Universität Potsdam
Institut für Informatik
WS 2003/2004
Prof. Dr. K. Rebensburg
Dipl. Inf. T. J. Wilke

Seminar

Umfassende Absicherung komplexer IT-Infrastrukturen

Autoren: Mario Bärmann, Lisa Böhringer, Marc Coym, Manuel Dräger, Kolja Engelmann, Christian Fricke, Michael Hübner, Peter Kirchner, André Kloth, Sebastian Krahmer, Björn Mehler, Vivien Mende, Jan Schlößin, Tobias Tebner, Matthias Thränhardt, Stephan Uhlmann, Ron Vollandt, Sebastian Weber

Datum: Potsdam, 09. Februar 2004

Inhaltsverzeichnis

1 E-Government

M. Coym

1.1 Einleitung

Am 18. September 2000 hat Bundeskanzler Gerhard Schröder die E-Government-Initiative "BundOnline 2005" der Bundesregierung vorgestellt. Diese besagt, das bis zum Jahr 2005 alle internetfähigen Dienstleistungen der Verwaltungen des Bundes online verfügbar sein sollen. Dabei soll die Webpräsenz http://www.bund.de die zentrale Anlaufstelle für E-Government der Bundesregierung und Bundesverwaltung sein. E-Government soll in Zukunft nicht nur Informationen wie Öffnungszeiten oder Kontaktinformationen bereitstellen, sondern online Dienstleistungen für Bürger und Unternehmen bieten, sodass die oft umfangreichen Behördengänge reduziert und optimiert werden können. Als schönes Beispiel wäre hier z.B. das Onlineangebot des Arbeitsamtes http://www.arbeitsamt.de zu nennen. Hierbei wird der Entwicklung Rechnung getragen, die die Wirtschaft innerhalb der letzten fünf Jahre im E-Businnes-Sektor erreicht hat und die sich als sehr erfolgreich herausgestellt hat.
Die E-Government-Initiative fordert jedoch nicht nur eine verbesserte elektronische Kommunikation nach aussen, sondern auch eine innerhalb der Behörden und Verwaltungen, mit der Zielsetzung die Bearbeitungsvorgänge zu beschleunigen und zu rationalisieren. Dabei betreten viele Behörden und Verwaltungen absolutes Neuland und müssen daher nicht nur über die Vorteile von E-Government, sondern auch über die Risiken und möglichen Fehler aufgeklärt werden.
So werden beim E-Government oft sensible Daten von Bürgern und Unternehmen über unsichere Medien, wie dem Internet übertragen, die ursprünglich persönlich bzw. per Post zugesandt wurden und seither abgeschottete IT-Systeme an das Internet angeschlossen um freien Informationsaustausch mit anderen Behörden zu gestatten.

"Wenn wir die Risiken im Blick haben, können wir die Chancen der Informationstechnik nutzen"

- Innenminister Otto Schily -

1.2 E-Government-Handbuch

Das Bundesministerium für Sicherheit in der Informationstechnik (BSI) hat daher ein Handbuch für die Einführung von E-Government erstellt, mit dem den Behörden ein Leitfaden gegeben wird, der auf Gefahren und mögliche Fehler bei der Einführung dieser

neuen Techniken hinweisst und auch Lösungs-vorschläge bietet, die jedoch reinen Empfehlungscharakter haben.
So steht ganz klar die "E-Government-Devise" im Vordergrund:

"Erst planen, dann handeln"

Dadurch sollen kostspielige Inkompatibilitäten und Doppelentwicklungen in den unterschiedlichen Behörden von vornherein vermieden werden. Des Weiteren müssen die Systeme anwenderfreundlich und sicher sein, denn im Endeffekt entscheidet der Bürger durch das Nutzen der Systeme über deren Erfolg oder Misserfolg. Kein Anwender wird ein System nutzen bei dem er erst Unmengen von Zusatzsoftware installieren oder komplizierte Einstellungen und Schlüsseleingaben vornehmen muss. Und genauso wenig erhöht ein System das Vertrauen der Nutzer in dem an jedem zweiten Tag eine Sicherheitslücke oder Manipulation bekannt wird.
Aus diesen Forderungen wird klar das E-Government auf bekannten und weit verbreiteten IT-Plattformen aufsetzen und dabei die z.Z. höchstmögliche Fehlerunanfälligkeit des Systems erreicht werden muss. Das BSI Handbuch sagt zu dem Kontext folgendes.

"Der Einsatz von exotischen oder unerprobten Systemen sollte - wo möglich - vermieden werden. Darüber hinaus ist eine weitgehende Standardisierung von Datenformaten und -schnittstellen (z. B. Online-Formularen) anzustreben."

Daraus ergibt sich eine weitere Forderung an die Systeme. Es müssen Datenformate gefunden werden die für die nächsten 100 Jahre gültig und lesbar bleiben, was bei der rasanten Evolution in der IT-Branche nicht selbstverständlich ist, wie sich gerade im Automobilbau bzw. beim Militär zeigt. Denn dort besteht die Forderung, das für die verbaute Hardware mindestens 20 Jahre lang Support gewährleistet sein muss.
Und wie Max Weber mit den Worten "quod non est in actis, non est in mundo" ausdrückte, kann ein Schriftstück in Behörden und Verwaltungen ein Beweis für die Existenz oder Nichtexistenz eines Menschen sein, und wer hätte es schon gern das man kein Bürger diese Staates mehr ist nur weil seine digitale Geburtsurkunde nach ein paar Jahrzehnten nicht mehr lesbar ist, weil das Dateiformat von keinem Programm mehr verstanden wird?

1.3 Anforderungen an den Datenschutz

Doch auf welche Sicherheitsaspekte muss man bei der Wahl der IT-Systeme, die für den Einsatz im E-Government gedacht sind, denn nun eingehen? Das BSI-Handbuch geht bei der Frage der Sicherheit und des Datenschutzes auf folgende Anforderungen ein.

1.3.1 Vertraulichkeit

Daten und Informationen, die zwischen Behörde und Bürger ausgetauscht werden, müssen vor dem Mitlesen durch Dritte geschützt werden.
Die bisherige Kommunikation zwischen Behörde und Bürger wird per Post oder persönlichem Gespräch realisiert. Dabei ist die Vertraulichkeit im Allgemeinen als hoch einzuschätzen, was sich aber bei einem Gespräch an einem Schalter schon nicht mehr behaupten lässt.

Um diese hohe Vertraulichkeit auch in der elektronischen Kommunikation zu gewährleisten muss auf entsprechende Verschlüsselung der Kommunikation zurückgegriffen werden. Dabei ist die allgemeine Bewertung der Stärke eines Verschlüsselungsmechanismus zu beachten. Verschlüsselungen die behaupten unbrechbar zu sein und dabei nicht den Algorithmus veröffentlichen sind als unzureichend anzusehen. Vielmehr sollte man auf Verschlüsselungen zurückgreifen, deren Algorithmus bekannt ist und die von Kryptologen als sicher eingestuft worden sind. Die eben, für Verschlüsselungen, genannte Forderung wird auch als Kerckhoff'sches Prinzip bezeichnet.

1.3.2 Integrität

Die Integrität von Daten besagt, das bei einer Übertragung keinerlei Manipulationen, ob bewusst durch Dritte oder unbewusst durch Hard- oder Softwarefehler auftreten. Um Manipulationen zu verhindern bzw. festzustellen werden digitale Signaturen und Message Authentication Codes benutzt, die eine fälschungssichere Checksumme der Daten enthalten und mit den zu schützenden Daten übertragen werden.
Falls die Daten verändert worden sind, so ist dieser Umstand spätestens beim Erhalt festzustellen. Dabei ist weder die Rekonstruktion der Daten möglich noch der Umfang der Manipulation festzustellen. Bei einer fehlerhaften Übertragung ist in jedem Fall eine Neusendung der Daten nötig.

1.3.3 Authentizität

Die Authentizität unterscheidet zwischen der Authentizität der Daten und der Authentizität des Kommunikationspartners.
Die Authentizität der Daten besagt, dass die Daten tatsächlich von meinem Kommunikationspartner stammen und damit auch in vollem Umfang seiner Zustimmung unterliegen. Die Authentizität der Daten wird, wie die Datenintegrität durch digitale Signaturen oder Message Authentication Codes erreicht. Denn im Endeffekt ist bei einer Manipulation der Daten die Authentizität dieser ebenfalls nichtig, da man nicht mehr davon ausgehen kann das der Kommunikationspartner die Daten in dieser Form gesendet hat und damit seine Zustimmung zu den Aussagen der Daten erlischt.
Die Authentizität des Kommunikationspartners lässt sich auf der Bürgerseite über einen elektronischen Ausweis oder über eine Passworteingabe (PIN/TAN etc.) realisieren. Der elektronische Ausweis besteht aus einer von der Behörde signierten Datei, die der Bürger in jedes von ihm versendete Dokument einbinden kann. Hierbei ist aber immer der Punkt 3.5 (Erforderlichkeit) zu beachten.
Die Behörde ihrerseits kann ihr Webangebot mit Zertifikaten signieren und eine SSL Verbindung anbieten, die eine eindeutige Zuordnung der Website zur Behörde garantiert.

1.3.4 Verfügbarkeit

Die Verfügbarkeit gibt an, wie gross die Zeitdauer der Ausfälle, verursacht durch Fehler oder Wartungsarbeiten, in einem IT-System ist.
Dieser Punkt wird auf den ersten Blick meistens dem Service zugesprochen, jedoch ist diesem im E-Government, insbesondere in Bezug auf Einhaltung von Fristen wegen juristischer Forderungen, eine größere Beachtung zu schenken. Bei Output-Servern, z.B. Formular- oder Informations-Server etc. ist die Anforderung an die Verfügbarkeit nicht so hoch wie bei Input-Server, z.B. die virtuelle Poststelle, bei denen wie schon erwähnt Fristen einzuhalten sind.

1.3.5 Erforderlichkeit

In jeder E-Government Anwendung ist vorher festzustellen inwieweit persönliche Daten der Nutzer bzw. Bürger abzufragen oder zu speichern sind.
Bei rein informellen Dienstleistungen, wie Öffnungszeiten, Bekanntmachungen oder Formulardownloads sind keine persönlichen Daten erforderlich. Dies gilt ebenso für IP-Adressen, die, falls sie zur statistischen Auswertung benötigt werden, so weit anonymisiert werden müssen, z.B. indem man die letzten drei Ziffern der IP-Adresse löscht, dass keine Rückschlüsse auf den Bürger vorgenommen werden können.
Auch Cookies sind in E-Governmentanwendungen höchsten temporär, also für die Zeit des Besuches eines Webangebots zulässig. Falls doch Daten zu speichern sind, z.B. bei Anträgen müssen die Systeme so konfiguriert sein, dass die persönlichen Daten zum frühstmöglichen Zeitpunkt wieder gelöscht werden.

1.3.6 Datenvermeidung und Datensparsamkeit

Der Anspruch der Datenvermeidung und Datensparsamkeit resultiert aus dem Punkt 3.5 (Erforderlichkeit). Schon in der Konzeptionsphase der E-Government-Systeme soll auf einen sparsamen Umgang mit persönlichen Daten eingegangen werden. Dadurch wird nicht nur der Zweckdurchbrechung (siehe 3.7) entgegengewirkt, sondern auch die rechtlich zugesicherten Anonymität der Nutzer gefördert.

1.3.7 Zweckbindung

Einen wichtigen Platz im Datenschutz nimmt die Zweckbindung ein. Sie fordert das Daten die zu einem bestimmten Zweck, z.B. von einer bestimmten Behörde oder Abteilung erhoben worden sind, nur der Behörde zur Verfügung stehen, die für diese Aufgabe/-Zweck verantwortlich ist. Jeder anderen Abteilung oder Behörde ist der Zugriff auf diese Daten untersagt. Wird von Unberechtigten auf die Daten zugegriffen, so spricht man von einer Zweckdurchbrechung, die nur auf gesetzlicher Grundlage legitim ist oder wenn der Betroffene eingewilligt hat.
Hier gilt der Grundsatz der informationellen Gewaltenteilung, die einer Profilbildung von Bürgern und Unternehmen entgegenwirken soll. Denn in der digitalen Welt ist diese Profilbildung leichter zu realisieren als in der bisherigen analogen, in der schon die physikalische Masse an Informationen dem entgegenwirkt.

1.4 Schutzbedarf

Jedoch muss man auch zwischen zu schützenden und weniger zu schützenden Daten unterscheiden. Ein Webangebot in dem nur die Öffnungszeiten bzw. eine Auflistung aller benötigten Unterlagen für die jeweiligen Anträge veröffentlicht sind, braucht einen minderen Schutz als z.B. der Eingangsserver für die digitale Steuererklärung. Das BSI hat sich auf vier Einstufungen geeinigt, die die Schutzwürdigkeit, in Abhängigkeit von dem zu erwartenden Schaden für Anbieter und Nutzer einer E-Government Dienstleistung angibt.

Ein **“niedriger“** Schutzbedarf besteht wenn:
- eine Beeinträchtigung des informationellen Selbstbestimmungsrechtes durch den Einzelnen nicht vorliegt;

- ein möglicher Missbrauch personenbezogener Daten vernachlässigbare Auswirkungen auf die gesellschaftliche Stellung oder die wirtschaftlichen Verhältnisse des Betroffenen hätte;
- für den Betreiber der Anwendung nur eine sehr geringe Ansehens- oder Vertrauensbeeinträchtigung zu erwarten wäre.

Ein **"mittlerer"** Schutzbedarf besteht wenn:
- eine Beeinträchtigung des informationellen Selbstbestimmungsrechtes durch den Einzelnen noch als geringfügig eingeschätzt würde;
- ein möglicher Missbrauch personenbezogener Daten nur geringfügige Auswirkungen auf die gesellschaftliche Stellung oder die wirtschaftlichen Verhältnisse des Betroffenen hätte;
- für den Betreiber der Anwendung nur eine geringe Ansehens- oder Vertrauensbeeinträchtigung zu erwarten wäre.

Ein **"hoher"** Schutzbedarf besteht wenn:
- eine erhebliche Beeinträchtigung des informationellen Selbstbestimmungsrechtes des Einzelnen möglich erscheint;
- ein möglicher Missbrauch personenbezogener Daten erhebliche Auswirkungen auf die gesellschaftliche Stellung oder die wirtschaftlichen Verhältnisse des Betroffenen hätte;
- für den Betreiber der Anwendung eine breite Ansehens- oder Vertrauensbeeinträchtigung zu erwarten wäre.

Ein **"sehr hoher"** Schutzbedarf besteht wenn:
- eine besonders bedeutende Beeinträchtigung des informationellen Selbstbestimmungsrechtes des Einzelnen möglich erscheint;
- ein möglicher Missbrauch personenbezogener Daten für den Betroffenen den gesellschaftliche oder wirtschaftlichen Ruin bedeuten würde;
- für den Betreiber der Anwendung eine landes- bzw. bundesweite Ansehens- oder Vertrauensbeeinträchtigung denkbar ist.

Das hier angesprochene informationelle Selbstbestimmmungsrecht, ist das Recht jedes Bürgers der Bundesrepublik, selbst zu entscheiden, wer an welche persönlichen Informationen über ihn gelangen kann.
Dabei muss für jede E-Government-Anwendung diese Bewertung festgestellt werden und daraufhin entsprechende technische Sicherungsmaßnahmen im IT-System ergriffen werden, um das informationelle Selbstbestimmungsrecht des Einzelnen konsequent zu beachten. Ist die Beachtung dieses Rechtes gar nicht oder nur teilweise möglich, so muss unter Umständen auf die Realisierung dieser Dienstleistung verzichtet werden.

1.5 Sicherer Webauftritt

Da alle E-Government-Dienstleistungen online angeboten werden hat das BSI auch Empfehlungen für einen sicheren Webauftritt herausgegeben. Sie sind im Kapitel 4 "Sicherer Internet-Auftritt im E-Government" beschrieben.
Falls die Behörde den Server selber stellt, dem so genannte Inhouse-Betrieb, ist im trivialsten Fall eine strikte Trennung des Behördenlan mit dem Webserver einzuhalten. Inhalte des Webservers sind also per Wechselmedien zu übertragen.

Die kostenintensiveren Varianten des Inhouse-Betriebes erfordern einen oder mehrere Application Gateways und Paketfilter, um einen hohen Schutz des Behördenlan und der darin enthalten Daten und Datenbanken zu gewährleisten. Zusätzlich ist natürlich entsprechend qualifiziertes Servicepersonal nötig um die Technik zu konfigurieren und zu warten.
Eine weitere Möglichkeit E-Government-Dienste bereitzustellen, ist Webhostingangebote externer Firmen zu nutzen. Der Informationsverbund Berlin-Bonn (IVBB) bietet obersten Bundesbehörden die Möglichkeit Webangebote zu hosten und übernimmt dabei sämtliche Sicherheitsaufgaben. Der IVBB hat verschiedenste Möglichkeiten, unter anderem auch einen abgesicherten Inhousebetrieb, das E-Government-Angebot der Behörde online zu stellen.
Um einer Manipulation des Webauftrittes vorzubeugen sind in allen besprochenen Fällen entsprechende Zugriffsrestriktionen bei Servicearbeiten zu beachten. Darunter fallen verschlüsselte Verbindungen wie SSL bzw. SecureShell und Einmal-Passwortverfahren wie S/Key. Des Weiteren sind die Datenformate für die Onlineformulare so zu wählen, das einer einfachen Manipulation entgegen gewirkt wird. Diesen Schutz bietet z.B. das PDF-Format. Selbstverständlich gelten weiterhin die gängigen Anforderungen im Bereich des Brandschutzes und der Sicherheitsvorkehrungen für die schon bestehende IT-Infrarstruktur.
Falls die Behörde einen externen Webhoster mit der Bereitstellung des E-Government-Angebotes beauftragt, so hat sie darauf zu achten das es sich um dedizierte Server handelt, d.h. nur Angebote der Behörde laufen auf dem Server. Dies ist nötig, um einen Angriff oder eine Manipulation durch andere Servernutzer zu verhindern.
Ebenso sind alle genannten Sicherheitsforderungen zu übertragen, diese vertraglich festzuhalten und deren Einhaltung im Nachhinein durch die Behörde zu kontrollieren.

1.6 Datenschutz in der Verwaltung

Um dem Schutz der Daten innerhalb der Daten verarbeitenden Stellen zu genügen sind auch hier einige Regeln zu beachten.
Hierzu gehört eine überlegte und restriktive Benutzer- und Rechteverwaltung, die jedem Mitarbeiter der Behörde seine individuellen Rechte vorgibt. So braucht eine Sekretärin im Allgemeinen keine Administrationsrechte auf ihrem PC. Ausserdem wird so das Prinzip der informationellen Gewaltenteilung unterstützt, da nur die Mitarbeiter einer Abteilung Zugriff auf die von ihnen erhobenen Daten haben. Da mit der Einführung von
E-Government Neuanschaffungen anstehen ist dabei nur auf Betriebssysteme zurückzugreifen, die diese Rechtevergabe auch anbieten und unterstützen, wie z.B. Windows NT/2000/XP oder Linux Distributionen.
Die Kommunikation der Behördenmitarbeiter sollte über eine virtuelle Poststelle laufen, die als zentraler Kommunikationsgateway fungiert. Ein- und ausgehende E-Mails werden hier ver- bzw. entschlüsselt und Zertifikate eingefügt bzw. überprüft. Interner E-Mailverkehr der das sichere Behördennetz nicht verlässt muss nicht verschlüsselt werden.
Der Zugang der Mitarbeiter zum Internet kann daher per Firewall auf Port 80 für http reduziert bzw. für speziell benötigte Programme erweitert werden. Auf Downloads sollte im allgemeinen und besonders bei unbekannten oder unsicheren Quellen verzichtet werden. Um eine völlige Sichheit vor Trojanern und Viren zu erreichen wäre ein Remote-Desktop-System denkbar, auf das sich die Mitarbeiter einloggen und dann von dort aus

das Web durchsuchen. In diesem Fall ist nur der Port für die Remote-Desktop-Software zu öffnen.
Ebenso unerlässlich ist die systematische Schulung der Mitarbeiter, sowie deren Personalvertretungen. Dabei sollen die Grundlagen über die eingesetzten Systeme, deren Techniken und die Risiken, aber auch Konsequenzen bei Verletzen der Datenschutzvorschriften verständlich gemacht werden, um z.B. einer unzulässigen Datenübermittlung an Dritte, egal ob fahrlässig oder vorsätzlich, vorzubeugen. Hierbei ist im besonderen auf den Punkt Social Engineering einzugehen.
Um die IT-Systeme ständig auf dem sicherheitstechnischen Stand der Dinge zu halten sind Administratoren für das Einspielen von Patches und Bugfixes zuständig, um mögliche Schwachstellen des Systems so schnell wie möglich zu schließen und die Virensignaturen der Scanner zu aktualisieren.
Um einen Einbruch in das System frühzeitig zu bemerken sind in regelmässigen Abständen Logdateien auszuwerten und auf mögliche Angriffsversuche von aussen, wie auch von innen zu untersuchen. Daher sind alle Änderungen an sicherheitsrelevanten Systemen, wie Firewalls oder Servern zu loggen, um alle Eingriffe von aussen im Bedarfsfalle nachvollziehen zu können.

1.7 Und der Bürger?

Doch ein System ist immer nur so stark wie sein schwächstes Glied, namentlich Otto-Normal-Bürger zuhause vor seinem Rechner, der E-Government nutzen möchte. Hier ist Aufklärungsarbeit gefordert, damit nicht alle Sicherheitsbemühungen der E-Government-Initiative vergebens sind weil die Nutzer kein Sicherheitsverständnis beim eigenen PC haben.
Doch wie soll diese Aufklärung aussehen? Der Blaster Wurm hat es im Spätsommer 2003 eindeutig gezeigt. Innerhalb kürzester Zeit verbreitete er sich über hundertausende von Rechnern obwohl der Patch für das Sicherheitsloch im DCOM Service seit mehreren Wochen bereitstand.
Ein aktuellerer Fall (Februar 2004) zeigt die immer noch grosse Sorglosigkeit der Bürger, aber auch das hohe Erfolgspotential von Social Engineering. Der E-Mail-Wurm “MyDoom“ sprengt alle bisherigen Verbreitungsrekorde. Dies ist umso unverständlicher, wenn man bedenkt, das es sich um einen Schädling handelt der eine aktive Starthilfe, in Form von Öffnen von Mailanhängen benötigt um sich auszubreiten, im Gegensatz zum Blaster Wurm, der selbständig ein unsicheres System infiziert und für seine Verbreitung nutzt.

1.8 Zusammenfassung

Der Leitfaden, den das BSI Handbuch zur Einführungen von E-Government darstellt ist eine gute und umfangreiche Richtlinie, an die sich die jeweiligen E-Government-Initiatoren der Behörden möglichst halten sollten, um den beschriebenen Schutz der Daten auch wirklich zu gewährleisten, das Vertrauen der Bürger zu stärken und damit letztlich den Erfolg von E-Government in Deutschland zu sichern.
Es wird auf viele Kriterien für sicheres E-Government eingegangen und deren Umsetzung wird klar skizziert.
Das Handbuch hinterlässt jedoch auch einige Unklarheiten. So ist z.B. nicht eindeutig geklärt wie sich die informationelle Gewaltenteilung konsequent einhalten und auch

kontrollieren lässt. Das aktuelle Beispiel Mautsystem zeigt wie schnell es zu einer Zweckdurchbrechung kommen kann. Das System ist noch nicht mal in Betrieb und schon fordern Strafverfolgungsbehörden Zugriff auf die Daten, obwohl deren Erfassung nur auf Grund der Mautgebührenabrechnung legitimiert ist.
Ein weiteres ungelöstes Problem ist der Rechner des Bürgers, der nicht zum Verfügungsbereich der Behörde gehört und daher bei allen Sicherheitsvorkehrungen weitgehend aussen vor bleibt. Hier muss eine Aufklärung und Sensibilisierung des Bürgers für mehr Sicherheit am eigenen Rechner erfolgen.

Literaturverzeichnis

[1] Dr. Hauschild (BSI) Dr. Isselhorst. Einleitung und Übersicht. *E-Government-Handbuch*, 3:1–7, 2002.

[2] Dr. Fuhrberg (BSI) Dr. Niggemann (BSI), Dr. Wolf (BSI). Sicherer Internetauftritt im E-Government. *E-Government-Handbuch*, 6:1–49, 2002.

[3] Reinhard Gantar. Tauziehen ums Bürger-Paradies. *c't*, 11:88–90, 2003.

[4] Thomas Knaak. Datenschutzgerechtes E-Government. *E-Government-Handbuch*, 2:1–66, 2003.

[5] Stefan Reinke. Stadt, Land, Verdruss. *Chip*, 3:204–209, 2003.

[6] Dr. Marc Fischlin (FhI-SIT) Robert Nitschke (NOVOSEC), Dr. Harald Ritter (NOVOSEC). Authentisierung im E-Government. *E-Government-Handbuch*, 2:1–61, 2002.

[7] Christiane Schulzki-Haddouti. Wunderdroge Signatur. *c't*, 21:42, 2003.

2 Architektur des Microsoft Windows NT-Betriebssystems, DCOM und .NET

M. Bärmann, P. Kirchner

2.1 Einführung

Diese Dokumentation behandelt Grundlegende Sicherheitstechnische Erkenntnisse beim Umgang mit .NET und DCOM sowie dem NT Betriebssystem.
Windows NT Betriebssystem: Microsoft Windows 2000 ist das Resultat aus elf Jahren Produktentwicklung, die 1988 mit der Entwicklung von Windows NT 3.0 begann. Die Ziele für das erste Release im Sommer 1993 waren damals Kompatibilität zu OS/2 und POSIX, Sicherheit, Unterstützung für Multiprozessorsysteme und Netzwerke und nicht zuletzt sollte das System zuverlässig sein. Das erste Release unterstütze als Zielplattform Intel i386, Intel i486, MIPS R4000 und Digital Alpha. Da die erste Version noch sehr groß und langsam war, folgte im Herbst des nächsten Jahres schon die Version 3.5, die kleiner, schneller und zuverlässiger war. Zudem wurde nun auch der IBM PowerPC unterstützt.
Im Sommer 1996 erschien dann die Version 4.0, die mit dem gleichen User-Interface wie Windows 95 aufwartete und zudem neue Technologien von Windows 95 mit aufnahm. Windows 2000, im Grunde Windows NT 5.0, wurde schließlich am 15.12.1999 veröffentlicht. Windows XP wurde dann im Jahr 2001 veröffentlicht und enthält gegenüber Windows 2000 architektonisch nur geringfügigere Änderungen.
DCOM (Distributed COM): DCOM wurde 1996 von der Firma Microsoft als neuer Begriff eingeführt, um einen Fokus auf die Verteilte Programmierung zu setzen. MS erkannte das es nicht ausreicht Komponenten nur auf lokalen Maschinen zu haben. Generell gesagt ist DCOM ist es eine Erweiterung des Common Objekt Modells (COM). Es liefert einen vollständigen Object Request Broker für die Transparente Kommunikation über die Rechnergrenzen hinweg.
.NET: Das Microsoft .NET-Framework ist eine noch sehr junge Plattform zum Erstellen, Verteilen und Ausführen von Web-Services und Anwendungen. Es liefert eine sehr effiziente Umgebung, die auf Standards basiert und viele Programmiersprachen unterstützt.

2.2 Allgemeine Aspekte zu Windows NT

2.2.1 System- und Gebrauchsbeschreibung

Vorweg sei zu erwähnen, dass die Windows NT-Architektur viel zu umfangreich ist, um sie hier nur annähernd zu erfassen. Daher wird nur auf die konzeptionelle Struktur eingegangen und die Bereiche erläutert, die eine Relevanz für die Sicherheit des Systems aufweisen.

Die Windows NT-Architektur teilt sich in zwei Teile, in dem Code ausgeführt werden kann: dem Kernel-Modus und dem User-Modus. Der Kernel-Modus besitzt privilegierte Rechte, die es ausführbarem Code erlauben, auf den gesamten Systemspeicher zuzugreifen und alle CPU-Instruktionen auszuführen. Diese Modi werden durch die Hardware unterstützt, indem die Intel-Architektur und ähnliche Plattformen so genannte Ringe definieren, in denen abgestuft der Zugriff auf den Arbeitsspeicher und CPU-Instruktionen eingeschränkt wird. Geschichtlich bedingt verwendet Windows NT nur zwei von vier verfügbaren Ringen, da Hardwareplattformen wie Compaq Alpha und Silicon Graphics MIPS nur zwei Ringe definieren. Ring 0 ist in Windows NT der Kernel-Modus und Ring 3 der User-Modus.
Das Betriebssystem (BS) und Gerätetreiber, die im Kernel-Modus ausgeführt werden, teilen sich einen einzelnen, gemeinsamen virtuellen Adressraum. Dadurch bedingt bietet Windows NT keinerlei Schutz gegen Lese- und Schreibvorgänge innerhalb des Kernel-Modus. D.h. auch, dass das BS und Gerätetreiber jegliche Sicherheitskonzepte umgehen können.
Gerätetreiber sind also eine potenzielle Gefahr für die Sicherheit des Systems. Daher ist es auch nur Administratoren erlaubt, dem System Gerätetreiber hinzuzufügen. Eine kleine Hilfestellung bietet die in Windows 2000 eingeführte Signierung von Gerätetreibern, die sicherstellt, dass signierte Gerätetreiber von Microsoft überprüft wurden.
Prozesse im User-Modus besitzen jeweils einen eigenen virtuellen Adressraum. Dieser Speicherbereich ist vor Zugriffen von anderen Prozessen geschützt, es sei denn, es werden gemeinsame Speicherbereiche definiert.
Für Prozesse ist erwähnenswert, dass jeder Prozess eine Liste von geöffneten Systemressourcen verwaltet, die für alle Threads in diesem Prozess verfügbar sind, und einen Sicherheitskontext besitzt, der festlegt, unter welchem Zugangskonto dieser Prozess läuft und welche Privilegien dieser Prozess hat. Jeder Thread innerhalb eines Prozesses kann zudem einen eigenen Sicherheitskontext besitzen. Dies ermöglicht es Serveranwendungen, die Person des Clients anzunehmen und in dessen Sicherheitskontext Funktionen auszuführen. Windows NT kann mehrere Prozesse in so genannten Jobs zusammenfassen. Jobs dienen u.a. der Nachverfolgung von Überwachungsdaten und dem Mangel, dass Windows keine strukturierten Prozessbäume besitzt. Prozesse wissen dennoch, welcher Prozess sie gestartet hat. Diese üntergeordneten"Prozesse werden allerdings nicht automatisch beendet, wenn ein übergeordneter Prozess beendet wird.
Es gibt vier Prozesstypen, die im User-Modus laufen: System-Support-Prozesse, Dienst-Prozesse, Umgebungs-Subsysteme und Benutzeranwendungen.
System-Support- Prozesse sind beispielsweise der Logon-Prozess, der Session-Manager, der Service Control Manager und das Local-Security-Authority-Subsystem, welches essentiel für die Sicherheit in Windows NT ist, weswegen wir darauf später noch detaillierter eingehen. Service-Prozesse sind Prozesse, die Dienste hosten, wie z.B. dem Taskplaner und dem Spooler.
Die Umgebungs-Subsysteme sind ein wesentlicher Bestandteil der Windows NT-Architektur. Sie bieten den Zugriff auf native BS-Dienste. Mit Windows 2000 werden drei Subsysteme mitgeliefert: Win32, POSIX und OS/2. Win32 ist von den drei System der mit Abstand wichtigste und ist vital für das System. Die anderen beiden Subsyteme werden nur bei Bedarf ausgeführt.
Das Win32-System ist zweigeteilt, dessen beiden Teile im User-Modus und im Kernel-Modus existieren. Der User-Modus-Part übersetzt dokumentierte Win32-API-Funktionen in die entsprechenden undokumentierten Kernelmodus-Systemdienstaufrufe in die Exe-

kutive (Bereich innerhalb des Kernel-Modus) und den Kernel-Modus-Part des Win32-Subsystems. Dieser enthält zudem den Fenster Manager und das Graphic Device Interface (GDI). Diese Funktionalität wurde erst mit Windows NT 4.0 in den Kernel-Modus verschoben, um eine Leistungssteigerung zu erzielen, da so weniger Kontextwechsel notwendig sind. Die Systemstabilität wurde dadurch nicht beeinträchtigt, da das Win32-Subsystem ein vitaler Prozess ist, ist es gleich, welche Lokalität das System zum Absturz bringt.

2.3 Sicherheitstechnische Aspekte

2.3.1 Sicherheitsmechanismen und Bedrohung

In Windows NT werden Ressourcen, wie z.B. Dateien, Sockets oder Ports, durch Objekte dargestellt. Objekte ermöglichen eine gemeinsame Nutzung von Ressourcen durch verschiedene Prozesse. Objekte sind im Gegensatz zu normalen Datenstrukturen, wie sie im Kernel-Modus verwendet werden, sicherungsfähig. D.h. Objekten können Zugriffskontrolllisten (access control list, ACL) zugewiesen werden und Objekte können daher auch auch überwacht werden.
Sicherheitszugriffskontrollen der Objekte geschehen im Kernel-Modus durch den Security Reference Monitor (SRM). Zudem ist es Aufgabe des SRM für Objekte die Privilegien (Benutzerrechte) zu modifizieren und Sicherheitsüberwachungsnachrichten generieren.
Verantwortlich für Systemsicherheitsstrategie ist das Local Security Authority Subsystem (Lsass), das im User-Modus läuft und die Authentifizierung von Benutzer übernimt und Sicherheitsüberwachungsnachrichten zum Ereignisprotokoll (Event Log)übermittelt. Die Systemsicherheitsstrategie kann sehr umfangreich sein und granulare Einstellungen für das System enthalten. Die wichtigsten sind z.B. welche Benutzer sich am System anmelden dürfen, welche Passwortrichtlinien beachtet werden müssen, die bewilligten Privilegien für Benutzer und Gruppen und Einstellungen für die Systemüberwachung. Der Lsass-Prozess beherbergt den assoziierten Dienst Lsasrv sowie viele weitere sicherheitsrelevante Dienste, die im folgenden aufgeführt werden. Der Security Accounts Manager (SAM) läuft als Dienst im Lsass-Prozess. SAM bietet Dienste für die Verwaltung von Benutzern und Gruppen auf dem lokalen System. Die Datenbanken für Lsass und SAM sind über die Registrierung erreichbar unter den Schlüsseln HKLM/SECURITY bzw. HKLM/SAM.
Der Verzeichnisdienst Active Directory läuft ebenfalls im Lsass-Prozess. Dieser Verzeichnisdienst speichert Informationen über Objekte in einer Domäne. Objekte können hierbei z.B. Benutzer, Gruppen oder Computer sein. Wichtig für die Anmeldung ans System sind die sogenanten Authentication Packages, die ebenfalls im Lsass-Prozess laufen. Die Authentication Packages implementieren die Authentifizierungsstrategie, indem sie die Überprüfung von Benutzernamen und dem zugehörigem Passwort übernehmen.
Die Verwaltung der interaktiven Anmeldung und die Antwort auf eine SAS-Anforderung (Secure Attention Sequence, B-Level-Anforderung nach einen vertrauenswürdigen Pfad) übernimt der Winlogon-Prozess, der im User-Modus läuft. Nach erfolgreicher Anmeldung am System erzeugt der Winlogon-Prozess den Shell-Prozess für den Benutzer. Die Anmeldung selbst wird durch die Graphical Identification and Authentication Bibliothek (GINA) realisiert, die innerhalb des Winlogon-Prozesses ausgeführt wird. GINA liefert Winlogon den Benutzernamen mit Passwort oder die PIN einer SmartCard. Für proprietäre Authentifizierungen kann die GINA-Bilbiothek gegen eigene Implementierungen ausgetauscht werden. Wenn bspw. biometrische Zugangsverfahren eingesetzt

werden. Remoteanmeldungen über das Netzwerk werden von dem Netlogon-Dienst entgegen genommen und als lokale Anmeldungen abgewickelt. Der Netlogon-Dienst wird ebenfalls im Lsass-Prozess ausgeführt.

2.3.2 Bewertungssystem

Windows NT 4.0 ist nach TCSEC C2-zertifiziert. TCSEC steht für ein Bewertungssystem des Department of Defense. Es sind Klassen von A bis D definiert, wobei A ein System mit einem überprüften Design ist und die höchste Sicherheitsstufe darstellt. Derzeit erfüllt kein BS die Kriterien eines A1-Systems. Systeme der Klasse D bieten minimale Sicherheit.
Die Schlüsselanforderungen für den C2-Level sind:

- Eine sichere Anmeldungseinrichtung, damit Benutzer eindeutig identifiziert werden können.
- Diskrete Zugriffskontrolllisten, damit Besitzer von Ressourcen festlegen können, wer auf die Ressourcen zugreifen darf und was damit gemacht werden darf.
- Vorhandensein einer Sicherheitsüberwachung zum Erkennen und Speichern von sicherheitsrelevanten Ereignissen, insbesondere der Erstellung, des Zugriffs und Löschung von Ressourcen.
- Schutz gegen Wiederbenutzung von zerstörten Objekten. Daten von deallokierten Dateisystembereiche dürfen nicht erneut ausgelesen werden dürfen.

Windows NT erfüllt auch zwei B-Level-Anforderungen:

- Sichere Pfade sind in Windows NT durch die Ctrl+Alt+Delete-Sequenze gegeben.
- Einrichtungen zur vertrauenswürdigen Verwaltung von Benutzer- und Gruppenkonten.

2.4 Bewertung der Sicherheitsfunktionen

Die Sicherheitsfunktionen von Windows NT sind geeignet, das BS und die assoziierten Ressourcen gegen Fremdzugriff zu schützen. Die Schwierigkeit besteht darin, das System entsprechend zu konfigurieren und, dieser Punkt ist mindestens genauso wichtig, das System entsprechend zu warten.
Häufige Fehlerursache ist die Ausführung von Anwendungen in einem zu vertrauenswürdigen Sicherheitskontext. D.h. Anwendungen werden im Sicherheitskontext des Administrators ausgeführt, obwohl viele Anwendungen nicht alle Privilegien erfordern, die ihnen bewilligt werden.
Das Problem wird oft durch Anwendungen verschärft, die aufgrund schlechter Programmierung mehr Rechte einfordern, als sie für die Ausführung der geforderten Operationen benötigen.
Diese Thematik betrifft auch Viren und Trojaner, die oft nur sehr geringen Schaden anrichten könnten, wenn Benutzer nicht als Administrator eingeloggt sind. Windows NT hat eine Funktion namens Windows-Dateischutz (Windows File Protection, WFP), die wichtige Systemdateien automatisch wiederherstellt, falls diese gelöscht oder mit einer älteren Version überschrieben worden sind. Diese Funktion richtet sich an Benutzer, die

noch unerfahren mit dem Windows NT-System sind und auf die Korrektheit von Installationsprogrammen hoffen. WFP ist allerdings nur eine unsaubere Lösung, da Benutzer nur Rechte erhalten sollten, die sie unbedingt benötigen.

2.5 Allgemeine Aspekte von DCOM

2.5.1 Funktionsweise

Die Architektur wurde als ein objektbasierter Binärstandard definiert. Die Kommunikation zwischen Client und COM-Objekt findet über das Interface statt. Der Client bekommt eine Referenz auf ein Interface des COM-Objektes. Es ist dabei unerheblich, ob das eigentliche Objekt lokal oder remote vorliegt. In COM/DCOM ist der eigentliche Zugriff transparent.

2.5.2 Art der Implementierung

Es werden ein Objektmodell definiert, sowie Namensraum-, Speicher und- Verarbeitungsmodelle beschrieben. Ziel der Implementierung ist die Wiederverwendung von (verteilten) Bauelementen auf Binärebene. Für uns bedeutet das, dass wir Bauelemente unabhängig von der Programmiersprache nutzen können. Die Klassen haben eine definierte Schnittstelle, welche die nutzbaren Dienste beschreibt. Die Implementation ist nach außen gekapselt. Die Kommunikation selbst erfolgt orts- und zugriffstransparent. DCOM basiert auf einer RPC-Implementierung. COM implementiert die Sicherheit durch die Nutzung von Microsofts SSPI (Security Support Provider Interface). DCOM verwendet zur Kommunikation über das Netzwerk DCE RPC, was so viel bedeutet wie "Distributed Computing Environment Remote Procedure Call".

2.5.3 Systeme zur Funktionserbringung?

DCOM wird bei allen aktuellen Betriebssystemen zur Verfügung gestellt und kann auch für Windows 95 und 98 von der MS COM Website herunter geladen werden. Es ist auch eine Portierung für Linux und Mac vorhanden. DCOM greift nicht direkt auf die NT Sicherheitsprotokolle zu. Der Grund dafür ist, dass COM auch für andere Plattformen zur Verfügung gestellt werden sollte (UNIX und Mac). Jedoch konnte es sich dort nicht, wie ursprünglich erwartet, etablieren. Zur Funktionserbringung sollten bestimmte Anforderungsfelder erfüllt werden. Die sind Interoperabilität, Skalierbarkeit, Sprachunabhängigkeit, Verteilungstransparenz und Robustheit gegenüber Weiterentwicklung.

2.5.4 Vorgesehener Einsatzzweck und Zweckmäßigkeit der Architektur

COM gab es schon eine Weile, als sich die Überlegung auftat, dass es möglich sein sollte, nun auch wie in Corba, verteilte Anwendungen zu nutzen. Konkret gesagt musste COM um den Zugriff auf das Netzwerk (per DCE RPC) erweitert werden.
COM hat sich eigentlich auf komponentenbasierte Softwareentwicklung in Windows-Umgebungen beschränkt, wird aber nun auch als Basistechnologie für verteilte Systeme anerkannt. MS Strategie lautet ¡COM überall¿ und daher sollte COM im Betriebssystem und im Internet genutzt werden. Alle modernen MS Technologien basieren auf COM, ActiveX, OLE und DNA.

2.5.5 Einsatz in der Praxis

Im Grunde genommen soll eine Applikation die Funktionaliät beliebiger binärer Komponenten in einer objektorientierten Art und Weise nutzen. Dies sollte weiterhin unabhängig von den verwendeten Implementationssprachen und von Aufenthaltsort der Akteure möglich sein.

2.5.6 Interaktion

Es kommunizieren immer ein Client und Server über das DCE RPC miteinander. Dabei greift der Client auf Komponenten des Servers zu. Das heißt, der Client möchte Komponenten des Servers nutzen und der Server möchte Komponenten bereitstellen. COM folgt dabei dem Broker-Architekturmuster. Jedoch erfolgt die Kommunikation nicht direkt sondern über Proxies. Dadurch bleiben alle Aktivitäten für den Client unsichtbar, die für die verteilte Kommunikation benötigt werden.

2.5.7 Nutzung und Kopplung von DCOM Systemen

Technologien wie ActiveX und DNA (Dynamic InterNet Applications) nutzen als tragende Säule COM/DCOM.
Das Security Service Provider Interface (SSPI) stellt ein gemeinsames API für den Zugriff für DCOM Anwendungen auf die somit isolierten Sicherheitsprotokolle dar.

2.6 Sicherheitstechnische Aspekte von DCOM

2.6.1 Vorgesehene Bedrohungen und Schutzmassnahmen

Es wurde vorgesehen einen Sicherheitsmechanismus zu schaffen, der vor nicht autorisierten Zugriff schützt. Man musste sicherstellen, dass unerlaubter Zugriff auf Informationen sowie die Modifikation von Informationen nicht ermöglicht wird. Auch der Missbrauch und der Diebstahl von Diensten (Denial of Service) wurden vorgesehen. Dies ist durch Ausführung von Code möglich. Um dies zu unterbinden müssen also ausreichende Sicherheitsmechanismen geschaffen werden.
Der Schutz in DCOM lässt sich in drei Hauptaspekte untergliedern. Authentifikation dient der sicheren Identifizierung eines Benutzers. Das Vortäuschen einer falschen Identität darf nicht möglich sein. Die Autorisierung ermittelt Attribute und Privilegien eines Benutzers. Token Mangement ermöglicht die Kontrolle über Credentials bei Start eines Serverprozesses und beim Aufruf einer Methode.
Wichtig ist weiterhin die Zugriffskontrolle. Diese untergliedert sich in
Ausführungserlaubnis (beschreibt wer einen Serverprozess starten darf) und Zugriffserlaubnis (beschreibt wer auf Objekte des gestarteten Servers zugreifen darf).
Weitere Sicherheitsaspekte

- Identität ist die Sicherheit eines Objektes selbst.
- Können Nachrichten von außen abgefangen werden durch andere Personen (Vertraulichkeit)
- Integrität ist die Modifikation von Information nur für autorisierte und berechtigte Benutzer.

- Benutzer sind für sicherheitsrelevante Aktionen verantwortlich (Vertraulichkeit)
- Verfügbarkeit besagt, dass ein System den berechtigten Benutzern den Zugang nicht verweigern darf.
- Es ist auch möglich die Datenübertragung über das Netzwerk mit http über SSL bzw. TCP/IP über PPTP zu sichern.

2.6.2 Gegenüberstellung: Sicherheitsmechanismen vs. Bedrohungen

Den Anforderungen an die Schutzmechanismen wird DCOM nur bedingt gerecht, denn gehäuft auftretende Sicherheitslücken in der DCOM/RPC Kommunikation sind auf Fehler in der Programmierung zurückzuführen. Buffer Overflows im Windows-Betriebssystem und die Ausführung beliebigen Codes im RPC-Interface offenbaren Schwächen im System (MS03 - 038). Eine weitere Sicherheitslücke (MS03-039) ermöglicht sogar einen Denial of Service Angriff, wiederum durch Ausführung beliebigen Codes. Diese Breite der Bedrohungen ist also noch nicht abschätzbar. Man kann also davon ausgehen, dass weitere Lücken vorhanden sind.

2.6.3 Schutzbedürftigkeiten des Systems

Da DCOM fast überall im Betriebssystem verwendet wird, gibt es eine große Angriffsfläche. Ein gestarteter Server kann sowohl nur über die Netzwerkverbindung angegriffen werden. DCOM ist sehr tief in das Betriebssystem integriert, wobei sich weitere Möglichkeiten für einen Angriff eröffnen. Es gilt also besonders die Zugriffe von Nutzern zu prüfen und nur authentifizierten Nutzern Zugriff zum System zu gestatten.

2.6.4 Gefährdungen in der Praxis

Das Vortäuschen (Impersonation) einer falschen Identität ist eine der größten Bedrohungen für ein verteiltes System. Gelingt dieser Zugriff, ist jegliche Bedrohung für den Angreifer leicht zu realisieren. Ein weiterer wichtiger Aspekt ist das abfangen und verändern von Nachrichten und das Sniffen von Passwörtern. Einige Parteien haben auch die Absicht, andere Rechner zu nutzen, um Code auszuführen, um Würmer im Internet zu verbreiten. Durch die Zusammenarbeit mit Software wie z.B. Outlook Express ist es möglich Daten Viren über die Email Adressen im Adressbuch zu verschicken.

2.6.5 Sind die vorhandenen Schutzmechanismen ausreichend?

Implementierung und Wirkmechanismen

Die vorhandenen Schutzmechanismen sind soweit ausreichend, jedoch verbergen sie einige Implementationslöcher. Ein verteiltes System sollte unbedingt in der Middleware sicher sein. Moderne Betriebssysteme haben eine Rechtevergabe und Authentifizierung. Ein einzelner unsicherer Knoten des Sicherheitskonzeptes hebelt das ganze System aus, so sollte auf alle Fälle der Server eines Security Services sicher sein.

Betriebsumfeld: Umgebungsgestaltung

Die Schutzmechanismen sollten leicht administrierbar sein und aus Sicht der Benutzer nur die wichtigsten Informationen aufzeigen. Die Schutzmechanismen von DCOM selbst

sind jedoch nicht ausreichend. Es sollten Firewalls verwendet werden um unzulässige Angriffe zu vermeiden. Es sollten nur Ports offen sein die auf dem System benötigt werden.

Neue Angriffsmöglichkeiten

Angriffe finden zumeist nur gegen Schwachstellen statt, d. h. es werden Sicherheitslücken werden von Hackern nicht direkt in den Sicherheitssystemen gesucht. Vielmehr werden Lücken ausgenutzt die durch vorhandene Patches und Updates nicht behoben wurden. So kommt es immer wieder vor das abgewandelte Virusvarianten zu neuen Attacken führen können. Eine Abwehr ist dahingehend nur selten schnell und effektiv möglich, da die Verbreitung zu schnell fortschreitet. Sicherungsmechanismen wie Firewalls bieten keine weiteren Angriffsmöglichkeiten für Angreifer.

Unbeachtete Bedrohungen

Der Zeit ist DCOM/RPC durch weltweite Angriffe in Verruf geraten. Von Seiten vieler Sicherheitsexperten wird geraten, den Dienst bei Nichtnutzung abzuschalten, da er ein großes Risiko darstellt. Sobald Bedrohungen bekannt werden, kann man davon ausgehen das in kürzester Zeit ein Patch zur Verfügung steht. Gerade Systeme wie MS Windows, welches ein Großteil der PC User nutzen, müssen gesichert werden.

2.7 Komplexe Sicherheitstechnische Aspekte von DCOM

2.7.1 Illegitime Vorgänge, die legitim sind

Einige Fehler, die auftreten, betreffen nicht direkt DCOM, sondern die DCOM-Schnittstelle in Verbindung mit dem RPC-Protokoll. Wenn nun Nachrichten mit ungültigen Informationen gesendet werden ist es möglich die DCOM Objektaktivierungsanforderung (z.B. auf Port 135) zu beeinflussen. Diese werden normalerweise von Clientcomputern an Server gesendet. Der Angreifer muss nun eine spezielle ungültige Nachricht an den offenen Port 135 des DCOM Servers senden. Dies erfordert keine weiteren Berechtigungen für den Angreifer. Das Problem dieser Sicherheitsanfälligkeit ist, dass der RPC-Dienst die Nachrichteneingaben nicht korrekt überprüft. Durch weitere Fehler ist es möglich Denial of Service Attacken zu starten und Code auf einem Rechner auszuführen.

2.7.2 Schutzmechanismen bei Interaktionen mit anderen Systemen

Alle im Abschnitt Sicherheitstechnische Aspekte genannten Schutzmechnismen gelten generell für die Interaktion mit anderen Systemen.

2.7.3 Bewertung der Schutzmechanismen

Es treten immer wieder neue Lücken Windows Betriebssystem auf, die nicht direkt auf DCOM zurückzuführen sind. Man kann sollte also das gesamte Konzept überarbeiten. Die große Menge an Diensten die mit DCOM zusammenarbeiten muss ebenfalls gesichert werden. Es ist also erkennbar das eine Systempflege dauerhaft nötig ist.

2.7.4 Wartung des Systems

Die Wartung des Systems ist sehr komfortabel. Bei COM ist es möglich die Funktionalität von Objekten zu erweitern, ohne dabei inkompatibel zu den bestehenden Anwendungen zu sein. Neuere Clients können schon die Erweiterte Funktionalität nutzen.

2.7.5 Kooperative Behandlung von illegitimen Vorgängen

Es werden Patches eingespielt um illegitime Vorgänge zu unterbinden. Wenn die Angriffe nicht zu großen Schaden anrichten, kann man das Versenden von Würmern eher als Hinweis auf eine Verbesserungsmöglichkeit des Systems verstehen.

2.8 Allgemeine Aspekte von .NET

2.8.1 Architektur

Das .NET Framework besteht, wie aus der folgenden Abbildung ersichtlich ist, aus verschiedenen Komponenten, wobei im Rahmen dieser Betrachtung die unteren drei Ebenen von besonderer Bedeutung sind: Äuf der untersten Ebene ist das Betriebssystem vorzufinden. Derzeit sind dies Windows 98, Windows NT 4.0, Windows Millennium Edition, Windows 2000, Windows XP Home Edition, Windows XP Professional, Windows Server 2003 family sowie Systeme, die das .NET Compact Framework enthalten (derzeit nur Windows CE 3.0 und höher) und andere Betriebssysteme, für eine CLR-Implementierung nach dem Common Language Infrastructure (CLI) Standard existiert (in Entwicklung für Linux das Projekt Go-Mono).

Auf der zweiten Ebene ist die Common Language Runtime zu finden, die MSIL-Code (Format der ausführbaren Programme, die auf dem .NET-Framework basieren.) übersetzt und dann ausführt. Äuf der dritten Ebene ist der Kern des .NET-Frameworks, die Base Class Library angeordnet. Die BCL ist eine Ansammlung tausender Klassen, die die Programmierung in sämtlichen Bereichen, wie Threading, Dateizugriff, Fenster, Remoting, Verschlüsselung, Mathematische Funktionen u.ä. unterstützt. (siehe Bild Architektur)

2.8.2 Vorgesehener Einsatzzweck und Zweckmäßigkeit der Architektur

Das .NET Framework wurde im Hinblick auf folgende Zielsetzungenentwickelt:

- Bereitstellung einer konsistenten, objektorientierten Programmierumgebung. Die Ausführung erfolgt lokal oder über verteilte Objekte auf Remotecomputern.
- Bereitstellung einer Codeausführungsumgebung.
- Bereitstellung einer Codeausführungsumgebung, die eine sichere Ausführung gewährleistet.
- Bereitstellung einer Codeausführungsumgebung ohne auftretende Leistungsprobleme.
- Schaffung einer konsistenten Entwicklungsumgebung für Windows- und webbasierte Anwendungen.
- Die Kommunikation wurde auf Industriestandards aufgebaut

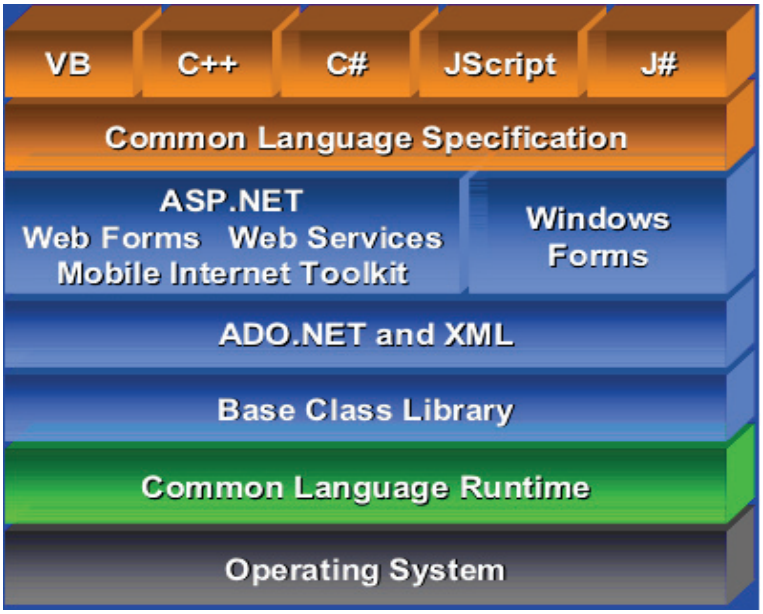


Abbildung 2.1: Architekur des .NET Frameworks

2.8.3 Einsatz in der Praxis

Der Einsatz in der Praxis ist breit gefächert. In der Programmierwelt hat sich .NET bereits jetzt mit großem Tempo verbreitet und bietet im Segment von Web- und Windowsanwendungen sehr gute Performance. Diese Eigenschaft macht .NET zu einer interessanten Zielplattform für alle Entwickler, die an der Wiederverwendbarkeit von Code und Sprachunabhängigkeit interessiert sind.

2.9 Sicherheitstechnische Aspekte von .NET

2.9.1 Vorgesehenen Bedrohungen und Schutzmassnahmen

Das .NET Framework bietet verschiedene Mechanismen, mit denen Ressourcen und Code gegen nicht autorisierten Code bzw. nicht autorisierte Benutzer geschützt werden können. Begonnen wird damit Berechtigungen auf Codeebene festzulegen. Dabei müssen nach Assembly- und Methoden-Berechtigungen unterschieden werden. Weiterhin kann der Administrator Sicherheitsrichtlinien festlegen, welche dann von der CLR bei der Ausführung des Codes beachtet werden. Für die Sicherheitsrichtlinien sind verschiedene Merkmale festgelegt. Diese unterteilen sich u.a. in Codesignaturen, Codeidentität und Lokalität des Codes. Vertrauenswürdigkeit wird allerdings nicht starr auf eine bestimmte Assembly (.NET-Terminologie für eine DLL, die ausführbaren Code enthält) angewandt, sondern es wird ein so genannter Stackwalk durchgeführt, wobei überprüft wird, ob alle Assemblies in der Aufrufhierarchie die nötigen Berechtigungen besitzen, die Operation durchzuführen. (siehe Bild Stackwalk)
Mithilfe des Stackwalks sollen Täuschungsmanöver verhindert werden, wobei nicht vertrauenswürdiger Code vertrauenswürdigen Code aufruft. Als Beispiel: eine .NET-Anwendung aus dem Internet verwendet Methoden der BCL um Dateizugriffe zu realisieren. Da die BCL lokal vorliegt, digital signiert und volle Zugriffsrechte auf das Dateisystem be-

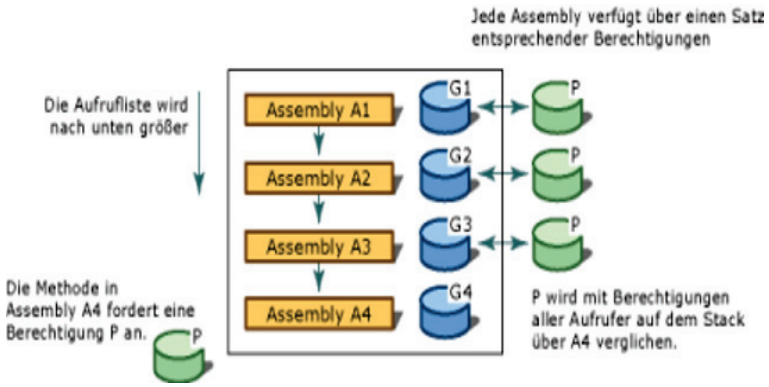


Abbildung 2.2: Stackwalk

sitzt, genießt die BCL eine sehr hohe Vertrauensstellung. Die aus dem Internet herunter geladene Anwendung darf jetzt allerdings keinerlei Schreibzugriffe auf das Dateisystem durchführen, obwohl die entsprechenden Methoden der BCL es dürften, der Stackwalk führt jetzt allerdings nach oben zur Internet-Anwendung, die per se keinerlei Vertrauen genießt.

Es gibt die Möglichkeit den Stackwalk abzuschalten, da der Stackwalk besonders bei Anwendungen, die häufige Methodenaufrufe verwenden, wie dies z.B. bei rekursiven Algorithmen der Fall ist, zu Lasten der Performance geht. Diese Vorgehensweise wird von Microsoft nicht empfohlen, da es hierbei möglich wäre, dass nicht-vertrauenswürdiger Code ausgeführt werden würde.

Code erhält die Berechtigungen, die ihm aufgrund der lokalen Sicherheitsrichtlinie zugestanden werden. Allerdings kann Code anfordern, geringere Berechtigungen zu bekommen, um sicherzustellen, dass die Funktionalität des Programms nicht in einer nicht gewollten Art und Weise ausgenutzt wird. Eine Sicherheitsrichtlinie ist ein konfigurierbarer Satz von Regeln, auf dessen Grundlage die Common Language Runtime Code Berechtigungen erteilt.
Es können drei Arten von Berechtigungen unterschieden werden.

1. Codezugriffsberechtigungen
    - Zugriff auf geschützte Ressourcen
    - Ausführen von geschützten Operationen
2. Identitätsberechtigungen
    - Code muss über bestimmte Anmeldeinformationen verfügen. Bspw. wenn eine Operation nur von einer bestimmten Person ausgeführt werden darf.
3. Berechtigungen der rollenbasierten Sicherheit
    - Es wird überprüft, ob ein Benutzer oder Agent Mitglied einer bestimmten Rolle ist.

2.9.2 Schutzbedürftigkeiten des Systems

Ein derzeitiges Problem von .NET-Anwendungen, das auch gleichzeitig viele der neuen Funktionen ermöglicht, ist das MSIL-Format der ausführbaren Anwendungen: es ist sehr leicht lesbar. Schon jetzt gibt es Programme, die aus kompilierten Assemblies wieder Quellcode erzeugen können. Dies war zwar bisher auch schon bei nativen Ausführbaren möglich, jedoch ist es nun durch die Menge an Metainformationen noch einfacher. Algorithmen, die auf dem Konzept Security by Obscurity aufsetzen, sollten daher weiterhin in native Maschinensprache übersetzt werden, um das Reverse Engineering nicht zu leicht zu machen.

2.9.3 Bewertung der Schutzmechanismen

Die neuen Schutzmechanismen des .NET-Frameworks sind eine seit langem notwendige Ergänzung der Sicherheitsfunktionen von Windows, da es erstmals möglich ist Berechtigungen auf einzelne Methoden innerhalb von Assemblies zu setzen. Das bisherige Prinzip des Alles oder Nichts ist damit durchbrochen. Das Framework ist noch zu jung, um wirklich Schwachstellen zu entdecken. Eine realistische Einschätzung des Einsatzes der neuen Sicherheitsfunktionen kann erst gegeben werden, wenn sich das .NET-Framework weiter etabliert hat und mindestens eine solche Verbreitung besitzt wie bspw. Java.

2.10 Zusammenfassung

Dieser Beitrag zeigt die Architektur von Windows NT, DCOM und dem .NET-Framework auf, wobei besonderes Augenwerk auf die sicherheitsrelevanten Aspekte der drei Systeme gelegt wird. Es wird sehr kurz auf die geschichtliche Entwicklung der Systeme eingegangen, die u.a. Einflüsse zu bestimmten Implementierung bei Sicherheitskonzepten aufzeigen. Es werden die Schlüsselkomponenten der Sicherheitskonzepte der Systeme vorgestellt und ausgewählte praktische Einsatzgebiete, Mechanismen und Gefährdungen gezeigt.

Literaturverzeichnis

[1] Randy Abernethy. *COM/DCOM Unleashed*. Macmillan Computer Publishing, 1999.

[2] Chris Anderson. .NET Framework Overview. *Microsoft TechEd 2002*, 2002.

[3] Don Box. *Essential COM*. Addison-Wesley, 1 edition, 1997.

[4] Don Box. *50 Ways to Improve Your COM and MTS-based Applications*. Addison Wesley, 1999.

[5] Richard Grimes. *Professional DCOM Programming*. Peer Information Inc., 2 edition, 1997.

[6] MSDN. Microsoft Developer Network. *Microsoft MSDN*, 2003.

[7] Markus Horstmann und Marty Kirtland. DCOM Architecture. *MSDN Libary*, 3, 1997.

[8] Solomon und Russinovich. *Inside Microsoft Windows 2000*. Microsoft Press, 3 edition, 2000.

3 Grundsätzliche Strukturen von Microsoft Windows-Software

V. Mende, M. Hübner

3.1 Einleitung

Sicherheit ist auch in Bezug auf Microsoft Windows^{TM} Software (Abk.: Windows Software) ein umfassendes und weitreichendes Thema. In den nächsten Kapiteln erhalten Sie einen groben Überblick über die Struktur von Windows Software und über verschiedene Sicherheitsaspekte.

3.2 Allgemeine Aspekte

Für die bessere Betrachtung der Software und der mit ihr involvierten Umgebung wird der Begriff System eingeführt. Das System besteht aus zwei Teilbereichen, dem Betriebssystem und der Software. Das Betriebssystem bildet die Grundlage für die Software, welche wiederum die Aspekte Entwicklung und Nutzung beinhaltet.

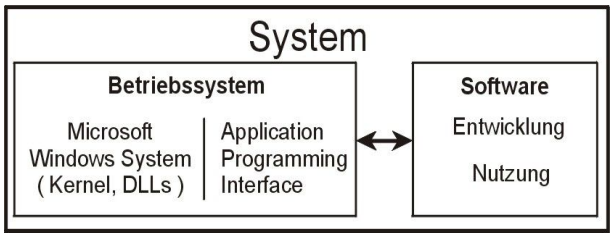


Abbildung 3.1: System

3.2.1 Übersicht

Zu Beginn soll ein Überblick über die Interaktion der Software mit dem Betriebssystem (Multiuser-Systeme der Windows Familie: Windows NT, 2000 und XP) gegeben werden. Zunächst wird die vereinfachte Struktur des Betriebssystems erläutert, mit der die Software in Form von Anwendungssystemen (Applikationen) interagiert.

3.2.2 Struktur des Betriebssystems

Entscheidend für die Arbeit mit Windows sind die Begriffe Kernel Mode und User Mode. In den meisten Multiuser-Systemen laufen Applikationen und Betriebssystem separat ab.

Das Betriebssystem wird in einem privilegierten Ausführungsmodus (sog. Kernel Mode) ausgeführt und hat Zugang zu den Systemdaten und der Hardware. Applikationen befinden sich in einem nicht privilegierten Ausführungsmodus (sog. User Mode) und verfügen über keinen Zugriff auf die Hardware und nur über eine begrenzten Zugriff auf Schnittstellen und Systemdaten.
Abbildung 2 veranschaulicht grob die Struktur des Betriebssystems. Alle Prozesse (auch die Applikationen) oberhalb der Linie befinden sich im User Mode. Die Komponenten unterhalb stellen die im Kernel Mode arbeitenden System Services dar. Die für unser Thema entscheidenden User Mode Teile sind die Environment Subsysteme, User Applikationen, die Subsystem DLLs und Teile der Service Prozesse.

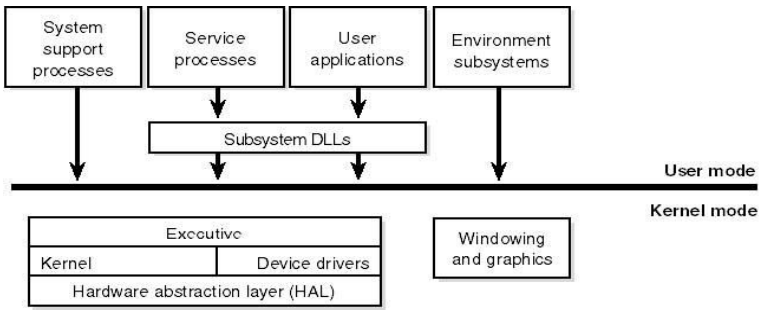


Abbildung 3.2: Struktur des Betriebssystems[6]

3.2.3 User Mode

Die *Service Prozesse* enthalten im wesentliche Prozesse mit denen Applikationen arbeiten. Unter den *User Applikationen* werden alle Applikationen zusammengefasst. Es ist Software, welche vom Benutzer des Rechners installiert wurde, als auch Software, die seitens des Betriebssystems bereits installiert wurde, darin enthalten. Alle User Applikationen arbeiten innerhalb eines *Environment Subsystems*, welche die nativen Betriebsystem Services (innerhalb des Kernel Mode) den User Applikationen in Form von Funktionen zur Verfügung stellen. Windows unterstützt drei Environment Subsysteme: Win32, POSIX, and OS/2. Diese Subsysteme sind die Schnittstellen für die Arbeit mit Betriebsystem. Zu welchen Subsystem eine Applikation gehört, wird durch den Entwickler festgelegt. Wir beschränken uns auf Software, die auf dem Win32 API basieren. Eng mit dem Subsystem verbunden sind die Subsystem DLLs, die sämtliche Win32 Funktionen für Applikationen bereitstellen.

Win32 API

Das Win32 API (Application Programming Interface) ist die primäre Programmierschnittstelle für die Windows Betriebssysteme. Die Hauptfunktionen umfassen das Arbeiten mit Prozessen, Threads, Speichermanagement, Sicherheit, I/O, Fenster und Grafik. Enthalten ist das Win32 API im Plattform SDK von Windows und in Entwicklungsumgebungen. Sie ist die bevorzugte, der unterstützten Programmierschnittstellen, da sie den größten Zugang zu den System Services im Kernel Mode bietet.
User Applikation rufen nicht direkt die nativen System Services auf, sondern sie benutzen dafür Funktionen aus den Subsystem DLLs, in dem sie diese dynamisch laden. Die

Aufgabe der Subsystem DLLs ist es, den Aufruf einer dokumentierten (API)-Subsystem-Funktion in einen Aufruf einer undokumentierten Betriebssystem-Funktion zu transformieren.

3.2.4 Kernel Mode

Die meisten Betriebssystemkomponenten und Gerätetreiber arbeiten in dem geschützten Kernel Mode Speicher und können somit direkt die Systemdaten bearbeiten. Die Kernel Mode Komponenten sind:
Executive enthält die grundlegenden Betriebssystem Services wie Speichermanagement, Prozess- und Threadmanagement, Sicherheit, I/O und Interprozesskommunkation (IPC).
Kernel besteht aus Low-Level System Funktionen wie das Thread Scheduling, Interrupts und Exception Dispatching.
Device drivers beinhalten sowohl Hardware Gerätetreiber, welche I/O Funktionen in spezifische Hardwaregeräte I/O Funktionen umsetzen, als auch Dateisystem- und Netzwerktreiber.
Der *Hardware Abstraction Layer* (HAL) ist die Schicht von Code, welche den Kernel, Gerätetreiber und den Rest der Windows Executives von der plattformspezifischen Hardware trennt.
Das *windowing* und *graphics system* implementiert die grafischen User Interface (GUI) Funktionen (für Entwickler die bekannten GDI Funktionen), wie das Arbeiten mit Fenstern, Steuerelementen und Zeichen.

3.2.5 Implementierung der Software

Im Weiteren wird Software im Allgemeinen beschrieben. Sie kann in verschiedene Kategorien, je nach Implementation und Zweck, unterteilt werden.

Bezeichnung			Beispiele
User Interface	grafisch	Konsole	Applikationen, Kommandozeilentools
Lokalität	lokal	verteilt	Online-Spiele
Ausführungsart	Maschinen-code	Interpreter-code	exe-Dateien, Java Applikationen
Betriebsmodi	Singleuser	Multiuser	Datenbanksysteme

Ein wichtiger Punkt ist die Ausführungsart. Software, welche als Maschinencode vorhanden ist, kann ohne zusätzlichen Aufwand ausgeführt werden. Sie nutzt die Möglichkeiten des Betriebssystems, indem sie notwendige Bibliotheken (Subssystem DLLs) lädt. Andere Software, welche als Interpretercode vorhanden ist, benötigt zur Funktionserbringung eine spezielle Laufzeitumgebung. Typische Vertreter sind Visual-Basic-Skripte in den Microsoft Office Paketen, als auch Java-Skripte im Browser und Java-Applikationen.

Einsatzzweck
Ausgehend davon, dass das Betriebssystem zweckmäßig installiert und konfiguriert ist (notwendige Treiber, Bibliotheken, Servicepacks und ähnliches sind vorhanden), wird nun auf die Software eingegangen.

1. Zweckmäßige Entwicklung der Software:

- Wurde Vorgaben des Betriebssystem beachtet:
  sauberes Eintragen und Löschen (z.B. aus der Windows Registry), Bibliotheken laden und wieder freigeben, Ressourcenverschwendung (CPU-Zeit), Berechtigungen, Prioritäten ...
- notwendige Behandlung und Beseitigung von Applikationsfehlern:
  Speicherung von Daten, Integritätsprüfungen(fehlerhafte Eingaben, Buffer Overrun), Stabilität, Sicherung und Wiederherstellen von Daten bei Abstürzen
- Unterstützung für den Nutzer:
  verständliche Oberfläche, Manuals, Hilfsdateien, Support, Updates
- Erfüllt die Software die Absicht des Entwicklers (Viren und Würmer sind auch Software)

2. Zweckmäßige Nutzung der Software:

   - Kompetenz und Fähigkeiten des Nutzer stehen der Komplexität der Software gegenüber:
     Konfigurieren und Anpassen der Software, Nutzen der vorhandenen Hilfe, Aktualisieren der Software (Updates), Unerfahrener Benutzer $\asymp$ Handhabung der Software nicht nachvollziehbar

Beim Umgang mit dem System treten die zwei Parteien Benutzer und Entwickler als direkte Nutzer des Systems auf. Indirekte Nutzung tritt dann auf, wenn Systeme sich untereinander nutzen oder Software mit andere Software innerhalb eines Systems arbeitet. Ein typisches Beispiel für die Arbeit von Software mit Software sind Excel Tabellen, die in Word-Dokumenten auftauchen. Beispiele für die indirekte Nutzung von Systeme findet man bei der Arbeit mit Netzwerken oder dem Internet.

3.3 Sicherheitstechnische Aspekte

In diesem Abschnitt geht es darum, den Einfluss von Sicherheitsaspekten bei Windows Applikationen zu zeigen. Es wird auf verschiedene Möglichkeiten der vom Betriebssystem bereitgestellten Funktionen eingegangen. Applikationseigene Entwicklungen (z.B. eigene Funktionen und Mechanismen von Datenbanksystemen) werden nicht berücksichtigt. Sicherheitsaspekte lassen sich in einen oder in mehrere der 3 folgenden Punkte einordnen:

- **Schutz sensitive Nutzerdaten:** Wie wird Sicherheit, d.h. Schutz von Daten realisiert und wie effektiv sind die Mechanismen in der Praxis. In diesen Bereich fällt z.B. die Interprozesskommunikation, d.h. der Austausch von Daten zwischen Applikationen.
- **Stabilität des Systems:** Das betrifft sowohl die Stabilität der Software (Programmfehler, Abstürze, Exceptions, ...), als auch die Stabilität des Betriebssystems, d.h. inwieweit können sich beide Teilsystem gegenseitig (voreinander) bzw. gemeinsam vor äußeren Einwirkungen schützen.
- **Manipulation:** Hierbei geht es um Software, die nicht ausspioniert oder beschädigt wird, sondern als Werkzeug gegen das System oder andere Software benutzt wird.

Das Betriebssystem Windows bietet viele Sicherheitsaspekte für Entwickler und deren Software. Darüber hinaus existieren weitere Tools, über die der Benutzer seine Sicherheitsbedürfnisse einstellen kann.

3.3.1 Schutzmechanismen

Windows unterstützt die C2-Level Sicherheit. Dies umfasst den Schutz aller gemeinsam zu nutzenden Systemobjekte (Dateien, Verzeichnisse, Prozesse, Threads usw.), Sicherheitsüberprüfungen für Benutzer und deren Aktionen, Passwort Authentikation beim Login und Vorkehrungen beim Zugriff auf nicht initialisierten Speicher.

Digitale Zertifikate (Digital Ids, public key Zertifikate)
Digitale Zertifikate sind eine Basis, die für Authentikation (Überprüfung der Identität), für Verschlüsselung oder für beides genutzt wird.

Beispiel für Authentikation
ActiveX Komponenten sind native Windows Programme, die wie andere Programme auf das Betriebssystem zugreifen, aber durch die Berechtigungen der Nutzers eingeschränkt sind. Wenn Benutzer mit sehr vielen Berechtigungen, ActiveX Komponenten aus dem Internet herunterladen, haben diese somit Zugriff auf Dateisystem, Registry und Hardware und können dann auch automatisch gestartet werden. Als Schutz für den Benutzer können Entwickler von solchen Komponenten eine digitale Id von Microsoft beantragen. Darüber hinaus kann der Benutzer bereits installiert Komponenten deaktivieren oder löschen. Komponenten mit Zertifikaten bedeuten jedoch nicht absolute Sicherheit. [5]

Encryption File System Security (EFS)
Das EFS basiert auf den Cryptographic Services von Windows (laufen im User Mode) und nutzt sowohl die Sicherheitsfeatures des Dateisystems, als auch Subsystem DLLs, welche mit dem Local Security Authority Subsystem (Lsass.exe) und den Cryptographic DLLs kommunizieren. Zur Verschlüsselung wird der bekannte RSA public key-basierende Encryption Algorithmus verwendet. Die Schlüssel sind Passwort geschützt. Dies soll den Schutz bei geklauten Dateien bzw. ganzen Laptops gewährleisten.
Applikationen können EFS nutzen, in dem sie die API-Funktionen *EncryptFile* und *DecryptFile* zum Ver- bzw. Entschlüsseln benutzen.[6]

Benutzer können auch unabhängig von der Software ihre Daten verschlüsseln, in dem sie im Explorer (oder anderen Dateibrowsern) auf die Eigenschaften der Dateien oder Ordner gehen und dort im erweiterten Modus die Verschlüsselung aktivieren. Das Windows Dateisystem NTFS erlaubt nicht die Verschlüsselung von Dateien die sich auf der Root-Partition, bzw. im Ordner Winnt befinden, da viele dieser Dateien während des Bootens des Rechners benötigt werden, EFS aber während des Bootens nicht aktiv ist. [6]

Objektschutz
Der Objekt Manager ist Teil der Executive Services im Kernel Mode. Seine Aufgabe in Bezug auf Sicherheit ist die Umsetzung der C2-Sicherheitsrichtlinien. Er liefert ein Grundgerüst zur Erzeugung, Löschung, Schutz, Aufzeichnung und anderer allgemeiner Verwaltungsaufgaben von Systemobjekten. Er ordnet jedem Objekt einen Objekt-Header zu, in dem die Prozesse vermerkt sind, die diese Objekt benutzen und ein Security Descriptor, der angibt, wer das Objekt benutzen darf und was damit getan werden darf.
Wenn Applikationen auf solche Objekte ihres Subsystems zugreifen, geschieht dies indirekt über Handles. Dieser erzwingt die Nutzung der Systemfunktionen des Objekt-Managers.
Der Objekt-Manager verwaltet zu dem Handle das interne Objekt, z.B. eine Datei, und

prüft die Gültigkeit des Zugriffs durch die Applikation. Da Applikation nie auf interne Objekte zugreifen (sollten), ist dadurch eine zentrale Zugriffskontrolle möglich, welche zur Stabilität des Betriebssystems beiträgt. Darüber hinaus existieren Win32 Sicherheitsfunktionen, die es Applikationen erlauben ihre eigene Objekte zu definieren und diese, wie Systemobjekte zu schützen. So existiert zur internen Funktion *SeAccessCheck* das User Mode Äquivalent *AccessCheck*.

Impersonation
Sogenannte Impersonation (Verkörperung) bedeutet, dass ein Thread ein anderes Sicherheitslevel hat, als der ihn umschließende Prozess. Bei der Sicherheitsüberprüfung wird das Sicherheitslevel des Threads geprüft, nicht das des Prozesses. Falls der Thread keine Impersonation besitzt, wird das Sicherheitslevel des ihn umgegebenen Prozesses geprüft. Wichtig ist, wenn ein Thread eines Prozesses Zugriff auf ein Objekt bekommt, können alle anderen Threads des Prozesses, unabhängig von deren Sicherheitslevel auch auf das Objekt zugreifen, da alle Threads ein und dieselbe Handle-Tabelle benutzen.
Impersonation wird von einigen Mechanismen genutzt. Wenn ein Server mit einem Client mittels Pipe kommuniziert, kann der Server die *ImpersonateNamedPipeClient* Win32 API Funktion benutzen um das Sicherheitslevel des Clients zu übernehmen. Für die Kommunikation über Dynamic Data Exchange (DDE) und RPC existieren ähnliche Aufrufe. Impersonation wird von Netzwerk- Sicherheitsprotokollen wie dem LAN Manager 2 oder Kerberos benutzt.

3.3.2 Verwendung in der Praxis

Im Weiteren werden Beispiele für die Nutzung oder Nichtnutzung der Schutzmechanismen vorgestellt.
Named Pipes und Mailslots
Named Pipes und Mailslots sind Speicherbereiche, die für die Kommunikation zwischen Prozessen benutzt werden. Named Pipes unterstützen eine bidirektionale, verbindungsorientierte Kommunikation. Im Gegensatz zu einer Named Pipe ist ein Mailslot ein verbindungsloser, unidirektionaler Datentransfer. Der Vorteil gegenüber Named Pipes ist die Fähigkeit Broadcast- Nachrichten zu unterstützen. Beide Techniken verwenden in den unteren Schichten Protokolle wie IPX, TCP/IP und NetBEUI.
Sicherheitseinstellungen für Pipes und Mailslots liegen beim Entwickler. Beim Erzeugen einer Pipe muss ein Öffnungmodus (OpenMode) angegeben werden, der den Zugriffsmodus, den Overlapped Mode, den Write-Through-Mode und den Sicherheits Mode spezifiziert. Sowohl Client als Server müssen diese Rechte beim Benutzen der Verbindung (OpenFile ...) bestätigen.
WinSock
Winsock ist die Microsoft Implementation der BSD (Berkeley Software Distribution) Sockets. Da Winsock in dem Win32 I/O integriert ist, benutzt es Dateihandles für die Sockets. Damit gelten für Sockets die gleichen Sicherheitseinstellungen, wie für Dateien.

Remote Procedure Call (RPC)
RPC bietet einer Applikation die Möglichkeit Funktionen lokal und auf entfernten Rechner auszuführen. Ab Windows 2000 enthält RPC die Integration der Security Support Provider (SSP), so dass Client und Server zur Kommunikation Authentikation oder Verschlüsselung benutzen können. Wenn ein RPC Server eine sichere Verbindung wünscht,

muss er seinen SSP-spezifischen Hauptnamen mit einem SSP registrieren. Ein Client registriert sich mit seinen Einstellungen, wenn er sich mit einem Server anhand dessen Namen verbindet. Während der Verbindung gibt der Client das Authentifikationslevel an, das verwendet werden soll. Da verschiedene Authentifikationslevel existieren, können nur authorisierte Clients sich mit einem Server verbinden. Zur Prüfung erfolgt ein Check der Integrität einer RPC Nachricht, um Manipulationen festzustellen. Ein SSP übernimmt die Details der Authentifikation und Verschlüsselung von Netzwerkkommunikation nicht nur für RPC, sondern auch für Winsock. Windows 2000 und XP enthält bereits einige Built-In SSPs, wie Kerberos SSP. Ein weiteres Feature der RPC-Sicherheit ist die Fähigkeit eines Servers die Sicherheitsidentität eines Clients zu verkörpern, d.h. er nimmt für die Arbeit mit diesem Client dessen Sicherheitsniveau an, und kehrt nach Beendingung der Arbeit mit dem Client in sein eigenes Sicherheitsniveau zurück.
Neben RPC, das ohne zusätzliche Installation verfügbar ist, existieren noch weiter Middleware-Systeme, wie CORBA und Java RMI, für verteilte Applikationen. Diese enthalten eigne Sicherheitsfunktionen auf die hier nicht weiter eingegangen werden soll.
Benutzer können zwar selten die Sicherheitsmechanismen der Applikation festlegen (SSL-Verschlüsselung), dafür aber Einstellungen für die Hardware vornehmen, z.B. WEP-Sicherheit für WLAN-Karten und die Installation von Ipv6 für die Arbeit mit dem Internet-Protokoll. Diese Einstellungen gelten dann auch für die damit arbeitenden Software. Für die Arbeit im Netzwerk bietet Windows den Benutzer u.a. das Kerberos-Protokoll für sichere Authentikation.

Nachrichtenschleife
Nicht direkt zur Interprozesskommunikation gehörend, soll auch die unterschätzte Nachrichtenschleife betrachtet werden. Jede Win32 Applikation hat im Hintergrund eine Nachrichtenschleife zu laufen, womit sie mit dem Betriebssystem oder anderer Software kommunizieren kann. Nachteilig ist, dass jede Applikation an jede andere laufende Applikation Nachrichten senden kann. Dabei spielt es keine Rolle, ob die Empfänger-Applikation solche Nachrichten empfangen möchte. Es existiert kein Mechanismus für die Überprüfung des Senders. So kann durch Senden der Nachricht EM_SETLIMITTEXT an ein Edit-Fenster die Anzahl von Bytes angegeben werden, die maximal eingefügt werden dürfen. Die Nachricht EM_SETTEXTLIMIT benutzt zur Übermittelung der neuen Längenangabe einen Datentyp (unsigned int) der Werte bis zu 2^{32} zulässt, so dass theoretisch über 4 Gigabyte in das Edit-Feld eingefügt werden könnten. Microsoft bewertet solche Szenarien nicht als gravierend und hat vorläufig keine Schutzmechanismen dahingehend entwickelt. Dies liegt auch daran, dass die Nachrichtenschleife ein zentraler Bestandteil der Win32 API ist und Veränderungen zu immensen Problemen bei bisherigen Applikationen führen würden. [1]

Interpretercode
Software in Form von Interpretercode kann sicherer sein, wenn die Laufzeitumgebung die notwendigen Einschränkungen macht. Ein typisches Beispiel sind Applets, die nur in der sog. Sandbox laufen und somit strikteren Regeln der Ausführung unterworfen sind, als dies bei andere Applikationen der Fall ist. Das Gegenbeispiel sind VisualBasic-Skripts, die zuviel Zugriffsmöglichkeiten auf das Betriebssystem haben, und daher als unsicher eingestuft werden.

Clipboard

Das Clipboard ist ein Speicher, der von Applikationen zum Ablegen und Auslesen von Dateninhalten benutzt wird. Falsche Sicherheitseinstellungen im Internet-Explorer ermöglichen JavaScripts den Inhalt des Clipboards auszulesen. Der Benutzer kann im Internet-Explorer über Internetoptionen Sicherheiten einstellen, dass nur Skripte auf das Clipboard zugreifen, wenn der Inhalt des Clipboard auch dem Internet-Explorer gehört.

3.3.3 Stabilität

Wie im ersten Kapitel erläutert, benutzt jede Applikation Services des Betriebssystem in Form von Schnittstellenaufrufen für ihr Subsystem. Diese Aufrufe laufen über die Subsystem DLLs im User Mode wie der Kernel32.dll zu den DLLs im Kernel Mode (Ntdll.dll). Sollte auch nur ein Teil dieser Ablaufkette fehlen, kann keine Applikation arbeiten. Mögliche Fehler:

- Fehlende Subsystem DLLs
  die benötigten Subsystem DLLs liegen nicht im erwarteten Ordner oder wurden gelöscht. Hier kann der Software Vertreiber die benötigten Subsystem DLLs mitliefern, was aber aufgrund der von Betriebssystem zu Betriebssystem verschiedenen Ausprägungen zu einer große Menge von zusätzlicher Software führt.

- Fehlende Kernel Services
  Wenn Services aus dem Kernel fehlen, besteht eine eingeschränkte Funktionsfähigkeit, so dass auch Applikationen hierfür keinerlei Schutzmechanismen bieten.

- Fehlende Bibliotheken
  Dabei handelt es sich ebenfalls um DLLs, die zu den Anwendungssystemen gehören. So nutzen viele Softwareentwickler gemeinsame Ordner, welche Bibliotheken für mehrere Applikationen enthalten. Typische Beispiele sind Firmen wie Adobe, Symantec und Microsoft. Problematisch wird das Verändern bzw. Löschen dieser Dateien, da andere momentan nicht aktive Applikationen diese nutzen.

3.4 Zusammenfassung

Software bietet meist wenige Sicherheitsfeatures, da sie nur solche benötigt, die für ihre Arbeit entscheidend sind. Grund dafür ist zum einen der hohe Aufwand für den Nutzer, da dieser die Sicherheitsmerkmale verstehen und berücksichtigen muss (z.B. Verschlüsselung von Dateien), als auch der Aufwand seitens der Entwickler. Nicht jede Software benötigt komplette Sicherheitsausstattungen, die alle Sicherheitsfeatures des jeweiligen Betriebsystems nutzen. Vielmehr wird davon ausgegangen, dass diejenigen, die ihre Daten schützen, spezielle Tools (VirenScanner, Firewalls etc.) verwenden, um so globalen Schutz zu erhalten. Ein Problem ist die Schnittstelle Win32 bzw. das Win32 Subsystem. Dortige Schwächen können nur schwer bekämpft werden, da meist eine Veränderung der darunterliegenden Kernel Services notwendig ist, was aber zu Inkompatibilitäten mit älteren Versionen führt. Ein weiteres nicht lösbares Problem sind die Benutzer. Die Einteilung nach Fähigkeiten ist nur dann wirksam, wenn die Person, die sich als Administratoren authentifizieren, auch die administrativen Fähigkeiten besitzen. Der Idealfall, dass ein erfahrener Administrator die Software installiert und konfiguriert und der Nutzer nur auf seinen Daten arbeitet, ist selten der Fall und sollte von Entwicklern nicht vorausgesetzt werden. Ebenso ist abzuwägen, wie viel Sicherheit

und wie viel Performanz benötigt wird. Zuviel Sicherheit kann das Arbeiten mit der Software behindern bzw. unmöglich machen.

Literaturverzeichnis

[1] Attacken über die Windows Nachrichtenschleife. http://security.tombom.co.uk/shatter.html.

[2] Betriebssysteme 3, Architektur von Windows XP. http://wwwbs.informatik.htw-dresden.de/svortrag/ai99/Kolbe/betriebssysteme_3.htm.

[3] Der Windows NT Objekt Manager, Teil 1. http://wwwbs.informatik.htw-dresden.de/svortrag/ai95/Kurbjuhn/ObjektManager.html.

[4] Passwortmanagement mit Kerberos, Authentisierung im Netzwerk. http://www.networkcomputing.de/heft/solutions/sl-2002/sl_0702_42.htm.

[5] Ed Bott, Carl Siechert. *Microsoft Security Inside Out for Windows XP and Windows 2000*. Microsoft Press, 2003.

[6] David A. Solomon, Mark E. Russinovich. *Inside Microsoft 2000*. Microsoft Press, 3 edition, 2000.

4 Architektur der unterschiedlichen UNIX-Betriebssysteme

M. Dräger, J. Schlössin

Vorwort

Das folgende Dokument ist auch online unter

⟨`http://burns.cs.uni-potsdam.de/~mdraeger/`⟩

bzw.

⟨`http://burns.cs.uni-potsdam.de/~jschloes/`⟩

als LaTeX- bzw. PDF-Version erhältlich.

4.1 Einleitung, Geschichte

Es war einmal vor langer langer Zeit (1965), da gab es einen Zusammenschluss aus dem MIT, General Electric, Bell Labs, Honeywell und IBM die das Projekt "Multics" in Angriff nahmen. Multics ist ein Betriebssystem, dessen Kriterien im Rahmen der *Fall Joint Computer Conference* zusammengestellt wurden. Kurz danach stellte sich heraus, dass "Multics" viel zu komplex geplant war, um es auf der Hardware der damaligen Zeit zu realisieren. 1969 zog sich Bell Labs aus dem Projekt zurück.
Ein Team bei Bell Labs interessierte sich aber für einige Ideen aus Multics, denn sie brauchten persönlich ein Mehrbenutzersystem um gleichzeitig an Dateien arbeiten zu können (was damals keineswegs selbstverständlich war). Die erste lauffähige Version hieß UNICS und lief auf einer PDP-7 von DEC[1] und hatte bereits ein primitives Prozessmanagement, ein selbstkonzipiertes Dateisystem sowie einige kleine Tools, um die Entwicklung direkt auf der PDP-7 weiterzuführen. Brain W. Kernighan, Projektmitarbeiter, nannte das Betriebssystem ironischerweise UNICS[2], da es im Gegensatz zur ursprünglichen Multics Planung sehr beschränkt war.
1970 wurde bei Bell Labs eine PDP-11 angeschafft und Unix darauf portiert. Bereits 1971 lief darauf im Patentbüro der Bell Labs ein Textverarbeitungsprogramm. Unix unterstützte zu dieser Zeit lediglich 2 Benutzer, was bei 16kb Systemspeicher, 8kb Programmspeicher und einer 512kb Festplatte nicht verwunderlich war.
Bei Bell Labs wurden bereits die Programmiersprachen A und B entwickelt. 1972 entwickelten Dennis Ritchie und Brain W. Kernighan die Sprache B weiter. Sie fügten der

[1] Digital Equipment Corporation
[2] UNiplexed Information & Computing Service

Sprache Datentypen hinzu und nannten sie ab sofort newB[3].
Unix wurde 1973 in C komplett neu geschrieben[4] und erhielt den Namen *Unix V4*. AT&T, die Muttergesellschaft von Bell Labs durfte aus kartellrechtlichen Gründen den Computermarkt nicht erschließen und verteilte deshalb Unix zum Selbstkostenpreis an Universitäten. Besonderen Anklang fand es in Berkeley wo unter Ken Thompson ab 1976 Neuerungen wie z.B. ein anderes Dateisystem, TCP/IP, eine Socket-Schnittstelle, eine virtuelle Speicherverwaltung und einige neue Tools beigesteuert wurden. Dieses UNIX-Derivat wurde dann als BSD[5] veröffentlicht.
Die Parallelentwicklung zwischen BSD und System V begann 1979, als sich AT&T doch entschloss Unix in größerem Umfang zu vermarkten.
1979 erwarb Microsoft[6] eine Unixlizenz und arbeitete an dem Derivat Xenix[7]. Microsoft stellte dann aber die Entwicklung zu gunsten von DOS ein und übergab später seine Unixlizenz an SCO[8].
AT&T betrat 1983 offiziell den Computermarkt und vermarktete ein auf *Unix V7* basierendes System, genannt *System V*[9]. Zeitgleich brachte die Universität Berkeley die Version 4.2 ihres BSD heraus.
Das POSIX[10]-Standardisierungsprojekt stellte sich die Aufgabe für die verschiedenen UNIX-Derivate eine einheitliche Schnittstelle zu definieren[11].
1983 begann Richard Stallman mit dem UNIX-ähnlichen Betriebssystem *GNU*[12] weil er über die Kommerzialisierung verärgert war. 1985 begründete er mit der Veröffentlichung des GNU-Manifests die freie Softwarebewegung. Andrew S. Tanenbaum entwickelte 1987 auch ein UNIX-ähnliches Betriebssystem namens *Minix* welches der Lehre dienen sollte. Dies erreichte zwar keine grosse Bedeutung, inspirierte aber Linus Torvalds.
Linus Torvalds stellte am 05.10.1991 seinen ersten Linux[13]-Kernel in der Version 0.02 vor.
1992 gründete die Universität Berkeley die Firma BSDi[14] und begann mit der Vermarktung von BSD. Daraufhin wurden 1993 *NetBSD* und *FreeBSD* gegründet. 1995 entwickelte sich *OpenBSD* aus *NetBSD*.
In den folgenden Jahren schritt die Entwicklung bei BSD und System V parallel voran, wobei die Unterschiede immer mehr in der Werbung als im eigentlichen System zu finden waren und sind.

[3] newB wurde dann einfach in C umbenannt
[4] Unix war bis dahin lediglich in Assembler verfügbar
[5] Berkeley Software Distribution
[6] Microsoft wurde 1975 durch Bill Gates und Paul Allen gegründet. 1980 hatten sie bereits 40 Angestellte.
[7] Xenix war eine Portierung für Intel 8086, Motorola 68000 und Zilog Z8000 Prozessoren.
[8] SCO wurde 1979 von Doug und Larry Michels gegründet und durch Microsoft mitfinanziert. Xenix stellte das erste SCO-eigene Unix dar.
[9] System V ist eigentlich ein spezielles Release von AT&T bezeichnet aber inzwischen eine ganze Klasse von UNIX-Derivaten.
[10] Portable Operating System Interface for UniX
[11] 1988 wurde POSIX.1 vergestellt, 1990 kamen die IEEE1003.1- und 1992 IEEE1003.2-Standards hinzu. Diese sind inzwischen auch DIN EN ISO genormt.
[12] Gnu is Not Unix
[13] Linux ist die Abkürzung für *Linus UNIX*.
[14] BSD Incorporation

4.2 Zielgruppe und Anwendungsgebiet

BSD und System V wurden vorwiegend für Mehrbenutzersysteme geschaffen. Historisch bedingt werden sie deshalb hauptsächlich auf Servern und Großrechnern eingesetzt. Viele große Firmen haben Ihre eigenen UNIX-Derivate entwickelt und sind somit Entwickler als auch Endanwender. Da Firmen die ihr UNIX entwickeln auch das Know-How dazu haben, ist der Schwerpunkt der Entwicklung eben nicht die intuitive Anwendung gewesen. Noch heute sind viele IT-Experten der Ansicht, man brauche für UNIX-Betriebssysteme einen höheren Schulungsaufwand als für Microsoft-Produkte. Distributoren von Linux haben viel Aufwand in Installationsroutinen, Paketverwaltung und Frontends gesteckt, um es auch im Desktopbereich für Privatanwender vermarkten zu können.

4.2.1 Konflikte bei der Nutzung (admin, user)

UNIX-System Nutzer arbeiten normalerweise nicht mit den Rechten eines Systemverwalters. Dadurch ist das System zu guten Teilen nicht kompromittierbar. Anwender sind in der Regel trotzdem in der Lage Software (unter ihren Home-Verzeichnissen) zu installieren. Dies ist ein klarer Vorteil gegenüber Windows-Betriebssystemen, wo Privatanwender aber auch Firmen meist mit Administrator-Rechten arbeiten, weil alles andere auf Dauer kaum praktikabel ist.
In großen Produktionsumgebungen gibt es aber viele Nutzerwünsche. Der Administrator steht nun in dem Konflikt für die Anwender eine komfortable Nutzung zu ermöglichen und trotzdem der Sicherheitspolitik der Firma gerecht zu werden. Besondere Beachtung ist der Verwaltung von Accounts über Angestelltenfluktuationen und Systemgrenzen hinweg zu schenken. Dazu muss auch der Informationsfluss von der Personalabteilung zum Systemverwalter genau bestimmt und eingehalten werden.

4.2.2 Wie kann man Administratoren von bestimmten Daten fernhalten

Unter UNIX haben Administratoren sehr viele Rechte. Dies ist nicht immer vorteilhaft. Einerseits kann ein (zum Beispiel übermüdeter) Admin großen Schaden anrichten. Andererseits ist ein Angreifer, der Administratonsrechte bekommen hat nur schwer aufzuhalten[15].
Diese Nachteile lassen sich mit Konzepten wie LIDS[16] abbauen. Dieser Patch enthält eine W^X Funktionalität (siehe Kapitel 4.3.3 auf Seite 37) aber auch die Möglichkeit Zugriffsrechte viel feingranularer einzustellen. Dazu zählen Ladeschutz für Kernelmodule, erweiterte Dateisystemschutzmechanismen auch für root[17], Zugriffsrestriktionen für einzelne Prozesse, zusätzliche Netzwerksicherheit und eine Funktionalität die unter *intrusion detector* zusammengefasst werden.
Hiermit kann man zum Beispiel verhindern, dass ein Angreifer seine Spuren verwischt, indem er die Logdateien modifiziert. Ebenso kann man das Ändern von Konfigurationsdateien unterbinden. Das System ist zur Zeit nur als Patch für Linux verfügbar und aus diesem Grund gibt es bisher keine einfache Unterstüzung in Distributionen oder anderen Derivaten. Der Ansatz ist aber sehr vielversprechend und wird möglicherweise die Sicherheit nachhaltig beeinflussen.

[15] Es gibt insbesondere keine Beschränkung für Gerätezugriff oder Dateisystemobjekte.
[16] Linux Intrusion Detection System
[17] So nennt man den Administrator in UNIX

4.3 Strukturen

4.3.1 Kernelstrukturen, Rechteverwaltung von Modulen

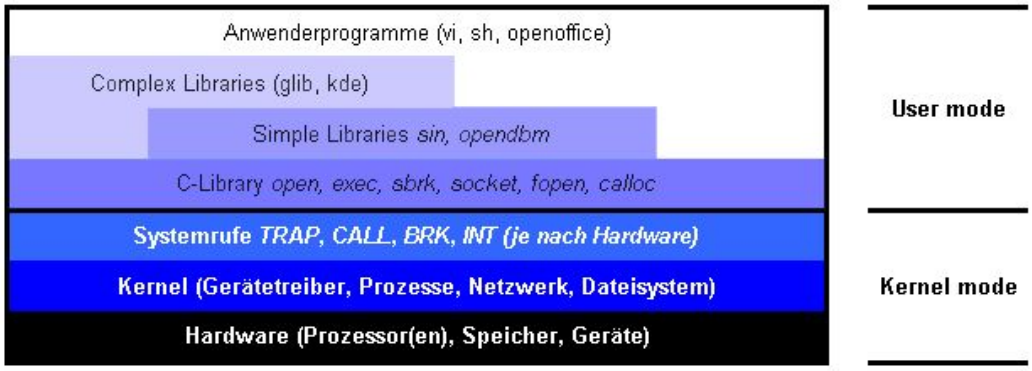
Der *Kernel*[18] eines Betriebssystems verwaltet elementare Prozess- und Datenorganisationen. Auf diesem baut die restliche Software des Systems auf. Ein *monolithischer Kernel* lässt sich in Kernbasis und Kernmodule unterteilen. Die Kernbasis wird von einem Bootloader beim Systemstart in den Arbeitsspeicher geladen und ist dann unveränderlich[19]. Kernmodule enthalten Funktionalität. Sie können zur Laufzeit geladen und auch Teilweise wieder aus dem Arbeitsspeicher entfernt werden. Dies kann für spezielle Gerätetreiber sinnvoll sein, wenn sie zum Beispiel nur selten verwendet werden. Wenn diese Funktionalitäten niemals benötigt werden, belegen diese keinen Speicher. Ist ein Modul geladen, so ist es Teil des Kernels und kann im Allgemeinen nicht mehr restriktiv überwacht werden. Ein fehlerhaftes oder korrumpiertes Modul kann also das gesamte System in mitleidenschaft ziehen[20].

Demgegenüber stehen *Microkernel*, welche wesentlich weniger Funktionen enthalten[21]. Nicht von Grund auf verfügbare Funktionen werden hier durch Programme angeboten oder müssen vom Administrator nachgeladen werden. Diese Zusätze[22] sind explizit nicht Teil des Kernels und können daher restriktiver überwacht werden. Dazu können auch Emulatoren ganz anderer Betriebssysteme gehören. Durch die geringere Komplexität des Kernels, ist er für den Entwickler leichter zu warten. Sicherheitspolitiken sind im Allgemeinen nicht Teil eines Microkernels.

4.3.2 Usermode vs Kernelmode

Verhält sich ein Programm destruktiv, muss das übrige System davor geschützt werden. Aus diesem Grund werden Prozesse im *user mode*[23] ausgeführt. Diese Prozesse sind nicht berechtigt auf Hardwareressourcen direkt zuzugreifen, sondern müssen dazu das API des Betriebssystem nutzen. Durch Prozessraumtrennung können sich diese Prozesse im Speicher nicht gegenseitig beeinflussen. Die Einhaltung der Grenzen wird durch die Hardware[24] überwacht und bei Überschreiten wird ein Interrupt ausgelöst. UNIX bricht den Prozess dann in der Regel ab.
Previlegierte Prozesse, wie kerneleigene müssen natürlich auf Geräte und andere Adressbereiche zugreifen. Sie laufen dazu im sogenannten *kernel mode*. *Rootkits* versuchen in genau diesen Modus zu gelangen. Dann können sie das Systemverhalten so beinflussen, dass ihr Vorhandensein nur schwer bemerkt wird.

[18] oder auch einfach Kern
[19] Diese Aussage bezieht sich auf das normale Verhalten. Ein Angreifer könnte auch die Kernbasis modifizieren.
[20] Sicherheitsprobleme können auch schon durch den Lademechanismus selbst hevorgerufen werden, wie das ptrace-exploit von Alan Cox im März 2003 für Linux demonstrierte.
[21] zum Beispiel nur Funktionen zur Speicherverwaltung und Interprozesskommunikation
[22] zum Beispiel Dateisystem- oder Benutzerverwaltung
[23] auch *user land* genannt
[24] Da die Peripheriegeräte auch in den Speicher eingeblendet werden, kann die gesamte Zugriffsüberwachung eines Prozesses relativ leicht durch die Memory Management Unit übernommen werden.



4.3.3 BSD: bufferoverflow, W^X (write xor execute)

Eine der am häufigsten genutzten Lücken in IT-Systemen sind sogenannte Bufferoverflows. Diese können bei Eingabedaten auftreten, deren Länge zur Entwicklungszeit unbekannt ist. Mit einer ungewöhnlich großen Datenmenge kann man den reservierten Platz überlaufen lassen und dadurch, im nachfolgenden Stack, beliebigen Code plazieren. Nun braucht der Angreifer nur noch die ebenfalls im Stack befindliche Rücksprungadresse der aktuellen Routine so zu überschreiben, dass der Prozess den eingeschleusten Code anspringt[25].
Ein allgemeiner Gegenansatz ist zur Zeit, im Stack zusätzliche zum Teil zufällige Daten abzulegen um die Veränderung zu erkennen. Etwas weiter geht OpenBSD[26]. In vielen Architekturen kann man Speicherseiten zum Lesen und Ausführen getrennt markieren. Der Stack enthält normalerweise keinen auszuführenden Code und wird nur als lesbar markiert. Der eingeschleuste Code ist dann nicht ausführbar und viele Ansätze für Bufferoverflows scheitern.
Leider funktioniert dies nicht auf allen Architekturen und es gibt auch Programme, die selbst modifizierenden Code verwenden[27].

4.4 Opensource vs Closedsource

4.4.1 Unterschied zwischen Distributionen & Derivaten

Die Unterschiede der einzelnen UNIX-Zweige System V und BSD sind kaum noch erkennbar. Das UNIX-Derivat OpenBSD gilt als sehr sicher[28], von dem Derivat NetBSD sagt man, dass es auf vielen Plattformen läuft und von FreeBSD ist man der Meinung, es ließe sich einfach installieren.
Die Unterschiede zwischen den einzelnen Linux-Distributionen liegen hauptsächlich in der Paketverwaltung. Da alle Distributionen auf dem mehr oder weniger gleichen Kernelgerüst basieren, werden keine großartigen Vor- bzw. Nachteile auf Kernelebene sichtbar.

[25] Dieses Szenario lässt sich fast beliebig variiren und funktioniert daher in sehr vielen Programmen.
[26] und inzwischen auch andere Distributionen
[27] Ein populäres Beispiel ist XFree86.
[28] Nach Aussagen der Entwickler gab es in 7 Jahren nur ein Remote-Einbruch bei einer Standardinstallation

4.4.2 weitere Probleme: Quellen auf WEB/FTP/RSYNC-Servern

Der Quelltext freier Software ist auf Internetservern verfügbar, dass jeder sie herunterladen kann. Dies stellt insofern ein Risiko dar, als dass diese Server leichter angegriffen werden können[29] als proprietäre Entwicklungslabore[30]. Ausserdem sind die zum Herunterladen verwendeten Protokolle oft unverschlüsselt und erlauben es Angreifern eigenen Code in den Downstream einzuschleusen. Dieses Risiko lässt sich mildern, wenn man mitgelieferte Prüfsummen mit verschiedenen Mirrors des Servers und des erhaltenen Codes vergleicht.

4.4.3 Hoffnung: offene Quellen

Die Quelltexte einzelner Programme sind offen und können somit von jedem eingesehen werden. Durch die Community werden Fehler oder Sicherheitslücken schnell bekannt und landen bei den zuständigen Entwicklern oder Paket-Betreuern einzelner Distributionen[31]. Open Source Software ist beim gewissenhaften Einsatz generell weniger anfällig für Makroviren, Dialer oder sonstige Schädlinge heutiger Software.

4.4.4 Rootkit, Viren, Fehlkonfiguration, Datenintegrität

Durch die strikte Trennung zwischen Benutzer- und Administratorrechten (siehe Kapitel 4.2.1 auf Seite 35) haben es Viren unter UNIX-Systemen recht schwer. Sollte sich tatsächlich mal ein Benutzer einen Virus einfangen, so hat dieser nur die Rechte des Benutzers[32]. Die Verbreitung von Viren für UNIX ist auch wegen dem heterogenen Umfeld schwierig. Hier finden sich oft andere Architekturen und untertützte Binärformate mit denen ein einfacher Virus nicht kompatibel ist. Durch die Architekturvielfalt können sich Vieren schwer epedemieartig ausbreiten. Das ist Ähnlich der größeren Resistenz von Mischkulturen gegenüber Monokulturen in der Natur. Viel Problematischer sind unter UNIX-Systemen die sogenannten Rootkits. Hierbei handelt es sich um Programme bzw. Programmfragmente die eine Person, bei einem erfolgreichen Angriff, auf dem System installiert. Sie dienen hauptsächlich dazu, dem Angreifer eine Hintertür offenzuhalten. Ist ein Rechner erst einmal kompromittiert, ist es sehr schwer und zeitaufwendig die Rootkits zu entdecken und zu entfernen[33].
Ein weiteres großes Problem können Fehlkonfigurationen darstellen. Mittlerweile existieren unter UNIX-Systemen viele Werkzeuge[34] zur Systemverwaltung, die unter anderem dazu dienen Fehlkonfigurationen zu vermeiden.

4.4.5 allgemeine Exploits - Angriffe von Innen vom User aus

Viele Angriffe auf UNIX-Systeme finden von innen, durch einen User statt. In der Regel wird versucht, z.B. durch abfangen des Passworts über unverschlüsselte Kanäle, Tastaturlogger oder “gelbe Zettel” am Bildschirm einen Benutzeraccount einzunehmen. Hat

[29] Ein Beispiel ist der Einbruch bei Debian im November 2003. Hier allerdings von einem User-Account aus über ein doBreak-Exploit
[30] Diese können durch Mandatory Access Control Mechanismen geschützt werden, was bei freier Software naturgemäß nicht geht.
[31] Bei JAP konnte zum Beispiel spy-code ohne Zutun der eigentlichen Betreiber (leicht) entdeckt werden.
[32] Wenn der Benutzer root heißt, wäre das natürlich schlecht.
[33] In den vielen Fällen hilft hier nur eine Neuinstallation.
[34] Meist handelt es sich hierbei um Shellscripts.

ein versierter Angreifer erst einen normalen Benutzeraccount eingenommen stehen ihm mehr Möglichkeiten als zuvor zur Verfügung um sich root-Rechte zu verschaffen[35].

4.4.6 Admins vertrauenswürdig?

Es gibt einige Projekte, womit man in der Lage ist, die Verantwortung für die Administration eines UNIX-Systems zu verteilen (siehe hierzu Kapitel 4.2.2 auf Seite 35).

4.4.7 Interaktion mit Software

Mittlerweile ist es auch für angehende Administratoren möglich ein einigermaßen ordentliches System zum Laufen zu bringen. In vielen Fällen ist noch nicht einmal großes Hintergrundwissen für UNIX-Systeme nötig, da viele Teile (mit den entsprechenden Tools) per Mouse einstellbar sind[36].

4.4.8 Konfigurationstools zur praktikablereren Sicherung der Systeme

Auf UNIX-Systemen existieren einige Tools zur Sicherung des Systems. Angefangen mit der Konsistenzhaltung der Benutzer- und Gruppenkonfiguration oder dem Zusammenspiel zwischen Serverdiensten und der Firewallkonfiguration eines Systems bis hin zur systemübergreifenden Verwaltung zum Beispiel über LDAP[37].

4.4.9 Restriktionen von Anwendungssoftware

In UNIX-Systemen sollten die meisten Anwendungen unter normalen Benutzeraccounts laufen. Um einzelnen Anwendungen erweiterte Rechte zu geben existieren verschiedene Einstellmöglichkeiten[38]. Man kann besimmten Nutzern gestatten bestimmte Programme mit den Rechten eines privilegierten Nutzers laufen zu lassen. Besonders ist darauf zu achten, dass die Programme dann nur von Privilegierten Nutzern verändert werden können und diese deren Funktionsumfang genau kennen.

4.4.10 Dateisystemsicherheit (Vor- und Nachteile von ACL's)

Die traditionellen Zugriffsrechte in UNIX-Systemen reichen den meisten Anwendern nicht mehr aus. Abhilfe versprechen die in mittlerweile fast jeder UNIX- und Linux-Variante enthaltenen ACL's[39].
Mit ACL's ist man in der Lage Rechte feingranularer zu definieren als in dem klassischem Dateisystem. Bisher konnte man lediglich die Rechte zum Lesen, Schreiben und Ausführen für die Bereiche User, Group und Other festlegen. Mit ACL's ist es nun möglich, beliebig vielen Gruppen oder Einzelnutzern Rechte an einem Dateisystemobjekt[40] einzuräumen oder wegzunehmen. Eine ACL besteht aus mehreren ACE's[41]. Eine ACE ist ein einzelner Eintrag einer ACL. Sie beinhaltet die Rechte eines bestimmten Nutzers oder einer bestimmten Gruppe am aktuellen Dateisystemobjekt.
Für den normalen User sind ACL's sicherlich eine gute Sache, z.B. um mal schnell die

[35] z.B. ptrace- oder doBreak-Exploit
[36] z.B. mit webmin
[37] Solche Pakete sind oft Distributionsabhängig (z.B. yast unter SuSE-Linux).
[38] z.B. das Kommando sudo bzw. visudo.
[39] Access Control Lists
[40] einer Datei oder einem Verzeichnis
[41] Access Control Entry's

Datei xyz ausschliesslich für die Benutzer meier, müller, lehmann freizugeben. Dies war bisher nur möglich, wenn der Administrator eine Gruppe für diesen Zweck einrichtete. Mit diesen Erweiterungen kann man den exponentiellen Blow-Up der Gruppen vermeiden.
Für den Administrator erhöhen ACL's die Komplexität der Rechteverwaltung auf den ersten Blick. Es ist aber nicht zwingend notwendig für den Administrator ACL's zu verwenden[42], er kann die Rechteverwaltung auch wie bisher handhaben.
Die ACL's sind im POSIX-Draft 1003.1e ausführlich beschrieben, jedoch hat es der POSIX-Draft 1003.1e nie zur Verabschiedung geschafft.

4.4.11 Aktualisierung (Updates)

Mittlerweile gibt es in allen UNIX-Systemen Tools für die Aktualisierung des Systems. In einigen Fällen muss der Administrator sich selbst um die Aktualisierung kümmern, indem er Paket für Paket per Hand installiert, in anderen Fällen reicht dazu ein einfacher Aufruf[43].

4.4.12 standard Dienste, offene Ports

Ein gut konfiguriertes System bietet nur die Dienste an, die es auch wirklich anbieten soll. Je weniger offene Ports desto weniger mögliche Fehlerquellen, unso weniger Administrationsaufwand hinsichtlich der Sicherheit. Bei einigen UNIX- bzw. Linux-Varianten sind nach der Installation standardmäßig viele Ports offen. Das liegt meist daran, dass installierte Dienste sofort aktiviert werden. Dies mag für einige Nutzer wesentlich sein, da sie das aus dem Umgang mit Microsoft-Produkten gewohnt sind[44].
Bei relevanten System (wie z.B. Servern) ist es daher wichtig diese Punkte nach der Installation zu überprüfen und entsprechend umzukonfigurieren.

4.4.13 Sinn und Unsinn von CryptoFS

In vielen UNIX- bzw. Linux-Varianten besteht die Möglichkeit Daten zu verschlüsseln. Bisher war es relativ einfach an die Daten eines lokalen Systems[45] heranzukommen. Mittlerweile ist es möglich ganze Dateisysteme für den Nutzer transparent zu verschlüsseln. Dies macht UNIX auch für portable Systeme interessant, die sensible Daten enthalten. Da die Verschlüsselung zugunsten der Geschwindigkeit symmetrisch erfolgt, gelten natürlich auch hier die Einschränkungen ungünstig gewählter Kennwörter.

4.5 Ausblick

4.5.1 Technischer Fortschritt vs Sicherheit

Bei dem technischen Fortschritt stellt man sich nun die Frage ob es immer sinnvoll ist die aktuellste Software einzusetzen. Sicherlich ist es durchaus richtig die Software zu aktualisieren, um damit bekannte Sicherheitslücken zu schließen. Bei einigen Linux-Distributionen legt man sehr großen Wert auf sichere Software und verzichtet dann

[42] ACL's sind in keiner uns bekannten UNIX- oder Linux-Variante bereits vordefiniert.
[43] z.B. bei Debian mit apt-get upgrade
[44] In der Tat ist der durchschnittliche Windows-Nutzer gewohnt, dass ein Programm nach wenigen Mausklicks läuft. Ein eventuell nachfolgendes Desinteresse geht dann auf Kosten der Systemssicherheit.
[45] Ein System wozu man physikalisch Zugang hat.

lieber auf neue technische Features. Debian Linux, z.B. bietet seine Software in den drei Kategorien an: stable, unstable und testing. Manch einen wird es verwundern, dass in der stable-Version ein 2.2.x-Linux-Kernel zum Einsatz kommt (der aktuelle Kernel hat die Version 2.6.0). Für die meisten Anwendungsgebiete der stable-Version (wie z.B. jegliche Art von Servern) werden Features des neuesten Kernels oft nicht benötigt und man legt daher mehr Wert auf Stabilität und Sicherheit.

4.5.2 Zukünftige Sicherheitskonzepte (TCPA, DRM)

Von IBM gibt es bereits jetzt erste Treiber für TCPA, inklusive Quelltext. Diese Technologie wird vor UNIX keinen Halt machen. Aufgrund des hohen Anteils an offenen und freien Systemen bei BSD besteht die Hoffnung, dass sie für den Nutzer und nicht gegen ihn eingesetzt wird.

4.5.3 Zukünftige Anwendungsgebiete

UNIX Derivate allen voran Linux werden mehr und mehr auch für den Privatgebrauch bzw. für den Desktopbetrieb interessant. Dies liegt nur zum Teil an der großen Leistungsfähigkeit heutiger PCs. Vorreiter ist hier Apple, die mit *MAC OS X* ein BSD basiertes Microkernel-System für Privatnuzer herausgebracht haben. Grafisch einfach gestaltete Installationsroutinen, riesige Softwarepackete, sowie die geringen Anschaffungskosten machen BSD und Linux hier immer attraktiver.
Noch einen Schritt weiter geht man bei Embedded Systemen[46]. Als Universalbetriebssystem ist UNIX zwar nicht für klassische Kleinstsysteme geeignet[47], aber wenn jeder Toaster einen Webserver laufen lässt, dann werden auch UNIX-Systeme das Betriebssystem stellen.

4.6 Zusammenfassung

Der Systemverwalter hat klassischerweise sehr viele Rechte. Dies lässt sich bisher nur schwierig beherrschen, aber neue Lösungsansätze sind vielversprechend. Ein komplexes IT-System zu administrieren ist eine sehr umfangreiche Aufgabe. Naturgemäß unterlaufen Menschen Fehler, die hier fatal sein können. Um der Sache Herr zu werden muss man die Komplexität der Aufgabe mit Hilfswerkzeugen verringern. Welches Tool geeignet ist, muss an Hand der Umgebung immer wieder neu Festgelegt werden. Die Sicherheit von IT-Systemen ist immer auch von deren Umfeld abhängig. Der Informationsfluß zwischen den zuständigen Personen einer Institution muss dazu genau definiert und eingehalten werden.

Heutige Hard- und Software enthält Fehler! Man kann nur versuchen sicherheitsrelevante Fehler zu beseitigen, bevor sie von einem Angreifer ausgenutzt werden können. Die der UNIX-Welt inhärente Vielfältigkeit kann auch vorteilhaft sein, da die Fehler oft jeweils andere sind. Damit sinkt die Warscheinlichkeit ein komplexes System zu unterlaufen.

[46] Diese Entwicklung birgt generell Risiken wenn man beachtet, dass sich schon heute Hacker auf Netzwerdruckern wohl fühlen. Sie ist allerdings nicht aufzuhalten.
[47] broken by design

Literaturverzeichnis

[1] FreeBSD. http://www.freebsd.org/.

[2] Linux from scratch. http://www.linuxfromscratch.org/.

[3] Linux intrusion detection system. http://www.lids.org/.

[4] Linux kernel. http://www.kernel.org/.

[5] NetBSD. http://www.netbsd.org/.

[6] OpenBSD. http://www.openbsd.org/.

[7] Suse linux. http://www.suse.com/.

[8] Webmin. http://www.webmin.com/.

[9] Benjamin Benz. Abgeschottet: Die Rechte des Linux-Administrators einschränken. *c't magazin für computer und technik*, 25:212–214, 2003.

[10] Dr. Oliver Diedrich. Dateisystem: Access Control Lists unter Linux. *c't magazin für computer und technik*, 23:218–221, 2003.

[11] Marcel Gagne. *Linux System Administration*. Addison-Wesley, 2001.

[12] Torbjoern Gripp, Günter Klappheck, Peter Glinsky, and Frank Gehrke. *LINUX - Das Buch*. SYBEX-Verlag, Duesseldorf (Germany), 1999.

[13] Jörg Holzmann and Jürgen Plate. *Linux-Server für Intranet und Internet*. Hanser-Verlag, Muenchen (Germany), 2000. Den Server einrichten und administrieren.

[14] John R. Levine and Margaret Levine Young. *Unix für Dummies. Gegen den täglichen Frust mit Unix*. MITP, 2000.

[15] Evi Nemeth, Garth Snyder, Trent R. Hein, Adam Boggs, Rob Braun, Ned McClain, and Ned MacClain. *Linux Administration Handbook*. Prentice Hall PTR, 2002.

[16] Alessandro Rubini and Jonathan Corbet. *Linux Device Drivers*. O'Reilly and Associates, 2001.

[17] Shelly Powers and Jerry Peek and Tim OReilly and Mike Loukides. *UNIX Power Tools*. O'Reilly and Associates, 2002.

[18] Andrew S. Tanenbaum. *Computernetzwerke*. Addison-Wesley, 2000. 3. Auflage.

[19] Andrew S. Tanenbaum. *Moderne Betriebssysteme*. Addison-Wesley, 2002.

[20] Andrew S. Tanenbaum and Goodman. *Computerarchitektur - Strukturen, Konzepte, Grundlagen.* Addison-Wesley, 2001.

[21] Arnold Willemer. *Wie werde ich UNIX-Guru? - Einführung in UNIX, Linux und Co.* Galileo Press, 2002.

5 Grundsätzliche Strukturen von UNIX-Software

C. Fricke, S. Krahmer

5.1 Einleitung

Macht man sich Gedanken über Unix, so fallen jedem, der mit der Unterstützung von Computern arbeitet, sofort die Schlagworte Webserver, Mainframes und so weiter ein. In Bezug auf den sicherheitstechnischen Aspekt denkt man sofort an grosse Firewalls und Scanner. Weiterhin hat man immer die Vorteile der Stabilität und Sicherheit gegenüber anderen Systemen im Hinterkopf.
Mit der zunehmenden Akzeptanz von Unix im Endanwenderbereich und speziell im öffentlichen Leben [1] erweitern sich die Anforderungen hinsichtlich der Sicherheit, wie sie bis dahin lediglich an andere Systeme gestellt worden sind. Sind diese Ziele doch für "Standardangriffe" ebenso attraktiv geworden.
Bedenkt man jedoch, dass 80 - 90 Prozent aller Angriffe auf ein System von "innen", also unter Mitwirkung von Menschen innerhalb der jeweiligen Systems stattfinden, so wird schnell klar, dass Firewalls und Virenscanner lediglich einen kleinen Teil der Arbeit erledigen können, um ein Computersystem zu schützen und abzusichern. Ebenfalls muss man sich darüber im Klaren sein, das eine "Tür", die existiert, niemals gegen Einbruch vollkommen gesichert werden kann. Es ist lediglich möglich, Eindringlingen ihr Handwerk so schwer wie möglich zu gestalten und den Kreis potentieller Angreifer so klein wie möglich zu halten.

Hinsichtlich dieser Anforderungen gilt auch in diesem Bereich, wie auch in fast allen Anderen des Lebens, der Grundsatz *"Wissen ist Macht"*.
Macht macht jedoch auch immer jede Menge Arbeit.

Im Verlauf dieser Arbeit nehmen alle gemachten Aussagen, Bezug auf Linux- und BSD-Software. Hierbei wird dabei davon Gebrauch gemacht, dass es sich bei Linux und BSD um Unixderivate handelt und somit die behandelten Daten und Fakten für die diversen Ausprägungen von Unix ebenso gelten.

5.2 Softwareprinzip

Für die diversen Unix Systeme gibt es von den Herstellern direkt eine Fülle von proprietärer Software, die neben den eigentlichen Betriebssystemen die eigentliche Arbeit im jeweiligen Anwendungsspektrum erst ermöglichen. Daneben gibt es aber auch, bedingt durch die Systemstruktur, jede Art von Software, welche frei und in Quellen verfügbar ist. Hierbei spielt das Unixderivat "Linux"[2] eine besondere Rolle. Hier ist die komplette Software einschließlich des Kernels im Quellcode frei, unter der "General Public License"[3] verfügbar.
Die Hauptidee von Unixsoftware, gleich welcher Art, ist jedoch die Verteilung der eigentlichen Arbeit auf viele kleine Tools, sowie die klare und strenge Trennung von Kernel und Userspace innerhalb einer Anwendung. In Windowssystemen finden sich im Gegensatz hierzu für verschiedene Anwendungen, wie zum Beispiel dem "Internet Explorer", ganze Teile einer Anwendung im Kernelspace. Dieser Umstand wirkt sich jedoch nachteilig auf die Stabilität des gesamten Betriebssystems aus.
Das Prinzip von Unixsoftware ist somit nicht ein Programm zu haben, mit dem man alles "erschlagen" kann, sondern viele kleine Tools mit denen man zusammengenommen genau die gleiche Arbeit erledigen kann.

5.2.1 Architektur

Für die unter Unix Verwendung findende Architektur der vielen kleinen Tools ist jedoch oberste Vorsicht geboten. Der Idee der größtmöglichen Flexibilität steht hierbei die Eventualität des Austauschens einer kleinen Komponente und die damit verbundende Möglichkeit das Verhalten der Gesamtanwendung so zu verändern, das ein Angriff auf das System möglich oder sogar getätigt wird gegenüber. Ein Austausch des kleinen Programms "ls" (list directory contents) verhindert beispielsweise das Entdecken von Fremdsoftware. Andere Software, die dieses Programm benutzt, hat somit ebenso ein Fehlverhalten.
Für diejenigen Benutzer die dazu berechtigt sind Programme zu installieren oder zu verwalten gilt daher, sich bei jeder Aktion genauestens über etwaige Reaktionen zu informieren.
Um Angriffe von Innen oder problematische Aktionen seitens der Anwender zu verhindern obliegt es dem jeweiligen Administrator hinsichtlich der Benutzerrechte den kleinsten gemeinsamen Nenner zu finden. Diese Rechteverwaltung liegt aber ebenso in den Händen des Besitzers von Dateien, sofern diese nicht zum System gehören. Die Gefahr für Benutzer beschränkt sich jedoch, bei guter Administration durch den Systemverwalter, lediglich auf das Benutzerverzeichnis selbst.

5.2.2 Distribution

Für die unter diversen Distributionen verwendeten RPM-Pakete [4] ist eine sogenannte Skriptfähigkeit eingebaut. Hiermit ist es möglich, während der Installation Befehle im Rootkontext auszuführen. Somit reicht es oftmals schon, einen Blick vor der Installation in das Paket zu tätigen um drohendes Unheil abzuwenden.
Weiterhin werden für fertige Pakete wie auch für Programme, welche im Quellcode erhältlich sind, MD5-Hashsummen ausgeschrieben. Verglichen mit einer neu erstellten Hashsumme für den aktuellen Download kann sichergegangen werden, das es sich bei dem jeweiligen Paket um ein Original handelt.

Oben genannte Maßnahmen gelten natürlich genauso für etwaige Updates der vorhandenen Software, sei es aus Gründen der Sicherheit oder des Lebenszyklus der Programme.

5.3 Softwareentwicklung

Stimmt man eine Disskusion über das Für und Wider freier Software an, so erhitzen sich sehr schnell die Gemüter. Für die Autoren dieses Beitrags stellen sich Linux und FreeBSD als ein Musterbeispiel dezentraler Softwareenwicklung dar.

5.3.1 Kompilation

Ein Bestandteil der Defaultinstallation diverser Unixderivate ist fast immer ein C-Compiler. Die Begründung hierfür findet sich in der Idee, bei der Auswahl der benutzten Software im jeweiligen Anwendungskontext, die größtmögliche Flexibilität zu haben. So bleibt in jedem Fall nicht nur die Auswahl aus diversen Binärdistributionen, sondern ebenso die Möglichkeit Quellen von Progammen zu übersetzen und systemweit verfügbar zu machen.
Durch die Möglichkeit unbemerkt Dateien auszutauschen, siehe Kapitel 5.2.1, ergibt sich hier eine weitere Gefahr. Tauscht man beispielsweise den Compiler selbst oder eine wichtige Komponente wie beispielsweise die libc.so aus, ist es möglich, bei jedem Kompilationsvorgang von beliebigen Programmen Trojaner in die Software einzubauen. Durch die Distribution auf diese Art und Weise erstellter Software werden diese Trojaner auf diverse Rechner verteilt. In den 80er Jahren soll auf diese Art und Weise ein Virus mittels starkt frequentierter Software im Umlauf gewesen sein.
Auf reinen Server- oder Hochsicherheitssystemen sollte auch darüber nachgedacht werden ob die Installation eines Compilers unbedingt notwendig ist, da ein System ohne Compiler einem (lokalen) Angreifer das Leben erheblich erschwert.

5.3.2 Installation

Für die Installation und Distribution von Unixsoftware gelten die im Kapitel 5.2.2 geschilderten Gefahren sowie Vorsichtsmaßnahmen.
Des Weiteren stellt die Installation eines Tools oder Programmsystems immer eine Gefahr dar, erfolgt diese doch im Rootkontext, wenn die Installation systemweit verfügbar sein soll. Diese Gefahr äußert sich nicht gleich immer in Form eines Angriffes. Oftmals reicht schon die Installation oder das Update eines Programmsystems welche das Überschreiben oder Verändern von Systemdateien und somit die Inkonsistenz des Systems zur Folge hat. Ebenso tragen Updates nicht immer zur Verbesserung bei. Tragende Rolle spielen hierbei unentdeckte Fehler.
Für viele Hersteller proprietärer Unix- und anderer Betriebssysteme stellt sich diese Gefahr insbesondere durch die Art und Weise in der freie Software produziert und distributiert wird dar. Für die Autoren dieses Berichts liegt in der Infrastruktur der freien Software jedoch der immense Vorteil, das auf etwaige Fehler oder Unzulänglichkeiten viel schneller reagiert werden kann als bei unfreier Software.

5.3.3 Skriptunterstützung

Unter Unix wird die Skriptfähigkeit per Verwendung der verschiedenen Shells, beispielsweise csh oder tcsh [5], von beiden Seiten, also Angriffsquelle und Angriffsziel, wegen

ihrer Mächtigkeit sehr geschätzt. Shellskripte erleichtern dem Admistrator die Arbeit, indem ständig wiederkehrende Vorgänge automatisiert werden können. Diese Fähigkeiten werden ebenso durch Tools, wie beispielsweise "expect" [6] erbracht. Für die Anwendung dieses Werkzeuges gilt jedoch äußerste Vorsicht. Ist es doch möglich beispielsweise ganze FTP- oder SSH-Sessions per Script zu absolvieren. Somit sind in den entsprechenden Dateien Passwörter im Klartext abgelegt. Man sollte sich deshalb sehr genau überlegen ob und wie man dieses Werkzeug verwendet und wem der Zugriff auf die Skripte erlaubt ist.
Das Wissen um die Funktionsweise und Nutzung der Skriptsprachen erleichtern jedoch nicht nur Unixadministratoren sowie -benutzern die tägliche Arbeit, sondern sind natürlich auch für potentielle Angreifer attraktiv um um die notwendigen Schritte schnell und effizient zu erledigen.

5.4 Softwarenutzung

5.4.1 Systemstruktur

Für Umsteiger auf ein Unixsystem bildet die Filesystemstruktur eine der grössten Hürden beim Umgang mit dem System. Nach kurzer Einarbeitung scheint das System dennoch sehr logisch. Systemprogramme befinden sich immer unter "/sbin", normale Tools in "/bin" und so weiter. Heutzutage findet man Programme moderner Benutzeroberflächen auch unter "/opt". Jeder Benutzer hat zudem die Möglichkeit "private" Installationen in seinem Homeverzeichnis abzulegen.
Globale Konfigurationen findet man in "/etc" und solche die nur für den Benutzer selbst gelten in seinem Homeverzeichnis. Solch ein Verzeichnis bekommt jeder Benutzer bei Erstellung vom Administrator zugewiesen.
Die eigentliche Schwierigkeit ist es Unterschiede verschiedener Distributoren zu kennen und bei der Administration zu beachten.

5.4.2 Bibliotheken

Unter Unix unterscheidet man zwischen statisch und dynamisch gelinkten oder zu linkenden Bibliotheken. Die einzubindenden Bibliotheken befinden sich unter den Systempfaden "/lib" und "/usr/lib". Die Gefahr besteht hierbei beim Austausch der dynamischen "libs". Dadurch sind alle Programme betroffen, die mit Hilfe solcher Bibliotheken zur Ausführung kommen.
Wichtige und kritische Programme Programme sollten daher die benutzten Bibliotheken immer statisch linken, um solche Gefahren auszuschließen.

5.4.3 Dienste

Dienste, sogenannte Daemons, spielen in Unix- wie auch in anderen Betriebssystemen, eine tragende Rolle. Mit deren Hilfe werden Netzwerkdienste über Sockets oder andere Interprozesskommunikation (FIFO,...) angeboten. Solche Daemons müssen, da diese meisstens durch alle Benutzer eines Systems oder weltweit benutzbar sind, natürlich möglichst fehlerfrei und sicher programmiert sein. So ist es beispielsweise mit Hilfe von Bufferoverflows möglich, Aktionen auszuführen, für die die Dienste eigentlich nicht gedacht waren, zum Beispiel das Starten einer Shell.

5.4.4 IPC

Interprozesskommunikation spielt unter Unix, wie auch unter allen anderen Systemen, eine tragende Rolle für die Funktionsfähigkeit moderner Betriebssysteme. Hierbei muss immer sichergestellt sein, dass auch die richtigen Prozesse miteinander reden. Sei es über shared Memory, Sockets oder anderes. Für die Kommunikation über UNIX-Sockets (/dev/log) stellt sich ebenso das Rechteproblem. So sollten "private" Sockets so administriert sein, das diese auch nur privat genutzt werden können und nicht etwa durch alle Nutzer des Systems.

5.5 Administration

Eine vernünftige Administration ist, wie bei allen anderen Betriebssystemen auch, eine der wichtigsten Voraussetzungen für den sicheren und stabilen Betrieb eines Unixsystems. Es ist nicht möglich ein System auf Grund der in Kapitel 5.4.1 beschriebenen Probleme 100%ig sicher zu machen. Eine gründliche Vorbereitung bei Aufsetzen eines neuen Systems sowie ein gewissenhaftes und streng logisches Vorgehen bei der täglichen Arbeit helfen bei der Absicherung gegenüber Angriffen von Innen und Aussen.
Im folgenden wird davon ausgegangen, dass der Leser Kenntnis über die Existenz von Benutzern und Gruppen unter Unix besitzt.

5.5.1 suid

Welche Rechte User und Gruppen haben wird mit Permissionbits festgelegt, die man per chmod-Befehl ändern kann, siehe Kapitel 5.5.3. Zusätzlich kann festgelegt werden, was Leute tun dürfen, die nicht der Benutzer sind oder einer bestimmten Gruppe angehören. Ein Benutzer ist so in einer Art Käfig eingesperrt. Er kann nur auf Files zugreifen für die er, oder eine Gruppe der er angehört, entsprechende Rechte hat. Auch wenn der Benutzer Programme ausführt dann haben diese genau die selben Rechte wie er.
In einigen Fällen muss ein Benutzer allerdings Arbeiten erledigen dürfen, die er eigentlich nicht darf. Um zum Beispiel mit dem passwd-Befehl sein Passwort zu ändern, muss man Schreib- und Leserechte auf den Dateien "/etc/passwd" bzw "/etc/shadow" haben. Jeder Benutzer soll normalerweise sein Passwort ändern können aber nicht generelle Schreib- und Leserechte auf diese sensiblen Files haben.
Es wird also eine Mechanismus benötigt, mit dem ein Benutzer, beziehungsweise die Programme die ein Benutzer startet, eingeschränkt höhere Rechte bekommt. Der in Unix hauptsächlich verwendete, ist das Setzen des "set user id" Bits. Ist dies bei einem ausführbaren Programm gesetzt so wird im Gegensatz zur normalen Ausführung das Programm nicht mit den gleichen Berechtigungen des Benutzers ausgeführt, sondern mit den Rechten des Eigentümers der Datei, der als der Benutzer für diese eingetragen ist.

ls -la /usr/bin/passwd zeigt uns beispielsweise:

```
-rwsr-xr-x 3 root shadow  77204 2003-09-23 23:04 /usr/bin/passwd
```

Das "s" an der Stelle, an der normalerweise das "x", für eXecutable, stehen würde ist das suid Bit. Es besagt in obigem Fall, dass Jeder, der das passwd-Programm aufruft, und aufrufen dürfen es alle Benutzer auf diesem System, es mit den Berechtigungen vom "root" User tut. Das Programm, wenn es also einmal läuft hat "root"-Rechte und darf

alles tun was auch Root tun dürfte.
Hierdurch ergibt sich für diese Programme eine hohe Verantwortung. Sie müssen so programmiert sein, dass sie nur diejenigen Funktionen ausführen für die sie gedacht sind und nicht missbraucht werden können. Das passwd-Program aus oben gezeigtem Fall, funktioniert auf eine solche Art und Weise, dass es prüft, wer das Programm wirklich aufgerufen hat und jedem user ausser Root nur das Ändern des eigenen Passwortes erlaubt.
Analog hierzu ist die Funktionsweise des "set groupid" Bit. Ein Programm, mit gesetztem "group id" Bit wird mit den Gruppenberechtigungen der Gruppe ausgeführt der es angehört und nicht mit den Rechten der Gruppe der der Aufrufende angehoert.
Man sollte also niemals ein "s" Bit setzen, ohne genau zu wissen was man tut. Nur ein Programm, dass auch sicher geschrieben ist, sollte mit solchen Rechten versehen werden können. Suid Bits auf Skripten sind generell zu vermeiden da Interpreter eine zu grosse Fülle an Möglichkeiten bieten unerwartete Aktionen auszulösen.

5.5.2 chroot/jail

Chroot-Umgebungen werden zum Einen benutzt um neugierige oder möglicherweise nicht vertrauenswürdige Benutzer davon abzuhalten auf alle Dateien oder Programme zuzugreifen. Zum Anderen wird eine solche Umgebung dazu verwendet um potentiellen Angreifern die "Arbeit" zu erschweren, indem zum Beispiel der Zugriff auf unterstützende Programme, welche bei einem Angriff benötigt werden, verwehrt wird. Es ist beispielsweise möglich die Shellumgebung und somit die Skriptfähigkeit vollständig dem Jail vorzuenthalten. Hiermit werden natürlich alle in Kapitel 5.3.3 beschriebenen Möglichkeiten und Gefahren unterbunden. Jail als Konzept geht hier noch ein ganzes Stück weiter. Es wird nicht nur das Dateisystem beschränkt, sondern auch das Netzwerk als Ressource.

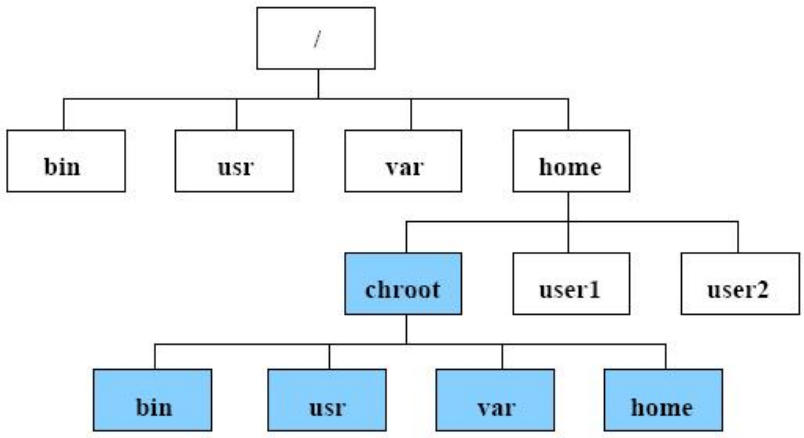


Abbildung 5.1: Beispiel für eine chroot-Umgebung.

Chroot ist verfügbar als Systemkommando, mit dem das Wurzelverzeichnis (siehe Kapitel 5.4.1) geändert wird. Die so erzeugte Umgebung wird jail genannt. Beispielsweise, wie in Bild [5.1] gezeigt, wird aus dem Originalverzeichnis "/home/chroot" einfach "/". Alle Dateien die nicht unterhalb von "/home/chroot" liegen, sieht man auch nicht mehr im Chroot-Jail.

Anbieter von Webspace oder Hosts [7], machen von dieser Möglichkeit gebrauch, indem sie für ihre Kunden ganze Distibutionen in ein solches Chroot-Verzeichnis packen um diesen grösstmögliche Flexibilität für Webanwendungen zu bieten.

Die chroot-Funktionalität ist in Daemons wie ftpd (und neuerdings auch sshd) bereits integriert, sodass fehlerhafte Programmierung einem Angreifer nicht gleich das gesamte System opfert. Wird chroot aus einem Programm heraus benutzt (mittels chroot() Systemcall) so ist darauf zu achten danach auch ein chdir() in dieses Verzeichnis zu machen, sonst ist es möglich aus dem jail auszubrechen.
Die Idee der Trennung von Prozessen durch Subsysteme, die durch Chroot bereitgestellt werden, kann durch einen typischen Administratorfehler zunichte gemacht werden. Prozesse in einem Jail dürfen unter keinen Umständen mit root-Rechten ausgeführt werden. Anderenfalls ist es möglich, unter Ausnutzung von Schwachstellen dieser Programme im root-Kontext, aus dieser Umgebung auszubrechen. Gerade diese Schwachstellen sind es jedoch, die durch chroot unter anderem ausgeglichen werden soll.
Ein Allheilmittel stellt die Möglichkeit der Schaffung von solchen Subsystemen natürlich nicht dar. So existieren bekannte, wenn auch nicht ganz triviale, Workarounds um aus den Jails auszubrechen.

5.5.3 Permissions

BSD beziehungsweise Linux verfügen wie jedes Unixsystem über ein ausgeklügeltes System zur Verwaltung der Zugriffsrechte auf Dateien und Devices. Solche Zugriffsrechte sind für Multiuser-Systeme unumgänglich, um die Dateien eines Benutzers vor Zugriffen der anderen Anwender zu schützen.
Die Zugriffsrechte sorgen auch dafür, dass Viren, wie man sie unter Windows findet, unter Unix praktisch keine Bedeutung haben. Ausserdem schützen sie den Anwender davor, aus Versehen sein System zym Beispiel mit einem "rm -rf /" zu beschädigen.
Jeder Benutzer erhält eine eindeutige Benutzerkennung, die auch UID (engl. user identification) genannt wird. In der Datei /etc/passwd sind alle Benutzer aufgeführt. Die erste Spalte enthält den Loginstring, mit dem man sich einloggt. In der dritten Spalte wird dem Loginstring dann die UID zugeordnet.
Neben der Benutzerkennung existieren unter Unix noch Gruppen. Auch einer Gruppe ist eine eindeutige Zahl, die sogenannte GID (engl. group identifcation) zugeordnet. Jeder Benutzer ist Mitglied wenigstens einer Gruppe. So ist beispielsweise "root" Mitglied der Gruppe "root" und ein Anwender Mitglied der Gruppe "users". Welcher Gruppe ein Benutzer angehört, sieht man in der vierten Spalte der Passwortdatei. Eine Liste der Gruppen findet man in der Datei "/etc/group". Der Name einer Gruppe wird in der ersten und die GID in der dritten Spalte definiert. Falls ein Benutzer Mitglied mehrerer Gruppen sein soll, kann man seinen Loginstring in die vierte Spalte der "/etc/group" Datei eintragen.

chmod

Im Unix Filesystem ist für jede Datei und jedes Verzeichnis gespeichert, wer dieses lesen (r), schreiben (w) und ausführen (x) darf.
Die Zugriffsrechte sind in drei Gruppen aufgeteilt. Die Erste bestimmt die Rechte des Besitzers der Datei. Die Zweite legt die Rechte für die Gruppe fest, der die Datei gehört. Als letztes werden die Rechte für alle übrigen Benutzer definiert.

Mit dem chmod-Befehl kann man nur die Rechte der Dateien verändern. Hierbei sind die zu setzenden Rechte oktal kodiert. Den Leserechten ist die Zahl 4, den Schreibrechten die Zahl 2 und den Ausführungsrechten die Zahl 1 zugeordnet. Die zu setzenden Rechte sind jeweils zu addieren. Um also zum Beispiel einer Datei test.txt die Rechte zu geben, die dem Eigentümer Lesen und Schreiben, sowie allen anderen Benutzern das Leserecht einräumt, würde man folgenden Befehl benutzen:

```
chmod 644 test.txt
```

Die erste Zahl von links bestimmt die Benutzerrechte (4+2), die zweite die Gruppenrechte (4) und die dritte die Rechte der übrigen Benutzer (4). Dieses Verfahren erscheint im ersten Moment vielleicht etwas kompliziert, lässt sich aber relativ schnell lernen.
Neben den r/w/x-Rechten gibt es, wie schon im Kapitel 5.5.1 beschrieben, noch die SUID (engl. set UID) und SGID (engl. set GID) Rechte. Diese sorgen bei einer ausführbaren Datei dafür, dass das Programm nicht unter der UID bzw. GID des aufrufenden Benutzers läuft, sondern die UID bzw. GID des Besitzer der Datei benutzt wird.
In jedem Fall obliegt es dem Administrator als oberste Instanz und den Benutzern für ihre eigenen Dateien, Rechte wohlüberlegt und sorgsam zu setzen. So ist es beispielsweise notwendig, um Angriffe von innen zu verhindern, den "normalen" Benutzern in Verzeichnissen welche Systemdateien oder Bibliotheken enthalten, keine Schreibrechte zu gewähren.

umask

Mit dem umask-Befehl kann man bestimmen, welche Rechte beim Anlegen einer neuen Datei nie gesetzt sein sollen. So möchte man in der Regel nicht, dass die Gruppe und alle anderen die eigenen Dateien modifizieren können. Deshalb benutzt man

```
umask 022
```

um jeweils die Schreibrechte zu löschen. Die "umask" wird typischerweise in "/etc/profile" gesetzt.
Beispielsweise legt der "vi", der wohl wichtigste Editor im Notfall, Dateien mit den Rechten "-rw-rw-rw-", oder oktal 0666, an. Das bedeutet, dass jeder Benutzer die Datei schreiben kann. Mit der oben angegebenen Maske verhindert man genau dies. Aus 0666 wird jetzt immer 0644. Das heisst, die Datei ist lediglich lesbar für andere Benutzer oder Gruppen.

5.5.4 ACL

*A*us nicht näher benannten Gründen wurden die betreffenden Standardentwürfe POSIX 1003.1e und POSIX 1003.2c zurückgezogen. ACL-Implementationen verschiedener UNIX-artiger Betriebssysteme basieren allerdings auf diesen Dokumenten. Eine letzte Referenz auf diese Dokumente findet sich unter [9]

Traditionell ist ein Dateiobjekt unter Linux mit drei Sets von Berechtigungen assoziiert. Diese Sets, wie auch schon in Kapitel 5.5.3 beschrieben, geben die Lese- (r), Schreib- (w) und Ausführungsrechte (x) für die drei Benutzerklassen Eigentümer der Datei (engl. owner), Gruppe (engl. group) und Rest der Welt (engl. other) wieder. Zusätzlich können noch die set user id, set group id und sticky Bits gesetzt werden.
Für die meisten in der Praxis auftretenden Fälle reicht dieses schlanke Konzept völlig

Typ	Textform
owner	user::rwx
named user	user:name:rwx
owning group	group::rwx
named group	group:name:rwx
mask	mask::rwx
other	other::rwx

Tabelle 5.1: Typen von ACL-Einträgen

aus. Für komplexere Szenarien oder fortgeschrittenere Anwendungen mussten Systemadministratoren bisher eine Reihe von Tricks anwenden, um die Einschränkungen des traditionellen Rechtekonzepts zu umgehen.
In Situationen, in denen das traditionelle Dateirechte-Konzept nicht ausreicht, helfen ACLs. Sie erlauben es, einzelnen Benutzern oder Gruppen Rechte zuzuweisen, auch wenn diese nicht mit dem Eigentümer oder der Gruppe einer Datei übereinstimmen. Mit Hilfe der Access Control Lists können komplexe Szenarien umgesetzt werden, ohne dass auf Applikationsebene komplexe Rechtemodelle implementiert werden müssen.
Ein bekanntes Beispiel der Verwendung von ACL ist Samba, mit Hilfe dessen Windowsclients Datei- und Druckdienste auf einem Unixsystem angeboten werden und andererseits Linuxrechner auf Windowsfreigaben zugreifen können.
Da Samba Access Control Lists unterstützt, können Benutzerrechte sowohl auf dem Linux-Server als auch über eine grafische Benutzeroberfläche unter Windows eingerichtet werden. Über einen Daemon ist es sogar möglich, Benutzern Rechte einzuräumen, die nur in der Windowsdomain existieren und über keinen Account auf dem Unix-Server verfügen.
ACLs werden grundsätzlich in zwei Klassen eingeteilt. Eine minimale ACL besteht ausschliesslich aus den Einträgen vom Typ owner (Besitzer), owning group (Besitzergruppe) und other (Andere), und entspricht den herkömmlichen Berechtigunen für Dateien und Verzeichnisse. Eine erweiterte ACL geht über dieses Konzept hinaus. Sie muss einen mask (Maske) Eintrag enthalten und darf mehrere Einträge des Typs named user (namentlich gekennzeichneter Benutzer) und named group (namentlich gekennzeichnete Gruppe) enthalten. Tabelle 5.1 fasst die verschiedenen verfügbaren Typen von ACL-Einträgen zusammen.
In den Einträgen owner und other festgelegte Rechte sind immer wirksam. Vom mask Eintrag abgesehen, können alle übrigen Einträge (named user, owning group und named group) entweder wirksam oder maskiert werden. Sind Rechte sowohl in einem der oben genannten Einträge als auch in der Maske vorhanden, werden sie wirksam. Rechte, die nur in der Maske oder nur im eigentlichen Eintrag vorhanden sind, sind nicht wirksam. Das nachfolgende Beispiel verdeutlicht diesen Mechanismus (Tabelle 5.2)
Grundsätzlich werden die ACL-Einträge in folgender Reihenfolge untersucht: owner, named user, owning group oder named group und other. Über den Eintrag, der am besten auf den Prozess passt, wird schliesslich der Zugang geregelt. Komplizierter werden die Verhältnisse, wenn ein Prozess zu mehr als einer Gruppe gehört, also potentiell auch mehrere group Einträge passen könnten. Aus den passenden Einträgen mit den erforderlichen Rechten wird ein beliebiger ausgesucht. Für das Endresultat "Zugriff gewährt" ist es natürlich unerheblich, welcher dieser Einträge den Ausschlag gegeben hat. Enthält

Typ	Textform	Rechte
named User	user:krahmer:r-x	r-x
mask	mask::rw-	rw-
	wirksame Rechte	r–

Tabelle 5.2: Beispiel eines ACL-Eintrages

keiner der passenden group Einträge die korrekten Rechte, gibt wiederum ein beliebiger von ihnen den Ausschlag für das Endresultat "Zugriff verweigert".

5.6 Zusammenfassung

Wie schon im Kapitel 5.1 erwähnt gibt es kein 100%ig sicheres Betriebssystem. Das gilt, wie aus den vorangegangenen Erläuterungen ersichtlich, auch für Unix und die diversen Derivate.
Wichtigstes Instrumentarium für ein "ruhigeres Leben" ist immer eine gute Administration. Dies gilt nicht nur für den laufenden Systemstatus sondern auch für folgende Installationen diverser benötigter Programme.

Mit UNIX als Betriebssystem hat man jedoch schon einen sicheren Grundstein gewählt das in den Punkten Softwaredesign bzw. -prinzip, Softwareentwicklung und Administration solide ausgelegt ist. Das von Anfang an integrierte User/Group Konzept sowie das in neuerer Zeit ergänzend hinzugekommende ACL Konzept erlauben es leicht eine sichere Basiskonfiguration zu erstellen. Werden die beschriebenen Administrationsfehler (weltweite Leserechte, suid bit aus Skripten usw.) vermieden und Softwareupdates regelmäßig eingespielt so erreicht man zwar keine 100%ige Sicherheit jedoch eine ausreichende Sicherheit für ein Produktions-, Endanwender- oder Serversystem.

Literaturverzeichnis

[1] München arbeitet an Konzept für Linux-Migration: http://www.heise.de/newsticker/data/anw-15.12.03-003/default.shtml

[2] Linux: http://www.linux.de/linux/

[3] GPL: http://www.gnu.org/copyleft/gpl.html

[4] RPM: http://www.rpm.org/

[5] Paul DuBois, *U*sing csh & tcsh, 1st Edition July 1995

[6] Don Libes, *E*xploring Expect, 1st Edition December 1994

[7] Strato Server Pakete: http://www.strato.de/server/index.html

[8] Posix ACL: POSIX Access Control Lists on Linux, Andreas Grünbacher *U*SENIX Annual Technical Conference San Antonio, Texas, June 2003

[9] Posix ACL Standardentwürfe: http://wt.xpilot.org/publications/posix.1e/

6 Standard IP-Dienste

A. Kloth, S. Uhlmann

6.1 Einleitung

In unserem Teil der Betrachtungen zur Absicherung komplexer IT-Infrastrukturen, analysieren wir den Bereich der IP-Dienste. Darunter verstehen wir all die netzwerkbasierten Dienste, die über das Internet Protocol (IP) verfügbar sind. Dies wären z.B. die klassischen Dienste wie WWW, Email, FTP usw. aber auch die neueren Kommunikationsdienste wie VoIP [1]. Wir beschränken uns jedoch auf ein typisches Einsatzszenario. Als Beispiel dient eine kleinere mittelständische Firma im Bereich der Softwareentwicklung mit 20 Mitarbeitern. Die Firma hat sich auf den Namen Bingo! GmbH getauft.

6.2 Allgemeine Aspekte

6.2.1 Systembeschreibung

In der Firma werden folgende Dienste angeboten:

- Email
- Intranet-Informationssystem
- zentrale Dateiverwaltung

Auch wenn das nur ein Teil der Dienste sind, die in einer Firma benötigt werden, wollen wir exemplarisch daran die von den Bereich der IP-Dienste analysieren.

Analyse der Funktionsweise

E-Mail: Von jedem Arbeits-PC aus kann der Benutzer mittels eines Email-Clients elektronische Nachrichten verschicken; entweder intern an andere Bingo!-Mitarbeiter oder extern an andere Benutzer des Internets. Dazu wird die Email-Adresse des Empfängers benötigt. Die Email wird vom Client-System an einen zentralen Mailserver (Mail-Relay, „smart host“) übertragen, welcher sich dann um den Versand der Email an den Empfänger kümmert. Ankommende Emails werden vom gleichen Email-Server angenommen und zwischengespeichert bis sie vom Benutzer abgeholt werden.

[1] Voice over IP, IP-Telefonie

Informationssystem: Die Mitarbeiter von Bingo! können sich auf Intranet-Webseiten diverse Informationen einholen, z.B. interne Dokumente der Firma, Urlaubs- und Anwesenheitszeiten sowie die Adressen und Telefonnummern der anderen Mitarbeiter oder technische Dokumentationen. Weiterhin gibt es ein Wiki, auf dem alle Mitarbeiter Informationen ablegen und untereinander zusammenarbeiten können. Um auf die Intranet-Webseiten und das Wiki zugreifen zu können, wird lediglich ein Webbrowser benötigt.

Dateiverwaltung: Zur Speicherung eigener Dateien und zum gemeinsamen Dateiaustausch gibt es einen zentralen Fileserver, auf den die Arbeits-PCs über das Netzwerk zugreifen können. Sowohl die Entwickler, die auf Linux-Systemen arbeiten, als auch die Mitarbeiter von Vertrieb, Marketing und der Vorstand, die auf Microsoft Windows-Systemen arbeiten, können mit ihren PCs auf die selben Daten zugreifen.

Art der Implementierung

E-Mail: Der de facto Standard zur Übertragung von Email ist das SMTP-Protokoll, das die Bingo! GmbH auch verwendet. Eingehende Emails werden vom SMTP-Server entweder an den Empfänger weitergeleitet oder wenn es sich um einen lokalen Empfänger handelt, gleich auf dem Server gespeichert. Die Benutzer können ihre Emails mittels dem POP3-Protokoll abholen. Theoretisch könnte man auf POP3 verzichten und die Emails per SMTP direkt in eine für den jeweiligen Benutzer zugängliche Datei schreiben. Dazu müssten aber alle Email-Clients das gleiche Format zur Speicherung der Emails verwenden. Dies ist jedoch in heterogenen Umgebungen wie bei der Bingo! GmbH oft nicht der Fall. SMTP kennt in seiner Grundform keine Authentifizierung und die Bingo! GmbH benutzt keine der vorhandenen Erweiterungen, wie SMTP Auth oder POP-before-SMTP, da der Email-Server nur intern von den eigenen Mitarbeitern verwendet wird. Zum Abholen der Emails sieht das POP3-Protokoll jedoch eine Authentifizierung mittels Benutzername und Passwort vor. Eine Verschlüsselung der SMTP- und POP3-Verbindungen ist nicht vorgesehen. Die Protokolle sind in den RFCs 2821 (SMTP) und 1939 (POP3) genau definiert. Daher soll an dieser Stelle darauf nicht weiter eingegangen werden.

Informationssystem: Die Intranet-Webseiten und das Wiki werden von einem Webserver mittels des HTTP-Protokolls an die Clients übermittelt. Dies ist ein einfaches request/response-Protokoll, das die jeweiligen Dokumente mittels URLs lokalisiert. Eine vorherige Authentifizierung oder Verschlüsselung ist von Bingo! nicht vorgesehen, da davon ausgegangen wird, dass nur Mitarbeiter auf das interne Netz und damit den Webserver Zugriff haben. Das HTTP-Protokoll ist genauer im RFC 2616 definiert. Das Wiki erfordert eine Datenbank. Daher wird vom Webserver noch eine Verbindung zu einem RDBMS [2] aufgebaut. Das verwendete Netzwerkprotokoll ist nicht weiter standardisiert. Die zur Abfrage der Datenbank verwendete Sprache SQL, ist in mehreren Versionsschritten standardisiert worden, zuletzt als SQL-92 und ANSI SQL 99. Die verschiedenen RDBMS unterstützen meist mehr oder weniger große Teile des Standards.

[2] Relational Database Management System

Dateiverwaltung: Die zentrale Dateiverwaltung kommuniziert mittels der Protokolle NFS und SMB mit den verschiedenen Clients. NFS ist ein von Sun Microsystems entwickeltes „Network File System“ und ein Protokoll, über das ein Client-Computer auf Dateien eines Servers über das Netzwerk genauso zugreifen kann, als wären sie auf der lokalen Festplatte vorhanden. NFS benötigt das Sun RPC [3] Protokoll. NFS ist genauer im RFC 1094 spezifiziert. SMB wurde von IBM entwickelt, doch die bekannteste Implementation stammt von Microsoft und wird dort CIFS [4] genannt. CIFS wird in Microsoft Windows für Datei- und Druckerfreigaben genutzt. Das Protokoll enthält etliche undokumentierte Erweiterungen des ursprünglichen Protokolls, doch es gelang dem Samba Projekt, eine Implementation von SMB/CIFS zu schreiben, die dafür geeignet ist, Windows-Clients Zugriff auf einen Unix-Server zu geben. Es gab einen Versuch seitens Microsoft SMB bzw. CIFS beim IETF zu standardisieren, doch dieser scheiterte an der Unvollständigkeit und Ungenauigkeit der Dokumentation.

Alle beschriebenen Protokolle benötigen als Einsatzumgebung lediglich eine vorhandene IP-Infrastruktur (Internet oder Intranet). SMB bzw. das darunter liegende NetBIOS könnte auch auf anderen Protokollen laufen (z.B. IPX, NetBEUI), doch ist dies in der Praxis heutzutage nicht mehr üblich, da sich TCP/IP durchgesetzt hat. Samba erfordert auch TCP/IP.

Durch das plattformunabhängige Design von IP, können sowohl Microsoft Windows- als auch GNU/Linux-Systeme auf die Dienste zugreifen. Eine kleine Ausnahme bilden hier SMB und NFS. Auch wenn es theoretisch möglich ist, GNU/Linux-Systeme mittels Samba und Windows-Systeme mittels NFS zu verbinden, hat sich dies in der Praxis nicht bewährt. SMB ist stark Windows-orientiert und Samba auf Linux dient meist nur Windows-Clients als Server. NFS wiederum ist eher Unix-orientiert (wenn auch innerhalb von Unix sehr plattformunabhängig) und es gibt wenig NFS-Server und Client-Software für Windows und keine davon hat sich breit durchgesetzt.

Welche Systeme werden zur Funktionserbringung benötigt?

Email: Für den Email-Dienst wird ein Server-System benötigt, auf dem eine SMTP- und eine POP3-Server-Software laufen. Sowohl für das Betriebssystem als auch für die jeweilige Server-Software gibt es eine breite Auswahl an Angeboten. Die Bingo! GmbH hat hier konsequent auf freie Software gesetzt, da sie flexibel bleiben und sich nicht an einen Hersteller binden wollte. So wurden als Betriebssystem GNU/-Linux, als SMTP-Server exim und als POP3-Server teapop verwendet.

Sowohl der SMTP-Server als auch der POP3-Server können auf dem selben System installiert werden. Beides zu trennen würde für eine Firma dieser Größe nicht verhältnismäßig sein.

Informationssystem: Das Informationssystem benötigt einen Webserver zur Präsentation der Informationen und einen Datenbankserver zur Speicherung der Daten im Wiki. Auch hier verwendet die Bingo! GmbH freie Software und wählte als Webserver

[3] Remote Procedure Call
[4] Common Internet Filesystem

den Marktführer Apache, der auch auf einem GNU/Linux-System installiert wurde. Oft werden Webserver und RDBMS auf einem Server installiert und für das Informationssystem wäre das auch ausreichend. Da die Bingo! GmbH aber auch für größere Kunden entwickelt, die sog. Multi-Tier Systeme betreiben, hat sie sich entschieden, Webserver und RDBMS auf zwei verschiedene Systeme zu verteilen, damit die Entwickler beide auch für Tests und Demo-Installationen nutzen können. Als Wiki-Software wurde Wikki Tikki Tavi aus der breiten Auswahl ausgesucht. Dadurch ergab sich auch die Software für das RDBMS, denn Tavi benötigt MySQL, welches ebenso auf einem GNU/Linux-System läuft. Tavi benötigt weiterhin die dynamische Scriptsprache PHP, die als Modul in den Webserver integriert werden kann.

Dateiverwaltung: Für den Fileserver wird ein zentrales Server-System benötigt. Es stand kurzzeitig zur Diskussion, ob ein Server unter Microsoft Windows verwendet werden soll, damit es keine Probleme mit der Anbindung der Windows Clients gibt. Dieser Gedanke wurde jedoch wieder verworfen, nachdem festgestellt wurde, dass Samba als freie Implementation des SMB-Protokolls, genauso gut geeignet ist, und man damit auch GNU/Linux als Betriebssystem einsetzen kann. Die Linux-Clients der Entwickler werden jedoch über einen NFS-Server an den Fileserver angebunden, da NFS besser mit dem Dateiverwaltungskonzept von Linux harmoniert.

Alle drei Server befinden sich hinter einer Firewall in einem internen privaten Netzwerk. Die Firewall bildet gleichzeitig das Gateway für alle anderen Hosts im Netzwerk und ermöglicht ihnen mittels IP-Masquerading (NAT) den Zugriff auf das Internet.

Von der Konstruktion (Architektur der Implementierung) vorgesehener Einsatzzweck, -umgebung

Email: Der Einsatzzweck des SMTP-Protokolls ist die Weiterleitung von Email. Das Protokoll POP3 ist geeignet Emails von einem Server abzuholen.

Informationssystem: Das HTTP-Protokoll ist prinzipiell dafür geeignet beliebige Daten zwischen Client und Server zu transferieren. Welche Daten das sind (HTML-Seiten, Bilder, Videos, ...) ist dabei völlig egal. Der Einsatzzweck von MySQL ist, wie bei jeder anderen relationalen Datenbank auch, Daten in strukturierter Form abzuspeichern und mittels der SQL-Sprache die Möglichkeit geben diese Daten zu manipulieren.

Dateiverwaltung: Der Einsatzzweck des SMB-Protokolls ist es Dateien, Drucker und andere Ressourcen eines Computers mit anderen Computern gemeinsam zu nutzen. Es ist seit einigen Jahren in Microsoft Windows integriert und eignet sich daher besonders zum Dateiaustausch zwischen und mit solchen Systemen. NFS hat seinen Einsatzzweck allgemein auf Unix-artigen Systemen und dient ebenso dazu, Dateien eines Computers einem anderen Computer zugänglich zu machen.

Ist die Konstruktion des Systems zweckmäßig für den vorgesehenen Einsatzzweck?

Email: Das SMTP und POP3 sind der allgemeine Standard zur Email-Kommunikation. Quasi alle Email-Client-Programme unterstützen diese Protokolle. Da keine es weiteren Anforderungen, wie z.B. Workgroup-Funktionalitäten, gibt, reicht dies auch aus.

Informationssystem: Der Webserver und damit das verwendete HTTP-Protokoll ist sehr zweckmäßig für ein Informationssystem, da dieser sehr gut mit Standardprogrammen (Webbrowser) auf allen eingesetzten Client-Plattformen zugänglich ist.

Dateiverwaltung: Durch die Bereitstellung sowohl eines NFS-Servers für die Linux-Clients als auch eines Samba-Servers für die Windows-Clients, ist die Dateiverwaltung gut kompatibel zu beiden System-Typen. Die zu erfüllende Aufgabe, nämlich der Dateiaustausch zwischen allen Mitarbeitern, ist damit erfüllt.

6.2.2 Gebrauchsbeschreibung

Welche Parteien interagieren mit dem System (mittel- und unmittelbar)?

Es sind folgende Nutzergruppen bzw. Rollen identifizierbar:

- Benutzer
- Administratoren
- Management
- Kunden

Benutzer sind alle Mitarbeiter der Bingo! GmbH, die mit dem System arbeiten. Sie haben keine besonderen Rechte auf den jeweiligen Systemen und auch keinen physischen Zugang. Sie arbeiten greifen nur über das Netzwerk auf die Dienste zu.

Administratoren haben besondere Rechte auf den Systemen, um diese zu warten. Sie müssen die Server installieren und konfigurieren können und haben auch physischen Zugang zu den Servern. Neben ihrer Rolle als Administrator sind Administratoren aber auch normale Benutzer der Systeme.

Das *Management* der Bingo! GmbH hat sich die gleichen besonderen Rechte erbeten, wie sie auch die Administratoren haben. Sie nehmen diese Rechte aber nicht aktiv wahr, sondern verfügen nur darüber, um im Notfall auf die Systeme und damit kritische Ressourcen der Firma, zugreifen zu können. Regulär verhält sich aber auch das Management, wie alle anderen Benutzer.

Die *Kunden* der Firma nutzen das System nicht direkt. Es werden jedoch sie betreffende Daten, auf den Systemen verarbeitet und gespeichert.

Welche Interessen haben diese Parteien?

Die *Benutzer* der Systeme wünschen sie sich eine möglichst störungsfreie Verfügbarkeit der Dienste. Die im Informationssystem und der Dateiverwaltung enthaltenen Daten, sollen authentisch und integer sein. Sie wünschen sich, dass Emails Spam-frei und ohne Viren bei ihnen ankommen. Ressourcen, insbesondere Speicherplatz, sollen immer in ausreichender Menge vorhanden sein. Es besteht weiterhin der Bedarf, am Arbeitsplatz auch private Emails zu lesen und es liegt den Benutzern hier besonders am Herzen, dass auf diese Daten nicht von Dritten eingesehen werden.

Die *Administratoren*, haben ebenso ein Interesse, dass die Dienste störungsfrei verfügbar sind, denn danach wird die Qualität ihrer Arbeit beurteilt. Dazu ist es notwendig, bestimmte Systemparameter, wie z.B. Prozessor-, Speicher- und Festplattenauslastung, zu überwachen und bei Bedarf diese Ressourcen für die Benutzer zu limitieren. Damit unautorisierte Zugriffe bemerkt werden können, muss protokolliert werden, wer, wann Zugriff auf die Diensten genommen hat. Daten und insbesondere Emails müssen auf Spam und Viren untersucht werden, damit diese sich nicht auf die Systeme ausbreiten.

Das *Management* hat zusätzlich zu den Interessen als Benutzer der Systeme, ein besonderes Interesse an der Vertraulichkeit der Daten. Es möchte nicht, dass interne oder nur für ihre Kunden bestimmte Dokumente öffentlich gemacht werden. Weiterhin sollen bestimmte Daten, z.B. bzgl. der Personalverwaltung, von den restlichen Mitarbeitern nicht einsehbar sein. Das Lesen privater Emails ist bei der Bingo! GmbH vom Management erlaubt worden, doch ansonsten wird eine ausschließlich Firmen-bezogene Nutzung aller Ressourcen gewünscht.

Die *Kunden* der Bingo! GmbH haben ein Interesse an den sie betreffenden Daten, die auf dem System gespeichert werden. Dies ist das einzige Interesse dieser Partei, es ist für sie aber umso wichtiger.

Wie wird das System in der Praxis eingesetzt?

Email: Alle Benutzer verwenden auf ihren Arbeits-PCs Email-Clients, welche die abgehenden Emails per SMTP an den Mailserver schicken. Der Mailserver kümmert sich dann um die Zustellung. Eingehende Emails werden mittels POP3 vom Mailserver abgeholt. Das dafür benötigte Passwort wird in der Konfiguration des Email-Clients gespeichert.

Informationssystem: Die Einstiegs-URL zu den Intranet-Webseiten ist allen Mitarbeitern bekannt gegeben. Von dort kann man sich über Links zu den veröffentlichten Dokumenten durchklicken. Änderungen an den Webseiten kann nur ein Administrator vornehmen. Diese bekommen meist per Email den Hinweis vom Management oder anderen Benutzern, bestimmte Dokumente zu veröffentlichen und zu verlinken. Wie bei Wikis üblich, kann dort jedoch jeder den Inhalt ändern. Es wird lediglich die IP-Adresse des Hosts, von wo die Änderung gemacht wurde, in der Datenbank abgespeichert.

Dateiverwaltung: Die meisten Verzeichnisse sind frei für alle les- und beschreibbar, nur einige wenige sind nur dem Management vorbehalten. Durch deren Sonderrechte sind sie aber auch den Administratoren zugänglich.

Von welchen Systemen wird das System benutzt? Mit welchen Systemen wird das System gekoppelt?

Der Email-Server und der Fileserver sind beide mit einem System zur Benutzerverwaltung (welches hier nicht Teil der allgemeinen Betrachtungen gemacht wurde) gekoppelt. Dies wurde gemacht, damit zur Authentifizierung per POP3 und zum Zugriff auf die Dateiverwaltung der gleiche Benutzername und das gleiche Passwort benutzt werden

können.

Auf dem Informationssystem ist lediglich das Wiki auf dem Webserver ist mit dem RDBMS gekoppelt.

6.3 Sicherheitstechnische Aspekte

Im Folgenden wird detailliert auf die sicherheitstechnischen Aspekte der oben beschriebenen Dienste eingegangen. In der Literatur finden sich fünf verschiedene Grundbedrohungen, die auf ein System der Informationstechnik einwirken können - der Verlust von ...:

- Vertraulichkeit
- Integrität
- Verfügbarkeit
- Authentizität
- Verbindlichkeit

Neben diesen Bedrohungen kann noch weiter unterschieden werden in *natürliche Bedrohungen*[5], *passive Angriffe*[6], *aktive Angriffe*[7] und *zufällige Verfälschungen*[8]. In dieser Arbeit werden jedoch nur rudimentäre theoretischen Grundlagen gelegt. Der interessierte Leser sei zum Selbststudium auf die angegebenen Quellen verwiesen.

6.3.1 Vorgesehene Bedrohungen und Sicherheitsmechanismen

Vorgesehene - im Sinne von unvermeidbare - Bedrohungen sowie Sicherheitsmechanismen existieren in jedem Protokoll, wenn man sich Umgebungen, in denen sie laufen, nur abstrakt genug vorstellt. Man muss Systeme (insb. Protokolle und damit Dienste) aus ihrer historischen Bedeutung betrachten und bedenken, dass beim Design eines Protokolls immer nur ein spezieller Dienst erfüllt werden soll. Somit ist es kein Wunder, dass manche Dienste erst durch zusätzliche Maßnahmen z.B. abgesichert werden müssen[9].

Vertraulichkeit: lediglich berechtigte Personen dürfen auf bestimmte Informationen bzw. Daten oder Systeme zugreifen dürfen.

Integrität: Zustand eines Systems, das das unbefugte oder unabsichtliche Verändern von Daten und Programmen nicht zulässt. Alle sicherheitsrelevanten Objekte sollen vollständig unverfälscht und korrekt sein.

Verfügbarkeit: dem Benutzer stehen sowohl Daten als auch das ganze System zu einem bestimmten Zeitpunkt in ihrer vollen Funktionalität zur Verfügung.

[5] Brand, Elektrostatik, ...
[6] Abhören von Daten, Verkehrsflussanalyse, ...
[7] Modifikation von Daten, DoS, Hijacking, ...
[8] Übertragungsfehler, Softwarefehler, ...
[9] Standard E-Mail-Protokolle sind nur ein Beispiel für eine nachträglich eingeführte Sicherungsschicht (z.B. SSL/TLS)

Authentizität: es muss verifizierbar sein, ob die behauptete Identität mit der tatsächlichen Identität übereinstimmt. Die Echtheit oder Glaubwürdigkeit von Kommunikationspartnern und Daten soll dabei nachvollziehbar sein.

Verbindlichkeit: verbindet den Schutz der Urheberschaft mit der Integrität einer Information. Es muss ein eindeutiger Nachweis zu einer Verbindung oder Transaktion möglich sein.

In den nächsten Abschnitten werden diese Anforderungen an IP-Dienste immer wieder anhand konkreter Beispiele dargestellt und näher beleuchtet.

Konstruktiv bzw. implementierungstechnisch bedingte Bedrohungen

Beim Thema IP-Dienste liegt die maßgebliche Bedrohung darin, dass das IP-Protokoll[10] eigentlich keine der o.g. Anforderungen an ein System umsetzt und sich somit Protokolle, die darauf aufsetzen, um jegliche Fragen der Sicherheit kümmern müssen.

E-Mail: Sowohl POP3 als auch SMTP sind Protokolle, die sämtliche Daten unverschlüsselt übertragen[11]. Lediglich POP3 sieht eine Benutzerauthentifikation gegenüber dem System (also seiner Mailbox) vor. Der SMTP Dienst nimmt im Normalfall E-Mails jedes Benutzers entgegen und verschickt sie. Dabei muss keine Authentifizierung erfolgen[12]
Eine weitere Bedrohung für fehlerhafte E-Mailclients sind Viren, diverser mitgeschickter Code, der beim Öffnen einer Mail ausgeführt wird oder auch durch Spam verursachte Schäden[13]

Informationssystem: Da das hier behandelte Informationssystem nur im internen Netzwerk verfügbar ist[14], besteht „nur“ die Möglichkeit Daten und Informationen von innerhalb des Netzwerks der Firma Bingo! zu manipulieren, etc. Falls über eine interne Internetseite Mehrdienste angeboten werden, können andere Mitarbeiten z.B. komplette Verbindungen abhören und dadurch Accounts übernehmen.

Dateiverwaltung: Die Protokolle NFS und SMB verschlüsseln standardmäßig genauso wenig wie IPv4. Also kann ein motivierter Angreifer aus dem Intranet eine Menge Schaden anrichten. Die Daten sind also nicht vertrauenswürdig.

Implementierte Schutzmaßnahmen

E-Mail: Um an die eigene Mailbox zu gelangen, muss sich der Benutzer vorher gegenüber dem Mailserver mittels Login/Passwort authentifizieren. Virenscanner und Spamfilter kommen nicht zum Einsatz.

Informationssystem: Nur autorisierte Benutzer dürften die Daten der internen Internetpräsenz verändern. Dazu zählen nicht nur HTML-Seiten und PHP-Skripte, sondern auch Zugriffe auf die MySQL Datenbank.

[10] IPv4
[11] Vgl. telnet-Protokoll
[12] Die meisten Systeme nehmen dabei nur E-Mails entweder ihres Netzwerks an oder setzen ein Authentifizierungsprotokoll ein.
[13] Vorstellbar sind jegliche bösartige Programme, die über das beliebte Einfallstor „fehlerhafte E-Mailclients“ den Rechner, das Netzwerk oder einzelne Dienste im weitesten Sinne stören.
[14] Die Firewall sperrt Zugänge von Außen

Dateiverwaltung: NFS sichert sich durch unbefugten Zugriff dadurch ab, dass Benutzer nur an Daten gelangen können, wenn sie sich auf ihrem lokalen System authentifizieren können. Es ist also genau wie beim SMB Protokoll ein Loginname und Passwort erforderlich. Wenn nicht explizit, darf auch kein Benutzer die Daten eines anderen lesen, verändern oder gar löschen.

Eignung der Sicherheitsmechanismen

E-Mail: Die Sicherheitsmechanismen - gemessen an den Anforderungen - könnten durch einen vertretbaren Aufwand erhöht werden. Aufgrund des Einsatzes von Windowsrechnern sollte auf dem Mailserver in jedem Fall ein Virenscanner zum Einsatz kommen. Spamfilter können durch die z.T. extrem hohe Last, die sie erzeugen, auch auf den Clientrechnern installiert werden.
Der Mailserver kann dazu auch um die Funktion erweitert werden, SSL-basierte Verbindungen zu akzeptieren. Innerhalb eines Firmennetzwerks können die Zertifikate auch selbstsigniert sein, ohne dass gravierende Sicherheitsprobleme entstehen sollten[15]

Informationssystem: Genau wie beim Mailserver kann auch beim internen Webserver eine SSL-Verbindung angeboten werden. Dies hindert neugierige Personen mit technischen Mitteln daran, fremde Daten zu erspähen.

Dateiverwaltung: Die Dateiverwaltung ist in dieser Firma auf Verfügbarkeit ausgelegt. Das sollte aber nicht zwangsweise Bedeuten, dass andere sicherheitstechnische Aspekte vernachlässigbar sind. Somit kann ebenfalls darüber nachgedacht werden, ob nicht ein System benutzt werden sollte, dass mehr Sicherheit in dem Netzwerk bietet. Leider führen alle Systeme, die Sicherheit für Verbindungen jeglicher Art bieten, einen erheblichen Overhead an Netzlast und administrativen Aufwand mit sich. Daher müsste anhand einer Kosten-Nutzen Rechnung bestimmt werden, wo sich Investitionen lohnen und wo sie kaum Vorteile bringen.

6.3.2 Einsatzbedingte Gefährdungen

Werte und Schutzbedürftigkeit des Systems

Da die Firma im Bereich der Softwareentwicklung tätig ist, sind die Informationen und Daten in Form von Software ihr schützenswertestes Gut. Betrachtet man die einzelnen Dienste, so ergeben sich folgende Gewichtungen:

E-Mail: Für die interne und externe Kommunikation ist die E-Mail mittlerweile als wichtiger einzustufen als das Telefon. Dieser Dienst ist für den alltäglichen Betrieb unerlässlich. Nicht nur Kundenanfragen werden darüber beantwortet, auch Entwickler klären Programmierdetails über interne Mailinglisten oder der Vorstand verschickt aktuelle Informationen an die Angestellten bequem über dieses Medium.

Informationssystem: Die zum internen Informationssystem gehörenden Dienste werden genau wie die E-Mail ständig benötigt. Ausfälle für kurze Zeit können zwar verschmerzt werden, aber stundenweise Störungen im Betrieb sind nicht akzepta-

[15] Bevor ein solches Zertifikat dauerhaft vom Client akzeptiert wird, ist es vertretbar, eine verantwortliche Person nach dem Fingerprint des Schlüssels zu fragen.

bel. Zu Wartungszwecken kann provisorisch immer ein Ersatzsystem die Aufgaben übernehmen.

Dateiverwaltung: Ohne die Benutzerdaten, welche auf dem Fileserver gespeichert sind, kann quasi nichts mehr entwickelt werden. Die Produktion steht im wahrsten Sinne des Wortes still. Ein Ausfall kommt einem GAU gleich.

Typische Gefährdungen

E-Mail: Typische Gefährdungen bei E-Mail-Diensten sind auch hier gegeben. Viren, Trojaner, Spam, etc. kann den normalen Betrieb stark einschränken. Außerdem liegt ein entscheidender Schwachpunkt in der Konfiguration des SMTP-Servers. Mails, die aus dem Intranet verschickt werden, sollen idealerweise angenommen und von Server verschickt werden - wenn gewollt, auch mit geändertem Absender[16]. Falls dabei in der Konfiguration oder durch Bugs Fehler auftauchen, und es passiert, dass der Mailserver auch von Außen als offenes Relay erreichbar ist, kann es zu erheblichen Problemen für die Kommunikation mit diesem Medium für die Firma Bingo! bedeuten[17]. Weiterhin können Mitarbeiter Mails mit gefälschtem Absender versenden und damit auch Schaden anrichten.

Informationssystem: Typische Gefährdungen entstehen hier meistens durch schlecht gewartete Systeme oder durch fehlerhafte Benutzung. Dazu sollte einerseits der Administrator immer aktuelle Patches einspielen und zum anderen auch nur entsprechend geschulte Mitarbeiter direkt an dem System arbeiten[18].

Dateiverwaltung: Hier kann es ein Problem mit NFS geben. Wenn es Benutzern möglich ist, auf einem System[19] Rootrechte zu erlangen, können sie auch Root im NFS werden, also auf dem Fileserver. Damit stünden dem Angreifer uneingeschränkte Rechte zur Verfügung. Dies lässt sich zwar durch spezielle Optionen in der NFS-Server-Konfiguration vermeiden, wird aber oft nicht bedacht.

Schutzmaßnahmen in Reaktion auf typische Gefährdungen

Die Firma Bingo! schützt sich nach Außen hin mit einer Firewall gegen Gefahren aus dem Internet ab. Diese gängige Maßnahme täuscht jedoch oft darüber hinweg, dass die Mehrzahl der (erfolgreichen) Angriffe auf ein System einer Firma aus dem Netzwerk selbst erfolgen.

Möglichkeiten das Netzwerk auch vor Internen stärker zu sichern, kann in erster Linie auch durch passive Maßnahmen erfolgen[20].

Seiteneffekte durch zusätzliche Absicherung

Eine zusätzliche Absicherung von Systemen jeder Art ist immer mit einem höheren Aufwand bei der Wartung und oft auch bei der Benutzung verbunden. Es reicht nicht aus,

[16] Der Mailserver fungiert als offenes sog. offenes Relay für alle Rechner innerhalb des Netzwerks.
[17] Sog. Blacklists registrieren offene Relays, auf die wiederum Spamfilter reagieren und Mail von den entsprechenden Servern ablehnen.
[18] Beispielsweise könnte man durch fehlerhaft geschriebene PHP-Skripte nicht nur eine DoS-Attacke auf den Web-Server starten, sondern gleichzeitig den Datenbank-Server unter Last setzen.
[19] Eigenes Notebook, etc.
[20] IDS, Netzwerkanalyse-Tools, Logfiles, etc.

eine Firewall einmal aufzubauen und zu denken, das Netz sei vor Angriffen von Außen sicher. Es gab/gibt genug Möglichkeiten Technik zu überwinden.

Also muss für eine höhere Sicherheit der einzelnen Dienste auch die erhöhte Komplexität und die geringere Transparenz in Kauf genommen werden. Im Endeffekt bedeutet es auch erhöhte Kosten für die Firma Bingo!.

6.4 Komplexe sicherheitstechnische Aspekte

6.4.1 Illegitime Vorgänge - aus Systemsicht legitim?

Als Beispiel kann hier die Designschwäche von NFS herangezogen werden. Sobald sich ein Benutzer lokal authentifiziert hat, ist er gleichzeitig im Netzwerk als der *gleiche* Benutzer authentifiziert[21]. Obwohl die Policy der Firma Bingo! davon weiß, sieht sie z.Z. keine Alternativen zum vorhandenen Konzept.

6.4.2 Bedrohungen aufgrund von System-/Funktionsinteraktion

Hier können als Beispiel diverse PHP+MySQL Implementierungen herangezogen werden. Seit etwa einem Jahr sind Cross Site Scripting[22] und SQL-Injection beliebte Angriffsmöglichkeiten entsprechend motivierter Internetbenutzer. Um solchen (und anderen) Fehlfunktionen von Soft- und Hardware aus dem Weg zu gehen, ist es unabkömmlich die Mitarbeiter zu schulen und auf Gefahren hinzuweisen.

6.4.3 Schutzmechanismen für Systeminteraktionen

Hier gilt es die Software und Hardware, die man einsetzt, grundsätzlich auf ihre korrekte Funktion hin zu überprüfen oder überprüfen zu lassen. Dazu gibt es jede Menge Programme[23], die der Administrator der Firma Bingo! kennt und auch benutzt. Schulungen der Mitarbeiter ist natürlich auch in diesem Bereich unerlässlich und sorgt unangenehmen Überraschungen vor.

6.4.4 Schutz verschiedener Parteien

Grundsätzlich besteht kaum ein Schutz aller Parteien vor der Gruppe der Administratoren. Aufgrund ihrer weitreichenden Rechte, haben sie Zugriff auf alle Daten und Ressourcen. Hier kann man sich nur mittels Einsatz von Verschlüsselung schützen. Das Management kann zusätzlich mittels entsprechender Arbeitsverträge das Verhältnis rechtlich absichern.

Ebenso können die Kunden von Bingo! mittels Verschlüsselung sicherstellen, dass vertrauliche Daten auch vertraulich bleiben.

Vor illegitimen Zugriffen durch die Mitarbeiter, kann sich das Management und die Administratoren mittels eingeschränkter Rechtevergabe schützen.

[21] Ein eigenes Notebook im Firmennetzwerk reicht.
[22] auch bekannt als XSS
[23] beliebtes Tool zum Überprüfen von Netzwerken ist z.B. Nessus

6.5 Zusammenfassung

Zusammenfassend kann man sagen, dass es sehr schwierig sein kann, schon einfachste Infrastrukturen von IP-Diensten adäquat abzusichern. Fällt einem die Formulierung einer normalen Security-Policy schon schwer, so wird es mit dem Total IT-Security Ansatz nicht eben einfacher. Doch nur mittels einer vollständigen Betrachtung aller betroffenen Elemente und ihrer Schnittstellen, lässt sich ein wirklich umfassender Schutz von IP-Diensten erzielen.

Literaturverzeichnis

[1] Christopher R. Hertel. *Implementing CIFS*. Prentice-Hall, http://ubiqx.org/cifs/SMB.html, 2003.

[2] Philippe Maurer. *Bedrohungen und Risiken im Internet*. Diplomarbeit Universität Zürich, http://www.daph.com/docs/DIPARB98.pdf, 1998.

[3] Norbert Pohlmann. *Firewall-Systeme*. MITP, 2000.

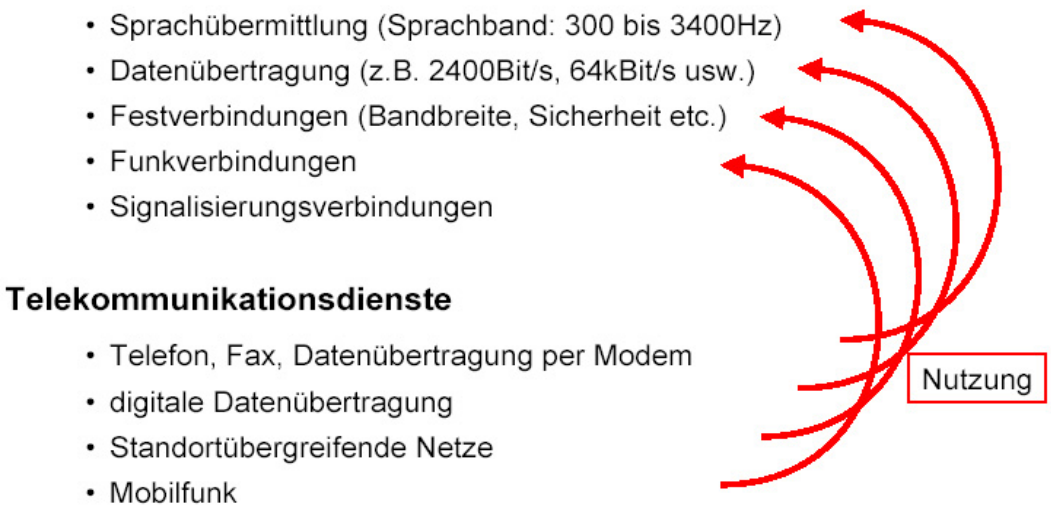
[4] Dr. H. Reiser Prof. Dr. H.-G. Hegering. *IT-Sicherheit - Sicherheit vernetzter Systeme*. Skript zur Vorlesung TU Münschen, http://wwwhegering.informatik.tu-muenchen.de/Vorlesungen/ws0304/itsec.shtml, 2003.

7 Telekommunikationsdienste

R. Vollandt

7.1 Einleitung

Telekommunikationsdienste sind standardisierte Telekommunikationsmöglichkeiten, die ein Anbieter zur Verfügung stellt. Hierbei handelt es sich um hauptsächlich um Dienstleistungen, die auf dem Markt gewinnbringend verkauft werden sollen. Telekommunikationsdienste haben einen außerordentlich hohen Stellenwert von 80% des westeuropäischen Telekommunikationsmarktvolumens (die anderen 20% setzen sich aus Vermarktung von Telekommunikationsnetzen und Netzinfrastrukturen zusammen). Anwender (Unternehmen, Haushalte, Organisationen) kommen somit immer mehr in den Genuss von den nutzbringenden Einsatzmöglichkeiten von Telekommunikationsdiensten. Die nachfolgende Darstellung zeigt die Nutzungsmöglichkeiten von den bekanntesten Telekommunikationsdiensten und deren Bezug zu den Bearer Services (Transportdienste).



Zu den wichtigsten Telekommunikationsdiensten gehört das Telefon und Telefax. Knapp die Hälfte der Betriebe gab bei einer Umfrage an, dass für den Geschäftsverkehr die Nutzung von Telefon und Telefax allein ausreicht. Weiterhin sind Mobiltelefone trotz erhöhter Kosten das am häufigsten genutzte Überbrückungsinstrument bis zum Anschluss an das Festnetz.

7.2 Telekommunikation: Netze und Dienste

7.2.1 Historischer Überblick

Noch vor über hundert Jahren hatte die Telekommunikationsbranche keine größere Bedeutung für die Wirtschaft. Sicherheitsaspekte spielten damals nur aus politischer Sicht (z.B. im Krieg) eine wesentliche Rolle, während diese Aspekte aus heutiger Sicht für den Anwender nicht mehr wegzudenken sind.
Das erste elektrische Nachrichtensystem war ein Netz bestehend aus Telegrafenleitungen. Dieses kam noch gänzlich ohne eigentliche Vermittlungstechnik aus, da sich seine Struktur an den zuvor bestehenden nicht elektrischen (z.B. optischen) Nachrichtensystemen orientierte. Die Leitungen wurden durch einzelne Relaisstationen miteinander verbunden. Als nächstes System entstanden, nach der Erfindung und Verbreitung des Telefons durch Alexander Graham Bell, die Fernsprechnetze. Das Prinzip beruht darauf, dass es verschiedene Möglichkeiten gibt Nachrichten elektronisch zu übertragen. Das Fernsprechnetz wird seit den 1930er Jahren auch für die Übertragung von Bildern benutzt. Letztendlich hat sich das eigenständige Telefonnetz (und nicht das wohl billiger gewordene Fernsprechnetz) durchgesetzt. Grund dafür war, dass bei der Telegrafie die Informationen in Form von digitaler Gleichstromsignalen übertragen werden, beim Fernsprechen hingegen die Sprache in Form analoger Wechselstromsignale. Bei der Nutzung von Telegrafenleitungen konnte man die kontinuierlich analogen Signale nicht verstärken und somit kam eine Nutzung für die Telegrafenleitungen für den Telefondienst nicht in Frage.

7.2.2 Situation der Branche

Die gegenwärtige Situation der Telekommunikationsbranche wird durch die dringende Notwendigkeit zu sparen geprägt. Ganz besonders die Telekommunikationsanbieter stehen in einem gewissen Dilemma, denn zum einen steht da die ungünstige wirtschaftliche Entwicklung in Verbindung mit Verlusten im herrschenden Verdrängungswettbewerb, die Investitionen dringend notwendig machen würde. Zum anderen haben diese Anbieter bereits hohe Investitionsbelastungen wie z. B. die Kosten von UMTS Lizenzen.
Die Telekommunikationsbranche befindet sich in einem ständigen Prozess der Orientierung und Strukturierung (z.B. wandeln neue Technologien die Sprachübermittlung in einen digitalen paketorientierten Datentransport um). Außerdem wird eine homogene Netzstruktur durch heterogene Technik und Komplexität ersetzt. Im Zuge der Liberalisierung der Telekommunikation fällt die zentrale Steuerung und Systemverwaltung der Telekommunikationsinfrastruktur weg und das Marktgeschehen wird von vielen großen und kleinen Unternehmen gestaltet. Immer mehr werden eigenständige Geschäftskonzepte einzelner Konzerne mit dem Ziel der Unabhängigkeit von anderen in Angriff genommen, später Kooperationen von Konzerngruppen organisiert, um Teilmärkte einheitlich zu bedienen. Dabei kommt es zum Splitting der Funktionsbereiche Mobilfunk, Festnetz, Internet und IT Systeme. Die Telekommunikationsanbieter konzentrieren sich immer mehr auf eine Kernaufgabe und versuchen es nicht mehr möglichst viele Bereiche zu bedienen. Stattdessen werden sogar weitere Aufgabenfelder ganz bewusst „ausgelagert“.

7.2.3 Telekommunikationsnetze

Die Entwicklung der Telekommunikationsdienste nahm in den letzten Jahren rapide zu. Zu der „alten Welt“ der Telekommunikationsnetze und -dienste gehören das Fern-

sprechnetz und Datennetze. Zu den Fernsprechnetzen zählt man an erster Stelle die Telephonie, direkt gefolgt von dem Telefax und dem analogen Onlinezugang mit z.B. einem Modem. Datennetze werden durch das Telex (kontinuierlicher Abbau) mit heutzutage immer mehr Schnittstellen zu anderen Netzen, die leistungsvermittelte Datenübertragung Datex-L (wurde bereits eingestellt), sowie die paketvermittelte Datenübertragung (Datex-P) und das Standleitungsnetz mit Datendirektverbindungen gekennzeichnet.
Zu den heutzutage üblichen standortbezogenen Telekommunikationsnetzen gehört das schmalbandige ISDN, Datex-P, Datex-M, Framelink, T-ATM, T-InterConnect und T-DSL. Nachfolgend gehe auf diese Netze ein und ich werde die entsprechenden Anwendungsbereiche mit ihren unterstützen Diensten vorstellen.
Das Integrated Services Digital Network (ISDN) ist ein universelles digitales Telekommunikationsnetz über das alle Dienste abgewickelt werden können. Nebst den bekannten Diensten wie z.B. Telefon und Telefax sind auch neue Dienste wie z.B. Bildtelefonie, Telefon- und Videokonferenzen möglich. Der Normalanschluss besteht aus zwei Kanälen zu je 64 kbps und einem Steuerkanal. Als Mehrwertendgeräte sind besonders PCs mit ISDN-Karte typisch. Das Datex-P ist ein weltweit verfügbares vermittelndes Netz mit einem hohen Sicherheitsstandard. Die Bandbreite liegt zwischen 9,6 kbit/sek. bis 1,92 Mbit/sek. und die Zwischenspeicherung der Informationen erfolgt auf Vermittlungsknoten. Datex-P findet besonders Anwendung bei der gelegentlichen Übertragung kleiner oder mittlerer Datenmengen wie beispielsweise für E-Cash und Sicherheitsanwendungen. Framelink ist ein vermittelndes Datennetz mit einer Bandbreite von 2 Mbit/sek. mit einer festen virtuellen Verbindung, also quasi einer Standleitung. Framelink bietet die Möglichkeit zur Sprach-/Datenintegration und die Deutsche Telekom garantiert sogar für das FrameLink Plus-Netz eine Verfügbarkeit von 99,99T-ATM beruht auf der ATM-(Asynchronous Transfer Mode-)Technologie und ist ein vermittelndes Netz. Die Anschlussgeschwindigkeiten liegen bei 2 Mbit/s bis zu 155 Mbit und der Anwender braucht dafür nur eine trafficunabhängige monatliche Grundgebühr bezahlen. Mit T-InterConnect sind symmetrische Internet-Anbindungen von 64 kbit/s bis zu 155 Mbit/s bei einem direkten Zugang oder per DSL möglich. DSL (Digital Subsciber Line) bietet eine Anschlussleitung für breitbandige, digitale Datenübertragungen über das herkömmliche Kupferkabel-Anschlussnetz. Bei einem DSL-Standardanschluss sind beispielsweise 768 kbit/s downstream und 128 kbit/s upstream möglich.
Festnetzverbindungen sind fest geschaltete Verbindungen und stellen einen eigenen Bereich dar. Hierbei unterteilt man in drei Bereiche, um eine Kommunikation über z.B. Telefon zu gewährleisten. Das City-Netz ist eine Hochgeschwindigkeitsplattform mit Übertragungsleistungen von 2, 34 und 155 Mbit/s. Festnetzverbindungen sind Standard-Festverbindungen und Datendirektverbindungen mit bis zu 155 Mbit/s und optischen Schnittstellen. Exklusive Übertragungswege stellen im letzten dritten Bereich die internationalen Mietleitungen dar, die Übertragungsgeschwindigkeiten bis zu 155 Mbit/s ermöglichen.

7.3 Absicherung von Telekommunikationsdiensten

7.3.1 Einführung

Die Absicherung von Telekommunikationsdiensten gewinnt heutzutage immer mehr an Bedeutung. Die Anforderungen an IT-Systeme mit Bezug zur Netzsicherheit umfassen dabei die Vertraulichkeit, Verfügbarkeit, Integrität, Zurechenbarkeit und Rechtsverbindlichkeit. Diese Festlegung auf diese fünf Basisanforderungen wird in der Fachwelt jedoch ständig kontrovers diskutiert, da man sich bis heute nicht auf eine eindeutige Definition geeinigt hat.

7.3.2 Netzsicherheit

Im nachfolgenden Abschnitt gehe ich auf die Netzsicherheit ein und ich stelle die Sichtweisen der dualen Sicherheit und der mehrseitigen Sicherheit vor. Diese Sichtweisen orientieren sich an den Bedürfnissen aller an einem Dienst Beteiligten. Sie bieten deshalb eine sinnvolle Perspektive im Hinblick auf eine Informationsgesellschaft, bei der die Kommunikationsformen (Briefe, direkte Kommunikation etc.) durch automatisierte IT-Systeme basierende Kommunikationsformen immer mehr abgelöst werden.

Duale Sicherheit

Zu der dualen Sicherheit gehört zum einen das maschinelle IT System mit seinen Randbedingungen sowie die Benutzer, die mit ihren Unzulänglichkeiten in die Sicherheitsbetrachtungen miteinbezogen werden. Die „duale“ Sichtweise dient dabei hauptsächlich als Brücke zwischen den menschlichen Benutzern (und deren Sicherheitssicht) und der innerhalb der technischen Kommunikationsdienstumgebung diskutierten Sicherheitsdienste.
Nachfolgend gehe ich auf die Forderungen an einen Kommunikationsdienst und die zugrunde liegende Kommunikationsinfrastruktur ein, die sich im Bezug auf den Umgang von Menschen mit diesen Diensten und Systemen definieren lassen.
Zurechenbarkeit (Accountability)
Darunter steht man die eindeutige Zuordnung von Ereignissen bei der Benutzung eines Kommunikationsdienstes zu dem jeweiligen Verantwortungsträger (Initiator).
Rechtsverbindlichkeit (Legal Liability)
Für die Nutzung von Kommunikationsdiensten und den aus der Nutzung resultierenden Ergebnissen müssen die verantwortlichen Instanzen gegenüber Dritten beweiskräftig nachweisbar sein.
Verlässlichkeit
Diese Kategorie beschreibt die Sicherheit, die das System während der Diensterbringung bietet. Ein IT System gilt als verlässlich, wenn weder die Daten noch die Datenverarbeitung in ihrem Bestand, ihrer Nutzung oder ihrer Verfügbarkeit beeinträchtigt werden. An die Kommunikationsdienste wird dabei zum einen die Forderung der *Vertraulichkeit* gestellt. Informationen sollen also vor unautorisierter Kenntnisnahme geschützt werden. Außerdem soll eine *Integrität* Schutz vor unautorisierter und unerkannter Veränderung von Informationsdaten bieten. Die *Verfügbarkeit* soll gewährleisten, dass die Kommunikationsdienste oder damit in Verbindung stehende Netzfunktionen für den autorisierten Benutzer bei Bedarf verfügbar sind.
Ein Kommunikationsdienst wird bezüglich der dualen Sicht als sicher bezeichnet, wenn die ausgeführten Aufgaben zurechenbar und rechtsverbindlich gestaltet werden können und der Dienst die an ihn gestellten Leistungsaspekte, wie z. B. geforderte Schutzziele

aus den Bereichen Vertraulichkeit, Verfügbarkeit und Integrität erfüllt. Außerdem muss der Kommunikationsdienst beherrschbar und verlässlich sein.

Mehrseitige Sicherheit

Mehrseitige Sicherheit bedeutet in der Kommunikationstechnik die Einbeziehung der Sicherheitsanforderungen aller Beteiligten sowie das faire Austragen daraus resultierender Schutzkonflikte beim Benutzen von Kommunikationsdiensten. Bei einer mehrseitigen Sicherheit müssen Schutzkonflikte gegensätzlicher Sicherheitsanforderungen von verschiedenen Beteiligten aufgelöst werden. Ziel ist eine Konfliktauflösung durch eine ausgewogene Berücksichtigung der Schutzinteressen aller Beteiligten.
Ein auszuhandelnder Konflikt könnte beispielsweise darin bestehen, dass gängige Verfahren zur Entgeltdatenerfassung und zur Zugriffskontrolle auf der Identität des Benutzers beruhen (z.B. anonym telefonieren). Gelöst könnte dieser Konflikt, indem der Benutzer auf das Kommunikationsnetz durch Barzahlung oder Debitkarte zugreift und somit die Erfassung von Entgeltdaten (z.B. auf einer kompletten Monatsrechnung) wegfällt. Gegenüber dem Anbieter weist sich der Benutzer mit Hilfe eines Pseudonyms aus, das er von einer ihm vertrauten Instanz erhält. Dabei werden die Spannungen zwischen Anonymität und Zugriffskontrolle aufgelöst. Der Dienstanbieter akzeptiert, dass eine nicht bekannte Person auf seine Datenbank zugreift, solange sie beweisen kann, dass die vertraute Instanz sie dazu ermächtigt hat.
Sicherheit dient aber nicht nur den Kommunikationspartnern selbst, sondern auch all jenen, die mit den Partnern oder mit dem jeweiligen Kommunikationsinhalt (keine Weitergabe vertraulicher Daten) in Beziehung stehen oder mit der Bereitstellung der Kommunikationsmittel (sicheren Zugriff gewährleisten) zu tun haben. Ein mehrseitig sicherer Kommunikationsdienst berücksichtigt also die Schutzziele aller Betroffenen in ausgewogener Weise.

Schutzziele

Nachfolgend stelle ich wichtige Schutzziele dar, die im Sinne der mehrseitigen Sicherheit von Kommunikationsdiensten erfüllt und gewährleistet werden müssen. Sie sind das Ergebnis des aktuellen Stands einer fortdauernden Entwicklung.

- Vertraulichkeit
  - Nutzdaten
  - Kommunikationsinhalte
  - Dienstesteuerungsdaten
  - Zustandsdaten der Kommunikationsinfrastruktur
  - Benutzer von Kommunikationsdiensten
- Integrität
  - Fälschungen von Daten
  - Fälschungen von Kommunikationsdiensten
  - Identitäten
  - Kommunikationsinfrastruktur

  - Kommunikationsdienste
- Verfügbarkeit
  - Kommunikationsdienste
  - Kommunikationsnetze
- Zurechenbarkeit
  - Benutzung für Anwender
  - Entgeltbezahlung an Netzbetreiber
- Rechtsverbindlichkeit
  - Kommunikationsumstände
  - Kommunikationsinhalte

7.3.3 Modelle für die Bewertung der Sicherheit

Es gibt insgesamt drei Modelle für eine sicherheitstechnische Bewertung eines Kommunikationsnetzes oder Kommunikationsdienstes aus netztechnischer Sicht (Leistungsfähigkeit, Dienstqualität, Ressourcenbelegung, etc.). Diese Modelle gelten als Grundlage für eine Bewertung der Sicherheit.

Angreifermodell

Ein Angreifermodell beschreibt die Attribute angenommener Angreifer die bei der Sicherung eines Systems berücksichtigt werden müssen. So versucht sich ein Angreifer durch eine unautorisierte Handlung Zugriff zu verschaffen. Die Grundlage für die Sicherheitsbewertung eines Angreifermodells besteht aus folgenden wesentlichen Charakteristiken: Ein wichtiger Punkt ist die Motivation des Angreifers. So hat er eine Gewinnerwartung durch seinen Angriff, die im Erfolgsfall in einem für ihn realistischen Aufwand-Nutzen-Verhältnis stehen muss. Viele Angreifer streben nach einer erhöhten Selbstdarstellung und wollen gegebenenfalls sogar Rachegelüste gegen Institutionen befriedigen. Auch spielt die generelle Befriedigung des Spieltriebs eines Angreifers eine wichtige Rolle. Das Systemwissen eines Angreifers entscheidet letztendlich, ob ein realistischer Erfolg versprechender Angriff möglich ist. Systemwissen kann beispielsweise durch praktische Erfahrungen oder Fachliteratur erworben werden.
Die Durchführbarkeit eines Erfolg versprechenden Angriffs hängt u. a. von folgenden Kriterien ab:

- Zugangsmöglichkeiten des Angreifers zum angegriffenen System
- Verfügbarkeit technischer Geräte und Erfahrungen mit deren Bedienung
- Zeit zur Vorbereitung und Durchführung von Angriffen
- Verfügbarkeit finanzieller Mittel

Ziel ist es effizient gegen die Angreifer vorzugehen, indem man sich in einen Angreifer und deren Motivation versetzt, um frühzeitig Lücken im System zu schließen. Insiderwissen darf gar nicht erst zur Verfügung gestellt werden und die Sicherheitsarchitektur[1] muss flexibel genug sein, um den höchsten Sicherheitsstandard zu gewährleisten.

[1] Unter einer Sicherheitsarchitektur versteht man jene Systembestandteile (Komponenten und Beziehungen zwischen Komponenten), die Einfluss auf die Sicherheit eines Systems haben.

Benutzermodell

Vertraut ein Benutzer einem technischen Gerät, so zeigt er damit, dass er von einer erwartungsgemäßen Funktion ausgeht, obwohl er gar nicht genügend Informationen besitzt, um dieses nachvollziehbar und sicher vorhersagen zu können. Innerhalb des Vertrauensbereichs des Benutzers werden keine Angreifer oder mögliche Angriffe auf das System oder technische Gerät angenommen. Somit fühlt sich ein Benutzer in argwöhnischer Sicherheit, obwohl generell kein technisches System gegen allmächtige Angreifer vollständig gesichert werden kann.
Ziel ist es hier die sicherheitstechnischen Prioritäten von Benutzern zu erkennen und besonders in den wesentlichsten Aspekten eine höchstmögliche Sicherheit zu gewährleisten.

Bedrohungsmodell

Eine Bedrohung beschreibt Schwachstellen eines Systems, an denen ein erfolgreicher Angriff auf Schutzziele angesetzt werden könnte. Ein Bedrohungsmodell liefert dabei die Kriterien zur Klassifizierung von Bedrohungen und zur Analyse von Systemen im Hinblick auf bestehende Bedrohungen.
Wichtig ist der Einsatz von Schutzmechanismen, um aktiv gegen mögliche Bedrohungen im System zu wirken. So erschweren Zugangskontroll- und Zugriffskontrollmechanismen den Zugang zum System auf schutzwürdige Systembestandteile. Auch erreicht man mit der Verschlüsselung durch leistungsfähige Algorithmen, dass sich Angriffe auf die Vertraulichkeit verschlüsselter Daten stark verringern. Eine Protokollierung von Systemvorgängen würde außerdem sofort unautorisierte Zugriffe aufdecken.
Ziel ist es hier ein ideales System zu schaffen, bei dem die Sicherheitsaspekte wesentlicher Parameter ersichtlich sind. Auswirkungen von Änderungen am System sind dann im Voraus abschätzbar, so dass gegebenenfalls das durch die Veränderung von Systemparametern entstehende Angriffspotential proaktiv durch entsprechende Schutzmechanismen ausgeglichen werden kann.

7.3.4 Ausblick

In der Zukunft wäre es möglich, dass die für einzelne Kommunikationsnetze verfügbaren Sicherheitsdienste netzübergreifend nutzbar und damit benutzerfreundlicher gemacht werden. Um dies zu gewährleisten, müssten beispielsweise die netzunabhängigen Sicherheitsdienste z.B. Authentisierung von den Netzdiensten getrennt realisiert werden. Netzspezifische Dienste sollen also für alle Betroffenen nachvollziehbar und steuerbar werden. Insgesamt lässt sich feststellen, dass viele von Netzbetreibern und Dienstanbietern angebotene Sicherheitsdienste nicht benutzerkontrollierbar und -steuerbar sind. Insbesondere für den Schutz von Kommunikationsumständen gibt es kaum befriedigende Lösungen. Dieses steht im Widerspruch zu der Tatsache, dass die Dienste für den Benutzer gemacht sind. Es kann heute durchaus der Eindruck entstehen, die Benutzer seien aus Sicherheitssicht dem Netzbetreiber und Dienstanbieter mehr verpflichtet, als dieser ihnen.
Es sollte auch noch einmal erwähnt werden, dass eine große Herausforderung darin besteht die bereits genannten Voraussetzungen zu schaffen. Hierzu gehören nachvollziehbar sichere Ablaufumgebungen und Endgeräte, prüfbare Betriebssysteme und die Zertifizierung kontrollierter Software. Insbesondere die zunehmende Vernetzung (Internet) und die damit einhergehenden Dienste erschweren die Erzielung einer nachhaltig sicheren Laufzeitumgebung. Diese Probleme sind deshalb so schwer zu lösen, weil sie nicht nur von technischen und finanziellen sondern auch von politischen Faktoren abhängig sind.

Absolut sichere komplexe Systeme sind auch zukünftig nicht zu erwarten. Benutzer, Hersteller, Netzbetreiber oder Dienstanbieter werden in ihrer Unvollkommenheit immer als Angreifer bezüglich der (möglicherweise selbst formulierten) Schutzziele betrachtet werden müssen. Dieses ist bei der Verteilung von Aufgaben an automatisierte technische Systeme stets zu bedenken. Möglicherweise ist die Nachvollziehbarkeit mit dem Wissen um die Unvollkommenheiten und die Kontrollierbarkeit der Technik letztendlich wichtiger, als der Versuch der Annäherung absoluter Sicherheit.

7.4 Zusammenfassung

Dienstanbieter sind bestrebt die Sicherheitsfunktionen entsprechend der zu garantierenden Schutzziele mit herkömmlichen Telekommunikationsdienstfunktionen zu verknüpfen und somit sichere Telekommunikationsdienste zu ermöglichen. Eine gute Sicherheitsarchitektur[1] unterstützt dabei die Verlässlichkeit von Telekommunikationsdiensten durch die Kopplung mit Sicherheitsanwendungsdiensten und dem optionalen Zuschalten von Sicherheitsfunktionen. Die flexible und optionale Zuschaltung von Sicherheitsdiensten fördert dabei außerdem die Kontrolle der gegebenen Sicherheit durch den Benutzer.
Durch gezielte Erweiterungen der Kommunikationsnetze können die Effektivität und Effizienz der Sicherheitsarchitektur entscheidend verbessert werden. Dies gilt zum Beispiel für folgende Bereiche:

- Übermittlungsdienste für Sicherheitssteuerungsdaten
- Adressierungsmöglichkeiten für zentrale Server
- Erweiterung des Netzzuganges um Sicherheitsdienstmerkmale

Das Einführen dieser und anderer Dienstmerkmale ist letztendlich eine Sache des Abwägens von Aufwand und Ertrag. Einer technischen Realisierbarkeit unter gleichzeitig garantiertem Investitionsschutz bestehender Netzinfrastruktur steht dabei jedoch nichts im Wege.

Literaturverzeichnis

[1] Fridhelm Bergmann and Hans-Joachim Gerhardt. *Taschenbuch der Telekommunikation.* Hanser Fachbuchverlag, 2003.

[2] Lutz Frühbrodt. *Die Liberalisierung der Telekommunikationsdienste. Vom nationalen Monopol zum globalen Wettbewerb.* Dt. Universitätsv., 2002. ISBN: 3824406241.

[3] Horst Jansen. *Telekommunikation mit ISDN.* Europa-Lehrmittel, 2003.

[4] Niels Klußmann. *Lexikon der Kommunikations- und Informationstechnik.* Hüthig Verlag, 3 edition, 2001.

[5] Geschichtlicher Überblick. http://www.handy-telefon.de.

[6] Linksammlung Telekommunikation. http://www.folden.de/telekommunikationinformationen.html.

[7] Telecommunication Networks Group. http://www-tkn.ee.tu-berlin.de/.

[8] Network Computing. http://www.networkcomputing.de.

8 GSM, SMS, MMS

B.Mehler, T. Tebner

8.1 Einleitung

Dieses Paper beschreibt die Entwicklung, den Aufbau und das Zusammenspiel der Komponenten eines GSM Netzes. Dabei werden die verwendeten Sicherheitsmechanismen ebenso wie vorhandene Sicherheitslücken und Angriffsmöglichkeiten erläutert. In diesem Zusammenhang werden ebenfalls die Dienste SMS und MMS, so wie die Rolle des Providers unter sicherheitstechnischen Aspekten näher untersucht. Mit einer Betrachtung der technischen Möglichkeiten von aktuellen Endgeräten, unter dem Gesichtspunkt "Spionage", schließt dieses Paper ab.

8.2 GSM

8.2.1 Historie

1982 verständigte man sich erstmals auf eine europaweite Zusammenarbeit in der Mobilfunkbranche. Dies hatte die Gründung der Arbeitsgruppe "Global System for Mobile Communication" (kurz GSM) zur Folge, welche mit der Entwicklung eines neuen digitalen Mobilfunkstandards beauftragt wurde. 1991 war es soweit, das GSM-Pilotnetz wurde erfolgreich auf der ITU-Messe in Genf vorgestellt. 1992 wurden bereits die ersten GSM-Netze offiziell freigegeben. Kurz darauf wurden die ersten Roaming-Abkommen zwischen den Netzbetreibern geschlossen. 1993 gingen die ersten GSM Netze außerhalb Europas in Betrieb. 1999 gab es bereits 239 GSM-Netze in 108 Ländern weltweit. Bis zum heutigen Tag wurde GSM durch über 160 Drittländer in Asien, Afrika und Amerika übernommen. Über 500 Millionen Teilnehmer telefonieren in über 400 Netzen weltweit und verschicken über 15 Milliarden SMS pro Monat.

8.2.2 Komponenten & Funktionsweise

Ein GSM Netz besteht aus den folgenden 4 Hauptkomponenten:

Mobilstation (MS)

Eine GSM-Mobilstation besteht aus dem Mobilfunkgerät selbst und dem SIM (Subscriber Identity Module). Das Mobilfunkgerät ist durch seine international eindeutige Seriennummer (IMEI) gekennzeichnet. Der Nutzer wird durch seine, auf der SIM-Karte gespeicherte, Kundennummer (IMSI) identifiziert. Damit wird im GSM-Netz zwischen Nutzer und Gerät unterschieden

Basisstation (BTS)

Eine GSM-Basisstation (BTS) bildet die Sende- und Empfangsstation einer oder mehrerer Funkzellen. Sie stellt die Schnittstelle zwischen Netzbetreiber und der Mobilstation dar. Die Kontrollstation (BSC) verwaltet dabei die Sende- und Empfangsressourcen der angeschlossenen Basisstationen.

Vermittlungsknoten (MSC)

Die Basisstation wird über den Vermittlungsknoten (MSC) gesteuert. Dieser übernimmt technische Funktionen wie Wegsuche, Signalwegschaltung und Dienstmerkmalsbearbeitung. Damit der Netzbetreiber in der Lage ist, alle gewünschten Dienste zu erbringen, speichert er zu diesem Zweck verschiedene Kundendaten (Aufenthaltsort, Nutzerberechtigungen, letzter Status, usw) in den Registern des Vermittlungsknoten (EIR, AUC, HLR, VLR).

Festnetz (GMSC)

Festnetze bilden die Basis für alle GSM Verbindungen (außer für Verbindungen innerhalb der gleichen Funkzelle).

8.2.3 Sicherheitsmechanismen

Das GSM Sicherheitskonzept, basiert im Wesentlichen auf den folgenden 4 Sicherheitsmechanismen:

Teilnehmeridentifizierung

Jeder Mobilfunkteilnehmer besitzt eine international eindeutige Kennung (IMSI), die auf der SIM-Karte gespeichert ist. Beim Einschalten der Mobilstation muss sich der Teilnehmer, der SIM-Karte gegenüber, mit einem PIN Code identifizieren. Nach erfolgreicher Eingabe der PIN verbindet sich die Mobilstation zu einer nahe gelegenen Basisstation.

Authentifizierung

Jeder Verbindungsaufbau und jedes Location-Update erfordert eine Authentifizierung. Dabei wird ein symmetrisches Schlüsselverfahren angewendet. Ein Schlüssel ist dabei im Heimat-Register (HLR) des Vermittlungsknoten (MSC) gespeichert, der andere auf der jeweiligen SIM Karte des Teilnehmers. Der Teilnehmer bekommt eine Zufallszahl von der Authentisierungszentrale (AuC) übermittelt. Beiderseitig wird aus der Zufallszahl und dem A3-Algorithmus nun ein 32 Bit Schlüssel errechnet. Anschließend erfolgt ein Vergleich. Stimmen die errechneten Schlüssel überein, war die Authentifizierung des Nutzers erfolgreich.

Verschlüsselung

Die SIM Karte des Teilnehmers bekommt eine Zufallszahl von der Authentisierungszentrale (AuC) übermittelt. Beiderseitig wird aus der Zufallszahl und dem A8-Algorithmus ein 64-Bit Übertragungsschlüssel (cipher key) errechnet. Danach werden aus den zu übertragenen Daten und dem zuvor berechneten cipher key, mittels des A5-Algorithmus, chiffrierte Datenblöcke erstellt. Diese werden anschließend übermittelt.

Anonymisierung

Nach erfolgreicher Authentifizierung weist das Besucher-Register (VLR) der Mobilstation eine temporäre Kennung zu (TMSI), die zur Identifizierung der Mobilstation an der Luftschnittstelle dient. Die TMSI ist nur innerhalb einer Location-Area gültig, wobei diese aus einer oder mehreren Zellen bestehen kann. Ein regelmäßiges Location-Update prüft, ob die Mobilstation sich noch innerhalb der zuletzt ermittelten Location-Area befindet.

8.2.4 Sicherheitslücken und Angriffsmöglichkeiten

Risiken im Sicherheitskonzept

Das GSM Sicherheitskonzept birgt bereits von Haus aus einige Design-Schwächen. So ist z.B. nur eine einseitige Authentifizierung vorgesehen, d.h. das Netz muss sich nicht gegenüber der Mobilstation authentifizieren, es gilt stets als vertrauenswürdig. Ebenso sind die Algorithmen A3 und A8 nicht standardisiert. Ihrer Implementierung ist nach wie vor geheim. Somit kann keine Überprüfung durch unabhängige Institute stattfinden. Weiterhin sind die verwendeten Übertragungsschlüssel mit einer Länge von 64 Bit nach heutigen Anforderungen zu knapp bemessen, 128 Bit sind mittlerweile Standard. Außerdem findet eine Verschlüsselung und Authentifizierung lediglich an der Luftschnittstelle statt. Außerhalb werden die Daten ohne jegliche Sicherung übertragen.

geklontes Handy - aus Eins mach Zwei

Die Algorithmen A3 und A8 benutzen COMP128 als Referenzimplementierung. Comp128 weißt allerdings eine signifikante Schwachstelle auf, die es ermöglicht, den geheimen Teilnehmerschlüssel aus einer SIM-Karte zu extrahieren. Dazu ist eine bestimmte Kombinationen von Anfragen nötig, die bei gleichen Ergebnissen, Rückschlüsse auf den Schlüssel ermöglichen. Für eine komplette Entschlüsselung werden ca. 150.000 Anfragen getätigt, die die SIM jeweils auffordern sich zu authentifizieren. Ein solcher Angriff dauert bis zu 12 Stunden. Die Konsequenz daraus ist, dass man eine SIM-Karte klonen kann. Dazu wird lediglich der extrahierte Schlüssel und ein SIM-Karten Emulator benötigt. Dann ist es möglich, mit einem beliebigen Handy auf Kosten des eigentlichen SIM-Karten Besitzers zu telefonieren. Dieses Verfahren wurde von "Chaos Computer Club"(CCC) 1998 erfolgreich nachgestellt.

Man-in-the-middle-Attack

Eine Mobilstation meldet sich immer bei der Basisstation mit dem stärksten Signal an. Das kann durch einen IMSI-Catcher ausgenutzt werden, indem dieser eine Basisstation simuliert und ein stärkeres Signal sendet als die wirklichen Basisstationen. Der IMSI-Catcher kann dabei angemeldete Mobilstationen fangen oder abhören. Im Fangmodus kann er die Kundennummer (IMSI) und die Gerätenummer (IMEI) im Klartext abfragen. Im Abhörmodus verhält sich der IMSI-Catcher zusätzlich gegenüber der echten Basisstation wie die gefangene Mobilstation. Durch einen speziellen GSM-Befehl (den der IMSI-Catcher nutzt), kann nun der Netzbetreiber veranlasst werden, die Verschlüsselung abzuschalten. Das hat zur Folge, dass alle abgehenden Gespräche der gefangenen Mobilstation vom IMSI-Catcher im Klartext aufgezeichnet werden können. Die gefangene Mobilstation bekommt davon nichts mit.

Gesprächsentschlüsselung

GSM verwendet für die Gesprächsverschlüsselung den A5-Algorithmus in den Varianten A5/1 und A5/2. Bereits 1999 gelang es Alex Biryukov und Adi Shamir, die A5/1 Variante adäquat zu knacken, indem sie eine große Menge an Zuständen, die der Algorithmus einnehmen kann, bereits im Voraus berechneten. Durch geeignete Verfahren, gelang es außerdem die Komplexität des Schlüssels auf ein akzeptables Maß zu senken. So war es nunmehr möglich, mit einem nur 2 minütigen Datenstrom und vorausberechneten Daten in einer Größe von ca. 150 GB, den A5/1 Algorithmus innerhalb einer Sekunde zu entschlüsseln. Da A5/2 wesentlich schwächer konzipiert ist als A5/1, kann dieser Algorithmus sogar noch schneller entschlüsselt werden, wie eine veröffentliche Kryptoanlayse von Briceno, Goldberg und Wagner ebenfalls 1999 zeigte.

8.3 SMS und MMS

Mit **SMS** (**S**hort **M**essage **S**ervice) können Textnachrichten mit bis zu 160 Zeichen an Mobilfunkteilnehmer in aller Welt versendet werden. Eine **EMS**-Nachricht (**E**nhanced **M**essaging **S**ervice) besteht aus mehreren aneinander gereihten SMS-Nachrichten und bietet die Möglichkeit, animierte Grafiken, (Klingel-) Töne und formatierte Texte zu verschicken. Mit einer **MMS** (**M**ultimedia **M**essage **S**ervice) lassen sich durch Nutzung gesteigerter Mobilfunk-Bandbreiten farbige Bilder und kurze Filmsequenzen auf entsprechend ausgestattete Mobiltelefone übertragen.

8.3.1 Historie

März 1994: Die erste SMS wird erfolgreich über das D1-Netz verschickt.[1] Der Short Message Service war ein überraschender Erfolg in der Mobilkommunikation. Nur Wenige haben diesem schwer zu bedienendem Dienst diesen Erfolg prognostiziert.
Paradoxerweise war gerade die schwierige Benutzung von SMS ein Schlüssel zum Erfolg bei der jungen Generation. Die Tatsache, das die Eintrittsbarriere beim Umgang mit der SMS hoch genug war, so dass Eltern, Lehrer und andere Autoritätspersonen nicht in der Lage oder nicht Willens waren, die Benutzung der SMS zu erlernen, machte diesen Dienst zu einer Möglichkeit sich von eben diesen Personen abzugrenzen.
Mit Einführung der Prepaid-Karten stieg die Zahl der versendeten SMS rapide an, denn die Prepaid-Karten-Telefone eigneten sich bereits für den SMS-Versand, wenngleich diese Funktion bewusst nicht dokumentiert wurde. Die Mobilfunkbetreiber waren Anfangs nicht in der Lage versandte SMS vom Prepaid-Guthaben abzurechnen. Sie benötigten einige Monate um dieses Problem in den Griff zu bekommen, worauf die Menge der versandten SMS auf 25-40% des vorherigen Wertes fiel. Dieser Zustand war jedoch nicht von Dauer und so werden heute Millionen SMS am Tag weltweit versendet.
Am Ende der bisherigen Entwicklung steht die MMS, mit der die Beschränkung auf 160 Zeichen Text entfällt und die das Versenden von Ton- und Bilddateien möglich macht.[2]

8.3.2 Technische Realisierung

SMS

SMS Nachrichten werden von und zur Mobilstation paketorientiert und ohne gesonderte Belegung eines Verkehrskanals übertragen, indem freie Kapazitäten in den Signalisierungskanälen der GSM-Protokollarchitektur ("SignalisierungsSystem Nr. 7") genutzt

werden. Die SMS werden über ein "**S**hort **M**essage **S**ervice **C**enter" (**SMSC**) weitergeleitet. Zusätzlich besitzt das SMSC meistens auch eine TCP/IP-Verbindung für den direkten SMS-Versand aus dem Internet und eine Anbindung über Standleitung für Firmen mit einem hohen SMS-Aufkommen. Ein Mobilfunkteilnehmer muss nicht das SMSC seines Heimatnetzbetreibers verwenden. Internationale SMSC-Rufnummern sind im Internet zu finden.

EMS

EMS erweitert die Funktionalität dahingehend, dass es nun auch möglich ist Bilder, Ruftöne oder Melodien zwischen verschiedenen Mobiltelefonen auszutauschen. Das wird durch Verketten mehrerer SMS sowie die Verwendung von 8 Bit statt der bisherigen 7 Bit pro Zeichen möglich. Die EMS ist ein Zwischenschritt von der bisherigen SMS zum heutigen "Multimedia Messaging Service" (**MMS**).[3]

MMS

Im Gegensatz zu EMS basiert die MMS-Technik nicht auf dem SMS-Standard, so dass nun auch größere Nachrichten übertragen werden können. In der momentan ersten Phase von MMS ist die Maximalgröße einer MMS-Nachricht auf 100kB begrenzt. Die Gründe dafür sind die noch knappe Speicherkapazität auf aktuellen Mobiltelefonen sowie die GPRS-Übertragungskapazität. Mit UMTS-Mobiltelefonen und -Trägerdiensten werden sich aber zukünftig größere Datenmengen leicht handhaben lassen.
Zentrales Element in der Mobilfunkarchitektur für MMS-Dienste ist das MMS-Center (**MMS-C**), das die Nachrichten verwaltet, bearbeitet und weiterleitet und das für die Gebührungszentrale ein Ticket ausstellt, damit die Kosten für den MMS-Dienst dem Kunden in Rechnung gestellt werden können. Das MMS-C kann analog zum SMSC gesehen werden, nur dass die Funktionalität des MMS-C komplexer geartet ist.
Im MMS-C kann jeder Teilnehmer ein Profil ablegen, in dem er definiert welche Art von Nachrichten er empfangen möchte bzw. empfangen kann. So läßt sich bestimmen, dass MMS-Dienste, die nicht am Mobiltelefon dargestellt werden sollen bzw. können per Internet abgerufen werden. Auch kann das Teilnehmerprofil Filter enthalten, mit denen unerwünschte Nachrichten ausgefiltert werden können. Um dieses Profil selber verwalten zu können, kann der Teilnehmer über ein WAP-Formular die Parameter seines Profils einstellen.

8.3.3 Einsatzgebiete und Anwendungen

Ein Telefonat ist nicht immer angemessen. Doch es gibt einen Ausweg: Jederzeit lassen sich kurze Textnachrichten empfangen und versenden. Das Mobiltelefon signalisiert den Eingang nur dezent und speichert die Nachricht ab. Lesen kann man sie dann, wenn man Zeit dazu hat.[4] Die Zahl der in Deutschland verschickten SMS stieg im Jahr 2003 auf den Rekordwert von 36 Milliarden (31,2 Milliarden im Vorjahr).[5] In der Schweiz wurden 66,4 Millionen SMS über die Neujahrstage verschickt und die Franzosen versandten 42,5 Millionen SMS nur zu Silvester.[6]
Das Haupteinsatzgebiet der Nachrichtendienste ist der private Versand von Textnachrichten zum Informationsaustausch (oder zumindest zum Austausch von Wörtern). Kommerzielle Einsatzgebiete setzen sich jedoch immer mehr durch. So gibt es z.B. das Produkt *mTaxi* der norwegischen Firma Cellvision AS, welches auf "Location based Services" setzt und damit einem Taxiunternehmen erlaubt schnell festzustellen, wo der

Kunde der das Taxi ruft steht (auf 75 bis 150 Meter genau) und welches Taxi in seiner Nähe ist. Die Anwendung kann über SMS, MMS und WAP genutzt werden.[8]
Weitere Anwendungen sind Live-Abstimmungen, SMS-Chat über Videotext sowie Alarmierungs- und Telemetrieanwendungen. Wie bereits auch schon bei der EMS können Netzbetreiber und Dienstleister auch mit der MMS gegen Gebühr Klingelmelodien, Bilder und kleine Videosequenzen auf Abruf zur Verfügung stellen. Wettervorhersagen mit kleinen Wetterkarten sind möglich, Börsendaten mit Kursverläufen, Politik mit Schlagzeilen und Foto oder gar Anfahrbeschreibungen mit Stadtplan und markanten Punkten im Bild.[7]

8.3.4 Sicherheitskonzepte und Schwachstellen

Generell gelten für SMS, EMS und MMS natürlich die Sicherheitskonzepte und Schwachstellen der zu Grunde liegenden Mobilfunktechnologie GSM, welche bereits im Kapitel 8.2.3 "Sicherheitsmechanismen" und im Kapitel 8.2.4 "Sicherheitslücken und Angriffsmöglichkeiten" ausführlich behandelt wurden. Kurznachrichten lassen sich also ebenso "abhören" wie ein über das GSM-Netz geführtes Telefonat. Hierzu kommt die Tatsache, dass Kurznachrichten im Message-Center unverschlüsselt gespeichert werden. D.h. also, wenn sich eine unbefugte Person Zugang zum Message-Center verschafft, ist diese in der Lage, alle Nachrichten im Klartext zu lesen.
Durch das Ausnutzen von Software-Fehlern auf bestimmten Mobiltelefonen ist es Hackern bereits gelungen, diese durch einen per SMS erzeugten "Buffer-Overflow" abstürzen zu lassen. Möglich ist auch, dass Mobiltelefone nach Eingang einer Hacker-SMS nicht mehr löschbare Symbole auf dem Display anzeigen. Solche "Angriffe" per SMS sind in der Regel jedoch ungefährlich, da die meisten Funktionsstörungen leicht rückgängig gemacht werden können. Außerdem können die so genannten "SMS-Viren" im Gegensatz zu "richtigen" Viren sich nicht selbstständig weiterverbreiten. Dieser Zustand könnte sich mit Programm-Downloads über das Mobilfunknetz und einer leistungsstarken Software-Umgebung (Java) auf dem Mobiltelefon durchaus ändern. Die Geräte-Hersteller wollen diesem Sicherheitsrisiko mit Zertifikaten, die den Urheber eines Java-Programms vor der Installation identifizieren, entgegentreten und so potenziellen Virenprogrammierern das Leben schwer machen.[11]
Neben der technischen Seite kann es aber auch Schwachstellen auf der Prozessebene geben. Wie ein aktueller Vorfall zeigt, sollten vertrauliche Nachrichten nicht per SMS verschickt werden. Vor kurzem hat die englische Telekommunikationsfirma MMO2 zwei Mitarbeiter entlassen, da sie einem Kollegen SMS-Nachrichten seiner Ex-Freundin weiterleiteten. Dieser hat die intimen Daten anschließend auf der Website seiner ehemaligen Liebe veröffentlicht. Dieser Vorfall weißt auf ein großes Sicherheitsrisiko bei SMS-Meldungen hin: Die Nachrichten sind dem jeweiligen Netzbetreiber bekannt. Während es relativ schwer ist, sich in das System dieser Unternehmen zu hacken, lässt sich die Schwachstelle "Mensch" durch das sog. "social engineering" viel leichter ausnutzen.[12]
Nicht zuletzt lässt die Möglichkeit eine Kurznachricht mit beliebigem Absender zu versenden der Fantasie noch viel Spielraum, um sich Einsatzzwecke auszudenken mit denen dem Empfänger oder dem angeblichen Absender viel Schaden zugefügt werden kann. Über kurz oder lang werden die Mobilfunkprovider zu den gleichen Mitteln greifen müssen, wie es die Webmailprovider schon längst getan haben. Ohne eine aktive Filterung von Nachrichten, Untersuchung auf gefährliche Inhalte und Virenüberprüfung wird der SMS/MMS/EMS Verkehr in absehbarer Zukunft nicht mehr auskommen.

8.4 Provider

8.4.1 allgemeine Sicherheitskonzepte

Zusätzlich zu dem GSM Sicherheitskonzept, setzen die Provider zunächst auf den Schutz ihrer Hardware, durch Firewalls, Intrusion-Detection-Systeme, etc, denn innerhalb eines Rechenzentrums eines Providers liegen die Kundendaten komplett in unverschlüsselter Form vor. Außerdem kommt ein Fraud-Detection-System zum Einsatz, was Betrug und dem so genannten "social-engineering" entgegenwirken soll. Dieses System überwacht den Datenverkehr auf ungewöhnliche Vorgänge. So kann z.B. ein Mobilfunkteilnehmer, logisch betrachtet, nicht erst in Hamburg telefonieren und 3 Minuten später in Paris. Ist das der Fall, schlägt das System an.

8.4.2 vertrauenswürdige Provider?

Die meisten Slogans von Mobilfunkbetreibern suggerieren dem Kunden Sicherheit und Anonymität. Die Realität sieht etwas anders aus. So ist es lange kein Problem mehr, den Standort eines Handys bis auf wenige Meter genau zu bestimmen. Das die staatlichen Behören sich diesem Verfahren, zwecks Erstellung von Bewegungsprofilen, schon seit längerem bedienen, ist auf keiner Providerseite zu finden. Ebenso wenig wie die Tatsache, dass staatliche Ermittlungsbehörden zukünftig über für sie eigens entwickelte Schnittstellen einen Vollzugriff bekommen. Somit ist es dann möglich, beim "Abhören von Gesprächen", den Provider komplett außen vor zu lassen.

8.4.3 sicheres Roaming?

Das Prinzip des Roaming klingt einfach, die Realisierung in der Praxis weißt jedoch Tücken auf. Ein Problem dabei sind die ungleich starken Kryptoverfahren. Andere Länder haben diesbezüglich andere Bestimmungen. Im schlechtesten Fall einigen sich die Provider auf den kleinsten gemeinsamen Nenner und damit auf eine sehr geringe oder gar keine Verschlüsselung. Ein weiteres Problem ist, dass der Teilnehmerschlüssel nur dem Heimnetz-Provider des Nutzers bekannt ist. Ein auswärtiger Mobilfunkbetreiber muss den Schlüssel erst beim Heimnetz-Provider erfragen. Diese Übertragung zwischen den Netzen findet unverschlüsselt statt. Ein ebenso problematischer Sachverhalt ist, dass die konkrete Implementierung der A5 und A8 Algorithmen dem jeweiligen Netzbetreiber obliegt. Es gibt keine internationalen Standards dafür. Demzufolge können die Standards von Land zu Land abweichen.

8.5 Kleine Spione im Hosentaschenformat

8.5.1 Abhören und Ausspionieren

Mobiltelefone können dazu verwendet werden, unbemerkt Gespräche abzuhören. Im einfachsten Fall wird dazu ein Mobiltelefon mit einer Verbindung zu einem interessierten Mithörer unauffällig in einem Raum platziert. Durch eine geschickte Wahl von Leistungsmerkmalen und eventuell auch in Kombination mit einer Freisprechanlage lässt sich ein Mobiltelefon von außen in den Gesprächszustand versetzen, ohne das dies durch einen Rufton signalisiert wird.
Ein solches Abhören kann nur dann ausgeschlossen werden, wenn das Einbringen von Mobiltelefonen in den "sensiblen" Bereich verhindert wird. Um das sicher zu stellen

kann man GSM-Mobiltelefon-Detektoren einsetzen. Dabei handelt es sich um passive Warngeräte, die Mobiltelefone, die sich im Sendebetrieb befinden, anzeigen. Aktive Mobiltelefondetektoren, die alle Mobiltelefone in einem bestimmten Radius auffordern in den Sendebetrieb zu gehen sowie GSM-Störsender sind in Deutschland nicht zulässig.

8.5.2 Kamerawahn

Die Begeisterung für die neue Technologie ist groß, die Missbrauchsgefahr allerdings ebenso. Mit den neuen Kamera-Mobiltelefonen können heimlich aufgenommene Fotos (MMS), aus dem Zusammenhang gerissen, irgendwo publiziert werden. Beispiel: In einer Sauna geschossene Nacktbilder landen im Internet. Die fotografierte Person bemerkt nichts, weil die Kamera nicht zwangsläufig als solche erkennbar ist.
Eine Konsequenz daraus wird zukünftig sein, dass es zunehmend Handy freie Zonen geben wird. Die Hersteller reagieren ebenfalls auf die zunehmenden Beschwerden und wollen zukünftig Mechanismen in ihre Mobiltelefone einbauen, die bei jeder Bild-Aufnahme einen Signalton auslösen.[9]
In Japan und Italien gibt es bereits erste Verbote für die neuen Kamera-Handys (Fitnessclubs und öffentliche Bäder). In Saudi-Arabien hat die "Kommission zur Förderung der Keuschheit" den Verkauf dieser Handys komplett verboten.[10]
Ein weiterer, nicht unerheblicher Punkt ist die Industriespionage. In vielen Konzernen sind bereits jetzt Kamera-Handys aus Angst vor Spionagefotos verboten. Diese Verbote werden sich mit Sicherheit in den Unternehmen weiter ausweiten. Daher ist der Trend dazu, in alles kleine Kameras einzubauen durchaus problematisch. Man stelle sich nur einmal vor man würde in einer solchen Firma einen Geschäftstermin haben, müsste aber seinen PDA und das Handy beim Pförtner lassen. Das würde den Besuch sicherlich nicht erleichtern, besonders wenn man während des Besuches plötzlich auf Informationen aus dem PDA oder Handy zugreifen müsste.

8.6 Fazit

Seit der Einführung der GSM hat sich die zu Grunde liegende Technik kaum verändert. Im Gegensatz dazu, sind die Sicherheitsanforderungen jedoch rasant gewachsen. So steigen die getätigten Transaktionen per Mobiltelefon, von Jahr zu Jahr an. Das Geldabheben am Automaten per Identifizierung über die SIM, rückt ebenfalls in greifbare Nähe. Alles das sind Anforderungen, für die die GSM Technik nie vorgesehen war. Kurzum, die aktuellen Sicherheitsstandards werden den heutigen Anforderungen nicht mehr gerecht, da GSM nicht mehr in der Umgebung arbeitet, für die es einst konzipiert war. Dass diese Erkenntnis schon länger bekannt ist, zeigt die rasante Entwicklung in der Mobilfunktechnik. UMTS, als langfristiger GSM Nachfolger und der neue A5/3 Algorithmus, als Nachfolger von A5/1 und A5/2, stehen kurz vor der Einführung. Auch auf der Providerseite ist Einiges in der Entwicklung. Elektronische Signaturen, eine Biometrieerkennung (Daumenabdruck) zur Ablösung der Pin sowie sichere Zahlungssysteme sollen bald für den Endanwender zur Verfügung stehen. Doch bis es soweit ist, sei jedem empfohlen, den Autokauf, seine Investitionen sowie die monatliche Überweisung auf ein Nummernkonto in der Schweiz auf die althergebrachte Art und Weise zu tätigen und nicht unbedingt über die GSM-Dienste.

Abkürzungsverzeichnis

3GPP	3rd Generation Partnership Project
AuC	Authentication Center
BSC	Base Station Controller
BSS	Base Station Subsystem
BTS	Base Transceiver Station
CCC	Chaos Computer Club
CEPT	Conferénce Européenne des Administrations des Postes et des Télécommunications
EIR	Equipment Identity Register
EMS	Enhanced Messaging Service
ETSI	European Telecommunications Standards Institute
GMSC	Gateway MSC (Übergang zum Festnetz)
GSM	Global System for Mobile Communication (Groupe Speciale Mobile)
GPRS	General Packet Radio Service
HLR	Home Location Register
IMEI	International Mobile Equipment Identity
IMSI	International Mobile Subscriber Identity
ISAAC	Internet, Security Applications Authentication and Cryptography (Research Group)
ITU	International Telecommunication Union
LAI	Location Area Identifier
LBS	Location based Services
LDAP	Lightweight Directory Access Protocol
MAP	Mobile Application Part
MMS	Multimedia Messaging Service
MMS-C	MMS-Center
MS	Mobile Station
MSC	Mobile Switching Center
NMT	Nordic Mobile Telephon
PIN	Personal Identify Number
PUK	Personal Unblocking Key
SDA	Smart Card Developers Association
SIM	Subscriber Identity Module
SMIL	Synchronized Multimedia Integration Language
SMS	Short Message Service
SMSC	Short Message Service Center
SMS-GMSC	SMS-Gateway-MSC
SMS-IWMSC	SMS-Interworking-MSC
SMS-SC	SMS-Service-Center
SM-TP	Short Message Transport Protocol
TMSI	Temporary Mobile Subscriber Identity
UMTS	Universal Mobile Telephone Service
VLR	Visitor Location Register

Literaturverzeichnis

Rudolf Riemer, SMS Technik, http://www.umtslink.at/sms/sms_technik.htm

Rudolf Riemer, "MMS Technik", http://www.umtslink.at/sms/mms.htm

H. Federrath, "Sicherheit mobiler Kommunikation", Vieweg 1998

B. Walke, "Mobilfunknetze und ihre Protokolle", Teubner 2001

Chaos Computer Club, "GSM Cloning", http://www.ccc.de/gsm

Bundesamt für Sicherheit in der Informationstechnik, "GSM-Mobilfunk - Gefährdungen und Sicherheismaßnahmen"

"SMS History", http://www.umtslink.at/sms/mms.htm

1 T-Mobile Deutschland Unternehmensprofil

2 Dirk Schmidt, "Analyse: spam im Lebenszyklus der email-Technologie"

3 http://www.golem.de/0105/14133.html

4 http://www.teltarif.ch

5 Wiener Zeitung vom 27.12.2003

6 derstandard.at

7 http://www.teltarif.de/i/mms-inhalt.html

8 NZZ Online, "Im neuen Universum des Mobiltelefons", http://www.nzz.ch/2002/12/06/em/page-article8K703.html

9 Brueckenbauer, "Sicherheit im Datenjungle", http://www.brueckenbauer.ch/INHALT/0330/30gesell1.htm

10 RTL-World News vom 10.07.2003, http://www.rtl.de/rtlworld.html

11 tomorrow, "Sicherheitslücke Java?", http://tomorrow.msn.de/handy/handyfun/handy-games?page=6

12 PCTipp Meldung vom 29.11.2002, http://www.pctip.ch/webnews/wn/22688.asp

9 UMTS

L. Böhringer, K. Engelmann

9.1 Einführung

Die aktuellen Entwicklungen im Bereich der Mobilfunkkommunikation haben den Markt stark verändert. Bereits seit einiger Zeit hat die Zahl der mobilen Nutzer die Zahl der Festnetzanschlüsse in Deutschland überholt [1] und der Wettbewerb um die Schaffung immer neuer Dienste, zwingt die Mobilfunkanbieter zu immer neuen Investitionen im Bereich der Netzinfrastruktur und Netzdienste. Während für den privaten Endkunden Performanz und Effizienz für hohe Akzeptanz sorgen, steht für Firmenkunden die Sicherheit der neuen Dienste eindeutig im Vordergrund.
Neue Dienste wie Videotelefonie oder Audiostreaming und natürlich auch die neuen, immer leistungsfähigeren Endgeräte[2], fordern immer höhere Datenübertragungsraten ein. Selbst die vor wenigen Jahren eingeführte -als ,,2-B“ bekannte- Generation rund um den Quasi-Standard GPRS (General Paket Radio Service) kann den bandbreiten hungrigen Anwendungen kaum noch gerecht werden. So wurden neue Übertragungstechnologien zwar zeitweilig mit den gestiegenen Anforderungen fertig, die dem zugrunde liegenden Sicherheitsmechanismen änderten sich derweil jedoch kaum. Abhilfe soll hier die dritte Mobilfunkgeneration UMTS, das Universal Mobile Telecommunication System, schaffen. Im Folgenden soll nun die Mobilfunkgeneration UMTS vorgestellt werden, deren Beitrag zur Sicherheit des Netzes und der Nutzdaten geklärt und ein Blick auf möglicherweise auftretende Sicherheitslücken geworfen werden.

9.2 GSM

9.2.1 Sicherheitslücken im GSM Standard

GSM, das Global System for Mobile communications, ist der direkte Vorgänger von UMTS. GSM wies bei seiner Einführung einen hohen Sicherheitsstandard auf und verwendete einen sehr komplizierten Protokoll- und Signalstack. Diese waren auf teurer Hardware implementiert und verwendeten der Öffentlichkeit unzugängliche Leitungen. Trotz der Umstellung auf neuere Übertragungsverfahren und Mobilfunkgenerationen (GPRS/3G) und deren wachsender Funktionalität, wurden die verwendeten Sicherheitsmechanismen seit Jahren nicht verändert. Das Ergebnis sind unzureichende, unsichere Sicherheitsmechanismen, wie sie bereits in [8] erklärt wurden.
Die Sicherheitsmechanismen beschränkten sich auf

- User Authentifizierung mittels Abfrage einer PIN Nummer

[1] UMTS Security Awareness - Report from the UMTS Forum, Seite 7
[2] Als ständiger Anreiz die neuen Möglichkeiten auch auszureizen.

- Radio Interface Verschlüsselung mittels des symmetrischen Verschlüsselungsverfahrens A5
- Identitäts- und Ortverschleierung mittels temporärer Useridentifikationen

Eine wechselseitige Authentifizierung aller Kommunikationsteilnehmer, stärkere Verschlüsselung durch offene Verschlüsselungsverfahren realisiert, sowie ein austauschbares Sicherheitsverfahren und ein weltweit einheitlicher Standard zur Absicherung der Netze, hätten Abhilfe schaffen können. Die Sicherheitsmechanismen in Ihrer damaligen Implementierung gerieten so jedoch bald in Kritik, da beispielsweise

- die nötige Hardware, um eine aktive Attacke mittels gefälschter Basisstationen durchzuführen, immer günstiger wurde
- Schlüssel für die Datenverschlüsselung im Klartext zwischen den Netzwerken ausgetauscht werden
- die User- und Verbindungsdaten von der BTS (base transceiver station) zur BSC (base station controller) im Klartext übermittelt werden.
- die Verschlüsselung betreiberseitig ausgeschaltet werden kann, um in Ländern, die starke kryptographische Algorithmen nicht erlauben, dennoch verwendet werden zu dürfen
- Sicherheitsfeatures nicht verändert und auf einen aktuelleren Stand gebracht werden können (Auch heute wird noch immer der gleiche Kryptographiealgorithmus wie vor 5 Jahren verwendet)
- die geheim gehaltenen Verschlüsselungsalgorithmen im Falle von D2 Vodafon an die Öffentlichkeit gerieten, schnell zum Ziel von Kryptoanalysen wurden und gravierende Sicherheitsmängel besaßen.

9.3 UMTS

9.3.1 Definition von UMTS

UMTS, das Universal Mobile Telecommunication System, ist ein mehrere Technologien beinhaltendes modulares System. Es verbindet existierende und zukünftige mobile und fixe Netzwerke. Ähnlich wie in 2G Netzen, ist das UMTS Netz zellulär aufgebaut. *Die Leistungsfähigkeit des UMTS-Netzes wird durch eine Zellenstruktur erzielt, wobei Zellen unterschiedlicher Größe und mit unterschiedlichen Datenraten kombiniert werden. [...]Die Netz-Infrastruktur des zellularen UMTS-Netzes kennt als zentrale Zellenstation die Basisstation, die [...] Node B genannt wird. Diese leitet die Daten an die übergeordnete Steuereinheit weiter, den Radio-Network-Controller (RNC). Von dort geht es über eine Schnittstelle, dem Media-Gateway, ins Kernnetz. Der funktechnische Teil des UMTS-Netzes heißt UTRAN (UMTS Terrestrial Radio Access Network).[5]* Das UMTS Konzept umfasst außerdem alle Applikationen und Dienste, die dem Enduser angeboten werden können. *UMTS wird im Gegensatz zu Mobilfunksystemen der 2. Generation kein abgeschlossenes System darstellen. Letztere waren über ein Vermittlungsnetz, dem sogenannten Signal System No. 7 mit dem Festnetz verbunden. Durch die Verschmelzung von UMTS mit dem Festnetz und dem Internet werden sowohl der*

Mobilfunkteilnehmer, als auch der Festnetzteilnehmer neuen Gefahren[3] *ausgesetzt.*[4]Auf diese Gefahren wird später genauer eingegangen.

9.3.2 Von GSM zu UMTS

In 21 Ländern laufen bereits UMTS-Testnetze, davon vier in Asien (China, Taiwan, Malaysia und Singapur) und sieben in Europa (Italien, Deutschland, Österreich, Monaco, Großbritanien, Frankreich und Finnland). Dort wurden bereits Sprachtelefonate und auch Live-Bilder über UMTS-Testnetze übertragen oder Verbindungen von den UMTS-Netzen mit den bestehenden GSM- und Festnetzen hergestellt[5]. Wichtige Anforderungen an die dritte Mobilfunkgeneration sind höhere Datenraten, optimale Implementierung paketorientierter Dienste und die längst überfällige Realisierung eines weltweiten Standards. *,,[...]Um diese Kriterien zu erfüllen, wurde 1992 von der ITU (International Telecommunications Union) der IMT-2000 (International Mobile Telecommunications at 2000 MHz) Standard ins Leben gerufen. Um allen Nationen wirtschaftlich entgegenzukommen, gliederte man in den IMT-2000 mehrere Einzelstandards ein. Das ermöglicht diversen Netzbetreibern, dass diese ihre zum Teil bereits bestehende Netzstrukturen aus der zweiten Mobilfunkgeneration in die zukünftigen 3G-Netze implementieren können. [...] In den USA und weiteren Ländern Nord- und Südamerikas kann das bestehende IS-95 2G-Mobilfunknetz in die Architektur des 3G Standards CDMA2000 integriert werden, da beide Netze im Funknetzteil bereits CDMA-Multiplexverfahren (CDMA - Code Division Multiple Access) verwenden.“[10]* Dies bedeutet auf der einen Seite eine finanzielle Entlastung der Netzbetreiber. Auf der anderen Seite werden jedoch bestehende Netze, die bereits genannte oder noch nicht bekannte Sicherheitslücken aufweisen, in ein neues System eingegliedert, welches die Mängel der bestehenden Netze doch eigentlich beseitigen und vor allen Dingen sicher sein sollte. Ob nun diese Eingliederung in UMTS durch Aufwertung der Sicherheitsmechanismen der bestehenden Netze, oder durch starke Kompatibilitätserweiterungen seitens UMTS von statten geht, entzieht sich unserer Kenntnis.[6]
Obwohl viele Mängel in den Sicherheitsmechanismen der zweiten Mobilfunkgeneration entdeckt wurden, sind nicht alle grundsätzlich als schlecht zu bewerten. Um weiterhin möglichst kompatibel zu bleiben, baut UMTS auf dem Sicherheitssystem der zweiten Mobilfunkgeneration auf. Einige der übernommenen Mechanismen sind

- die Authentifizierung von Benutzern für Servicedienste
- die Chiffrierung der Daten auf dem Radiointerface
- die SIM Karte als austauschbarer Hardware-Identifizierungsmechanismus
- die verminderten Sicherheitsmechanismen zwischen dem Node B (entspricht BTS) und dem RNC(entspricht BSC), da diese sich gegenseitig vertrauen

[3] Diese neuen Gefahren sind in Ihrer Gänze noch nicht vorhersehbar, was deren Verhinderung stark erschwert

[4] Stefan Kralicek,UMTS Security, Ruhr-Universität Bochum Lehrstuhl Kommunikationssicherheit, ITS-Seminar, Seite 12

[5] Einige Netze wurden bereits offiziell gestartet, darunter das österreichische UMTS Netz, das UMTS Netz in Tokio, Japan und auch das D1 Netz in Deutschland. Eine vollständige Liste aller bereits öffentlich laufenden Netze konnten wir leider nicht finden.

[6] Da es beispielsweise bisher nicht möglich war die Sicherheitsmechanismen bestehender 2G-Netze auf den neuesten Stand zu bringen, oder es aus wirtschaftlichen Gründen nicht gemacht wurde, nehmen wir die genannte Kompatibilitätserweiterung als Lösung an.

Obwohl UMTS auf den unsicheren Sicherheitsmechanismen von GSM aufbaut, impliziert dies nicht unbedingt eine Unsicherheit für UMTS. Gemeint ist hier lediglich die Übernahme der Mechanismen, nicht deren Implementierung– diese ist auf die neuen Bedürfnisse der 3G zugeschnitten. Um UMTS aber nicht die gleichen Fehler wie 2G machen zu lassen, bedarf es zunächst der Klärung, was Sicherheit für das Netz bedeutet und welche Sicherheitsverletzungen bei einem Mobilfunknetz zu erwarten sind.

9.3.3 Erwartete Sicherheitsrisiken im UTRAN

Das UMTS-Forum, als internationaler Zusammenschluss von Herstellern und Lizenznehmern für Produkte der 3G, definiert Sicherheit in UMTS anhand der folgenden vier Punkte[7]

- *Integrity* - Daten dürfen in keiner Richtung verändert werden, bevor sie am Ziel ankommen
- *Privacy/secrecy* - Nachrichten dürfen nicht abgefangen oder auf einem unsicheren Medium zwischengespeichert werden
- *Authentication* - Nachrichten müssen von einem autorisierten Sender kommen
- *Non repudiation* - Nachricht müssen belegen können, dass sie vom Gegenpart der aufgebauten Verbindung kommen

Was aus dieser Liste nicht hervorgeht, aber dennoch von großer Bedeutung ist, ist der Schutz aller Gegenstände, die Teil des Netzes sind und aller Informationen, die über das Netz übertragen werden. Diese Erweiterung ist notwendig, da sich die vom UMTS-Forum ausgeführten Punkte nur auf die Daten bezieht, welche übertragen werden, nicht jedoch auf die enthaltene Information. Selbst wenn es gelungen sein sollte die Daten einer Verbindung oder den physikalischen Träger der Daten zu kompromittieren, so sollte ein Zugriff auf die enthaltenen Informationen nicht möglich sein. Der Schutz vor unrechtmäßigen Zugriffen jeglicher Art muss folglich gewährleistet werden für

- Informationen privater, firmeninterner oder gemeinschaftlicher Art
- Rechte für den Zugriff auf Netz und Dienste
- Netzressourcen in physikalischer und funktionaler Ausprägung

Um die nötigen Maßnahmen zur Absicherung des Netzes und seiner Inhalte festzustellen, muss geklärt werden, auf welche Art und Weise diese gefährdet werden können. Einige der häufigsten Bedrohungen in einem Mobilfunknetz seien im Folgenden genannt.[8]

- Unerlaubter Zugriff auf Daten
- Verletzung der Datenintegrität
- Fraud[9]

[7] UMTS Security Awareness - Report from the UMTS Forum, Seite 10
[8] UMTS Security Awareness - Report from the UMTS Forum, Seiten 10-12
[9] Eine Übersetzung in „Betrug“ kann den Charakter dieser Gruppe an Bedrohungen nicht korrekt wiedergeben, weswegen weiterhin Fraud verwendet wird

- Denial of Service[10]
- Diebstahl

Was aus der Liste nicht hervorgeht, jedoch nicht fehlen darf sind Bedrohungen durch

- Fehlerhafte Anwendungssoftware
- Malware(malicious software)
- Legitime Benutzer

wobei jeder dieser Punkte eine oder mehrere Bedrohung aus der vorhergehenden Liste mit sich bringen kann. Und als letzten Punkt die Kommunikation mit anderen unsicheren Netzen, hauptsächlich jedoch 2G.
Unter ,,Unerlaubtem Zugriff auf Daten“ versteht man

- das Abhören einer Verbindung - Eine Nachricht wird abgefangen oder mitgehört. Ist der Angreifer in der Lage den Protokollheader zu entfernen liegt ihm die Nachricht offen
- das Masquerading - Ein Eindringling gibt sich als legitimer Teil des Netzwerkes aus
- die Verkehrsanalyse - Anhand der Sammlung von Daten, erhält ein Eindringling Informationen und kann diese in Verbindung zu bestimmten Abläufen bringen
- das Durchsuchen von Speichermedien nach wichtigen Informationen

Die Datenintegrität wird durch das illegitime Ändern, Löschen oder Einfügen von Daten verletzt. Dies kann absichtlich herbeigeführt werden, zum Beispiel durch Viren oder Anwender mit böswilligen Absichten, oder unabsichtlich durch Fehlbedienungen oder fehlerhafte Anwendungssoftware.
Als Fraud bezeichnet man unter anderem

- die Fehlnutzung von Benutzerrechten - Rechte werden verwendet, um an verbotene Informationen zu gelangen
- die Nutzung eines Services, um an Daten zu gelangen, die normalerweise nicht zugänglich sind - Ein Eindringling kann sich durch den Missbrauch eines Services einen Vorteil beschaffen
- den Missbrauch von Privilegien, um an Services oder Daten zu gelangen, ohne dafür zu bezahlen

Denial of Service Attacken auf ein Mobilfunknetz enstehen unter anderem durch

- die übermäßige Belastung des Traffics, um einen Service für autorisierte Benutzer zu behindern
- die übermäßige Benutzung eines Service, um andere Benutzer von dessen Nutzung abzuhalten

[10] Dieser Begriff hat sich in der IT eingebürgert, weswegen von einer Übersetzung abgesehen wird

- Services, die dazu verwendet werden können das Netzwerk übermäßig zu belasten
- Anwendungsfehler, die zu einem Fehlverhalten führen
- Malware(malicious software), die die Behinderung eines Dienstes zum Ziel haben

Oftmals erfolgt Diebstahl

- der Identität eines legitimen Users, um unter anderen Rechten agieren zu können oder finanziellen Schaden hervorzurufen
- von geistigem Eigentum
- von Dokumenten und Informationen
- von Endgeräten und darauf aufbauend die illegitime Nutzung eines Services

Fehlerhafte Anwendungssoftware kann, genau wie Malware zu einer der anderen Bedrohungen führen. Ist eine Sicherheitslücke erst einmal gefunden, oder passende Malware installiert, stehen meist Tür und Tor für den Zugriff auf Netzressourcen und Informationen offen.
Legitime Benutzer sind eines der am schwierigsten zu beseitigenden Bedrohungen in einem Netz. Durch fehlende Aufklärung eines Nutzers oder böswillige Absicht, kann jede mögliche andere Bedrohung hervorgerufen werden. Diese Art der Bedrohung ist deswegen so schwierig zu unterbinden, da eine Masse von Nutzern, seien es nun Endkunden oder Mitarbeiter, nicht adäquat auf Fehlverhalten überwacht werden können. Zwar können beispielsweise Benutzerrechte sehr restriktiv gehandhabt werden, dennoch gibt es immer einen Benutzer oder eine Benutzergruppe, die mehr Rechte besitzt und erneut zu einer potentiellen Bedrohung wird.
Der wichtige Aspekt der Abwärtskompatibilität zu Netzen der zweiten Mobilfunkgeneration oder dem Festnetz stellt indes nur teilweise eine Bedrohung dar, die abhängig von der Aufgeklärtheit des Benutzers und von den Mitteilungen des Netzanbieters sind, die der IMT-2000 Standard als Sichtbareit von Sicherheitsmechanismen für den Nutzer vorschreibt. So ist die Kommunikation beispielsweise zwischen einem 2G Netz und einem 3G Netz ohne weiteres möglich, allerdings erfolgt hierbei keine End-To-End Verschlüsselung wie sie bei einer Kommunikation innerhalb des 3G Netzes stattfinden würde. Statt dessen werden 2G verschlüsselte Verbindungen bis zum UMTS Kern-Netzwerk transportiert und Antworten auch von dort zurück geschickt. Die empfangenen Datenpakete werden nach dem 2G Verfahren dechiffriert, decodiert und anschließend im entsprechenden 3G Verfahren wieder codiert und chiffriert, an den Empfänger der Information weitergeleitet zu werden. Um der angesprochenen Sichtbarkeit von Sicherheitsmechanismen zu genügen, muss dem 3G Endgerät der Kommunikation der Umstand der verminderten Verbindungssicherheit mitgeteilt werden. Ob und wie diese Mitteilung letztendlich dem User angezeigt wird ist jedoch nicht standardisiert.[11] Wird dem User mitgeteilt, dass seine Verbindung nicht End-To-End gesichert wurde, so obliegt es ihm selbst entsprechende Maßnahmen zu treffen(Anruf nicht annehmen, Datentransfer nicht starten etc.). Wird

[11] Bei einem Test mit aktuellen UMTS Telefonen von Motorola und Siemens, konnten wir weder im D2 noch D1 UMTS-Testnetz eine optische oder akustische Mitteilung entnehmen, die eine verminderte Sicherheit bei den Gesprächen zwischen UMTS-Endgerät und Telekom Festnetz oder Handynetz angezeigt hätte. Hieraus lässt sich jedoch nicht schließen, ob es sich um ein noch nicht implementiertes Feature, oder um eine bewusste Unterdrückung dieser Nachricht handelt.

dem User keine Nachricht angezeigt, so entsteht in jedem Fall eine Bedrohung, die vom Diebstahl der Daten, bis hin zur Einschleusung von gefährlichem Code auf dem Endgerät führen kann[12]. Egal ob der Benutzer über die Sicherung einer Verbindung Bescheid weiß oder nicht, ob er sich selbst der Sicherheitsrisiken einer solchen Übetragung bewusst ist oder nicht, das Risiko bleibt bestehen. Die Abwärtskompatibilität wurde teuer erkauft.

9.3.4 Was muss gewährleistet werden?

Das die 3GPP (Third Generation Partnership Project) bei der Standardisierung von UMTS und der Implementierung des Sicherheitssystems das Rad nicht komplett neu zu erfinden hatte, steht außer Frage.
The UMTS standardisation is driven by 3GPP, whereas IETF is the standardisation organisation body that drives all IP (related) activities. 3GPP adopts the IETF recommendations (RFC's) with respect to the definition of the IP part of the future UMTS network[. . .] [13]
Aus den Sicherheitsbestrebungen der 3GPP lassen sich zwei Hauptaspekte ableiten. Die Sicherheit der Privatsphäre des Endusers und die Integrität der Daten. Es soll weiterhin gelten, dass

- ein Sicherheitssystem skalierbar sein muss, damit es den verschiedensten Ansprüchen in Hinblick auf die verwendeten Algorithmen oder die Geschwindkeit seiner Berechnung gerecht werden kann
- ein Sicherheitssystem End-to-End funktionieren sollte, um die als schützenswert erachteten Daten auf keinem Teil einer Gesamtübertragungsstrecke unsicher zu übertragen
- das Sicherheitssystem minimal sein muss, damit beispielsweise keine unnötigen Pakete übertragen werden
- das Sicherheitssystem aufgrund der begrenzten Resourcen der Endgeräte sehr sparsam hinsichtlich der benötigten Speicherkapazität und Rechenleistung sowie schnell zu berechnen sein muss.

Leider stellt man bei der Realisierung dieser Bestrebungen immer wieder fest: *,,Absolute security is absolute impractical¡'* [14] und impliziert stets hohe Kosten, weshalb es *,,best practice"* [15] ist, einen Kompromiss zwischen Sicherheit, Nutzbarkeit und Kostenintensität zu finden. Der dahinter stehende Gedanke lautet also. Je mehr Geld investiert wird, desto sicherer kann das System gestaltet werden.

9.3.5 UMTS Sicherheit–Vorschlag der 3GPP

Die 3GPP schlägt eine Einteilung des UMTS Netzes in drei große Bereiche vor, wobei sie sich nach den typischen Service Funktionalitäten richtet

[12] als Beispiel wären hier SMS zu nennen, die ein Handy zum Abstürzen bringen können
[13] 3rd Generation Partnership Project, UMTS Security Awareness - Report from the UMTS Forum, 4. Auflage Seite 13
[14] 3rd Generation Partnership Project, UMTS Security Awareness - Report from the UMTS Forum, 4. Auflage Seite 1
[15] 3rd Generation Partnership Project, UMTS Security Awareness - Report from the UMTS Forum, 4. Auflage Seite 2

- Applikationsebene - u.a. Ablauf der Dienste
- Netzwerkkernebene - u.a. Authentifizierung / Verschlüsselung
- Transportebene - u.a. Transport der Daten auf physikalischer Ebene

Diese Ebenen stellen die Grundlage der UMTS Sicherheitsarchitektur dar, welche sich auch in das ISO/OSI Referenzmodell eingliedern lassen(siehe Bild 9.2 Seite 95).

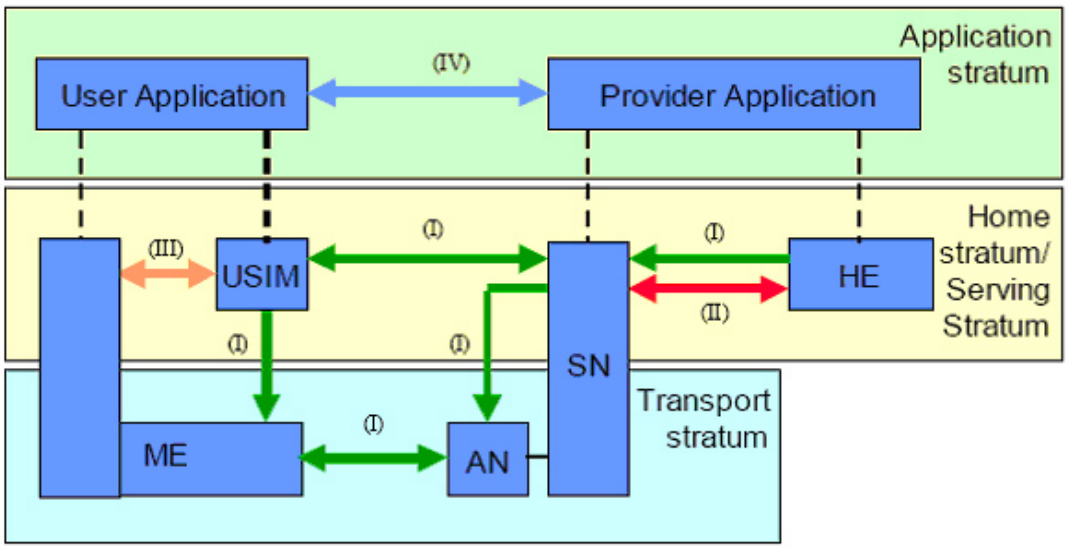


Abbildung 9.1: Schematische Darstellung der Sicherheitsebenen eines UMTS Netzes

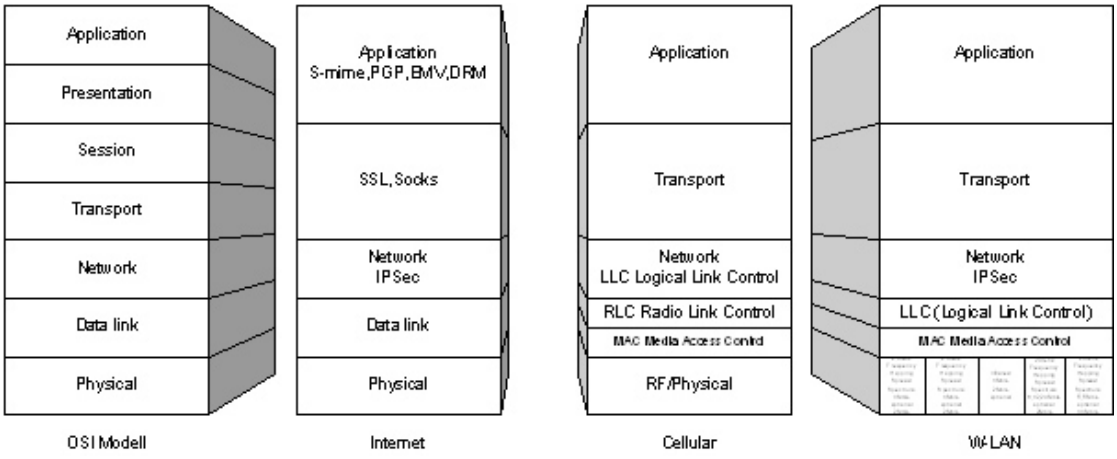


Abbildung 9.2: Versuch der Einordnung in das ISO/OSI Referenzmodell

Wie man allerdings feststellen muss, fehlen der Darstellung zwei weitere äußerst wichtige Ebenen. Wie bereits erklärt ist UMTS kein abgeschlossenes System, sondern möchte andere Netze möglichst nahtlos integrieren. Die Sicherheitsebene auf der sich Fremdnetze befinden und deren Übergang zum UMTS Netz gehören eng zur Netzwerkebene, werden durch diese jedoch nicht vollständig abgedeckt, wie man am Beispiel der

Abwärtskompatibilität (s.S. 93) erkennen kann. Außerdem bietet UMTS noch eine weitere Ebene, das IMS (IP Multimedia Subsystem). Diese umgibt die drei erstgenannten Sicherheitsebenen(s. 9.3.6 Seite 98) und darf bei der Betrachtung ebenfalls nicht fehlen. Man unterscheidet fünf definierte Sicherheitsgruppen(siehe Bild 9.1 auf Seite 95).

1. Netzzugangssicherheit
2. Sicherheit im Netzwerkbereich
3. Sicherheit im Benutzerbereich
4. Sicherheit im Anwendungsbereich
5. Sichtbarkeit und Konfigurierbarkeit

Jede dieser Gruppen unterliegt gewissen Gefahren und ist für bestimmte Sicherheitsaspekte zuständig.
Die Netzzugangssicherheit ist zuständig für einen sicheren Zugang zu den Diensten der 3G. Die wichtigsten Aufgaben dieses Bereiches sind der Schutz vor Angriffen auf die Funkverbindungen, durch Sicherung der übertragenen Daten und Informationen mittels Verschlüsselung(secrecy) und der Schutz der Benutzeridentität und dessen Aufenthaltsort(privacy). Ein Benutzer wird in der angemeldeten Basisstation durch eine temporäre Benutzerkennung eindeutig identifiziert. Um die Rückverfolgung seiner Identität oder des Aufenthaltsortes zu verhindern, wird diese Kennung periodisch erneuert und übertragene Daten mittels eines Privat-Key Verfahrens verschlüsselt. [16] Um aktive Attacken zu verhindern, erfolgt eine gegenseitige Authentifizierung der Kommunikationsteilnehmer und durch einen pro Nachricht berechneten Integritätsschlüssel, der vom geheimen Sitzungsschlüssel abhängig ist, werden die Checksummen der Nachrichten überprüft.
Die Sicherheit im Netzwerkbereich ist für die wechselseitige Authentifizierung der einzelnen Netzwerkinstanzen zuständig. Die einzelnen Sicherheitsmechanismen gleichen denen der Netzzugangssicherheit[17]
Für den sicheren Zugriff auf das Endgerät ist die Sicherheit im Benutzerbereich zuständig. Um diese zu garantieren, ist eine Authentifizierung durch eine PIN notwendig. Die Methode unterscheidet sich dabei nicht von der bereits in 2G verwendeten Userauthentifikation. Dies ist natürlich ein mehr als dürftiger Schutz vor ungewollter Verwendung eines Endgerätes. Dem erraten einer PIN (Geburtstage, einfache Zahlenfolgen) wird so ebensowenig ein Riegel vorgeschoben, wie der unfreiwilligen Herausgabe einer PIN seitens des legitimen Users(PIN steht auf Zettel). Das bessere Verfahren zur Userauthentifizierung, nämlich Biometrie steht jedoch kurz vor der Marktreife. Durch die offene Sicherheitsschnittstelle ist die Umstellung auf dieses oder ein anderes Verfahren in Zukunft schnell und einfach möglich. Ein Restrisiko bleibt dennoch bestehen, wenn nicht alle Benutzer umgehend auf ein neues Authentifizierungsverfahren umsteigen. Sicherheit im Benutzerbereich bedeutet außerdem, dass Informationen, die auf dem Endgerät gespeichert sind vor unerlaubtem Zugriff geschützt sind. Hierzu kann zum Beispiel eine Verschlüsselung der Daten auf dem Endgerät vorgenommen werden. Die konkrete Implementierung eines solchen Schutzes hängt von den Herstellern der Endgeräte ab.

[16] 3GPP TS 33.105; Technical Specification Group Services and System Aspects;3G Security;Cryptographic Algorithm Requirements
[17] Das uns vorliegende Dokument enthielt in seiner von uns verwendeten, aktuellen Fassung noch keine Informationen über die Netzwerksicherheit in UMTS.

Die Sicherheit im Anwendungsbereich behandelt in erster Linie den sicheren Datenaustausch zwischen Anwendungen im Benutzerbereich und Providerbereich. Um dies zu ermöglichen, muss ein netzweit vertraulicher Datenverkehr vorliegen, d.h. es sollte eine vollständige End-to-End Verschlüsselung bestehen, was jedoch ausschließlich innerhalb eines oder mehrerer 3G Netze gewährleistet werden kann. Dies ist nicht mehr möglich, sobald ein Kommunikationspartner aus einem unsicheren Netz stammt, wie beispielsweise dem Internet. Wie bereits vermerkt, basieren UMTS Dienste auf dem IP Protokollstack und sind eng mit dem Internet verzahnt(s. Seite 94). Das Erzielen von Sicherheit gleicht einer Sisyphusarbeit, absolute Sicherheit ist unmöglich zu erreichen[18]. Um dennoch ein möglichst hohes Maß an Sicherheit, gemessen an den gegebenen Bedingungen(s.S.91), zu erreichen, werden nahezu alle aktuellen Sicherheitsverfahren zum Einsatz gebracht.[19]

- Firewalls
- Biometrie
- Intrusion Detection Systeme-Intrusion Detection System (IDS)
- Digital Rights Management
- Public Key Infrastructure (PKI)
- Sichere Übertragungsprotkolle (IPSec/VPN Tunnel...)
- Physische Sicherheit
- Antivirus Mechanismen

„Sichtbarkeit und Konfigurierbarkeit“ gibt dem Benutzer die Möglichkeit, jederzeit Einsicht in die aktivierten und deaktivierten Sicherheitsmechanismen zu haben. Werden bei einem Dienst gewisse Sicherheitseigenschaften benötigt, so muss dies dem Nutzer zuvor mitgeteilt werden. Sichtbarkeit besagt außerdem, dass der Benutzer informiert werden muss, bevor eine verschlüsselte bzw. unverschlüsselte Verbindung aufgebaut wird. Auch bei einem Handover (Zellwechsel), zum Beispiel von einer UMTS-Zelle in eine GSM-Zelle, ist eine Mitteilung des Benutzers notwendig(s. Beispiel Abwärtskompatibilität S. 93).
Durch die geforderte Konfigurierbarkeit kann der Benutzer entscheiden, ob eine Authentifizierung mit der USIM-Karte durchgeführt wird, ob unverschlüsselte Anrufe zugelassen bzw. entgegengenommen werden und welche Verschlüsselungsalgorithmen verwendet werden sollen. Fraglich ist jedoch, ob eine solche Möglichkeit den User überfordert und in welcher Art und Weise dies dem User präsentiert werden kann, wenn beispielsweise eine Änderung des Sicherheitssystems zu einem neuen Interface führen würde. Durch mangelndes Fachwissen oder sogar böswillige Absicht, wäre es somit einem User möglich das Sicherheitssystem seines Endgerätes stark herunterzufahren und potentiell gefährliche Inhalte auf seinem Endgerät auszuführen. Die Einspielung eines Trojaners wäre folglich eine mögliche Konsequenz, die das gesamte Sicherheitssystem von UMTS aushebeln könnte, in dem, vom Benutzer unbemerkt, Informationen legitim über das Netz an einen

[18] Absolute Sicherheit ist nicht nur bei der Kommunikation mit einem unsicheren Netz nicht zu erreichen, auch innerhalb des UMTS Netzes ist dieses Ziel nicht zu erreichen. Dank der gesamten Sicherheitsvorkehrungen ist man jedoch ein ganzes Stück näher dran!

[19] 3rd Generation Partnership Project, UMTS Security Awareness - Report from the UMTS Forum, 4. Auflage Seite 13

illegitimen Empfänger gesendet werden. Der Gedanke auf diese Weise an den auf der USIM gespeicherten geheimen Schlüssel und Identifikationsnummern zu gelangen liegt nahe, sollte aber dennoch fehl schlagen, da dieser nicht auslesbar auf der USIM abgespeichert werden.[20]. Dieses Horroszenario, bereits hervorgerufen durch eines der fünf grundlegenden Sicherheitsgruppen (s. S. 96), ist wahrscheinlich auch der Grund dafür, warum man in aktuellen Endgeräten noch keine Implementierung der Konfigurierbarkeit findet.[21]

9.3.6 IMS – Das IP Multimedia Subsystem

Das IMS ist eine Erweiterung für UMTS, welche verschiedene dezentralisierte multimediale Dienste wie Multimedia Calls, Sessions und Service Kontrolle ermöglicht und auf dem SIP Protokoll des IETF (RFC 2543) basiert, um für neue Dienste ein möglichst einheitliches Interface zu definieren. Dabei werden nicht nur Nutzdaten, sondern auch Signaldaten übertragen und für erstere QoS(Quality of Service) Parameter festgelegt. Das bei einem solchen allumfassenden Konzept die Sicherheitseinteilung in eine einzelne Netzwerkschicht nicht möglich ist, liegt auf der Hand. Das IMS versteht sich deswegen als eine, den gesamten UMTS Kern umfassende, Netzwerkschicht.

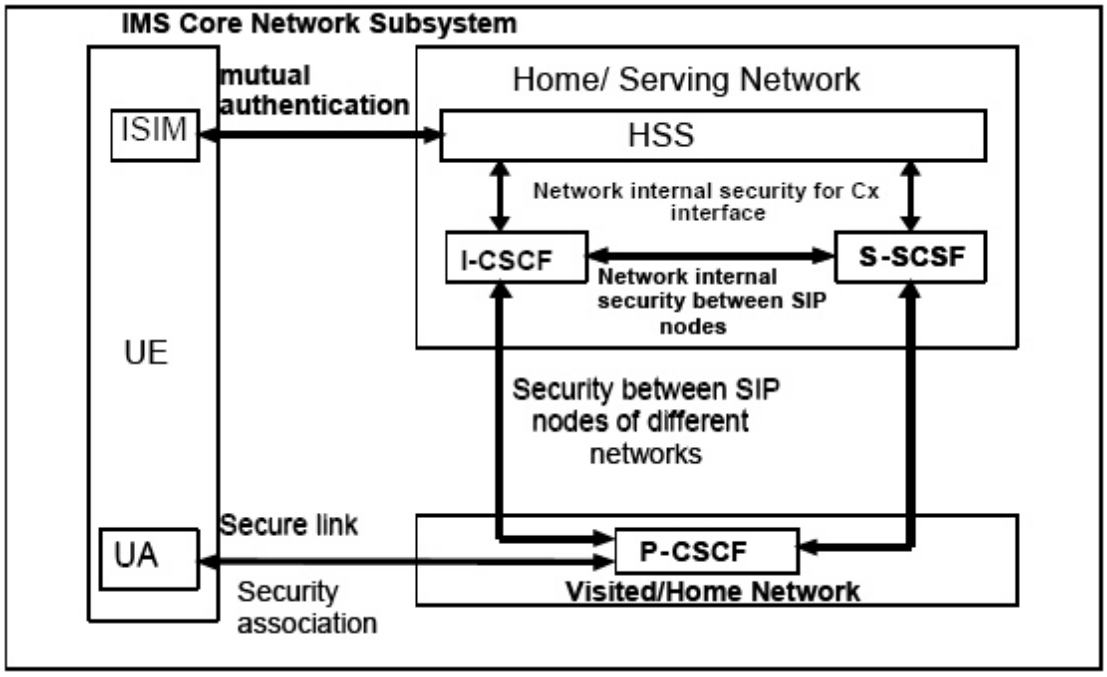


Abbildung 9.3: Schematische Einordnung des IMS in ein UMTS Netzwerk

Die Abbildung 9.3(s.S. 98)macht schematisch die Funktionsweise deutlich. Die ISIM Karte (IMS Subscriber Identity Modul) speichert Authentifizierungs- und Verschlüsselungsdaten, um sich beim HSS, dem Home Subscriber Server, anzumelden und umgekehrt. Das HSS übernimmt die Aufgaben des Home Location Registers, die IP Verwaltung und stellt die nötige Datenbank, um eine sichere Userauthentifizierung durchzuführen.

[20] Dies dachte man allerdings auch bis vor einiger Zeit von Informationen auf den GSM SIM Karten
[21] Es ist mir nicht bewusst ob dies für alle 21 UMTS-Testnetze gilt, weshalb sich die Aussage auf D2 und D1 Testnetze in Deutschland bezieht.

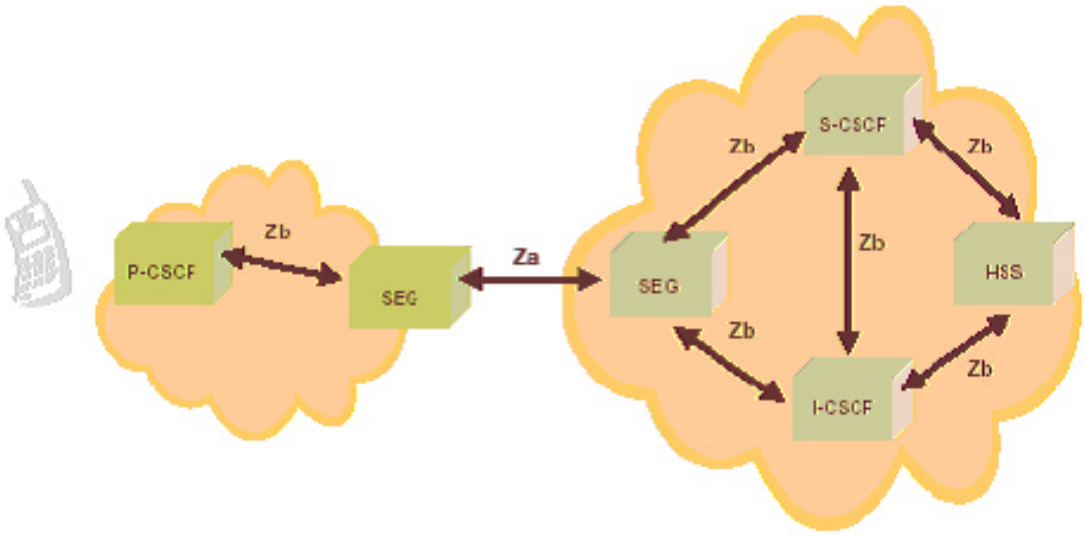


Abbildung 9.4: Sicherheitsbeziehung zwischen dem IMS und der Network Domain Security(NDS)/IP Netzwerk

Anhand der Abbildung auf Seite 99, werden hier nun kurz die wichtigsten Sicherheitsaspekte des IMS vorgestellt.
Ein- und ausgehender Traffic zwischen unterschiedlichen PLMNs (Public Land Mobile Networks) muss treuhändisch durch sichere SEGs (Security Gateways) geleitet werden. Ein Netzwerk kann mehrere Security gateways beinhalten. Die dabei fließenden Daten werden durch AES (Advanced Encryption Standard) verschlüsselt und über IPSec gesicherte VPN Tunnel von einem Sicherheitsnetz in ein anderes übertragen. Diese Netze sind stets weitere UMTS Netze.
Die IMS Sicherheitsmechanismen arbeiten unabhängig von den anderen bisher vorgestellten Sicherheitsmechanismen. Es steht noch zur Entscheidung, ob das ISIM auf der gleichen physischen UICC Karte (Universal Integrated Circuit Card)liegen wird, oder nicht.

9.4 Sicherheitstechnisch bedenkliche Punkte

Bei der Bearbeitung der Quellen, welche die Verfahren zur Authentikation, Verbindungsaufbau und Verschlüsselung zum Thema haben([3], [2], [4], [1]), vielen uns weiterhin folgende sicherheitstechnisch bedenkliche Punkte auf.

9.4.1 Klartextübertragung von Identifikationsdaten

Um eine Verbindung zu einer Basisstation zu initiieren, sieht das Protokoll für das initiierende Endgerät den Broadcast der TMSI (temporary mobile subscriber identity) vor, welche keine Rückschlüsse auf die Identität des Endgerätes gewähren soll. Ein Verbindungsaufbau über die permanente IMSI ist jedoch auch möglich. Der Broadcast der IMSI stellt damit einen Bruch in der vertraulichen Behandlung der Userdaten dar, die Möglichkeit dazu ist jedoch noch immer vorgesehen. Wozu diese Möglichkeit noch immer vorhanden ist entzieht sich leider unserer Kenntnis, genauso ob diese Möglichkeit auch

tatsächlich implementiert wird, oder ob der User die Möglichkeit erhält diesen Datentransfer abzuschalten oder nicht.

9.4.2 Übertragung der IMEI

Die Übertragung der IMEI(International Mobile Equipment Identification) Nummer eines Endgerätes an ein Serving Network wird im Normalfall nicht durchgeführt, oder erst, wenn eine gegenseitige Authentifizierug stattgefunden hat, schließlich identifiziert sie ein Endgerät eindeutig und sollte geheim gehalten werden. Laut Spezifikation kann ein Serving Network unter gewissen Umständen, welche nicht näher spezifiziert werden, die Herausgabe der IMEI fordern und erhält dadurch die eindeutige Benutzerkennung. Es ist nicht geplant diese Funktionalität zu entfernen. Allein die Tatsache, dass die ,,gewissen Umstände“ unter denen die eindeutige Identifizierungsnummer herausgegeben wird nicht näher bestimmt sind, lässt aufhorchen. Es ließe sich ohne weiteres ein Benutzer ausfündig machen oder überwachen, ohne das dieser etwas davon weiß. Hier wird erneut eine der fünf grundlegenden Sicherheitsgruppen (s. S. 96) gebrochen, nämlich ,,privacy“. Es stellt sich nun die Frage, warum überhaupt die Identität eines Nutzers verschleiert wird, wenn das Serving Network diese erfragen kann und ohne umschweife ausgehändigt bekommt. Die Antwort darauf kann leider erst dann gegeben werden, wenn die ersten UMTS-Testnetze für eine öffentliche Testphase freigegeben werden, und die Dokumente, welche diesen Umstand (eventuell unbeabsichtigt) beschrieben haben, eine finale Überarbeitung und Vervollständigung erfahren.

9.5 Fazit

Als direkter Nachfolger zu GSM war es die Aufgabe von UMTS dessen Sicherheitsmängel auszumerzen [8]. Dies ist dank vieler Verbesserungen auch gelungen, obwohl sich UMTS die Bürde auferlegt hat zu 2G kompatibel zu bleiben und somit auf dessen Sicherheitseigenschaften, aber auch dessen Sicherheitslücken aufbaut. Aktive Attacken mit gefälschten Basisstationen gehören dank der wechselseitigen Authentifikation der Vergangenheit an. Daten selbst werden End-To-End übertragen und dabei mit 128bit verschlüsselt. Selbst wenn dieses Verschlüsselungsverfahren geknackt werden sollte, so bietet der während der Verbindung ständig wechselnde Übertragungsschlüssel ausreichenden Schutz für die Daten und die temporären Useridentifikationen lassen keinen Rückschluss auf Ort und Identität eines Users zu. Zudem ist das Verschlüsselungsverfahren nicht fix, sondern kann ausgetauscht oder korrigiert werden, falls doch einmal eine Kryptoanalyse einen Fehler entdeckt. *Da die UMTS Algorithmen von einer eigenen Gruppe, der ,,Security Algorithms Group of Experts“ (SAGE), unter strengen statistischen und mathematischen Gesichtspunkten und unter besonderer Berücksichtigung der bereits bekannten Angriffspunkte entworfen werden, ist eine weltweite Kompatibilität der UMTS-Netze untereinander gewährleistet.*[22] Nach eigener Aussage sind die Algorithmen für ihren Zweck gut geeignet und es wurden keine praktischen Angriffsmöglichkeiten ermittelt. Da die Algorithmen öffentlich gemacht wurden, ist die Möglichkeit gegeben, das sich dritte Parteien mit dem Problem der Sicherheit beschäftigen, und gegebenenfalls Verbesserungsvorschläge entwickeln.

[22] Stefan Kralicek,UMTS Security, Ruhr-Universität Bochum Lehrstuhl Kommunikationssicherheit, ITS-Seminar, Seite 23

Die Implementierung der auf UMTS laufenden Netzdienste erfolgt auf Basis des standardisierten IP Protokollstacks der IETF und hat somit den Vorteil bereits ausgereifte Protokolle zu verwenden. Der somit entstandene Übergang zwischen den Sicherheitsebenen von UMTS und dem Internet ist jedoch eine Verletzung der Forderung nach End-To-End Sicherheit und erfordert weitere Sicherheitsmaßnahmen, die jedoch nicht definiert wurden.
Dank der Liste an Sicherheitsfeatures, welche man schon fast paranoid nennen mag, sind momentan noch keine Angriffe gegenüber UMTS bekannt, die alle Sicherheitsvorkehrungen durchdringen, allerdings wurde UMTS auch noch nicht in hohem Maße der Öffentlichkeit zur Verfügung gestellt, um diese Aussage zu validieren. Bedenklich jedoch ist, das mehrere grundlegende Sicherheitsgruppen, welche vom UMTS-Forum als essentiell vorgestellt wurden und sogar die Definition von Sicherheit, bereits als gebrochen gelten können (s. S.97/100). Auf der anderen Seite sind die Dokumente, welche den Sicherheitsstandard der 3G definieren sollen bisher noch nicht in der finalen Fassung vorhanden, weswegen eine Änderung der angesprochenen Punkte noch möglich ist. Was die Sicherheitsmechanismen letztendlich taugen, wird sich erst in der Praxis zeigen, denn leider hat UMTS momentan noch etwas zuviel Hype und noch viel zu wenig Erfahrung.

Abbildungsverzeichnis

Literaturverzeichnis

[1] 3rd Generation Partnership Project. 3GPP TS 21.133 V4.1.0 3G Security; Security Threats and Requirements. *3GPP Reference*, 4:341–356, 2001.

[2] 3rd Generation Partnership Project. 3GPP TS 33.105; Technical Specification Group Services and System Aspects;3G Security;Cryptographic Algorithm Requirements. *3GPP Reference*, 4:341–356, 2001.

[3] 3rd Generation Partnership Project. Technical Specification Group Services and System Aspects; 3G Security;Cryptographic Algorithm Requirements. *3GPP Reference*, 4:341–356, 2001.

[4] 3rd Generation Partnership Project. 3GPP TS 33.102 V4.3.0 3G Security; Technical Specification Group Services and System Aspects; Security Architecture. *3GPP Reference*, 5:341–356, 2002.

[5] Siemens AG. Siemens ag lexikon der datenkommunikation. In *Siemens AG Lexikon*. Siemens AG, 2004.

[6] UMTS Forum. UMTS Security Awareness - Report from the UMTS Forum. *UMTS Forum Report 30*, 4:341–356, 2003.

[7] BörseGo GmbH. Grundlagenserie umts. In *Grundlagenserie UMTS*, chapter 1. BörseGo GmbH, 2004.

[8] kenn ich noch nicht. UMTS Security, Universität Potsdam,Seminar ,,Umfassende Absicherung komplexer IT Infrastrukturen“. *Seminararbeit*, 1, 2004.

[9] Stefan Kralicek. UMTS Security, Ruhr-Universität Bochum Lehrstuhl Kommunikationssicherheit, ITS-Seminar. *ITS-Seminar*, 1:341–356, 2002.

[10] Rudolf Riemer. Umtslink.at gsm-gprs-umts - alles zum thema mobilfunk. In *UMTSLink.at GSM-GPRS-UMTS - Alles zum Thema Mobilfunk*, chapter 1. UMTS-Link.at, 2004.

[11] Reinhard Wobst. *Abenteuer Kryptologie*, volume 3. erweiterte Auflage. Addison Wesley, 2001.

10 Trusted Computing

M. Thränhardt, S. Weber

10.1 Einleitung

Die rechnergestützte Datenverarbeitung nimmt einen hohen Stellenwert auf der Welt ein und hat mittlerweile Einfluß auf nahezu alle Teile der Gesellschaft. Darüberhinaus sind einzelne Rechnersysteme immer mehr vernetzt. Es gibt viele Firmennetze, die mit dem Internet verbunden sind und auch ein Großteil der Privathaushalte haben zunehmend Zugang zum Internet. Damit ergibt sich ein großes Gefahrenpotential, z.B. durch Viren und Trojaner. Sie erreichen so eine Vielzahl von Rechner und haben außerdem meist ein leichtes Spiel, ihr Ziel zu erreichen. Besonders Rechner, die mit einem Betriebssystem der Reihe „Windows“ ausgestattet sind, werden oft zum Ziel. Das ist weit verbreitet und durch unsichere Standard-Einstellungen (z.B. in Mail-Clients) offen für Angriffe dieser Art. Desweiteren gibt es immer öfter Sicherheitslücken, die erst durch Software-Updates geschlossen werden müssen. Die hohe Komplexität von Betriebssystemen wird dann u.a. als Begründung angeführt. Auch die Manipulation von Hardware, z.B. das Entwenden der Festplatte, stellt eine Gefahr dar. Besonders wichtig ist dabei die Sicherheit von Rechner-Systemen in sensiblen Bereichen, wie etwa der Energie-Versorgung, Sicherheitsbehörden oder Militär.
Entwicklungen, die diese Bedrohungen bannen sollen, sind im wesentlichen nur auf der Ebene von Software zu vernehmen, wie z.B. Anti-Viren-Software oder Betriebssystem-Aktualisierungen. Von Erfolg sind diese Maßnahmen offenbar nicht gekrönt. Die Angriffe erlangen vielmehr eine immer größere Tragweite, wie man anhand jüngster Viren wie „Blaster“ sah.
„Trusted Computing“ soll hier wesentliche Veränderungen herbeiführen. Dazu wurde im Oktober 1999 die Organisation „Trusted Computing Platform Alliance“ (TCPA) gegründet. Mitglieder sind ein Großteil der Hardware- und Software-Industrie, wie z.B. Microsoft, Hewlett-Packard, Intel, AMD oder Via. Mittlerweile ist diese Organisation im wesentlichen in der neuen seit April 2003 existierenden „Trusted Computing Group“ (TCG) aufgegangen. Hier müssen bei Abstimmungen keine einstimmige Mehrheit, sondern nur eine 2/3 Mehrheit erreicht werden. Diese Organisation erstellt eine Spezifikation, die von grundsätzlich unsicheren Rechnersystemen ausgeht und grundlegende Veränderungen bei der Rechnerarchitektur vorsieht. Die Plattform soll dann eine vertrauensvolle Basis darstellen, so dass die darauf aufbauenden Systeme und Interaktionen ebenfalls als sicher gelten können. Intel arbeitet in diesem Zusammenhang bereits am Projekt „La Grande“, der diese Spezifikation umsetzen soll. AMD will dies mit dem „Hammer“ erreichen und Microsoft plant ein darauf aufbauendes Betriebssystem NGSCB (vorher: „Palladium“).
Erreichen will man vor allem die breite Masse der PC-Anwender mit ihren Notebooks und stationären Rechnern. Aber auch Mobiltelefone und PDAs sind ein Anwendungs-

feld für TCG. Es wird versprochen, die Anwender von Problemen wie Viren, Trojanern oder Spam zu befreien. Mit der in jedem TCG-System vorhandenen ID kann sich der Rechner identifizieren und authentifizieren. Dazu werden die vorhanden Komponenten des Systems an den Rechner gebunden (engl.: „dongle“). Besonders interessant für Unternehmen aber auch Privatanwender ist die Möglichkeit, Dokumente signieren zu lassen und somit vor Manipulation zu schützen. Die Spam-Flut ließe sich mittels Zertifikaten eindämmen, in dem jede E-Mail, die nicht signiert ist, z.B. gelöscht wird.
Gelöst werden soll dies zum einen durch grundlegende Änderungen an der Hardware. Zentrales Element ist ein Chip zunächst auf dem Motherboard. Später wird sich dieser direkt im Prozessor befinden. Er kontrolliert den Start des PC's („Secure Booting“). Wenn alle Hardware-Komponenten und das BIOS für vertrauenswürdig befunden wurden, startet das aufbauende Betriebssystem. Eine Komponente erhält keinen Zugriff auf geschützte Inhalte, wenn es kein gültiges Zertifikat vorweisen kann. Mehr über den Aufbau und die Funktionsweise von TCG und NGSCB ist im Teil 2 zu lesen.
Gern dementiert und heruntergespielt wird die Tatsache, dass „Trusted Computing“ auch und vorallem zur Kontrolle der Einschränkungen, die Anbieter von digitalen Inhalten den Nutzern auferlegen, dienen soll. Ohne Genehmigung des Rechteinhabers lassen sich geschützte Musiktitel nicht mehr kopieren. Ferner wird kontrolliert, dass nur genehmigte Software auf Medieninhalte zugreifen können. Insofern wird vorallem die Medien- und Softwareindustrie als Nutznießer gesehen, die mit einer vertrauensvollen Rechnerumgebung vorallem ihre Rechte durchgesetzt sieht. Hier wird häufig „Digital Rights Management“ (DRM) und NGSCB in Zusammenhang gebracht. Welche technischen Beziehungen zwischen TCG und DRM bestehen, wird ebenfalls in Teil 2 behandelt.
Das Projekt „Trusted Computing“ hat auch viel Kritik laut werden lassen. Die häufigen Namenswechsel von Organisation und Produkten sowie die zögerliche Informationspolitik wird als Verschleierungstaktik empfunden, weshalb das Vorhaben auch als „Treacherous Computing“ bezeichnet wird. Die potentielle Nutzung als Rechtekontroll-Instanz wird als Verlust über die Kontrolle des Rechners und damit als Entmündigung des Nutzers gesehen. Auch die Verwendung zur Zensur von Konkurrenz-Produkten und eine damit einhergehende Wettbewerbsverzerrung wird befürchtet. Mehr über diese und weitere Kritikpunkte ist im Teil 3 zu lesen.

10.2 Technische Realisierung

Die technische Realisierung von TCPA bzw. Palladium ist besonders interessant, um mögliche Angriffsmöglichkeiten und/oder Gefahren zu erkennen. Zentraler Gesichtspunkt dabei ist die hinzukommende Hardware und ihre Nutzung durch die Software, speziell des Betriebssystems. Grundsätzlich muss man hier unterscheiden, ob man sich gerade mit Palladium befasst oder mit der nicht so tiefgreifenden Spezifikation von TCPA. Größter Unterschied ist, dass die TCPA-Spezifikation nicht das Betriebssystem umfasst. Bestehende Betriebssysteme laufen auf dieser Plattform wie bisher. Das komplexere Palladium-Konzept umfasst das gesamte sichere Betriebssystem und stellt auch an die Hardware höhere Anforderungen. Doch bevor auf die einzelnen Unterschiede konkret eingegangen wird, erst ein Blick auf die Gemeinsamkeiten. Dazu soll die Frage geklärt werden, wozu überhaupt zusätzliche Hardware nötig ist bzw. wo die Grenzen des mit Software Machbaren liegen.

10.2.1 Probleme/Bedrohungen

Das erste Problem der bestehenden Hardwareplattform ist das Nichtvorhandensein eines hardwareseitigen Zufallszahlengenerators. Mit Software kann zwar das Erstellen über Eingabegeräte (z.B. zufällige Mausbewegungen) und den Timer simuliert werden, jedoch ist dies aufwendig und auch nicht manipulationssicher. Das zweite Problem schließt sich direkt daran an. Der Timer bisheriger Systeme ist nicht vor Veränderung geschützt, d.h. Software kann der Gültigkeit der Systemzeit nicht vertrauen. Bekannt sind kleine Programme, die während des Zeitchecks einer auf einen bestimmten Zeitraum befristet lauffähigen Demoversion die Systemzeit verändern, um so den Testzeitraum unendlich erweitern. Drittes Problem ist die eingeschränkte Kontrolle des Betriebssystems über Hardwarezugriffe. Bestes Beispiel ist Busmaster DMA. Diese Fähigkeit besitzt nahezu jede PCI-Steckkarte und erlaubt es dieser (mit entsprechenden Treibern) direkt auf den Speicher zuzugreifen. Was als Performance-steigerndes Feature entwickelt wurde, birgt eine nicht zu unterschätzende Gefahr. Über den Treiber eines solchen Gerätes lässt sich auf den gesamten Speicher zugreifen, auch eben auf den Bereich, wo sicherheitsrelevante Programme ihre Passwörter unverschlüsselt ablegen. Das vierte Problem wird besonders in Zukunft eine starke Bedeutung bekommen. Der Trend zum allgegenwärtigen Rechnen ist deutlich sichtbar. Ob es nun ein Laptop oder PDA ist, die Daten auf diesen Geräten sind deutlich gefährdeter als die auf einem normalen Heim-PC. Hat ein Angreifer direkten Zugriff auf die Hardware (durch Diebstahl, Unachtsamkeit des Eigentümers,...) ist es wesentlich leichter, einen Software-Schutz auszuhebeln und so an sensible Daten zu gelangen. Fünftes Problem stellt die Unversehrheit des PCs dar. Für den Anwender ist es nicht ersichtlich, ob an der Hardware manipuliert wurde, wenn sich die Anwendungen augenscheinlich wie immer verhalten. Gleiches gilt auch für die Software selbst. Für den normalen Anwender gibt es keine Möglichkeit zu verifizieren, ob die Anwendung seit dem letzten Zugriff modifiziert wurde oder ob es sich überhaupt um die gewollte Anwendung handelt, wenn am Aussehen und am Benutzverhalten nichts geändert wurde. So ist es möglich, dem Benutzer sein Online-Banking-Programm so zu verändern, dass eingegebene Passwörter an einen Internet-Server übermittelt werden.

10.2.2 Lösungen

Um diese Schwächen der bisherigen Hardware-Plattform zu eliminieren, sind durch TCPA/Palladium folgende Erweiterungen vorgesehen: Erstens, ein in Hardware realisierter Zufallszahlengenerator. Dieser soll das Problem der Erstellung kryptographischer Schlüssel lösen und den bisherigen sehr aufwendigen Weg deutlich vereinfachen. Die zweite Erweiterung ist eine Timer, der nur sehr eingeschränkt veränderbar ist. Damit kann die Software jederzeit auf eine gültige Systemzeit zurückgreifen und anhand dieser neue Sicherungsmöglichkeiten realisieren. Die dritte Erweiterung ist ein manipulationssicherer Speicherbereich. In diesem lassen sich z.B. wichtige kryptographische Schlüssel speichern. Da dieser Bereich in „abgeschirmter“ Hardware umgesetzt werden soll, ist er vor Angriffen gut geschützt. Durch diesen separaten Speicher ist es Busmasterfähigen Geräten nicht mehr möglich, alle sensible Daten auszulesen. Sealing ist das Schlagwort der vierten Erweiterung. Sensible Daten werden an die Konfiguration der vorhandenen Hardware gebunden. Die Idee dabei ist, dass aus den im PC befindlichen Hardwarekomponenten(Produkt-ID, Seriennummer) ein eindeutiger Schlüssel erstellt wird. Mit diesem Schlüssel werden sensible Daten verschlüsselt. Bei jedem Boot-Vorgang wird die Konfiguration neu ermittelt und auch ein neuer Schlüssel generiert.

Stimmt dieser Schlüssel nicht mit dem gespeichert Schlüssel überein, lassen sich die Daten nicht mehr entschlüsseln. Damit werden Manipulationen an der Hardware sofort aufgedeckt und der Benutzer kann indirekt die Integrität der Hardware prüfen. Denkbar wäre z.B. ein mit Passwort und Hardware-Konfig-Schlüssel codiertes Dokument, welches er als Original identifizieren kann. Wird etwas Falsches angezeigt, kann es nur an einem ungültigen Passwort oder einer veränderten Konfiguration liegen. Sehr eng mit der vorherigen Erweiterung verbunden ist die Möglichkeit des "Reporting". Dabei kann ein System seine Vertrauenswürdigkeit gegenüber einem anderen System beweisen. Hiermit kann der Schutz vor dem Anwender realisiert werden.

10.2.3 TCPA-Spezifikation

Um die eben aufgezählten Schutzziele technisch umzusetzen, muss zuerst das BIOS um neue Funktionen ergänzt werden. In der TCPA-Spezifikation heißt diese Ergänzung "Core Root of Trusted Module"(CRTM). Hinzu kommt ein Chip, namens "Trusted Platform Module"(TPM), der später näher erläutert wird. Nach dem Einschalten des TCPA-konformen Rechners bekommt das CRTM als erstes den Zugriff (s. Abbildung 10.1). Es bestimmt die aktuelle BIOS-Version und seine Einstellungen und speichert sie mittels eines Hashwertes (SHA-1) in einem speziellen Bereich des TPMs. Nach dieser Aktion bekommt das BIOS den Zugriff auf alle Hardwarekomponenten. Von jeder Komponente wird ein Hashwert gebildet. Alle Hashwerte werden gebündelt in einem oder mehreren Hashwerten wiederum in einen geschützten Bereich des TPMs geschrieben. Das TPM kann jetzt den Hashwert der aktuellen Konfiguration mit dem einer vorherigen vergleichen und bei Ungleichheit Funktionen wie Auslesen von Teilen des geschützten Speichers verweigern. Das BIOS ermittelt beim Abfragen der Komponenten auch den Master Boot Record des Boot-Laufwerkes. Dies ist notwendig, damit das aufbauende sichere Betriebssystem nicht selbst z.B. durch einen Bootsektor-Virus befallen ist. Wie das Betriebssystem weiter agieren soll, sagt die TCPA-Spezifikation nicht aus. Sie schafft also nur die Grundlagen.
Wesentliches Element der Spezifikation ist das schon angesprochene TPM. Dieser Hardware-Chip ist mit dem Motherboard fest verbunden und wird über den Low-Pin-Count-Bus benutzt. Das TCP beinhaltet u.a. eine RISC-CPU mit ROM, EEPROM und RAM. Nach aussen bietet es die Leistungen eines Zufallszahlengenerators, eines Timers und einer kryptographischen Engine an. Die kryptographische Engine stellt Funktionen wie Bildung von Hashwerten, asymmetrisch/symmetrische Verschlüsselung, Schlüsselgenerierung bereit. Optional kann ein Selbst-Schutz (Tamper Detection) vorhanden sein. Die Möglichkeit der Schlüsselspeicherung und -verwaltung bedarf einer genaueren Betrachtung.
Grundsätzlich werden die gespeicherten Schlüssel in migrierbar und nicht-migrierbar unterteilt. Wie der Name schon andeutet, lassen sich die nicht-migrierbaren nicht von einem TPM in eines anderes übertragen (bzw. nur mit erheblichen Aufwand). Ausserdem haben die Schlüssel eine unterschiedliche Gewichtung. Die zwei wichtigsten Keys sind: 1. Der Endorsement Key identifiziert eindeutig das TPM. Er ist nicht veränderbar und kann deshalb nicht auf ein anderes TPM übertragen werden. 2. Der Storage Root Key wird zur TPM-internen Verschlüsselung anderer Keys verwendet. Alle diese anderen Keys sind auf die Integrität dieses Schlüssels angewiesen. Auch er ist nicht migrierbar.

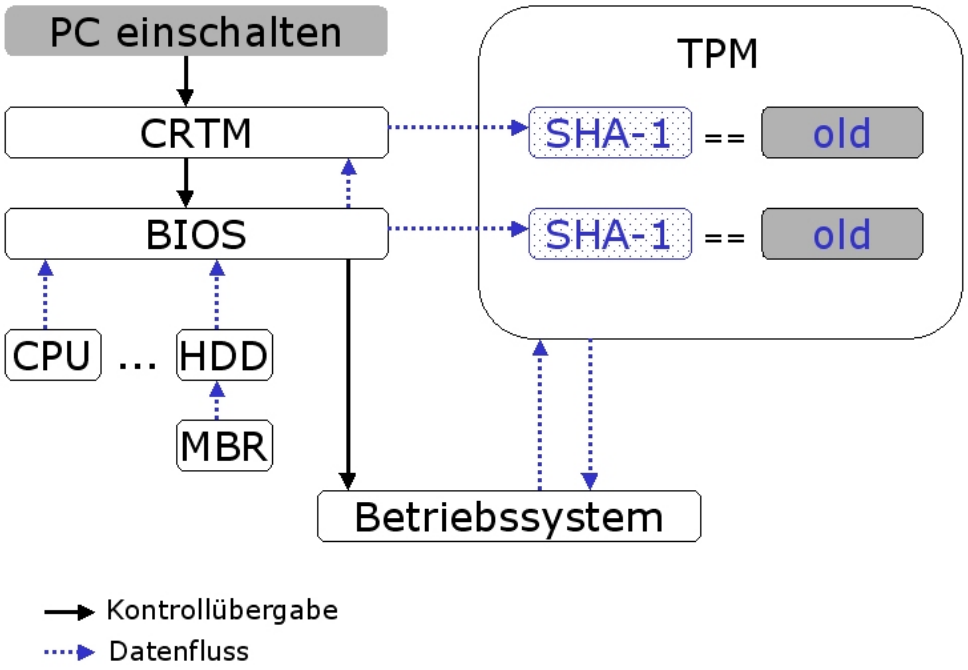


Abbildung 10.1: Vorgang des Secure Booting

10.2.4 Microsofts Palladium

In Palladium werden hardwareseitig die gleichen Funktionen bereitgestellt, genutzt wird im normalen PC das TPM. Doch Microsofts Pläne stellen auch an die restliche Hardware neue Anforderungen. Das bisherige Windows-Betriebssystem soll danach nicht grundsätzlich geändert werden, eher soll durch neue Hardware ein sicheres Umfeld geschaffen werden. Bei diesem Konzept spielt besonders die Software-Abwärtskompatiblität eine große Rolle. Jetzige Anwendungen sollen weiterhin laufen, nur in einem unsicheren Bereich. In den gesicherten Bereich, Nexus genannt, kommen dann die abgesicherten Anwendungen, die Agents. Diese Agents sollen sogar dann noch sicher laufen, auch wenn der PC schon kompromittiert wurde. Diese Agents laufen in isolierten Speicherbereichen und greifen nie direkt auf die Hardware zu. Zentrales Steuerungselement ist der Nexus-Manager. Er verifiziert die Daten, die zwischen beiden Modi ausgetauscht werden, und kontrolliert, ob sich die Agents an die ihnen zustehenden Rechte halten. Um die Funktion des Nexus umzusetzen, braucht dieser mehr Zugriffsmöglichkeiten auf die Hardware als bisherige Hardware erlaubt. Dabei bekommt er durch einen neuen CPU-Modus ("Ring -1") mehr Zugriffsrechte. Diese gestatten es z.B. Zugriff auf einen geschützten Speicher zu unterbinden. Dabei muss nicht nur die CPU einen neuen Modus bereitstellen, nahezu alle elementaren Hardwarekomponenten wie Chipsatz, Speicher müssen mit neuen Funktionen ausgestattet werden, um möglichst viel Kontrolle den Nexus zu gewährleisten. Weiterhin müssen sogar Ein-/Ausgabegeräte ihre angegebene Funktion nachweisen, um einen "Trusted Path"vom und zum Anwender aufzubauen (siehe u.a. vorgesehene Neuerungen in USB-Spez.2.3). Damit ein Programm Ägent"wird und in den sicheren Modus ausgeführt werden kann, muss es zertifiziert werden. Unklar ist, wie der Zertifizierungsvorgang abläuft und viel wichtiger, wer überhaupt berechtigt ist, zu zertifizieren.

10.2.5 Einsatz und damit verbundene Gefahren

Microsofts Weg über die Schaffung zweier Modi (Standard Mode, Nexus Mode) ist zwar sehr elegant, macht aber gleichzeitig die Schwächen von Windows deutlich. Microsoft nutzt hier eine Variante, nicht Windows sicher zu machen, sondern eine sichere Umgebung um Windows zu bauen. Weiterhin hält Microsoft immer wieder an Altlasten fest. Eine konsequentere Durchsetzung neuer Sicherheitskonzepte auf Kosten der Abwärtskompatiblität würde viele Sicherheitslücken erst gar nicht entstehen lassen. Des weiteren geht Palladium stark auf Software-Angriffe ein, das Problem der Mobilität wird untergeordnet behandelt. Der TCPA-Weg sieht zwar ein sicheres Betriebssystem vor, spezifiziert es jedoch nicht. Die entstehenden Sicherheits-Vorteile des Secure-Bootings bringen wenig, wenn das Betriebssystem nicht sicher ist. Genau hier liegt auch das Problem bei TCPA. Wird es jemals möglich sein, ein komplexes Betriebssystem wie Linux abzusichern, das in seiner Konzeption nicht die Sicherheit als oberstes Entwicklungsziel hatte. Palladiums ungeklärter Zertifizierungsvorgang stellt ein Risiko dar. Hier können Anwendungen ausgeschlossen werden, andere wiederum zertifiziert werden, die Gefahren in sich bergen. Der Anwender wird stark in seinen Möglichkeiten beschränkt, zertifizierte Anwendungen zu untersuchen. Die meist als Hacker-Tools verschrieenen Programme, die Ein-/Ausgaben testen, Speicherabbilder auslesen oder Sniffer, mit den überprüft werden konnte, was an einen Server übertragen wird, sollen geblockt werden. Auch wenn diese Programme tatsächlich auch zum Hacken missbraucht wurden, dienten sie auch zur Anwendungsabsicherung und zur Kontrolle. Wie z.B. kann der Anwender überprüfen, welche privaten Daten an einen Server übertragen werden und welche nicht? Hier wird zu viel Vertrauen in den Hersteller der Anwendung und die Zertifizierungstelle gesteckt, was sich nicht durch unabhängige Dritte überprüfen lässt. Genauso könnte ein nach aussen hin "wohl-zertifizierterÄgent mit vielen Rechten großen Schaden am System anrichten. Der Benutzer hat aufgrund der Einschränkungen dann eventuell nicht mal die Möglichkeiten, diesen Agent mittels Tools einfach zu entfernen, weil der Agent anderen Programmen den Zugriff auf seine Programmteile verwehrt. Wie sich der Nexus gegen solche ïnneren Bedrohungen"verhält, ist unklar. Die Fähigkeit des Reporting stellt eine weitere Gefahr dar. Hat der Anwender keine Kontrolle über dieses "Feature", ist die Anonymität nicht gegeben. Wenn also ein externes System ohne Zustimmung des Benutzer eindeutige Daten über das System bekommen kann, lässt sich z.B. das Surfverhalten gut aufzeichnen. Insgesamt wird für die meisten konstruktiv vorgesehenen Bedrohungen/-Probleme adäquate Mittel bereitgestellt. Dem Problem des allgegenwärtigen Rechnen wird allerdings keine ausreichende Lösung entgegengesetzt. Hat ein Angreifer Zugriff auf ein komplettes System (z.B. Notebook), gewährt es ihm große Angriffsmöglichkeiten, denn die Bindung Software an Hardware(Sealing) besteht. Hier helfen bloß altbekannte Sicherungsverfahren wie ein wirksamer Passwortschutz. Auch ausreichender Schutz vor Hardwareangriffen ist nicht gegeben. Betroffen ist z.B. das TPM, auf dem die Sicherungsmechanismen aufbauen. Der Schutz vor Softwareangriffen konzentriert sich auf das Erkennen von Veränderungen. Was dagegen durch die Konzepte nicht verhindert werden kann, ist, dass Dateien durch Schädlingsprogramme gelöscht werden können.

10.2.6 Beziehungen zu Digital Rights Management(DRM)

Das Thema DRM ist einer der Hauptkritikpunkte an TCPA/Palladium. Obwohl es in letzter Zeit eher bestritten wurde, dass DRM jemals ein Entwicklungsziel der Systeme gewesen ist, deutet sehr viel darauf hin. Wegen heftiger Kritik wird von vielen TCPA-

Befürwortern gerne die Umsetzung eines DRM-System mit Hilfe von TCPA in Frage gestellt. Dagegen bietet TCPA ein gute Grundlage für ein solches System. Wichtigster Ansatzpunkt ist dabei das weiter oben beschriebene "Reporting". Damit ist es möglich, sich gegenüber anderen Systemen zu identifizieren. Mit Hilfe einer Datenbank könnte dann geprüft werden, ob das System in dieser Konfiguration berechtigt ist, einen Film abzuspielen. Der manipulationssichere Speicherbereich könnte zur Speicherung von Keys benutzt werden, die zum Entschlüsseln eines Medium notwendig sind. Da diese Keys durch das TPM geschützt sind, können diese nicht einfach ausgelesen werden und zur Entschlüsselung auf einen anderen Rechnersystem benutzt werden. Mit diesem beiden TCPA-Funktionalitäten eröffnen schon viele Möglichkeiten, solange sie garantiert werden können. Die Garantie steckt hauptsächlich im TPM und sogar TCPA-Befürworter erwähnen (als Indiz für die Nicht-Umsetzbarkeit), dass die aktuelle Version des TPMs Schwächen bei Hardware-Angriffen aufweist. Zwar sind diese Gefahren vorhanden, jedoch ist dies kein starkes Argument, denn erstens ist das TPM gut gegen Software-Angriffe geschützt und eine DRM-Umsetzung auf Basis von TCPA wesentlich sicherer als reine Softwarelösungen und zweitens ist abzusehen, dass in Zukunft TPMs stärker gegen diese Angriffe geschützt werden. Ob nun DRM Entwicklungsziel war oder nicht, auf jeden Fall bieten sich durch TCPA wesentlich bessere Umsetzungsmöglichkeiten als bisher. Und der Bedarf an einem funktionierenden DRM-System ist vorhanden. Besonders die Musik- und Filmindustrie braucht neue Schutzmechanismen, nachdem in der Vergangenheit sich die meisten Techniken (z.B. Verschlüsselung von Filmmaterial auf DVD) einfach um gehen ließen. Aufgrund der riesigen neuen Vermarktungsmöglichkeiten, die ein funktionierendes DRM-System bringen könnte, ist dieses Thema für viele Heimanwender mit Horrorszenarien behaftet, denn es bedeutet für den Anwender Einschränkungen.

10.3 Kritische Betrachtung

Ein sicheres PC-System, dem man vertrauen kann, ist in der Tat ein reizvoller Gedanke. Doch derzeit sieht es nicht so aus, als würde den Befürwortern von „Trusted Computing“ sehr viel Vertrauen entgegengebracht werden. Es gibt sogar einige Kritiker, wie Richard Stallman (Free Software Foundation) die das Kürzel „TC“ bereits in „Treacherous Computing“ umgedeutet haben. Das Verhalten von TCG und NGSCB gegenüber der Öffentlichkeit spielt dabei neben den inhaltlichen Vorstellungen eine nicht ganz unwesentliche Rolle. Schon allein in den häufigen Namens- und Organisationswechseln sowohl von TCG (zuvor TCPA) als auch der Microsoft-Produktbezeichnung NGSCB (zuvor Palladium, Trusted Systems) sieht manch ein Kritiker eine Verschleierungstaktik um vom eigentlichen Ziel abzulenken. Dies passt auch zur bisherigen Informationspolitik. Wenn etwa der Einsatz von Trusted Computing zum Entfernen von Raubkopien von außen einerseits bestritten wird, andererseits eine Dementierung erfolgt, ist dies nicht unbedingt vertrauensbildend. Zumal die einflussreichsten beteiligten Firmen Intel und Microsoft mit bisherigen Vorhaben wie z.B. die Seriennummer im Pentium 3 oder die Datenübertragungen des Microsoft-Betriebssystems „Windows XP“ eher Misstrauen ernteten. Daher basiert die eine oder andere Kritik zum Teil eher auf Vermutungen und Szenarien, als auf tatsächliche Fakten als Bestandteil der Spezifikationen.

10.3.1 Bedeutung für den Privatanwender

Die wichtigste Partei hat bisher von den bevorstehenden Plänen wohl am wenigsten erfahren: der allgemeine Nutzer. Mangels Präsenz des Themas in den Massenmedien wird der Anwender wohl erst mit dem Thema beim PC-Kauf an der Laden-Theke konfrontiert werden. Ob er dann weiß, in wessen Hände er sein Vertrauen legt? Das in Wirklichkeit in erster Linie nicht dem Computer, sondern dem Menschen, der davor sitzt, misstraut wird, ist der Tenor in vielen kritischen Fachartikeln. Der Anwender soll die Macht über seinen Rechner verlieren und an den TCG-Chip mit seinem zentralen Baustein TPM im Inneren des PCs abgeben. Befürworter wie z.B. Microsoft beschwören indes die Vorteile des Nutzers und der Sicherheit seiner Daten. Zudem sei der Chip passiv und könne nicht aktiv in das System eingreifen. Diese Aussage ist sicherlich richtig, da TCG nur die sichere Basis liefert. Nur muss man dies im Zusammenhang mit der Nutzung durch ein entsprechend ausgelegtes sicheres Betriebssystem sehen. Das soll die Fähigkeiten der neuen Architektur nutzen, worin viele Experten Probleme sehen.

10.3.2 Zusammenhang mit Urheberrechte-Kontrolle

Oft bestritten wurde der Zusammenhang zwischen TCG und der Durchsetzung der Rechte der Medien- und Inhalteindustrie mittels Digital Rights Management (DRM). Dennoch sehen viele im Zusammenhang mit einem Betriebssystem eine geeignete Architektur. Die aktuelle Urheberrechte-Diskussion und die damit verbundenen gesetzlichen Veränderungen beispielsweise mit dem Digital Millennium Copyright Act (DMCA) in den USA, und das Vorhaben, den ausschließlichen Verkauf von TCG-fähigen Geräten zu erzwingen und bei Nichtbeachtung rigide Strafen vorzusehen, untermauern diesen Verdacht. US-Senator Fritz Howling, einer der energischsten Vertreter der Unterhaltungsindustrie und Initiator dieses Gesetzes, ist auch der Namensgeber des „Fritz-Chip“, wie das TPM unter Kritikern oft genannt wird. Microsoft könnte NGSCB für DRM benutzen und damit nicht nur der Unterhaltungsindustrie einen Dienst erweisen sondern auch die Raubkopien der eigenen Produkte insbesondere in Fernost eindämmen. Musikdateien z.B. wären an einen Rechner gebunden. Eine weitere Inanspruchnahme der Nutzungsrechte etwa auf einem MP3-Player wäre nicht möglich. Derselbe Mechanismus könnte zur Löschung von unliebsamen Inhalten und damit zur Zensur der Nutzerdaten zum Einsatz kommen. So könnte sich etwa der US-Kryptoexperte Lucky Green vorstellen, dass es mit DRM und NGSCB möglich ist, den über das Internet empfangenen und eigentlich harmlos anmutenden Dateien wie E-Mails oder Word-Dateien auch Anweisungen an das Betriebssystem beizufügen, um nach kritischen Inhalten zu suchen sowie gegebenenfalls den Master Key-Inhaber zu informieren. Die Frage ist also, ob es Software wie z.B. ein Betriebsystem geben könnte, die diese Schlüssel-Informationen dazu benutzt, den Anwender und seine Daten zu zensieren und ihn somit zu entmündigen.

10.3.3 Forderungen nach mehr Transparenz und Kontrolle

Damit ist für Gegner klar, dass TCG im wesentlichen nur kommerzielle Interessen der Medienindustrie verfolgt und möglicherweise auch politisch genutzt werden könnte. Der Chaos Computer Club fordert stellvertretend für die Anwender z.B. die „Kontrolle über alle Schlüssel im TPM“ [3] in der Hand des Nutzers. Dieser solle selbst entscheiden können, welche Schlüssel auf seinem Rechner für welche Zwecke benutzt werden. Wichtig sind dabei vor allem die im TPM versiegelten Schlüssel „Endorsement Key“ (EK) und „Storage Root Key“ (SRK). Wird dem Nutzer der Zugang zu diesen Schlüsseln

gewährt, wäre die Verwendung von DRM nicht mehr möglich. Denn DRM basiert ja darauf, dass der Nutzer die Schlüssel nicht manipulieren kann. Mit der Spezifikation 1.2 der TCG kann der Endorsement Key gelöscht und für ungültig erklärt werden. Damit verbunden ist aber auch der Verlust des Vertrauensystems. Relativ unproblematisch in dieser Schlüssel-Frage ist die Situation, wenn die erworbenen Schlüssel auf einen anderen Rechner migriert werden sollen. Dies ist möglich. Auch der Storage Root Key kann exportiert werden. Nicht jedoch der Endorsement Key. Die Kosten für die Migration von einem alten TPM auf einen neuen stehen hingegen noch nicht fest. Liegt ein Hardware-Defekt vor, ist man vom Hersteller abhängig, um die Schlüssel zu ermitteln. Darüber hinaus ist auch der Verlust der gespeicherten Schlüssel zu beklagen. Ein Daten- und Eigentumsverlust von Dokumenten und gekauften Inhalten ist die Folge.

10.3.4 Bedeutung für Wirtschaft und Wettbewerb

Die Experten des CCC fordern ferner mehr Offenheit hinsichtlich der Zertifizierung von Software im Zusammenhang mit TCG [3]. Dieser Forderung wurde von IBM entgegnet, dass sie jeder Grundlage entbehrt, da die TCG-Spezifikation „keine Zertifizierung von Software vorsehe“ [14]. Die Forderung wäre sicherlich bei Betriebssystem-Herstellern wie Microsoft oder HP (entwickeln ein TCG-fähiges Linux) besser aufgehoben. Beispielsweise ist in Microsofts NGSCB vorgesehen, vertrauenswürdige Software anhand ihrer Zertifikate zu erkennen. Da stellt sich die Frage, wer diese Zertifizierung durchführen soll. Es wird befürchtet, das Microsoft selbst diese Rolle übernehmen möchte. Damit hätte der Software-Konzern, der durch die starke Verbreitung der Betriebssystem-Reihe Windows ohnehin schon eine marktbeherrschende Stellung in der Software-Industrie inne hat, noch mehr Möglichkeiten, diese zu festigen und auszubauen. Schließlich haben sie dann in der Hand, wessen Produkte zertifiziert werden. Konkurrenten in bestimmten für Microsoft interessanten Segmenten könnten hier mit ihren Produkten benachteiligt werden. Es ist abzusehen, dass Microsoft ein Programm nicht offensichtlich wegen dessen Herkunft von der Konkurrenz abweisen würde, da ansonsten sicherlich wettbewerbsrechtliche Konsequenzen die Folge wären. Aber eine Benachteiligung kann auch durch Verzögerung der Zertifizierung geschehen, um somit möglicherweise dem Wettbewerber zwischenzeitlich selbst mit einem eigenen Produkt Marktanteile wegzunehmen. Oder die Zertifizierung ist mit überhöhten Kosten verbunden. Dies ist auch die Gefahr, die besonders Vertreter der OpenSource-Szene sehen. Diese Projekte könnten dann nicht mehr ohne weiteres umgesetzt werden, weil sie nicht finanzierbar wären. Dies gilt insbesondere auch für die zahlreichen Releases die während der Entwicklung entstehen. Gefährdet in ihrer Existenz wären auch freiberufliche Programmierer sowie kleine und z.T. auch mittelständische Unternehmen. Damit besteht die Gefahr, dass der Wettbewerb auf dem Software-Markt dauerhaft nur zwischen den großen „Global Playern“ abläuft. Die Folge für den Kunden wären weniger alternative Produkte und insbesondere weniger meist kostenfreie OpenSource-Software. Daher wird der Ruf nach unabhängigen Zertifizierungsinstanzen und solchen, die nur für freie Software zuständig sind, laut, die diese Machtposition nicht so leicht für ihre Zwecke missbrauchen können. Ebenfalls ein Zensur-Instrument sehen Kritiker in den so genannten Black und White-Listen. Die HCL (mit geprüfter Hardware) und SCL (mit gesperrten Seriennummern) wird konsultiert, um erlaubte Hardware und verbotene Software zu identifizieren. Es wäre damit möglich, bestimmte Hardware z.B. aus Fern-Ost auszuschließen, weil diese etwa nicht als sicher gilt. Und auch Software sowie Dokumente könnten so zensiert werden. Besonders heikel wäre hier auch ein Angriff auf diese Datenbanken. Würde es einem Hacker gelingen, eine

zentrale wichtige Software wie etwa den Bootloader des NGSCB auf die SCL einzutragen, wäre ein Großteil der Datenverarbeitung auf der Welt stillgelegt. Dieses Szenario basiert nicht auf den paranioden Vorstellungen eines Kritikers. Wie die Geschichte zeigt, ist dies durchaus möglich. Microsoft wurde Opfer eines Unbekannten, der in den Besitz eines VeriSign-Zertifikates unter dem Namen „Microsoft Corporation“ kam und damit bei Internet-Nutzern trügerische Sicherheit vorgaukeln konnte [2]. Darüber hinaus hatte man Mitte 2001 vergessen, die Server-Zertifikate für den Passport-Server zu erneuern, wodurch Passport und MSN-Nutzer sich nicht mehr einloggen konnten [19]. Mit TCG wird ein hohes Maß an Sicherheit versprochen, die der einzelne Nutzer aber schwer nachprüfen kann. Er weiß z.B. nicht, was ein vermeintlich sicheres Programm im Hintergrund wirklich tut. Schließlich geht es auch gerade darum, die Debugger daran zu hindern, das Verhalten eines Programmes zu analysieren.

10.3.5 Rolle der Hardware-Hersteller und Folgen für die Nutzer

Dass die Beteiligung großer Hardware-Hersteller wie Intel, AMD, Via und Nvidia nicht nur darauf beruht, den PC-Eigentümern und Nutzern mehr Sicherheit zu bieten, glauben viele Kritiker. Der Bedarf an neuer Hardware sowie die Verbesserung der Performance steigt nicht in dem Maße, so dass auf diesem Sektor neue Funktionalitäten gefragt sind, um die Update-Zyklen der Kunden kurz zu halten. TCG im Zusammenhang mit DRM-geschützten Inhalten könnte die Anwender zwingen, neue Hardware zu kaufen, wenn sie die Inhalte konsumieren möchten bzw. neue Software-Versionen nutzen wollen, die eventuell auch nur noch auf sicheren Systemen ausgeführt werden können. Negativ betrachtet wird oft die dauerhafte Verknüpfung der Hardware mit dem System (engl: dongled). Dies sei notwendig, um die Integrität des Systems zu überprüfen und vor Manipulation zu schützen. Auch wird es für die Authentifizierung und Identifizierung gebraucht. Doch der Nutzer zahlt dies mit der Flexibilität, die er sonst von einem PC verlangt.

10.3.6 Staatliche Sicht

Wo ein Projekt mit so viel Misstrauen begleitet wird, liegt die Frage auf der Hand, ob in der TPM auch „dunkle Kanäle“ [3] eingebaut wurden, um z.B. von staatlichen Sicherheitsorganen (wie Geheimdiensten) oder ggf. auch anderen unautorisierten Parteien benutzt zu werden und auf diese Weise Schlüssel und Daten zu kompromittieren. Solche „Covert Channels“ sind in der TCG-Spezifikation natürlich nicht zu finden und von Seiten der TCG auch nicht beabsichtigt, würde dies doch weiteres Misstrauen säen. Es könne jedoch nicht ausgeschlossen werden, dass ein Hersteller in seiner Implementation davon abweicht. Letztlich sollte eine unabhängige Zertifizierung der Komponenten durchgeführt werden, um diesem Sachverhalt auf den Grund zu gehen. Von offiziellen Stellen der Bundesrepublik Deutschland ist hingegen auch Zurückhaltung gegenüber Trusted Computing zu verspüren. Der Bundesminister für Wirtschaft und Arbeit Wolfgang Clement antwortete auf eine Anfrage aus dem Bundestag, dass die weitreichende Verbreitung von „Palladium“ (nun NGSCB) zu höheren Kosten führen könnte, da Lizenzkosten zugunsten Microsoft zu zahlen wären [15]. In punkto „Trusted Computing“ wurde Klärungsbedarf erkannt und im Juli 2003 ein Symposium im Bundesministerium für Wirtschaft und Arbeit mit Befürwortern und Kritikern durchgeführt [7]. Das „Bundesamt für Sicherheit in der Informationstechnik“ (BSI) begrüßt jegliche Bestrebungen, Bereiche in der Informationstechnik sicherer zu machen. In „allen Anwendungsbereichen [sei ein] starkes Bedürfnis nach größerer Sicherheit“ vorhanden. Es wird aber auch dar-

auf verwiesen, dass der Anwender „selbstbestimmt über alle seine Daten verfügen“ soll und einzelne Wettbewerber nicht ausgegrenzt werden dürfen [6]. Das BSI ist daher auch u.a. mit Microsoft im Kontakt, um Einfluß zu nehmen [5]. Von Seiten der US-Regierung werden die geplanten Sicherheitsmaßnahmen eher als guter Anfang, aber längst nicht als ausreichend angesehen, wie der Berater für Sicherheit im Cyberspace des Weißen Hauses, Richard Clarke verlautbarte.
Ob indes überhaupt die versprochenen Funktionalitäten geboten werden können, fragen sich einige Experten. Die technischen Aspekte wurden bereits im Teil 2 Technische Realisierung behandelt.

10.4 Zusammenfassung

Bis zum gegenwärtigen Zeitpunkt bleibt „Trusted Computing“ stark umstritten. So sehr eine sichere Rechner-Umgebung auch zu begrüßen ist, so wären viele der angepriesenen Funktionalitäten wie Schutz vor Viren und Spam auch heute schon durch Software einschränkbar. Auch die bessere Gestaltung der existierenden Betriebssysteme in punkto Sicherheit wäre ein positiver Beitrag. E-Mail-Würmer wie etwa „Blaster“ sind schließlich durch Lücken in diesen so folgenreich. Wir wissen, dass softwareseitige Absicherung ihre Grenzen haben und eine Betrachtung des Gesamtsystems notwendig ist, wie das die TCG in Angriff genommen hat. Mit der gewählten Konstruktion von Hardware-Veränderungen ist ein System erdacht worden, das durchaus zweckmässig wäre. Auch energische Kritiker, wie CCC-Experte Rüdiger Weis, können mit der nun vorliegenden TPM-Spezifikation und den Veränderungen ”langsam relativ glücklich leben”[13]. Dennoch ist die geäußerte Kritik, auch wenn sie zum Teil auf Spekulationen basiert, berechtigt. Schließlich ist es auch nachwievor nur eine einzige Instanz, die die Sicherheit garantieren soll und von der das System sicherheitstechnisch abhängig und kein verteilter Ansatz.
Wichtig ist nicht nur, das der Nutzer entscheiden kann, ob er TC nutzen möchte oder nicht. Da entsprechend ausgestattete Rechner zunächst mit deaktivierter Sicherheitsfunktion ausgeliefert werden sollen und man auch bei NGSCB die Wahl hat, die Funktionen zu nutzen oder nicht, ist dies zunächst gegeben. Auch wird darauf verwiesen, dass Linux auch weiterhin mit deaktiviertem TCG-Chip funktionieren wird und damit auch freie Software nutzbar sei. Diese Wahlmöglichkeit mag zunächst einleuchten, verharmlost aber die eigentliche Tragweite. Schließlich geht es offenbar u.a. (bzw. vor allem) um die Nutzungskontrolle von Medien. Wer hier TCG abschaltet bzw. alternative Betriebssysteme nutzt, wird womöglich mit Einschränkungen bei der Nutzung seines Equipments leben müssen. So ist denkbar, dass Medien nur in Zusammenhang mit bestimmten Playern mit Rechtekontrolle auf bestimmten Betriebssystemen abspielbar sind. Oder die Kommunikation per E-Mail kann nur eingeschränkt genutzt werden, da viele Kommunikations-Partner E-Mails mit Hilfe von TCG verschlüsseln.
Die Gefahr, dass der Alltag zur Nutzung von TCG zwingt, ist also vorhanden. Daher ist es wichtig, dass dieses Vorhaben auf einer Grundlage realisiert wird, die den Bürger nicht entmündigen. Als einen Schritt in die richtige Richtung wertet auch der CCC die Aussage, dass der Nutzer die Kontrolle über die im TCG-Chip gespeicherten Schlüssel haben wird. Damit wäre eine Rechtekontrolle auf Wunsch nicht mehr möglich.
Für einen abschließenden Ausblick braucht man nicht sehr weit in der Zeit voraus zu schauen. Die Zukunft hat zum Teil schon begonnen. IBM liefert seit 2002 seine ThinkPad-Notebooks mit TCG-Chip aus. Microsoft hat mit einer veränderten End-User License Agreement (EULA) beim Media Player 9 im Juni 2002 u.a. festgelegt, dass Microsoft

jederzeit Softwareaktualisierungen durchführen kann um damit DRM-geschützte Inhalte vor dem abspielen und kopieren zu schützen [16]. Darüber hinaus kann auch weitere nicht erwünschte Software deaktiviert werden. Ferner hat Intel jetzt mit dem „D865GRH“ „das erste einzeln erhältliche Mainboard“ mit TPM-Chip von Infineon vorgestellt [18].

Literaturverzeichnis

[1] Ross Anderson. Trusted Computing Frequently Asked Questions. http://www.cl.cam.ac.uk/ rja14/tcpa-faq.html, Aug 2003.

[2] Jo Bager. Microsoft warnt vor Cracker-Zertifikat. http://www.heise.de/newsticker/data/jo-24.03.01-001/, Mär 2001.

[3] Chaos Computer Club. TCPA - Whom do we have to trust today? http://www.ccc.de/digital-rights/forderungen/, Mär 2003.

[4] Zentrum für Europäische Integrationsforschung. Konferenz: Trusted Computing - Neue Herausforderungen für das deutsche und europäische Wirtschaftsrecht. URL: http://www.zei.de/zei_deutsch/veranstaltung/konf_2003_05_09.htm, Mai 2003.

[5] Bundesamt für Sicherheit in der Informationstechnik. Häufig gestellte Fragen und die dazugehörigen Antworten. http://www.bsi.bund.de/faq/palladium.htm.

[6] Bundesamt für Sicherheit in der Informationstechnik. Sichere Plattformen und die Trusted Computing Group (TCG). http://www.bsi.bund.de/tcg/tcgi0312.htm, Dez 2003.

[7] Bundesministerium für Wirtschaft und Arbeit. Symposium: "Trusted Computing Group (TCG)äm 2. und 3. Juli 2003 (Berlin). http://www.webpk.de/bmwa/willkommen.php.

[8] Christian Stüble Gerald Himmelein. Vertrauensfragen. *c't*, 15:21–22, 2003.

[9] Lucky Green. How will Microsoft respond to Lucky's patent application? URL: http://www.mail-archive.com/cryptography@wasabisystems.com/msg02554.html, Aug 2002.

[10] Trusted Computing Group. *TPM Main 1.2 - Part 1: Design Principles*, Nov 2003.

[11] Trusted Computing Group. *TPM Main 1.2 - Part 2: TPM Structures*, Nov 2003.

[12] Trusted Computing Group. *TPM Main 1.2 - Part 3: TPM Commands*, Nov 2003.

[13] Gerald Himmelein. 20C3: Trusted Computing soll Farbe bekennen. http://www.heise.de/newsticker/data/ghi-28.12.03-000/, Dez 2003.

[14] Gerald Himmelein. Ganz im Vertrauen. *c't*, 15:20, 2003.

[15] Declan McCullagh. Germany cautious on Microsoft security. http://zdnet.com.com/2100-1105-976620.html, Dez 2002.

[16] Christian Persson. Microsoft will Recht zum Direktzugriff auf private PCs. http://www.heise.de/newsticker/data/cp-30.06.02-001/, Jun 2002.

[17] Bruce Schneier. Crypto-Gram Newsletter: Palladium and the TCPA. URL: http://www.schneier.com/crypto-gram-0208.html, Aug 2002.

[18] Christof Windeck. Pentium-4-Mainboard mit TPM. *c't*, 26:30, 2003.

[19] Volker Weber Wolfgang Stieler. Microsoft Server-Zertifikate abgelaufen (Update). http://www.heise.de/newsticker/data/wst-06.05.01-003/, Mai 2001.