BRAIN-MEDIA.DE

PROFESSIONELLES OPEN-SOURCE- UND LINUX-KNOW-HOW

Holger Reibold

Nessus 3.x kompakt

2., aktualisierte und erweiterte Auflage

Alles Wichtige für den Einsatz des Security-Scanners

Holger Reibold

Nessus kompakt

2., aktualisierte und erweiterte Auflage

© 2008 Brain-Media.de

Herausgeber: Dr. Holger Reibold

Umschlaggestaltung: Brain-Media.de

Satz: Brain-Media.de

Coverbild: Photocase/Lutz Wallroth

Druck: COD

ISBN: 978-3-939316-54-1

Inhaltsverzeichnis

Vorwort

Sicherheitsprodukte haben Hochkonjunktur. Kein Wunder, denn fast täglich kann man der Presse Meldungen über kritische Sicherheitslücken, erfolgreich gehackte Server, Datenklau und Manipulation entnehmen. So steigt die Nachfrage nach Werkzeugen, die helfen, Sicherheitslücken aufzuspüren und zu schließen.

Nessus ist ein solches Tool, das zur Kategorie der Security Scanner gehört. Solche Scanner nehmen beliebige Netzwerkgeräte, Server, Netzwerkdrucker, Router und natürlich auch Einzelplatzrechner unter die Lupe, identifizieren Schwachstellen und geben Hilfestellung beim Schließen von Lücken.

In der Welt der Sicherheitsscanner ist Nessus inzwischen der Quasi-Standard. Kaum ein anderes Programm bietet einen ähnlichen Funktionsumfang. Die neue Version 3.x bringt gegenüber ihrem Vorgänger eine Vielzahl an Neuerungen, aber auch einige weniger schöne Änderungen.

Daher ist es an der Zeit, die Erstauflage von **Nessus kompakt** auf den neuesten Stand zu bringen. Dieses Buch führt Sie in die Arbeit mit Nessus 3.2.1 ein. Es zeigt, wie Sie den Security Scanner einsetzen, wie Sie Tests durchführen und Berichte interpretieren. Außerdem zeigt es Ihnen, wie man eigene Test-Skripts erzeugt. Tipps und Tricks für den Praxiseinsatz sowie interessante Hintergrundinformationen runden das Buch ab.

Viel Erfolg beim Scannen wünscht

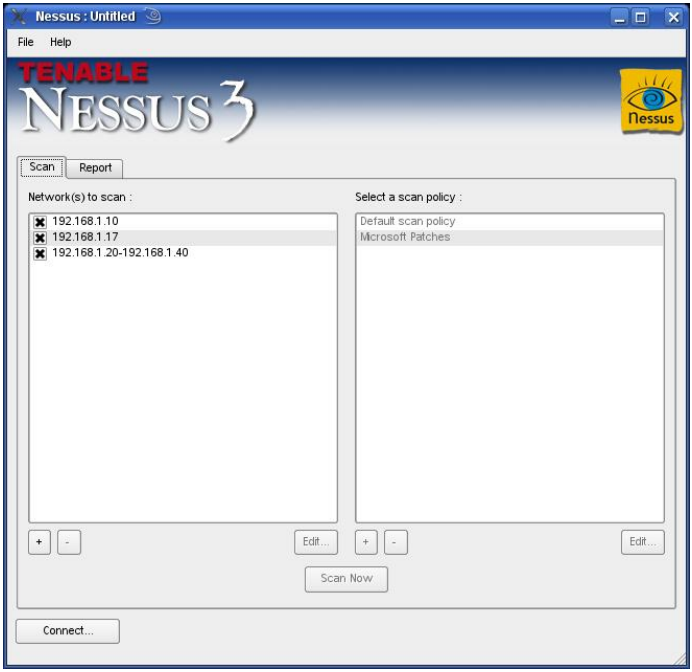
Holger Reibold

(November 2008)

1 Nessus 3.x − der Einstieg

Wenn Sie Ihr Netzwerk und Ihre kritischen Systeme auf ihre Zuverlässigkeit hin überprüfen wollen, müssen Sie diese wohl oder übel echten Attacken aussetzen.

Da Sie sich oder einen Ihrer Mitarbeiter nicht erst zum Hacker ausbilden wollen und können, sollten Sie zu einem Spezialisten greifen, der all das beherrscht. Die Rede ist von einem Securityscanner. Solche Systeme setzen die zutestenden Systeme typischen Attacken aus – oder simulieren dies zumindest.

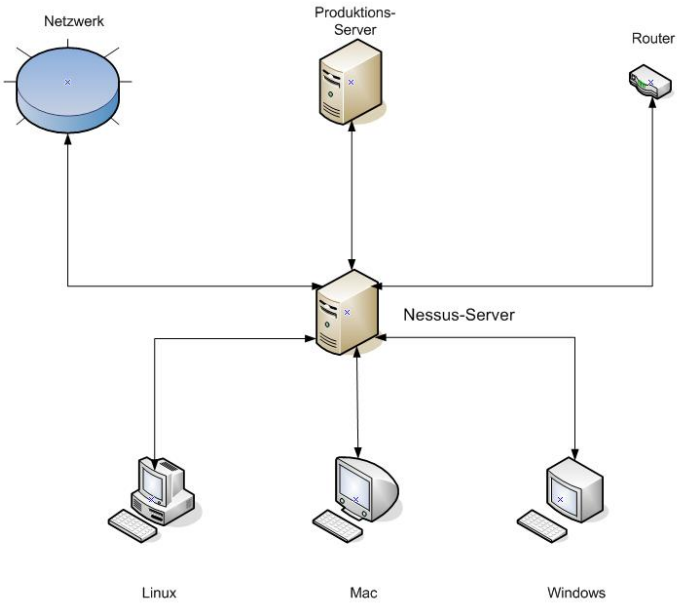


Ein erster Blick auf Nessus 3.x – genauer auf NessusClient 3.2.1, mit dem Sie den Nessus-Server steuern.

Der wohl bekannteste Sicherheitsscanner ist Nessus. Dieses Tool ist ein typischer Vertreter der Netzwerk- oder Vulnerability-Scanner (nicht nur für Linux- und Unix-Systeme).

Das Tool basiert auf einer Client-Server-Architektur. Der Server wird meist auf einem Linux-System (nessusd) ausgeführt und kann von einem lokalen oder auch entfernten Client gesteuert werden. Sowohl der Server als auch die Clients sind für den Mac, Linux und Windows verfügbar.

Beim Start des Servers werden automatisch sogenannte Plug-ins geladen, mit denen sich diverse Sicherheitslücken des Betriebssystems bzw. der Dienste aufdecken lassen, die auf den zu scannenden Hosts laufen. Die Plug-ins werden in der Nessus-eigenen Skriptsprache Nessus Attack Scripting Language (NASL) erstellt.



Die typische Nessus-Architektur: Die Nessus-Clients auf Mac-, Linux- oder Windows-Rechnern steuern den Nessus-Server, der die Sicherheitstests auf beliebigen Systemen ausführt.

Das Prinzip ist ansonsten recht einfach: Der Client stellt eine Verbindung zu dem Server her, erzeugt eine Session, in der die Plug-ins, der oder die Ziel-Hosts und weitere Scan-Einstellungen definiert werden.

Wurde der Scan auf einen Host ausgeführt, gibt der Nessus-Client eine Übersicht über die offenen Ports und eventuell gefundene Sicherheitslücken aus.

Zu erheblichen Verstimmungen in der Fangemeinde haben 2005 die Lizenzänderungen des Scanners geführt. Dieser unterlag der GPL. Ab Version 3.0 wurde Nessus unter eine Hersteller-eigene Lizenz gestellt. Diese sieht zwar weiterhin die freie Nutzung vor, wer aber in den Genuss der vollen Leistungsfähigkeit kommen will, muss eine Art Jahresabo erwerben.

Seitdem bekannt wurde, dass Nessus nicht mehr der GPL unterstellt ist, wurden verschiedene Projekte ins Leben gerufen, um das Tool weiter in einer freien Version verfügbar zu machen. Das bislang beste Ergebnis liefert das OpenVAS (*http://www.openvas.org*), das im August 2007 einen ersten Durchbruch mit der Einführung des Clients in Version 1.0 erzielte.

Andere Projekte wie gNessus sind inzwischen eingestellt. Auch das Bundesamt für Sicherheit in der Informationstechnik hat eine auf Nessus basierende Open-Source-Software entwickelt: BOSS (BSI OSS Security Suite).

1.1 Nessus-Quickstart

Im Zusammenhang mit Security-Scannern taucht auch immer wieder der Begriff des Penetrationstests auf. Ein Penetrationstest sollte jedoch ein zielgerichteter Angriff mit den Mitteln eines Angreifers sein. Angreifer verwenden keine Security-Scanner für ihren Angriff, da diese Programme zu massiv Eintragungen in Logfiles bzw. Intrusion-Detection-Systemen hinterlassen. Ein Penetrationstest ist also der Versuch, mit den Mitteln eines Angreifers und innerhalb einer gegebenen Zeitspanne Lücken in der IT-Sicherheit aufzudecken. Auch hierfür gibt es Hilfsmittel.

Beim Einsatz eines Security-Scanners sind folgende Punkte zu beachten:

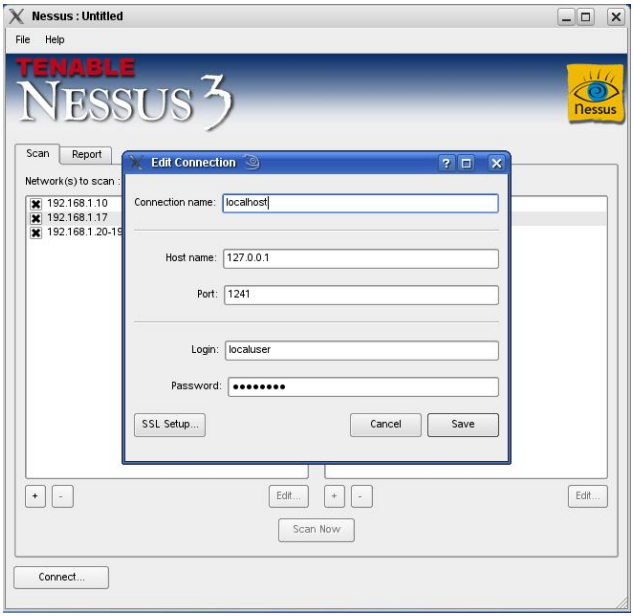
- Portscans können ein System lahmlegen. Daher sollte bei unbekannter Hard- oder Software zuerst ein Test auf Nicht-Produktionssystem stattfinden.
- Kritisch ist der Scan von Produktionssystemen, da ein Ausfall des Systems aufgrund eines Scans nie ganz ausgeschlossen werden kann.
- Besonders kritisch ist eine Überprüfung von relevanten Systemen.

Der Begriff Nessus

Nessus ist als Sohn des Ixion und der Nubes einer der Kentauren/Zentauren. In der griechischen Mythologie ist ein Kentaur ein Mischwesen aus Mensch und Pferd. Sie haben Kopf, Oberkörper und Hände wie ein Mensch, Rumpf, Schweif und Beine wie ein Hengst. Der menschliche Kopf ist mal bärtig, mal bartlos.

Nessus stritt mit Herakles um dessen Gattin Deianira. Als er sie über den Fluss Evenus trug, verliebte er sich in sie. Am anderen Ufer fiel er über sie her, doch Herakles erschoss ihn mit einem Pfeil, der ihn im Rücken traf und durchbohrte. Manche Darstellungen sprechen davon, dass die Göttin Venus Nessus zu dieser Tat getrieben habe, weil Herkules ihr Nebenbuhler um die Gunst des Adonis war.

Unternehmen wir die ersten Schritte. Ist Nessus installiert und der Nessus-Server gestartet, lässt sich mit dem Nessus-Client eine Verbindung zum Server aufbauen. Dazu starten Sie den NessusClient mit dem Befehl *./NessusClient*. Verschaffen wir uns einen ersten Überblick. Der Nessus-Client präsentiert Ihnen einen einfachen Dialog, in dem die zu scannenden Systeme bzw. Netzwerke und die Scan-Richtlinien aufgeführt werden.

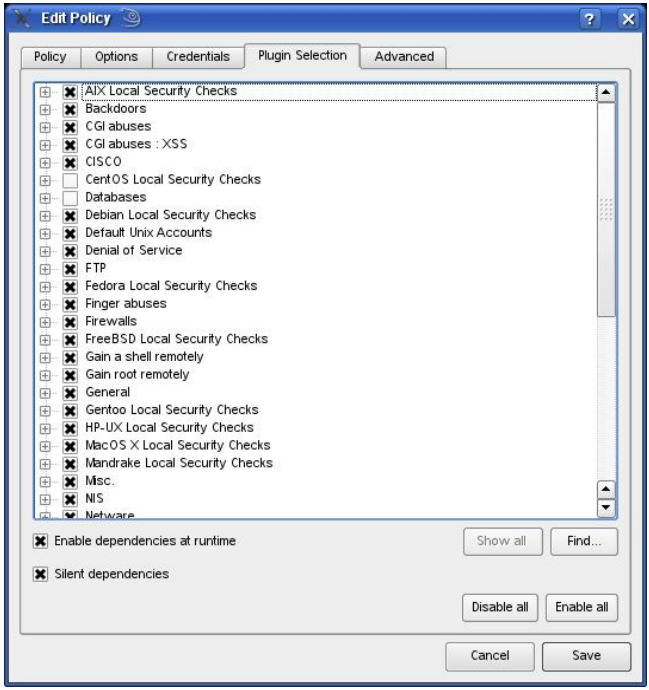


Die Einstellungen für den Verbindungsaufbau zum lokalen Nessus-Server.

Prinzipiell können Sie mit dem NessusClient mehrere Nessus-Server steuern. Um die Verbindung zu einem herzustellen, klicken Sie auf die Schaltfläche *Connect*.

Im sogenannten Connection Manager verwalten Sie die Server-Einstellungen. Sie finden im Verbindungsmanager bereits den Eintrag *localhost*, der die Nutzung eines lokal installierten Servers vorsieht. Markieren und editieren Sie den Eintrag. In den Verbindungseinstellungen geben Sie den Hostnamen, den Standardport des Servers (in der Regel 1241) und die Zugangskennung an.

Wenn Sie Nessus-Server und -Client auf verschiedenen Systemen einsetzen, sollten Sie außerdem die SSL-Einstellungen bearbeiten, um die Verbindung zwischen beiden zu sichern. Nach dem Speichern der Verbindungseinstellungen stellen Sie die Verbindung mit einem Klick auf *Connect* her. Steht die Verbindung, verwandelt sich die Connect- in eine Disconnect-Schaltfläche.



Die Plug-in-Auswahl.

Als Nächstes sind die Scan-Richtlinien dran. Auch hier kommt NessusClient mit zwei vordefinierten Policys daher:

- Default Scan Policy
- Microsoft Patches

Sie können diese Richtlinien über die Schaltfläche *Edit* bearbeiten. Neben den vordefinierten Richtlinien können Sie natürlich auch eigene mit einem Klick auf das Pluszeichen erstellen.

Achtung

Da die Plug-ins vom Server, nicht aber vom Client bereitgestellt werden, ist die Plug-in-Liste nur dann verfügbar, wenn Sie eine Verbindung zu einem Nessus-Server aufgebaut haben.

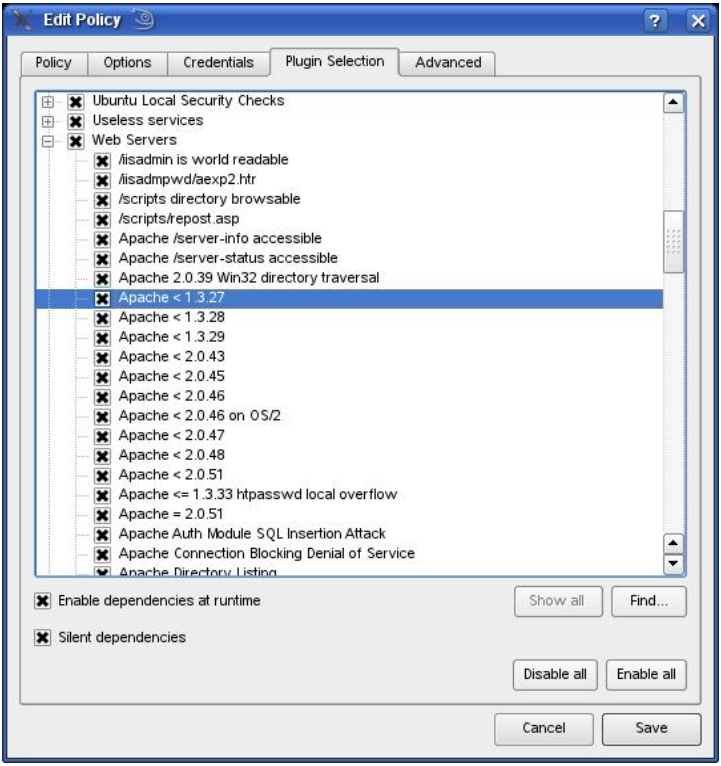
1.2 Das Herzstück: die Plug-ins

Nessus präsentiert Ihnen vier Registerkarten. Auf dem Register *Plugin Selection* finden Sie die Einstellungen für die Plug-ins. Wie bereits erwähnt, handelt es sich bei den Plug-ins um das eigentliche Herzstück des Scanners. Hinter den Plug-ins verbergen sich die Sicherheitstests, die einen möglichen Angriffspunkt entdecken. Im Herbst 2008 gibt es über 40 Plug-in-Kategorien mit mehr als 20.000 Plug-ins. Die Kategorien:

- AIX Local Security Checks
- Backdoors
- CGI abuses
- CGI abuses: XSS
- CISCO
- CentOS local Security Checks
- Databases
- Debian Local Security Checks
- Default Unix Accounts
- Denial of Service

- FTP
- Fedora Local Security Checks
- Finger abuses
- Firewalls
- FreeBSD local Security Checks
- Gain a shell remotely
- Gain root remotely
- General
- Gentoo Local Security Checks
- HP-UX Local Security Checks
- Mac Local Security Checks
- Mandrake Local Security Checks
- Misc.
- Netware
- NIS
- Peer-To-Peer File Sharing
- Red Hat Local Security Checks
- Remote file access
- RPC
- Service Detection
- Settings
- Slackware Local Security Checks
- SMTP problems
- SNMP
- SuSE Local Security Checks
- Ubuntu Local Security Checks
- Useless services

- Web Servers
- Windows
- Windows: Microsoft Bulletins
- Windows: User management



Die der Kategorie *Web Servers* zugeordneten Plug-ins.

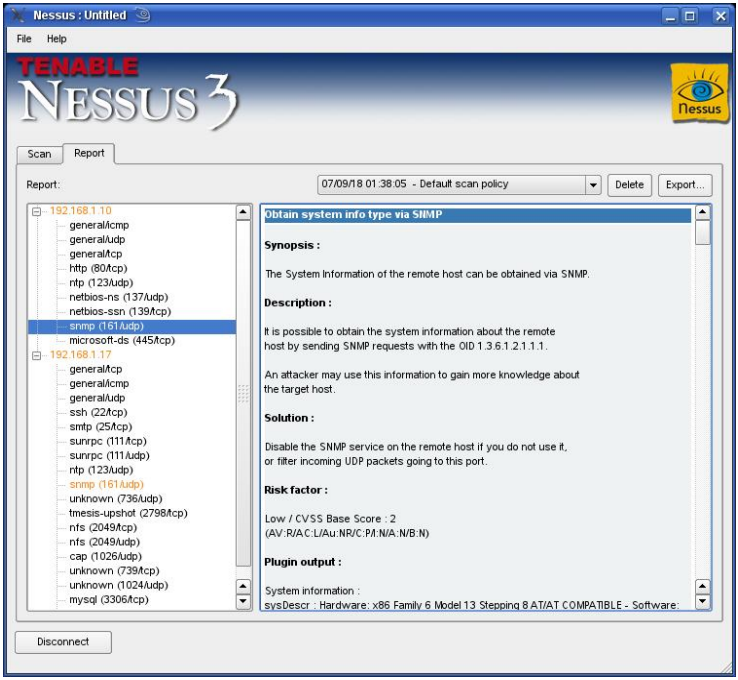
Wenn Sie eine Kategorie mit einem Klick öffnen, erkennen Sie, wie viele Plug-ins einer Kategorie zugeordnet sind. In der Vorgängerversion Nessus 2.x waren immer auch über den Client Details zu den einzelnen Test-Skripts abrufbar. Das ist in der aktuellen Version Nessus 3.2.1 leider nicht mehr der Fall. Hier hilft im Zweifelsfall nur ein Blick auf die Plug-in-Übersicht der Entwickler. Über die Find-Schaltfläche können Sie zudem recht einfach die gewünschten Plug-ins suchen.

Tipp: Alle Nessus-Plug-ins im Überblick

Auf der Plug-in-Seite (*http://cgi.nessus.org/plugins/*) der Nessus-Homepage stehen die Module zum Download bereit. Sie sind nach Kategorien sortiert. Eine Liste der beliebtesten Module und eine Suchfunktion erleichtern die Auswahl neuer Module.

Insbesondere über die Register *Options* und *Advanced* können Sie eine Vielzahl von Einstellungen anpassen. Sichern Sie die Einstellungen mit einem Klick auf *Save*. Sie landen wieder in der NessusClient-Übersicht.

Den eigentlichen Scanvorgang leiten Sie mit einem Klick auf die Schaltfläche *Scan Now* ein. Den Scan-Vorgang zeigt der Client durch rotierende Punkte an. Die Ergebnisse präsentiert Ihnen der Client auf dem Register *Report*.



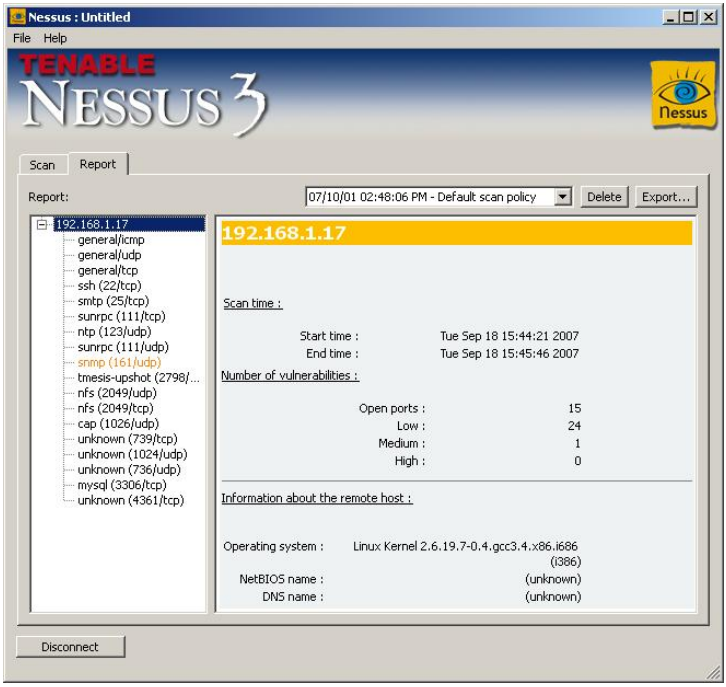
Eine typische Berichtausgabe.

1.3 Nessus-Berichte

Die Berichtfunktion des Nessus-Clients zeigt im linken Bereich die gescannten Systeme an, rechts finden Sie die zugehörigen Detailinformationen.

Kritische Ereignisse werden farbig gekennzeichnet und beispielsweise rot markiert. Die Berichtinfo liefert Ihnen in der Regel die notwendigen Informationen, um die (potenzielle) Schwachstelle schließen zu können.

Sie können die Berichte in zwei Nessus-spezifischen Formaten und als HTML-Daten speichern. Leider scheint die XML-Exportfunktion in Nessus 3.x nicht mehr integriert zu sein. Auch der ASCII-Export ist nun nicht mehr möglich.



Der Windows-Client entspricht exakt der Linux-Variante.

Die Berichtausgabe unterliegt einer weiteren Einschränkung. Bislang ist ein direktes Schreiben der Berichtinformation in eine Datenbank nicht möglich. Dieses Problem löst ein kleines Open-Source-Tool mit der Bezeichnung NNP (*http://www.eless.fi/nnp/*). Es handelt sich um einen Parser, der als CGI-Skript

agiert und NBE-Dateien in SQL-Statements umwandelt, die dann beispielsweise an eine MySQL-Datenbank übergeben und weiter verarbeitet werden können. Es scheint, als dürfte man nicht mit der Weiterentwicklung des Tools rechnen.

Tipp: Professioneller Berichtexport

Der aktuelle Nessus-Client unterliegt gewissen Einschränkungen. So kann man beispielsweise die Scan-Ergebnisse nicht editieren oder bearbeiten, bevor man den Bericht exportiert. Beim Windows-Client NessusWX, dem Vorläufer von NessusClient ist dies möglich. Der Windows-Client kann gar Berichte ins PDF-Format und in MySQL-Statements exportieren. Der Windows-Client unterstützt sogar den Vergleich von Berichten.

Damit haben Sie einen ersten Überblick, was Sie mit Nessus anstellen können. Schauen wir uns als Nächstes an, wie Sie den Server und den Client in Betrieb nehmen.

2 Nessus in Betrieb nehmen

Mit einem soliden Grundverständnis von Nessus sind Sie bestens für die Installation und Konfiguration gerüstet. Der Sicherheitsscanner ist für alle wichtigsten Linux-Distributionen, Solaris, Mac OS und aktuelle Windows-Plattformen verfügbar. Konkret sind es folgende Plattformen:

- Red Hat Enterprise Server 3 und 4
- Fedora Core 1, 3, 4 und 5
- SuSE Linux 9.3 und 10.0
- Debian 3.1
- FreeBSD 5.4 und 6.0
- Solaris 9 und 10
- Mac OS X
- Windows 2000, XP und Server 2003

Für die Installation, Inbetriebnahme und Nutzung genügen in der Regel grundlegende Linux-Kenntnisse und Grundkenntnisse im Bereich Sicherheit und Security Scanning.

2.1 Voraussetzungen

Die Voraussetzungen für die Nutzung des Scanners sind recht bescheiden. Laut Angaben der Entwickler genügt ein Pentium, der mit 256 MB RAM ausgestattet ist, um ein lokales Netzwerk zu scannen. Bei großen Netzen sollten es allerdings schon 1 bis 4 GB sein.

Zu Testzwecken können Sie Nessus auch in einer VMware-Umgebung ausführen. Doch das sollten Sie auch wirklich nur zum Testen tun.

Für all jene, die den Server unter Windows einsetzen wollen, sei darauf hingewiesen, dass der Scanner unter Windows XP mit dem Service Pack 2 sehr langsam arbeitet und unnötige False Negatives produziert. Setzen Sie den Scanner daher lieber auf einer Windows-Server-Plattform ein.

2.2 Installation unter Linux

Bei einer Installation muss man prinzipiell zwischen mehreren Szenarien unterscheiden:

- Neuinstallation
- Upgrade von Nessus 3.x
- Upgrade von Nessus 2.x

Und dann gibt es – wenn auch meist recht geringe – Unterschiede zwischen den verschiedenen Linux-Varianten.

Achtung

Die Installation des Nessus-Servers setzt unter Linux immer Root-Rechte voraus. Melden Sie sich also vor der Installation entsprechend an. Falls Sie einen anderen Benutzer-Account verwenden, passiert entweder nichts oder Sie werden entsprechend hingewiesen.

2.2.1 Neuinstallation unter Linux

Wir befassen uns zunächst mit einer Neuinstallation. Sie finden im Download-Bereich der Nessus-Website die Installationpakete für die verschiedenen Plattformen. Vor jedem Download des Servers ist die Registrierung erforderlich. Für OpenSuse 10.2 lautet die Dateibezeichnung beispielsweise *Nessus-3.0.6-suse10.2.i586.rpm*. Wählen Sie das Paket aus, das zu Ihrer Plattform passt und starten Sie die Installation mit folgendem Kommando:

```
# rpm –ivh Nessus-3.2.1-suse10.2.i586.rpm
```

Dieser Befehl installiert den Scanner in das Verzeichnis */opt/nessus/*. Dabei werden auf Konsolenebene etwa folgende Ausgaben produziert:

```
# rpm –ivh Nessus-3.2.1-suse10.2.i586.rpm

Preparing... ########################################### 
[100%]

1:Nessus ########################################### [100%]

nessusd (Nessus) 3.2.1 for Linux
```

```
(C) 2005 Tenable Network Security, Inc.

Processing the Nessus plugins...

[##########################################]

All plugins loaded

- Please run /opt/nessus/sbin/nessus-add-first-user to add an
admin user

- Register your Nessus scanner at
http://www.nessus.org/register/ to

obtain all the newest plugins

- You can start nessusd by typing /sbin/service nessusd start

#
```

Wenn Sie mit Debian arbeiten, so verwenden Sie folgenden Befehl:

```
dpkg -i Nessus-3.2.1-debian3_i586.deb
```

Auch bei diesem Kommando wird Nessus in das Verzeichnis */opt/nessus/* installiert. Die typische Ausgabe sieht wie folgt aus:

```
# dpkg –i Nessus-3.2.1-debian3_i586.deb

(Reading database ... 10027 files and directories currently
installed.) Unpacking nessus (from Nessus-3.0.6-
debian3_i586.deb) ...

Setting up nessus (3.2.1) ...

nessusd (Nessus) 3.2.1 for Linux (C) 2005 Tenable Network
Security, Inc.

/opt/nessus/var/nessus/CA created /opt/nessus/com/nessus/CA
created

nessusd (Nessus) 3.2.1 for Linux (C) 2005 Tenable Network
Security, Inc.

Processing the Nessus plugins...
[##########################################]

All plugins loaded

- Please run /opt/nessus/sbin/nessus-add-first-user to add an
admin user - Register your Nessus scanner at
http://www.nessus.org/register/ to
```

```
obtain all the newest plugins - You can start nessusd by typ-
ing /etc/init.d/nessusd start
#
```

Auf die Installation auf weiteren Plattformen wollen wir hier verzichten. Schauen wir uns lieber noch die Vorgehensweise beim Upgrade an.

2.2.2 Upgrade-Varianten

Wenn Sie bereits mit einer mehr oder minder aktuellen Nessus-Installation arbeiten, können Sie auch ein Upgrade durchführen. Besonders einfach ist das, wenn Sie mit einer Nessus-3.x-Installation arbeiten, denn hier können Sie auch die bestehenden Konfigurationen übernehmen.

Laden Sie sich zunächst die aktuellste Version auf Ihren Nessus-Rechner herunter. Vor der eigentlichen Installation sollten Sie erst den Nessus-Daemon beenden:

```
# killall nessusd
```

Dieser Befehl stoppt ihn abrupt, ohne dass weitere Scans durchgeführt werden.

Installieren Sie dann mit folgendem Kommando die neue Version:

```
# rpm -Uvh <neustes_release>
```

Nach der Installation starten Sie die neu installierte Version mit dem bekannten Befehl:

```
# /opt/nessus/sbin/nessusd –D
```

Die Vorgehensweise ist bei allen Linux-Distributionen identisch. Unterschiede gibt es lediglich bei dem auszuführenden Befehl.

Auch ein Upgrade von Nessus 2.x auf Nessus 3.x ist möglich, allerdings mit ein wenig mehr Aufwand als bei einer Neuinstallation.

Da die neue Nessus-Version in ein anderes Installationsverzeichnis (*/opt/nessus/*) kopiert wird, müssen Sie zunächst die Dateien der Vorgängerversion manuell in dieses Verzeichnis kopieren. Die Vorgängerversion finden Sie meist unter */usr/local/nessus*.

Beenden Sie dann die Ausführung der Vorgängerversion:

```
# killall nessusd
```

Im nächsten Schritt installieren Sie die aktuelle Version entsprechend voranstehender Beschreibung.

Dann müssen Sie die Nessus-2.x-Benutzer zu Benutzern der aktuellen Version machen. Da das Benutzermanagement von Nessus übernommen wird, müssen Sie die Benutzer einfach verschieben. Das geschieht mit folgendem Kopierbefehl:

```
# cp –r /usr/local/var/nessus/users/*
/usr/local/nessus/var/nessus/users/
```

Für das vollständige Upgrade sind weitere Aktionen erforderlich: Als Nächstes müssen Sie die Datei *nessus-fetch.rc* in das neue Nessus-Verzeichnis kopieren, um die Übernahme des Aktivierungscodes sicherzustellen. Der entsprechende Befehl sieht wie folgt aus:

```
# cp /usr/local/etc/nessus/nessus-fetch.rc
/usr/local/nessus/etc/nessus/
```

Stellen Sie dann sicher, dass die Dateiberechtigungen korrekt sitzen:

```
-rw------- 1 root root 398 Oct 2 10:00 nessus-fetch.rc
```

Als Nächstes sind einige Anpassungen der Nessus-Konfigurationsdatei erforderlich. Sie finden sie meist unter */usr/local/nessus/etc/nessus/nessusd.conf*. Hier müssen Sie sicherstellen, dass der Benutzer *admin_user* korrekt eingetragen ist. Die korrekten Einstellungen sehen wie folgt aus:

```
plugin_upload = yes
admin_user = <ADMIN>
```

Dabei ist <*ADMIN*> der Benutzername des von Ihnen eingerichteten Nessus-Server-Administrators in der Nessus-2.x-Konfigurationsdatei */usr/local/etc/nessus/nessud.conf*.

Prinzipiell sind Sie damit am Ende des Upgrade-Vorgangs angelangt und können im nächsten Schritt Nessus 3.x konfigurieren und den Nessus-Server starten.

Nachdem Sie sichergestellt haben, dass die neue Installation korrekt arbeitet, sollten Sie die Vorgängerversion mit folgendem Kommando deinstallieren:

```
# /usr/local/sbin/uninstall-nessus
```

Nachdem Sie Nessus – gleich auf welchem Weg – installiert haben, ist als Nächstes die Konfiguration dran.

2.2.3 Konfiguration

Der Nessus-Server wird über einen Client angesprochen und gesteuert. Das setzt voraus, dass der Nessus-Server zumindest einen einzigen Benutzer kennt, der ihn steuern darf.

Um einen ersten Nessus-Benutzer zu erzeugen, führen Sie auf dem Nessus-Server-System folgenden Befehl aus:

```
nessus-add-first-user
```

Wenn Sie ein Upgrade durchführen und die bestehenden Benutzer für die neue Version nutzbar gemacht haben, verwenden Sie den Befehl, um einen weiteren Benutzer einzurichten:

```
nessus-adduser
```

Der Adduser-Befehl verlangt einige Eingaben wie den Benutzernamen und das Passwort. Hier ein Beispiel für den typischen Vorgang beim Erzeugen eines ersten Benutzers:

```
# /opt/nessus/sbin/nessus-add-first-user
nessusd (Nessus) 3.2.1 for Linux
(C) 2005 Tenable Network Security, Inc.
Using /var/tmp as a temporary file holder
Add a new nessusd user
--------------------
Login : admin
Authentication (pass/cert) [pass]:
Login password:
```

```
Login password (again):

User rules

----------

nessusd has a rules system which allows you to restrict the
hosts that admin has the right to test. For instance, you may
want

him to be able to scan his own host only.


Please see the nessus-adduser(8) man page for the rules syn-
tax.


Enter the rules for this user, and hit ctrl-D once you are
done:

(the user can have an empty rules set)

Login :admin

Password :********

DN :

Rules :

Is that ok ? (y/n) [y]

User added.

Thank you. You can now start Nessus by typing:

/opt/nessus/sbin/nessusd –D

#
```

Weitere Benutzer werden mit folgendem Kommando erzeugt:

```
# /opt/nessus/sbin/nessus-adduser
```

Wie Sie als Nächstes fortfahren ist sicherlich von Ihren Kenntnissen und Anforderungen abhängig. Der eine Benutzer interessiert sich für weitere Konfigurationsmöglichkeiten, die der Nessus-Daemon zu bieten hat, ein anderer will lieber direkt erste Tests starten. Wieder ein anderer befasst sich mit den verschiedenen Plug-in-Optionen.

2.2.4 Die Nessus-Konfigurationsdatei

Wenn Sie zu den eher technisch interessierten Anwendern gehören, interessieren Sie sich vermutlich für die Einstellungen, die Ihnen die Nessus-Server-Konfigurationsdatei *nessusd.conf* bietet. Damit Sie einen ersten Eindruck von den Möglichkeiten und vielfältigen Einstellungen bekommen, hier eine typische Beispieldatei mit den notwendigen Erläuterungen:

```
# Konfigurationsdatei der Nessus Security
# Scanner

# Pfad zu dem Sicherheitscheck-Ordner:
plugins_folder = /opt/nessus//lib/nessus/plugins

# Plug-ins werden automatisch aktualisiert.
plugins.nessus.org automatically
auto_update = yes

# Anzahl an Stunden zwischen zwei
# Update-Vorgängen.
auto_update_delay = 24

# Soll die Plug-in-Datenbank bei jedem
# Update bereinigt werden?
purge_plugin_db = no

# Maximale Anzahl an gleichzeitigen Test.
max_hosts = 40

# Maximale Anzahl an Tests pro Host.
max_checks = 5

# “Niedlichkeit.
```

```
be_nice = no

# Drossle Scan, wenn die CPU überlastet ist?
throttle_scan = yes

# Protokolldateien:
logfile = /opt/nessus//var/nessus/logs/nessusd.messages

# Sollen alle Details einer Attacke
# protokolliert werden? Die Aktivierung
# ist sehr speicherplatzintensiv.
log_whole_attack = no

# Protokollierung der Plug-ins, die vom
# Server geladen werden?
log_plugins_name_at_load = no

# Dump-Datei für den Debug-Output.
dumpfile = /opt/nessus//var/nessus/logs/nessusd.dump

# Pfad zur Regeldatei.
rules = /opt/nessus//etc/nessus/nessusd.rules

# Benutzerdatenbank
users = /opt/nessus//etc/nessus/nessusd.users

# CGI-PFaf für Prüfungen.
cgi_path = /cgi-bin:/scripts

# Port-Bereich, den es zu scannen gilt.
```

```
# Dabei bedeutet default, dass Nessus die
# Ports scannt, die in der Service-Datei
# gefunden werden.
port_range = default

# Optimiert die Tests (empfohlen).
optimize_test = yes

# Sprache der Plug-ins.
language = english

# Optimierung:
# Liest den Timeout-Wert für die Sockets.
checks_read_timeout = 5

# Ports, gegen die nicht zwei Plug-ins
# gleichzeitig attackieren sollen.
non_simult_ports = 139, 445

# Maximale Lebensdauer eines Plug-ins
# in Sekunden.
plugins_timeout = 320

# Sichere Checks, das bedeutet, dass
# System nicht zum Absturz gebracht werden
# soll.
safe_checks = yes

# Aktiviert automatisch die abhängigen
# Plug-ins. Es gibt Tests, die auf
```

```
# einander aufbauen.
auto_enable_dependencies = yes

# Eine weitere Abhängigkeitseinstellung.
silent_dependencies = yes

# Bestimmt Hosts durch Ihre MAC- und nicht
# durch Ihre IP-Adresse.
use_mac_addr = no

# -- Einstellungen für die Knowledgebase --
# Sichert die Knowledgebase auf Festplatte.
save_knowledge_base = no

# Stellt die KB für jeden Test wieder her.
kb_restore = no

# Nur die Hosts, deren KB nicht
# vorhanden ist.
only_test_hosts_whose_kb_we_dont_have = no

# Nur die Hosts, deren KB bereits
# verfügbar ist.
only_test_hosts_whose_kb_we_have = no

# KB-Testwiederholung
kb_dont_replay_scanners = no
kb_dont_replay_info_gathering = no
kb_dont_replay_attacks = no
kb_dont_replay_denials = no
```

```
kb_max_age = 864000

# -- Ende der KB-Konfiguration

# Können Benutzer Ihre Plug-ins uploaden?
plugin_upload = yes

# Suffixe der Plug-ins, die Benutzer
# hochladen können.
plugin_upload_suffixes = .nasl, .nasl3, .inc, .inc3, .nbin,
.audit

# Name der Benutzer, die Plug-ins aus
# der Ferne hochladen können.
admin_user = holger

# Wenn Sie diese Option aktivieren, führt
# Nessus keine inkrementellen Scans durch
# (also 10.0.0.1, dann 10.0.0.2, 10.0.0.3
# und so weiter), sondern versucht die Last
# auf das gesamte Netzwerk zu verteilen
# (also 10.0.0.1, dann 10.0.0.127, dann
# 10.0.0.2, dann 10.0.0.128 und so weiter).
slice_network_addresses = no

# IP-Adresse, auf die der Server für
# eingehende Verbindungen hört.
listen_address = 0.0.0.0

# Sollen alle NASL-Skripts als signiert
# behandelt werden?
```

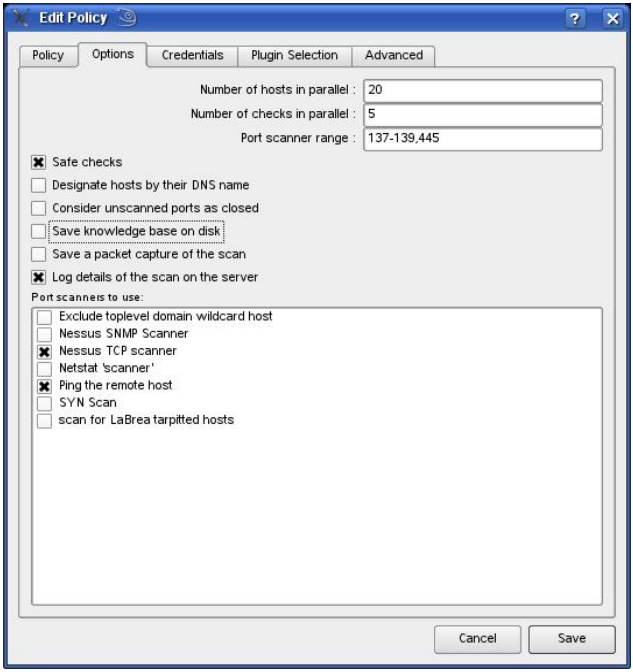
```
nasl_no_signature_check = no

# Dieser Abschnitt wird durch nessus-mkcert
# hinzugefügt.
cert_file=/opt/nessus//com/nessus/CA/servercert.pem
key_file=/opt/nessus//var/nessus/CA/serverkey.pem
ca_file=/opt/nessus//com/nessus/CA/cacert.pem

# Wenn Sie den privaten Schlüssel durch
# ein Passwort schützen wollen. Eun If you
pem_password=password

# Erzwingt die Verwendung eines Client-
# Zertifikats.
force_pubkey_auth = yes
```

Wie Sie voranstehender Beispielkonfiguration entnehmen können, ist über die Nessus-Konfigurationsdatei eine Vielzahl an Einstellungs- und Anpassungsmöglichkeiten geboten. Einige besonders Wichtige bedürften der Erläuterung.

Die beiden wichtigsten Einstellungen für die Ausführung von Sicherheitsscans sind *max_hosts* und *max_checks*. Wie Sie obiger Beispielkonfiguration entnehmen können, werden in der Regel nicht mehr als 40 Systeme und 5 Tests pro Ziel gleichzeitig ausgeführt.

Tipp

Diese Voreinstellungen können durch das kommerzielle Security Center von Tenable oder durch einen Client wie NessusClient oder den Vorgänger NessusWX überschrieben werden.



Auf dem Register *Options* des NessusClients bestimmen Sie die maximale Anzahl an Hosts, die der Server prüft.

Prinzipiell ist die Anzahl der Systeme, die ein Nessus-Server prüfen kann, von der Leistungsfähigkeit der eigenen Hardware abhängig. Daneben gibt es weitere Faktoren, die die Scan-Möglichkeiten beeinflussen, beispielsweise Sicherheitsrichtlinien für die Nutzung von internen Systemen etc. Hier ist etwa darauf zu achten, dass kritische Systeme nicht harten Attacken ausgesetzt werden.

Die Entwickler empfehlen generell, konservativ an die Sache heranzugehen. Wenn Sie Nessus auf einem typischen Linux-Server ausführen, so sollten Sie die Anzahl der gleichzeitigen Tests auf 20 setzen. Wenn Sie Ihre Tests mit einem Windows-System ausführen, sollten Sie bei maximal zehn zu scannenden Hosts beginnen.

Ähnlich zurückhaltend sollten Sie bei der Anzahl an gleichzeitigen Tests pro Host sein. Beginnen Sie hier bei drei oder vier.

2.2.5 Nessus-Server starten und anhalten

Sind Sie mit der Grundkonfiguration des Systems zufrieden, können Sie den Nessus-Server in Betrieb nehmen. Dazu führen Sie den bereits vorgestellten Startbefehl aus:

```
# /opt/nessus/sbin/nessusd –D
```

Soll der Scanner bei jedem Systemstart automatisch gestartet werden, so fügen Sie obigen Befehl in das Startskript ein:

```
/etc/rc.d/rc.local
```

Auf die Ausgabe beim Server-Start müssen wir nicht mehr eingehen. Sie ist bereits zuvor beschrieben. Wichtig ist, dass Sie Nessus als Root ausführen, denn nur Root kann den Scanner starten und beenden.

Abhängig von dem von Ihnen verwendeten Betriebssystem können Sie auch folgende Kommandos verwenden:

- Red Hat: # /sbin/service nessusd start
- SUSE: # /etc/rc.d/nessusd start
- Debian: # /etc/init.d/nessusd start
- FreeBSD: # /usr/local/etc/rc.d/nessusd.sh start
- Solaris: # /etc/init.d/nessusd start

Auch das Stoppen des Servers wurde bereits erwähnt. Hier der Vollständigkeit halber noch einmal der zugehörige Befehl:

```
# killall nessusd
```

2.2.6 Kommandozeilenoptionen des Nessus-Servers

Für die Steuerung des Nessus-Servers stehen Ihnen einige weitere Kommandozeilenoptionen zur Verfügung. Hier ein Beispiel:

```
# /opt/nessus/sbin/nessusd [-vhD] [-c <konfigdatei>] [-p
<port-nummmer>] [-a <adresse>] [-S <ip[,ip,...]>]
```

Nachstehende Tabelle fasst die Optionen und ihre Bedeutung zusammen:

Option	Beschreibung
-c <konfigdatei>	Beim Start können Sie auch eine alternative Konfigurationsdatei angeben, um mit einer anderen Konfiguration ergänzende Informationen zu erzielen.
-a <adresse>	Hier können Sie beim Start bereits die IP-Adresse bestimmen, auf die der Server anspricht.
-S <ip1, ip2, …>	Hiermit erzwingen Sie bestimmte IP-Adressen.
-p <portnummer>	Wenn Sie eine alternative Port-Nummer verwenden wollen, so definieren Sie diese mit diesem Schalter.
-D	Hiermit starten Sie den Server im Daemon-, also Hintergrundmodus.
-v	Dieser Schalter gibt die Versionsnummer aus und beendet sich.
-h	Dieser Schalter gibt die Hilfetexte aus.

2.2.7 Aktivierungscode installieren

Wenn Sie eine kommerzielle Version erstanden haben, so müssen Sie den Aktivierungscode für die Plug-ins einrichten. Dessen Installation erfolgt mit folgendem Befehl:

```
# /opt/nessus/bin/nessus-fetch --register <license code>
```

Wichtig bei der Aktivierung ist, dass eine Internet-Verbindung verfügbar ist, die die Korrektheit der Daten überprüft. Es geht zwar auch ohne, doch das ist etwas umständlich.

Hier ein Beispiel für die typische Vorgehensweise bei der Aktivierung Ihrer Lizenz:

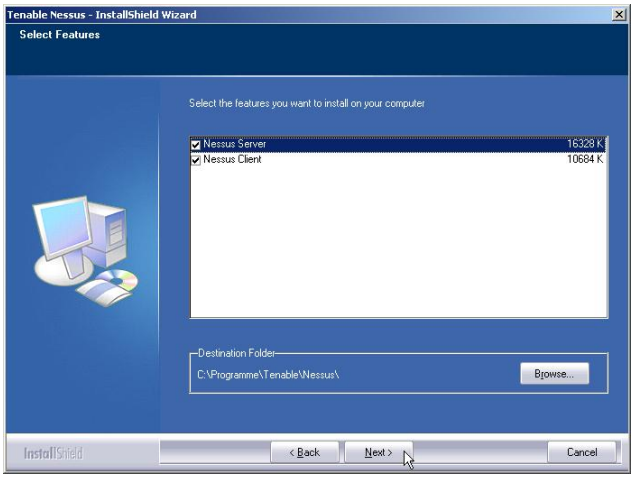
```
# cat /opt/nessus/lib/nessus/plugins/plugin_feed_info.inc

PLUGIN_SET = "200810021500";

PLUGIN_FEED = "Release";

# /opt/nessus/bin/nessus-fetch --register XXXX-XXXX-XXXX-
XXXX-XXXX
```

```
Your activation code has been registered properly – thank
you.
Now fetching the newest plugin set from plugins.nessus.org...
Your Nessus installation is now up-to-date.
If auto_update is set to 'yes' in nessusd.conf, Nessus will
update the plugins by itself.
# date
Tue Oct 02 15:00:00 EDT 2008
# cat /opt/nessus/lib/nessus/plugins/plugin_feed_info.inc
PLUGIN_SET = "200810021500";
PLUGIN_FEED = "Direct";
```

In der Datei *plugin_feed_info.inc* im Verzeichnis */opt/nessus/lib/nessus/plugins/* ist hinterlegt, welche Plug-ins verfügbar sind. Hier können Sie prüfen, ob die neuesten Plug-ins bei Ihrer Installation verfügbar sind.

2.3 Nessus unter Windows in Betrieb nehmen

Die Installation von Nessus unter Windows ist einfach. Laden Sie sich zunächst das Programm von der Tenable-Website. Starten Sie dann mit einem Doppelklick auf *Nessus-3.2.1.ex*e das Installationsprogramm. Nach der Wahl des Zielverzeichnisses bestimmen Sie als Nächstes, ob Sie den Server, den Client oder beide installieren wollen.

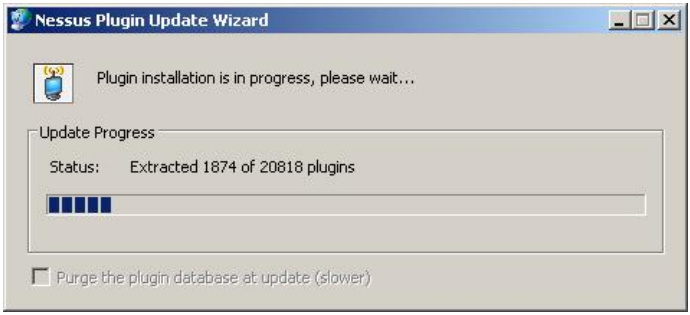


Die Installationsroutine von Nessus 3.2.1 erlaubt die Installation des Nessus-Clients und -Servers in einem Zug.

Achtung

Für all jene, die den Server unter Windows einsetzen wollen, sei darauf hingewiesen, dass der Scanner unter Windows XP mit dem Service Pack 2 womöglich langsam arbeitet und unnötige False Negatives produziert. Setzen Sie den Scanner daher lieber auf einer Windows-Server-Plattform ein.

Im nächsten Schritt holt sich das Installationsprogramm die neuesten Plug-ins von der Website der Entwickler. Diese werden auf das lokale System heruntergeladen und dann installiert. Bei Version 3.2.1 waren es über 20.800 Testskripts – eine beeindruckende Zahl.



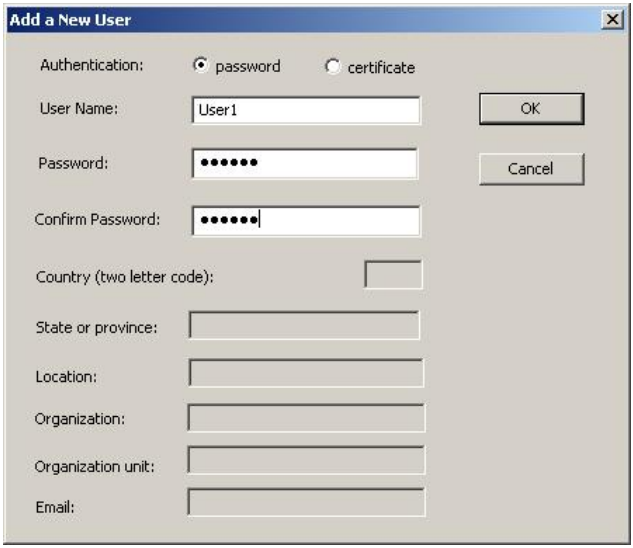
Die Installation des Plug-ins.

Zum Abschluss der Plug-in-Installation präsentiert Ihnen Nessus eine Erfolgsmeldung und Sie können zum ersten Mal den Server über die Windows-Startleiste starten. Sie finden die relevanten Tools unter *Start> Programm> Tenable Network Security*. Hier finden Sie drei zentrale Einträge:

- **Nessus Client**: Startet den Client für den Zugriff auf den Server.
- **Nessus Server Configuration**: Öffnet einen einfachen Konfigurationsdialog, mit dem Sie die Einstellungen anpassen können.
- **User Management**: Öffnet die Nessus-Benutzerverwaltung.

In der Grundkonfiguration kommt Ihre Nessus-Installation ohne einen Benutzer daher. Also müssen Sie zunächst die Benutzerverwaltung starten und zumindest einen ersten Benutzer erstellen. Dazu führen Sie den Befehl *Start> Programm> Tenable Network Security> User Management* aus.

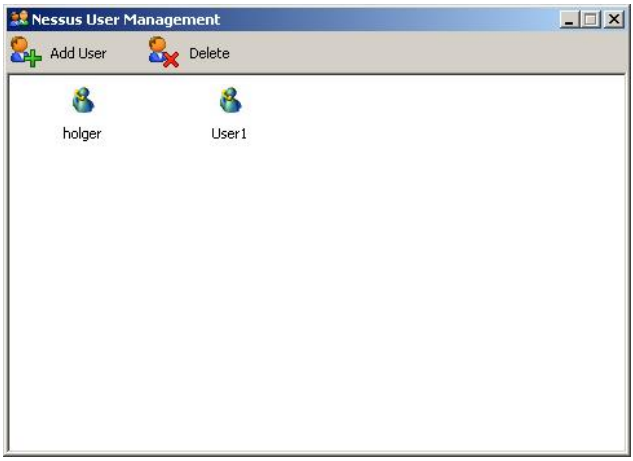
Wie Sie der einfachen Benutzerverwaltung entnehmen können, besitzt Ihre Nessus-Installation nicht einmal einen Admin-Account.



Ein erster Benutzer-Account für Ihre Nessus-Installation entsteht.

Um einen ersten Nessus-Benutzer anzulegen, klicken Sie auf das Add-User-Icon. Im zugehörigen Dialog geben Sie den gewünschten Benutzernamen und das Passwort an. Wenn Sie die Authentifizierungsmethode *Certificate* verwenden, so sind neben dem Benutzernamen verschiedene Adressdaten anzugeben.

Der neue Account ist nach dem Sichern in der Benutzerverwaltung zu finden. Sie können quasi beliebig viele Benutzer erzeugen, doch sollten Sie nur so viele einrichten, wie Sie tatsächlich benötigen.

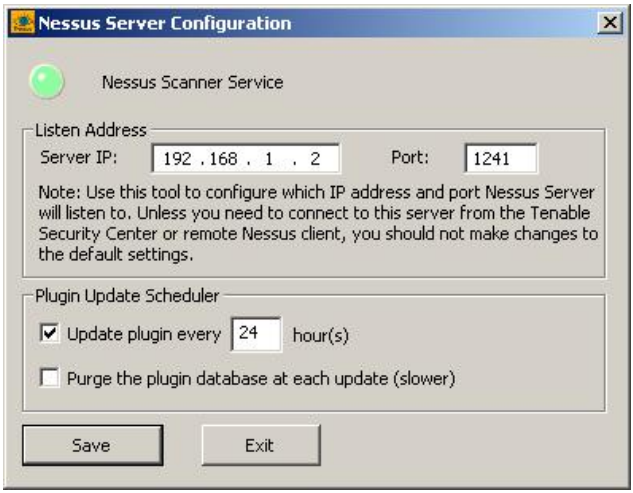


Die Nessus-Benutzerverwaltung mit den ersten beiden Benutzern.

Der nächste Schritt dient der Konfiguration des Nessus-Servers. Hierzu führen Sie den Befehl *Start> Programm> Tenable Network Security> Nessus Server Configuration* aus.

Die Handhabung der Server-Konfiguration ist ebenfalls sehr einfach. Hier geben Sie unter *Listen Address* die IP-Adresse des Nessus-Servers an. Der Nessus-Server verwendet standardmäßig den Port 1241. Diese Daten benötigen Sie später bei der Client-Konfiguration, damit dieser eine Verbindung zum Server aufbauen kann.

Der zweite Bereich der Server-Konfiguration dient der Einrichtung der Plug-in-Updates. Der Nessus-Server ist standardmäßig so konfiguriert, dass Updates alle 24 Stunden geprüft und gegebenenfalls installiert werden.



Die Konfiguration des Nessus-Servers.

Mit einem Klick auf *Save* übernehmen Sie Ihre Einstellungen. Schließen Sie die Server-Einstellungen mit einem Klick auf *Exit* ab.

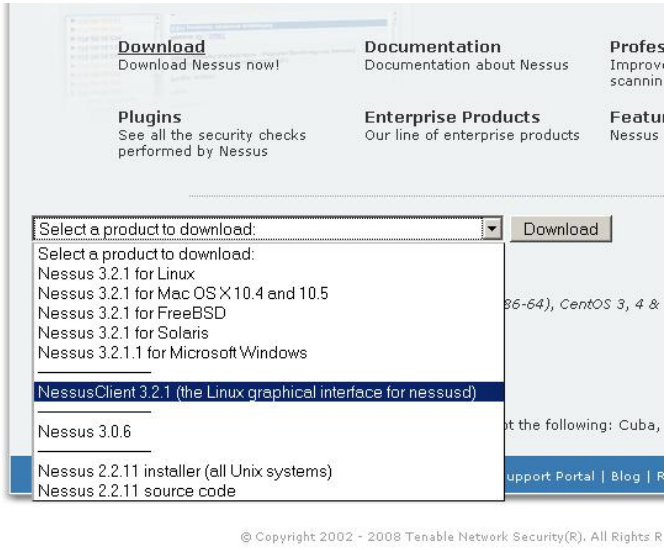
Im Unterschied zur Linux-Variante gibt es bei Nessus für Windows keine textbasierte Konfigurationsdatei. Vielmehr sind die Konfigurationseinstellungen binär gespeichert. Das ist schade, denn so ließen sich auch weitergehende Änderungen an der Server-Konfiguration vornehmen, ohne dass ein Zugriff mit dem Client erfolgt. Alle weiteren Server-Einstellungen sind nur über einen Client anpassbar.

Die Server-Konfiguration zeigt Ihnen durch das farbige runde Symbol im Kopfbereich übrigens den Status des Systems an. Grün steht für *In Betrieb*, rot für *Außer Betrieb* und orange für einen Neustart.

3 Der optimale Client: NessusClient

Nachdem Sie den aktuellen Nessus-Server installiert haben, benötigen Sie einen Client, um diesen zu steuern. Die Nessus-Umgebung setzt, wie bereits erwähnt, auf eine typische Client-Server-Architektur. Sie können den Nessus-Server von jedem beliebigen Client aus steuern – sofern ein entsprechender Client verfügbar ist. Sie können Ihren Linux-basierten Server also beispielsweise von einem Mac-, Linux- oder Windows-Client aus führen.

Wenn Sie den Server von einem Linux-System steuern wollen – womöglich auch von dem gleichen System, auf dem der Server installiert ist –, so müssen Sie den sogenannten NessusClient installieren, der von Tenable stammt. Sie finden diesen im Download-Bereich der Nessus-Homepage.



Der Download des Linux-Clients für die Steuerung des Nessus-Servers.

Neben der Steuerung mit dem neuen NessusClient können Sie Nessus auch mit alternativen Clients nutzen. Dazu gibt es mehrere Möglichkeiten:

- Sie nutzen den Vorläufer in Version 2.x. Das bringt verschiedene Vorteile. So können Sie beispielsweise die Knowlegdebase-Funktion des Nessus-Servers nutzen. Diese ist im NessusClient (leider) nicht mehr verfügbar.
- Sie können den im Windows-Paket enthaltenen Client nutzen. Damit steht Ihnen ein versionsgleicher Client zur Verfügung.
- Sie können einen alternativen Client wie Nessconnect (*http://sourceforge.net/projects/nessconnect/*) verwenden. Dieser Client ist mein Favorit, weil er den beachtlichsten Funktionsumfang bietet. So können Sie beispielsweise Vergleiche zwischen zwei Scans durchführen. Nessconnect ist für Mac, Linux und Windows verfügbar.
- Schließlich könnten Sie beispielsweise den inzwischen (ebenfalls leider) eingestellten Windows-Client NessusWX (*http://nessuswx.nessus.org/archive/*) verwenden. Der Vorteil von diesem Tool: NessusWX bietet verschiedene Exportfunktionen für die Berichte, auch eine PDF-Exportfunktion. Auch der Vergleich zweier Scans miteinander ist möglich. Nachteil: Das Tool wird seit 2005 nicht mehr weiterentwickelt. Daher kann es zu Problemen bei der Ausführung kommen. Profis, die damit umzugehen verstehen, können Änderungen am offenen Quellcode vornehmen.

Damit stehen Ihnen ausreichend Möglichkeiten für die Steuerung des Nessus-Servers zur Verfügung. Im Folgenden konzentrieren wir uns zunächst auf die Nutzung von NessusClient.

3.1 NessusClient installieren

Wenn Sie den Nessus-Server mit NessusClient steuern wollen, müssen Sie zunächst dieses Tool installieren. Der NessusClient ist, wie bereits erwähnt, nur für Linux-Plattformen verfügbar. Laut Tenable ist die Ausführung des Clients auf folgenden Plattformen möglich:

- Debian 4
- Red Hat Enterprise Server 4 und 5
- Fedora Core 7, 8 und 9

- SuSE 10.3
- Ubuntu 7.10 und 8.04

Aus eigenen Tests kann ich anmerken, dass der Client auch unter OpenSuSE 11 arbeitet.

Für die Ausführung muss außerdem das X-Window-System (X11) auf dem Client installiert sein, damit NessusClient ausgeführt werden kann.

Die RPM-Pakete für die verschiedenen Linux-Varianten stehen auf der Nessus-Website zum Download bereit. Sie haben eine Größe von ca. 5 MB. Ihre Bezeichnung besitzt folgendes Format:

```
NessusClient-X.X.X-OS.hardware.format
```

Das NessusClient-RPM für den Red Hat Enterprise Server 4 trägt demnach die Bezeichnung *NessusClient-3.2.1-es4.i386.rpm*.

Wenn Sie nun den NessusClient auf einem SuSE-System installieren wollen, so loggen Sie sich als Root-User ein, laden die SuSE-Varianten herunter und führen folgenden Befehl aus:

```
# rpm -ivh NessusClient-3.2.1-suse10.3.i586.rpm
```

Für (K)Ubuntu 8.04 entsprechend:

```
# rpm -ivh NessusClient-3.2.1-(k)ubuntu804.i386.deb
```

Dieser Befehl erzeugt das Verzeichnis */opt/nessus* und installiert den Client hinein. Das können Sie mit folgendem Kommando prüfen:

```
# ls -l /opt/nessus
total 16
drwxr-xr-x 2 root root 4096   17, Okt 07:34 bin
drwxr-xr-x 2 root root 4096   17, Okt 07:34 lib
drwxr-xr-x 3 root root 4096   17, Okt 07:34 plugins
drwxr-xr-x 3 root root 4096   17, Okt 07:34 var
```

Der Nessus-Server wird standardmäßig übrigens in das gleiche Verzeichnis installiert. Ist er bereits installiert, sieht die Ausgabe natürlich ein wenig anders aus, weil insbesondere mehr Dateien in dem Verzeichnis zu finden sind.

Wenn Sie bereits eine frühere Version von NessusClient einsetzen und ein Upgrade durchführen wollen, so verwenden Sie dazu folgenden Befehl:

```
# rpm -Uvh NessusClient-suse10.3..i586.rpm
Preparing... ##################################[100%]
1:NessusClient ##################################[100%]
```

Nun kann es natürlich auch vorkommen, dass Sie NessusClient wieder entfernen wollen. Im ersten Schritt müssen Sie dazu den RPM-Namen des Pakets identifizieren. Dazu verwenden Sie folgenden Befehl:

```
# rpm –qa | grep NessusClient
```

Die Ausgabe sieht dann wie folgt aus:

```
NessusClient-3.2.1-suse10.3
```

Um den Client zu entfernen, verwenden Sie dann folgenden Befehl:

```
# rpm -e NessusClient-3.2.1-suse10.3
```

Beachten Sie, dass dabei nicht die Konfigurationsdateien und/oder Dateien entfernt werden, die nicht zur Ausgangsinstallation gehörten.

Um Nessus vollständig zu entfernen, verwenden Sie folgenden Befehl:

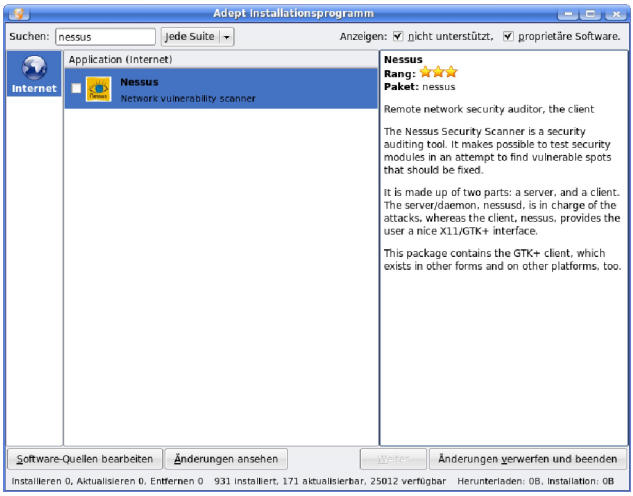
```
# rm -rf /opt/nessus
```

Diesen Befehl sollten Sie natürlich nicht verwenden, wenn auch der Server auf diesem System im Standardinstallationsverzeichnis installiert ist.

Nach der erfolgreichen Installation können Sie NessusClient dann wie folgt starten:

```
# /opt/nessus/./NessusClient
```

NessusClient präsentiert Ihnen dann seine Schnittstelle und erlaubt über den Connection Manager die Verbindungsaufnahme zu einem Nessus-Server.



Kubuntu erlaubt über seinen Paketmanager Adept die Installation von Nessus, allerdings nur die veraltete Version 2.4.

Achtung: Alte Nessus-Version

In verschiedenen Distributionen wie Kubuntu 8.04 und anderen ist über den jeweiligen Paketmanager die Installation von Nessus möglich. Beachten Sie, dass es sich dabei nicht um die aktuelle Version Nessus 3.x handelt, sondern um den freien Vorgänger 2.x.

3.2 Nessus-Erstkonfiguration mit dem NessusClient

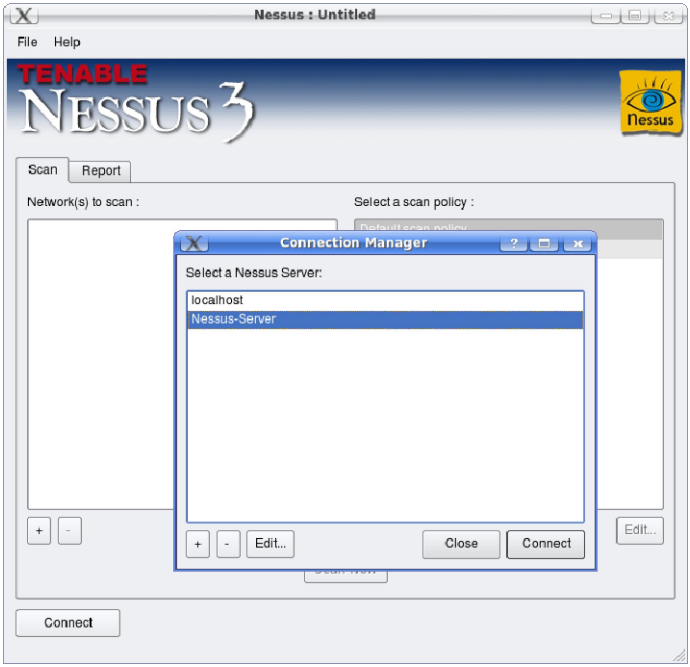
Nachdem Sie Nessus in Betrieb genommen haben, können Sie sich als Nächstes der Server-Konfiguration widmen. Dazu verwenden wir im Folgenden den von Tenable entwickelten NessusClient.

Stellen Sie zunächst sicher, dass der von Ihnen gewünschte Server verfügbar ist, und starten Sie dann NessusClient. Der Client präsentiert Ihnen seine übersichtli-

che Schnittstelle mit seinen beiden Registern *Scan* und *Report*. Auf dem Register *Scan* legen Sie fest, welche Netzwerkbereiche (es kann sich auch um einzelne Systeme handeln) einem Sicherheits-Scan unterzogen werden sollen. Klicken Sie auf das Pluszeichen, um die bzw. das Ziel zu definieren. Klicken Sie zum Speichern der Zieleinstellung auf *Save*.

Als Nächstes bestimmen Sie über den Connection Manager, mit welchem Nessus-Server die Tests durchgeführt werden sollen. Bei der Erstinstallation finden Sie im Verbindungsmanager den Eintrag *localhost* für den Verbindungsaufbau. Erstellen Sie gegebenenfalls einen oder weitere neue Verbindungseinträge.

Das Erstellen einer neuen Verbindung ist einfach. Klicken Sie auf das Pluszeichen, um eine neue Verbindung zu erstellen. Unter *Connection Name* weisen Sie der Verbindung eine Bezeichnung zu. Im Eingabefeld *Host name* geben Sie die IP-Adresse bzw. den Hostnamen des Nessus-Servers an. Der Nessus-Server verwendet standardmäßig den Port 1241.



Über den NessusClient rufen Sie den Verbindungsmanager auf und stellen die Verbindung zum Nessus-Server her.

Dann geben Sie den Benutzernamen und das Passwort für den Verbindungsaufbau an. Beide haben Sie bei der Einrichtung des Nessus-Servers mit *nessus-adduser* eingerichtet. Wenn Sie weitere Benutzer benötigen, erzeugen Sie diese einfach auf dem Nessus-Serversystem. Außerdem können Sie mit der Option *SSL Setup* für eine SSL-gesicherte Verbindung zwischen dem Nessus-Client und -Server sorgen.

Aus dem Verbindungsmanager heraus stellen Sie die Verbindung zwischen Client und Server her. Wählen Sie dazu den Server aus und klicken Sie einfach auf die Schaltfläche *Connect*. Kann die Verbindung hergestellt werden, verwandelt sich die Connect- in eine Disconnect-Schaltfläche.

Bevor wir auf die unzähligen Scan-Einstellungen zu sprechen kommen, sollten Sie noch die Funktionen des NessusClient-Menüs kennen. Der Client stellt Ihnen zwei Menüs zur Verfügung:

- **File**: Hier finden Sie typische dateibezogene Funktionen wie das Öffnen, Speichern etc.
- **Help**: Hier finden Sie lediglich die typische Produktinformation.

Der Info-Dialog des NessusClients verrät Ihnen die exakte Produktversion.

Über das Menü *File* können Sie verschiedene Aktionen durchführen, beispielsweise die folgenden:

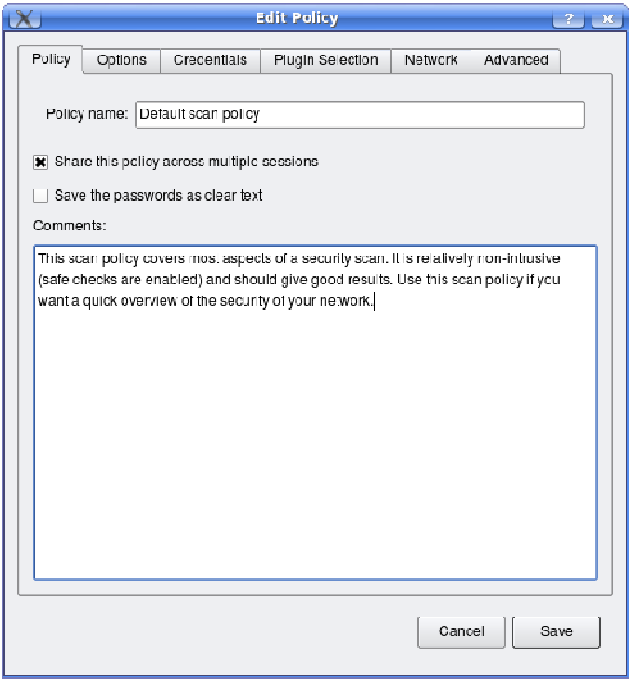
- Sie können über den Eintrag *New Session* einen zweiten Dialog öffnen, um mit diesem andere Ziele und andere Scan-Profile zu testen.
- Sie können eine bestehende Scan-Konfiguration über *File> Open* öffnen. In solchen Scan-Konfigurationen sind Ziele, Policys und Scan-Einstellungen gespeichert. Sie besitzen die Dateierweiterung *.nessus*.
- Sie können den Dialog mit *File> Close* schließen.
- Sie können eine Konfiguration sichern. Ihr wird standardmäßig die Dateierweiterung *nessus* zugewiesen.
- Sie können eine Policy über *File> Import* importieren und so den einfachen Austausch von Scan-Richtlinien ermöglichen.

Schließlich können Sie eine Policy mit dem Befehl *File> Export Policy* in das Format *nessusrc* exportieren und dann beispielsweise in weiteren Umgebungen einsetzen.

Wie es sich für einen grafischen Benutzer-Client gehört, können Sie außerdem NessusClient mit dem Befehl *File> Quit Nessus* beenden.

3.3 Policy-Einstellungen

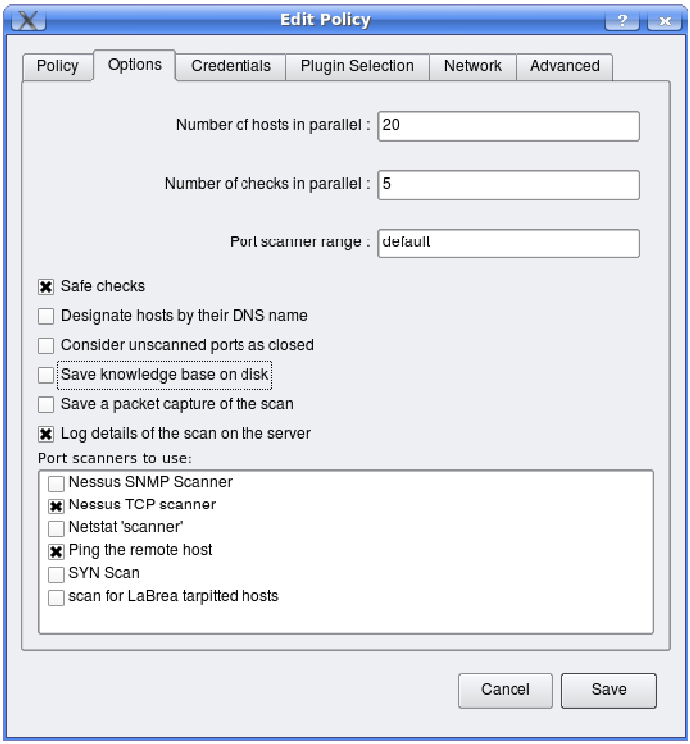
Steht die Verbindung zwischen dem Nessus-Client- und Server, können Sie wie bereits erwähnt über den Client zunächst die Systeme bestimmen, die Sie Ihren Sicherheitstests unterziehen wollen.



Die Konfiguration einer Scan-Policy erfolgt auf sechs Registern.

Im nächsten Schritt bestimmen Sie die Scan-Richtlinien. Damit bestimmen Sie eine Fülle an Scan-Einstellungen. Für den Einstieg ist es ausreichend, wenn Sie sich mit der Standardscan-Richtlinie *Default Scan Policy* befassen. Wenn Sie ein wenig mit Nessus vertraut sind, können Sie natürlich auch eigene Policys erstellen. Doch zunächst sollten Sie die Policy-Einstellungen kennenlernen.

Markieren Sie einen Policy-Eintrag und öffnen Sie dessen Einstellungen mit einem Klick auf die Edit-Schaltfläche unterhalb der Policy-Liste.



Auf dem Register *Options* des Nessus-Clients bestimmen Sie die maximale Anzahl an Hosts, die der Server prüft.

3.3.1 Register Options

Die Policy-Einstellungen sind über sechs Register verteilt. Das erste Register trägt die Bezeichnung *Policy*. Hier finden Sie die Bezeichnung der Scan-Richtlinie und legen fest, ob die Policy auch über mehrere Sessions hinweg verwendet werden kann. Die Option *Save the passwords as clear text* sollten Sie nicht aktivieren, außer vielleicht zu Testzwecken. Unter *Comments* können Sie außerdem eine Beschreibung hinterlegen.

Die Anpassungs- und Konfigurationsmöglichkeiten der weiteren Register fallen umfangreicher aus.

Prinzipiell ist die Anzahl der Systeme, die ein Nessus-Server prüfen kann, von der Leistungsfähigkeit der eigenen Hardware abhängig. Daneben gibt es weitere Faktoren, die die Scan-Möglichkeiten beeinflussen, beispielsweise Sicherheitsrichtlinien

für die Nutzung von internen Systemen etc. Hier ist etwa darauf zu achten, dass kritische Systeme nicht harten Attacken ausgesetzt werden.

Für das Testen von Verwundbarkeiten eines Systems gibt es zwei Ansätze: eindringendes Scannen und nicht-eindringendes Scannen. Im ersten Fall sendet man an den jeweiligen Dienst Daten, die die Schwachstelle ausnutzen und beispielsweise ein System zum Absturz bringen. Bei der zweiten Methode sendet man Anfragen an den Dienst, die die Schwachstelle verifizieren, den Dienst aber selbst nicht lahmlegen oder auf eine andere Art beschädigen.

Nessus unterstützt beide Verfahren. Um die ungefährlichere Variante zu wählen, aktivieren Sie die Option *Safe checks* auf dem Register *Options*. Diese Option sorgt auch dafür, dass nicht unnötige Informationen in den Berichten landen.

Für den Administrator ist es besonders schwierig, sie als solche zu identifizieren. Es dauert nicht nur lange, bis sie als solche erkannt und behoben sind, sondern sie können sogar ganze Testreihen (auch solche über einen längeren Zeitraum hinweg) in Frage stellen. Also muss man Wege finden, wie man mit solchen Problemen umgeht. Auch hierfür ist Nessus gerüstet.

Die Option *Designate hosts by their DNS name* hat Auswirkungen auf die Scan-Performance – gerade bei sehr umfangreichen Testvorgängen. Normalerweise gibt Nessus die Ergebnisse seiner Tests nach IP-Adressen sortiert aus. Wenn Sie die Sortierung nach Hostnamen vorziehen, aktivieren Sie diese Option. Sie hat auch positive Auswirkungen auf die Scan-Vorgänge, wenn Sie viele Systeme den Sicherheitstests unterziehen. Der Grund für die schnellere Ausführung: Die DNS-Einträge werden nicht sauber aktualisiert. Das kann pro Host-Adresse bis zu 10 Sekunden Zeit einsparen.

Nessus greift bei seinen Tests auch auf altbekannte Werkzeuge wie Netstat & Co. zurück und ist darauf angewiesen, dass diese korrekt arbeiten. Soweit es die Scans betrifft, hat die Option *Consider unscanned ports as closed* einen erheblichen Einfluss auf die Berichtausgabe. Wenn Sie diesen Schalter aktivieren, geht Nessus davon aus, dass nicht gescannte Ports sicher sind. Dass das nicht notwendigerweise der Fall sein muss, versteht sich von selbst.

In Vorgängerversionen war auch NMAP in Nessus integriert. Aber dieser ist inzwischen nicht mehr Bestandteil des Scanners. Der Grund: NMAP ist auf das Scannen von großen Netzwerken spezialisiert, nicht aber so sehr auf das Scannen von Systeminterna und Systemschnittstellen.

Wenn Sie Nessus 2.x kennen, so vermissen Sie vermutlich beim ersten Zugriff auf die Policy-Einstellungen das Register *KB*, das für Knowledgebase steht. Bei der Knowledgebase handelt es sich eine textbasierte Wissensdatenbank, die Daten über die durchgeführten Tests und die Testumgebung speichert.

Das Register ist leider verschwunden. Die Speicherung der Daten, aus denen Nessus quasi aus bereits durchgeführten Scans etwas lernen kann, ist nach wie vor möglich. Um die Daten zu sichern, aktivieren Sie die Option *Save knowledge base on disk*. Nach dem Aktivieren dieser Einstellungen landen die Daten im Ordner *KBs* des betreffenden Users.

Standardmäßig ist auch das Aufzeichnen der Scan-Aktivitäten des Servers mit der Option *Log details of the scan on the server* aktiviert. Nessus erzeugt zwei Protokolldateien, die im Unterverzeichnis logs abgelegt werden: *scan.txt* und *server.txt*. Die eine enthält die Protokolle der durchgeführten Scans, die andere die Protokolle der Server-Aktivitäten.

Hier ein Ausschnitt aus einer typischen Server-Protokolldatei:

```
[Tue Sep 16 11:15:17 2008][1996] Successful login of User1
from 192.168.1.2

[Tue Sep 16 11:15:24 2008][1996] user User1 starts a new
scan. Target(s) : 192.168.1.13, with max_hosts = 20,
max_checks = 5 and safe_checks = yes

[Tue Sep 16 11:20:08 2008][1996] user User1 : test of
192.168.1.13 completed

[Tue Sep 16 11:22:40 2008][1996] Client closed the communica-
tion
```

Der Scan-Log-Datei können Sie eine Fülle an Details zu den geladenen Plug-ins, dem Scan-Zeitpunkt etc. entnehmen. Aufgrund der Fülle an Informationen kann das Scan-Protokoll schnell hundert MB und mehr groß werden. Auch hier ein Blick auf einen Ausschnitt:

```
[Tue Sep 16 11:15:25 2008][3896] Use default port range

[Tue Sep 16 11:15:31 2008][3896] user User1 : testing
192.168.1.13 (192.168.1.13) [3896]

[Tue Sep 16 11:15:31 2008][3896] Scan 192.168.1.13 using
20200 plugins

[Tue Sep 16 11:15:31 2008][3896] user User1 : launching
clrtxt_proto_settings.nasl against 192.168.1.13 [1]

[Tue Sep 16 11:15:31 2008][3896] user User1 : launching
dont_scan_settings.nasl against 192.168.1.13 [2]

[Tue Sep 16 11:15:31 2008][3896] user User1 : launching
ssh_settings.nasl against 192.168.1.13 [3]
```

```
[Tue Sep 16 11:15:31 2008][3896] clrtxt_proto_settings.nasl
(process 1) finished its job against 192.168.1.13 in 0.015
seconds

[Tue Sep 16 11:15:31 2008][3896] dont_scan_settings.nasl
(process 2) finished its job against 192.168.1.13 in 0.000
seconds

[Tue Sep 16 11:15:31 2008][3896] ssh_settings.nasl (process
3) finished its job against 192.168.1.13 in 0.000 seconds

[Tue Sep 16 11:15:31 2008][3896] user User1 : launching
snmp_settings.nasl against 192.168.1.13 [4]

[Tue Sep 16 11:15:31 2008][3896] snmp_settings.nasl (process
4) finished its job against 192.168.1.13 in 0.000 seconds

[Tue Sep 16 11:15:31 2008][3896] user User1 : launching
ping_host.nasl against 192.168.1.13 [5]

[Tue Sep 16 11:15:57 2008][3896] ping_host.nasl (process 5)
finished its job against 192.168.1.2 in 25.187 seconds

[Tue Sep 16 11:15:57 2008][3896] user User1 : launching
dont_scan_printers.nasl against 192.168.1.13 [6]

[Tue Sep 16 11:15:57 2008][3896] dont_scan_printers.nasl
(process 6) finished its job against 192.168.1.13 in 0.000
seconds

[Tue Sep 16 11:15:57 2008][3896] user User1 : launching syn-
scan.nes against 192.168.1.13 [0]

…

…
```

Es folgt der Auswahlbereich, über den Sie bestimmen, welche Scanner Nessus verwenden soll. Sie haben die Wahl zwischen sechs verschiedenen Typen:

1 **Nessus SNMP scanner**: Nessus verfügt über einen eigenen SNMP-Scanner. Mit diesem können Sie SNMP-Requests an die zu scannenden Systeme verschicken und diese auf mögliche SNMP-Schwachstellen prüfen. Insbesondere Router lassen sich mit diesem Scanner auf Schwachstellen hin überprüfen.

2 **Nessus TCP scanner**: Dieser Scanner ist auf das Aufdecken von offenen Ports spezialisiert. Dieser Scanner ist standardmäßig aktiviert. Auf dem Register *Advanced* finden Sie verschiedene Konfigurationsmöglichkeiten.

3 **Netstat scanner**: Der Netstat-Scanner kommt insbesondere auf Linux-Systemen zum Einsatz. Er zeigt die Protokollstatistiken und aktuelle Rechnernetzverbindungen an. Mit diesen Statistiken finden Sie heraus, welche Ports geöffnet sind oder welche Verbindungen zu entfernten Rechnern bestehen.

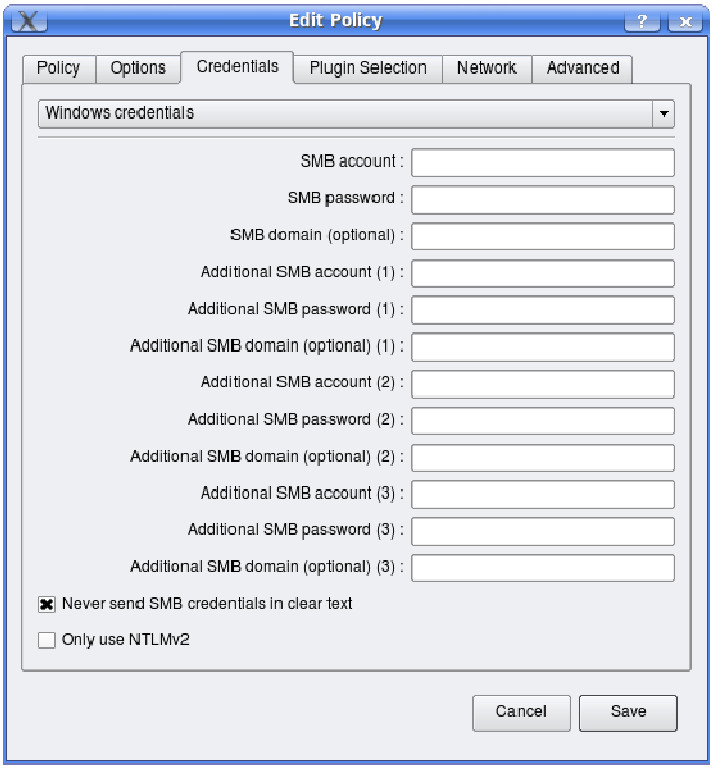
4 **Ping the remote host**: Auch diese Option ist standardmäßig aktiviert. Sie sorgt dafür, dass vor dem eigentlichen Scan-Vorgang ein Ping an das Zielsystem verschickt wird, um sicherzustellen, dass das System auch erreichbar ist.

5 **SYN Scan**: Dieser Scanner sendet ein SYN-Paket an das Ziel und wartet auf die Antwort. Kommt keine, geht Nessus davon aus, dass der Port geschlossen ist.

6 **Scan for LaBrea tarpitted hosts**: LaBrea (*http://labrea.sourceforge.net*) ist eine sogenannte TCP-Teergrube (Tarpit). Dabei handelt es sich um eine spezielle Honeypot-Form, bei der Netzwerkverbindungen künstlich verlangsamt werden und der Verbindungspartner möglichst lange blockiert wird. Wenn Sie diese Option aktivieren, sucht Nessus auch nach den LaBrea-Teergruben.

3.3.2 Register Credentials

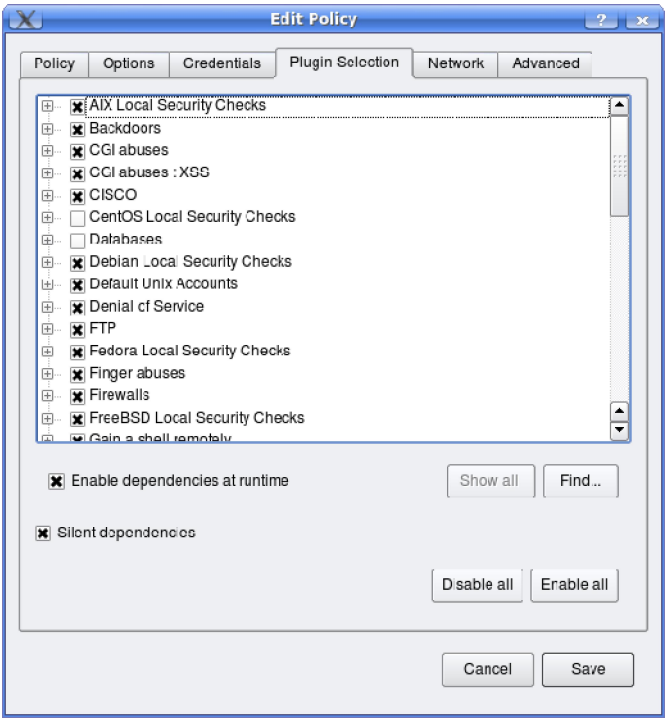
In den Policy-Einstellungen warten vier weitere Register auf Sie. Nessus unterschied sich von Anfang an von anderen Produkten dadurch, dass es nicht nur typische Netzwerkschwachstellen erkennen, sondern auch systemspezifische Sicherheitslücken identifizieren kann. Die dafür relevanten Funktionen finden Sie auf dem Register *Credentials*.



Die Funktionen des Registers Credentials erlauben das Aufdecken von systemspezifischen Schwachstellen.

Das Credentials-Register stellt Ihnen ein Auswahlmenü zur Verfügung, über das Sie aus drei Bereichen Einstellungen wählen können. Jede Option stellt Ihnen eigene Konfigurationsmöglichkeiten zur Verfügung.

- **Windows Credentials**: Erlaubt Ihnen das Testen von mehreren Windows-Domainen. Sie können bis zu vier Domains prüfen.
- **SSH settings**: Erlaubt es Ihnen, sich über SSH Zugang zu Unix-basierten Systemen zu verschaffen.
- **Kerberos configuration**: Erlaubt die Verwendung von Kerberos, um sich Zugriff auf Drittsysteme zu verschaffen.



Die Auswahl der Plug-ins, also der eigentlichen Test-Skripts, erfolgt auf dem Register *Plugin Selection*.

3.3.3 Plug-in-Auswahl

Sind die allgemeinen Scan-Einstellungen definiert, so widmet man sich in der Regel im nächsten Schritt der Auswahl der Plug-ins. Wie bereits erwähnt, verbergen sich dahinter die eigentlichen Tests. Die Tests sind in die oben erwähnten Kategorien eingeteilt. Das erleichtert das Auffinden und die Auswahl der gewünschten Tests. Mit einem Häkchen aktivieren Sie einfach die Tests, die Nessus an Ihren Zielen durchführen soll.

Wenn Sie einen bestimmten Test suchen, klicken Sie auf die Find-Schaltfläche und geben in das Eingabefeld *Contains* den Suchbegriff ein.