Michael Messner

Hacking mit Metasploit

Das umfassende Handbuch
zu Penetration Testing und Metasploit

dpunkt.verlag

Michael Messner arbeitet als IT Security Consultant bei der Corporate Technology der Siemens AG in München und führt dort technische Sicherheitsanalysen und Penetrationstests durch. Neben der technischen Analyse von hausinternen Enterprise-Applikationen testet er auch Produkte und Lösungen der Siemens AG auf Schwachstellen. In seiner Freizeit entwickelt er aktiv am Metasploit-Framework mit und hat dabei bereits eine Vielzahl unterschiedlichster Module in das Open-Source-Framework eingepflegt.

Michael Messner

Hacking mit Metasploit

Das umfassende Handbuch zu Penetration Testing und Metasploit

3., aktualisierte und erweiterte Auflage

dpunkt.verlag

Michael Messner
msf@s3cur1ty.de
Twitter: @s3cur1ty_de

Lektorat: René Schönfeldt
Copy-Editing: Annette Schwarz, Ditzingen
Satz und Herstellung: Nadine Thiele
Umschlaggestaltung: Helmut Kraus, www.exclam.de
Druck und Bindung: Media-Print Informationstechnologie, Paderborn

Bibliografische Information der Deutschen Nationalbibliothek
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

ISBN:
Print 978-3-86490-523-0
PDF 978-3-96088-362-3
ePub 978-3-96088-363-0
mobi 978-3-96088-364-7

3., aktualisierte und erweiterte Auflage 2018

Wieblinger Weg 17
69123 Heidelberg

Die erste Auflage dieses Buches erschien unter dem Titel »Metasploit. Das Handbuch zum Penetration-Testing-Framework«.

5 4 3 2 1 0

Für meine Eltern

❖

Für Carina

❖

Für meine Kids

Geleitwort zur ersten Auflage

Penetration Testing hat sich in den letzten Jahren stark etabliert: War das Thema vor einigen Jahren noch in der Domäne des Militärs und der Geheimdienste, ist Penetration Testing mittlerweile fester Bestandteil von Richtlinien wie dem Payment Card Industry Data Security Standard (PCI DSS). Die Besucherzahlen von Konferenzen wie DefCon in Las Vegas sind in den letzten Jahren explodiert. Kein Wunder, denn das Thema hat nicht nur einen gewissen technischen Sex-Appeal, sondern auch einen handfesten Nutzen. Als technischer Anwender von Metasploit haben Sie eine erfolgversprechende Zukunft vor sich: 40% der Stellenausschreibungen im Sicherheitssektor bleiben dieses Jahr wegen Fachkräftemangels unbesetzt, und Penetrationstester sind chronisch überbucht.

Ein Thema, mit dem sich viele Ihrer Kollegen – und vielleicht auch Sie – oft schwertun, ist, ein neues Sicherheitsprogramm an ein nichttechnisches Management zu verkaufen. Beide Seiten »sprechen einfach nicht dieselbe Sprache«. In diesem Geleitwort möchte ich daher versuchen zu erklären, wie Sie die Vorzüge eines Penetrationstests im Unternehmen vermitteln und dadurch benötigtes Budget sicherstellen können.

Wie sage ich es am besten?

Wir haben alle vor dem Angst, was wir nicht verstehen. Daher sollten Sie erst einmal Ihr Management mit dem Konzept eines Penetrationstests vertraut machen. Probieren Sie es einfach mit diesem Beispiel: Wir sollten uns alle in regelmäßigen Abständen einer Gesundheitsuntersuchung unterziehen, auch wenn wir uns eigentlich gesund fühlen. Nur so können schwere Erkrankungen früh erkannt und behandelt werden. Eine solche Untersuchung gehört zu den Aufgaben eines verantwortungsvollen Erwachsenen, der seine Familie und sich langfristig schützen möchte.

Dieses Beispiel lässt sich eins zu eins auf Penetrationstests anwenden, denn auch diese sollten in regelmäßigen Abständen an wichtigen Systemen durchgeführt werden. Nur so können wir erkennen, wo unsere Systeme verletzbar sind. Wir müssen diese Schwachstellen finden, bevor Kriminelle, Spione und Cyber-Vanda-

len unserem Unternehmen Schaden zufügen können. Penetrationstests gehören zu den Instrumenten einer verantwortungsvollen Unternehmensführung, die Risiken identifizieren und mindern möchte. Wie bei einer Gesundheitsuntersuchung vertrauen wir hierfür auf die Meinung ausgebildeter Experten: Ärzten und Penetration-Testern.

Aber wir haben doch eine Firewall!

»Wir haben schon so viel Geld für Sicherheitssysteme ausgegeben, und Sie sagen mir, wir wissen immer noch nicht, ob unsere Systeme sicher sind?«, mag Ihr Manager sagen. Außerdem, sollten Sie Ihre Systeme nicht gut genug kennen, um ihre Schwachstellen zu wissen? Nicht wirklich. Wenn Sie ehrlich sind, können Sie wahrscheinlich nicht einmal beschwören, dass Sie in Ihrem Unternehmen noch keine Datenpanne hatten, denn diese sind nicht immer offensichtlich.

Unsere IT-Systeme sind komplex: organisch gewachsen und mit der Außenwelt an vielen Punkten verknüpft. Es ist in vielen Netzen für einzelne Personen kaum noch möglich, einen Überblick zu behalten. Außerdem könnten Sie die intelligentesten Netzwerk-Spezialisten einstellen, und sie würden trotzdem Fehler machen. Wir brauchen also eine Art Nagelprobe, einen Realitäts-Check, eine Qualitätssicherung für unsere Netzwerksicherheit.

Der Penetrationstest stellt eine solche Qualitätssicherung dar. Sie prüft, ob all unsere Firewalls, Berechtigungssysteme, Intrusion-Detection-Systeme und Data Loss Prevention auch das tun, was wir von ihnen erwarten.

Das Geschäft mit der Angst

Vom Fahrradschloss bis zum Düsenjäger wird Sicherheit primär mit dem Angstfaktor verkauft. Bei Penetrationstests ist dies denkbar einfach: Nehmen Sie die Kosten einer Datenpanne und multiplizieren Sie diese mit der Wahrscheinlichkeit des Eintreffens in einem beliebigen Jahr. So erhalten Sie die potenziellen jährlichen Kosten mangelnder Sicherheit.

Daten hierzu gibt es zur Genüge: Das Ponemon Institute, Verizon Business, Forrester Research, und das FBI veröffentlichen hierzu regelmäßig Daten. Berechnet werden die Wahrscheinlichkeit einer Datenpanne, Kosten von Systemausfällen, der Wert gestohlener/gelöschter/manipulierter Daten, Rechtskosten und verlorener Umsatz durch Kunden, die das Unternehmen verlassen oder wegen des Vorfalls gar nicht erst zum Kunden werden. Aktuell schätzt das Ponemon Institute die Kosten pro verlorenem Kundendatensatz auf 130 Euro (145 US-Dollar). Die durchschnittlichen Kosten pro Datenpanne belaufen sich auf 3,1 Millionen Euro (3,5 Millionen US-Dollar).

Diese Zahlen sind auch sicherlich hilfreich, helfen IT-Sicherheitsfachleuten in Unternehmen aber oft nicht weiter, da die Summen so hoch sind, dass keiner sie für realistisch hält. Außerdem stammen viele der Zahlen aus den USA, wo eine Gesetzgebung, der sogenannte »Data Breach Notification Acts«, die Kosten einer Datenpanne in die Höhe getrieben hat. In Deutschland sind diese Zahlen daher, zumindest bisher, nicht direkt anwendbar. Außerdem müssen diese Zahlen den Kosten aller Sicherheitssysteme gegenübergestellt werden, nicht nur einem einzelnen Penetrationstest.

Sicherheit als Erfolgsfaktor

Penetrationstests über Angst zu verkaufen ist also möglich, aber es gibt auch andere Wege, die bei Ihrem Management eventuell besser ankommen, denn das Geschäft mit der Angst kann im Zweifel als »Erpressungsversuch« interpretiert werden. Und darauf lässt sich keine langfristige Geschäftsbeziehung aufbauen.

Penetration Testing in Kombination mit Vulnerability-Management

Eine Möglichkeit ist zum Beispiel, Penetrationstests als Kostensenker einzusetzen. Viele Unternehmen setzen bereits ein etabliertes Programm für Vulnerability-Management ein, können aber aufgrund der schieren Menge nicht alle Schwachstellen beheben. Eine Penetration-Testing-Software wie Metasploit kann in diesem Fall prüfen, welche Schwachstellen ausnutzbar sind und daher als Erstes behoben werden müssen. Durch eine solche Verfeinerung des Sicherheitsprogramms werden nicht nur die wichtigsten Schwachstellen zuerst behoben, sondern auch die Gesamtkosten für das Beseitigen von Schwachstellen gesenkt, da nicht direkt ausnutzbare Schwachstellen im ersten Schritt ignoriert werden können.

Compliance

Compliance ist oft die Brücke, über die IT-Sicherheitsfachleute mit dem Management kommunizieren können. Manager wissen, dass sie für ihren Geschäftszweig Compliance mit bestimmten Richtlinien benötigen, um Strafen zu vermeiden. Auf der anderen Seite wissen IT-Sicherheitsfachleute, dass sie über diesen Weg neues Budget beantragen können. Compliance bedeutet nicht gleich Sicherheit, aber das Compliance-Budget kann, wenn es sinnvoll eingesetzt wird, zu einer höheren Sicherheit beitragen.

Business Continuity

Viele Argumente für Penetrationstests beziehen sich darauf, was es kostet, wenn Daten gestohlen werden. Kaum eine Argumentation beleuchtet, was es bedeutet, wenn Systeme stillstehen, obwohl dies ebenfalls erhebliche Kosten verursachen

kann. Stellen Sie einfach die Frage: »Was passiert, wenn unser ERP-System eine Woche lang stillsteht?« Dieses Szenario ist für Manager wahrscheinlich deutlich greifbarer, als sich vorzustellen, was passiert, wenn die Kundendaten auf Hackerseiten verkauft werden. Auch die Kosten dürften etwas einfacher zu berechnen sein.

Unternehmensimage

Der Ruf des Unternehmens kann bei einer Datenpanne erheblichen Schaden erleiden, ist aber auch am wenigsten greifbar. Wir werden hier den Ruf des Unternehmens gleichsetzen mit seiner Marke (dem »Brand«). Besonders für Techniker ist das Konzept einer Marke nicht immer offensichtlich, daher nehmen wir einen kurzen Ausflug ins Marketingland.

Bevor wir den Schaden an einer Marke berechnen können, müssen wir uns erst einmal überlegen, wie man den Wert einer Marke berechnet: Stellen Sie sich vor, heute brennen alle Gebäude von Coca-Cola ab. Alle Fabriken, alle Abfüllanlagen, alle Verwaltungsgebäude – alles weg. Ihnen bietet jemand die Rechte an, die Marke Coca-Cola in Zukunft zu verwenden, um Getränke zu verkaufen. Was wäre Ihnen dieses Recht wert? Obwohl das gesamte Unternehmen nicht mehr existiert, hat die Marke noch einen gewissen Wert. Er ist auf jeden Fall nicht null.

Eine Marke ist ein Wiedererkennungsmerkmal für Konsumenten, um mein Produkt gegen das meines Konkurrenten abzugrenzen. Wenn ich das erste Mal in den Supermarkt gehe, um Zuckerwasser zu kaufen, habe ich ohne Marken keine Ahnung, welches ich kaufen soll. Welches schmeckt mir? Habe ich einmal »meine Marke« gefunden, kann ich sie einfach identifizieren und baue ein Vertrauensverhältnis mit ihr auf. Ich weiß, meine Marke steht für gleichbleibende Qualität und wird mich nicht enttäuschen. Sie erleichtert mir die Entscheidung beim nächsten Einkauf.

Außer über einen direkten Kontakt mit dem Produkt versuchen Unternehmen auch durch Werbung mein Vertrauensverhältnis mit der Marke aufzubauen, damit ich ihre Marke als erste ausprobiere oder von einer anderen Marke wechsle.

Viele Unternehmen investieren viel Geld für Werbung – mit steigender Tendenz, denn die Produkte in vielen Segmenten werden immer generischer. Was unterscheidet Ihr Girokonto bei der Sparkasse von dem bei der Deutschen Bank? Wahrscheinlich wenig. Falls Sie nicht Ihren besten Kumpel als Bankberater haben, war Ihre Wahrnehmung vom Unternehmen und Ihre Vertrauensbeziehung zur Marke der größte Entscheidungsträger.

Selbst bei Elektronikgeräten wird der emotionale Teil der Kaufentscheidung immer größer, da Konsumenten immer weniger zwischen den komplexen Modellen verschiedener Hersteller unterscheiden können. Wo eine rationale Entscheidung nicht mehr möglich ist, tritt eine emotionale Enscheidung an dessen Stelle, teilweise unbewusst. Dies mag für Sie als sehr technischen Penetrationstester nicht zutreffen, für das Gros der Konsumenten aber schon.

Überlegen wir uns jetzt, was passiert, wenn dieses Vertrauensverhältnis zu »meiner Marke« durch eine Datenpanne verletzt wird. Als Konsument fühlen wir uns in unserer Privatsphäre verletzt, wenn unser Online-Buchhändler die Kaufhistorie der letzten drei Jahre offenlegt. Vielleicht müssen wir sogar unsere Kreditkarte sperren lassen und haben eine Menge Scherereien. Wenn das Produkt der Konkurrenz identisch mit meinem eigenen ist, fällt die emotionale Entscheidung leicht, das Produkt zu wechseln. Dies hat direkten Einfluss auf den Umsatz des Unternehmens.

Je austauschbarer das Produkt, desto höher der Schaden. Denken wir beispielsweise an wohltätige Organisationen, würde ich wohl kaum ein zweites Mal an Brot für die Welt spenden, wenn diese meine Kreditkartendaten verschlampt haben. Dann ginge ich doch lieber zum Roten Kreuz!

Wie berechne ich einen Business Case?

Da Sie gerade ein Buch über Metasploit lesen und kein Wirtschaftsstudium absolvieren wollen, werden wir an dieser Stelle einen einfachen, pragmatischen Weg wählen. Wenn Sie tiefer in die Thematik einsteigen wollen, empfehle ich das White Paper von Marcia Wilson bei Symantec mit dem Titel »Demonstrating ROI for Penetration Testing« [1], in dem Themen wie Payback Period, Net Present Value, und Internal Rate of Return angeschnitten werden.

Für einen Business Case stellen Sie grundsätzlich zwei Dinge gegenüber: Was ist, und was könnte sein. Das »was könnte sein« ist Ihr Vorschlag. Wenn dieser Vorschlag weniger Geld kostet (oder mehr Umsatz bringt) als das, »was ist«, haben Sie einen guten Business Case. In der IT-Sicherheit lässt sich ein solcher Business Case nicht immer gut berechnen – in manchen Fällen aber schon. Wir müssen hier je nach Szenario unterscheiden.

Neue Einführung von Penetrationstests

Wenn Sie bisher keine Penetrationstests durchgeführt haben, haben Sie aktuell keine merklichen Kosten. Um einen Business Case aufzubauen, müssen Sie die Kosten einer Datenpanne oder eines Systemausfalls berechnen und diesen mit der Wahrscheinlichkeit des Eintretens multiplizieren. Hier bleibt leider nur die Angst vor einer Datenpanne als Argumentation.

Beispiel: Ihr ERP-System beinhaltet 10.000 Kundendaten. Laut The Ponemon Institute belaufen sich die Kosten pro verlorenem Datensatz auf 130 Euro (145 US-Dollar) und bei einem Gesamtschaden auf 1.300.000 Euro. Forrester schätzt, dass 60% der Unternehmen im Jahr 2015 mindestens eine Datenpanne erleiden werden, also ist die Wahrscheinlichkeit des Eintretens 60%. Der Wert des Risikos einer Datenpanne ist also 1.300.000 Euro × 60% = 780.000 Euro.

Alternativ rechnen wir aus, was der Ausfall des ERP-Systems kosten würde. Nehmen wir an, die Kosten eines Ausfalls belaufen sich auf 1 Million Euro pro Tag, und das System wäre für 3 Tage außer Gefecht gesetzt. Bei einer Eintrittswahrscheinlichkeit von 10% wären dies 3 × 1.000.000 Euro × 10% = 300.000 Euro.

Im Kontrast zu diesen potenziellen Kosten dürften Ihre geplanten Kosten für Penetrationstests recht gut aussehen. Die Frage ist, ob Ihre Berechnungen als realistisch angesehen werden.

Alternativ können Sie einfach etwas Business Jiu Jitsu anwenden, indem Sie den Penetrationstest nicht im luftleeren Raum, sondern als Teil eines Projekts unterbringen. Suchen Sie sich ein Projekt aus, das aktuell auf der Liste der Management-Ziele Ihres CIO steht. Wenn Sie die Ziele Ihres CIO nicht kennen, fragen Sie ihn einfach – und bieten Sie Ihre Hilfe an! Nehmen wir an, Ihr CIO soll in diesem Quartal 20% der externen Zulieferer per Web Services an das ERP-System anbinden. Sie können nun Ihre Hilfe für dieses Projekt anbieten und damit einen Penetrationstest in die Abnahme der Systeme einbauen. Statt nur die Web Services selbst im Penetrationstest zu prüfen, sollte selbstverständlich das gesamte ERP-System getestet werden. So werden Sie mit Ihrem Sicherheitsfachwissen zum Berater und helfen, die Technologie im Unternehmen sicher voranzutreiben.

Penetrationstests für Vulnerability-Management

Wollen Sie Penetrationstests einführen, um die Remediation-Kosten für Ihr Vulnerability-Management-Programm zu senken, sieht die Berechnung etwas anders aus:

Nehmen wir an, Sie haben drei Netzwerkadministratoren, die im Schnitt 65.000 Euro kosten. Wenn jeder dieser Mitarbeiter 20% seiner Zeit damit verbringt, Updates zu installieren und Schwachstellen zu beheben, kostet dies das Unternehmen jährlich 39.000 Euro. Wenn Penetrationstests diese Arbeit auf je 10% minimieren können, weil die Mitarbeiter nur Schwachstellen beheben, die ausnutzbar sind, spart das Unternehmen dadurch 19.500 Euro. Sie sollten außerdem in die Überlegung einbeziehen, dass die Mitarbeiter nun an Schwachstellen arbeiten, die wirklich ausnutzbar sind, und dadurch das Unternehmensnetz besser geschützt ist.

Penetrationstests intern durchführen

Wenn Sie bisher Penetrationstests durch ein externes Beratungsunternehmen haben durchführen lassen, möchten Sie diese Tests vielleicht jetzt intern durchführen und damit Geld sparen. In diesem Fall ist die Berechnung einfach, da Sie die aktuellen externen Kosten einfach den neuen internen Kosten gegenüberstellen können.

Gerade wenn Sie regelmäßig interne Penetrationstests durchführen, lohnt sich auch ein Blick auf Metasploit Pro, die kommerzielle Version von Metasploit, mit der Sie die Penetrationstests effizienter durchführen können, weniger Training benötigen und eine größere Anzahl Maschinen mit weniger Aufwand testen können.

Ziele eines Penetrationstests

Wichtig bei der Präsentation eines Business Case ist es auch, die Ziele deutlich zu kommunizieren, zum Beispiel:

- Demonstration der Verwundbarkeit der Systeme, um die Aufmerksamkeit und Unterstützung des Managements für neue Sicherheitsprogramme zu erlangen
- Senkung der Kosten eines Vulnerability-Management-Programms
- Bestandsaufnahme für neue CIOs oder CISOs
- Hilfe für Entscheidung, worauf das Sicherheitsbudget verwendet werden soll
- Testen der Response-Mechanismen von IDS-, IPS- und DLP-Systemen (Metasploit-vSploit-Module)
- Penetrationstest aus Compliance-Gründen

Fazit

Wie eine regelmäßige Gesundheitsuntersuchung gehört ein Penetrationstest zum verantwortungsvollen Verhalten eines Unternehmens. Mit der Auswahl von Metasploit als Werkzeug für dieses Unterfangen haben Sie eine hervorragende Wahl getroffen. Metasploit ist mit mehr als einer Million Downloads pro Jahr das am weitesten verbreitete Penetration-Testing-Werkzeug der Branche. Somit sind Tests mit Metasploit nahe an der Realität eines echten Angriffs.

Pentester sind aktuell sehr gefragt und werden gut bezahlt. Mit dem Spezialwissen über Metasploit, das Sie sich mit diesem Buch aneignen, werden Sie Ihren persönlichen Wert am Arbeitsmarkt nachhaltig steigern. Wichtig ist aber in jedem Fall ein solides Fachwissen, damit Sie mit dem Penetrationstest keine Systemabstürze oder Netzwerküberlastungen erzeugen.

Sollten Sie Penetrationstests zu Ihrer Haupttätigkeit machen, können Sie Ihr erworbenes Wissen auch in den kommerziellen Versionen von Metasploit weiter nutzen, die Ihnen durch Automatisierungen und Teamkollaboration ein effizienteres Arbeiten ermöglichen und dröge Aufgaben wie Beweismittelsicherung und Berichteschreiben weitgehend abnehmen.

In jedem Fall sollten Sie in Ihrem Unternehmen daran mitarbeiten, Penetration Tests in den Sicherheits-Lebenszyklus zu integrieren, so dass kein neues System ohne Penetrationstest in Produktion geht. Wenn Ihre Kollegen fragen, wann sie einen Penetrationstest durchführen sollten, antworten Sie einfach: »Wann sollten Sie im Auto einen Sicherheitsgurt anlegen?« Immer.

Christian Kirsch[1]

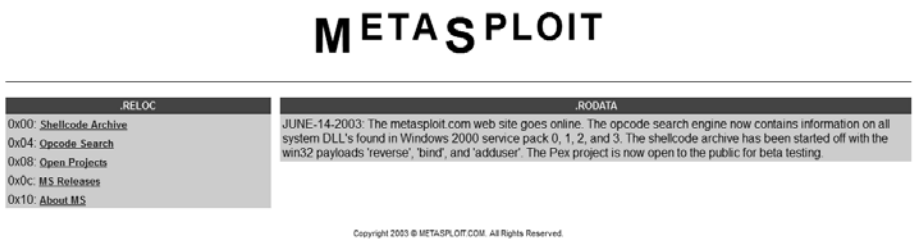
1 Christian Kirsch war Principal Product Marketing Manager bei Rapid7, der Firma, die seit 2009 für die Entwicklung des Metasploit-Framework verantwort-lich ist.

Vorwort

Das Metasploit-Framework ist dort, wo es um Penetrationstests, Sicherheitsanalysen und Forschung im IT-Security- und speziell im Schwachstellenbereich geht, nahezu immer anzutreffen. Wenn von Metasploit gesprochen wird, geht es aber nicht um ein einziges Tool, sondern um eine sehr umfangreiche und komplexe Toolbox, die in Fachkreisen als Framework bezeichnet wird. Dieses Framework besteht aus unterschiedlichsten Teilbereichen, Teilprojekten und Modulen und ist fester Bestandteil der Werkzeugkiste nahezu jedes Pentesters. Der große Umfang ermöglicht einen Einsatz, der weit über typische Exploiting-Vorgänge hinausgeht und eine Anwendung in nahezu allen Phasen eines Penetrationstests bzw. einer technischen Sicherheitsanalyse erlaubt.

Das Framework unterstützt aber nicht nur den Pentester bei seiner täglichen Arbeit, sondern auch den Sicherheitsforscher bei der Erkennung und Analyse potenzieller Schwachstellen und den Administrator bei der besseren Einschätzung vorhandener Schwachstellen.

Die Entwickler von Metasploit gehörten zu den ersten Sicherheitsexperten, die durch ihre Forschungsarbeiten unterschiedliche Exploit-Technologien einem breiten Publikum zugänglich machten. Bereits mit der ersten Veröffentlichung dieses Frameworks im Jahr 2003 sorgten dessen freie Natur und der damit verbundene freie Zugang zu Informationen zur Erkennung und Ausnützung von Schwachstellen für erheblichen Diskussionsstoff. Speziell die Hersteller der betroffenen Produkte sind an keinem freien Zugang zu solchen Informationen interessiert und versuchen, diesen entsprechend zu verhindern.



Metasploit-Webseite aus dem Jahr 2003 [2]

Diese Diskussionen sind in all den Jahren nicht verstummt und werden bis heute regelmäßig erneut entfacht. Hier seien nur kurz die wichtigsten Methoden der Schwachstellenveröffentlichung *Full Disclosure* [3], *Coordinated* und *Responsible Disclosure* [4] [5] angeführt. Für weitere Informationen zu den einzelnen Methoden der Veröffentlichung wird auf die im Anhang angegebenen Online-Ressourcen verwiesen.

Das Jahr 2009/2010 war für das Metasploit-Framework wie auch für die Community wohl eines der spannendsten in der mittlerweile achtjährigen Entwicklungsgeschichte. Durch den neuen Mitspieler Rapid7, einen Hersteller von Vulnerability-Scanning-Lösungen, machte das Metasploit-Framework einen enormen Sprung nach vorne. Mittlerweile lassen sich jeden Tag Änderungen in der Entwicklerversion beobachten. Diese enorm schnelle Entwicklung führte in der jüngeren Vergangenheit zur Veröffentlichung von sechs neuen Versionen innerhalb eines Jahres. Zusätzlich kam es durch den Einfluss von Rapid7 zur Etablierung von zwei neuen, kommerziellen Versionen des Frameworks: Metasploit Express und Metasploit Pro. Durch diese Entwicklungsgeschwindigkeit ist es kaum mehr möglich, alle aktuellen Neuerungen zu kennen und möglichst zeitnah zu testen. Die oftmals nur sehr spärlich über verschiedenste Blogs verteilte Dokumentation macht es neuen Benutzern zudem nicht unbedingt einfacher, sich mit dem Thema *Pentesting mit Metasploit* im Detail zu befassen.

Dieses Buch soll das Metasploit-Framework möglichst umfassend dokumentieren und Interessierten einen Einstieg in diese spannende Thematik ermöglichen. Gleichzeitig will es diejenigen, die sich bereits längere Zeit mit dem Framework befassen, das eine oder andere weitere und spannende Detail oder die eine oder andere neue Idee vermitteln.

Dieses Buch soll sozusagen die Basis abdecken, mit der ein Pentester arbeiten kann und auf der er aufbauen kann. Neue Versionen zu testen, die aktuellen Entwicklungen beobachten und evtl. auch Codeteile des Frameworks zu lesen, wird durch dieses Buch aber sicherlich nicht weniger aufwendig.

Wie ist dieses Buch aufgebaut?

Nach einer ersten Erklärung, was das Metasploit-Framework ist, stellt das Buch zunächst das Thema Informationsgewinnung vor und beschreibt einen ersten Exploiting-Vorgang. Anschließend werden Automatisierungsmöglichkeiten des Frameworks betrachtet, gefolgt von weiteren sehr speziellen Themengebieten, die im Rahmen eines Penetrationstests und im IT-Security-Prozess von Belang sind.

Im ersten Abschnitt wird das Thema Pentesting und Exploitation möglichst allgemein betrachtet, wodurch dem Leser ein Einstieg in diese Thematik ermöglicht wird. Es werden beispielsweise alternative Exploiting-Frameworks und Tools dargestellt, die den Pentester im Rahmen seiner Dokumentationserstellung unterstützen können.

In folgenden Abschnitten werden unterschiedlichste Module für Informationsgewinnungs- und Scanning-Vorgänge behandelt. Zudem wird betrachtet, wie unterschiedlichste Exploits und Payloads eingesetzt werden. Neben Automatisierungsmechanismen werden zudem Penetrationstests von Webapplikationen und Datenbanken betrachtet, gefolgt von einer detaillierten Vorstellung unterschiedlichster Methoden der Post-Exploitation-Phase. Die abschließenden Abschnitte des Buches behandeln dann die kommerziellen Versionen des Frameworks und den IT-Security-Research-Bereich. In dem Abschnitt zur Schwachstellenerkennung und Exploit-Entwicklung wird eine Schwachstelle in einer von KMDave speziell entwickelten Testapplikation gesucht und analysiert. Anhand dieser Analyse, mit einem sogenannten Fuzzer, wird dargestellt, wie eine Entdeckung dieser Schwachstelle möglich ist, um im Anschluss einen voll funktionsfähigen Exploit zu erstellen.

Wer sollte dieses Buch lesen?

Dieses Buch richtet sich an Pentester sowie an IT-Sicherheitsverantwortliche und Systemadministratoren mit vorwiegend technischen, aber auch organisatorischen Berührungspunkten zur IT-Security. Darüber hinaus ist es für den Einsatz in IT-Security-Studiengängen bzw. in Studiengängen mit IT-Security-Schwerpunkt geeignet und für jeden, der Interesse an Pentesting- und Exploiting-Frameworks mitbringt und sein Wissen in diesen Bereichen vertiefen möchte.

Im Rahmen dieses Buches werden keine typischen IT- und Security-Grundlagen, wie beispielsweise TCP/IP und Portscans, behandelt. Es wird vorausgesetzt, dass Sie als Leser die Grundlagen der Netzwerk- und Systemtechnik sowie der IT-Security bereits mitbringen oder sich dieses Wissen bei Bedarf anderweitig aneignen. Relevante Grundlagen des Pentesting-Vorgangs werden in den ersten Abschnitten kurz dargestellt, umfassen allerdings keine vollständige Abhandlung von Penetrationstests.

> Der Leser dieses Buches wird durch die Lektüre zu keinem Pentester. Dieses Buch kann den geneigten Leser aber auf dem Weg dorthin begleiten.

Dieses Buch wird unterschiedlichste Beispiele aus dem praktischen Leben eines Pentesters darstellen und sie in einem Testlabor umsetzen. Um diese Beispiele im eigenen Labor nachzustellen, sollten Sie die Möglichkeit haben, verschiedene Windows- und Linux-Systeme in einer physikalischen oder virtualisierten Umgebung einzurichten. Sie sollten dabei imstande sein, diese Systeme mit unterschiedlichsten Diensten, Konfigurationen und/oder weiterer Software auszustatten.

Allein das Lesen dieses Buches macht aus Ihnen keinen Pentester. Sie müssen sich schon »die Hände schmutzig machen« und Systeme in einer Testumgebung wirklich angreifen.

Strafrechtliche Relevanz

Die in diesem Buch dargestellten Tools und Techniken lassen sich neben den hier behandelten legalen Einsatzszenarien unter Umständen auch für nicht legale Aktivitäten nutzen.

An dieser Stelle muss ausdrücklich festgehalten werden, dass die in diesem Buch beschriebenen Vorgänge ausschließlich in einer gesicherten Testumgebung oder mit der Einwilligung des Systembesitzers zur Anwendung gebracht werden dürfen. Werden Angriffe dieser Art auf Systemen durchgeführt, für die keine ausdrückliche Erlaubnis erteilt wurde, stellt dies im Normalfall eine strafrechtlich relevante Handlung dar. Der Autor oder der Verlag können dafür in keinster Weise belangt werden.

Danksagungen

Irgendwann im Laufe eines persönlich wie beruflich sehr spannenden Jahres 2010 sprach mich jemand im IRC darauf an, ob ich nicht ein Buch zu Metasploit im Pentesting-Umfeld schreiben wolle. Eineinhalb Jahre später gibt es dieses Buch nun. Ich habe leider keine Ahnung mehr, wer mir diese Idee in meinen Kopf eingepflanzt hat. Falls sich einer der Leser angesprochen fühlt, möchte ich mich bei ihm bedanken und hoffe, dieses Buch entspricht seinen Vorstellungen und bereitet dem Ideengeber wie auch allen anderen Lesern möglichst viel Freude!

Folgenden Personen möchte ich speziell danken:

- Meiner ganzen Familie,
- Carina und den Mädels für eine traumhafte Zeit, ihr seid die Besten,
- ChriGu – ihr zwei seid einfach spitze! Vielen Dank für die Unterstützung …
- Viktoria Plattner für eine wunderschöne Reise, durch die dieses Buch wohl erst ermöglicht wurde, zudem möchte ich dir für die Abbildung 1–1 und Abbildung 8–1 danken,
- Dave für die Zusammenarbeit am Kapitel zur Exploit-Entwicklung,
- Holger und dem PS-ISM-Team für die Unterstützung seitens der Integralis,
- René und dem dpunkt.verlag für das Vertrauen, die Unterstützung und alle Einflüsse,
- Christian Kirsch für die Unterstützung und das tolle Geleitwort,
- HDM und dem gesamten Metasploit-Team für ein geniales Framework,
- allen Freunden, Gutachtern und Helfern, die dieses Buch erst möglich gemacht haben und mich im letzten Jahr etwas weniger zu Gesicht bekamen ;),
- allen Lesern der ersten und zweiten Auflage. Zudem noch ganz speziell Thomas Wallutis, Klaus Gebeshuber, Jörn A., Pascal Winkler und Christian Kunze für das Feedback.

Michael Messner, im September 2017

Inhaltsverzeichnis

1 Eine Einführung in das Pentesting und in Exploiting-Frameworks

Bevor ich im weiteren Verlauf des Buches mit einer detaillierten Darstellung des Metasploit-Frameworks und dessen praktischer Anwendung beginne, betrachten wir im folgenden Kapitel zunächst einige grundlegende Aspekte rund um die Pentesting-Thematik.

Unter anderem werden wir die einzelnen Phasen eines Penetrationstests betrachten. Ich werde außerdem erläutern, worum es sich bei einem Exploiting-Framework handelt und was es typischerweise umfasst. Neben Metasploit gibt es noch weitere, weit verbreitete Frameworks, die in einem eigenen Abschnitt vorgestellt werden. Ebenso lernen Sie einige Dokumentationswerkzeuge kennen. Schließlich stellen wir Überlegungen zum eigenen Testlabor an und betrachten unterschiedliche Lern- und Testsysteme.

1.1 Was ist Pentesting?

Prinzipiell geht es im ersten Schritt eines Pentests darum, Schwachstellen zu erkennen und sie im Anschluss zu bewerten, um darauf basierend geeignete Gegenmaßnahmen erarbeiten zu können. Während automatisierte Vulnerability-Scans im Grunde genommen dieselbe Zielsetzung haben, werden die Ergebnisse eines professionellen Penetrationstests erheblich detaillierter und durch die manuelle Arbeit umfangreicher und korrekter sein. Durch die manuellen Tätigkeiten des Pentesters werden die Ergebnisse eines Penetrationstests in der Regel keine bzw. kaum Schwachstellen der Kategorie *False-Positive* beinhalten.

> Als False Positives werden »falsch« gemeldete Schwachstellen bezeichnet, die zwar häufig von automatisierten Tools als Schwachstellen eingestuft werden, allerdings auf dem Zielsystem entweder gar nicht vorhanden sind oder aufgrund vorhandener Gegenmaßnahmen nicht ausnutzbar sind.

Während Vulnerability-Scanner typischerweise ausschließlich Schwachstellen erkennen, wofür der Hersteller dieses Scanners entsprechende Module integriert hat, verfügt ein Pentester über weitere Möglichkeiten, potenzielle Schwachstellen

auszumachen. Im einfachsten Fall reicht bereits eine einfache Suche nach einer erkannten Versionsnummer auf einem der bekannten Internetportale für Exploit-Code aus. Zudem haben Vulnerability-Scanner typischerweise das Problem, dass sie nicht imstande sind, potenzielle Schwachstellen zu verifizieren, wodurch es zur bereits erwähnten False-Positive-Problematik kommt.

Information: Es gibt auch Vulnerability-Scanner, die Exploits integriert haben und dadurch oftmals die dargestellte Problematik in Teilbereichen umgehen können.

Der Scanner glaubt bei False-Positives, eine Schwachstelle erkannt zu haben, kann sie allerdings nicht durch den Einsatz von Exploit-Code oder weiteren Tools bzw. Angriffsmethoden bestätigen. Im darauf basierenden Bericht wird dementsprechend eine kritische Schwachstelle aufgeführt, die das geprüfte System allerdings nicht aufweist. Ein Pentester wird typischerweise im Rahmen seiner Tätigkeiten einen Schritt weitergehen und die Schwachstelle durch manuelle Arbeiten wie den Einsatz weiterer Tools, Module oder eines Exploits verifizieren. Dieser zusätzliche manuelle Schritt ermöglicht in den meisten Fällen eine klare Bewertung, ob eine Schwachstelle nicht nur *möglicherweise* vorhanden ist und sich *möglicherweise* für eine Kompromittierung eines Systems eignet, sondern dass es sich um ein *tatsächlich* vorhandenes und kritisches Bedrohungsszenario handelt. Auf Basis solcher Ergebnisse lassen sich entsprechend klare Empfehlung aussprechen. Solche Empfehlungen mit einem tatsächlich vorhandenen Bedrohungsszenario sind ungemein wichtig, um eine korrekte Priorisierung seitens der Verantwortlichen erst möglich zu machen. Diese sollten sofort erkennen, um welche Schwachstellen sie sich unverzüglich kümmern müssen und welche eine weitere, interne Bewertung nach sich ziehen können.

Viele Systeme und Applikationen sind zudem hochkomplex. Als Beispiel sei hier eine spezielle intern programmierte Webapplikation angeführt. Auch für Analysetools, die auf Webapplikationen optimiert sind, ist es häufig nicht möglich, solche Applikationen vollständig und automatisiert auf Schwachstellen zu testen. Ein Pentester wird an dieser Stelle durch manuelle Analyse die Funktionsweise der Applikation analysieren, wodurch es überhaupt erst möglich wird, weitere Schwachstellen zu erkennen und diese beispielsweise im Anschluss für verkettete Angriffe zu nutzen. Durch solche verketteten Angriffe kann eine mögliche Eskalationskette ermittelt werden, in der unterschiedliche Schwachstellen miteinander kombiniert werden, um dadurch das tatsächliche Bedrohungsszenario darzustellen.

Folgendes Szenario stellt ein kleines Beispiel einer möglichen Eskalationskette dar, die sich im Rahmen eines durchgeführten Penetrationstests in ähnlicher Weise abgespielt hat:

Im Rahmen einer umfangreichen Sicherheitsanalyse eines international tätigen Konzerns wird eine Simulation eines gestohlenen Notebooks durchgeführt. Unternehmen bzw. IT-Abteilungen, die eine hohe Anzahl mobiler Geräte verwalten und absichern müssen, sind häufig von einer entsprechend hohen Verlustzahl dieser Geräte betroffen. Werden keine speziellen Sicherheitsmaßnahmen zum Schutz sensibler Daten eingesetzt, ist es einem Angreifer unter Umständen möglich, ein gestohlenes Notebook für einen erfolgreichen Zugriff auf das interne Unternehmensnetzwerk zu nutzen.

Bei der durchgeführten Analyse des Notebooks ist es wegen fehlender Festplattenverschlüsselung möglich, das System nach Datenspuren und Passwörtern zu analysieren. In der History des Browsers lässt sich die Internetadresse der SSL-VPN-Verbindung auslesen, und der nicht gesicherte Passwortsafe liefert die benötigten Informationen für einen erfolgreichen Anmeldevorgang.

Der Pentester liest noch den Windows-Passwort-Hash des lokalen Administrator-Accounts aus und meldet sich über das SSL-VPN im Unternehmensnetzwerk an. Hierfür konnten die bereits ermittelten Benutzerinformationen des nicht gesicherten Passwortsafes genutzt werden. An dieser Stelle hat der Angreifer einen nicht privilegierten Zugriff auf das Unternehmensnetzwerk erhalten. Dieser nicht privilegierte Zugang dient im weiteren Verlauf sozusagen als Sprungbrett in das interne Netzwerk und ermöglicht weiterführende Angriffe.

Anmerkung: Eine sogenannte Zweifaktor-Authentifizierung hätte einen erfolgreichen Anmeldevorgang an dieser Stelle erheblich erschwert oder sogar unmöglich gemacht.

Nachdem die Administratoren auf allen Systemen dasselbe lokale Administrator-Passwort einsetzen, konnte sich der Pentester unter Zuhilfenahme des ausgelesenen Windows-Hash sowie der *Pass-the-Hash*-Methode (diese wird im Verlauf des Buches, in Abschnitt 9.2, noch detailliert dargestellt) und ohne Wissen des Klartext-Passwortes direkt an weiteren Systemen anmelden. Dies ermöglichte ihm weiteren Systemzugriff mit lokalen administrativen Berechtigungen. Als lokaler Administrator angemeldet lässt sich erkennen, dass er unter anderem auf einem System gelandet ist, auf dem vor kurzem ein Domain-Administrator angemeldet war. Bei einer solchen Anmeldung hinterlässt der Benutzer automatisch sein Authentifizierungstoken auf dem System, das sich unter Umständen weiterhin auf dem System befindet und sich für Angriffe einsetzen lässt. Im folgenden Schritt ist es dem Pentester dann möglich, das Token des Domain-Administrators zu übernehmen und dadurch die Identität dieses wichtigen Domain-Users (siehe Abschnitt 5.8.1). Der Pentester kann sich ab sofort im internen Netzwerk als vollwertiger Domain-Administrator bewegen, einen neuen administrativen Domain-User anlegen und dadurch seinen weiteren Zugang zum Netzwerk sichern.

Dem Pentester war es in unserem Beispiel durch die Kombination mehrerer Schwachstellen bzw. teilweise durch Konfigurationsfehler möglich, ausgehend von

einem mobilen System die vollständige interne Windows-Domäne erfolgreich anzugreifen und zu kontrollieren. Was sich als ein schönes Ergebnis für einen Pentester darstellt, ist im typischen, unkontrollierten Fall eines Angriffs für das betroffene Unternehmen eine sicherheitstechnische Katastrophe.

Hinweis: Bei solchen Pentests ist unbedingt vorab der Umfang (Scope) des Tests abzuklären. Das Ziel eines Pentests ist es nicht, die zu analysierende Infrastruktur zu gefährden.

1.2 Die Phasen eines Penetrationstests

Wenn es um die Durchführung von Penetrationstests geht, wird häufig von Voodoo, geheimen Hackertricks und undurchsichtiger, oftmals nicht vollständig legaler Vorgehensweise gesprochen. Jeder Pentester fand sich wohl schon das eine oder andere Mal in einem solchen Gespräch und überlegte schmunzelnd, ob er diese Gerüchte nun wirklich auflöst oder ob er den Gegenüber besser in seinem Glauben lassen solle.

Professionelle Penetrationstests haben nichts mit Magie, Voodoo und auch sehr wenig mit geheimen Hackertricks gemein. Die Vorgehensweise von Penetration-Tests ist normalerweise sehr einheitlich und wurde von unterschiedlichsten Institutionen formuliert. Folgende Darstellung bezieht sich auf die fünf Phasen eines Penetrationstests, wie sie vom Bundesamt für Sicherheit in der Informationstechnik (BSI) dargestellt wurden [6]:

- Phase 1: Vorbereitung
- Phase 2: Informationsbeschaffung und auswertung
- Phase 3: Bewertung der Informationen/Risikoanalyse
- Phase 4: Aktive Eindringversuche
- Phase 5: Abschlussanalyse

Im weiteren Verlauf dieses Abschnitts werden diese einzelnen Phasen eines Penetrationstests dargestellt, wobei dabei bereits die Eignung des Metasploit-Frameworks in den einzelnen Bereichen einfließt.

Hinweis: In unterschiedlichsten Dokumenten werden die dargestellten Phasen oftmals in etwas anderen Aufteilungen und dadurch in weniger oder mehr Phasen dargestellt. Die durchzuführenden Punkte und Aufgaben unterscheiden sich allerdings prinzipiell nicht. In Abschnitt 1.2.6 wird eine etwas andere Aufteilung grafisch dargestellt.

Weitere Informationen zur typischen Vorgehensweise bei Penetrationstest sind neben den dargestellten Details vom BSI in den Dokumenten der OISSG (Open Information System Security Group) mit dem »Information Systems Security Assessment Framework« (ISSAF) [7] oder im »Technical Guide to Information Security Testing and Assessment« vom NIST [8] zu finden.