



Penetration Testing mit Metasploit

Praxiswissen für mehr IT-Sicherheit



Inhaltsverzeichnis

1	Einleitung	11
1.1	Ziel und Inhalt des Buches	11
1.2	Rechtliches	14
1.3	Die Einverständniserklärung	15
1.4	Begrifflichkeiten und Glossar	15
2	Metasploit-Framework: Hintergrund, Historie	17
2.1	Die Geschichte des Metasploit-Frameworks	17
2.2	Die Editionen von Metasploit	18
2.3	Die Wahl der Open-Source-Framework-Edition	19
3	Kali-Linux-Umgebung aufsetzen	21
3.1	Die richtige Plattform	21
3.2	Hintergründe zu Kali Linux	22
3.2.1	Offensive Security	22
3.2.2	Kali-Versionen – Rollierende Releases	23
3.3	Muss es unbedingt Kali Linux sein?	23
3.4	Fluch und Segen zugleich	24
3.5	Nativ oder als VM?	25
3.5.1	Kali als native Installation – volle Performance	25
3.5.2	Kali als VM – immer dabei	26
3.5.3	32 Bit oder 64 Bit?	27
3.6	Erste Schritte nach der Installation	27
3.6.1	Kali-Passwort setzen	27
3.6.2	Zurechtfinden	28
3.6.3	Aktualisieren des Systems	34
3.7	Ein erstes Zielsystem: Metasploitable2	37
4	Pentesting-Grundlagen	39
4.1	Begriffsdefinition	39
4.2	Der Scope	40
4.3	Schwarz, Weiß, Grau	41
4.4	Häufigkeit und Zeitpunkt	41
4.5	Verschiedene Arten von Pentests	42

4.5.1	Infrastruktur-Pentests	42
4.5.2	Windows-Domain-/Active-Directory-Pentest	42
4.5.3	Webapplication-Pentests	42
4.5.4	Application-Pentests	43
4.5.5	Wireless-, Physical-Pentests ... and so much more	43
4.5.6	All-In!	44
4.6	Pentesting-Phasen	44
4.6.1	Phase 1: Reconnaissance/Information Gathering	44
4.6.2	Phase 2: Exploitation	45
4.6.3	Phase 3: Privilege Escalation & Post Exploitation	46
4.6.4	Phase 4: Go back to 1 / Pivot and Escalate	46
4.7	Unterschied zwischen Schwachstellenscans, Pentests und Schwachstellenmanagement	46
4.7.1	Penetrationstest	46
4.7.2	Schwachstellenscan	47
4.7.3	Schwachstellenmanagement	47
4.7.4	Der Mix macht's!	48
5	Schwachstellen und Exploits	49
5.1	Softwareschwachstellen/Schwachstellen im Quelltext	49
5.2	Konfigurationsschwachstellen	50
5.3	Standard-, keine und unsichere Passwörter	51
5.3.1	Standard-Passwörter	52
5.3.2	Keine Passwörter	52
5.3.3	Unsichere Passwörter	52
5.3.4	Eine Lösung für das Passwort-Problem?	53
5.4	Exploits	54
5.4.1	Remote Buffer Overflow in SLmail 5.5	54
5.4.2	Der passende Exploit	56
5.4.3	Verschiedene Kategorien von Exploits	60
6	Nmap-Exkurs	63
6.1	Wie funktionieren Portscanner?	63
6.1.1	Ein wenig TCP/IP-Theorie	64
6.1.2	Wie erhebt ein Portscanner offene Ports?	66
6.2	Keine Angst vor der Nmap-Hilfe	71
6.3	Erste Gehversuche mit Nmap	72
6.3.1	Spezifizieren der zu scannenden Ports	73
6.3.2	Offene Ports vs. lauschende Dienste	76

6.3.3	Nmap-Script-Scanning.....	77
6.3.4	Schwachstellenscanning mit Nmap	80
6.4	Intensität und Datenmengen von Portscans verstehen	82
6.5	Die wichtigsten Nmap-Parameter	84
7	Metasploit-Basics.....	85
7.1	Der erste Start	85
7.1.1	Die Datenbankanbindung	85
7.1.2	Initialisierung der Datenbank.....	86
7.2	Erste Gehversuche im Framework.....	89
7.2.1	Tab Autocomplete und Befehlshilfen	91
7.2.2	Modul-Workflow – Arbeitsschritte	91
7.3	Metasploit-Module.....	95
7.3.1	Die Modulfamilien.....	97
7.4	Eine Ablaufkette am Beispiel von Metasploitable2.....	105
7.4.1	Portscan mit Nmap	105
7.4.2	Validierung mit Metasploit-Auxiliary-Modul	106
7.4.3	Eindringen mittels Metasploit-Exploit und Payload	107
7.4.4	Post-Exploitation.....	114
7.4.5	Übungsaufgabe.....	117
8	Metasploit in der Verteidigung.....	119
8.1	Von der Heise-Meldung über das Patchen bis zur Validierung.....	119
8.1.1	Transparenz und Awareness schaffen mit Metasploit	122
8.1.2	Auffinden und Ausnutzen von Heartbleed mit Metasploit	123
8.1.3	Schließen von Schwachstellen validieren.....	127
8.2	Andere Einsatzmöglichkeiten in der Verteidigung.....	128
9	Praxisbeispiele	129
9.1	Vom Word-Dokument zum Domänen-Administrator	130
9.1.1	Die Laborumgebung	130
9.2	Initialvektor: Word-Makro	134
9.2.1	Der soziale Aspekt – Social Engineering	134
9.2.2	Research, Research, Research	135
9.2.3	Erstellung der Word-Datei mit Metasploit	136
9.2.4	Reverse Listener starten.....	137
9.2.5	Übertragen und Ausführen der Word-Datei	139
9.2.6	Local Privilege Escalation	143
9.2.7	Situational Awareness	151

9.3	Eskalation in der Windows-Domäne	156
9.3.1	Pivoting	156
9.3.2	Pass-The-Hash	164
9.3.3	Clientside-Exploitation	173
9.3.4	Letzte Hürde: Domänen-Administrator	176
9.4	Erkenntnisse für die Verteidigung	181
9.4.1	User-Awareness-Maßnahmen als erste Verteidigungslinie	182
9.4.2	Antiviren- und Endpoint-Security-Software	182
9.4.3	Office-Makros und Sicherheit allgemein	183
9.4.4	Keinerlei Ports aus dem LAN ins Internet öffnen	184
9.4.5	Berechtigungs hygiene	186
9.4.6	Privilege Escalation mittels UAC unterbinden	189
9.4.7	Auffinden von Schwachstellen durch aktive Schwachstellenscanner	190
9.4.8	Alarmierung mittels Microsoft Defender for Identity (MDI)	190
9.4.9	Office-Angriffsfläche mit Windows Defender Exploit Guard verringern	191
9.5	Das IT-Security-Wettrüsten	194
10	Anti-Virus-Evasion	195
10.1	Grundlagen der Anti-Virus-Evasion	196
10.1.1	Wann kann man von einem Virensorter erkannt werden?	196
10.1.2	Pattern-Matching	197
10.1.3	Wenn Virensorter selbst zur Schwachstelle werden	198
10.1.4	Network-, Filetype-, Clientside- und Post-Exploitation	200
10.2	Generierung von Stand-alone-Payloads mit Metasploit	201
10.2.1	msfvenom generiert Payloads	201
10.2.2	Standardtechniken: Packer und Encoder	206
10.3	AV-Evasion mit PowerShell	206
10.3.1	Gar nicht erst in Berührung mit dem Virensorter kommen: PowerShell	206
10.3.2	Invoke-Shellcode.ps1	207
10.3.3	Base64-All-The-Things by Hand	210
10.3.4	Klickbare Payload	212
10.3.5	Verteidigung: PowerShell sperren	214
10.4	Katz-und-Maus-Spiel – Neue Techniken und Tricks nutzen	214

11	Nessus-Schwachstellenscanner	217
11.1	Schwachstellen scannen	218
11.2	Vergleich Schwachstellenscanner	219
11.2.1	Unterschied Schwachstellenscanner zu VirensScanner	219
11.2.2	Funktionsweise von Schwachstellenscannern	220
11.3	Nessus-Versionen	225
11.4	Nessus-Anwendung in der Praxis	225
11.4.1	Installation und Start	226
11.4.2	Erstellen einer passenden Policy	232
11.4.3	Scan der bekannten Labor-Umgebung	248
11.4.4	Schwachstellen-Bewertungssysteme	255
11.4.5	Auswertung der Nessus-Scan-Ergebnisse	259
11.4.6	Effiziente Filtermöglichkeiten	265
11.5	Schwachstellenmanagement	274
11.5.1	Tenable.sc-SecurityCenter	274
12	Schlusswort	277
A	Glossar	279
	Stichwortverzeichnis	285

Einleitung

1.1 Ziel und Inhalt des Buches

Ziel dieses Buches soll nicht sein, dem erfahrenen Metasploit-Nutzer die letzten Tricks und Kniffe des Metasploit-Frameworks beizubringen. Vielmehr soll der unbedarfte (Security-)Administrator gezeigt bekommen, dass Metasploit ein mächtiges Werkzeug ist, das zum Verstehen und Nachstellen von gängigen Angriffsmethoden genutzt werden kann.

Die Veröffentlichung von Schwachstellen und Patches, die wiederum selbige schließen, ist heutzutage an der Tagesordnung. So vergeht kein Monat, in dem nicht eine kritische Schwachstelle im Internet bekannt gemacht wird und Hersteller Hals über Kopf als »kritisch« deklarierte Patches herausbringen.

Wie allerdings kann der normalsterbliche Administrator sicher sein, dass eine Lücke in erster Linie überhaupt vorhanden war? Und wie kann er dafür geradestehen, dass nach dem Einspielen von Patches und/oder begleitenden Konfigurationsänderungen diese Sicherheitslücken auch tatsächlich geschlossen sind?

Inhaltlich sind moderne Schwachstellen mittlerweile so komplex geworden, dass es ein eigener Karrierepfad ist, Schwachstellen aufzufinden und auszunutzen (sogenannte Exploits dafür zu entwickeln). Gerade abseits der Programmier-Berufe fehlt der Einblick und Tiefgang in die Funktionsweise moderner Technologien und Programmiersprachen, um die technischen Hintergründe von Schwachstellen zu verstehen. Selbst Programmierer mit jahrelanger Erfahrung sind nicht automatisch befähigt, Schwachstellen zu verstehen, geschweige denn, sie zu vermeiden. Dies stellt unter anderem auch einen Grund dar, weshalb immer wieder neue Schwachstellen in Software auftauchen.

Wie also soll der Administrator hiermit umgehen und sich sicher sein, dass Lücken existieren und geschlossen werden? Ein Weg ist es, die veröffentlichten Schwachstellen und Angriffstechniken unter sicheren Rahmenbedingungen selbst anzuwenden; Schwachstellen selbst auszunutzen und Exploits selbst einzusetzen, um mit eigenen Augen zu sehen, dass diese vor dem Patchen noch und nach dem Patchen nicht mehr funktionieren.

Genau hier kommt Metasploit ins Spiel.

Hinweis: Was ist ein Exploit?

Ein Exploit kann ein beliebig komplexes Stück Software (oft auch nur ein kleines Script) sein, das es ermöglicht, eine Schwachstelle gezielt auf Knopfdruck auszunutzen.

Bemerkenswert ist hierbei, dass der Anwender eines Exploits die genaue Funktionsweise nicht unbedingt verstehen muss. Oft sind Exploits »schlüsselfertig« und müssen nur noch in die richtige »Richtung« (IP-Adresse und Port) gezielt werden.

Es gibt viele Kategorien von Exploits, zwei der wichtigsten sollen hier kurz erwähnt werden:

Remote-Code-Execution-(RCE)-Exploits:

RCE-Exploits ermöglichen das Ausführen von Code auf Ziel-Systemen über das Netzwerk, ohne einen autorisierten Zugang zum Zielsystem zu besitzen.

Local Exploits:

Lokale Exploits dienen dazu, eingeschränkte Rechte auf einem Zielsystem zu erweitern: So bekommt man über einen RCE Exploit für eine Webapplikation gegebenenfalls eine eingeschränkte Kommandozeile auf einem Linux-Webserver unter dem niedrig privilegierten Apache-User. Ein Local Exploit kann nun helfen, die eingeschränkten Rechte z. B. auf Root-Berechtigungen zu erweitern.

Mehr dazu in Kapitel 5 »Schwachstellen und Exploits«.

Metasploit ist ein mächtiges Framework mit vielschichtigem Einsatzgebiet. So bringt das Metasploit-Framework z.B. Tools mit, die dabei helfen können, Schwachstellen zu finden und Exploits dafür zu entwickeln. Spannender für dieses Buch ist aber die Tatsache, dass mithilfe von Metasploit das Ausnutzen (Exploiten) von Schwachstellen von den technischen Hintergründen der Schwachstellen abstrahiert wird.

Gemeint ist damit, dass das Metasploit eine Vielzahl (2218 zum Zeitpunkt der Fertigstellung dieser Auflage – Mitte 2022) von Exploits für die unterschiedlichsten Betriebssysteme und Programmiersprachen mitbringt. Sie als Anwender müssen aber nur einmal die grundlegende Funktionsweise des Frameworks verstehen und können danach z. B. Linux- sowie Windows-Ziele angreifen, ohne eingefleischter Experte auf der Zielplattform zu sein.

Sie müssen also kein PHP-Guru oder Linux-Kernel-Entwickler sein, um eine PHP-Webapplikationsschwachstelle auszunutzen und danach mittels einer lokalen Linux-Kernel-Schwachstelle vom Apache-Nutzer zum Root-Benutzer zu eskalieren.

Bei all dem kann Metasploit Sie unter den richtigen Voraussetzungen mit ein paar einfachen, wenigen Befehlen unterstützen.

Wichtig: Voraussetzungen zum Verständnis des Buches

Ich möchte niemandem vorschreiben, ob und wann er dieses Buch lesen sollte. Allerdings wurde es mit der Zielgruppe Administratoren und IT-Security-Verantwortlichen im Hinterkopf geschrieben und setzt ein gewisses Vorwissen voraus:

Erfahrungen in der Systemadministration, Netzwerkgrundlagen und dem Programmieren werden an vielen Stellen vorausgesetzt. Es wird jedoch versucht, die technischen Begebenheiten möglichst einfach darzulegen und auf unnötige Verkomplizierungen zu verzichten.

Sollten Sie trotzdem irgendwo nicht folgen können, so kann ich nur dazu ermutigen, einfach die Suchmaschine Ihrer Wahl anzuwerfen und die unklaren Begriffe oder Zusammenhänge zu kombinieren.

Mittlerweile gibt es einen riesigen Schatz an völlig frei zugänglichen Informationen zu diesem Themengebiet, der aber zum größten Teil in der englischen Sprache zu finden ist.

Metasploit kommt mittlerweile in verschiedenen Formen. So wurde das Open-Source-Projekt durch die Firma Rapid7 übernommen und parallel zu einem kommerziellen Produkt weiterentwickelt und vertrieben. Welche Version Sie einsetzen können und wie Sie diese am einfachsten verwenden, wird in Kapitel 2 thematisiert.

Wie zu Beginn schon erwähnt, ist es Ziel dieses Buches, Metasploit nicht nur als reines Angriffswerkzeug (z. B. für Penetration Testing) darzustellen, sondern es auch Systemadministratoren und IT-Security-Verantwortlichen für die Verteidigung zugänglich zu machen.

An dieser Stelle verzichte ich auf das in anderen Werken obligatorische Sun-Tzu-Zitat, da es meiner Meinung nach relativ logisch ist, Angriffstechniken zu verstehen und zu erlernen, um sich effektiv gegen selbige verteidigen zu können.

Kapitel 9 wird hierfür einige gängige Angriffspfade und Techniken erläutern und mithilfe von Metasploit nachstellbar machen.

Bevor dieses Buch entstand, habe ich bereits einige freie Community-Metasploit-Workshops sowie kommerzielle Metasploit-Trainings abgehalten. Für diesen Zweck entstand über die Jahre eine Laborumgebung, anhand der die Teilnehmer der Workshops Metasploit in der Praxis anwenden und testen können.

Da gewisse Angriffstechniken, wie z. B. clientseitige Angriffe sowie die Eskalation in modernen Windows-Domänen-Umgebungen, voraussetzen, dass man auch

entsprechende Systeme zur Verfügung hat, ist diese Laborumgebung deutlich komplexer geworden, als einfach nur ein bis zwei ungepatchte Windows-Installatio-nen zur Verfügung zu haben.

Im Verlaufe dieses Buches werde ich in Abschnitt 9.1.1 diese Laborumgebung zu-mindest teilweise erläutern und als Grundlage zur Demonstration von Metasploit verwenden. Das Kapitel wird außerdem Anregungen und Tipps zum Aufbau eines eigenen Labors geben.

Trotz Fokussierung auf Metasploit wird dieses Buch jedoch an vielen Stellen über den Tellerrand blicken und weitere Tools erwähnen und erklären, die sich mit Metasploit ergänzen.

1.2 Rechtliches

Wahrscheinlich kommt kein Buch, das sich um IT-Security dreht, ohne einen ent-sprechenden Warnhinweis aus:

Das unbedarfe und unkontrollierte Anwenden von Werkzeugen wie Metasploit und anderen Programmen, die sich in Kali Linux befinden, kann (gegebenenfalls versehentlich) zu Straftaten führen.

Es verstößt gegen deutsches Gesetz, ohne Erlaubnis der Eigentümer von IT-Syste- men diese auf Schwachstellen zu überprüfen oder gar Schwachstellen in diesen Systemen auszunutzen.

Doch selbst mit Erlaubnis und Einverständniserklärung der Eigentümer von IT- Systemen kann es durchaus nicht rechtens sein, IT-Systeme zu auditieren. Neh- men wir einmal das Beispiel eines Mailservers in der eigenen Firma. Auf diesem Mailserver liegen gegebenenfalls vertrauliche oder private E-Mails, die nach dem deutschen Postgeheimnis zu betrachten sind.

Auch Shared-Hosting-Umgebungen, wie sie z.B. bei jeglichen Cloud-Providern vorliegen, stellen ein Problem dar: Decken oder nutzen Sie gar eine Schwachstelle in der unterliegenden Infrastruktur des Cloud-Providers auf, so kommen Sie gege- benenfalls an Daten anderer Nutzer dieser Infrastruktur.

Dies gilt es unbedingt zu vermeiden und bedarf ganz klarer vertraglicher Regelun- gen mit dem jeweiligen Provider.

Lassen Sie sich hiervon aber auch nicht einschüchtern. Sicherheitsaudits sind auch in diesen Umgebungen sehr nützlich und wichtig. Sicherheitsaudits lassen alle guten Cloud-Provider unter abgesteckten Bedingungen zu.

Auch könnte wiederum der Internet-Service-Provider, über dessen Infrastruktur ein einfacher Portscan läuft, ein Problem damit haben. Viele Internet-Service-

Provider haben hierzu Klauseln in den Verträgen. Gerade bei privaten Anschlüssen wird das Portscanning gern pauschal verboten. Selbst habe ich zwar noch keine Fälle davon erlebt, dass Internet-Provider deshalb Anschlüsse gekündigt oder Kunden abgemahnt haben, aber auf Nummer sicher geht, wer sich auch hier explizit eine Freigabe einholt.

Zu guter Letzt sollte klar sein, dass es schon ein Kündigungsgrund sein kann, wenn Sie unbedarf mit Metasploit bei Ihrem Arbeitgeber experimentieren. Selbst wenn Sie dabei nichts zerstören und nur gute Beweggründe haben.

1.3 Die Einverständniserklärung

Zu jedem Penetrationstest und Schwachstellenaudit gehört also immer eine schriftlich und vertraglich festgehaltene Einverständniserklärung des Eigentümers der Infrastruktur und aller beteiligten Provider.

Vorlagen hierfür bekommen Sie beim Beauftragen von Schwachstellenscans und Penetrationstests bei professionellen Anbietern oder sicherlich auch frei verfügbar im Internet.

Vorsichtshalber lassen Sie eine solche Vorlage aber lieber durch Anwälte prüfen, bevor Sie größere Audits unternehmen.

Wichtig: IANAL – I am not a Lawyer

Dieses Buch stellt keine fundierte Rechtsberatung dar.

Es soll lediglich an dieser Stelle davor warnen, dass die rechtlichen Rahmenbedingungen der IT-Security sehr ernst genommen werden sollten.

Im Notfall bleiben Sie beim Lesen und Nachverfolgen dieses Buches komplett auf virtuellen Maschinen auf Ihrem privaten Computer oder besuchen entsprechend vorbereitete Workshops oder Weiterbildungen, die abgeschottete Demo-Umgebungen bereitstellen.

1.4 Begrifflichkeiten und Glossar

Zu guter Letzt möchte ich drauf hinweisen, dass es bei tiefgehenden Themen wie Metasploit und IT-Security immer mal wieder vorkommen kann, dass Ihnen einzelne Begriffe oder Hintergründe unklar sind.

Ich habe daher versucht, entsprechende Begriffe im Glossar am Ende des Buches zu beschreiben.

Sollte Ihnen trotzdem beim Lesen noch etwas unklar sein, scheuen Sie sich nicht davor, den Begriff einfach in der Suchmaschine Ihrer Wahl einzugeben. Ich versichere Ihnen, dass Sie zu all dem Geschriebenen in diesem Buch eine Vielzahl von Webseiten finden werden, die Ihnen die Hintergründe weiter erläutern.

Metasploit-Framework: Hintergrund, Historie

2.1 Die Geschichte des Metasploit-Frameworks

Bevor das Buch sich in die technischen Tiefgründe von Metasploit stürzt, möchte ich an dieser Stelle erst einmal ein wenig die Hintergründe und die Historie von Metasploit aufzeigen.

Das Metasploit-Projekt wurde im Sommer 2003 von H. D. Moore gegründet. Die ersten Versionen des Frameworks wurden in Perl geschrieben, mit dem Ziel, die Entwicklung und Anwendung von Exploits zu vereinheitlichen und zu vereinfachen.

Gegen Ende 2003 trat der zweite Kern-Entwickler »spoonm« dem Projekt bei und schaffte die Grundstruktur, nach der auch heute noch das Metasploit-Framework aufgebaut ist und bedient wird.

Kurz danach kam auch Matt Miller »skape« an Bord und komplettierte als drittes Mitglied das Metasploit-Kernteam.

Im Jahr 2006 kam mit der Version 2.7 die letzte in Perl geschriebene Version des Frameworks heraus. Bereits 2005 begann das Team damit, Metasploit in Ruby neu zu schreiben, und brachte letztendlich 2007 mit Version 3.0 eine komplett in Ruby geschriebene Version heraus.

In einem weiterhin auf Github verfügbaren Artikel wird als einer der Hauptgründe hierfür angeführt, dass Perl gewisse Nachteile in fehlender Objektorientierung aufweist, sowie der einfache Grund, dass das Entwickler-Team mehr Spaß an Ruby als an Perl hatte.

Im Oktober 2009 wurde dann durch das Metasploit-Projekt-Team verkündet, dass das Projekt von der Firma Rapid7 gekauft und übernommen wurde. Rapid7 ist eine Firma mit dem Fokus auf IT-Security-Software, die neben Metasploit auch für ihren Schwachstellenscanner »Nexpose« bekannt ist.

Seit 2020 ist das Framework in Version 6.0 und derzeit in Version 6.1.41 verfügbar.

2.2 Die Editionen von Metasploit

Nach der Übernahme durch Rapid7 war Metasploit zwischenzeitlich in vier Versionen verfügbar:

- Metasploit Framework Edition
- Metasploit Community Edition (eingestellt)
- Metasploit Express (eingestellt)
- Metasploit Pro

Im Jahr 2022 bietet Rapid7 allerdings nur noch eine kommerzielle Version an: Metasploit Pro.

Metasploit-Framework-Edition

Die Framework-Edition ist weiterhin Open Source und steht unter einer »BSD Style«-Lizenz.

Das Framework ist unter der URL <https://github.com/rapid7/metasploit-framework> quelloffen verfügbar und kann weiterhin von jedem mitentwickelt und unter Berücksichtigung der geltenden Lizenz weiterverwendet werden.

Dieses Buch wird sich ausschließlich mit dieser Version des Metasploit-Frameworks befassen.

Metasploit Pro

Für diese Edition werden keine Preise öffentlich auf der Rapid7-Webseite genannt. Sie dürften allerdings im fünfstelligen Bereich zuzüglich jährlicher Support-Kosten liegen.

Also ganz klar für den professionellen Einsatz durch große Pentesting-Firmen oder Firmen mit großen hausinternen Penetrationstest-Abteilungen gedacht.

Diese Edition weist neben dem Webinterface auch wieder ein erweitertes Konsolen-Interface ähnlich dem der Framework-Edition auf.

Eine ausführliche Auflistung aller Features findet sich auf der Rapid7-Homepage. Ein paar der interessantesten Features sind im Folgenden aufgelistet:

- Interaktion mit dem kommerziellen Schwachstellenscanner Nmap
- Social-Engineering-Module (z.B. Phishing)
- AV-Evasion (Umgehen von Virenscannern)
- IDS/IPS-Evasion (Umgehen von Netzwerk-Traffic-Scannern)
- VPN-Pivoting (siehe auch Abschnitt 9.3.1 »Pivoting«)
- 24/7-Support durch Rapid7

Um welche Metasploit-Edition dreht sich dieses Buch?

Dieses Buch befasst sich abseits der kurzen Einleitung in das Metasploit-Framework ausschließlich mit der quelloffenen Framework-Edition.

Zwar bieten die kommerziellen Ableger der Firma Rapid7 sicherlich nützliche Zusatzfeatures, allerdings sind sie dafür auch kostenpflichtig und für das Erlernen des Frameworks nicht unbedingt notwendig.

Wer das Bedienen von Metasploit mit der freien Open-Source-Framework-Edition erlernt, wird ohne Weiteres in der Lage sein, bei Bedarf die kommerziellen Editionen zu benutzen.

2.3 Die Wahl der Open-Source-Framework-Edition

Warum habe ich mich auf die Open-Source-Framework-Edition fokussiert und basiere dieses Buch auf dieser Version?

Zum einen aus Überzeugung. IT-Security-Tools müssen nicht kostenpflichtig, kommerziell und Closed Source sein. Metasploit ist eines der Vorzeigeprojekte hierfür.

Auch die Firma Rapid7 gehört an dieser Stelle dafür gelobt, dass sie dem »Open Core«-Modell folgt und den Kern – also die Framework-Edition – weiterhin als kostenlose Open-Source-Version am Leben hält.

Darüber hinaus bin ich der Meinung, dass das Erlernen der Metasploit-Konsole gegenüber den vereinfachten Webinterfaces den großen Vorteil bringt, dass man sich genau überlegen muss, was man vorhat und erreichen will.

Den professionellen Penetrations-Tester unterscheidet vom »Script Kiddie«, dass er genau weiß, was er tut und seine Tools zielgerichtet und so schadfrei wie möglich einsetzt.

Genau dieses zielgerichtete Wissen und Anwenden möchte ich den Lesern dieses Buches zugutekommen lassen.

Als letzten Grund möchte ich dann noch die »Offenheit« der Version anführen, die dazu führt, dass kurz nach Bekanntwerden von großen Schwachstellen sehr häufig zeitnah ein Metasploit-Modul zum Auffinden der Schwachstelle (Auxilliary-Modul) sowie ein Exploit für die Schwachstelle für Metasploit bereitsteht.

Ein Beispiel stellt die Sicherheitslücke MS17-010 in Microsofts SMB-Dienst von Windows dar, die am Wochenende 13.05./14.05.2017 weltweite Bekanntheit dadurch erlangte, dass sie von der Ransomware (Verschlüsselungstrojaner)

»WannaCry« wormartig ausgenutzt wurde und die schadhafte Verschlüsselung vieler Computer auf der ganzen Welt ermöglichte.

Das passende Scanner-Modul (`smb_ms17_010`) tauchte erstmals am 29.03.2017 in Metasploit auf.

Der passende Exploit (`ms17_010_永恒之蓝`) war am 14.05.2017 kurz nach den Schlagzeilen verfügbar und wäre wahrscheinlich durch seine Bekanntheit wegen seiner Herkunft bei der NSA sicherlich auch schon früher in Metasploit integriert gewesen, wenn er nicht große »SMB-Protokoll-Binary-Blobs« enthalten hätte, die mühselig von den Exploit-Modul-Entwicklern revers-engineert, also ohne Dokumentation manuell nachvollzogen und erprobt werden mussten.

Ein weiteres aktuelleres Beispiel stellt zur Zeit des Verfassens dieses Buches die Sicherheitslücke CVE-2021-44228 Log4Shell dar. Log4Shell wurde am 09.12.2021 veröffentlicht und löste so bei vielen Administratoren ein arbeitsintensives Wochenende und einen arbeitsintensiven Dezember im Jahr 2021 aus.

Das passende Scanner-Modul (`log4shell_scanner`) tauchte erstmals am 15.12.2021 – also 6 Tage nach Veröffentlichung der Schwachstelle – im Metasploit-Github-Repository auf.

Passende Exploits-Module wie beispielsweise `log4shell_header_injection` und `vmware_vcenter_log4shell` wurden erstmalig am 07.01. respektive 13.01.2022 im Github-Repository eingechekpt.

Metasploit ermöglicht es also, seine Systeme gezielt und zeitnah nach der Bekanntmachung solcher Schwachstellen zu auditieren.

Ich ermutige trotzdem jeden Leser dieses Buches dazu, auch die Pro-Edition von Metasploit zu erproben und zu testen. Gegebenenfalls weisen sie ja trotzdem Funktionen auf, die das Verwenden oder gar Erwerben dieser Editionen rechtfertigt.

IT-Security-Expertise basiert nicht zuletzt darauf, möglichst viele Systeme zu verstehen und so einen möglichst großen Überblick und Wissensschatz aufzubauen.

Stichwortverzeichnis

0-Day-Exploit 61
2-Faktor-Authentifizierung 54
3-Wege-Handshake 67

A

ACK 279
Administratorrechte 145
Adobe Flash-Player 253
Advanced Threat Analytics 190
agentenbasierter Scan 222
Alive Check 220
Antivirensoftware 182
Anti-Virus-Evasion *Siehe* AV-Evasion
Applikationspentest 43
AppLocker 214
ARP-Protokoll 153
ASR 191
Assembly 279
Assembly Code 43
Attack Surface Reduction 191
AV-Evasion 135, 195, 279
 Grundlagen 196
Awareness 122
Awareness-Maßnahmen 128

B

background-Befehl 145, 158
Baiting 279
Banner 279
Banner-Grabbing 77, 279
Base64 210
Bash 279
Beeinträchtigung der Zielsysteme 93
Befehl
 background 145, 158
 creds 152
 exit 145
 getsystem 145, 150, 177
 hashdump 143, 151, 164
 ifconfig 141
 ipconfig 141
 jobs 139, 148

msfvenom 202
net 144, 177
net localgroup administratoren 144
route 158
run 148
set target 146
shell 144
show targets 146
Berechtigungs hygiene 186
Binary 279
Blackbox-Pentest 41
Bourne Again Shell 279
Browser
 umleiten 175
Browser Redirect 175
Bruteforcing 107, 279
Buffer Overflow 49, 279
bypass_uac-Trick 178
Bypassuac-Exploit 149
Bypassuac-Modul 147

C

Clientside-Exploit 60, 135, 173
Clusternodes 279
Common Vulnerabilities and Exposures 255
Common Vulnerability Scoring System 256
Compliance-Scan 246
Credential Guard 188
Credentialed Scans 221
Credentials 143, 279
creds-Befehl 152
Cross-Site-Scripting 223
CVE 255, 279
CVE-ID 255
CVSS 256

D

Databreaches 280
Debian 280
Debugger 280
Defender of Identity 190
Domain-Administrator-Passwort 179

Domänen-Administrator 176

Domänen-User 269

Dump 280

E

Einverständniserklärung 15

Encoder 204, 206, 280

Endpoint-Security-Lösung 182

Endpoint-Security-Suite 197

Eskalation 156

horizontale 46

vertikale 46

Evasion-Module 104

exit-Befehl 145

Exploit 12, 49, 54, 280

0-Day- 61

alter 173

Beispiel 54

Bypassuac- 149

Clientside- 60, 135, 173, 200

Exploit-DB 56

Fileformat- 136

Filetype- 200

Kategorien 60

Local Exploit 115

lokaler 12

MS14-064- 174

Network- 60, 200

Post- 200

PSEXEC- 168

RCE- 12

Social-Engineering- 135

Exploitation 280

Exploit-Chain 51

F

Fileformat-Exploit 136

Filtermöglichkeiten (Nessus) 265

Firefox 29, 163

Firewall 200

Foca 45

Foothold 142

Fuzzzen 43, 98

G

getsystem-Befehl 145, 150, 177

Github 280

GNOME 280

GPO 280

Greybox-Pentest 41

H

Hashdump 143, 151

hashdump-Befehl 143, 151, 164

Heartbleed 119

auffinden 123

Scanner-Modul 124

Heuristik 197

HiDPI Display 28

Host-Antivirus-Produkt 200

Hostfirewall 182

HTTPS Tunnel Opening 280

HTTPS-Interception 185, 199, 280

I

IANA 280

IDS 280

ifconfig-Befehl 141

IFrame 175

Information Gathering 281

Infrastruktur-Pentest 42

In-House Pentests 128

Injection-Lücken 223

Invoke-Shellcode.ps 207

ipconfig-Befehl 141

IPS 280

IT-Security-Wettrüsten 194

J

JavaScript 175, 212

jobs-Befehl 139, 148

K

Kali Linux 21, 281

64 Bit 27

aktualisieren 34

installieren 25

Konsole 32

native Installation 25

Netzwerkeinstellungen 30

Passwort 27

Repositories 34

Tools von Github installieren 35

Versionen 23

Verzeichnisstrukturen 33

VirtualBox 26

VM 26

VMware 26

Kali Linux ARM Images 26

Kali Linux Nethunter 26

Keine Passwörter 52

Kill-Chain 51, 281

Klartextpasswort 169
 anzeigen 169
 Konfigurationsschwachstelle 50
 Kritikalität 122

L

Labor
 aufbauen 130
 Umgebung 130
 LanManager-Hash 151
 LAPS 186
 Lauschende Dienste 76
 LHOST 136
 Live-Hacking 128
 LM-Hash 151, 186
 Local Administrator Password Solution 186
 Local Privilege Escalation 143
 Local Security Authority Subsystem Service
 143
 LSASS 143

M

Makro 183
 MDI 190
 Metasploit 85
 Autocomplete 91
 Auxiliary-Module 97
 Basics 85
 Befehlshilfen 91
 Core Commands 90
 Credentials Backend Commands 91
 CVE-ID 124
 Database Backend Commands 90
 Datenbankanbindung 85
 Datenbankinitialisierung 86
 Dokumentation 85
 Exploit 107
 Exploit-Module 99
 Geschichte 17
 Heartbleed 123
 Hilfe 89
 Indexierung der Module 85
 Job Commands 90
 Module 95
 Module Commands 90
 Module-Workflow 91
 Modulfamilien 97
 Payload 100, 107, 201
 Post-Exploitation-Module 102
 Projekt 17
 Resource Script Commands 90
 richtiges finden 124

Sessions 111
 Validierung 106
 verfügbare Exploits 271
 Versionen 18
 Workspaces 88
 Metasploit Pro 18
 Metasploitable2 37
 Metasploitable2-VM 72, 105
 Metasploit-Framework 17
 Metasploit-Framework-Edition 18
 Metasploit-Route 158
 Metasploit-Word-Template 136
 Meterpreter 101, 115, 141, 143, 200, 281
 Mimikatz 169
 verhindern 187
 MS14-064 173
 MS14-064-Exploit 174
 MS17-010 252
 msfconsole-Befehl 202
 msfvenom-Befehl 202
 Multihandler 281
 LHOST 138

N

NAT 281
 Nessus 47, 190, 217
 aktivieren 229
 Anwendung 225
 Compliance-Scans 246
 Description 262
 Download 226
 Exploitable With 261
 File-System-Scans 241
 Filtermöglichkeiten 265
 installieren 226
 Nessus Essentials 225, 237
 Nessus Professional 225
 Performance 244
 Plugin Details 260
 Plugin-Output 263
 Plugins 247
 Risk Information 260
 Scan Credentials 244
 Scan-Ergebnisse 259
 Scan-Policy 232
 See Also 262
 Solution 262
 starten 227
 Versionen 225
 Vulnerability Information 261
 net localgroup administratoren-Befehl 144
 net-Befehl 144, 177

- Network-Exploit 60, 200
- Netzwerkdesign 133
- Netzwerkeinstellungen 30
- Netzwerk-Pentest 42
- Netzwerk-Port-Scan 221
- NewTechnology-LanManager-Hash 151
- Nexpose 218, 281
- Nmap 63, 105
 - Ausgabe 64
 - Debug-Ausgabelevel 75
 - Default-Script 79
 - Hilfe 71
 - Ncat 77
 - NSE-Script 79
 - Parameter 84
 - Schwachstellenscanning 80
 - Script 106
 - Script-Scanning 77
 - Scriptsets 79
 - Statuszeile 75
 - TCP-Full-Connect-Scan 68
 - TCP-Stealth-Scan 69
 - TCP-SYN-Scan 69
 - UDP-Portscan 70
- NOPs 281
- NTLM-Hash 151
- O**
 - Offene Ports 76
 - Offensive Security 22
 - Office-Makro 183
 - OpenSSL 120
 - OSI-Schichtenmodell 65
 - OWASP 223
- P**
 - Packer 206
 - Pass-The-Hash 107, 164, 165, 187
 - verhindern 187
 - Password Spraying 281
 - Passwordhash 143
 - Password-Reuse 53
 - Passwort
 - Rotation 187
 - sicheres 187
 - Passwort-Bruteforcing 165
 - Passwortgenerator 53
 - Passworthash 151, 281
 - Passwortmanager 187
 - Passwort-Safe 53, 187
 - Patch 281
 - vs. Schwachstelle 270
 - Patch-Verfügbarkeit 269
- Patch-Zyklus 269
- Pattern-Matching 197, 200
- Payload 49, 100, 137, 281
 - Encoder 204
 - generieren 201
 - Klartext 110
 - klickbar 212
 - Meterpreter 101
 - Reverse-CMD-Payload 109
 - Single-Payload 100
 - Staged-Payload 101
 - verschlüsselt 110
- Penetrationstest *Siehe* Pentest
- Pentest 39, 46
 - Applikations- 43
 - Arten 42
 - Blackbox- 41
 - Exploitation 45
 - Go back to 1 46
 - Greybox- 41
 - Häufigkeit 41
 - Information Gathering 44
 - Infrastruktur- 42
 - Netzwerk- 42
 - Phasen 44
 - Physical- 44
 - Pivot and Escalate 46
 - Post Exploitation 46
 - Privilege Escalation 46
 - professioneller 39
 - Reconnaissance 44
 - Scope 40
 - vs. Schwachstellenscan 47
 - Webapplication- 43
 - Whitebox- 41
 - Wireless- 43
 - Zeitpunkt 41
- Pentesting 281
 - Phasen 44, 104
- Phishing 282
- Physical-Pentest 44
- Pivoting 46, 156, 164, 282
- Plattform 21
- Podcast-Adressen 277
- Policy 282
- Port 184
- Portforwarding 133, 156, 282
- Portscan 105, 282
 - Datenmengen 82
 - Intensität 82
- Portscanner 63, 66
 - Nmap 63
- Post-Exploitation 114

- Post-Exploitation-Phase 114
 Post-Hashdump-Modul 152
 PowerMeta 45
 PowerShell 206
 sperren 214
 PowerSploit 207
 Privilege Escalation 114, 282
 verhindern 189
 verifizieren 151
 Proxychains 161
 PSEXEC 167
 PSEXEC-Exploit 168
 PWK/OSCP-Zertifizierung 23
- Q**
 Qualitätssicherung 24
- R**
 Ransomware 282
 RCE-Exploit 12
 RDP 282
 RDP-Sitzung 171
 aufbauen 170
 Rechtliche Situation 14
 Reconnaissance 282
 Recon-NG 44
 Remote Buffer Overflow 54
 Request for Comments 65, 131
 Reverse Engineering 43, 282
 Reverse Listener 137, 282
 RFC 65, 131, 282
 rockyou.txt 34
 route-Befehl 158
 run-Befehl 148
- S**
 SAM 143
 SCADA 282
 Scan
 agentenbasiert 222
 Credentialled 221
 Netzwerk-Port- 221
 Webanwendungen 223
 Webapplication- 223
 Scanner-Modul
 Heartbleed 124
 Schadsoftware
 generieren 201
 Schwachstelle 282
 bewerten 255
 im Code 49
 Kategorien 49
- Passwort 51
 scannen 218
 Schließung validieren 127
 vs. Patch 270
 Schwachstellen
 Konfigurations- 50, 190, 222
 Software- 190
 Schwachstellen-Assessment 221
 Schwachstellenmanagement 46, 47, 274
 Schwachstellenscan 46
 Ergebnisse auswerten 259
 vs. Pentest 47
 vs. Schwachstellenmanagement 219
 Schwachstellenscanner 47, 190
 Funktionsweise 220
 Nessus 47, 190, 217
 Nexpose 218
 Nmap 80
 und Metasploit 123
 Vergleich 219
 vs. Virenscanner 219
 Scope 282
 Pentest 40
 Script-Scanning 77
 Security Account Manager 143
 Server-Management-Webinterface 120
 Service Detection 221
 set target-Befehl 146
 Severity 270
 shell-Befehl 144, 176
 Shell-Code 60
 Shikata Ga Nai 204
 Shouldersurfing 135
 show targets-Befehl 146
 Sicherheitsaudit 283
 Situational Awareness 151
 SMB-Dateifreigabe 164
 SMBv1-Protokoll 222
 Sniffer 283
 Social Engineering 44, 134, 283
 Social-Engineering-Exploit 135
 Socks4a-Modul 160
 Socks4a-Proxy 161
 Softwareschwachstelle 49
 Spearphishing 45
 Spoofing 283
 SSH-Verbindung 185
 Standard-Passwörter 52
 Stuxnet 40
 Subnetz 153
 SYN 283
 SYSTEM-Rechte 145

T

TCP 66
 Ports 65
TCP/IP 64
tenable.io 275
Tenable.sc-SecurityCenter 274
Transparenz 122
Tunneln 185

U

UAC 145, 189, 283
UAC-Einstellung 150
UAC-Stufe 149
Ubuntu 123
UDP 65, 66
UDP-Portscan 70
UEFI 188
Unsichere Passwörter 52
UPD
 Ports 65
User Account Control 145
User-Agent-String 171
User-Awareness-Maßnahme 182

V

Validierung
 Schließung von Schwachstellen 127
Verschlüsselungstrojaner 282
Verteidigung 128, 181
 IT-Systeme 119
 PowerShell 214
Virenscanner 135
 als Schwachstelle 198
 Funktionsweise 196
 Geschichte 196
 umgehen 195
 vs. Schwachstellenscanner 219
Virtuelle Maschine 283

Virustotal.com 204

VNC-Server 222
Vorher-Nachher-Vergleich 127
VPN 120
Vulnerability Assessment 47, 283

W

WannaCry 20
Webapplication-Audit 224
Webapplication-Pentest 43
Webapplication-Scan 223
Whitebox-Pentest 41
Windows Defender 198
Windows Defender Exploit Guard 191
Windows Server Update Services 50
Wireless-Pentest 43
WMI 283
Word-Datei
 ausführen 139
 mit Makro 136
 mit Metasploit erstellen 136
Word-Makro 134
Word-Template 140
Workspace 88
Wörterbuchliste 34
Wrapper 212

X

XAMPP-Webserver 171
XFCE 28

Y

Yara Rules 241, 283

Z

Zertifikat 278
zsh 32