University of Zurich
Department of Informatics

Burkhard Stiller
Thomas Bocek
Peter Racz
Gregor Schaffrath
(Eds.)

Mobile Systems III

July 2008

TECHNICAL REPORT – No. ifi-2008.09

University of Zurich
Department of Informatics (IFI)
Binzmühlestrasse 14, CH-8050 Zürich, Switzerland

B. Stiller, T. Bocek, P. Racz, G. Schaffrath (Eds.): Mobile Systems III
Technical Report No. ifi-2008.09, July 2008
Communication Systems Group (CSG)
Department of Informatics (IFI)
University of Zurich
Binzmühlestrasse 14, CH-8050 Zurich, Switzerland
URL: http://www.csg.uzh.ch

Introduction

The Department of Informatics (IFI) of the University of Zurich, Switzerland works on research and teaching in the area of communications. One of the driving topics in applying communications technology is addressing investigations on mobility aspects and support for mobile users. Therefore, during the spring term FS 2008 a new instance of the Mobile Systems seminar has been prepared and students as well as supervisors worked on this topic.

Even today, the increasing number of mobile and wireless networks as well as their users or customers drive many developments of systems and protocols for mobile systems. The areas of underlying networking and development technology, of services assisting security or Quality-of-Service (QoS), and of mobility support determine an important part of future wireless networks. Therefore, this year's seminar addressed such areas in more depth. The understanding and clear identification of problems in technical and organizational terms have been prepared and challenges as well as weaknesses of existing approaches have been addressed in a mobile and wireless environment. All talks in this seminar provide a systematic approach to judge dedicated pieces of systems or proposals and their suitability.

Content

This third edition of the seminar entitled "Mobile Systems III" discusses a number of selected topics in the area of mobile communication. The first talk "Unterstützung der Mobilität in HIP" presents the Host Identity Protocol, discusses its mobility support, and compares it with mobile IPv4 and mobile IPv6. Talk two "RFID Karten und RFID Kartenleser für BioXes" provides an overview of the RFID technology and RFID-based access control and discusses biometric authentication. Talk three "Mobile Agents and Security" presents the concept of mobile agents, their security concerns, and discusses different approaches to make mobile agents secure.

Seminar Operation

Based on well-developed experiences of former seminars, held in different academic environments, all interested students worked on an initially offered set of papers and book chapters. Those relate to the topic titles as presented in the Table of Content below.

They prepared a written essay as a clearly focused presentation, an evaluation, and a summary of those topics. Each of these essays is included in this technical report as a separate section and allows for an overview on important areas of concern, sometimes business models in operation, and problems encountered.

In addition, every group of students prepared a slide presentation of approximately 45 minutes to present his findings and summaries to the audience of students attending the seminar and other interested students, research assistants, and professors. Following a general question and answer phase, a student-lead discussion debated open issues and critical statements with the audience.

Local IFI support for preparing talks, reports, and their preparation by students had been granted by Thomas Bocek, Peter Racz, Gregor Schaffrath, and Burkhard Stiller. In particular, many thanks are addressed to Peter Racz for his strong commitment on getting this technical report ready and quickly published. A larger number of pre-presentation discussions have provided valuable insights in the emerging and moving field of Mobile Systems, both for all groups of students and supervisors. Many thanks to all people contributing to the success of this event, which has happened in a lively group of highly motivated and technically qualified students and people.

Zürich, July 2008

Contents

Kapitel 1

Unterstützung der Mobilität in HIP

Svetlana Gerster

Die Internet Technologie entwickelt sich über die Grenzen des Festnetzes. Geräte, die mobil genutzt werden, möchten ihre Verbindungen während ihrer Bewegung aufrechterhalten. Die Mobilität führt zu neuer Herausforderung: um ein Paket an einen mobilen Host weiterzuleiten muss er zuerst gefunden werden. Dabei ist auch zu beachten, dass bei einem Standortwechsel auch die IP Adresse geändert wird. Diese Änderung sollte für die Applikationen transparent gehalten werden, sonst ist eine Kommunikation wegen der Gebundenheit an die IP Adresse nicht möglicht. Es wurden unterschiedliche Ansätze vorgeschlagen um diese Herausforderungen zu bewältigen. Im Folgenden wird der von IETF entwickelte Host Identity Protocol (HIP) näher betrachtet [1]. Um HIP mit anderen Technologien zu vergleichen und Unterschiede sichtbar zu machen wird zuerst kurz auf IPv4 und IPv6 aus der Sicht der Unterstützung von Mobilität eingegangen [6],[7].

Inhaltsverzeichnis

1.1 Problematik bei der Mobilität

Die Mobilität bringt mehrere neue Aufgaben mit sich. Wenn ein Host mobil ist, ändert er seine IP Adresse. Daraufhin ist der mobile Host nicht mehr erreichbar, da die Kommunikation an die IP Adresse gebunden ist. Um die Verbindung wieder aufzunehmen muss zuerst die neue Adresse den anderen Knoten bekannt gemacht werden. Dafür gib es unterschiedliche Vorschläge. Bei einem Vorschlag soll ein mobiler Host immer auch eine stationäre Adresse, unter der er ständig erreichbar ist, haben. Oder der mobile Host sollte andere Knoten selbst über seinen neuen Standort informieren. Eine weitere Möglichkeit besteht darin die Informationen in die Netzinfrastruktur abzulegen. Die Verwaltung von Standortinformationen ist mit heutiger Infrastruktur nicht zu bewältigen. Es bedarf weiteren Ausbau beziehungsweise Ereiterungen in der Netzinfrastruktur. Eine weitere Aufgabe ist die Session Pflege. Der Übergang zu neuer Adresse bricht die Verbindungen mit anderen Hosts ab, da die IP Adresse auch als Host Identifier gilt. Mobile Hosts müssen imstande sein sich gegenüber ihren Kommunikationspartner zu authentifizieren und die Verbindung zu pflegen oder wieder aufzunehmen.

1.2 Mobile IP

Die beiden IP Protokolle, IPv4 und IPv6, wurden erweitert um die Mobilität zu unterstützen. Das Ziel ist die Veränderungen der IP Adresse für Schichten oberhalb der Vermittlungsschicht transparent zu machen.

1.2.1 Mobile IPv4

In IPv4 wird angenommen, das ein Knoten durch die IP Adresse eindeutig identifiziert ist. Ein Knoten ohne Netzwerkanschluss kann die Datenpakete nicht empfangen. Ändert ein Knoten seine Position, wird üblicherweise seine IP Adresse auch verändert. Dadurch wird es für einen Knoten unmöglich die Verbindungen der Transport - und höherer Schichten aufrecht zu erhalten. Mobile IPv4 hat einen neuen Mechanismus entwickelt, um Geräten den Wechsel von einem Rechnernetz in ein anderes zu ermöglichen und dabei eine feste IP Adresse zu behalten. Für diesen Zweck enthält Mobile IPv4 Architektur neue Entitäten [8]: **Mobile Rechner (mobile Host)** - ist ein Rechner, welcher seine Anschlussstelle wechselt. Jeder mobile Rechner besitzt einen `Heimatnetz` und eine `primäre Adresse` auch `Home Address` genannt. **Home Agent** ist ein Router im Heimnetzwerk. Wenn der mobile Rechner sein Heimatnetz verlässt, leitet dieser Home Agent ankommende Datenpakete an den aktuellen Foreign Agent weiter. **Foreign Agent** ist ein Router in einem fremden Netz, in welchem sich der Rechner gerade befindet. Er übernimmt die Leitung der Pakete an den mobilen Host. Abbildung 1.1 zeigt die Nachrichtenübermittlung. Auffallend ist die mögliche Überlastung von Home und Foreign Agenten, falls sie viele Knoten bedienen oder falls der Verkehr sehr gross ist.

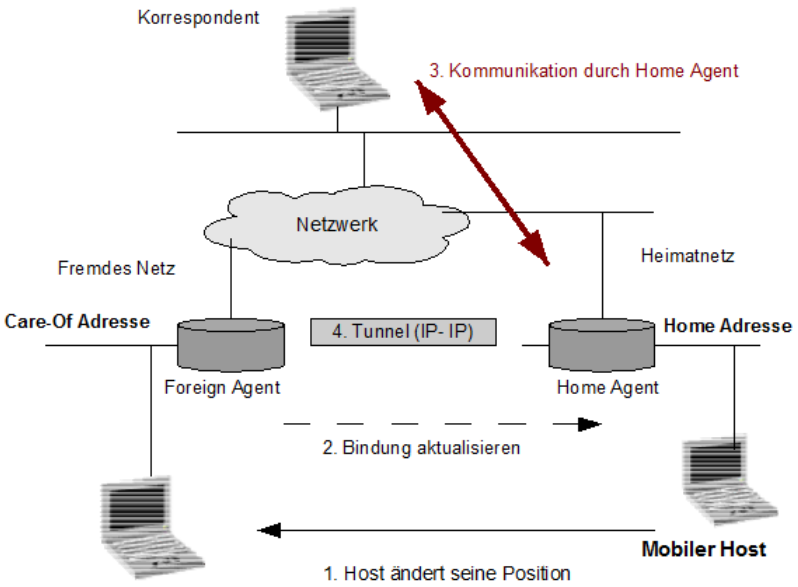

Abbildung 1.1: Mobile IPv4 Nachrichtenaustausch

Die Agenten signalisieren ihre Präsenz mittels Agent Advertisement - Nachricht. Diese Nachricht ist eine Erweiterung der gewöhnlicher ICMP Router Advertisement - Nachricht. Üblicherweise wird eine solche Nachricht jede Paar Sekunden verschickt. Mit dieser Nachricht kündigt der Router seine Bereitschaft die Rolle der Agenten zu übernehmen. Der mobile Rechner nutzt die Daten aus dieser Nachricht um zu prüfen wo er gerade angeschlossen ist und wie die Adresse der Agenten lautet. Der Host kann sich immer noch in seinem Heimatnetz befinden, oder er kann sich von seinem Heimatnetz in ein fremdes Netz verschoben haben, oder er kann sich auch von einem fremden Netz in ein anderes fremdes Netz bewegt haben. In letzten beiden Fällen registriert sich der mobile Host beim fremden Router und bekommt eine `sekundäre Adresse` zugewiesen. Diese Adresse entspricht der Adresse von Foreign Agent und wird als `Care-Of Adresse` des mobilen Hosts bezeichnet. D.h. mobiler Host bekommt eigentlich keine eigenständige Adresse. Nachdem der Host seine Care-Of Adresse hat registriert er sich bei seinem Home Agenten und teilt ihm damit seine aktuelle Position mit. Die Registrierung erfolgt mittels REGISTRATION REQUEST und REGISTRATION REPLAY Nachrichten und wird nicht direkt sondern durch den Foreign Agenten verschickt. REGISTRATION REQUEST Nachricht enthält die neue Care-Of Adresse und ihre Laufzeit. Die Registrierung erzeugt oder aktualisiert die gespeicherte Bindungen. Diese Bindungen assoziieren primäre Adresse - Home Address mit momentaner Care-Of Adresse. Für die berechtigte Durchführung der Registrierung müssen mobiler Rechner und Home Agent sich authentifizieren sonst kann ein Angreifer falsche Registrierungen durchführen. Deshalb muss jede Registrierung Nachricht eine Authentifizierung haben damit die Registrierung autorisiert wird. Sonst werden die Nachrichten verworfen. Ist die Registrierung erfolgreich, können die Pakete an den Host weiter geleitet werden. Die Kommunikation zwischen mobilem Host und seinem Korrespondenten erfolgt immer über den Home Agenten. Der Korrespondent hat keine Kenntnisse über den aktuellen Standort von mobilem Host. Alle Pakete werden an die primäre (statische) Adresse geschickt. Ist der Host unter dieser Adresse momentan nicht erreichbar, werden an ihn geschickte Pakete von Home Agenten abgefangen in einen zusätzlichen IP Paket

verpackt und an den Foreign Agenten d.h. an die Care-Of Adresse weitergeleitet, sog. Tunneling. Foreign Agent entpackt die Nachricht und überträgt sie an den Host. Der Host kann eine Antwort auf üblichem Weg, ohne seinen Agenten, abschicken. Wichtig ist, dass die Sender Adresse immer die Home Address ist und der Korrespondent immer seine Nachrichten an die Home Address schickt. So bleibt die Mobilität für Transportschicht völlig unbemerkt.

1.2.2 Mobile IPv6

IPv6 geht wie IPv4 davon aus, dass jeder mobiler Host immer durch seine primäre Adresse - Home Address identifiziert ist, unabhängig davon wo er sich gerade befindet. Die aktuelle Position ist durch Care-Of Adresse gegeben. Das Protokoll ermöglicht die Speicherung von Bindungen zwischen primären und Care-Of Adresse und die Weiterleitung der Pakete durch Home Agent an die aktuelle Care-Of Adresse. Der Ablauf ist dem aus IPv4 ähnlich mit der Ausnahme, dass IPv6 keine Foreign Agent benötigt. Der mobile Rechner kann seine eigene Care-Of Adresse mittels konventionellen IPv6 Mechanismus, zum Beispiel Auto - Konfiguration, anfordern. Ein weiterer Unterschied besteht darin auf welche Art und Weise die Nachrichten vom mobilen Host verschickt werden. In IPv6 ist Tunneling entgegengesetzt. Das bedeutet nichts anderes als die mobile Host Pakete auch über Home Agenten gehen. Unterstützt der Korrespondent IPv4 oder IPv6 Protokoll kann die Kommunikation mit mobilem IPv6 Host mittels der zu IPv4 ähnlicher Umleitung, d.h. durch Home Agenten geschehen. Für die Kommunikation zwischen zwei IPv6 Rechner bietet der Protokoll eine zweite Möglichkeit, sog. `Route Optimization` [9]. Route Optimization unterstützt eine direkte Verbindung, ohne dass der Home Agent beansprucht wird. Dafür notwendige Mechanismen werden hier näher vorgestellt. Bei Route Optimization Modus kann mobiler Rechner mit einem Korrespondenten direkt kommunizieren. Um es zu ermöglichen sollten die Korrespondenten die lokale Speicherung von Bindungen, wie Home Agent, unterstützen. Denn die mobile Hosts müssen ihre aktuelle Bindung bei den Korrespondenten registrieren. Korrespondent schlägt in seinen Einträgen die aktuelle Care-Off Adresse nach und schickt die Pakete direkt dorthin. Die Care-Of Adresse ist die Empfänger Adresse. Sendet der mobile Host seine Pakete an den Korrespondenten so ist Care-Of Adresse die Sender Adresse. Jeder Paket beinhaltet aber auch die primäre Adresse damit die Nutzung der Care-Of Adresse für Transportschicht transparent bleibt. Die Abbildung 1.2 zeigt den Route Optimization Modus.

Ändert der Knoten eine Adresse, so soll er die Bindungen sowohl bei seinem Home Agent als auch bei seinem Korrespondenten aktualisieren. Zwischen Home Agenten und mobilem Host muss IPsec Sicherheitsassoziation benutzt werden. Dadurch wird die Integrität und Authentifizierung geschützt. Zwischen mobilem Host und dem Korrespondenten verlangt keine solche Assoziation. Stattdessen wird eine Methode namens `Return Routability Procedure` verwendet. Dabei werden vier Nachrichten, die durch verschlüsselten Hash Algorithmus geschützt sind ausgetauscht. Abbildung 1.3 stellt die Methode dar. Mobiler Rechner sendet gleichzeitig zwei Nachrichten an den Korrespondenten ab. Home Test Init wird durch den Home Agenten geschickt. Die Sender Adresse entspricht der primären Adresse, Empfänger Adresse ist die Adresse der Korrespondenten. Außerdem beinhalten

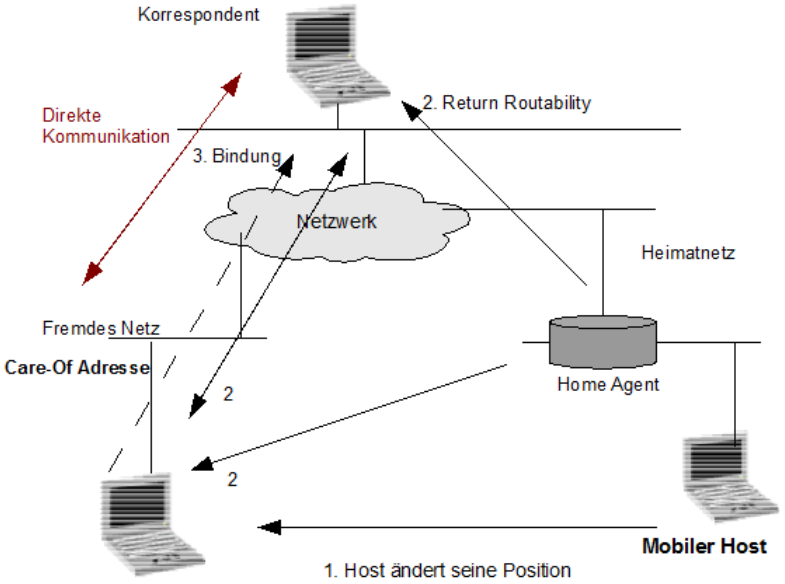

Abbildung 1.2: Mobile IPv6: Nachrichtenaustausch in Route Optimization Modus.

die Nachricht einen Cookie. Care-Of Test Init wird direkt verschickt. Die Sender Adresse ist die Care-Of Adresse. Auch diese Nachricht enthält einen Cookie.

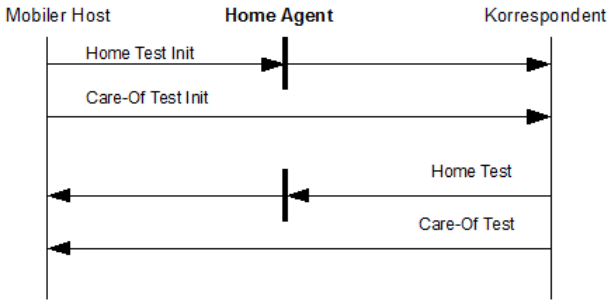

Abbildung 1.3: Return Routability Procedure

Die entsprechende Antworten der Korrespondenten werden auf gleichem Weg zurückgeschickt. Die beiden Cookies werden auch mit zurückgeschickt. Zusätzlich beinhaltet jede Nachricht den, aus dem Inhalt der enthaltenen Nachricht, verschlüsselt berechneten Token. Erst jetzt kann der mobile Host die Aktualisierung seiner Bindung starten. Die Aktualisierung erfolgt mittels BINDING UPDATE Nachricht [9]. In der BINDING UPDATE Nachricht schickt der mobile Host auch Informationen mit die er nur aus den vier Return Routability Procedure Nachrichten berechnen kann. So wird auch sichergestellt, dass der Host tatsächlich unter der neuen Adresse erreichbar ist und so eine Aktualisierung zu veranlassen auch berechtigt ist. Wäre der Host nicht unter der neue Adresse erreichbar hätte er wenigstens die Care-Of Test Nachricht nicht erhalten. Hätte so ein Rechner versucht die Bindung trotzdem zu aktualisieren, wäre der Wert in der BINDING UPDATE , der sich aus allen vier Nachrichten berechnet falsch gewesen. In einem solchen Fall findet keine Aktualisierung statt.

1.3 Host Identity Protocol

Host Identity Protokoll wurde von IETF entwickelt. Dabei handelt es sich um ein kryptographisch basierten Protokoll, welcher auch eine bessere Unterstützung für mobile Hosts bieten soll. Im Folgenden werden die Elemente der HIP Architektur näher vorgestellt. Zuerst wird betrachtet wie ein HIP Kommunikation Kontext zwischen zwei Hosts aufgebaut wird. Danach folgt die Diskussion über die Unterstützung der Mobilität sowie die Auswirkungen von HIP auf die Applikationen. Die Fragen wie Sicherheit, DNS Erweiterung, Vor- und Nachteile, HIP im Vergleich sollen auch nicht unbehandelt bleiben.

1.3.0.1 Referenzmodell und Rolle der IP Adresse

Um mit Host Identity Protocol verbundene Veränderungen im Protokollstapel sichtbar zu machen wird hier ein modifiziertes OSI-Referenzmodell nach A.S. Tanenbaum verwenden [5]. Wie die Abbildung 1.4 zeigt, besteht das Modell aus fünf Schichten.

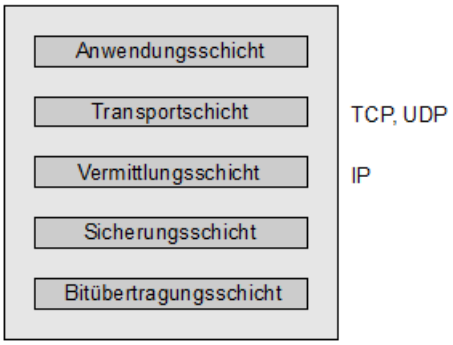

Abbildung 1.4: Modifiziertes OSI Schichtenmodell

Möchte ein Prozess in der Anwendungsschicht mit einem entferntem Prozess kommunizieren, muss er den gewünschten Partner angeben. Der Prozess verbinden sich mit seinem lokalen Port, um eine Verbindung mit einem entfernten Port aufzubauen. Diese Verbindung läuft über IP Adressen der beiden Hosts . In der Abbildung 1.5 wird die Beziehung zwischen Port und IP Adresse dargestellt [5].

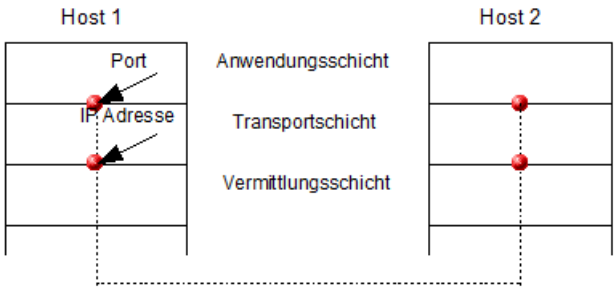

Abbildung 1.5: Port zu Port Kommunikation.

Die IP Adresse ist eindeutig und verbindet in sich zwei unterschiedliche Rollen: Host zu identifizieren und sein Standort zu bestimmen.

1.3.1 HIP Architektur

Host Identity Protocol schlägt die Einführung einen neuen Namensraums sowie einer neuen Schicht zwischen Vermittlungs- und Transportschicht vor [1]. Damit wird der Transportschicht von der IP Adresse, die nur zu Lokalisierung dienen soll, entkoppelt. Die Identifizierung der Hosts übernimmt der neue Namensraum, der als **Host Identity Namespace** bezeichnet wird. Dieser Namensraum besteht aus einer Menge von sogenannten **Host Identifiers (HI)**. Ein Host Identifier ist der Name, welcher die **Host Identität** eng. **Host Identity** im Host Identity Namespace repräsentiert. Da ein Host Identifier eindeutig sein muss, wurde die Verwendung der öffentlicher Schlüssel, wie bei asymmetrischen Verschlüsselung[1], als HI empfohlen [1]. Jeder Host sollte wenigstens eine Host Identität haben. Die Identität und entsprechender Host Identifier können in Domain Name System (DNS) veröffentlicht werden. Da die Nutzdaten bei der Kommunikation mit HIP aktuell durch IPsec geschützt werden, kann der Host Identifier zu Authentifizierung eingesetzt werden. Als Konsequenz des neuen Namensraums enthält das Schichtenmodell eine neue Schicht, **HIP Schicht** genannt, Abbildung 1.6. HIP Schicht transformiert Identifier der Transportschicht in dazugehörige IP Adresse der Vermittlungsschicht. Auf diese Art und Weise wird die Verbindung zwischen zwei Rechner auf der Transportschicht nicht unterbrochen, auch wenn die Adresse sich in der Zwischenzeit geändert hat.

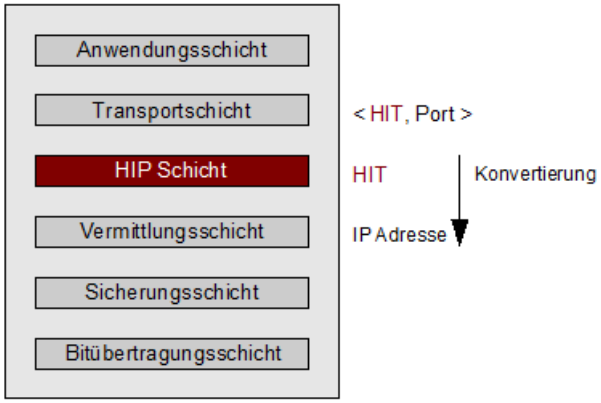

Abbildung 1.6: HIP Schicht befindet sich zwischen Transport- und Vermittlunsschicht.

1.3.1.1 Host Identity Namespace und Host Identifier (HI)

Wie oben erwähnt, sollte jeder Host eine Identität haben um angesprochen werden zu können. Die Host Identität ist nur durch öffentlicher - privater Schlüssel Paar vollständig definiert. Verweis auf die Identität wird mittels öffentlichen Schlüssels d.h. HI gemacht. HIs bilden den neuen Namensraum. Es wird vorgeschlagen die Host Identifiers im DNS Verzeichnis abzulegen und bei Bedarf anzufragen. In den Internetprotokollen soll **Host**

[1]Eine asymmetrische Verschlüsselung ist dadurch charakterisiert, dass jede Partei einen öffentlichen und einen privaten Schlüssel hat. Der öffentliche Schlüssel ist bekannt. Der private Schlüssel ist von dem öffentlichen verschieden und nur derjenigen Partei bekannt, welche als Halterin von entsprechendem öffentlichem Schlüssel identifiziert ist. Privater Schlüssel wird für die Authentisierung der Identität verwendet.

Identity Tag (HIT) verwendet werden. Host Identity Tag wird mittels SHA-1 Hash-Funktion über HI berechnet und hat eine konstante Länge von `128-Bit`. In einem IP Paket, der eine HIP Nachricht überträgt - HIP Paket, wird der Sender und der Empfänger durch HIT identifiziert. Eine weitere Möglichkeit die Host Identität zu präsentieren ist die Verwendung von lokalem Identifier sog. **Local Scope Identifier (LSI)**. LSI ist `32-Bit` lang und wurde entwickelt um den HIP auch bei den Hosts anzuwenden die noch IPv4 einsetzen.

1.3.1.2 HIP Kommunikation

Eine HIP Nachricht wird im IP Paket übertragen. Dadurch verändert sich die übliche Paketstruktur, Abbildung 1.7.

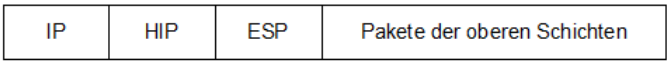

Abbildung 1.7: Veränderungen in Paketstruktur

Dieses Kapitel stellt die Nachrichtentypen des Protokolls vor und zeigt wie eine Nachricht aufgebaut ist.

HIP Pakete Jeder HIP Paket besitzt einen `Typ` und einen `Namen`. Zur Zeit gibt es `acht` grundlegende HIP Pakettypen. Vier davon werden eingesetzt um einen Kommunikationskontext zwischen zwei Rechner aufzubauen, siehe Base Exchange. Geschlossen wird dieser Kontext durch den Austausch von CLOSE und CLOSE-ACK Nachrichten. UPDATE Nachricht findet bei der Aktualisierungen ihre Verwendung. NOTIFY ist u.a. für eintretende Fehler vorgesehen. Die acht Nachrichten sind in der Abbildung 1.8 dargestellt.

Typ	Name	Verwendung
1	**I1**	Initiator Paket bei Base Exchange
2	**R1**	Responder Paket bei Base Exchange
3	**I2**	Zweites Initiator Paket
4	**R2**	Zweites Responder Paket
5	**UPDATE**	Adressen Aktualisierung
6	**NOTIFY**	v.a. Fehler Meldung
7	**CLOSE**	Beendet eine HIP Assoziation
8	**CLOSE_ACK**	Bestätigung für Ende der Assoziation

Abbildung 1.8: 8 wichtige HIP Pakettypen

Alle HIP Nachrichten beginnen mit einem `HIP Header` gefolgt von null bis mehreren `HIP Parameter`, Abbildung 1.9. Der Header enthält Feld für nächsten Header, welcher das Protokoll der im Paket übertragenen Daten identifiziert, ein Feld für die Angabe von welchem Typ der Paket ist, sowie Felder für Sender und Empfänger HIT.

Die Parameter bestehen aus `Namen`, `Typ`, `Länge` und `Wert`. Sie werden vor allem für Authentifizierung und Adressenänderung bzw. Prüfung benötigt. Beispielhaft wird der Parameter namens LOCATOR im Kapitel näher betrachten Vollständige Parameterliste kann unten IETF RFC 5201 nachgeschlagen werden [2].

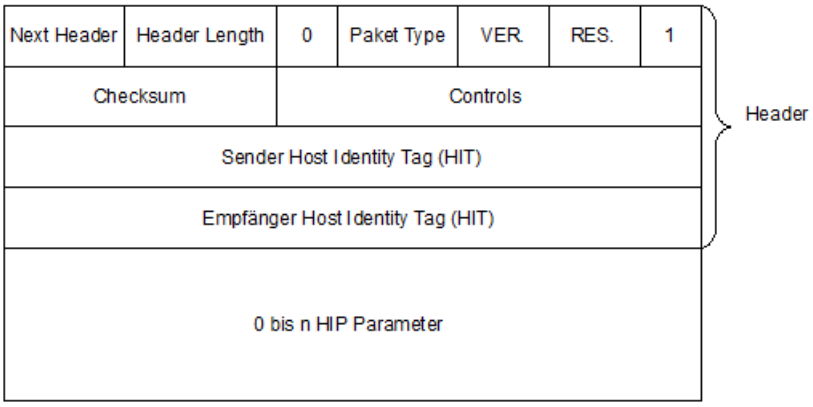

Abbildung 1.9: HIP Paket

1.3.1.3 Base Exchange

HIP wurde als ein Authentifizierung Protokoll entwickelt, der mit Encapsulated Security Payload[2] (ESP) genutzt werden soll. **HIP Base Exchange** ist ein Protokoll, welcher für Nachrichtenaustausch zwischen zwei Hosts bei der Aufbau von HIP Assoziation, verwendet wird. Eine `HIP Assoziation` ist ein Kommunikationskontext der IP Schicht und wird aufgebaut bevor die Übertragung von Nutzdaten statt finden kann.

HIP Assoziation Base Exchange schreibt den Austausch von insgesamt vier Nachrichten zwischen Sender und Empfänger vor. Dadurch wird auch versucht die Denial-of-Service Angriffe zu unterbinden. Der Sender wird entsprechend der IETF Empfehlung nachfolgend als `Initiator` genannt, der Empfänger als `Responder` [2]. Abbildung 1.10 verdeutlicht den Vorgang. Als erstes schickt der Initiator eine Nachricht **I1**, die nur den Initiator HIT und möglicherweise auch den Responder HIT enthält und als Auslöser dient. Die nachfolgende drei Nachrichten implementieren den Diffie-Hellman Schlüsselaustausch Protokoll. Mit ihm erzeugen beide Parteien einen nur ihnen bekannten geheimen Schlüssel, der verwendet wird um verschlüsselte Nachrichten zu übertragen. Die Antwortnachricht **R1** von Responder starten den Austausch. R1 enthält Responder Signatur, sein HIT, öffentlichen Diffie-Hellman Schlüssel und ein Rätsel als Parameter. Der Initiator nutzt den enthaltener Diffie-Hellman Schlüssel um den Session Schlüssel abzuleiten mit dem er die HIP Assoziation bildet. Darauffolgende **I2** Nachricht des Initiators muss in ihrer Parameterliste die Lösung des Rätsels haben, sonst wird sie nicht akzeptiert. Ausserdem enthält I2 die Signatur des Initiators, sein HIT und ein weiterer öffentlicher Diffie-Hellman Schlüssel. Der Responder verwenden seinerseits den empfangenen Diffie-Hellman Schlüssel für die Berechnung von Session Schlüssel und erzeugt damit dazugehörige HIP Assoziation. **R2** beendet den Austausch. Zu beachten ist die Tatsache, dass nur die Signaturen in I2 und R2 zu Authentifizierung verwendet werden, da die Signatur in der R1 Nachricht nicht über den vollen Datenpaket gebildet wird. Das Ergebnis von Base Exchange ist eine sichere HIP Assoziation.

[2]ESP ist eine der IPsec Protokolle und stellt die Integrität der Daten sicher. Die Nutzdaten werden verschlüsselt übertragen.

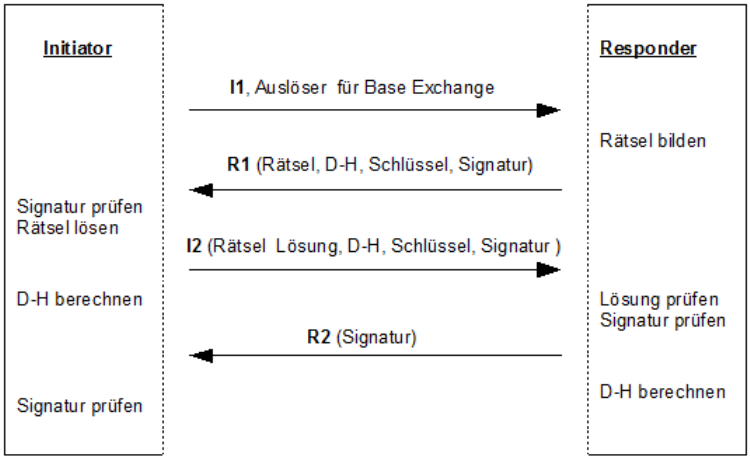

Abbildung 1.10: HIP Base Exchange

Base Exchange soll die beide Parteien gegen die Replay Angriffe schützen. Initiator ist gegen die R1 Wiederholungen mittels R1 Zähler, der im R1 als Parameter enthalten ist, geschützt. Responder ist gegen Wiederholungen von falschen I2 Nachrichten durch den Rätsel geschützt. Ausserdem schützen sich die Hosts gegen R2 Wiederholungen durch die Prüfung der Signatur. Erst nach einem erfolgreichen Base Exchange können die Hosts weitere Nachrichten austauschen. Zum Beispiel UPDATE Nachricht, die über neue IP Adresse informiert, wie es im nächsten Kapitel gezeigt wird.

HIP Assoziation in opportunistischem Modus Es ist auch möglich eine HIP Assoziation zu erstellen wenn die Hostidentität respektive HIT der Korrespondenten nicht bekannt ist. In diesem Fall enthält die I1 Nachricht ein null Wert als Empfänger HIT. Auf diese Weise aufgebaute Verbindung wird opportunistischer Modus genannt. Der Korrespondenten HIT wird mit R1 Paket an den Initiator verschickt. Nach der Prüfung vom R2 Paket ist der HIT auch authentisiert. Allerdings bringt der opportunistischer Modus zusätzliche Herausforderungen für API mit sich. Der Initiator muss Base Exchange, auf IP Adressen basierend, starten können.

1.3.2 HIP Mobilität und Multihoming

HIP Protokoll entkoppelt den Transportschicht von der IP Adresse was eine neue Lösung für die Mobilität und Multihoming ermöglicht. Der Begriff Multihoming bedeutet den Besitzt von mehreren IP Adressen für eine Hostidentität. Dieses Kapitel beschreibt die im HIP vorgeschlagene Erweiterungen für die Unterstützung von mobilen Rechner. Im Mittelpunkt steht der durch die Mobilität ausgelöste Prozeduren. Wird die IP Adresse verändert wird zuerst die UPDATE Nachricht versenden. Hier spielt der LOCATOR Parameter eine wichtige Rolle da er den Korrespondenten über die neue IP Adresse in Kenntnis setzt. Es wird empfohlen den LOCATOR zu schicken sobald der mobile Host eine Veränderung in seiner IP Adresse erkannt hat. So wird die bestehende Assoziation aufrechterhalten. Schlussendlich aber hängt es von der zugrunde liegenden Bestimmungen ab ob es tatsächlich gemacht wird.

1.3.2.1 Adressen Aktualisierung

Wenn ein Host sich zu einer anderen IP Adresse bewegt teilt er den Korrespondenten seine neue Adresse mit. Dazu übermittelt der Host eine UPDATE Nachricht, welche den LOCATOR Parameter enthält. Der Empfänger pflegt eine Liste, die die Bindungen zwischen Adressen und Hosts enthält. Hier wird angenommen, dass die HIP Assoziation bereits existiert, sonst muss zuerst Base Exchange ausgeführt werden um Sicherheit zu gewährleisten. Es sei auch daran erinnert, dass in HIP zur Zeit ESP für Nutzdatenschutz benutzt wird. ESP SPI ist der Kontextindex.

Abbildung 1.11 stellt den Prozess dar. Der mobile Host schickt eine UPDATE Nachricht, welche den LOCATOR Parameter enthält. Der LOCATOR Parameter beinhaltet die neu IP Adresse und ihre Laufzeit. Fehlt die Bestätigung seitens der Korrespondenten wird die Nachricht wiederholt übertragen. Kommt die Nachricht an, authentifiziert der Korrespondent den UPDATE Paket Anhang der Signatur und dem verschlüsselt berechnetem Hashwert des Pakets. Daraufhin aktualisiert der Empfänger die lokal gespeicherte Bindung zwischen der HIP Assoziation und der Adresse von mobilem Host. Um die Angriffe zu vermeiden muss der Korrespondent die Adressenprüfung unterstützen. Dazu wird ein schwer zu erratender Wert berechnet und als ECHO-REQUEST Parameter in der UPDATE Nachricht an den mobilen Host zurückgeschickt. Der Korrespondent kann die neue Adresse sofort nutzen, die Anzahl der Nachrichten ist aber limitier. Erst nachdem der mobile Host die Umadressierung durch einen weiteren UPDATE Paket bestätigt, können die Daten im vollem Umfang übertragen werden. Dieser dritte UPDATE Paket schickt den Wert aus ECHO-REQUEST als ECHO-RESPONSE an den Korrespondenten zurück. So wird sichergestellt, dass der mobile Host tatsächlich unter der angegebenen Adresse erreichbar ist. Erst jetzt wird die alte Adresse endgültig entfernt und die neue gespeichert.

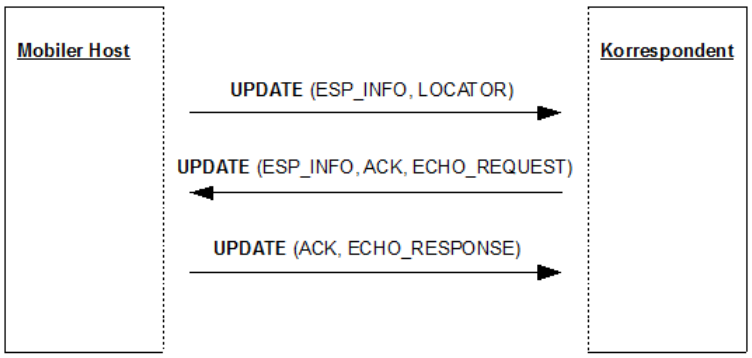

Abbildung 1.11: Austausch bei Aktualisierung der Adresse(n).

Ein mobiler oder stationärer Host kann auch mehr als nur eine Adresse besitzen. Der Host kann auch die UPDATE Nachricht dazu benutzen um den Korrespondenten über die zusätzliche Adressen zu informieren. Bei mehreren Adressen kann eine als bevorzugt gezeichnet werden. Um Adressen anzugeben wird LOCATOR Parameter verwendet. Es besteht auch die Möglichkeit den Parameter schon bei Base Exchange in der I2 Nachricht zu übermitteln. Allerdings kann nur die aktuelle Senderadresse als bevorzugt gesetzt werden. Alle neue Adressen werden der Prüfung mit ECHO unterzogen. Der Korrespondent hat auch die Möglichkeit den LOCATOR bei Base Exchange zu übertragen. Dafür wird sein R1 Paket benutzt.

1.3.2.2 LOCATOR Parameter

Ein Parameter besteht wie schon erwähnt aus Namen, Typ, Länge und Wert. Der Parameter Namens LOCATOR kann zusätzlich dazu mehrere Subparameter haben. Die Subparameter bestehen ihrerseits aus Datenverkehrtyp, Locator Typ, Länge, Laufzeit und Adressenwert. Abbildung 1.12 zeigt die mögliche Aufbau.

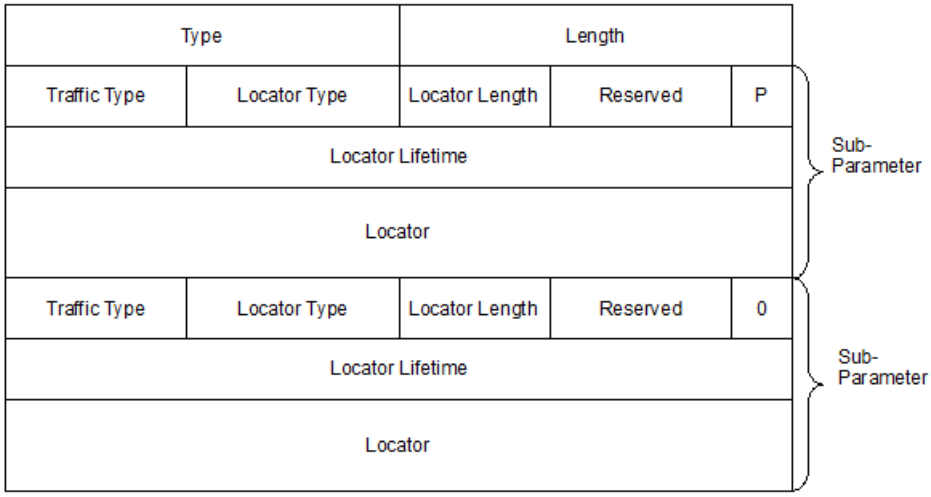

Abbildung 1.12: HIP LOCATOR Parameter Aufbau

Der Wert für Typ Feld ist festgelegt. Die Länge wird über dem LOCATOR aber ohne Typ und Länge Felder berechnet.

- `Datenverkehrtyp` ermöglicht es festzulegen ob die Adresse nur für HIP Kontrollpakete - Typ 1, nur für Benutzerdaten - Typ 2 oder für beides - Typ 0 verwendet wird.
- `Locator Typ` definiert die Semantik vom Locator Feld. Ist es eine IPv6 oder eine auf IPv6 abgebildete IPv4 Adresse mit 128 Bit länge, wird der Feld auf null gesetzt. Handelt es sich um eine Aneinanderkettung von ESP aus 32 Bit gefolgt von der IPv6 oder abgebildeten IPv4 128 Bit Adresse, wird der Feld auf eins gesetzt.
- `P` Feld gibt an ob die Adresse für diesen Datenverkehrtyp bevorzugt ist - Wert 1, oder nicht - Wert 0.
- `Laufzeit`, d.h. wie lange ist der Host unter dieser Adresse erreichbar, wird in Anzahl Sekunden angegeben. Wird die Zeit überschritten muss der UPDATE erneuert werden.
- Und schlussendlicht wird die Adresse im `Locator` Feld angegeben.

Ein Host soll im Stande sein den LOCATOR Parameter zu empfangen. Nachdem die UPDATE Nachricht authentifiziert ist, werden die Adressen im LOCATOR Parameter verarbeitet. Bislang wurde nur die Verarbeitung für folgende Kombination untersucht:

Datenverkehrtyp `0` und Locator Typ `1` [3]. Für jede Adresse aus dem LOCATOR Parameter prüft der Host ob die Adresse schon an eine Assoziation gebunden ist. Ist es der Fall, wird die Laufzeit aktualisiert sonst wird die Adresse in die Liste übernommen und zuerst als nicht geprüft markiert. Erst nach der Prüfung werden sie als aktive bezeichnet. Alte Adressen die in der Liste aber nicht mehr im LOCATOR sind werden als abgelehnt markiert. Wird eine der neuen Adressen als bevorzugt gesetzt, muss sie zuerst überprüft werden bevor es entsprechend markiert wird.

1.3.2.3 Rendezvous Mechanismus

Der HIP Protokoll erlaub einem mobilen Host seinen Korrespondenten über die neue Adresse mit UPDATE Nachricht zu informieren. Es ist aber nur möglich falls die Hosts wissen wie sie einander erreichen können. Ein mobiler HIP Host möchte vielleicht auch für andere Korrespondenten, die keine Informationen darüber haben wo der Host sich gerade befindet, erreichbar sein. HIP Architektur besitzt dafür einen **Rendezvous Server (RVS)**. Ein HIP Host kann sich beim RVS mit seinem Host Identity Tag und seiner aktueller Adresse registrieren. Möchte ein HIP Host mit einem anderen kommunizieren, kann er den Base Exchange starten und die erste d.h. I1 Nachricht an die IP Adresse des Servers anstatt des Hosts schicken, Abbildung 1.13. Damit der Initiator wiederum die IP Adresse des Servers bekommt, benutzt er zuerst die Dienste von DNS. DNS wird im nächsten Unterkapitel behandelt. Der Server schlägt Anhang vom HIT die richtige Adresse nach und leiten den Paket weiter. Dafür überschreibt der RVS seine IP Adresse mit der IP Adresse vom Responder und führt die Berechnung der Prüfsumme erneut durch. Der Responder kann die Base Exchange ohne die Serverdienste weiter führen indem er die R1 Paket direkt an den Initiator zurück schickt. Die Adresse des Initiators entnimmt er aus dem I1 Paket.

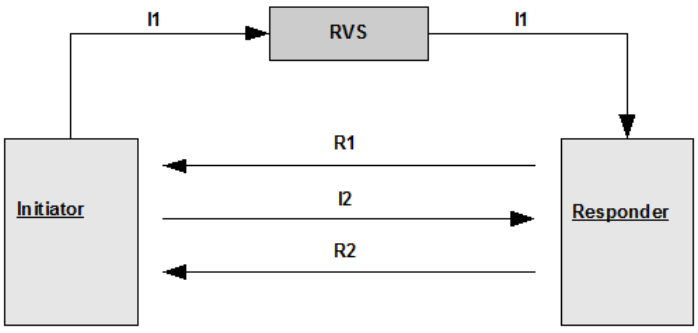

Abbildung 1.13: Base Exchange mit RVS

IETF Spezifikation nimmt an, dass die Klienten vom Server alle Responder Rolle Haben [7]. Andere Möglichkeiten wurden noch nicht behandelt. Der Klient muss sich beim Server registrieren. Für die Registrierung werden die vier Base Exchange Pakete verwendet, allerdings mit Erweiterungen um den Registrierungsprozess zu unterstützen [RFC 5203]. Zu Untersuchen ist auch in wie weit der Rendezvous Server dafür genutzt werden kann um die Position von mobilem Host zu verbergen, so dass die Korrespondenten nicht erfahren werden wo der Host sich befindet.Der ganze Verkehr wird dann über den RVS gehen. In solchem Fall mutiert der Server zu Home Agent , wie ein in IPv4. Ein auf HIP und

Rendezvous Server basierender Vorschlag untersucht eine neue Architektur, welche die sog. Location Privacy bittet [10]. Die HIT zu IP Auflösung übernimmt eine neue Entität, Rendezvous Agent (RVA) genannt. Rendezvous Agent ist nichts anderes als ein verbesserter RVS. Den zentralen Bestandteil neuer Architektur bilden die durch RVA geschützte Bereiche. Die Hosts im RVA Bereich können stationär oder mobil sein. Die Bereiche sind mit dem üblichem Netz verbunden. RVS und DNS liegen ausserhalb dieser Bereiche. RVA übernimmt die HIT-IP, oder IP-IP Überführung. Bekommt der RVA eine Paket von ausserhalb, adressiert er diesen Paket zu HIT oder lokaler IP Adresse um und leitet es an den Host weiter. Die ausgehende Pakete werden mit einer globalen IPv6 Adresse versehen. Die globalen IP Adressen limitieren die Informationen über die Hostposition.

1.3.2.4 DNS Erweiterung

Möchte ein Host eine Kommunikation mit einem anderem Host initiieren, wird als erstes Base Exchange durchgeführt und dabei HIP Assoziation aufgebaut. Bisher wurde impliziert, dass die Hostidentität des Empfängers dem Sender bekannt ist. Es musste möglich sein den Nachrichtenaustausch auch ohne diese Kenntnisse zu starten. Es kann durch ein zusätzliches Service erreicht werden, der auch einen Lookup durchführt. IETF [6] schlägt die Erweiterung von DNS und nicht die Ausbau von einem neuen Dienst vor, Abbildung 1.14.

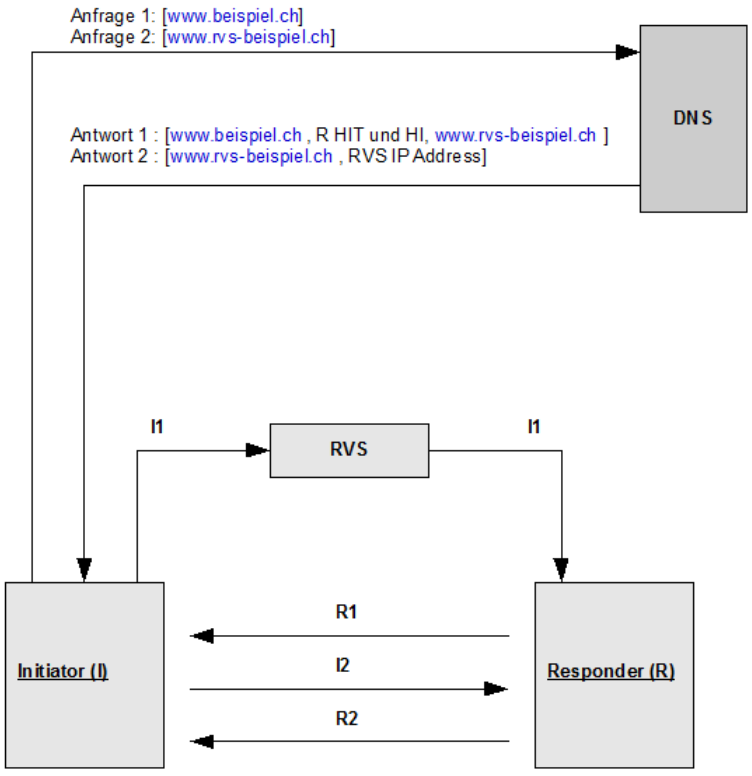

Abbildung 1.14: Base Exchange mit RVS und DNS.

Möchte ein mobiler Host mittels Namen erreichbar sein, sollen folgende Informationen im DNS Eintrag abgelegt werden:

- Host Identity (HI)
- Host Identity Tag (HIT)
- IP Adresse
- Alle Rendezvous Servers (RVS) an welchen der Host registriert ist.

Wenn ein HIP Host seine Adresse häufig ändert, kann es dazu kommen, dass wegen zu grossen Wartezeiten sein DNS Eintrag nicht aktualisiert wird. Aus diesem Grund wurde der oben kurz vorgestellter Rendezvous Server vorgeschlagen. Der RVS soll für ständige Erreichbarkeit sorgen und die Adressen rechzeitig aktualisieren. Initiator schickt zuerst eine Anfrage mit dem Namen von Responder an DNS. Der zurückgeschickte DNS Paket beinhaltet Responder HI, HIT und RVS Domänenname(n). Danach übermittelt der Initiator eine weitere Anfrage an DNS. Diesmal um mit dem enthaltenem RVS Domänenname(n) die IP Adresse(n) von RVS abzufragen. Erst nachdem der Initiator die RVS IP Adresse hat sendet er an diese Adresse I1 Paket ab. Der wird dann an den Responder weitergeleitet. Die drei folgende Base Exchange Nachrichten werden direkt übertragen.

1.3.2.5 Sicherheit

Der HIP Mobilität Mechanismus bietet Sicherheitsmechanismen für die Aktualisierung der Adressen an. Der Empfänger führt zuerst eine kryptographische Prüfung des UPDATE Senders durch. Auf diese Weise wird es schwierig die Nachricht zu fälschen oder für Replay Angriffe zu nutzen. Die Sicherheitsrisiken liegen in anderen Bereichen v.a. `Men-in-The-Middle` und `Denial-of-Service Angriffe`. Men-in-The-Middle Angriffe sind möglich falls der Angreifer während Base Exchange präsent ist und die Kommunikationspartner ihre Identitäten gegenseitig nicht prüfen. Sobald aber die HIP Assoziation korrekt aufgebaut ist wird es sehr schwierig den UPDATE Paket zu erzeugen und es als legitim zu verkaufen. Richtige Signatur ohne den richtigen Schlüssel ist nicht realisierbar. Bei Denial-of-Service ist es vorstellbar, dass der Angreifer versucht die Ressourcen seiner Opfer auszuschöpfen. Zum Beispiel kann der Netzwerk angegriffen werden in dem der Angreifer sehr viele umfangreiche Nachrichten wiederholt verschickt. Gegen solche Angriffe bittet HIP keinen verbesserten Schutz. Versucht der Angreifer aber die Pakete an seiner Opfer nicht selbst zu verschicken sondern sie umzuleiten wird es durch die Prüfung der Adresse bzw. Erreichbarkeit beim HI Protokoll nicht mehr möglich.

Allerdings bietet HIP neue Angriffsmöglichkeiten. Die Frage ist was passiert wenn ein Angreifer beispielweise I1 oder I2 Pakete wiederholt verschickt. Sie werden zwar als falsch von dem Responder erkannt aber er wird trotzdem seine Ressourcen für die Berechnungen verbrauchen. Es ist auch möglich die R2 Nachricht zu kopieren und sie wiederholt an den Initiator zu verschicken so das er immer wieder mit der Lösung von altem Rätsel beschäftigt ist. Die Frage ob eine Identität gestohlen werden kann und wie lange es dauert bis es entdeckt wird bleiben unbehandelt. Eine Identität zu stehlen bedeutet einen privaten Schlüssel zu stehlen. Es ist zwar sehr schwierig, hat aber den Einschein, dass es danach um so einfacher wird. Der Angreifer braucht nur die Adresse mit UPDATE zu ändern und bei mehreren Adressen die neue Adresse als bevorzugt anzumelden. Wie viel Schaden kann der

wohl einrichten bis es erkannt wird. Mechanismen, welche darauf ausgerichtet sind solche *unwahrscheinliche* Angriffe schnell zu entdecken werden vermutlich kaum entwickelt.

Der Einsatz von Rendezvous Server und DNS Erweiterungen verlangen auch nach zusätzlichen Schutzmechanismen damit die Einträge ohne Integrität und Authentifizierung nicht möglich sind. Der Rendezvous Server bittet zusätzliche Fläche für Men-in-The-Middle Angriffe gegen Base Exchange. Die Kommunikation mit DNS Server ist auch für Men-in-The-Middle und Denial-of-Service Angriffe anfällig. Die Aufbau eines sicheren Kanals ist ein MUSS. Sonst sollen die Einträge nicht erlaubt sein.

1.3.3 HIP Vor- und Nachteile

Einer der Vorteile von HIP ist sicherlich die Trennung der Applikationen von der IP Adressen. So kann die Mobilität an besten unterstützt werden. Für die oberen Schichten Spielt es dann überhaupt keine Rolle wo der Kommunikationspartner sich gerade befindet. Ein weiterer Vorteil ist die Trennung der Identifier Funktion von der IP Adresse. Der Host Identity Namespace fühlt eine Lücke zwischen IP und DNS Namensräumen. Außerdem Nutz HIP die erworbene Erfahrungen in der Kryptographie um die Sicherheit zu erhöhen. Weiterer Vorteil ist die Integration der IP Mobilität. Die IPv4 und IPv6 Systeme/Applikationen können über HIP miteinander Kommunizieren. So kann ein Rechner bei Bedarf gleichzeitig beide Schnittstellen - IPv4 und IPv6 haben. Die Nachteile liegen primär in grossen Aufwendungen, die sich aus notwendigen Erweiterungen bzw. Ausbau der Netzinfrastruktur ergeben. So muss ein Dienst zur Verfügung gestellt werden, welcher die Verwaltung und häufige Aktualisierung der Abbildungen zwischen Host Identitäten und Host Adressen pflegt. Ob dazu ein neuer Dienst ausgebaut werden soll oder DNS erweitert werden soll, der Aufwand ist in beiden Fällen gross. Die Mögliche Einführung der Rendezvous Servers steigert die Wartung. Es ist auch nicht klar ob es möglich ist den Host Anhang seiner Identität zu finden obwohl seine IP Adressen nicht registriert sind. Nachteilig ist auch die notwendige Anpassungen an den API. Die Hosts selbst müssen natürlich auch für HIP nachgerüstet werden.

Es besteht die Befürchtung, dass die Entscheidung getroffen wird sich lieber auf die breite Einführung von IPv6 zu konzentrieren und mit den Erfahrungen lieber die bestehenden Dienste so weit wie möglich verbessern und sich nicht gleich mit der Einführung von HIP beschäftigen. Der notwendige Aufwand könnte dazu beitragen, dass HIP keine breite Unterstützung findet.

1.3.4 Zusammenfassung

HIP trennt den Transportschicht von der IP Adresse ab und bindet es stattdessen an die Host Identität an. Ermöglicht wird es durch die Einführung einen neuen Namensraums für Host Identitäten. Für Identitäten ist der Einsatz der öffentlichen Schlüsseln vorgesehen. Die Kommunikation zwischen Host Identitäten hält die Veränderungen in der IP Adressen transparent. So bittet HIP eine sichere Unterstützung für Host Mobilität und Multihoming mittels kryptographisch basiertem Namensraums. Ein sehr wichtiger Mechanismus ist

der sogenannter Base Exchange. Seine primäre Aufgabe besteht darin eine sichere HIP Assoziation aufzubauen, bevor die eigentliche Nutzdaten übertragen werden. Dafür sind Schlüsseln und Signatur notwendig, die machen eine Erstellung von falschen Nachrichten unmöglich. Der Protokoll besitzt auch ein Pakettyp - UPDATE Paket, der speziell dafür entwickelt wurde um die Korrespondenten über die neue IP Adresse(n) zu informieren. Alle HIP Pakete besteht aus HIP Header mit Initiator und Responder HIT und HIP Parameter. Ein sehr wichtiger und als kritisch empfundener Parameter ist der LOCATOR. Weitere Parameter sind zum Beispiel Rätsel, seine Lösung und Diffie-Hellman Schlüsseln, die in Base Exchange ausgetauscht werden. Um auch die Kommunikation zwischen den Hosts zu ermöglichen die sich noch nicht kennen bzw. die Ihre aktuelle Positionen nicht kennen sind zusätzliche Dienste notwendig, welche die Einträge über Hostidentität, sein HIT und IP Adressen verwalten. Ein Vorschlag ist die Erweiterung der DNS. Bei häufigem Adressenwechsel besteht aber die Möglichkeit, dass die Einträge nicht aktualisiert werden. Deshalb soll ein neuer Server - Rendezvous Server eingeführt werden. Sein Dienst ist die Verwaltung und Pflege von IP Adressen der mobiler HIP Hosts. Solche Server werden dann auch bei DNS in den Host Einträgen festgehalten. In Sicherheitsfragen bietet HIP zumindest keine schlechtere Lösungen als die IP Protokolle. Allerdings entstehen neue Angriffsflächen. Base Exchange kann durch die wiederholte Übertragung der Pakete dazu genutzt werden um die Host Ressourcen zu beanspruchen. Die Registrierungspakete an Rendezvous und DNS sollen in einem sicherem Kanal übermittelt werden, sonst wird es ein leichtes Unterfangen die falsche Adressen einzuschleusen. Entweder bekommt der Host dann einfach seine Nachrichten nicht oder sie werden beispielweise an den Angreifer geschickt.

1.4 HIP und Applikationen

Dieser Kapitel behandelt die Anforderungen, welche HIP an die Applikationen stellt.

1.4.1 HIP mit IP Applikationen

Auch wenn ein Host den HI Protokoll unterstützt ist es kaum anzunehmen, dass alle Applikationen HIP fähig sind. Aus der Sicht der Applikationen unterstützen die HIP Systeme eine neue Adressenfamilie von Host Identifiers. Es kann aber sehr viel Zeit vergehen bis HIP Anwendungen breitflächig zum Einsatz kommen, auch wenn die Rechner schon für HIP ausgerüstet wären. Vielmehr werden HIP gerüstete Systeme noch viele IP Adressen gebundene Anwendungen haben. Um HIP in solchen Situationen zu nutzen gibt es grundsätzlich drei unterschiedliche Ansätze:

- Die Applikationen nutzen weiterhin IP Adressen. Das System sorgt für die Abbildung von IP Adressen auf die Hostidentitäten und zurück.
- DNS zu nutzen um die Anwendungen entweder mit Alias - Namen für Hostidentitäten oder, im Fall von IPv6, direkt mit HITs zu versorgen.
- Die direkte Nutzung von HITs anstatt der IPv6 Adressen unterstützen.

Weitere Nutzung der IP Adressen Hier setzen die Applikationen weiterhin IP Adressen ein. Für die Sicherung von Verbindungen wird möglicherweise HIP Assoziation erwünscht. Weder die Anwendung noch der Benutzer haben aber einen Weg um explizit anzugeben den HIP zu nutzen. Aus diesem Grund muss die Entscheidung den HIP aufrufen in Host Policy verankert sein. Die Abbildung der IP Adresse auf HI könnte mittels Modifizierungen im Betriebsystem oder durch Unhüllung von Socket API erfolgen. Die IP Adresse - HIT Bindungen sind dafür lokal oder im DNS gespeichert. Wobei die Annahme, dass jeder Rechner alle Bindungen für sich lokal speichert, eher unrealistisch erscheint. DNS ist eher für die transparente Nutzug der Hostidentitäten geeignet.

DNS Nutzung In dem voran gegangenen Unterkapitel wurde die Verwendung von DNS für durch HIP Systeme erwähnt. Für die Applikationen, die DNS nutzen, bittet die Namensauflösung eine weitere Möglichkeit an. Falls DNS über die Informationen bezüglich der Bindung von Domänennamen an Host Identifier verfügt, kann anstatt der IP Adresse der 32 Bit Local Scope Identifier (LSI) oder 128 Bit HIT zurück gegeben werden, abhängig davon ob es sich um IPv4 oder IPv6 handelt. Anderenfalls wird wie üblich IP Adresse zurück gegeben. Die Applikationen nutzen LSI oder HIT als ob es eine IP Adresse wäre. Die Nutzung von LSI bzw. HIT anstatt IP ist nicht bei allen Anwendungen möglich. Was passiert, zum Beispiel, wenn eine Anwendung LSI an ein System, welches LSI nicht zurück zu Host Identifier oder IP Adresse auflösen kann, weiterleiten.

Direkte HIT Verbindung Direkte HIT Verbindung ist nur für IPv6 Applikationen möglich. Die Applikationen könne so konfiguriert werden, dass sie die Verbindung direkt mit einem HIT aufnehmen. D.h. anstatt connect(IP) führt ein Socket connect(HIT)durch. Dieser Ansatz ist sicherer als die ersten beide, weil die Applikation den Empfänger benennt. Die Herausforderung ist die entsprechende IP Adresse Anhang von HIT zu finden. Spezieller Dienst muss dafür vorhanden sein, wie beispielweise DNS Erweiterung oder Rendezvous Server. Ein weiteres Problem tritt auf wenn die Hosts, trotz des vorhandenen Dienstes, sich nicht registriert haben.. Resultat: der Initiator kennt den HIT, IP Adresse ist nicht bekannt und IP Adresse kann nicht nachgeschlagen werden, da der Host sich (noch) nicht registriert hat.

Fazit Auch wenn Systeme für HIP ausgebaut werden, ist nicht damit zu rechnen, dass alle Applikationen in so einem System auch sofort HIP fähig sein werden. Viel mehr ist es ein langer und kontinuierlicher Prozess. Um HIP trotzdem zu nutzen, sind viele Anpassungen notwendig. Wird weiterhin die IP Adresse verwendet, muss entweder am Socket oder am Betriebsystem gearbeitet werden. Hier ist die Frage ob die Hersteller der Ansatz unterstützen. Jeder Ansatz erfordert die Ausbau eines Dienstes für die Pflege und Verwaltung von Bindungen. Die Hosts sollen aufgefordert sein sich bei dem Dienst anzumelden, anderenfalls ist unklar wie die Adresse v.a. von mobilem Host Anhang von HIT gefunden werden kann.

1.4.2 HIP Applikationen

HIP Vorschlag beinhaltet zwei umfangreiche Änderungen im Netzwerkstapel, nämlich einen zusätzlichen, kryptographisch basierten Namensraum und eine neue Protokollsschicht. Um diese Veränderungen für die Applikationen nutzen zu können sollen Dienste für HIs und HIs-IP Adressen ausgebaut werden. Folgende Ausführungen widmen sich kurz der Frage an, welche funktionale Anforderungen muss eine HIP basierte Applikation erfüllen. Dabei wird angenommen, dass die Applikationen den DNS Dienst nach Identitäten und ihre Adressen abfragen kann.Die von M.K. implementierte HIP Applikation hat die Realisierbarkeit der Anforderungen in der Praxis bestätigt [14].

Funktionale Anforderungen Die funktionale Anforderungen konzentrieren sich auf Identitäten, Mobilität und Sicherheit. Die Applikationen sollten die beiden Repräsentationen, variablen und fixen Länge, für Host Identity unterstützen. Ausserdem müssen sowohl öffentliche wie auch private Schlüssel unterstützt werden. Die Applikationen sollen auch ihre eigene Host Identity kennen und bei Bedarf übergeben können. Weiter gilt es auch die genaue Prozesse der Applikationsschnittstelle zu definieren und damit festzulegen wie Identity an der Schnittstelle genutzt werden kann. Eine Anwendung soll auf die Dienste der Vermittlungsschicht mittels HIP Schicht zugreifen. HIP Schicht übernimmt dabei die transparente Suche nach der richtigen Adresse. Es sollte keine Rolle spielen, ob es sich um die Adresse eines Rendezvous Servers oder eines Rechners handelt. API soll imstande sein eine Verbindung ohne HI Kenntnis aufzunehmen, d.h. im opportunistischen Modus. Für die Mobilität gilt die Annahme, dass die Hosts ihre Adressen dynamisch und völlig transparent ändern. Die Applikation soll die Veränderungen nicht beobachten können. Für die Fragen der Sicherheit sollte es den Applikationen erlaubt sein die Sicherheitsattribute festzulegen. Dabei handelt es sich um Verschlüsselung, kryptographische Algorithmen für die Kommunikation, Schwierigkeitsgrad von Rätsel usw.

Zusätzlich zu den funktionalen gibt es eine Reihe von nicht-funktionalen Anforderungen. Dazu gehören backward und forward Kompatibilität.

1.5 HIP im Vergleich

Dieser Kapitel stellt die Mobile IPv4 und IPv6 den HIP gegenüber. Den Schwerpunk bilden die Mechanismen für die Einrichtung einer Assoziation und Mobilitätsunterstützung.

1.5.1 Leistung

Die Leistung von IPv4 ist durch den Einsatz von Home Agenten beeinträchtigt. Zusätzlich verlangt das Protokoll nach einem Foreign Agent. Die beide Agenten verarbeiten jede Nachricht, welche an einen registrierten Host adressiert ist. Bei IPv6 ist dagegen eine Route Optimierung Möglicht, vorausgesetzt es handelt sich um Mobile IPv6 Korrespondenten. Bei der Kommunikation zwischen IPv4 und IPv6 kommt wieder der Home Agent

zum Einsatz. HIP dagegen benutzt kein Heimnetzwerk Konzept. Der Aktualisierungsprozess ist bei HIP besser gelöst als in IPv6. Beide Protokolle schicken hierzu zuerst vier Nachrichten ab. Kommunikation zwischen HIP Rechner - Base Exchange erfolgt direkt. Return Routability Procedure von IPv6 schickt zwei der Pakete durch den Home Agenten. Bei weiten Strecken und vielen registrierten Rechner dürfte es als Nachteil bewertet werden. Um die Unterschiede in messbaren Werten anzubieten wurden HIP Systeme und IPv6 Systeme prototypisch realisiert und die Performance gemessen [12]. Die Messungen wurden an mobilen Knoten durchgeführt und zwar aus der Sicht des Endbenutzers., wobei v.a. Zeit für Paketübertragungen von Interesse war. Bei Mobile IPv6 enthält die Verzögerung die Zeiten für Registrierung bei Home Agent, Return Routability Procedure und Binding Update. Bei HIP sind es die Zeiten für Base Exchange und UPDATE. Zu beachten ist aber, dass IPv6 Implementierung einen Fehler in vertical handover gehabt hat.Dieser Fehler hat dazu geführt, dass obwohl die Bestätigung zu neuer Bindung von dem Korrespondenten noch nicht angekommen war,der Initiator die Daten gesenden hat. Die Ergebnisse sind 8,05 Sekunden für IPv6 und 2,46 Sekunden für HIP.

HIP bewirkt Erweiterungen in DNS oder den Ausbau von neuen Diensten um die HIT- IP Bindungen zu verwalten. HIP schlägt den Einsatz von Rendezvous Server vor. Einerseits ist DNS für häufigen Adressenwechsel zu langsam, andererseits soll der RVS den DNS entlasten. DNS sollte dann aber zusätzlich die Informationen über RVS enthalten. In IPv6 werden die Adressen bei Home Agent nachgefragt.

1.5.2 Sicherheit

Für die Authentifizierung in IPv4 wird MD5 Hash - Funktion benutzt. Die Nachrichten zwischen Host und seinem Home Agent werden immer überprüft. Sonst besteht die Gefahr von Remote Redirection. Gegen Replay Angriffe werden Pakete mit Timestamp oder einem zufällig erzeugtem Wert geschützt. Wobei Timestamp die Uhrensynchronisierung mit Abweichungsgrenzen benötigt. IPv6 kann mit IPsec genutzt werden. Verschlüsselt wird mit SHA Algorithmus. Falsche Bindungen resultieren in Denial-of-Service Angriffen. Aus diesem Grund sind die Bindung Update Nachrichten zu schützen. Return Routability Procedure erhöht zwar die Sicherheit sollte aber in der Zukunft durch stärkere Mechanismen ersetzt werden. HIP ist dagegen eng mit IPsec verbunden und bietet einen schnellen Schlüsselaustausch an. Der Protokoll schlägt auch vor die Identitäten in DNS zu speichern. In diesem Fall werden die Man-in-The-Middle Angriffe erschwert. Bevor eine Bindung aktualisiert werden kann, wird eine sichere ESP HIP Assoziation erstellt. Der Erstellungsmechanismus basiert auf Cookies. Responder kann den Schwierigkeitsgrad seiner Cookies wählen. Je schwieriger der Rätsel desto länger nimmt seine Berechnung in Anspruch.Es wurde eine HIP System auf Linux mit 266 MHz Pentium. aufgebaut und die Abhängigkeit zwischen dem Schwierigkeitsgrad und Berechnungszeit getestet [13] . Es hat sich gezeigt, dass der Aufwand sich exponentiell mit Anzahl Bits erhöht, d.h je schwieriger der Rätsel desto grösser seine Länge. Um so aufwendiger ist es auch für den Angreifer. Bei diesem Experiment wurde auch die Zeit für HIP Base Exchange berechnet. Der Ergebnis lag unter 1 Sekunde.

1.6 Zusammenfassung

Internet Protokolle wurden für Netzwerke mit stationären Rechner entwickelt und hat keine Unterstützung für die Mobilität. Die immer steigende Anzahl an mobilen Geräten verlangt nach neuen Technologien. Mobile IP bietet die Erweiterungen für IP Protokolle, mit dem Ziel die Mobilität für die Applikationen transparent zu halten. Auf diese Weise kann die Kommunikation nach der Adressenänderung fortgesetzt werden. IPv4 assoziiert einen mobilen Host immer mit seiner primären IP Adresse. Alle Verbindungen laufen über diese Adresse ab. So ein vorgehen kann aber zu Überlastung der Home Agent führen. IPv6 ermöglicht zusätzlich eine direkte Kommunikation. Dazu müssen die mobile Hosts ihre Bindungen bei ihren Korrespondenten aktualisieren. Direkte Verbindung kann aber nur zwischen IPv6 Rechner stattfinden, da IPv4 Rechner keine Bindungstabellen verwalten können. For IETF entwickelte Host Identity Protocol stellt einen neuen Ansatz vor. Die Idee ist die Identität eines Hosts von seiner Adresse zu trennen. Die IP Adresse soll nur zu Lokalisierung genutzt werden. Die Identität der Hosts wird durch zusätzlichen Namensraum definiert. Da die Identität eindeutig sein muss, wurden die Erfahrungen aus Kryptographie genutzt und für Identitäten die öffentliche Schlüssel vorgeschlagen. Als Konsequenz wird die Schichtenstapel um eine Schicht, HIP Schicht, grösser. Die Hosts sprechen einander nicht mit IP Adressen an sondern mit Identitäten. HIP Schicht übernimmt unter anderem die Transformation von HI auf die entsprechende IP Adresse(n). Eine der Fragen ist hier die Pflege von Host Identitäten und ihrer Bindungen an die IP Adressen oder der Auffindung eines Hosts Anhang seiner Identität. Dazu ist die Ausbau von entsprechenden Diensten notwendig. DNS Erweiterung ist eine solche Möglichkeit. Allerdings ist DNS für dynamische Adressen Aktualisierung zu langsam. Aus diesem Grund wurde ein weiterer Dienst entwickelt, sog. Rendezvous Server, der DNS unterstützen soll indem er die Pflege von Bindungen übernimmt. HIP ist stark mit IPsec verankert und bietet bessere Sicherheiten als IP Protokolle. Es wurden einige HIP Systeme prototypisch implementiert um die Machbarkeit und Leistung gegenüber IP Protokollen zu testen und es konnte dabei gezeigt werden, dass HIP z.B. den Adressen Update schneller durchführt. HIP hat einen grossen Leistungs-, Sicherheit- und Adressierungspotential. Es befinden sich aber noch in sehr früher Entwicklungsphase und bedarf noch viele Untersuchungen über seinen Einfluss auf Internet.Ein HIP Prototyp kann unter [25] geladen werden.

Literaturverzeichnis

[1] R. Moskowitz, P. Nikander: Host Identity Protocol (HIP) Architecture. IETF RFC 4423, May 2006.

[2] R. Moskowitz, P. Nikander, P. Jakela, T. Henderson: Host Identity Protocol. IETF RFC 5201, April 2008.

[3] P. Nikander, T. Henderson, J. Arkko: End-Host Mobility and Multihoming with the Host Identity Protocol. IETF RFC 5206, April 2008.

[4] T. Henderson, P. Nikander, M. Komu: Using the Host Identity Protocol with Legacy Applications. Internet Draft, draft-ietf-hip-applikations-02, 18 November, 2007.

[5] A.S. Tanenbaum: Computernetzwerke 4., überarbeitete Auflage. Pearson Studium Verlag, 2003.

[6] P. Nikander, J. Laganier: Host Identity Protocol (HIP) Domain Name System (DNS) Extension. IETF 5205, April 2008.

[7] J. Laganier, L. Eggert: Host Identity Protocol (HIP) Rendezvous Extension. IETF RFC 5204, April 2008.

[8] C. Perkins: IP Mobility Support for IPv4. IETF RFC 3344, August 2002.

[9] D. Jonson, C. Perkins, J. Arkko: Mobility Support in IPv6. IETF RFC 3775, June 2004.

[10] A. Matos, J. Santos, S. Sargento, R. Aguiar, J. Girao, M. Liebsch: HIP Location Privacy Framework. ACM, Mai 2006.

[11] T.R. Henderson, B.P. Works: Host Mobility for IP Networks : A Comparison. IEEE, November/December 2003.

[12] P. Jokela, T. Rinta-aho, T. Jokikyyny, J. Wall, M. Kuparinen, H. Mahkonen, J. Melén, T. Kauppinen, J. Korhonen: Handover Performance with HIP and MIPv6.

[13] T. Henderson, J.M. Ahrenholz, J.H. Kim: Experience with the Host Identity Protocol for Secure Host Mobility and Multihoming. IEEE WCNC, March 2003.

[14] M. Komu: Application Programming Interfaces for the Host Identity Protocol. Helsinki University of Technology, September 2004.

[15] P. Nikander: Applying Host Identity Protocol to the Internet Addressing Architecture. IEEE, May 2004.

[16] F. Al-Shraideh: Host Identity Protocol. IEEE, 2006.

[17] P. Jokele, P. Nikander, J. Melen, J. Ylitalo, J. Wall: Host Identity Protocol - Extended Abstract. Ericsson Research NomadicLab.

[18] P. Jokela: Host Identity Protocol, Study Cluster. Ericsson, 19 April 2005.

[19] P. Jokela, P. Nikander, J. Melen, J. Ylitalo, J. Wall: Host Identity Protocol: Achieving IPv4 - IPv6 handovers without tunneling. Ericsson Research NomadicLab.

[20] P. Nikander, J. Ylitalo, J. Wall: Integrating Security, Mobility, and Multi-homing in a HIP Way. Ericsson Research NomadicLab.

[21] N.B. Jee, W. Wu, S.K. Das, S. Dawkins, J. Pathak: Mobility Support in Wireless Internet. IEEE Wireless Communication, October 2003.

[22] S. Navaczki, L. Bokor, S. Imre: Micromobility Support in HIP, Survey and Extension of Host Identity Protocol. IEEE MELECON, 16-19 May 2006.

[23] M. Bagnulo, A. Garcia-Martinez, A. Azcorra: Efficient Security for IPv6 Multihoming. ACM SIGCOMM Computer Communications Review, April 2005.

[24] A. Rakotonirainy: Trends and Future of Mobile Computing. The University of Queensland.

[25] P. Jokele, P. Nikander, J. Melen, J. Ylitalo, J. Wall: HIP for inter.net Project. `http://www.hip4inter.net`, last visited: June 2008.

Kapitel 2

RFID Karten und RFID Kartenleser für BioXes

Monir Mahdavi

RFID ermöglicht berührungslose Authentifizierung von Gegenständen oder Lebewesen sowie die Speicherung und Bearbeitung von Daten. Eine Einführung in die RFID Technologie und ihre Integration in Anwendungen im Bereich der Zugriff- und Zutrittskontrolle sind Gegenstand der vorliegenden Arbeit. Eine Übersicht über die gängigen Marktentwicklungen der RFID Systemen, die Key-Players und ihre Produkte runden das Thema ab. RFID wird anschliessend mit der biometrische Authentifizierung verglichen. Bei dieser Methode werden die biometrischen Merkmale wie z.B. Iris, Fingerprint, Gesicht, Augen usw. zur Authentifizierung eingesetzt. Am Schluss wird das Softwaresystem BioXes vorgestellt. BioXes, entwickelt an der Universität Zürich und der Universität der Bundeswehr München(UniBwM) wird zur Verwaltung und Verteilung biometrischer Daten auf biometrische Geräte eingesetzt. Nachfolgend werden die Architektur sowie Funktionalitäten der BioXes Applikation vorgestellt. Ziel ist es, dass BioXes in Zukunft nebst der Verwaltung biometrischer Daten und Geräte auch RFID Geräte integriert. Es soll nach Lösungsvorschlägen zur Integration von RFID in BioXes gesucht werden.

Inhaltsverzeichnis

2.1 Einleitung

Bei Auslandsreisen, bei Geldüberweisen oder bei Online Einkäufen kann man mit einem Erkennungs- oder Authentifizierungsprozess konfrontiert werden. Wichtig ist dabei, dass der Erkennungsprozess abgesichert ist. Die folgenden Ansätze sind typisch in Authentifizierungsprozessen:

- Etwas haben.
- Etwas wissen.
- Etwas sein. [5]

Ein Beispiel zu dem Ansatz - etwas haben - ist der Besitz einer speziellen Karte, welche die Person eindeutig in einem System authentifiziert. Es gibt verschiedene Technologien, die zur Authentifizierung mittels einer Karte eingesetzt werden können.

RFID steht für Radio Frequency Identification und ist eine Technologie, die seit mindestens zwei Jahrzehnten in vielfältigen Applikationen Anwendung findet. In den vergangenen Jahren wurde RFID beispielsweise zur Tieridentifikation, in Wegfahrsperren oder auch in Abonnementen für Skipisten eingesetzt. RFID wird ebenfalls in Lieferketten und Logistikprozessen von Handelsunternehmen eingesetzt. Ein verbreiteter Einsatzbereich von RFID ist das Authentifizieren von Personen beim physischen, wie auch beim logischen Zugriff, z.B. bei Eintritt in ein Zimmer/Gebäude oder beim Einloggen in einen Computer [3].

Die zu authentifizierende Person besitzt eine spezielle Karte, in der ein Mikrochip integriert ist (Abbildung 2.1). In dem Mikrochip sind Informationen zur eindeutigen Authentifizierung der Person gespeichert. Eine Besonderheit des RFID Systems ist, dass Informationen mittels Radiowellen berührungslos und ohne Sichtkontakt zu einem Lesegerät übertragen werden. In den folgenden Abschnitten wird diese Technologie näher beschrieben.

Abbildung 2.1: Kontaktlose Authentifizierung mittels einer Chipkarte [21]

Um den Authentifizierungsprozess sicherer zu gestalten, wird das obige Szenario (etwas haben) mit dem Ansatz - etwas wissen - kombiniert. Werden die Informationen im Mikrochip verschlüsselt gespeichert, so fragt das Lesegerät nach dem Empfangen der Daten nach einem Kennwort. Wird das richtige Kennwort eingegeben, werden die Daten entschlüsselt und weitergeleitet, ansonsten wird der Prozess blockiert.

Bekanntlich sind keine der Sicherheitsansätze hundertprozentig sicher, sodass der Authentifizierungsprozess keine Garantie gewährt [5]. Die Gefahr bei den Ansätzen - etwas haben und etwas wissen - ist das Verlieren, Diebstahl der RFID Karte oder das Vergessen des Kennworts. Unter Umständen können Zugriffsschlüssel aus der Karte gelesen werden: Wissenschaftler aus dem Umfeld des Chaos Computer Clubs konnten den Zugriffsschlüssel der Mifare Karte innerhalb einiger Minuten mit Hilfe von Spezialhardware offen legen [1].

Bei dem Ansatz - etwas sein - fallen im Normalfall die oben genannten Gefahren weg. Die biometrische Authentifizierung basiert auf biometrischen Merkmalen einer Person (z.B. Fingerprint, Iris, Handfläche). Diese kann man nicht verlieren oder vergessen [7].

Eine andere Kombination oben genannter Sicherheitsansätze zeigt das folgende Beispiel: Etwas sein und etwas haben. Die Authentifizierung ist erfolgreich, wenn die Person ihre biometrischen Merkmalen (z.B. ihre Fingerabdrücke) mit den biometrische Daten - gespeichert in der Chip-Karte - nachweisen kann.

Diese Seminararbeit behandelt die Analyse und die Integration zwei verschiedener Technologien: RFID und biometrische Authentifizierung. Das Softwareprodukt BioXes, welches biometrische Authentifizierung in verschiedenen Arten und Formen zur Authentifizierung der Personen realisiert, soll mit RFID Technologien erweitert werden.

2.2 RFID Komponenten und ihre Mechanismen

Radiofrequenz-Identifikation oder RFID ermöglicht es, Daten mittels Radiowellen berührungslos und ohne Sichtkontakt zu übertragen. Eine typische RFID-Systeminfrastruktur kann in drei Komponenten aufgeteilt werden:

- Ein Transponder oder Tag.
- Ein Sende-Empfangs-Gerät.
- Ein im Hintergrund wirkendes IT-System.

Herzstück der Technologie ist ein Transponder oder Tag - ein winziger Computerchip mit Antenne. Der Transponder ist in ein Trägerobjekt integriert, beispielsweise in eine Plastikkarte. Auf dem Chip sind die Daten zur Authentifizierung gespeichert. In der Regel wird in dem Chip eine eindeutige Nummer gespeichert. Diese Nummer referenziert Informationen, die in einer Datenbank hinterlegt sind. Dadurch erhält jeder Gegenstand mit RFID-Transponder eine unverwechselbare Identität. Es gibt aber auch Methoden die die Speicherung aller erforderlichen Informationen auf dem Chip ermöglichen. Bei dieser Methode müssen die Lesegeräte nicht mit einer Datenbanken verbunden sein. Stattdessen ist eine dezentrale Verwaltung und Steuerung möglich. Ein Vorteil dieser Methode ist, dass die Daten sich in der Regel auf dem Chip einfacher verändern lassen als im System. Der Nachteil ist dass die Lesevorgänge mehr Zeit benötigen und die Transponder teurer sind.

Ein RFID System funktioniert in der Regel wie folgt: RFID Transponder und Lesegeräte sind auf verschiedene Weise mit Antennen ausgerüstet (Abbildung 2.2). Zwischen Antennen werden die Funksignale (Radio Welle) ausgetauscht. Die Antenne der Lesegeräte sendet Funksignale aus, um den Transponder anzusteuern und Daten zu lesen bzw. zu schreiben. Die Daten des Mikrochips werden durch die Antenne von Transponder zum Lesegerät geschickt. Die Besonderheit der RFID Systeme ist, dass der Datenaustausch kontaktlos durch Radio Wellen erfolgt [6].

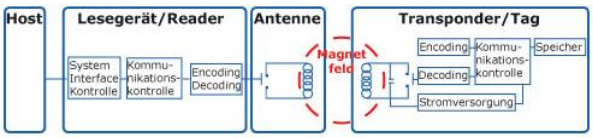

Abbildung 2.2: RFID Komponenten und ihre Mechanismus [23]

2.3 RFID Funktionsweise

Ein RFID System lässt sich durch mehrere technische Faktoren, wie z.B. Energieversorgung, Sendefrequenz, Kopplung und Speicherstruktur von Transpondern beschreiben und klassifizieren.

2.3.1 Energieversorgung

RFID Transponder brauchen Energie um auf die Befehle der Lesegeräte zu antworten und um ihre Mikrochips zu betreiben. Dabei unterscheiden sich wie folgt drei Formen von RFID Transpondern:

Passive RFID Diese Art von Transponder benutzen ihre Energie aus dem Feld, das vom Lesegerät erzeugt wird.

Aktive RFID Diese Art von Transponder besitzen interne Batterien.

Semi-Aktive RFID Dabei handelt es sich um eine Kombinationsart von oben genannten Transpondern. Semi-Aktive RFID besitzen interne Batterien, die zum Betreiben des Mikrochips erforderlich sind. Zum Senden der Daten an das Lesegerät benutzen sie Energie des Feldes, die vom Lesegeräte erzeugt werden. [3]

2.3.2 Sendefrequenz und Kopplung

Ähnlich wie beim Radio, das auf Ultrakurzwellen-, Mittelwellen-, und Langwellenfrequenzen sendet, gibt es auch für RFID verschiedene Funkfrequenz-Bereiche. Damit keine Kollisionen entstehen, werden Funkregularien festgelegt, welche die Frequenzstufen für Anwendungen reservieren. RFID Systeme können die so genannter ISM-Frequenzen benutzen,

die für Industrielle, wissenschaftliche (Science) und medizinische Anwendungen freigegen sind [6]. Welcher Frequenzbereich geeignet ist, hängt von der Art der Anwendungen ab. Die verschiedenen Frequenzbereiche haben unterschiedliche Eigenschaften. Zum Beispiel: je höher die Frequenzstufe ist, desto höher werden die Lesereichweite und Lesegeschwindigkeiten. In der Tabelle 2.1 sind die Frequenzstufe für RFID Systeme und ihre Anwendungen und Lesereichweite dargestellt.

Tabelle 2.1: RFID Frequenzstufen, Anwendungsbeispiele und Reichweite

RFID-Frequenzen	Anwendung (Beispielen)	Typische Reichweite
Niederfrequenz(lf) 125-135 kHz	Tieridentifikation, Produktionskontrolle, Zutrittkontrolle	1-1.5 Meter
Hochfrequenz (hf) 13.56 MHz	Handelsgüter, Bibliothekmanagement, Zutrittkontrolle	1-1.5 Meter
Ultrahochfrequenz(uhf) 860-960 MHz	Paletenidentifikation, Kartonindentifikation(Handel)	3-4 Meter(Europa), 7Meter (USA)

2.3.3 Speicherstruktur und Speicherkapazität der Transponder

Die Daten werden auf den Transpondern in definierten Strukturen abgelegt. Zu den Stammdaten werden noch Ergänzungen wie Datenerkennungs- und Fehlererkennungsbit gespeichert. Im Allgemeinen werden die folgenden Daten auf den Transpondern, die zur Eintritts- bzw. Zutrittskontrolle eingesetzt werden, gespeichert:

- Eine Kennungen, in denen eine numerische oder alphanumerische Zeichenkette zum Zweck der Authentifizierung gespeichert wird.
- Weitere notwendige Informationen abhängig von der Speicherkapazität und dem Anwendungsbereich der Transponder.

Im Allgemeinen variiert die Speicherkapazität der Transponder von 1 Bit bis hin zu Kilobits. Die Speicherkapazität der Transponder lässt sich wie folgt kategorisieren:

- Die 1-Bit-Transponder werden hauptsächlich zu Überwachungszwecken verwendet. Ein typischer Anwendungsfall für diese Transponder ist die elektronische Warensicherungen in Warenhäuser.
- Transponder mit Speicherkapazität bis zu 128 Bit reichen aus, um eine Serien- oder Identifikationsnummer eventuell zusammen mit Prüfsbits zu speichern. Solche Transponder können durch den Anwender programmiert werden.

- Transponder mit Speicherkapazität bis zu 512 Bit sind durch Anwender programmierbar und eignen sich zur Speicherung von Identifikations- oder andere bestimmte Daten wie Seriennummer, Paketinhalte und wichtige Verfahrensanweisungen.
- Transponder mit Speicherkapazität bis 64 Kbit können als Träger portabler Dateien betrachtet werden.
- Bei höheren Speicherkapazitäten gibt es Möglichkeiten die Daten in Felder bzw. Seiten einzuteilen [6].

In vielen Anwendungsfällen genügen Transponder, die nur eine Identifikationsnummer besitzen. Die Identifikationsnummer ist eine eindeutige Referenz zu weiteren Informationen in einer Datenbank oder in einem Informationssystem. In den Situationen, wo der Zugriff auf eine Datenbank oder auf ein System nicht immer möglich ist, werden Transponder mit zusätzlichen Datenspeicher eingesetzt, damit die Daten direkt auf den Transpondern gespeichert werden können.

2.4 RFID in der Zutrittskontrolle

In dem folgenden Abschnitte werden RFID Karten bzw. Geräten für die Zutrittskontrolle vorgestellt.

2.4.1 RFID Karten

Ein weit verbreiteter Anwendungsbereich für RFID Systeme ist die Eintritts- und Zutrittskontrolle. Hierbei werden Transponder und Antenne in eine RFID Karte eingebaut. Es gibt verschiedene Bauformen und Modelle von RFID Karten, die in den folgenden Abschnitten beschrieben sind.

2.4.1.1 Kontaktlose Smartkarten (Contactless Smartcard)

Bei den kontaktlosen Smartkarten werden Chip und Antenne in einer Karte eingebettet (Abbildung 2.3). Die Antenne transformiert die gespeicherten Daten auf dem Chip und sendet diese zu einem entfernten Computer. Die kontaktlosen Smartkarten müssen in Reichweite der Frequenz des Lese/Schreibgeräts gestellt werden um Daten zu Lesen bzw. Schreiben. Diese Reichweite ist abhängig von Lese/Schreibgerät Typen und beträgt ca. 63.5 mm - 99.06 mm.

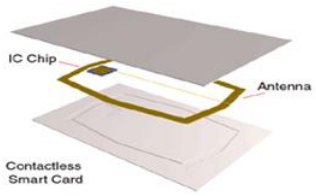

Abbildung 2.3: Kontaktlose Smartkarte [10]

2.4.1.2 Proximity Karten (Proximity Cards)

Proximity Karten (Abbildung 2.4) beinhalten wie kontaktlose Smartkarten einen eingebetteten Chip und Antenne zur Kommunikation mit einem Lesegerät. Im Unterschied zu kontaktlosen Smartkarten können Proximity Karten nur gelesen werden. Sie können aber in grösserer Reichweite vom Lesegerät gelesen werden (63.5 mm - 508 mm). Die gespeicherten Datenmengen auf dem Chip sind jedoch kleiner und haben in der Regel Platz für einen Identifikationscode. Proximity Karten sind populär für Anwendungen wie: Identifikation und Access Control.

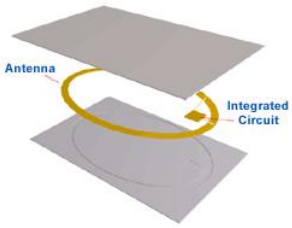

Abbildung 2.4: Proximity Karte [10]

2.4.1.3 Hybride Karten(Hybrid Cards)

Hybride Karten (Abbildung 2.5) beinhalten zwei oder mehrere eingebettete Chips: einen kontaktlosen Chip mit einer Antenne und einen kontaktgebundenen Chip mit einem Kontaktpad. Die kontaktgebundenen Chips werden für Applikationen eingesetzt, die einen höheren Sicherheitsgrad erfordern. Die beiden Chips haben keine Verbindung zueinander.

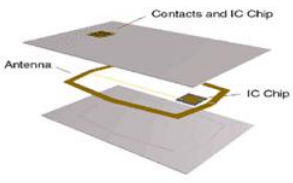

Abbildung 2.5: Hybride Karte [10]

2.4.1.4 Kombi Karten (Combi Cards)

Kombi Karten, auch bekannt als ein Dual Interface Card (Abbildung 2.6), haben einen eingebetteten Chip, auf den entweder durch ein Kontaktpad oder durch eine eingebettete Antenne zugegriffen werden kann. Diese Kartenart ist aufgrund einfacher Anwendbarkeit sehr populär und wird oft bei Applikationen eingesetzt, die viele IDs verarbeiten [10].

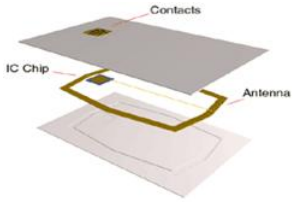

Abbildung 2.6: Kombi Karte [10]

2.4.2 RFID Lesegeräte

Ebenfalls wie bei RFID Karten gibt es bei RFID Lesegeräten verschiedene Bauformen und Modelle. RFID Lesegeräte sind aktive Einheiten, die Daten aus den Transpondern auslesen und ggf. Daten auf Transponder schreiben können (Lese/Schreibgeräte). Sie schicken Befehle zu den Transpondern über eine integrierte oder externe Antenne und empfangen die Antworten. RFID Lese/Schreibgeräte werden grundsätzlich in zwei Gruppen unterteilt: mobile und stationäre Lese/Schreibgeräte. Abhängig von der Leistung und des verwendeten Frequenzbereichs kann das Gerät Radiowellen über eine Distanz von 1 cm bis zu 33.3 cm senden.

2.4.2.1 Stationäre Reader

Bei vielen der eingesetzten RFID-Systeme handelt es sich um stationäre Reader, die Schnittstellen zu externen Antennen besitzen. Die Steuerung der Reader und die Verarbeitung der Informationen erfolgt durch Systeme, die über diverse Schnittstellen (Ethernet, RS232, RS485, usw.) mit dem RFID-Reader kommunizieren. (Abbildung 2.7, Abbildung 2.8, Abbildung 2.9)

Abbildung 2.7: Stationäre Lesegerät [22]

Abbildung 2.8: Antenne eines Stationären Lesegeräts [22]

Abbildung 2.9: Eine RFID Lesegeräte integriert in einem Tür [17]

2.4.2.2 Mobile Reader (RFID-Handhelds)

Mobile RFID Lese- Schreibgeräte sind in einer kompakten Bauform integriert. Dem Benutzer ist es möglich Authentifizierung vor Ort durchzuführen. Die Daten können in einem Handheld zwischengespeichert und werden zu einem späteren Zeitpunkt zum Host übertragen werden [10].

Abbildung 2.10: Intermec IP4, ein mobiler UHF RFID Leser / Schreiber [20]

2.5 Die Key-Player und ihre Produkte

MIFARE ist ein Akronym für Mikron Fare System (Mikron Fahrgeld-System). Mifare Smartcards von NXP Semiconductors sind kontaktlose Chipkarten. Entsprechend Herstellerangaben wurden davon bislang insgesamt 500 Millionen Karten und 5 Millionen Kartenlesegeräte verkauft. Die Mifare-Chipkarte wird von Infineon Technologies hergestellt. Die Standardkarte arbeitet in einer Distanz von bis zu 10 cm und nutzt dabei eine Frequenz von 13,56 MHz. Der Speicher (1024 Byte) ist in 16 Sektoren unterteilt, die jeweils unabhängig vor unerlaubtem Lesen bzw. Schreiben geschützt sind. Diese 16 Sektoren sind weiter in jeweils 4 Blöcke zu je 16 Byte unterteilt. Der letzte Block in jedem Sektor wird als Sector Trailer bezeichnet und beinhaltet zwei Schlüssel sowie die Access Conditions für den betroffenen Sektor. Dadurch wäre es möglich mit einer Mifare Karte mehrere,

unterschiedliche Applikationen zu bedienen (Multiapplikation). Dieser Chip findet z. B. in Studentenausweisen, wo er u. a. als bargeldloses Bezahlungs-Mittel für die Mensa dienen kann.

Mittlerweile ist die MIFARE-Technologie mit 13,56 MHz Basis für RFID geworden, da bei dieser Frequenz im Gegensatz zu anderen Systemen, die z.B. mit 125 kHz arbeiten, zwar die Reichweite geringer ist, jedoch in der selben Zeit mehr Information übertragen werden kann.

MIFARE hat auch den Markt für kontaktlose Chipkarten ohne eigene Stromversorgung mitgeprägt. Der Datenträger kommt ohne Batterie aus und wird durch das Magnetfeld der Basis-Station (Schreib-/ Lesegerät) und die im Datenträger vorhandene Drahtspule beim Durchführen durch das Magnetfeld mit Energie versorgt und kann mit dem Schreib-/ Lesegerät kommunizieren [9].

LEGIC Identsystems AG (CH) ist ein Anbieter kontaktloser Smartcard-Technologie auf der Frequenz von 13,56 MHz. Zum Produktangebot gehören Lese/Schreib-Chip-Sets, Lesermodule und Transponder-Chips, welche den LEGIC RF-Standard unterstützen sowie mit den Normen ISO 15693 und ISO 14443 konform sind. Die LEGIC-Partner bieten eine Vielzahl von LEGIC all-in-one-card Applikationen für Zeiterfassung, Zutrittskontrolle, Biometrie, E-Payment, Parken, Ticketing sowie weitere Multiapplikationen an [8].

2.6 Begriffliche Klärungen zur biometrischen Authentifizierung

Die Begriffe Biometrie und Biometrik haben verschiedene Bedeutungen. Die folgende Liste zeigt die Definitionen zu den Grundbegriffen:

- Biometrie (engl. biometrics), ein aus dem Griechischen stammender Begriff, bedeutet laut Lexikon: biologische Statistik, Zählung und Messung von Lebewesen.
- Biometrik ist das automatisierte Messen eines oder mehrerer spezifischer Merkmale eines Lebewesens (e.g. einer Person).
- Biometrische Authentifizierung verfolgt das Ziel, eine mittels Biometrik spezifizierte Person von anderen unterscheidbar zu machen [7].

In den technikorientierten Literaturen werden meistens biometrische Authentifizierung zu Biometrie verkürzt. Biometrie bedeutet jedoch in anderen wissenschaftlichen Bereichen wie Medizin, Mathematik, Statistik usw. unterschiedliches. Ebenfalls biometrisches Systeme bzw. biometrische Verfahren werden auch als biometrische Authentifizierungsverfahren interpretiert. Weitere begriffliche Unterscheidungen im Zusammenhang mit biometrischer Authentifizierung fallen auf Identifikation und Verifikation:

- Identifikation bedeutet, dass ein Individuum aus einer vorgegebenen Menge heraus erkannt wird (1:n).

- Bei der Verifikation wird eine geforderte Identifikation (1:1) entweder bestätigt oder verworfen.

2.7 Basismodell der biometrischen Authentifizierung

In den folgenden Abschnitten werden die Abläufe der biometrischen Authentifizierung gezeigt.

2.7.1 Merkmalsauswahl

Biometrische Authentifizierung basiert auf biometrischen Merkmalen einer Personen. Diese Merkmale können Eigenschaften der Personen oder ihre Verhaltensweise sein. Es können auch mehrerer Merkmale kombiniert eingesetzt werden.

2.7.1.1 Eignungskriterien von Merkmalen

Merkmale können nach folgenden Kriterien zur biometrischen Authentifizierung unterschieden werden:

- Universalität, ein Merkmal ist bei jeder Person vorhanden.
- Einzigartigkeit, ein Merkmal ist bei jeder Person anders.
- Permanenz, ein Merkmal ist zeitlich invariant.
- Erfassbarkeit, ein Merkmal lässt sich quantitativ erheben. Ausser den oben genannten Kriterien zum Eignung eines Merkmales muss es an Umsetzbarkeit und Implementierung auch gedacht werden.

Die folgenden Kriterien beschreiben die technischen Umsetzbarkeit:

Ökonomische Machbarkeit d.h. die Kosten müssen angemessen sein.

Überlistungsresistenz d.h. das Verfahren darf durch betrügerische Techniken zumindest nur schwer beeinflussbar sein.

Akzeptanz d.h. bei den Individuen muss die Bereitschaft bestehen, das zum Verfahren gehörende Merkmal zur biometrischen Authentifizierung zu verwenden.

2.7.1.2 Synopsis von Verfahren zur biometrischen Authentifizierung

Die Tabellen 2.2 und 2.3 zeigen die in der Praxis verwendeten Verfahren. Auf das DNA Verfahren wurde verzichtet, da es nur in speziellen Situationen verwendet wird.

Tabelle 2.2: Biometrischer Authentifizierungsverfahren, die auf Körpermerkmalen basieren

Biometrische Authentifizierungsverfahren	Biometrische Merkmale	Charakteristiken
Fingerbilderkennung	Muster der Haut der Fingerkuppe	Bild der Fingerlinie, Klassenbildung, Charakteristische Merkmale(Minuzien)
Gesichtserkennung	Gesichtsbild und geometrische Merkmale	Attribut-Ansatz: Attribute wie Nase, Augen und die spezifischen Grössen und Anordnungen
Handerkennung	Masse, Form und Figuren von Fingern und Handballen	Länge der Finger, Profil der Hand
Iriserkennung	Muster des Gewebes um die Pupille	Texturanalyse
Retinaerkennung	Muster der Blutgefässe im Augenhintergrund	Texturanalyse der Netzhaut bzw der Retina

Tabelle 2.3: Biometrischer Authentifizierungsverfahren, die auf Verhaltensmerkmalen basieren

Biometrische Authentifizierungsverfahren	Biometrische Merkmale	Charakteristiken
Sprechererkennung	Stimme	Sowohl von vorgegebenen Texten abhängige wie unabhängige Lösungen sind bekannt
Unterschrifterkennung und Schrifterkennung	Schreibverhalten	Geschwindigkeit, Druck, Beschleunigung des Schreibvorgangs
Tastendruckdynamik	Tipprhythmus	Gemessen werden Druckdauer und Zwischenzeiten der Tastenbetätigung
Optische Sprechererkennung	Mimik	Analyse von Bewegungsabläufen beim Sprechen vereinbarter Texte

2.7.2 Messung von Merkmalen

Bei Messungen bzw. Erfassung der Biometrische Merkmale können Probleme auftreten:

- Ungenauigkeiten des Messgerätes: beispielsweise bei der Gesichtserkennung wird eine Kamera eingesetzt. Die Genauigkeiten (Auflösung, Signalqualität) der Kamera kann variieren.
- Instabilität der Merkmale: die Merkmale einer Person können sich ändern. Müdigkeit, Schönheitsoperationen und Alter sind Beispiele von veränderbaren Merkmalen einer Person.
- Ungenauigkeiten bei der Bearbeitung von aufgenommenen Merkmalen: bei der Verarbeitung der Messwerte können Ungenauigkeiten und Rundungsfehler entstehen. Die Abbildung 2.11 zeigt die Abweichungen bei einem Fingerbild während des Messungs- und Erfassungsprozesses.

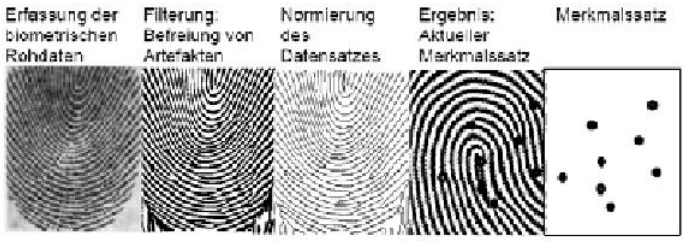

Abbildung 2.11: Messung und Weiterbearbeitung eines Fingerabdrucks [7]

2.7.3 Individualisierung

Individualisierung bei der biometrischen Authentifizierung heisst die Erkennung eines einzelnen Elements aus einer Gesamtmenge. Die Menge von Elementen wird solange in Teilmengen geteilt bis das gesuchte Element gefunden bzw. nicht gefunden wird. Bei der Individualisierung sind folgenden Schritte zu betrachten.

2.7.3.1 Registrierungsprozess (enrollment process)

Das biometrische Merkmal der Person muss in einem Vorprozess registriert werden. Das Ergebnis der Registrierung wird als ein Template bezeichnet. Ein Registrierungsprozess besteht normalerweise aus folgenden Schritten:

- Erfassung der relevanten Merkmale.
- Bearbeitung der relevanten Merkmale.
- Speicherung als Datensatz (Template).

2.7.3.2 Identifikation

Bei der 1:n Identifikation handelt es sich um das Vergleichen eines aktuell erfassten biometrischen Merkmals mit vielen anderen voraufgenommen Merkmalen, die bereits im System vorhanden sind. Eine Identität gilt als gefunden wenn im System ein ähnliches gespeichertes biometrisches Merkmal als Datensatz existiert. Es gibt jedoch einige kritische Punkte bei diesem Verfahren zu erwähnen:

- Das Vergleichen der aktuell erfassten Merkmale mit den gespeicherten Datensätzen kostet bei wachsenden Eingabemengen in der Datenbank (Skalierbarkeit) mehr Aufwand. Bei den Wachsenden Eingabemenge wird ebenfalls mehr Speicherkapazität benötigt.
- Datenschutz Probleme und Restriktionen, z.B. bei Fingerabdrücken und daraus extrahierten Minuzien handelt es sich um statische biometrische Daten. Sie sind zeitlich unveränderliche und unverwechselbare Wesensmerkmale einer Person, die sich andere Personen nicht aneignen können. Ohne Verbindung zur Identität stellen die biometrischen Elemente keine Personendaten dar. Mit Verbindung zur Identität dagegen bilden die Fingerabdrücke schützenswerte Daten, da sich daraus die Rassenzugehörigkeit rekonstruieren lässt. [19].

2.7.3.3 Verifikation

Im Gegensatz zu der Identifikation ist das Vergleichen bei der Verifikation 1:1. Es wird ein aktuell erfasstes biometrisches Merkmal mit einem vorgelegten Datensatz verglichen. Der vorgelegte Datensatz ist ein Template mit biometrischen Merkmalen der zu verifizierenden Person. Dieser Datensatz kann beispielsweise auf eine Chip-Karte gespeichert werden [7].

2.8 BioXes

BioXes ist ein Softwaresystem zur Verwaltung biometrischer Daten und Geräte, welches von der Universität Zürich und der Universität der Bundeswehr München(UniBwM) entwickelt wurde. BioXes wird für folgende Zwecke eingesetzt:

- Zur physischen bzw. logischen Zutrittskontrolle (z.B. Eintritt in ein Zimmer bzw. in die Gebäude).
- Zur Ansteuerung und Verwaltung unterschiedlicher biometrischen Geräte.
- Die Verwaltung von Templates und biometrischen Geräten.
- Die Verwaltung von biometrischen Daten.
- Die Verteilung, Updaten, Löschen von biometrischen Templates und Benutzerdaten.

Diverse biometrische Geräte werden mit einer seriellen Schnittstelle, USB oder über TCP/IP mit der BioXes Applikation verbunden. Diese biometrischen Geräte sind beispielsweise Fingerabdruck-Printer, Irisscanner oder Handfläche-Scanner. Die Abbildung 2.12 veranschaulicht den schematischen Aufbau von BioXes. In der obersten Schicht befinden sich der Administrator. In der untersten Schicht sind diverse biometrische Geräte, die zur Eintrittskontrolle dienen. Im Mittelpunkt steht BioXes, der die erwähnten Schichten verbindet [12].

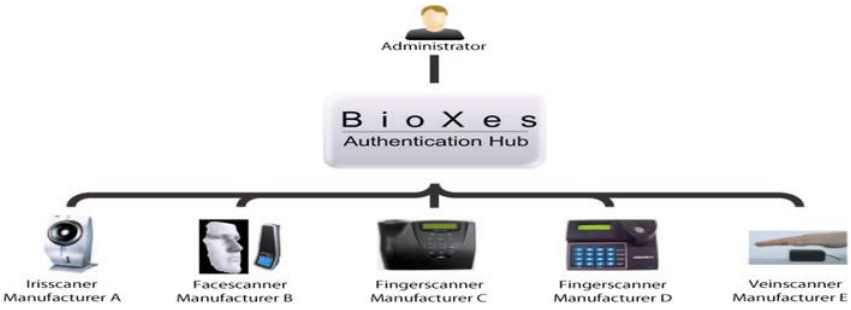

Abbildung 2.12: Schematische Aufbau von BioXes-Applikation [12]

2.8.1 BioXes Funktionalitäten

Benutzer von BioXes werden in zwei Gruppen klassifiziert:

- Endbenutzer, die mittels eines biometrischen Gerätes zu identifizieren sind. Ein Benutzer hat direkte Interaktionen mit den biometrischen Geräten.
- Administratoren, die direkte Interaktion mit der BioXes Applikation haben. Ein Administrator benutzt die Funktionalitäten, die von BioXes angeboten werden um die Benutzer- bzw. Geräte zu verwalten.

Die Funktionalitäten, die von BioXes angeboten werden, sind wie folgt:

- Verwaltung von Geräten (Devices). Der Administrator kann ein neues Gerät in das System hinzufügen, konfigurieren oder entfernen.
- Verwaltung von Benutzern. Der Administrator kann Benutzerdaten im System hinzufügen, editieren, löschen, aktivieren oder deaktivieren.
- Benutzergruppen. Der Administrator kann Benutzer bzw. Geräte eingruppieren, damit die Verwaltung von Benutzerdaten und Geräte schneller und einfacher werden.
- Reports generieren. Der Administrator kann die Systemereignisse in eine externen Datei exportieren.
- Job Verwaltungen. Die Job Verwaltungen bietet die Möglichkeit widerkehrende Aufgaben zu automatisieren.
- Zugriffsplan Verwaltungen. Der Zugriffsplan bietet Optionen zum Hinzufügen, Editieren, Löschen, Gewähren oder Ablehnen von Zugriffen zu bestimmten Zeiten.

2.8.2 BioXes Architektur

BioXes ist in Java geschrieben. Die Abbildung 2.13 zeigt die Architekturkomponenten von BioXes.

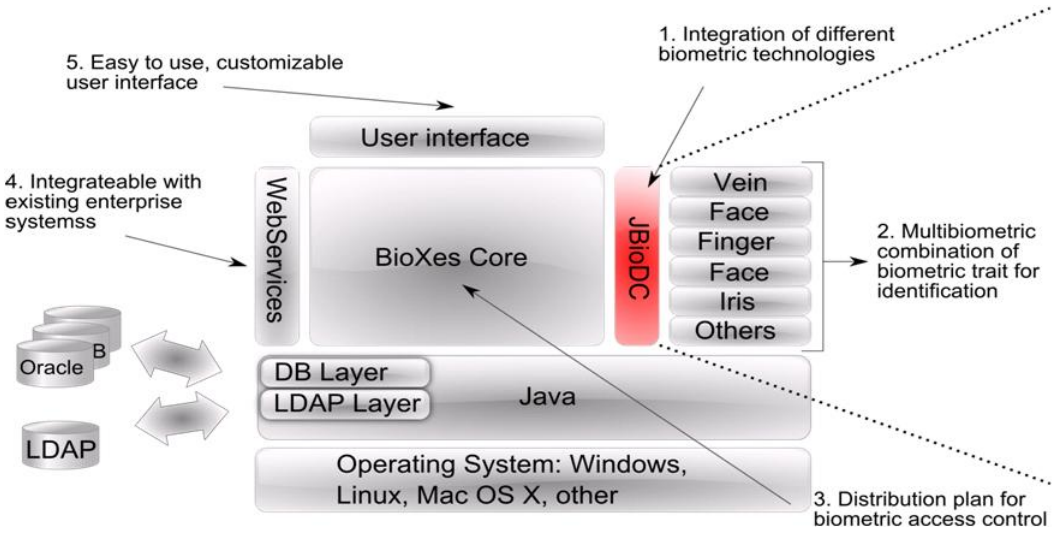

Abbildung 2.13: Architekturkomponenten von BioXes-Applikation [12]

BioXes folgt dem Architekturmuster Model-View-Controller (MVC). Das Model ist zuständig für Datenentitäten, wie Users, Devices und Zugriffe. Die View ist zuständig für die Interaktion mit dem Benutzer. Der Controller steht zwischen View und Model und definiert die Businesslogic von BioXes.

BioXes steuert verschiedene biometrische Geräte an. Die Geräte unterscheiden sich aufgrund ihrer Funktionalitäten oder ihrer Leistungen. Z.B. kann ein biometrisches Gerät einen internen Speicher für Benutzerdaten oder spezielle Tasten besitzen. Um diese Heterogenität der biometrischen Geräte zu bewältigen, wurde zwischen BioXes und den biometrischen Geräten einen Schicht entwickelt, der die Heterogenität für BioXes transparent macht. Diese Schicht heisst Java Biometric Direct Connectivity (JBioDC).

2.8.2.1 JBioDC

Java Biometric Direct Connectivity (JBioDC) ist eine high level API zur Verwaltung von biometrischen Geräten. JBioDC stellt generische Schnittstellen zur Verfügung. JBioDC bietet einige obligatorische Schnittstellen an. Diese Schnittstellen müssen zwingend für jede Art von biometrischen Geräten implementiert werden. Neben diesen obligatorischen Schnittstellen werden optionale Schnittstellen angeboten, die je nach Type und gewünschter Funktionalitäten zu implementieren sind. JBioDC beinhaltet Klassen, die bei der Implementierung von Schnittstellen verwendet werden können. Diese Klassen modellieren Fehlerbehandlungen(exceptions), Ereignisse(events) und Daten Strukturen(data containers) [12].

2.9 Vorschläge zur RFID-Erweiterung von BioXes

Zurzeit verwaltet und steuert BioXes nur biometrische Geräte. In Zukunft soll BioXes derart weiterentwickelt werden, sodass auch die Verwaltung und die Ansteuerung von RFID Geräten möglich ist. Im Folgenden werden einige Vorschläge zur RFID-Erweiterung von BioXes gegeben. Zu diesem Zweck werden die Unterschiede und Gemeinsamkeiten der Authentifizierung zwischen RFID und Biometrie untersucht.

2.9.1 Unterschiede zwischen RFID und Biometrie

- Das Datenformat der RFID Transponder unterscheidet sich von dem Datenformat der biometrischen Systeme. Bei RFID Transpondern handelt es sich in der Regel um Seriennummer, hingegen bei der Biometrie in der Regel um geometrische Daten.
- Störungen: Daten beim RFID System werden mittels elektromagnetischer Felder ausgetauscht. In dem Feld können spezielle Materialien zur Signalverstärkung (z.B. Metall) bzw. Signalschwächung (Flüssigkeiten) führen.
- RFID Authentifizierung ist eine Art von Identifikation(1:n Vergleich zw. ID auf der Chipkarte und IDs auf DB), hingegen in der biometrischen Authentifizierung sind Identifikation und Verifikation (1:1 Vergleich) möglich.

2.9.2 Analogien zwischen RFID und Biometrie

Die Analogie zwischen RFID- und biometrischen Systemen ist, dass die RFID Lese/Schreibgeräte wie biometrischen Geräten heterogen sind. Die Heterogenität der RFID Geräte bezieht sich auf folgende Punkte:

- Verwendet verschieden Frequenzstufen (lf/hf).
- Interaktionen mit verschiedenen Geschwindigkeiten.
- Unterschützung von verschieden Kartenformaten.
- Bieten unterschiedliche Funktionalitäten (Buttons).
- Setzen unterschiedliche Kommunikationsprotokolle (serielle oder TCP/IP Protokollen) ein.

2.9.3 Vorschläge und Schlussfolgerungen zur Erweiterung von BioXes

Eine Erweiterung der BioXes Applikation betrifft die drei Schichten des MVC. Auf alle diese Schichten müssen neue Funktionalitäten implementiert werden. Zum Beispiel sind

zusätzliche Komponenten in der View erforderlich, die spezifisch sind für RFID, vie z.B. die Geräteverwaltungen (Device Management) wie auch Benutzerverwaltungen (User Management). Neue Klassen, Methoden und Attribute im Controller und neue Entitäten im Model sind ebenfalls erforderlich.

Eine zusätzliche Schicht zwischen RFID und der BioXes-Applikation innerhalb der JBioDC ermöglicht die Verwendung von RFID in BioXes:

1. Die RFID Lesegerät Schnittstellen Komponente (ConnectionRFID) ermöglicht die Anbindung von heterogenen RFID Lesegeräten in BioXes. Diese Komponente bietet die Schnittstellen saveData(id, data), getData(id), setData(id, data) und removeData(id) an.
2. RFID Ereignisverwaltung Komponente (Event Management Component) ermöglicht das Reagieren auf Ereignisse von RFID Lesegeräten. Zusätzlich müssen folgende Aspekten betrachtet werden:
    - Pufferung (Speicherung der gelesen Daten)
    - Filterung (Kriterien zur Auswahl der Tag Daten)

Literaturverzeichnis

[1] Heise Online: *Schwächen des RFID-Systems Mifare Classic bestätigt*, Heise Zeitschriften Verlag, 19 März 2008. http://www.heise.de/security/Schwaechen-des-RFID-Systems-Mifare-Classic-bestaetigt--/news/meldung/105315

[2] Universität Ulm Abt. Pädagogische Psychologie: *RFID im Alltag, Reisepass*, Wettbewerbsbeitrag zum FOCUS Schülerwettbewerb,1 Mai 2006. http://www.schulemachtzukunft2006-068.de/projekterg_alltag_pass.html

[3] Matthias Lampe, Christian Flörkemeier,Institut für Pervasive Computing, ETH Zürich, Stephan Haller,SAP Research, Karlsruhe, SAP AG: *Einführung in die RFID-Technologie*,research document. http://www.vs.inf.ethz.ch/res/papers/mlampe-rfid-2005.pdf

[4] Prof. Peter Straub: *iFridge - Der intelligente Kühlschrank*, Elektro- und Kommunikationstechnik. http://www.hti.bfh.ch/fileadmin/img/HTI/Diplomarbeiten06/book_074.pdf

[5] Prof. Dr. Rolf Oppliger: *Vorlesung 491, Sicherheit in der Informationstechnik*, Vorlesungsfolien, Februar 2008. http://www.rfid-journal.de/rfid-energieversorgung.html

[6] Dr. Andrea Huber Geschäftsführerin Informationsforum RFID: *Basis Wissen RFID*, Informationsforum. http://www.info-rfid.de/downloads/basiswissen_rfid.pdf

[7] Michael Behrens,Richard Roth(Hrsg.): *Biometrische Identifikation*, EBook. http://books.google.com/books?hl=en&lr=&id=KnWEypCRqbgC&oi=fnd&pg=PA8&dq=Michael+Behrens+und+Richard+Roth&ots=KLM-LirU4k&sig=VKox3OMEH0-MAGyZDTswQJJD_Gg#PPP1,M1

[8] Wikipedia: *Legic*, Freie Enzyklopedia. http://de.wikipedia.org/wiki/Legic

[9] Wikipedia: *Mifare*, Freie Enzyklopedia. http://de.wikipedia.org/wiki/Mifare

[10] Alphacard: *RFID Cards- Contactless Smart Cards*, EBook. http://www.alphacard.com/id-cards/rfid-cards.shtml

[11] Veröffentlicht von AIM, Inc.125 Warrendale-Bayne Road Suite 100 Warrendale, PA 15086: *Radiofrequenz Identifikation Grundlage* , Document Type: AIM Inc. White Paper, Document Version: 1.3, 2004-04-15.http://www.kompetenzzentrum-autoid.de/contents/pdfs/AIM_RFIDprim.pdf

[12] Universität Zuerich,Institut fuer Informatik, Communication Group System: *BioXes Documentations*,April 2008. http://www.csg.uzh.ch/staff/bocek/publications/

[13] Communication System Group, IFI, UZH: *Biometric Local Area Network Control Center (BioLANCC)* http://www.csg.uzh.ch/research/biolancc/

[14] Wikipedia: *Radio Frequency Identification*,Freie Enzyklopedia. http://de.wikipedia.org/wiki/RFID

[15] Mirco Sander, Karsten Stieler: *RFID Geschäftsprozesse mit Funktechnologie unterstützen*,EBook. http://www.hessen-media.de/mm/rfid.pdf

[16] *RFID, das kontaktlose Informationssystem*, Informationen zur RFID-Technik des BMWi, Stand November 2007. http://www.bmwi.de/BMWi/Redaktion/PDF/P-R/rfid,property=pdf,bereich=bmwi,sprache=de,rwb=true.pdf

[17] ARCHI EXPO: *RFID reader for access control SIGNATURE*, Katalog. http://www.archiexpo.com/prod/vingcard/rfid-reader-for-access-control-10481-34254.html

[18] ARCHI Expo: *RFID reader for access control SIGNATURE*, workshop. http://de.wikipedia.org/wiki/Fu%C3%9F_%28Einheit%29

[19] EDÖB: *Anwesendheit Kontrolle mit Fingerandruecke*, online hompage von schweizerischer Eidgenossenschaft. http://www.edoeb.admin.ch/dokumentation/00445/00509/00510/00544/index.html?lang=de

[20] OPAL Associates Holding AG: *Intermec IP4 mobiler UHF RFID Leser*, Servive. http://www.intermec.ch/RFID/ip4.htm

[21] OPAL Associates Holding AG: *RFID readers,RFID writers*, Servive. http://www.directindustry.com/cat/storage-packaging-marking/rfid-readers-rfid-writers-U-663.html

[22] Professionals in Mobile Solusions and Automatic Identification: *Stationäre RFID - Reader und Antennen*,Katalog. http://at.rodata.ch/web/cms/system/galleries/download/produkte/Rodata_Produktexbersicht_RFID_Stat._RFID_-_Reader_und_Antennen_V0705.pdf

[23] RFID ready: *Funktionsweise der RFID Technologie*,rfid ready - Das Informationsportal und Branchenbuch für RFID-Technologie liefert aktuelle RFID-News, Anwendungs- und Praxisbeispiele zum Thema RFID. http://www.rfid-ready.de/65-0-wie-funktioniert-rfid.html

Kapitel 3

Mobile Agents and Security

Besa Canolli-Hasanmetaj

Mobile Agenten sind wichtige Programme, die sich innerhalb eines Netzwerkes migrieren. Sie haben viele mögliche Anwendungsgebiete, bei denen verschiedene Vorteile, aber auch Risiken zu sehen sind. Daher betrachtet man näher den Aspekt der Sicherheit als ein wichtiger Faktor bei der Einsatz der Mobile Agenten. Ein zukünftiger Anwendungsgebiet der Mobile Agenten ist der elektronische Markt, wobei deren Grundlagen von der Mobile Agenten Systeme gebildet werden. Eine Motivation für den Einsatz mobiler Agenten ist die Automatisierung der Handlungen, die bisher durch Menschen durchgeführt wurden [3], als Beispiel dazu sind die Mobile Agenten zum Preisvergleich.

Inhaltsverzeichnis

3.1 Einleitung und Motivation

Mobile Agenten sind Programme, die sich innerhalb eines Netzwerkes autonom von einer Umgebung zu einer anderen Umgebung bewegen (migrieren) und alle dabei gesammelten Daten mitführen [9]. Mobile Agenten haben die Fähigkeit mit vielen anderen Rechner innerhalb eines Netzwerks zu kommunizieren. Sie können mobilen Agenten, Agentensysteme oder anderen Partnern sein.

Es werden mögliche Anwendungsgebiete für mobile Agenten vor allem in den Bereichen Elektronische Märkte, Netzwerkmanagement, Informationsbeschaffung, mobile Endgeräte, Unterstützung von Gruppenarbeit, Personalisierte Dienste und Fernwartung beschrieben. Der Anwendungsbereich scheint aber unbegrenzt zu sein.

Es werden für mobile Agenten eine ganze Reihe von möglichen Vorteilen geltend gemacht [3]. Mobile Agenten als eine schnell entwickelte neue Technologie [2] bringt neben eine ganze Reihe von Vorteilen auch neue Nachteile und Risiken mit.

In diesem zunehmend stark vernetzten Welt, um diese neue Technologie überhaupt zu akzeptieren, spielt der Aspekt der Sicherheit eine immer wichtigere Rolle. Für den Schutz des ausführenden Rechners werden einige Sicherheitselemente betrachtet [3].

Diese Arbeit beschäftigt sich im Besonderen mit der Frage unter welchen Bedingungen ein mobiler Agent vor Angriffen geschützt wird. Somit werden im Kapitel 3.3.3 einige mögliche Angriffen von Mobile Agenten untersucht und im Kapitel 3.4 werden dann verschiedene Lösungsansätze beschrieben, die zur Beantwortung dieser Frage dienen. Ausser andere Teilschutzansätze wird auch der Blackbox als einer wichtiger Ansatz hingedeutet, deswegen wird er auch separat im Kapitel 3.4.3 behandelt. Er reduziert das Gesamtproblem des Schutzes mobiler Agenten auf ein kleineres Problem, um mobile Agenten vor allen Angriffen zu schützen, indem die Blackbox-Eigenschaft mobiler Agenten angenommen wird [3].

Nachdem im Kapitel 3.2.3 eine Menge von Anwendungsgebieten erläutert wurden, wird im Kapitel 3.5.1 näher der elektronische Markt als zukünftiger Anwendungsgebiet der Mobile Agenten beschrieben. Wobei deren Grundlagen von der Mobile Agenten Systeme gebildet werden. Eine Motivation für den Einsatz mobiler Agenten ist die Automatisierung der Handlungen,die bisher durch Menschen durchgeführt wurden [3]. Mobile Agenten zum Preisvergleich wird im Kapitel 3.5.2 in einem wirtschaftlichen Beispiel behandelt.

3.2 Konzept Mobile Agenten

In diesem Kapitel wird der Mobile Agent durch seine Eigenschaften definiert. Dank seiner Fähigkeiten ist Mobile Agent an einen breiten Anwendungsgebiet zu finden. Trotz seiner vielen Vorteilen, bringt der Mobile Agent auch Nachteile mit sich, die sehr wichtig sein könnten, wie z.B. die Sicherheitsproblematik.

3.2.1 Definition

„Mobile Agenten sind Programminstanzen, die in der Lage sind, sich selbstständig zwischen verschiedenen, eventuell fremden, Ausführungsumgebungen zu bewegen und, unter Ausnutzung lokaler Ressourcen, Aufgaben zu erfüllen“ [3]. Wenn man ein Mobile Agent als eine Programminstanz betrachtet, dann lässt er sich in veränderliche (variable) Daten und statische (konstante) Daten unterteilen. Ein Agent kann diese variablen Daten zur Laufzeit erweitern oder verringern. Das ausführbare Agentenprogramm gehört zum statischen Daten [3] [1].

3.2.1.1 Eigenschaften

Die mobile Agenten haben wichtige *Eigenschaften*. Hier sind deren Beschreibungen im Kürze. Gemäss [18] ist ein Mobil Agent:

- *Autonom* Der Mobile Agent entscheidet selbstständig über seine nächste Aktion und arbeitet unabhängig von der kontrollierenden Instanzen.
- *Mobil* Anhand dieser Fähigkeit, können Mobile Agenten ihr Programm zusammen mit Zusatzinformationen und weitere Daten von einem Agentensystem über ein Netzwerk zu einem anderen Agentensystem transportieren [7].
- *Reaktiv* Der Mobile Agent reagiert entsprechend auf Änderungen seiner Umgebung, dabei benützt er seine Sensoren. Dazu besitzt er auch Regeln, womit er eine bestimmte Aktion auslösen kann.
- *Proaktiv* Mobile Agenten besitzen einen internen Zustand und sind fähig, zielgerichtet zu planen und zu handeln. Sie ergreifen die Initiative.
- *Kommunikativ* Der Agent kommuniziert mit anderen Agenten, Agenten-Systemen und anderen Partnern im Netzwerk. Als Vorraussetzung für eine Kommunikation sind gemeinsame Protokolle, Ontologien, Interaktionsvereinbarungen sowie Sprachen (z.B. Java, TCL). Mehr darüber gibt es bei Mandry [5].

Eine ausführliche Beschreibung der verschiedene Agentenfähigkeiten findet man z.B. in [9].

3.2.2 Migrationsprozess

Beim Prinzip der Migration [8] kann sich die Bearbeitung einem Mobilen Agent unterbrechen, und dann nach dem Transaktion auf einen neuen Wirtsystem wieder fortsetzten. Somit folgen Mobile Agenten auch keine ausgezeichnete Client / Server Architektur. Weil, nachdem der Client auf seinem Agentenserver den mobilen Agenten gestartet hat, entscheidet dann er selbst, welchen Service und welche Ressource er von welchem Ort benötigt. Eine Migration findet immer genau zwischen zwei Orten statt [3]. Damit das

volle Potential der Agenten ausgenutzt wird, müssen die Agenten auf vielen verschiedenen Plattformen und in vielen Zielsystemen ausführbar sein.

Für die einzelne Schritte, die von einem Agentensystem durchgeführt werden, wird hier eine weiterführende Literatur der [3] empfohlen. Einen Überblick der Migrationsablauf ergibt sich hier die Abbildung 3.1, basierend auf [18].

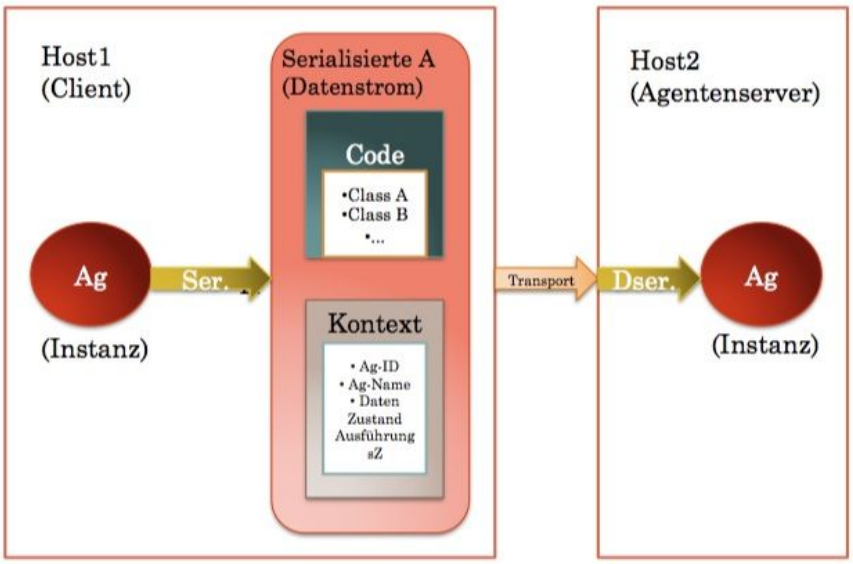

Abbildung 3.1: Migrationssprozess

Aus Sicht der Bearbeitungszustand der Mobilen Agenten unterscheiden sich die Migration [8] nach:

Schwache Migration Der Agent migriert, während seiner Ausführung, auf einem neuen Wirtsystem (Zielsystem). Dabei wird, neben dem Programmcode, auch der Status der Daten mit übertragen. Das Programm selber wird auf dem neuen Wirtsystem neu gestartet. Ein Programmstatus wird nicht mit übertragen, aber er hat die Möglichkeit, im Vergleich mit den transparenten Migration, einfacher zu reprodutieren. Er muss vom Programmierer explizit in Form von Daten festgehalten werden. Hier unterscheidet man Migrationen mit festem Einstiegspunkt (z.B. Aglets) und die mit beliebigem Einstiegspunkt (z.B. Voyager) [18].

Starke Migration bezeichnet man als eine transparente Migration, weil hier neben dem Programmcode auch Statusinformationen eines Agenten zum Programmablauf an eine neue Ausführungsumgebung transportiert werden. Dies erfordert zum einen ein globales Agentenstatus-Modell und zum anderen eine Syntax, die es erlaubt diese Statusinformationen zu übertragen. Da der Status sehr grosse Datenmengen umfassen kann, ist die Starke Migration eine sehr teure und zeitaufwendige Operation [18].

Ausser schwache und starke Migration gibt es noch andere Migrationsarten, wie Remote Execution, Remote Evaluation und Code on Demand. Die nach individuelle Interesse bei [9] zu finden sind.

Zu den Mobile Agenten allgemein, wird eine entsprechende Infrastruktur benötigt, bestehend aus Agentenservern, die Agenten empfangen, starten, anhalten und wieder versenden können [7].

3.2.3 Anwendungsgebiete

Elektronische Märkte Mobile Agenten haben hier eine breite Auftrittsmöglichkeit, vom Anbieter, Käufer, Verkäufer und bis zur Vermittler von Waren und Dienstleistungen. Sie bewegen sich im Internet dann von einem Ort zum Anderen. Man kann es ein Mobile Agent auch so vorstellen, dass er im Auftrag eines Nutzers eine Einkaufstour durchführt, sich über Angebote informiert und auf seiner Rundreise das billigste Angebot ermittelt [13]. Eine ausführliche Erklärung hierfür findet sich hier im Kapitel 3.5.1.

Netzwerkmanagement Mobile Agenten sind generell als aktive Elemente in diesem Bereich einsetzbar. Sie übernehmen hier insbesondere Aufgaben wie z.B. Monitoring, wo sich die mobilen Agenten von Netzknoten zu Netzknoten bewegen, um die lokalen Daten einzusammeln und auszuwerten.

Informationsbeschaffung Mit dem Profil des Auftraggebers werden hier die Mobile Agenten versehen und suchen auf entsprechende Datenbankserver nach relevanten Informationen. Dabei kommunizieren sie auch mit Informationsvermittlern und tauschen gegebenenfalls Informationen mit anderen mobile Suchagenten, die ein ähnliches Profil besitzen, aus [13].

Mobile Endgeräte Mobile Endgeräte müssen während der Ortsveränderung die Aufrechterhaltung der Kommunikation sicherstellen. Da sie in ihrem PDA (Personal Digital Assistant) nicht über die nötigen Ressourcen verfügen, schickt der mobile Endgerät einen Mobilen Agenten an einen leistungsfähigen Rechner, auf dem er einen Arbeitsauftrag erledigt und zusammen mit dem Ergebnis auf den PDA zurückkehrt.

Unterstützung von Gruppenarbeit Durch Einsatz der Mobile Agenten als aktive Dokumente können die zur Bearbeitung notwendige Funktionalität zur Bearbeitungsstelle mitbringen. Damit können auch die Terminvereinbarungen und Projektplanungen gut unterstützt werden [13].

Personalisierte Dienste Hier kann ein mobiler Agent von einem Dienstanbieter zum Kunden gesendet werden, um dort vor Ort einen generischen Dienst in spezieller Weise zu erweitern. Beispiele hierfür wären etwa Funktionserweiterungen bei Telekommunikationsgeräten oder die Ergänzung von Softwarekomponenten.

Fernwartung Agenten können vor Ort Systemkomponenten überwachen und gegebenenfalls auch die Diagnose von Fehlverhalten durchführen sowie Reparaturmaßnahmen treffen. Ein Beispiel wäre die Monitoring, die die Netze managiert.

3.2.3.1 Vorteile

Bei der der Einsatz von mobiler Agenten gibt es eine Reihe von Vorteilen [3], die hier beschrieben werden.

Höhere Fehlertoleranz Durch die endsystemnahe Platzierung und die Eigenschaft der Autonomie des Mobilen Agenten ergibt sich auch eine höhere Fehlertoleranz. Auch wenn ein Ausfall der Netzverbindung zwischen Manager und Mobilem Agenten gibt, kann dieser autonom weiterarbeitet und dann seine Daten zu einem späteren Zeitpunkt an den Manager übermitteln.

Asynchrones Arbeiten Ein mobiler Agent benötigt bei der speziellen Arbeitsauftrag keine dauerhafte Kommunikationsverbindung. Der Rechner, der einen mobilen Agenten erzeugt, muss nur für die Dauer der Migration und zum Entgegennehmen der Ergebnisse (z.B. durch eine Rückmigration) mit einem anderen Rechner verbunden bleiben [3]. Somit ergibt sich tiefere Verbindungskosten.

Reduzierte Netzwerkbelastung Bei Mobile Agenten besteht die Möglichkeit, die Ergebnisse noch vor der Übertragung zu evaluieren, einzuschränken oder umzuformen. Dafür muss die Funktionalität des Servers nicht verändert werden. Bei diesem Ansatz werden nur noch die eingeschränkte Ergebnismenge und der Agent selbst übertragen. Dadurch ergibt sich ein erhebliches Einsparpotenzial im Umfang der zu übertragenden Daten.

Dynamische Anpassung Die Eigenschaften „reaktiv“ und „autonom“ von den Mobile Agenten kommen hier zum Einsatz, wobei nach [8] können die Mobile Agenten ihre Ausführungsumgebung erkennen und autonom auf Veränderungen im Wirtsysteme reagieren. Das heisst, die Mobile Agenten erlauben es als Mobile-Code-Einheiten, Funktionalität, sprich Code, dynamisch auf Rechner zu bringen, auf denen diese Funktionalität vorher nicht vorhanden war[3].

Es gibt mehrere potenzielle Vorteile Mobiler Agenten nach ([9] s. 18), wo, bei jeder auch eine kurze Begründung gegeben wird.

3.2.3.2 Nachteile

Der Einsatz mobiler Agenten hat viele Vorteile, dazu bergen sie aber durchaus auch Gefahren und Probleme.

Sicherheitsproblematik Wegen der hohe Flexibilität und Mobilität der Mobilen Agenten sind die Sicherheitsrisiken nicht zu unterschätzen. Weil dadurch sowohl das Rechnersystem des Auftraggeber als auch der Agent selbst betroffen sind [8].

Fehlertoleranz In einer offenen, vernetzten Welt ist es oft der Fall, dass es Fehler auftreten können. Mobile Agenten sollten ein derartiges Fehlverhalten nach Möglichkeit überleben, damit realisierte Anwendungen von diesen Fehlern möglichst nichts mitbekommen.

Keine direkte Kontrolle Ein mobiler Agent wird auf einem anderen Rechner (Zielsystem) ausgeführt, daher ist es ohne weitere Massnahmen schwierig, diese Mobile Agenten zu kontrollieren.

3.2.4 Zusammenfassung

Mobile Agenten wurden anhand einer kurzen Definition und ihrer Eigenschaften definiert. Die Migration selbst bezeichnet man als wichtiger Prozess der Mobile Agenten. Die möglichen Anwendungsgebiete wurden erläutert und abschliessend noch deren Vor- und Nachteile kurz diskutiert.

3.3 Sicherheitsprobleme

Eine der diskutablen Nachteil der Mobile Agenten Einsatz ist die Sicherheitsproblematik. Die als ungelöste Problem da steht. Hier wird beginnen, mit einer Beschreibung der Sicherheit in verteilte Systemen mit einigen Sicherheitsaspekten. Zunächst wird definiert, was ein sicheres System ist. Dann werden die Sicherheitsgefährdungen und einige Sicherheitsmechanismen unterschieden und einzeln betrachtet.

3.3.1 Begriff Sicherheit

Der Begriff Sicherheit impliziert aus sprachlicher Sicht eine gewisse Absolutheit, die normalerweise in den meisten Fällen nicht zu finden ist. Tatsächlich ist der Begriff der Sicherheit eine relative Größe, weil in der Technik allgemein und in der Computertechnik im speziellen, keine absolute Sicherheit geben kann. Daraus folgt, dass es hier die Aufgabe der Sicherheit vielmehr darin besteht, das Risiko bis auf einen gewissen Mass zu reduzieren, die vertretbar ist.

Wichtig dabei sind diese vier Sicherheitspunkte, die möglichst einzuhalten sind:

1. Verschlüsselung
2. Authentifizierung
3. Autorisierung
4. Auditing

3.3.2 Sicherheitsbereiche

Wie schon im Kapitel 3.2.1 erwähnt wurde, kann ein mobiler Agent mit unterschiedlichen Stellen kommunizieren. Dabei ergeben sich gewisse Anforderungen an die Sicherheit, die sich auch folglich eine zentrale Bedeutung haben. Um deren zu gewährleisten, je nach abgebildeten Kommunikationsebene in der Abbildung 3.2, gibt es unterschiedliche Möglichkeiten [5].

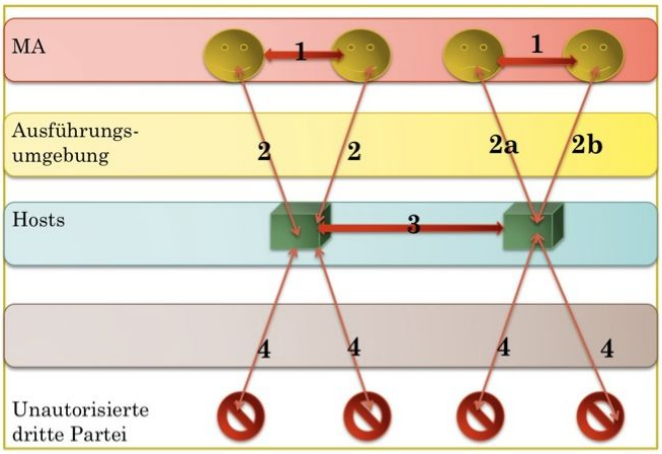

Abbildung 3.2: Sicherheitsbereiche von Mobile Agenten

In [5] werden diese vier Sicherheitsbereiche von Mobile Agenten etwas deutlicher unterschieden:

3.3.2.1 Sicherheit zwischen zwei Mobile Agenten

Bei dieser Kommunikation zwischen zwei Mobile Agenten (z.B. über RPC) migriert der einer Agent lokal auf das System der Anderer Agent. Diese Mobile Agenten sollen hier vor anderen böswiligen Agenten geschützt werden. Weil sie versuchen könnten, den Programmcode oder die Daten ihres Opfers zu lesen oder gar zu manipulieren. Solche Angriffe wären hier möglich [5]:

- direkte Zugriffe auf die Speicherbereiche des Agenten.
- Maskierung (Vortäuschen einer fremden Identität),
- Betrug allgemein (z.B. die Benutzung eines Dienstes, ohne ihn zu bezahlen),
- und Denial of Service (Verhindern der Ausführung).

Diese Angriffe können zu ihrer Bekämpfung auf die Standardverfahren aus diesem Kommunikationsbereich zurückgegriffen werden. In diesem Fall könnten sein [5], z.B.:

- Die Benutzung einer Programmiersprache, die den Privatzugriff auf Speicherbereiche,

- Isolierte Adressräume für die einzelnen Mobile Agenten,
- Authentifizierung der Agenten, z.B. anhand digitale Signaturen,
- Benutzung eines Service mit dem Vertragssystem.

3.3.2.2 Sicherheit zwischen Mobile Agent und Host

Dieser Bereich lässt sich weiter in zwei unterschiedliche Unterbereiche aufteilen:

1. **Sicherheit von Hosts gegenüber böswilligen Mobile Agenten** Diese Angriffe sind dabei dieselben, wie zwischen Agenten, die auch Viren sein können [16]. Nur, hier hat ein Wirtsystem den größeren Einfluss, weil er den Ablauf des Agenten kontrolliert. Er kann böswilligen Mobile Agenten sofort in ihrer Ausführung unterbrechen, und dann sich vor eventuellen Angriffen schützen. Außerdem hat er Schutzmöglichkeiten [5], wie z.B.:
   - Benutzung von sicheren Sprachen oder isolierten Adressräumen (z.B. bei Java Applets),
   - Authentisierung,
   - Benutzung von Kontosystemen,
   - Benutzung von Vertragssystemen,
   - Benutzung von Mechanismen, die die Ressourcen kontrollieren.
2. **Sicherheit von Mobile Agenten gegenüber betrügerischen Hosts** Dieser Sicherheitsbereich im Vergleich mit den Vorherigen, stellt ein völlig neues Problem dar. Weil, der Host ist hier für die Ausführung des Agenten zuständig und insofern kann er jeden Programmschritt beobachten. In dieser Bereich gibt es sowohl Hardware als auch Software Lösungsansätze. Die in [5] Kapitel 3.5 ausführlich besprochen wird.

3.3.2.3 Sicherheit zwischen zwei Hosts

Das ist eine Sicherheitsbereich, die speziell den Schutz der Kommunikation zwischen zwei Wirts über ein unsicheres Netzwerk (z.B.: das Internet) beschreibt. Deren möglichen Angriffe können sein:

1. **passive Angriffe**, sind gewisse Abhörvarianten, die die Vertraulichkeit der Kommunikation bedrohen können, und
2. **aktive Angriffe**, bei denen das Ziel ist, aktiv in die Kommunikation einzugreifen und so die Daten und User zu manipulieren.

3.3.2.4 Sicherheit zwischen Host und einer unautorisierten Person

Es bezieht sich auf den Zugriff auf Systembereiche des Hosts. Diese Verfahren kommen besonders hier zu Einsatz:

1. Authentifizierung (z.B.: Passwort-Verfahren) und
2. Autorisierung (z.B.: Zugriffsrechte-Verfahren).

Nach [3] wird der Schutz Mobiler Agenten vor böswilligen Wirten als Voraussetzung für einen Einsatz Mobiler Agenten in offenen Systemen genannt und somit als wichtig bezeichnet. Das ist auch der eine Grund, warum in den kommenden Abschnitt mit dieser Sicherheitsbereich (Sicherheit von Mobile Agenten gegenüber betrügerischen Wirt) näher um Vertiefung eigegangen wird. Das andere Grund, warum die anderen Sicherheitsbereiche (hier im Kapitel 3.3.2) nicht weiter diskutiert werden ist, dass es in dieser Arbeit hauptsachlich um die Mobile Agenten und deren Sicherheit geht.

3.3.3 Mögliche Angriffe

Hier sind einzelne Angriffe beschrieben, die ein Wirt auf einen Mobilen Agenten ausüben kann. Sie lassen sich in drei Hauptkategorien der Grundbedrohungen einteilen [5].

3.3.3.1 Leseangriffe

Die Leseangriffe richten sich gegen die Vertraulichkeit des Programmcodes, der Daten und des Kontrollflusses. Die Vertraulichkeit auf den Schutz vor dem jeweils ausführenden Server ist hier von Bedeutung, die beim Übertragen des Mobile Agenten zwischen zwei Agentenservern durch eine normale asymmetrische Verschlüsselung erreicht werden kann [5].

1. **Auslesen von Programmcode** Für eine korrekte Ausführung der Mobile Agenten auf den Wirt, muss der Programmcode generell für den Wirt lesbar sein. Es gibt dabei zwei Wissensunterschiede:

   - was das Programm als ganzes tut, und
   - was jede Zeile des Programms bewirkt (z.B.: der Interpreter, um das Programm auszuführen.)

   Es besteht die theoretische Möglichkeit, den Wirt nur auf die nächste Programmzeile zuzugreifen. Wegen den grossen Teile der ausgeführten Codes ist diese Lösung, keine generelle Lösung des Problems, da einige Hosts trotzdem den gesamten Code sehen können [5].

2. **Auslesen von Daten** Wenn ein Mobile Agent beim Auslesen von Daten keine Spuren hinterlässt, kann es sehr kritisch sein. Weil es kann später dazu kommen, dass sowohl der Agent als auch sein Besitzer von dem ausgeführten Angriff nichts merken. Wenn es besonders um die geheime Daten (wie z.B.: Schlüssel) oder elektronisches Geld handelt. Dieser Art von Datenverlust kann sich beheben, nur mit erhebliche Konsequenzen für den Besitzer [5].

3. **Auslesen des Kontrollflusses** Selbst der Wirt entscheidet hier für den nächsten Programmschritt, sobald er den Programmcode und die Daten eines Agenten gelesen hat. Es ist möglich, die benutzen Daten vor dem Wirt zu verbergen, aber nicht den eigentlichen Kontrollfluss des Mobile Agenten zu schützen. Alle diese Informationen erlauben es einem Wirt, Informationen über den Ausführungsstatus des Agenten herzuleiten. Wodurch er ohne Inhaltskenntnisse der einzelnen Variablen ermitteln kann, ob ein Angebot besser oder schlechter als das gespeicherte Angebot des Mobile Agenten ist [5].

3.3.3.2 Manipulationsangriffe

Die folgenden drei Angriffe betreffen die Integrität der drei Sicherheitsbereiche, wobei der Integrität bei der Übertragung durch digitale Signaturen erzielt werden kann. Der ausführende Server wird hier als Angreifer betrachtet.

1. **Manipulation von Programmcode** Der Wirt kann den Programmcode eines Mobile Agenten lesen und kann auch Zugang zu deren Speicherbereichen haben. Insofern besteht die Möglichkeit, dass er den Code auch ändert und zwar in zwei unterschiedliche Formen:

   - **permanent** oder
   - **temporär**.

2. **Manipulation von Daten** Es ist wichtig, dass der Wirt Kenntnisse über den physischen Speicherplatz der Daten und deren Semantik, weil kann er Änderungen vornehmen. Als Beispiel könnte er zuerst seinen Preis als besten eintragen, dann die nachfolgenden Wirts aus der Liste der zu besuchenden Wirts streichen. Das könnte sein, dass der Mobile Agent die gesuchte Anfrage bei diesem Host erledigt, und keine anderen mehr besucht [5].

3. **Manipulation des Kontrollflusses** Wenn der Wirt auch nicht Zugangsberechtigt zu den Daten des Mobile Agenten ist, kann er immer noch den Kontrollfluss des Agenten manipulieren. Als Beispiel kann er falsche Interpretation bei der Bedingungen in Conditional Statements geben, oder ganze Programmteile (bspw. Sicherheitsüberprüfungen) überspringen. Dazu haben die Absicherungen keinen Effekt. Ein kritischer Punkt dabei ist, dass eine Veränderung des Kontrollflusses nicht über Checksummen geprüft werden kann. Eine Schutzmöglichkeit dazu wäre, indem die Relation zwischen Kontrollfluss und Ausführungssemantik vor dem Wirt verborgen wird [5].

3.3.3.3 Verfügbarkeit des gesamten Mobile Agenten

Das ist der letzte Angriff, die lediglich die absichtliche Zerstörung durch den ausführenden Wirt betrachtet [5].

1. **Denial of Service Angriff** Das ist eine Möglichkeit für den Wirt, die Ausführung des Mobile Agenten in zwei Arten zu beeinflussen. Einerseits kann er ihn einfach nicht ausführen, oder durch Zerstörung die Ausführung verhindern. Hier entscheidet der Wirt frei darüber. Deswegen ist es auch schwer ihn abzusichern. Nun, könnte man den Wirt durch organisatorische Massnahmen (z.B. durch Protokolle) dazu zwingen, für eine korrekte Ausführung und dadurch kann man den angreifenden Wirt identifizieren und als „betrügerisch“ markieren. Was ergibt sich, dass solche Wirts nicht mehr von einem Mobile Agenten aufgesucht werden.

Für eine ausführliche Erklärung zu den allen Angriffen siehe [5].

3.3.4 Zusammenfassung

In diesem Kapitel wurde die Sicherheitsproblematik der Mobile Agenten diskutiert. Es wurde mit einer Beschreibung der Sicherheitsbegriff begonnen. Dann wurden die Sicherheitsgefährdungen und einige Sicherheitsmechanismen unterschieden und einzeln betrachtet.

3.4 Lösungsansätze

Wenn man den Mobile Agent mit einem Programm und den Wirtssystem mit einem Interpreter vergleicht, dann folgt, dass es im Falle deren Absicherung sehr ähnlich ist. Weil, der Wirt hat hier Zugang zu den Programm- und Datenbereichen des Mobile Agenten und zu seinem Ausführungsstatus (durch Interpretation seiner Programmcode). Daher ist es nicht einfach dieses Problem zu lösen [5].

In diesem Kapitel werden zunächst verschiedene Lösungsansätze zu den im Kapitel 3.3 genannten Sicherheitsprobleme, beschrieben. Danach wurden einige Teillösungsansätze anhand der Vertraulichkeit, Integrität und Verfügbarkeit verglichen. Anschliessend ist der Gesamtschutzansatz, der Blackbox-Ansatz etwas näher beschrieben.

3.4.1 Verschiedene Ansätze

Für eine Sicherheit von Mobile Agenten gegenüber betrügerischen Wirts sind verschiedene Ansätze nach [5] entwickelt worden. Diese Ansätze sind mit Overhead verbunden und dementsprechend mit eine Frage des Kosten- / Nutzen- Verhältnisses, ob sie sich bei deren Einsatz lohnen. Generell können diese in zwei Gruppen eingeteilt werden:

1. **Vorbeugende**, indem die Mobile Agenten vor Angriffen geschützt werden.
2. **Erkennende**, die lediglich im nachhinein einen Angriff erkennt.

In den folgenden Abschnitt werden einige Ansätze kurz vorgestellt. Deren Ausgang ist im Bezug auf das Vertrauen. Ein Benutzer hat kein Vertrauen in die Server, die von seinem Mobile Agenten besucht wird.

3.4.1.1 Kein Schutz

Hier geht es um keine Benutzung von Schutzmechanismen. Man lässt das Risiko einfach eingehen. Es wird die Kosten (des Schutzes) - Nutzen (der Sicherheit) Funktion betrachtet und dementsprechend verhalten.

Als Beispiel hierfür sind gewisse Daten, die keinen grossen Wert besitzen, besonders nicht für einen Schutz. Der grösste Vorteil besteht bei ungeschützten Agentensystemen darin, dass kein zusätzlicher Aufwand für die Berechnungen allgemein benötigt wird. Es besteht noch eine Möglichkeit für den privaten kleinen Benutzer, wobei ein ungeschützter Mobile Agent die effizientere Alternative ist [5].

3.4.1.2 Gesetze / Verträge

Dieser Schutzansatz ist auch einfach. Hier geht es um die Betreiber der Wirts per Gesetz oder durch entsprechende Verträge zu versichern. Das Ziel ist hier den Server vor externen Angreifern zu schützen, in dem man versucht die Vertraulichkeit und Integrität der Mobile Agenten nicht zu verletzen. Bei dieser Methode benötigt man keine kryptographischen Protokolle und es entstehen auch kein Laufzeit-Overhead, was zur Vorteil führen. Das Problem liegt aber bei der Beweis im Falle einer Verletzung des Vertrages oder des Gesetzes. Dazu braucht man ein zusätzliches erkennendes Verfahren, wo man Manipulationsnachweise aufzeichnen kann und als Beweise gelten. Beispielsweisse könnte man ein vertraglicher Schutz festgelegen, was zu einer Beschränkung der Marktteilnehmer entspricht und gleich zu eine Verletzung der gewünschten Marktoffenheit führt [5].

3.4.1.3 Trust

Bei dieser Ansatz geht es darum, den kritischen Prozess der Berechnung aller Ergebnisse auf einen vertrauten Server auszulagern. Es sind dann nur die Daten gegen Angriffen zu sichern, die der Mobile Agent aus den unbekannten Servern mitgebracht hat. Dabei wird dann deren Integrität vor der Ausführung der Berechnung auf dem vertrauenswürdigen Server geprüft. Alle Daten (wie private Schlüssel und elektronisches Geld) können bei dieser Verfahren für die kritischen Prozesse benutzt werden. Das ist auch ein wichtiger Vorteil dieses Verfahrens. Eine Frage der Vertraulichkeit stell sich hier, indem der Benutzer dem Server seine privaten Schlüssel übergibt [5].

3.4.1.4 Reputation

Hiermit wird eine bisherige Verhalten der Mobile Agenten (Teilnehmern) auf einem Markt verstanden. Aufgrund dessen Verhalten besitzen die Marktteilnehmern einen solchen bestimmten Ruf, auf dem man für (nach einer korrekten Ausführung) oder gegen (betrügerische Server) eine Weiterempfehlung basieren kann. Es kann auch sein, dass die betrügerische Server einen falschen Ruf bekommen. Theoretisch ist es möglich, dass ein Server über eine gewisse Zeit einen guten Ruf aufbaut, um diesen dann später zum Betrug einsetzt. Ein erstes Vorteil des Reputation ist, dass nicht so viel Aufwand von den einzelnen Servern gefordert wird. Das Zweite ist die direkte Eliminieren den betrügerische Server, aufgrund eines korrekt funktionierenden Systems [5]. Ein Beispiel dazu ist, wenn man die Einwohner eines kleinen Dorfes und deren einer grossen Stadt vergleicht. In dem Kleindorf kennt jeder jeden und ihrem Verhalten. Wenn es um eine Wahl-Kampagne geht, dann wird der Richtige gewählt.

Detection Objects bietet auf der Basis von Detection Objects die Möglichkeit, betrügerische Modifikationen der Daten des Mobile Agenten zu erkennen.

Tracing ist ein Protokoll, bei dem durch Statusmeldungen und Ausführungsprotokolle eines Mobile Agenten jede unrechtmässige Manipulation identifiziert und einem bestimmten Server zugeordnet werden kann.

Fault Tolerance ist eine Art des Schutzes gegen Angreifer, der benutzt wird, um daraus ein relativ einfaches Sicherungsprotokoll gegen betrügerische Wirts abzuleiten.

Code Mess Up Im Normalfall braucht jeder Angreifer eine gewisse Zeit, um die Daten und den Programmcode zu lesen und zu verstehen, bevor er eine gezielte Manipulation des Mobile Agenten vornimmt. Dieser Code Mess Up Verfahren versucht, diese Zeitspanne zu maximieren, indem er inzwischen den Programmcode so umwandelt wird, dass er für einen Angreifer dannextrem schwer zu analysieren ist.

3.4.2 Vergleich der Ansätze

Alle die scon vorgestellten Ansätze gehen davon aus, dass der Mobile Agent direkt nach seiner Ankunft auf dem Agentenserver von diesem ausgeführt wird. Ein Server hat aber die Möglichkeit, den Mobile Agenten beliebig zu kopieren, bevor er mit dem eigentlichen Mobile Agenten zu ausführen beginnt. Der Server kann diese Kopien durchtesten, und daraus Rückschlüsse auf die gespeicherten Daten, den Programmcode und auch den Kontrollfluss ableiten [5].

In Abbildung 3.3 wird eine kurze Zusammenvergleich der oben genannten Ansätze, anhand drei Sicherheitsmechanismen basierend auf [5] dargestellt.

Wie in der Abbildung 3.3 steht, soll der „Kein Schutz“ Ansatz immer dort aufgeführt werden, wo kein Grundbedürfnis an Sicherheit gegeben ist.

	Vertraulich			Integrität			Verfügbarkeit
	C	D	K	C	D	K	
Kein Schutz							
Gesetze							
Trust	++	++	++	++	++	++	
Reputation				+	+	+	+

C = Code
D = Daten
K = Kontrollfluss
++ = vertrauenswürdiger Server notwendig
+ = Nicht völlig sicher

Abbildung 3.3: Vergleich der Ansätze

Die Sicherung über Gesetze und Verträge allein stellt nur zur Erkennung von Angriffen dar und macht keine Veränderung der gegebenen Situation. Dafür sollte neben einem anderen Verfahren angewandt werden und eine Rechtsgrundlage für entsprechende Streitfälle bilden.

Wie schon erwähnt, bietet der Trust Ansatz einen gleichermassen umfassenden und vor allen Dingen leicht zu realisierenden Schutz.

Der Reputation Ansatz schützt theoretisch die Integrität der einzelnen Mobile Agentenbereiche und ist gewünscht nur für Anwendungsgebiete, deren Sicherheit nicht zwingend notwendig ist [5].

3.4.3 Blackbox-Schutz

Eine weitere Lösungsansatz ist auch diese Blackbox-Schutz Ansatz. Absichtlich wurde dieser Ansatz in diesem separaten Kapitel behandelt, weil es eine Gesamtlösung der Sicherheitsproblem der Mobile Agenten bietet. Deswegen wird Blackbox in dieser Arbeit als wichtiger Ansatz genannt und spezieller betrachtet.

Blackbox-Schutz ist ein Verfahren, deren Anliegen es ist, den Mobile Agenten möglichst vor allen Angriffen zu schützen.

Bei diesem Ansatz wird versucht eine „Blackbox“ aus Agentencode unter Verwendung von Verwürfelungstechniken zu generieren [3]. Jeder Angreifer benötigt eine gewisse Zeit, um den Blackbox-Code zu analysieren. Das Ziel dieses Ansatzes ist, den Code für mindestens für diese Zeitintervall geschützt zu halten. Danach wird der Mobile Agent und die von ihm transportierten Daten ungültig [3].

3.4.3.1 Die Idee

Gemäss [3] wird ein beliebiger Ursprungsagent durch eine Konvertierung einen äquivalenten Agenten erzeugen, der zwar andere Struktur hat, aber dieselbe Funktionalität bietet, und somit auch weiter ausführbar ist.

„Als Blackbox wird ein Agent dann bezeichnet, wenn es nicht möglich ist, dass die Datenelemente und Codeteile eines Ursprungsagenten in der Blackbox erkannt werden können“ [3].

Das Ergebnis zeigt, dass der mögliche Angreifer die Werte der Datenelemente des Ursprungsagenten nicht bestimmen kann. Dabei ist es wichtig, dass diese Datenelemente und Codeteile auch temporär gesichert sind. Wenn ein Mobile Agent Blackbox ist, dann kann ein Angreifer nur noch Eingaben in und Ausgaben aus der Blackbox beobachten. Beobachten heisst, er kann die Zustandsänderungen des Mobile Agenten wahrnehmen, diese aber nicht ändern. Somit ist ein Blackbox ein Agent mit der Eigenschaft „autonom“, der diese um andere Angriffe zu verhindern nutzt.

3.4.3.2 Eigenschaften der Blackbox

Nach dem obigen Definition ist ein ursprünglicher Mobile Agent durch die Konvertierung in eine Blackbox, ein konvertierter Mobiler Agent entstanden, der vor bestimmten Angriffen geschützt ist. Im Abschnitt 3.3.3 war die Rede immer von einem Mobile Agent, während hier um zwei davon geht. Dem ursprünglichen und den konvertierten Mobile Agent.

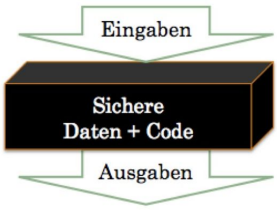

Abbildung 3.4: Blackbox

Das Verhalten von beiden ist eigentlich auch für den Wirt identisch. Sie unterscheiden sich von der äußeren Form, die zum Schutz gehört und entsprechend für das äussere Interaktionsverhalten dem Code und dem Ausführungszustandes des mobilen Agenten bedeutet.

Wenn man den Datenzustand des mobilen Agenten betrachtet, gilt eine Art Inklusionsbeziehung. Das bedeutet, „für jedes Datenelement aus dem Datenzustand des Ursprungsagenten gilt, dass eine Funktion existiert, die dieses Datenelement aus dem Datenzustand der Blackbox dieses Ursprungsagenten erzeugen kann“ [3]. Eine Blackbox kann auch weitere Daten enthalten, die nicht zum Interaktionsverhalten beitragen.

Eine Blackbox unterscheidet von ihrem Ursprungsagenten, sonst wäre ein Schutz nicht zu gewährleisten. Die Ausführung beider Agenten erfolgt durch den Host in verschiedener Weise. Es sind folgende Angriffsmethoden bezogen auf den Ursprungsagenten ausgeschlossen [3]:

- Lesen der Agentendaten
- Zielgerichtete Modifikation von Agentendaten

- Lesen des Codes
- Inkorrektes Ausführen von Code
- Lesen der Ausführung des Agenten
- Temporäre Modifikation der Agentenausführung

Eine weiterführende Literatur zur Thema Sicherheit der Mobile Agenten findet man in [3], wo auch der technischer Aspekt detailliert erklärt ist.

3.4.4 Zusammenfassung

Die verschiedene Teillösungsansätze wurden zu den genannten Sicherheitsproblemen der Mobile Agenten kurz beschrieben. Danach wurden einige Schutzansätze anhand der Vertraulichkeit, Integrität und Verfügbarkeit verglichen. Anschliessend wurde noch der Blackbox-Ansatz als Gesamtlösungsansatz näher beschrieben.

3.5 Wirtschaftliche Sicht

Hier wird der Wirtschaftliche Aspekt der Mobile Agenten betrachtet. Wobei die Einsatzmöglichkeit der Mobile Agenten auf elektronischen Märkten erkannt wird. Anhand der Prozessmodell wird der Benutzung der Mobile Agenten in verschiedene Phasen näher betrachtet. Der wichtige Einblick der Sicherheit wird zunächst in die Elektronischen Märkten geworfen. Danach wird der Mobile Agent zum Preisvergleich in praktischen Sicht gesehen. Anschliessend wird ganz kurz über den heutigen Verbreitung und den Zukunft der Mobile Agenten diskutiert.

3.5.1 Elektronische Märkte

Eine allgemeine Marktdefinition aus der Ökonomie definiert die elektronische Märkte als „ökonomische Orte des Tausches“, wo „die aggregierte Nachfrage auf das aggregierte Angebot trifft“ [5]. Das Ziel dabei ist die gesamtwirtschaftliche Nutzen möglichst zu maximieren. Wobei verschiedene ökonomische Faktoren und Ressourcen zu berücksichtigen bzgl. zu optimieren sind. Nach [3] sind „elektronische Märkte im engeren Sinne mit Hilfe der Telematik realisierte Marktplätze, d.h. Mechanismen des marktmässigen Tausches von Gütern und Leistungen, die alle Phasen der Transaktion unterstützen“.

In diesem Abschnitt soll der Prozess des Gütertausches anhand der Phasenmodell analysiert.

3.5.1.1 Phasenmodell der Koordination

Es wird in Abbildung 3.5 basierend auf [5] eine einheitlichen Schema gezeigt, wonach der Prozess des Tausches von Gütern oder Leistungen, möglichst unabhängig von der Art der getauschten Ressourcen, abläuft.

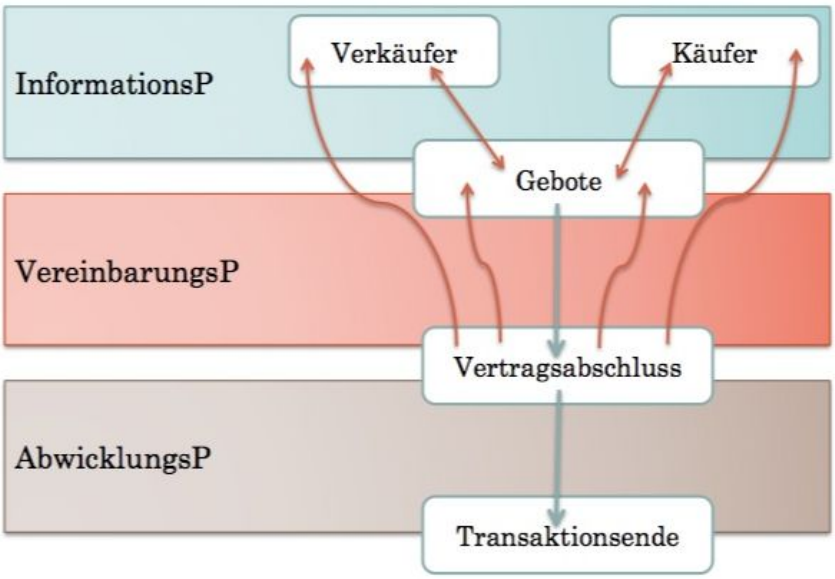

Abbildung 3.5: Phasenmodell

Informationsphase In der Informationsphase, werden nötigen Informationen über die zu tauschenden Güter oder Leistungen eingeholt. Als allgemeine wichtige Standardinformationen sind:

- von der Nachfrageseite
  1. Produkt,
  2. Anbieter, und
  3. Preis.
- von der Anbieterseite
  1. Kundenname,
  2. Kundenadresse,
  3. Kundenpreisvorstellungen,
  4. Kundeneinkommensverhältnisse, ect.

Wenn eine Kunde (Nachfrage) für ein bestimmtes Produkt entschieden hat, das heisst er hat die Informationsphase schon abgeschlossen, erst dann beginnt der grösste Teil dieser Phase von der Anbieterseite [5].

Vereinbarungsphase Hier geht es um wesentliche Vereinbarungen und Entscheidungen zwischen Kunde und Anbieter, wie z.B.: die Konditionen und die Durchführung der Transaktion. Was zu einem höheren Kommunikationsaufwand zwischen den Geschäftspartnern führt. Wichtiger ist hier auch der Vertragsabschluss am Ende der Phase, wo alle rechtlichen Voraussetzungen für die Transaktion stehen [5].

Abwicklungsphase In diesem Phase geht es um die Durchführung der eigentlichen Transaktion (Güteraustausch). Dabei kann je nach Art der Ware noch: Verpackung, Transport, Zwischenlagerung, Versicherung, usw. dazukommen [5].

3.5.2 Mobile Agent zum Preisvergleich

In bisherigen Abschnitten wurde vieles über den Mobile Agent diskutiert. Auch der Prozessmodell wurde erläutert, anhand dem wir jetzt die paktische Seite der Mobile Agenten betrachten. Es wird bei jeder Phase kurz eingegangen.

Bei der Informationsphase in Abbildung 3.6 wird graphisch mehr zu den Ablauf des gleichen Geschäftes unter Benutzung eines mobilen Agenten gezeigt. Sobald der Mobile Agent von einem Benutzer beauftragt wird, wird er zum ersten Marktserver gesendet. Wo er von dem erreichten Marktserver bestimmte Anfragen startet, die für ihn wichtige Informationen liefern. Der Mobile Agent wandert von einem Marktserver der Anbieter (An) zum Anderen und sammelt Informationen. Beim letzten Marktserver vergleicht er alle gesammelten Informationen und ermittelt dadurch den günstigsten davon. Am Schluss besucht der Mobile Agent nochmals den billigsten Anbieter und führt weitere Geschäftsprozesse, siehe hier im Kapitel 3.5.1, durch [5].

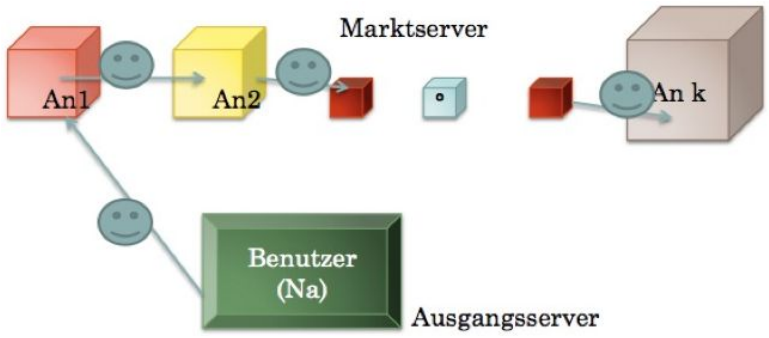

Abbildung 3.6: Mobile Agent zum Informationssphase

In diesem Fall hat der Informationsumfang des Geschäfts allgemein keinen Einfluss auf die zu übertragenden Informationen. Somit wird wie Abbildung 3.7 zeigt, eine letzte Verbindung benötigt, um den Mobile Agenten zum Benutzer (Na) zurücksenden, um den Beauftragten über den Status des Geschäfts zu informieren. In dieser

Die Mobile Agenten sind somit betrachteten Rolle des Nachfragers (Na) von Bedeutung. Für den Anbieter ist nicht möglich, ihrerseits einen Mobile Agenten als „Handelsvertreter“ zu verschiedenen potentiellen Käufern (Na) zu senden. Weil dazu eine sehr breite Informationspalette zu den nachgefragten Produkten oder Serviceleistungen ausgetauscht werden müssen [5].

In der Abwicklungsphase ist das Vertrauen eine wichtige Punkt, die in die Mobile Agenten gefordert wird.

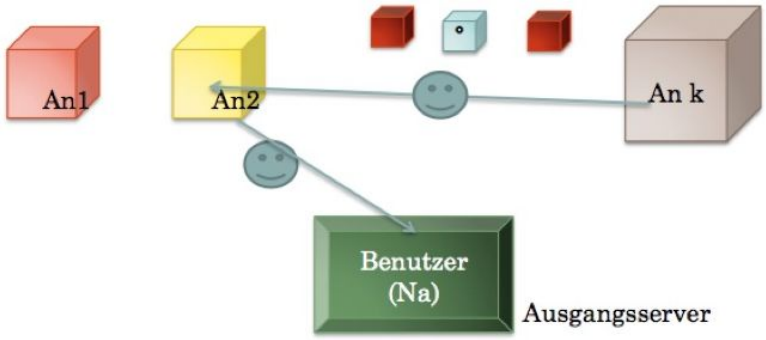

Abbildung 3.7: Mobile Agent zum Vereinbarungsphase

3.5.2.1 Sicherheit auf elektronischen Märkten

Wie es allgemein erläutert wurde, ist die Sicherheit auch auf elektronischen Märkten ein wichtiger Punkt. Hier kann man zwei unterschiedliche Probleme sehen:

1. Völlige Kontrolle: Das Agentenprogramm steht völlig unter der Kontrolle des entfernten ausgeführten Rechners. Aufgrund der Eigenschaft „Autonomie“ des Mobilen Agenten kann der Benutzer folgende Möglichkeiten nicht überprüfen: die richtige Programmausgeführung, die Datenmanipulierung oder die Programmzerstörung [5].
2. Fehlende Kenntnisse: Der entfernte Rechner kann nicht wissen, welche Funktionen das Programm überhaupt ausführt hat, ohne es vorher analysiert zu haben. Weil im Falle eines unbekanntes Programmausführung kann zu grossen Risiko kommen, dass durch Virenprogramme lokale Daten des Rechners ausgelesen, manipuliert oder zerstört werden könnten [5].

3.5.3 Marktdurchdringung

Bei der Einsatz von Mobile Agenten, wie schon hier im Abschnitt 3.2.3 erklärt wurde, gibt es viele Vorteile. Auch [16] zitiert, dass „Mobile agents are a solution in search of a problem“ (J. Ousterhout). Trotzdem sind noch nicht so viele solcher Systeme im Einsatz. Die grössten Gründer dafür seien die Sicherheitsbedenken [7].

Für die Zukunft zeichnet sich eine Interesse an vielen kommerziellen Systeme [16] als eine eigenständige Technologie. Ansonsten besteht auch die Möglichkeit der Kombination der klassischen Prinzipien, wobei eine breite Anwendungsentwicklung zur Verfügung steht.

3.5.4 Zusammenfassung

Der Mobile Agent aus der wirtschaftlichen Sicht wurde hier betrachtet. Zunächst wurde die Anwendungsmöglichkeit der Mobile Agenten auf elektronischen Märkten näher erkannt, worauf auch ein Einblick der Sicherheit geworfen wurde. Danach wurde der Mobile Agent zum Preisvergleich in wirtschaftlichen Sicht gesehen. Und anschliessend wurde ganz kurz über den heutigen Verbreitung und den Zukunft der Mobile Agenten diskutiert.

3.6 Zusamenfassung und Schlussfolgerung

Mobile Agenten sind Programme, die sich innerhalb eines Netzwerkes autonom von einer Umgebung zu einer anderen Umgebung bewegen und alle dabei gesammelten Daten mitführen [9]. Darausfolgt, dass sie die Fähigkeit mit vielen anderen Rechner innerhalb eines Netzwerks zu kommunizieren haben.

Es wurden mögliche Anwendungsgebiete für mobile Agenten kurz beschrieben. Der Anwendungsbereich scheint aber unbegrenzt zu sein. Dabei wurden für mobile Agenten eine ganze Reihe von möglichen Vorteilen genannt [3], die daneben auch gewisse Nachteile mitbringen.

Der Aspekt der Sicherheit wurde auch nicht vernachlässigt, im Gegensatz als ganz Wichtig betrachtet. Für den Schutz des ausführenden Rechners wurden dazu einige Sicherheitselemente betrachtet [3].

In diese Arbeit gingt es im Besonderen mit der Frage unter welchen Bedingungen ein mobiler Agent vor Angriffen geschützt wird. Daher wurden möglichen Angriffe geschildert, die zur Beantwortung dieser Frage dienen. Ausser andere Teilschutzansätze wurde auch der Blackbox als einer wichtiger Ansatz hingedeutet [3].

Als zukünftiger Anwendungsgebiet der Mobile Agenten wurde der elektronische Handel näher anhand von Beispielen beschrieben.

Die Auseinandersetzung mit dem Paradigma mobiler Agenten zeigt, dass diese Technologie ein grosses Potential für zukünftige Entwicklungen und Einsatzszenarien bietet, nicht nur im elektronischen Handel, sonder auch in die Produktentwicklung [8] und besonders für die zukünftige Automatisierungstechnik, wobei die Mobile Agenten eine zukunftsträchtige Ergänzung der Web-technologien darstellt. Die Frage der Sicherheit und des Schutzes mobiler Agenten ist und bleibt jedoch das Kernproblem. Der vorgestellte Szenario bringt die mobilen Agenten aus dem praktikablen Einsatz einen Schritt näher.

Dennoch ist das Grundproblem, der Schutz der Ausführung von Agenten auf nicht vertrauenswürdigen Plattformen, nicht grundsätzlich gelöst.

Literaturverzeichnis

[1] Volker Roth: *Über die Bedeutung eines Statischen Kernes für die Sicherheit mobiler Software-Agenten*; http://www.volkerroth.com/download/Roth2001a.pdf; Februar 2008.

[2] Ivan Gladenko: *Makings Agents Secure on the Web*; http://ivs.cs.uni-magdeburg.de/~dumke/OperWeb/Gladenko/Folien.ppt; Februar 2008.

[3] Fritz Hohl: *Sicherheit in Mobile-Agenten-Systemen*; http://www.agentensystem.de/dipl-kap6.html; Februar 2008.

[4] Kai Simeth: *Intelligente Software-Agenten*; http://simeth-online.de/swagents.htm; Februar 2008.

[5] Torsten Mandry: *Sicherheit von mobilen Agenten auf elektronischen Märkten*; http://www.agentensystem.de/publist.html; Februar 2008.

[6] Chr. Tschudin, G. Di Marzo, M. Murhimanya, J. Harms: *Welche Sicherheit für mobilen Code?*; http://arvo.ifi.uzh.ch/ikm/tschudin/research/sis96-folien.ps.gz; März 1996.

[7] Volker Roth:*Mobile Software Agenten und Sicherheit – eine Haßliebe?* http://www.volkerroth.com/download/Roth2000b.pdf; März 2008.

[8] Josef Renner: *Mobile Agenten für den Fernzugriff auf eingebettete Systeme* http://www.atypon-link.com/OLD/doi/abs/10.1524/auto.2007.55.8.394; März 2008.

[9] Frank Joachim Leitner: *Architektur eines sicheren Mobile-Agenten-Systems für das Netzmanagement* http://deposit.ddb.de/cgi-bin/dokserv?idn=969360045; Februar 2008.

[10] Volker Roth:*Mobile Software-Agenten* http://www.funkschau.de/heftarchiv/pdf/2000/fs06/fs0006048.pdf; März 2008.

[11] Diana Fechtemeier, Filiz Sen:*Prototypische Implementierung einer Agentenkommunikation mit XML* http://www.coagens.de/german/information/Sen-Fechtemeier_Prototypische_Implementierung_einer_Agentenkommunikation_mit_XML.pdf; März 2008.

[12] Dietmar Feckelmann:*Rechtekonzepte für die Mobile Agent System Architecture (MASA)* http://www.nm.ifi.lmu.de/pub/Diplomarbeiten/fack01/HTML-Version/index.html; März 2008.

[13] Friedemann Mattern:*Mobile Agenten* http://www.vs.inf.ethz.ch/publ/papers/mobags.html; März 2008.

[14] Reto Trinkler:*TCP/IP in drahtlosen Netzen* http://www.basis06.com/media/pdf/speech_wexpo.pdf; April 2008.

[15] Helmut Reiser: *Sicherheitsarchitektur für ein Managementsystem auf der Basis Mobiler Agenten* http://www.nm.ifi.lmu.de/pub/Dissertationen/reis01/; März 2008.

[16] Friedemann Mattern, Stefan Fuenfrocken :*Mobile Agenten als Architekturkonzept Internet-basierter Anwendungen* http://www.vs.inf.ethz.ch/publ/slides/kivs99.pdf; März 2008.

[17] Stefan Fuenfrocken:*Mobile Agenten im Internet* http://www.vs.inf.ethz.ch/publ/papers/online98.pdf; März 2008.

[18] Matthias Rohr:*Mobile Agenten* http://www.fh-wedel.de/~si/seminare/ss04/Ausarbeitung/4.Rohr/mobile_agenten_ss04_seminar_dok.pdf; März 2008.

[19] Prof. Dr. Dr. h.c. mult. Gerhard Krueger, Roland Bless, Hartmut Ritter, Dr. Jochen Schiller, Rainer Ruggaber:*Klausurtagung des Instituts für Telematik* http://digbib.ubka.uni-karlsruhe.de/volltexte/documents/1278; April 2008.

[20] Ying Lin, Florian Michahelles:*Mobile Agenten* http://www.inf.ethz.ch/personal/michahel/RefServ/mobil.htm; Juli 1999.