2 Die Testumgebung

Wenn man sich ausgiebig mit den Möglichkeiten von Metasploit beschäftigen möchte, ist es sinnvoll, sich eine Testumgebung einzurichten. Wie bereits erwähnt, ist das Framework auf verschiedenen Plattformen nutzbar. Wir werden hier zumindest die Betriebssysteme Windows und Linux etwas näher betrachten. Außerdem bietet es sich an, verschiedene virtuelle Umgebungen einzurichten, um die Wirkung auf den unterschiedlichen Plattformen auszutesten.

Derzeit gibt es einige kostenlose Produkte, um virtuelle Server oder Workstations in einem Netzwerk darzustellen. Citrix stellt eine kostenfreie Vollversion seines XenServers[1] zur Verfügung. Die Installation erfolgt über ein CD-Image und ist in wenigen Minuten erledigt. Auf der gleichen CD wird die Administrationskonsole *XenCenter* als Applikation für Windows bereitgestellt. Sie beinhaltet eine hierarchisch gegliederte Bedienoberfläche. Damit ist man in der Lage, mehrere XenServer über das Netzwerk zu verwalten. Nach der Registrierung erhält man eine Lizenz, die man für ein Jahr nutzen kann.

Für einfache Tests in einer virtuellen Umgebung reicht oftmals der VMware-Player[2] aus. Auch hier bietet sich die Möglichkeit, mehrere Betriebssysteme gleichzeitig auf einem PC auszuführen. Auch wenn es der Name der Software nicht klar ausdrückt – im Gegensatz zu älteren Versionen kann man mit dem aktuellen Produkt nicht nur virtuelle Maschinen (VM) »abspielen«, sondern auch VMs erstellen.

In diesem Buch wurde die Testumgebung auf der Basis von VirtualBox[3] angelegt. Diese kostenfreie Lösung hat sich besonders bei kleinen und mittelgroßen Testumgebungen bewährt und zeichnet sich durch eine hohe Zuverlässigkeit, einfache Administration und gute Performance aus. Derzeit läuft es unter den Betriebssystemen Windows, Linux, Mac OS X und Open Solaris. Es wird ständig in der Community weiterentwickelt und entspricht professionellen Standards. Als Hostsystem wird ein Ubuntu-Server LTS in der 64bit-Version eingesetzt.

1. *http://www.citrix.de/produkte/xenserver/*
2. *http://www.vmware.com/de/products/desktop_virtualization/player/overview.html*
3. *https://www.virtualbox.org/*

Zum Anlegen von virtuellen Maschinen wurde ein Webinterface auf der Basis von PHP installiert. Dieses stellt einen Assistenten zur Verfügung, der durch die einzelnen Schritte bei der Installation von virtuellen Maschinen führt. Es lassen sich außerdem RAM- und Festplattengröße definieren und vorgefertigte Festplatten- bzw. ISO-Images auswählen. Wie von kommerziellen Lösungen (z.B. VMware Workstation) bekannt, bietet auch VirtualBox bestimmte Grundeinstellungen für unterschiedliche Betriebssysteme an, die man nach eigenen Vorstellungen modifizieren kann.

Die im Buch verwendeten virtuellen Maschinen wurden auf folgender Hard- und Software erstellt:

Hardware: Intel Core 2 Quad Q8300, 8 GB RAM, 1 TB Festplatte
Software: Ubuntu 10.04.3 Server 64 bit, Linux (2.6.32-33-Server),
VirtualBox: 4.1.8 (75467)

2.1 VirtualBox 4.1 und phpVirtualBox installieren

VirtualBox von Oracle ist eine hervorragende Lösung, die eine kostengünstige Virtualisierung von x86- und AMD64/Intel64-Umgebungen ermöglicht und sowohl im professionellen Umfeld als auch von privaten Nutzern geschätzt wird. Die Software ist als Open Source unter der GNU General Public License frei verfügbar und ist für alle gängigen Betriebssysteme erhältlich. Es ist also nicht verwunderlich, dass die Nutzer-Community ständig wächst und dadurch ein ausgezeichneter Support ermöglicht wird.

In diesem Abschnitt wird zunächst erläutert, wie VirtualBox 4.1 auf einem Ubuntu-Server ohne grafische Oberfläche installiert und später mittels phpVirtualBox ein Ajax-Webinterface hinzugefügt wird. Mithilfe dieser Oberfläche lassen sich dann alle Operationen von der Erstellung virtueller Maschinen, dem Anfertigen von Snapshots bis zur Veränderung der Hardwareeinstellungen bequem ausführen. Zur Bedienung wird dann nur ein Browser Ihrer Wahl benötigt.

Einem sachkundigen Anwender sollte die Installation eines Ubuntu-Servers keine Probleme bereiten. Aus diesem Grund wird sie auch hier nicht weiter beschrieben. Als Grundlage für das Hostsystem wird in diesem Buch Ubuntu-Server 10.04.3 LTS[4] (Lucid Lynx) verwendet. Für die Realisierung unseres Vorhabens brauchen wir keine grafische Oberfläche (wie z.B. Gnome oder KDE) und können daher auf diese schlanke Installation zurückgreifen.

Für die weitere Vorgehensweise setzen wir voraus, dass der Ubuntu-Server installiert ist, der Zugriff auf die Konsole mittels einer SSH-Session erfolgt und eine Internetverbindung zur Verfügung steht.

In einem ersten Schritt (Listing 2-1) wird zunächst geprüft, ob sich alle Pakete auf dem aktuellen Stand befinden und die richtige Version installiert ist.

4. *http://www.ubuntu.com/download/server/download*

```
sudo su
apt-get update
apt-get upgrade

root@virtualbox: lsb_release -a
No LSB modules are available.
Distributor ID:  Ubuntu
Description:     Ubuntu 10.04.3 LTS
Release:         10.04
Codename:        lucid
```

Listing 2–1 *Software aktualisieren*

Nun wird das Quellverzeichnis um das VirtualBox-Repository ergänzt und die benötigten Schlüssel hinzugefügt.

```
echo "deb http://download.virtualbox.org/virtualbox/debian lucid contrib non-free"
| sudo tee -a /etc/apt/sources.list

wget -q http://download.virtualbox.org/virtualbox/debian/oracle_vbox.asc -O- |
sudo apt-key add –
```

Listing 2–2 *Listing 2-2 VirtualBox-Repository hinzufügen*

Listing 2-3 zeigt die notwendigen Befehle für die Installation von VirtualBox, den Kernel-Ressourcen bzw. dem Apache Webserver und PHP5. Zum Abschluss wird noch der »Oracle VM_VirtualBox Extension Pack« heruntergeladen und installiert. Dieser ermöglicht später u.a. die Nutzung von USB-2.0-Geräten bzw. die Unterstützung des »VirtualBox Remote Desktop Protocol«.

```
apt-get update
apt-get install virtualbox-4.1
apt-get install linux-headers-`uname -r` dkms unzip
apt-get install apache2 php5 libapache2-mod-php5 unzip

wget
"http://download.virtualbox.org/virtualbox/4.1.4/Oracle_VM_VirtualBox_Extension_
Pack-4.1.4.vbox-extpack"

VBoxManage extpack install Oracle_VM_VirtualBox_Extension_Pack-4.1.4.vbox-extpack
```

Listing 2–3 *Benötigte Software installieren*

Die Virtualisierungssoftware wird bald in einem eigenen Prozess mit Nutzerrechten laufen. Um dies zu gewährleisten, sind einige Vorbereitungen zu treffen. Zunächst legen wird die Datei `/etc/default/virtualbox` an und fügen gemäß Listing 2-4 den Nutzer `virtualbox` in einer Zeile hinzu. Mit den folgenden Befehlen legen wir den Benutzer im System an und fügen ihn der Gruppe `vboxusers` hinzu.

```
root@virtualbox:~# echo „VBOXWEB_USER=virtualbox" > /etc/default/virtualbox
root@virtualbox:~# adduser virtualbox
Lege Benutzer »virtualbox« an ...
Lege neue Gruppe »virtualbox« (1001) an ...
Lege neuen Benutzer »virtualbox« (1001) mit Gruppe »virtualbox« an ...
Erstelle Home-Verzeichnis »/home/virtualbox« ...
Kopiere Dateien aus »/etc/skel« ...
Geben Sie ein neues UNIX-Passwort ein:
Geben Sie das neue UNIX-Passwort erneut ein:
passwd: password updated successfully
Changing the user information for virtualbox
Enter the new value, or press ENTER for the default
       Full Name []: Frank Neugebauer
       Room Number []:
       Work Phone []:
       Home Phone []:
       Other []:
Sind diese Informationen korrekt? [J/n] j

root@virtualbox:~# adduser virtualbox vboxusers
Füge Benutzer »virtualbox« der Gruppe »vboxusers« hinzu ...
Adding user virtualbox to group vboxusers
Fertig.
```

Listing 2–4 *Nutzer anlegen und Gruppe hinzufügen*

Mit diesen ersten Schritten haben wir die Virtualisierungssoftware erfolgreich installiert und die notwendigen Nutzer erstellt. Um VirtualBox über eine Webschnittstelle administrieren zu können, wird nun phpVirtualBox[5] eingerichtet. Auch diese Software ist Open Source und wird unter der GNU GPL-Lizenz zur Verfügung gestellt. Gemäß Listing 2-5 wird das in PHP geschriebene Programmpaket zunächst mittels dem `wget`-Befehl heruntergeladen und entpackt. Nachdem der notwendige Ordner in der Verzeichnisstruktur des Webservers angelegt worden ist, muss nun die mitgelieferte Beispielkonfiguration kopiert und angepasst werden. In diesem Fall wird die Datei `config.php` mit dem Editor Ihrer Wahl geöffnet und die Variablen `$username` und `$password` entsprechend angepasst. Tragen Sie dazu das in Listing 2-4 genutzte Passwort für den Nutzer `virtualbox` ein. Mit dem Befehl `vbox-service start` kann nun das Webinterface gestartet werden.

```
wget http://phpvirtualbox.googlecode.com/files/phpvirtualbox-4.1-5.zip
unzip phpvirtualbox-4.1-5.zip
mkdir /var/www/phpvirtualbox
cp -R phpvirtualbox-4.1-5/* /var/www/phpvirtualbox
```

5. *http://code.google.com/p/phpvirtualbox/*

```
sudo cp /var/www/phpvirtualbox/config.php-example
/var/www/phpvirtualbox/config.php

nano /var/www/phpvirtualbox/config.php

/* Username / Password for system user that runs VirtualBox */
var $username = 'virtualbox';
var $password = 'deinPasswort';

sudo /etc/init.d/vboxweb-service start
```

Listing 2–5 *phpVirtualBox installieren und Konfiguration anpassen*

Nachdem alle Anpassungen gemacht wurden, sollte der Server zunächst erst einmal heruntergefahren und neu gestartet werden. Alle Dienste sind standardmäßig so eingestellt, dass sie beim Hochfahren des Systems automatisch gestartet werden. Bevor Sie nun weiterarbeiten, sollten einmal die Kernel-Module von VirtualBox konfiguriert werden. Nutzen Sie dazu den Befehl gemäß Listing 2-6. Weist die Konfiguration keine Fehler auf, so kann das Webinterface über folgende URL gestartet werden:

http://IP-Adresse/phpvirtualbox/

```
root@virtualbox:/home/frank# /etc/init.d/vboxdrv setup
 * Stopping VirtualBox kernel modules                              [ OK ]
 * Uninstalling old VirtualBox DKMS kernel modules                 [ OK ]
 * Trying to register the VirtualBox kernel modules using DKMS     [ OK ]
 * Starting VirtualBox kernel modules                              [ OK ]
```

Listing 2–6 *Konfiguration der Kernel-Module von VirtualBox*

Nutzen Sie dazu den Browser Ihrer Wahl und verwenden Sie folgende Login-Informationen für die erste Anmeldung:

Login: admin
Password: admin

Nun sollten Sie zunächst das Login-Passwort ändern und die Sprache auf »Deutsch« umstellen. Diese Einstellungen erreichen Sie über das Menü *Ablage – Globale Einstellungen – Sprache bzw. Benutzer* (siehe Abbildung 2-1).

Sollte es Probleme beim Aufrufen von VirtualBox geben, so prüfen Sie, ob die Dienste ordnungsgemäß gestartet sind. Listing 2-7 zeigt eine Ausgabe mittels des netstat-Befehls.

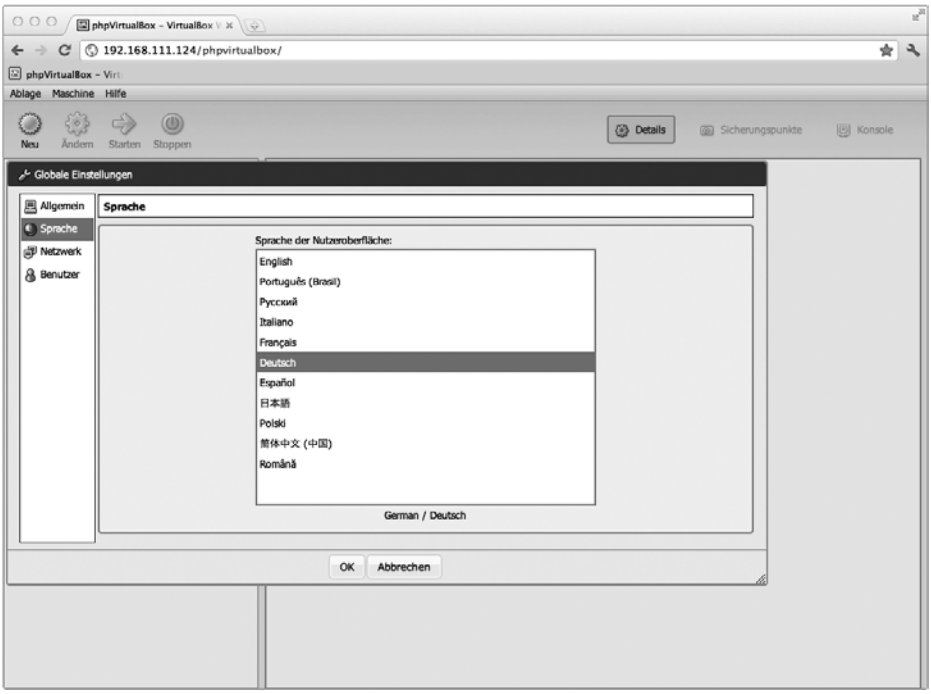

Abb. 2–1 *Globale Einstellungen vornehmen*

```
root@virtualbox:/home/frank# netstat -tulpe
Aktive Internetverbindungen (Nur Server)
Proto  Recv-Q Send-Q Local Address    Foreign Address  State  User         Inode
PID/Program name
tcp    0      0      localhost:18083  *:*              LISTEN virtualbox   5032
960/vboxwebsrv
tcp    0      0      *:www            *:*              LISTEN root         4953
972/apache2
```

Listing 2–7 *Listing 2-7 Apache und VirtualBox sind ordnungsgemäß gestartet*

Hinweis: In den Supportforen schildern einige Nutzer Probleme mit KVM (Kernel-based Virtual Machine) in der Zusammenarbeit mit VirtualBox. Hier wird dringend empfohlen, diese Module abzuschalten. Listing 2-8 zeigt die entsprechende Vorgehensweise.

```
rmmod kvm-intel
rmmod kvm-amd
/etc/init.d/vboxdrv setup
```

Listing 2–8 *KVM abschalten*

Die einzelnen virtuellen Maschinen können über das VirtualBox Remote Desktop Protocol (VRDP) in separaten Fenstern angesprochen werden. Dabei ist VRDP kompatibel mit dem von Windows bekannten Remote Desktop Protocol (RDP). Demzufolge kann jeder beliebige RDP-Viewer genutzt werden. Um dies auszunutzen, muss nur die IP-Adresse des Hostsystems (nicht der virtuellen Maschine (VM)) und der VRDP-Port der jeweiligen VM angegeben werden. Den genutzten VRDP-Port kann man in den Eigenschaften der VM unter *Anzeige – Port für Fernsteuerung* einsehen. Der zuerst gestarteten VM wird dabei der Port 3389 zugeordnet. Alle weiteren virtuellen Maschinen erhalten den jeweils höheren Port bis 4000.

Beispiel: Nutzt man ein Linux-System, so kann man die als Zweites gestartete virtuelle Maschine über folgenden Befehl im Full-Screen-Modus fernsteuern:
`rdesktop –f  192.168.111.124:3390`

2.2 Virtuelle Maschinen erstellen

Um unsere Tests an verschiedenen Systemen vornehmen zu können, sollten virtuelle Maschinen in ausreichender Anzahl erstellt werden. Auch hier erweist sich VirtualBox als sehr flexible Software und lässt praktisch keine Wünsche offen. Haben Sie z.B in der Vergangenheit die verschiedenen Produkte von VMware genutzt, so können Sie die dort erstellten virtuellen Maschinen unter bestimmten Voraussetzungen weiternutzen bzw. importieren.

Um Problemen aus dem Weg zu gehen, sollten Sie Ihre virtuellen Maschinen wenn immer möglich neu erstellen. In den nächsten Abschnitten wird aber zunächst gezeigt, wie Sie VMware-Image in VirtualBox weiternutzen bzw. konvertieren können.

Wenn Sie die Defaulteinstellungen bei der Installation des Servers übernommen haben, sollten die neu erstellten virtuellen Maschinen im Ordner `/home/virtualbox/ VirtualBox VMs/` zu finden sein. In Listing 2–9 werden weitere Verzeichnisse angelegt, in denen wir später ISO-Dateien und VMware-Images ablegen und weiternutzen.

```
sudo su
mkdir /home/virtualbox/ISO
kkdir /home/virtualbox/VMware
cd  /home/virtualbox
chown –R virtualbox:virtualbox  ISO/ VMware/
```

Listing 2–9 *Zusätzliche Verzeichnisse erstellen und Rechte anpassen*