Penetration Test Report

bknaus.de

Stefan Nicolin - Alexander Seith

Inhaltsverzeichnis

1. Zusammenfassung

Es zeigt sich dass die Sicherheitsimplementierung durchdacht ist und mehr als ausreichenden Schutz für die gehosteten Applikationen bietet. Es konnten lediglich nicht kritische Informationen über das System offengelegt werden.

Die Systemüberprüfung hat folgende Informationslecks ergeben:

I. Das Contentmanagementsystem Joomla! erlaubt durch die direkte Angabe eines Themes (individuelle Anzeige der Seite) die lokale und nicht dauerhafte Änderung des Aussehens der Seite.
II. Durch eine präparierte URL ist es möglich Informationen über die Installation des Visualisierungstools Cacti zu erfahren und für weitere Angriffe ggf. auszunutzen.
III. Eine veraltete Mamboinstallation (Vorgänger von Joomla!) hat die Registrierung eines Benutzers erlaubt. Durch eine fehlgeschlagene Sicherheitsüberprüffung der verwendeten Datenbank wurden eine Vielzahl von eMails generiert und haben ein spammerähnliches Verhalten hervorgebracht.
IV. Es konnte die Firewall mit einem TCP-Traceroute umgangen werden.
V. Offenlegung von Systeminformationen im eMail-Header

Folgende Maßnahmen werden empfohlen:

I. Die nichtbenötigten Joomla! Themes sollten entfernt werden.

II. Die Aktualisierung auf die neueste Joomla Version wird empfohlen sowie die Änderung der Webserverkonfiguration um im Fehlerfall die Menge an systemrelevanten Informatinen zu reduzieren.

III. Es wird ein Update von Mambo auf die aktuellste Joomla! Version angeraten. Desweiteren soll geprüft werden ob die Anmeldung von neuen Benutzer ein Freigabeprozess bedarf.

IV. Der Einsatz eines Intrusion Detection System (IDS) zur Feststellung von Evasionsangriffen auf die Firewall

V. Unterbinden des Versenden von Systeminformationen im eMail-Header

2. Ziele

Im Rahmen der Vertiefungsveranstaltung “Telekommunikation” an der Fachhochschule Wiesbaden wurde die Seite bknaus.de und alle damit assoziierten Domains und Systeme einem Penetrationstest unterzogen. Der Test wurde auf allen Ebenen durchgeführt und war unbeschränkt in seinen Angriffsmitteln. Es wurde lediglich auf „Social Engineering“ verzichtet.

Ziel des Tests war es, Schwachstellen ausfindig zu machen, auszunutzen und ggf. eine Handlungsempfehlung zu geben.

3. Methodik

Der Penetrationstest erfolgte als sogennanter „Black-Box Test“, d.h. es lagen keine weiteren Informationen als die öffentlich zugänglichen Informationen zur Verfügung. Es gestaltete sich deshalb das Szenario eines echten, externen Angreifs auf das System.

Genauer betrachtet wurden:

I. Betriebssystem
   - Version und bekannte Schwachstellen (sog. Exploits)

II. Dienste
   - Arten, Versionen und bekannte Schwachstellen

III. Firewall
   - Art der Firewall, Regelwerk und Umgehung (sog. Evasion)

IV. Gehostete Domains und Anwendungen
   - z.B. Mambo, Joomla!, Cacti, Modul- und Versionsbestimmung

V. Anfälligkeiten für verschiedene Angriffe aus der Anwendungsschicht
   - Cross-Site-Scripting (XSS), SQL-Injection, Path Disclosure, Remote File Inclusion, Directory Traversal, ...

Zur Durchführung der Tests wurde folgende Software verwendet:

I. nmap (Portscanner)

II. Nessus (Vulnerability-Scanner)

III. fpdns (DNS Fingerprinter)

IV. Firefox & Safari (Webbrowser)

V. tcptraceroute (Tracerouting auf TCP-Basis)

4. Systemanalyse

4.1 Allgemeine Systeminformation

Unter Verwendung des bekannten Opensource Scanners Nmap von insecure.org wurde ein Profil des Systems erstellt.

Dabei ließen sich folgende Informationen zusammentragen:

IP Adresse: 85.14.217.227

Serverbetriebsystem: Gentoo Linux

Virtualisierungssoftware: XEN

Offene Ports:

Port 21	FTP Server, TLS getunnelt	Software unbekannt
Port 25	Mailserver	Net-Qmail
Port 53	DNS Server	BIND 9.2.3rc1 – 9.4.0a0
Port 80	Webserver	Apache (mit PHP5)
Port 993	IMAP Server	Courier Imap
Port 443	Webserver SSL	Apache
Port 222	SSH Daemon	OpenSSH

Zu den gehosteten Anwendungen zählen Joomla, Mambo, Cacti und Squirrelmail.

Diese (meist) PHP basierenden Anwendungen sollten das Hauptziel des Penetrationstests darstellen.

4.2 FTP Server

Der Server wird über TLS getunnelt. Ein eigens dafür erstelltes Zertifikat kommt zum Einsatz. Die Verwendung des Opensource Tools curl hat folgendes ergeben:

```
curl --ftp-ssl ftp://85.14.217.227 -v

* About to connect() to 85.14.217.227 port 21 (#0)
*   Trying 85.14.217.227... connected
* Connected to 85.14.217.227 (85.14.217.227) port 21 (#0)
< 220 Unberechtigter Zugriff wird strafrechtlich verfolgt!
> AUTH SSL
< 234 AUTH SSL successful
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/certs/ca-certificates.crt
  CApath: none
* SSLv2, Client hello (1):
* SSLv3, TLS handshake, Server hello (2):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Server key exchange (12):
* SSLv3, TLS handshake, Request CERT (13):
* SSLv3, TLS handshake, Server finished (14):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Client key exchange (16):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSL connection using DHE-RSA-AES256-SHA
* Server certificate:
*        subject: /CN=bknaus.de
*        start date: 2007-07-06 12:26:07 GMT
*        expire date: 2008-01-02 12:26:07 GMT
* SSL: certificate subject name 'bknaus.de' does not match
target host name '85.14.217.227'
* Closing connection #0
* SSLv3, TLS alert, Client hello (1):
curl: (51) SSL: certificate subject name 'bknaus.de' does not
match target host name '85.14.217.227'
```

Ein Textbanner weist beim Verbinden darauf hin dass solche Aktionen straffrechtlich verfolgt werden.

4.3 Mail Server

Durch das Verbinden auf Port 25 zeigt sich die Verwendung der Mailserversoftware NetQmail.

```
telnet bknaus.de 25
Trying 78.47.111.98...
Connected to bknaus.de.
Escape character is '^]'.
220 bknaus.de ESMTP
help
214 netqmail home page:
http://qmail.org/netqmail
quit
221 bknaus.de
Connection closed by foreign host.
```

Weitere sensitive Informationen ließen sich durch das Versenden einer “Bogus Mail” aus dem Header der Antwortmail herauslesen:

```
Received: from 209.85.128.187 by gondor (envelope-from
<xxx@gmail.com>, uid 201) with qmail-scanner-1.25st
(clamdscan: 0.88.5/2132. spamassassin: 3.1.3. perlscan:
1.25st.
Clear:RC:0(209.85.128.187):SA:0(-2.6/5.0):.
Processed in 5.293867 secs); 24 Feb 2008 15:11:20 -0000
X-Spam-Status: No, hits=-2.6 required=5.0
Received: from fk-out-0910.google.com (209.85.128.187)
by static.98.111.47.78.clients.your-server.de with SMTP; 24
Feb 2008 15:11:15 -0000
```

4.4 DNS Server

Die Version der eingesetzten Software ließ sich mit Hilfe des Tools “fpdns” (FingerprintDNS) herausfinden:

```
fingerprint (85.14.217.227, 85.14.217.227): BIND 9.2.3rc1 - 9.4.0a0
```

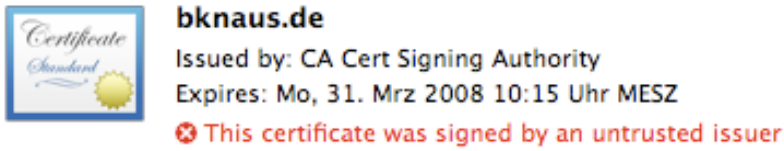
4.5 Webserver

Die Signatur des Webservers Apache wurde vom Betreiber ausgeschaltet. Eine Analyse eines HTTP Requests ergab die Information über die Verwendung von PHP5 unter Gentoo Linux. Die Untersuchung des Seitenquelltextes deutete auf dem Einsatz von Joomla hin.

```
X-Powered-By: PHP/5.2.2-pl1-gentoo

<meta name="Generator" content="Joomla! - Copyright (C)
2005 - 2006 Open Source Matters. All rights reserved." />
```

Grundlage der SSL gesicherten Kommunikation bildet ein freies Zertifikat von Cacert.org. Dadurch dass Cacert.org nicht in der Liste der Stammzertifikate der meisten Browser vorhanden ist, wird beim Wechsel auf https eine Browserwarnung ausgegeben.



4.6 IMAP Server

Informationen über den verwendeten Mailserver konnten nicht erhoben werden.

4.7 SSH Daemon

Auf dem ungewöhnlichen Port 222 lauscht ein Dienst zur sicheren Terminalanmeldung. Die Wahl der Portnummer deutet auf dem Einsatz der Linux-Firewallsoftware IPCop hin.

5. Ergebnisse

5.1 Joomla! CMS

Das Contentmanagement System Joomla basierend auf PHP ist anfällig für verschiedene Schwachstellen. Keine der getesteten Schwachstellen kann jedoch als kritisch eingestuft werden. Es folgt eine Auflistung der Tests:

5.1.1 Lokale Änderung des Joomla Templates

Durch die Manipulation des URL Strings ist es möglich andere Joomla Templates als die vom Webseitenbetreiber vorgesehen zu aktivieren. Diese Änderung betrifft jedoch nur die eigene Anzeige der Website und ist somit nicht dauerhaft.

http://www.bknaus.de/index.php?jos_change_template=rhuk_solarflare_ii

5.1.2 Cross-Site-Scripting (XSS)

Es wurden zwei Cross-Site-Scripting Schwachstellen getestet, jedoch ohne Erfolg.

I. com_search component Vulnerability
http://nvd.nist.gov/nvd.cfm?cvename=CVE-2007-5427

   Die Eingabe von HTML Code im Suchfeld ermöglicht das Ausführen von Code innerhalb der Anwendung.

II. com_colorlab remote file inclusion Vulnerability
http://nvd.nist.gov/nvd.cfm?cvename=CVE-2007-5451

   Durch eine Komponente im Administrationsbereich ist es Angreifern unter Umständen möglich PHP Code über die URL einzuschleußen.

5.2 Mambo CMS

Bei Mambo handelt es sich um den Vorgänger von Joomla!, da eine Reihe von Hauptentwicklern das Projekt gewechselt haben.

Es bestand die Möglichkeit einen freien Benutzeraccount zu registrieren. Dieser wurde ohne vorherige Überprüfung freigeschaltet und stand sofort zur Nutzung bereit.

Mambo SQL-Injection: http://retrogod.altervista.org/mambo_46rc1_sql.html

Durch einen fehlgeschlagenen Exploit wurde eine sehr große Anzahl von eMails generiert. Die blinde SQL-Injection hat zum Ziel gehabt das Superadministrator Password in den Datenbankserverlogs auszugeben.

5.3 Cacti

Die eingesetzte Visualisierungssoftware Cacti ist innerhalb der Xendomain ohne Authentifizierung erreichbar. Es wird kein Benutzername benötigt um die Auswertung anzuzeigen. Die angezeigten Graphen enthalten sensitive Informationen über interne Devices sowie Netzwerkinformationen.

Das interne Klasse C Netz (192.168.100.0/24) wird genutzt. Der Einsatz einer VPN Software wird duch das Vorhandensein von /dev/tun0 signalisiert. In Verbindung mit Linux kommt die beliebte Software OpenVPN in Frage.

5.3.1 Cross-Site-Scripting (XSS) & SQL-Injection

Für Cacti wurden verschiedene Exploits getestet: http://www.ush.it/team/ush/hack-cacti087a/cacti.txt

Die Durchführung eines der Tests führte zur Offenlegung des vollständigen Pfades der Cactiinstallation. Auch die Verwendung der Datenbanksoftware MySQL wurd ersichtlich.

http://monitor.itg-em.de/cacti/tree.php?action=edit&id=1

```
Warning: mysql_pconnect() [function.mysql-pconnect]: Access
denied for user 'cactiuser'@'localhost' (using password:
YES) in /usr/share/php5/adodb/drivers/adodb-mysql.inc.php
on line 382
02/24/2008 11:23:40 PM - CMDPHP: Poller[0] FATAL: Cannot
connect to MySQL server on 'localhost'. Please make sure
you have specified a valid MySQL database name in
'include/config.php'.
Warning: Cannot modify header information - headers already
sent by (output started at
/usr/share/php5/adodb/drivers/adodb-mysql.inc.php:382) in
/var/www/monitor.itg-em.de/htdocs/cacti/include/auth.php on
line 31
```

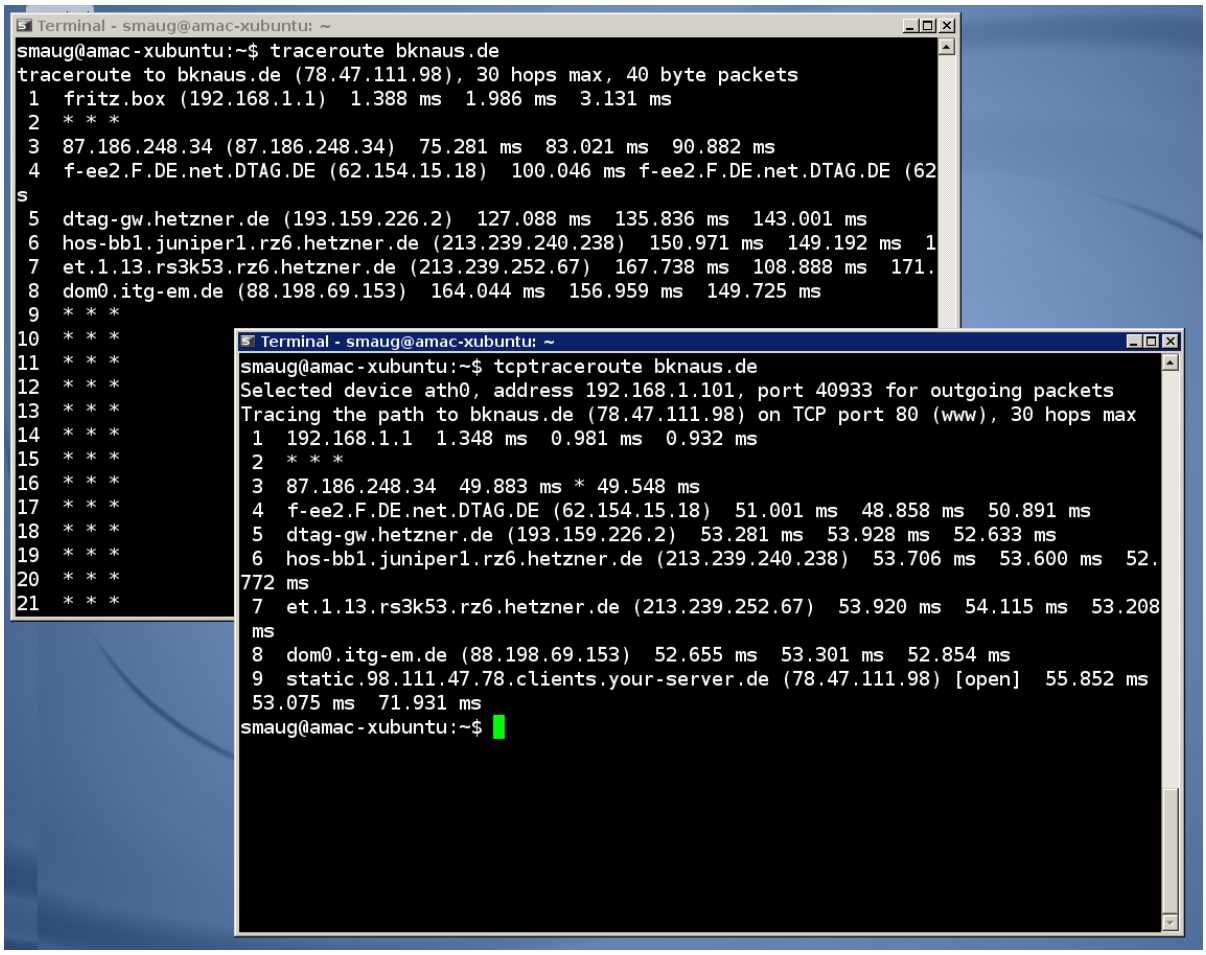
Alle weiteren Exploits in o.g. Dokument führten jedoch zu keinem geänderten Ergebnis. Es wird vermutet, das die PHP-Konfigurationsvariable „Magic Quotes“ auf „ON“ gestellt ist, weshalb alle weiteren Exploits daran scheitern (siehe Exploit).

5.4 Squirrelmail

Die Webmailsoftware Squirrelmail ist auf dem aktuellsten Stand. Es konnten keine Schwachstellen gefunden werden. Ein kurzer Test mit einer Firefoxerweiterung zum Testen von Standard XSS und SQL-Injections scheiterete erwartungsgemäß und führt zu keinem Ergebnis.

5.5 System

Gegenüber traceroute welches das UDP Protokoll nutzt konnte mit Hilfe von tcptraceroute ein vollständiger Traceroutelauf durchgeführt werden. Dadurch war es möglich, die Firewall zu umgehen und so eventuell weitere Systeme hinter der Firewall ausfindig zu machen und ggf. seperat einem Penetrationstest zu unterziehen. Leider gab es jedoch wie vermutet hinter der Firewall keine weiteren Angriffsziele.



6. Bewertung

Der Penetrationstest verlief aus der Sicht des Betreibers positiv. Insgesamt kann gesagt werden, dass das Ziel bknaus.de auf allen Ebenen sehr gut abgesichert ist. Kein einziger aller gefährlichen Angriffe führte zum Erfolg.

Es konnten lediglich weitere Kenntnisse über das Angriffsziel in Erfahrung gebracht werden, die jedoch nicht für andere Angriffe von nutzen waren.

7. Handlungsempfehlung

Folgende Maßnahmen werden empfohlen, um die in Kapitel 4& 5 festgestellten Schwachstellen zu beheben:

I. Die lokale Änderung der Templates in Joomla! sollte entweder unterbunden werden, in dem alle weiteren Templates gelöscht oder ordnungsgemäß der Seite angepasst und dem Nutzer per Menü zur Verfügung gestellt werden.

II. Hinsichtlich des Full Path Disclosure in Cacti sollte die Installation und Konfiguration überprüft und ggf. aktualisiert werden. Falls der Fehler dadurch nicht behoben ist, sollte der Quellcode in der entsprechenden Datei gegenüber falschen URL-Parametern immunisiert werden.

III. Bezüglich des Angriffs auf die Mamboinstallation ist anzuraten, auf die neuste Version von Joomla! umzusteigen und/oder ein manuellen Freigabeprozess für Neuregistrierungen einzuführen.

IV. Um das Umgehen der Firewall durch z.B. tcptraceroute zu verhindern, wird der Einsatz eines Intrusion Detection System (IDS) empfohlen.

V. Das Übermitteln von internen Informationen per eMail-Header ist überflüssig und sollte unterbunden werden. Dazu ist eine entsprechende Änderung in der Konfiguration von qMail notwendig.

8. Literaturverzeichnis

[nmap] http://nmap.org, Portscanner

[Firefox] http://www.mozilla.com/en-US/firefox/, Sehr erfolgreicher Opensource Browser

[XSS-Me, SQL Inject-Me] http://www.securitycompass.com/exploitme.shtml, Firefoxerweiterungen zur automatischen Schwachstellenüberprüffung

[fpdns] http://www.rfc.se/fpdns, Fingerprint DNS Server

[nessus] http://www.nessus.org/nessus/, Sicherheitsscanner

[tcptraceroute] http://michael.toren.net/code/tcptraceroute/, Traceroute auf TCP Basis