Webseiten- und Datenbanksicherheit

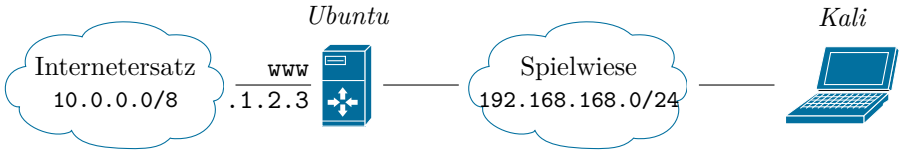
Lösen Sie die nachfolgenden Aufgaben und bereiten Sie diese bis zum nächsten Lehrveranstaltungstermin vor.

01.

a) Konfigurieren Sie den Webserver auf dem *Ubuntu*-Server so, dass die IP der in der Grafik unten abgebildeten entspricht und der Webserver alle Anfragen der Domäne bearbeitet. Folgen Sie den Hinweisen zur Einrichtung.

b) Kopieren Sie die zur Verfügung gestellten Beispieldateien ins Verzeichnis `/var/www` auf dem *Ubuntu*-Server. Überprüfen Sie anschließend die Funktionsfähigkeit der Webseite, indem Sie sie in der *Kali*-VM mit *Firefox* öffnen und alle Links auf der Startseite einmal anklicken.
Hinweis: Sie können im Rahmen dieses Laborblockes den Befehl `scp` auf der Kali-*VM verwenden, um Dateien auf den* Ubuntu-*Server zu kopieren. Beachten Sie allerdings die fehlenden Zugriffsrechte des Benutzers `bruce` auf das Zielverzeichnis. Kopieren Sie daher die Dateien zuerst nach `/tmp` und verschieben Sie sie von dort als Benutzer `wartung` auf dem* Ubuntu-*Sever mit dem Befehl `mv` ins Zielverzeichnis.*

c) Überprüfen Sie Möglichkeit von HTML bzw. JavaScript Injection, indem Sie in das Eingabefeld der eingerichteten Seite `top.html` in *Firefox* in der *Kali*-VM `<b>Test</b>` bzw. `<script language="javascript"> alert("Hallo Welt");</script>` eingeben. Erläutern Sie die Sicherheitsimplikationen dieser beiden Angriffsvektoren.



Domäne `spielwiese-<Nachname>.tld.`, z.B. `spielwiese-unterweger.tld.`

02.

a) Erstellen Sie in der *Kali*-VM eine HTML-Seite mit dem Dateinamen `malicious.html` in `/var/www/html`, die einem Nachbau der Ergebnisseite von `top.html` entspricht, aber eine zusätzliche Fehlermeldung sowie ein Passworteingabefeld enthält. Starten Sie einen weiteren Webserver auf der *Kali*-VM mit dem Befel `service apache2 start` und überprüfen Sie, ob die erzeugte HTML-Seite von anderen Rechnern (z.B. aus der *Windows*-VM) abrufbar ist.

b) Führen Sie eine JavaScript Injection über das Eingabefeld auf `top.html` durch, bei der nach dem Anzeigen der Ergebnisseite `malicious.html` vom Webserver der *Kali*-VM geladen wird. Diskutieren Sie, welche URL in der Adresszeile angezeigt wird.

c) Erstellen Sie auf dem *Ubuntu*-Server eine Kopie von `customize_v1.php`, z.B. `customize_v2.php`, in der Sie `$_POST['name']` mit dem Aufruf der Funktion `htmlspecialchars` umschließen. Modifizieren Sie außerdem `top.html` entsprechend. Wiederholen Sie einige Ihrer bisherigen Angriffe und diskutieren Sie, inwieweit diese nun verhindert werden können.

03.

a) Loggen Sie sich mit dem Befehl `mysql -p<Passwort>` (z.B. `mysql -pwartung`) auf dem *Ubuntu*-Server in die SQL-Konsole ein und legen Sie eine Datenbank namens `Spielwiese` an. Legen Sie in dieser Datenbank eine Tabelle `products` an, die zumindest über die beiden Spalten `name` und `description` verfügt. Befüllen Sie die Tabelle mit mindestens drei Datensätzen und überprüfen Sie die korrekte Einrichtung der Datenbank(-Tabelle) über das Suchformular auf der Beispielseite `search.html` in einem Browser.
Hinweis: Beachten Sie die Hinweise zur Verwendung von PHP, um temporär Fehlermeldungen bei Seitenaufrufen anzeigen zu lassen.

b) Überprüfen Sie Möglichkeit von SQL Injection, indem Sie in das Eingabefeld auf der eingerichteten Seite `search.html` in *Firefox* in der *Kali*-VM `" OR 1=1 OR "` bzw. `" OR "*"; --` (beachten Sie das zusätzliche Leerzeichen am Ende!) eingeben. Beschreiben Sie, welche Datenbankabfrage im Hintergrund effektiv ausgeführt wird und erläutern Sie die Sicherheitsimplikationen dieses Angriffsvektors.

c) Führen Sie eine erneute SQL Injection mit der Eingabe `" UNION ALL SELECT database(), LOAD_FILE("/etc/passwd"); --` (beachten Sie das zusätzliche Leerzeichen am Ende!) durch. Erläutern Sie die Ausgabe, deren Ursache und die Sicherheitsimplikationen dieser Art von Abfrage.

d) Erstellen Sie auf dem *Ubuntu*-Server eine Kopie von `search_v1.php`, z.B. `search_v2.php`, in der Sie `$_POST['query']` mit dem Aufruf der Funktion `mysql_real_escape_string` umschließen. Modifizieren Sie außerdem `search.html` entsprechend. Wiederholen Sie einige Ihrer bisherigen Angriffe und diskutieren Sie, inwieweit diese nun verhindert werden können.

Hinweis zur Einrichtung des Webservers auf einem *Ubuntu*-Server

Stellen Sie sicher, dass Sie eingeloggt sind und führen Sie alle nachfolgenden Befehle als Superuser aus. Öffnen Sie die Port-Konfigurationsdatei des Webservers mit dem Editor `nano`, indem Sie

`nano /etc/apache2/ports.conf`

eingeben.
Ändern Sie in der Zeile

```
1 NameVirtualHost *:80
```

den Platzhalter `*` durch die IP-Adresse, die der Webserver bedienen soll. Entfernen Sie außerdem die Zeile

```
1 Listen 80
```

Speichern Sie die Datei und beenden Sie `nano`.
Öffnen Sie anschließend die Konfigurationsdatei `/etc/apache2/apache2.conf` mit dem Editor `nano` und fügen Sie am Ende die Zeile

```
1 Listen <IP-Adresse>:80
```

ein, wobei `<IP-Adresse>` die oben angegebene IP-Adresse des Webservers ist. Speichern Sie die Änderungen und beenden Sie `nano`.
Öffnen Sie anschließend die Datei `/etc/apache2/sites-enabled/000-default` zur Standardwebseitenkonfiguration mit `nano`. Ändern Sie `VirtualHost *:80` in der ersten Zeile wie oben beschrieben ab. Fügen Sie anschließend vor der letzten Zeile (`</VirtualHost>`) die folgenden beiden Zeilen ein:

```
1 ServerName <Servername>.<Domänenname>
2 ServerAlias <Domänenname>
```

`<Servername>` und `<Domänenname>` müssen dabei durch den Rechner- bzw. den Domänennamen ersetzt werden. Die spitzen Klammern sind **nicht** zu inkludieren. Beispiel:

```
1 ServerName www.spielwiese-unterweger.tld
2 ServerAlias spielwiese-unterweger.tld
```

In dieser Beispielkonfiguration bedient der Webserver alle Anfragen an den Servernamen `www.spielwiese-unterweger.tld`, bearbeitet aber auch Anfragen an die gesamte Domäne, d.h. ohne Angabe des Servernamen.
Speichern Sie die Datei und beenden Sie `nano`.
Öffnen Sie abschließend die HTTP-Konfigurationsdatei `/etc/apache2/httpd.conf` mit `nano` und fügen Sie die Zeile

```
1 ServerName localhost
```

hinzu, damit der Webserver seinen eigenen Namen nicht über ein Reverse DNS Lookup ermittelt, sondern fix auf dem lokalen Rechnernamen belässt. Speichern Sie die Datei und beenden Sie `nano`.
Nach der Konfiguration muss der Webserver neu gestartet werden:

`/etc/init.d/apache2 restart`

Abschließend kann mit dem Befehl

```
netstat -4at
```

überprüft werden, ob nur ein HTTP-Port auf der gewünschten IP-Adresse offen ist und keine weiteren IP-Adressen bedient werden.

Hinweis zur Verwendung von PHP auf einem als Webserver konfigurierten *Ubuntu*-Server

Um Fehlermeldungen bei der Erstellung von Seiten mit PHP anzeigen zu lassen, muss die Konfigurationsdatei `/etc/php5/apache2/php.ini` mit `nano` geöffnet werden und die Zeile

```
1 display_errors = Off
```

durch

```
1 display_errors = On
```

ersetzt werden. Nach dem Speichern der Datei und dem Beenden von `nano` muss der Webserver – wie oben beschrieben – neu gestart werden.