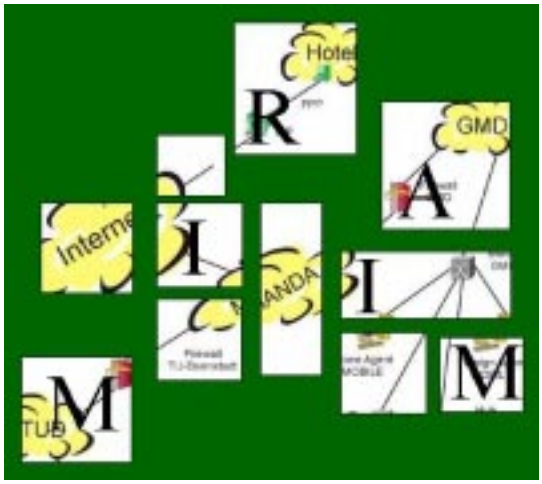
Projektbericht MIRIAM: Meilenstein 4

Mobile IP Referenzinstallation mit Entwicklung einer Arbeitsumgebung für den mobilen Wissenschaftler

Darmstadt, 28. Februar 2001



GMD - Forschungszentrum Informationstechnik GmbH

Institut für Integrierte Publikations- und Informationssysteme (IPSI)
Dolivostr. 15, 64293 Darmstadt

Technische Universität Darmstadt

Industrielle Prozess- und Systemkommunikation (KOM)
Merckstr. 25, 64283 Darmstadt

Allgemeines

Projektname:	Mobile IP Referenzinstallation mit Entwicklung einer Arbeitsumgebung für den Mobilen Wissenschaftler
Projektkürzel:	MIRIAM
Internet:	http://www.darmstadt.gmd.de/mobile/projects/miriam
Projektleitung:	Dipl.- Wirtsch.- Inf. Nicole Berier Tel. Nr.: 06151 / 869-817 E-Mail: berier@darmstadt.gmd.de
Teilnehmende Einrichtungen:	GMD-Forschungszentrum Informationstechnik GmbH, Institut für Integrierte Publikations- und Informationssysteme (IPSI), Doliviostrasse 15, 64293 Darmstadt Technische Universität Darmstadt, Fachgebiet für Industrielle Prozess- und Systemkommunikation (KOM), Merckstrasse 25, 64283 Darmstadt
Projektlaufzeit:	24 Monate: 1. März 1999 - 28. Februar 2001

Gliederung

Allgemeines 1
Gliederung 2
Abbildungsverzeichnis 4
Tabellenverzeichnis 5

1. Einleitung 6

2. Projektbeschreibung 6

2.1 Motivation 6
2.2 Projektziel 6
2.3 Arbeitspakete 7
2.3.1 Anwendungsszenario 7
2.3.2 Referenzinstallation 7
2.3.3 Testumgebung 7
2.3.4 Datensicherheit 7
2.3.5 Service Location Protocol 7
2.4 Meilensteine 8
2.4.1 Meilenstein 1 8
2.4.2 Meilenstein 2 8
2.4.3 Meilenstein 3 8
2.4.4 Meilenstein 4 8

3. Aktueller Stand von Mobile IP 9

3.1 Standardisierung 9
3.1.1 Internet-Drafts: 9
3.1.2 Request For Comments: 9
3.2 Implementierungen 10

4. Testumgebung 11

5. Testmechanismen 11

5.1 Testfolgen 11
5.1.1 Testfolgen der Schnittstellen der Instanzen von Mobile IP 11
5.1.2 Testfolgen der Instanzen unter Extrembedingungen 23
5.2 Testbögen 27
5.2.1 Heimatagent 27
5.2.2 Fremdagent 30
5.2.3 Mobiler Rechner 33
5.3 Paketgeneratoren 34
5.3.1 Registrierungspaketgenerator 34
5.3.2 Registrierungsantwortgenerator 36

6. Testergebnisse 38

7. Mobile IP und PPP 39

7.1 Motivation 39
7.2 PPP Allgemein 39
7.3 Architektur 40
7.4 Installation / Konfiguration 40
7.5 Registrierung eines Mobilen Knotens beim Fremdagenten 41
7.6 Testdurchführung 43

7.7 Zusammenfassung....43
8. Sicherheit....44
8.1 Mobile IP und Verfügbarkeit....44
8.1.1 Replay Angriff (Kontroll- und Datenebene)....45
8.1.2 Überflutung mit gefälschten Agent Advertisment Nachrichten (Kontrollebene)46
8.1.3 Getunneltes Ping Flooding (Datenebene - Tunnel)....48
8.1.4 IP Tunnelmissbrauch gegen Rechner im Heimatnetz....49
8.1.5 IP Tunnelmissbrauch gegen Rechner im Fremdnetz....51
8.1.6 Entführen des Tunnels....53
8.1.7 Angriffe auf das ARP Protokoll....53
8.1.8 Zusammenfassung Angriffe auf die Verfügbarkeit....55
8.2 Zusammenfassung Sicherheit in Mobile IP....55
9. Integration von SLP....57
9.1 Beschreibung....57
9.2 Implementierungen....57
9.3 Funktionen für SLP....58
9.4 Grundlegende und leistungsbezogene Tests....58
9.4.1 Auffinden von DA oder SA....58
9.4.2 Registrieren von Diensten....59
9.4.3 Deregistrieren von Diensten....59
9.4.4 Suche nach Servicetypen....59
9.4.5 Suche nach einem Dienst....60
9.5 Testdurchführung....60
9.6 Zusammenfassung....62
10. Ergebnisse....64
Referenzen....68

Abbildungsverzeichnis

Abbildung 1. Architektur Mobile IP und PPP 40
Abbildung 2. Erweiterte Schemazeichnung des Testrack 41
Abbildung 3. Erfolglose Registrierung eines Mobilen Knotens 42
Abbildung 4. Erfolgreiche Registrierung eines Mobilen Knotens 42
Abbildung 5. Schemazeichnung für einen Angriffspunkt einer Replay Attacke 45
Abbildung 6. Angriff durch gefälschte Agent Advertisment Pakete 46
Abbildung 7. Auslastung während eines Agent Advertisment Angriff 47
Abbildung 8. Senden von gefälschten Agent Advertisment Nachrichten 47
Abbildung 9. Aufbau eines getunnelten Ping IP Paket 48
Abbildung 10. Systemlast bei einem getunnelten Ping mit Überflutung 49
Abbildung 11. IP Tunnel als Angriffsweg im Heimatnetz 50
Abbildung 12. Aufbau eines getunnelten Ping IP Pakets, in das Heimatnetz 50
Abbildung 13. Einbringen eines IP Pakets in einen IP Tunnel 51
Abbildung 14. IP Tunnel als Angriffsweg im Fremdnetz 52
Abbildung 15. Aufbau eines getunnelten Ping IP Pakets, in das Fremdnetz 52
Abbildung 16. Man in the Middle Angriff mit hunt 53
Abbildung 17. Adressauflösungsprotokolle: ARP, RARP 54
Abbildung 18. ARP Request mit darauffolgendem ARP Reply 54
Abbildung 19. Gefälschte ARP Reply Pakete 55

Tabellenverzeichnis

Tabelle 1. Frei verfügbare Mobile IP Implementierungen 10
Tabelle 2. Testfolgen der Schnittstellen der Instanzen von Mobile IP 12
Tabelle 3. Testfolgen der Instanzen unter Extrembedingungen 23
Tabelle 5. Testbogen Heimatagent 29
Tabelle 6. Testbogen Fremdagenten 32
Tabelle 7. Testbogen Mobiler Rechner 34
Tabelle 8. Mögliche Sicherheit in Mobile IP 56
Tabelle 9. SLP Implementierungen 57
Tabelle 10. Basistests 61
Tabelle 11. Testszenarios 62
Tabelle 12. Leistungstests 62

1. Einleitung

Der vierte Meilenstein des Projekts Miriam baut auf den Ergebnissen der bisher erarbeiteten Meilensteine auf.

Themen und Ziele dieses Meilensteins sind:

- der aktuelle Stand von Mobile IP
- das Point-to-Point Protokoll (PPP) im Zusammenspiel mit Mobile IP
- die Testumgebung und Testergebnisse
- sowie einige Untersuchungen im Bereich Sicherheit und SLP

Abschließend wird eine Zusammenfassung und eine Beurteilung der Ergebnisse gegeben.

2. Projektbeschreibung

2.1 Motivation

Mobile IP (MIP) wurde von der Internet Engineering Task Force (IETF) entwickelt und 1996 als Proposed Standard [10] veröffentlicht. Es setzt auf dem IP Protokoll auf und ermöglicht die Mobilität im Internet auf der Basis von IPv4. Mobile IP erlaubt es mobilen Rechnern, ihren Zugangspunkt zum Internet zu verändern, wobei sie ihre ursprüngliche Identität, welche durch die IP Adresse festgelegt wird, beibehalten. Derzeit befinden sich sowohl frei erhältliche als auch kommerzielle Produkte auf dem Markt. Es lässt sich jedoch feststellen, dass trotz dramatischer Verbesserungen von Notebooks bezüglich Größe, Gewicht und Leistungsfähigkeit sowie der wachsende Bedeutung von Netzwerken, insbesondere des Internets, Mobile IP noch nicht weit verbreitet ist. Es befindet sich immer noch in der Testphase. Ein Grund für die gerine Verbreitung liegt vor allem an der Frage nach der Notwendigkeit von Mobile IP, die sich für viele potenzielle Anwender stellt. Als Beispiel dienen hier oft die weitverbreiteten Internetdienste, wie der Zugriff zum World Wide Web (WWW) oder der Mail Dienst. Diese Dienste benötigen keine Mobilitätsunterstützung auf der Netzwerkschicht durch Mobile IP, da die Mobilität auf der Anwendungsebene unterstützt wird. Darüber hinaus lassen die vielen ungelösten Sicherheitsaspekte insbesondere Netzwerkadministratoren noch zögern.

2.2 Projektziel

Ziel des Projektes ist daher der Aufbau einer Referenzinstallation für Mobile IPv4 an den Instituten IPSI und KOM. Grundlage hierfür ist das im Projektvorschlag skizzierte Anwendungsszenario (siehe Abschnitt 2.3.1 Anwendungsszenario), in dem Wissenschaftler die Referenzinstallation innerhalb eines Feldversuches benutzen werden. Durch die Nutzung von Mobile IP sollen Erfahrungen gesammelt werden, die dann an die Mitglieder des DFN Vereins als potenzielle Anwender weitergegeben werden können. Hierbei sollen unter anderem Kenntnisse von Installation, Betrieb und Wartung von Mobile IP gewonnen werden. Darüber hinaus sollen spezielle mobile Arbeitsumgebungen untersucht werden, die die Mobile IP Referenzinstallation verwenden und benötigen. Weiterhin sollen Alternativen für die Implementierung untersucht, sowie Anwendungen für Mobile IP angepasst werden. Auch sollen im Zusammenhang mit Mobile IP die

Gebiete Dienstidentifizierung und -bereitstellung (Service Location Protocol, SLP) und Datensicherheit untersucht und Lösungsansätze erarbeitet werden.

2.3 Arbeitspakete

Im Folgenden werden die verschiedenen Arbeitspakete, die in MIRIAM untersucht werden, vorgestellt.

2.3.1 Anwendungsszenario

In diesem Arbeitspunkt geht es um die Implementierung und den Betrieb des Szenarios, die Evaluierung in Bezug auf Bedienung, Installation und Wartbarkeit und schließlich die Akzeptanz der gesamten Lösung. Weiterhin müssen spätere Vorführungen des Szenarios sowie eine Anwenderunterstützung vorbereitet werden.

2.3.2 Referenzinstallation

Dieses Paket umfasst den Aufbau der Infrastruktur sowie die Installation und Anpassung von Software. Als mögliche Betriebssystemalternativen, für die Mobile IP Implementierungen verfügbar sind, werden FreeBSD, Linux, Win95/-98 oder WinNT diskutiert. Die Installation und Konfiguration wird erst im Festnetz und dann im Funknetz vorgenommen.

2.3.3 Testumgebung

Dieses Arbeitspaket wurde im Laufe des Projektes neu definiert und ist somit nicht in dem ursprünglichen Antrag enthalten. Es umfasst den Aufbau einer möglichst effizienten Testumgebung für Mobile IP. Hierfür wurde innerhalb eines privaten Testsegmentes die Möglichkeit geschaffen auf eine flexibel Art und Weise unterschiedliche Topologien zu erzeugen. Zusätzlich wurden Testfolgen konzipiert und entwickelt, die ein systematischen Testen der vorhandenen Mobile IP Implementierungen innerhalb dieser Topologien ermöglichen.

2.3.4 Datensicherheit

Dieses Paket beschäftigt sich mit Algorithmen der Authentifizierung und Verschlüsselung zur Wahrung der Authentizität und Vertraulichkeit der übertragenen Daten. Existierende Implementierungen, die innerhalb der Referenzinstallation eingesetzt werden können, werden diskutiert. Darüber hinaus soll die Wirksamkeit und Durchlässigkeit von bestehenden Firewall-Techniken, wie z. B. Ingress Filtering, in Bezug auf die Funktionsweise von Mobile IP untersucht werden.

2.3.5 Service Location Protocol

Dieses Paket sieht die Integration des Service Location Protocol (SLP) in die MIRIAM Referenzinstallation vor. Zu untersuchen sind Mechanismen zur Registrierung, Deregistrierung und zur Abfrage von Diensten. Eine mögliche Lösung für den Einsatz von SLP in Mobile IP soll gegeben werden.

2.4 Meilensteine

2.4.1 Meilenstein 1

Meilenstein 1 sah vor, eine Infrastruktur aufzubauen und auf ihr Mobile IP zu installieren. Das Funknetz wurde hierbei noch nicht berücksichtigt. Weiterhin ist für das Anwendungsszenario ein Detailentwurf erfolgt. Die Technologien, die der Referenzinstallation zugrunde liegen, sollten auf einer geeigneten deutschen Tagung vorgestellt werden.

2.4.2 Meilenstein 2

Meilenstein 2 sah die Integration der Funknetze in die in Meilenstein 1 erarbeitete Infrastruktur vor. Weiterhin wurde eine allgemeine Beschreibung von Mobile IP und dessen Einsatz in dem Anwendungsszenario gegeben. Darüber hinaus wurde eine Demonstration der Referenzinstallation mit ersten Ergebnissen, sowie der Aufbau des Anwendungsszenarios auf einer geeigneter Veranstaltung (Oktober 1999, GMD: Tag der offenen Tür) präsentiert.

2.4.3 Meilenstein 3

Im Meilenstein 3 wurden die Ergebnisse des Arbeitspaketes Datensicherheit in einem Bericht und zusätzlich erste Testergebnisse für die vorher beschriebenen Testfolgen präsentiert. Weiterhin wurden die Arbeitspakete um die Themen “Sicherheit” und “Integration von SLP” erweitert.

2.4.4 Meilenstein 4

In Meilenstein 4 soll zum Abschluss ein aktueller Stand von Mobile IP, eine Zusammenfassung sämtlicher Projektergebnisse und eine abschliessende Bewertung gegeben werden. Eine Aussage über das im ersten Meilenstein definierte Szenario Hotel soll ebenfalls getroffen werden. Ein komplette Übersicht über die durchgeführten Tests wird ebenfalls Bestandteil dieses Meilensteins sein.

3. Aktueller Stand von Mobile IP

3.1 Standardisierung

Relevante RFCs und Drafts innerhalb der IP Routing for Wireless / Mobile Hosts (mobileip) [11]:

3.1.1 Internet-Drafts:

Route Optimization in Mobile IP [12]

Mobility Support in IPv6 [13]

Registration Keys for Route Optimization [14]

Mobile IP Regional Registration [15]

AAA Registration Keys for Mobile IP [16]

IP Mobility Support for IPv4, revised [17]

Mobile IP Based Micro Mobility Management Protocol in The Third Generation Wireless Network [18]

Generalized NAI Extension (GNAIE) [19]

Hierarchical MIPv6 mobility management [20]

3.1.2 Request For Comments:

Applicability Statement for IP Mobility Support (RFC 2005) [21]

Minimal Encapsulation within IP (RFC 2004) [22]

IP Encapsulation within IP (RFC 2003) [23]

IP Mobility Support (RFC 2002) [24]

The Definitions of Managed Objects for IP Mobility Support using SMIv2 (RFC 2006) [25]

Sun’s SKIP Firewall Traversal for Mobile IP (RFC 2356) [26]

Mobile IP Network Access Identifier Extension for IPv4 (RFC 2794) [27]

Mobile IP Authentication, Authorization, and Accounting Requirements (RFC 2977) [28]

Mobile IP Challenge/Response Extensions (RFC 3012) [29]

Reverse Tunneling for Mobile IP, revised (RFC 3024) [30]

Mobile IP Vendor/Organization-Specific Extensions (RFC 3025) [31]

3.2 Implementierungen

Die Tabelle 1, “Frei verfügbare Mobile IP Implementierungen” gibt eine Übersicht der frei verfügbaren Implementierungen, die nach Aktualität geordnet wurden. Derzeit werden, noch die Implementierungen Mosquito Net, HUT und Monarch gepflegt und auf die jeweils neuen Betriebssystem Versionen portiert. In Testergebnisse werden die einzelnen Mobile IPv4 Implementierungen im Detail besprochen.

Tabelle 1. Frei verfügbare Mobile IP Implementierungen

BS	IP	Version	Kommentar	Organisation
Linux	IPv4	2.2.x	RedHat 6.2	Standford University MosquitoNet [1]
			RedHat	Helsinki University of Technology HUT [2]
		2.0.x	nur für mobile Rechner	Portland State University Secure MIP [3]
				State University of New York Binghamton [4]
		2.0.37	RedHat 5.1 Suse 6.0	University of Singapore NUS [5]
		2.0.34	Slackware	University of Singapore NUS [5]
			RedHat 5.1	SUN Microsystems SUN [6]
	IPv6	2.1.9x	draft-ietf-mobileip-ipv6-05.txt	Lancaster University Lancaster [7]
		2.1.59	draft-ietf-mobileip-ipv6-04.txt	University of Singapore NUS [5]
FreeBSD	IPv4	3.3		Carnegie Mellon University Monarch [1]
		2.2.6		Portland State University Secure MIP [3]
	IPv6	3.3	draft-ietf-mobileip-ipv6-07.txt	Inria HMIPv6 [8]
Solaris	IPv4	2.5.1		SUN Microsystems SUN [6]
Windows	IPv4	NT 4.0		Politehnica University of Bucharest [9]
			nur für Agenten	University of Singapore NUS [5]
		95	nur für mobile Rechner	University of Singapore NUS [5]

4. Testumgebung

Die Beschreibung der Miriam Testumgebung kann dem dritten Meilenstein entnommen werden.

5. Testmechanismen

Zur Untersuchung der einzelnen Mobile IP Instanzen und deren Implementierungen haben wir folgende Mechanismen verwendet:

- die Log-Dateien der Implementierungen.
- `ping`, `traceroute`, `ethreal`, `top` sowie ein von uns erweitertes `tcpdump`.
- Implementierung der Testfolgen (siehe Kapitel 5.1 “Testfolgen”).
- Testbögen für die einzelnen Instanzen (sieheKapitel 5.2 “Testbögen”)
- Implementierung von Paketgeneratoren (sieheKapitel 5.3 “Paketgeneratoren”).

Im Gegensatz zum dritten Meilenstein wurden die Tests in eine andere Reihenfolge gebracht, so dass eine systematische Fehlersuche ermöglicht wird. Wenn ein Test negativ ausfällt, dann entfallen eventuell sämtliche nachfolgende Tests. In den Testbeschreibungen wurden Details ergänzt, wie die Tests durchzuführen sind und welche Ergebnisse erwünscht sind. In den Beschreibungen wird jeweils auf die Seitenzahl des RFC hingewiesen.

In Kapitel 5.2 “Testbögen” wurden die Tests zusätzlich noch nach Instanzen sortiert aufgeführt. Diese Blätter lassen sich als Checkliste verwenden.

5.1 Testfolgen

Die Testfolgen sind in zwei Bereiche unterteilt. Zum einen wurden Tests entwicklet, die MobileIP auf Funktionsfähigkeit der Implementierung überprüfen sowie Anhaltspunkte zur Fehlersuche bei nicht operablen Systemen bieten. Die Tests überprüfen die Konformität der Implementierungen mit den Normen der RFCs. Zum weiteren wurden Tests entwickelt, die der Untersuchung der Leistungsfähigkeit und Verfügbarkeit der Mobile IP Implementierungen dienen. Daher werden im folgenden zwei Aspekte unterschieden: Kapitel 5.1.1 “Testfolgen der Schnittstellen der Instanzen von Mobile IP” (Heimatagent, Fremdagent und Mobiler Knoten) und Kapitel 5.1.2 “Testfolgen der Instanzen unter Extrembedingungen”.

5.1.1 Testfolgen der Schnittstellen der Instanzen von Mobile IP

Zwischen den einzelnen Rechnern findet ein reger Austausch von Informationen statt. Diese Kommunikation wird mit den, in Tabelle 2, “Testfolgen der Schnittstellen der Instanzen von Mobile IP” zusammengefassten Tests untersucht. Die Tests sind in ihrer Reihenfolge chronologisch sortiert und entsprechen den getesteten Schnittstellen wie sie beim Aufbau einer Mobile IP Verbindung benötigt werden. In den folgenden Kapiteln Test 1: “Mobility Agent Solicitations” bis Test 3: “Registrierung” werden die einzelnen Tests beschrieben, sowie deren detaillierte Durchführung und Überprüfung angegeben.

Testnummer	Beschreibung	Details siehe
Test 1:	**Mobility Agent Solicitations**	**Seite 13**
Test 1.1:	*Festnetz*	*Seite 13*
Test 1.2:	*Funknetz*	*Seite 13*
Test 2:	**ICMP Router Advertisement**	**Seite 13**
Test 2.1:	*Unsolicitated Router Advertisement*	*Seite 13*
Test 2.1.1:	Festnetz	Seite 14
Test 2.1.2:	Funknetz	Seite 14
Test 2.2:	*Solicitated Router Advertisement*	*Seite 14*
Test 2.2.1:	Festnetz	Seite 14
Test 2.2.2:	Funknetz	Seite 14
Test 3:	**Registrierung**	**Seite 14**
Test 3.1:	*Registration Request des mobilen Knotens*	*Seite 15*
Test 3.2:	*korrekte Registrierungsanfrage*	*Seite 15*
Test 3.2.1:	Verbindungsaufbau	Seite 15
Test 3.2.2:	Timeout	Seite 16
Test 3.2.3:	Rückkehr ins Heimatnetz	Seite 16
Test 3.2.4:	Booten im Heimatnetz	Seite 16
Test 3.2.5:	Wechsel des Fremdnetzes	Seite 17
Test 3.2.5.1:	kein gesetztes S-Bit	Seite 17
Test 3.2.5.2:	gesetztes S-Bit – simultaneous bindings	Seite 17
Test 3.3:	*fehlerhafte Registrierungsanfrage*	*Seite 18*
Test 3.3.1:	fehlerhafte Authentifizierung mobile-home	Seite 18
Test 3.3.2:	mehrere Authentifizierung mobile-home	Seite 18
Test 3.3.2.1:	vor einer korrekten mobile-home Authentifizierung	Seite 18
Test 3.3.2.2:	nach einer korrekten mobile-home Authentifizierung	Seite 18
Test 3.3.3:	fehlende Authentifizierung mobile-home	Seite 18
Test 3.3.4:	fehlerhaftes ID-Feld (time-stamps)	Seite 19
Test 3.3.5:	fehlerhaftes ID-Feld (nonces)	Seite 19
Test 3.3.6:	Überschreitung der Lebensdauer des Heimatagenten	Seite 19
Test 3.3.7:	Überschreitung der Lebensdauer des Fremdagenten	Seite 20
Test 3.3.8:	fehlerhafter Heimatagent	Seite 20
Test 3.3.8.1:	es existiert kein Tunnel	Seite 20
Test 3.3.8.2:	es existiert ein Tunnel	Seite 20
Test 3.3.9:	unbekannter Fremdagent	Seite 21
Test 3.3.10:	fehlerhafte Anfrage bei bestehender Verbindung	Seite 21
Test 3.3.11:	Broadcast Anfrage an Heimatagent	Seite 21
Test 3.3.12:	Duplikate	Seite 21
Test 3.4:	*fehlerhafte Registrierungsantwort*	*Seite 21*
Test 3.4.1:	fehlerhafte Authentifizierung mobile-home	Seite 22
Test 3.4.2:	mehrere Authentifizierungen mobile-home	Seite 22
Test 3.4.3:	fehlende Authentifizierung mobile-home	Seite 22
Test 3.4.4:	fehlerhafte ID	Seite 22
Test 3.4.4.1:	Antwort wird vom Fremdagenten simuliert	Seite 22
Test 3.4.4.2:	Antwort kommt vom Heimatagenten zum Fremdagenten	Seite 22
Test 3.4.5:	keine Antwort	Seite 22

Tabelle 2. Testfolgen der Schnittstellen der Instanzen von Mobile IP

Test 1: Mobility Agent Solicitations

Beschreibung:

Um festzustellen, in welchem Heimat- oder Fremdnetz sich der mobile Rechner befindet, können ICMP Nachrichten versendet werden. Auf diese antworteten die anwesenden Heimat- und Fremdagenten.

Durchführung:

Die Mobility Agent Solicitations des mobilen Rechners sind mit tcpdump o.ä. abzuhören, ohne dass sich ein Mobility-Agent im Netz befindet. [[Per96a]], S.20.

Überprüfung:

- Das TTL-Feld des IP-Headers muss 1 sein. [10], S.18.
- Die Empfängeradress im IP-Header muss eine der folgenden sein: 255.255.255.255 (limited broadcast) oder 224.0.01 (“all systems on this link” multicast). [10], S. 14.
- Die Nachricht muss einer normalen ICMP Router Solicitation entsprechen. [10], S14 und [37].
- Nach den ersten drei Nachrichten, die in Sekundenabständen versant werden dürfen muss ein exponential binary backoff Algorithmus angewandt werden, der eine Maximale Zeitspanne zwischen zwei Solicitations von mindestens 60 Sekunden haben sollte. Außerdem müssen die Zeitabstände innerhalb bestimmter Fenster zufällig sein. [10], S.20.

Test 1.1: Festnetz

Test 1.2: Funknetz

Test 2: ICMP Router Advertisement

Beschreibung:

Router Advertisements werden in regelmäßigen Zeitabständen oder auf Anfrage eines Mobilen Rechners (vgl. Test 1) versendet.

Test 2.1: Unsolicitated Router Advertisement

Durchführung:

Die Router Advertisements sind zunächst ohne Anwesenheit eines mobilen Rechners mit tcpdump o.ä. abzuhören. Router mpssen kein Unsolicitated Router Advertisements versenden, wenn sie über den Link Layer entdeckt werden können, oder wenn sie auf Router Solicitations antworten. [10], S.18f.

Überprüfung:

- Das TTL-Feld des IP-Headers muss 1 sein. [10], S.14.

- Die Empfängeradress im IP-Header muss eine der folgenden sein: 255.255.255.255 (limited broadcast) oder 224.0.01 (“all systems on this link” multicast). [10], S. 14.
- Die Nachricht enthötl ein ICMP Router Advertisement mit Type = 0, Code = 0 oder 16, eventuell Router Adressen, die unabhängig von Mobile IP angegeben werden können. [10], S.14.
- An die ICMP Nachricht schließt sich die Mobile IP Extention an. [10], S.15.
- Bei einem Heimatagenten muss das H-Bit gesetzt sein, bei einem Fremdagenten das F-Bit. [10], S.16.
- Die Advertisements dürfen nicht öfter als einmal pro Sekunde versandt werden. [10], S.18.

Test 2.1.1: Festnetz

Test 2.1.2: Funknetz

Test 2.2: Solicitated Router Advertisement

Durchführung:

Die Router Advertisements sind wiederum mit tcpdump o.ä. abzuhören. Gleichzeitig werden aber Router Solicitations mit bekannter Absenderadresse versandt. Dies kann entweder von einem mobilen Knoten oder durch andere Maßnahmen (z.B. Abspielen aufgezeichneter Pakete) geschehen.

Überprüfung:

- Das Router Advertisement muss aussehen wie unter Test 2.1: “Unsolicitated Router Advertisement” beschrieben, mit dem Unterschied, dass die IP-Empfängeradress mit derr Absenderadresse derr Router Solicitation übereinstimmen kann. [37], S.4.
- Ebenso muss die MAC-Empfängeradresse mit der Absenderadresse der Router Solicitation übereinstimmen. [10], S.14.

Test 2.2.1: Festnetz

Test 2.2.2: Funknetz

Test 3: Registrierung

Beschreibung:

Die Registrierungsnachrichten werden vom mobilen Knoten versandt, laufen dann eventuell über den Fremdagenten zum Heimatagenten. Dieser schickt - eventuell wiederum über den Fremdagenten - die Antwort an den mobilen Knoten zurück.

Test 3.1: Registration Request des mobilen Knotens

Beschreibung:

Im Folgenden wird überprüft, ob sich der mobile Knoten beim Versenden der Registrierungsanfragen regelgerecht verhält.

Überprüfung:

- Der mobile Knoten sollte versuchen, sich über den Fremdagenten zu registrieren. [10], S.24.
- Der Empfänger Port des UDP-Headers muss 434 sein. [10], S.26.
- Nach dem der UDP-Header müssen sich die Mobile IP Felder befinden. [10], S.26.
- Die Extensions beinhalten mindestens einen Authentication-Header. [10], S.27.
- Die Absenderadresse muss die Heimatadresse des mobilen Knotens sein. [10], S.35.
- Das D-Bit darf nicht gesetzt sein, wenn ein Fremdagent verwendet wird [10], S.36.
- Die Registrierung muss entweder an die IP und MAC-Adresse des Fremdagenten gerichtet sein, oder an die IP-Adresse 224.0.011 und die MAC-Adresse des Fremdagenten (wenn Fremd Agent Detection auf der Sicherungs-Schicht verwendet wird). [10], S.35.

Test 3.2: korrekte Registrierungsanfrage

Beschreibung:

Mit diesem Test wird die Registrierung von Seiten der Agenten überprüft. Hierzu wird wenn möglich nur ein Agent betrachtet und der Registration-Request mit einem Paketgenerator erzeugt, so dass ausgeschlossen werden kann, dass sich Fehler in Fremd- und Heimatagent gegenseitig aufheben. Ist dies nicht möglich, sollte nur eine der Instanzen bislang ungetestet sein. Auf diese Weise kann einerseits ausgeschlossen werden, dass sich wie erwähnt Fehler gegenseitig aufheben, oder dass Fehler nicht auffindbar sind, da sie von einer Instanz verursacht werden, die momentan nicht getestet werden soll.

Bei sämtlichen Überprüfungen von Tunnels ist zu berücksichtigen, dass mit Hilfe eines pings nur festgestellt werden kann, wenn ein Tunnel zwischen Heimatagent und Fremdagent besteht. Unter Umstaänden wird jedoch nur ein Tunnelendpunkt abgebaut. Dann funktioniert der Tunnel als gesamtes zwar nicht mehr und die ping-echos kommen nicht an, aber die drei eventuell noch bestehenden Tunnelendpunkte lassen sich unter Umständen für Hacker-Angriffe nutzen.

In der Regel sollte auf eine Registrierungsanfrage eine Registrierungsantwort erfolgen. Auch diese wird in einem UDP-Paket versandt. Wie die Registrierungsanfrage muss auch die Registrierungsantwort mindestens eine Erweiterung zur Authentifizierung enthalten.

Test 3.2.1: Verbindungsaufbau

Durchführung:

Senden einer korrekten Registrierungsanfrage und Abhören der Kommunikation:

- mobiler Rechner - Fremdagent
- Fremdagent - Heimatagent

Überprüfung:
- Überprüfung der Paketinhalte.
- Überprüfung der Besuchstabellen. [10], S.43 und S.48.
- Überprüfung der ARP Tabellen in Heimatagent, Fremdagent und mobilem Rechner
- Überprüfung des Tunnels durch bspw. `ping`.
- Registrierungsantwort muss vom Heimatagenten an den Fremdagenten gesendet und von diesem an den mobilen Rechner weitergeleitet werden. [10], S.51.

Test 3.2.2: Timeout

Durchführung:

Verhindern, dass weitere Registrierungspakete gesendet werden.

Überprüfung:
- Nach Ablauf der Lebenszeit der Verbindung Überprüfung der Besuchstabellen und der ARP-Tabellen in Heimatagenten, Fremdagent und mobilem Rechner.
- Überprüfung des Tunnels durch bspw. `ping`.
- Der Tunnelendpunkt im Heimatagent muss auf jeden Fall auslaufen. [10], S.50.

Test 3.2.3: Rückkehr ins Heimatnetz

Durchführung:

Regulärer Aufbau eines Tunnels zu einem mobilen Rechner im Fremdnetz, anschließend direkte Registrierung des Rechners beim Heimatagenten vom Heimatnetz aus - mit einer Lebenszeit für die Registrierung = 0. [10], S.38. Care-off Adresse gleich der Adresse des Heimatagenten – bevor die Lebenszeit der ursprünglichen Registrierung abgelaufen ist (entspricht einer Deregistrierung); Abhören der Kommunikation mobiler Rechner-Heimatagent.

Überprüfung:
- Überprüfung der Besuchstabellen und der ARP-Tabellen in Heimatagent und mobilem Rechner.
- Der mobile Rechner muss aus der Tabelle gelöscht werden. [10], S.50.

Test 3.2.4: Booten im Heimatnetz

Durchführung:

Booten eines mobilen Rechners im Heimatnetz; abhören der Kommunikation zwischen mobilem Rechner und Heimatagent.

Überprüfung:

- Der mobile Rechner muss sich beim Heimatagenten registrieren mit der care-of Adresse gleich der IP-Adresse des Heimatagenten und einer Lebensdauer = 0. Da er bootet weiß er nicht, ob eventuell Tunnel in Fremdnetze bestehen. Deshalb muss die Deregistrierung durchgeführt werden.

Test 3.2.5: Wechsel des Fremdnetzes

Durchführung:

Regulärer Aufbau eines Tunnels zu einem mobilen Rechner im Fremdnetz, anschließend Registrierung des mobilen Rechners über einen anderen Fremdagenten in einem anderen Fremdnetz.

Abhören der Kommunikation:

- mobiler Rechner – neuer Fremdagent
- neuer Fremdagent – Heimatagent

Überprüfung:

- Überprüfung der Paketinhalte.
- Überprüfung der Besuchstabellen und der ARP-Tabellen in Heimatagent, beiden Fremdagenten und mobilem Rechner.
- Überprüfung des Tunnels über den neuen Fremdagenten bspw. mit `ping`.
- Die Registrierungsantwort muss zum neuen Fremdagenten geroutet werden.
- Der Tunnel zwischen Heimatagent und neuem Fremdagent muss sofort (nach der Registrierung) aufgebaut werden und die Besuchstabelle im Heimatagent muss aktualisiert werden.
- Die ARP-Tabelle des neuen Fremd-Agenten muss sofort angepasst werden.
- Die Besuchstabellen des alten Fremdagenten müssen auslaufen.

Test 3.2.5.1: kein gesetztes S-Bit

Überprüfung:

- Der Tunnel vom Heimatagenten zum alten Fremdagenten muss sofort abgebaut werden. [10], S.26.

Test 3.2.5.2: gesetztes S-Bit – simultaneous bindings

Überprüfung:

- Der Tunnel vom Heimatagenten zum alten Fremdagenten muss nach Timeout der Verbindung abgebaut werden. [10], S.26.

Test 3.3: fehlerhafte Registrierungsanfrage

Beschreibung:

Hier ist zu bedenken, dass zwar eine Reihe von Fehlermeldungen in [10] definiert werden, dass aber nur wenige auch vom Heimatagenten gesendet werden müssen. Ein Test, ob Fehlermeldungen gesendet werden kann also im allgemeinen nicht durchgeführt werden. Unter Umständen ist es sinnvoll keine Fehlermeldungen zu senden, sondern diese zu verwerfen (z. B. um kein Ziel von DoS-Attacken zu werden, oder um Hackern keine Anhaltspunkte über Ablehnungsgründe der Registrierung zu geben).

Test 3.3.1: fehlerhafte Authentifizierung mobile-home

Durchführung:

Senden einer Registrierungsanfrage ohne mobile-home Authentifizierung und Überwachung der Kommunikation zwischen Heimatagent und mobilem Rechner.

Überprüfung:

- Die Registrierung muss fehlschlagen. [10], S.30 und S.49.

Test 3.3.2: mehrere Authentifizierung mobile-home

Durchführung:

Senden einer Registrierungsanfrage mit fehlerhafter mobile-home Authentifizierung.

Abhören der Kommunikation zwischen:

- Heimatagent - mobilem Rechner

Test 3.3.2.1: vor einer korrekten mobile-home Authentifizierung

Überprüfung:

- Die Registrierung muss fehlschlagen. [10], S.49.

Test 3.3.2.2: nach einer korrekten mobile-home Authentifizierung

Überprüfung:

- Die Registrierung muss fehlschlagen. [10], S.49.

Test 3.3.3: fehlende Authentifizierung mobile-home

Durchführung:

Senden einer Registrierungsanfrage ohne mobile-home Authentifizierung und Überwachung der Kommunikation zwischen Heimatagent und mobilem Rechner.

Überprüfung:

- Die Registrierung muss fehlschlagen. [10], S.30 und S.49.

Test 3.3.4: fehlerhaftes ID-Feld (time-stamps)

Durchführung:

Einstellen von time-stamps als Replay-Protection; senden einer Registrierungsanfrage mit fehlerhaftem ID-Feld im Mobile IP Paket. Bei Verwendung einer realen Mobile IP Implementierung kann eine fehlerhafte ID durch Verstellen der Systemzeit des mobilen Rechners gegenüber dem Heimatagenten geschehen.

Abhören der Kommunikation:

- Heimatagent - Fremdagent
- Fremdagent - mobiler Rechner

Überprüfung:

- Die Registrierung muss fehlschlagen. [10], S.49.
- Im ersten Teil des ID-Felds der Registrierungsantwort muss die aktuelle Zeit des Heimatagenten übermittelt werden, im zweiten Teil muss der zweite Teil des ID-Feldes der Anforderung wiederholt werden.
- Wird eine reale Mobile IP Implementierung zum Testen verwendet, darf diese nun keine Registrierungsanfrage mit fehlerhafter ID mehr versenden. [10], S.41.

Test 3.3.5: fehlerhaftes ID-Feld (nonces)

Durchführung & Überprüfung:

Test analog 1.3.3.4

Test 3.3.6: Überschreitung der Lebensdauer des Heimatagenten

Durchführung:

Senden einer Registrierungsanfrage, die die maximale Lebensdauer die beim Heimatagenten eingestellt wurde überschreitet

Abhören der Kommunikation:

- mobiler Rechner – Fremdagent
- Fremdagent – Heimatagent

Überprüfung:

- gegebenenfalls Überprüfung der Paketinhalte.
- Überprüfung der Besuchstabellen in Heimatagent, Fremdagent und mobilem Rechner.
- Die Registrierungsantwort muss die Lebenszeit des Heimatagenten zurückliefern.
- Fremdagent und mobiler Rechner müssen diese übernehmen [10], S.40.

Test 3.3.7: Überschreitung der Lebensdauer des Fremdagenten

Durchführung:

Senden einer Registrierungsanfrage, die die maximale Lebensdauer die beim Fremdagenten eingestellt wurde überschreitet.

Abhören der Kommunikation:

- mobiler Rechner – Fremdagent
- Fremdagent – Heimatagent

Überprüfung:

- gegebenenfalls Überprüfung der Paketinhalte.
- Überprüfung der Besuchstabellen in Heimatagent, Fremdagent und mobilem Rechner.
- Überprüfung des Tunnels durch bspw. `ping`.
- Die Registrierung muss bereits vom Fremdagenten abgelehnt werden.
  aber: widersprüchliche Angaben zur Ablehnungsmeldung in [10]:
  S.43: Eine Ablehnungsmeldung durch den Fremdagenten ist Pflicht.
  S.45: Eine Ablehnungsmeldung durch den Fremdagenten ist keine Pflicht.
- Es darf kein Tunnel aufgebaut werden.
- Der mobile Rechner darf einen erneuten Request senden, allerdings nur mit der maximalen TTL die er aus der Registrierungsantwort (Ablehnung) erhält. [10], S.41.

Test 3.3.8: fehlerhafter Heimatagent

Durchführung:

Senden einer Registrierungsanfrage über einen Fremdagenten, wobei ein nicht existenter Heimatagent angegeben wird.

Test 3.3.8.1: es existiert kein Tunnel

Beschreibung:

Es besteht vorher kein Tunnel für den Mobilen Rechner am Fremdagent.

Überprüfung:

- Es sollte kein Eintrag in die Besuchstabelle des Fremdagenten stattfinden.

Test 3.3.8.2: es existiert ein Tunnel

Beschreibung:

Es besteht vorher ein Tunnel für den mobilen Rechner am Fremdagenten

Überprüfung:

- Der alte Eintrag in der Besuchstabelle des Fremdagenten sollte gelöscht werden.

Test 3.3.9: unbekannter Fremdagent

Durchführung:

Senden einer Registrierungsanfrage von einem dem Heimatagenten unbekannten Fremdagenten.

Abhören der Kommunikation zwischen:

- Heimatagent - Fremdagent

Überprüfung:

- Die Registrierung sollte nicht akzeptiert werden.

Test 3.3.10: fehlerhafte Anfrage bei bestehender Verbindung

Durchführung:

Verbindung aufbauen, gleicher mobiler Knoten sendet falsche Registrierungsanfrage.

Überprüfung:

- Die Verbindung muss bestehen bleiben. [10], S.43f.

Test 3.3.11: Broadcast Anfrage an Heimatagent

Durchführung:

Senden einer Registrierungsanfrage an die Broadcast-Adresse des Heimatnetzes.

Überprüfung:

- Die Anfrage muss abgelehnt werden. [10]. S.50.

Test 3.3.12: Duplikate

Durchführung:

Registrieren eines mobilen Rechners, später Wiederholung der gleichen Registrierung (gleiche home-address, gleiche care-off-address, gleiches ID-Feld).

Überprüfung:

- Die Lifetime darf im Anschluss an die zweite Anfrage im Heimatagent nicht erhöht werden. [10], S.51.

Test 3.4: fehlerhafte Registrierungsantwort

Durchführung:

Test 3.3.1: “fehlerhafte Authentifizierung mobile-home” bis Test 3.3.3: “fehlende Authentifizierung mobile-home” lassen sich analog für die Registrierungsantwort durchführen.

Um die Tests durchzuführen muss jeweils vom mobilen Knoten eine Registrierung initiiert werden. Die Antwort des Heimatagenten wird dann entsprechend des jeweiligen Tests verändert.

Überprüfung:

- In jedem der Fälle muss die Registrierung fehlschlagen. [10], S.39f.

Test 3.4.1: fehlerhafte Authentifizierung mobile-home

Test 3.4.2: mehrere Authentifizierungen mobile-home

Test 3.4.3: fehlende Authentifizierung mobile-home

Test 3.4.4: fehlerhafte ID

Durchführung:

Versuch einer Registrierung durch einen mobilen Rechner. In der Registrierungsantwort wird eine andere ID (2. Teil) verwendet

Test 3.4.4.1: Antwort wird vom Fremdagenten simuliert

Überprüfung:

- Die Registrierung muss fehlschlagen; vom mobilen Rechner “silently discarded”. [10], S.39.

Test 3.4.4.2: Antwort kommt vom Heimatagenten zum Fremdagenten

Überprüfung:

- Die Registrierung muss fehlschlagen; vom Fremdagent “silently discarded”. [10], S.46.

Test 3.4.5: keine Antwort

Durchführung:

Versuch einer Registrierung durch einen mobilen Rechner. Er erhält keine Antwort.

Überprüfung:

- Wiederholung des Registrierungsversuchs mit anderen timestamps (ID) oder gleichen nonces. [10], S.42.
- Zeit zwischen den Versuchen >= 1s mit binary exponential backoff Algorithmus. [10], S.42.

5.1.2 Testfolgen der Instanzen unter Extrembedingungen

Diese Tests für Mobile IP Implementierungen zielen auf die Leistungsfähigkeit sowie die Verfügbarkeit aller Instanzen. Hierfür wurden Hochlastsimulationen sowie Denial of Service Angriffe implementiert. Die Testfolgen sind in der folgenden Tabelle 3, “Testfolgen der Instanzen unter Extrembedingungen” zusammengefasst und in Kapitel Test 4: “Hochlastsimulationen” und Kapitel Test 5: “Denial of Service Attacken” detailiert beschrieben.

Testnummer	Beschreibung	Details siehe
Test 4:	**Hochlastsimulationen**	**Seite 23**
Test 4.1:	*viele Tunnel im Fremdagenten*	*Seite 23*
Test 4.2:	*ein Tunnel, viele mobile Knoten im Heimatagenten*	*Seite 24*
Test 4.3:	*viele Tunnel im Heimatagenten*	*Seite 24*
Test 4.4:	*Hochlast auch mit falschen Registrierungsanfragen*	*Seite 24*
Test 4.5:	*Hochlast nur mit falschen Anfragen*	*Seite 24*
Test 5:	**Denial of Service Attacken**	**Seite 25**
Test 5.1:	*gefälschte Agent Advertisment Pakete*	*Seite 25*
Test 5.2:	*gefälschte Solicitation Pakete*	*Seite 25*
Test 5.3:	*gefälschte Registration Request Anfragen im Heimatnetz*	*Seite 25*
Test 5.4:	*gefälschte Registration Request Anfragen im Fremdnetz*	*Seite 26*

Tabelle 3. Testfolgen der Instanzen unter Extrembedingungen

Test 4: Hochlastsimulationen

Beschreibung:

Testläufe mit einer hohen Anzahl (ca. 10 000) regulärer Benutzer.

Test 4.1: viele Tunnel im Fremdagenten

Durchführung:

Senden von ca. 10 000 Registrierungsanfragen und Antworten an einen Fremdagenten. Dabei simulieren die Anfragen und Antworten eine große Anzahl verschiedener mobiler Rechner mit verschiedenen Adressen der Heimatagenten. Die Lebensdauer wird zweckmäßigerweise auf unendlich gestellt (0xffff), damit die Einträge in der Besuchstabelle nicht gelöscht werden bevor der Test beendet ist.

Überprüfung:

- Kontrolle von ARP- und Besuchstabelle.
- Sämtliche Registrierungen müssen sich in der ARP- und der Besuchstabelle wiederfinden.

Test 4.2: ein Tunnel, viele mobile Knoten im Heimatagenten

Durchführung:

Senden von ca. 10 000 Registrierungsanfragen direkt an einen Heimatagenten. Dabei simulieren die Anfragen eine große Anzahl verschiedener mobiler Rechner, die sich jedoch alle im gleichen Fremdnetz befinden. Die Lebensdauer wird auch hier auf unendlich gestellt.

Überprüfung:

- Kontrolle der Besuchstabelle des Heimatagenten.
- Sämtliche Registrierungen müssen sich in der Besuchstabelle wiederfinden.

Test 4.3: viele Tunnel im Heimatagenten

Durchführung:

Senden von ca. 10 000 Registrierungsanfragen direkt an einen Heimatagenten. Dabei simulieren die Anfragen eine große Anzahl verschiedener mobiler Rechner, die auch verschiedene care-of Adressen verwenden. Die Lebensdauer wird auch hier auf unendlich gestellt.

Überprüfung:

- Kontrolle der Besuchstabelle des Heimatagenten.
- Sämtliche Registrierungen müssen sich in der Besuchstabelle wiederfinden.
- Es müssen alle Tunnel aufgebaut worden sein.

Test 4.4: Hochlast auch mit falschen Registrierungsanfragen

Durchführung:

Senden von ca. 10 000 Registrierungsanfragen über den Fremdagenten an den Heimatagenten mit Einstreuung falscher Registrierungsanfragen.

Überprüfung:

- Die falschen Registrierungsanfragen dürfen nicht dazu führen, dass korrekte Anfragen abgelehnt oder verworfen werden.

Test 4.5: Hochlast nur mit falschen Anfragen

Durchführung:

Senden vieler falscher Anfragen an den Fremdagenten.

Überprüfung:

- der Fremdagent muss diese ablehnen.

- bei mehr als einer Anfrage pro Sekunde von einem mobilen Rechner ist keine Antwort mehr nötig. [10], S.43.

Test 5: Denial of Service Attacken

Beschreibung:

DoS Attaken wurden bereits in Kapitel 8. “Sicherheit” ausführlich behandelt. Insbesondere die Ergebnisse aus Test 5.1: “gefälschte Agent Advertisment Pakete” und Test 5.4: “gefälschte Registration Request Anfragen im Fremdnetz” wurden dort bereits erörtert. In diesem Kapitel erscheinen sie - neben weiteren Tests - der Vollständigkeit halber.

Test 5.1: gefälschte Agent Advertisment Pakete

Durchführung:

Überflutung des Mobilen Rechners durch gefälschte ICMP Pakete mit einer Agent Advertisment Nachricht.

Überprüfung:

- Die Belastung des mobilen Rechners kann mit den Werkzeugen `top` und `ping` überprüft werden.

Test 5.2: gefälschte Solicitation Pakete

Durchführung:

Überflutung des Fremdagenten durch ICMP Pakete mit einer gefälschten Solicitation Nachricht. Diese Nachricht fordert den Fremdagenten auf mit einem Agent Advertisement zu antworten.

Überprüfung:

- Die Belastung des Fremdagenten kann mit den Werkzeugen `top` und `ping` überprüft werden.
- Die Belastung des mobilen Rechners aufgrund der Agent Advertisements kann mit den Werkzeugen `top` und `ping` überprüft werden.

Test 5.3: gefälschte Registration Request Anfragen im Heimatnetz

Durchführung:

Überflutung eines Heimatagenten durch UDP Paketen mit einer gefälschten Registration Request Anfrage von unterschiedlichen Mobilen Rechnern. Diese Nachrichten fordern den Heimatagenten auf, die Mobilen Rechner zu registrieren und einen Tunnel - genauer einen Tunnelendpunkt - zu etablieren.

Überprüfung:

- Die Belastung des Heimatagenten kann mit den Werkzeugen `top` und `ping` überprüft werden.
- Die Anzahl der Tunnel kann mit entsprechenden Monitor Programmen (bzw. lsof) oder über die Registrierungsdateien des Heimatagenten überprüft werden.

Test 5.4: gefälschte Registration Request Anfragen im Fremdnetz

Durchführung:

Überflutung eines Fremdagenten durch UDP Paketen mit einer gefälschten Registration Request Anfrage von unterschiedlichen Mobilen Rechnern.

Überprüfung:

- Die Belastung des Fremdagenten kann mit den Werkzeugen `top` und `ping` überprüft werden.

5.2 Testbögen

5.2.1 Heimatagent

TABELLE 4.

Testnummer	Beschreibung/Überprüfung	Details siehe
Test 2:	**ICMP Router Advertisement**	**Seite 13**
Test 2.1:	*Unsolicitated Router Advertisement*	*Seite 13*
Test 2.1.1:	Festnetz • IP-TTL • IP-Empfängeradresse (255.255.255.255 und 244.0.0.1) • Tpye, Code • Mobile IP-Extension (H-Bit, F-Bit) • Minimales Intervall	Seite 14
Test 2.1.2:	Funknetz • IP-TTL • IP-Empfängeradresse (255.255.255.255 und 244.0.0.1) • Tpye, Code • Mobile IP-Extension (H-Bit, F-Bit) • Minimales Intervall	Seite 14
Test 2.2:	*Solicitated Router Advertisement*	*Seite 14*
Test 2.2.1:	Festnetz • IP-TTL • Type, Code • Mobile IP Extension (H-Bit, F-Bit) • MAC Empfängeradresse	Seite 14
Test 2.2.1:	Funknetz • IP-TT. • Type, Code • Mobile IP Extension (H-Bit, F-Bit) • MAC Empfängeradresse	Seite 14
Test 3:	**Registrierung**	**Seite 14**
Test 3.2:	*korrekte Registrierungsanfrage*	*Seite 15*
Test 3.2.1:	Verbindungsaufbau • HA -> FA • Besuchstabelle HA • ARP-Tabelle HA • Tunnel	Seite 15
Test 3.2.2:	Timeout • Besuchstabelle HA • ARP-Tabelle HA • Tunnel	Seite 16
Test 3.2.3:	Rückkehr ins Heimatnetz • Besuchstabelle HA • ARP-Tabelle HA • Tunnel	Seite 16

TABELLE 4.

Testnummer	Beschreibung/Überprüfung	Details siehe
Test 3.2.5: Test 3.2.5.1:	Wechsel des Fremdnetzes kein gesetztes S-Bit • HA -> neuer FA • Besuchstabelle HA • APR-Tabelle HA • Tunnel altes Fremdnetz • Tunnel neues Fremdnetz	Seite 16 Seite 17
Test 3.2.5.2:	gesetztes S-Bit – simultaneous bindings • HA -> neuer FA • Besuchstabelle HA • APR-Tabelle HA • Tunnel altes Fremdnetz • Tunnel neues Fremdnetz	Seite 17
Test 3.3:	*fehlerhafte Registrierungsanfrage*	*Seite 18*
Test 3.3.1:	fehlerhafte Authentifizierung mobile-home • Fehlschlag	Seite 18
Test 3.3.2: Test 3.3.2.1:	mehrere Authentifizierung mobile-home vor einer korrekten mobile-home Authentifizierung • Fehlschlag	Seite 18 Seite 18
Test 3.3.2.2:	nach einer korrekten mobile-home Authentifizierung • Fehlschlag	Seite 18
Test 3.3.3:	fehlende Authentifizierung mobile-home • Fehlschlag	Seite 18
Test 3.3.4:	fehlerhaftes ID-Feld (time-stamps) • Fehlschlag • ID-Feld (Teil1) • ID-Feld (Teil2)	Seite 19
Test 3.3.5:	fehlerhaftes ID-Feld (nonces) • Fehlschlag • ID-Feld	Seite 19
Test 3.3.6:	Überschreitung der Lebensdauer des Heimatagenten • TTL HA -> FA • Besuchstabelle HA	Seite 19
Test 3.3.10:	fehlerhafte Anfrage bei bestehender Verbindung • Tunnel	Seite 21
Test 3.3.11:	Broadcast Anfrage an Heimatagent • Fehlschlag	Seite 21
Test 3.3.12:	Duplikate • Liefetime im HA	Seite 21
Test 4:	**Hochlastsimulationen**	**Seite 23**
Test 4.2:	*ein Tunnel, viele mobile Knoten im Heimatagenten* • Besuchstabelle HA • ARP-Tabelle HA	*Seite 24*
Test 4.3:	*viele Tunnel im Heimatagenten* • Besuchstabelle HA • ARP-Tabelle HA • Routing -Tabelle HA	*Seite 24*
Test 4.4:	*Hochlast auch mit falschen Registrierungsanfragen* • Besuchstabelle HA • ARP-Tabelle HA • Routing Tabelle HA	*Seite 24*

TABELLE 4.

Testnummer	Beschreibung/Überprüfung	Details siehe
Test 5:	**Denial of Service Attacken**	**Seite 25**
Test 5.1:	*gefälschte Agent Advertisment Pakete* • CPU Last mittels top	*Seite 25*
Test 5.3:	*gefälschte Registration Request Anfragen im Heimatnetz* • CPU Last mittels top • Routingtabelle HA • Login Dateien	*Seite 25*

Tabelle 5. Testbogen Heimatagent

5.2.2 Fremdagent

Testnummer	Beschreibung/Überprüfung	Details siehe
Test 2:	**ICMP Router Advertisement**	**Seite 13**
Test 2.1:	*Unsolicitated Router Advertisement*	*Seite 13*
Test 2.1.1:	Festnetz • IP-TTL • IP-Empfängeradresse (255.255.255.255 und 224.0.0.1) • Type, Code • Mobile IP Extension (H-Bit, F-Bit) • Minimales Intervall	Seite 14
Test 2.1.2:	Funknetz • IP-TTL • IP-Empfängeradresse (255.255.255.255 und 224.0.0.1) • Type, Code • Mobile IP Extension (H-Bit, F-Bit) • Minimales Intervall	Seite 14
Test 2.2:	*Solicitated Router Advertisement*	*Seite 14*
Test 2.2.1:	Festnetz • IP-TTL • Tpye, Code • Mobile Extension (H-Bit, F-Bit) • MAC-Empfängeradresse	Seite 14
Test 2.2.2:	Funknetz • IP-TTL • Tpye, Code • Mobile Extension (H-Bit, F-Bit) • MAC-Empfängeradresse	Seite 14
Test 3:	**Registrierung**	**Seite 14**
Test 3.2:	*korrekte Registrierungsanfrage*	*Seite 15*
Test 3.2.1:	Verbindungsaufbau • FA -> HA • FA -> MN • Besuchstabelle FA • ARP-Tabelle FA • Tunnel	Seite 15
Test 3.2.2:	Timeout • Besuchstabelle FA • ARP-Tabelle FA • Tunnel	Seite 16
Test 3.2.3:	Rückkehr ins Heimatnetz • Besuchstabelle FA • ARP-Tabelle FA • Tunnel	Seite 16

Testnummer	Beschreibung/Überprüfung	Details siehe
Test 3.2.5:	Wechsel des Fremdnetzes	Seite 17
Test 3.2.5.1:	kein gesetztes S-Bit • Neuer FA -> HA • Neuer FA -> MN • Besuchstabelle alter FA • Besuchstabelle neuer FA • APR-Tabelle alter FA • APR-Tabelle neuer FA • Tunnel altes Fremdnetz • Tunnel neues Fremdnetz	Seite 17
Test 3.2.5.2:	gesetztes S-Bit – simultaneous bindings • Neuer FA -> HA • HA -> neuer FA • Neuer FA-> MN • Besuchstabelle alter FA • Besuchstabelle neuer FA • ARP-Tabelle alter FA • ARP-Tabelle neuer FA • Tunnel altes Fremdnetz • Tunnel neues Fremdnetz	Seite 17
Test 3.3:	*fehlerhafte Registrierungsanfrage*	*Seite 18*
Test 3.3.1:	fehlerhafte Authentifizierung mobile-home • Fehlschlag	Seite 18
Test 3.3.2:	mehrere Authentifizierung mobile-home	Seite 18
Test 3.3.2.1:	vor einer korrekten mobile-home Authentifizierung • Fehlschlag	Seite 18
Test 3.3.2.2:	nach einer korrekten mobile-home Authentifizierung • Fehlschlag	Seite 18
Test 3.3.3:	fehlende Authentifizierung mobile-home • Fehlschlag	Seite 18
Test 3.3.6:	Überschreitung der Lebensdauer des Heimatagenten • TTL FA -> MN • Besuchstabelle FA	Seite 19
Test 3.3.7:	Überschreitung der Lebensdauer des Fremdagenten • FA -> HA • FA -> MN • Besuchstabelle FA • Tunnel	Seite 20
Test 3.3.8:	fehlerhafter Heimatagent	Seite 20
Test 3.3.8.1:	es existiert kein Tunnel • Besuchstabelle FA	Seite 20
Test 3.3.8.2:	es existiert ein Tunnel • Besuchstabelle FA	Seite 20
Test 3.3.9:	unbekannter Fremdagent • Fehlschlag	Seite 21
Test 3.3.10:	fehlerhafte Anfrage bei bestehender Verbindung • Tunnel	Seite 21
Test 3.3.11:	Broadcast Anfrage an Heimatagent • Fehlschlag	Seite 21
Test 3.3.12:	Duplikate • Lifetime im FA	Seite 21

Testnummer	Beschreibung/Überprüfung	Details siehe
Test 3.4:	*fehlerhafte Registrierungsantwort*	*Seite 21*
Test 3.4.1:	fehlerhafte Authentifizierung mobile-home • Fehlschlag	Seite 22
Test 3.4.2:	mehrere Authentifizierungen mobile-home • Fehlschlag	Seite 22
Test 3.4.3:	fehlende Authentifizierung mobile-home • Fehlschlag	Seite 22
Test 3.4.4: Test 3.4.4.2:	fehlerhafte ID Antwort kommt vom Heimatagenten zum Fremdagenten • Besuchstabelle FA • FA -> MN	Seite 22 Seite 22
Test 4:	**Hochlastsimulationen**	**Seite 23**
Test 4.1:	*viele Tunnel im Fremdagenten* • Besuchstabelle FA • ARP-Tabelle FA	*Seite 23*
Test 4.4:	*Hochlast auch mit falschen Registrierungsanfragen* • Besuchstabelle FA • ARP-Tabelle FA	*Seite 24*
Test 4.5:	*Hochlast nur mit falschen Anfragen* • FA -> MN	*Seite 24*
Test 5:	**Denial of Service Attacken**	**Seite 25**
Test 5.1:	*gefälschte Agent Advertisment Pakete*	*Seite 25*
Test 5.2:	*gefälschte Solicitation Pakete* • CPU Last mittels top	*Seite 25*
Test 5.4:	*gefälschte Registration Request Anfragen im Fremdnetz* • CPU Last mittels top	*Seite 26*

Tabelle 6. Testbogen Fremdagenten

5.2.3 Mobiler Rechner

Testnummer	Beschreibung	Details siehe
Test 1:	**Mobility Agent Solicitations**	**Seite 13**
Test 1.1:	*Festnetz* • IP-TTL • IP-Empfängeradress • Nachrichtenaufbau • binary exponential backoff	*Seite 13*
Test 1.2:	*Funknetz* • IP-TTL • IP-Empfängeradress • Nachrichtenaufbau • binary exponential backoff	*Seite 13*
Test 3:	**Registrierung**	**Seite 14**
Test 3.1:	*Registration Request des mobilen Knotens* • Registrierungsversuche • Empfänger-Port • MIP-Felder • Authentication-Header • Absenderadresse • D-Bit • Empfängeradresse (IP und MAC-Adresse des Fremdagenten / IP-Adresse 224.0.0.1 und MAC-Adresse des Fremdagenten)	*Seite 15*
Test 3.2:	*korrekte Registrierungsanfrage*	*Seite 15*
Test 3.2.1:	Verbindungsaufbau • Besuchstabelle MN • ARP-Tabelle MN • Tunnel	Seite 15
Test 3.2.2:	Timeout • Besuchstabelle MN • ARP-Tabelle MN • Tunnel	Seite 16
Test 3.2.3:	Rückkehr ins Heimatnetz • Besuchstabelle MN • ARP-Tabelle MN • Tunnel	Seite 16
Test 3.2.4:	Booten im Heimatnetz • Care-off Adresse • Lifetime	Seite 16
Test 3.2.5: Test 3.2.5.1: Test 3.2.5.2:	Wechsel des Fremdnetzes kein gesetztes S-Bit • MN -> neuer FA • Besuchstabelle MN • ARP-Tabelle MN • Tunnel altes Fremdnetz • Tunnel neues Fremdnetz gesetztes S-Bit – simultaneous bindings • MN -> neuer FA • Besuchstabelle MN • APR-Tabelle MN • Tunnel altes Fremdnetz • Tunnel neues Fremdnetz	Seite 17 Seite 17 Seite 17

Testnummer	Beschreibung	Details siehe
Test 3.3:	*fehlerhafte Registrierungsanfrage*	*Seite 18*
Test 3.3.4:	fehlerhaftes ID-Feld (time-stamps) • Fehlschlag • Evtl. korrekte Registrierung	Seite 19
Test 3.3.5:	fehlerhaftes ID-Feld (nonces) • Fehlschlag • Evtl. korrekte Registrierung	Seite 19
Test 3.3.6:	Überschreitung der Lebensdauer des Heimatagenten • Besuchstabelle MN	Seite 19
Test 3.3.7:	Überschreitung der Lebensdauer des Fremdagenten • Besuchstabelle MN • Tunnel • Erneuter Request vom MN	Seite 20
Test 3.4:	*fehlerhafte Registrierungsantwort*	*Seite 21*
Test 3.4.1:	fehlerhafte Authentifizierung mobile-home • Fehlschlag	Seite 22
Test 3.4.3:	fehlende Authentifizierung mobile-home • Fehlschlag	Seite 22
Test 3.4.4: Test 3.4.4.1: Test 3.4.5:	fehlerhafte ID Antwort wird vom Fremdagenten simuliert • Besuchstabelle MN • MN -> FA keine Antwort • timestamps • nonces • binary exponential backoff	Seite 22 Seite 22 Seite 22
Test 5: *Test 5.1:*	**Denial of Service Attacken** *gefälschte Agent Advertisment Pakete* • CPU Last mittels top	**Seite 25** *Seite 25*

Tabelle 7. Testbogen Mobiler Rechner

5.3 Paketgeneratoren

5.3.1 Registrierungspaketgenerator

Syntax und Funktion.

Im Rahmen der Tests wurde ein C-Programm entwickelt, das in der Lage ist, verschiedene Registrierungsanfragen zu senden die einen mobilen Rechner oder einen mobilen Rechner und einen Fremdagenten simulieren. Die Syntax für dieses Programm werden im folgenden erläutert. Kleinbuchstaben sind immer von einem Parameter gefolgt, Großbuchstaben beziehen sich normalerweise auf den entsprechenden Befehl mit Kleinbuchstabe, werden jedoch ohne weitere Parameter angegeben.

```
dt [-l lifetime] [-m MN_IP] [-h HA_IP] [-f FA_IP]
[-i ID1] [-p SPI] [-k key] [-s SRC_IP] [-d DST_IP]
[-c count] [-w wait] [-M] [-I] [-S] [-F] [-U] [-V] [-
N] [-A] [-H]
```

(V und N schließen sich aus. Nicht jedoch A und N oder A und V.)

Dabei bedeuten:

- lifetime: angefragte Registrierungs-Lebenszeit
- MN_IP: IP-Adresse des mobilen Rechners im Heimatnetz
- HA_IP: IP-Adresse des Heimatagenten
- FA_IP: IP-Adresse des Fremdagenten
- ID1: zweiter Teil des ID-Feldes (sollte eigentlich Zufallszahl sein)
- SPI: security parameter index (Zugriffsidentifikator für den geheimen Schlüssel)
- key: “shared secret” zwischen Heimatagent und mobilem Rechner
- SRC_IP: IP-Absenderadresse der Registrierungsanfrage
- DST_IP: IP-Zieladresse der Registrierungsanfrage
- count: Anzahl der zu sendenden Registrierungsanfragen
- wait: Zeit, die nach jedem Paket gewartet werden soll in Mikrosekunden
- M: MN_IP bei jedem Paket um 1 hochzählen
- I: ID1 bei jedem Paket um 1 hochzählen
- S: SRC_IP bei jedem Paket um 1 hochzählen
- F: FA_IP bei jedem Paket um 1 hochzählen
- U: Die unbekannte HUT-Extension nicht senden[1]
- V: vor der korrekten Authentifizierungs-Erweiterung eine falsche senden
- N: nach der korrekten Authentifizierungs-Erweiterung eine falsche senden
- A: korrekte Authentifizierung weglassen
- H: HA_IP bei jedem Paket um 1 hochzählen

Anmerkungen zum Registrierungspaketgenerator:

- Der Paketgenerator berechnet UDP-Checksummen für das erste gesendete Paket. In den nachfolgenden Paketen eines Programmlaufs wird das Checksummenfeld mit dem Wert 0 gefüllt. Das heißt, der Empfänger sollte die Checksumme ignorieren.
- Die md5 C-Routinen wurden aus der HUT MIP-Implementierung (dynamics 0.6) übernommen.
- Standardmäßig sind die Parameter so eingestellt, dass der mobile Testrechner der GMD mit der IP 192.168.11.4 simuliert wird. Die Standardeinstellungen befinden sich jedoch gesammelt weit oben im Quelltext, so dass Änderungen einfach durchgeführt werden können.

1. In der Regel muss die unbekannte HUT-Extension ignoriert werden, da sie die Kennung 255 besitzt. Falls dies nicht der Fall ist, kann sie mit -U deaktiviert werden. Sie wurde ursprünglich in die Pakete aufgenommen, um diese einfacher mit den tatsächlichen HUT-Paketen vergleichen zu können.

Anmerkungen zur Verwendung des Registrierungspaketgenerators:

- Da die Registrierungspakete auf dem Weg zwischen mobilem Rechner und Fremdagent sowie Fremdagent und Heimatagent weitestgehend identisch sind, lässt sich der Generator sowohl an Stelle des Fremdagenten als auch an Stelle des mobilen Rechners einsetzen. Soll der Rechner einen oder mehrere Fremdagenten simulieren, muss die Zieladresse des IP-Headers entsprechend angepasst werden (ist standardmäßig der Fremdagent) außerdem muss eventuell das TTL-Feld erhöht werden (standardmäßig: 1).
- Damit der Generator Pakete korrekt versendet, muss der Empfänger im gleichen Subnetz sein wie der versendende Rechner. Das heißt auch, dass die Netzwerkkarte entsprechend konfiguriert sein muss. Unter Umständen kann man der Netzwerkkarte auch über den Befehl `route` mitteilen, dass sie an ein weiteres Subnetz angeschlossen wurde. Dann muss die Konfiguration der Karte selbst nicht geändert werden.
- Soll das Programm zur Hochlastsimulation mit Aufbau vieler Tunneldevices verwendet werden, müssen entsprechend viele verschiedene Adressen für die mobilen Knoten bereitstehen, deren Registrierung vom Heimatagent auch akzeptiert wird. Hierzu bietet sich an, im Heimatagent eine Netzwerkkarte oder ein NULL-device mit der IP-Adresse 2.0.0.0 und der Subnetmask 255.0.0.0 zu konfigurieren. Das Routing muß im Fremdagenten natürlich entsprechend angepasst werden. Ebenso muss der Heimatagent so konfiguriert werden, dass er sämtliche Anfragen positiv beantwortet. Am Beispiel von HUT müssen in einer Konfigurationsdatei sowohl die maximale Anzahl zugelassener gleichzeitiger Registrierungen hochgesetzt werden, als auch dem gesamten Netz 2.0.0.0 eine Berechtigung erteilt, IP-in-IP Verbindungen aufzubauen. Eventuell muss auch die Lifetime der Verbindungen hochgesetzt werden, um zu verhindern dass die ersten Tunnel schon abgebaut werden wenn die letzten noch generiert werden.

5.3.2 Registrierungsantwortgenerator

Syntax und Funktion

Für die Überprüfung der Leistungsfähigkeit von Fremdagenten ist es unter Umständen notwendig, viele Heimatagenten zu simulieren. Diese müssen auf die Registrierungsanfragen mit korrekten Antworten reagieren. Versuche, zunächst nur die Antworten zu senden sind fehlgeschlagen, da die Fremdagenten Tabelleneinträge immer erst nach Abgleich einer Antwort mit einer vorhergehenden Anfrage vornehmen.

Das entwickelte Programm untersucht alle empfangenen IP-Pakete, ob es sich um eine Registrierungsanfrage handelt. Dabei werden nur Teile des Headers überprüft, so dass auch Pakete akzeptiert werden, die an andere IP-Adressen gerichtet sind, aber auch Pakete, die zufällig an den untersuchten Positionen die gleichen Inhalte aufweisen wie ein typisches Mobile IP Registrierungspaket. Es findet hier keine Überprüfung der Gültigkeit der Registrierung statt! Zu jedem akzeptierten Paket wird ein Antwortpaket generiert und an den Absender zurückgesandt.

Die Syntax entspricht der des Registrierungspaketgenerators und ist im Folgenden dargestellt:

```
prom [-i ID1] [-p SPI] [-k key] [-w wait] [-I] [-V]
[-N] [-A]
```

(V und N schließen sich aus. Nicht jedoch A und N oder A und V.)

Dabei bedeuten:

- ID1: zweiter Teil des ID-Feldes (sollte eigentlich Zufallszahl sein)
- SPI: security parameter index (Zugriffsidentifikator für den geheimen Schlüssel)
- key: “shared secret” zwischen Heimatagent und mobilem Rechner
- wait: Zeit, die nach jedem Paket gewartet werden soll in Mikrosekunden
- I: ID1 bei jedem Paket um 1 hochzählen
- V: vor der korrekten Authentifizierungs-Erweiterung eine falsche senden
- N: nach der korrekten Authentifizierungs-Erweiterung eine falsche senden
- A: korrekte Authentifizierung weglassen

Anmerkungen zum Antwortpaketgenerator:

- Obwohl die Authentifikation der Registrierungsanfrage nicht überprüft wird, kann ein gültiger Authentifizierer für die Antwort generiert werden. Die MD5 C-Routinen wurden hierzu aus der HUT Mobile IP-Implementierung (dynamics 0.6) übernommen.
- Die UDP-Checksum scheint immer nur im ersten Paket das generiert wird korrekt zu sein, deshalb wurde sie behelfsmäßig auf Null gesetzt. Dieser Wert bedeutet, dass die UDP-Checksum im Empfänger nicht überprüft werden soll.

Anmerkungen zur Verwendung des Antwortpaketgenerators:

- Der Antwortpaketgenerator schaltet die Netzwerkkarte nicht selbständig in den promiscous-mode. Das heißt es werden nur Pakete empfangen, die an die MAC-Adresse der Netzwerkkarte gesandt werden. Sollen alle Pakete auf Leitung untersucht werden, muss die Netzwerkkarte in den promiscous-mode geschaltet werden. Dies geschieht unter RedHat zum Beispiel mit dem Befehl “ifconfig ethx promisc” in einer root-shell.
- Zu berücksichtigen ist außerdem, dass die Registrierungspakete unter Umständen gar nicht bei dem Rechner ankommen, der die Heimatagenten simulieren soll, da er sich nicht tatsächlich an ihrer Position im Netz befindet. Hierzu bietet sich an, den “Simulationsrechner” an den Gateway des Fremdagenten zu hängen. Dieser sollte dann so konfiguriert sein, dass er die fraglichen Pakete an den Simulationsrechner weiterleitet. Außerdem sollte sich der Simulationsrechner nicht mit dem Fremdagenten in einem Subnetz befinden, da ansonsten die Registrierungsantworten von einer anderen MAC-Adresse kommen, als die MAC-Adresse, an die die Anfragen gesendet werden. Das hat in Versuchen zu Problemen geführt.[1]

1. Im GMD-Test-Rack bietet sich also an, für eine gewisse Zeit das Rack vom GMD-Netz zu trennen und statt dessen den Simulationsrechner an den Router anzuschließen. Zusätzlich muss dann noch der Router im Fremdagenten als Standard-Gateway eingetragen werden.]

6. Testergebnisse

Derzeit existieren zwei aktuelle Mobile IPv4 Implementierungen:

- Dynamics - HUT [2] Mobile IP System der Technischen Universität Helsinki
- Mosquito Net [36] der Standford University

Beide Implementierungen wurden innerhalb der Testumgebung installiert und eingesetzt. Zusätzlich wurden mit Hilfe der HUT Implementierung die für den dritten Meilenstein entwickelten Testfolgen überprüft. Hierbei wurde festgestellt, dass die ursprünglich vorgesehene Reihenfolge und Aufteilung der Tests sinnvoller geordnet werden kann, um eine systematische Fehlersuche zu ermöglichen. Das wesentliche Ergebnisse der Testfolgen ist daher nicht das Austesten der Implementierungen, sondern die Entwicklung der Testfolgen selbst.

Wir haben uns aus zwei Gründen für diesen Weg, der von dem ursprünglichen Antrag abweicht, entschlossen. Zum einen hat die Entwicklung des Mobile IPv4 Bereiches gezeigt, dass zwar einige Implementierungen durch universitäre Gruppen entwickelt wurden, diese sind jedoch aus dem Status des Prototypen nicht herausgetreten. Zum anderen dienten diese Implementierungen den jeweils dazugehörigen Forschungsinhalten als “proof-of concept”. So wurde durch HUT ein hierarchisches Fremdagenten Konzept verwirklicht und getestet [34]. Mosquito Net dagegen implementiert das Konzept einer “mobile policy table” wodurch ein mobiles Endgerät eine Entscheidung darüber treffen kann, ob es zum Versenden seine IP-Pakete die “care-of” oder die “co-located care-of” Adresse verwenden [36]. Beide Implementierungen haben daher darauf verzichtet, den kompletten Funktionsumfang wie er in [10] spezifiziert wurde, zu implementieren.

Es erschien uns daher sinnvoller, ein Verfahren zu entwickeln, wonach eine “mobilitätsunterstützende Implementierung” getestet werden kann. Hierzu haben wir sowohl den Aufbau einer flexiblen Testumgebung als auch die Formalisierung und Implementierung der dazugehörigen Testfolgen gezält. Wir sind sicher, dass die hier erzielten Ergebnisse und Erfahrungen in folgenden Projekten verwendet werden können.

7. Mobile IP und PPP

7.1 Motivation

Das Point-To-Point Protokoll (PPP) unterstützt für verschiedene Instanzen von Mobile IP die Einwahl in Heimat- und Fremdnetz. Somit hat der Mobile Knoten die Möglichkeit, Mobile IP von verschiedenen Standorten zu verwenden. [32] spezifiziert eine Erweiterung für Mobile IP, die es mobilen Knoten erlaubt, sich in ein Netz einzuwählen und sich bei einem Fremdagenten zu registieren.

Für das definierte Anwendungsszenario "Hotel" (siehe Anwendungsszenario aus Meilenstein 3) besteht für jeden Teilnehmer die Möglichkeit, eine Einwahl in ein Wide Area Network (WAN) durchzuführen.

Im Folgenden wird zunächst ein allgemeiner Überblick über PPP gegeben. Anschließend wird die Installation und Konfiguration des im Anwendungsszenario eingesetzten Szenarios beschrieben. Daran anschließend wird die in [32] erwähnte Lösungsmöglichkeit erklärt. Abschließend werden die Testergebnisse und eine Zusammenfassung gegeben.

7.2 PPP Allgemein

Das Protokoll ermöglicht den Aufbau einer Verbindung über eine Telefon- oder ISDN-Leitung und die Übertragung verschiedener Protokolle über eine serielle Datenleitung. Zusätzlich verfügt es über eine Vielzahl von Funktionen wie Fehlerkorrektur oder die Möglichkeit der Zuweisung von Netzwerkadressen.

PPP besteht aus den folgenden drei Komponenten, auf die hier aber nicht näher eingegangen werden soll:

- Die Enkapsulierung der Datenpakete
- dem Link Control Protocol (LCP) zum Aufbau der Verbindung
- sowie verschiedenen Netzwerk-Protokollen

7.3 Architektur

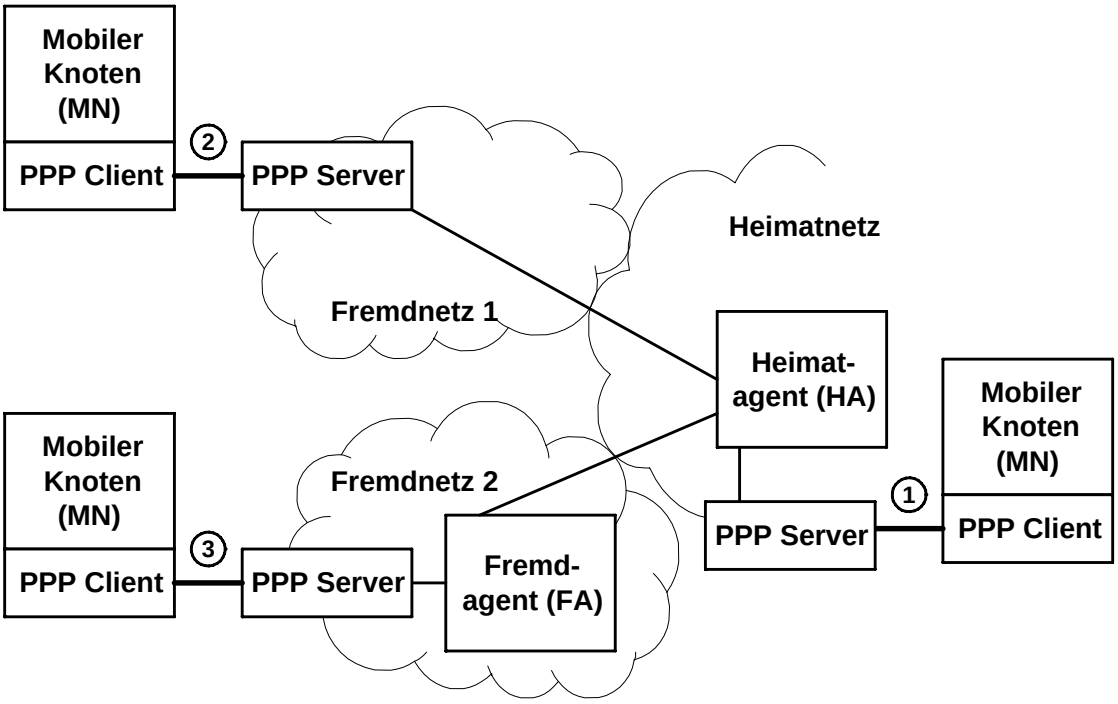


Abbildung 1. Architektur Mobile IP und PPP

Das oben dargestellte Schaubild (Abbildung 1, “Architektur Mobile IP und PPP”) zeigt drei Varianten, wie Mobile IP und PPP eingesetzt werden können:

1. Der Mobile Knoten (MN) wählt sich in das Heimatnetz ein (1)
2. Der MN wählt sich in ein Fremdnetz ein und findet keinen Fremdagenten (2)
3. Der MN wählt sich in ein Fremdnetz ein und findet einen FA (3)

Für das im Anwendungsszenario beschriebene Hotelszenario treffen im wesentlichen die Varianten (2) und (3) zu. Für (1) ist keine spezielle Konfiguration erforderlich, um eine Kommunikation mit dem Heimatagenten zu ermöglichen. Unser Augenmerk galt deshalb den zwei zuvor genannten Varianten.

7.4 Installation / Konfiguration

Der Einsatz von Mobile IP und PPP erfordert nur unwesentliche Veränderungen der bisherigen Infrastruktur (siehe Abbildung 1, “Architektur Mobile IP und PPP”). Die Installation eines PPP Servers wurde auf einem separaten Rechner umgesetzt, um die einzelnen Möglichkeiten zu testen.

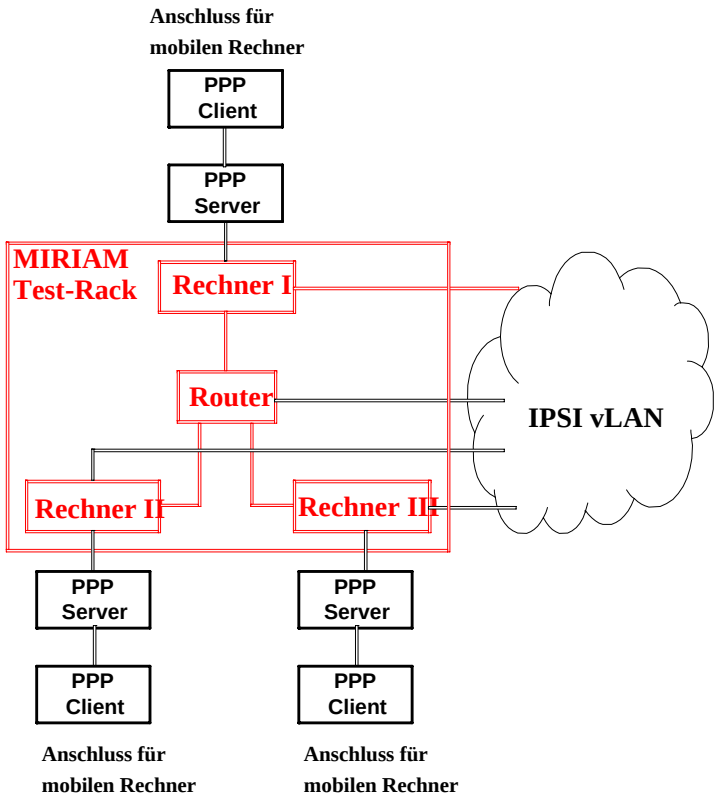


Abbildung 2. Erweiterte Schemazeichnung des Testrack

Die Konfiguration setzt voraus, dass entsprechende Anpassungen erfolgen für:

- PPP-Unterstützung für Kernel
- Konfiguration von zwei ISDN-Leitungen
- Installation und Konfiguration des Mobilen Knotens
- Routeneintragungen
- ISDN Support.
- Support für synchronous PPP

7.5 Registrierung eines Mobilen Knotens beim Fremdagenten

Die Untersuchung beschränkte sich auf die gezeigten Varianten (2) und (3) - es können aber grundsätzlich beliebig viele PPP-Verbindungen existieren.

Im Verlauf des Tests hat sich herausgestellt, dass die von uns vermutete Problematik in Bezug auf die Registrierungsanfrage, die über Broadcast gestellt, aber nicht über PPP weitergesendet wird nicht existiert: Über eine Konfigurationseinstellung leitet PPP Broadcast Pakete weiter.

Für den Fall, wenn sich der Mobile Knoten über PPP und über einen Fremdagenten beim Heimatagenten registrieren möchte, gibt [32] eine Lösung. Problem und Lösung werden

nachfolgend anhand der Abbildung 3, “Erfolglose Registrierung eines Mobilen Knotens” und Abbildung 4, “Erfolgreiche Registrierung eines Mobilen Knotens” erläutert werden.

Die nachfolgende Abbildung soll dieses Problem deutlich machen:

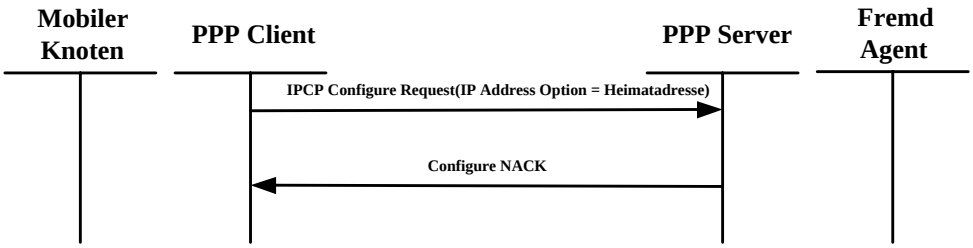


Abbildung 3. Erfolglose Registrierung eines Mobilen Knotens

Der Mobile Knoten möchte sich bei einem Fremdagenten registrieren. Über den PPP Client sendet er ein IPCP Configure Request an den PPP Server mit der IP Address Option seiner Heimatadresse. Der PPP Server nimmt die Anfrage entgegen und stellt fest, dass die Heimatadresse keine gültige IP Adresse für das Fremdnetz ist. Darauf sendet er eine Configure NACK (Negative Acknowledgement) zurück. Nach mehreren fehlgeschlagenen Versuchen (bis Timeout abgelaufen ist) geht der Mobile Knoten davon aus, dass sich kein Fremdagent im Fremdnetz befindet und verwendet eine Co Located Care-Of-Adresse, um sich beim Heimatrechner zu registrieren.

Wird die entsprechende Erweiterung, die in [32] beschrieben ist, verwendet, so besteht die Möglichkeit, sich bei einem Fremdagenten zu registrieren. Danach sendet dieser ein IPCP Configure Request an den PPP Server. Sie sieht vor, dass der Mobile Knoten zusätzlich zu dem IPCP Configure Request dem PPP Server mitteilt, dass es sich hierbei um eine Anfrage eines Mobilen Knoten handelt. Er übergibt ebenfalls die IP Adresse des Heimatnetzes. Diese Anfrage ist für den PPP Server also gültig (funktioniert nur bei den PPP Servern, die diesen [32] unterstützen). Er sendet dem Mobilen Knoten ein Configure ACK zurück und dieser hat jetzt die Möglichkeit, sich beim Fremdagenten zu registrieren.

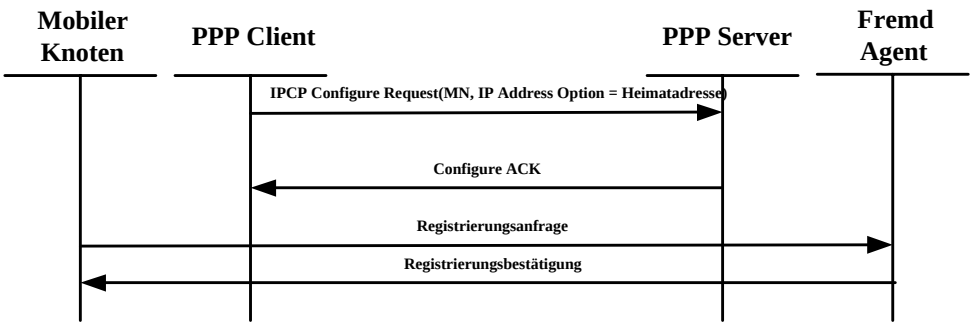


Abbildung 4. Erfolgreiche Registrierung eines Mobilen Knotens

7.6 Testdurchführung

Die von uns durchgeführten Tests waren erfolgreich für die Varianten (1) und (2). Jedoch war es uns nicht möglich, die dritte Variante funktionsfähig umzusetzen. Das Problem, das an dieser Stelle auftauchte, waren die Registrierungsnachrichten, die vom Mobilen Knoten gesendet wurden. Sie wurden nicht in der vorgegebenen Zeit zurückgesendet. Nach dem vorgegeben Timeout hat der Mobile Knoten eine Co-Located Care-Of-Adresse verwendet.

7.7 Zusammenfassung

Einem Mobilen Knoten soll ein Zugang über das PPP Protokoll in das Netz des Hotels gewährt werden. Für die Einsatz von Mobilen Knoten in Fremdnetzen empfehlen wir die Verwendung einer Co-Located Care-Of-Adresse. Es ist davon auszugehen, dass nur standkonforme PPP Implementierungen sich im Einsatz befinden und somit der Mobile Knoten keine Möglichkeit hat sich bei einem Fremdagenten zu registrieren. Der Registrierungsversuch bei einem Fremdagenten ist insofern nicht erfoderlich, da der Mobile Knoten nach dem abgelaufenen Timeout mit einer Co-Located Care-Of-Adresse operiert.

8. Sicherheit

Zusätzlich zu den in den vorangegangenen Meilensteinen vorgestellten Inhalten werden im Folgenden die im letzten Projektabschnitt durchgeführten sicherheitsrelevanten Tests beschrieben.

Schwerpunktmässig liegt hierbei der Fokus auf Angriffen, die auf die Verfügbarkeit von Mobile IP abzielen. Die zugehörige Angriffsart - Denial of Service - stellt insbesondere bei einem großflächigen Einsatz von Mobile IP eine herausragende Gefahr dar. Nach der kurzen wiederholenden Beschreibung von Denial of Service Angriffen aus Meilenstein 3 wird im folgenden der Schwerpunkt auf die Beschreibung durchgeführter Attacken gegen Mobile IP liegen.

Denial of Service Angriffe auf die Verfügbarkeit von Netzdiensten verhindern die normale Benutzung von Rechnern und Diensten, indem sie diese beispielsweise gezielt mit falschen Anfragen lahmlegen und im Hintergrund weitere Angriffe durchführen. Die gängigsten Denial of Service Angriffe umfassen:

- Das Senden von falschen Daten zu Netzdiensten oder Applikationen, was zu abnormalen Reaktionen bis hin zum Ausfall dieser Dienste führt
- Überfluten von Rechnern mit Anfragen bis diese auf Grund der Überlastung nicht mehr reagieren und vom Netz genommen werden müssen
- Blockade von gültigem Netzverkehr, so das autorisierte Benutzer keinen Zugriff erhalten

8.1 Mobile IP und Verfügbarkeit

Die Erbringung des Dienstes Mobile IP erfordert insbesondere die Verfügbarkeit aller Entitäten, die bei der Aushandlung bzw. Nutzung der Mobilitätsbeziehung beteiligt sind. Da Mobile IPv4 auf die Unterstützung von Heim- und Fremdagenten angewiesen ist, stellen diese besondere Angriffspunkte dar. Ebenfalls müssen die Protokollmechanismen zur Registrierung und zum Tunnelaufbau untersucht werden. Die Datenphase selbst ist durch das getunnelte Verschicken der Datenpakete gekennzeichnet. Hier stellt der Tunnel selbst ein weiteres Angriffsziel dar.

DoS Angriffe können von einem Angreifer oder bei Distributed Denial of Service Angriffen von mehrerern Angreifern ausgehen. Denial of Service Angriffe, die es schaffen, das Zielsystem vollständig auszulasten oder zum Absturz zu bringen bzw. das Netzwerk mit Antwortpaketen auf gefälschte Anfragen zu sättigen, können als erfolgreich bezeichnet werden. Ziel eines Angreifers ist die verwundbarste Stelle eines Systems zu finden - in einem bzgl. Verfügbarkeit optimierten Netz sollte der entsprechende Dienst auch weiterhin auf korrekte Anfragen antworten und die bösartigen Pakete mit möglichst geringer Prozessorlast und ohne Netzlast zu generieren verwerfen.

Die von uns durchgeführten Angriffe wurden größtenteils als Überflutungsangriffe durchgeführt. Überflutungsangriffe eignen sich besonders gut, um einen Angriff auf die Verfügbarkeit durchzuführen. Von einem Drittrechner werden wiederholt IP Pakete entweder zum HA, FA oder MN geschickt werden. Im Folgenden werden exemplarisch die wichtigsten von uns gegen das Schutzziel Verfügbarkeit durchgeführten Angriffe erläutert. Eine zweckmässige Unterscheidung über die einzelnen Angriffe hinaus kann

hierbei durch das Angriffsziel getroffen werden, welche sich in die Kontrollebene und die Datenebene unterteilt.

8.1.1 Replay Angriff (Kontroll- und Datenebene)

Bei diesem Test wird überprüft, wie sich Mobile IP bei einer Replay Attacke verhält. Dabei wird die integrierte Replay Protection auf ihre Funktion untersucht. Für diesen Test kann das Programm tcpreplay in der Version 1.0.1 verwendet werden.

Angriff:

Der MN hat sich über einen FA an seinem HA authentisiert und kann den vollen Funktionsumfang von Mobile IP benutzen. Jetzt wird auf der Verbindungsstrecke zwischen HA und FA eine Sequenz des Netzwerkverkehrs aufgezeichnet. Dazu wird das Programm tcpdump, welches zum Systemumfang von Linux gehört, verwendet. Durch den Aufruf tcpdump -i eth1 -w dump.tcp -n als Benutzer root wird der gesamte Netzwerkverkehr an der Netzwerkkarte mit dem device eth1 in der Datei dump.tcp gespeichert. Die Option -n besagt, dass die IP Adressen nicht in ihre Namen konvertiert werden sollen. Jetzt öffnet der MN zum Beispiel eine Telnet Session auf dem HA oder einen anderen Rechner (was problemlos funktionieren sollte). Nun benutzt der Angreifer das Programm tcpreplay und die zuvor erstellte Datei dump.tcp, somit wird der zuvor aufgezeichnete Netzwerkverkehr neu in das Netz eingespielt (siehe Abbildung 5, “Schemazeichnung für einen Angriffspunkt einer Replay Attacke”).

Der Programmaufruf wird ebenfalls mit dem Benutzer root durchgeführt und sieht wie folgt aus: tcpreplay -i eth1 -l 100 -r 0.01 dump.tcp. Es werden alle in der Datei dump.tcp gespeicherten Pakete an die Netzwerkkarte eth1 gesendet. Dabei wird eine Verzögerung von 0.01 Sek. verwendet. Die Pakete werden im Beispiel 100 mal versandt.

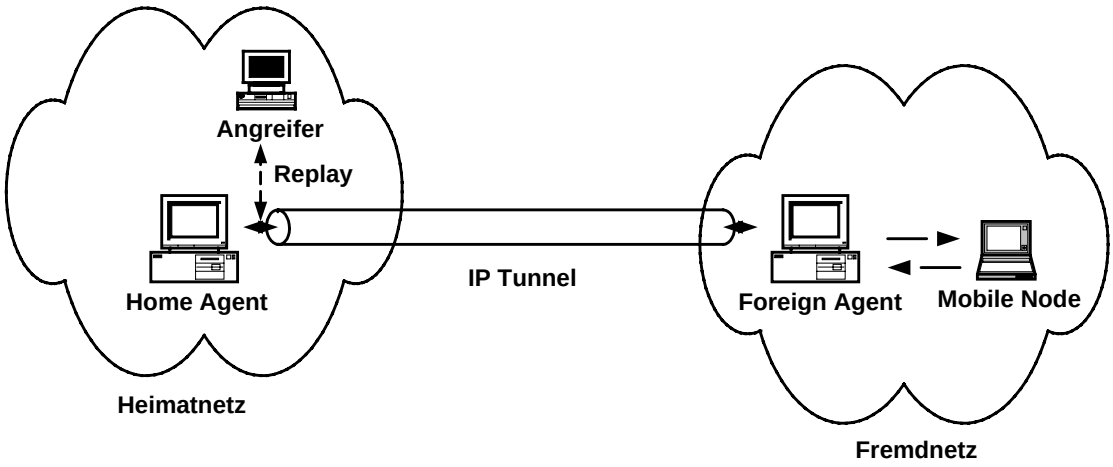


Abbildung 5. Schemazeichnung für einen Angriffspunkt einer Replay Attacke

Ergebnis:

Versucht der MN jetzt beispielsweise eine Telnet Session zu einen anderen Rechner aufzubauen, funktioniert das nicht. Er reagiert bei dem Versuch sich anzumelden nicht mehr und Fehlermeldungen werden auch nicht ausgegeben. Wenn der Angreifer seinen Angriff stoppt, benötigt der MN einen Zeitraum von bis zu zwei Minuten, um die

Kommunikation wieder aufzunehmen oder sie bleibt dauerhaft gestört. Die in Mobile IP integrierte Replay Protection weisst in der von uns untersuchten Implementierung also noch Schwachstellen auf.

8.1.2 Überflutung mit gefälschten Agent Advertisment Nachrichten (Kontrollebene)

An dieser Stelle wird untersucht, wie sich Mobile IP verhält, wenn ein Angreifer gefälschte Agent Advertisment Pakete (welche zur Bekanntmachung des FA dienen) an den MN sendet. Dazu sendet der Angreifer (wie in Abbildung 6, “Angriff durch gefälschte Agent Advertisment Pakete” zu sehen) gefälschte IP Pakete mit dem ICMP Anhang Agent Advertisment an den MN. Da der MN nicht in seinem Heimatnetz ist, werden diese vom HA an den FA getunnelt. Dort entkapselt, werden sie an den MN weitergeleitet und von diesem empfangen. Diese Pakete teilen dem MN mit, an welcher IP Adresse er einen MA findet. Die Pakete des Angreifers beinhalten als Nachricht für den MN eine falsche IP Adresse des Mobilen Agenten und eine gefälschte Absenderadresse. Die Pakete werden fortlaufend an den MN gesendet (weitere gleichartige Angriffsmöglichkeiten würde ein Versenden der Advertisments direkt im Fremdnetz bzw. ein Einspielen dieser Pakete in den Tunnel darstellen).

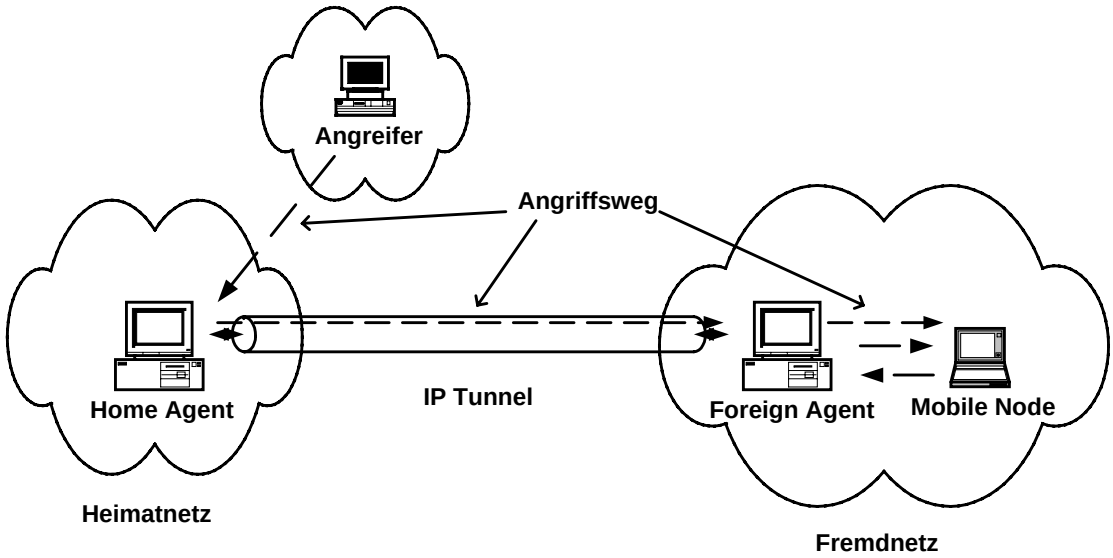


Abbildung 6. Angriff durch gefälschte Agent Advertisment Pakete

Für diesen Test wird ein Programm verwendet, das von uns entwickelt wurde. Dieses Programm generiert eine Struktur, die am Anfang einen IP Header, gefolgt von einem ICMP Header, besitzt. Die Variablen des IP Headers legen die Ziel IP Adresse, den Absender und weitere Werte fest. Im ICMP Header ist definiert, dass es sich hierbei um ein Agent Advertisment handelt. Deshalb werden die Variablen der Struktur auf die entsprechenden Werte gesetzt. Mit Hilfe einer Schleife wird diese Struktur, die gleichzeitig dem IP Paket entspricht, fortlaufend in das Netzwerk gesendet. Der MN, der diese IP Pakete empfängt, verhält sich wie folgt.

Das Werkzeug top zeigt an, wie die CPU dadurch belastet wird. In der Abbildung 6.3 ist zu sehen, dass die CPU Last auf 98,6% steigt.

```
12:20pm up 9 days, 22:07, 5 users, load average: 1.97, 1.57, 1.23
31 processes: 29 sleeping, 2 running, 0 zombie, 0 stopped
CPU states: 53.6% user, 46.3% system, 0.0% nice, 0.0% idle
Mem: 47184K av, 40508K used, 6676K free, 12040K shrd, 21576K buff
Swap: 100760K av, 2280K used, 98480K free 8812K cached
PID    USER PRI NI SIZE  RSS SHARE STAT LIB %CPU %MEM TIME COMMAND
7155   root  11  0  512  496   396    R   0 98.6  1.0 2:44  dynmnd
8203  alang   2  0 1000 1000   808    R   0  1.3  2.1 0:00     top
1      root   0  0  160  116    96    S   0  0.0  0.2 0:02    init
2      root   0  0    0    0     0   SW   0  0.0  0.0 0:00 kflushd
3      root   0  0    0    0     0   SW   0  0.0  0.0 0:00 kupdate
4      root   0  0    0    0     0   SW   0  0.0  0.0 0:00   kpiod
5      root   0  0    0    0     0   SW   0  0.0  0.0 0:00  kswapd
240     bin   0  0  380  356   304    S   0  0.0  0.7 0:00 portmap
256    root   0  0  272  248   216    S   0  0.0  0.5 0:00    apmd
309    root   0  0  208  156   116    S   0  0.0  0.3 0:00 syslogd
320    root   0  0  448  164   128    S   0  0.0  0.3 0:00   klogd
336  daemon   0  0  144  104    76    S   0  0.0  0.2 0:00     atd
352    root   0  0  164  108    80    S   0  0.0  0.2 0:00   crond
367    root   0  0  152    0     0   SW   0  0.0  0.0 0:00 cardmgr
383    root   0  0  216  172   152    S   0  0.0  0.3 0:00   inetd
```

Abbildung 7. Auslastung während eines Agent Advertisment Angriff

Wenn der MN gleichzeitig ein ping zu seinem HA ausübt, verhalten sich die Antwortzeiten wie es in Abbildung 8, “Senden von gefälschten Agent Advertisment Nachrichten” zu sehen ist. Das Programm ping sendet ICMP Pakete (Echo Request) an den Zielrechner und dieser antwortet ebenfalls mit einem ICMP Paket (Echo Reply) darauf. Die Pakete werden anhand von Sequenznummern nummeriert, um festzustellen, welche Pakete auf dem Übertragungsweg verloren gehen und wie lange die entsprechenden IP Pakete unterwegs sind.

```
PING 192.168.11.1 (192.168.11.1) from 192.168.11.4 : 56(84) bytes of data.
64 bytes from 192.168.11.1: icmp_seq=0 ttl=254 time=1.2 ms
64 bytes from 192.168.11.1: icmp_seq=1 ttl=254 time=1.1 ms
64 bytes from 192.168.11.1: icmp_seq=2 ttl=254 time=1.0 ms
64 bytes from 192.168.11.1: icmp_seq=3 ttl=254 time=1.0 ms -Start Angriff-
64 bytes from 192.168.11.1: icmp_seq=9 ttl=254 time=145.7 ms
64 bytes from 192.168.11.1: icmp_seq=11 ttl=254 time=215.0 ms
64 bytes from 192.168.11.1: icmp_seq=27 ttl=254 time=254.8 ms
64 bytes from 192.168.11.1: icmp_seq=31 ttl=254 time=1.0 ms -Stop Angriff-
64 bytes from 192.168.11.1: icmp_seq=32 ttl=254 time=0.9 ms
--- 192.168.11.1 ping statistics ---
33 packets transmitted, 9 packets received, 72% packet loss
round-trip min/avg/max = 0.9/69.0/254.8 ms
[alang@pc-rutile alang]$
```

Abbildung 8. Senden von gefälschten Agent Advertisment Nachrichten

Die ersten vier ping Pakete zeigen die Antwortzeiten des Mobile IP Protokolls ohne Netzlast und auch ohne einen Angriff. Das fünfte ICMP Paket (icmp_seq=4) fehlt. Zu diesem Zeitpunkt wurde der Angriff gestartet und der MN wurde mit Agent Advertisment Paketen generiert. Der Angriff endet zu dem Zeitpunkt des 32. Pakets (icmp_seq=31). Ab dann sind die Antwortzeiten wieder akzeptabel. Dazwischen ist zu sehen, dass viele Pakete verloren gegangen sind. Einige wenige (icmp_seq=9, 11, 27) haben den MN wieder erreicht und teilen mit, dass die entsprechenden Antwortzeiten sich um das 200-fache erhöht haben.

Die Überflutung mit gefälschten Solicitation Paketen werden im folgenden nicht näher erläutert, da der Angriff die angegriffenen Systeme nicht nachhaltig beeinflussen konnte.

8.1.3 Getunneltes Ping Flooding (Datenebene - Tunnel)

Im Folgenden wird als Referenz für die nachfolgenden Angriffe ein Überflutungsangriff mittels ping dargestellt. Das verwendete Programm zum getunnelten Ping Flooding generiert ein IP Paket, welches eine Struktur besitzt, die in der Abbildung 9, “Aufbau eines getunnelten Ping IP Paket” dargestellt ist. In dem äußeren IP Header steht als Empfänger die IP Adresse des FA und die Absender IP Adresse ist die eines HA da der FA nur getunnelte IP Pakete akzeptiert wenn ihm der HA bekannt ist. Der innere IP Header beinhaltet als Empfänger- und Sende IP Adresse die des MN. Die dort angehängte ICMP Struktur besitzt den Type 8 (Echo Request), was bedeutet, dass der Empfänger aufgefordert wird, auf das Echo Request mit einem Echo Reply (Type 0) zu antworten. Da als Sende- und Empfangs IP Adresse der MN eingetragen ist, antwortet er an sich selbst. Dadurch wird der MN mehr belastet, da er nicht nur die Antwort senden, sondern auch noch empfangen muss.

Dieser Angriff kann als Referenz für einen klassischen (für normale IP Systeme gedachten) Angriff auf ein Mobile IP System gelten.

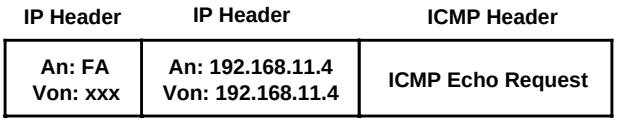


Abbildung 9. Aufbau eines getunnelten Ping IP Paket

Aus dem Angriff resultiert eine Systemlast, die in Abbildung 10, “Systemlast bei einem getunnelten Ping mit Überflutung” zu sehen ist. Diese Abbildung entspricht einem Bildschirmausdruck des Programms top. Das Programm zeigt den Systemstatus, die laufenden Prozesse und die dazugehörenden CPU-Zeiten sowie den Speicherplatzbedarf an.

```
625:19pm up 7 days, 3:07, 3 users, load average: 0.78, 1.04, 0.96
29 processes: 27 sleeping, 2 running, 0 zombie, 0 stopped
CPU states: 11.0% user, 56.8% system, 0.0% nice, 32.1% idle
Mem: 47184K av, 41272K used, 5912K free, 9876K shrd, 20916K buff
Swap: 100760K av, 1756K used, 99004K free 10564K cached


PID    USER PRI  NI SIZE  RSS SHARE STAT LIB %CPU %MEM TIME COMMAND
4996   root  17   0  688  688   576    R   0 59.8  1.4 1:10  dynmnd
5073   root   3   0 1012 1012   820    R   0  7.3  2.1 0:16     top
1      root   0   0  160  116    96    S   0  0.0  0.2 0:02    init
2      root   0   0    0    0     0   SW   0  0.0  0.0 0:00 kflushd
3      root   0   0    0    0     0   SW   0  0.0  0.0 0:00 kupdate
4      root   0   0    0    0     0   SW   0  0.0  0.0 0:00   kpiod
5      root   0   0    0    0     0   SW   0  0.0  0.0 0:00  kswapd
240     bin   0   0  380  356   304    S   0  0.0  0.7 0:00 portmap
256    root   0   0  272  248   216    S   0  0.0  0.5 0:00    apmd
309    root   0   0  296  280   216    S   0  0.0  0.5 0:00 syslogd
320    root   0   0  448  164   128    S   0  0.0  0.3 0:00   klogd
336  daemon   0   0  144  104    76    S   0  0.0  0.2 0:00     atd
352    root   0   0  168  112    84    S   0  0.0  0.2 0:00   crond
383    root   0   0   76    0     0   SW   0  0.0  0.0 0:00   inetd
```

Abbildung 10. Systemlast bei einem getunnelten Ping mit Überflutung

Dabei wird nicht nur ein einzelnes IP Paket über den FA an den MN gesendet, sondern fortlaufend eine ganze Folge von Paketen. Es ist ersichtlich, dass der Prozess des Mobilen Daemons dynmnd eine Systemlast von 59,8% verursacht, die aus der benötigten Rechenzeit zur Verarbeitung der empfangenen und gesendeten Pakete entsteht. Die zwar starke jedoch nicht "extreme" Auslastung veranschaulicht, dass die klassischen IP basierten Angriffe das System nicht so stark belasten wie die speziell auf die Funktionsweise von Mobile IP angepassten Angriffe.

8.1.4 IP Tunnelmissbrauch gegen Rechner im Heimatnetz

Der Test des IP Tunnelmissbrauchs gegen Rechner im Heimatnetz ist dem vorhergehenden sehr ähnlich. Der Unterschied besteht aber darin, dass der Angreifer im Fremdnetz ist und durch den Tunnel Pakete an andere Rechner im Heimatnetz schickt. In Abbildung 11, “IP Tunnel als Angriffsweg im Heimatnetz” ist zu sehen, an welcher Stelle der Rechner plaziert ist.

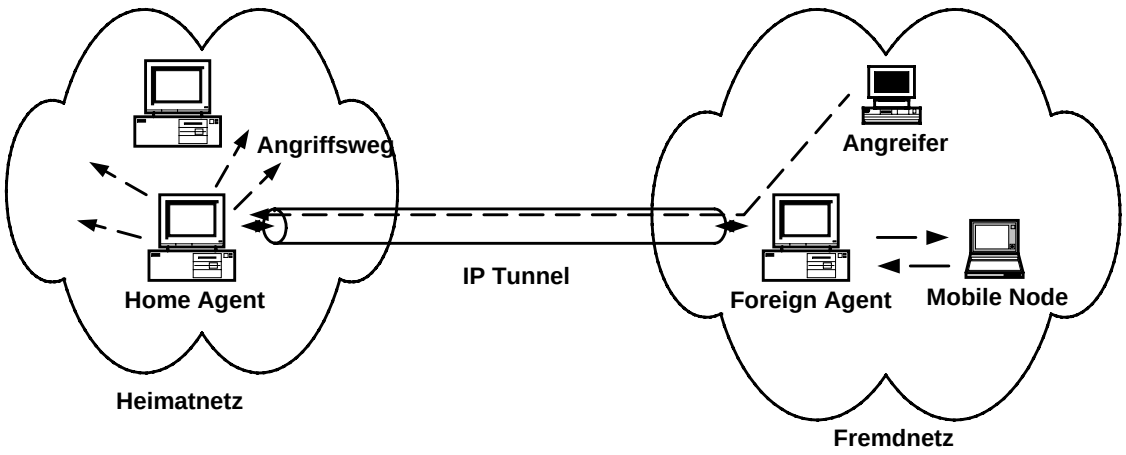


Abbildung 11. IP Tunnel als Angriffsweg im Heimatnetz

Die gesendeten Pakte des Angreifers sind ebenfalls IP-in-IP Pakete und haben den Aufbau gem. Abbildung 12, “Aufbau eines getunnelten Ping IP Pakets, in das Heimatnetz”. Der Zielrechner im Heimatnetz, der Mobile IP nicht kennt, besitzt die IP Adresse 192.168.11.6.

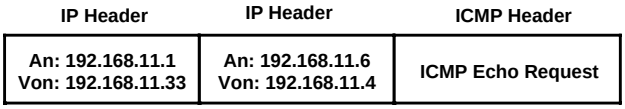


Abbildung 12. Aufbau eines getunnelten Ping IP Pakets, in das Heimatnetz

Vom Aufbau des IP-in-IP Pakets her ist es mit einem Paket, welches der MN sendet, identisch. Das führt auch dazu, dass es vom FA akzeptiert und an den HA weitergeleitet wird. Welche Wege das Paket nimmt ist im Folgenden zu sehen. Das ICMP Echo Request nimmt folgenden Weg:

```
Angreifer -> Router
Router -> Home Agent
Home Agent -> Jeder Rechner im Heimatnetz
```

Das ICMP Echo Replay nimmt folgenden Weg:

```
Rechner -> Foreign Agent
Foreign Agent -> Router
Router -> Home Agent
Home Agent -> Router
Router -> Jeden Rechner im Fremdnetz
```

Dieser Test zeigt eine nicht zu verachtende Schwachstelle. Wird zum Beispiel das Heimatnetz durch eine Firewall geschütz, welche nur IP Pakete durchlässt, die von dem FA stammen und an den HA gerichtet sind, kann ein Angreifer dadurch IP Pakete in das Heimatnetz zu anderen Rechnern bringen. Es fehlt eine Überprüfung, ob die entsprechenden IP Pakete auch wirklich vom MN und FA stammen oder nicht. Weiterhin wurde getestet, ob es auch möglich ist, in den IP Tunnel einzudringen um dort mit IP Spoofing gefälschte IP Pakete versenden zu können. In der Abbildung 13, “Einbringen eines IP Pakets in einen IP Tunnel” ist dargestellt, dass der Angreifer in diesem Fall nicht im Heimat- oder Fremdnetz sein muss, um einen Angriff durchzuführen. Diese Problematik kann mit den gleichen Programmen getestet werden, die im vorhergehenden Test benutzt wurden.

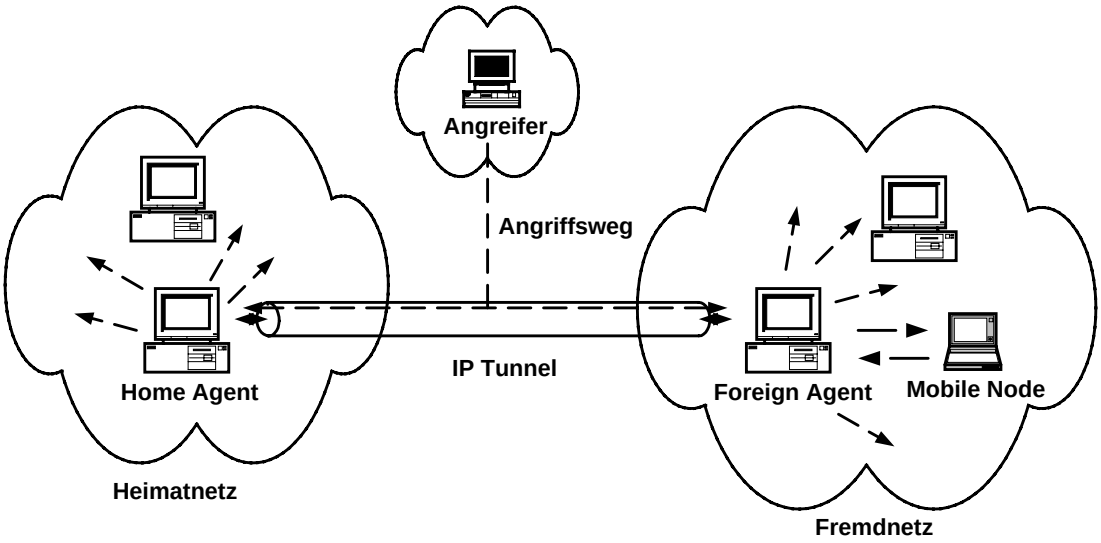


Abbildung 13. Einbringen eines IP Pakets in einen IP Tunnel

Das Ergebnis dieses Versuches ist, dass die IP Pakete aus einem anderen Subnetz in den IP Tunnel gebracht und an den HA oder FA geschickt werden können. Der Agent behandelt das Paket so, als käme es von der Gegenseite des IP Tunnels und sendet es entsprechend der IP Adressen weiter. Ein Angreifer hat demzufolge die Möglichkeit, IP Pakete an einen beliebigen Rechner im Heimatnetz des MN oder einen beliebigen Rechner im Fremdnetz des MN zu senden.

Angriffe, die dergestalt gegen den Tunnel und somit die Datenebene gerichtet sind, können durch den Einsatz von IPSec effektiv unterbunden werden; Angriffe auf die Verfügbarkeit werden in diesem Fall von den IPSec Instanzen erkannt und ausgefiltert (und belasten somit nicht den Mobile IP Daemon sondern die IPSec Komponenten).

8.1.5 IP Tunnelmissbrauch gegen Rechner im Fremdnetz

Der Test IP Tunnelmissbrauch gegen Rechner im Fremdnetz soll zeigen, dass es möglich ist, IP Pakete an Rechner zu senden, die weder im eigenem Subnetz noch als mobile Rechner eingesetzt werden. Die Struktur des Netzes und der Anschluss der jeweiligen Rechner ist in der Abbildung 14, “IP Tunnel als Angriffsweg im Fremdnetz” zu sehen. Der Angreifer befindet sich im Heimatnetz und möchte IP Pakete an einen Rechner im Fremdnetz schicken. Der Angreifer könnte die Pakete an den Zielrechner direkt senden, aber hier soll gezeigt werden, dass es auch mit Hilfe von Mobile IP möglich ist.

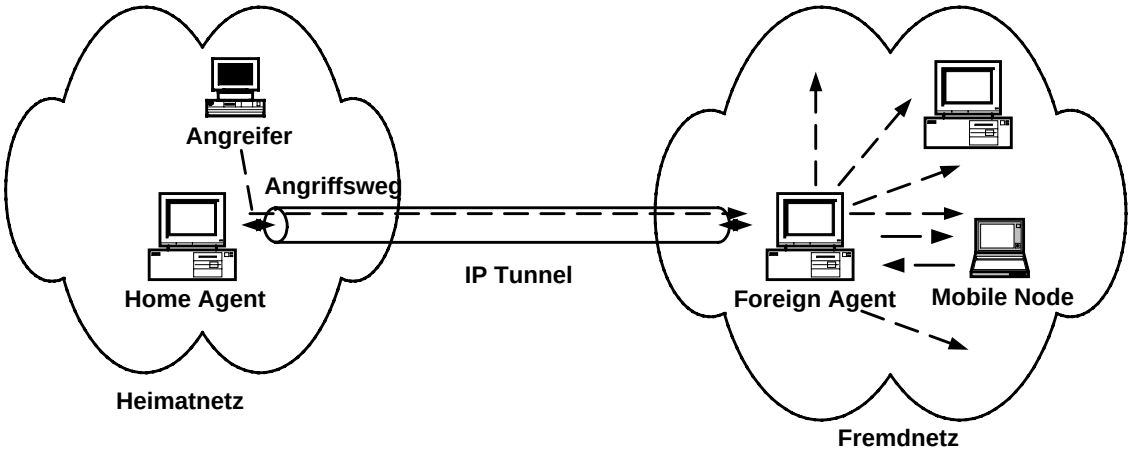


Abbildung 14. IP Tunnel als Angriffsweg im Fremdnetz

Der Angreifer sendet IP-in-IP Pakete mit der Sendeadresse des HA und der Empfängeradresse des FA. Das innere IP Paket ist an den MN gerichtet und trägt als Absenderadresse die eines anderen Rechners im Fremdnetz. Daran ist ein ICMP Echo Request angehängt, um zu sehen, ob der MN auch zu dem Fremdrechner antwortet. Der Fremdrechner im dritten Subnetz besitzt die IP Adresse 192.168.11.35. Diese Struktur des Pakets wird in der Abbildung 15, “Aufbau eines getunnelten Ping IP Pakets, in das Fremdnetz” verdeutlicht.

IP Header	IP Header	ICMP Header
An: FA Von: HA	An: 192.168.11.4 Von: 192.168.11.35	ICMP Echo Request

Abbildung 15. Aufbau eines getunnelten Ping IP Pakets, in das Fremdnetz

Wird während des Sendens eines Pakets mit dieser Struktur vom Angreifer aus, das Netzwerk überwacht, ist zu sehen, welche Wege das IP Paket nimmt. Anhand der folgenden Darstellung wird der Weg des Pakets verdeutlicht.

Das ICMP Echo Request nimmt folgenden Weg:

```
Angreifer -> Router
Router -> Foreign Agent
Foreign Agent -> Mobile Node
```

Das ICMP Echo Replay nimmt folgenden Weg:

```
Mobile Node -> Foreign Agent
Foreign Agent -> Router
Router -> Home Agent
Home Agent -> Router
Router -> Jeden Rechner im Fremdnetz
```

Tunnelmissbrauch kann durch IPSec vorgebeugt werden, hier trägt IPSec dann die Last den Angriff auszufiltern.

8.1.6 Entführen des Tunnels

Das Entführen (Hijacking) von Verbindungen in Form von Man-in-the-Middle Angriffen ist für beispielsweise Telnet Sitzungen wohlbekannt. Es gibt hierzu ein Werkzeug des Autors Pavel Kraus mit dem Namen hunt, welches zur Zeit in der Version 1.5 vorliegt. Mit Hilfe dieses Werkzeuges ist es möglich, eine bestehende Telnetverbindung zu belauschen, zurückzusetzen oder zu entführen. Dabei wird der gesamte TCP Verkehr mitgehört. Bei Bedarf werden TCP Pakete in das Netzwerk eingespielt, so dass die bestehende Telnetverbindung gestört wird. Der Angreifer kann sogar Zeichenfolgen oder Befehle an den Telnetserver schicken, um dort unter dem Namen des Telnetbenutzers Befehle auszuführen. Die Abbildung 16, “Man in the Middle Angriff mit hunt” veranschaulicht diesen Angriffstyp. Zusätzlich zu der beschriebenen Funktionsweise ist es natürlich auch möglich auf Sitzungen im Tunnel zuzugreifen (so dieser nicht gesichert ist), hierzu müsste jedoch das verwendete Programm angepasst sein.

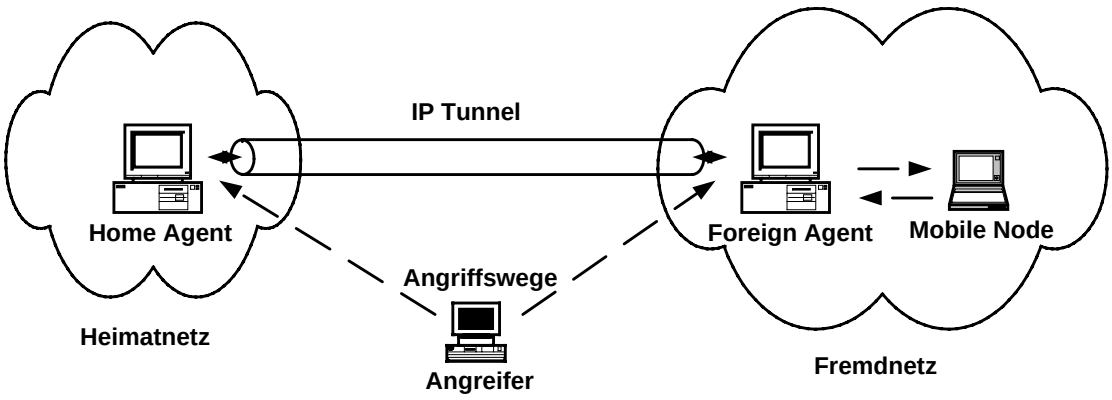


Abbildung 16. Man in the Middle Angriff mit hunt

Der Angreifer kann ohne Probleme die Kommunikation eines MN beobachten und entführen, wenn der MN eine Telnetverbindung aufbaut oder schon führt. Der erfolgreiche Hijackingangriff verdeutlicht die benötigte mehrstufige Sicherheit: Um sich gegen Angreifer im öffentlichen Netz zu schützen muss der Tunnel geschützt werden, um sich gegen Angreifer im Heim- oder Fremdnetz zu schützen müssen Ende-zu-Ende Sicherheitsmechanismen zum Einsatz kommen.

8.1.7 Angriffe auf das ARP Protokoll

Mobile IP stützt sich bei der Zustellung der Pakete - wie ein standardkonformes Ethernet bzw. Fast Ethernet auch auf das Address Resolution Protokoll um eine Zuordnung von Netzwerkadressen zu Hardwareadressen herzustellen. In Mobile IP übernimmt der HA darüber hinaus mittels eines Proxy ARP alle Pakete, die an mobile Systeme unterwegs sind, geschickt werden.

Die nicht vorhandenen Sicherheitsmechansimen des ARP Protokolls erlauben es einem Angreifer sich anstelle des HA als verantwortlich für den/die MNs auszugeben. Die

Funktionsweise des ARP Protokolls ist in Abbildung 17, “Adressauflösungsprotokolle: ARP, RARP” beschrieben.

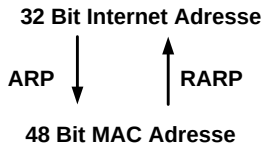


Abbildung 17. Adressauflösungsprotokolle: ARP, RARP

Das ARP Protokoll wird benutzt um die Zuordnung von IP Adresse zu MAC Adresse herzustellen. In der umgekehrten Richtung wird das Reverse ARP (RARP) Protokoll verwendet. Dazu werden ARP Anfragen (ARP Request) und ARP Antworten (ARP Reply) verwendet, wie in Abbildung 18, “ARP Request mit darauffolgendem ARP Reply” dargestellt. Eine Anfrage an die Ethernet Broadcastadresse bedeutet, dass das Paket an jeden Rechner in diesem Segment gesendet wird (ff:ff:ff:ff:ff:ff).

```
15:05:55.952545 arp who-has 192.168.11.5 (Broadcast) tell 192.168.11.14
15:05:55.952755 arp reply 192.168.11.5 is-at 0:60:8:23:5f:ca
```

Abbildung 18. ARP Request mit darauffolgendem ARP Reply

In diesem kurzen Auszug aus dem Netzverkehr ist zu sehen, wie ARP Anfragen und Antworten aussehen. Man sieht, dass der Rechner mit der IP Adresse 192.168.11.14 fragt, welche MAC Adresse der Rechner mit der IP Adresse 192.168.11.5 besitzt. Dabei stellt er die Anfrage an die Broadcastadresse. Die zweite Zeile zeigt, dass der Rechner mit der IP Adresse 192.168.11.5 antwortet und seine MAC Adresse (00:60:08:23:5f:ca) mitteilt.

Angriff:

Ist der MN nun in einem Fremdnetz und ein anderer Rechner möchte eine Verbindung zum MN aufbauen, werden die Pakete in das Heimatnetz geschickt. Dort fragt der Router, der das IP Paket zustellen würde, nach, welche MAC Adresse der MN hat. Der HA antwortet stellvertretende für den MN mit seiner eigenen MAC Adresse auf die Anfrage des Routers. Der HA wird deshalb auch ARP Proxy genannt, weil er die MAC Adressen aller abwesenden mobilen Rechner kennt und sich an deren Stelle zu erkennen gibt. Der Angriff wurde wie folgt durchgeführt. Ein korrespondierender Rechner will mit dem MN kommunizieren. Der MN befindet sich in einem Fremdnetz. Ein anderer Rechner führt währenddessen einen Angriff im Heimatnetz aus. Dabei sendete der angreifende Rechner fortlaufend gefälschte (spoofing) ARP Pakete mit dem Inhalt, dass der MN immer eine andere MAC Adresse hat. In Abbildung 19, “Gefälschte ARP Reply Pakete” ist ein Auszug aus dem Netzwerk zu sehen. Man sieht, dass der Rechner mit der IP Adresse 192.168.11.4 immer vorgibt, dass er die darauf folgende MAC Adresse besitzt. Diese ändert sich per Zufall in jedem gesendeten Paket. (Ein vergleichbarer Angriff wäre das Konfigurieren weiterer Rechner ausser dem HA als Proxy ARP für den MN)

```
15:58:09.360650 arp reply 192.168.11.4 is-at a3:97:a2:55:53:be
15:58:09.360709 arp reply 192.168.11.4 is-at f1:fc:f9:79:6b:52
15:58:09.360772 arp reply 192.168.11.4 is-at 14:13:e9:e2:2d:51
15:58:09.360848 arp reply 192.168.11.4 is-at 8e:1f:56:08:57:27
15:58:09.360908 arp reply 192.168.11.4 is-at a7:05:d4:d0:52:82
```

```
15:58:09.360978 arp reply 192.168.11.4 is-at 77:75:1b:99:4a:ed
15:58:09.361046 arp reply 192.168.11.4 is-at 58:3d:6a:52:36:d5
15:58:09.361115 arp reply 192.168.11.4 is-at 24:4a:68:8e:ad:95
15:58:09.361183 arp reply 192.168.11.4 is-at 5f:3c:35:b5:c4:8c
```

Abbildung 19. Gefälschte ARP Reply Pakete

Ständig und in einer hohen Anzahl schickt der Angreifer IP Pakete in das Netzwerk. Der korrespondierende Rechner möchte eine Verbindung zum MN aufbauen. Der Router benötigt dafür die MAC Adresse des MN. Wenn der Router nach der MAC Adresse des MN fragt (ARP Request), erhält er neben der korrekten Antwort des HA ständig falsche MAC Adressen vom angreifenden Rechner. Dadurch kann der Router die Pakete des korrespondierenden Rechner nicht weiterleiten, die Verbindung zwischen dem korrespondierenden Rechner und dem MN kommt nicht zu Stande.

8.1.8 Zusammenfassung Angriffe auf die Verfügbarkeit

Die dargestellten Angriffe auf die Verfügbarkeit von Mobile IP veranschaulichen das zusätzlich in den Kommunikationsprozess eingebettete Entitäten auch ein zusätzliches Risiko darstellen. Mobile IP als Erweiterung zu IP stellt mit relativ überschaubaren Protokollmechanismen und dem Komponenten HA, FA und MN ein deutlich “besseres” Angriffsziel dar als bisherige stationäre IP basierte Rechner. Die von uns untersuchten Implementierungen sowie der Einsatz des Basisprotokolls (ohne IPSec) sind als hauptverantwortlich für den Erfolg der Angriffe zu benennen. In realen Mobile IP Systemen werden Schutzmechansimen wie IPSec - die jedoch eine funktionierende AAA Struktur voraussetzen um einen skalierbaren Einsatz zu ermöglichen - eine deutlich gesteigerte Sicherheit bieten. Angriffe auf die Verfügbarkeit können dann, abhängig von den zum Einsatz kommenden IPSec Filterkomponenten, einerseits effizienter abgeblockt werden, andererseits werden sich neue Angriffspunkte bzgl. Verfügbarkeit ergeben wie z. B. der Schlüsselaustauschprozess der vor dem Einsatz von IPSec nötig ist.

8.2 Zusammenfassung Sicherheit in Mobile IP

Zusammenfassend kann Mobile IP bzgl. der Netzsicherheit ein enormes Gefährdungspotential attestiert werden. Diese konnte in den drei vorangegangenen Meilensteinen theoretisch gezeigt werden und wurde in diesem Meilenstein anhand des Aspektes Verfügbarkeit praktisch untermauert. Die grundlegende Architektur von Mobile IP sorgt für systemimmanente Schwachstellen, die selbst durch aufwändige Maßnahmen nur abgeschwächt, nicht aber vollständig beseitigt werden können. Die Funktionsweise von Mobile IP wird weiterhin durch bestehende Sicherheitsinfrastrukturen bzgl. Anwendbarkeit, Umsetzbarkeit und Leistungsfähigkeit stark beeinträchtigt.

Wie in der vorangegangenen Evaluation von Mobile IP gezeigt wurde, existieren (durchaus aufwändige) Mechanismen, um die Sicherheit sowohl auf Kontroll- als auch auf Datenebene zu unterstützen. Diese im vorangegangenen Meilenstein beschriebenen Ansätze ermöglichen, die im Kernstandard beschriebenen Sicherheitsmechanismen bzgl. Authentizität, Integrität und Vertraulichkeit deutlich zu erweitern (vgl. Tabelle 8, “Mögliche Sicherheit in Mobile IP”). Hierbei stellen sich jedoch aus administrativer Sicht mehrere Hindernisse dar. Firewallsysteme sind aktuell nur mit großem Aufwand oder

unter Schwächung der kompletten Netzsicherheit in der Lage, mit Mobile IP zusammenzuarbeiten.

Die in diesem Meilenstein dokumentierten Angriffe auf die Verfügbarkeit des Dienstes Mobile IP spiegeln auch die Problembereiche im realen Einsatz wieder - die Sicherstellung der Verfügbarkeit des Dienstes sollte gewährleistet sein, wenn sie mit kommerziellen Interessen oder notwendigen Systemanforderungen verbunden ist. Aktuelle Implementierungen und die Funktionsweise von Mobile IP an sich stellen hier die Schwachstellen dar.

Tabelle 8. Mögliche Sicherheit in Mobile IP

	Kontrollebene	**Datenebene**	**Einfluss auf bestehende Sicherheitsinfrastrukturen**
Authentizität	Authentisierung mittels SPI + Pre-shared Secret	IPSec	ja
Integrität	Zeitmarken, Zufallszahlen	IPSec	ja
Verbindlichkeit	nicht vorhanden	nicht vorhanden	nein
Verfügbarkeit	nicht vorhanden	nicht vorhanden	nein
Vertraulich	IPSec	IPSec	ja

Die in Meilenstein 3 vorgestellten Designkriterien für zukünftige AAA Architekturen bieten als Kompromiss sowohl für das Heimatnetz als auch für das Fremdnetz eine grundlegende Sicherheit, die heutigen Anforderungen entspricht. Der hierbei administrativ zu betreibende Aufwand ist jedoch relativ hoch, insbesondere, weil bestehende Mobile IP Implementierungen im Zusammenspiel mit IPSec bisher kaum erforscht sind.

Zukünftige Mobilitätsansätze auf Vermittlungschicht (Mobile IP für IPv6) sehen abweichend von der bisher schwerpunktmässig untersuchten Foreign Agent basierten Mobilitätsunterstützung vor, dass der Mobilen Knoten selbst mittels Co-Located Care-Of Adresse als Tunnelendpunkt eingesetzt wird. Diese Konfiguration kommt aktuell, insbesondere aus dem Grund, dass ohne Foreign Agent kaum Kontrollmechanismen für das Fremdnetz bestehen, um mit mobilen Rechnern umzugehen, kaum zum Einsatz und wird in Zukunft Bestandteil weiterer Sicherheitsbetrachtungen sein.

9. Integration von SLP

9.1 Beschreibung

Im vorangegangenen Meilenstein 3 wurde die Funktionsweise von SLP beschrieben. Der Schwerpunkt dieses Meilensteins liegt auf die Evaluierung der definierten Funktionen, die in der API des Service Location Protocols in RFC 2614 standardisiert sind. Hierfür haben wir verschiedene Implementierung untersucht.

9.2 Implementierungen

Tabelle 9, “SLP Implementierungen” zeigt die aktuell verfügbaren Implementierungen:

Tabelle 9. SLP Implementierungen

Nr.	Implementierung	Prog.	OS	Beschreibung
1	http://www.cs.columbia.edu/~zwb/project/slp/mslp.html	Java	alle die eine Java VM unterstützen	Mesh-enhanced Service Location Protocol (mSLP) beschreibt einen SLP-Prototyp für das Cache Aktualisiserungsproblem.
2	http://www.sun.com/research/slp/	C/C++	ab Solaris 2.6 ab NT 4.0 ab Linux 2.0	Kommerzielle SLP-Implementierung von SUN Microsystems. Unterstützt SLPv1
3	http://www.srvloc.org	C/C++	für Windows95/98/2000/NT	Kommerzielle SLP-Implementierung von James Kempf
4	http://www.openslp.org/	C/C++	Linux	OpenSLP ist eine Implementierung von Caldera, die die Funktionalitäten des SLPv2 Standards implementiert hat.

Nachfolgend wird von uns eine grobe Einschätzung zu den verschiedenen Implementierungen gegeben:

Mesh-enhanced Service Location Protocol ist eine Implementierung von Weibin Zhao von der Columbia University. Für sie spricht der plattformübergreifende Einsatz. Dem Leistungstest von 9000 registrierten Diensten war diese Implementierung jedoch nicht gewachsen. Deshalb ist nach unserer Einschätzung diese Implementierung für den kommerziellen Einsatz nicht geeignet (Abbildung 12, “Leistungstests”).

Die *Implementierung von SUN* unterstützt nur SLPv1. Bestimmte Tests, wie z. B. der Einsatz von mehreren DAs konnten nicht durchgeführt werden, da SLPv2 hierfür vorausgesetzt wird. Die Implementierung von SUN wurde nicht für SLPv2 erweitert und ist somit auch nicht zu empfehlen.

Seit Ende letzten Jahres ist eine *Implementierung von James Kempf* verfügbar. Sie bietet einen Einsatz auf Windowsplattformen an. Diese Implementierung stellt keinen Quellcode zur Verfügung, jedoch sind Schnittstellen mitgeliefert. Nach unserer Einschätzung stellt sie den am weitgehendst fertigen Code zur Verfügung und scheint uns am stabilsten. Ein wesentlicher Nachteil ist jedoch, dass sich diese Implementierung nur für Windows-Betriebsysteme eignet.

Für die Untersuchung der verschiedenen Funktionen haben wir uns für die *OpenSLP-Implementierung* von Caldera entschieden. Sie schien uns aus folgenden Gründen am geeignetsten. Das Paket liefert Quellcode für Windows, Solaris und Linux und implementiert SLPv2. Zusätzlich wird eine Testumgebung bereitgestellt. In der neuesten Version 0.82 gibt es noch einige Fehler in der Implementierung und es bleiben noch einige Punkte offen für SLPv2 wie Sprachunterstützung, Verwendung von Scopes und mehr (siehe http://www.openslp.org/roadmap.html).

9.3 Funktionen für SLP

Im Folgenden wird eine Übersicht über die definierten Funktionen gegeben, die als Basis für nachfolgende Tests verwendet werden. *SLPOpen* versucht DA oder SAs ausfindig zu machen, die nach einer bestimmten Vorgehensweise arbeiten (siehe “Auffinden von DA oder SA” auf Seite 58). *SLPClose* gibt alle Ressourcen wieder frei (SLP Handle Functions). Zum Registrieren und Deregistrieren werden die Funktionen *SLPReg* und *SLPDereg* bereitgestellt. Weiterhin werden für die Suche nach Diensten, Servicetypen und Attributen sogenannte Service Location Functions zur Verfügung gestellt (*SLPFindSrvs, SLPFindSrvTypes und SLPFindAttrs*). Konfigurationsfunktionen wie *SLPFindScopes* ermöglichen dem User Agent die Abfrage nach Scopes. Mit den Funktionen *SLPSetProperty* und *SLPGetProperty* können Eigenschaften der einzelnen Agenten gesetzt und abgefragt werden. Auf weitere Funktionen, die ebenfalls über die API bereitgestellt werden, wird an dieser Stelle nicht eingegangen, da diese bisher noch nicht implementiert wurden. Die übrigen Funktionen haben wir in Basistests sowie Leistungstest untersucht, die im nachfolgenden Abschnitt beschrieben werden.

9.4 Grundlegende und leistungsbezogene Tests

Als Basis für das Registrieren, Deregistrieren und Abfragen von Diensten ist das Auffinden von Kommunikationspartnern, entweder DA oder SAs, notwendig.

9.4.1 Auffinden von DA oder SA

Mit der Funktion *SLPOpen* wird ein Mechanismus zum Auffinden in Gang gesetzt. Dieser ist in vier Schritten aufgebaut. Ist einer dieser Schritte erfolgreich, so sind alle nachfolgenden Schritte unrelevant:

1. **DHCP und wahlfreie Optionen** - Die über das Dynamic Host Configuration Protocol zur Verfügung gestellte IP Adresse liefert mit wahlfreien Optionen zusätzliche Informationen über die im Netz erreichbaren Directory Agents (SLP Directory Agent Option: Code 78). Die Unterstützung von Scopes können über Code 79 in Erfahrung gebracht werden.
2. **Agent Discovery** - Eine weitere Möglichkeiten, die in den meisten Fälle erfolgreich ist, ist die Verwendung des Agent Discovery mit Multicast. Ein UA sendet beispielsweise eine Nachricht mit dem standardisierten Port 427 und der wohldefinierten Multicastadresse 239.255.255.253. Er fordert zusätzlich alle Instanzen, die die Nachricht "service:directory-agent" verstehen, auf, sich bei ihm zu melden.
3. **Statische Konfiguration** - Über eine statische Konfiguration können DAs vorab definiert werden (siehe Konfiguration eines SLP Daemons).

4. **Agent Discovery nach Service Agents** - Wird der vierte Mechanismus verwendet, so ist davon auszugehen, dass sich in dem gegenwärtigen Subnetz kein DA befindet. Danach wird versucht, existierende SAs ausfindig zu machen. Hierfür wird der Mechanismus Multicast Agent Discovery verwendet, jedoch wird in diesem Fall die Nachricht "service:service-agent" gesendet.

Verschiedene Fehlermeldungen, die während unserer Testdurchführung auftauchten, waren: keine Unterstützung für Multicast; der User Agent hat keine DAs oder SAs im Netz gefunden; da keine im Netz verfügbar waren; ungültige statische Konfiguration von IP Adressen.

9.4.2 Registrieren von Diensten

- SLPReg() - Diese Funktion ermöglicht eine Registrierung von Diensten.

Die Registrierung von Diensten kann von verschiedenen Dienstanbietern verwendet werden. Hierfür wird wieder das in Meilenstein 2 vorgestellte Service URL Schema verwendet.

allgemein:

SLPReg(<URL>, <TTL>, <servicetype>, <scopelist>, <attributlist>)

Beispiel:

SLPReg("service:telnet://miriam02", 5555, "service:telnet", "DEFAULT","(available=true)");

9.4.3 Deregistrieren von Diensten

- SLPDereg() - Funktion zur Deregistrierung von Diensten.

Die Deregistrierung von Diensten kommt eher selten zum Einsatz und findet vermutlich mehr Einsatz in der Mobilität. Vorraussetzung ist natürlich, dass mobile Geräte Dienste anbieten. In den meisten Fällen ist es jedoch so, dass der DA keine Einträge mehr über bestimmte Dienste hat, wenn die definierte TTL Zeit abgelaufen ist. Nachfolgend ein Beispiel für die Deregistrierung:

allgemein:

SLPDereg(<URL>, <TTL>, <scopelist>, <taglist>, <authentification>)

Beispiel:

SLPDereg("service:telnet://miriam02", 0, "DEFAULT","", "")

9.4.4 Suche nach Servicetypen

- SLPFindSrvTypes - Suche zum Auffinden von Servicetypen

Die Suche nach Servicetypen ist dann notwendig, wenn man verschiedene DAs oder SAs ausfindig gemacht hat und wissen möchte, welche dieser Instanzen welche Servicetypen unterstützen.

allgemein:

SLPFindSrvTypes(<prList>, <naming authority>, <scopelist>)

Beispiel:

SLPFindSrvTypes("*", "", "DEFAULT")

9.4.5 Suche nach einem Dienst

- SLPFindSrvs - Suche zum Auffinden von Diensten

Die Suche nach einem Dienst setzt die Kenntnis von Servicetypen sowie eine Scopeliste voraus.

allgemein:

SLPFindSrvs(<servicetype>, <scopelist>)

Beispiel:

SLPFindSrvs("service:telnet", "DEFAULT")

9.5 Testdurchführung

Für die Testdurchführung haben wir die existierende Infrastruktur von MIRIAM verwendet (siehe Meilenstein 3). Für das Starten des SLP-Daemons können verschiedene Parameter verwendet werden (Angabe der Konfigurationsdatei, Registrierungdatei oder Logdatei). So ist es z. B. möglich, eine statische Konfiguration über registrierte Dienste vorzugeben, die die Testdurchführung vereinfachen.

Nach dem Start des SLP-Daemons lädt er seine Konfigurationsdatei, die sich standardmäßig und /etc/slpd.conf befindet. In ihr befinden sich verschiedene Einstellungen, die vor dem Start festgelegt werden sollen. Über eine DA spezifische Konfiguration ist es möglich, den Daemon als DA zu konfigurieren. Statische festgelegte Scopes sowie DA Adressen sind ebenfalls möglich. Dies ist dann von Bedeutung, wenn der Daemon als Service Agent eingesetzt werden soll. Die Konfiguration wird in der Datei slpd.conf umgesetzt. Die Formatierung sowie der Inhalt richten sich nach der Spezifikation, die in RFC 2614 festgelegt ist. Der SLPD operiert als DA oder SA. Ebenso können Konfigurationseigenschaften wie Timeouts für Multicastanfragen oder die max. Lebenszeit von registrierten Einträgen, die Verwendung von Broadcast, Aktive und Passive DA Erkennung bestimmt werden. Im Abschnitt Sicherheit wird der Hinweis gegeben, dass bisher hierfür keine Unterstützung erfolgt.

Für die Überprüfung haben wir folgende Werkzeuge verwendet sowie Logdateien ausgewertet.

- **Ethereal** ist ein frei verfügbarer Protokollanalysierer für Unix und Windows. Pakete können entweder durch einen Live-Capture eingefangen werden oder aus einem File geladen werden. Die eingefangen Pakete werden detailliert aufgeschlüsselt und analysiert.
  Da Ethereal mittlerweile Informationen über den Aufbau der meisten SLP-Pakete besitzt, eignet es sich hervorragend zur Fehlersuche bzw. Funktionsprüfung im in der SLP Umgebung.
- **tcpdump** Ein textorientierter Netzwerkpacketanalysierer, der sich vor allem für die allgemeinere Überwachung des Datenflusses eignet.
- **nmap** Ein Portscanner, der für Windows und Linux/Unix verfügbar ist. Er erlaubt das Überprüfen von Netwerkressourcen auf Verfügbarkeit.

Die Basistests basieren auf typischen Fällen, die im normalen Einsatz auftauchen. Bei den nachfolgenden Tests ist davon auszugehen, dass sich im aktuellen Subnetz ein DA befindet.

Tabelle 10. Basistests

Test Nr.	Beschreibung	Durchführung	verwendet
1	Registrierung eines Dienstes und Suche nach Diensten von dem selben Servicetyp.	• Senden der Registrierung • Abfrage des selben Servicetyps	SLPOpen, SLPClose, SLPReg, SLPFindSrvs
2	Registrierung eines Dienstes und Suche nach Diensten von dem selben Servicetyp. Der URL String wird dannach in seine Bestandteile aufgesplittet.	• Senden der Registrierung • Abfrage desselben Servicetyps • URL parsen	SLPOpen, SLPClose, SLPReg, SLPFindSrvs, SLPParseSrvURL
3	Registrierung eines Dienstes und setzen eines Attributes. Suche nach diesem Dienst und Abfrage nach dem gesetzten Attribut.	• Senden der Registrierung • Abfrage desselben Servicetyps	SLPOpen, SLPClose, SLPReg, SLPFindSrvs, SLPFindAttrs
4	Registrierung eines Dienstes. Suche nach diesem Dienst. Deregistrierung des registrierten Dienstes.	• Senden der Registrierung • Abfrage des Dienstes • Deregiestrierung des Dienstes	SLPOpen, SLPClose, SLPReg, SLPFindSrvs, SLPDereg

Beim Einsatz von mehreren Instanzen haben wir sog. Testszenarien eingeführt. Diese überprüfen den Austausch von Registrierungsdatensätzen zwischen DAs, sowie die Registrierung gleicher Dienste an unterschiedlichen DAs und Deregistrierung von gleichen Diensten am gleichen DA.

Tabelle 11. Testszenarios

Test Nr.	Beschreibung	Durchführung	verwendet
1	Registrierung eines Dienstes bei DA1. Abfrage des Dienstes bei DA2.	• Senden der Registrierung an DA1 • Suche des Dienstes bei DA2	SLPOpen, SLPClose, SLPReg, SLPFindSrvs
2	Registrierung gleicher Dienste an unterschiedlichen DAs	• Senden der Registrierung an DA1 • Senden der Registrierung an DA2 • Suche des Dienstes bei DA1 • Suche des Dienstes bei DA2	SLPOpen, SLPClose, SLPReg, SLPFindSrvs
3	Registrierung eines Dienstes an einem DA. Deregistrierung des Dienstes an unterschiedlichen DAs	• Senden der Registrierung an DA1 • Suche des Dienstes bei DA1 • Suche des Dienstes bei DA2 • Senden der Deregistrierung an DA1 • Senden der Deregistrierung an DA2	SLPOpen, SLPClose, SLPReg, SLPFindSrvs, SLPDereg

Verschiedene Leistungstests, die wir durchgeführt haben, waren erforderlich, um die Belastung der DAs zu überprüfen. Für diese Tests haben wir ein Directory Agent im Einsatz, an den wir die verschiedenen Anfragen gerichtet haben.

Tabelle 12. Leistungstests

Test Nr.	Beschreibung	Durchführung	verwendet
1	Registrierung von 4000 Diensten. Suche nach zufällig ausgewählten Diensten.	• Senden der Registrierungen • Suche nach verschiedenen Diensten	SLPOpen, SLPClose, SLPReg, SLPFindSrvs
2	Registrierung von 9000 Diensten. Suche nach zufällig ausgewählten Diensten. Deregistrierung von zufällig ausgewählten Dienstes.	• Senden der Registrierungen • Suche nach verschiedenen Diensten • Deregistrierung von Diensten	SLPOpen, SLPClose, SLPReg, SLPFindSrvs, SLPDereg

9.6 Zusammenfassung

Aus unseren Tests hat sich ergeben, dass SLP für den Einsatz in heterogenen Umgebungen geeignet ist. Die einzelnen Implementierungen unterstützen jedoch noch nicht komplett die standardisierte Beschreibung von SLPv2. Es hat sich gezeigt, dass die OpenSLP Implementierung von Caldera die Implementierung ist, die das meiste Potenzial mit sich bringt.

Die Basistest waren alle erfolgreich und bieten somit eine gute Grundlage für eine Anwendung von SLP in Applikationen. Ausserdem hat sich herausgestellt, dass bei einer

Durchführung eines Leistungstests eine Timeout erfolgt, was allerdings scheinbar nur aus einer Überlast resultierte. Wenige Sekunden später nahm der DA erneut Anfragen an.

In den neueren Versionen von OpenSLP ist das Auffinden von DAs nach dem genannten Agent Discovery Verfahren vollständig implementiert, da in den vorhergehenden Versionen eine Findung nur über Multicast erfolgreich war.

In Zukunft gilt es, eine Infrastruktur für SLP so aufzubauen, dass auch verschiedene Anwendungen die verschiedenen Mechanismen nutzen. Weiterhin ist ein umfangreicher Test erforderlich, der die Skalierung der DAs überprüft.

10. Ergebnisse

Innerhalb des MIRIAM Projektes wurde Mobile IPv4 untersucht. Hierbei wurden speziell Stärken und Schwächen existierender Implementierungen innerhalb einer realen Umgebung getestet. Das Projekt sollte eine Einschätzung über den Entwicklungsstand sowie die Einsetzbarkeit von Mobile IPv4 geben. Untersucht wurden daher neben der Installierbarkeit, der Wartbarkeit auch das alltägliche Arbeiten im wissenschaftlichen Umfeld innherhalb einer mobilen Umgebung.

Ein wesentlicher Schwerpunkt des Projektes war es daher, an den beteiligten Instituten GMD und KOM eine Referenzinstallation aufzubauen. Hierfür wurden in einem ersten Schritt die folgende realen Altagssituationen identifiziert und im Meilenstein 1 dargestellt:

1. *GMD:* Das Arbeiten in einem fremden Zugangsnetz über Fest- und Funknetz, innerhalb der eigenen administrativen Domäne.
2. *KOM:* Subnetzwechsel von einem fremden Zugangsnetz in ein anderes über Fest- und Funknetz, innherhalb der eigenen administrativen Domäne.
3. *Konferenz:* Das Arbeiten in einem fremden Zugangsnetz über Fest- und Funknetz, ausserhalb der eigenen administrativen Domäne.
4. *Hotel:* Das Arbeiten in einem fremden Zugangsnetz über eine Punkt-zu-Punkt Verbindung.

Beim Aufbau dieser Referenzinstallation haben sich jedoch Schwierigkeiten ergeben. So wurde im Laufe des Projektes festgestellt, dass Firewall Techniken ein wesentlicher Hindernisgrund für den Einsatz von Mobile IP Implementierungen ausserhalb der eigenen administrativen Domäne sind. Das Senden mit topologisch nicht korrekten Adressen sowie der Aufbau von IP Tunneln werden durch die Firewalls in den meisten Fällen unterbunden. Hierdurch war es uns nicht möglich das ursprünglich geplante Konferenz Szenario durchzuspielen. Auch das Hotel Szenario musste auf den Heimbereich, d.h. innherhalb der eigenen administrativen Domäne verlegt werden. Statt dessen wurde jedoch ein Forschungsbedarf identifiziert, der sich mit der Ansteuerung von Firewalls zur dynamischen Freischaltung von Mobile IP Registrierungen und dem Datentransporten über IP Tunnels beschäftigt (vergl. Meilenstein 1 und 2). Dieser Forschungsbereich wurde bei KOM aufgegriffen und wird derzeit im Rahmen einer Promotion bearbeitet.

Wesentliche Ergebnisse durch die Konzeption und Realisierung der Referenzinstallation konnten daher nur innerhalb der eigenen administrativen Domäne gewonnen werden, wo wir in der Lage waren die Firewalls entsprechend unseren Anforderungen zu konfigurieren. Zusätzlich wurde hierbei noch das sogenannte Reverse Tunneling eingesetzt, so dass unsere Mobilen Endgeräte immer mit einer topologisch korrekten Adresse senden konnten. Als Ergebnis wurde festgestellt, dass der Zugriff auf ein Heimatnetz aus einem Fremdnetz heraus nach der Installation von Mobile IP ohne Probleme möglich ist. Dies galt sowohl nach dem Anschluss an ein LAN - über eine Fest- oder Funkverbindung (vergl. Meilenstein 2 und 3) - als auch über eine Punkt-zu-Punkt Verbindung (vergl. Meilenstein 4). Auch der Subnetzwechsel von einem funkunterstützten Fremdnetz in ein anderes konnte innerhalb der Referenzinstallation überprüft werden. Ergebnisse hierzu sind in Meilenstein 2 dargestellt. Zusätzlich wurden die Ergebnisse - wie im Antrag gefordert - auf einer geeigneten deutschen Veranstaltung, dem "Tag der

offenen Tür der GMD" in den Jahren 1999 und 2000 einem breitem Publikum vorgestellt. Es ausserdem Webseiten erstellt, so dass auf die Ergebnisse jederzeit zugegriffen werden kann.

Im Zuge des Projektes sollten neben der Einsetzbarkeit auch sämtliche verfügbaren Mobile IPv4 Implementierungen getestet werden. Hierbei hat sich im Laufe unserer Arbeit herausgestellt, dass ein Zielkonflikt zwischen der Referenzinstallation und den Tests besteht. Auf der einen Seite erforderte die Referenzinstallation stabil installierte Agenten und mobile Endsysteme, mit denen täglich gearbeitet werden konnte. Auf der anderen Seite mussten zum Testen regelmäßig neue Betriebssystem und Mobile IP Versionen eingespielt werden. Die beiden Aufgabenstellungen haben sich hierdurch bei der Durchführung behindert. Dieser Sachverhalt wurde ausführlich in dem dritten Meilenstein dargestellt.

Dieser Konflikt wurde durch die Schaffung einer mobilen Testumgebung aufgelöst. Hierfür wurde ein Test-Rack angeschafft, welches vier baugleiche Rechner über einen Switch verbindet. Mit Hilfe dieser Testumgebung konnten so in kürzester Zeit unterschiedliche Topologien konfiguriert und Mobile IP Implementierungen installiert werden. Aufgrund der Baugleichheit der Rechner mussten hierfür nur die Platten entsprechend gespiegelt werden. Die Testumgebung wurde in einem eigens geschaffenen Testsegment untergebracht, so dass ohne störende Firewall Techniken getestet werden konnte. Den Subnetzen wurden private Adressbereiche vergeben, so dass diese entsprechend flexibel konfiguriert werden konnten. Der Aufbau der Testumgebung wurde in Meilenstein 3 ausführlich beschrieben. Darüber hinaus wurden Testfolgen entworfen und implementiert, welche die Anforderungen des Mobile IPv4 Standards spezifizieren und überprüfen. Hierdurch wurden die Test der Mobile IPv4 Implementierungen formalisiert und konnten nach einem immer gleichen Schema ablaufen. Durch die Formalisierung der Test, ist es möglich die Ergebnisse und somit die Implementierungen zu vergleichen. Die Testfolgen und deren Implementierung sind in Meilenstein 4 beschreiben.

In Hinblick auf Installierbarkeit und Wartbarkeit konnten keine Ergebnisse gewonnen werden. Die derzeit frei verfügbaren Mobile IPv4 Implementierungen wurden alle an Forschungseinrichtungen entwickelt und dienen daher im wesentlichen dem *proof-of-concept* der dahinter stehenden Forschungsinhalten. Ein Großteil der Implementierungen sind veraltet und daher natürlicherweise nur mit einem erheblichen Aufwand zu installieren. Zudem liegt der Schwerpunkt bei keiner der Implementierung auf dem Aspekt der Benutzerfreundlichkeit. Dennoch kann gesagt werden, dass die aktuellen Implementierungen leicht zu installieren und zu warten sind. Details hierzu befinden sich in Meilenstein 3; Meilenstein 4 gibt zudem eine Tabelle der aktuellen Mobile IPv4 und IPv6 Implementierungen.

Ein Schwerpunkt im Rahmen des MIRIAM Projektes war die Untersuchung von Mobile IP bzgl. Sicherheit. Hierzu wurde in Meilenstein 2 die theoretische Evaluierung des Themenbereiches begonnen. Ausgehend von generischen Überlegungen zum Thema Netzsicherheit wurde Mobile IP anhand klassischer Schutzziele *(Authentizität, Integrität, Verbindlichkeit, Vertraulichkeit, Verfügbarkeit)* und zugehöriger Angriffe getrennt für die Kontroll- und Datenebene betrachtet. Die in Mobile IP bereits enthaltenen Sicherheitsmechanismen wurden vorgestellt. Ausgehend von existierenden

Lösungsansätzen zur besseren Sicherung von Mobile IP wurden am Beispiel des eigenen Testbetriebs Lösungsansätze entworfen. In Meilenstein 3 wurde die systematische Betrachtung der theoretischen Arbeiten zu Mobile IP und Sicherheit abgeschlossen. Die Ergebnisse - eine Bestandsaufnahme der Entwicklung von Mobile IP unter dem Gesichtspunkt Sicherheit - wurden im Rahmen eines Workshops auf der GI-Jahrestagung vorgestellt und publiziert. In Meilenstein 3 wurde ebenfalls der Themenkomplex zukünftige Entwicklung von Mobile IP bzgl. Sicherheit betrachtet. Hierbei wurde auf Mobile IP verwandte Themen mit Bezug zur Datensicherheit eingegangen (wie z. B. das AAA-Framework (Authentication, Authorization, Accounting) der IETF), um die Entwicklungsperspektive von Mobile IP aufzuzeigen. Der letzte Berichtsabschnitt des MIRIAM-Projekts umfasste die Erforschung des bisher im Zusammenhang mit Mobile IP nur unzureichend untersuchten Schutzzieles Verfügbarkeit. Die Mobilitätsunterstützung auf Netzwerkebene mit Mobile IP bietet insbesondere hinsichtlich der aktuellen "Popularität" von Denial of Service Angriffen systemimmanente Schwachstellen, die von uns in umfangreichen Testreihen untersucht und dokumentiert wurden. Meilenstein 4 fasst diese zusammen und stellt zum Projektabschluss am Ende des Kapitels Sicherheit noch eine zusammenfassende Betrachtung der wichtigsten sicherheitsrelevanten Aspekte von Mobile IP dar, die wir im Rahmen des MIRIAM Projekts hinterfragt und beleuchtet haben.

Ein weiterer Untersuchungs-Gegenstand des MIRIAM Projekts war der Einsatz des Service Location Protokolls in unserer mobilen Arbeitsumgebung. Die aus der Referenzinstallation gewonnenen Erfahrungen haben gezeigt, dass ein mobiler Knoten zwar mittels Mobile IP an ein Fremdnetz angeschlossen und ihm transparenten Zugriff auf sein Heimatnetz gewährt werden kann - er dann jedoch durch dieses Verfahren isoliert in dem entsprechenden Fremdnetz agiert. Hierdurch ergibt sich die Notwendigkeit einen Weg zu schaffen, wie der mobile Rechner auf die Dienste im Fremdnetz zugreifen kann. Innerhalb von MIRIAM wurde hierfür das Service Location Protokoll untersucht. Ausgehend von den grundlegenden Funktionen des Protokolls, wie der Registrierung, Deregistrierung und die Suche nach Diensten, wurde ein möglicher Einsatz in einer mobilen Arbeitsumgebung konzipiert und anhand existierenden Implementierungen evaluiert. Das Ergebnis dieser Untersuchung war, dass ein Mechanismus zur Dienstfindung und -bereitstellung eine hilfreiche Unterstützung für das mobile Arbeiten innerhalb der Referenzumgebung darstellt. Die Beschreibung dieser Arbeiten befinden sich in Meilenstein 2 und 3 (Grundlagen) sowie Meilenstein 4 (Evaluierung). Zusätzlich wurde die "Verschmelzung" des Heimat- und des Fremdnetzes zu einer "Welt" des mobilen Rechners als Forschungsgegenstand identifiziert. Hierfür ist es nötig Dienst als lokal bzw. entfernt zu spezifizieren und die IP Pakete in die entsprechende Netze zu routen. Mosquito Net, eine der untersuchten Mobile IPv4 Implementierung bietet hierfür einen entsprechenden Ansatz. Dieser wurde innerhalb dieses Meilensteins beschrieben.

Zusammenfassend kann gesagt werden, dass die Entwicklung von Mobile IPv4 von einem IETF Standard hin zu einem Produkt derzeit zu einem Stillstand gekommen ist. Mobile IPv4 fand in der Vergangenheit, trotz dramatischer Verbesserung von Notebooks bezüglich Größe, Gewicht und Leistungsfähigkeit sowie der wachsenden Bedeutung von Netzwerken, insbesondere des Internets, noch keine weite Verbreitung. Professionelle Anwender entwickeln eigene proprietären Mobile IP Implementierungen. Die Forschungsgemeinschaft dagegen beschäftigt sich derzeit hauptsächlich mit den

ungelösten Sicherheitsfragen und Weiterentwicklung von Mobile IPv6. Wurde durch Mobile IPv4 die Mobilität doch recht künstlich und auch durchaus umständlich in das IPv4 basierte Internet eingebracht, so wird sie bei IPv6 schon von Anfang an dazugehören und wesentlich eleganter realisiert sein. Konzeptionelle Annahmen, wie die Transparenz mobiler Kommunikation - ein wichtiges Design Kriterium von Mobile IPv4 - können ersetzt werden durch IPv6 fähige Knoten, die sich der Existenz mobiler Kommunikation bewusst sind und diese sogar unterstützen. Durch IPv6 werden viele offene Fragen bezüglich der Sicherheit, der Administrierbarkeit sowie der Performance gelöst. Daher sind wir der festen Überzeugung, dass sich Mobile IP mit der Einführung von IPv6 durchsetzen wird.

Referenzen

[1] Monarch, Mobile IPv4/IPv6 Implementation, FreeBSD, 2000. http://www.monarch.cs.cmu.edu/.

[2] HUT, Mobile IPv4 Implementierung, Linux, 2000.

[3] Secure MIP, Mobile IPv4 Implementation, FreeBSD/Linux, 2000.

[4] Binghamton, Mobile IPv4 Implementation, Linux, 1996.

[5] National University of Singapore, Mobile IPv4/IPv6 Implementation, Linux and Windows, 2000. http://mip.ee.nus.edu.sg/mipv6.

[6] SUN, Mobile IPv4 Implementation, Solaris, 1999.

[7] Lancaster, Mobile IPv6 Implementation, Linux, 1998. http://www6.cs-ipv6.lancs.ac.uk/start.htm.

[8] HMIPv6, Mobile IPv6 Implementation, FreeBSD, 2000. http://www.inrialpes.fr/planete/people/bellier/hmip.html.

[9] Politehnica University of Bucharest, Mobile IPv4 Implementation, Windows, 1999.

[10] C. E. Perkins. RFC 2002 - IP Mobility Support. Standards Track RFC, October 1996.

[11] IP Routing for Wireless/Mobile Hosts (mobileip), 2000. IETF Working Group. http://www.ietf.cnri.reston.va.us/html.charters/mobileip-charter.html.

[12] C. E. Perkins and D. B. Johnson. Route Optimization in Mobile IP. Internet Draft (draft-ietf-mobileip-optim-09.txt), February 2000. Work in Progress.

[13] C. E. P. David B. Johnson. Mobility Support in IPv6. Internet Draft (draft-ietf-mobileip-ipv6-13.txt), November 2000. Work in Progress.

[14] C. E. Perkins, D. B. Johnson, and N. Asokan. Registration Keys for Route Optimization. Internet Draft (draft-ietf-mobileip-regkey-03.txt), July 2000. Work in Progress.

[15] E. Gustafsson, A. Jonsson, and C. E. Perkins. Mobile IP Regional Registration. Internet Draft (draft-ietf-mobileip-reg-tunnel-03.tx), July 2000. Work in Progress.

[16] C. E. Perkins and P. R. Calhoun. AAA Registration Keys for Mobile IP. Internet Draft (draft-ietf-mobileip-aaa-key-03.txt), January 2001. Work in Progress.

[17] C. E. Perkins. IP Mobility Support for IPv4, revised. Internet Draft (draft-ietf-mobileip-rfc2002-bis-03.txt), September 2000. Work in Progress.

[18] Y. Xu, R. Bhalla, E. Campbell, K. Freter, E. M. Hadwen, G. Dommety, K. Joshi, P. Yegani, T. Matsumura, A. Teshima, L. D. Hyun, N. Itoh, K. Ohki, B.-K. Lim, P. J.McCann, T. Towle, J. Jayapalan, P. W. Wenzel, C. B. Becker, J. Jiang, S. Shikano, W. Kim, Y. Chang, B. Semper, J. M. Koo, M. A. Lipford, F. Leroudier, and J. Gately. Mobile IP Based Micro Mobility Management Protocol in The Third Generation Wireless Network. Internet Draft (draft-ietf-mobileip-3gwireless-ext-05.txt), November 2000. Work in Progress.

[19] M. M.Khalil, E. Qaddoura, and P. R. Calhoun. Generalized NAI Extension (GNAIE). Internet Draft (draft-ietf-mobileip-gnaie-01.txt), March 2001. Work in Progress.

[20] H. Soliman, C. Castelluccia, K. El-Malki, and L. Bellier. Hierarchical MIPv6 mobility management. Internet Draft (draft-ietf-mobileip-hmipv6-01.txt), September 2000. Work in Progress.

[21] J. Solomon. Applicability Statement for IP Mobility Support. Standards Track RFC, October 1996.

[22] C. E. Perkins. RFC2004 - Minimal Encapsulation within IP. Standards Track RFC, October 1996.

[23] C. E. Perkins. RFC 2003 - IP Encapsulation within IP. Standards Track RFC, October 1996.

[24] C. E. Perkins. IP Mobility Support. Standards Track RFC, October 1996.

[25] D. Cong, M. Hamlen, and C. E. Perkins. RFC2006 - The Definitions of Managed Objects for IP Mobility Support using SMIv. Standards Track RFC, October 1996.

[26] G. E. Montenegro and V. Gupta. Sun’s SKIP Firewall Traversal for Mobile IP. Standards Track RFC, June 1998.

[27] P. Calhoun and C. E. Perkins. RFC2794 - Mobile IP Network Access Identifier Extension for IPv4. Standards Track RFC, March 2000.

[28] S. Glass, T. Hiller, S. Jacobs, and C. E. Perkins. RFC2977 - Mobile IP Authentication, Authorization, and Accounting Requirements. Standards Track RFC, October 2000.

[29] C. E. Perkins and P. Calhoun. Mobile IPv4 Challenge/Response Extensions. Standards Track RFC, November 2000.

[30] G. Montenegro. RFC3024 - Reverse Tunneling for Mobile IP, revised. Standards Track RFC, January 2001.

[31] G. Dommety and K. Leung. RFC3025 - Mobile IP Vendor/Organization-Specific Extensions. Standards Track RFC, February 2001.

[32] J. Solomon and S. Glass. RFC 2290 - Mobile-IPv4 Configuration Option for PPP IPCP. Standards Track RFC, February 1998.

[33] KOM MPEG-1 Player for Linux, 2000. http://www.kom.e-technik.tu-darmstadt.de/kom-player.

[34] D. Forsberg, J. T. Malinen, J. K. Malinen, T. Weckstrˆm, and M. Tiusanen. Distributing Mobility Agents Hierarchically under Frequent Location Updates. In *Proceeding of the Mobile Multimedia Communications Workshop 1999 (MoMuC'99), San Diego, USA*. IETF, November 1999.

[35] R. Droms. RFC 2131 - Dynamic Host Configuration Protocol (DHCP), March 1997.

[36] M. G. Baker, X. Zhao, S. Cheshire, and J. Stone. Supporting Mobility in Mosquito Net. In *Proceeding fo the 1996 USENIX Technical Conference, San Diege, USA*, January 1996.

[37] S. E. Deering. RFC 1256 - ICMP Router Discovery Messages. Standars Track RFC, September 1991.