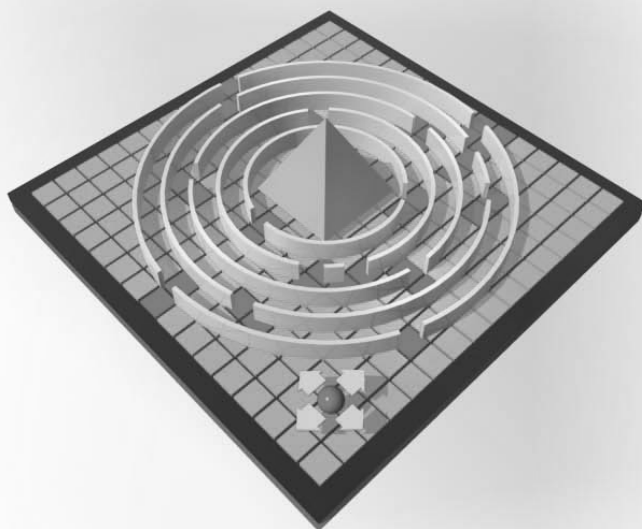




Computer & Literatur Verlag GmbH

DIE KUNST DES PENETRATION TESTING

Marc Ruef



Deutsche Nationalbibliothek – CIP-Einheitsaufnahme
Bibliografische Information der Deutschen Nationalbibliothek

Ein Titeldatensatz für diese Publikation ist bei
der Deutschen Nationalbibliothek erhältlich und im Internet über
<http://dnb.ddb.de> abrufbar.

Alle Rechte vorbehalten. Ohne ausdrückliche, schriftliche Genehmigung des
Herausgebers ist es nicht gestattet, das Buch oder Teile daraus in irgendeiner Form
durch Fotokopie, Mikrofilm oder ein anderes Verfahren zu vervielfältigen
oder zu verbreiten. Dasselbe gilt für das Recht der öffentlichen Wiedergabe.

Der Verlag macht darauf aufmerksam, daß die genannten Firmen- und
Markenzeichen sowie Produktbezeichnungen in der Regel marken-, patent-,
oder warenzeichenrechtlichem Schutz unterliegen.

Die Herausgeber übernehmen keine Gewähr für die Funktions-
fähigkeit beschriebener Verfahren, Programme oder Schaltungen.

1. Auflage 2007

© 2007 by C&L Computer und Literaturverlag
Zavelsteiner Straße 20, 71034 Böblingen
E-Mail: info@cul.de
WWW: <http://www.cul.de>

Coverdesign: Hawa & Nöh, Neu-Eichenberg
Satz: C&L-Verlag
Druck: PUT i RB DROGOWIEC
Printed in Poland

Dieses Buch wurde auf chlorfrei gebleichtem Papier gedruckt

ISBN: 978-3-936546-49-1

INHALT

**Geleitwort
Seite 15**

Kapitel 1 Einführung Seite 17

1.1	Über dieses Buch	17
1.2	Grundwissen.....	21

Kapitel 2 Organisation von Sicherheitsüberprüfungen Seite 27

2.1	Bedrohung und Risiko.....	29
2.2	Assets festlegen.....	41
2.3	Überprüfungsmodi	53
2.4	Informationen einholen und zusammentragen	62
2.5	Dokumentation, Reporting und Präsentation	68
2.6	Gegenmaßnahmen und Controlling	82

Kapitel 3 Footprinting – Informationen sammeln Seite 87

3.1	Informationssammlung in Suchmaschinen	87
3.2	Usenet-Newsgruppen	91
3.3	whois-Abfragen	96

Kapitel 4

Reconnaissance – Zielnetzwerk auswerten

Seite 103

4.1	IP-Adressen herausfinden.....	104
4.1.1	Namensauflösung	105
4.1.2	Erweiterte DNS Zonetransfers	108
4.1.3	whois-Abfragen	110
4.1.4	Suchmaschinen abfragen	112
4.1.5	E-Mail und Postings untersuchen.....	116
4.1.6	Social Engineering: Direktverbindungen.....	118
4.1.7	HTML-Mail mit Internet-Referenz.....	123
4.1.8	Zugriffspfad mit Route-Tracing ermitteln.....	125
4.1.9	Netzwerkverkehr durch Sniffing auswerten.....	127
4.1.10	Lokale Netzwerkkonfiguration auswerten.....	129
4.1.11	Routing-Tabelle auswerten.....	133
4.2	Broadcast-Adressen ermitteln	136
4.2.1	whois-Abfragen	139
4.2.2	Broadcast-Pings.....	140
4.2.3	DHCP-Einstellungen auswerten	141
4.3	Zugriffspfaddiagramme	143
4.3.1	TTL-Auswertung	144
4.3.2	Restriktive Hops erkennen.....	152
4.3.3	Firewall-Elemente erkennen.....	155
4.3.4	Firewall-Systeme mit ICMP-traceroute umgehen.....	160
4.3.5	Exotisches TCP-traceroute	162
4.3.6	Alternative Portnummern gegen Portfilter	165
4.3.7	tcptraceroute mit exotischen TCP-Segmenten	169
4.3.8	Erweiterte Fragmentierung.....	172
4.3.9	Load-Balancing und multiple Routen erkennen	175
4.3.10	Route durch IP Source Routing selbst definieren.....	181
4.3.11	Route Record-Option von Ping als Alternative.....	183
4.4	Das Netzwerkmedium erkennen	186
4.4.1	Hostnamen analysieren	187
4.4.2	MTU und Path MTU analysieren.....	189
4.4.3	MTU der lokalen Schnittstelle	191
4.4.4	Datenübertragungsanalyse	193
4.4.5	Latenzanalyse.....	196

Kapitel 5

Mapping – Aktive Systeme erkennen

Seite 201

5.1	ICMP-Mapping.....	203
5.1.1	Ping-Zugriff auf ein einzelnes System	206
5.1.2	Nicht vorhandene Zielsysteme	210
5.1.3	Existente, aber nicht antwortende Zielsysteme	213
5.1.4	Verlorene ICMP-Datagramme	217
5.1.5	Ablauf eines Ping-Mappings	220
5.1.6	Kurze Pings	221
5.1.7	Auf Namensauflösung verzichten.....	223
5.1.8	Ping-Suchlauf in großen Netzwerken	227
5.1.9	Broadcast-Pings.....	233
5.1.10	Ping-Suchlauf mit gefälschter Absenderadresse.....	237
5.1.11	Online-Dienste als zusätzliche Meinung	240
5.1.12	Exotische ICMP-Typen als Alternative	244
5.2	ARP-Mapping.....	253
5.2.1	arping	257
5.2.2	Dynamische Zugriffe.....	261
5.2.3	ARP-Ping mit gefälschter Absender-IP-Adresse.....	264
5.2.4	ARP-Ping mit gefälschter Absenderadresse	267
5.2.5	Passives ARP-Mapping	270
5.2.6	Grenzen und Nachteile des ARP-Mappings	272
5.3	TCP-/UDP-Mapping.....	274
5.3.1	TCP und TCP-Mapping	275
5.3.2	Einfaches full-connect TCP-Mapping	281
5.3.3	Half-open TCP-Mapping bei offenem Port.....	286
5.3.4	Half-open TCP-Mapping bei geschlossenem Port	289
5.3.5	ACK-Mapping zur Umgehung von Firewalls.....	292
5.3.6	FIN-Mapping als unterschätzte Alternative	295
5.3.7	TCP-Mapping in Adreßbereichen	298
5.3.8	UDP-Mapping	303
5.3.9	Automatisiertes Mapping auf verschiedenen Ebenen	309

Kapitel 6

Portscanning – Dienste ermitteln

Seite 315

6.1	Full-connect und half-open TCP-Scans	317
6.1.1	Einfacher full-connect Portscan mit NetCat.....	319
6.1.2	Full-connect TCP-Scans mit nmap	323
6.1.3	Optimales full-connect Verhalten bei offenem Port.....	327
6.1.4	Das typische Verhalten bei geschlossenem Port	328
6.1.5	Einschränkungen bei full-connect TCP-Portscans	331
6.1.6	Half-open SYN-Scan	333
6.1.7	Das Verhalten bei einem offenen Port	334
6.1.8	Unzuverlässigkeit und Schäden.....	337
6.2	FIN-, Xmas- und Null-Scan	340
6.2.1	FIN-Scan	341
6.2.2	Xmas-Scan	344
6.2.3	Null-Scan	348
6.3	Window-, FTP bounce- und Idle-Scan.....	352
6.3.1	Window-Scan	353
6.3.2	Idle-Scan.....	355
6.3.3	FTP Bounce Scan	358
6.3.4	Einschränkungen und Möglichkeiten	361
6.4	UDP-Portscan.....	364
6.4.1	Normales Verhalten bei einem UDP-Portscan	365
6.4.2	Unerwartete Reaktion bei einem offenen Port.....	368
6.4.3	Zuverlässigkeit.....	372
6.5	Der Weg zum perfekten Portscan	375
6.5.1	Auf Namensauflösungen verzichten	376
6.5.2	Zugriff ohne Mapping.....	378
6.5.3	Portscans verstecken.....	380

Kapitel 7

Application Mapping – Dienste ermitteln

Seite 395

7.1	Zuweisung anhand der Portnummer.....	396
7.2	Banner-Grabbing.....	404
7.3	Application Mapping	413
7.3.1	Willkommens-Banner	419
7.3.2	Reaktion auf allgemeine Eingaben.....	422

7.3.3	Close-Verhalten.....	427
7.3.4	Daytime-Sitzung	429
7.3.5	NOOP/Timeout-Verhalten.....	431
7.3.6	Analyse des Type of Service	433

Kapitel 8

OS Fingerprinting – Betriebssystemerkennung

Seite 439

8.1	Ports und Dienste auswerten.....	440
8.1.1	Typische Portbelegung von Microsoft Windows.....	442
8.1.2	Ermittlung der Windows-Version	444
8.1.3	Typische Portbelegung eines Unix-Systems	445
8.1.4	Funktion eines Hosts	447
8.1.5	Bannerauswertung	448
8.1.6	Anwendungstechnologien erkennen.....	449
8.1.7	Verzeichnisstrukturen auswerten	450
8.2	ICMP-Fingerprinting	452
8.2.1	Reaktion auf »ICMP echo request«-Anfragen	453
8.2.2	Reaktion auf Broadcast-Anfragen	455
8.2.3	echo-request-Datenteil	456
8.2.4	Precedence Bit TOS Echoing	460
8.2.5	TOS Unused Bit.....	463
8.2.6	Address mask request-Fragmentierung.....	465
8.3	TCP-Fingerprinting	468
8.3.1	IP-ID-Generierung.....	469
8.3.2	Teilung des gleichen IP-ID-Zählers	473
8.3.3	Initial Sequence Number	477
8.3.4	Sequenznummern und Bestätigungsnummern	480
8.3.5	Timestamp-Berechnung.....	482
8.3.6	Window Size.....	483
8.3.7	Don't Fragment Flag.....	485
8.3.8	RST-Nutzdaten.....	486

Kapitel 9

Application Fingerprinting – Anwendungen ermitteln

Seite 491

9.1	Direkte oder indirekte Nennung	492
9.1.1	Klassisches Banner-Grabbing	493
9.1.2	Fehlermeldungen provozieren	495
9.1.3	Vermerk im About oder Help	496
9.1.4	Versteckte Hinweise	497
9.2	SMTP-Fingerprinting	500
9.2.1	Die Implementierung von smtpscan	502
9.2.2	Das Begrüßungsverhalten	505
9.2.3	Definition von Absender und Empfänger	508
9.2.4	Onlinehilfe	512
9.2.5	Gefährliche und obsoletere Kommandos	514
9.2.6	Optionale Befehle	518
9.2.7	Das NOOP-Verhalten	521
9.2.8	Erweiterte Tests	523
9.2.9	Erweiterte Statuscodes nach RFC 1893	524
9.3	HTTP-Fingerprinting	530
9.3.1	Statuscodes	531
9.3.2	Lange Anfragen	533
9.3.3	Status Messages	536
9.3.4	Header-Wording	537
9.3.5	Header-Ordering	538
9.3.6	List-Ordering	540
9.3.7	Formatierungen	541
9.3.8	Zeilentrennzeichen	543

Kapitel 10

Denial of Service – Destruktive Attacken

Seite 547

10.1	Auslastung der Netzwerkanbindung	550
10.2	Flooding und Stürme	560
10.3	Fragmentierung	569
10.4	Verbindungen unerlaubt trennen	579

Kapitel 11

Firewall-Systeme – Auswertung und Angriffe

Seite 585

11.1	Paketfilter	587
11.1.1	Route-Traceing	589
11.1.2	Deny- oder Drop-Rules	591
11.1.3	Alternative Portnummern	595
11.1.4	Beglaubigte TCP-Sitzungen durch Flags vortäuschen	597
11.1.5	IP-Fragmentierung.....	598
11.1.6	Quellen vortäuschen mit IP-Spoofing	599
11.1.7	Indirekte Identitäten mit Proxy-Hubs	600
11.2	Application-Gateways und Proxies	602
11.2.1	Portscan für Proxy-Dienste	603
11.2.2	Proxyspezifisches Application Mapping.....	605
11.2.3	Filter erkennen	607
11.2.4	Blacklists durch andere Referenzierungen umgehen.....	609
11.2.5	Schwache Keyword-Filter mit Codierung umgehen.....	610
11.2.6	Port-Redirection und Proxy als Hop	611
11.2.7	Application Inspection durch Tunneling untergraben	612
11.3	Formale Analyse des Firewall-Regelwerks	614
11.3.1	Vorbereitungen	616
11.3.2	Formale Bewertung bestehender Regeln	622
11.3.3	Object-Spanning	630
11.3.4	Inbound- und Outbound-Rules	631
11.3.5	Stealth-Rules.....	632
11.3.6	Kommunikationsbeziehungen visualisieren	633
11.3.7	Komplexität und Fehleranfälligkeit	636
11.4	Konzept und Design	638
11.4.1	Zonen und Kommunikationen.....	639
11.4.2	Fehlender Common Point of Trust.....	642
11.4.3	Single Point of Failure	644
11.4.4	Mehrstufige Firewall-Systeme	646
11.4.5	Virtualisierung durch VLANs.....	650
11.4.6	Virtuelle Betriebssysteme.....	652
11.4.7	Firewalls und VPN-Endpunkte	655
11.4.8	Firewall-Management und -Administration.....	657
11.4.9	Logging, Log Correlation und Analyse	658
11.4.10	Automatische Lockout- und Strike-Back-Mechanismen	660

Kapitel 12

Debugging – Systematische Fehlersuche

Seite 665

12.1	Warum passieren Fehler?	667
12.1.1	Fehler bei der Programmkonzeption.....	668
12.1.2	Fehler bei der Installation und Wartung	669
12.1.3	Anwenderfehler	671
12.2	Fehler finden	675
12.2.1	Grundlegende Schwachstellen suchen	676
12.2.2	Typische Schwachstellen suchen.....	677
12.2.3	Verwundbarkeitsdatenbanken.....	678
12.2.4	Automatische Scanning-Software	681
12.2.5	Automatisiertes Vulnerability Scanning	682
12.2.6	Vulnerability Scanning mit Reizen.....	686
12.2.7	Kombination von Fingerprinting und Reizen/Reaktionen.....	690
12.2.8	Vulnerability Scanning durch Exploiting	692
12.3	Nach neuen Schwachstellen suchen.....	696
12.3.1	Systematisches Finden von Schwachstellen	697
12.3.2	Fuzzing.....	701
12.3.3	Publizieren einer neuen Sicherheitslücke	707
12.4	Quelltext-Analysen	711
12.5	Debugging	714
12.6	Proof-of-Concept	717
12.6.1	Möglichkeiten und Ziele.....	718
12.6.2	Automatisieren durch einen Proof-of-Concept.....	719

Kapitel 13

Exploits – Schwachstellen ausnutzen

Seite 725

13.1	Fehlerhafte Datei-/Verzeichnisrechte.....	727
13.1.1	Fehlendes und fehlerhaftes Nutzungsrecht	728
13.1.2	Zugriffsrechte in Windows-Umgebungen.....	730
13.1.3	Zugriffsrechte in Unix/Linux-Umgebungen	733
13.1.4	SUID und GUID.....	736
13.1.5	Fehlerhafte Freigaben ausnutzen	738
13.2	Directory Traversal	741
13.2.1	Legitime Navigation im Dateisystem	742
13.2.2	Content Management System	744

13.2.3	Von statischen zu dynamischen Eingaben.....	746
13.2.4	Fehlerhafte Filter umgehen	748
13.2.5	Statische Eingabefilter umgehen	751
13.2.6	Filter mit Codierung umgehen	752
13.2.7	Statische Zeichenketten mit Delimiter abtrennen	754
13.3	Code Injection und Command Injection	759
13.3.1	Beispiel einer direkten Kommandobestimmung.....	759
13.3.2	Injizierung von Meta- und Steuerdaten.....	762
13.3.3	Umgehen eines client-basierten Schutzes	764
13.3.4	Aneinanderreihung von Kommandos.....	769
13.4	HTML Injection und Cross Site Scripting.....	773
13.4.1	Injection von statischem HTML	774
13.4.2	Injizieren von Skriptcode	777
13.4.3	Cookies auslesen und verschicken	780
13.4.4	Weitere Möglichkeiten von JavaScript-Angriffen	786
13.4.5	Whitelists ermitteln.....	787
13.4.6	Filter anhand alternativer Schreibweisen umgehen.....	791
13.4.7	Filter umgehen mit Codierung	794
13.4.8	Event-Handler umgehen	795
13.4.9	Cross Site Request Forgery.....	800
13.4.10	Injizierte POST-Abfragen mit XMLHttpRequest	804
13.4.11	Browser-Schwachstellen ausnutzen	807
13.5	SQL Injection	810
13.5.1	Manuelle Zugriffe durch SQL-Parameter-Injection.....	813
13.5.2	SQL-Injection durch Eingabeungültigkeit.....	817
13.5.3	Manipulierte WHERE-Abfragen	820
13.5.4	Komplexe Injections einschleusen	825
13.5.5	Angriffe auf Stored Procedures	827
13.5.6	Denial of Service-Attacken auf Datenbanken	829
13.6	Pufferüberlauf-Attacken	832
13.6.1	Off-by-One-Pufferüberlauf	835
13.6.2	BSS-Overflow	836
13.6.3	Heap Overflow.....	838
13.6.4	Shellcodes für Linux.....	839
13.6.5	Shellcodes für Windows.....	841
13.6.6	Optimierung von Shellcodes.....	843
13.7	Format-Strings	848
13.7.1	Auslesen des Speichers.....	850
13.7.2	Schreiben an eine beliebige Stelle.....	853
13.7.3	Manipulation der Global Offset Table.....	855
13.8	Race Condition	858

13.9	Schwache Authentisierung.....	862
13.9.1	Schwache Paßwörter	864
13.9.2	Benutzername und Paßwort.....	866
13.9.3	Lineare Bruteforce-Attacken.....	867
13.9.4	Klassische Dictionary Attacks.....	870
13.9.5	Optimierte Dictionary Attacks	873
13.9.6	Standardkonten und -paßwörter	876
13.9.7	Authentisierung mitlesen und Replay-Attacken.....	878
13.9.8	Token-basierte Systeme	882
13.9.9	Biometrische Authentisierung	884
13.9.10	Zugriff ohne Authentisierung.....	887
13.9.11	Sessions kopieren, erraten und übernehmen.....	891

Danksagung
Seite 897

Stichwortverzeichnis
899