4.2 Webserver-Log-Forensik

Prof. Dr. rer. nat. Daniel Hammer

Abstract
In recent years, PHP-based malware is gaining popularity amongst attackers on the Internet. Such hacking scripts allow an attacker to hijack and control web servers that run the Hypertext Preprocessor scripting language in order to maintain dynamic web pages. We introduce a logfile parser that finds traces of such malware on infected systems, analyses and reports the actions of the attacker.

Logdatei-Forensik
Für den professionellen Einsatz von Computersystemen müssen die darauf stattfindenden Ereignisse – ähnlich wie bei einem Flugschreiber in Flugzeugen – geeignet protokolliert werden. Die gesammelten Informationen werden in Form von Ereignisprotokollen in Logdateien gespeichert.

Im Rahmen einer effizienten Administration von IT-Systemen und -Netzwerken, bei der Prozess-Kontrolle bzw. -Synchronisation und bei Optimierung und Automatisierung finden Logdateien eine wichtige Anwendung. Klassisch und besonders transparent ist der Einsatz von Logdateien auf Webservern, wo durch dieser Ereignisprotokolle Statistiken zur Nutzung und Optimierung von Internetauftritten oder zu Analysen von Benutzerfreundlichkeit oder dem Erfolg von Marketing-Kampagnen erstellt werden.

Vor allem dann, wenn es zu Fehlverhalten von IT-Anlagen kommt, dienen Einträge in Systemlogbüchern der Fehleranalyse und Ursachenbekämpfung.

Sieben kriminalistische Ws
Nach dem Versuch oder der erfolgreichen Ausführung eines Hackerangriffs auf IT-Systeme können Logdateien von zentraler Bedeutung sein, um solche Vorfälle zu entdecken und aufzuklären. Als Teil einer computer-forensischen Analyse werden dabei Aufzeichnungen der Logdateien eines betreffenden Zeitraums nach bestimmten Kriterien untersucht, Anomalien festgestellt und Rückschlüsse auf die Angriffsszenarien gezogen. Um ein möglichst umfassendes Bild der Vorgänge zu bekommen, werden oft die Aufzeichnungen verschiedener Logdateien über vorhandene Zeitstempel miteinander korreliert. Dies geschieht unter anderem, wenn die Logdateien von Webservern oder Firewalls mit denen von Routern in einer zusammenfassenden Analyse abgeglichen werden.

Wie bei allen digitalen forensischen Untersuchungen werden die einzelnen fallrelevanten Daten und Fakten jeweils für sich bewertet und in den zentralen Analysefeldern mit anderen Fakten in Beziehung gesetzt. Diese zentrale Analysefelder Täter, Tatort, Tatzeit, Opfer bzw. angestrebtes Gut, Modus Operandi, Tatmittel und Motiv manifestieren sich in den sogenannten „Sieben kriminalistischen Ws“, d. h. den Fragen: ~Wer? ~Wo? ~Wann? ~Wen/was? ~Wie? ~Womit? ~und ~Warum?

Gerade weil Logdateien wichtige Informationen über den Tathergang eines Angriffs enthalten können, bemühen sich manche Angreifer, ihre Spuren zu verwischen und Logdateien zu bereinigen oder gar ganz zu löschen.

Sind Logdateien allerdings vorhanden und nicht manipuliert worden, so sind deren Einträge oft nicht einfach zu lesen. Gerade wenn auf einem kompromittierten System sich Spuren einer Schadsoftware in den Logdateien finden, müssen diese erst extrahiert, mit den Gegebenheiten des Systems korreliert und interpretiert werden.

PHP-basierte Schadsoftware
Im Zug der IT-forensischen Ermittlungen nach Einbrüchen in eines der größten deutschen Internetportale wurde im Labor für IT-Sicherheit und Computer Forensik der Hochschule Offenburg ein Forschungsprojekt gestartet, das sich mit der Analyse von Schadsoftwarespuren in Logfiles beschäftigt. Ein im Zug dieser Forschungsarbeit entstandenes Programm, der „Analyzer of Death“, analysiert und interpretiert Spuren, die PHP-basierte Backdoor-Programme in den Webserver-Logfiles hinterlassen.

PHP-(Hypertext Preprocessor-) basierte Backdoor-Programme werden entworfen, um einem Angreifer die Kontrolle über einen infizierten Computer zu geben. Häufig bei Angriffen benutzte Skripte, die in diese Kategorie fallen sind „destroyer“, „C99Shell“, „c99ud“, „r57“, „safe0ver“, „bypass“ oder „zehir“.

Mit dieser Schadsoftware werden vor allem XAMPP oder verwandte Installationen angegriffen. Hierbei handelt es sich um leicht zu installierende Zusammenstellungen eines Webservers-Apache mit der Datenbank MySQL und den Skriptsprachen Perl und PHP. Sie existieren für die Betriebssysteme Linux (LAMPP), MS-Windows (WAMPP), Mac OS X (MAMPP) und Solaris und enthalten in der Regel noch weitere komfortable Module wie phpMyAdmin, einen ProFTPD-Server, OpenSSL, GD, Freetype2, libjpeg, libpng, gdbm, zlib, expat, Sablotron, libxml, Ming, Webalizer, pdf class, ncurses, mod_perl, FreeTDS, gettext, mcrypt, mhash, eAccelerator, SQLite und einen IMAP C-Client.

Solche Backdoor-Schadsoftware ist sehr gefährlich; angegriffenen Opfersystem liefert sie u. a. folgendes Aktionsportfolio:

- voller Zugriff auf Dateien des Systems inklusive Änderung der Zugriffsrechte
- Starten von Kommandointerpretern, wobei der Standard Input/Output an einen spezifischen TCP-Port gebunden oder mit Daten eines IRC-Servers (Datapipe) verknüpft werden kann
- Ausführen von beliebigem PHP-Code und beliebigen Shell-Kommandos
- Download/Upload von Dateien
- Festplatten nach beliebigen Inhalten/Dateien durchsuchen
- MySQL-Datenbanken vollständig administrieren
- FTP-Server Accounts nach schwachen Passwörtern scannen
- Benutzeraktivitäten verfolgen und Accounts ohne Passwort anlegen
- Spuren der eigenen Aktivität aus den Logdateien des Webservers und sogar sich selbst löschen.

Dazu bringt z. B. die C99Shell ein leicht zu bedienendes grafisches User-Interface mit, sodass der Angreifer effizient und kurz seinen Auftritt gestaltet.

Infektion
Das entwickelte Programm Analyzer of Death kann Spuren aller oben genannter Backdoors analysieren, ist aber besonders auf die Schadsoftware Destroyer und C99Shell ausgelegt.

Eine solche PHP-basierte Backdoor-Schadsoftware kann auf drei Arten auf einem Zielsystem genutzt werden.

Abb. 4.2-1: *Destroyer aka C99Shell*

Bei der ersten Methode wird die Backdoor-Schadsoftware direkt auf das System durch eine dort vorhandene Upload-Routine für Bilder, Dateien etc. gebracht. Solche Upload-Mechanismen existieren zum Beispiel auf allen sozialen Netzwerken und vielen Informationsportalen. Einmal auf das Opfersystem gebracht, lässt sich die Schadsoftware dann direkt über den Browser des Angreifers nutzen. Bei dieser Methode werden Schwachstellen der PHP-Installation ausgenutzt, die es erlauben, den PHP-Code des Backdoors direkt auszuführen.

Bei einer zweiten Methode wird die PHP-basierte Backdoor-Schadsoftware ebenfalls auf das System geschleust, die Ausführung wird aber an eine im System bereits vorhandene PHP-Seite gekoppelt. Enthält eine solche PHP-Webseite, angenommen die index.php, zum Beispiel das Codefragment

```
<?php
if(isset($_GET[„page“])) {
  require_once($_GET[„page“]);
}
...
?>
```

dann kann das Backdoor-Programm in einer sogenannten Local File Inclusion mit

http://www.opfersystem/index.php?page=backdoor.txt

über den Browser des Angreifers genutzt werden.

Bei der dritten Methode, der Remote File Inclusion, muss die PHP-basierte Backdoor-Schadsoftware gar nicht erst auf das Zielsystem gebracht werden. Bei dieser Methode nutzt man Schwachstellen im Code von PHP-Seiten des Opfersystems und belässt die Schadsoftware auf dem System des Angreifers. Beinhaltet eine PHP-Webseite, wieder angenommen die index.php, das bereits oben erwähnte Codefragment, so kann das Backdoor-Programm vom Angreifersystem in einer sogenannten Remote File Inclusion mit

http://www.opfersystem/index.php?page=http://www.angreifersystembackdoor.txt

Abb. 4.2-2: *Analyzer of Death*

wiederum über den Browser des Angreifers geladen werden.

Spurensuche

Das sogenannte Combined Log-Format gehört zu den am häufigsten benutzten Formaten des sehr ausgiebig konfigurierbaren Apache-Webservers. Das Log-Schema hat dabei folgendes Aussehen:

LogFormat „%h %l %u %t \“%r\“ %>s %b \“%{Referer}i\“ \“%{User-agent}i\““ combined

Die Spuren etwa des Destroyers oder einer C99Shell gliedern sich in dieses Format ein, folgen aber einer dieser Schadsoftware eigenen Syntax.

Ein typischer Log-Eintrag für den Destroyer aka C99Shell hat zum Beispiel etwa folgendes Aussehen:

vvv.xxx.yyy.zzz - [02/Aug/2010:06:00:33 +0100] „GET /Destroyer99.php.rar?act=f&f=.nsconfig&d=%2Fhome pages%2F20%2Fd69932965%2Fhtdocs%2Fforum%2F140581-ada& HTTP/ 1.1“ 200 3238 forum.Opfersystem „http://forum.Opfersystem/Destroyer99.php.rar?act=ls&d=%2Fhomepages%2F20%2Fd69932965%2Fhtdocs%2Fforum% 2F140581-ada&sort=0a“ „Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)“ „-“

Backend

Der entwickelte Analyzer of Death fungiert als Log-Parser, der die Logdatei analysiert und dabei einzelne Strings durch Filter passieren lässt. Der Parser legt dabei eine eigene Datei an, wobei jede Zeile in einzelne Teilstrings unterteilt in ein Array geschrieben wird. Dieses Array beinhaltet den exakten Zeitpunkt des Logeintrags (Systemzeit), die IP-Adresse, die Übertragungsmethode (GET/POST), Verzeichnis, Datei und deren Format bzw. Dateigröße, die URL und den Referrer.

Im URL-Teil befinden sich die Synonyme für die Funktionen der Schadsoftware, z. B. „act=“, „d=“ oder „f=“. Anhand dieser lassen sich in den meisten Fällen eindeutige Rückschlüsse auf die vom Angreifer durchgeführten Aktionen mit der Schadsoftware ziehen. Das Ergebnis der Analyse wird als HTML-Seite ausgegeben.

Frontend

Das Frontend des Analyzer of Death hat fünf Hauptelemente (siehe Abb. 4.2-3: Analyser of Death im Einsatz):

1. In das Suchfeld wird der Suchbegriff eingegeben, nachdem die Logdatei durchsucht wird.

2. Mit dem „Durchsuchen"-Button kann man eine Logdatei aus dem Dateisystem des Analysesystems einlesen. Der vollständige Pfad wird anschließend via JavaScript ausgelesen.

3. Dem Benutzer stehen zwei Ausgabemöglichkeiten zur Verfügung:
   a) „show data" zeigt die relevanten Bestandteile des Logeintrags an, ohne die Angreiferaktionen zu analysieren.
   b) „show important data" zeigt ebenfalls die relevanten Bestandteile des Logeintrags, jedoch wird jetzt nach Angreiferaktionen gefiltert und textuell beschrieben.

4. Analyse starten.

5. Die Ergebnistabelle listet die Analyseergebnisse chronologisch auf.

Der Analyzer of Death hat den Praxistest schon überstanden. Mit seiner Hilfe konnten bereits Angriffe auf Webserver analysiert und aufgeklärt werden. Unter anderem konnte die zuvor beschriebene Methode ermittelt werden, wie die PHP-basierte Backdoor-Schadsoftware auf die Opfersysteme gelangt ist und welche Informationen der/die Angreifer dazu im Vorfeld von Quellen aus dem Internet eingeholt haben.

Entsprechende forensische Gutachten wurden an die Webseiten-Betreiber respektive die Ermittlungsorgane weitergeleitet.

Referenzen/References

[1] Web Server Log Forensics App Wanted, http://ha.ckers.org/blog/20100613/web-server-log-forensics-app-wanted, 13. Juni 2010

[2] Fingerprinting port 80 attacks: a look into web server, and web application attack signatures, Zenomorph: http://www.cgisecurity.com/papers/fingerprintport80.

[3] Web Server Log Forensics App Wanted, http://ha.ckers.org/blog/20100613/web-server-log-forensics-app-wanted, 13. Juni 2010

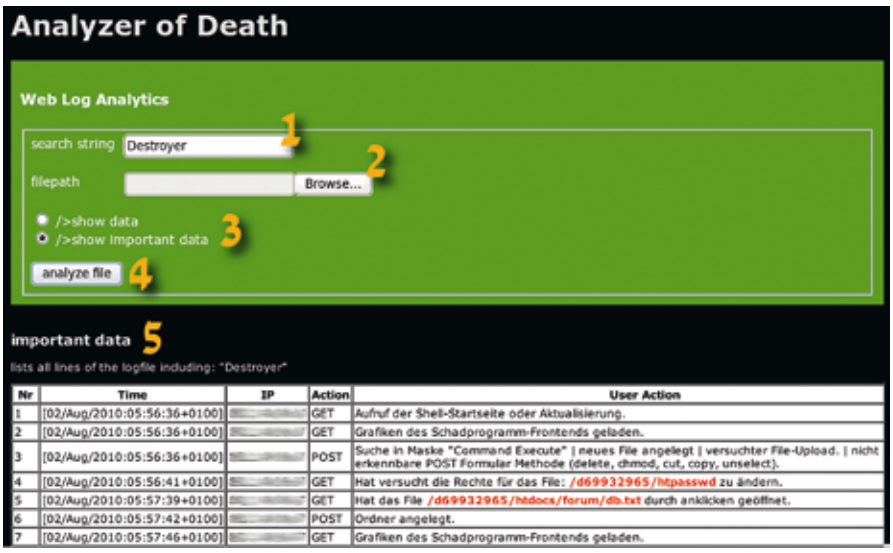

Abb. 4.-2-3: *Analyzer of Death im Einsatz*