

Steffen Wendzel, Johannes Plötner

Praxisbuch Netzwerk-Sicherheit

Risikoanalyse, Methoden und Umsetzung

Auf einen Blick

1	Gefahrenanalyse	35
2	Physikalische Sicherheit	59
3	IT-Grundschutz	75
4	Firewalls und Proxys	85
5	Topologien	137
6	Einbruchserkennung	151
7	Netzwerksicherheit überwachen	197
8	Remote-Access auf Netzwerke	235
9	Penetrationstests und Audits	273
10	Serveraufbau	281
11	Serversicherheit	289
12	Websicherheit	313
13	Secure Shell	335
14	Unix-Sicherheit	355
15	Windows-Sicherheit	381
16	Viren, Würmer, Trojaner & Rootkits	391
17	Disaster Recovery	411
18	Security Policys	431
19	Backups	437
20	Sichere Kommunikation für Benutzer	451
21	Authentifizierungen	471
22	Kerberos	489
23	Sichere Software entwickeln	499
24	Organisationen und Informationen	529
A	Einführung in die Kryptografie	537
B	TCP/IP	577
C	WLAN-Standards nach IEEE 802.11	627
D	kap23.c	631
E	Das Skript vpnstart.sh	633
F	Buffer-Overflow-Exploit	635
G	Glossar	637
H	Literatur	641

Inhalt

Vorwort zur 2. Auflage	21
Einleitung	23

TEIL I: EINFÜHRUNG

1 Gefahrenanalyse 35

1.1 Die Räumlichkeiten	35
1.1.1 Der Serverraum	35
1.1.2 Die Computerarbeitsplätze	36
1.2 Social Engineering	37
1.3 Handys, PDAs & Co.	38
1.3.1 PDAs	39
1.3.2 Handys	40
1.3.3 Speichermedien	41
1.4 Hacker, Cracker & Spione	41
1.4.1 Panikmache	41
1.4.2 Und das Know-how?	42
1.4.3 Zahlen	42
1.4.4 Industriespionage	45
1.5 Viren, Würmer & Trojaner	46
1.5.1 Schädlinge	47
1.5.2 Botnetze	48
1.6 Spam	49
1.7 Selbst in der Verantwortung stehen	49
1.7.1 Ansprechpartner	50
1.7.2 Information	51
1.8 Informationsbeschaffung	53

2 Physikalische Sicherheit 59

2.1 Ein klassisches Unternehmensnetzwerk	60
2.1.1 Zugangskontrollen	60
2.1.2 Das Netzwerk	62
2.2 Wireless LAN	63
2.2.1 Der Standard	63
2.2.2 Sicherheit	64
2.2.3 WEP	64

2.2.4	WPA und VPN	68
2.3	Bluetooth	69
2.3.1	Die Funktionsweise	69
2.3.2	Sicherheitsaspekte	70
2.4	Mobilgeräte	72
2.4.1	Datenverlust durch Diebstahl	72
2.4.2	Gefahr durch Handys	72
2.5	Zusammenfassung	73

3 IT-Grundschutz 75

3.1	Das GSHB des BSI	75
3.1.1	Aufbau	75
3.1.2	Einschränkungen	77
3.2	Das GSHB anwenden	77
3.2.1	Strukturanalyse	78
3.2.2	Schutzbedarf	78
3.2.3	Modellierung	79
3.3	Weitergehende Maßnahmen	81
3.3.1	Sicherheitscheck und -revision	81
3.4	Zusammenfassung	81

TEIL II: SICHERHEIT VON NETZWERKEN

4 Firewalls und Proxys 85

4.1	Grundlagen und Konzepte	85
4.1.1	Default Deny	86
4.1.2	Firewalls	86
4.1.3	NAT und Masquerading	89
4.2	Paketfilter	91
4.2.1	Typische Einsatzgebiete	91
4.2.2	Funktion	92
4.2.3	NAT	93
4.3	Personal Firewalls	93
4.3.1	Szenario 1: auf den Clients eines Firmen-LANs ..	94
4.3.2	Szenario 2: auf einem Heimrechner	94
4.3.3	Absicherung von Clients und Heimrechnern	95
4.3.4	Gefahren	96
4.4	iptables/netfilter für Linux	96
4.4.1	Support im Kernel	98

4.4.2	Die Funktionsweise	99
4.4.3	Beispiele für den Einsatz	101
4.4.4	Iptables im Detail	104
4.5	pf für OpenBSD	114
4.5.1	pf aktivieren	114
4.5.2	pfctl	114
4.5.3	Regeln erstellen	115
4.5.4	pfsync und CARP	121
4.6	IPFilter – eine kurze Einführung	122
4.6.1	Portabilität	122
4.6.2	Filterregeln	123
4.6.3	IPF-Administration	126
4.7	Firewall-Konfiguration mit fwbuilder	126
4.8	Proxyserver	128
4.8.1	Funktion	129
4.8.2	Einsatz	130
4.8.3	Beispiel: Squid unter Linux	131
4.9	Zusammenfassung	136

5 Topologien 137

5.1	Switches oder Router?	137
5.1.1	Hubs	137
5.1.2	Switches	138
5.1.3	Repeater	139
5.1.4	Bridges	140
5.1.5	Router	140
5.2	Bastion Hosts	141
5.3	DMZ	142
5.3.1	Aufbau	142
5.3.2	Ein Beispiel	144
5.4	Terminal-Server vs. Desktops	145
5.5	Honeypots	147
5.5.1	Argumentation	148
5.5.2	Honeypot-Farm	150
5.6	Zusammenfassung	150

6 Einbruchserkennung 151

6.1	Honeypots in der Praxis	151
6.1.1	Honeyd	151

6.1.2	Honeytokens	155
6.2	Darknets	157
6.2.1	Welche Software?	159
6.3	Linux-Syslog-Server	159
6.3.1	Motivation	159
6.3.2	syslogd	160
6.3.3	Konfiguration des Servers	161
6.3.4	Konfiguration der Clients	162
6.3.5	logrotate	163
6.3.6	logcheck	164
6.3.7	Sicherheitsaspekte	165
6.3.8	Ausblick: syslogd-ng	167
6.4	Intrusion Detection	167
6.4.1	Mögliche Implementierungen von IDS	169
6.5	Snort	171
6.5.1	Aufbau der Intrusion Detection	174
6.5.2	snort.conf	174
6.6	Dateisystem-Intrusion-Detection	183
6.7	mtree	184
6.7.1	Ein Abbild des Dateisystems erstellen	184
6.7.2	Auf Veränderungen überprüfen	186
6.8	Prozesse überwachen	187
6.8.1	Unix-Prozess-Accounting	188
6.8.2	FUPIDS	188
6.8.3	fupids2	190
6.8.4	SID	190
6.8.5	systrace	191
6.8.6	IDMEF	191
6.9	Intrusion Prevention	191
6.10	systrace	192
6.11	Intrusion Response	194
6.12	Zusammenfassung	195

7 Netzwerksicherheit überwachen 197

7.1	Netzwerkmonitoring mit Nagios	199
7.1.1	Die Installation	200
7.1.2	Die Konfiguration	204
7.1.3	Die Plugins	211
7.2	Nmap: Der wichtigste Portscanner	214
7.2.1	Prinzip eines Portscanners	214

7.2.2	Techniken des Scannens	215
7.2.3	Weiterer Informationsgewinn	221
7.2.4	Nmap in der Praxis	223
7.3	Nessus: Ein Security-Scanner	227
7.3.1	Die Installation	228
7.3.2	Die Konfiguration	229
7.3.3	Nessus benutzen	230
7.4	Sniffer	231
7.4.1	tcpdump	232
7.4.2	Wireshark (ehemals ethereal)	233
7.4.3	dsniff	233
7.5	Zusammenfassung	234

8 Remote-Access auf Netzwerke 235

8.1	Remote Desktop, VNC & Co.	235
8.1.1	Terminalserver	236
8.1.2	RDP und VNC	236
8.2	Tunneling und VPNs	237
8.2.1	VPN-Typen	238
8.2.2	Datenverschlüsselung	239
8.2.3	Tunneling	240
8.2.4	Artikel: Firewalls umgehen	242
8.3	IPSec	251
8.3.1	Die Tunnel-Modi	252
8.3.2	Authentication Header	253
8.3.3	ESP	254
8.3.4	IKE	256
8.3.5	Security Assoziationen	257
8.3.6	IPSec Security Policys	257
8.3.7	IPSec Praxis: Solaris	258
8.3.8	IPSec Praxis: OpenBSD	262
8.3.9	Support unter Windows	266
8.4	VPNs mit OpenVPN	266
8.4.1	Pre-shared Keys	267
8.4.2	Zertifikate mit OpenSSL	269
8.4.3	OpenVPN als Server einrichten	270
8.4.4	OpenVPN als Client	271
8.5	Zusammenfassung	272

9 Penetrationstests und Audits 273

- 9.1 Was sind Penetrationstests? 273
- 9.2 Security Audits 274
- 9.3 Bevor es los geht 275
- 9.4 Die Durchführung 276
 - 9.4.1 Aufteilung der Tätigkeiten 276
- 9.5 Der Abschlussbericht 277
- 9.6 IT Grundschutz 278
- 9.7 Zusammenfassung 278

TEIL III: SYSTEMSICHERHEIT

10 Serveraufbau 281

- 10.1 Der Aufbau der Hardware 281
 - 10.1.1 USV 282
 - 10.1.2 RAID 283
 - 10.1.3 Backups 284
- 10.2 Der Serverraum 285
 - 10.2.1 Brandschutz 285
 - 10.2.2 Klimaanlage 285
 - 10.2.3 Zutritt 286
- 10.3 Die Entsorgung 286
 - 10.3.1 Alte Festplatten 286
 - 10.3.2 Kopier- und Faxgeräte 286
- 10.4 Zusammenfassung 287

11 Serversicherheit 289

- 11.1 Banner 289
 - 11.1.1 Die Lösung des Problems 291
- 11.2 Nameserver-Absicherung 292
 - 11.2.1 Restricted Zonetransfers 293
 - 11.2.2 Hochverfügbarkeit 294
 - 11.2.3 Versionsnummer verstecken 294
 - 11.2.4 Serverzugriff 294
- 11.3 SSH-Absicherung 295
 - 11.3.1 SSH devicespezifisch 295
 - 11.3.2 Root-Login 296
 - 11.3.3 Listen-Port 296

11.3.4	Leere Passwörter	297
11.3.5	Benutzer-Wrapper	297
11.3.6	Group-Wrapper	297
11.3.7	Zugriff nur über Key oder auch über Passwort? ..	297
11.3.8	Weitere Möglichkeiten der Authentifizierung	298
11.4	X11-Absicherung	298
11.4.1	xhost	298
11.4.2	Keine Remoteverbindungen	299
11.4.3	Weitere Möglichkeiten	299
11.5	Absicherung des Network Filesystems	299
11.5.1	Absicherung der zu exportierenden Verzeichnisse	300
11.5.2	Weitere Möglichkeiten zur Absicherung	302
11.6	NIS-Absicherung	302
11.7	FTP-Absicherung	303
11.7.1	Das Protokoll	303
11.7.2	chroot	304
11.7.3	Quotas und Speicherlimits	304
11.7.4	Anonymous-Zugriff	304
11.7.5	Authentifizierung	307
11.7.6	Verschlüsselung	307
11.8	DHCP-Absicherung	308
11.8.1	Maßnahmen zur Absicherung	308
11.9	E-Mail-Absicherung	309
11.9.1	Authentifizierung	309
11.9.2	SMTP, POP3, IMAP	310
11.9.3	MailScanner: Spam- und Viruscheck auf SMTP-Servern	311
11.10	Zusammenfassung	312

12 Websicherheit 313

12.1	Webserver-Absicherung	314
12.1.1	Authentifizierung mit htaccess	314
12.1.2	Verwendung von HTTPS	316
12.1.3	mod_security	318
12.1.4	Umgebungsvariablen im Apache	320
12.1.5	Weitere Möglichkeiten zur Absicherung	321
12.2	Microsoft IIS absichern	321
12.3	nikto	322
12.4	PHP-Sicherheit	323

12.4.1	Variablen	324
12.4.2	Referer und Co.	326
12.5	Cross-Site-Scripting	327
12.6	CSS und Cookies	329
12.7	Cookies	330
12.8	Cross-Site-Authentication (XSA)	330
12.9	Angriffe auf Proxyserver und Datenbanken	331
12.9.1	Proxyserver	331
12.9.2	Datenbanken	332
12.9.3	SQL-Injection	333
12.10	Zusammenfassung	334

13 Secure Shell 335

13.1	Das Protokoll	336
13.1.1	SSH Protokoll 1	336
13.1.2	SSH Protokoll 2	336
13.2	Konfiguration eines OpenSSH-Servers	337
13.2.1	Die /etc/ssh/sshd_config	337
13.3	SSH nutzen	341
13.3.1	Remote-Login	341
13.3.2	Secure Copy	343
13.3.3	Authentifizierung über Public-Key-Verfahren	345
13.3.4	Secure File Transfer	349
13.3.5	X11-Forwarding	350
13.3.6	SSH-Port-Forwarding	351
13.4	Zusammenfassung	353

14 Unix-Sicherheit 355

14.1	Access Control	355
14.2	Benutzer, Passwörter, Gruppen	357
14.3	Trojanische Pferde	359
14.4	Logging	359
14.4.1	Bei der Analyse Zeit sparen	360
14.5	Partitionierungen	360
14.6	Restricted Shells	361
14.7	su und sudo	362
14.8	chroot	363
14.9	Patches	364
14.9.1	OpenBSD	364

14.9.2 Solaris	366
14.9.3 Linux	368
14.10 LKMs	369
14.11 Stealth Interfaces	370
14.12 Dateisystemverschlüsselung	370
14.12.1 Swap-Verschlüsselung	371
14.12.2 Dateisystemverschlüsselung unter Linux	371
14.13 Kernel-Erweiterungen und gcc-propolice	373
14.13.1 gcc propolice	374
14.13.2 SeLinux und SeBSD	375
14.13.3 OpenWall (OWL)	375
14.13.4 grsecurity	376
14.13.5 PaX	376
14.14 Sichere Derivate und Distributionen	376
14.14.1 Trusted Solaris (jetzt Teil von Solaris)	377
14.14.2 OpenBSD	377
14.14.3 TrustedBSD	377
14.14.4 Hardened Gentoo	378
14.14.5 OpenWall	378
14.14.6 Adamantix	378
14.14.7 Hardened Linux	378
14.15 Zusammenfassung	379

15 Windows-Sicherheit 381

15.1 Dienste, die die Welt nicht braucht	381
15.2 Patches	383
15.3 BIOS-Bootreihenfolge	383
15.4 Standardfreigaben deaktivieren	384
15.4.1 NetBIOS und SMB	385
15.5 Deaktivieren der Druckerfreigabe	386
15.6 Zugriff auf Wechselmedien	386
15.7 Benutzer-Accounts	386
15.8 Unterschiedliche Passwörter im Netzwerk	386
15.9 Partitionierung	387
15.10 Sicherheit im Dateisystem	387
15.11 Sicherheitsfeatures und Systemversion	388
15.12 Zusammenfassung	389

16 Viren, Würmer, Trojaner & Rootkits 391

- 16.1 Viren 391
 - 16.1.1 Grundlegende Funktionsweise 392
 - 16.1.2 Viren kennen mehr als nur Win32! 393
 - 16.1.3 Schutz vor Viren 394
- 16.2 Würmer 395
- 16.3 Rootkits 396
 - 16.3.1 Rootkit-Geschichte 397
 - 16.3.2 Rootkit Technologien 398
 - 16.3.3 Backdoors und Malware 402
 - 16.3.4 Firewalls und Hidden Channels 404
 - 16.3.5 Rootkits entdecken 404
- 16.4 Angriffe verhindern 410
- 16.5 Zusammenfassung 410

17 Disaster Recovery 411

- 17.1 Problemdefinition 411
- 17.2 Ursachenanalyse 411
 - 17.2.1 Serverprobleme 412
 - 17.2.2 Hackerangriffe 412
- 17.3 Forensik 413
 - 17.3.1 Erste Schritte 413
 - 17.3.2 Forensik-Tools 414
 - 17.3.3 Backdoors 415
 - 17.3.4 Rootkits 417
 - 17.3.5 Nach einem Angriff 420
 - 17.3.6 Ein Blick über die Schulter 422
 - 17.3.7 Letzte Worte zur Forensik 426
- 17.4 Server wieder aufsetzen 427
 - 17.4.1 Neuinstallation? 427
 - 17.4.2 Backups 427
- 17.5 Zusammenfassung 428

TEIL IV: DATENSICHERHEIT UND -INTEGRITÄT

18 Security Policys 431

- 18.1 Gute Vorsätze für das neue Jahr! 431
- 18.2 Der Inhalt 432

18.2.1	Festlegung der Dokumentationsstruktur	432
18.2.2	Überwachungssystem und -intervalle	432
18.2.3	Autorisierungs- und Authentifizierungsmaßnahmen	432
18.2.4	Weitere Möglichkeiten	433
18.3	Zusammenfassung	435

19 Backups 437

19.1	Die Backupstrategie	438
19.2	Backuparten	441
19.2.1	Vollständiges Backup	441
19.2.2	Inkrementelles Backup	441
19.2.3	Datenbackup	441
19.2.4	Systembackup	442
19.2.5	Backupmedien	442
19.2.6	Komprimieren	443
19.3	BackupPC	443
19.3.1	Das Projekt	444
19.3.2	Die Installation	446
19.3.3	Die Konfiguration	447
19.3.4	Bedienung	448
19.4	Zusammenfassung	449

20 Sichere Kommunikation für Benutzer 451

20.1	Richtige Kommunikation	451
20.1.1	Übersehen und Ignorieren von E-Mails	451
20.1.2	Von sich auf andere schließen	451
20.1.3	Romane statt kurzer Mails	452
20.1.4	Im Ton vergriffen	452
20.1.5	Mails an alle	453
20.1.6	Grammatik und Rechtschreibung	453
20.2	E-Mail-Standards	453
20.2.1	SMTP	453
20.2.2	POP3	456
20.3	Sichere Übertragung: SSL/TLS	458
20.3.1	Das Protokoll	458
20.3.2	SSL in der Praxis: HTTPS	459
20.4	Mails verschlüsseln: PGP und S/MIME	461
20.4.1	PGP/GPG	461

20.4.2	S/MIME	468
20.5	Zusammenfassung	469

21 Authentifizierungen 471

21.1	Die grundlegendsten Möglichkeiten	471
21.2	Biometrische Authentifizierung	472
21.3	S/Key	474
21.3.1	S/Key aktivieren	475
21.3.2	Benutzerinitialisierung	475
21.3.3	Anmelden am System	477
21.4	PAM	477
21.4.1	Die Datei /etc/pam.conf	479
21.4.2	Die Module	482
21.4.3	Linux-PAM	482
21.4.4	Entwickeln mit der PAM-Library	483
21.5	Authentifizierung via NIS, NIS+ und LDAP	487
21.6	Zusammenfassung	488

22 Kerberos 489

22.1	KDC und Tickets	489
22.2	Master-Server und Realms	491
22.3	Konfiguration des Kerberos-Servers und des KDC	491
22.3.1	Konfiguration der krb5.conf	491
22.3.2	Konfiguration der kdc.conf	493
22.3.3	Erstellung einer Datenbank	493
22.3.4	Administrative Rechte klären	494
22.3.5	Starten des KDC und des Master-Servers	495
22.3.6	Principals hinzufügen	495
22.4	Konfiguration des Clients	496
22.4.1	kinit	496
22.4.2	Verwalten von Kerberos-Tickets	497
22.5	Kerberos unter Windows	497
22.6	Zusammenfassung	498

23 Sichere Software entwickeln 499

23.1	Sicheres Design	499
23.1.1	Der Kreislauf der Softwaresicherheit	500

23.2	Buffer-Overflows	501
23.2.1	Eine Einleitung	501
23.2.2	Begriffserklärungen	502
23.2.3	Ein Beispielangriff	505
23.2.4	Vermeiden von Buffer-Overflows	509
23.2.5	Weitere Funktionen	510
23.3	Return to Libc	511
23.4	Formatstring-Overflows	511
23.5	Heap-Overflows	514
23.6	Integer-Overflows	517
23.6.1	Der Overflow	518
23.6.2	Vorsicht ist geboten: Multiplikation	519
23.7	Race Conditions	521
23.7.1	File-Locking	521
23.7.2	Atomare Operationen – Teil 1	522
23.7.3	Atomare Operationen – Teil 2	523
23.8	Off-by-One	525
23.9	Integer-Promotions	526
23.10	Abschließende Worte	527
23.10.1	(Symbolische) Links	527
23.10.2	Signale	527
23.11	Zusammenfassung	528
24	Organisationen und Informationen	529
24.1	ICANN	529
24.2	RIPE	529
24.3	IETF	530
24.4	ETSI	530
24.5	BSI	530
24.6	CERT/CC	531
24.7	HERT	531
24.8	Securityfocus	531
24.9	Newsgroups	532
Anhang	535
A	Einführung in die Kryptografie	537
A.1	Motivation	537
A.1.1	Kryptografie	537
A.1.2	Kryptoanalyse	538

	A.1.3	Kryptologie	542
	A.1.4	Steganografie	542
A.2		Grundlagen	544
	A.2.1	Symmetrische Verschlüsselungen	544
	A.2.2	Asymmetrische Verschlüsselungen	552
	A.2.3	Hashes	553
	A.2.4	Digitale Signaturen	554
A.3		DES	557
	A.3.1	Die Geschichte	557
	A.3.2	Der Algorithmus selbst	558
	A.3.3	Sicherheitsfragen	560
	A.3.4	Und doch: DES in der Praxis	561
	A.3.5	Weitere wichtige symmetrische Algorithmen	561
A.4		RSA	563
	A.4.1	Schlüsselerzeugung	563
	A.4.2	Ver- und Entschlüsseln mit RSA	564
	A.4.3	Die Sicherheit von RSA	566
	A.4.4	RSA in der Praxis	566
	A.4.5	Weitere wichtige asymmetrische Algorithmen	567
A.5		MD5	567
	A.5.1	Das Verfahren	567
	A.5.2	Sicherheit von MD5	568
	A.5.3	MD5 in der Praxis	569
	A.5.4	Weitere populäre Hashverfahren	570
A.6		Kryptografie und Politik	570
	A.6.1	USA	571
	A.6.2	Wassenaar	574
	A.6.3	Kryptografie in Deutschland	574
A.7		Zusammenfassung	576
B		TCP/IP	577
	B.1	Grundlegendes zu TCP/IP	577
		B.1.1 Network-Access-Layer	579
		B.1.2 Internet-Layer	579
		B.1.3 Transport-Layer	581
		B.1.4 Application-Layer	582
	B.2	Das OSI-Modell	583
	B.3	Die wichtigen Protokolle	585
	B.4	ARP	585
		B.4.1 ARP-Spoofing	587
		B.4.2 Reverse-ARP	587
		B.4.3 Proxy-ARP	587

B.5	IP	588
	B.5.1 Der IP-Header	589
	B.5.2 Fragmentierung	595
	B.5.3 MTU	596
	B.5.4 Sicherheit	598
B.6	ICMP	598
	B.6.1 ICMP-Type 0 und 8	599
	B.6.2 ICMP-Type 3	600
	B.6.3 ICMP-Type 4	602
	B.6.4 ICMP-Type 5	602
	B.6.5 ICMP-Type 9 und 10	602
	B.6.6 ICMP-Type 11	603
	B.6.7 ICMP-Type 12	603
	B.6.8 Weitere ICMP-Typen	604
B.7	IGMP	604
	B.7.1 Der IGMPv2-Header	604
	B.7.2 IGMPv3	605
B.8	IPv6	606
	B.8.1 Die Verbreitungsproblematik	606
	B.8.2 Der IPv6-Header	607
	B.8.3 Extension-Header	608
	B.8.4 Sicherheit von IPv6	610
B.9	ICMPv6	611
	B.9.1 ICMPv6-Typen	611
	B.9.2 Sicherheit von ICMPv6	613
B.10	Das UDP-Protokoll	613
	B.10.1 Der UDP-Header	613
	B.10.2 Sicherheitsaspekte	614
B.11	Das TCP-Protokoll	615
	B.11.1 Reliability	616
	B.11.2 Sende- und Empfangspuffer	617
	B.11.3 Flow-Control	617
	B.11.4 Header	618
	B.11.5 Grundlegendes zur Datenkommunikation	620
	B.11.6 TCP: Sicherheitsaspekte	623
B.12	Weitere Protokolle	624
B.13	Routing	624
B.14	Zusammenfassung	625
C	WLAN-Standards nach IEEE 802.11	627
D	kap23.c	631
E	Das Skript vpnstart.sh	633

Inhalt

F	Buffer-Overflow-Exploit	635
G	Glossar	637
H	Literatur	641
	Nachwort	647
	Index	649

Einleitung

Wir wollen uns in diesem Buch mit dem Thema »**praktische Netzwerksicherheit**« auseinandersetzen. Wenn Sie nach deutschsprachigen Büchern Ausschau halten, die sich ebenfalls daran versuchen, werden Sie **meistens** entweder bei den Büchern fündig, die sich völlig auf »Script-Kiddies« konzentrieren, oder bei denen, die die Einstiegsthemen nicht auslassen wollen, damit die Seitenzahl ansteigt.¹

Warum also dieses Buch? Stellen Sie sich einen Architekten vor, der Gebäude plant. Er wird die Statik hoffentlich korrekt berechnen, wird dem Brandschutz Rechnung tragen und im günstigsten Fall dem neuen Haus auch noch zu einem hübschen Erscheinungsbild verhelfen. Und nun stellen Sie sich eine Welt vor, in der die Häuser ihre Einganstüren im dritten Stock haben, in der die Aufzüge sich waagrecht im Haus bewegen und in der es offenkundig kein Konzept bei der Planung gab.

Diese surreal anmutende Welt ist nun leider Realität im Bereich IT-Sicherheit. Es gibt die bekannte Theorie, die oft gebetsmühlenartig wiederholt und regelrecht gepredigt wird, und es gibt die Praxis, in der ein IT-System erst einmal *funktionieren* und vor allem wenig *kosten* soll. Unser Ziel ist es, Sie zu einem »Architekten« zu machen, der preiswerte, funktionale – und wenn Sie so wollen: sichere – »Häuser« respektive IT-Systeme bauen kann.

Theorie vs. Praxis

Unser Ziel ist also ein Mittelweg. Wir werden Sie nicht mit Themen wie »Hilfe, mein PC wurde von einem bösen Hacker mit einem Virus infiziert« langweilen, aber trotzdem ein Buch für Einsteiger und Fortgeschrittene gleichermaßen schreiben. Genau wie die Zielgruppe ist auch die Zielsetzung des Buches nicht ganz homogen: Als »Architekt« für IT-Sicherheit können Sie einerseits komplexe Systeme und Umgebungen *analysieren* und Schwachstellen finden, andererseits können Sie aber auch solche Systeme konstruktiv *planen* und *umsetzen*.

Auch Architekten machen Fehler. Man erinnere sich an das Terminal des Pariser Flughafens, das plötzlich in sich zusammenfiel, oder an die amerikanische Hängebrücke, die sich bei einem Orkan aufschaukelte, weil sie dummerweise in Resonanz mit den Windböen geraten ist. Die Brücke schaukelte sich immer weiter auf, bis sie zusammenstürzte. Auch Sie und

¹ Es gibt natürlich einige Ausnahmen, auf die wir an entsprechenden Stellen verweisen. Zudem gibt es sehr gute Bücher für einzelne Spezialthemen der IT-Sicherheit.

wir machen Fehler. Und das Schlimmste: Es gibt keinen Königsweg und kein Allheilmittel zu allumfassender, ewiger Sicherheit.

**Tunnelblick &
Betriebsblindheit**

Schon allein aus philosophischen Gründen ist so ein Zustand unerreichbar. Das Einzige, was wir Ihnen immer wieder mit auf den Weg geben möchten, ist, einen offenen Geist zu bewahren. Die Feinde der Sicherheit sind der Tunnelblick und die Betriebsblindheit.

Da hat doch zum Beispiel ein Programmierer der Software eines Jumbojets festgelegt, das beim Abstellen *eines* Triebwerkes auch alle anderen Triebwerke deaktiviert werden. Das macht auch Sinn, schließlich werden die Triebwerke ja am Boden abgeschaltet, sobald das Flugzeug gelandet ist. Oder nicht? Nun ja, dieses Problem fiel erst auf, als in der Luft einmal ein Triebwerk zu brennen anfang und der Pilot dieses Triebwerk gezielt abschalten wollte.

Man könnte Tausende solcher Beispiele bringen, und dies werden wir im Verlauf des Buches auch tun. Jedenfalls sollten Sie nach der Lektüre dieses Buches das grundlegende praktische Wissen und etwas Handwerkszeug haben, die Ihnen zugetragenen oder auffallenden Probleme und Risiken in den Griff zu bekommen.

Unsere Philosophie

Sicherheit ist ein extrem umfangreiches und teilweise kompliziertes Thema. Es ist also nicht verwunderlich, dass so ziemlich jedes Buch andere Schwerpunkte setzt und auch die jeweiligen Begriffe anders definiert. Es kann also durchaus passieren, dass Sie in anderer Literatur auf leicht andere Aussagen und Definitionen stoßen. Lassen Sie sich dadurch nicht irritieren, und bilden Sie sich Ihre eigene Meinung.

Wenn Sie im Zuge Ihrer weiteren Beschäftigung mit dem Thema auf fachliche Fragen oder Probleme stoßen, geben Sie nicht auf. Versuchen Sie es weiter, stellen Sie uns Fragen (s. Nachwort), stellen Sie Fragen in verschiedenen Internetforen, posten Sie Ihre Probleme in Newsgroups – aber geben Sie nicht auf. Als Hilfe zum besseren Verständnis und zur weiterführenden Recherche haben wir uns außerdem um ein umfangreiches Literaturverzeichnis bemüht, das wir Ihnen ausdrücklich ans Herz legen wollen.

100 % Sicherheit?

Die Unternehmen verlangen den perfekten Sicherheitsexperten und möglichst 100-prozentige Sicherheit. Einige Dienstleister verkaufen 100-prozentige IT-Sicherheit, andere wiederum glauben, diese zu haben. Aber genauso klar ist auch, dass es so etwas nicht geben kann.

Experten

Nehmen wir zum Beispiel moderne Software: Jede Software, die aus ein paar tausend Zeilen Sourcecode besteht, hat irgendwelche Bugs, die irgendjemand finden und ausnutzen kann. Und bloß, weil der Hersteller und Ihre IT-Abteilung davon nichts wissen, heißt das nicht, dass sich nicht bereits jemand über diesen Bug Zugang zu Ihrem System verschafft haben könnte. In den sprichwörtlichen tausend Zeilen Code gab es übrigens 1977 circa 7 bis 20 »Defekte«, also wie auch immer geartete Fehler und Bugs, während 1994 nur noch 0,7 bis 0,05 Defekte zu verzeichnen waren – ein positiver Trend, der sich sicherlich auch bis heute noch fortgesetzt hat. Damit Sie allerdings wissen, in welchen Dimensionen wir uns hier bewegen, sollten Sie wissen, dass heutige Software-Systeme aus mehreren Millionen Zeilen Quellcode bestehen. Hier zwei Beispiele.

- ▶ Man schätzt den Umfang des Windows-Betriebssystems auf etwa 30 bis 60 Mio. Zeilen Quellcode.
- ▶ Zum Vergleich: Ein OpenBSD 4.0-Current System (aufgesetzt im August 2006) besteht derzeit aus etwas über 2,3 Mio. Zeilen Code, also aus deutlich weniger als beim Windows-System. Zudem wird ein Großteil von diesem Code im normalen GENERIC-Kernel nicht einmal verwendet!²

Zudem befinden wir uns, was die Sicherheit anbelangt (mit Ausnahme einiger Experten), in einer Know-how-Wüste. Nun sollte man nicht verlangen, dass jeder alles kann. Jemand, der niemals mit Kerberos oder Windows arbeitet, benötigt auch keine Kenntnisse auf diesen Gebieten. Wer jedoch der Meinung ist, Experte für ein Gebiet, etwa der Absicherung von Linux, zu sein, sollte nicht nur über die bloße Absicherungspraxis selbst, sondern auch über das Hintergrundwissen dafür verfügen. Im Falle von Linux sollte man sich im Optimalfall auch einmal mit der Systemprogrammierung, eventuell der Kernelprogrammierung und der Konzeption des Kernels auseinandersetzen.

² Den Test haben wir mit einem `wc -l `find /sys -name '*.chly'|sort` durchgeführt. Zum direkten Vergleich mit Windows sollte evtl. noch der User-space-Code mit eingerechnet werden. Allerdings wissen wir nicht, ob in den 30 bis 60 Mio. Zeilen des Windows-Codes auch der entsprechende Windows User-space-Code enthalten ist.

Betriebssysteme und deren Software

Die Frage nach dem Betriebssystem ist sozusagen eine der Gretchenfragen der IT-Sicherheit, immerhin handelt es sich für den einen oder anderen auch um eine Art Religion.

Windows vs.
Linux?

Schließlich gibt es zum Teil gravierende Unterschiede, wenn man einen Server oder eine ganze Infrastruktur plötzlich mit Linux statt mit Windows aufbaut. Andererseits gibt es auch viele Gemeinsamkeiten: So gibt es den Webserver »Apache« für alle möglichen und unmöglichen Betriebssysteme, genau wie das Einbruchererkennungssystem »Snort«.

Angesichts der Fakten haben wir uns entschieden, uns auf freie und portable Software zu konzentrieren. Dabei werden wir den Fokus auf Unix-artige Betriebssysteme wie Linux lenken, ohne Windows dabei aus dem Blick zu verlieren. Das hat vor allem zwei Gründe: Zum einen sind wir »Experten« im Bereich Linux/Unix, und zum anderen wollten wir erreichen, dass die vorgestellten Lösungen nicht unnötig Geld kosten.

Schließlich werden wir nicht dafür bezahlt, Werbung für einzelne Hersteller zu machen. Stattdessen setzen wir voraus, dass Sie selbstständig genug sind, die hier vorgestellten Lösungen kritisch zu analysieren. Und nichts wäre uns am Ende lieber, als wenn Sie mit der einen oder anderen hier angepriesenen Variante am Ende aus handfesten Gründen heraus unzufrieden wären und selbst eine passendere und vielleicht sogar sicherere Lösung umsetzen oder auch einkaufen könnten. Denn dann hätte das Buch sein Ziel erreicht.

Eigeninitiative

Sie werden merken, dass wir in diesem Buch auch sehr selten auf wirklich konkrete Fehler und Fehlermeldungen eingehen. Stattdessen verlieren wir uns manchmal in philosophischen Diskussionen und theoretischen Konzepten, die Sie aber letztendlich weiter bringen als wertloses Detailwissen. Denn sollten Sie nicht selbst herausfinden, wie Sie unter Windows XP einen neuen Benutzeraccount anlegen, dann haben Sie mit diesem Buch ein Problem. Schließlich sind wir nicht beim »Malen nach Zahlen«, sondern wollen ein Thema zumindest halbwegs seriös behandeln. Wir werden Ihnen also maximal erklären, warum eine Benutzerverwaltung sinnvoll ist und wie diese vielleicht an einem Beispiel realisiert wird. Für Ihren konkreten Fall werden Sie also immer noch eine gewisse Eigeninitiative brauchen.

Was ist IT-Sicherheit?

Die IT-Sicherheit ist ein sehr komplexes Thema mit vielen wissenschaftlichen Teilgebieten, etwa der Kryptografie. IT-Sicherheit beschäftigt sich mit der Absicherung von Daten und dem Schutz vor ungewolltem Zugriff.

Heutzutage verfügt zumindest jedes mittelständische Unternehmen über eine gewisse IT-Infrastruktur. Diese besteht zum Beispiel aus diversen Sicherheitsrichtlinien, einigen Firewalls, den Workstations der Arbeitnehmer, den Servern, die für den Betrieb notwendigen Dienste bereitstellen, computergesteuerten Produktionssystemen, Terminal-Systemen, ...

Prinzipiell gibt es verschiedenste Ansätze, »Sicherheit« für diese Infrastruktur zu definieren. Viele Leute beschränken sich dabei mit dem Schutz vor Angreifern rein auf die Softwareseite, und vergessen so eine ganzheitliche Betrachtung der Thematik. Aus diesem Grund definieren wir IT-Sicherheit wie folgt:

Viele Ansätze

Der Begriff »IT-Sicherheit« bezeichnet einen Zustand, in dem die IT-Infrastruktur exakt den Zweck erfüllt, für den sie vorgesehen ist.

Diese Definition ist nun recht problematisch. Es stellen sich viele Fragen, und es ergeben sich viele Implikationen:

- ▶ Zuerst einmal müssen der Zweck und die Aufgabe der IT-Infrastruktur genau bekannt und auch definiert und irgendwo festgehalten sein.
- ▶ Damit die IT-Infrastruktur ihren Zweck erfüllen kann, muss sie in erster Linie einmal »laufen«. Im nächsten Schritt muss sie für ihre Aufgabe konfiguriert werden, denn kein System ist *out-of-the-box* einsatzbereit – selbst wenn alles auf Anhieb »funktioniert«.
- ▶ Die Anforderung der Definition beschränkt sich nicht auf die eingesetzte Hard- und Software – die Benutzer müssen diese auch richtig nutzen (können).
- ▶ Es können nur entsprechend autorisierte Benutzer die verschiedenen Dienste nutzen.
- ▶ Der Datenbestand ist sicher – sowohl vor fremdem Zugriff als auch vor Verlust.

Wie man sieht, ergibt sich eine Fülle von Themen, die es in einem Buch über IT-Sicherheit anzusprechen gilt. Der viel zitierte *Hacker* ist da nur ein – und nicht unbedingt das allerwichtigste – Szenario in einer komplexen Betrachtungsweise der IT-Sicherheit.

Geht man von dieser Definition aus, so könnte man die Aufgaben der IT-Sicherheit wie folgt umreißen:

► **Integrität der Daten**

Die Daten müssen vor unautorisierter Manipulation geschützt werden. Ein Beispiel hierfür sind Zugriffsrechte in Dateisystemen.

► **Authentizität**

Wer ist wer

Es muss sichergestellt werden, dass ein Objekt oder Subjekt, welches Daten modifizieren möchte, auch tatsächlich dasjenige ist, für das es sich ausgibt. Dies wird beispielsweise durch Logins erreicht, bei denen der Benutzer seinen Namen und ein zugehöriges Passwort angeben muss, um auf Daten zugreifen zu können.

► **Verbindlichkeit**

Es muss nachweisbar sein, dass der Zugriff auf Daten erfolgte. Dies ist beispielsweise durch die Protokollierung von Zugriffen und Unterschriften möglich.

► **Vertraulichkeit**

Das Preisgeben der Daten an Unbefugte ist zu verhindern, zum Beispiel mit der Unterbringung von Verträgen in einem Safe.

► **Verfügbarkeit**

Die Daten müssen jederzeit verfügbar sein. Dies kann beispielsweise durch die Installation von Hochverfügbarkeitsrechnern, die Daten spiegeln und über redundante Hardwarekomponenten verfügen, erreicht werden.

Englisch?

Diese Vielseitigkeit ergibt sich auch aus dem reinen Wort »Sicherheit«. Würden wir dieses Wort ins Englische übersetzen wollen, damit es aufregender klingt, stünden wir nämlich vor einem Problem. Der Brite hat für die deutsche »Sicherheit« nämlich schon zwei Wörter: *Security* und *Safety*. Mit *Security* würde man zum Beispiel einen profanen Passwortschutz beschreiben, während die *Safety* dafür sorgt, dass wir beim Einschalten des PCs keinen Stromschlag bekommen.

Da man sich also recht schnell in Definitionen, Kleinigkeiten und Formulierungen verrennt, belassen wir es bei der einfachen deutschen Definition, dass Sicherheit genau dann gewährleistet ist, wenn ein System einen vorher genau definierten Zweck erfüllt – nicht mehr und nicht weniger. Sie sind nun in den folgenden Kapiteln eingeladen, alles Mögliche in diese bewusst weit gefasste Formulierung hineinzuzinterpretieren.

Damit Sie dieses Wissen gleich anwenden können, beschäftigt sich das nächste Kapitel mit Sicherheitsproblemen und allem, was wir darunter verstehen. Und das ist eine Menge.

Die Kapitel im Überblick

Bevor wir dieses einleitende Kapitel nun beenden, möchten wir Ihnen noch einen Überblick über den Inhalt des Buches geben. Das Buch ist so aufgebaut, dass die Theorie, die unbedingt benötigt wird, in den ersten Kapiteln besprochen wird. Mit Hilfe dieser dann vorhandenen Theorie kann die praktische Umsetzung dieser Themen im Hauptteil des Buches hoffentlich recht problemlos vonstatten gehen.

Im **1. Kapitel** geht es vor allen Dingen darum, ein Bewusstsein für die Notwendigkeit der IT-Sicherheit zu schaffen. Wir erläutern die möglichen Gefahren und Folgen von Angriffen und definieren einige grundlegende Begriffe. Gefahrenanalyse

In **Kapitel 2** sehen wir uns die Gefahren des physikalischen Zugriffs und Möglichkeiten der (physikalischen) Absicherung von Computern und Funknetzen an. Physikalische Sicherheit

Im nächsten Kapitel dreht sich alles um das IT-Grundschutzhandbuch des BSI (Bundesministerium für Sicherheit in der Informationstechnik). Dabei werden die typischen Vorgehensweisen bei einer Absicherung nach diesem Standard vorgestellt sowie die Benutzung der Grundschutz-Kataloge erklärt. IT-Grundschutz

Kapitel 4 behandelt Thematiken rund um den Einsatz von Firewall-Systemen. Dabei werden zunächst grundlegende Prinzipien erläutert, dann erfolgt die Konfiguration von Paketfiltern und *Network Address Translation* mit der Linux-Firewall `iptables` und OpenBSDs `pf`. Zudem werden wir die Konfiguration eines Proxy-Servers genauer erläutern und Ihnen erklären, was dieser in einem Kapitel über Firewalls zu suchen hat. Firewalls, Proxy-Server

Die Topologien stehen in **Kapitel 5** im Mittelpunkt. Wir konzentrieren uns primär auf die Einrichtung einer Demilitarisierungszone und die Risiken diverser Strukturen und Komponenten im Netzwerk. Topologie

Im **6. Kapitel** soll es um die Erkennung von Einbrüchen gehen. Zunächst beschäftigen wir uns mit der Konfiguration eines Honeypot-Systems, anschließend mit dem Netzwerk-Logging via Syslog und der Intrusion Detection. Dabei gehen wir genauer auf das Network Intrusion Detection System »Snort« ein. Honeypots und IDS

In **Kapitel 7** setzen wir diese Thematik aus dem Blickpunkt der Netzwerküberwachung, also mit aktiveren Mitteln, fort. Das Ziel besteht primär darin, Ihnen einfache und effektive Lösungen für die permanente Überwachung der Netzwerksysteme – zum Beispiel durch den Einsatz des Netzwerkmonitors Nagios – zu geben. Auch werden wir den Security Monitoring

Scanner Nessus und den Netzwerkscanner Nmap genauer unter die Lupe nehmen. Zudem wird noch auf die Netzwerkpaket-Analyse durch Sniffer wie tcpdump oder ethereal eingegangen.

- Remote-Access** **Kapitel 8** setzt sich mit dem Remotezugriff auf Netzwerke, jedoch im Speziellen mit der Implementierung eines virtuellen privaten Netzwerks (VPN) mit IPsec und OpenVPN auseinander.
- Audits** In **Kapitel 9** geht es vor allem um die prinzipielle Vorgehensweise bei Sicherheitüberprüfungen, sogenannten Audits.
- Serveraufbau** **Kapitel 10** beschäftigt sich mit dem Aufbau der Server, deren Umgebung und der Entsorgung alter Hardware.
- Dienste** Es folgt die Absicherung einzelner Dienste wie die des Nameservers Bind oder die des Secure Shell Servers SSH.
- Websicherheit** **Kapitel 12** befasst sich mit der Absicherung von Webserver-Software, PHP-Skripten, MySQL und allgemeineren Sicherheitsaspekten eines Webauftretens.
- SSH** Die Secure Shell bekam aufgrund ihrer Bedeutsamkeit ein eigenes Kapitel von uns verpasst. Wir werden Ihnen zeigen, was mit SSH möglich ist und wie es unter verschiedenen Betriebssystemen umgesetzt wird.
- Unix und Windows** Die **Kapitel 14 und 15** geben jeweils einen Einblick in die Absicherung von Unix- bzw. Windows-Systemen.
- Viren, Würmer & Co.** Anschließend befassen wir uns in **Kapitel 16** kurz mit einem leider nicht zu vernachlässigenden Thema – den Viren. Dabei sollen aber nur ein paar kurze, generelle Aspekte betrachtet werden.
- Disaster Recovery** Im Kapitel über Disaster Recovery (**Kapitel 17**) wenden wir uns dem Vorgehen im Problemfall zu. Was sollte man tun, und was sollte man möglichst unterlassen?
- Security Policys** Darauf folgt ein sehr kurzes **Kapitel 18** über Security Policys, in dem kurz erläutert wird, was in solch eine Security Policy hineingehört und wie man sie anwendet.
- Backups** Backups sollten regelmäßig durchgeführt werden, doch dabei gibt es verschiedene Arten von Backups und verschiedene Strategien diese zu realisieren – vorgestellt werden diese in **Kapitel 19**.
- Kommunikation** Was macht die digitale Kommunikation sicher bzw. unsicher? Diese Frage steht im Zentrum des **20. Kapitels**.

- Die Authentifizierung von Benutzern und Computern in Netzwerken ist äußerst wichtig. Aus diesem Grund haben wir dieses Kapitel zweigeteilt: **Kapitel 21** enthält einen Überblick über die wichtigsten Authentifizierungsmöglichkeiten, etwa durch Biometrie oder S/Key. Außerdem erläutern wir PAM und die Programmierung mit der PAM-Library. **Kapitel 22** wendet sich der Authentifizierung mittels Kerberos zu. **Authentifizierung**
- Bei der Entwicklung von Software gibt es vieles, was man falsch machen kann. Wir setzen uns in diesem Kapitel mit Themen wie Buffer-Overflows oder Formatstring-Overflows auseinander und gehen auf Race-Conditions und deren Vermeidung sowie auf einige weitere Sicherheitsaspekte bei der Softwareentwicklung ein. **Softwaresicherheit**
- Zum Schluss stellt Ihnen **Kapitel 24** die wichtigsten Organisationen im Bereich der (Netzwerk-)Sicherheit und einige hilfreiche Newsgroups vor, an die Sie sich wenden können, falls Sie einmal hilfreichen Rat benötigen sollten. **Organisationen und Newsgroups**
- Im ersten Kapitel des Anhangs befassen wir uns mit den Grundlagen der Kryptografie. Dabei werden die wichtigsten Begriffe und Verschlüsselungsalgorithmen, etwa eine simple XOR-Verschlüsselung, aber auch RSA und Ähnliches erläutert. **Kryptografie**
- Im zweiten Kapitel des Anhangs beschäftigen wir uns mit der TCP/IP-Protokollsuite, wobei wir uns primär auf den Internet- und Transport-Layer konzentrieren werden, um Ihnen die für die Konfiguration von Firewalls (Kapitel 4) und Network-Intrusion-Detection-Systemen (Kapitel 6) wichtigen Grundlagen zu vermitteln. Im Mittelpunkt stehen die Protokolle ARP, IP, ICMP sowie IPv6, ICMPv6 und die beiden Transport-Layer-Protokolle UDP und TCP. **TCP/IP**
- Die Icons in diesem Buch**
- Wichtige Hinweise finden Sie in Abschnitten, die mit diesem Symbol **[«]** gekennzeichnet sind.
- Tipps und Tricks finden Sie neben diesem Symbol. **[!]**
- Beispiele werden hingegen durch dieses Symbol eingeleitet. **[zB]**

»Und wie viel kostet das Gratiswochenende?«
– Homer Simpson

1 Gefahrenanalyse

In diesem Kapitel soll eine Gefahrenanalyse erstellt werden, um ein Gefühl für die Komplexität des Themas Sicherheit zu vermitteln. An dieser Stelle kann man auch gut ein wichtiges Gesicht zu dem Satz »So eine Analyse kann aber nie komplett sein« machen, aber wir wollen Ihnen solche platten Sprüche ersparen.

Auf jeden Fall werden wir im Folgenden keine Lösungen vorstellen, sondern erst einmal nur Probleme aufzeigen. Und auch wenn die folgenden Anmerkungen teils trivial und teils weit hergeholt erscheinen, so muss man doch alle Eventualitäten in einem ganzheitlichen Sicherheitskonzept berücksichtigen.

1.1 Die Räumlichkeiten

Fangen wir also mit der einfachsten Form der Sicherheit an. Ganz trivial und für jeden einzusehen ist der Zweck einer **Tür**: Sie hindert potenzielle Diebe daran, nachts Ihre Computer oder andere Wertgegenstände zu entwenden.

Der Sinn einer Tür

Das kann man natürlich weiterdenken. Vor einiger Zeit lief im Fernsehen eine Werbung für einen Versicherer, in der eine Putzfrau versehentlich in einer Art Kontrollraum Knöpfe drückt und damit vermutlich ein mittelgroßes Chaos anrichtet: vielleicht ein Anlass zur Frage, wer denn eigentlich alles bei Ihnen in den Serverraum kann.

1.1.1 Der Serverraum

Apropos Serverraum. Gerade bei diesem Thema kann man sehr viel falsch machen, zum Beispiel:

- ▶ Den Hauptwasseranschluss in den Serverraum legen
- ▶ Nicht für ausreichenden Brandschutz sorgen
- ▶ Keine Klimaanlage zur Verfügung haben
- ▶ Die Akkus der USV nicht regelmäßig überprüfen
- ▶ Und noch viel mehr

Natürlich gibt es auch die absurdesten Situationen, die man sich nicht einmal mit viel Fantasie auszudenken vermag. Schließlich kann man nicht nur mit den Räumlichkeiten an sich viel falsch machen, sondern eben wie gesagt auch beim Zugang dazu.

Lokaler Zugriff Hat ein Angreifer nämlich erst einmal lokalen Zugriff auf den Server, ist es meist nur eine Frage der Zeit, bis der Server übernommen wird. Dazu ein Beispiel: Bei einem Linux-Server genügt das Hinzufügen der Bootoption »init=/bin/sh« im Bootprompt beim LILO bzw. Grub, um nach dem Booten eine Rootshell zu präsentieren – ohne vorherige Passwortabfrage. Dort ist es nun ein Leichtes, das System schnell so zu manipulieren, dass man den root-Zugang später auch remote nutzen kann.

```
vmlinuz root=/dev/hda2 init=/bin/sh
```

Listing 11 Bootprompt für eine Rootshell (Linux)

Knoppix Alternativ könnte der Angreifer beispielsweise auch über eine Linux-Live-Distribution wie Knoppix gleich alle benötigten Tools zur Verfügung haben und auf der Festplatte diverse Backdoors hinterlegen.

Dies ist, von knopfdrückenden oder kabelziehenden Putzfrauen einmal abgesehen, also durchaus eine ernst zu nehmende Bedrohung.

1.1.2 Die Computerarbeitsplätze

Lassen Ihre Mitarbeiter bzw. Kollegen oder gar Sie selbst auch den Computer laufen, während Sie zum Mittagessen gehen? Prinzipiell ist das kein Problem, solange ein Bildschirmschoner mit Passwort verwendet wird. Ansonsten könnte, falls so ein Problem auch in der Personalabteilung auftritt, vielleicht jemand ...

Absicherung der Clients Schließlich ist eine der wichtigsten Aufgaben der Clients eines Netzwerkes, auf Daten zuzugreifen und diese zu verarbeiten. Wenn man also nicht sicherstellt, dass nur autorisierte Benutzer Zugang zu den Clients

und damit zu den Daten haben, dann folgen daraus schwerwiegende Sicherheitsprobleme.

Doch damit ist es nicht getan. Wenn es nämlich kein ordentliches Rechtemanagement gibt und jeder prinzipiell auf alle Daten zugreifen kann, kommt man ebenfalls schnell in Teufels Küche. Dabei tut es nichts zur Sache, ob beispielsweise ein bestimmtes Netzlaufwerk standardmäßig nicht eingebunden ist oder ob die Daten anderweitig »versteckt« werden. Immerhin beschäftigen sich einige Leute auch privat mit Computern, und der Spieltrieb ist manchmal größer, als man denkt – nicht umsonst kommt die überwältigende Mehrheit aller Angriffe von »innen«.

Schließlich haben Mitarbeiter im Gegensatz zu Hackern zum einen oftmals eine egoistische Motivation, weil ihnen zum Beispiel die Nase vom Chef nicht passt oder Kollegen gemobbt werden sollen, und zum anderen haben sie auch eine gewisse Kenntnis über die Netzwerkstruktur und den Datenbestand des Unternehmens. Natürlich ist auch die Versuchung groß, sich noch mit wichtigen Daten – und wenn es nur »die Früchte der eigenen Arbeit« sind – einzudecken, bevor man zur Konkurrenz wechselt.

Doch auch die Gestaltung der Arbeitsplätze ist unabhängig von allen Computern ein wichtiger Faktor der IT-Sicherheit. Was nützt schließlich ein gutes, gesichertes System, wenn die wichtigsten Ausdrucke offen herumliegen? Oder wenn vielleicht vertrauliche Unterlagen in den einfachen Hausmüll wandern, ohne vorher vernichtet worden zu sein?

Gestaltung der
Arbeitsplätze

Auch würde jeder Angreifer dank Vorbildung durch Film und Fernsehen auf der Suche nach Passwörtern

- ▶ zuerst unter der Tastatur nach kleinen gelben Zetteln Ausschau halten,
- ▶ danach Trivialpasswörter wie »asdf« ausprobieren
- ▶ oder einfach fragen.

Wobei Letztere die eindeutig gewinnversprechendste Möglichkeit ist, sobald man nur eloquent genug und mit einem halbwegs glaubwürdigen Vorwand gewappnet auftritt.

1.2 Social Engineering

Das nächste große Thema in diesem Zusammenhang ist *Social Engineering*, ein Begriff der auch langsam an Bekanntheit gewinnt:

Wie das Wort bereits zum Ausdruck bringt, wird über soziale Kontakte versucht, an geheime Daten zu kommen. Ob der Angreifer nun im Anzug bei der Sekretärin vorspricht oder sich im geliehenen Blaumann Zutritt zum Serverraum verschafft – alles ist möglich, solange es der Beschaffung der gewünschten Daten dient. Dieses *soziale Hacking* basiert also im Allgemeinen auf Überredungs- sowie Überzeugungskunst.

Am besten kann man sich das am Beispiel von Journalisten vorstellen. Journalisten leben vom Publizieren von Informationen und sind folglich darauf angewiesen, immer mit diesen versorgt zu werden. Abgesehen von den »offiziellen« Nachrichtenquellen wird sich ein guter Journalist also zuerst einmal eine Reihe von Informanten halten. Diese Informanten wissen natürlich, dass ihre Informationen in der Presse landen werden, genau wie der Journalist weiß, dass die Veröffentlichung den Informanten einen gewissen Nutzen bringen wird. Insofern benutzt der Journalist also sein soziales Netzwerk, um an Informationen zu gelangen.

Soziales Hacking So richtig deutlich wird der Zusammenhang zum »sozialen Hacking«, wenn ein Journalist unter einem wie auch immer gearteten Deckmantel versucht, an Informationen zu gelangen. Ein prominentes Beispiel dafür ist sicherlich Günter Wallraff mit seinem Buch »Ganz unten«. In dem bereits 1985 erschienenen, aber immer noch höchst interessanten Buch erzählt er, wie er von 1983 bis 1985 mit dunklen Kontaktlinsen, einem schwarzen Haarteil und gebrochenem Deutsch als Türke Ali Levent lebte und die berufliche und gesellschaftliche Diskriminierung der ausländischen Minderheit in Deutschland erfuhr.

Natürlich ist der Alltag des *Social Engineering* im Bereich der IT-Sicherheit etwas anders und weit weniger spektakulär, aber die Folgen sind dafür umso größer. Meist setzen sie nämlich alle weiteren Sicherheitsmaßnahmen außer Kraft. Und dazu braucht es manchmal nicht mehr als den Geburtstag als Passwort oder die kleinen gelben Zettel unter der Tastatur. Denken Sie nur daran, wie schlecht Sekretärinnen oft bezahlt werden und welchen Zugang diese zu wichtigen Daten und Informationen haben.

1.3 Handys, PDAs & Co.

Gefahr durch Statussymbole Die Gefahren der neuen Technikspielzeuge sollte man auch nicht unterschätzen. Die Technik wird immer kleiner, und die Geräte werden immer leistungsfähiger. Gleichzeitig werden sie »trendy« und gar zu Statussymbolen, sodass der pure Besitz entsprechender Gerätschaften zur normalsten Sache der Welt wird.

Im Folgenden sollen kurz mögliche Implikationen angesprochen werden, die sich aus diesen Fakten ergeben. Im Sinne einer Gefahrenanalyse sollen allerdings keine Lösungsmöglichkeiten aufgezeigt oder die Konsequenzen für den konkreten Fall betrachtet werden.

1.3.1 PDAs

Der Trend zur Miniaturisierung ist bei den *Personal Digital Assiants* – kurz PDAs – klar erkennbar. Diese Minicomputer im Notizblockformat erfreuen sich seit der Mitte der 90er-Jahre immer größerer Beliebtheit und bieten meistens die folgenden Funktionen:

- ▶ Adressbuch
- ▶ Kalender samt Termin- und Aufgabenplaner
- ▶ Notizblock
- ▶ Projektmanagement

In vielen neuen Geräten findet man auch erweiterte Funktionalitäten wie

- ▶ E-Mail & Synchronisation mit diversen Groupwaresystemen
- ▶ Textverarbeitung & Tabellenkalkulation
- ▶ Bluetooth, WLAN oder IrDA

Das bietet natürlich einen weitreichenden Spielraum für die kreative Nutzung der Technik.

WLAN etc.

In diese, meist für Laptopzugänge genutzten Funknetze können sich auch einige PDAs¹ einklinken. Mit anderen Worten: Ein Angreifer kann sich so »Zugang« zu einem eigentlich für ihn nicht zugänglichen Netzwerksegment verschaffen.

Natürlich kann man da mit einem »normalen« PDA nicht viel anstellen, doch es gibt diverse Projekte, die sich mit der Portierung und Optimierung von Linux auf PDAs beschäftigen. Hat man so ein System wie zum Beispiel OPIE (*Open Palmtop Integrated Environment*) auf seinem Sharp Zaurus, Siemens SimPad oder HP iPAQ installiert, hat man natürlich die Möglichkeit, auch eigene Software wie diverse Port- oder Vulnerability-scanner nachzuinstallieren.

¹ PDAs mit WLAN-Chip bezeichnet man auch als WDAs.

Server fernsteuern Da man dann meist auch eine Shell mit diversen Tools zur Verfügung hat, kann man auch mit SSH eine Verbindung zu einem Server aufbauen und diesen theoretisch dann mit dem PDA fernsteuern. Außerdem sind entsprechende PDAs bei Warwalkern recht beliebt, um so nach offenen Funknetzen suchen.

Kontakte & Termine

Ein weiterer Problemkreis sind die auf einem Gerät möglicherweise gespeicherten sensiblen Daten. Verliert man nämlich den PDA – oder wird dieser gar vorsätzlich entwendet –, können diese Daten natürlich in falsche Hände fallen. Besonders peinlich wird das Ganze, wenn man den PDA nutzt, um Passwörter zu sichern oder Details über eigentlich geheime Projekte mit sich herumzutragen.

Aber schon der Verlust von möglicherweise privaten oder anderweitig sensiblen Kontaktdaten kann ärgerlich sein. Vor allem wenn es sich bei diesen Daten um Kundendaten oder ähnliche Kontakte handelt, bei denen Vertrauen sehr schnell verspielt sein kann.

1.3.2 Handys

Dieselbe Problematik gibt es natürlich bei Handys, die neben dem »klassischen« Telefonbuch oft auch einen Speicher für Adressen und Termine haben. Auch sonst entwickeln sich die tragbaren Telefone immer mehr zu Alleskönnern: Neben den klassischen Organizerfunktionalitäten ist heutzutage eine Kamera schon obligatorisch.

Fotohandys

Wenn früher unscharfe und falsch belichtete Bilder der Handykameras die Regel waren, die an die ersten Digitalkameras erinnerten, gibt es mittlerweile sogenannte »Megapixelhandys«. So kommt man in die Situation, auch mit dem Handy scharfe Bilder machen zu können.

Datenmitnahme Dadurch erhöht sich natürlich die Gefahr der Datenmitnahme, wenn wichtige Dokumente von Unbefugten einfach abfotografiert werden können. Schließlich hat man oft einfach »nur so« ein Handy in der Hand, um zum Beispiel eine SMS zu schreiben oder nach einem Kontakt zu schauen. Gerade in sensiblen Bereichen wie der Entwicklung oder der Buchhaltung könnte ein entsprechender Missbrauch zu großen Problemen führen. Sollte man also private Handys eventuell in solchen Bereichen verbieten?

Bluetooth etc.

Natürlich verfügen Handys, ähnlich wie PDAs, über diverse Netzwerkschnittstellen wie IrDA oder Bluetooth. Nicht zu vergessen ist auch die Möglichkeit, via UMTS, GPRS oder GSM ins Internet zu gehen. Inwieweit das ein Problem sein kann, kommt immer auf das Handymodell an. Aber wie wir später noch sehen werden, gibt es zumindest bei einigen Modellen handfeste Probleme.

1.3.3 Speichermedien

Mit der Fotothematik wurde bereits das Problem der Datenmitnahme angesprochen. Es kommt leider gelegentlich vor, dass einem Mitarbeiter aus den unterschiedlichsten Gründen gekündigt wird. Da die meisten weiterhin in ihrem Beruf tätig sein wollen, ist es wahrscheinlich, dass sie früher oder später bei der Konkurrenz landen.

Gerade wenn die Trennung nicht in beiderseitigem Einvernehmen stattfand, kommen Mitarbeiter manchmal auf die Idee, »die Früchte ihrer Arbeit« zu »behalten«. Da werden also Arbeitsergebnisse, Dokumente etc. mitgenommen, um später daraus Nutzen zu ziehen. Dabei können die Daten fast unauffällig mitgenommen werden: Schließlich werden die Speichermedien immer kleiner.

USB-Sticks beispielsweise – und darunter fällt auch eine Reihe recht populärer MP3-Spieler – können viele hundert Megabyte speichern; genug Platz für sehr viele wichtige Dokumente.

1.4 Hacker, Cracker & Spione

Ein Lieblingsthema vieler Security-Bücher soll auch in unserem Buch nicht zu kurz kommen: die Geschichte der Menschen, die zumindest partiell etwas mehr über Computer und Technik wissen als mancher Administrator oder Programmierer.

1.4.1 Panikmache

Es gibt TV-Sendungen, die hier nicht näher benannt werden sollen, die Ihnen die extremsten, unglaublichsten und super-mega-tollsten Berichte liefern, damit Sie auch ja dran bleiben, Sendungen, die einfach alles dramatisieren müssen und so tun, als ob das, was sie berichten, entweder

zum ersten Mal erzählt wird oder aber das Schicksal jedes Einzelnen beeinflussen wird.

In unserer Gesellschaft ist es anscheinend gern gesehen, sich mit solcherlei Informationen zu versorgen bzw. versorgen zu lassen. Dieser Trend hat selbstverständlich keinen Halt vor der Hackerkultur gemacht. Es gibt daher Bücher mit Titeln wie »Hackers Blackbook«, »Hackers Dirty Tricks« oder auch »Der Hacker. Ein Insider packt aus«, bei denen schlicht gilt: Finger weg und Geld sparen. Das Image des bösen Hackers, vor allen Dingen des bedrohlichen Hackers wird all zu gerne verwendet, um Angst zu schüren und Aufsehen zu erregen.

Die Leute, die solcherlei Inhalte verfassen, haben meistens keine Ahnung, was sie schreiben, auch wenn sie das selbst wohlmöglich anders sehen werden. Gern auch setzen sie den Begriff »Hacker« mit dem von Raubkopierern gleich.

1.4.2 Und das Know-how?

Natürlich gibt es auch Angreifer, die ihr Handwerk sehr gut verstehen und durchaus über enormes Security-Know-how verfügen. In diesem Fall taucht die Frage auf, womit man sich wohl konfrontiert sehen muss?

Diese Frage lässt sich durch die folgende Faustregel beantworten: Ein Angreifer muss mindestens genauso viel wissen wie der Administrator selbst. Es gibt hervorragende Administratoren auf dieser Welt, aber auch hervorragende Angreifer, die im Optimalfall ihre Tätigkeit als Kunst betrachten.

[»] Wir möchten Sie nicht überbeanspruchen, aber wenn Sie nach diesem Buch noch 20 bis 50 weitere sowie ein paar hundert Security-Papers lesen, die wichtigsten Security-Mailinglisten abonnieren und Security-Newsgroups und -Foren besuchen, kommen Sie der Sache schon relativ nah' ;-)

1.4.3 Zahlen ...

Die »Computerkriminalität« hat in den vergangenen Jahren stark zugenommen. Einerseits lässt sich dies natürlich mit der zunehmenden Verbreitung des Computers und des Internets erklären, allerdings wird diese Erklärung allein der Sache natürlich kaum gerecht. Aber wie sieht dieser starke Anstieg im Detail aus? Gibt es mehr Hackerangriffe, oder sind die Probleme anderer Natur?

Laut statistischem Bundesamt² haben sich die der Computerkriminalität zuzurechnenden Delikte³ im Zeitraum von 1995 bis 2001 von rund 27.000 auf knappe 80.000 erfasste Fälle verdreifacht. Interessant an dieser Tatsache ist nun, dass in den alten Bundesländern⁴ die typischerweise Hackern zuzurechnenden Delikte wie

Etwas Statistik

- ▶ Ausspähen von Daten,
- ▶ Datenveränderung
- ▶ und Computersabotage

mit kaum mehr als 10 Verurteilten pro Jahr sehr gering ausfallen. Und trotzdem steigt die Zahl der Einbrüche und der Verletzung der Sicherheitsrichtlinien ständig – allerdings werden diese in den seltensten Fällen auch angezeigt und tauchen so nicht in der Statistik auf. Das Anzeigen von solchen Vorfällen bei der Polizei ist schlicht nicht üblich, und meistens bringt es für den Geschädigten auch keine Vorteile, sondern im Gegenteil eher Kosten, Aufwand und im schlimmsten Fall schlechte Publicity. Wer gibt schon gern zu, dass er seine IT nicht im Griff hat?

Dies beweisen zum Beispiel Studien wie die vom Magazin Informationweek und von Mummert Consulting im September 2004 veröffentlichte Umfrage zum Thema IT-Security. In dieser hatten im Vergleich zu 2003 knapp zwei Drittel der Befragten mehr oder wesentlich mehr Verstöße gegen die IT-Sicherheit zu verzeichnen.

Wachsendes Risiko

Mit anderen Worten: Hacker, oder wie auch immer man sie bezeichnen will, sind trotz konstant weniger Verurteilter bei einer insgesamt steigenden Computerkriminalität und einem auch sonst stark steigenden Gefahrenpotenzial weiterhin ein nicht zu unterschätzendes Risiko.

Motivation

Hinter Angriffen stecken, wenn es sich um fachlich qualifizierte Personen handelt, drei verschiedene Typen (auf die nicht qualifizierten kommen

Whitehat,
Blackhat, Greyhat

-
- 2 Das Bundesamt war so freundlich, uns die hier zitierten Informationen aus der polizeilichen Kriminalstatistik des Bundeskriminalamtes sowie der hauseigenen Strafverfolgungsstatistik zur Verfügung zu stellen.
 - 3 Summe aus den Einzeldelikten: Betrug mittels rechtswidrig erlangter Kreditkarten mit PIN; Computerbetrug; Betrug mit Zugangsberechtigungen zu Kommunikationsdiensten; Fälschung beweisheblicher Daten, Täuschung im Rechtsverkehr bei Datenverarbeitung; Datenveränderung, Computersabotage; Ausspähen von Daten; Softwarepiraterie (private Anwendung, z. B. Computerspiele); Softwarepiraterie in Form gewerbsmäßigen Handelns
 - 4 In den neuen Ländern wird die Strafverfolgungsstatistik nicht flächendeckend durchgeführt.

wir gleich noch zu sprechen): Whitehats, Greyhats und Blackhats. Die Begriffsdefinitionen sind an dieser Stelle leider nicht so klar, wie man es sich wünschen würde:

► **Whitehat**

Whitehats suchen Sicherheitslücken und schließen selbige. Manche von ihnen suchen solche Lücken auch mal ungefragt und testen die Möglichkeiten aus, die sich ihnen durch eine solche Lücke bieten, ohne jedoch Schaden anzurichten. Im zweiten Fall spricht man auch von »*ethical hacking*«. Diverse Security-Unternehmen bieten ethical hacking auch als Dienstleistung an.

► **Blackhat**

Blackhats kann man als das Gegenteil von Whitehats ansehen. Sie greifen Systeme an und manipulieren diese auch. Sie verfügen ebenfalls über ein gehobenes fachliches Niveau.

► **Greyhat**

Als Greyhat bezeichnet man Personen, die je nachdem, wonach ihnen gerade »der Sinn steht«, entweder Whitehat oder Blackhat sind.

Die Motivation dieser Menschen ist recht interessant. Von purem Interesse an der Technik über diverse Egoprobleme bis hin zu reiner Lust am Zerstören reicht die Palette der Gründe für einen Einbruch in ein fremdes System. Wo man nun den Schwerpunkt dieser breit gefächerten Motivationen zu setzen hat, kommt auf den Bereich an, den man betrachten möchte.

»proof-of-concept« Klassische Einbrüche – mithin die Übernahme eines Servers – sind meist aus prinzipiellem Interesse und dem »*proof-of-concept*«-Gedanken motiviert, man möchte also zeigen, »dass es geht«. Dagegen sind sogenannte *Defacements* meist ein Ausdruck mangelnden Selbstwertgefühls. Da zeigt nun ein Webserver statt der gewohnten Homepage eine meist in Grün/Schwarz gehaltene Seite mit groß tönenden Worten wie: »Lernt einen Server richtig aufzusetzen, dann passiert sowas nicht. Hacked by xyz.«.

Natürlich hat sich ein Administrator in so einem Fall nicht mit Ruhm bekleckert, jedoch sind die vermeintlichen Sicherheitsspezialisten oft auch nicht besser. Schließlich muss man für so eine Nachricht einen Server nicht unbedingt vollständig übernehmen und kommt auch mit abgeschauten und nachgemachten Methoden zum Ziel. Und: Jeder kann mit von diversen Webseiten heruntergeladenem Code Webseiten »hacken«, solange man nur stoisch nach verwundbaren Systemen sucht.

Aber da so ein Vorgehen nach dem Prinzip »Ich kenne eine Schwachstelle und suche ein entsprechendes Ziel« einen ganz anderen Anspruch hat als ein Prinzip »Ich habe ein System und suche dort nach Schwachstellen«, ist so ein Vorgehen nicht sehr hoch angesehen – man nennt »Hacker«, die so vorgehen, auch abwertend *Script-Kiddies*.

Script-Kiddies

1.4.4 Industriespionage

Im Gegensatz dazu steht der Problembereich der Industriespionage. Dort werden nämlich Einbrüche in Computersysteme so verübt, dass man sie nicht aufspürt⁵ und der Einbrecher einen in aller Regel wirtschaftlich motivierten Zweck verfolgt. Dieser Zweck ist meist die Beschaffung – oder in seltenen Fällen auch die Manipulation – von Daten, um das Opfer zu schädigen und selbst im Vorteil zu sein.

Das klingt leider unrealistischer, als es ist. Zwischen legaler und illegaler Informationsbeschaffung existiert nun mal eine Grauzone, die natürlich von fremden »Geheimdiensten« oder auch Wirtschaftsunternehmen gerne ausgenutzt wird. Gefährdet sind dabei vor allem Branchenführer und Unternehmen mit besonderem Know-how, unabhängig von ihrer Größe. Besonders interessieren natürlich dabei Firmen, bei denen Wissen in konzentrierter Form und am besten von mehreren Quellen gleichzeitig vorliegt. Das können Zulieferfirmen, Übersetzerbüros, Unternehmensberater oder Technologiezentren sein.

Grauzone

Dabei muss es sich nicht einmal um klassische Spionage handeln, denn schon durch die reine Teilnahme am Wirtschaftsleben können wichtige und sensible Informationen gesammelt werden:

- ▶ Ankauf von Firmen
- ▶ Gründung von Joint Ventures
- ▶ Einholung von Angeboten
- ▶ Kauf und Analyse von Produkten
- ▶ Inanspruchnahme von Serviceleistungen
- ▶ Umfragen

Bezogen auf die IT gibt es natürlich die Gefahr, dass wirtschaftlich motivierte Hacker das gesamte Know-how quasi auf Knopfdruck »abziehen«. Es subventionieren – wie bereits angedeutet – auch Staaten über ihre

⁵ Und im Gegensatz zu Hackern auch niemand etwas daraus lernt ...

Nachrichtendienste die Industriespionage, um zum Beispiel ihre eigene Volkswirtschaft zu stärken oder Forschung und Entwicklung im eigenen Land zu optimieren.

Im weiteren Verlauf des Buches wird noch einmal genauer auf konkrete Beispiele und Lösungsmöglichkeiten im Hinblick auf die elektronische Überwachung eingegangen, die laut verschiedenen Schätzungen⁶ jährlich eine Summe im zweistelligen Milliardenbereich beträgt (Euro wohlge-merkt).

1.5 Viren, Würmer & Trojaner

Eng im Zusammenhang mit Hackern & Co. stehen natürlich auch »Viren, Würmer & Trojaner« sowie sonstiger möglicher und unmöglicher Unrat der digitalen Welt. Denn nichts entsteht im Softwarebereich von selbst; irgendjemand muss damit Profit machen oder daran zumindest anfänglich seine Freude gehabt haben.

Des Weiteren soll ausdrücklich darauf hingewiesen werden, dass es sehr wohl einen Unterschied in diesen Begrifflichkeiten gibt, auch wenn ein »Virens Scanner« heutzutage natürlich auch Würmer oder Trojaner erkennt und beseitigt.

Ein **Virus** ist ein kleines Programm beziehungsweise eine Programmroutine, die selbstständig andere Programme »infiziert«. Ein Virus wird erst aktiv, wenn das entsprechende infizierte Programm gestartet wird.

Ein **Wurm** verbreitet sich dagegen selbstständig in Netzwerken. Typischerweise ist ein Wurm für eine begrenzte Anzahl bekannter Sicherheitslücken »programmiert« und kann so anfällige Serverdienste infizieren. Dies geschieht normalerweise über die Anwendung von Exploits für diese Sicherheitslücken, die dann einen Code auf dem Server starten, der dann den restlichen Wurm nachlädt.

Ein **Trojaner** ist dagegen ein angeblich harmloses Programm, das aber »bösen« Code zur Öffnung einer Hintertür für Angreifer oder andere Spitzfähigkeiten enthält.

⁶ Unter anderem durch die Gewerkschaft der Polizei und der Frankfurter Wirtschaftsprüfungsgesellschaft KPMG.

Natürlich gibt es heute auch immer mehr Mischformen. So versenden sich zum Beispiel Viren sehr gerne selbst als Dateianhänge, die dann vom Benutzer geöffnet werden sollen. So werden also Eigenschaften aller drei »Gattungen« kombiniert: Schließlich infiziert der Schädling das lokale System wie ein Virus, verbreitet sich selbst ähnlich einem Wurm und muss aber vor der Infektion – ob bewusst oder unbewusst – immer noch wie ein Trojaner vom Benutzer ausgeführt werden.

1.5.1 Schädlinge

Die Schädlinge, die man gemeinhin als »Viren« bezeichnet, sind heutzutage meistens eben solche Mischformen. Vor allem PCs mit Internetzugang sind dabei gefährdet, und zwar eher Client- als Serversysteme.

Schließlich sind Clients im Netz zahlreicher vertreten und somit als Ziel interessanter. Außerdem werden Clients oft weniger sorgfältig gewartet als Server, und es wird häufiger Software installiert. Außerdem spielt die Betriebssystemmonokultur des vorherrschenden Microsoft Windows eine große Rolle. Da die Schädlinge eine größtenteils einheitliche Umwelt vorfinden, können sie sich besser verbreiten – eine solche Aussage kann jeder Biologe bestätigen.

Auch überwiegen im Windows-Bereich generell eher virenartige als wurmartige Schädlinge. Schließlich haben Clients meist weniger Ports und Dienste geöffnet und sind daher eher für Infektionen über installierte Programme anfällig.

Unter Unix/Linux

Unter Unix-artigen Betriebssystemen wie zum Beispiel Linux sind Viren dagegen überhaupt kein Problem. Es gibt zwar auch Virens Scanner für Linux, aber mit denen kann man auch nur nach Windows-Viren suchen. Das macht zum Beispiel auf Mailservern oder Fileservern Sinn, die zwar unter Linux laufen, aber Daten für Windows-Clients bereitstellen und verarbeiten.

Nun verleitet der Satz »*Unter Unix-artigen Betriebssystemen [...] sind Viren [...] kein Problem*« leicht zu der Annahme, dass es keine Viren unter diesen Systemen gäbe. Falsch – es gibt sie, sie spielen nur keine Rolle. Das liegt zum einen natürlich an der geringeren Verbreitung dieser Systeme, zum anderen aber auch am intelligenten Rechtesystem.

Keine Gefahr?

Führt ein Benutzer nämlich ein Programm aus, so läuft dies mit den – intelligenterweise sehr beschränkten – Rechten des Benutzers. So darf

dieser keine Binärdateien ändern oder Programme installieren, und so hat ein Virus natürlich auch kaum eine Chance, sich dauerhaft auf einem System einzunisten. Bringt man dagegen den Superuser durch Tricks oder geschickte Exploits doch dazu, eine infizierte Datei auszuführen, so könnte man die Modifikationen zumindest unter Linux dank der Checksummen herausfinden, die die Distributoren von ihren natürlich nicht infizierten Originaldateien bilden. Diesen Schutz könnte man natürlich wieder umgehen, indem man entweder die Checksummen manipuliert oder den Kernel⁷ entsprechend modifiziert ...

Wie Sie sehen, gibt es viele Möglichkeiten, diverse Schutzmaßnahmen beziehungsweise deren Überwindung zu implementieren. Die Praxis zeigt jedoch, dass Unix-Viren bis auf »*proof-of-concept*«-Implementierungen⁸ quasi nicht existent sind.

1.5.2 Botnetze

Botnetze sind eines der größten Probleme des heutigen Internets. Sie wurden größtenteils aus den hier vorgestellten Schädlingen entwickelt. Schließlich liegt es nahe, einen infizierten Rechner für den Remotezugriff durch Hacker zu öffnen und mit anderen befallenen Systemen zu einem »Netzwerk« zusammenzuschließen.

Wirtschaftlichkeit Diese Netzwerke lassen sich auch prima verkaufen, zum Beispiel für verteilte Angriffe (DDoS, Distributed Denial of Service) oder auch zum Versenden von Spam. Die Miete für eine Stunde Zugriff auf ein solches Netzwerk liegt übrigens heutzutage bei ca. 120 Dollar, einen einzelnen Rechner kann man schon ab 10 Cent pro Woche mieten.

Die Netze selbst sind sehr ausgereift. Die wenigsten benötigen noch »klassische« und damit offensichtliche Protokolle wie TCP oder UDP zur Kommunikation. Stattdessen kann man auch über den Payload in ICMP-Nachrichten Daten austauschen und ganze Sessions organisieren. So könnte ein gehackter Rechner quasi über »ping« – was die ICMP-Echo und ICMP-Echo-Reply Nachrichten der TCP/IP-Protokollsuite nutzt – mit einem Hacker kommunizieren. Außerdem sind diese Netze zum Teil selbstorganisierend, neue Maschinen buchen sich also ähnlich wie bei Peer-to-Peer-Tauschbörsen selbstständig in ein von zentralen Servern weitgehend unabhängiges Netz ein.

⁷ Den »Kern« eines Betriebssystems

⁸ Bei diesen Implementierungen soll nur gezeigt werden, dass etwas möglich ist. An einen Einsatz oder sonstigen praktischen Nutzen wird dabei nicht gedacht.

Im Übrigen wurden laut der renommierten Firma Symantec im Jahr 2003 jeden Tag 2000 Computer unter fremde Kontrolle gebracht; 2004 sollen es schon 30.000 täglich gewesen sein. Es gibt also ein Problem.

1.6 Spam

Vom letzten Thema lässt sich leicht eine Überleitung zur größten Plage des heutigen Internets finden: dem Spam. Die unverlangte Werbung kostet die Wirtschaft Milliarden und den Nutzern den letzten Nerv. Mindestens.

Stellen Sie sich einfach mal vor, Sie hätten eine Firma mit 200 Angestellten samt Internetzugang. Diese Angestellten sollen intern mit einem Stundensatz von 50 Euro verrechnet sein. Müsste nun jeder Arbeitnehmer jeden der angenommenen 280 Arbeitstage im Jahr geschätzte 10 Spam-Mails löschen, wofür insgesamt 10 Sekunden gebraucht werden sollen, so ergibt sich für das Unternehmen bereits ein jährlicher Schaden von circa 7800 Euro.

Selbst aus dieser wirklich konservativen Milchmädchenrechnung⁹ sollte ersichtlich sein, dass sich der Kampf gegen Spam im Unternehmen rechnet. So bekommt man schon leistungsfähige Mail-Gateways für den halben Preis, die Spam weitgehend erfolgreich filtern.

Und so kostet der Fortschritt natürlich Geld: ob Virenschanner oder Spamfilter, umsonst ist nichts zu haben. Allerdings merkt man zumindest beim Spam, dass Sicherheit wirklich wichtig und kostensparend ist.

1.7 Selbst in der Verantwortung stehen

Es gibt auch viele Fälle, bei denen man selbst in der Verantwortung steht. Das könnte zum Beispiel der Fall sein, wenn man selbst entwickelte Software oder vielleicht Serversysteme im Bundle mit einem Wartungsvertrag vertreibt. Wie auch immer, aus solchen Dienstleistungen ergeben sich gewisse Verpflichtungen, die wir im Folgenden ansprechen wollen.

Verkauf von
Software

⁹ Rechnen Sie mal aus, wie viel Geld die Zigarettenpause von 5 Minuten kostet. Oder der Gang zum Kaffeeautomaten. So viel Geld kann man kaum erwirtschaften, wie einem scheinbar verloren geht. Aber die eigentliche Produktivität abzüglich dieser menschlichen Pausen oder Unkonzentriertheiten spiegelt sich ja schon im Stundensatz wider. Trotzdem ist und bleibt Spam ein teures Problem.

1.7.1 Ansprechpartner

Der wichtigste Punkt ist vielleicht, dass man für seine Kunden einen **qualifizierten** Ansprechpartner benennt, der im Problemfall eine Lösung sieht und diese auch koordinieren kann. Auf keinen Fall – wirklich unter gar keinen Umständen – sollte in so einem sensiblen Bereich eine halbkompetente Callcenterbesetzung als Hotline fungieren.

Softwareentwicklung

Was passiert, wenn eine Software entwickelnde Firma keinen ordentlichen Ansprechpartner auf der Webseite bekanntgibt, zeigt dieses Posting von einer bekannten Security-Mailingliste:

```
Have a vulnerability in an IBM product.
```

```
sent alert to security@ibm.com, secure@ibm.com and
cert@ibm.com, all three bounced. Can anyone tell me the
official address or procedure to notify IBM?
```

Listing 1.2 Anfrage an eine Security Mailingliste

Mailinglisten für Sicherheitsfragen

Auf dieser Mailingliste werden normalerweise Bugs und anderweitige Sicherheitslücken gepostet. Allerdings schreiben die Richtlinien vor, dass man zuvor den Hersteller zu informieren und eine Stellungnahme abzuwarten hat, um diesem eine Möglichkeit zur Behebung des Problems zu geben.

In der Mail wurde nun gefragt, ob jemand einen entsprechenden Kontakt für IBM-Produkte hat, da entsprechende E-Mail-Adressen nicht funktioniert haben.

Aber offensichtlich konnte dem armen Menschen nicht geholfen werden, wie die folgende Antwort zeigt:

```
For AIX-related flaws, the contact is
security-alert@austin.ibm.com
```

```
For other products... good luck. I also have a
vulnerability in an IBM product but I wasn't able to
get in touch with anyone.
```

```
Online forms told me to call a number that is
unreachable outside USA.
```

```
The AIX security officer told me he would find the
right contact but I never got anything else since.
```

Listing 1.3 Eine Antwort

Vor allem ein Punkt ist für den Hersteller verwerflich: Dass sich ein angesprochener Kontaktmann trotz gegenteiligem Versprechen nicht mehr meldet, ist nicht hinzunehmen; erst recht nicht, wenn es um Sicherheitsfragen geht.

Wie die Geschichte weiterging und ob es vielleicht ein Happy End gab, ist uns leider nicht bekannt. Allerdings nehmen wir es stark an.

Betreuung

Wenn man »nur« Serversysteme betreut – zum Beispiel im Rahmen eines Outsourcings beim Kunden –, ist es auch wichtig, entsprechende Sicherheitsprobleme im Auge zu behalten. Dazu gehören selbstverständlich das regelmäßige Einspielen von Updates und Patches sowie die Benachrichtigung des Kunden bei möglichen Problemen.

Es gibt also gewichtige Gründe dafür, einen kompetenten Ansprechpartner für Sicherheitsfragen zu benennen und mit weitreichenden Kompetenzen auszustatten.

1.7.2 Information

Wenn man als Dienstleister oder in welcher Form auch immer selbst in die Verantwortung tritt, wird auch ein verantwortungsvoller Umgang mit sicherheitsrelevanten Informationen Pflicht¹⁰. Sie müssen nämlich Ihre Kunden informieren, wenn Probleme bekannt werden, die deren Sicherheit beeinträchtigen.

Das ist leider nicht selbstverständlich. Viele, gerade auch große Firmen nehmen es mit der Informationspolitik aus verschiedensten Gründen nicht immer ganz ernst. Diese Gründe wollen wir im Folgenden kurz aufzeigen, bevor wir sie auseinandernehmen wollen ...

Mangelhafte Informationspolitik

Natürlich ist es irgendwo ein Imageschaden, wenn man Fehler eingestehen muss, erst recht, wenn sich eventuelle Fehler häufen. Auch vergeht natürlich zwischen dem Bekanntwerden und der Beseitigung einer Sicherheitslücke eine gewisse Zeit, in der sich Kunden zu Recht »unsicher« fühlen und vielleicht auch am Produkt zweifeln.

Latentes Unsicherheitsgefühl

¹⁰ Wir sehen hier einmal komplett von der rechtlichen Lage ab, sondern wollen die rein ethische Verantwortung betrachten.

Insofern ist es eigentlich nachvollziehbar, wenn Firmen entsprechende Sicherheitsprobleme eher zurückhaltend bewerten und mit der Veröffentlichung vielleicht bis zum Bereitstehen einer Lösung warten. Aber manche Firmen gehen noch weiter: Wieso sollte man einen Patch beziehungsweise eine Lösung für ein Problem herausgeben, wenn dieses der Öffentlichkeit noch gar nicht bekannt ist, vielleicht weil man das Problem nur zufällig entdeckt hat?

Aber auch für reine Zwischenhändler stellen sich entsprechende Fragen. Soll man die eigenen Kunden informieren, wenn ein ausgeliefertes Produkt einen Fehler aufweist, für den man vielleicht nicht verantwortlich ist? Gerade bei Softwarebugs ist dies eine schwierige Frage, die heutzutage allerdings kaum mit »Ja« beantwortet wird. Immerhin ist man nicht direkt verantwortlich, und es ist die Aufgabe des Herstellers, auf entsprechende Probleme hinzuweisen und diese zu lösen.

Jedoch ...

Die richtigen Zweifel an der Kompetenz kommen einem Kunden erst, wenn Probleme nachweislich unterdrückt oder heruntergespielt werden. Auch ist es kein gutes Zeichen, wenn bekannt wird, dass in Hackerkreisen bereits seit geraumer Zeit diverse Sicherheitslücken bekannt sind und genutzt werden. So wird das Unsicherheitsgefühl bezüglich einer konkreten Bedrohung zum unterschweligen Dauerzustand.

Microsoft und Sicherheit

Leidiges Beispiel dafür ist sicherlich Microsoft Windows. Der in der Vergangenheit schon mal durch längerfristig nicht geschlossene Sicherheitslücken sowie eine manchmal zweifelhafte Sicherheitspolitik in die Schlagzeilen geratene Konzern hat ein Produkt, mit dem sich viele Kunden schlicht »unsicher« fühlen. Dieses Gefühl ist nun mal sehr subjektiv und hat nichts damit zu tun, dass ein Windows-System bei einer guten Konfiguration auch sehr sicher ist – immerhin ist Windows 2000 mit der Sicherheitsstufe EAL4 zertifiziert¹¹, während zum Beispiel RedHat Linux oder SuSE Linux nur mit EAL3+ erreicht haben – obwohl Linux im Allgemeinen als sicherer angesehen wird.

Ein Problem in diesem spezifischen Beispiel ist natürlich auch die Verbreitung von MS Windows, die es natürlich für Angriffe attraktiv macht. Außerdem ist ein Betriebssystem eine extrem komplexe Software, in der sich sehr leicht Fehler einschleichen können. Dementsprechend gab und gibt es also immer wieder Probleme mit nicht behobenen Sicherheits-

¹¹ Die EAL-Stufen der Common Criteria (ISO 15408) beschreiben präzise Anforderungen an eine IT-Sicherheitsprüfung. Siehe auch bsi.de/cc/eal_stufe.htm.

lücken, während die Benutzer irritiert und frustriert von der ständigen Aktualisierung ihres Systems sind.

Um dieser Frustration Herr zu werden und gleichzeitig das Sicherheitsproblem in den Griff zu bekommen, hat Microsoft nun zwei sehr interessante Lösungsmöglichkeiten umgesetzt. Da wäre zum einen das Auto-Update, das einen – sofern man einen Internetzugang besitzt – informiert, wenn Updates und Patches zur Verfügung stehen. Diese können dann automatisch heruntergeladen und installiert werden.

Zum anderen gibt es seit November 2003 den sogenannten *Patchday*, einen Tag im Monat, an dem alle Bugfixes und Patches für die neuesten und aktuellsten Sicherheitslücken herausgegeben werden. Das hat natürlich den Vorteil, dass Benutzer und Administratoren sich auf diesen speziellen Tag vorbereiten und es so einrichten können, dass das Einspielen der Updates ohne Probleme vonstatten geht.

Aber natürlich ist diese Praxis auch zu hinterfragen. Schließlich können zu jedem Zeitpunkt Sicherheitslücken auftauchen, die dann natürlich bis frühestens zum nächsten Patchday »offen« bleiben. Allerdings bringt Microsoft bei »besonders schweren« Sicherheitslücken auch unabhängig vom Patchday Fixes heraus – nur hier spielt wieder die immer subjektive Meinung eine wichtige Rolle, ab wann ein Problem nun als »besonders schwer« zu klassifizieren ist.

1.8 Informationsbeschaffung

Angreifer, die gezielt in ein bestimmtes System/Netzwerk eindringen wollen, sammeln so viele Informationen wie nur irgendwie möglich über ihr Ziel. Es kann Monate dauern, bis die Informationen ausreichend sind, um den Angriff möglichst mit Erfolg durchzuführen. Zu diesem Zweck werden natürlich zunächst die großen Suchmaschinen nach Informationen durchsucht. Dabei gibt es die interessantesten Vorgehensweisen, die spezielle Suchfunktionen wie »?intitle=Index.Of .passwd« oder »?inurl=...« verwenden.

Außerdem gibt es mittlerweile simple Google-IP-Scanner. Hier ein Posting mit dem Code eines solchen Scanners, das an die *SecurityFocus.com Penetration Testing*-Mailingliste gesendet wurde. Das Skript setzt einfach IP-Adressen in Google-URLs ein und hofft dann auf Resultate.

Subject: "Ping scan" through Google

The way I do a "Google Ping scan" is so trivial and badly programmed that I'm almost ashamed to publish it. But since two people asked about it, I'll publish it anyway.

This thing could be programmed much better using Perl and the Google API, but I haven't taken the time to do this.

Suppose I want to scan the range: 221.208.146.1-255 (This is a random IP range that I got from one of the most recent SPAM mails that I received.)

The Google search URL belonging to the spam sender's address 221.208.146.138 is:
<http://www.google.nl/search?hl=en&q=%22221.208.146.138%22&btnG=Search>

Now I cut off the last IP range from this string, replace it by a counter and wrap this in a VBScript file:

```
=====
Option Explicit

Const IPRange = "221.208.146"
Const ForWriting = 2

Dim objFSO, objFile, filename
Dim urlpart1, urlpart2, url
Dim i
Dim objHTTP

for i = 1 to 255

    urlpart1="http://www.google.nl/search?hl=en&q=%22"
    urlpart2="%22&btnG=Search"
    url= urlpart1 & IPRange & "." & i & urlpart2

    Set objHTTP = CreateObject("MSXML2.XMLHTTP")
    Call objHTTP.Open ("GET", url, FALSE)
    objHTTP.Send

    If InStr (objHTTP.ResponseText,
```

```

        "did not match any documents") > 0
Then
    ' do nothing
Else
    filename =
        "Googlescan_" & IPRange & "." & i & ".htm"
    Set objFSO = CreateObject(
        "Scripting.FileSystemObject")
    Set objFile = objFSO.OpenTextFile(
        filename, ForWriting, True,
        vbTrue)
    objFile.Write objHTTP.ResponseText
    objFile.Close
End If
wscript.sleep 6000
next

```

=====
 That's all. (I hope I didn't leave a dumb bug in there ...)

Now the above example (IP Range) is not the most suitable, because it looks like it's from a provider with a lot of outgoing IP-adresses, and it is in Chinese, so the websites are not very informative.

But try this for your own company's IP-range and it should get more interesting.

Listing 1.4

Das Skript funktioniert auf eine äußerst simple Weise: Die URL wird nach dem Schema »*http://www.google.nl/search?hl=en&q=%22*«, gefolgt von der IP-Adresse und dem String »*22&btnG=Search*«, zusammengesetzt. Bekommt man die Meldung »did not match any documents« zurück, dann war die Suche erfolglos.

Probieren wir diese Technik doch gleich mal mit der IP-Adresse von *www.OpenBSD.org*. Die Adresse (129.128.5.191) bekommt man über einen simplen Ping-Aufruf. Die URL lautet somit:

»*www.google.nl/search?hl=en&q=%22129.128.5.191%22&btnG=Search*«.

Wenn wir diese URL in den Browser eingeben, dann erhalten wir entsprechende Resultate: Google zeigt, dass die Webseite verfügbar ist (oder

zumindest einmal verfügbar war, denn die Einträge in Google könnten theoretisch veraltet sein).

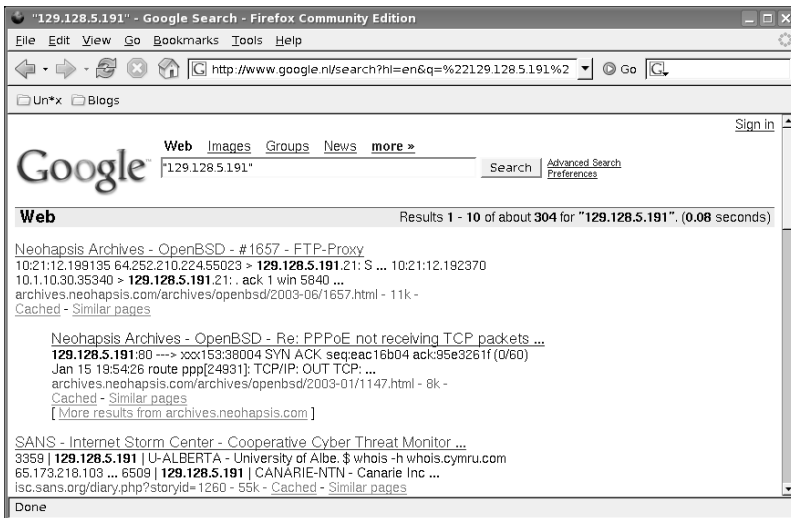


Abbildung 1.1 Der manuelle IP-Scan mit Google

Doch damit nicht genug. Klicken wir ein paar Links an, dann erfahren wir auch gleich, dass ein FTP-Server, nämlich der der Universität von Alberta, hinter der Webseite steckt, die Seite also durch eine Universität und nicht durch das Projekt selbst gehostet wird. Zudem erfahren wir etwas über das Betriebssystem, auf dem die Universität diese Server laufen lässt: Solaris.

Auf `isc.sans.org/diary.php?storyid=1260` bekommen wir sogar einen `whois`-Query (inkl. des gesamten Adressbereiches der Universität, der für weitere Scans durchaus hilfreich sein dürfte) geliefert und müssen diesen folglich nicht selbst tätigen. Ein Angreifer hat somit bereits überaus wichtige Informationen gesammelt, ohne selbst direkt zu scannen!

Sie sollten **niemals** sensible Informationen im Web verfügbar machen! Ein Angreifer erfährt schon genug Informationen über Sie durch Einträge in `whois`-Datenbanken und ähnlichen Informationsquellen!

- [»] Um möglichst viele Informationen über das Suchverhalten von Angreifern zu sammeln, hat Steffen ein Projekt gestartet, das mit scheinbar verlockenden Informationen auf Angreifer wartet:

Auf *cdp.doomed-reality.org/home/www* werden Google-Search-Querys geloggt und später ausgewertet. Nachdem dann sozusagen die Spreu vom Weizen getrennt wurde, wird immer mal wieder ein aktueller Bericht über die interessantesten Search-Querys auf *cdp.doomed-reality.org* geladen.

Wenn Sie die typischen Suchkriterien kennen, können Sie bei Ihrem Webauftritt auch mehr darauf achten, solche Strings zu vermeiden und Ihre Seite so für Angreifer unauffälliger zu gestalten.

Index

/etc/group 357
/etc/login.conf 475
/etc/master.passwd 357
/etc/pam.conf 479
/etc/passwd 357
/etc/protocols 592
/etc/services 337
/etc/shadow 357
/etc/ssh/sshd_config 337
/etc/sudoers 363
/var/adm 360
/var/log 360
~/.ssh/authorized_keys 347
~/.ssh/config 350
~/.ssh/id_rsa 346
~/.ssh/id_rsa.pub 346

A

ACCEPT 92
Access Control Lists 355
Account 637
Account Management 478
accton 188
ACL 355
Adamantix 378
Adaptive Chosen Ciphertext 541
Adaptive Chosen Plaintext 541
Adaptive Chosen Text 541
address space layout randomization 511
airsnort 67
Application Layer 582
ARP 585, 586, 637
 Angriffe 587
asymmetrische Verschlüsselung 336,
 552, 563, 637
Authentication Header 253
Authentication Management 478
Authentifizierung 471, 538
 Biometrie 472
 LDAP 487
 NIS 487
 NIS+ 487

authorized_keys 348

B

Backup 62, 427, 437
 Arten 441
 inkrementell 441
 Komprimierung 443
 Medien 442
 Strategie 438
 vollständig 441
BackupPC 443
Bastion Hosts 141
Bigramme 539
Biometrie 472
Bit-Operatoren 546
Blackhat 44
Blockchiffre 548, 558
 Modi 549
Blowfish 561
Bluejacking 70
Bluesnarfung 71
Bluetooth 69
 Bluejacking 70
 Bluesnarfung 71
 Funktionsweise 69
 Sicherheit 70
Bootvirus 392
boundary-checking 501
BSD 637
BSI 75, 530
Buffer-Overflows 501
 Programmiersprachen 502

C

C 637
 Buffer-Overflows 502
CA.sh 269
Caesar-Verschlüsselung 545
CAST 562
Catch-all 88
CERT/CC 531
Checksumme 538, 553
Chiffre 538
Chipkarten 61

Chosen Ciphertext 541
Chosen Key 541
Chosen Plaintext 541
Chosen Text 541
chroot 363
Citrix 236
Compiler 637
consh 414
Cookies 330
coreography 414
Cracker 41
Credentials 490
Cross-Site-Scripting 327

D

Daemonprozess 637
Darknet 157
Data Encryption Standard 557
Dateisystem IDS 183
Datenbackup 441
Datenbanken 332
Deauthentication Flooding 64
Default Deny 86, 92, 604
Default Gateway 581
Denial-of-Service 64
DENY 92
DES 557, 637
DHCP
 Absicherung 308
Diffie-Hellman-Algorithmus 567
Diffusion 547
Digitale Signatur 554, 637
DMZ 142
DNAT 91
DNS
 Server 292
Druckerfreigabe 386
dsniff 233
Dumpster Diving 433

E

E-Mail
 Absicherung 309
 Standards 453
EBP 503
Echelon 572, 637
EIP 503
ElGamal 567

ESP 254, 503
ethereal 233
Ethernet 62
ETSI 530
ext2/3 637

F

Faxgeräte 286
Feistelnetzwerk 558
FIFO 637
File locking 521
Firewall 86, 142
Forensik 413
 Tools 414
Formatstring-Angriffe 511
Fotohandy 72
fscanf 510
FTP 335, 637
 Absicherung 303
 Anonymous-Login 304
 Protokoll 303
Funktionen
 fcntl() 523
 gets() 510
 snprintf() 510
 sprintf() 510
 strcat() 509
 strcpy() 509
 strlcat() 510
 strncpy() 509
FUPIDS 188
fupids2 190
fwbuilder 126

G

galleta 414
Gateway 581
gcc
 propolice 374
Geheimtext 538
Gentoo
 Hardened 378
getcwd 510
getenv 510
getfac 356
getwd 510
GMR 567
gnupg 463

GPG 461, 637
 GPL 637
 GRE 241
 Greyhat 44
 group 357
 grsecurity 376
 GSHB 75, 77

H

Hacker 41, 214
 Handy 38, 40, 72
 Fotohandy 72
 Viren 394
 Hardened Gentoo 378
 Hardened Linux 378
 Hardening 289
 Hardware
 Backup 284
 Entsorgung 286
 Hash 553, 567, 637
 hdb 414
 Heap-Overflow 514
 HERT 531
 HMAC 251
 honeyd 151
 Installation 152
 Konfiguration 153
 Honeyd 151
 Honeytoken 155
 htaccess 314
 HTTP 314, 459, 637
 Referer 326
 HTTPS 316, 459
 httptunnel 243
 Hubs 137

I

ICANN 529
 ICMP 598
 Echo 599
 ICMPv6 611
 Typen 611
 ICV 254
 IDEA 336, 562
 IDS 167, 169
 IETF 530
 ifconfig 596
 IGMP 604

IGMPv3 605
 IKE 256
 IMAP 310
 Industriespionage 45, 572
 inetd 359
 Integer
 signed 517
 Integer-Overflow 517
 Integer-Promotion 526
 Internet Layer 579
 Interpreter 637
 Intrusion Detection 167
 Intrusion Prevention 191
 Intrusion Response 194
 IP 588
 Adressen 593
 Fragmentierung 590, 595
 Header 589
 Optionen 593
 Protokolle 592
 IP-Spoofing 598
 IPC 637
 ipf 122
 IPFilter 122
 IPIP 241
 iproute2 98
 IPS 191
 IPsec 251
 AH 253
 Authentifizierung 251
 ESP 254
 IKE 256
 Policies 257
 Replay Protection 252
 SPD 258
 Tunnel-Modi 252
 Windows 266
 ipsecadm 264
 ipsecconf 258, 260
 ipseckey 261
 iptables 96
 FORWARD 100
 Funktionsweise 99
 INPUT 100
 Kernelkonfiguration 98
 Masquerading 104
 NAT 112
 OUTPUT 100
 Referenz 104
 IPv6 606

Extension-Header 608
Header 607
Sicherheit 610
Iris-Scan 472
IRS 194
isakmp 240
IT-Grundschutz 75
IT-Sicherheit
 Definition 26
IT-Verbund 75

J

Java
 Buffer-Overflows 502
Javascript 327
john 419

K

kadmin.local 494
KAME BSD IPv6 607
kdb5_util 493
KDC 489
 Konfiguration 491
kdc.conf 493
Kerberos 489
 Client 496
 Master-Server 491
 Principal 491
 Realm 491
 Tickerverwaltung 497
 Windows 497
Kernel 638
Kernel hardening 373
Kernel-Modul 359
Kernelmodul 638
Kernelspace 638
Key Distribution Center 489
kinit 496
Klartext 538
klist 497
Knoppix 36
Known Ciphertext 540
Known Plaintext 540
Konfusion 547
Konten
 umbenennen 386
Kopiergeräte 286
krb5.conf 491

krb5.conf (Client) 496
Kryptoanalyse 68, 538
 Angriffstechniken 540
 Statistik 539
Kryptografie 537, 638
 Export 573
 Ziele 538
Kryptologie 542

L

L2TP 241
Lauschangriff 572
libc 511
Line-Interactive-USV 283
Link Layer 579
Link-Angriff 527
Linux 26, 97
 Patches 368
 Viren 393
Linux-PAM 482
LKM 359, 369, 638
Loadable Kernel Modules 369
logrotate 163

M

MAC-Adresse 138
Mail
 Absicherung 309
MailScanner 311
Makroviren 392
Mandatory Access Control 377
mapping 638
Masquerading 89, 104
master.passwd 357
Maximum Transmission Unit 596
MD2 570
MD4 570
MD5 567, 638
memfetch 414
mkstemp 524
mod_security 318
Monitoring 197
MS-Terminalserver 236
mtree 184, 377
MTU 596
Multicasting 604
Multitasking 638
Multiuser 638

MX-Record 455
MySQL 332

N

Nagios 199
Hostgruppen 207
Hostobjekte 206
Installation 200
Kommandos 211
Konfiguration 204
Kontakte 209
Kontaktgruppen 209
Plugins 202, 211
Serviceobjekte 208

Nameserver
Sicherheit 292

NAT 89, 90, 93, 119

Nessus 227
Installation 228
Konfiguration 229

netfilter 96

netstat 579

Network Access Layer 579

Network Address Translation 89

Netzwerk Dateisystem → NFS

Newsgroups 532

NFS 638
Absicherung 299
Optionen 300

NIS
Absicherung 302

Nmap 214
Praxis 223

NNTP 638

NOP-Slide 508

NRPE 213

NSA 557, 570, 571, 638

NSCA 213

nsswitch.conf 487

O

Oakley 240

Off-by-One 525

One-Time-Pad 550, 638

Open Source 638
Vorteile 97

OpenBSD 377
Patches 364

pf 114

OpenPAM 377

OpenSSH 335
Konfiguration 337

OpenSSL 269

openssl 316

OpenVPN 266
Client 271
Konfiguration 270
Server 270

OpenWall 375, 378

Organizer 72

OSI-Modell 583

OWL 375, 378

P

p0f 159

Paketfilter 87, 91
Einsatzgebiete 91
Funktion 92

PAM 477
Library 483
Linux 482
Module 482
pam_conv 485
pam_end 485
pam_start 485

Passwörter 471
sicher generieren 433

passwd 357

Password Management 478

Patches 383

Path-MTU 598

PaX 376

PDA 39, 72

Perfect Forward Secrecy 267

Personal Firewall 88, 93

pf 114
flags 121
keep state 120
Macro 115
NAT 119
port 116
quick 117
redirect 119
scrub 119
table 117

pfctl 114

PGP 461, 556, 573, 575, 638

PHP

Sicherheit 323

Physikalische Sicherheit 59

PocketPC

Viren 394

POP3 310, 456

Portscan

Bannerscanning 222*Fingerprinting* 223*Fragmentierungsscan* 217*Reverse Ident Scan* 218*TCP Connect Scan* 215*TCP Fin Scan* 216*TCP Idle Scan* 218*TCP Null Scan* 217*TCP Syn Scan* 216*UDP Scan* 221

Portscanner 214

PPTP 241

Prüfsumme 538, 553

Pre-Shared Keys 267

Principal 491

Produktalgorithmus 548, 558

proof-of-concept 44

propolice 374

Proxy

Squid 131*Transparent* 130

Proxy-ARP 587

Proxyserver 128, 638

Absicherung 331

Prozess Accounting 188

Prozesse 638

Public-Key-Verfahren 552

PuTTY 335, 341

Port Forwarding 353*pscp* 344*psftp* 349*PuTTYgen* 347*X11 Forwarding* 350**Q**

Quantenkryptografie 550

Quota 638

R

R-Tools 335

Race Condition 521

Racks 281

RAID 283

Level 0 283*Level 1* 284*Level 5* 284

RARP 587

RC4 562

RDP 236

Realm 491

realpath 510

Referer 326

Remote Desktop 236

Restricted Shells 361

ret2libc 511

Return to Libc 511

Reverse Proxy 131

RIPE 529

RIPEND-160 570

rkhunter 405

rksh 362

root 339

Rootkit 396, 417

Detector 396

ROT 13 546

route 579

Router 140

Routing 581, 624

RSA 336, 563, 638

RSBAC 378

Rule Set Based Access Control 378

Ruleset 97

S

S/Key 474

S/MIME 461, 468

SAD 257

scanf 510

Schlüssel 538

Script-Kiddies 45

SEAM 495

SeBSD 375

Secure Copy 343

Secure Shell 335

Security Association 257

Security Parameter Index 253

- Security through Obscurity 89
- Security-Policy 575
- Securityfocus 531
- SeLinux 375
- Serveraufbau 281
- Serverraum 62, 285
- Serversicherheit 289
- Session Management 478
- Session-ID 329
- setfacl 356
- SFTP 340
- SHA-1 570
- shadow 357
- Shoreline Firewall 101
- SID 190
- Side Channel Angriff 541
- Signale 527
- Signaturgesetz (SigG) 555
- Signaturverordnung (SigV) 555
- skeyinfo 477
- skeyinit 475, 476
- Skipjack 562
- Skriptviren 392
- sleuthkit 414
- SMB 638
- SMTP 453, 638
 - Authentifizierung* 309
- SNAT 91
- Sniffen 59
- Sniffer 231, 416
- snort 171
 - config* 176
 - Installation* 171
 - preprocessor* 178
 - Sniffer* 172
- Social Engineering 37, 539
- Socket 639
- Software
 - Sicherheit* 499
- Solaris
 - Patches* 366
 - Stealth Interface* 370
 - Trusted Solaris* 377
- Spam 49
- SpamAssassin 312
- SPI 253, 257
- Squid 131
 - Konfiguration* 133
- SSH 335
 - Absicherung* 295
 - Device-Sicherung* 295
 - Group-Wrapper* 297
 - Leere Passwörter* 297
 - Listen-Syscall* 296
 - Port-Forwarding* 351
 - Protokoll 1* 336
 - Protokoll 2* 336
 - Public-Key Login* 345
 - PuTTY* 335, 341, 344, 347, 349, 350, 353
 - Remote Login* 341
 - Root-Login* 296
 - scp* 343, 347
 - Secure Copy* 343
 - Secure File Transfer* 349
 - Serverkonfiguration* 337
 - sftp* 340, 349
 - ssh* 341, 352
 - User-Wrapper* 297
 - X11-Forwarding* 340, 350
- SSL 458, 459
- SSL-Modul 316
- Stack 503
- Stack Smashing Protection 511
- Stack-Smashing-Protection 376
- Standardfreigaben 384
- State-Regel 87
- Statistische Angriffe 539
- Stealth Interfaces 370
- Steganografie 542, 639
- Stromchiffre 548
- su 362
- Substitution 539
- sudo 362
 - /etc/sudoers* 363
- Swap 639
 - Verschlüsselung* 371
- Switches 138
- Symlink-Angriff 527
- symmetrische Verschlüsselung 336, 544, 639
- Syscall 639
- syslog 159, 338
 - Konfiguration* 161
- syslog-ng 167
- Systembackup 442
- systrace 192, 377

T

Tapebackup 442
 TCP 615
 Flow-Control 617
 Handshake 620
 Header 618
 Puffer 617
 Reliability 616
 Sicherheitsaspekte 623
 TCP/IP 577, 639
 tcpdump 232, 586
 Telnet 335
 Terminal (PC) 236
 Terminal IDS 190
 Terminal Server 145
 Terminalserver 236
 testdisk 415
 TGT 490
 Thin Client 236
 Ticket 489
 Ticket Garanting Ticket 490
 TLS 458
 Topologien 137
 Transparenter Proxy 93, 130
 Transport Layer 581
 Transposition 539
 Tripwire 183
 Trojaner 46
 Trusted Debian 378
 TrustedBSD 377
 TTL 591
 Tunnel 240
 CDP 250
 DNS 249
 ICMP 248
 IP 248
 POP3 246
 Reliability 251
 TCP 247
 Turtle Firewall 101
 Twofish 561

U

UDP 613
 Header 613
 Sicherheit 614
 UFS2-Attribute 377
 Unix 355

Logging 359
Partitionierung 360
Patches 364
Signale 527
trojanische Pferde 359
Zugriffsrechte 355
 Userspace 639
 USV 282
 Offline-USV 282
 Online-USV 282

V

Viren
 Schutzmaßnahmen 394
 Virus 46, 391
 Typen 392
 VNC 237
 VPN 68
 Extranet 239
 Intranet 239
 Remote-Access 239
 Site-to-Site 238
 Typen 238
 Verschlüsselung 239
 vstt 245

W

Würmer 395
 WarWalking 67
 Wassenaar-Abkommen 574, 639
 Webmin 101
 Webserver-Absicherung 314
 Wechselmedien 386
 Wendel Linux 378
 WEP 64, 639
 Funktionsweise 65
 Schwachstellen 66
 Whitehat 44
 Windows 26, 236, 381
 Partitionierung 387
 Patches 383
 wireshark 233
 WLAN 39, 63, 639
 Ad-Hoc Netzwerk 63
 IEEE 802.11 63
 Infrastruktur 63
 VPN 68
 WEP 64

WPA 68
Wurm 46
WWW 639

X

X11 639
 Absicherung 298
 nolisten 299
xhost 298

XOR 547

Z

Zahlentheorie 538
Zertifikat 269
Zertifizierungsstelle 269
Zonetransfer 293
Zugangskontrollen 60