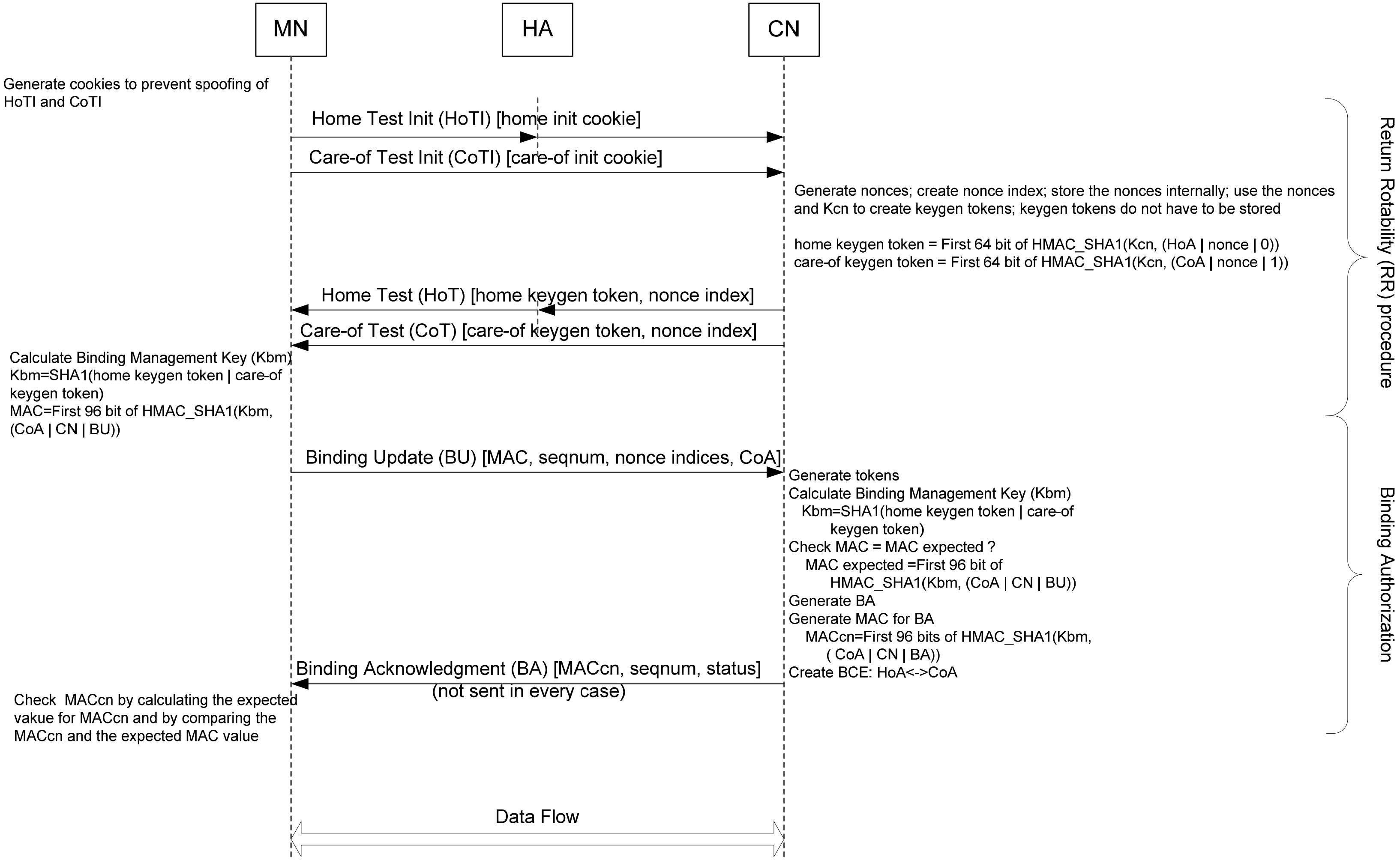
MN
HA
CN
Generate cookies to prevent spoofing of HoTI and CoTI
Home Test Init (HoTI) [home init cookie]
Care-of Test Init (CoTI) [care-of init cookie]
Generate nonces; create nonce index; store the nonces internally; use the nonces and Kcn to create keygen tokens; keygen tokens do not have to be stored
home keygen token = First 64 bit of HMAC_SHA1(Kcn, (HoA | nonce | 0))
care-of keygen token = First 64 bit of HMAC_SHA1(Kcn, (CoA | nonce | 1))
Home Test (HoT) [home keygen token, nonce index]
Care-of Test (CoT) [care-of keygen token, nonce index]
Calculate Binding Management Key (Kbm)
Kbm=SHA1(home keygen token | care-of keygen token)
MAC=First 96 bit of HMAC_SHA1(Kbm, (CoA | CN | BU))
Binding Update (BU) [MAC, seqnum, nonce indices, CoA]
Generate tokens
Calculate Binding Management Key (Kbm)
Kbm=SHA1(home keygen token | care-of keygen token)
Check MAC = MAC expected ?
MAC expected =First 96 bit of HMAC_SHA1(Kbm, (CoA | CN | BU))
Generate BA
Generate MAC for BA
MACcn=First 96 bits of HMAC_SHA1(Kbm, (CoA | CN | BA))
Create BCE: HoA<->CoA
Binding Acknowledgment (BA) [MACcn, seqnum, status]
(not sent in every case)
Check MACcn by calculating the expected vakue for MACcn and by comparing the MACcn and the expected MAC value
Data Flow
Return Rotability (RR) procedure
Binding Authorization

Figure 14. Route optimization in MIPv6 containing the RR process and the binding authorization of the MN.

Home Address destination option

The Home Address destination option is used in packets sent from the MN to the CN, or from the MN to the HA. The Home Address destination option contains the HoA of the MN. The usage of this field is restricted to the destination address, i.e., the CN or the HA. The CNs uses this field to present the HoA of the MN for the upper-layers. HA uses it to check if the packet was tunneled by a registered MN. The management of the Home Address destination option field is defined by MIPv6 in a way to restrict the threats, e.g. the reflection attacks. The strict usage restrictions of this field aim to protect the Home Network, HA, HoA, and consequently the MN (e.g. prevention of reflection attacks).

Type 2 routing header

The processing of routing header of type 2 is restricted to the MN. It contains generally the HoA of the MN, so a packet arriving to the CoA is “routed” to the HoA within the MN. Consequently, the upper-layer protocols in the MN see that packets were destined to sockets using the HoA. It is important, that normal routers on-the-path do not use the HoA as a routing destination address. This explains the introduction of type 2 routing header and its restricted use. Normally, routing within the MN could have been solved by normal routing headers, but in that case, if a wrong HoA would be given, the packets may end up in unsuspicious nodes or networks. Hence normal routing headers would make possible reflection attacks.

8. Summary of security solutions

The state-of-the-art security solutions for IPv6 based mobility services can be summarized as follows. When a previous relationship, business contract can be established between the parties, then IPSec SAs can be used to protect the authenticity, integrity and if encryption is also used then confidentiality of the signaling. These security tunnels will also protect payload data. However IPSec is no more usable between parties who do not preliminary know each other, because they may difficultly provide verifiable certificate of their identity and public keys for the other party. The IPSec has also scalability problems due to the limitations of PKI infrastructure in the real world. Moreover, in case of Route Optimization, the traditional IPSec solution would require certificate for authenticating the MN at a given CoA. It could e very costly to generate certificates for the MN at each CoA.

There are drafts of the mip6 Working Group dealing with the dynamic SA establishment between the MN and the HA. Firstly the MN communicates securely with an AAA server, resulting in the authentication of the MN and possibly the AAA server and in the generation of the cryptographic key material at the MN and the AAA. Then the AAA sends the keys to the HA. This is called the bootstrap phase of the MN in a home network. There exist different scenarios for the different possibilities for which parts relate to the same administrative domain.

Typically we do not want to make restrictions on the CN, i.e., it is supposed to belong to a foreign administrative domain. However, the binding update procedure at the CN has also to be protected. The Return Routability procedure in the route optimization makes possible to check, whether the Binding Update came from really a MN that is reachable through the claimed CoA at the moment. Return Routability protects only against off-the-path attackers with the use of cryptographic tokens sent in plaintext as challenges for the MN. Anybody who gets these tokens could easily masquerade as the MN and fabricate fake Binding Cache Entries in the CN.

Ingress filtering is used at the routers to filter egress traffic with spoofed source IP addresses. Ingress filtering reduces the potential packet injection possibilities.

Each IPv6 mobility standard achieves some part of protection by careful routing policies and header field verifications at the HA, MR, MN and CN. Routing policy rules limit where to propagate routing information. The careful check of source addresses and destination addresses in the original and the external IPv6 header, moreover the checking of Home Address destination option and routing header type 2 at the parties in different phases of the binding update process, the data transfer, the tunneling are the main tools to protect the participants and other nodes and networks from potential attacks.

An alternative method for securing MIPv6 signaling between the MN and HA is proposed by the Network Working Group in RFC 4285 [55]. It introduces new mobility message authentication options to authenticate the control message between the MN and the HA or a home AAA server (as an alternative to IPSec). The new options provide for authentication of BU and Back messages. The standard also proposes an optional replay protection.

9. References

[1] Peter G. Neumann, *Computer-Related Risks*, ACM Press / Addison-Wesley Professional, 1995, ISBN 0-201-55805-X, 384pp. Paperback.

[2] CP & SL Pfleeger, *Security in Computing*, 3rd International edition, Prentice Hall PTR, Upper Saddle River, N.J, 2003.

[3] Matt Bishop, *Introduction to Computer Security*, Addison-Wesley, 2004, ISBN 0-321-24744-2, Hardcover.

[4] Frank Swiderski and Window Snyder, *Threat Modeling*, Microsoft Press, 2004, ISBN 0-7356-1991-3, Paperback.

[5] Mehmet Sahinoglu, "Security Meter: A Practical Decision-Tree Model to Quantify Risk," *IEEE Security and Privacy*, vol. 03, no. 3, pp. 18-24, May/Jun, 2005.

[6] B. Schneier, "Attack trees: Modeling security threats," *Dr. Dobb's Journal. Software Tools*, vol. 24, no. 12, pp. 21-29, December 1999.

[7] Stuart E. Schechter, "Toward Econometric Models of the Security Risk from Remote Attack," *IEEE Security and Privacy*, vol. 03, no. 1, pp. 40-44, 2005.

[8] Salim Hariri, Guangzhi Qu, Tushneem Dharmagadda, Modukuri Ramkishore, Cauligi S. Raghavendra, "Impact Analysis of Faults and Attacks in Large-Scale Networks," *IEEE Security and Privacy*, vol. 01, no. 5, pp. 49-54, , 2003.

[9] Axelle Apvrille, Makan Pourzandi, "Secure Software Development by Example," *IEEE Security and Privacy*, vol. 03, no. 4, pp. 10-17, Jul/Aug, 2005.

[10] Peter Torr, "Demystifying the Threat-Modeling Process," *IEEE Security and Privacy*, vol. 03, no. 5, pp. 66-70, Sept/Oct, 2005.

[11] David M. Nicol, "Modeling and Simulation in Security Evaluation," *IEEE Security and Privacy*, vol. 03, no. 5, pp. 71-74, Sept/Oct, 2005.

[12] Helayne T. Ray, Raghunath Vemuri, Hariprasad R. Kantubhukta, "Toward an Automated Attack Model for Red Teams," *IEEE Security and Privacy*, vol. 3, no. 4, pp. 18-25, Jul/Aug, 2005.

[13] Andrew P. Moore, Robert J. Ellison, Richard C. Linger, “Attack Modeling for Information Security and Survivability”, Technical Note, CMU/SEI-2001-TN-001, March, 2001. URL: http://www.cert.org/archive/pdf/01tn001.pdf , (checked at 28 September, 2006).

[14] B. Wood, “An Insider Threat Model for Adversary Simulation”, *Proc. 2nd Workshop Research with Security Vulnerability Databases*, SRI Int’l, 1999.

[15] Dimitrios Lekkas, Diomidis Spinellis, "Handling and Reporting Security Advisories: A Scorecard Approach," *IEEE Security and Privacy*, vol. 03, no. 4, pp. 32-41, Jul/Aug, 2005.

[16] C. E. Landwehr, David M. Goldschlag, “Security Issues in Networks with Internet Access,” *Proceedings of the IEEE.* Vol.85, No.12 (December 1997) 2034-2051.

[17] J. Steffan and M. Schumacher, “Collaborative Attack Modeling,” *Proc. 17th ACM Symp. Applied Computing (SAC 2002)*, ACM Press, 2002, pp. 253-259.

[18] Kanta Jiwnani, Marvin Zelkowitz, "Susceptibility Matrix: A New Aid to Software Auditing," *IEEE Security and Privacy*, vol. 02, no. 2, pp. 16-21, , 2004.

[19] Shelby Evans, David Heinbuch, Elizabeth Kyule, John Piorkowski, James Wallner, "Risk-based Systems Security Engineering: Stopping Attacks with Intention," *IEEE Security and Privacy*, vol. 02, no. 6, pp. 59-62, , 2004.

[20] Peder Jungck, Simon S.Y. Shim, "Issues in High-Speed Internet Security," *IEEE Computer Society*, vol. 37, no. 7, pp. 36-42, Jul., 2004.

[21] Hao Yang, Fabio Ricciato, Songwu Lu, and Lixia Zhang, “Securing a Wireless World,” Proceedings of the IEEE, Vol. 94, No. 2, pp. 442-454, February 2006.

[22] Gerald A. Marin, "Network Security Basics," IEEE Security and Privacy, vol. 3, no. 6, pp. 68-72, Nov/Dec, 2005.

[23] L. Zhou and Z. J. Haas, “Securing Ad Hoc Networks,” *IEEE Network Magazine*, vol. 13, no.6, November/December 1999.

[24] Thomas J. Walsh, D. Richard Kuhn, "Challenges in Securing Voice over IP," IEEE Security and Privacy, vol. 03, no. 3, pp. 44-49, May/Jun, 2005.

[25] Marco Gruteser, Xuan Liu, "Protecting Privacy in Continuous Location-Tracking Applications," IEEE Security and Privacy, vol. 02, no. 2, pp. 28-34, , 2004.

[26] J.M. Park, E.K.P. Chong, H.J. Siegel, “Efficient multicast stream authentication using erasure codes,” *ACM Transactions on Information and System Security*, Vol. 6 No. 2, pp. 258-285, May, 2003.

[27] Richard Ford, "Malcode Mysteries Revealed," *IEEE Security and Privacy*, vol. 03, no. 3, pp. 72-75, May/Jun, 2005.

[28] M. Cai, K. Hwang, Y.-K. Kwok, S. Song, and Y. Chen, "Collaborative Internet worm containment," *IEEE Security and Privacy Magazine*, vol. 3, no. 3, pp. 25–33, May/June 2005.

[29] D. Dolev, A. C. Yao, "On the security of public key protocols ," *IEEE Transactions on Information Theory*, vol. 29, no. 2, pp. 198-208, 1983.

[30] M. Bakes, B. Pfitzmann, and M. Waidner, "A universally composable cryptographic library", IACR Cryptology ePrint Archive 2003/015, Jan. 2003. URL: http://citeseer.ist.psu.edu/backes03universally.html (checked at 29 September, 2006)

[31] Sean Barnum, Gary McGraw, "Knowledge for Software Security," IEEE Security and Privacy, vol. 03, no. 2, pp. 74-78, Mar/Apr, 2005.

[32] Tuomas Aura, Pekka Nikander, Gonzalo Camarillo, "Effects of Mobility and Multihoming on Transport-Protocol Security," sp, p. 12, 2004 IEEE Symposium on Security and Privacy, 2004.

[33] M. Burrows, M. Abadi, R. Needham, "A logic of authentication, " ACM Transaction on Computing Systems, vol. 8, no. 1, pp.18-36, 1990.

[34] L. Gong, R. Needham, R. Yahalom, "Reasoning about belief in cryptographic protocols", Proceedings of the 1990 IEEE Computer Society Symposium on Research in Security and Privacy , Oakland, CA, IEEE Computer Society Press, 7-9 May 1990, pp. 234-248.

[35] P. Bieber, "A logic of communication in a hostile environment", Proceedings of IEEE Computer Security Foundations Workshop III , Los Alamitos, CA, IEEE Computer Society Press, 12-14 June 1990, pp. 14-22.

[36] M. Abadi, M. R. Tuttle, "A Semantics for a logic of authentication", Proceedings of the 10th ACM Symposium on Principles of Distributed Computing , ACM Press, August 1991, pp. 201-216.

[37] P. Syverson, P. C. van Oorschot, "On unifying some cryptographic protocol logics", Proceedings of the 1994 IEEE Computer Society Symposium on Research in Security and Privacy , May 1994, pp. 14-28.

[38] G. Wedel, V. Kessler, "Formal semantics for authentication logics", Computer Security -- ESORICS'96 , Rome, Italy, Springer-Verlag, Septemper 1996, pp. 219-241.

[39] Pekka Nikander, "Modelling of cryptographic protocols", Licenciate's thesis, Helsinki University of Technology, December 1997.

[40] M. Matsui, "Linear Cryptanalysis Method for DES Cipher", Advances in Cryptology - EUROCRYPT '93 (Lecture Notes in Computer Science no. 765), Springer-Verlag, pp. 386-397, 1994.

[41] E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Journal of Cryptology, vol. 4, no. 1, pp. 3-72, 1991.

[42] E. Biham and A. Shamir, Differential Cryptanalysis of the Data Encryption Standard, Springer-Verlag, 1993.

[43] Howard M. Heys, *A Tutorial on Linear and Differential Cryptanalysis*, URL: http://www.engr.mun.ca/~howard/Research/Papers/ldc_tutorial.html

[44] J. P. McDermott, "Attack net penetration testing, " In Proceedings of the 2000 Workshop on New Security Paradigms (Ballycotton, County Cork, Ireland,

September 18 - 21, 2000). NSPW '00. ACM Press, New York, NY, 15-21. DOI= http://doi.acm.org/10.1145/366173.366183

[45] Dianxiang Xu, Kendall E. Nygard, "Threat-Driven Modeling and Verification of Secure Software Using Aspect-Oriented Petri Nets," IEEE Transactions on Software Engineering, vol. 32, no. 4, pp. 265-278, Apr., 2006.

[46] Guy Helmer, Johnny Wong, Mark Slagell, Vasant Honavar, Les Miller, Yanxin Wang, "Software Fault Tree and Colored Petri Net Based Specification, Design and Implementation of Agent-Based Intrusion Detection Systems, " Submitted to ACM Transactions on Information and System Security, 2000. URL: http://citeseer.ist.psu.edu/helmer01software.html

[47] Stefan Lindskog and Erland Jonsson: "Different Aspects of Security Problems in Network Operating Systems". In Proceedings of the Third Annual International Systems Security Engineering Association Conference (2002 ISSEA Conference), Orlando, Florida, USA, March 13-15, 2002.

[48] E. Nordmark, T. Li, "Threats Relating to IPv6 Multihoming Solutions, " RFC 4218, October 2005.

[49] D. Johnson, C. Perkins, J. Arkko, "Mobility Support in IPv6, " RFC 3775, June 2004.

[50] V. Devarapalli, R. Wakikawa, A. Petrescu, P. Thubert, "Network Mobility (NEMO) Basic Support Protocol, " RFC 3963, January 2005.

[51] H. Soliman, C. Castelluccia, K. El Malki, L. Bellier, "Hierarchical Mobile IPv6 Mobility Management (HMIPv6), " RFC 4140, August 2005.

[52] R. Koodli Ed., "Fast Handovers for Mobile IPv6, " RFC 4068, July 2005.

[53] R. Moskowitz, P. Nikander, "Host Identity Protocol (HIP) Architecture, " RFC 4423, May 2006.

[54] J. Arkko, V. Devarapalli, F. Dupont, "Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents, " RFC 3776, June 2004.

[55] G. Giaretta, I. Guardini, E. Demaria, J. Bournelle, R. Lopez, "AAA Goals for Mobile IPv6, " draft-ietf-miv6-aaa-ha-goals-03.txt, September 12, 2006.

[56] A. Patel, K. Leung, M. Khalil, H. Akhtar, K. Chowdhury, "Authentication Protocol for Mobile IPv6, " RFC 4285, January 2006.

[57] S. Kent, R. Atkinson, "IP Authentication Header, " RFC 2402, November 1998.

[58] S. Kent, R. Atkinson, "IP Encapsulating Security Payload (ESP), " RFC 2406, November 1998.

[59] D. Harkins, D. Carrel, "The Internet Key Exchange (IKE), " RFC 2409, November 1998.

[60] V. Devarapalli, F. Dupont, "Mobile IPv6 Operation with IKEv2 and the revised IPsec Architecture" draft-ietf-mip6-ikev2-ipsec-07.txt, October 22, 2006.

[61] E. Nordmark, T. Li, "Threats Relating to IPv6 Multihoming Solutions, " RFC 4218, October 2005.