

Inhalt

0x100	Einführung	1
0x200	Programmierung	7
0x210	Was ist Programmierung?	8
0x220	Pseudocode	9
0x230	Kontrollstrukturen	10
0x231	If-then-else	10
0x232	While/Until-Schleifen	12
0x233	For-Schleifen	13
0x240	Weitere grundlegende Programmierkonzepte	14
0x241	Variablen	14
0x242	Arithmetische Operatoren	15
0x243	Vergleichsoperatoren	17
0x244	Funktionen	19
0x250	Die Hände schmutzig machen	22
0x251	Das große Ganze	24
0x252	Der x86-Prozessor	27
0x253	Assembler-Sprache	29
0x260	Zurück zu den Wurzeln	43
0x261	Strings	43
0x262	Signed, Unsigned, Long und Short	48
0x263	Zeiger	50
0x264	Formatstrings	54
0x265	Typecasting	58
0x266	Kommandozeilenargumente	65
0x267	Geltungsbereich von Variablen (Scoping)	69

0x270	Speichersegmentierung	77
0x271	Speichersegmente in C	85
0x272	Den Heap nutzen	87
0x273	Fehlerprüfendes malloc()	89
0x280	Auf den Grundlagen aufbauen	91
0x281	Dateizugriff	91
0x282	Zugriffsrechte	97
0x283	Benutzer-IDs	99
0x284	Strukturen (structs)	108
0x285	Funktionszeiger	112
0x286	Pseudozufallszahlen	113
0x287	Ein Glücksspiel	115
0x300	Exploits	127
0x310	Allgemeine Exploit-Techniken	130
0x320	Pufferüberlauf (Buffer Overflow)	131
0x321	Stackbasierte Pufferüberlauf-Schwachstellen	134
0x330	Experimente mit BASH	146
0x331	Die Umgebung nutzen	156
0x340	Überläufe in anderen Segmenten	165
0x341	Ein grundlegender heapbasierter Überlauf	165
0x342	Überflutung von Funktionszeigern	171
0x350	Formatstrings	183
0x351	Formatparameter	183
0x352	Die Formatstring-Sicherheitslücke	186
0x353	Beliebige Speicheradressen lesen	188
0x354	An beliebige Speicheradressen schreiben	189
0x355	Direkter Parameterzugriff	197
0x356	Short Writes nutzen	199
0x357	Umwege mit .dtors	201
0x358	Eine weitere notesearch-Sicherheitslücke	207
0x359	Überschreiben der globalen Offset-Tabelle	209

0x400	Netzwerke	213
0x410	OSI-Modell	213
0x420	Sockets	216
0x421	Socket-Funktionen	217
0x422	Socket-Adressen	219
0x423	Netzwerk-Bytereihenfolge	221
0x424	Konvertierung von Internetadressen	222
0x425	Ein einfaches Server-Beispiel	222
0x426	Ein Webclient-Beispiel	227
0x427	Ein Tinyweb-Server	233
0x430	Die tieferen Schichten	237
0x431	Sicherungsschicht	238
0x432	Vermittlungsschicht	240
0x433	Transportschicht	242
0x440	Netzwerk-Sniffing	245
0x441	Raw-Socket-Sniffer	247
0x442	libpcap-Sniffer	249
0x443	Decodierung der Schichten	251
0x444	Aktives Sniffing	261
0x450	Denial of Service	274
0x451	SYN-Flooding	274
0x452	Der Ping des Todes	279
0x453	Teardrop	279
0x454	Ping-Flooding	280
0x455	Verstärkende Angriffe	280
0x456	Verteiltes DoS-Flooding	281
0x460	TCP/IP-Hijacking	281
0x461	RST-Hijacking	282
0x462	Fortlaufendes Hijacking	287
0x470	Port-Scanning	287
0x471	Verdeckter (Stealth) SYN-Scan	288
0x472	FIN-, X-mas- und Null-Scans	288
0x473	Spoofing mit »Lockadressen«	289
0x474	Idle-Scanning	289
0x475	Proaktive Abwehr (shroud)	291
0x480	Hole aus und hacke jemanden	297
0x481	Analyse mit GDB	298
0x482	Zählt nur rohe Gewalt?	300
0x483	Port-bindender Shellcode	303

0x500	Shellcode	307
0x510	Assembler versus C	307
0x511	Linux-Systemaufrufe in Assembler	310
0x520	Der Weg zum Shellcode	313
0x521	Den Stack nutzende Assembler-Instruktionen	313
0x522	Untersuchung mit GDB	315
0x523	Nullbytes entfernen	317
0x530	Shellcode zum Öffnen einer Shell	322
0x531	Eine Frage der Privilegien	327
0x532	Und noch kleiner	330
0x540	Port-bindender Shellcode	331
0x541	Standarddateideskriptoren duplizieren	336
0x542	Sprung-Kontrollstrukturen	338
0x550	Shellcode zum Herstellen von Verbindungen	343
0x600	Gegenmaßnahmen	349
0x610	Erkennende Gegenmaßnahmen	350
0x620	System-Daemons	350
0x621	Signal-Crashkurs	352
0x622	Tinyweb-Daemon	354
0x630	Handwerkszeug	358
0x631	Exploit-Tool tinywebd	359
0x640	Log-Dateien	364
0x641	Mischen Sie sich unters Volk	365
0x650	Das Offensichtliche übersehen	367
0x651	Ein Schritt nach dem anderen	367
0x652	Die Dinge wieder zurechtrücken	372
0x653	Kinderarbeit	378
0x660	Tarnung für Fortgeschrittene	380
0x661	Geloggte IP-Adresse fälschen	380
0x662	Exploit ohne Log	385
0x670	Die gesamte Infrastruktur	388
0x671	Sockets wiederverwenden	388
0x680	Nutzdaten einschmuggeln	393
0x681	Stringcodierung	393
0x682	Wie man einen NOP-Block versteckt	397

0x690	Puffer-Beschränkungen	397
0x691	Polymorpher druckbarer ASCII-Shellcode	400
0x6a0	Die Gegenmaßnahmen verbessern	411
0x6b0	Nicht ausführbarer Stack	411
0x6b1	ret2libc	412
0x6b2	Rückkehr nach system()	412
0x6c0	Zufälliger Stackbereich	414
0x6c1	Untersuchungen mit BASH und GDB	416
0x6c2	Aus linux-gate herauspringen	420
0x6c3	Angewandtes Wissen	424
0x6c4	Ein erster Versuch	424
0x6c5	Überlegenheit ausspielen	426
0x700	Kryptologie	429
0x710	Informationstheorie	430
0x711	Unbedingte Sicherheit	430
0x712	One-Time-Pads	430
0x713	Quanten-Schlüsselverteilung	431
0x714	Effektive Sicherheit	432
0x720	Laufzeit eines Algorithmus	433
0x721	Asymptotische Notation	434
0x730	Symmetrische Verschlüsselung	434
0x731	Lox Grovers Quanten-Suchalgorithmus	436
0x740	Asymmetrische Verschlüsselung	436
0x741	RSA	437
0x742	Peter Shors Quanten-Faktorisierungsalgorithmus	441
0x750	Hybride Chiffren	442
0x751	Man-in-the-Middle-Angriffe	443
0x752	Unterschiedliche Host-Fingerprints der SSH-Protokolle	447
0x753	Fuzzy Fingerprints	451
0x760	Passwörter knacken	455
0x761	Dictionary-Angriffe	457
0x762	Vollständige Brute-Force-Angriffe	460
0x763	Hash-Lookup-Tabelle	461
0x764	Passwort-Wahrscheinlichkeitsmatrix	462
0x770	Drahtlose 802.11b-Verschlüsselung	472
0x771	Wired Equivalent Privacy	472
0x772	Stromchiffre RC4	474

0x780	WEP-Angriffe	475
0x781	Offline Brute-Force-Angriffe	475
0x782	Wiederverwendung von Schlüsselströmen	476
0x783	IV-basierte Entschlüsselungstabellen	477
0x784	IP-Redirection	477
0x785	Fluhrer-Mantin-Shamir-Angriff	479
0x800	Fazit	489
	Literatur	491
	Referenzen	491
	Quellen	492
	Index	493