Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Seminararbeit

Sicherung von Mobile IPv6-Handovers durch Cryptographically Generated Adresses

Olaf Christ

Fakultät Technik und Informatik
Studiendepartment Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Inhaltsverzeichnis

1 Motivation

Cryptographically Generated Addresses (CGAs) [12] können auftretende Authentifizierungsprobleme bei der Kommunikation über unsichere Netze wie das Internet lösen. So verwendet beispielsweise das Secure Neighbor Discovery Protokoll (SEND) [11] CGAs, um der Gefahr des Address Resolution Protocol (ARP)-Spoofings entgegenzuwirken. Bei den derzeit aktuellen Authentifizierungsmethoden muss vor der Kommunikation entweder ein gemeinsamer, geheimer Schlüssel oder die öffentlichen Schlüssel der Kommunikationspartner bekannt sein. CGAs hingegen benötigen diesen vorherigen Schlüsselaustausch nicht. Durch geschickten Einsatz des bei IPv6 frei konfigurierbaren Interface Identifiers (siehe Kapitel 2.1) kann eindeutig bewiesen werden, dass der Sender des Pakets auch der legitime Besitzer der IP-Adresse ist (Proof of Address Ownership). Die beim Secure Socket Layer (SSL) angewandte Authentifizierungsmethode ist das Aufsetzen einer Public Key Infrastructure (PKI). Dieser Ansatz ist allerdings in vielen Anwendungsszenarien nicht gewünscht, da er administrativen Aufwand mit sich bringt und dadurch Verwaltungskosten entstehen.

CGAs können für die unterschiedlichsten Einsatzzwecke verwandt werden, z.B. zum Sichern des Binding Updates bei Mobile IPv6 (siehe Kapitel 3.2.3). Dabei können die Kommunikationspartner im Voraus kein Vertrauensverhältnis aufbauen. Nach einem Adresswechsel muss sichergestellt werden, dass die Kommunikation nicht von einem Dritten belauscht oder verändert wird. Durch CGAs kann ein Vertrauensverhältnis zwischen den Kommunikationspartnert aufgebaut werden. Zusätzlich kann die Handover-Latenz bei Mobile IPv6 durch die Verwendung von CGAs im Gegensatz zur Benutzung der Return Routability Procedure deutlich verringert werden (vgl. [1, 3]), da für das Binding Update lediglich ein Datenpaket mit einem zusätzlichen Header versehen werden muss.

2 Einleitung

Im Folgenden soll ausschließlich MIPv6, also das Mobile IP für IP Version 6, betrachtet werden, da CGAs einen frei wählbaren Interface Identifier benötigen und somit nur mit IPv6 funktionieren.

2.1 IPv6

IPv6 [5] ist der Nachfolger des aktuell verbreiteten Internetstandards IPv4. Einige der herausragenden Merkmale werden im Folgenden vorgestellt.

2.1.1 IPv6 Adressen

Die IP-Adresslänge wurde von 32 auf 128 Bit erhöht, um einen Adressvorrat auch für kommende Anwendungen zur Verfügung zu stellen. Dies ist beispielsweise wichtig für eine Inter-Fahrzeug-Kommunikation im öffentlichen Straßenverkehr, bei der jedes Fahrzeug mit einer eigenen IP-Adresse ausgestattet werden muss.

IPv6-Adressen bestehen aus einem Prefix und einem Interface Identifier. Sie werden üblicherweise in der folgenden Notation dargestellt:

```
2001:0d45:1b57:0159:3645:c3ff:0730:7635/64
```

Das an die IP-Adresse angehängte `/64` kennzeichnet, wie viele Bits der Adresse zum Prefix gehören. In diesem Fall entsprechen die vorderen 64 Bit dem Prefix und die hinteren 64 Bit dem Interface Identifier:

```
Prefix:               2001:0d45:1b57:0159
Interface Identifier: 3645:c3ff:0730:7635
```

Zusätzlich besitzen Prefixe einen hierarchischen Zusammenhang. Das oben angegebene Prefix gehört z.B. zum übergeordneten Prefix `2001:0d45:1b57/48`, welches wiederum dem Prefix `2001:0d45/32` untergeordnet ist usw. Durch diese Hierarchie kann das Routing deutlich effizienter stattfinden.

Im Gegensatz zu IPv4 kann bei IPv6 ein IP-Interface mehrere IP-Adressen besitzen. So besitzt ein Rechner standardmäßig eine Link-lokale und eine global Eindeutige (Global Unique) IP-Adresse [13]. Zwar konnte eine Netzwerkkarte bereits bei IPv4 mehrere IP-Stacks mit jeweils einer Adresse verwalten, die Bindung mehrerer IP-Adressen an denselben Stack ist jedoch neu.

2.1.2 Address Configuration

Die IP-Adresse kann beim Anmelden eines Geräts an ein Netzwerk entweder durch Stateful oder Stateless Configuration vergeben werden. Die Stateful Address Configuration funktioniert analog zur DHCP-Adressvergabe bei IPv4. Hierbei vergibt ein DHCPv6-Server freie Adressen und weitere Netzwerkdetails wie z.B. das Default Gateway oder DNS-Server-Adressen. Bei der Stateless Address Configuration sendet ein Router regelmässig sog. Router Advertisment (RA) Pakete, die Informationen wie z.B. Prefix und DNS-Server über das Netzwerk enthält. Ein Gerät bildet aus dem erhaltenen Prefix zunächst eine Link-lokale, dann zusätzlich eine globale IP-Adresse. Vor der ersten Verwendung muss geprüft werden, ob diese Adresse bereits genutzt wird. Dazu wird das Duplicate Address Detection-Verfahren (DAD) eingesetzt [6].

2.2 Mobile IPv6

Mobile IPv6 bezeichnet ein Protokoll, das es erlaubt, nach einem IP-Adresswechsel - z.B. durch Wechsel des Access Points - bestehende Transportbindungen aufrecht zu erhalten. Der sich bewegende Kommunikationspartner wird als Mobile Node (MN) bezeichnet, der feste Kommunikationspartner als Correspondent Node (CN) (siehe Abbildung 1). Zusätzlich muss ein Home Agent (HA) existieren. Diese Funktion wird üblicherweise von einem Router übernommen.

Zunächst meldet sich der MN an seinem HA an (siehe Abbildung 2). Beispielhaft befinden sich der MN und der HA hier zu Beginn im selben physikalischen Netz. Möchte der CN mit dem MN kommunizieren, kann dieses direkt stattfinden. Bewegt sich der MN in ein anderes Subnetz, reisst die Kommunikation zwischen MN und CN zunächst ab. Der MN bildet nach dem Adresswechsel eine neue Care-of Adresse (CoA), meldet sich dann beim HA und gibt seine aktuelle IP-Adresse (CoA) bekannt. Dieser Vorgang wird als Binding Update (BU) bezeichnet. Der HA bestätigt das Update mit dem Binding Update Acknowledge (Back) Paket. Er nimmt von nun an die IP-Pakete für die HoA des MN entgegen und tunnelt diese an seine CoA. Pakete vom

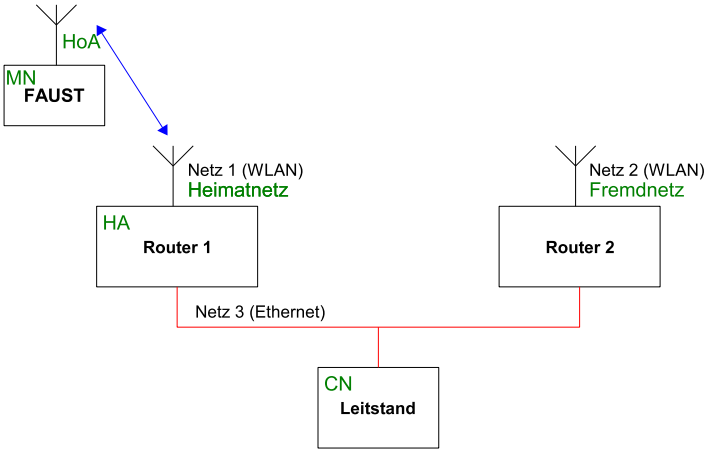

Abb. 1: Testnetz Mobile IPv6

MN zum CN werden weiterhin direkt gesendet. Um auch dem CN das BU zukommen zu lassen wird es als Extension Header in das erste Paket nach dem Handover eingefügt. Eine Ausnahme dazu ist die Return Routability Procedure (siehe Kapitel 3.1). Mobile IPv6 ist standardisiert in [9]. Ausführungen finden sich in [19], ein Anwendungsszenario in [20].

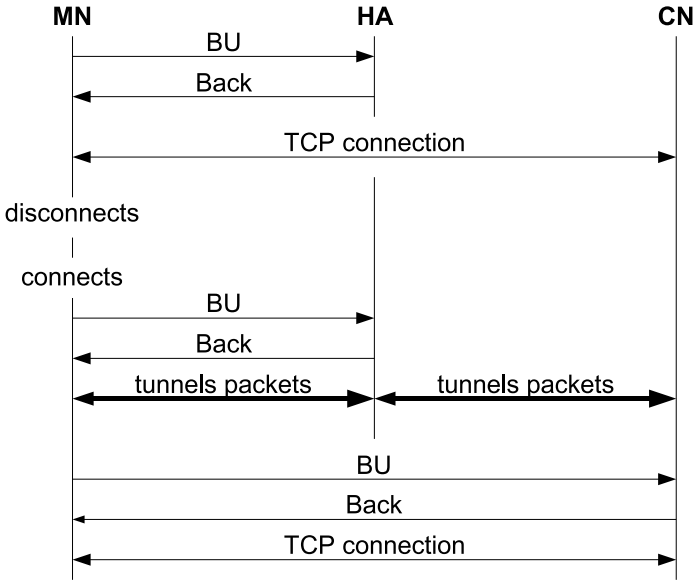

Abb. 2: Protokollablauf Mobile IPv6 zwischen Mobile Node, Home Agent und Correspondent Node

3 Sicherung der Binding Updates

Bei der Entwicklung von IP-Version 6 wurde besonderen Wert darauf gelegt, dass Verschlüsselungs- und Authentifizierungsmechanismen direkt implementiert sind. Diese Mechanismen sind standardisiert als IP Security (IPSec) [14]. Das Authentifizierungsprotokoll Authentication Header (AH) [15] bietet Schutz vor Paketveränderungen bei der Übertragung. Dazu müssen

die Kommunikationspartner gegenseitig bekannt sein, wie z.B. der MN und der HA [10]. Da sich MN und CN jedoch meist nicht bekannt sind, kann IPSec zwischen ihnen nicht angewandt werden.

3.1 Return Routability Procedure

Eine Gefahr, die bei der Kommunation über das Internet auftritt, ist die sog. Man in the Middle-Attacke (MitM). Dabei fängt der Angreifer das BU-Paket an den CN ab und manipuliert es, so dass fortan sämtlicher Traffic über ihn läuft. Dieser Angriff kann von jedem Rechner im Internet durchgeführt werden, von dem das BU-Paket weitergeleitet wird. Eine Lösung dieser Gefahr ist der Einsatz der Return Routability Procedure.

Die Return Routability Procedure [9] erweitert den Vorgang des BUs durch eine Folge von Paketen. Anstelle des BU-Extension Headers werden die Pakete Home-Test Init (HoTi) und Care-of Test Init (CoTi) versendet (siehe Abbildung 3). Das CoTi-Paket wird auf direktem Wege an den CN gesendet, das HoTi-Paket nimmt den Umweg über den HA. Beide Pakete sind mit einem Cookie versehen, um die Beziehung der beiden Pakete zueinander abzubilden.
Der CN sendet als Antwort auf diese Pakete, das Home Test (HoT) und das Care-of Test (CoT)-

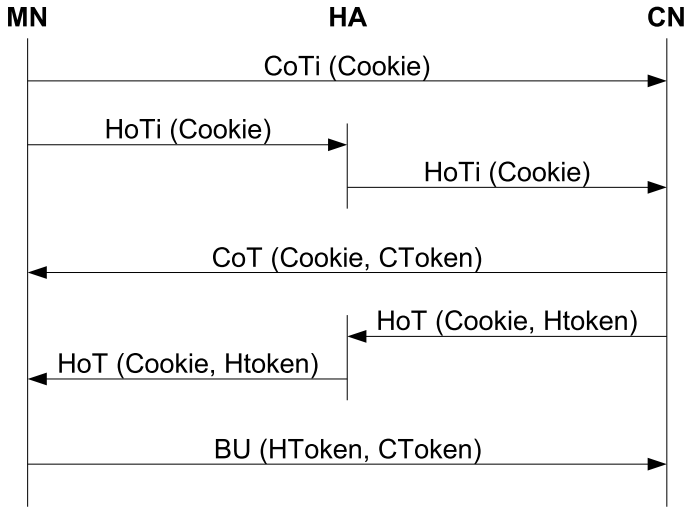

Abb. 3: Protokollablauf Return Routability Procedure

Paket auf denselben Wegen zurück zum MN. Diese Pakete sind mit dem Home Token (HToken) bzw. Care-of Token (CToken) versehen. Diese Zufallszahlen werden vom MN empfangen und in einem weiteren Paket (BU) auf direktem Wege an den CN gesendet. Durch das gleichzeitige Senden beider Token kann der MN beweisen, dass er sowohl über die CoA-Adresse als auch über die HoA erreichbar ist.

Ein Angreifer müsste nun sowohl das HToken als auch das CToken erlangen, um einen Angriff durchführen zu können. Da diese Beiden über völlig unterschiedliche Wege vom CN zum MN geschickt werden, ist es unwahrscheinlich, dass ein Angreifer Beide abfangen kann. Es bestehen trotzdem noch zwei Angriffspunkte: Beide Token werden im Subnetz des MN (siehe Abbildung 4 ①) und des CN (②) auf den gleichen Wegen übertragen. Findet die Übertragung zwischen MN

und HA jedoch verschlüsselt über IPsec statt, so fällt auch ① als möglicher Angriffspunkt aus. Somit bleibt nur das Subnetz des CN übrig. Dies ist allerdings keine Mobile IPv6-spezifische Gefahr und kann damit für diese Untersuchung vernachlässigt werden.
Eine mögliche Lösung ist der Einsatz von CGAs für das BU, welche im nächsten Abschnitt

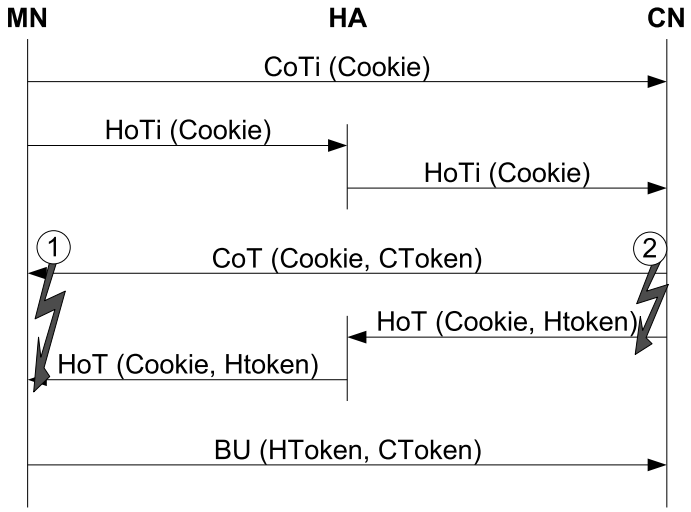

Abb. 4: Man in the Middle-Attacke bei der Return Routability Procedure

behandelt wird.

Zusätzlich ist zu erwähnen, dass durch diese Prozedur Latenzen auftreten, die bei einem Echtzeit-Datenstrom unerwünschte Folgen haben kann. Dem kann durch sog. Credit-Based Authorization [16] entgegengewirkt werden.

3.2 CGAs

Beim Mobile IPv6-Handover besteht für den MN die Notwendigkeit, zu beweisen, dass er legitimer Besitzer der HoA ist und dass das BU unverfälscht übertragen wird. Eine mögliche Lösung dazu wäre das Authentifizieren mit Hilfe einer Public Key Infrastructure (PKI) [7]. Dazu vergibt eine Zertifizierungsstelle (Certificate Authority (CA)) signierte Zertifikate an die einzelnen MNs. Diese sind durch die Signatur eindeutig mit der CA verbunden. Ein CN erhält dieses Zertifikat vom MN und prüft es gegen das öffentliche Zertifikat der CA. Damit kann sichergestellt werden, dass der MN ein gültiges Zertifikat besitzt.
Eine PKI birgt allerdings auch einige Nachteile. Das Aufsetzen einer PKI ist kostspielig und mit hohem Administrationsaufwand verbunden. Für jeden MN müsste ein eigenes Zertifikat erstellt und sicher übermittelt werden. Zusätzlich müsste auch jeder CN mit den aktuellen öffentlichen CA-Zertifikaten und Certificate Revocation Lists (CRLs) ausgestattet sein.

Die Grundidee kryptographisch generierter Adressen (CGAs) ist die Benutzung eines kryptographischen Merkmals als Interface Identifier. Pakete können so selbstkonsistent authentifiziert werden, ohne vorher ein Vertrauensverhältnis durch ein gemeinsames Passwort (shared secret) oder durch einen Public Key-Austausch aufzubauen.

3.2.1 Generierung von CGAs

Zum Bilden der CGAs ist die CGA-Parameter Datenstruktur nötig (vgl. Abbildung 5). Zusätzlich

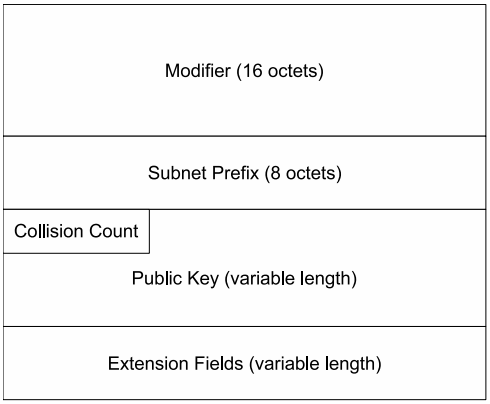

Abb. 5: CGA-Parameter Datenstruktur (aus [12])

existiert ein Sicherheitsparameter *Sec*, der die Sicherheitsstufe darstellt. Er kann die Werte zwischen Null und Sieben annehmen. Je höher der Wert, umso höher ist die Sicherheit der CGA (siehe Kapitel 4).
Das Verfahren zur Generierung von CGAs ist in Abbildung 6 dargestellt.

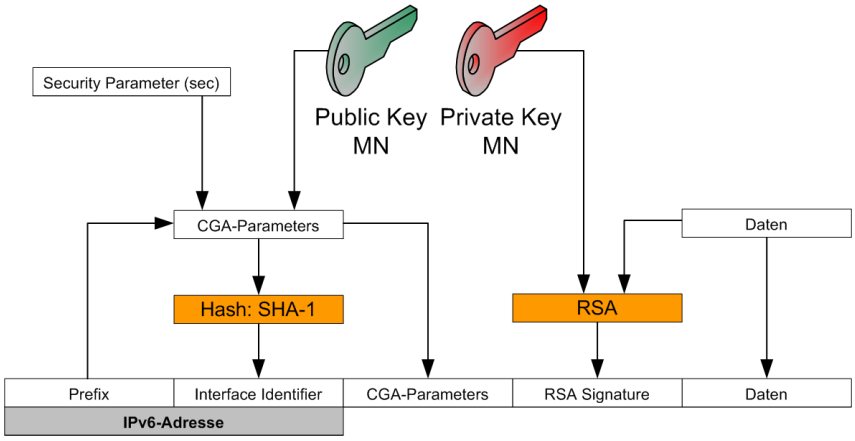

Abb. 6: Generierung einer CGA

Es werden folgende Schritte durchgeführt:

1. Der Modifier der CGA-Parameter Datenstruktur wird mit einer Zufallszahl belegt.
2. Der Modifier, 9 Null-Octets und der Public Key werden von links nach rechts aneinandergehängt. Hiervon wird der SHA-1 Hashwert gebildet. Die 112 ersten Bits sind der Hash2.
3. Die 16*sec ersten Bits des Hash2-Werts werden mit Null verglichen. Sind alle Null, geht es weiter bei Schritt 4. Ansonsten wird der Modifier um 1 inkrementiert und Schritt 2 wird wiederholt.
4. Der 8-bittige collision count wird auf Null gesetzt.
5. Es wird ein SHA-1 Hashwert über die gesamte CGA-Parameter Datenstruktur, einschließlich der Extension Fields, gebildet. Die ersten 64 Bit bilden den Hash1.

6. Ein interface identifier wird aus dem Sicherheits-Parameter sec und dem Hash1 gebildet.
7. Die IPv6-Adresse ergibt sich aus dem subnet prefix und dem gerade gebildeten interface identifier.
8. Es muss nun geprüft werden, ob die IPv6-Adresse bereits benutzt wird. Dazu wird der Mechanismus des duplicate address detection (siehe [11]) verwendet. Falls ein Adresskonflikt besteht, wird das collision count Feld um 1 erhöht und bei Schritt 5 fortgefahren. Diese Wiederholung darf lediglich 3 Mal durchgeführt werden. Sollte danach immer noch keine eindeutige IPv6-Adresse gefunden worden sein, so ist ein Fehler aufgetreten.

3.2.2 Überprüfung von CGAs

Das CGA-authentifizierte Paket, das den Zielhost erreicht, enthält die CGA-Parameter Datenstruktur, welche im Folgenden verwendet wird.
Zur Überprüfung von CGAs werden folgende Schritte durchgeführt (vgl. Abbildung 7):

1. Der Wert des collision count-Feldes muss zwischen 0 und 2 liegen.
2. Das subnet prefix aus der CGA-Parameter Datenstruktur muss mit dem subnet prefix der IP-Adresse übereinstimmen.
3. Der SHA-1 Hash wird über die komplette CGA-Parameter Datenstruktur gebildet. Die ersten 64 Bit sind der Hash1.
4. Der Hash1-Wert muss mit dem interface identifier der IP-Adresse übereinstimmen. Ausgenommen ist hierbei der security parameter *Sec*, der die ersten 3 Bits des interface identifier belegt.
5. Der Modifier, 9 Null-Octets, der Public Key und eventuelle extension fields werden von links nach rechts aneinandergehängt. Hiervon wird der SHA-1 Hashwert gebildet. Die 112 ersten Bits sind der Hash2.
6. Die 16*sec ersten Bits des Hash2-Werts werden mit Null verglichen. Sind sie Null, ist die Überprüfung erfolgreich abgeschlossen.

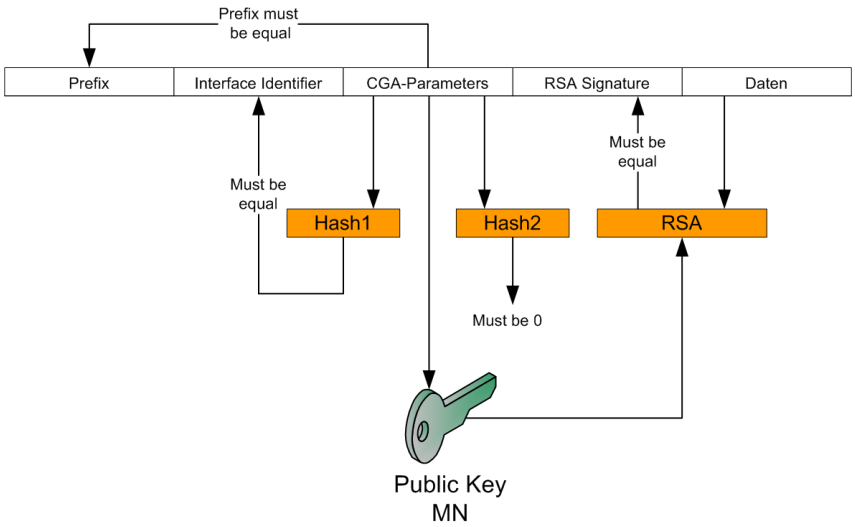

Abb. 7: Überprüfung einer CGA

Von nun an kann davon ausgegangen werden, dass der Public Key aus der CGA-Parameter Datenstruktur während der Übertragung nicht verändert wurde. Er kann somit für weitere Sicherheitsmaßnahmen benutzt werden.

3.2.3 Enhanced Route Optimization für Mobile IPv6

Das Enhanced Route Optimization-Verfahren [16] zeichnet sich im Gegensatz zur Return Routability Procedure dadurch aus, dass lediglich ein Paket vom MN zum CN verändert werden muss. Dazu wird das eigentliche Binding Update-Paket um die CGA-Parameter Datenstruktur erweitert und in das aktuell zu übertragende IP-Paket eingefügt (siehe Abbildung 8). Durch den CGA-Mechanismus wird der Besitz der HoA eindeutig nachgewiesen. Außerdem wird der Public Key fälschungsfrei übertragen. Signiert der MN das BU mit seinem privaten Schlüssel digital, kann es vom CN mit Hilfe des öffentlichen Schlüssels überprüft werden.

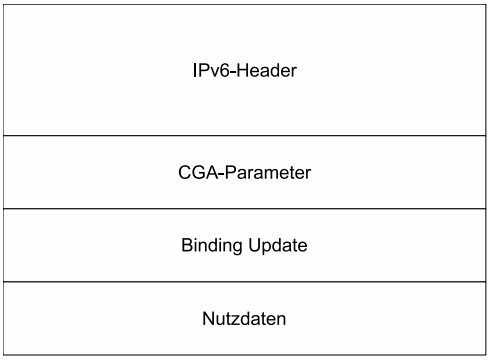

Abb. 8: CGA-Optimiertes BU-Paket

4 Aufwandsanalyse zur Generierung von CGAs

Zur Generierung von CGAs werden zwei Hashes erstellt: Hash1 und Hash2. Als Hashfunktion kommt SHA-1 [17] zum Einsatz. Dabei wird jedes Eingabebit dazu verwandt, den Hashwert zu berechnen. Das Verändern eines einzelnen Bits führt weiterhin zu einem völlig veränderten Hashwert. Damit steigt allerdings auch der Berechnungsaufwand linear mit der Länge der Eingabe.

Während die Generierung des Hash1-Wertes einen gleichbleibenden Aufwand erfordert, ist zum Berechnen des Hash2-Wertes deutlich mehr Aufwand nötig. Wie in Kaptiel 3.2.1 gesehen, wird die Eingabe in den SHA-1 Algorithmus so lange angepasst, bis eine definierte Anzahl an Bits des Hash2-Werts Null sind. Diese Anzahl hängt vom Sec-Parameter ab:

```
Sec=0 0x00000000 00000000 00000000 0000
Sec=1 0xffff0000 00000000 00000000 0000
Sec=2 0xffffffff 00000000 00000000 0000
Sec=3 0xffffffff ffff0000 00000000 0000
...
```

Die sich aus dem Sec-Parameter ergebende Maske wird über die ersten 112 Bits des Hash-Werts gelegt. Danach wird überprüft, ob der resultierende Wert gleich Null ist.

Um den Rechenaufwand, einen Hash2-Wert zu erstellen, abschätzen zu können, wird experimentell untersucht, wie weit zwei mögliche Hash2-Werte bei gleichem Sec-Parameter auseinanderliegen.
Die verwandten Eingabewerte sind in Tabelle 1 dargestellt.

Feldname	**Länge (byte)**	**Wert**
Modifier	16	Variabel
Subnet Prefix	1	Null
Collision Count	1	Null
Public Key	20 (variabel)	Null
Extension Fields	0	

Tabelle 1: Eingabewerte in die SHA-1 Berechnung

Der Modifier wird dazu verwandt, die Eingabe der Hash-Funktion anzupassen, damit das gewünschte Ergebnis entsteht. Dieser sollte im Produktiveinsatz zufällig gewählt werden, in diesem Experiment beginnt er jedoch mit dem Wert Null und wird bei jeder Iteration um 1 erhöht. Das Subnet Prefix sowie der Collision Count werden ebenfalls auf Null gesetzt. Die Länge des Public Key-Feldes ist variabel, da der Key als DER-enkodierte [18] ASN.1-Struktur [8] dort abgelegt wird. Testhalber wird hier ein 20 Byte langer Null-Schlüssel abgelegt. Die Extension Fields sind optional und werden nicht benutzt.

Ein Testprogramm berechnet für alle möglichen Modifier-Werte den SHA-1 Wert, nimmt die ersten 112 Bits, legt die Sec-Maske darüber und überprüft, ob der Wert gleich Null ist. Ist dies der Fall, so ist ein gültiger Modifier gefunden. Das Testprogramm gibt dafür die Modifierveränderung zwischen zwei gültigen SHA-1-Werten (Dist), die Mittlere Modifierveränderung (Mean) und den resultierenden SHA-1-Wert aus:

```
CGA-Tests
=========
sec = 0xFFFF0000
OK 0x00000000000000000000000000000A29   2601
   Dist:   2601 Mean:  2601
   SHA1: 0000F444 8762DF71 CE144BBC D9B23EE3 007081D1
OK 0x00000000000000000000000000000A41C  42012
   Dist:  39411 Mean: 21006
   SHA1: 00007F4E ECAF1038 1E495D4E 7F2D8ECE 079BFF92
OK 0x000000000000000000000000000301BD 197053
   Dist: 155041 Mean: 65684
   SHA1: 00009DDA 45CA3576 5DD20639 C5589888 E64216ED
```

Die Ergebnisse sind in Tabelle 2 dargestellt. Ist der Sec-Parameter auf Null gesetzt, darf jeder Hash-Wert benutzt werden. Damit entspricht der Gesamtaufwand selbst bei zufällig gewähltem Modifier lediglich dem von zwei SHA-1 Hash-Berechnungen.
Hash-Werte für Sec gleich 1 können noch mit geringem Rechenaufwand gebildet werden. Zwei gültige, aufeinanderfolgende Werte können dabei nach durchschnittlich 66.113 Versuchen gefunden werden. Der Aufwand, zusätzlich den Hash1-Wert zu bilden, ist dabei vernachlässigbar. Schwieriger wird es bei einem Sec-Wert von 2. Hierbei kann ein gültiger Hash-Wert lediglich

Sec	Durchschnittliche Modifierveränderung	Standardabweichung
0	1	0
1	66.113	65.747
2	2.591.220.608	2.590.985.113

Tabelle 2: Versuchsergebnisse der CGA-Generierung

noch alle etwa 2,6 Millionen Versuche gefunden werden. Da die gefundenen Zahlen Durchschnittswerte darstellen, ist es auch möglich, wesentlich schneller oder auch langsamer zu einem gültigen Wert zu gelangen. Dazu ist in der Tabelle auch die Standardabweichung angegeben.

Der Versuch offenbart eine signifikante Schwäche des CGA-Verfahrens: Der nötige Aufwand zum Generieren einer CGA steigt mit der Sicherheitsstufe (Sec). Diese Parametrierbarkeit der Sicherheitsstufe wurde vorgesehen, um Attacken durch schneller werdene Computer entgegenzuwirken. Sollte z.B. Sec gleich 2 zukünftig leicht zu errechnen sein, so wird Sec einfach erhöht, um dem engegenzuwirken.

Mobile Geräte sind meist mit energiesparenden Prozessoren ausgestattet. Daher können sie diese komplexen Aufgaben nicht in annehmbarer Zeit berechnen. Dieser Nachteil könnte jedoch mit Spezialhardware (FPGAs) ausgeglichen werden. Alternativ können die Hash-Werte bereits im Voraus berechnet oder auf andere, schnellere Computer auslagert werden.

Auf der Empfängerseite entspricht der Aufwand übrigens dem zweier SHA-1-Hashwertbildungen, unabhängig von der Wahl des Security-Parameters Sec. Auch ist die Berechnung nur abhängig von der Länge des vom MN benutzen Public Keys.

5 Zusammenfassung

Ein mit dem Internet verbundener Computer ist gezielten Attacken ausgesetzt. Die beim Binding Update auftretende Man in the Middle-Attacke wurde untersucht und mögliche Lösungen dargestellt. Mit der Return Routability Procedure besteht ein effektives Protokoll, um Man in the Middle-Attacken entgegenzuwirken. Da dafür allerdings viele Pakete übertragen werden müssen, ist das Enhanced Route Optimization-Verfahren wesentlich effektiver beim Handover. Dieses nutzt das CGA-Verfahren, bei dem ein Vertrauensverhältnis zwischen den Kommunikationspartnern etabliert wird. Hierzu wird ein kryptographisches Merkmal im Interface Identfier hinterlegt, mit dessen Hilfe authentifizierte Daten wie z.B. das Binding Update übertragen werden können. Nachteilig ist der hohe Rechenaufwand, CGAs zu generieren. Hierzu wurde der Aufwand zum Berechnen von CGAs für verschiedene Werte des Security-Parameters *Sec* experimentell bestimmt. Geräte mit geringer Rechenleistung können die Berechnung alternativ im Vorwege auf andere Rechner auslagern.

6 Ausblick

Durch den Einsatz von CGAs ist es möglich, vorher unbekannten Empfängern mittels eines selbstkonsistenten IPv6-Pakets authentifizierte Nachrichten zu schicken. Ebenso ist eine Authentifizierung gegenüber der Infrastuktur (Routern) möglich. Dies ist besonders interessant für Sendermobilität bei Multicastübertragungen. Dort gibt es weder einen Rückkanal von jedem Empfänger, noch bestehen Vertrauensverhältnisse zu den Empfängern. CGAs können dabei genutzt werden, einen Multicast-Sender auch nach einem Handover gegenüber den Empfängern eindeutig zu identifizieren.

In der folgenden Masterarbeit soll das Protokoll für die Authentifizierung gegenüber Kommunikationspartnern und Infrastruktur definiert werden. Weiterhin soll die Effizienz der Handover, die Paketverzögerungen und -verluste, sowie die Effektivität des „Tree Morphings“ (vgl. [2]) untersucht werden.

Literatur

[1] T. C. Schmidt, M. Wählisch: **Predictive versus Reactive - Analysis of Handover Performance and Its Implications on IPv6 and Multicast Mobility**
Telecommunication Systems 30:1/2/3, 123-142, Springer 2005
http://users.informatik.haw-hamburg.de/~schmidt/papers/telesys05.pdf

[2] T. C. Schmidt, M. Wählisch: **Morphing Distribution Trees - On the Evolution of Multicast States under Mobility and an Adaptive Routing Scheme for Mobile SSM Sources**
Telecommunication Systems 33:1/2/3, 131-154, Springer 2006
http://users.informatik.haw-hamburg.de/~schmidt/papers/sw-mdtem-06.pdf

[3] C. Voigt: **A Comprehensive Delay Analysis for Reactive and Proactive Handoffs with Mobile IPv6 Route Optimization**
Telematics Technical Reports, Institute of Telematics, University of Karlsruhe, 2006 *http://doc.tm.uka.de/2006/vogt-2006-delay-analysis-for-reactive-and-proactive-handoffs.pdf*

[4] T. Dierks, C. Allen: **RFC 2246: The TLS Protocol Version 1.0**
http://www.ietf.org/rfc/rfc2246.txt

[5] S. Deering, R. Hinden: **RFC 2460: Internet Protocol, Version 6 (IPv6) Specification**
http://www.ietf.org/rfc/rfc2460.txt

[6] S. Thomson, T. Narten: **RFC 2462: IPv6 Stateless Address Autoconfiguration**
http://www.ietf.org/rfc/rfc2462.txt

[7] S. Chokhani, W. Ford: **RFC 2527: Internet X.509 Public Key Infrastructure - Certificate Policy and Certification Practices Framework**
http://www.ietf.org/rfc/rfc2527.txt

[8] R. Housley, W. Polk, W. Ford, D. Solo: **RFC 3280: Internet X.509 Public Key Infrastructure**
http://www.ietf.org/rfc/rfc3280.txt

[9] D. Johnson, C. Perkins, J. Arkko: **RFC 3775: Mobility Support in IPv6**
http://www.ietf.org/rfc/rfc3775.txt

[10] J. Arkko, V. Devarapalli, F. Dupont: **RFC 3776: Using IPsec to Protect Mobile IPv6 Signaling Between Mobile Nodes and Home Agents**
http://www.ietf.org/rfc/rfc3776.txt

[11] J. Arkko, J. Kempf, B. Zill, P. Nikander: **RFC 3971: SEcure Neighbor Discovery (SEND)**
http://www.ietf.org/rfc/rfc3971.txt

[12] T. Aura: **RFC 3972: Cryptographically Generated Addresses (CGA)**
http://www.ietf.org/rfc/rfc3972.txt

[13] R. Hinden, S. Deering: **RFC 4291: IP Version 6 Addressing Architecture**
http://www.ietf.org/rfc/rfc4291.txt

[14] S. Kent, K. Seo: **RFC 4301: Security Architecture for the Internet Protocol**
http://www.ietf.org/rfc/rfc4301.txt

[15] S. Kent: **RFC 4302: IP Authentication Header**
http://www.ietf.org/rfc/rfc4302.txt

[16] J. Arkko, C. Vogt, W. Haddad: **Network Working Group: Internet-Draft, 2006: Enhanced Route Optimization for Mobile IPv6**
http://www.ietf.org/internet-drafts/draft-ietf-mipshop-cga-cba-02.txt

[17] Federal Information Processing Standards: **Publication 180-2: Secure Hash Standard**
http://csrc.nist.gov/publications/fips/fips180-2/fips180-2withchangenotice.pdf

[18] International Telecommunication Union (ITU): **X.690**
http://www.itu.int/ITU-T/studygroups/com17/languages/X.690-0207.pdf

[19] H. Soliman: **Mobile IPv6 - Mobility in a Wireless Internet**
Addison-Wesley (2004). ISBN 0-201-78897-7

[20] O. Christ: **Pflichtenheft AW - INF-M3 Technische Informatik (SS 2006), HAW Hamburg**

Abbildungsverzeichnis

Tabellenverzeichnis

A Glossar

AH	Authentication Header
ARP	Address Resolution Protocol
ASN.1	Abstract Syntax Notation One
Back	Binding Update Acknowledgement
BU	Binding Update
CA	Certificate Authority
CGA	Cryptographically Generated Address
CN	Correspondent Node
CoA	Care of Address
CoT	Care of Test
CoTi	Care of Test Init
CRL	Certificate Revocation List
CToken	Care of Keygen Token
DAD	Duplicate Address Detection
DER	Distinguished Encoding Rules
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Service
DoS	Denial of Service
FPGA	Field Programmable Gate Array
HA	Home Agent
HoA	Home Address
HoT	Home Test
HoTi	Home Test Init
HToken	Home Keygen Token
IP	Internet Protocol
IPsec	Internet Protocol security
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
MIPv6	Mobile IPv6
MitM	Man in the Middle

MN	Mobile Node
PKI	Public Key Infrastructure
RRP	Return Routability Procedure
RSA	Rivest, Shamir, Adleman
Sec	Security Parameter
SEND	Secure Neighbor Discovery
SHA-1	Secure Hash Algorithm 1
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
WLAN	Wireless Local Area Network