HARD CORE IT SECURITY MAGAZINE

Hakin9 EXTRA

Hakin9 Extra Ausgabe 3/2012

BACKTRACK 5

DATENSAMMLUNG UND REPORTERSTELLUNG FÜR PENTESTER MIT MAGICTREE

IDENTIFIZIERUNG VON SCHWACHSTELLEN BEI WEBANWENDUNGEN MIT SKIPFISH

PLUS

EINFÜHRUNG IN DIE MAN-IN-THE-MIDDLE-ATTACKEN

Hakin9

Herausgegeben vom Verlag:
Hakin9 Media Sp. z o.o. Sp. Komandytowa

Geschäftsführer:
Paweł Marciniak

Managing Director:
Grzegorz Tabaka

Chefredakteurin:
Katarzyna Kwapińska
katarzyna.kwapinska@software.com.pl

Redaktion:
Bernd Schwedler, Andy Stern,
Thomas Kreichgauer, Sven Amberger,
Tobias Willers, Nils Kuhnert,
Martin Weber, Igor Kochanow.

Produktion:
Andrzej Kuca

DTP:
Marcin Ziółkowski

Umschlagsentwurf:
Marcin Ziółkowski

Werbung: *adv@software.com.pl*

Anschrift:
Hakin9 Media Sp. z o.o. Sp. Komandytowa
ul. Bokserska 1, 02-682 Warszawa,
Poland
Tel. +48 22 427 36 56,
Fax +48 22 244 24 59
www.hakin9.eu

Liebe Hakin9 Leser,

Backtrack, die Linux-Distribution, die zur Überprüfung der Sicherheit einzelner Rechner sowie der Gesamtsicherheit des Netzwerks dient, hat seit März 2012 eine neue Version. Backtrack 5 beinhaltet viele Programme und Funktionen, die die Sicherheit eines Computers oder eines ganzen Netzwerkes gewährleisten können.

Die vorliegende Ausgabe von Hakin9 Extra beinhaltet Artikel, die verschiedene Funktionen und Möglichkeiten von Backtrack 5 präsentieren. Sie erfahren aus ihnen, welche Tools von Backtrack 5 Ihnen zur Verfügung stehen, um die Sicherheit eines Netzwerkes zu gewährleisten und wie ein Sicherheitstest durchgeführt werden kann, um Sicherheitslücken zu vermeiden.

Ausserdem präsentieren wir in dieser Ausgabe, wie die Angriffe mithilfe von Backtrack 5 durchgeführt werden können oder wie es einfach ist, in andere Netzwerke einzudringen oder ein Passwort zu knacken. Damit wollen wir Sie darauf aufmerksam machen, dass die Computer- und Netzwerksicherheit eine sehr wichtige Rolle spielt. Deshalb zeigen wir Ihnen auch Möglichkeiten, welche uns zur Verfügung stehen, um sich vor solchen Angriffen zu schützen.

Diese Ausgabe sollte Ihnen einen Überblick über verschiedene Möglichkeiten von Backtrack 5 geben. Wir legen grossen Wert auf die praktische Seite der beschriebenen Inhalte, deshalb ist jeder Schritt mit Grafiken veranschaulicht. Wir hoffen, dass Sie viel Vergnügen am Lesen finden und nach der Lektüre Lust haben werden, Backtrack 5 zur Überprüfung der Sicherheit Ihrer Computer und Netzwerke zu verwenden.

Viel Spass bei der Lektüre!
Katarzyna Kwapińska

Falls Sie Interesse an einer Kooperation oder Themenvorschläge hätten, wenden Sie sich bitte an unsere Redaktion: de@hakin9.org

4. BACKTRACK 5 R2 – EINFÜHRUNG IN DIE MAN-IN-THE-MIDDLE-ATTACKEN

Lothar Serra Mari

„Backtrack" ist eine unter Security-Spezialisten beliebte Linux-Distribution zur Überprüfung der Sicherheit einzelner Rechner oder ganzer Netzwerke. Die Distribution beinhaltet viele Programme, mit denen unterschiedlichste Aspekte der Netzwerk- und Computersicherheit überprüft werden können. Darunter befinden sich auch Tools, mit denen sog. „Man-in-the-middle"-Angriffe (kurz: MITM) durchgeführt werden können. Der folgende Artikel demonstriert verschiedene, einfache MITM-Angriffe mithilfe von Netzwerk-Sniffern, ARP- und DNS-Spoofing. Alle Tools sind dabei in Backtrack 5 R2 enthalten. Dabei sollen die Schwächen der Protokolle ARP, DNS und SSL demonstriert und einfache Schutzmöglichkeiten gezeigt werden.

8. BACKTRACK 5

Martin Schagerl

Da Sicherheit in IT Netzwerken und Applikationen immer mehr an Bedeutung gewinnt, versuchen viele Unternehmen ihre Infrastruktur sowie Ihre eingesetzte Software auf Sicherheitslücken zu durchsuchen. Die verantwortlichen Personen wissen dabei oft nicht, welche Tools für diesen Zweck zu Verfügung stehen und sind unschlüssig, welches Betriebssystem verwendet werden sollen. Die Antwort ist einfach: Backtrack 5 – Eine Linux Distribution mit einer großen Anzahl von vorinstallierten Securitytools. In dem folgenden Artikel erfahren Sie mehr über die wichtigsten Applikationen von Backtrack 5 sowie eine Anleitung um eine fremde Maschine anzugreifen.

12. BACKTRACK 5: DATENSAMMLUNG UND REPORTERSTELLUNG FÜR PENTESTER MIT MAGICTREE

Hans Höfken, Marko Schuba

Bei der Durchführung von Schwachstellen- und Penetrationstests fallen häufig eine große Menge Daten an. Hier den Überblick zu behalten ist oft nicht so einfach. MagicTree kann Sie unterstützen und das (Pentester-) Leben vereinfachen. Es liegt aktuell in der Version 1.1 vor und wird stetig gepflegt und erweitert. Was dieses Tool kann und wie Sie es für Ihre Zwecke einsetzen können, zeigt dieser Artikel.

18. BACKTRACK 5

Julius Biermann

Die kostenlose Linux Distribution Backtrack umfasst viele integrierte Funktionen und Tools. Unter anderem auch das Tool: „Aircrack". Dieses Programm, welches sich über die systemeigene Shell bedienen lässt, knackt durch das Mitschneiden von Datenpaketen und der bekannten Wörterbuchattacke das Wlan Passwort eines Routers. Dieser Arktikel bringt Ihnen Schritt für Schritt die Einzelheiten näher und erleutert diese.

21. IDENTIFIZIERUNG VON SCHWACHSTELLEN BEI WEBANWENDUNGEN MIT SKIPFISH

Mike Kuketz

Ein Alltag ohne Webanwendungen ist heute kaum noch vorstellbar. Unzählige Anfragen werden täglich von Webservern verarbeitet und anschließend im Browser des Nutzers dargestellt. Eine gute Strategie, um eine Webanwendung bzw. den dahinter liegenden Webserver abzusichern ist eine potenzielle Schwachstelle vor einem Angreifer zu finden. Dazu eignen sich diverse Tools – sogenannte Web Vulnerability Scanner. Diese automatischen Scanner werden eingesetzt, um zunächst eine Grundlage zu schaffen, die wiederum als Ausgangspunkt für manuelle Tests dient. Eines dieser automatisch agierenden Scanner nennt sich skipfish und wird in diesem Artikel vorgestellt.

25. BACKTRACK 5 R2 – DAS SCHWEIZER ARMEEMESSE FÜR IT-NINJAS

Patrick Blom

In diesem Artikel bekommen Sie einen praktischen Einblick in das Betriebssystem BackTrack5 R2 sowie in die AirCrack-ng Suite. Wir werden Informationen über BackTrack5 R2 sammeln und die Fragen klären, was BackTrack5 überhaupt ist und wozu BackTrack5 verwendet wird.

An einem praktischen Step by Step Beispiel demonstrieren wir dann, wozu BackTrack5 R2 genutzt werden kann. Die AirCrack-ng Suite wird ein zentraler Bestandteil dieses Artikels sein, da wir mit ihrer Hilfe die Schwachstellen der WEP-Verschlüsselung ausnutzen, sowie die Verschlüsselung an sich aushebeln werden. Des weiteren macht dieser Artikel sehr gut deutlich, wie unsicher die WEP-Verschlüsselung ist und warum Sie diese nicht nutzen sollten.

30. BACKTRACK 5 -WLAN MIT WPA/2-VERSCHLÜSSELUNG KNACKEN

Tysonpower

Aus diesem Artikel erfahren sie, wie man mit Hilfe von Backtrack 5 R2 ein WLAN-Netzwerk mit WPA/WPA2-Verschlüsslung knackt, um das Passwort herauszufinden.

BACKTRACK 5 R2

– Einführung in Man-in-the-middle-Attacken

LOTHAR SERRA MARI

„Backtrack“ (*http://www.backtrack-linux.org*) ist eine unter Security-Spezialisten beliebte Linux-Distribution zur Überprüfung der Sicherheit einzelner Rechner oder ganzer Netzwerke. Die aktuelle Version von Backtrack trägt die Versionsnummer 5 R2 und wurde am 1. März 2012 veröffentlicht. Backtrack basiert auf der Linux-Distribution Ubuntu 10.04 und dem Linux-Kernel 3.2.6. Der eingesetzte Kernel wurde von den Entwicklern hinsichtlich der WLAN-Unterstützung optimiert.

Die Distribution beinhaltet viele Programme, mit denen unterschiedlichste Aspekte der Netzwerk- und Computersicherheit überprüft werden können. Darunter befinden sich auch Tools, mit denen sog. „Man-in-the-middle“-Angriffe (kurz: MITM) durchgeführt werden können. Unter einer MITM-Attacke versteht man einen Angriff, bei dem sich der Angreifer in den Datenverkehr eines Netzwerkes einklinkt und somit die komplette Kontrolle über den gesamten Netzwerkverkehr zwischen zwei oder mehreren Rechnern besitzt. Damit können beispielsweise Benutzereingaben abgefangen oder manipuliert werden.

Der folgende Artikel demonstriert verschiedene, einfache MITM-Angriffe mithilfe von Netzwerk-Sniffern, ARP- und DNS-Spoofing. Alle Tools sind dabei in Backtrack 5 R2 enthalten. Dabei sollen die Schwächen der Protokolle ARP, DNS und SSL demonstriert und einfache Schutzmöglichkeiten gezeigt werden.

Achtung: Gemäß §202c StGB ist die Durchführung der gezeigten Angriffe in fremden Netzwerken in Deutschland strafbar. Sofern die Absicht der illegalen Nutzung besteht, ist bereits der Besitz oder Vertrieb der genannten Tools strafbar. Alle Angriffe wurden in einem eigens dafür eingerichteten Netzwerk durchgeführt. Alle Angriffe werden lediglich zu Unterrichtszwecken dokumentiert. Der Autor übernimmt keine Haftung, sollten mithilfe dieses Artikels strafrechtlich relevante Handlungen durch Dritte durchgeführt werden.

Grundlagen des ARP-Spoofing

Das „Address Resolution Protocol“, kurz „ARP“, ist ein Netzwerkprotokoll zur Ermittlung von MAC-Adressen zu gegeben IP-Adressen. Damit ein Rechner in einem Netzwerk den zu der IP gehörenden Zielrechner „findet“, muss die IP des Zielrechners mit der Hardware-Adresse (MAC-Adresse) des Netzwerkadapters verknüpft sein. Für diese Verknüpfung ist das ARP zuständig. Die Zuordnung der einzelnen IP-Adressen zu den zugehörigen MAC-Adressen wird in der sogenannten „ARP-Tabelle“ gespeichert. Jeder Netzwerkadapter besitzt eine theoretisch weltweit einmalige MAC-Adresse, was eine einwandfreie Identifikation jeder Netzwerkschnittstelle in einem Netzwerk ermöglicht. Während sich IP-Adressen verändern können (z.B. bei der Nutzung dynamischer IP-Adressen), bleibt die MAC-Adresse eines Netzwerkadapters konstant.

Mithilfe einer ARP-Spoofing-Attacke können die ARP-Tabellen in einem Netzwerk so manipuliert werden, dass Datenpakete, die für einen bestimmten Rechner gedacht sind, zu einem Drittrechner weitergeleitet werden. Dort können die Pakete dann abgehört oder manipuliert werden, bevor sie anschließend von dem angreifenden Rechner zu dem eigentlichen Zielrechner weitergeleitet werden. Der angreifende Rechner hat sich als „Man-in-the-middle“ zwischen Quell- und Zielrechner geschaltet. Wird eine ARP-Spoofing-Attacke auf den Gateway eines Netzwerks durchgeführt, kann der Angreifer den Datenverkehr des gesamten Netzwerks mitschneiden und analysieren.

Benutzereingaben in Webanwendungen mit DNS-Spoofing und set mitschneiden

Mit „set“ beinhaltet Backtrack ein umfangreiches Toolkit zum Durchführen von Social-Engineering-Attacken. set beinhaltet eine einfache Möglichkeit, eine genaue Kopie einer Website zu

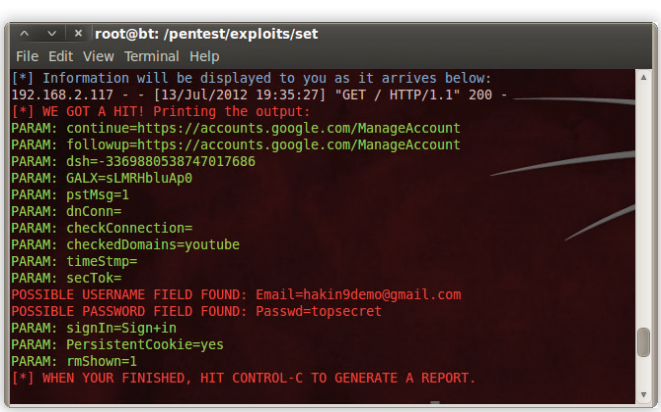


Abbildung 1: *Mitgeschnittene E-Mail-Adresse und Passwort*

erstellen und diese auf einem eigenen Webserver im Netzwerk verfügbar zu machen. Natürlich ist der einzige Zweck dieser Website, die Benutzereingaben beim Login-Vorgang mitzuschneiden. Darüber hinaus ist diese Methode äußerst einfach zu erkennen, da die Website selbst nicht mehr funktionsfähig ist. Damit lässt sich jedoch leicht eine DNS-Spoofing-Attacke demonstrieren, weshalb dieser Angriff hier dargestellt wird.

„set" befindet sich im Ordner /pentest/exploits/set/ und wird mit dem Aufruf ./set in einem Linux-Terminalfenster gestartet. set benötigt das Metasploit-Framework, welches sich im Paket framework3 befindet und einfach über die Paketverwaltung nachinstalliert werden kann. Nach dem Start lädt set ein einfaches Hauptmenü, über welches die einzelnen Funktionen des Toolkits aufgerufen werden können. Der Website-Cloner verbirgt sich hinter den Menüpunkten „1) Social Engineering Attacks", „2) Website Attack Vectors", „3) Credential Harvester Attack Method" und „2) Site Cloner".

Jetzt werden wir von set dazu aufgefordert, die URL der zu kopierenden Website anzugeben. Dabei müssen wir genau die URL eingeben, auf der sich auch der Login-Bereich der Website befindet. Bei Google-Konten wäre die dazu passende URL https://accounts.google.com. Nach erfolgter Eingabe startet set einen Webserver, über welchen die Kopie ausgeliefert wird. Sobald das Opfer nun die IP-Adresse des angreifenden Rechners im Browser aufruft, wird die täuschend echte Kopie der Website angezeigt. Alle Benutzereingaben werden von set mitgeschnitten und gespeichert.

Diese Methode hat jedoch zwei entscheidende Nachteile. Zum einen ist die Website nicht mehr funktionsfähig, da eben nur eine Kopie erzeugt wird, die eigentliche Webanwendung jedoch natürlich nicht kopiert werden kann. Zum anderen wird wohl kein Benutzer „versehentlich" im Browser gerade die IP-Adresse aufrufen, die zu dem angreifenden Rechner gehört, und anschließend auch noch seine Benutzerdaten eingeben.

Während set für das erste Problem keine Lösung bereithält, kann das zweite Problem mit einem zweiten Tool umgangen werden. Dazu bedienen wir uns einer DNS-Spoofing-Attacke. Beim DNS-Spoofing werden gefälschte DNS-Einträge im Netzwerk verbreitet. Da das DNS die Zuordnung von Domainnamen zu den passenden Server-IPs kontrolliert, können Domains auf andere IP-Adressen umgeleitet werden. Mit einem Tool namens „dnsspoof" können wir solche DNS-Spoofing-Attacken durchführen und beispielsweise die Domain der Website, deren Userdaten wir abgreifen wollen, auf die IP des angreifenden Rechners umleiten, auf dem dann eine Kopie mit set ausgeliefert wird. Damit der Netzwerkverkehr des zu attackierenden Rechners über den angreifenden Rechner umgeleitet wird, müssen wir zusätzlich eine ARP-Spoofing-Attacke durchführen, um anschließend die DNS-Queries manipulieren zu können.

ARP-Spoofing-Attacken sind unter Backtrack mit dem Tool „arpspoof" möglich.

Zuerst müssen wir den Linux-Kernel anweisen, allen eingehenden IPv4-Traffic zu seinem eigentlichen Bestimmungsort weiterzuleiten. Dieser Parameter ist jedoch nicht persistent, muss also nach jedem Reboot des Systems neu gesetzt werden.

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Um die ARP-Spoofing-Attacke zu starten, geben wir folgenden Befehl in einem Linux-Terminalfenster ein:

```
arpspoof -i eth0 -t 192.168.2.101 192.168.2.1
```

Dabei ersetzen wir „eth0" durch die Netzwerkschnittstelle des angreifenden Rechners. „eth0" bezeichnet dabei den Ethernet-Anschluss, während die WLAN-Schnittstelle als „wlan0" bezeichnet wird. Die IP-Adresse hinter „-t" bezeichnet die IP-Adresse des zu spoofenden Rechners, die zweite IP bezeichnet den Netzwerk-Gateway. Wenn der Parameter -t weggelassen wird, werden alle Rechner im Netzwerk manipuliert. Nach dem Start beginnt arpspoof mit seiner Arbeit und verbreitet die gefälschten ARP-Tabellen. Sobald das Terminalfenster, in welchem arpspoof läuft, geschlossen wird, wird die ursprüngliche ARP-Tabelle wiederhergestellt und vom eigentlichen Gateway propagiert.

Ob ein Rechner die falsche Tabelle erhalten hat, ist beispielsweise über die Abfrage der Traceroute zum Gateway überprüfbar. Während bei intakter ARP-Tabelle der Gateway üblicherweise im ersten Hop erscheint, befindet sich bei gefälschter ARP-Tabelle vor dem Hop zum Gateway noch ein „Umweg" zum angreifenden Rechner.

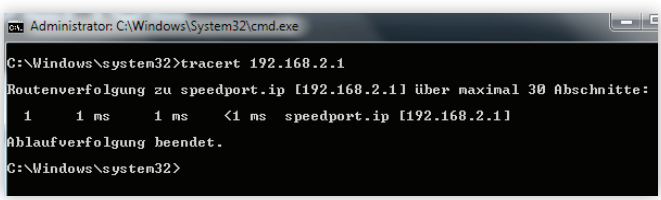


Abbildung 2: *Traceroute zum Gateway bei intakter ARP-Tabelle*

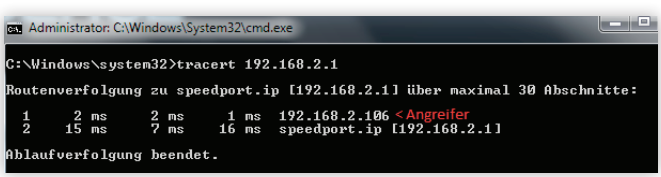


Abbildung 3: *Traceroute zum Gateway bei gefälschter ARP-Tabelle*

Nun können wir die DNS-Spoofing-Attacke vorbereiten. Dazu erstellen wir eine neue Textdatei mit dem Namen fakeresolv.conf, die nach dem folgenden Muster gefüllt wird:

192.168.2.106	facebook.com
192.168.2.106	*.facebook.com
192.168.2.106	google.com
192.168.2.106	accounts.google.com
192.168.2.106	*.ebay.com

Der Inhalt der Datei ist selbsterklärend: In der linken Spalte steht die IP-Adresse des Angreifer-Rechners, in der rechten Spalte die zu spoofenden Domains. Die Domainnamen können Wildcards enthalten, um beispielsweise alle Domainendungen einer bestimmten Domain oder alle Subdomains einzuschließen. Der Name der Datei kann frei gewählt werden.

Die DNS-Spoofing-Attacke wird gestartet, indem wir in einem neuen Linux-Terminal den folgenden Befehl eingeben:

```
dnsspoof -i eth0 -f fakeresolv.conf host 192.168.2.1
```

Auch hier müssen wieder Netzwerkschnittstelle und Host angepasst werden. Auch wenn der Gateway des Netzwerks gespooft werden soll, werden die DNS-Einträge nur auf den Rechnern verfälscht, zu denen eine Verbindung mit einer ARP-Spoof-Attacke besteht. Immer, wenn ein gespoofter Rechner eine gespoofte Domain aufruft, zeigt uns dnsspoof dies im Terminalfenster an.

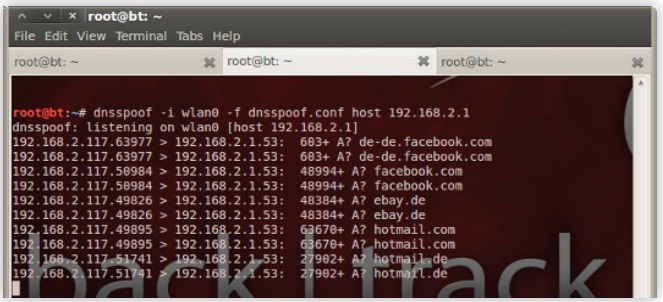


Abbildung 4: *dnsspoof bei der Arbeit*

Jetzt können wir in einem neuen Terminal-Fenster wie weiter vorne schon beschrieben mit set eine Kopie einer Website erstellen. Jetzt muss das Opfer nicht mehr „zufällig“ die IP-Adresse des Angreifer-PCs aufrufen, sondern erreicht die Kopie über die gewohnte Domain, was die Chancen des Angreifers, Benutzerdaten abzufangen, erheblich vergrößert.

Aber auch eine DNS-Spoofing-Attacke kann die größte Schwachstelle dieses Angriffsvektors nicht ausmerzen: Nach wie vor bekommt das Opfer nur eine funktionsunfähige Kopie angezeigt. Während weniger versierte Benutzer wahrscheinlich an einen Fehler auf der Website glauben werden und sich keine weiteren Gedanken machen, dürften fortgeschrittene Benutzer skeptisch werden und vermutlich bei dem Betreiber der Website nachfragen, ob aktuell eine Störung vorliegt. Abgesehen davon ist gerade auf Websites, die zum Login das sichere Protokoll HTTPS verwenden, der beschriebene Angriff nicht möglich, da der Webserver des Angreifers keine HTTPS-Inhalte ausliefert.

Zusammengefasst kann gesagt werden, dass der beschriebene Angriff mit set und DNS-Spoofing keine besonderen Erfolgschancen verspricht. Glücklicherweise haben wir dank der ARP-Spoofing-Attacke eine Möglichkeit geschaffen, weitaus eleganter nach Benutzerdaten zu suchen.

Benutzereingaben in Webanwendungen mit ettercap mitschneiden

Genau genommen stellt die Kombination aus ARP-Spoofing, DNS-Spoofing und set einen ziemlich umständlichen und wenig erfolgreichen Weg dar. In diesem Artikel kommt sie nur zum Einsatz, da sich die Grundlagen von ARP- und DNS-Spoofing vergleichsweise einfach erklären lassen. In der Realität ist es relativ unwahrscheinlich, dass ein solcher Angriff zum Einsatz kommt.

Anstatt erst eine Verbindung mithilfe einer MITM-Attacke herzustellen und DNS-Einträge zu fälschen, um eine Kopie einer Website an das Opfer auszuliefern, ist es wesentlich effizienter, den Traffic zwischen Internet und Opfer-PC direkt abzufangen und zu analysieren. Dank ARP-Spoofing ist dies auch in geswitchten Netzwerken möglich. Tools, die dazu dienen, den Netzwerkverkehr zu belauschen, werden als „Sniffer“ bezeichnet.

ettercap ist ein sehr umfangreicher Sniffer. Neben der Möglichkeit, den Traffic „nur“ mitzuschneiden und später mit externen Tools zu analysieren, besitzt ettercap Filter, mit denen nur Passwörter angezeigt werden. Der restliche Traffic wird für den Benutzer unsichtbar weitergeleitet. Neben HTTP(S) unterstützt Ettercap POP3, SMTP, IMAP und weitere Protokolle. Selbst SSL-Verbindungen sind vor Angriffen durch ettercap nicht sicher, wie weiter unten in diesem Artikel gezeigt wird. Ein weiterer entscheidender Vorteil ist, dass der Angriff für das Opfer vollkommen unsichtbar abläuft. Auch muss man sich nicht auf bestimmte Websites beschränken, wie es bei set der Fall ist. Ettercap übermittelt alle Passwörter und Benutzernamen, die das Opfer auf Websites eingibt, solange die MITM-Attacke aufrecht erhalten bleibt.

Vor der Benutzung bedarf ettercap einiger Konfiguration.

Dazu bearbeiten wir die Datei /usr/local/etc/etter.conf mit einem Editor unserer Wahl und bearbeiten sie folgendermaßen:

```
[…]
  [privs]
  ec_uid = 0          # nobody is the default
  ec_gid = 0          # nobody is the default
Diese Zeilen bewirken, dass ettercap mit root-Rechten
  ausgeführt wird.
Anschließend bearbeiten wir folgende Zeilen, die in etwa bei
Zeile 165 beginnen:
#---------------
#     Linux
#---------------
# if you use ipchains:
  #redir_command_on = „ipchains -A input -i %iface        […]
  #redir_command_off = „ipchains -D input -i %iface       […]
  # if you use iptables:
  #redir_command_on = „iptables -t nat -A PREROUTING -i  […]
  #redir_command_off = „iptables -t nat -D PREROUTING -i […]
```

Hier entfernen wir die Rautezeichen („#“) vor den beiden Zeilen unter if you use iptables. Alle anderen Zeilen müssen auskommentiert bleiben, sonst arbeitet ettercap nicht richtig. Abschließend wird die Datei abgespeichert.

Nun starten wir wie weiter vorne bereits beschrieben eine ARP-Spoofing-Attacke gegen einen bestimmten Rechner oder ein Gateway. Ettercap besitzt übrigens selbst die Möglichkeit, ARP-Spoofing-Attacken ausführen. Ich rate jedoch dazu, das ARP-Spoofing mit arpspoof durchzuführen, damit die falschen ARP-Tabellen auch dann aufrecht erhalten werden, wenn man ettercap beispielsweise beendet um den anfallenden Traffic mit anderen Tools zu analysieren.

Nun sollten wir noch den Passwortfilter von ettercap ein wenig nachrüsten. Dieser Filter scannt die anfallenden GET/POST-Requests nach Schlüsselbegriffen wie „username“ oder „password“. Auch wenn schon recht viele Schlüsselbegriffe in dem Filter enthalten sind, fehlen beispielsweise noch passende Filter für die populären Webanwendungen Wordpress, vBulletin und Woltlab Burning Board.

Der Passwortfilter befindet sich in der Datei /usr/local/share/ettercap/etter.fields.

In diese Datei fügen wir folgende Usernamen und Passwörter in die entsprechenden Abschnitte ein:

```
[USER]
vb_login_username
loginUsername
log

[PASS]
vb_login_password
loginPassword
pwd
```

Um die Parameter für weitere Websiten hinzuzufügen, um die Funktionalität des Passwortfilters zu verbessern, empfiehlt sich der Einsatz eines Tools zum Manipulieren von HTTP-Requests. Ein gutes Beispiel für ein solches Tool ist das Firefox-Plugin TamperData, welches auch für diesen Artikel verwendet wurde.Nun können wir ettercap mit den benötigten Parametern starten:

```
ettercap -Tq -i eth0
```

Der Parameter -T bedeutet, dass ettercap im Textmodus starten soll, ohne die grafische Oberfläche zu laden. Mit dem Parameter -q werden alle Ausgaben unterdrückt, die keine Passworteingaben enthalten oder sonst irgendwie interessant sein könnten. Wird dieser Parameter weggelassen, wird der komplette Netzwerkverkehr im Terminal-Fenster angezeigt. -i definiert den Netzwerkadapter, der zum Abhören der Verbindung eingesetzt wird.

Nach dem Start nimmt ettercap automatisch die Arbeit auf und wartet auf übertragene Benutzernamen und Passwörter. Gesniffte Benutzereingaben werden im Terminal-Fenster angezeigt.

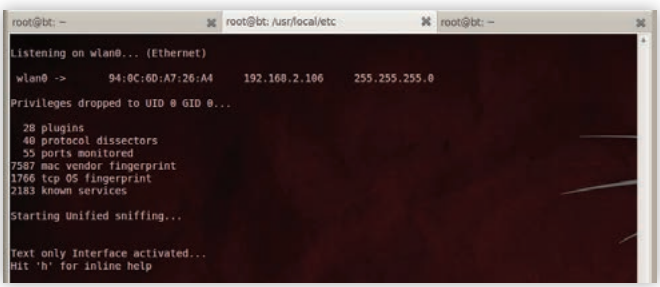


Abbildung 5: *Ettercap schneidet Benutzernamen und Passwörter mit*

Im Beispiel wurde immer der Benutzername „hakin9demo“ und das Passwort „topsecret“ verwendet. Diese Attacke ist für das Opfer vollständig unsichtbar. Abgesehen von den gefälschten ARP-Requests, die nur in den wenigsten Fällen auffallen dürften, funktionieren alle Websites wie gewohnt. Man kann sich also relativ sicher sein, dass die gesnifften Passwörter auch tatsächlich stimmen.

Dieser Angriff funktioniert jedoch nur dann, wenn die Website nicht über eine verschlüsselte Verbindung wie HTTPS ausgeliefert wird. Sobald SSL verwendet wird ist ettercap alleine nicht mehr in der Lage, die Verbindung abzuhören. Aus Sicherheitsgründen wird der Login-Vorgang auf Websites häufig verschlüsselt.

Doch dank eines zusätzlichen Tools ist es auch möglich, verschlüsselte Verbindungen so zu manipulieren, dass Benutzerdaten mitgeloggt werden können, selbst wenn sogar der Login-Vorgang selbst verschlüsselt abläuft.

Verschlüsselte Benutzereingaben mit ettercap und sslstrip mitschneiden

Im Jahre 2009 stellte der Hacker Moxie Marlinspike auf der Hacker-Konferenz Black Hat DC 2009 ein Tool namens sslstrip vor, welches in der Lage ist, Verbindungen so zu manipulieren, dass keine HTTPS-, sondern in gewöhnliche HTTP-Verbindungen aufgebaut werden. sslstrip ist nicht in der Lage, den Verschlüsselungsalgorithmus von SSL zu brechen, vielmehr wird dafür gesorgt, dass keine HTTPS-Verbindung zustande kommt. Zusätzlich besitzt sslstrip die Möglichkeit, eine eventuell laufende Session einer Website zu beenden, um so eine Neuanmeldung zu erzwingen.

Zuerst muss wie schon beschrieben eine ARP-Spoofing-Attacke gestartet werden und der Traffic vom Kernel weitergeleitet werden. Dann benötigen wir noch eine iptables-Regel, um den eingehenden Traffic von Port 80 zum Port 10000 weiterzuleiten. Port 10000 wird dann von sslstrip abgehört und auf die gewünschten Informationen hin untersucht.

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j
                        REDIRECT -to-port 10000
```

sslstrip befindet sich im Ordner /pentest/web/sslstrip und wird folgendermaßen gestartet: `python sslstrip.py -k -l 10000`.

Der Parameter -k sorgt dafür, dass laufende Sessions beendet werden, um den Benutzer dazu zu zwingen, seine Anmeldedaten erneut einzugeben. -l definiert den Port, auf dem sslstrip lauscht.

Der so bearbeitete Traffic kann wie gewohnt mit ettercap auf Usernamen und Passwörter untersucht werden. Dieses mal stellt die Verschlüsselung kein Hindernis für ettercap dar, da diese von sslstrip wirkungsvoll umgangen wird.

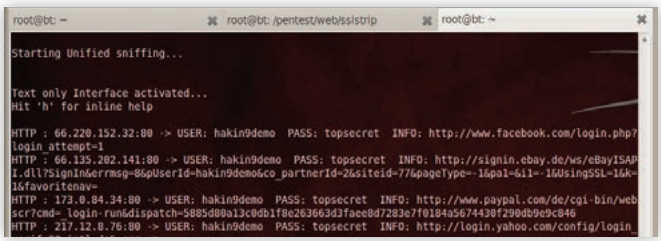


Abbildung 6: *sslstrip enthüllt SSL-Verbindungen*

Ein Angriff mit sslstrip, ettercap und arpspoof stellt gewissermaßen den „perfekten“ Angriff dar. Da keine Zertifikatfehler auftreten und nur wenige Benutzer darauf achten werden, ob im Browser tatsächlich eine SSL-Verbindung besteht, ist dieser Angriff als sehr gefährlich einzustufen.

Doch es gibt eine Möglichkeit, sich auch davor zu schützen.

Schutzmechanismus gegen sslstrip

Während eine MITM-Attacke mit set relativ einfach zu erkennen ist, ist dies bei der Verwendung von ettercap und sslstrip sehr viel schwerer. Website-Betreiber können jedoch Vorkehrungen treffen, die solche Angriffe unmöglich machen, indem sie HTTPS-Verbindungen erzwingen und keine „gewöhnlichen“ HTTP-Verbindungen zulassen.

Ruft das Opfer während einer Attacke mit sslstrip die Websites von Twitter oder Google Mail auf, wird eine Warnmeldung im Browser erzeugt.

Im Gegensatz zu der Fehlermeldung, die dann auftritt, wenn eine Website ein selbst signiertes Zertifikat besitzt, kann diese Fehlermeldung nicht übersprungen werden; es besteht keine Möglichkeit, die Website zu laden.

Diese Fehlermeldung wird deshalb erzeugt, da unter anderem Twitter und Google Mail eine Technik namens „HTTP Strict Transport Security“ (HSTS). Diese Technik sorgt dafür, dass alle unsicheren HTTP-Verbindungen einer Website auf eine sichere HTTPS-Verbindung umgeleitet werden. Da sslstrip jedoch darauf angewiesen ist, dass parallel zur HTTPS-Verbindung eine unsichere Verbindung möglich ist, ist es für sslstrip nicht möglich, die Verschlüsselung auszuhebeln. Jeder Webmaster, der HTTS-Verbindungen einsetzt, sollte auch HSTS nutzen.

LOTHAR SERRA MARI

Lothar Serra Mari (aka. „horrork1d“), Jahrgang 1993, ist Blogger, Computer- und Linux-Experte. Daneben beschäftigt er sich mit der Sicherheit von Webanwendungen und Computersystemen. Erbetreibt den Blog http://www.leetperium.de, auf dem regelmäßig wissenswerte Artikel rund um Linux, Server-Administration und anderen IT-bezogenen Themen veröffentlicht werden.
Kontakt: lserramari@gmail.com

BACKTRACK 5

MARTIN SCHAGERL

Da Sicherheit in IT Netzwerken und Applikationen immer mehr an Bedeutung gewinnt, versuchen viele Unternehmen ihre Infrastruktur sowie ihre eingesetzte Software auf Sicherheitslücken zu durchsuchen. Die verantwortlichen Personen wissen dabei oft nicht, welche Tools für diesen Zweck zu Verfügung stehen und sind unschlüssig, welches Betriebssystem verwendet werden soll. Die Antwort ist einfach: Backtrack 5 – eine Linux Distribution mit einer großen Anzahl von vorinstallierten Securitytools. In dem folgenden Artikel erfahren Sie mehr über die wichtigsten Applikationen von Backtrack 5 sowie eine Anleitung um eine fremde Maschine anzugreifen.

Bei Backtrack handelt es sich um eine Linux Distribution mit zusätzlichen Werkzeugen für Sicherheitsüberprüfungen. Seit 1. März 2012 steht Backtrack 5R2 kostenlos zum Download bereit. Diese Version basiert auf Ubuntu mit dem Kernel 3.2.6, wodurch die Ubuntu Paketverwaltung genutzt werden kann. Beim Download von Backtrack kann man zwischen GNOME und KDE als grafischen Oberflächenmanager entscheiden. Einsteiger sollten sich eher für GNOME entscheiden, da diese einfacher zu bedienen ist. KDE bietet mehrere Optionen, um das System zu konfigurieren und ist daher für fortgeschrittene Benutzer zu empfehlen.

Backtrack wird ursprünglich als Live-CD eingesetzt. Dadurch ergibt sich der große Nachteil, dass alle Änderung nach einem Neustart verloren geht. Daher kann das System auch auf einer Festplatte installiert werden. Dazu muss Backtrack von der CD gestartet werden und danach die vorgefertigte Anwendung „Install Backtrack“ auf dem Desktop ausgeführt werden (siehe Abbildung 1).

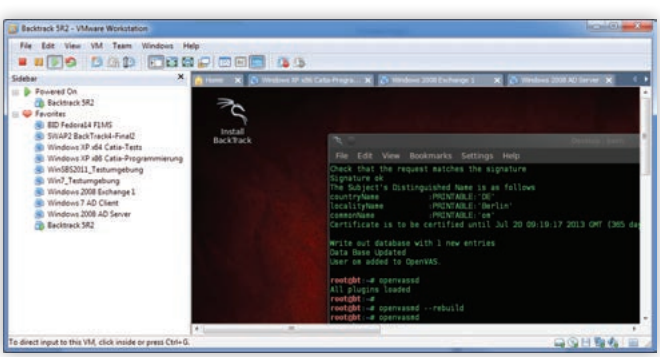


Abbildung 1: *Desktop Icon zum Installieren von Backtrack*

Da Backtrack für alle möglichen Arten von Sicherheitstests (Testen von Webapplikation, Password Cracking, Sniffen im WLAN, …) eingesetzt werden kann, werden die bereitgestellten Anwendungen in Kategorien eingeteilt, um die Übersichtlichkeit zu gewährleisten. Dadurch findet man schnell jene Applikationen, die für das gewünschte Einsatzgebiet am besten passen. In der Abbildung 2 werden die Hauptkategorien gezeigt.

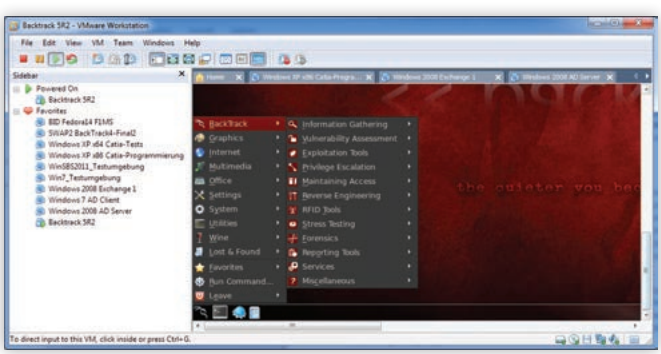


Abbildung 2: *Kategorisierung der installierten Anwendungen*

Da es den Rahmen sprengen würde, alle Programme anzuführen, werden wir uns auf die am häufigsten eingesetzten Tools konzentrieren.

Sammeln von Informationen

Häufig sind Daten bzw. Bereiche eines Unternehmens ungewollt öffentlich zugänglich. Um nun einen Angriff vorzubereiten, müssen bestimmte Informationen wie offene Ports, Dienste, usw. ermittelt werden. Dazu stehen unter Backtrack die Tools NMAP und OpenVAS zu Verfügung.

Bei NMAP handelt es sich um ein verbreitetes Programm für Portscans, welches alle gängigen Scan-Techniken beherrscht. Dabei beinhaltet NMAP nützliche Funktionen, um Dienste und Betriebssysteme sowie deren Versionsstände aufzudecken. Das Ziel eines Portscan ist es festzustellen, ob Ports auf bestimmten

Zielsystemen geöffnet sind und ein Dienst aktiv ist. Diese Informationen können für einen Angreifer hilfreich sein, da hiermit ältere und somit verwundbare Systeme aufgedeckt werden können. Bei einem Portscan werden entsprechend präparierte TCP- und UDP Datenpakete an die Zielsysteme gesendet und deren Antwort ausgewertet. Um nun offene TCP Ports zu finden, kann eine der folgenden Techniken angewendet werden:

- **SYN/"half open" Scan:** Bei dieser Scan-Technik wird keine vollständige TCP Session aufgebaut. Sollte vom Zielsystem durch Antworten eines TCP SYN|ACK-Flag ein offener Port signalisieren werden, so wird ein TCP RST-Flag gesendet, um den Verbindungsaufbau abzubrechen. Wird ein TCP RST|ACK-Flag empfangen, so werden keine weiteren Pakete gesendet und der Port wird als geschlossen behandelt.
  Da hier kein üblicher Verbindungsaufbau stattfindet, werden Administrator bzw. Root Rechte für das Scannertool benötigt: `nmap -sS <IP oder Domain>`
- **Connect Scan:** Hier wird versucht, eine vollständige TCP Session aufzubauen. Diese Überprüfung ist im Vergleich mit dem SYN/"half open" Scan zeitaufwändiger, da mehrere Pakete übertragen werden. Da hier ein üblicher Verbindungsaufbau stattfindet, können Standard APIs bzw. System Calls verwendet werden und es sind somit keine Administrator bzw. Root Rechte für das Scannertool notwendig:
  `nmap -sT <IP oder Domain>`

Nachdem verbreitete Dienste wie zum Beispiel DNS das Transportprotokoll UDP verwenden, müssen auch diese Ports aufgedeckt werden. Da bei UDP kein Verbindungsaufbau durchgeführt wird, muss eine andere Scan-Technik angewendet werden: Es wird ein leeres UDP Pakete an das Zielsystem gesendet. Antwortet dieses System mit einem „ICMP port unreachable" Nachricht, so ist der Port geschlossen. Wenn keine Antwort oder ein UDP Paket als Antwort empfangen wird, so ist dies ein Anzeichen für einen offenen Port. Der Nachteil dieser Technik ist, dass beim Blocken eines Ports durch ein Firewall falscherweise dieser als offener gekennzeichnet wird. Um diesem Problem entgegenzuwirken, können zusätzlich Anwendungsspezifische UDP Pakete gesendet und ausgewertet werden. So kann beispielsweise eine DNS Anfrage auf Port 53 geschickt und auf eine korrekte Antwort gewartet werden. Bei einen NMAP-UDP Scan wird dieses Konzept angewendet.

Ein UDP Scan kann mittels NMAP mit dem Parameter -sU" durchgeführt werden: `nmap -sU <IP oder Domain>`

Wie zuvor erwähnt, kann zusätzlich zu den Portstatus auch der eingesetzte Dienst und das Betriebssysteme sowie deren Versionsstände ermittelt werden. Um mit NMAP eine Betriebssystemerkennung durchzuführen, muss der Parameter -O angegeben werden. Durch diesen Parameter versendet NMAP präparierte Datenpakete an offene und geschlossene Ports. Die Antwort wird mit einer Datenbank abgeglichen, in der unterschiedlichste Betriebssystem-Signaturen abgespeichert sind. Dadurch kann eine Aussage über das entfernte Betriebssystem getroffen werden: `nmap -O <IP oder Domain>`

Bei der Versionserkennung kann spezifiziert werden, wie aggressiv die Überprüfung durchgeführt werden soll. Es kann ein Level zwischen 0 und 9 konfiguriert werden, wobei 7 der Standardwert ist. Je höher das Level, umso mehr Informationen werden ermittelt. Die Diensterkennung wird mit dem Parameter „-sV" angegeben, die Idensität mit „--version-intensity <intensity>". Durch setzten dieser Parameter baut NMAP eine vollständige Verbindung zu offene Ports auf. In Abhängigkeit von der Konfiguration der Dienste senden diese nach einer aufgebauten Verbindung einen sogenannten Banner, welcher Informationen über den laufenden Dienst und dessen Versionsstand enthält. Anhand der Daten des Banners können auch Rückschlüsse auf das Betriebssystem gezogen werden, da der Dienst eventuell nur für ein bestimmtes Betriebssystem verfügbar ist: `nmap -sV --version-intensity <intensity> <IP oder Domain>`

Wie man in Abbildung 3 sieht, können die Parameter beliebig kombiniert werden. Alle weiteren Parameter können in der Manpage von NMAP nachgelesen werden.

Abbildung 3: *Auszug eines NMAP Scan*

Der Vulnerability Scanner OpenVAS ist eines der beliebtesten Tools für Security-Audits. Die Entwicklung von OpenVAS begann 2005, als der Vorgänger Nessus nur noch gegen Lizenzgebühren eingesetzt werden konnte. So existiert weiterhin ein Open Source Scanner zum Sammeln von Sicherheitskritischen Informationen. In der aktuellen Backtrack 5R2 Version ist der OpenVAS 5.0 bereits vorinstalliert.

Um die Nachteile von einem clientbasierten Scanner entgegenzuwirken, wurde OpenVAS auf einem Client-Server Prinzip entwickelt. Das bedeutet, es muss auf einen zentralen Computer (Server) der OpenVAS Dienst laufen. Danach kann OpenVAS von berechtigten Benutzern (Clients) über das Netzwerk mittels einer Webapplikation, Clientapplikation oder Konsole bedient werden. Der Serverseitige Dienst besteht aus mehreren Komponenten, welche alle über gut etablierte Protokolle und einer durchgehend SSL-abgesicherten Verbindung kommunizieren. Mit dem Scanner kann eine Netzwerkinfrastruktur auf unterschiedlichste Sicherheitslücken wie z.B. unzureichende Verschlüsselung, unsichere Protokolle, usw. geprüft werden. Dabei werden während einer Überprüfung eine Reihe von einzelnen Tests durchgeführt, welche zuvor als Plugin geladen werden. Derzeit werden mehr als 26.000 Plugins zu Verfügung gestellt. Hat man bestimmte Anforderungen, welche durch die mitgelieferten Sicherheitstest nicht abgedeckt werden, wie z.B. Prüfen von eigenen Softwareprodukten, so können eigene Überprüfungen geschrieben werden und diese in der OpenVAS Umgebung ausgeführt werden. Am Ende

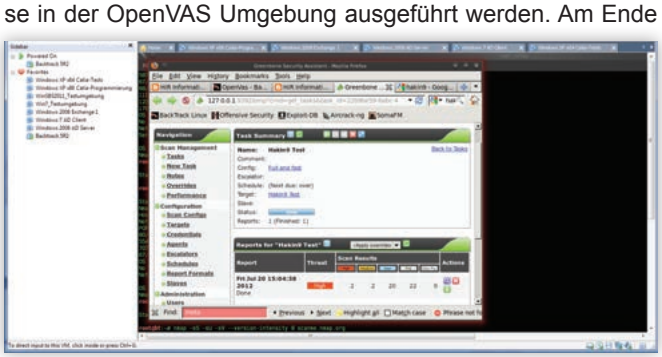


Abbildung 4: *Übersicht über die Schwachstellen einer Überprüfung*

einer Überprüfung werden die gefundenen Schwachstellen in vier Schweregrad kategorisiert (High, Medium, Low, Log, False Pos.) (siehe Abbildung 4). In der Detailansicht wird jede Schwachstelle detailliert beschrieben sowie teilweise ein Lösungsvorschlag und weitere nützliche Informationen gegeben (siehe Abbildung 5). Das Ergebnis kann als CPE, HTML, ITG, LaTeX, NBE, PDF, TXT oder XML Datei exportiert werden.

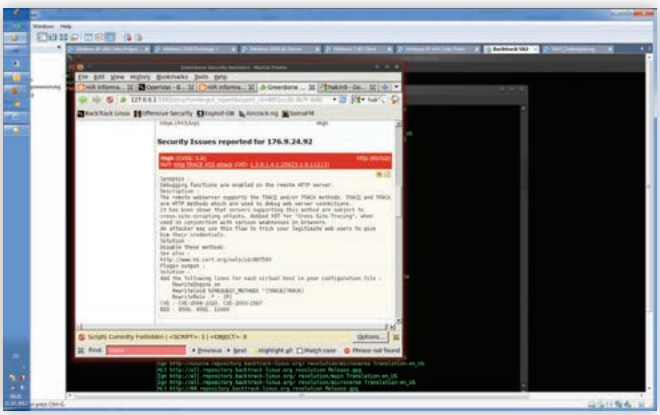


Abbildung 5: *Details von einer aufgedeckten http Schwachstelle*

Exploits durchführen

Hat man mit den zuvor beschriebenen Tools genügend Informationen gesammelt, um Schwachstellen zu erkennen, so können diese mithilfe von weiteren Backtrack-Applikationen ausgenützt werden. Es müssen jedoch immer die rechtlichen Rahmenbedienungen beachtet werden, da man hierbei die Schwachstelle aktiv ausnützt, um Schadcode einzuschleusen, welcher am Zielsystem unautorisiert ausgeführt wird.

Das bekannteste und umfangreichste Programm ist wohl das Metasploit Framework, welches in der Version 4.4.0 in der aktuellen Backtrack Version mitgeliefert wird. Das Programm befindet sich auf Platz 2 der 125 wichtigsten Sicherheits-Tools. Es handelt sich um ein freies Open-Source-Projekt, welches bereits über eine große Datenbank an vorgefertigten Exploits verfügt.

Hat man durch OpenVAS oder NMAP eine Schwachstelle identifiziert, so kann für diese aus der Metasploit-Datenbank ein geeigneter Exploit ausgewählt werden, um in das Zielsystem einzudringen. Zusätzlich kann ein Payload konfiguriert werden, welcher nach dem erfolgreichem Eindringen ausgeführt werden soll. Dies könnte zum Beispiel eine Remoteshell oder VNC Server sein, um alle weiteren Befehle bzw. spätere Einbrüche komfortabel ausführen zu können. Derzeit verfügt die Metasploit Datenbank über 909 Exploits sowie 250 Payloads.

Der Funktionsumfang von Metasploit ist sehr größer, eine komplette Beschreibung würde den Umfang dieses Beitrags sprengen.

Passwörter knacken

Unter Umständen gelangt man an durch Sniffen im Netzwerk oder anderer Methoden an verschlüsselten Passwörtern bzw. an eine Datei, welche verschlüsselte Passwörter enthält. Um diese verschlüsselten Passwörter entschlüsseln zu können, bietet Backtrack einige nützliche Tools.

Das Programm „John the Ripper" (JTR) ist bei Backtrack standardmäßig vorinstalliert und eines der meist genutzten Programme zum Entschlüsseln von Passwörtern. Die Entschlüsselung kann dabei über einen Brute-Force bzw. eine Dictionary Attacke erfolgen. Da beim Entschlüsseln von komplexen Passwörtern ein sehr hoher Rechenaufwand entstehen kann, unterstützt JTR das zusätzliche verwenden von Grafikkarten-Prozessoren (GPU). Dadurch können die Berechnungen um ein vielfaches beschleunigt werden.

Um beispielweise die Passwörter eines Unix Systems zu entschlüsseln, genügen zwei Befehle (siehe Abbildung 6). Mit dem ersten Befehl werden die beiden Dateien /etc/passwd und /etc/shadow kombiniert und an den angegeben Pfad abgespeichert. Diese Datei kann nun an den eigentlichen Befehl zum Entschlüsseln übergeben werden. In den Standardeinstellungen von JTR wird versucht, das Passwort anhand eines mitgelieferten Dictonary zu entschlüsseln. Sollte das nicht klappen, so wird eine Brute-Force Attacke durchgeführt, welche im Normalfall länger dauert.

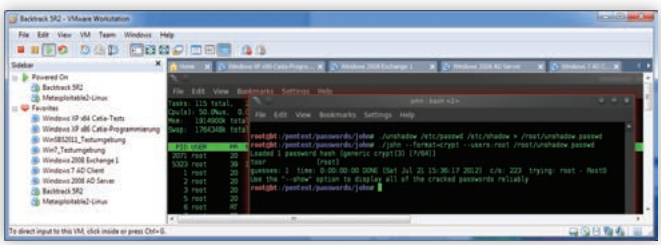


Abbildung 6: *Unix Passwort mit „John the Ripper" entschlüsseln*

Da nun alle benötigten Werkzeuge für einen erfolgreichen Angriff beschrieben wurden, werden wir nun einen Angriff Schritt für Schritt durchführen:

Schritt 1: Portscan sowie Dienst- und Betriebssystemerkennung mit NMAP

Durch den NMAP Scan in Abbildung 7 wurde erkannt, dass bekannte Port wie SSH offen sind und ein eine Linux Distribution eingesetzt wird. Diese Informationen können im Schritt 5 ausgenutzt werden.

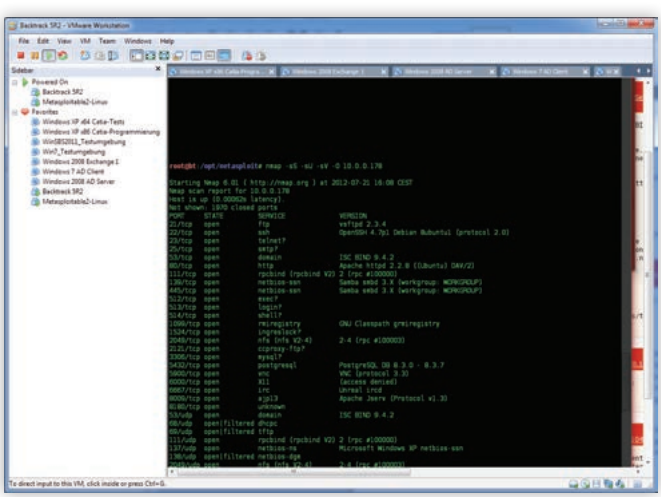


Abbildung 7: *Ausschnitt von einem NMAP Ergebnis*

Schritt 2: Vulnerability Scan mit OpenVAS

Durch den Vulnerability Scan wurde herausgefunden, dass am Zielsystem eine schwerwiegende Samba Schwachstelle existiert.

Schritt 3: Exploit mit Metasploit

Um die Schwachstelle aus Schritt 2 auszunützen, muss ein Exploit aus der Metasploit Datenbank geladen und konfiguriert werden. Da es sich um eine Samba Schwachstelle handelt,

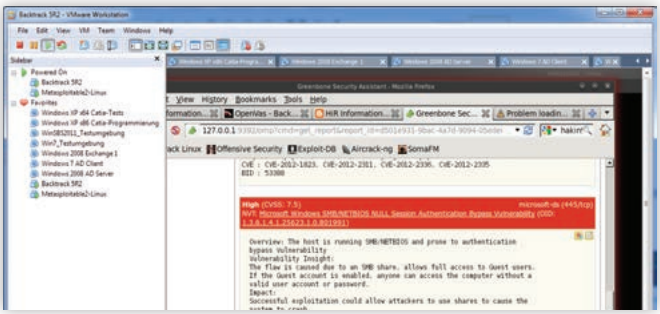


Abbildung 8: *Schwerwiegende Samba Schwachstelle*

wird der Exploit „exploit/multi/samba/usermap_script“ verwendet. Um nach dem Eindringen in das System auch Befehle absetzen zu können, wird ein Payload mitübertragen und ausgeführt. In unseren Fall wird der Payload „cmd/unix/bind_netcat“ verwendet, womit Daten von der Standardein- oder -ausgabe über das Netzwerk übertragen werden können. Ziel dieser Demonstration ist es, die Dateien /etc/passwd und /etc/shadow auszulesen und den Inhalt am lokalen System zu speichern. Dazu wird der Befehl „cat /etc/passwd“ und „cat /etc/shadow“ durchgeführt. Der ausgegebene Inhalt wird manuell markiert und in zwei separaten Dateien (attackPasswd.txt und attackShadow.txt) abgespeichert.

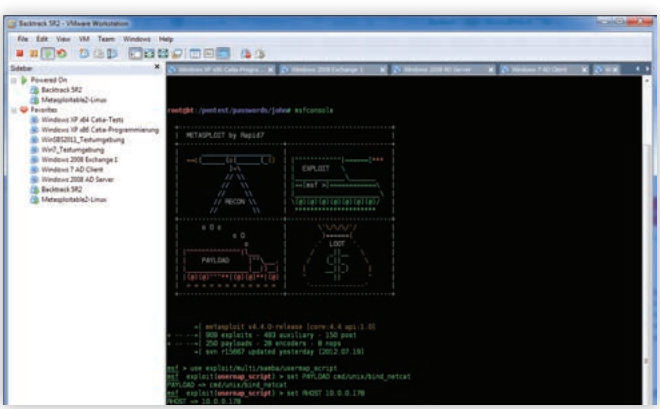


Abbildung 9: *Auszug einer Metasploit Attacke*

Schritt 4: Passwörter entschlüsseln

Da wir nun im Besitz der Benutzer und deren Passwörter sind, müssen wir diese nur noch entschlüsseln. Dazu werden die beiden Dateien mit den unshadow Kommando von JDR kombinieren und danach entschlüsselt.

Schritt 5: SSH Verbindung als Root aufbauen

Wie man aus Schritt 1 entnehmen kann, ist auf der Zielmaschine der SSH Dienst aktiv. Da wir nun mehrere Benutzer und deren Passwörter besitzen, versuchen wir einen SSH Verbindung zum Ziel aufzubauen. Danach können Befehle mit den Berechtigungen des entsprechenden Benutzers (in diesen Fall msfadmin) ausgeführt werden.

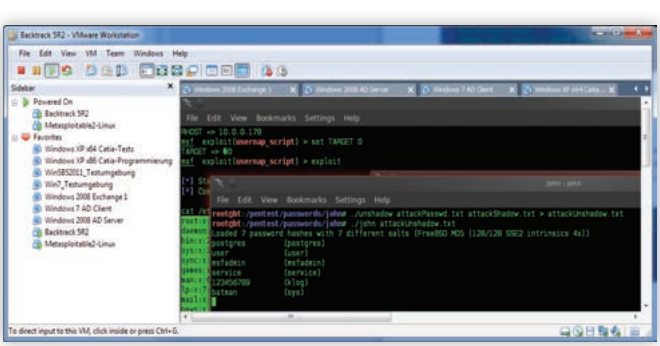


Abbildung 10: *Passwörter entschlüsseln*

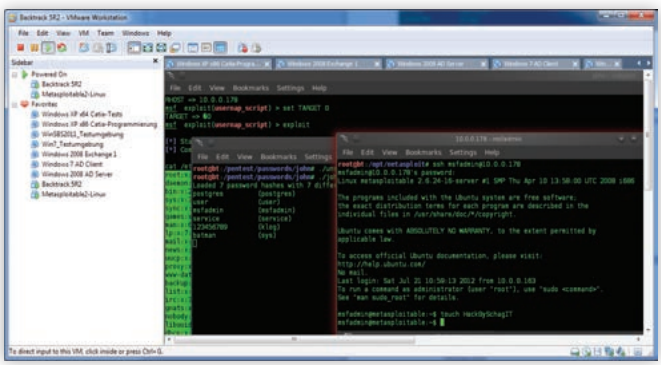


Abbildung 11: *SSH Verbindung zur angegriffenen Maschine aufbauen*

Fazit

Backtrack bietet eine Vielzahl an nützlichen Tools, um Sicherheitsüberprüfungen durchzuführen, Passwörter zu cracken, in fremde Systeme einzudringen usw. Trotz der vielen unterschiedlichen Anwendungen wird eine gute Übersicht gegeben und das System lässt sich komfortabel bedienen. Da viele Tools bereits vorinstalliert sind bzw. mit nur wenig Aufwand aus den Repositories nachinstalliert werden können, kann man schnell mit der tatsächlichen Arbeit loslegen, ohne sich um aufwendige Installationen zu kümmern.

Das Betriebssystem ist jedoch nicht für den Alltagseinsatz geeignet, schon alleine dass man als Root unterwegs ist, sollte dies jedem Linux-Anwender verdeutlichen.

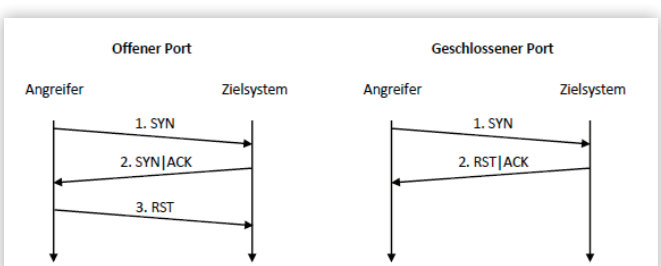


Abbildung 12: *Ablauf eines SYN Scan/”half open” Scan*

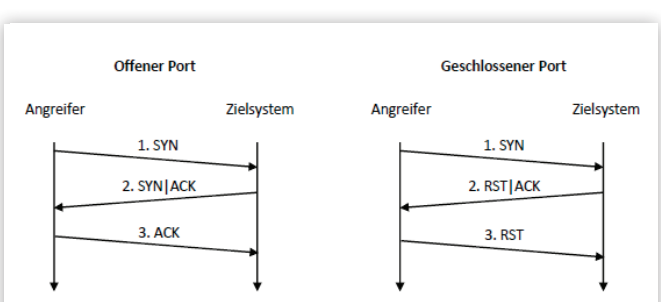


Abbildung 13: *Ablauf eines Connect Scan*

MARTIN SCHAGERL

geboren am 21.03.1988 absolviert derzeit das Masterstudium IT-Security an der Fachhochschule St. Pölten. Durch laufende Ausbildungen sowie praktische Erfahrungen besitzt er ein hervorragendes und breites Wissen im Bereich der Computersicherheit, Software Engineering sowie Webapplikationen. Er gründete 2010 ein eigenes Unternehmen, welches sich mit IT Lösungen beschäftigt und dabei einen Fokus auf IT Security setzt. Mehr Informationen über seine Firma können Sie unter http://www.schagit.at finden.

BACKTRACK5:
DATENSAMMLUNG UND REPORTERSTELLUNG FÜR PENTESTER MIT MAGICTREE

HANS HÖFKEN, MARKO SCHUBA

Bei der Durchführung von Schwachstellen- und Penetrationstests fallen häufig eine große Menge Daten an. Hier den Überblick zu behalten ist oft nicht so einfach. Manchmal hängt es vielleicht an nur einem Datum, einem offenen Port oder einer Programmversion, um es dem Pentester zu ermöglichen, einen erfolgreichen Angriff durchzuführen. Dann wäre es umso ärgerlicher, wenn genau dieses Datum in der Menge der ermittelten Daten unterginge.

In diesem Artikel erfahren Sie…
- wie Daten in das Tool MagicTree eingegeben bzw. importiert werden können
- welche Programme als Datenquelle eingesetzt werden können
- wie Reports erstellt werden können
- wie eigene Daten automatisch in einen Report importiert werden können

Einführung
- Das Programm MagicTree kann von *http://www.gremwell.com/* heruntergeladen werden. Hier findet sich auch ein Forum und ein Blog, in denen Sie bei Bedarf Hilfe suchen und finden können. Auf der neuesten Backtrack Distribution BT5R2 ist die aktuelle Version 1.1 schon vorinstalliert und kann unter BackTrack/Reporting Tools/Evidence Management/magictree sofort gestartet werden.

Die grundlegende Philosophie von MagicTree bei der Datenspeicherung ist, wie der Name schon sagt, die Erzeugung eines Datenbaums. Dieser Baum wird dann mit XPATH [1] durchsucht. Eine schnelle Einführung in XPATH findet sich auch in der Dokumentation von MagicTree [2]. Das Programm stellt mehrere Knotentypen zur Verfügung, die schon einen großen Teil der vorkommenden Daten repräsentieren können.

Knotentypen

- **Bereichsknoten:** werden verwendet, um die Struktur des Baumes zu erzeugen.
- **Einfache Knoten:** Hierbei handelt es sich um den gebräuchlichsten Kontentyp. Er speichert einfache Daten, wie z.B. IP-Adressen oder Domainnamen.
- **Textknoten:** In diesem Knotentyp können Sie Text abspeichern. Hier können Sie Anmerkungen über den abgelaufenen Test einpflegen, aber auch Informationen, die später im Report auftauchen sollen.
- **Datenknoten:** Speichert alle Daten, außer Bilder und XML Anhänge im Projektordner.
- **XML Datenknoten:** Speichert XML Daten.
- **Bildknoten**: Speichert Bilder.
- **Cross-Referenzen:** Erzeugt Verbindungen (Links) zwischen Knoten (um z.B. redundante Daten zu vermeiden).
- **Übersichtsknoten**: Hier können Testergebnisse eingegeben und mit betroffenen Geräten verbunden werden.
- **Spezialknoten**: Dieser Knoten wird nicht vom Anwender, sondern vom Programm selbst erzeugt. Er wird von der Anwendung für bestimmte Aufgaben eingesetzt.

Im weiteren Verlauf werden Sie den einen oder anderen Knotentypen einsetzen.

Auf geht's…
Nach dem Start des Programms werden zuerst die Lizenzbedingungen angezeigt, die für den weiteren Gebrauch akzeptiert werden müssen. Dann erscheint der Hauptbildschirm, in dem fast alle Aktionen durchgeführt werden.

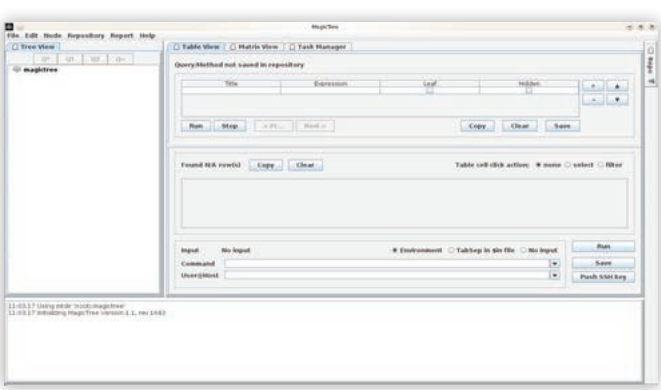


Abbildung 1: *Hauptbildschirm von MagicTree*

Daten können sowohl manuell als auch über einen automatischen Import von XML-Daten eingegeben werden. Hier soll zunächst ein Knoten manuell erzeugt werden. Das können Sie sowohl über das Menü (**Node/Autocreate**), als auch über den Shortcut *CTRL-N* erreichen.

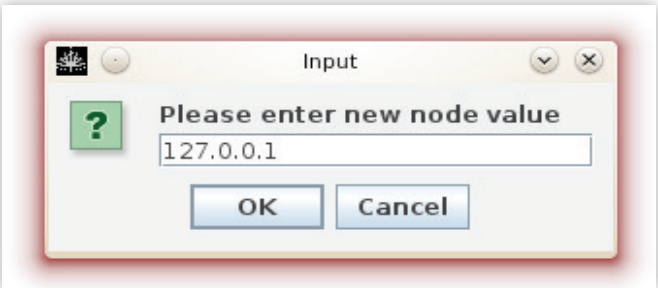


Abbildung 2: *Erzeugen eines neuen Knotens*

Als kleines Beispiel wird die Loopback-IP-Adresse des eigenen Backtrack-Rechners verwendet.

Von MagicTree (von jetzt ab nur noch MT genannt) wird automatisch ein Bereichsknoten erzeugt und in diesem ein einfacher Knoten mit unserer IP-Adresse. Diesem Knoten können nun weitere Daten zugefügt werden. Dazu reicht ein Links-Klick auf den Knoten und Sie können den Knotentypen auswählen, den Sie einfügen wollen.

Über *Create Child* lässt sich anschließend ein zugehöriger Text erzeugen.

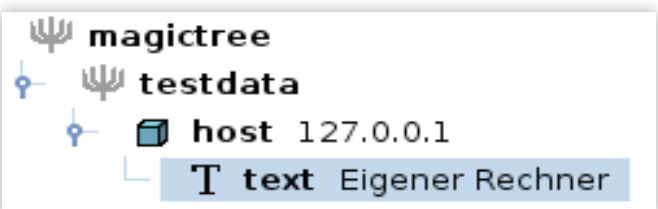


Abbildung 3: *Zufügen eines Textknotens zum Host-Knoten*

Auf diese Art können weitere Daten zum Host-Knoten zugefügt werden.

Interessant wird es aber erst, wenn automatisch Ergebnisse anderer Programme eingefügt werden. Das kann mit MT für eine Reihe von Programmen veranlasst werden. Aktuell werden von MT folgende Programme unterstützt (wobei die Liste ständig größer wird):

- R1apid 7 NeXpose
- Arachni
- OWASP Zed Attack Proxy
- Nessus (v1 and v2)
- Nikto
- Nmap
- Burp
- Qualys
- Imperva Scuba
- OpenVas

Als nächstes wird in diesem Beispiel mit **nmap** der eigene Rechner gescannt und die Ergebnisse in MT importiert. **Nmap** kann über die Befehlszeile im rechten Fenster gestartet werden. Zuerst muss dazu die Verknüpfung mit dem Host-Knoten hergestellt werden. Dazu klicken Sie oben im *Tree View* auf ***Q****. Die bisher bekannten Daten werden in das rechte Fenster (*Table View*) übernommen. Alle ausgeführten Befehle werden nun auf die im **Host**-Fenster des *Table View* angezeigten Geräte ausgeführt. Hier könnte statt einer einzelnen IP-Adresse auch ein IP-Bereich stehen, z.B. 192.168.1.0/24.

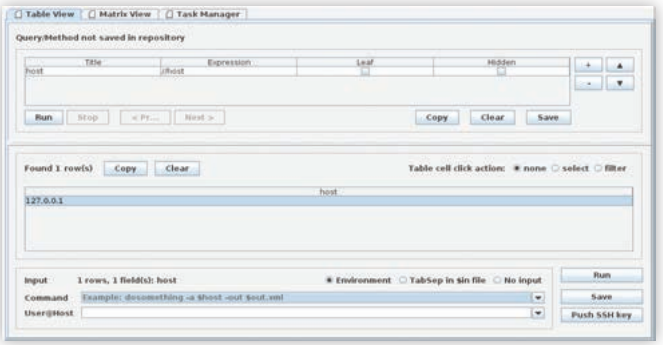


Abbildung 4: *Tabellenansicht der Daten vom Host-Knoten*

Es soll folgender Scan ausgeführt werden:

nmap –vv –O –sS –A –p- P0 127.0.0.1

Um die Ergebnisse in MT importieren zu können, müssen sie im XML-Format in eine Datei geschrieben werden. Weiterhin soll die IP-Adresse aus den Daten des Host-Knotens übernommen werden. Der in MT einzugebende Befehl sieht dann folgendermaßen aus:

nmap –vv –O –sS –A –p- P0 -oX $out.xml $host

Dieser Befehl wird in die *Command*-Zeile im **Input**-Abschnitt des *Table View* Fensters eingegeben.

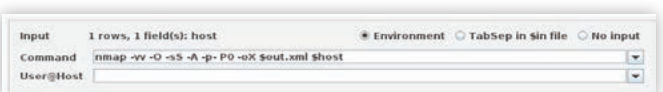


Abbildung 5: *Befehlseingabe für* **nmap** *in MT*

Starten Sie die Befehlsausführung mit einem Klick auf **Run**.

Auf der rechten Seite wird der Reiter *Task Manage*r angezeigt. Hier wird die gerade ausgeführte Task mit dem aktuellen Status gezeigt (nach dem Scan sollte hier *done* stehen) und die erzeugten Ausgabedateien (eine **Log**-Datei, in der exakt die Ausgabe von **nmap** steht, und die **$out.xml**-Datei, in der die Daten im XML-Format vorhanden sind). Auf der rechten Seite sind die Dateiinhalte aufgelistet.

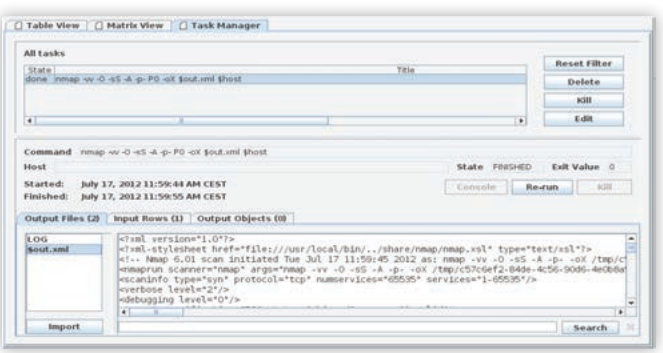


Abbildung 6: *Inhalt der erzeugten Dateien*

Noch sind aber die Daten nicht in MT importiert. Dazu muss im linken Fenster die Datei $out.xml ausgewählt und unten der Import gestartet werden.

Nachdem die Daten importiert wurden, werden automatisch generierte neue Knoten im linken *Tree View* Fenster angezeigt. Alle ermittelten Daten (Hostname, Betriebssystem, offene Ports und zugehöriger Dienst mit Versionsnummer) werden in eigenen Knoten angezeigt.

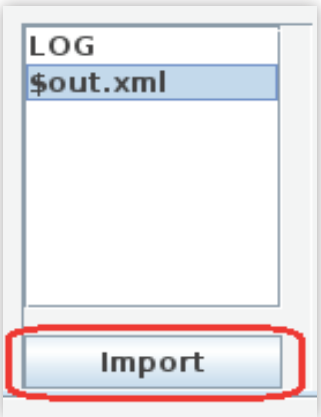


Abbildung 7: *Starten des Datenimports*

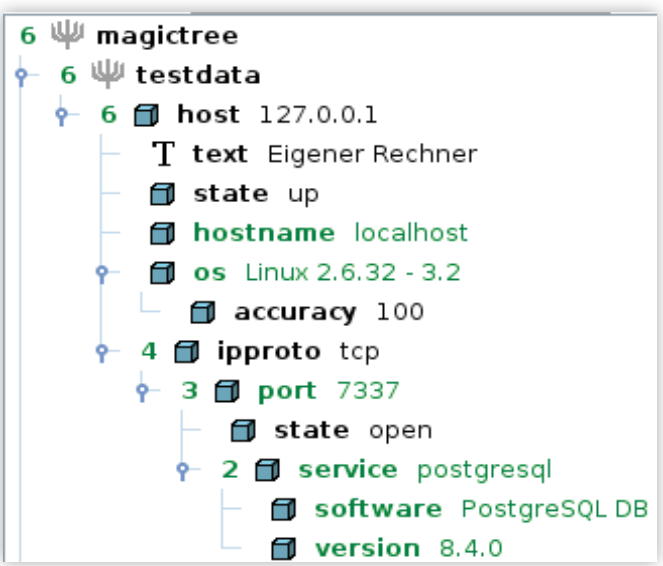


Abbildung 8: *Tree View nach dem Datenimport*

Falls Sie mehrere Knoten zusammenfassen wollen, steht Ihnen der Merge-Befehl im File-Menü zur Verfügung. Dabei werden Daten, die in beiden Knoten vorhanden sind, nicht doppelt übernommen, sondern nur einmal. Damit können z.B. IP-Adressbereiche, die sich überschneiden, problemlos zusammengefügt werden, jeder Host ist im resultierenden Knoten nur einmal vorhanden.

Sollten Knoten zwar doppelt vorhanden, aber mit unterschiedlichen Daten verknüpft sein, werden sie zusammengefügt und es gibt nach dem Zusammenfügen nur einen Knoten, aber mit allen Daten aus beiden Ausgangsknoten.

Raus mit den Daten…

Wenn auf diese Weise alle Daten gesammelt wurden, sollte der Report gestartet werden. Es ist möglich, die Daten sowohl im OpenOffice- als auch im MS-Word-Format zu erstellen.

Hinweis: Um das folgende Beispiel mit BackTrack durchführen zu können, muss OpenOffice [3] auf dem System installiert sein.

Je nach gesammelten Daten sieht so ein Report natürlich unterschiedlich aus. MT bringt zu diesem Zweck schon eine Reihe von Templates mit, die durch eigene Templates, ganz nach Belieben ergänzt werden können.

Als kleines Beispiel sollen nun unsere NMAP-Daten in ein OpenOffice-Dokument geschrieben werden. Zur Reporterstellung klicken Sie oben im Menü auf Report/Generate Report. Als Zieltemplate wählen Sie *open-ports-and-summary-of-findings-by-host.odt*. In diesem Template werden alle Hosts, deren ermittelten offenen Ports und Dienste, sowie alle von einem Schwachstellenscanner erkannten Schwachstellen (nach Hosts gruppiert) aufgelistet. Sie können Templates auch selber erstellen und dabei alles verwenden, was das Textverarbeitungssystem hergibt. Texte und Bilder, je nach Verwendung formatiert, können so vorgegeben werden. Die von MT zu übernehmenden Daten werden über Platzhalter in das Dokument eingefügt. Wie so etwas geht, wird später noch genauer erklärt. Sehr hilfreich ist auch die (sehr verständlich geschriebene) Hilfe, die über das Hauptmenü aufgerufen werden kann. Wer ein komplett eigenes Template erstellen will, findet hier wertvolle Hinweise.

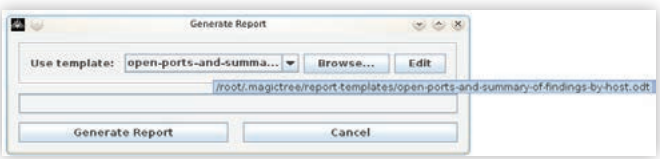


Abbildung 9: *Reportgenerierung*

Doch leider schlägt in der aktuellen Version 1.1 ein Java-Fehler zu.

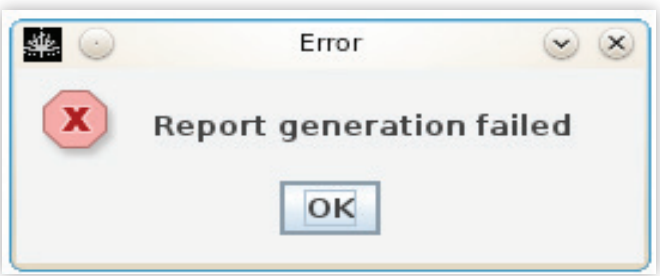


Abbildung 10: *„Falsche" Fehlermeldung*

Dabei wird nicht die Erstellung des Reports abgebrochen (wie in der Fehlermeldung angegeben), sondern das automatische Starten des erzeugten Reports mit OpenOffice ist das (Java-)Problem. Ein Bugfix ist für die nächste (Unter-)Version abgekündigt.

Eine zugehörige Fehlermeldung wird auch unten im Meldefenster von MT ausgegeben.

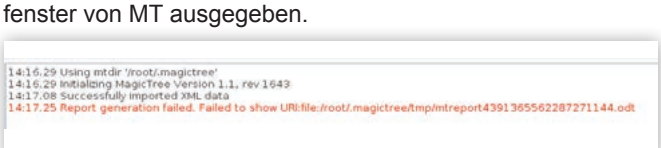


Abbildung 11: *Fehlermeldung im Meldefenster von MT*

Da aber der Report erstellt wurde, kann er manuell geöffnet werden.

Reports werden standardmäßig im Verzeichnis */root/.magictree/tmp* gespeichert (Achtung: verstecktes Verzeichnis!), wie auch in der Fehlermeldung (Abb.11) zu sehen ist. Dabei werden die Namen aus zwei Teilen gebildet: zuerst der immer gleiche Namensteil *mtreport*, dann eine zufällige Zahl.

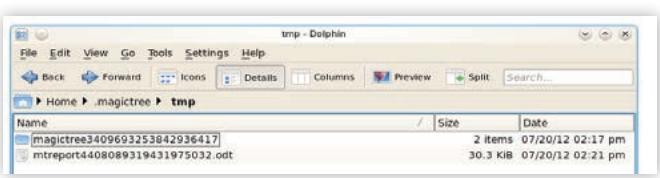
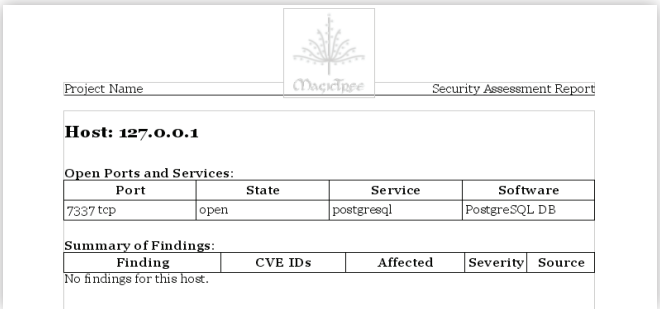


Abbildung 12: *Erzeugte Reportdatei im Verzeichnis /root/.magictree/tmp*

Mit einem Doppelklick lässt sich der Report (bei installiertem Openoffice) öffnen.



Project Name — Security Assessment Report

Host: 127.0.0.1

Open Ports and Services:

Port	State	Service	Software
7337 tcp	open	postgresql	PostgreSQL DB

Summary of Findings:

Finding	CVE IDs	Affected	Severity	Source
No findings for this host.				

Abbildung 13: *Report*

In dieser Datei sind nun die meisten (im vorliegenden Fall zugegebenermaßen nur wenige) Daten enthalten. So wurde z.B. das erkannte Betriebssystem (Linux 2.6.32 -3.2) nicht mit übernommen, weil es im Template fehlt.

Um die oben schon kurz beschriebenen Templates etwas zu veranschaulichen, wird nun das verwendete Template so geändert, dass auch das Betriebssystem angezeigt wird.

Dazu öffnen Sie die Template-Datei *open-ports-and-summary-of-findings-by-host.odt* mit OpenOffice.

Project Name — Security Assessment Report

Host: {{//host[@status!='ignore']}}

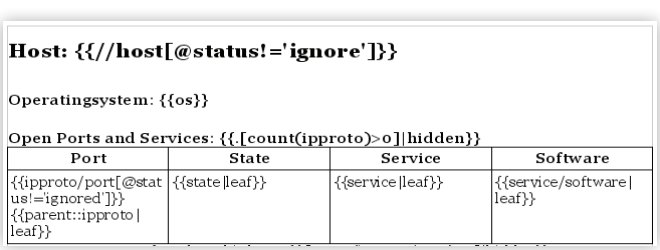
Open Ports and Services: {{.[count(ipproto)>0]|hidden}}

Port	State	Service	Software
{{ipproto/port[@status!='ignored']}} {{parent::ipproto|leaf}}	{{state|leaf}}	{{service|leaf}}	{{service/software|leaf}}

No open ports were found on this host. {{.[count(ipproto/port)=0]|hidden}}

Abbildung 14: *Ausschnitt aus der Datei open-ports-and-summary-of-findings-by-host.odt*

Die Platzhalter korrespondieren mit den Daten im linken *Tree View* Fenster von MT. Hier findet man z.B. den Host mit der IP-Adresse 127.0.0.1. Er wird im Template mit {{//host}}, die IP-Adresse mit {{ipproto}} referenziert. Wenn also das Betriebssystem mit aufgenommen werden soll, muss analog dazu {{os}} verwendet werden. Eine Änderung könnte also etwa so aussehen, wie in Abbildung 15 gezeigt.



Host: {{//host[@status!='ignore']}}

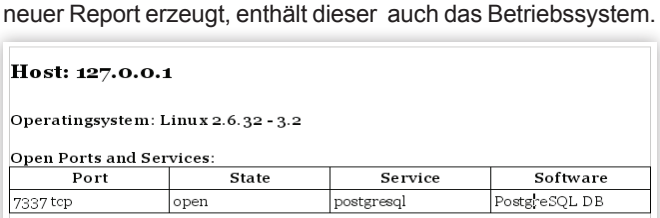
Operatingsystem: {{os}}

Open Ports and Services: {{.[count(ipproto)>0]|hidden}}

Port	State	Service	Software
{{ipproto/port[@status!='ignored']}} {{parent::ipproto|leaf}}	{{state|leaf}}	{{service|leaf}}	{{service/software|leaf}}

Abbildung 15: *Ausschnitt aus der Datei open-ports-and-summary-of-findings-by-host.odt mit Änderung*

Wird diese Datei unter dem Namen *my-open-ports-and-summary-of-findings-by-host* gespeichert und anschließend ein neuer Report erzeugt, enthält dieser auch das Betriebssystem.



Host: 127.0.0.1

Operatingsystem: Linux 2.6.32 - 3.2

Open Ports and Services:

Port	State	Service	Software
7337 tcp	open	postgresql	PostgreSQL DB

Abbildung 16: *Report mit Betriebssystemanzeige*

Import weiterer Daten…

Auf ähnliche Weise können Daten aus weiteren Programmen importiert werden. Als weiteres Beispiel soll das Programm **Nikto** verwendet werden, um die Schwachstellen eines Webservers (im Beispiel 192.168.1.2) festzustellen. Als Angriffsobjekt wird DVWA[5] in der Version 1.0.7 verwendet. Nachdem ein entsprechender Host-Knoten erstellt wurde, kann folgender Befehl zum Start von **nikto** eingegeben werden:

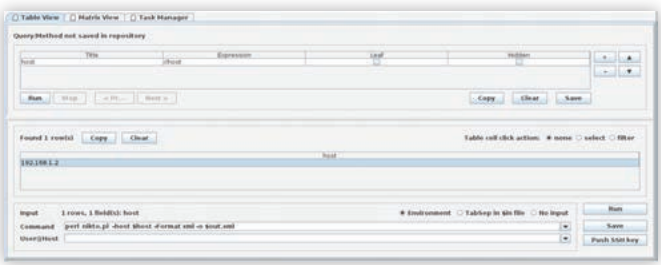


Abbildung 17: *Untersuchung eines Webservers mit* **nikto**.

Nachdem das Programm beendet wurde, kann die Outputdatei $out.xml in MT importiert werden.

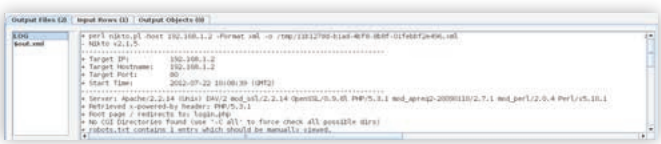
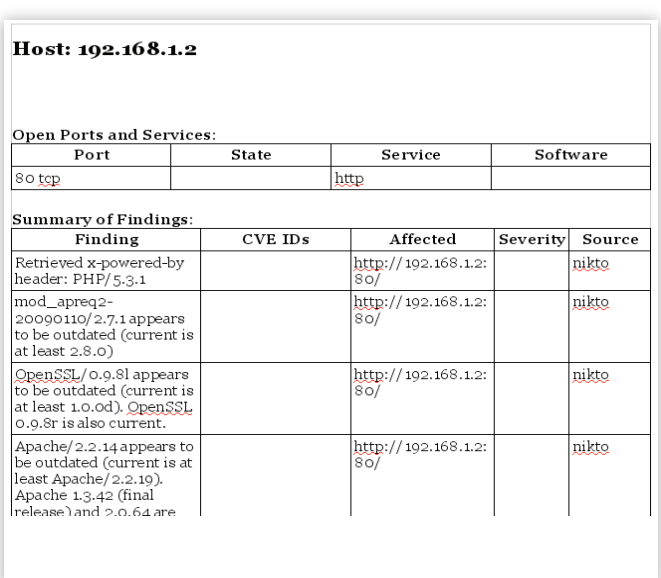


Abbildung 18: *Ergebnisausgabe des Programms* **nikto**

Nach dem Import werden die gefundenen Ergebnisse im *Tree-View* angezeigt.

Abbildung 19: *Ergebnisse im TreeView*

Ein Report enthält nun auch die entsprechenden Ergebnisse.



Host: 192.168.1.2

Open Ports and Services:

Port	State	Service	Software
80 tcp		http	

Summary of Findings:

Finding	CVE IDs	Affected	Severity	Source
Retrieved x-powered-by header: PHP/5.3.1		http://192.168.1.2:80/		nikto
mod_apreq2-20090110/2.7.1 appears to be outdated (current is at least 2.8.0)		http://192.168.1.2:80/		nikto
OpenSSL/0.9.8l appears to be outdated (current is at least 1.0.0d). OpenSSL 0.9.8r is also current.		http://192.168.1.2:80/		nikto
Apache/2.2.14 appears to be outdated (current is at least Apache/2.2.19). Apache 1.3.42 (final release) and 2.0.64 are		http://192.168.1.2:80/		nikto

Abbildung 20: *Report mit den Ergebnissen von* **nikto**

Eigene XML-Dateien importieren

Es können aber auch eigene Daten automatisch importiert werden. Dabei kommen XSLT Transforms zum Einsatz, die vorhandene XML-Dateien so aufbereiten, dass sie von MT verarbeitet werden können. Dieser Vorgang ist allerdings etwas komplexer und würde den Rahmen dieses Artikels sprengen. Einige in dieser Richtung interessante Webseiten sind in der Literaturliste angegeben [6], [7]. Beispiele sind aber auch in der MT Dokumentation zu finden [8].

Natürlich können Sie alle Aktionen und deren Ergebnisse sichern. Über File/Save ist das ganz einfach möglich. Das gesamte Arbeitsverzeichnis wird in einer Datei (komprimiert) gespeichert.

Resüme und Ausblick

MagicTree ist ein Programm mit Potential. Auch wenn hier nur einige grundlegende Möglichkeiten gezeigt wurden, kann der Wert dieses Programmes für den Pentester enorm groß sein. Die Flexibilität durch die weitreichende Anpassbarkeit an die eigenen Arbeitsgewohnheiten sorgt für eine Arbeitserleichterung bei der Erstellung und Strukturierung des Abschlussreports. Das Programm liegt erst in der Version 1.1 vor, aber die Entwickler geben schon Ausblicke auf die kommenden Erweiterungen. So soll weiter daran gearbeitet werden, dass das Programm auch besser im Team eingesetzt werden kann. So wäre es dann auch besser möglich MT im Team einzusetzten und alle Daten an zentraler Stelle zu sammeln.

Literatur

- [1] XML Path Language (XPath), *http://www.w3.org/TR/xpath/*
- [2] MagicTree Documentation: XPath Crash Course - Learning By Example, *http://www.gremwell.com/magictreedoc/2ac07abf.html*
- [3] OpenOffice, *http://www.openoffice.org/de/*
- [4] Lee Allen, „Advanced Penetration Testing for Highly-Secured Environments: The Ultimate Security Guide, [PACKT] Publishing May 2012
- [4] MagicTree Documentation, *http://www.gremwell.com/documentation*
- [5] Damn Vulnerable Web Application (DVWA), *http://www.dvwa.co.uk/*
- [6] Bending MagicTree (Burp import), *http://blog.astyran.sg/2012/03/bending-magictree-burp-import.html*
- [7] Lighting up Gremwells MagicTree with Arachni Data, *http://blog.astyran.sg/2012/01/lighting*-up-gremwells-magictree-with.html
- [8] Adding XSLT Transforms, *http://www.gremwell.com/magictreedoc/9a673327.html*

HANS HÖFKEN

ist Leiter der Rechenzentrale und wissenschaftlicher Mitarbeiter im Fachbereich Elektrotechnik und Informationstechnik der FH Aachen. Er arbeitet auf dem Gebiet IT-Sicherheit und IT-Management und ist verantwortlicher Trainer der Cisco Academy der FH Aachen, Kursleiter von Hacking- und IT- Forensikkursen und ISO 27001 Auditor.
hoefken@fh-aachen.de

MARKO SCHUBA

ist Professor an der FH Aachen. Davor war er über zehn Jahre in der Telekommunikations-Industrie tätig. Er lehrt und forscht in den Bereichen Datennetze und IT-Sicherheit sowie - als einer der wenigen Professoren in Deutschland – im Bereich IT-Forensik. Darüber hinaus ist er IT Sicherheitsbeauftragter der FH Aachen und geschäftsführender Gesellschafter der schuba & höfken Gbr.
schuba@fh-aachen.de

MONITOR STRONY

Innovative e-Dienstleistungen für die Überwachung von Internetseiten

SEOmonitor
Überwachung von Internetseiten für den Bedarf von SEO

SPEEDmonitor
Überwachung der Ladegeschwindigkeit einer Internetseite

CONTENTmonitor
Überwachung der sprachlichen Korrektheit
der auf einer Internetseite veröffentlichten Inhalte

www.monitorstrony.pl

INNOVATIVE ECONOMY
NATIONAL COHESION STRATEGY

ZUWENDUNGEN FÜR INNOVATIONEN

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND

DAS PROJEKT WIRD VON DER EUROPÄISCHEN UNION AUS DEM EUROPÄISCHEN FONDS FÜR REGIONALE ENTWICKLUNG MITFINANZIERT

Web Audit Authority

Ein Internetservice für eine automatische Durchführung von Audits von Internetseiten

technischer e-Audit einer Internetseite
e-Audit der Zugänglichkeit einer Internetseite
e-Audit der Verwendbarkeit einer Internetseite

www.webauthority.eu

INNOVATIVE ECONOMY
NATIONAL COHESION STRATEGY

ZUWENDUNGEN FÜR INNOVATIONEN

EUROPEAN UNION
EUROPEAN REGIONAL DEVELOPMENT FUND

DAS PROJEKT WIRD VON DER EUROPÄISCHEN UNION AUS DEM EUROPÄISCHEN FONDS FÜR REGIONALE ENTWICKLUNG MITFINANZIERT

BACKTRACK 5

JULIUS BIERMANN

Die kostenlose Linux Distribution Backtrack umfasst viele integrierte Funktionen und Tools. Unter anderem auch das Tool: „Aircrack". Dieses Programm, welches sich über die systemeigene Shell bedienen lässt, knackt durch das Mitschneiden von Datenpaketen und der bekannten Wörterbuchattacke das Wlan Passwort eines Routers. Dieser Artikel bringt Ihnen Schritt für Schritt die Einzelheiten näher und erleutert diese.

DISCLAIMER:
Dieser Arktikel stellt jedeglich Informationen da! Er dient nicht dazu, Sie in die Lage zu versetzten, ein Wlan Passwort zu knacken! Hier wird nur gezeigt, wie einfach es für einen Angreifer ist! Das Nachmachen ist jedeglich an eigenen Servern gestattet!

Das Tool: „Aircrack", welches zum Standert Equipment der Linux Distribution „Backtrack" gehört, ist für Hacker und IT-Experten schon zum Standard geworden. Das Tool lässt sich auch problemlos auf jeder anderen Linux-Distribution installieren.

Hierbei ist es ein leichtes den Namen des Programms in die Standardsuchmaschine einzugeben und das Tool zu downloaden.

Zunächst ist es wichtig zu wissen, dass die sogennate WEP-Verschlüsselung, welche trotz zunehmender Schwachstellen und Alters immernoch gebraucht wird, durch das Analysieren von mitgeschnittenden Datenpaketen (das so gennante „sniffen") geknackt bzw. errechnet wird.

Dagegen die WPA Verschlüsselung ist wesentlich sicherer, da diese von Aircrack nur über eine Wörterbuchattacke, bei der nacheinander vorher aufgelistete Wörter durchlaufen, gelöst werden kann.

Das Tool „Aircrack", das von Christophe Devine entwickelt wurde, ist eigentlich nur eine Zusammenstellung aus verschieden Programmen.

Hier einmal die einzelnen Programmteile aufgelistet:

- **aircrack-ng**
  Berechnung von WLAN-Schlüsseln
- **airodump-ng**
  Paket-Sniffer (schneidet Pakte mti)
- **aireplay-ng**
  schleust Pakete in Netzwerke ein
- **airdecap-ng**
  Entschlüsselung an Hand eines bekannten Schlüssels
- **airmon-ng**
  versetzt WLAN-Karten in den Monitor-Modus
- **airtun-ng**
  erzeugt virtuelle Tunnel
- **airolib-ng**
  Speichert Passwortlisten
- **wesside-ng**
  berechnet automatisch die WEP-Schlüssel von gefundenen Netzwerken
- **airdriver-ng**
  baut und installiert WLAN-Treiber unter Linux
- **airbase-ng**
  Simuliert Acces-Point

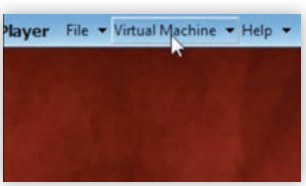
In diesem Artikel werden wir uns auf das Lösen eines WEP-Netzwerkes beschränken, da diese Methode am häufigsten angewandt wird und somit für Sie am erschwinglichsten ist.

Falls Sie den Vorgang an einer Virtuellen Maschine durchführen möchten, müssen Sie erst Ihre Wlan-Karte richtig anschließen. Dies wird unser erster Schritt sein auf dem Weg zum Wlan-Passwort.

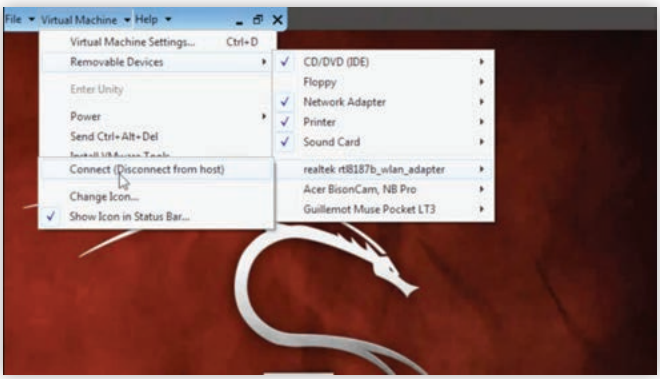
In diesem Fall wird davon ausgegangen VMWare-Player zu benutzen.

Schritt 1:

Nachdem die graphische Oberfläche gestartet worden ist, oben in der Leiste auf *Virtual Machine* klicken.



Schritt 2:
Removable Devices -> „Ihre Wlan-Adapter“ -> *Connect*



Bei dieser Methode koppelt sich die Wlan-Karte vom Host ab und schaltet sich an den Virtuellen Pc.

Für ein ausführliches Video Tutorial: http://www.youtube.com/watch?v=ZST3mq14x3I

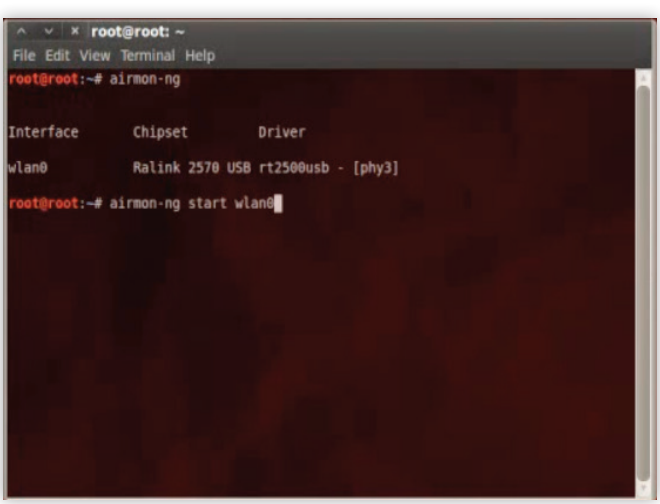
Nachdem dieser Schritt geschafft ist, können wir mit dem eigentlichen Teil beginnen.

Hier zu erstmal die Shell aufrufen. Diesen Schritt werde ich nicht erklären, da dies zu dem KnowHow eines erfahrenen Benutzers gehört.

Schritt 1: Wlan-Karte in den Monitor Modus setzen.
Der Befehl „airmon-ng“ listet uns alle verfügbaren Wlan-Karten auf, die in den Monitor Modus versetzbar sind.

In dem nächsten Befehl geben wir mit dem Parameter „start“ hinter airmong-ng an, dass wir das jewalige Interface, welches wir hinter den Parameter schreiben, starten wollen, bzw. in den „Monitor mode“ setzen wollen.

Ob die Aktion geglückt ist, sehen wir an der Nachricht: *„monitor mode enabled on mon0“*

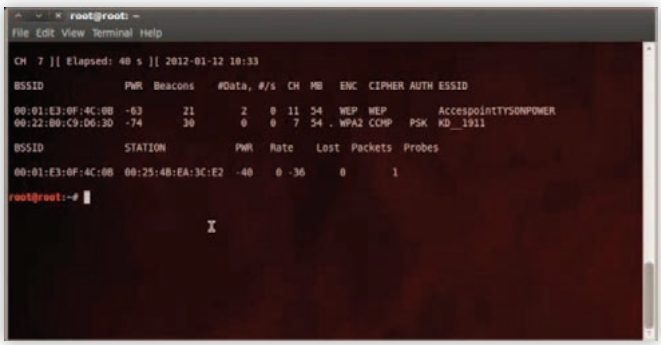


Schritt 2: Router bzw. Clients suchen.
Mit dem Befehl *„airodump-ng mon0“* werden uns alle Router aufgelistet

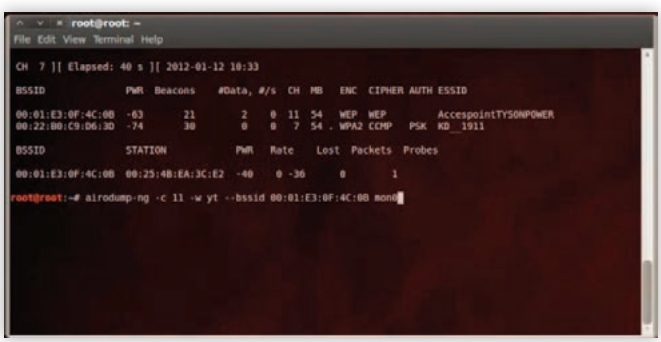
Desweiteren werden im unteren Bereich die Clients angezeigt, die mit den Routern Packete austauschen (der so genannte *Traffic*)

Um die Suche anzuhalten, muss einmal *„Strg+Z“* gedrückt werden, nachdem eine Zeit gesucht wurde.

(In diesem Fall ist der zu hackende Router der obige.)



Schritt 3: Router bzw. Clients überprüfen.
Dies wird mit folgendem Befehl gemacht: airodump-ng -c „Hier wird der Channel eingefügt oben in der Shell abgekürzt mit CH“ –w „Ein beliebiger Name“ –bssid „Die oben stehende BSSID vom zu hackenden Router.

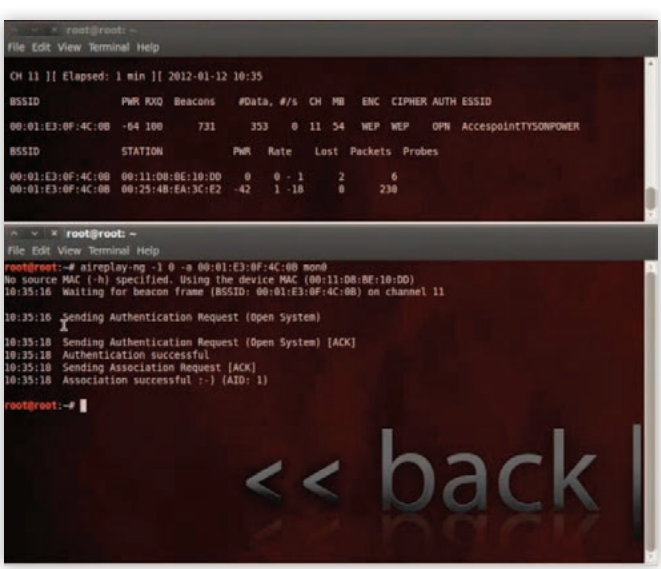


Schritt 4: Neue Shell + Packe an den Router schicken
Für diesen Schritt ist eine neue Shell erforderlich. Diese wird genauso wie die erste geöffnet.

Nun wollen wir auch Packete an den Router schicken. Die machen wir mit folgendem Befehl:

aireplay-ng -l -a „BSSID des zu hackenden Routers“ mon0

Sofort erscheint in der alten Shell ein neuer Client, der ebenfalls Traffic mit dem Router betreibt.



Schritt 5: Router das Passwort abknöpfen.
Hier wird nun mit dem Befehl: *aireplay-ng -3 -b „BSSID des zu hackenden Routers“ mon0*

gebraucht, um dem Router sehr viele Packete zu schicken. Dieser Anwortet ebenfalls mit Packeten.

Den Traffic können wir in der letzten Zeile der neuen Shell beobachten.

Sobald der Router auf ca. 5000 Packete geantwortet hat, kommt er ins Schlingern und schickt mit den Packeten Teile von dem Wlan-Passwort mit.

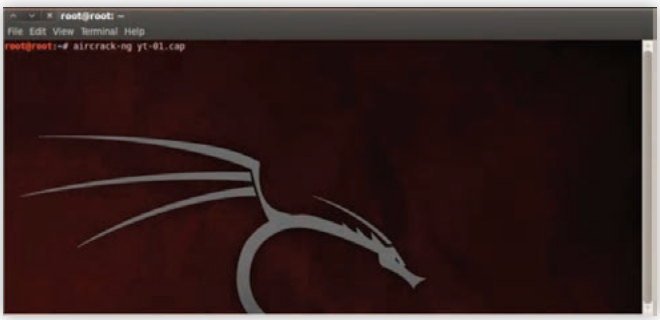
Für den nächsten Schritt ist eine neue Shell notwendig!

Schritt 6: Das Finale – Knacken des Passworts

In die neu geöffnete Shell wird nun der Befehl zum Cracken hineingetragen:

Aircrack-ng „Der Name der bei Schritt 3 ausgesucht wurde“ -01.cap

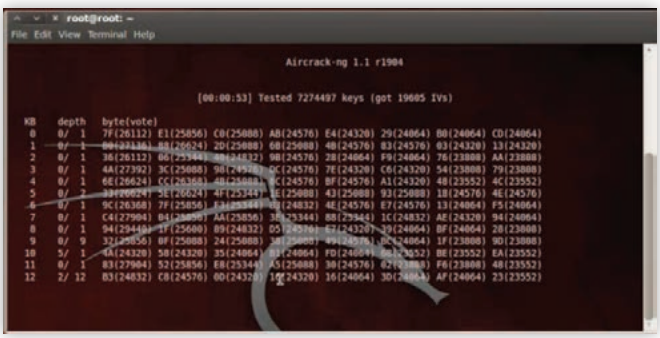
Bevor Sie den Befehl abschicken, sollten Sie über ca. 20.000 beantwortete Packete haben.



Schritt 6.1: Warten...

Das Passwort wird nun aus den beantworteten Packeten berechnet.

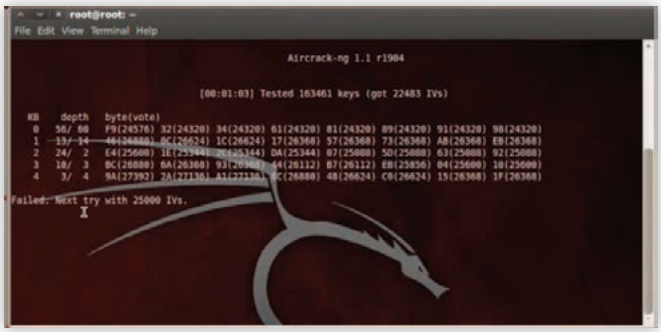
Dieser Vorgang kann etwas dauern.



Oooooops: Ein Fehler ist aufgetreten.

Es kann passieren, dass wie hier eine Fehlermeldung *„Failed. Next try with 25000 IVs.“* auftaucht.

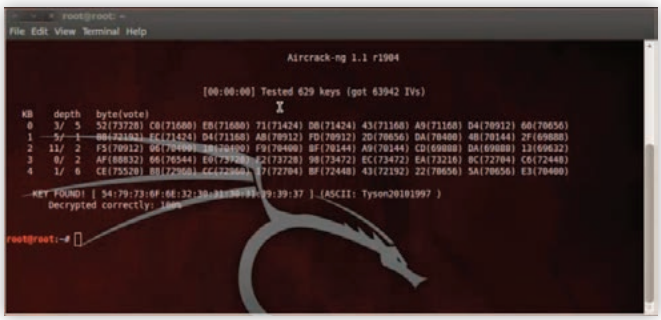
Falls dies bei Ihnen passiert, müssen Sie Schritt 5 wiederholen und auf noch mehr beantwortete Packete warten.



Schritt 7: Tatatataaaa!

An der Nachricht „KEY FOUND!“ ist zu sehen, dass das Passwort erfolgreich geknackt wurde.

In der ersten Klammer steht eine Zahlenkombination, welche verschlüsselt ist. In der zweiten steht allerdings hinter „ASCII“ das Passwort in „menschlicher Schrift geschrieben“



Herzlichen Glückwunsch, Sie haben es geschafft! Für weitere spannende Tutorials dieser und anderer Art besuchen Sie doch einmal meinen YouTube-Channel: *http://www.youtube.com/user/TheNewestTut?feature=watch*

Ich würde mich sehr über einen Besuch von Ihnen freuen

JULIUS BIERMANN

IDENTIFIZIERUNG VON SCHWACHSTELLEN BEI WEBANWENDUNGEN MIT SKIPFISH

MIKE KUKETZ

Ein Alltag ohne Webanwendungen ist heute kaum noch vorstellbar. Unzählige Anfragen werden täglich von Webservern verarbeitet und anschließend im Browser des Nutzers dargestellt. Angefangen bei der Suche nach Informationen, sozialer Vernetzung, über das Lesen von E-Mails bis hin zum Online-Shopping – all das und mehr wird von Webanwendungen realisiert.

In ihrer Komplexität übertreffen Webanwendungen oft die einer Desktopanwendung. Gerade die Möglichkeit, dass Webanwendungen von jedem und zur jeder Zeit nutzbar sind, machen sie zu einem sehr beliebten Angriffsziel. Kreditkartennummern und Kundendaten stehen dabei ganz oben auf dem Wunschzettel der Angreifer. Etablierte Sicherheitsmaßnamen wie Firewalls und Intrusion-Detection- bzw. Prevention-Systeme bieten dagegen kaum ausreichend Schutz.

Eine gute Strategie, um eine Webanwendung bzw. den dahinter liegenden Webserver abzusichern ist eine potenzielle Schwachstelle vor einem Angreifer zu finden. Dazu eignen sich diverse Tools – sogenannte **Web Vulnerability Scanner**. Diese automatischen Scanner werden eingesetzt, um zunächst eine Grundlage zu schaffen, die wiederum als Ausgangspunkt für manuelle Tests dient. Eine Auswahl dieser Tools befindet sich auf **BackTrack 5**, einer auf Ubuntu basierten Linux-Distribution speziell für IT-Sicherheitsexperten, Pen-Tester oder an IT-Security interessierten Personen. Eines dieser automatisch agierenden Scanner nennt sich skipfish und wird in diesem Artikel vorgestellt.

Abbildung 1: *skipfish Logo*

skipfish

Der von Google entwickelte Sicherheitsscanner für Webanwendungen ist als Open Source freigegeben. Entwickelt wurde der Scanner in der Programmiersprache C und soll durch sein optimiertes HTTP Handling besonders flink arbeiten. Je nachdem, ob sich das Ziel im Internet oder lokalen LAN befindet, setzt skipfish zwischen 500 bis 2000 Anfragen ab. Auf einem lokalen Rechner sind es sogar bis zu 7000 Anfragen. Ist der Scanner einmal gestartet, sucht er automatisiert nach diversen Sicherheitslücken, wie zum Beispiel Cross-Site-Scripting (XSS) oder SQL-Injections. Eine komplette Auflistung der von skipfish durchgeführten Tests findet sich in der Dokumentation.
http://code.google.com/p/skipfish/wiki/SkipfishDoc#Most_curious!_What_specific_tests_are_implemented?

Wie arbeitet skipfish?

Zunächst generiert skipfish durch rekursives Crawlen eine interaktive Sitemap der Webseite. Dort sind alle möglichen Links bestehend aus Dateien und Verzeichnisse hinterlegt, die während des Scan-Vorgangs untersucht werden. Interessant dabei ist das Vorgehen von skipfish, um wirklich jeden Link bzw. Datei oder Verzeichnis in der Sitemap zu indexieren. Schließlich macht ein Scan nach Sicherheitslücken nur dann Sinn, wenn auch alles geprüft wird. Um eine möglichst komplette Sitemap zu generieren, setzt skipfish Brute Force Techniken und Wörterbuch-Attacken ein. Mitgeliefert werden bereits unterschiedliche Varianten von Wörterbüchern, die bekannte und oft verwendete Namen kennen. Ergänzt wird die Technik durch ein adaptives Lernverfahren. Skipfish besitzt die Fähigkeit, neue Wörter und Wortkombinationen aus der Zielwebseite zu extrahieren. Die so

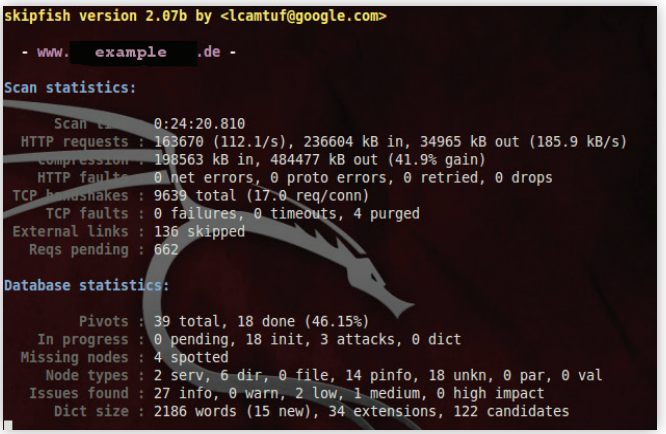


Abbildung 2: *Scan-Vorgang skipfish*

gewonnenen Daten können in einem Wörterbuch gespeichert und für weitere Scans genutzt werden.

Start des Scan-Vorgangs

Skipfish ist in BackTrack 5 bereits integriert. Die Installation von weiteren Paketen ist daher nicht notwendig. Das Tool befindet sich im Verzeichnis **/pentest/web/skipfish** und wird über die Konsole aufgerufen. Vor dem eigentlichen Scan-Vorgang lässt sich skipfish durch diverse Optionsparameter an die Umgebung bzw. das zu scannende Ziel anpassen.

Um einen ersten Scan zu initiieren sind folgende Schritte zu empfehlen:

```
cd /pentest/web/skipfish
touch new_dict.wl
./skipfish -S dictionaries/minimal.wl
-W new_dict-wl -o /home/skipfish http://www.example.com
```

Was bewirken die Optionen:-S: Definiert ein Wörterbuch. Auf das Wörterbuch wird nur lesend zugegriffen. **-W**: Im Gegensatz zur vorigen Option wird auf das Wörterbuch lesend und schreibend zugegriffen. Nach dem Scan wird das Wörterbuch mit neu gelernten Wörtern gefüllt. **-o**: Definiert ein Ausgabeverzeichnis für die Ergebnisse

Mit den gewählten Optionen wird aus dem **minimal.wl** Wörterbuch gelesen und neue Wörter werden mit der Auto-Learn Technik in das zuvor angelegte Wörterbuch **new_dict.wl** geschrieben. Da ansonsten keine Einschränkungen vorgenommen werden, arbeitet der Scanner im **normal-dictionary-fuzzing** Modus. In diesem Modus wird jedes Schlüsselwort aus dem Wörterbuch mit jeder Endung und jeder tatsächlich auf dem Webserver gefundenen Datei kombiniert. Aus den kombinierten Möglichkeiten werden im Anschluss Anfragen an den Webserver generiert, die dann alle möglichen Datei- und Verzeichnisnamen prüfen. Das ist der langsamste Modus, aber auch jener mit der höchsten Abdeckungsrate. Er eignet sich für schnell antwortende Server und kleine bis mittlere Webanwendung. Bei großen Projekten kann ein Scan-Vorgang sehr zeitintensiv sein. Je nach Größe des verwendeten Wörterbuchs und der Größe der Webanwendung ergeben sich unzählige kombinatorische Möglichkeiten. Es ist daher unmöglich vorauszusagen, wie viel Zeit eine Prüfung in Anspruch nimmt.

Während des Scan-Vorgangs werden kontinuierlich Statusupdates dargestellt. Je nach Umfang des ausgewählten Ziels nimmt der Scan einige Zeit in Anspruch. Zeit genug, um ein paar Optionen genauer unter die Lupe zu nehmen.

Optionsvielfalt von skipfish

Skipfish bietet zahlreiche Optionen, um den Scan an das Ziel anzupassen. Einige davon werden hier kurz vorgestellt. Eine Übersicht über alle Optionen erhält man mit folgendem Befehl:

```
./skipfish -h
```

Abbildung 3: *Optionen von skipfish*

Authentication und Access Optionen

Webanwendungen, die durch einen HTTP Authentifizierungsmechanismus gesichert sind oder bestimmte Cookies voraussetzen, stellen für skipfish kein Hindernis dar. Durch das Hinzufügen einer Option (-A user:pass) wird die HTTP Authentifizierung vorgenommen und der Scan kann wie gewohnt gestartet werden.

```
/skipfish -A user:password -S dictionaries/minimal.wl -o
   /home/skipfish http://www.example.com
```

Um die DNS-Namensauflösung zu umgehen, kann auch direkt die IP-Adresse (-F *http://IP*) des Ziels definiert werden. Ebenfalls sinnvoll falls das zu scannende Ziel über keinen DNS-Namen verfügt.

```
./skipfish -S dictionaries/minimal.wl -o /home/skipfish -F
                    http://127.0.0.1
```

Crawl Scope Optionen

Der Scan-Vorgang nimmt Zeit in Anspruch. Je nach gewählter Option kann der Zeitaufwand erheblich reduziert werden. Dazu stehen verschiedene Möglichkeiten zur Verfügung. Eine davon ist das gezielte Exkludieren von Links (-X STRING). Dazu wird skipfish ein String übergeben, der von der Prüfung dann ausgeschlossen wird.

```
./skipfish -X /manuals -S dictionaries/minimal.wl -o /home/
                    skipfish http://www.example.com
```

Umgekehrt ist dies ebenfalls möglich. Der Scan lässt sich durch die -I Option auf ein bestimmtes Verzeichnis reduzieren.

```
./skipfish -I http://www.example.com/dir1/ -S dictionaries/
minimal.wl -o /home/skipfish http://www.example.com
```

Kombinationsmöglichkeiten

Die Kombination aus verschiedenen Optionen ermöglicht skipfish auf Besonderheiten von Webanwendungen und Serverkonfigurationen zu reagieren. Zu viele Request Anfragen werden von Servern manchmal als DOS Angriff interpretiert, was die Blockierung der anfragenden IP-Adresse zur Folge haben kann. Ebenso kann es notwendig sein, den Scan auf ein bestimmtes Verzeichnis zu reduzieren. Sei es aus Geschwindigkeitsgründen oder weil dort eine Lücke geschlossen wurde, die anschließend geprüft werden muss. Anhand von zwei Beispielen wird im Folgenden die Kombination aus verschiedenen skipfish Optionsparametern dargestellt.

```
./skipfish -r 5000 -m 5 -L -o output_dir -b ie http://www.example.com/
```

- **-r 5000:** Beschränkung der maximalen Anzahl von Requests auf 5000 (Standard: 100000000)
- **-m 5**: Es werden maximal 5 Verbindungen zum Ziel initiiert.
- **-L**: Die Auto-Learn Funktion wird abgeschaltet.
- **-b IE**: Gegenüber dem Webserver gibt sich skipfish als Internet Explorer aus.

```
./skipfish -S dictionaries/complete.wl -P -I
    http://www.example.com/dir1/ -o output_dir -t 5
    http://www.example.com/dir1/
```

- **-P**: Vom Ziel werden keine HTML Links extrahiert
- **-I**: Das Ziel wird auf ein bestimmtes Verzeichnis eingeschränkt
- **-t 5**: Der Request-Timeout wird auf 5 Sekunden reduziert (Standard: 20 Sekunden)

Ergebnisse des Scan-Vorgangs

Der kurze Einblick in die Optionsvielfalt von skipfish sollte einen Eindruck über die Möglichkeiten des Tools vermitteln. Sobald der zuvor angestoßene Scan-Vorgang abgeschlossen ist, wird dies auf der Konsole ausgegeben. Skipfish informiert über die Anzahl der neu gelernten Wörter und wie viele Knoten vom Crawler verfolgt wurden. Alle Ergebnisse werden im Ordner **/home/skipfish/** zusammengefasst.

```
[+] Wordlist 'new_dict.wl' updated (40 new words added).
[+] Copying static resources...
[+] Sorting and annotating crawl nodes: 57
[+] Looking for duplicate entries: 57
[+] Counting unique nodes: 40
[+] Saving pivot data for third-party tools...
[+] Writing scan description...
[+] Writing crawl tree: 57
[+] Generating summary views...
[+] Report saved to '/home/skipfish/index.html'' [0xe87c614e]..
[+] This was a great day for science!
```

Abbildung 4: *Ende des Scan-Vorgangs*

Das Resultat liegt im HTML-Format vor. Die Meldungen werden übersichtlich untereinander dargestellt und können mit einem Mausklick expandiert werden, um detaillierte Informationen abzurufen. Eine Gruppierung des Risikos in Kombination mit allen relevanten Informationen hilft bei der Bewertung von potenziellen Schwachstellen.

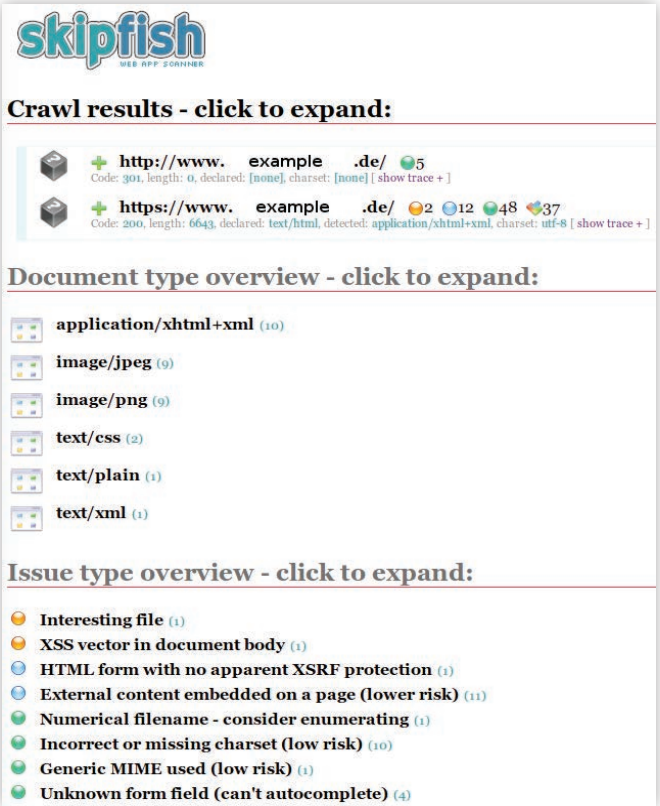


Abbildung 5: *Scan-Ergebnis*

Skipfish gruppiert die Meldungen des Scan-Vorgangs wie folgt:

- High risk flaws – Kann zur Kompromittierung des Systems führen
- Medium risk flaws – Kann zur Veränderung von Informationen / Daten führen
- Low risk issues – Eingeschränkte Auswirkungen
- Internal warnings – Warnungen
- Non-specific informational entries

Skipfish überflutet den Anwender mit vielen Meldungen. Darunter sind grundsätzlich auch etliche Fehlalarme. Hier steht der Tester in der Verantwortung. Bei der Ergebnissichtung müssen tatsächliche Alarme von Falschmeldungen getrennt werden, was bei der Masse von Meldungen mitunter einige Zeit in Anspruch nimmt.

Interessant sind Informationen, die auf externe URL Redirectors hinweisen. Dahinter kann sich fremd manipulierter Quellcode verbergen, der gerne von Spammern eingesetzt wird, um Besucher auf andere Webseiten umzuleiten. URL Redirectors können also ein Indiz für kompromittierten Quellcode darstellen und sollten untersucht werden.

Ein Beispiel:

Abbildung 6: *Detail aus Scan-Ergebnis*

Skipfish weist im Ergebnis auf eine externe Verknüpfung hin, die im Quellcode implementiert wurde. Im Beispiel stellt dies allerdings keine Gefahr dar. Es handelt sich um Piwik, einem Tool zur Webanalyse. Es wird vom Betreiber eingesetzt, um beispielsweise die Besucherzahlen oder Verweildauern auf Seiten zu messen. Skipfish schlägt dennoch Alarm, da sich die Auswertungskomponente von Piwik auf einer anderen Domäne befindet.

Um genau Informationen zu einer potenzielle Schwachstelle zu erhalten, reicht ein Klick auf**show trace** am Ende der Zeile. Mit Hilfe der dargestellten Informationen lässt sich die Position der Schwachstelle im Quelltext meist schnell lokalisieren.

Abbildung 7: *Show Trace*

Fazit

Für den schnellen Scan eignet sich skipfish nicht. Generell benötigt ein Scan etliche Stunden bis ein Ergebnis vorliegt. Im Anschluss wird der Anwender dann mit vielen Meldungen überflutet, die es gilt richtig zu analysieren. Dank der verwendeten Heuristik gekoppelt mit dem selbstlernenden Algorithmus ist die Bedienung grundsätzlich sehr einfach. In Kombination mit diversen Optionen, lässt sich der Scan je nach Ziel und Zweck variieren. Die Darstellung der Ergebnisse im HTML-Format ist gelungen und sorgt für ausreichend Übersicht. Mit der Detail-Ansicht lassen sich Schwachstellen genau lokalisieren und anschließend fixen.

Alleine auf skipfish sollte man sich beim Auffinden von Schwachstellen in Webanwendung allerdings nicht verlassen. Aus reinstrategischenGründenundmitdemHintergedanken,dassskipfish nicht alle Schwachstellen finden kann bzw. darauf ausgelegt ist, sollten immer mehrere Tools kombiniert werden. Dazu eignen sich weitere **Web Vulnerability Scanner**, wie beispielsweise **w3af** oder **Nikito**. Erst eine Kombination aus mehreren Tools und Techniken vervollständigen eine Schwachstellen-Analyse einer Webanwendung. Eine ganze Reihe dieser **Web Vulnerability Scanner** befindet sich auf BackTrack 5. Wer sich also tiefergehender mit dem Thema beschäftigen möchte, findet die entsprechenden Tools auf der Distribution.

MIKE KUKETZ

Der Autor beschäftigt sich seit vielen Jahren mit dem Thema IT-Sicherheit. Er ist Gründer von Kuketz IT-Security und unterstützt Unternehmen bei der Implementierung neuer IT-Sicherheitstechnologien. Durch die enge Zusammenarbeit mit lokalen IT-Systemhäusern in Karlsruhe entsteht daraus eine Komplettbetreuung im IT-Servicebereich.

Weitere Informationen: www.kuketz-security.de | www.kuketz-blog.de

Kontakt mit dem Autor: info@kuketz-security.de

BACKTRACK 5 R2

das Schweizer Armeemesse für IT-Ninjas

PATRICK BLOM

Mittlerweile gibt es einige freie Distributionen im Internet, die sich mit dem Thema IT-Sicherheit, Penetration Testing oder IT-Forensik befassen. Hierzu zählen bekannte Namen wie BackBox Linux, BlackBuntu, DEFT Linux oder auch NetSecL. Jedoch ist keine dieser Distributionen so bekannt und beliebt wie BackTrack Linux.

Das im Jahre 2006 erstmals veröffentlichte Betriebssystem zeichnet sich besonders durch seine sehr große Vielfalt an Programmen sowie eine sehr aktive Community aus. Die auf Debian/Ubuntu basierende Distribution nutzte anfangs ausschließlich das KDE-Environment als grafische Oberfläche. Im Laufe der Entwicklung kamen noch XFCE und Gnome hinzu, wobei XFCE immer noch nicht ganz ausgereift ist und auch nicht direkt als Download zur Verfügung steht. Für welche Oberfläche Sie sich entscheiden, ist den persönlichen Geschmack vorbehalten. BackTrack 5 kommt als Live-DVD einher, kann aber auch problemlos mit nur wenigen Klicks auf die Festplatte installiert werden.

Wenn man sich die Anzahl der mitgelieferten Tools und Scripte in der aktuellen Version BackTrack 5 R2 ansieht, stellt man schnell fest, dass man hier ein “rundum glücklich”-Paket runtergeladen hat, was nahezu alles beinhaltet, um das eigene Netzwerk auf Herz und Nieren zu testen. Bekannte Tools wie NMAP, Wireshark, Metasploit, John the Ripper und AirCrack-ng zählen ebenso zur Grundausstattung, wie der beliebte Neueinsteiger Armitage, mit dem die Bedienung des Metasploit Frameworks noch nie so einfach war. Kurz um von IT-Forensik bis hin zu Information-Gathering ist bei BackTrack 5 R2 alles in einem soliden Paket sauber verpackt und verschnürt worden. Sollte denn noch mal der Fall auftreten, dass ein persönlicher Programm-Liebling fehlt, kann man diesen meistens problemlos mittels APT nachinstallieren.

Eines der wohl meist genutzten Programme unter BackTrack ist die AirCrack-ng Suite. Diese Zusammenstellung von Programmen bietet verschiedene Möglichkeiten, ein W-LAN-Netz auf Sicherheitslücken zu testen. Hierzu zählen z.B. das Cracken von W-LAN Verschlüsselungen, Fake-Authentications oder das Aufsetzen eines Honypots. Um Ihnen einen Einblick in BackTrack 5 R2 und die AirCrack-ng Suite zu geben, möchte ich Ihnen Schritt für Schritt zeigen, wie Sie mit Hilfe der AirCrack-ng Suite ein mit der WEP-Verschlüsselung geschütztes W-Lan ausfindig machen und den für die Authentifizierung am Netzwerk benötigten W-LAN Schlüssel knacken.

Vorab sei gesagt:

Dieser Artikel soll kein Leitfaden für Skriptkiddis darstellen, die sich einen Spaß daraus machen, anderer Leute W-LAN zu “hacken” und nicht wissen, was sie da überhaupt machen. Ich möchte mit diesem Artikel zeigen, wie einfach es ist, ein mit WEP verschlüsseltes W-LAN zu knacken und somit auf die gravierenden Sicherheitslücken in dieser Verschlüsselung hinweisen. Ich rate jedem, der noch ein W-LAN mit WEP-Verschlüsselung benutzt, auf die weitaus sichere WPA/WPA2 Verschlüsselung umzusteigen. Des Weiteren weise ich Sie drauf hin, dass das Eindringen in fremde Netzwerke den Sachverhalt einer Straftat darstellt und nicht als Kavaliersdelikt abgetan wird.

In meinem Artikel verwende ich zu Demonstrationszwecken mein persönliches Heim-W-Lan, welches ich für diesen Artikel auf die WEP-Verschlüsselung umgestellt habe.

Vorbereitung

Beginnen wir mit unserem Beispiel und machen uns ein paar Gedanken zu unserem Vorhaben und was wir dafür brauchen. Als Erstes benötigen wir eine aktuelle Version von BackTrack 5R2. Diese gibt es auf *http://www.backtrack-linux.org/downloads/*. Wer möchte, kann sich vor dem Download in der BackTrack Community registrieren, dies ist aber nicht zwingend erforderlich. Als Release wählen wir BackTrack5 R2; die anderen Auswahlmöglichkeiten können nach belieben angewählt werden. Ich entscheide mich in diesem Fall für die 32-Bit-Architektur und Gnome als Windows Manager. Da mir zurzeit kein Torrentprogramm zur Verfügung steht, entscheide ich mich für den “Direct”-Download.

Während unser Download läuft, gönnen wir uns einen kurzen Einblick in die WEP-Verschlüsselung.

WEP steht für Wired Equivalent Privacy und ist der Vorgänger der WPA/WPA2 Verschlüsselung. Das Prinzip von WEP ist recht simple zu beschreiben, die Daten, die wir übertragen möchten, werden mit einem generierten Keystream exklusiv oder verknüpft und dann übertragen. Die Gegenstelle führt dann dieselbe Aktion für die Entschlüsselung der Daten durch.

Dieser sogenannte Keystream ist der für uns interessante Teil, er besteht aus unserem WEP-Passwort, welches wir im W-LAN Router festlegen und einem 24-Bit langen Initialisierungsvektor (kurz IV). Dieser IV wird für jedes Datenpaket neu generiert und bildet in Verbindung mit unserem WEP-Passwort die Grundlage für unsren Keystream. Aus dieser Verbindung wird dann mithilfe des RC4-Algorithmus der eigentliche Keystream generiert.

Um nun nicht zu tief ins Detail zu gehen und den Rahmen dieses Artikels nicht zu sprengen sei gesagt, wichtig sind die Initialisierungsvektoren des Keystreams. Wenn wir genug von diesen IV's mitlesen, kann Aircrack-ng das WEP-Passwort in einer relativ kurzen Zeit entschlüsseln.

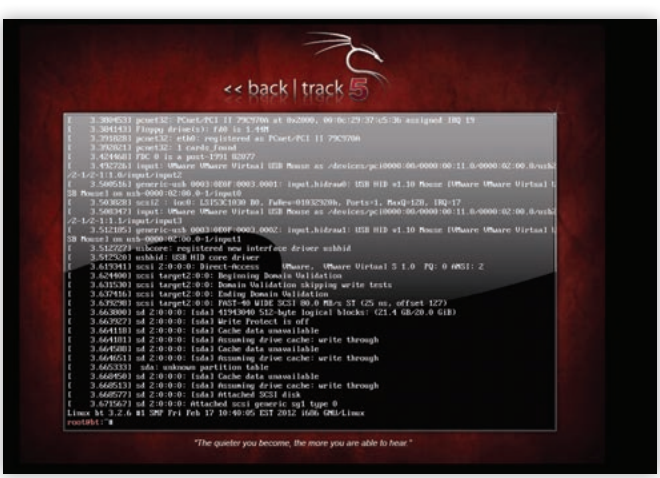
Falls Sie gerne mehr über die WEP und den RC4-Algorithmus erfahren möchten, empfehle ich Ihnen folgende Artikel: http://de.wikipedia.org/wiki/Wired_Equivalent_Privacy und http://de.wikipedia.org/wiki/RC4. Ebenfalls sehr hilfreich zu diesem Thema ist der "Wireless LAN Security and Penetration Testing Megaprimer" auf http://www.securitytube.net/ Vivek Ramachandran hat sich diesem Thema sehr detailliert gewidmet und meiner Ansicht nach hervorragende Arbeit geleistet.

Und los geht's

Unser Download sollte mittlerweile abgeschlossen sein. Also nichts wie los, noch schnell das Image auf DVD brennen und davon booten, schon sind wir mitten drin.

Nach dem Bootvorgang von der DVD begrüßt uns BackTrack5 R2 mit folgendem Startbildschirm.

Hier haben wir nun die Qual der Wahl. In welchem Modus möchten wir BackTrack starten? Ich empfehle für den Anfang erst mal den Default Modus "BackTrack Text" - in diesem Modus wird alles Wichtige mit gestartet und wir haben den vollen Umfang der Distribution verfügbar. Die anderen Optionen wie Stealth oder Forensics sind für spezielle Anwendungsgebiete gedacht, da gewissen Funktionen deaktiviert sind. Z.B. wird im Forensics-Mode kein Swap-Laufwerk (falls vorhanden) gemountet, um keine Daten zu verändern, die das Ergebnis einer forensischen Untersuchung verfälschen könnten.



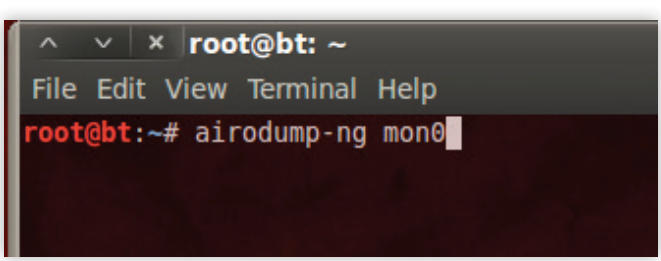
Nachdem wir unseren Startmodus gewählt haben, landen wir auch schon direkt im eigentlichen Betriebssystem.

Da sämtliche Anwendungen in Backtrack von der Konsole aus zu bedienen sind, ist es nicht nötig, einen X-Server zu starten. Dies belastet zum einen nicht so stark die Ressourcen des Systems und ist zum anderen in vielerlei Fällen deutlich schneller, als sich mit der Maus irgendwo durchzuklicken. Um unseren Komfort jedoch etwas zu steigern, starten wir den X-Server mit folgendem Befehl:

startx

Tipp: Sollte Ihnen der X-Server abstürzen nicht gleich verzweifeln, meistens genügt es sich die Fehlermeldung zu notieren und einen kurzen Blick in das BackTrack Forum zu werfen. Dort wurden bereits viele Probleme deutlich beschrieben und durch die Community gelöst.

Nachdem wir nun den X-Server gestartet haben, erwartet uns ein recht übersichtlicher Desktop. Sämtliche Annehmlichkeiten, die man von Gnome gewohnt ist, stehen einem zur Verfügung. Die interessanten Programme sind schnell über den "Applications"-Button unter dem Menüpunkt BackTrack zu finden. Hier zeigt sich eine ordentliche Strukturierung der einzelnen Applikationen nach Gruppen sowie Einsatzgebiete.



Am besten gehen Sie einfach mal auf Entdeckungsreise durch diesen Menüpunkt und lassen sich überraschen, auf was Sie alles stoßen werden. Sie werden feststellen, dass ich, was den Umfang der Programme und Scripte angeht, nicht zu viel versprochen habe.

Die AirCrack-ng Suite

Wie schon beschrieben ist die AirCrack-ng Suite wohl eins der beliebtesten Programme unter BackTrack, und gehört auch schon seit geraumer Zeit zu der Grundausstattung des Systems.

Um unserer Ziel-W-Lan zu Cracken, werden wir einige Programme aus dieser Suite verwenden. Hierzu zählen airmon-ng, airodump-ng, aireplay-ng und aircrack-ng. Wie Sie diese Programme benutzen und welches wir wofür verwenden werden, beschreiben wir in den einzelnen Etappen des Cracking-Vorgangs.

Damit wir einen strukturierten Ablauf haben und ggf. einen Schritt zurück gehen können, unterteilen wir unsere Vorgehensweise in folgende Schritte:

- Unseren W-LAN Adapter vorbereiten.
- Das Ziel-W-LAN lokalisieren und, dessen Traffic sniffen.
- Das Ziel-W-LAN provozieren, Initialisierungsvektoren zu erzeugen.
- Das eigentliche Cracken des Passworts

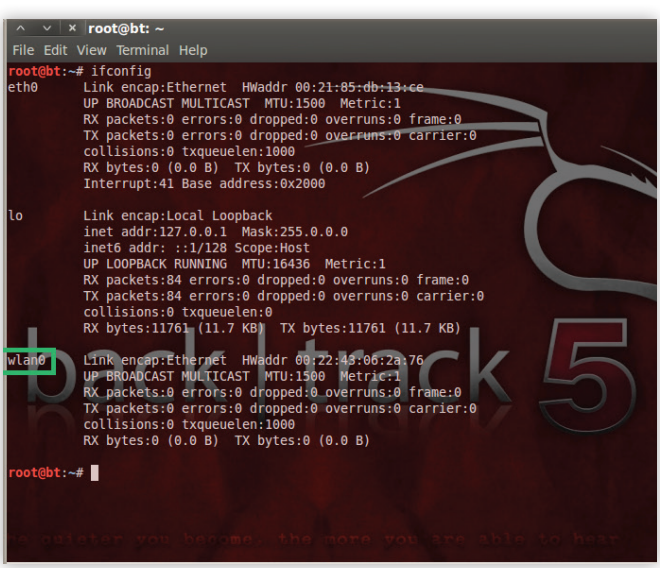
Nachdem wir uns eine klare Vorgehensweise zurecht gelegt haben, können wir nun damit beginnen, sie abzuarbeiten. Also nichts wie ran an Schritt 1.

Unseren W-LAN Adapter vorbereiten

Damit wir sämtliche Pakete, die über ein W-LAN übertragen werden, empfangen und mitschneiden können, versetzten wir unseren W-LAN Adapater in den promiskuos Mode (“freizügiger Modus”). In diesem Modus startet unser W-LAN Adapter quasi den Großen Lauschangriff und nimmt sämtliche Pakete entgegen, die ihn erreichen.

Beginnen wir also damit, das wir erst mal herausfinden, wie unser W-LAN Adapter von BackTrack erkannt wurde. Dies erreichen wir, indem wir einen Terminal öffnen und den Befehl ifconfig eingeben.

Dieser Befehl sorgt dafür, dass uns die aktuelle Netzwerkkonfiguration angezeigt wird.



Wie hier zu sehen ist, hat unser System den W-LAN-Adapter richtig erkannt und dem Verweis wlan0 zugeordnet. Nun können wir den Adapter in den promiskuos Mode versetzen. Dies machen wir mit dem Befehl.

airmon-ng start wlan0

Wird dieser Befehl erfolgreich ausgeführt, bestätigt uns airmon-ng, dass unser Adapter sich nun im promiskuos Mode befindet, mit der Ausgabe (Monitor Mode enabled on mon0) hier grün markiert. Von jetzt an werden wir nur noch mit dem Adapter mon0 arbeiten, den Adapter wlan0 brauchen wir für die folgenden Schritte nicht mehr.

Zeitgleich zum Erzeugen des Adapters mon0 haben wir auch den ersten Schritt auf unserer Liste abgearbeitet. Machen wir also nun weiter mit Schritt2.

Das Ziel-W-LAN lokalisieren und dessen Traffic sniffen

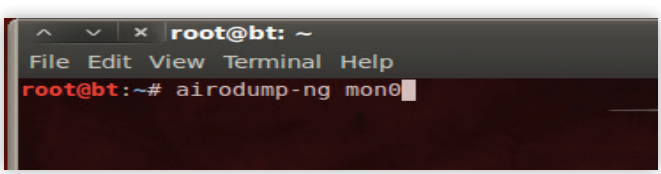
Wie wir im Vorhinein festgestellt haben, benötigen wir zum Entschlüsseln des W-LAN Passworts sogenannte Initialisierungsvektoren (IV’s). Diese müssen von unserem Ziel-W-LAN erzeugt werden, sodass wir diese mitschneiden können.

Diese IV’s befinden sich nahezu in jedem Datenpaket, welches von unserem Ziel-W-LAN ausgesendet wird, sprich wenn Daten auf dem Netzwerk hin und her geschickt werden, können wir diese einfach mitschneiden. Sollte auf dem Netzwerk kein Datenverkehr herrschen, können wir diesen provozieren. Diesem Schritt werden wir uns aber etwas später widmen.

Wir gehen davon aus, dass 1 Datenpaket = 1 Initialisierungsvektor enthält. Für eine erfolgreiche Entschlüsselung gilt je mehr IV’s desto besser. Der Grenzwert für einen 128-Bit-WEP Key liegt laut AirCrack-ng bei ca. 40.000 IV’s, das heißt also für uns das wir ca. 40.000 Datenpakete mitschneiden müssen. Dies klingt nun erst mal nach einer Menge Daten, ist aber bei einer halbwegs guten Empfangsrate in nur wenigen Minuten erledigt.

Finden wir also heraus, ob unser Ziel-W-LAN in Reichweite ist und ob auf ihm Daten versendet werden. Dies bewältigen wir mit dem Befehl

airodump-ng mon0



Nun wird uns ein Livescan der aktuellen W-LANs in unserer Reichweite angezeigt. Diese W-LANs bzw. Accesspoints werden in diesem Moment von unserem mon0 Adapter empfangen. Wir lassen diesen Scan nun ein paar Sekunden laufen und brechen ihn dann mit der Tastenkombination Strg+C ab.

Wir haben uns soeben einmal kurz umgesehen, was sich so alles in unserer W-LAN Umgebung tut. Interessant ist für uns die hier grün markierte Zeile. Dies ist unser Ziel-W-LAN mit dem Namen “hakin9” und soll von uns gecrackt werden. Bevor wir uns aber dem widmen, hier noch eine kurze Erklärung zu der Anzeige, die wir hier sehen.

An der ersten Stelle wird uns die BSSID unseres Ziel-W-LANs angezeigt. Das ist die Hardwareadresse unseres Accesspoints und quasi das Äquivalent zu der MAC-Adresse in einer normalen Netzwerkkarte.

Die nächste Spalte PWR zeigt uns die Signalstärke des Accessponits in dbm an. Die genaue Berechnung der Signalstärke ist recht kompliziert, daher eine Kurzanleitung wie diese werte zu interpretieren sind. Die Signalstärken Skala in airmon-ng geht von -0 bis -100. Das Ziel ist es nun so nah wie möglich an den Mittelwert -50 zu kommen. In unserem Beispiel haben wir eine Signalstärke von -48 was nahe zu an eine perfekte Signalstärke ran kommt. Somit haben wir fast perfekte Gegebenheiten für unseren Versuchsaufbau. Meine Tests haben gezeigt, dass sich mit einer Signalstärke bis -72 noch recht akzeptabel arbeiten lässt. Bei höheren Werten ist der Paketverlust einfach zu hoch.

In der Spalte Beacons wird uns die Anzahl der sogenannten Beacon Frames, die der Accesspoint versendet hat, angezeigt. Ein Beacon Frame kann man sich am besten wie eine Visitenkarte des Accesspoints vorstellen. Diese Datenpakete werden in regelmäßigen Abständen von dem Accesspoint ausgesendet und enthalten alle Informationen, die für die erste Phase eines Verbindungsaufbaus benötigt werden.

Die Spalten #Data und #/S zeigen die gesamte Anzahl der mitgeschnittenen Pakte und den Schnitt der letzten 10 Sekunden an. Das Feld #Data ist in unserem Fall besonders interessant, da wir hier dran ablesen können, wann wir die Grenze von 40.000 Pakten/ IV’s überschritten haben und das Mitschneiden beenden können.

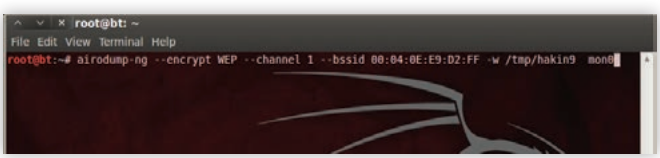
Anhand der Spalte CH können wir ablesen, auf welchem Kanal unser Ziel-W-LAN sendet. In unserem Fall sendet unser Accesspoint auf Kanal 1.

MB ist die Kurzform für Mega-Bit und zeigt uns die maximale Übertragungsrate des Accesspoints an. Da wir hier für unseren Aufbau eine FritzBox 7170 verwenden haben wie folglich eine maximale Übertragungsrate von 54 Mega-Bit.

Die Spalten ENC CIPHER und AUTH geben uns Informationen über die Verschlüsselung des Accesspoints. In unserer markierten Ziele sehen wir, dass unser Ziel-W-LAN mit der WEP-Verschlüsselung arbeitet.

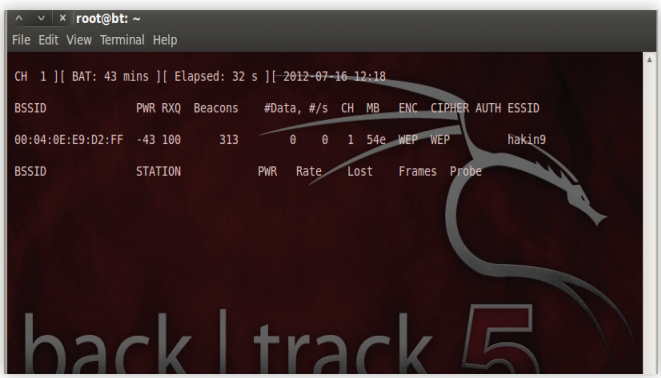
In der letzten Spalte wird uns schließlich noch die ESSID des Accesspoints angezeigt. Für diesen Testaufbau habe ich unserem Ziel-W-LAN den Namen “hakin9” gegeben.

Da wir nun unser Ziel-W-LAN lokalisiert haben, können wir airodump-ng erneut starten. Dieses Mal spezialisieren unseren Befehl etwas und sorgen auch gleichzeitig dafür, dass unsere mitgeschnittenen Pakete in eine Datei geschrieben werden.



Wir teilen so airodump-ng mit nur Accesspoints mit der Verschlüsselung (--encrypt) WEP, auf dem Kanal (--channel) 1 und mit der BSSID (--bssid) 00:04:0E:E9:D2:FF anzuzeigen. Diese Daten konnten wir aus unserem vorherigen Aufruf von airodump-ng entnehmen. Zusätzlich teilen wir airodump-ng mit, dass alle Pakete, die es mit diesen Bedingungen empfängt, in das Verzeichnis /tmp/ mit den Dateinamen hakin9 schreiben soll. Hierfür sorgt der Parameter -w . Als letzten Parameter übergeben wir airodump-ng unseren Monitorring Adapter mon0.

Wenn wir diesen Befehl ausführen, sehen wir folgende Ausgabe von airodump-ng.



Wir stellen fest, dass durch unsere Spezifizierung nun nur noch das W-LAN mit der ESSID “hakin9” in airodump-ng zu sehen ist. Nun öffnen wir ein neues Terminal, widmen uns dem nächsten Schritt und schieben unsere geöffnetes airodump-ng beiseite.

Das Ziel-W-LAN provozieren Initialisierungsvektoren zu erzeugen

Wie wir in unserer vorherigen Abbildung gesehen haben, steht die Paketzählende Zeile #Data auf 0. Dies ist ein klares Zeichen, dass auf diesem Netzwerk keinerlei Daten versendet werden. Somit können wir auch keine IV’s mitschneiden.

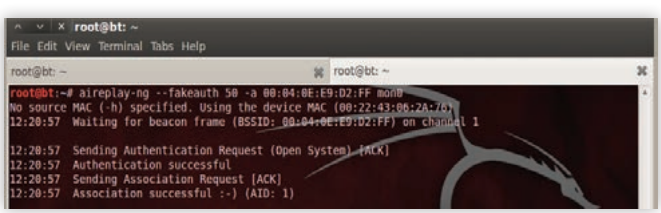
Aus diesem Grund müssen wir das Ziel-W-LAN dazu provozieren IV’s zu generieren. Dies erreichen wir mit einer sogenannten ARP-Replay Attacke. Diese Attacke basiert auf dem Address Resolution Protocol und bewegt den Accesspoint dazu immer wieder ein neues Datenpaket inkl. IV’s zu senden.

Bevor wir allerdings diese Attacke anwenden können, müssen wir herausfinden, ob wir diese Attacke bei unserem Accesspoint überhaupt funktioniert. Dies können wir mit einer Fake-Authentifizierung an unserem Accesspoint testen. Bei dieser Fake-Authentifizierung wird getestet, ob unser Accesspoint für sogenannte “Paket Injections“ anfällig ist. Sollte dies der Fall sein, können wir die ARP-Replay Attacke problemlos anwenden. Die Fake-Authentifizierung und die ARP-Replay Attacke werden wir mit dem Programm aireplay-ng durchführen.

Beginnen wir mit der Fake-Authentifizierung. Hierzu verwenden wir folgenden Befehl in unserem neu geöffneten Terminal.

```
aireplay-ng --fakeauth 50 -a 00:04:0E:E9:D2:FF mon0
```

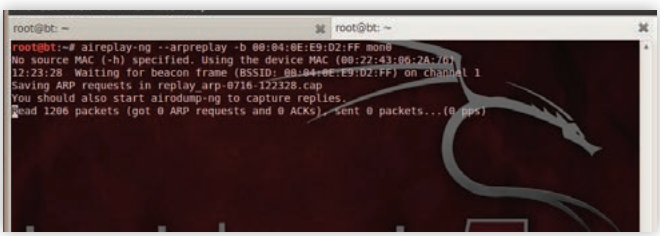
Wir teilen so aireplay-ng mit, die Fake-Authentizifierung in einem Intervall von 50ms (--fakeauth 50) auf den Accesspoint mit der BSSID 00:04:0E:E9:D2:FF (-a 00:04:0E:E9:D2:FF) über den Adapter mon0 durchzuführen. Im Erfolgsfall sollten wir folgendes Resultat von aireplay-ng angezeigt bekommen:



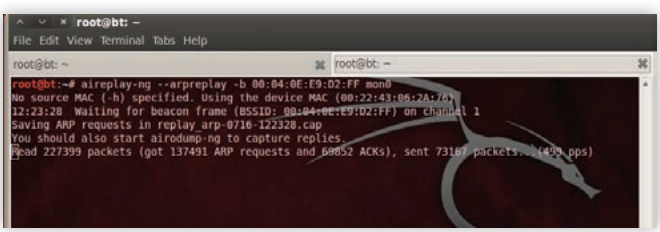
Wir wissen nun, dass wir die ARP-Replay Attacke anwenden können. Wir starten die Attacke mit folgendem Befehl:

```
aireplay --arpreplay -b 00:04:0E:E9:D2:FF mon0
```

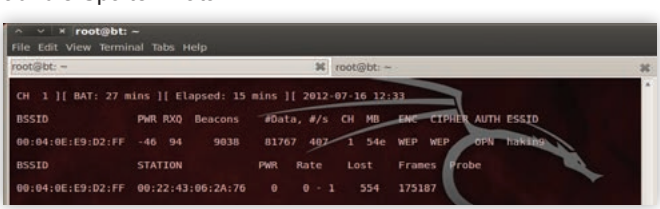
So führt aireplay-ng nun die ARP-Replay Attacke (-- arpreplay) gegen die BSSID 00:04:0E:E9:D2:FF (-b 00:04:0E:E9:D2:FF) über den Adapter mon0 aus.



Nach kurzer Zeit wird sichtbar, dass unsere Attacke funktioniert. Die Anzahl der von aireplay-ng gesendeten Pakete steigt rapide an.



Während diese Attacke läuft, betrachten wir einmal unser zuvor beiseitegeschobenes Fenster mit airodump-ng und achten auf die Spalte #Data.



Wie wir sehen, steigt die Anzahl unserer mit geschnittenen Pakete sehr schnell stark an. Nach nur 15 Minuten wurden bereits über 80.000 Pakete mitgeschnitten. Da wir somit die Grenze von 40.000 Paketen überschritten haben, können wir die ARP-Replay Attacke sowie airodump-ng mit Strg+C beenden und den letzten Schritt in unserer Reihenfolge abarbeiten.

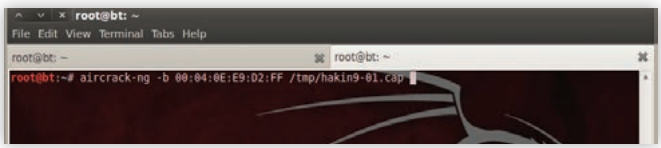
Das eigentliche Cracken des Passworts

Jetzt sind wir nur noch einen Katzensprung von unserem Ziel entfernt. Das Cracken des Passwortes ist mit einem recht kurzen Befehl erledigt.

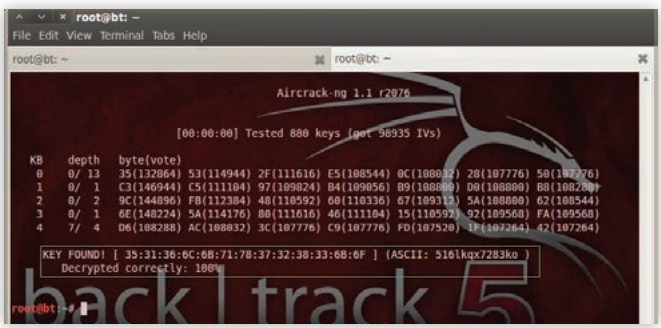
```
aircrack-ng  -b 00:04:0E:E9:D2:FF /tmp/hakin9-01.cap
```

Wir verwenden nun das eigentliche Programm aircrack-ng, um das Passwort zu cracken. Ähnlich wie bei aireplay-ng gibt der Parameter (-b) die BSSID des Accesspoints, an dessen Passwort wir cracken möchten. Wir müssen aircrack-ng dann lediglich noch die mitgeschnitten Daten von airodump-ng übergeben, welche wir unter /tmp/hakin9 abgelegt haben.

Tipp: Da airodump-ng den Namen der Datei mit den mitgeschnittenen Daten selbst vervollständigt hat, nutzen Sie die Tab-Taste, um den genauen Name der .cap Datei zu bekommen.



Nachdem wir diesen Befehl ausgeführt haben, beginnt aircrack-ng mit dem cracking-Prozess. Im Erfolgsfall sollte Ihnen aircrack-ng folgendes Ergebnis anzeigen:



In diesem Fall war es für AirCrack-ng keine sonderlich schwierige Aufgabe, das WEP-Passwort zur cracken. Wir können nun entweder den Hashkey oder das Klartext Passwort in einem beliebigen Networkmanager verwenden und uns problemlos mit unserem Ziel-W-LAN verbinden.

Sollte cracking-Prozess dennoch fehlschlagen und das Passwort nicht entschlüsselt werden, wiederholen Sie den Mitschneide-Abschnitt und versuchen Sie noch mehr Datenpakete mitzuschneiden.

Fazit

Wie wir gesehen haben, ist es mit Hilfe von BackTrack5 R2 nicht sonderlich schwierig, ein mit WEP verschlüsseltes W-LAN zu cracken. Der ganze Vorgang hat in dem von mir aufgebauten Testszenario keine 30 Minuten gedauert. Dies verdeutlicht sehr gut, wie unsicher die WEP-Verschlüsselung ist und welche Möglichkeiten BackTrack5 bieten kann. Natürlich ist AirCrack-ng nicht das einzige Feature von BackTrack5, aber es zeigt sehr gut, wie leicht die Bedienung des Systems ist und erklärt, warum es von vielen professionellen Pentestern so geschätzt wird.

PATRICK BLOM

ist leitender Anwendungsentwickler für einen Großhandel in NRW. Zuvor war er in einigen Internet-Agenturen beschäftigt und entwickelte dort hauptsächlich in den Bereichen eCommerce und Digital Signage. Seit über 10 Jahren beschäftigt sich Patrick Blom mit auf Unix basierenden Bestriebssystemen, und ist ständig auf der Suche nach ihm unbekannten Distributionen, um diese zu testen und deren Besonderheiten zu untersuchen.
Kontakt: info@bl0m.de oder über www.bl0m.de

BACKTRACK 5

WLAN MIT WPA/2-VERSCHLÜSSELUNG KNACKEN

TYSONPOWER

In diesem Artikel geht es um das unter IT-Fachleuten relativ bekannte Backtrack 5 und wie es mit dem bereits beinhalteten „airmon-ng" Tool-Kit zum Knacken von WLAN Netzwerken gebraucht werden kann.

Backtrack ist ein Betriebsystem, das auf dem bekannten Ubuntu Linux bzw. in der neuesten Variante 5 R2 auf Ubuntu Lucid (Kernel 2.6.38) basiert und viele Tools für „Hacker" bzw. IT-Fachleute beinhaltet.

Da es auf Ubuntu basiert und die Entwickler von Backtrack alles „for free" machen, ist Backtrack komplett kostenlos downloadbar (Link unten).

Außerdem kann man problemlos Backtrack auch von einem USB Stick laufenlassen ohne Installation.

Z.B. kann man mit Hilfe bestimmter Tools aus Backtrack 5 Server angreifen und lahmlegen, Passwörter klauen bzw. - wie es unter Fachleuten heißt - „sniffen" und noch vieles vieles mehr.

Da Backtrack so viele Tools enthält zeige ich nur ein Tool, das sehr bekannt ist und auch sehr ausgereift.

Das besagte Tool heißt airmon-ng und ist eine Terminal-Anwendung, mit der WLAN-Passwörter herausgefunden werden können, um so in andere WLAN-Netzwerke einzudringen.

Mit airmon-ng kann man WEP sowie WPA / WPA2 Verschlüsselungen knacken.

Da das Ausspähen eines WLAN-Passwortes natürlich gegen das „Hacker Gesetz (§ 202c StGB)" in Deutschland verstößt, sollte man es nur an seinem eigenen Netzwerk testen.

In diesem Tutorial benutze ich Backtrack 5 R2.

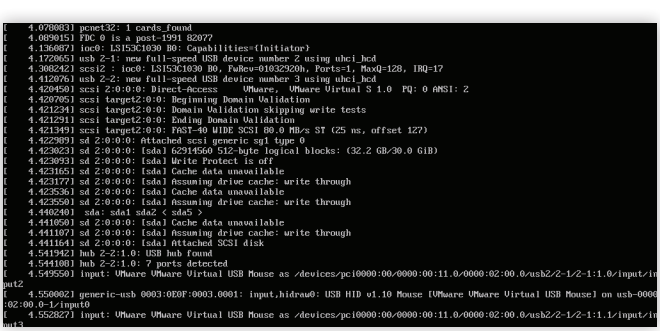
Schritt 1

Zuallererst muss man natürlich Backtrack erstmal starten und sich einloggen.

Am besten loggt man sich als „root" ein, da man so Administrator ist und über alle Rechte verfügt.

Standardmäßig ist der Login „root" und das Passwort „toor".

Zum Starten des grafischen Interfaces einfach „startx" eingeben.

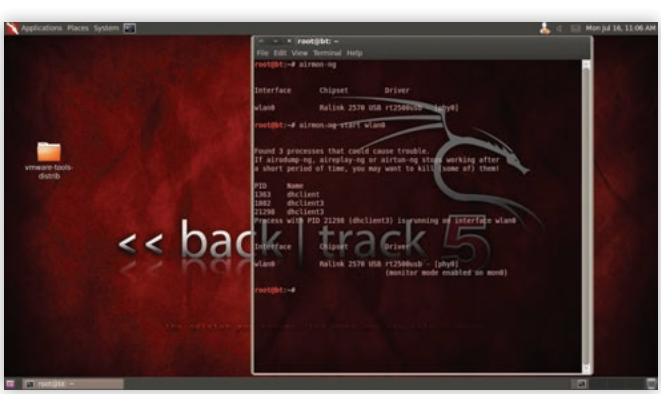


Schritt 2

Nun öffnen wir über das Menü den Terminal über „Applications → Accessories → Terminal".

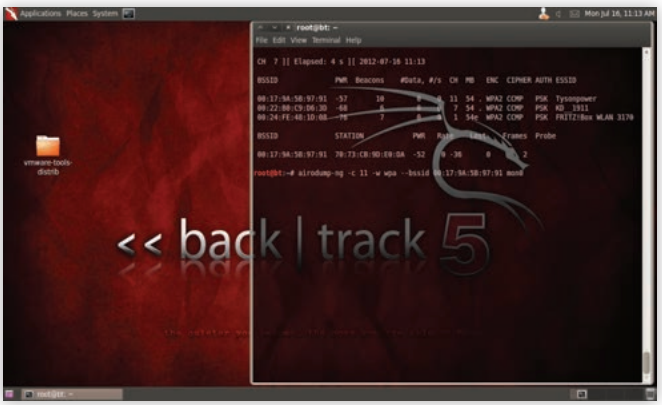
Schritt 3

Erstmal müssen wir unseren WLAN-Adapter für airmon-ng kompatibel machen. Dies geht mit den Commands „airmon-ng" und „airmon-ng start". Zuerst suchen wir nach WLAN Geräten mit „airmon-ng". Danach starten wir 1 WLAN gerät mit „airmon-ng start WLANGERÄT". Anstatt WLANGERÄT schreiben wir wlan0 oder wenn ihr mehrere WLAN-Geräte habt eben das WLAN, das ihr haben möchtet. Deshalb benutzen wir ab jetzt nur noch mon0 als Interface.



Schritt 4

Jetzt suchen wir nach WLAN-Netzwerken in der Umgebung mit dem Befehl „airodump-ng mon0" und beenden nach 1-2 Minuten die Suche mit Strg + c .



Danach suchen wir uns ein von den gefundenen Netzwerken aus, das wir angreifen wollen.

Wenn wir uns für ein entschieden haben, scannen wir nur noch dieses ein Netzwerk, um die Suche genauer zu machen mit dem Befehl *airodump-ng -c CH -w wpa --bssid BSSID mon0*.

„CH“ ersetzen wir mit dem Channel des Netzwerkes und „BSSID“ mit der Bssid des Netzwerkes.

Die Bssid ist die MAC-Adresse des Routers und wird ab jetzt in fast jedem weiterem Schritt benötigt.

Die Option -w gibt nur an, wie die Datei später heißen soll, deshalb muss man „wpa“ nicht ändern, kann man aber.

Schritt 5

Nun reconnecten wir alle verbundene Geräte um einen Handshake zum Kriegen.

Dies machen wir mit dem Command „aireplay-ng -0 5 -a BSSID mon0 “ in einem neuen Terminal.

Ein Handshake ist unter anderem das WLAN-Passwort in verschlüsselter Form.

Mit Hilfe von der Handshake kann man später das Passwort herausfinden.

Wenn nach diesem Command ein Fehler kommt, muss man diesen Command benutzen: „aireplay-ng -0 1 -a (BSSID) -c (Client Mac)“, da dann der Router eine Sperre hat.

Die Client MAC ist einfach die MAC des Clients, der natürlich vorhanden sein muss, um die Handshake zu bekommen.

Schritt 6

Wenn wir jetzt unsere Handshake haben, müssen wir noch das Passwort knacken, das in verschlüsselter Form in der Handshake ist.

Dies kann man auf viele Arten machen bzw. versuchen.

Die hier gezeigte Art ist die einfachste, aber auch nicht die erfolgreichste.

Dafür nehmen wir den Command „aircrack-ng (filename) -w (Pfad zur Wordlist)“.

(filename) ersetzen wir jetzt mit dem Namen der .cap-Datei - in diesem Fall müsste sie wpa-01.cap heißen.

(Pfad zur Wordlist) ist der Pfad der Wordlist, in dem hoffentlich das Passwort steht.

Um sich das Schreiben zu ersparen, kann man auch einfach die Wordlistdatei auf den Terminal ziehen, um den genauen Pfad einzufügen.

Da das Passwort in der Wordlist stehen MUSS, ist diese Art natürlich nicht immer erfolgreich.

Was 100% zum Erfolg führt, aber ewig dauern kann, ist das Cracken mit Hilfe von dem Wordlistgenerator Crunch, der ebenfalls enthalten in Backtrack ist.

Auf Crunch wollen wir aber hier nicht weiter eingehen.

Dort, wo jetzt der rote Balken auf dem Bild ist, steht das Passwort. Zur Sicherheit habe ich es weggestrichen.

FRAGEN:

- Warum findet Backtrack keine WLAN Geräte, obwohl ich eins angeschlossen habe?
- Dies kann daran liegen, dass es schon von etwas anderen benutzt wird. Dieses Problem kann man lösen mit den Befehlen: „airmon-ng stop wlan0“
- „ifconfig wlan0 down“
- Wie vergleicht airmon-ng die Passwörter?
- ganz einfach: Es wandelt das Wort in der Liste um in einen Hash und vergleicht ihn mit dem Hash in der Handshake.

LINKS:

Backtrack 5 R2 Download: http://backtrack-linux.org
Cracken mit Crunch (Bruteforce): http://www.youtube.com/watch?v=tLW_0gKYSIc

KONTAKT MIT DEM AUTOR:

Youtube: http://youtube.com/tysonpower2010
Twitter: http://twitter.com/Tysonpower
E-mail: help@tysonpower.de.vu
Website: http://tysonpower.de.vu

SchagIT.at

it-consulting | software engineering | web

IT Lösungen mit Fokus auf IT-Security

Egal ob Netzwerke, Datenbanken, Webanwendungen oder kundenspezifische Softwareprodukte, bei SchagIT wird IT-Security hoch priorisiert. Wir bieten Ihnen auch Sicherheitsüberprüfungen von bestehenden Lösungen an.

Jetzt NEU!

DocIT

Tool zur Dokumentation Ihrer IT-Komponenten, um Tätigkeiten und Änderungen in einer IT Umgebung zu erfassen und zu protokollieren. Dadurch ist man bei einem System- bzw. Serviceausfall in der Lage, systematisch an das Problem herangehen zu können. Weiters behält man den Überblick über die Vorgänge aller Komponenten.

+ Vollständig in eine Windows Domäne integrierbar
+ Schnittstelle zu vorhandenen Monitoring Systemen (Splunk, Nagios, ...)
+ Keine Clientinstallationen notwendig

SchagIT, Inh. Martin Schagerl | Holzing 52, 3252 Bergland, AUSTRIA
http://www.schagit.at | office@schagit.at | +43 676 / 7119622