Nutzung von Kryptographie im Zusammenhang mit Firewalls

Studie des DFN-FWL

Carsten Benecke und Uwe Ellermann
Inhaltsverzeichnis

1 Einleitung

2 Firewalls

2.1 Firewall-Komponenten

2.1.1 Packet Screen

2.1.2 Proxy-Server

2.1.3 Anwendungs-Gateway

2.1.4 Weitere Firewall-Komponenten

2.2 Firewall-Architekturen

2.2.1 Packet Screens

2.2.2 Kombinationen von Packet Screen und Bastion

2.2.3 Gateway-Firewalls

2.2.4 Firewall-Ergänzungen

2.3 Nicht durch Firewalls abgedeckte Sicherheitsprobleme

3 Kryptographische Verfahren und Protokolle

3.1 Informale Definition: Kryptographie und verwandte Begriffe

3.2 Kryptographie für die Netzwerksicherheit

3.3 Kryptographische Verfahren und Algorithmen

3.3.1 Algorithmen mit symmetrischen Schlüsseln

3.3.2 Algorithmen mit asymmetrischen Schlüsseln

3.3.3 Bitstrom-basierte und Block-basierte Algorithmen

3.3.4 Weitere elementare Funktionen für kryptographische Protokolle

3.4 Informale Definition: Kryptographische Protokolle

3.5 Infrastruktur für die Schlüsselverteilung

3.6 Protokolle für das Schlüsselmanagement

3.6.1 Protokolle basierend auf Diffie-Hellman

3.6.2 Protokolle basierend auf KDC
4 Kryptographische Protokolle in der TCP/IP-Familie

4.1 Protokolle unterhalb von IP ........................................................................ 36

4.2 Protokolle auf der IP-Ebene ...................................................................... 39

4.2.1 IPsec ................................................................................................. 40

4.2.2 SKIP ................................................................................................. 46

4.3 Protokolle auf der Transportebene ............................................................ 50

4.3.1 Das TLS Record Protocol ................................................................. 52

4.3.2 Das TLS Handshake Protocol ......................................................... 53

4.4 Protokolle auf der Anwendungsebene ....................................................... 56

4.4.1 Sicheres HTTP (S-HTTP) ............................................................... 56

4.4.2 Sicheres "Remote"-Login ................................................................. 57

4.4.3 Sicheres "Domain Name System" (Secure DNS) ............................... 61

4.4.4 ISAKMP/OAKLEY ............................................................................ 63

4.4.5 PKI-Protokolle ................................................................................. 64

4.5 Anwendungen außerhalb des Protokollstapels ........................................ 65

4.6 Nicht durch kryptographische Protokolle abgedeckte Sicherheitsprobleme 66

4.7 Ausblick ................................................................................................. 68

5 Vorteile durch einen kombinierten Einsatz

5.1 Nutzen kryptographischer Verfahren für Firewalls ................................. 69

5.1.1 Authentisierung ............................................................................... 69

5.1.2 Integrität übertragener Daten ......................................................... 72

5.1.3 Erweiterte Zugriffskontrolle ............................................................ 72

5.1.4 Fernadministration von Firewalls .................................................... 72

5.2 Nutzen von Firewalls für kryptographische Verfahren .......................... 73

5.2.1 Firewalls für offene Probleme ......................................................... 73

5.2.2 Durchsetzen von Verschlüsselungsrichtlinien am Firewall ................. 75

5.2.3 Firewall als Verschlüsselungs-Gateway ......................................... 75

5.2.4 Zertifikatskontrolle im Firewall ....................................................... 76

5.2.5 Anonymisierung der Kommunikation mit einem Firewall .................. 78
6 Wechselwirkungen des kombinierten Einsatzes
6.1 Technische Aspekte des kombinierten Einsatzes
6.1.1 Packet Screens
6.1.2 Proxies und Anwendungs-Gateways
6.2 Performanz bei kombiniertem Einsatz

7 Zusammenfassung
Abkürzungsverzeichnis

Literatur
<table>
<thead>
<tr>
<th>Abbildungsverzeichnis</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
<tr>
<td>12</td>
</tr>
<tr>
<td>13</td>
</tr>
<tr>
<td>14</td>
</tr>
<tr>
<td>15</td>
</tr>
<tr>
<td>16</td>
</tr>
<tr>
<td>17</td>
</tr>
<tr>
<td>18</td>
</tr>
<tr>
<td>19</td>
</tr>
<tr>
<td>20</td>
</tr>
<tr>
<td>21</td>
</tr>
<tr>
<td>22</td>
</tr>
<tr>
<td>23</td>
</tr>
<tr>
<td>24</td>
</tr>
<tr>
<td>25</td>
</tr>
</tbody>
</table>
1 Einleitung


Im zweiten Teil der Studie werden die Vorteile des kombinierten Einsatzes von Firewalls und kryptographischer Verfahren dargestellt (vgl. Abschnitt 5). Neben Lösungen für viele der diskutierten offenen Probleme werden auch völlig neue Möglichkeiten erläutert, die erst durch den gleichzeitigen Einsatz der Sicherungsmechanismen aus beiden Bereichen denkbar sind. Dazu wird wiederum unabhängig voneinander dargestellt, wie einerseits Firewalls von zusätzlichen kryptographischen Mechanismen profitieren und andererseits die Zugriffskontrolle nutzbringend in Netzen verwendet werden kann, in denen ausschließlich kryptographische Schutzmechanismen verwendet werden.


In der Zusammenfassung wird schließlich auf offene Probleme eingegangen und es wird ein Ausblick auf zukünftige Untersuchungen gegeben.

2 Firewalls

Firewalls sind Zugriffskontrollmechanismen in Netzen. Sie werden eingesetzt, um eine Trennung der Netze nach Sicherheitsgesichtspunkten zu erreichen. Üblicherweise wird durch einen Firewall ein internes, sicherheitskritisches Netz vor Angriffen aus einem externen Netz...

**Firewall:** "Ein Firewall ist eine Schwelle zwischen zwei Netzen, die überwunden werden muß, um Systeme im jeweils anderen Netz zu erreichen. Es wird dafür gesorgt, daß jede Kommunikation zwischen den beiden Netzen über den Firewall geführt werden muß. Auf dem Firewall sorgen Zugriffskontrolle und Audit dafür, daß das Prinzip der geringsten Berechtigung durchgesetzt wird und potentielle Angriffe schnellstmöglich erkannt werden."

Zunächst wurden Firewalls primär zur sicheren Anbindung von Unternehmensnetzen an das (öffentliche) Internet eingesetzt. Neben diesen “Internet-Firewalls” ist heute die Absicherung besonders sensitiver Abteilungen innerhalb eines Unternehmensnetzes ein wichtiger Einsatzbereich von Firewalls ("Intranet-Firewalls"). Diese beiden Einsatzbereiche unterscheiden sich durch:

**Kommunikationsanforderungen:** Zwischen verschiedenen Teilen eines Unternehmensnetzes bestehen in der Regel engere Kommunikationsbeziehungen als mit dem Internet. So ist meist der Zugriff vom abgesicherten Netz auf Datenbanken im Unternehmensnetz notwendig.

Da das durch einen Intranet-Firewall abgetrennte Netz oft nur wenige Systeme umfaßt, ist der Aufwand zur Absicherung selten genutzter Dienste kaum zu rechtfertigen. Die Liste der zu unterstützenden Protokolle wird sich daher für Internet- und Intranet-Firewalls unterscheiden.

**Sicherheitsanforderungen:** Intranet-Firewalls sind einer höheren Bedrohung ausgesetzt, da die zu schützenden Systeme besonders kritische Informationen enthalten. Auch der Kreis der potentiellen Angreifer unterscheidet sich, da primär mit Angriffen durch firmeninterne Mitarbeiter gerechnet wird. Externe Angreifer werden bereits durch den Internet-Firewall erkannt und aufgehalten.

**Performanzanforderungen:** Hochgeschwindigkeitsnetze werden innerhalb von Unternehmen früher eingesetzt als bei der Internet-Anbindung. Intranet-Firewalls bilden daher früher einen Engpaß.

---

1. Für das durch den Firewall geschützte Netz wird im folgenden die Bezeichnung “internes Netz” verwendet. Das Netz auf der anderen Seite des Firewalls wird als “externes Netz” bezeichnet.

2.1 Firewall-Komponenten


2.1.1 Packet Screen


---


4 Bei fragmentierten Dateneinheiten sind die Transport-Header nur im ersten Fragment enthalten, so daß Packet Screens folgende Fragmente (Offset > 0) ohne vollständige Überprüfung weiterleiten können. Durch nachfolgende Fragmente mit einem kleinen Offset können beim Empfänger Teile des im ersten Fragment enthaltenen Transport-Headers überschrieben werden.

**Dynamische Packet Screens:** Im Gegensatz zu den bisher beschriebenen “Statischen Packet Screens”, bei denen jede eingehende Dateneinheit nach einem festgelegten Satz von Filterregeln unabhängig von bereits verarbeiteten Dateneinheiten ausgewertet wird, kann bei “Dynamischen Packet Screens” die Filterentscheidung vom Kontext bereits empfangener Dateneinheiten abhängen.

Die Möglichkeit, Filterentscheidungen anhand des Kontextes bereits empfangener Dateneinheiten durchführen zu können, hat im Wesentlichen zwei Vorteile:

---


6 Gelegentlich wird auch die Bezeichnung “Stateful Packetfilter” verwendet.
2.1 Firewall-Komponenten

- Da bei UDP kein Verbindungsaufbau durchgeführt wird, kann die Richtung des Zugriffs nicht wie bei TCP über das SYN-Flag festgestellt werden. Ohne Kenntnis vorangegangener Dateneinheiten kann nicht entschieden werden, ob es sich bei einem empfangenen UDP-Datagramm um eine Reaktion (Antwort) oder um einen neuen Zugriff handelt. Bei einer “Dynamischen Packet Screen” wird eine Filterregel ergänzt, die Antworten auf übertragene UDP-Datagramme ebenfalls passieren läßt. Nach längerer Inaktivität dieser Kommunikationsbeziehung wird die Filterregel wieder entfernt.


Vorteile einer Packet Screen:


Performanz: Der Filter-Aufwand der Packet Screen ist relativ gering, so daß eine gute Performanz erreicht wird. Die Performanzauswirkungen der Packet Screen werden vom Benutzer nicht bemerkt.

Geringer Installationsaufwand: Die Packet Screen Funktionalität ist in heutigen Routern meist schon vorhanden, so daß nur die Definition geeigneter Filterregeln notwendig ist. Die Konfiguration erfordert nur einen geringen Zeitaufwand.

Kostengünstig: Da Packet Screens meist im Router schon verfügbar sind, ist dank des geringen Zeitaufwands zur Konfiguration der Filterregeln, der Einsatz einer Packet Screen relativ kostengünstig. Wenn kein vorhandener Router als Packet Screen verwendet werden kann, bietet frei verfügbare Packet Screen Software (z. B. “Drawbridge” oder “IP-Filter”), die auf einfacher PC-Hardware eingesetzt werden kann, eine ebenfalls ko-

---

7 Eine Authentisierung von Benutzern wäre zukünftig mit Hilfe von IPSEC denkbar [14], die Verfügbarkeit von entsprechenden Packet Screen Implementationen läßt sich heute noch nicht absehen.

8 In Hochgeschwindigkeitsnetzen kann eine Packet Screen jedoch zum Engpaß werden, wie die Analysen in [28] zeigen.

9 ftp://ftp.cert.dfn.de/pub/firewalls/software/drawbridge/

stengünstige Alternative. Kommerzielle Lösungen wie die dynamische Packet Screen “Firewall-1” sind deutlich teurer.

Nachteile einer Packet Screen:


2.1.2 Proxy-Server


**Granularität:** Mit Proxies ist eine sehr detaillierte Zugriffskontrolle möglich. Die Kontrolle kann abhängig von Host-Adressen, vom Dienst, vom Benutzer, von zeitlichen Beschränkungen, oder vom Kontext des Zugriffs erfolgen. Insbesondere stehen der Zugriffskontrolle alle zwischen Client und Server ausgetauschten Daten zur Verfügung.

**Vorteile eines Proxies:**

**Feine Granularität:** Größter Vorteil von Proxies ist die detaillierte Zugriffskontrolle, durch die auch ein sehr gutes Audit möglich wird (s. o.).

**Authentisierung:** Es ist eine einmalige Benutzerauthentisierung beim Verbindungsaufbau oder eine wiederholte Authentisierung für jede Anfrage möglich.
Nachteile eines Proxies:


2.1.3 Anwendungs-Gateway

Bei Protokollen, die im Stapelbetrieb (“Batch-Mode”) verarbeitet werden, wie zum Beispiel SMTP (E-Mail) und NNTP (Netnews), ist es vorteilhaft, Zugriffe nicht über einen Proxy an einen Server weiterzuleiten, sondern auf dem Firewall bereits wesentliche Teile der Serverfunktionalität zu realisieren.


Vorteile von Anwendungs-Gateways: Zusätzlich zu den Vorteilen von Proxies haben Anwendungs-Gateways folgende Vorteile:

Verdecken der internen Struktur: Durch Erbringen wesentlicher Serverfunktionen kann die interne Struktur des durch den Firewall geschützten Netzes verdeckt werden.
2.2 Firewall-Architekturen


Nachteile von Anwendungs-Gateways: Zusätzlich zu den Nachteilen von Proxies haben Anwendungs-Gateways folgende Nachteile:

Keine Benutzerinteraktion zur Authentisierung: Stapelorientierte Anwendungen schränken die Liste der einsetzbaren Authentisierungsverfahren ein, da eine Benutzerinteraktion zur Authentisierung nicht möglich ist.

2.1.4 Weitere Firewall-Komponenten

Audit: Die von den Zugriffskontrollkomponenten gemeldeten Ereignisse müssen aufgezeichnet und ausgewertet werden. Um Angriffe schnell erkennen zu können, ist eine maschinelle Auswertung der aufgezeichneten Daten notwendig.

Verschlüsselung: In zunehmendem Maße werden Verschlüsselungsverfahren in Firewall-Konzepte integriert. Neben den bereits eingesetzten “Virtual Private Networks” (VPNs) werden zukünftig auch in Proxies und Anwendungs-Gateways kryptographische Verfahren zur Verschlüsselung der übertragenen Daten oder zur Authentisierung angewendet werden.


2.2 Firewall-Architekturen

Firewall-Architekturen werden aus Firewall-Komponenten aufgebaut. Aus der Vielzahl möglicher Kombinationen werden im Folgenden die wichtigsten Architekturen mit ihren Vor- und Nachteilen kurz vorgestellt.
2.2.1 Packet Screens

Die Konfiguration einer Packet Screen stellt die einfachste Firewall-Architektur dar. Trotz der bereits diskutierten Einschränkungen (vgl. Abschnitt 2.1.1) bei der Granularität der Zugriffs kontrolle und beim Audit kann die Konfiguration einer Packet Screen ausreichen, wenn die erkannten Defizite durch zusätzliche Maßnahmen kompensiert werden (vgl. [13]).

Da die Möglichkeit, Filterregeln zu definieren, bereits in allen gängigen Routern enthalten ist und ein Router zudem zur Anbindung eines Netzes an das Internet oder an ein größeres Firmennetz ohnehin benötigt wird, ist die Packet Screen eine kostengünstige Lösung. Die konzeptionelle Sicht einer Firewall-Architektur auf Basis einer Packet Screen ist in Abbildung 3 wiedergegeben.

![Abbildung 3: Packet Screen: konzeptionelle Sicht](image)

Zur Erstellung der Filterregeln können zwei unterschiedliche Ansätze gewählt werden:

**Verbotsregeln:** Über eine Konfigurierung von Verbotsregeln für als gefährlich eingestufte Protokolle können Angriffe an der Packet Screen gestoppt werden. Da alle anderen Dienste von diesen Regeln nicht betroffen sind, bleiben bei diesem Ansatz viele Dienste erreichbar. Für Benutzer hat dies den Vorteil, daß ihre Kommunikation nur selten durch die Filterregeln unterbunden wird. Allerdings bleiben so auch viele Dienste für Angriffsversuche geöffnet, so daß dieser Ansatz nur als “Ad-hoc Lösung” bei konkreten Angriffen auf einzelne Dienste gewählt werden sollte.


Eine wichtige Aufgabe von Packet Screens ist der Schutz gegen IP-Spoofing Angriffe. Durch einfache Plausibilitätsüberprüfungen kann die Packet Screen feststellen, ob die Absender-Adresse einer eingegangenen Dateneinheit zu einem Endsystem gehört, das über die jeweilige Schnittstelle erreichbar ist. [16]
2.2 Firewall-Architekturen


Neben den grundsätzlichen Vor- und Nachteilen der Firewall-Komponente “Packet Screen” (s. Abschnitt 2.1.1) werden abschließend Aspekte vorgestellt, die aus dem Einsatz einer Packet Screen als Firewall-Architektur resultieren.

Vorteile bei ausschließlichem Einsatz einer Packet Screen:


Nachteile bei ausschließlichem Einsatz einer Packet Screen:


Granularität: Die Überprüfung von Protokoll-Headern reicht für die Abwehr vieler Angriffe nicht aus.
2.2.2 **Kombinationen von Packet Screen und Bastion**


Abbildung 4: Kombination von Packet Screen und Bastion: konzeptionelle Sicht – Bastion innen

Abbildung 5: Kombination von Packet Screen und Bastion: konzeptionelle Sicht – Bastion mittig

Bei der Kombination von Packet Screen und Bastion wird abhängig von der Position der Bastion zwischen vier Konzepten unterschieden: (siehe auch [2])
Abbildung 6: Kombination von Packet Screen und Bastion: konzeptionelle Sicht – Bastion außen

Abbildung 7: Kombination von Packet Screen und Bastion: konzeptionelle Sicht – beidseitig abgeschirmte Bastion

**Bastion innen** (Abbildung 5)

Die Installation der Bastion im internen Netz ist die am einfachsten zu realisierende Lösung. Jedoch besteht hierbei zwischen internem Netz und der Bastion keine weitere Abschirmung. Angriffe auf die Bastion, die von internen Systemen ausgehen, müssen in diesem Fall von der Bastion allein abgewehrt werden.

**Bastion mittig** (Abbildung 6)

Durch die Installation der Bastion in einem eigenen Netz an einer weiteren Schnittstelle der Packet Screen kann der Zugriff auf die Bastion vom externen und vom internen Netz besser eingeschränkt werden.

**Bastion außen** (Abbildung 7)

**Bastion beidseitig abgeschirmt** (Abbildung \(\text{Abb. 7}\))

Bei der vorhergehend beschriebenen Installation der Bastion außen werden zwei Router benötigt, von denen aber nur einer als Packet Screen verwendet wird. Der zum externen Netz hin installierte Router ist in den meisten Fällen zusätzlich notwendig, um die Anbindung an das Internet technisch zu realisieren. Aus diesem Grund liegt es nahe, auch den zum Internet hin installierten Router als Packet Screen zu nutzen. Es ergibt sich dann die in Abbildung \(\text{Abb. 7}\) vorgestellte Konfiguration mit einer beidseitig abgeschirnten Bastion. Das in beide Richtungen hin abgeschirmte Subnetz, an dem die Bastion installiert ist, wird als Grenznetz (“Screened Subnet”, “Demilitarized Zone” oder “DMZ”) bezeichnet.


**Vorteile der Kombination von Packet Screen und Bastion:**

**Granularität:** Durch die Kombination von Packet Screen und Bastion kann eine sehr feine Granularität bei Zugriffskontrolle und Audit erreicht werden. Insgesamt kann ein sehr hohes Sicherheitsniveau erreicht werden.

**Erweiterbarkeit:** Durch Einsatz zusätzlicher Bastionen kann diese Lösung bei Bedarf an höhere Leistungsanforderungen angepaßt werden.

**Nachteile der Kombination von Packet Screen und Bastion:**

**Kosten:** Zur Realisierung diese Lösung ist zunächst ein hoher Installationsaufwand notwendig. Während des Betriebs können durch niedrigeren Aufwand bei Pflege und Überwachung der internen Systeme Kosten eingespart werden.

2.2.3 **Gateway-Firewalls**

Ein Gateway ist ein Rechner, der an mindestens zwei Netze angeschlossen ist und der auf Anwendungssebene eine Kommunikation zwischen beiden Netzen ermöglicht. Durch Erweiterungen um Zugriffskontrolle und Audit zusammen mit einer besonderen Absicherung des Gateways ist der Einsatz als Firewall möglich (vgl. Abb. \(\text{Abb. 8}\)). Dabei muß sichergestellt werden, daß alle Kommunikationsbeziehungen zwischen den angeschlossenen Netzen nur über


Vorteile von Gateway-Firewalls:

Granulärität: Durch einen Gateway-Firewall kann eine sehr feine Granularität bei Zugriffs kontrolle und Audit erreicht werden. Insgesamt wird ein sehr hohes Sicherheitsniveau erreicht.

Viele kommerzielle Lösungen: Für Gateway-Firewalls gibt es viele Anbieter.

Nachteile von Gateway-Firewalls:


Erweiterbarkeit: Im Gegensatz zu Kombinationen von Packet Screen und Bastion besteht keine Erweiterungsmöglichkeit durch den Einsatz zusätzlicher Bastionen.

Ein Gateway-Firewall ist per Definition eine Bastion.
2.2.4 Firewall-Ergänzungen

Nach der Vorstellung der Firewall-Architekturen sollen in diesem Abschnitt die wichtigsten Ergänzungen und Varianten kurz angesprochen werden.

**Informationsdienste:** Zusätzlich zur Absicherung des internen Netzes wird von Firewall-Konzepten oft auch die sichere Integration von Informationsdiensten (Anonymous-FTP, WWW usw.) gefordert. Diese Dienste, die meist eingesetzt werden, um öffentliche Informationen für das externe Netz bereitzustellen, haben andere Sicherheits- und Kommunikationsanforderungen als die Systeme im internen Netz. Um eine Gefährdung für das interne Netz durch die Informationsdienste ausschließen zu können, ist das Bereitstellen der Informationsdienste durch Systeme im internen Netz oder auf dem Firewall selbst nicht akzeptabel. Besser ist es, die Informationsdienste außerhalb des Firewalls anzubieten. Da der Informationsserver so nicht mehr durch den Firewall geschützt werden kann, erfordert diese Lösung jedoch eine zusätzliche Absicherung des Informationsservers.


Die Planung eines VPNs muß im Zusammenhang mit dem Firewall-Konzept betrachtet werden, da je nach Positionierung der beiden Konzepte zueinander unterschiedliche Sicherheitseigenschaften erreicht werden [28, S. 6f]. In jedem Fall ist zu beachten, daß die miteinander gekoppelten Netzwerke dasselbe Sicherheitsniveau erreichen müssen, da andernfalls Zugriffe aus einem weniger stark gesicherten Netz durch das VPN eine Bedrohung für besser gesicherte Netze darstellen können.

2.3 Nicht durch Firewalls abgedeckte Sicherheitsprobleme

Firewalls werden oft als “Allheilmittel” für die Netzwerksicherheit angesehen. Ein Firewall ist ein Zugriffskontrollmechanismus, der Netze in verschiedene Sicherheitsbereiche aufteilt. Um falsche Erwartungen zu verhindern, soll auch aufgeführt werden, was Firewalls nicht leisten können.

**Keine Kontrolle des internen Verkehrs:** Über den Firewall findet nur die Kommunikation zwischen internem und externem Netz statt. Der Datenverkehr zwischen internen Systemen untereinander kann durch den Firewall nicht kontrolliert werden. Für den Schutz
gegen Insider-Angriffe ist daher weiterhin eine Absicherung der internen Systeme erforderlich. In manchen Fällen kann auch eine weitere Aufteilung des internen Netzes in mehrere über (Intranet-)Firewalls getrennte Sicherheitszonen sinnvoll sein.


**Keine Zugriffskontrolle auf verschlüsselten Daten:** Der Einsatz kryptographischer Verfahren soll das Abhören oder die Manipulation von übertragenen Daten verhindern. Dadurch wird gleichzeitig auch eine Kontrolle dieser Daten durch einen Firewall verhindert.

**Begrenzte Kontrollmöglichkeiten des Firewalls:** Proxies und Anwendungs-Gateways erhalten zwar Zugriff auf alle übertragenen Daten, was eine Vielzahl von Angriffen theoretisch erkennbar macht. In der Praxis übersteigen aufwendige Überprüfungen jedoch die Leistungsfähigkeit des Firewalls.

Zur Absicherung der aufgeführten Probleme müssen andere Sicherheitsmechanismen eingesetzt werden. Über die aufgeführten Probleme hinaus kann der Firewall selbst zum Ziel von Angriffen werden, was schwerwiegende Konsequenzen haben kann, wenn die Sicherheit allein vom Firewall abhängt und die Sicherheitsmaßnahmen auf den internen Systemen vernachlässigt wurden.


### 3 Kryptographische Verfahren und Protokolle

In diesem Abschnitt wird dargestellt, was *Kryptographie* ist (vgl. Abschnitt 3.1). Es wird diskutiert, warum es notwendig ist, Kryptographie zur Sicherung von einzelnen Kommunikationsbeziehungen einzusetzen. Danach können für immer wiederkehrende Aufgabestellungen grundlegenden Algorithmen diskutiert werden (vgl. Abschnitt 3.3). Darauf aufbauende Schutzmechanismen werden wiederum in kryptographischen Protokollen eingesetzt.

---

12 IP-Spoofing Angriffe können über einfache Regeln teilweise erkannt und abgewehrt werden.

Die Wahl der Protokolle orientiert sich am TCP/IP-Protokollstapel. Es wird für jede Schicht in diesem Modell mindestens ein Protokoll vorgestellt, um zu verdeutlichen, daß in Abhängigkeit vom (Schutz)Ziel die gleichen Mechanismen auf verschiedenen Ebenen eingesetzt werden können. Gleichzeitig sind die Protokolle so ausgewählt, daß mit hoher Wahrscheinlichkeit auch die zukünftig im Internet verwendeten kryptographischen Protokolle abgedeckt werden. Es ist zur Zeit noch nicht mit Sicherheit vorab aussagbar, welche der vorgestellten Protokolle sich längerfristig durchsetzen werden. Immerhin ist es möglich anzugeben, an welchen Protokollen mit dem größten Einsatz gearbeitet wird.

3.1 Informale Definition: Kryptographie und verwandte Begriffe

Als Kryptographie bezeichnet man die Kunst und Wissenschaft, Nachrichten geheim zu halten. Das Problem, Nachrichten geheim zu halten, entsteht dadurch, daß Botschaften normalerweise gezielt an einen Empfänger gesendet werden. Dabei gilt fast immer die implizite Annahme, daß diese Botschaft ausschließlich für den Empfänger bestimmt ist.


Im Bereich der Netzwerksicherheit wird die Kryptographie zunehmend eingesetzt, um Botschaften (Nachrichten) geheim zu halten. Der besondere Nutzen der Kryptographie besteht jedoch darin, daß sie die Grundlage für eine Reihe weiterer wichtiger Eigenschaften und dazugehörender Mechanismen bildet, die im nächsten Abschnitt vorgestellt werden.
3.2 Kryptographie für die Netzwerksicherheit

Die Kryptographie hat im Bereich der Netzwerksicherheit eine wichtige Rolle eingenommen. Sie bildet die Grundlage für eine Vielzahl von Mechanismen, mit denen erfolgreich die folgenden Eigenschaften durchgesetzt werden können:

**Vertraulichkeit:** Nachrichten sollen nur von den Empfängern lesbar sein, die der Sender angibt. Allen anderen potentiellen Empfängern der Nachricht soll deren Inhalt (Information) nicht zugänglich sein.

Im Bereich der Netzwerksicherheit hat das Konzept der Vertraulichkeit eine weitere Ausprägung: Es ist unter Umständen erwünscht, daß nicht nur die Nachricht sondern auch die Kommunikationsbeziehung selbst vertraulich ist. Dritte sollen keine Erkenntnisse über den Nachrichtenaustausch zwischen Sendern und Empfängern erlangen können.

**Integrität:** Integrität bezeichnet die Eigenschaft einer Nachricht, unverfälscht zu sein. Eine Manipulation durch Dritte an einer Nachricht muß beim Empfänger erkennbar sein.

**Authentizität:** Der Empfänger einer Nachricht muß die Möglichkeit haben, die Nachricht zweifelsfrei einem Sender zuordnen zu können. Es soll nicht möglich sein, beim Senden die Herkunft einer Nachricht unerkannt zu fälschen.

**Unleugbarkeit:** Ein Sender sollte nicht die Möglichkeit haben, das Versenden einer Nachricht später zu leugnen. Empfänger müssen die Möglichkeit haben, nachweisen zu können, daß eine Nachricht von einem bestimmten Sender tatsächlich gesendet wurde. Ebenso sollte es dem Sender möglich sein, den Empfang seiner Nachricht durch den Empfänger jederzeit nachweisen zu können. Der Empfänger kann somit den Empfang einer Nachricht nicht leugnen (vgl. 'Einschreiben' bei der Briefpost).

Diese Eigenschaften kann man grundsätzlich für alle Arten der Kommunikation fordern. Beispielsweise ist auch bei der öffentlichen "Briefpost" die Vertraulichkeit der Briefsendungen (Nachrichten) gesetzlich gefordert. Das Öffnen eines Briefes durch einen Dritten ist jedoch trivial; es gibt keinen einfachen Mechanismus, der das Öffnen des Briefes durch einen Dritten verhindert, um die Vertraulichkeit durchzusetzen. Durch die Kryptographie ist man jedoch heute, insbesondere bei der rechnerbasierten Kommunikation, in der Lage, diese Eigenschaften mit Hilfe entsprechender Mechanismen durchzusetzen.\(^{13}\)

**Resümee:** Kryptographische Verfahren werden eingesetzt, um die Geheimhaltung und Integrität übertragener Daten zu gewährleisten. Außerdem ermöglichen sie eine sichere Authentisierung der Kommunikationspartner und der übertragenen Dateneinheiten.

\(^{13}\) Andererseits wird auch erst durch die Computer-basierte Kommunikation ein effizientes und für die Kommunikationspartner transparentes Abhören möglich.
3

Kryptographische Verfahren und Algorithmen


Die gängigen Verfahren lassen sich anhand der Schlüssel in zwei Kategorien einteilen:

- Algorithmen mit symmetrischen Schlüsseln ("secret-key algorithms")
- Algorithmen mit öffentlichen (asymmetrischen) Schlüsseln (‘public-key algorithms’)

Die Algorithmen beschreiben meistens zwei Funktionen: \( V(T) \) ist die Verschlüsselungsfunktion, die den Klartext \( T \) in den Geheimtext \( G \) umwandelt. Die Entschlüsselungsfunktion \( E(G) \) wird eingesetzt, um den Geheimtext in den Klartext zurückzuwandeln.

3.3.1 Algorithmen mit symmetrischen Schlüsseln

Kennzeichen der symmetrischen Algorithmen ist, daß sie denselben Schlüssel zum Ver- und Entschlüsseln einer Nachricht benutzen. Ein Klartext \( T \) wird mit dem Schlüssel \( S \) unter Verwendung der Verschlüsselungsfunktion \( V \) derart verschlüsselt, daß die Entschlüsselungsfunktion \( E \) mit demselben Schlüssel aus dem Geheimtext wieder den Klartext zurückgewinnen kann. Formal dargestellt gilt für die symmetrischen Algorithmen:

\[
\forall S \in \Sigma : V_S(T) = G, E_S(G) = T
\]

(1)

\( \Sigma \) bezeichnet die Schlüsselmenge. Die Schlüsselmenge ist in der Regel endlich. Die verschiedenen Werte, die ein Schlüssel in \( \Sigma \) annehmen kann, bezeichnet man als Schlüsselraum ("Keypace").

\[\text{Im Idealfall hängt die Sicherheit eines Verfahrens ausschließlich vom gewählten Schlüssel ab. Eine Disziplin der Kryptoanalyse besteht darin, Schwächen eines Algorithmus aufzudecken, um unabhängig vom Schlüssel den Geheimtext in Klartext umzuwandeln. Der Vorteil der öffentlichen Algorithmen besteht gerade darin, daß sich ein öffentlicher Algorithmus bei einer eingehenden Kryptoanalyse “bewähren” kann. Die Güte eines öffentlichen Algorithmus ist somit bewertbar.}\]
Die Hauptvorteile von Algorithmen mit symmetrischen Schlüsseln sind:

- Die Komplexität dieser Algorithmen ist meist geringer als vergleichbar sichere asymmetrische Algorithmen. Daher erreicht man mit symmetrischen Verfahren im allgemeinen sehr viel höhere Ver- und Entschlüsselungsgeschwindigkeiten.

- Die Schlüssellänge zum Erreichen eines bestimmten Sicherheitsniveaus ist bei symmetrischen Algorithmen deutlich kleiner als bei vergleichbar sicheren asymmetrischen Verfahren. Dadurch sind symmetrische Verfahren auch gut in Bereichen einsetzbar, bei denen das Schlüsselmanagement unter “Platzproblemen” leidet.

Der größte Nachteil dieser Methode liegt im Aufwand für die Schlüsselverwaltung:

1. Bei $n$ Kommunikationspartnern, die alle untereinander verschlüsselte Nachrichten austauschen wollen, muß jeder der $n$ Teilnehmer $n-1$ symmetrische Schlüssel bereithalten, falls er eine Nachricht von einem der anderen Teilnehmer bekommt.


### 3.3.2 Algorithmen mit asymmetrischen Schlüsseln


Eine wesentliche Anforderung an diese Verfahren ist, daß der öffentliche Schlüssel keine Berechnung des privaten Schlüssels ermöglicht.

Analog zu Formel (1) gilt Formel (2) für die asymmetrischen Verfahren. $SV$ bezeichnet den öffentlichen Schlüssel zum Verschlüsseln. $SE$ bezeichnet den privaten Schlüssel zum Entschlüsseln der Nachricht.

$$ V_{SV}(T) = G, \ E_{SE}(G) = T $$.  

Die Hauptvorteile dieser Algorithmen sind:

---

15 Dieses Problem läßt sich durch Schlüsselmanagement-Protokolle lösen (s.u.)
16 In fast allen Fällen gibt es bei asymmetrischen Verfahren zwei verschiedene Schlüssel. Es gibt auch Verfahren, die mehrere Schlüssel verwenden.

Es wird kein sicherer Kanal für die Schlüsselverteilung benötigt. Alle Empfänger stellen ihre öffentlichen Schlüssel jedem zur Verfügung, der eine verschlüsselte Nachricht senden möchte.

Die Hauptschwächen dieser Algorithmen sind:

- Die Komplexität ist höher als bei vergleichbaren symmetrischen Verfahren. Es werden in der Regel geringere Ver- und Entschlüsselungsgeschwindigkeiten erreicht als bei symmetrischen Verfahren.

- Die Schlüssellänge muß im Vergleich zu der Schlüssellänge von symmetrischen Algorithmen deutlich größer sein, um ein vergleichbares Sicherheitsniveau zu erreichen.

- Das Problem der Geheimhaltung eines symmetrischen Schlüssels verlagert sich bei den asymmetrischen Algorithmen auf das Problem, die Authentizität des Schlüssels festzustellen.\(^\text{17}\)

### 3.3.3 Bitstrom-basierte und Block-basierte Algorithmen

Ein weiteres wichtiges Unterscheidungsmerkmal der Algorithmen ist, ob der Text (Klartext oder Geheimtext) bei den Ver- und Entschlüsselungsoperationen Bit für Bit verarbeitet wird oder ob mehrere Bits zusammen als “Block” verarbeitet werden. Im ersten Fall spricht man von einem Bitstrom-basierten Algorithmus, auch Bitstrom-Chiffre genannt. Im zweiten Fall spricht man von einem Block-basierten Algorithmus oder auch von einem Block-Chiffre.

Die kryptographischen Operationen, die der jeweilige Algorithmus ausführt, werden bei Bitstrom-basierten Algorithmen auf jedem einzelnen Bit des Textes ausgeführt. Bei den Bitstrom-basierten Algorithmen ist daher die Textlänge beliebig. Da die Block-basierten Algorithmen immer mehrere Bits auf einmal verarbeiten, müssen die Texte in der Regel ein Vielfaches der Blockgröße lang sein. Dazu sind die Texte vor dem Verschlüsseln gegebenenfalls durch Anhängen von weiteren Zeichen an die Blocklänge anzupassen. Typische Blocklängen orientieren sich in der Regel an der Größe verbreiteter CPU-Register. So sind beispielsweise die Blocklängen 64 Bit und 128 Bit gebräuchlich.

\(^{17}\) Einfach dargestellt bedeutet die Veröffentlichung eines Schlüssels mit dem Bezeichner “Schlüssel der Person X” nicht, daß dieser Schlüssel von Person X tatsächlich generiert wurde. Die Verknüpfung der Identität einer Person oder allgemein eines Schlüsselinhabers mit einem bestimmten Schlüssel ist nicht Bestandteil der asymmetrischen Algorithmen.
3.3.4 Weitere elementare Funktionen für kryptographische Protokolle

Es gibt weitere wichtige Funktionen, die in vielen kryptographischen Protokollen eingesetzt werden. Diese Funktionen übernehmen elementare, immer wiederkehrende Aufgaben bei den Protokollen. Die wichtigsten Funktionen werden kurz vorgestellt:


- Es muß unmöglich sein, einen Text zu finden, der einen vorgegebenen Hash-Wert ergibt. Wenn jemand einen bestimmten Hash-Wert vorgibt, dann darf es mit vertretbarem Aufwand nicht möglich sein, einen geeigneten Eingabetext für die Hash-Funktion zu finden, so daß der Vorgabewert errechnet wird.
- Es muß unmöglich sein, zu einem gegebenen Text mit einem dazugehörenden Hash-Wert einen zweiten — vom ersten verschiedenen — Text zu finden, der denselben Hash-Wert hat.
- Es muß unmöglich sein, zwei beliebige aber voneinander verschiedene Texte zu wählen, so daß sie denselben Hash-Wert haben.

Wenn die beiden ersten Bedingungen erfüllt sind, spricht man bereits von einer kryptographisch sicheren Funktion, einer sogenannten “one-way” Funktion. Die dritte Bedingung ist eine Verschärfung der zweiten. Funktionen, die zusätzlich der dritten Bedingung genügen, werden als *kollisionsfrei* (“collision resistant”) bezeichnet (vgl. [34]).

**Generieren von hochauflösenden Zeitstempeln:** Eine Nachricht, die verschlüsselt wurde, kann von einem Dritten auch ohne weitere Kenntnisse des Inhalts mißbraucht werden.

---

Es sind Angriffe ("replay attacks") bekannt, bei denen abgehörte Botschaften benutzt werden, um sie demselben Empfänger oder auch einem anderen Empfänger erneut zu übermitteln. Dazu werden die abgehörten Nachrichten vom Angreifer erneut in das Kommunikationssystem eingespielt. Ist die Botschaft beispielsweise eine verschlüsselte Banküberweisung, könnte durch einen Wiederholungsangriff die gleiche Überweisung mehrfach veranlaßt werden.


**Digitale Signaturen:** Mit asymmetrischen Verfahren ist es möglich, Dokumente digital zu "unterschreiben". Dazu verwendet der Unterzeichner seinen privaten Schlüssel, um entweder das Dokument oder einen Hash-Wert (s.o.) des Dokumentes zu verschlüsseln. Mit dem öffentlichen Schlüssel kann dann jeder die Herkunft des Dokumentes verifizieren. Das Entschlüsseln mit dem öffentlichen Schlüssel ergibt entweder wieder den Klartext des Dokumentes oder einen Hash-Wert, der mit dem berechneten Hash-Wert des Dokumentes verglichen werden kann.


### 3.4 Informale Definition: Kryptographische Protokolle

Die oben dargestellten Verfahren sind die Grundlage für eine Vielzahl kryptographischer Protokolle. Die kryptographischen Protokolle wurden entwickelt, um Dienste bereitstellen zu können, die Authentizität, Vertraulichkeit, Integrität und Unleugbarkeit für Kommunikationsbeziehungen anbieten. Der Begriff *kryptographisches Protokoll* wird auch für grundlegende kryptographische Algorithmen benutzt, wie beispielsweise das "Simultane Unterzeichnen von Dokumenten" (vgl. [38, S.118ff]). Im folgenden wird der Begriff allerdings auf Kommunikationsprotokolle eingeschränkt, die kryptographische Algorithmen zur Absicherung der
Kommunikation verwenden. Die folgende informale Definition faßt die wesentlichen Aspekte zusammen:

Ein *kryptographisches Protokoll* ist eine festgelegte Abfolge von Handlungen zweier oder mehrerer Kommunikationspartner, die der Erfüllung einer Kommunikationsaufgabe dienen. Teil der Handlungen ist die Verwendung von kryptographischen Verfahren, um die in Abschnitt 3.2 dargestellten Eigenschaften durchzusetzen oder zumindest Verstöße gegen diese Eigenschaften zu erkennen bzw. nachzuweisen.


### 3.5 Infrastruktur für die Schlüsselverteilung

Authentisierung ist eine sehr wichtige Funktion in verteilten Systemen. Wenn Dienste nicht öffentlich angeboten werden, sondern nur bestimmten Personen, Rechnern oder Prozessen zugänglich sein sollen, muß vor jedem Zugriff die Identität des Dienstanfordernden überprüft werden.

Bei der rechnerbasierten Kommunikation ist sicherzustellen, daß die für eine Identifizierung geeigneten Informationen nicht verfälscht oder von Dritten vorgetäuscht werden. Wir sprechen daher von *starker Authentisierung*, wenn mit Hilfe kryptographischer Protokolle eine sichere Identifizierung der beteiligten Kommunikationspartner möglich ist. Wenn für die *Autorisierung* eines Zugriffs über die Authentisierung hinaus weitere Informationen notwendig sind, müssen diese Informationen (beispielsweise “Credentials”, Tickets, etc.) ebenfalls sicher übertragen werden.


Für diese initiale Authentisierung werden in der Regel *Zertifikate* verwendet. Hierbei handelt es sich um beglaubigte öffentliche Schlüssel, bei denen eine Zuordnung des Schlüssels mit ei-

---

19 Es sind auch Dienste denkbar, bei denen sich nur einer der Kommunikationspartner identifizieren muß. Zum Beispiel authentisiert sich beim SSL-Protokoll nur der Server beim Client (vgl. Abschnitt 3.6).
ner Identität durch eine “Certification Authority” (CA) vorgenommen wurde. Im Internet, an das mehrere Millionen Rechner mit entsprechend vielen Benutzern angeschlossen sind, muß es effiziente Methoden geben, Schlüssel für die Kommunikation zu verwalten. Um beispielsweise asymmetrische Verfahren mit öffentlichen Schlüsseln im großen Umfang einsetzen zu können, ist eine Infrastruktur erforderlich, die es jedem Benutzer erlaubt, nach einem eindeutig spezifizierten Verfahren einen bestimmten öffentlichen Schlüssel aufzufinden.

Typische Dienste mit entsprechenden Protokollen, die diese Schlüsselverteilung und Schlüsselorganisation technisch realisieren sind:


**Spezialisierte Schlüsselverwaltung:** Neben dem allgemeinen Verzeichnisdienst X.500, der unter anderem auch öffentliche Schlüssel speichern kann, gibt es spezialisierte Dienste für die Schlüsselverwaltung und -verteilung. Das bekannteste Beispiel sind die “Key-Server” für PGP-Zertifikate. Diese spezialisierten Server sind ähnlich den DSAs im Internet verteilte und tauschen untereinander neue Schlüssel aus. Durch diesen Abgleich werden neue Schlüssel innerhalb kurzer Zeit weltweit verfügbar und können von den Anwendern lokal hinterlegt und abgeholt werden.


### 3.6 Protokolle für das Schlüsselmanagement

Für die Verschlüsselung von Kommunikationsbeziehungen werden in der Regel sogenannte *Sitzungsschlüssel* (“session keys”) vereinbart. Dies sind kurzlebige Schlüssel für die Dauer einer Kommunikationsbeziehung. Die Verwendung von kurzlebigen Schlüsseln hat mehrere Vorteile:

- Die Schlüssel werden bei Bedarf generiert und verteilt, sie brauchen nicht auf den lokalen Rechnerplatten gespeichert werden. Dadurch sinkt das Risiko, daß bei einem Angriff auf einen Rechner der Angreifer Zugriff auf fremde Schlüssel erhält.
Da die Schlüssel nur für kurze Zeit benutzt werden, ist der Wert eines durch Kryptanalyse gewonnenen Schlüssels für einen Angreifer sehr begrenzt. Er kann bestenfalls die Kommunikationsbeziehung entschlüsseln bzw. angreifen, dessen Schlüssel gewonnen wurde. Weitere Kommunikationsbeziehungen, die mit anderen Sitzungsschlüsseln abgesichert werden, bleiben unberührt.

Je weniger ein Schlüssel eingesetzt wird, desto weniger Geheimext kann mit diesem Schlüssel für eine Kryptanalyse in Verbindung gebracht werden, desto schwieriger ist es bei vielen Algorithmen, den Schlüssel zu berechnen.


3.6.1 Protokolle basierend auf Diffie-Hellman


Alice berechnet zuerst den Wert X und schickt diesen an Bob. Bob berechnet einen Wert Y und sendet diesen zurück an Alice. Die dafür benutzten Zahlen g, n müssen einigen Randbedingungen genügen (vgl. [30]), entscheidend ist jedoch, daß X, Y, g, n nicht geheim gehalten werden müssen. Es müssen lediglich x, y geheim gehalten werden und es gilt für den Sitzungsschlüssel k:

\[ k = k' = g^{xy} \mod n \]  

Das Diffie-Hellman Protokoll hat in seiner Urfassung ein wesentliches Manko: es ist anfällig gegen 'Man in the Middle' (MiM)-Angriffe. Bei dieser Form von Angriffen kann ein unbekannter Dritter die beiden Nachrichten abfangen und manipulierte Nachrichten an der Stelle der Ursprungsbotschaft weiterleiten.


Ein weiteres Schlüsselmanagementprotokoll mit dem Namen "Encrypted Key Exchange" geht ebenfalls vom kombinierten Einsatz asymmetrischer und symmetrischer Verfahren aus (vgl. [7]).
3.6.2 Protokolle basierend auf KDC

Eine andere Möglichkeit, Sitzungsschlüssel zu vereinbaren, besteht darin, sich an einen Schlüsselverteiler ("Key Distribution Center", KDC) zu wenden.

Dazu ist es erforderlich, daß alle Teilnehmer mit dem KDC einen symmetrischen Schlüssel vereinbart haben oder entsprechende asymmetrische Schlüsselpaare existieren. Weiterhin wird vorausgesetzt, daß alle Kommunikationspartner dem KDC vertrauen und dieser seinerseits kein Interesse hat, einen Angriff auf einen der Kommunikationspartner durchzuführen oder zu unterstützen. Man bezeichnet den KDC auch als Vermittler ("arbitrator").

In diesem Abschnitt wird das Konzept des KDC anhand des bekanntesten KDC-basierten Protokolls diskutiert. Es handelt sich um das Kerberos-Protokoll (vgl. [27]), das hier vereinfacht dargestellt wird.

Wenn Alice eine verschlüsselte Kommunikationsbeziehung mit Bob eingehen will, erfragt sie einen Sitzungsschlüssel beim KDC. Im einzelnen sind vereinfacht dargestellt folgende Schritte erforderlich:

1. Alice sendet die Anfrage $Anf(A, B)$ für einen Sitzungsschlüssel mit ihrer und Bobs Kennung an KDC.
2. Der KDC generiert einen zufälligen Sitzungsschlüssel $K$, einen Zeitstempel $T$ und eine Zeitspanne $L$, die die Gültigkeit des Schlüssels $K$ angibt.

3. Das Tupel $\{K,T,L\}$ wird nun mit den Schlüsseln von Alice und von Bob verschlüsselt und beide Geheime Texte ($V_A(K, T, L, B)$, $V_B(K, T, L, A)$) werden an Alice gesendet.

   - den Sitzungsschlüssel $K$ erhalten,
   - die Bestätigung erhalten, daß die Nachricht vom Vermittler verschlüsselt wurde, da nur dieser seinen Schlüssel $V_B$ kennt,
   - somit der Zeitstempel vom Vermittler generiert wurde,
   - der Vermittler Alice als Anfragende ansieht.

5. Bob bestätigt den Empfang des Sitzungsschlüssels, indem er die Nachricht $V_K(T + 1)$ an Alice sendet.


**Resümee:** Die für eine Ver- und Entschlüsselung wichtigen Schlüssel müssen vor der Verschlüsselung der Kommunikation sicher zwischen den beteiligten Kommunikationspartnern ausgetauscht werden bzw. sicher vereinbart werden. Diese Schlüsselabtäubung kann durch Schlüsselaustausch-Protokolle erfolgen. Verbreitet sind Schlüsselaustausch-Protokolle, die entweder direkt zwischen den Kommunikationspartner Schlüssel etablieren, und Protokolle, die einen vertrauenswürdigen Schlüsselverteiler in die Etablierung einbeziehen.

### 4 Kryptographische Protokolle in der TCP/IP-Familie


---

20 Die Kennung dient nur der Beschreibung der Endpunkte der Kommunikationsbeziehung, für die ein Sitzungsschlüssel angefordert wird. Hiermit wird noch keine Identifizierung durchgeführt.
4.1 Protokolle unterhalb von IP


Ethernet, Fast Ethernet und Gigabit Ethernet haben ein einheitliches Rahmenformat (vgl. Abbildung 12). Für diese Protokolle läßt sich ein einfacher Verschlüsselungsmechanismus implementieren, der entweder für die Endsysteme transparent ist oder von den Endsystemen selbst vorgenommen wird.

Um sicherzustellen, daß die Verschlüsselung auf Ethernet-Ebene mit vorhandenen Netzkomponenten (Hubs, Switches) kompatibel ist, müssen zwei Randbedingungen eingehalten werden:

1. Die Präambel, Senderadresse, Empfängeradresse und das Typ/Längenfeld dürfen nicht verschlüsselt werden. Die Präambel wird für die Synchronisation des Senders mit den
4.1 Protokolle unterhalb von IP

<table>
<thead>
<tr>
<th>Anwendungsschicht</th>
<th>Mail</th>
<th>WWW</th>
<th>Payment</th>
<th>Andere</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMTP, HTTP, ...</td>
<td>POP</td>
<td>S-HTTP</td>
<td>SET</td>
<td>Secure DNS</td>
</tr>
<tr>
<td>(Application Layer)</td>
<td>PEM</td>
<td>HTTP-S</td>
<td>STT</td>
<td>ISAKMP/Oakley</td>
</tr>
<tr>
<td></td>
<td>MOSS</td>
<td>S-MIME</td>
<td>SEPP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>EISS TESS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sesame</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Transportschicht</th>
<th>TLS</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP, UDP</td>
<td>SSL</td>
<td>PCT</td>
<td></td>
</tr>
<tr>
<td>(Transport Layer)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IP-Schicht (IP)</th>
<th>IPsec</th>
<th>SKIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Internet Layer)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Netzwerk Schnittstelle (Ethernet, FDDI, ATM, ...)</th>
<th>CryptoNodes</th>
<th>CelCase</th>
<th>KryptoLANs</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Network Interface)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hardware</th>
<th></th>
</tr>
</thead>
</table>

**Abbildung 11: Verfügbare kryptographische Protokolle**


2. Die maximale Länge des Rahmens von 1526 Byte darf nicht überschritten werden, da es hier zu unvorhersehbaren Fehlern bei der Übertragung und beim Empfang kommen kann. Dies ist besonders in den Fällen zu berücksichtigen, in denen für die Verschlüsselung Block-basierte Algorithmen (vgl. Abschnitt 3.3.3) verwendet werden, da hier eventuell zusätzliche Füll-Bytes zum Erreichen eines ganzzahligen Vielfachen der Blocklänge an die Daten angehängt werden müssen.

Die Verschlüsselung sollte sich daher auf den Datenbereich beschränken. Solange die maximale Datenlänge von 1500 Byte nicht überschritten wird, braucht der Verschlüsselungsalgorithmus auch nicht längenerhaltend zu sein.
Ein häufig vorkommendes Szenario für diese Art der Verschlüsselung ist die Koppelung von zwei Subnetzen über ein ungesichertes, abhörbares Netz. Man könnte in diesem Fall eine transparente Lösung für die beteiligten Rechner durch zwei “Bridges” erreichen, die zusätzlich eine Verschlüsselung bzw. Entschlüsselung vornehmen, bevor sie Dateneinheiten weiterleiten (vgl. Abbildung 13).

Die “Bridges” könnten die Ver- und Entschlüsselung auch auf Basis der Sender- und Empfängeradressen vornehmen. Das hat zwei Vorteile:


2. Bei speziellen Empfängeradressen (beispielsweise einem Router) wird keine Verschlüsselung vorgenommen. Damit ist es möglich, auch unverschlüsselt zu kommunizieren, beispielsweise mit Rechnern im unsicheren Netz.

In Abbildung 14 ist die Kommunikation von Alice aus dem Subnetz I mit Bob im Subnetz II und mit einem Internet-Servor (IS) im unsicheren Netz dargestellt. Die “Bridges” ver-
entschlüsseln bei Bedarf, also immer dann, wenn die Sende- und Empfängeradresse in verschiedenen Subnetzen eines virtuellen LANs liegen. 

\( V_B \) bezeichnet die Verschlüsselung mit dem symmetrischen Schlüssel der “Bridges”, 

\( E_B \) bezeichnet die Entschlüsselung der Nachricht.


**Resümee:** Die Verschlüsselung von Dateneinheiten durch Protokolle unterhalb von IP ist proprietär und erfordert spezialisierte Hardware, die auf die verwendete Netztechnologie zugeschnitten ist. Diese Verfahren sind unabhängig von IP und können deshalb auch Dateneinheiten von anderen Protokoll-Familien verschlüsseln.

### 4.2 Protokolle auf der IP-Ebene


---

21 Es sind hier auch Mechanismen zur Authentisierung denkbar. Diese Verfahren wurden aber nicht mit dem Ziel entworfen, einen Mechanismus für die Authentisierung zu entwickeln.
Ein weiteres wichtiges Protokoll auf der IP-Ebene ist SKIP (vgl. Abschnitt 4.2.2), das sich zu einem Quasi-Standard entwickeln könnte.

4.2.1 IPsec

Im folgenden werden die Spezifikationen der IPsec-Arbeitsgruppe ebenfalls als IPsec bezeichnet. Diese Terminologie ist konform mit der IETF-Terminologie.

IPsec bietet IP basierten Diensten folgende Funktionen an:

- Vertraulichkeit (für Daten, teilweise auch für den Verkehrsfluß)
- Authentisierung des Paketursprungs
4.2 Protokolle auf der IP-Ebene

- Integrität von verbindungslosen Dateneinheiten
- Erkennen und Ablehnen von wiederholten Paketen
- Zugriffskontrolle

Da diese Funktionen als Dienst der IP-Schicht angeboten werden, können sie von allen Protokollen auf der Transport- und Anwendungsebene verwendet werden.

Diese Funktionen werden wiederum durch zwei Protokolle erbracht:

- Spezifikationen bezüglich des “IP Authentication Header” (AH) definieren die Dienste für die Integrität von verbindungslosen Dateneinheiten, Authentisierung des Paketursprungs und für das Ablehnen von wiederholten Paketen.


Die Protokolle können auch kombiniert eingesetzt werden, je nachdem, welche Dienste der Benutzer anfordert.

Sowohl AH als auch ESP können in zwei verschiedenen Modi betrieben werden. Üblicherweise werden die Dienste für Transport- oder Anwendungsprotokolle der TCP/IP-Familie erbracht. AH und ESP werden in diesem Fall im sogenannten Transport Modus ("transport mode") betrieben. Es ist aber auch möglich, IP Dateneinheiten über ein beliebiges Netz hinweg zu “tunneln”, das heißt, IP Dateneinheiten wiederum in IP Dateneinheiten zu versenden. In diesem Fall werden ESP und AH im Tunnel Modus ("tunnel mode") betrieben.


Diese Tupel indizieren auf IPsec-Systemen eine “Security Association Database” (SAD), in der für jede SA Kontextparameter abgelegt sind. Dies sind in der Regel Angaben zum verwendeten Algorithmus beim Einsatz von AH oder ESP und zur Schlüssellänge sowie andere Informationen, die für die Bearbeitung der Dateneinheiten erforderlich sind.

22 Sollen ESP und AH kombiniert werden, sind zwei SAs notwendig.


![Abbildung 15: “IP Authentication Header” (vgl. [25], S. 4ff)](image)

Da die Sequenznummer streng monoton steigend ist und im ICV ebenfalls kryptographisch abgesichert wird, sind Replay-Angriffe leicht zu erkennen. Neben der Sequenznummer gehen die in Abbildung 15 dargestellten Informationen mit in die Berechnung des ICV ein. Es handelt sich hierbei um:

- die beim Routing unveränderlichen IP-Header-Informationen (beispielsweise Sender- und Empfängeradresse) und
die Informationen des AH-Header (Next Header, Länge, Reserved, SPI, Sequenznummer und Datenbereich mit notwendigen Füll-Byte, die zum Zweck der Berechnung auf Null gesetzt werden) sowie
die Nutzdaten des Datagramms (TCP, UDP, ICMP,...)

Zur Authentisierung kann eine Vielzahl von Algorithmen eingesetzt werden. Jede zu IPsec konforme Implementierung muß mindestens die Verfahren

**MD5** “Message Digest 5” (vgl. [37])

**SHA-1** “Secure Hash Algorithm 1” (vgl. [31])


4.2.1.2 **IP Encapsulating Security Payload:** Die Hauptaufgabe der ESP Spezifikation ist es, Vertraulichkeit bei der IPsec-basierten Kommunikation sicherzustellen. Daher wurde statt eines Headers ein ESP Paket definiert, um das kryptographische Material und die verschlüsselten Daten in IP-Datagrammen zu transportieren. Abbildung 87 stellt das Paketformat graphisch dar und zeigt die Einbettung des Paketes in ein Datagramm in Abhängigkeit vom jeweiligen Modus.


Die Felder SPI und Sequenznummer haben die gleiche Funktion wie beim AH. Optional kann man den ESP auch zum Authentisieren der Daten verwenden. In diesem Fall wird der Hash-Wert über die Nutzdaten und dem ESP-Kopf gebildet und an den ESP-Flipper angehängt. Eine der Funktionen Verschlüsseln oder Authentisieren muß immer vereinbart werden, beide können miteinander kombiniert werden.

Jede IPsec konforme Implementierung muß mindestens die folgenden Algorithmen für die Verschlüsselung und Authentisierung unterstützen:

**DES:** “Data Encryption Standard” im CBC Modus (vgl. [3])
MD5: “Message Digest 5” (s.o.)

SHA-1: “Secure Hash Algorithm 1” (s.o.)

NULL Authentisierung: Authentisierung wird nicht verlangt

NULL Verschlüsselung: Verschlüsselung wird nicht verlangt

MD5 und SHA-1 werden wie bei AH nur zur Authentisierung benutzt. Die beiden “NULL”-Algorithmen müssen definiert werden, da sie bei der Initialisierung der SA vereinbart werden müssen, falls keine Authentisierung oder keine Verschlüsselung erforderlich ist.

DES ist der einzige erforderliche Verschlüsselungsalgorithmus, der immer implementiert werden muß. Es handelt sich hierbei um einen Block-basierten Algorithmus mit einer Blocklänge von 64 Bit. Der “Cipher Block Chaining” (CBC) Modus (vgl. [2]) legt fest, daß jeweils der vorherige 64 Bit Geheimtextblock mit dem nächsten 64 Bit Klartextblock durch eine Exklusiv-ODer Operation verknüpft wird, bevor der Klartextblock verschlüsselt wird. Die einzelnen 64 Bit Blöcke des Geheimtextes sind dadurch nicht mehr unabhängig voneinander, sondern nur noch zusammen zu entschlüsseln.
Der Overhead durch ein ESP-Paket ergibt sich zu mindestens 10 Byte bei Verschlüsselung mit DES und mindestens 22 Byte bei einer DES-Verschlüsselung mit zusätzlicher SHA-1-Authentisierung. Bei der Berechnung sind eventuell notwendige Füll-Bytes zum Erreichen der DES-Blocklänge von 64 Bit noch nicht berücksichtigt.


Eine Lastverteilung scheitert aus einem ähnlichen Grund. Da die Datagramme nur von Rechnern entschlüsselt werden können, die die SA vereinbart haben, müssen die beiden Tunnelrouten in jedem Fall auf der gewählten Route liegen, normalerweise mit direktem Anschluß an das jeweilige Subnetz der Klienten, für die der Tunnel etabliert wurde.

Durch die Kontextinformationen ist IPsec außerdem auf eine neue Weise angreifbar. Wenn es gelingt, Kontextinformationen einer Seite zu manipulieren oder zu zerstören (mit anderen Worten die SA wird angegriffen), treten Fehler bei der Kommunikation auf (beispielsweise
werden gültige Datagramme auf der Empfangsseite als ungültig fehlinterpretiert und vernichtet). Diese Probleme müssen durch aufwendige Gegenmaßnahmen, die wiederum die kritisierte Verbindungsorientierung benötigen, angegangen werden.


4.2.2 SKIP

“Simple Key-Management for Internet Protocols” (SKIP, vgl. [6]) beschreibt eine weitere Spezifikation für ein Protokoll, das Vertraulichkeit und Authentizität auf der IP-Ebene sicherstellt. Die wesentlichen Unterschiede zu IPsec mit ISAKMP/OAKLEY bestehen darin, daß

- SKIP keine Verbindungskontexte vereinbart,
- SKIP verbindungslos arbeitet und somit weiterhin einen IP-Datagrammdienst anbietet,
- SKIP über ein eingebautes Schlüsselmanagementprotokoll verfügt und dadurch nicht den Dienst eines separaten Protokolls benötigt.

Das eingebaute Schlüsselmanagementprotokoll basiert auf zertifizierten öffentlichen Diffie-Hellman Schlüsseln, die entweder über ein beliebiges Schlüsselverteilungsprotokoll angefordert werden können, oder vom Systemmanager vor Inbetriebnahme von SKIP auf den beteiligten Rechnern installiert werden müssen.

Danach können zwei Rechner (I und J) sicher miteinander kommunizieren, indem sie einen impliziten Hauptschlüssel (“Master Key”) berechnen und die Dateneinheiten mit einem von diesem “Master Key” abgeleiteten Sitzungsschlüssel (hier Paketschlüssel genannt) verschlüsseln.

4.2.2.1 Berechnen von impliziten Hauptschlüsseln: Die Berechnung eines impliziten Hauptenschlüssels erfolgt aus dem gemeinsamen Geheimnis, das zwischen Sender I und Empfänger J etabliert ist. Der öffentliche Schlüssel von J ist gegeben durch \( g^j \mod n \). Nach Diffie-Hellman (vgl. Formel 61, S. 3) existiert dann ein gemeinsames Geheimnis zwischen J und I mit \( K_{IJ} = g^{ij} \mod n \), das in der SKIP-Terminologie auch als “Master Key” \( (K_{IJ}) \) für \( IJ \) bezeichnet wird. Mit diesem “Master Key” wird ein zufällig gewählter Paketschlüssel (Kp) mit einem symmetrischen Verfahren verschlüsselt. Mit dem Kp werden die Nutzdaten des Datagramms verschlüsselt. In einem SKIP-Paket wird dann der mit dem “Master Key”

Beim Empfang prüft der Empfänger, ob er bereits den Paketschlüssel im aktuellen Datagramm entschlüsselt hat. Dazu führt der Empfänger in typischen Implementationen eine Liste mit den Tupeln $\{V K_{ij}(K_p), K_p\}$ und kann so direkt über den verschlüsselten Paketschlüssel $(V K_{ij}(K_p))$ den Paketschlüssel $(K_p)$ finden. Andernfalls wird (im einfachsten Fall über die Zuordnung der IP-Senderadresse zum öffentlichen Schlüssel des Senders) mit Hilfe des öffentlichen Schlüssels des Senders der “Master Key” $(K_{MF})$ berechnet, um so den verschlüsselten Paketschlüssel zu entschlüsseln.


**4.2.2.2 Schlüssel für Authentisierung und Verschlüsselung:** Der Paketschlüssel $K_p$ wird nicht direkt zum Verschlüsseln benutzt. Da SKIP auch die Möglichkeit zur Authentisierung der Dateneinheiten bietet, werden zwei Schlüssel benötigt. Ein Schlüssel zum Verschlüsseln und ein Schlüssel zum Authentisieren. Diese beiden Schlüssel werden unter Verwendung ei-
ner “one-way” Hash-Funktion und \( K_p \) gewonnen, indem dieser Paketschlüssel zusammen mit einigen Konstanten und Bezeichnern für den verwendeten Verschlüsselungs- bzw. Authentisierungsalgorithmus verknüpft wird.

Die tatsächlich für die Verschlüsselung \( (V_{K_p}) \) und Authentisierung \( (A_{K_p}) \) verwendeten Schlüssel ergeben sich aus \( K_p \) zu:

\[
V_{K_p} = h(K_p \mid \text{Crypt Alg} \mid 0x02) \mid h(K_p \mid \text{Crypt Alg} \mid 0x00) \quad (4)
\]
\[
A_{K_p} = h(K_p \mid \text{MAC Alg} \mid 0x03) \mid h(K_p \mid \text{MAC Alg} \mid 0x01) \quad (5)
\]

Das Symbol \( \mid \) bedeutet, daß die Bits der Teilausdrücke aneinandergeschachtelt werden. SKIP verwendet für \( h() \) beispielsweise MD5 (vgl. [6]). “MAC Alg” und “Crypt Alg” sind numerische Werte, die den jeweiligen Algorithmus identifizieren, der zum Verschlüsseln oder Authentisieren benutzt wird. Von diesem Algorithmus hängt wiederum ab, wie viele der 256 Bits der Schlüssel tatsächlich benutzt werden.

4.2.2.3 Erneuerung des Hauptschlüssels: In dem SKIP-Datagramm ist ein Zähler untergebracht, der benutzt werden kann, um einen neuen “Master Key” durch den Sender festzulegen.
Abb. 19: SKIP-Empfangsmodul

Der Hauptschlüssel ist bei SKIP 256 Bit lang. Bei Diffie-Hellman kann man beliebig große Schlüssel vereinbaren, so daß hier immer nur die niederwertigsten 256 Bit benutzt werden.

Mit Hilfe des Zählers kann der Sender in jedem Paket anzeigen, welche Instanz \((K_{ijn})\) er zum Verschlüsseln des Pakets benutzt hat. Die Instanz wird aus \(K_{ij}\) und \(n\) mit Hilfe einer “one-way” Hash-Funktion \(h()\) (vgl. Abschnitt 3.4) berechnet:

\[
K_{ijn} = h(K_{ij} \mid n \mid 0x01) \mid h(K_{ij} \mid n \mid 0x00)
\]  

4.2.2.4 **Weitere Funktionen:** SKIP erlaubt auch die Vereinbarung von Gruppenschlüsseln. Alle Empfänger der Gruppe können dann die Nachrichten entschlüsseln. Ermöglicht wird dies durch entsprechende Identifikationsfelder im SKIP-Paket. Diese sogenannten “Name Space” Bezeichner ermöglichen die Identifikation der Gruppen. Den Gruppen sind wiederum ver-
schiedene Hauptschlüssel zugeteilt. Im einfachsten Fall erfolgt die Auswahl eines Hauptschlüssels jedoch über die IP-Adressen.

4.2.2.5 Probleme bei SKIP: Der SKIP-Paketkopf ist 28 Byte groß. Diese 28 Byte fallen zusätzlich zum ESP- und/oder AH-Overhead bei jedem IP-Datagramm an. Um Fragmentierungen im jeweiligen Endsystem zu vermeiden, wird durch SKIP in der Regel die MTU-Größe angepaßt. Protokolle wie TCP werden dann ihrerseits kleinere Segmente generieren, so daß der zusätzliche SKIP-Overhead nicht zu einer Fragmentierung führen muß (vgl. [5]).

Es gibt keine “perfect forward secrecy”. So lange die Hauptschlüssel geheim sind, nützt einem Angreifer das passive Abhören der SKIP-Kommunikation nichts. Sollte einem Angreifer jedoch zu irgend einer Zeit der Hauptschlüssel bekannt werden, dann kann er nachträglich alle in der Vergangenheit aufgezeichneten Datagramme entschlüsseln.


4.3 Protokolle auf der Transportebene


---

4.3 Protokolle auf der Transportschicht


TLS übernimmt außerdem die Verschlüsselung der Verbindung. Nach der Authentisierung setzt die Verschlüsselung ein, so daß die Geheimhaltung der Transaktionen zwischen Client und Server sichergestellt ist.

TLS soll die Geheimhaltung und Authentizität von Daten zwischen zwei Anwendungen garantieren. Dazu werden zwei Protokolle spezifiziert:

- “TLS Record Protocol” (TRP)
- “TLS Handshake Protocol” (THP)

Das TRP sichert die Geheimhaltung von übertragenen Daten mit Hilfe von symmetrischer Verschlüsselung. Außerdem wird die Integrität der Daten durch zusätzliche Berechnung eines “Message Authentication Codes” (MAC) gesichert, so daß Manipulationen erkennbar sind.


Problematisch an TLS ist die Verknüpfung mit der Anwendung bzw. die Einbindung in den Protokollstapel. Intuitiv würde man annehmen, daß TLS eine eigene “well known” Portnummer zugeteilt wird, über die das TLS Protokoll in TCP-Segmenten identifiziert werden kann. Die jeweilige Applikation könnte dann wiederum über einen TLS-Protokolltyp identifiziert werden. Dieses Vorgehen wird von der IETF allerdings nicht gewählt. Es gibt bisher zwei Ansätze, nach denen TLS mit den Anwendungen verknüpft werden soll:

1. Die mit TLS abgesicherte Variante eines Dienstes wird auf einem eigenen Port angeboten. Dieser Ansatz würde schlagartig die Anzahl der bekannten und festgelegten Portnummern (“well known ports”) verdoppeln.


---

24 Der Mechanismus zum Sichern der Authentizität von einzelnen Nachrichten zwischen diesen Kommunikationspartnern beruht auf der Verschlüsselung mit dem gemeinsamen Schlüssel, der im Zuge der Authentisierung zwischen Client und Server etabliert wird.
4.3.1 Das TLS Record Protocol

Das “TLS Record Protocol” (TRP) verwaltet für jede Verbindung zwei Zustandsvariablen in einem Verbindungskontext. Es gibt jeweils eine Zustandsvariable für ausgehende Daten (“written”) und für zu lesende Daten (“read”). Die Zustandsvariablen sind strukturiert durch die folgenden Bezeichner:

**Endpunkthebezeichnung** („connection end”) gibt an, ob es sich um den Dienstnehmer (Client) oder den Dienstleister (Server) bezüglich der Verbindung handelt.

**Verschlüsselungsalgorithmus** („bulk encryption algorithm”) gibt den Algorithmus zum Verschlüsseln der Anwendungsdaten an.

**Authentisierungsalgorithmus** („MAC algorithm”) gibt den Algorithmus zum Berechnen des “Message Authentication Codes” (MAC) an.

**Kompressionsalgorithmus** („compression algorithm”) bezeichnet den Algorithmus zum Komprimieren der Anwendungsdaten vor der Verschlüsselung oder Authentisierung.


Neben diesen eher statischen Werten, beinhaltet der Kontext weitere Informationen, die für jedes TLS-Paket aktualisiert werden:

**Kompressionszustand** ist der Zustand des Komprimierungsalgorithmus.

**Verschlüsselungszustand** ist der Zustand des Verschlüsselungsalgorithmus. Das kann beispielsweise der Initialisierungsvektor für einen Algorithmus sein, oder ein Zwischenwert, der neben den eigentlichen Daten in die Berechnung des folgenden Geheimtextes eingeht.

**MAC-Schlüssel** („MAC secret”) ist der geheime Schlüssel für den MAC-Algorithmus. Dieser Schlüssel wird nicht verändert, sondern bleibt nach der Initialisierung konstant.

**Sequenznummer** ist eine streng monoton steigende Zahl, die in jedes Paket kopiert und anschließend um eins erhöht wird. Damit sind Auslassungen und Wiederholungen von einzelnen Paketen erkennbar.

Die einzelnen Initialwerte für diese Zustandsvariablen werden am Anfang von einem gemeinsamen Geheimnis abgeleitet, das durch das THP (s.u.) etabliert wird. Das TRP bietet somit einen Dienst für die Verschlüsselung und Authentisierung von beliebig langen Datenblöcken an.

---

4.3.2 Das TLS Handshake Protocol


THP ist für diese Aufgaben unterteilt in die drei Sub-Protokolle:

**Handshake Protocol:** Das “Handshake Protocol” wird zur Authentisierung und zum Etablieren des gemeinsamen Geheimnisses ("shared secret") benutzt.

**Change Cipher Spec Protocol:** Dieses Protokoll wird zum Wechseln der Verschlüsselungsmethode einer bestehenden Verbindung benutzt.

**Alert Protocol:** Das “Alert Protocol” wird benutzt, um Fehlernachrichten und sonstige Metameldungen zwischen den Partnerinstanzen zu versenden.

Auf die beiden letztgenannten Protokolle braucht hier nicht weiter eingegangen zu werden. Wichtig für die Schlüsselvereinbarung und Authentisierung ist das “Handshake Protokoll”. Die dabei ausgetauschten Informationen sind sehr komplex, so daß eine Reihe von speziellen Nachrichten definiert wurde, um die Struktur der transportierten Informationen einfach abbilden zu können. Es handelt sich hierbei um die folgenden Nachrichten:

**Hello:** Es gibt ein “Client Hello” und ein “Server Hello”. Initiator einer Verbindung ist in der Regel der Client. Er sendet einen Zeitstempel und eine Zufallszahl, die für spätere Berechnungen von Sitzungsschlüsseln benutzt werden. Außerdem kann der Client die SitzungsID einer bereits etablierten Sitzung angeben, um diese entweder wieder aufzunehmen (“resume”) oder eine weitere zu dieser Sitzung identische Sitzung zu eröffnen. Normalerweise ist dieser Wert nicht belegt, so daß eine neue Sitzung erstellt wird. Der Client muß außerdem eine Liste der von ihm unterstützten kryptographischen Verfahren (LKrV) angeben. Diese Liste ist priorisiert, d.h. das vom Client favorisierte Verfahren sollte zuerst angegeben werden, dann mit absteigender Priorität die weiteren bekannten Verfahren. Analog zur LKrV gibt der Client eine Liste der bekannten Komprimierungsverfahren an (LKoV).

Der Server gibt neben einer Protokollversion und seinem Zufallswert die SitzungsID an. Im Fall einer wieder aufgenommenen Sitzung entspricht diese ID der vom Client vorgegebenen, andernfalls wählt der Server eine neue, eindeutige ID. Der Server gibt seine Wahl für die zu verwendenden kryptographischen Verfahren (Verschlüsselung und Authentisierung) in dem Feld GKrV an, das gewählte Komprimierungsverfahren im Feld GKoV.

**Certificate:** Die Zertifikat-Nachricht des Servers ist erforderlich, die des Clients wird nur auf explizite Anforderung des Servers gesendet. Die Nachricht beinhaltet eine Zertifikatsidentifizierung und eine Liste der jeweiligen Zertifikate.


Das THP Protokoll kennt weitere optionale Nachrichten, auf die nicht eingegangen werden muß, um das TLS-Konzept zu verstehen. Der generelle Verlauf der Nachrichten zwischen Client und Server ist in Abbildung 108 graphisch dargestellt.

4.3.2.1 Eingesetzte Verschlüsselungsverfahren: Zur Verschlüsselung der Daten in den TRP-Paketen wird DES oder 3DES eingesetzt. Dies sind symmetrische Verfahren, die wiederum im “Cipher Block Chaining” (CBC) Modus betrieben werden (vgl. [1, 2]).

26 Die Zertifikatsidentifizierung gibt den Typ des Zertifikats an.

4.3.2.2 **Eingesetzte Paket-Authentisierungsverfahren:** Eine Prüfsumme über die Paketinhale wird wiederum entweder mit MD5 oder SHA-1 erstellt (vgl. Abschnitt 4.2.1.1).

4.3.2.3 **Eingesetzte Authentisierungsverfahren:** Zur Authentisierung des Servers werden digital unterzeichnete Zertifikate verwendet. Diese Zertifikate müssen entweder Konform zum RSA Standard sein oder zum “Digital Signature Standard” (DSS). RSA ist ein bekanntes und weit verbreitetes “Public Key” Kryptosystem (vgl. [36]). Der DSS wurde von einem US-amerikanischen Standardisierungsinstitut (NIST) verabschiedet (vgl. [32]).

4.3.2.4 **Historische Entwicklung:** TLS baut auf einer Entwicklung der Firma “Netscape Communications Corporation” auf. Unter dem Namen “Secure Socket Layer” wurde eine Spezifikation veröffentlicht, die die Grundlage von TLS 1.0 ist.

4.4 Protokolle auf der Anwendungsebene

Auf Anwendungsebene können ebenfalls Sicherheitsmechanismen implementiert werden. Im Unterschied zu allen bislang diskutierten Mechanismen sind die Realisierungen auf der Anwendungsebene immer anwendungsspezifisch. Die Schutzfunktionen werden daher für eine ganz bestimmte Anwendung oder Klasse von Anwendungen spezifiziert und sind nicht universell einsetzbar wie IPsec und TLS.


4.4.1 Sicherer HTTP (S-HTTP)

Das “Secure HTTP” (S-HTTP) ist eine Spezifikation [35], die das “Hypertext Transfer Protocol” (HTTP, vgl. [17]) um Mechanismen zur Authentisierung, Geheimhaltung und Unleugbarkeit der Sendung erweitert.


Die unterstützten symmetrischen Algorithmen für die Verschlüsselung sind: DES, “Triple-DES” (3DES) mit zwei oder drei Schlüsseln, eine abgeschwächte DES Version und “International Data Encryption Algorithm” (IDEA) [38, S. 319ff].
Resümee: S-HTTP ist eine Erweiterung von HTTP. S-HTTP verwendet ebenfalls die bei TLS vorgestellte Authentisierung mit Zertifikaten.

4.4.2 Sicherer “Remote”-Login


Diese “Remote”-Kommandos mit den dazugehörenden Protokollen haben eine große Bedeutung in der Welt der verteilten Systeme, da so auf einfache Weise die Betriebsmittel weit entfernter oder spezialisierter Rechner benutzt werden können. Es gibt jedoch eine Reihe von Problemen mit diesen Protokollen:


28 In der Regel ist dies TCP. Es kann allerdings auch ein anderes Protokoll verwendet werden, da TLS keine TCP-spezifischen Funktionen benötigt. Das Protokoll muß lediglich die korrekte Übertragung der Daten und Einhaltung der Paketreihefolge sichern.
Drei Protokolle werden spezifiziert, die unterschiedliche Dienste erbringen:

**Transport Layer Protocol:** Dieses Protokoll (TLP) sorgt für eine Verschlüsselung der übertragenen Daten, führt Integritätsprüfungen durch und ermöglicht eine Authentisierung des Servers beim Client mit Hilfe von asymmetrischen Schlüsseln ("Public Key" Verfahren). Zusätzlich kann eine Komprimierung der Daten vorgenommen werden.

**Authentication Protocol:** Dieses Protokoll (AP) setzt auf dem TLP auf und ermöglicht die Authentisierung des Clients oder eines Benutzers beim Server.

**Connection Protocol:** Dieses Protokoll (CP) setzt wiederum auf dem AP auf und bietet interaktiven Programmen die Möglichkeit, diverse "login"-Funktionen auszuführen. Beispiele sind Passwortabfrage und -änderung. Außerdem ermöglicht das CP mehrere "Sessions" über eine authentisierte Verbindung zu multiplexen.

Bei Verwendung der SSH-Protokolle innerhalb der TCP/IP-Protokollfamilie ergibt sich somit der in Abbildung 21 dargestellte Protokollstapel. In den folgenden Unterabschnitten werden die einzelnen Protokolle kurz vorgestellt.

4.4.2.1 Das “Transport Layer Protocol”: Für die Authentisierung wird ein “Public Key”-Verfahren eingesetzt. Ein Client kann so überprüfen, ob er tatsächlich mit dem angenommenen Server verbunden ist. In einem zweiten Schritt wird mit Hilfe des Diffie-Hellman-Verfahrens ein gemeinsames Geheimnis etabliert, aus dem dann mehrere Schlüssel abgeleitet werden. Für jede Kommunikationsrichtung (Client zum Server und Server zum Client) wird ein Verschlüsselungsschlüssel und ein Schlüssel für die Integritätsprüfung erstellt. Der Hauptaufwand beim Protokoll entsteht durch die Nachrichten zum Aushandeln der verschiedenen

Ohne weiter auf die Struktur der einzelnen TLP-Nachrichten einzugehen, ist der Standardfall für einen Verbindungsaufbau in Abbildung 22 abgebildet.

Abbildung 22: Schlüsselaustausch und Authentisierung beim TLP

4.4.2.2 Das “Authentication Protocol”: Nachdem zwischen Client und Server eine gesicherte Verbindung mit dem TLP hergestellt wurde, kann die Authentisierung des Benutzers

```
ssh–tlp

Client

Proto_Version ("ssh–cx–cy", Kommentare)

Proto_Version ("ssh–sx–sy", Kommentare)

Protokollversionen und somit die "Features" sind identifiziert

SSH_MSG_KEXINIT (Liste der bekannten Algorithmen)

SSH_MSG_KEXINIT (Liste der bekannten Algorithmen)

Client und Server haben sich auf Algorithmen geeinigt

SSH_MSG_KEXDH_INIT (Diffie–Hellman Public Key)

SSH_MSG_KEXDH_REPLY (Diffie–Hellman Public Key, Signatur)

Gemeinsames Geheimnis (K) bekannt!

Gemeinsames Geheimnis (K) bekannt, Server hat sich authentisiert!

SSH_MSG_NEWKEYS()

SSH_MSG_NEWKEYS()

Schlüsselaustausch beendet, Alle folgenden Pakete werden mit den von K abgeleiteten Sitzungsschlüsseln verschlüsselt und authentisiert.

Server
```
beim Server erfolgen. Das Authentisierungsprotokoll unterstützt hierfür drei verschiedene Methoden:

- Authentisierung mit einem “Public Key”
- Authentisierung durch interaktive Abfrage eines Passworts
- Authentisierung auf Grundlage der IP-Adresse


Bei der zweiten Methode sendet der Server die Aufforderung, ein Passwort einzugeben. Das Passwort wird vom Client zum Server sicher übertragen, da AP auf dem verschlüsselnden TLP aufsetzt.


4.4.2.3 Das “Connection Protocol”: Nach erfolgreicher Authentisierung können mehrere verschlüsselte Sitzungen mit Hilfe des “Connection Protocols” (CP) über die authentisierte und verschlüsselte Verbindung verwaltet werden. Der Vorteil des Multiplex-Betriebs liegt darin, daß gegenüber dem Eröffnen von neuen Verbindungen alle aufwendigen kryptographischen Operationen (Schlüsselaustausch) nicht noch einmal durchgeführt werden müssen. Das CP bietet eine Reihe weiterer Dienste an, auf die hier nicht weiter eingegangen werden kann.


4.4 Protokolle auf der Anwendungsebene

4.4.3 Sicheres “Domain Name System” (Secure DNS)

Das “Domain Name System” (DNS) beschreibt einen hierarchischen, verteilten Namensraum für die Zuordnung von Internet-Adressen zu symbolischen Namen. In diesem Namensraum können weitere Informationen, wie beispielsweise der Name eines Ansprechpartners für einen Rechner, gespeichert werden.

Die verteilte Datenhaltung wird durch die DNS-Server erbracht. DNS-Clients können dadurch ihre Anfragen an einen lokalen Server senden, der diese gegebenenfalls weiterleitet, falls er die gewünschten Informationen nicht lokal gespeichert hat. Typischerweise sind auf jedem Endsystem DNS-Clients im Betriebssystem integriert, um für den Benutzer transparent die Auflösung eines symbolischen Namens in eine IP-Adresse zu veranlassen.


“Secure DNS” implementiert drei Mechanismen:

1. Authentisierung der Daten bei “Zone Updates” DNS-Server können die von ihnen versendeten “Zone Updates” digital signieren. Dadurch ist die Integrität der Daten und deren Herkunft überprüfbar. Dafür ist es erforderlich, eine “Public Key” Infrastruktur bereit zu stellen, so daß die Signaturen eines Servers von anderen Servern mit dem entsprechenden öffentlichen Schlüssel überprüft werden können.

2. Bereitstellen einer Schlüsselverteilung. Die für die Überprüfung der Signaturen der “Zone Updates” notwendigen öffentlichen Schlüssel können ebenfalls in der DNS-Datenbank abgespeichert werden. Damit stehen den DNS-Servern nach einem “Zone Update” auch die nötigen Schlüssel zur Verfügung, um für weitere “Zone Updates” die öffentlichen Schlüssel dynamisch zu lernen.


29 Bei “Zone Updates” werden die gesamten Datenbankinformationen, für die ein Server verantwortlich ist (also die Daten seiner Zone), an einen oder mehrere Server versendet, die stellvertretend für den Zone-Server antworten können.


4.4.3.1 Der “Key Resource Record”: Neben dem eigentlichen öffentlichen Schlüssel werden in “Key Resource Records” (Key-RR) weitere Informationen gespeichert, die Hinweise auf den verwendeten Verschlüsselungsalgorithmus geben als auch Hinweise, wofür dieser Schlüssel zu verwenden ist. Über mehrere Bits wird beispielsweise kodiert, ob der Schlüssel an eine Maschine (IP-Adresse) oder an eine Person gebunden ist.

4.4.3.2 Der “SIG Resource Record”: Der “SIG Resource Record” (SIG-RR) wird von den DNS-Servern benutzt, um digitale Signaturen für die anderen RR in ihrer Datenbank abzulegen. Der Empfänger kann anhand der RR und der Signaturen überprüfen, ob die Daten tatsächlich aus der jeweiligen Zone kommen und ob an den Daten durch einen Angreifer Änderungen vorgenommen wurden.


4.4.4 ISAKMP/OAKLEY


Grundsätzlich ist ISAKMP ein allgemeines Protokoll für die Verhandlung und Etablierung von SAs. Es kann auch für andere Protokolle eingesetzt werden, die ebenfalls — wie IPsec (AH und ESP) — ein eigenständiges Protokoll zur Authentisierung und Schlüsselabtiberung benötigen. ISAKMP stellt daher einen Mechanismus zum sicheren Verhandeln von Verschlüsselungs- und Authentisierungsmechanismen zur Verfügung.


ISAKMP etabliert die SA für ein anderes Protokoll in zwei Schritten. Zuerst wird ein gesicherter Kanal für ISAKMP etabliert, dann können über diesen gesicherten Kanal in einem zweiten Schritt weitere SAs für IPsec oder ein anderes Protokoll etabliert werden. Um die Verhandlungen der Kommunikationspartner zu sichern, schützt ISAKMP diese Verhandlungen durch eigene Mechanismen gegen:

- “Denial of Service” (DoS)-Angriffe
- Wiederholungsangriffe
- “Man-in-the-Middle” (MiM)-Angriffe
- Übernahme der Verbindungen (“Connection Hijacking”)

Um diesen Schutz erreichen zu können, erfordert ISAKMP von den beteiligten Kommunikationspartnern ihrerseits den Einsatz von starker Authentisierung. Die Authentisierung wird
mit einem initialen Schlüsselaustausch gekoppelt, so daß der Schlüssel und die Kommunikationspartner authentisiert sind. Danach kann auf dieser Grundlage die eigentliche Verhandlung über eingesetzte Zertifikate, Authentisierungsmechanismen und Verschlüsselungsalgorithmen beginnen (vgl. [29]).

Haben sich die Kommunikationspartner auf angemessene Mechanismen geeinigt, unterstützt ISAKMP anschließend auch den Austausch von Zertifikaten und die Vereinbarung von Schlüsseln und bietet somit eine universelle Plattform für die Etablierung beliebiger SAs. Das von ISAKMP bevorzugte Verfahren zum Schlüsselmanagement ist OAKLEY (vgl. [18]).

ISAKMP setzt kein Protokoll für den Zugriff auf öffentliche Zertifikatsverzeichnisse voraus. Die Überprüfung der ausgetauschten Zertifikate liegt in der Verantwortung der beteiligten Kommunikationspartner, die eine SA etablieren wollen.


### 4.4.5 PKI-Protokolle


4.5 Anwendungen außerhalb des Protokollstapels

Kryptographische Verfahren werden auch außerhalb des Protokollstapels angewendet. Grundsätzlich kann jedes Anwendungsprogramm auf einem Endsystem kryptographische Verfahren einsetzen.

31 Zugriff: Eintragen, Suchen, Ändern oder Löschen


**Resümee:** Es gibt viele Anwendungsprogramme, die kryptographische Verfahren zur Verarbeitung von Informationen außerhalb des Kommunikationssubsystems einsetzen. Etabliert hat sich vor allem PGP zum kryptographischen Absichern von elektronischer Post.

### 4.6 Nicht durch kryptographische Protokolle abgedeckte Sicherheitsprobleme

Im Abschnitt 3.4 wurde einleitend dargestellt, daß mit Hilfe von kryptographischen Verfahren die Vertraulichkeit, Integrität, Authentizität und Unleugbarkeit von Nachrichten in Kommuni-
kationssystemen gesichert werden kann. In den vorherigen Abschnitten wurden verschiedene Protokolle vorgestellt, die diese Verfahren nutzen, um auf verschiedenen Ebenen im Kommunikationssubsystem Nachrichten zu schützen.

Es gibt aber eine Reihe von Sicherheitsproblemen, die nicht durch kryptographische Protokolle gelöst werden können. Teilweise verschärfen sich sogar bekannte Probleme durch den Einsatz kryptographischer Verfahren. Diese Probleme lassen sich in folgende Kategorien einteilen:


**Softwarefehler:** Ein ähnliches Problem ergibt sich durch Softwarefehler. Kommerziell erworbene Software kann meist nicht anhand des Quellcodes verifiziert werden. Fehlerhafte Software könnte unerlaubt Daten verschlüsselt oder unverschlüsselt versenden (exportieren).

**kritische Kommunikationsanforderungen:** In Unternehmen gibt es oft ältere Anwendungen, die unternehmenskritisch sind und in jedem Fall eingesetzt werden müssen. Manchmal kann diese Software nicht mit kryptographischen Verfahren zusammen eingesetzt werden, da das jeweilige System nicht mehr nachträglich um neue Protokolle wie IPsec oder TLS erweitert werden kann. In diesem Fall muß eine andere Art der Authentisierung – zum Beispiel mit Hilfe eines Firewalls – gewählt werden. Kryptographische Verfahren sind nicht in jedem Fall verfügbar.


4.7 Ausblick


Im Zusammenhang mit dem “E-Commerce”, aber auch durch den zunehmenden Einsatz von PGP, wird der Ausbau von CAs weiter voranschreiten. Einher mit diesem Ausbau steigt der Bedarf nach Protokollen zum Zugriff auf die Zertifikate, so daß LDAP an Bedeutung gewinnen wird.

Für die sofortige Absicherung von interaktiven Sitzungen und “Remote”-Kommandos kann SSL uneingeschränkt empfohlen werden.

5 Vorteile durch einen kombinierten Einsatz

Aus den in Abschnitt 39 und Abschnitt 141 dargestellten Problemen, die einerseits bei reinen Firewall-Lösungen und andererseits bei reinen “Krypto”-Lösungen offen bleiben, ergibt sich der Wunsch, die jeweiligen Vorteile der Verfahren zu kombinieren.


Auf der anderen Seite kann man den Standpunkt vertreten, daß die kryptographischen Mechanismen die grundlegenden Sicherungsmöglichkeiten bieten. In diesem Fall stellt sich dennoch die Frage, wie Firewalls eingesetzt werden können, um die verbleibenden Sicherheitsrisiken einzudämmen (vgl. Abschnitt 5.2).

Die Symbiose eröffnet auch neue Möglichkeiten, die in Zukunft einen effizienteren und feiner granulierten Einsatz von Sicherungsmechanismen erlauben.

## 5.1 Nutzen kryptographischer Verfahren für Firewalls

In diesem Abschnitt wird gezeigt, wie durch kryptographische Verfahren die bei heutigen Firewalls bestehenden Schwachstellen beseitigt werden können.

### 5.1.1 Authentisierung


Um der Packet Screen die Überprüfung des AH zu ermöglichen, werden in [14] drei unterschiedliche Lösungsmöglichkeiten vorgeschlagen:

- Die naheliegende und einfachste Lösung wäre es, das Key-Management-Protokoll so zu erweitern, daß die Packet Screen den für die Generierung des AH verwendeten Schlüssel bei einem der Kommunikationspartner erfragen kann. Allerdings würde dies der Packet Screen nicht nur die gewünschte Kontrolle des AH ermöglichen, sondern auch die Gene-
  rierung von Paketen mit korrektem AH. Der Empfänger kann nicht mehr unterscheiden,
  ob es sich um ein Paket des vermeintlichen Senders oder um ein gefälschtes Paket der
  Packet Screen handelt. Diese Lösung ist daher nicht empfehlenswert.

- Um diese Möglichkeit des Fälschens von Paketen durch die Packet Screen zu vermei-
  den, muß der Sender mit der Packet Screen einen eigenen Schlüssel vereinbaren. Der
  Sender generiert zunächst wie auch im vorigen Fall ein Paket, das an den Empfänger
gesendet werden soll. Die Packet Screen weist dieses Paket ab und fordert den Sender
5.1 Nutzen kryptographischer Verfahren für Firewalls


Leider erfordern Signatur-Verfahren deutlich mehr Berechnungen als symmetrische Verfahren wie MD5. Selbst bei MD5 bestehen schon jetzt Bedenken, ob die Prozessorleistung und somit die Geschwindigkeit für alle Anwendungen ausreichend sein wird. Daher ist es nicht sehr wahrscheinlich, daß asymmetrische Verfahren eine große Verbreitung finden werden.


Ein weiteres Problem besteht darin, daß die Packet Screen auch bei den asymmetrischen Verfahren das Key-Management-Protokoll nutzen muß, um den Schlüssel zu erhalten, mit dem dann die Signaturen überprüft werden können. Bei der großen Anzahl von Verbindungen, die durch eine Packet Screen hindurch aufgebaut werden, wird durch das Key-Management-Protokoll eine deutliche Last erzeugt werden.

Inzwischen ist die zweite Lösung von verschiedenen Autoren aufgegriffen worden und scheint sich daher als zukünftige Lösung für dieses Problem abzuzeichnen.

Resümee: Durch den Einsatz von kryptographischen Verfahren ist eine sichere Authentisierung auf dem Firewall möglich. Diese Authentisierung kann außerdem transparent erfol-

5.1.2 Integrität übertragener Daten


5.1.3 Erweiterte Zugriffskontrolle


5.1.4 Fernadministration von Firewalls

5.2 Nutzen von Firewalls für kryptographische Verfahren


5.2 Nutzen von Firewalls für kryptographische Verfahren


5.2.1 Firewalls für offene Probleme

Die in Abschnitt 141 zusammengefaßten Probleme sollen hier wieder aufgenommen werden. Für die meisten Probleme gibt es offensichtliche Lösungen, die auf dem Einsatz von Firewallmechanismen beruhen.


5.2.1.2 **Kritische Kommunikationsanforderungen:** Ein Firewall bietet sich immer dann an, wenn kryptographische Verfahren zur Absicherung nicht eingesetzt werden können, der ungesicherte Dienst aber dennoch erforderlich ist. In einem Szenario, in dem ein Dienst aufgrund von Hardware- oder Software-Restriktionen nicht durch starke Authentisierung der Clients abgesichert werden kann, muß diese Authentisierung stellvertretend auf dem Firewall erfolgen. Der Firewall ermöglicht die starke Authentisierung eines Zugriffs von außerhalb und sichert somit den Zugriff auf kritische Dienste ab. Der Firewall bietet sogar die Möglichkeit, erstmalig sicher aus einem ungesicherten Netz auf die kritischen Dienste zuzugreifen (vgl. Abschnitt 5.2.3).

5.2.1.3 **DoS:** Die kryptographischen Verfahren sind besonders rechenaufwendig und bieten sich daher für “Denial of Service”-Angriffe geradezu an. Ein Firewall kann auch in diesem Fall deutlich das Risiko verringern. In der Regel gibt es nur wenige Benutzer oder Maschinen, die Zugriff von externen Netzen auf Dienste im internen Netz erhalten sollen und sich zuvor authentisieren müssen. Allen anderen Clients sollte nicht einmal das Aushandeln von Authentisierungs- und Verschlüsselungsverfahren gestattet werden.


5.2.1.4 **Zugriffskontrolle:** Die Möglichkeit, Daten zu verschlüsseln oder Zugriffe auf Server zu authentisieren, ist völlig orthogonal zu einer Zugriffskontrolle, die unerwünschte Kommunikation verhindert. Soll der Zugriff von einem internen Netz auf Betriebsmittel im Internet aus irgendwelchen Gründen verhindert werden, ist unbedingt ein Firewall erforderlich. Das Verhindern unerwünschter Kommunikation kann nicht durch kryptographische Mechanismen erreicht werden.

**Resümee:** Der zusätzliche Einsatz von Firewalls für die Zugriffskontrolle verhindert oder lindert verbleibende Sicherheitsrisiken, die nicht durch den alleinigen Einsatz von kryptographischen Verfahren ausgeschlossen werden. Ein Großteil der externen DoS-Angriffe auf rechenintensive, kryptographisch gesicherte Dienste kann bereits am Firewall abgefangen werden.
5.2 Nutzen von Firewalls für kryptographische Verfahren

5.2.2 Durchsetzen von Verschlüsselungsrichtlinien am Firewall

Ohne den Einsatz eines Firewalls ist es unmöglich, durchzusetzen, daß die kryptographischen Mechanismen auch tatsächlich eingesetzt werden. Ein Firewall kann hingegen bestimmte Verschlüsselungs- oder Authentisierungsverfahren erzwingen. Dies gilt sowohl für die eingehende Kommunikation als auch für die ausgehende Kommunikation aus dem zu schützenden Netz. Eine Firma könnte beispielsweise verlangen, daß Daten nur in verschlüsselter Form mit externen Stellen (Heimarbeiter, Vertreter mit Laptops) ausgetauscht werden, um die Firmendaten zu schützen. Die Einhaltung dieser Richtlinie ist jedoch nur an dem zentralen Zugangspunkt zum Netz — also am Firewall — durchsetzbar.

Derartige Verschlüsselungsrichtlinien können beliebig komplex gestaltet werden. Abhängig von Rechnern oder Personen könnten bestimmte Verfahren gefordert werden, die der Firewall individuell überprüft und durchsetzt.


5.2.3 Firewall als Verschlüsselungs-Gateway


---


Eine Umschlüsselung kann außerdem dazu beitragen, den sicheren Zugriff auf interne Betriebsmittel von außen zu vereinfachen. Falls verschiedene Dienste intern mit unterschiedlichen kryptographischen Verfahren arbeiten, kann die Umschlüsselung eingesetzt werden, um nach außen eine einheitliche kryptographische Schnittstelle anzubieten. Dies ist wiederum nur durch einen Firewall möglich, der Zugriff auf die netzübergreifende Kommunikation hat.

**Resümee:** Ein Firewall kann eine Umschlüsselung vornehmen, falls in den getrennten Netzen kein Protokoll vorhanden ist, daß von beiden Kommunikationspartnern für die Absicherung verwendet werden kann. Ein wichtiger Spezialfall ist das Anbieten einer einheitlichen kryptographischen Schnittstelle zum internen Netz durch einen Firewall.

### 5.2.4 Zertifikatskontrolle im Firewall

In einem Netz, das mit kryptographischen Mechanismen abgesichert wird, ist für die Authentisierung und anschließende Verschlüsselung der öffentliche Schlüssel des Kommunikationspartners erforderlich.

In großen verteilten Systemen ist es nicht mehr möglich, alle Zertifikate mit signierten öffentlichen Schlüsseln lokal auf jedem Client vorzuhalten. Vielmehr wird ein Client oder ein Server bei Bedarf ein Zertifikat von einem Zertifikatsserver anfordern.


Der Standardfall ist der Zugriff vom Internet auf ein X.509-Verzeichnis, das auf einem internen Rechner abgelegt ist. Typischerweise wird eine Firma in einem solchen Verzeichnis alle möglichen internen Zertifikate mit weiteren Zusatzinformationen abspeichern. Derartige Zusatzinformationen sollen oft nur intern zugänglich sein, beispielsweise Raum- und Telefonnummer eines Mitarbeiters oder seine Bürozeiten. Gleichzeitig wird man auch interne Zer-
5.2 *Nutzen von Firewalls für kryptographische Verfahren*

Zertifikate, die ausschließlich für firmeninterne Kommunikation bestimmt sind, auf demselben Server ablegen.


In Richtung des externen Netzes kann eine Zugriffskontrolle ebenfalls sinnvoll sein. Falls ein interner Rechner ein Zertifikat von einem externen Verzeichnisdienst (z. B. einem LDAP-Server) anfordert, kann der Firewall beispielsweise prüfen, ob das Zertifikat eine nach internen Richtlinien als “akzeptabel” klassifizierte Signatur trägt. Ist dies nicht der Fall, kann der Firewall die Weitergabe des Zertifikates unterbinden und sich gleichzeitig, sozusagen präventiv, die Zertifikats- und Schlüsselidentifikationen merken.

Sollte ein interner Benutzer später den Schlüssel des “inakzeptablen” Zertifikats dennoch benutzen, um beispielsweise ein E-Mail zu verschlüsseln, könnte der Firewall dies aufgrund der gespeicherten Schlüssel-IDs aufwandsarm\(^{33}\) erkennen und die Auslieferung der E-Mail verweigern.

Eine Zugriffskontrolle kann selbstverständlich auch auf anderen Attributen der jeweiligen Zertifikate bzw. der jeweiligen Verzeichnisdienste beruhen. Sehr interessant ist die Möglichkeit, über die in den Zertifikaten angegebenen Algorithmen zu erkennen, ob es sich um “Key Escrow”-Algorithmen handelt. Ein Firewall könnte somit Richtlinien durchsetzen, die die Benutzung dieser Algorithmen entweder vorschreiben oder auch verbieten.

**Resümee:** Ein Firewall kann die Benutzung und die Güte der verwendeten Zertifikate kontrollieren. Zertifikate, die nur für den internen Gebrauch bestimmt sind, werden von einem

\(^{33}\) Bei einer Policy, die verlangt, daß alle E-Mails mit bestimmten signierten Schlüsseln verschlüsselt werden, muß der Firewall in jedem Fall prüfen, welche Schlüssel-ID benutzt wird. Durch die vorangegangene Anfrage ist dieser Schlüssel in der Regel schon im “Firewall-Cache”, entweder in einer “allow”- oder einer “deny”-Liste. Sollte eine unbekannte ID benutzt worden sein, muß der Firewall mit erheblichen Mehraufwand seinerseits eine Anfrage an einen Verzeichnisdienst stellen, um den Schlüssel bewerten zu können.
Firewall gefiltert. Der Zugriff auf externe Zertifikate, die nicht den internen Anforderungen genügen, kann ebenfalls vom Firewall abgelehnt werden.

5.2.5 Anonymisierung der Kommunikation mit einem Firewall


Um auch auf der IP-Ebene Kommunikationsbeziehungen zumindest teilweise anonym eingehen zu können, kann wiederum der Firewall eingesetzt werden. Durch einen Proxy wird die Identität des internen Rechners geschützt. Anstelle des internen Rechners wird eine “Security Association” (im Falle von IPsec) zwischen Server und Firewall eingegangen. Der Server kann nun die Identität des Firewalls erkennen, jedoch nicht die Identität individueller Clients im internen Netz.

Resümee: Der Firewall kann trotz starker Authentisierung zur Anonymisierung von Kommunikationsbeziehungen eingesetzt werden.

6 Wechselwirkungen des kombinierten Einsatzes

In diesem Abschnitt werden die Aspekte diskutiert, die durch den kombinierten Einsatz von Firewall-Techniken und Kryptographie entstehen. Schwerwiegende Probleme entstehen dadurch, daß sich die eingesetzten Mechanismen gegenseitig behindern. Dies ist insbesondere dann der Fall, wenn ein Zugriffskontrollmechanismus Informationen benötigt, die verschlüsselt sind. Dieses Problem wird im Abschnitt 6.1 genauer dargestellt.

Ein weiteres schwerwiegendes Problem entsteht durch den hohen Rechenaufwand kryptographischer Verfahren, die auf dem Firewall eingesetzt werden. Der Engpaß Firewall wird noch prägnanter. Auf dieses Problem wird in Abschnitt 6.2 eingegangen.

6.1 Technische Aspekte des kombinierten Einsatzes

In diesem Abschnitt werden die Aspekte der Kombination von kryptographischen Verfahren (Abschnitt 4) mit den im Abschnitt 2.1 vorgestellten Firewall-Komponenten diskutiert. Das Vorgehen orientiert sich an den Firewall-Komponenten und diskutiert für jede Komponente die sich auch einer Kombination mit einem oder mehreren kryptographischen Verfahren ergebenden Aspekte.
6.1 Technische Aspekte des kombinierten Einsatzes

6.1.1 Packet Screens

IPsec und Skip: Diese beiden Protokolle können Teile von einzelnen IP-Datagrammen verschlüsseln und greifen somit direkt in die Möglichkeit der Packet Screen ein, die Datagramme zu überprüfen.


der Sender mit der Packet Screen die Verwendung von AH im “Tunnel-Mode” zusammen mit einem geeigneten Schlüssel aushandelt. Hierfür muß der Sender zunächst Kenntnis von der Packet Screen erlangen, was zum Beispiel durch ICMP-Nachrichten der Packet Screen erfolgen kann oder durch zusätzliche “Tunnel-Records” (TX) im DNS, die vom Sender abgefragt werden müssen. Derzeit wird noch erfor-scht, welche alternativen Verfahren ein Sender nutzen kann, um eine Packet Screen mit IPsec-Funktionen auf dem Weg zum Empfänger zu erkennen.


6.1.2 Proxies und Anwendungs-Gateways


6.2 Performanz bei kombiniertem Einsatz


Kombination mehrerer Verfahren: Wenn über Verfahren auf einer unteren Schicht (z. B. IPsec) bereits eine Authentisierung am Proxy durchgeführt wurde, kann der Proxy gegebenenfalls auf eine Entschlüsselung der Daten auf einer höheren Schicht verzichten und zum Beispiel TLS Verbindungen zwischen Client und Server ohne weitere Prüfung auch verschlüsselt durchlassen.

Resümee: Beim Einsatz von Proxy Servern gibt es keine Probleme mit IPSec, SKIP oder Protokollen unterhalb der IP-Schicht. Bereits auf der IP-Ebene wird eine Entschlüsselung durch IPSec oder SKIP vorgenommen, so daß die Daten unverschlüsselt auf der Anwendungs- ebene beim Proxy vorliegen.


6.2 Performanz bei kombiniertem Einsatz

Kryptographische Verfahren erfordern einen hohen Rechenaufwand. Firewalls sind bereits ohne den Einsatz kryptographischer Verfahren in HSN ein Engpaß, so daß sich dieses Problem durch den Einsatz kryptographischer Verfahren noch verschärfen wird.


Zur Verschlüsselung der Daten auf dem Proxy wurden die beiden Verfahren “3DES” und “RC4” verwendet, da dies die zur Zeit am häufigsten verwendeten Verfahren sind.

DES ist eine Block-Chiffre (“Block Cipher”), die bei jedem Durchlauf 64 Bit (bzw. 8 Byte) große Blöcke verschlüsselt. Kürzere Nachrichten müssen gegebenenfalls durch Anhängen von Fülldaten (“Padding”) vor der Verschlüsselung mit DES (oder 3DES) auf ein Vielfaches von 8 Byte verlängert werden. Dieses Padding bei Block-Chiffren führt zu einer Verlängerung der übertragenen Nachrichten. Im ungünstigsten Fall übersteigt die Nachricht nach dem Padding gerade die MSS (bzw. MTU), was die Generierung eines zusätzlichen, kurzen TCP-Segments oder allgemein eines zusätzlichen Datagramms erforderlich macht. Die Folge sind ein höherer Verarbeitungsaufwand in den Endsystemen. Außerdem müssen mehr Header-Informationen übertragen werden, wodurch ein geringerer Netzdurchsatz für die Übertragung von Nutzdaten verbleibt.


---


35 Da die einzelnen Blöcke unabhängig voneinander verschlüsselt werden, könnten Angreifer die Reihenfolge der Datenblöcke ändern. Um diese Manipulationsmöglichkeit auszuschließen, wird über das “Cipher Block Chaining” (CBC) eine Abhängigkeit zwischen den einzelnen Blöcken erzeugt.


Um die Performanzauswirkungen der Verschlüsselung aufzuzeigen, sind Messungen mit kurzen Transaktionen wenig sinnvoll, da deren Performanz stark durch die Verbindungsaufbauzeiten dominiert wird. Um Auswirkungen durch Verbindungsaufbauzeiten weitestgehend auszuschließen zu können, wurden Messungen mit 30 Sekunden langen Transaktionen durchgeführt. Da der Datendurchsatz bei etwa 40 parallelen Threads einbrach, wurden hauptsächlich Messungen mit bis zu 50 Threads durchgeführt. Zwei weitere Meßpunkte liegen bei 60 und 100 Threads.


Die Belastung des Firewalls durch die Verschlüsselung zeigen die CPU-Lastkurven\[40\] in Abbildung 25. Im Gegensatz zu der reinen Kopierfunktion, bei der die CPU-Last erst bei mehr als 40 parallelen Verbindungen sprunghaft ansteigt, benötigen die Verschlüsselungsfunktionen bereits bei einer Verbindung die gesamte Prozessorleistung des Firewalls. Dadurch steigt die CPU-Last auf dem Firewall auf einen Wert, der nur knapp unter der Anzahl der parallelen Verbindungen liegt.\[41\]


38 Die Verschlüsselungsfunktion stammt ebenfalls aus SSH. Aus patent- und namensrechtlichen Gründen (siehe auch [38], S. 397f) wird das Verfahren bei SSH als “Arcfour” bezeichnet.

39 Der Proxy wurde auf einer Bastion ausgeführt, also auf einem Rechner mit nur einer Netzschnittstelle (vgl. Abschnitt 2.2.2).

40 Die CPU-Last gibt die über einen Zeitraum (hier 15 Minuten) gemittelte Anzahl der Prozesse oder Threads an, die gleichzeitig die CPU beanspruchen.

41 Während zum Beispiel 100 Verbindungen aktiv sind, werden Benutzereingaben von dem Firewall nur mit sehr großen Verzögerungen bedient.

Diese Einschätzung gilt neben Firewalls auch für andere Systeme, die gleichzeitig mit vielen Clienten kommunizieren müssen (z. B. WWW-Server).
Abbildung 25: CPU-Last auf dem Firewall durch Verschlüsselung

7 Zusammenfassung


Der zweite wichtige Bereich für die Netzwerksicherheit ist der Einsatz kryptographischer Verfahren und Protokolle. In Abschnitt 3 wurde der Nutzen der Kryptographie für die Netzwerksicherheit benannt: Erstmalig ist es bei der Kommunikation über ein Rechnernetz möglich,


In Abschnitt 5.2 wurde der Nutzen von Firewalls für kryptographische Verfahren dargestellt. Firewalls werden zum einen als Ergänzung zu kryptographischen Verfahren eingesetzt, um die Auswirkungen von Administrations- und Softwarefehlern zu minimieren. Zum anderen kann ein Firewall eingesetzt werden, um Verschlüsselungsrichtlinien durchzusetzen oder als Verschlüsselungs-Gateway ein einheitliche kryptographische Schnittstelle zu einem Subnetz anzubieten.


In Zukunft gibt es mehrere Schwerpunkte, die noch genauer untersucht werden müssen. Es muß vertiefen analysiert werden, welche neuen Möglichkeiten sich durch den kombinierten Einsatz von Firewalls und kryptographischen Verfahren ergeben. Die in Abschnitt 5.4 und Abschnitt 5.2 dargestellten Aspekte bilden die Grundlage für die Erforschung neuer Anwendungen, die das volle Potential dieser wichtigen Schutztechniken für die Netzwerksicherheit nutzen können.


Abschließend ist festzustellen, daß die Vorteile für die Netzwerksicherheit, die sich aus einem kombinierten Einsatz kryptographischer Verfahren mit Firewalls ergeben, die bisher erkannten Nachteile bei weitem überwiegen. Das Internet braucht für eine weitere Entfaltung, insbesondere im kommerziellen Bereich, sowohl die Absicherung der lokalen Netze durch Firewalls als auch den Schutz individueller Kommunikationsbeziehungen durch kryptographische Verfahren.
<table>
<thead>
<tr>
<th>Abkürzung</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACL</td>
<td>Access Control List</td>
</tr>
<tr>
<td>AH</td>
<td>Authentication Header</td>
</tr>
<tr>
<td>ANSI</td>
<td>American National Standard Institute</td>
</tr>
<tr>
<td>AP</td>
<td>Authentication Protocol</td>
</tr>
<tr>
<td>ASCII</td>
<td>American Standard Code for Information Interchange</td>
</tr>
<tr>
<td>ATM</td>
<td>Asynchronous Transfer Mode</td>
</tr>
<tr>
<td>BSD</td>
<td>Berkeley Software Distribution</td>
</tr>
<tr>
<td>CA</td>
<td>Certification Authority</td>
</tr>
<tr>
<td>CBC</td>
<td>Cipher Block Chaining</td>
</tr>
<tr>
<td>CERT</td>
<td>Computer Emergency Response Team</td>
</tr>
<tr>
<td>CP</td>
<td>Connection Protocol</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>DAP</td>
<td>Database Access Protocol</td>
</tr>
<tr>
<td>DEA</td>
<td>Data Encryption Algorithm</td>
</tr>
<tr>
<td>DES</td>
<td>Data Encryption Standard</td>
</tr>
<tr>
<td>DFN</td>
<td>Deutsches Forschungs Netz</td>
</tr>
<tr>
<td>DMZ</td>
<td>Demilitarized Zone</td>
</tr>
<tr>
<td>DNS</td>
<td>Domain Name System</td>
</tr>
<tr>
<td>DSA</td>
<td>Directory Service Agent</td>
</tr>
<tr>
<td>DSS</td>
<td>Digital Signature Standard</td>
</tr>
<tr>
<td>DUA</td>
<td>Directory User Agent</td>
</tr>
<tr>
<td>ESP</td>
<td>IP Encapsulating Security Payload</td>
</tr>
<tr>
<td>FIPS</td>
<td>Federal Information Processing Standards</td>
</tr>
<tr>
<td>FTP</td>
<td>File Transfer Protocol</td>
</tr>
<tr>
<td>FWL</td>
<td>Firewall-Labor</td>
</tr>
<tr>
<td>HMAC</td>
<td>Hashed Message Authentication Code</td>
</tr>
<tr>
<td>HSN</td>
<td>High Speed Network</td>
</tr>
<tr>
<td>HTML</td>
<td>Hypertext Markup Language</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>ICMP</td>
<td>Internet Control Message Protocol</td>
</tr>
<tr>
<td>ICV</td>
<td>Integrity Check Value</td>
</tr>
<tr>
<td>IDEA</td>
<td>International Data Encryption Algorithm</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>IETF</td>
<td>Internet Engineering Task Force</td>
</tr>
<tr>
<td>IKE</td>
<td>Internet Key Exchange</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>IPSEC</td>
<td>IP Security (Working Group)</td>
</tr>
<tr>
<td>IS</td>
<td>Internet-Server</td>
</tr>
<tr>
<td>ISAKMP</td>
<td>Internet Security Association Key Management Protocol</td>
</tr>
<tr>
<td>ISO</td>
<td>International Standards Organisation</td>
</tr>
<tr>
<td>ITU</td>
<td>International Telecommunication Union</td>
</tr>
<tr>
<td>KDC</td>
<td>Key Distribution Center</td>
</tr>
<tr>
<td>Abkürzung</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------</td>
<td>------------</td>
</tr>
<tr>
<td>LAN</td>
<td>Local Area Network</td>
</tr>
<tr>
<td>LDAP</td>
<td>Lightweight Database Access Protocol</td>
</tr>
<tr>
<td>MAC</td>
<td>Message Authentication Code</td>
</tr>
<tr>
<td>MD</td>
<td>Message Digest</td>
</tr>
<tr>
<td>MSS</td>
<td>Maximum Segment Size</td>
</tr>
<tr>
<td>MTU</td>
<td>Maximum Transfer Unit</td>
</tr>
<tr>
<td>NAT</td>
<td>Network Address Translation</td>
</tr>
<tr>
<td>NIST</td>
<td>National Institute of Standards and Technology</td>
</tr>
<tr>
<td>NNTP</td>
<td>Net News Transport Protocol</td>
</tr>
<tr>
<td>OSI</td>
<td>Open System Interconnection</td>
</tr>
<tr>
<td>PGP</td>
<td>Pretty Good Privacy</td>
</tr>
<tr>
<td>PKI</td>
<td>Public Key Infrastructure</td>
</tr>
<tr>
<td>PKIX</td>
<td>Public Key Infrastructure (X.509)</td>
</tr>
<tr>
<td>PS</td>
<td>Primary Server</td>
</tr>
<tr>
<td>RFC</td>
<td>Request for Comments</td>
</tr>
<tr>
<td>RPC</td>
<td>Remote Procedure Call</td>
</tr>
<tr>
<td>RR</td>
<td>Resource Record</td>
</tr>
<tr>
<td>RSA</td>
<td>Rivest, Shamir, Adleman</td>
</tr>
<tr>
<td>SAD</td>
<td>Security Association Database</td>
</tr>
<tr>
<td>SDL</td>
<td>Specification and Description Language</td>
</tr>
<tr>
<td>SecDNS</td>
<td>Secure Domain Name System</td>
</tr>
<tr>
<td>SHA</td>
<td>Secure Hash Algorithm</td>
</tr>
<tr>
<td>SIG-RR</td>
<td>Signature Resource Record</td>
</tr>
<tr>
<td>SKIP</td>
<td>Simple Key-Management for Internet Protocols</td>
</tr>
<tr>
<td>SMTP</td>
<td>Simple Mail Transport Protocol</td>
</tr>
<tr>
<td>SPI</td>
<td>Security Parameter Index</td>
</tr>
<tr>
<td>SPKI</td>
<td>Simple Public Key Infrastructure</td>
</tr>
<tr>
<td>SSH</td>
<td>Secure Shell</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
<tr>
<td>THP</td>
<td>TLS Handshake Protocol</td>
</tr>
<tr>
<td>TLP</td>
<td>Transport Layer Protocol</td>
</tr>
<tr>
<td>TLS</td>
<td>Transport Layer Security</td>
</tr>
<tr>
<td>TRP</td>
<td>TLS Record Protocol</td>
</tr>
<tr>
<td>UDP</td>
<td>User Datagram Protocol</td>
</tr>
<tr>
<td>VLAN</td>
<td>Virtual LAN</td>
</tr>
<tr>
<td>VPN</td>
<td>Virtual Private Network</td>
</tr>
<tr>
<td>WWW</td>
<td>World Wide Web</td>
</tr>
</tbody>
</table>
Literatur


