Weniger ist mehr – Virtuelle Thin Clients auf Linux-Basis

Roland Herbst | herbst@cms.hu-berlin.de

Im Zeitalter immer leistungsfähigerer Prozessoren und des RAM-Ausbaus jenseits der 4-GB-Grenze mag es einem zunächst unsinnig erscheinen, schlanke, Ressourcen sparende Systeme zu designen. Im Zusammenhang mit der Anwendung von Virtualisierungstechnologien und aus dem Blickwinkel der IT-Sicherheit heraus betrachtet, wird der Nutzen einer solchen Überlegung aber schnell ersichtlich. Vorgestellt wird hier ein Beispiel für einen virtuellen Thin Client, der dies deutlich macht. Die integrierten Anwendungen sind frei verfügbar und können an die eigenen Anforderungen in großem Umfang angepasst werden. Nicht mehr als 128 MB RAM und ca. 2 GB Festplattenspeicher werden bei performanter Funktion mit grafischer Oberfläche beansprucht.

Warum virtuelle Thin Clients?

Aus Gründen der Praktikabilität und vor allem der Systemsicherheit werden in zunehmendem Maße sogenannte Appliances eingesetzt. Das sind IT-Systeme, welche speziellen Aufgaben dienen und die auf die dabei wesentlichen Funktionen optimiert sind. Beispiele hierfür sind neben Thin Clients u. a. Print-Server, Netzwerk-Speicher (NAS) und VPN-Appliances.

Als Thin Clients bezeichnet man spezielle Terminals, die gegenüber Standard-PCs mit geringeren Hardware-Anforderungen auskommen und nur komprimierte Bildschirm-Informationen sowie die Eingaben von Tastatur, Maus und angeschlossenen Peripheriegeräten, wie z. B. Chipkarten-Lesern, übertragen. Auf Thin Clients werden neben der notwendigen Konfiguration keinerlei Nutzerdaten gespeichert. Das Betriebssystem dieser Thin Clients ist allgemein nicht quelloffen und spezielle Änderungen oder Erweiterungen sind so nur kostenpflichtig über den Hersteller der Hardware möglich.

Großes Potential bietet der Einsatz von Virtualisierungs-Umgebungen, die flexible und zentral verfügbare virtuelle Thin Clients ermöglichen [1]. Um zu veranschaulichen, wie leicht die Vorteile der virtuellen Thin Clients genutzt werden können, wird im Folgenden ein konkretes Beispiel betrachtet.

Von der Virtuellen Maschine zum Thin Client

Das vorgestellte System besitzt folgenden Funktionsumfang:

- Linux-Kernel
- SSH-Client
- Webbrowser
- E-Mail-Client
- RDP-Client
- OpenSSL-Tools
- OpenVPN
- Datenaustauschfunktion

und wird aus einer Standard-Distribution vom Minimalsystem ausgehend aufgebaut.

Vorbereitend wird eine Virtualisierungsplattform ausgewählt. Prinzipiell bieten sich verschiedene Möglichkeiten wie z. B. Virtual Box [2], Xen [3], KVM [4] und der VMware Player [5] an. Da der CMS eine langjährige Erfahrung mit VMware-Produkten besitzt, fiel die Wahl auf diese Plattform. Als Host-System dient die Debian GNU/Linux Distribution.

Basis-Installation

Die Installation eines virtuellen Thin Clients unterscheidet sich nicht wesentlich von der eines gewöhnlichen Linux-Systems. Die Hardware-Anforderungen (128 MB RAM und ca. 2 GB Festplattenspeicher) sind Abbildung 1 zu entnehmen, wobei die Ausstattung des Host-Systems (RAM-Ausstattung bzw. Prozessorleistung und -kerne) die mögliche Konfiguration der Gast-Systeme limitiert.

Bevor mit der Installation des virtuellen Thin Clients begonnen werden kann, sind folgende Voraussetzungen auf dem Host-System zu erfüllen:

1. Download VMware Player [6]
2. Download debian netinst ISO [7]
3. Installation VMware Player

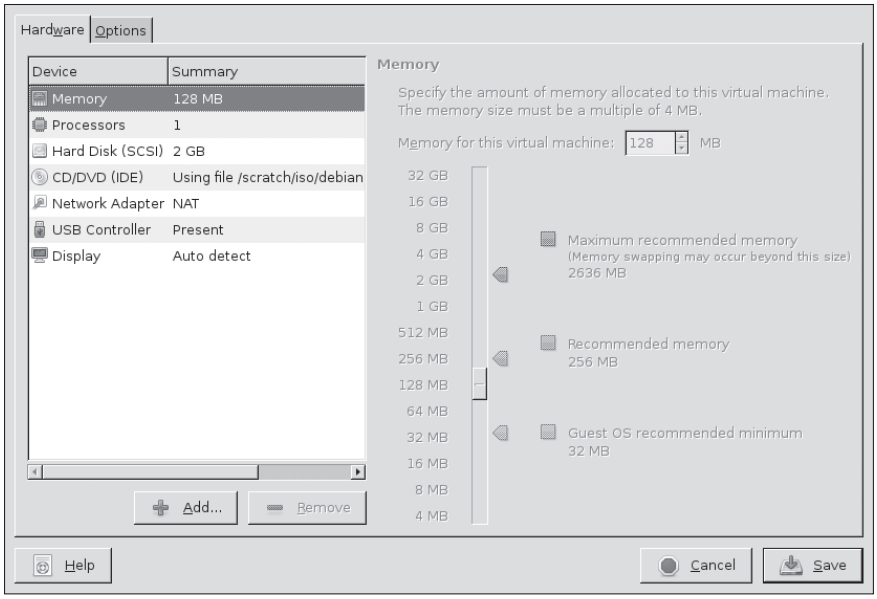

Abb. 1: Die Hardware wird auf das Wesentliche reduziert

Nach Installation der Virtualisierungs-Umgebung auf dem Host-System wird eine VM mit den o. g. Hardware-Anforderungen als Gast-System konfiguriert (Abbildung 1).

Der Einfachheit halber wählt der Administrator bei der Netzwerk-Konfiguration die Funktion NAT (Network Address Translation). Zuerst bindet er hierzu das im vorherigen Schritt heruntergeladene ISO-Image in die VM ein. Danach beginnt er mit der Installation des debian Basis-Systems. Diese unterscheidet sich nicht sehr von der eines normalen Desktop-Systems, deshalb soll hier nur auf die wesentlichen Optionen eingegangen werden. Der gesamte Thin Client wird mit Ausnahme der Boot-Partition in eine dm-crypt-verschlüsselte logische Partition hinein installiert [8]. Dies sorgt dafür, dass die Anforderung nach Vertraulichkeit der Daten des Systems erfüllt wird, auch wenn der Transport z. B. auf dem USB-Stick oder Smartphone des Administrators erfolgt. Damit wird ein hohes Maß an Sicherheit gewährleistet. Die Partitionierung sollte – wie vom Installationsprogramm vorgeschlagen – mit separaten Partitionen erfolgen. Der Vorschlag für die /home-Partition sollte zu Gunsten einer größeren /usr-Partition abgeändert werden. Dies kann jedoch erst im Anschluss an die Installation von der Kommandozeile aus erfolgen (siehe Kasten 2). Für das /home-Verzeichnis eines Thin Clients genügen etwa 100 MB Speicherplatz, denn hier werden maximal Profil-Daten der Anwendungen oder Konfigurations-Dateien abgelegt, welche allgemein wenig Platz beanspruchen (Abbildung 2).

Um ein Minimalsystem zu installieren, sind bei der Festlegung des Installationsumfangs alle Optionen zu deaktivieren. Später können dann die zusätzlich erforderlichen Komponenten nachgeladen werden. Zum Abschluss dieses Teilschrittes wird das System bootfähig gemacht und zum ersten Mal gestartet. Da die logische Partition verschlüsselt ist (siehe Abbildung 2), muss hier die bei der Installation vergebene Passphrase eingegeben werden, was einen Missbrauch unmöglich macht. Die nachfolgenden Schritte sind selbsterklärend.

Nachdem das Basis-System erfolgreich installiert worden ist, kann sich der Administrator mit dem innerhalb des Installationsvorgangs vergebenen Administrator-Passwort als root am System anmelden, um mit der Installation der zusätzlich erforderlichen Software-Pakete fortzufahren.

Erweiterungen und Anpassungen

Nachdem sich der Administrator erfolgreich angemeldet hat, werden für Anpassungen und Erweiterungen die im Kasten 1 genannten Schritte ausgeführt.

Danach kann optional die oben beschriebene Änderung der Partitionierung des Systems von der Kommandozeile aus vorgenommen werden. Die hierzu erforderlichen Schritte sind in Kasten 2 beschrieben.

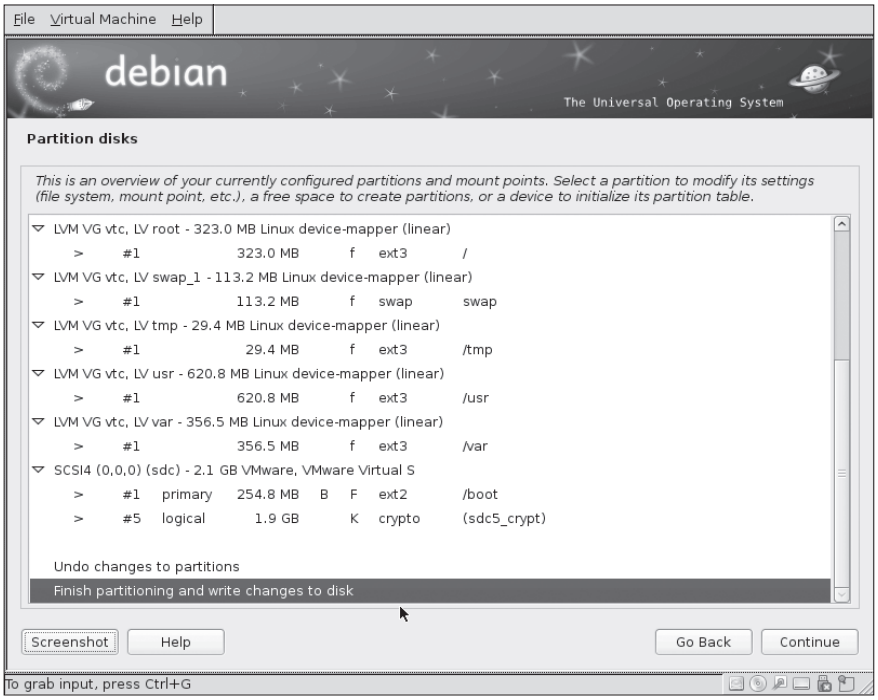

Abb. 2: Innerhalb der Logical Volume-Group vtc liegen die Logical Volumes LV, in denen sich die Partitionen befinden. Mit Ausnahme der Boot-Partition sind diese dm-crypt-verschlüsselt.

Zu Beginn der Installation der zusätzlichen Software-Pakete wird der aktuelle Stand in eine Text-Datei gesichert. Damit wird eine spätere automatisierte Installation vereinfacht, weil man sich durch einfaches Vergleichen einen Überblick über die Änderungen bzgl. der Basis-Installation verschaffen kann.

```
# su - paula
% dpkg --get-selections > default_install_`date +%Y-%b-%d`.txt
% logout
#
```

Danach wird das `sudo` Paket installiert. Es ermöglicht einem am System als Standard-Nutzer angemeldeten Administrator auf Anforderung Administrator-Privilegien zu erlangen, um z. B. System-Konfigurationseinstellungen vorzunehmen.

```
# aptitude install sudo
# export EDITOR=/usr/bin/vi
# visudo
```

Im sich nun öffnenden Fenster ergänzt man einen Eintrag für den bei der System-Installation angelegten nicht privilegierten Nutzer, hier z. B: `paula`.

```
paula ALL=NOPASSWD: ALL
```

und beendet den Editor.
Jetzt sollte `paula` über `sudo` Administrationsrechte erlangen können.

Der Befehl

```
% sudo head /etc/shadow
```

sollte die ersten Zeilen der Passwort-Shadow-Datei anzeigen. Danach kann der Administrator die Installation fortführen. Nacheinander werden das X-Windows-System, einige System-Programme, Security-Tools, die Programme der Mozilla-Suite und das Rdesktop-Programm aus dem während des Installationsprozesses konfigurierten Repository installiert.

X11

```
% sudo aptitude install xorg fluxbox
```

Xorg steht für das X-Windows-System, Fluxbox ist der dazugehörige Window-Manager.

System-Programme

```
% sudo aptitude install cron-apt dnsutils less ntp ntpdate vim
```

`Cron-apt` sorgt für den automatischen Download und die Installation von System-Updates. Vim ist eine Erweiterung des Standard-Unix-Editors vi, der u. a. das farbliche Hervorheben der Syntax von Quellcode erlaubt (Syntax-Highlighting).

Security-Tools

```
% sudo aptitude install iptstate openssh-client openssl openvpn
```

iptstate ist ein Programm, dass die aktuell bestehenden Netzwerk-Verbindungen in einem Terminal-Fenster anzeigt. Die drei letztgenannten Programme dienen der verschlüsselten Kommunikation über unsichere Kanäle und dürfen unter Administratoren als bekannt vorausgesetzt werden [11], [12].

Mozilla-Suite

Standardmäßig installiert der Administrator die debian-Pakete `iceweasel` und `icedove`, aber es besteht auch die Möglichkeit, die Original-Pakete des Mozilla-Projektes zu installieren. Hierfür bietet sich das `/opt`-Verzeichnis an. Beim ersten Start wird die Profil-Struktur für die Mozilla-Applikationen im Home-Verzeichnis der Nutzerin `paula` angelegt.

```
% sudo aptitude install bzip2 libdbus-glib-1-2 libgtk2.0-0
```

Firefox (aktuell 3.6.14)

```
% cd /tmp && wget ftp://ftp.mozilla.org/pub/firefox/releases/3.6.14/linux-i686/en-US/firefox-3.6.14.
tar.bz2
% cd /opt && sudo tar xfj /tmp/firefox-3.6.14.tar.bz2
```

Thunderbird (aktuell 3.1.8)

```
% cd /tmp && wget ftp://ftp.mozilla.org/pub/thunderbird/releases/3.1.8/linux-i686/en-US/thunderbird-
3.1.8.tar.bz2
% cd /opt && sudo tar xfj /tmp/thunderbird-3.1.8.tar.bz2
```

Rdesktop

```
% sudo aptitude install rdesktop
```

Rdesktop ist die Anwendung für den Remote-Zugang zu Microsoft-Windows-basierten Systemen.

Kasten 1: Das System wird mit zusätzlichen Software-Paketen erweitert.

Nachfolgend sind die Befehle aufgeführt, um die Dateisysteme nachträglich in der Größe zu verändern. Dazu muss man als `root` am System angemeldet sein. Zuerst wird das `/home`-Verzeichnis um 324 MB verkleinert, um den in diesem Schritt gewonnenen Platz dann dem `/usr`-Verzeichnis hinzuzufügen.

```
# umount /home
# fsck -f /dev/vtc/home
# resize2fs /dev/vtc/home 100M
# lvreduce -L -324M /dev/vtc/home
# fsck -f /dev/vtc/home
# mount /home
```

Der Erfolg kann durch Aufruf des Befehls `df -h` überprüft werden. Das `/home`-Verzeichnis sollte jetzt eine Größe von fast 100 MB besitzen.

Um das `/usr`-Dateisystem zu vergrößern, muss das System in den `Single User Mode (init 1)` gefahren werden. Danach sind folgende Schritte auszuführen:

```
# umount /usr
# fsck -f /dev/vtc/usr
# lvextend -L +324M /dev/vtc/usr
# fsck -f /dev/vtc/usr
# resize2fs /dev/vtc/usr
# fsck -f /dev/vtc/usr
# mount /usr
```

Die vergrößerte Partition müsste jetzt ca. 900 MB groß sein. Dies kann wiederum mit df -h überprüft werden.

Kasten 2: Anpassung der Datei-Systeme

Waren diese Schritte erfolgreich, kann man das System rebooten.

Nachdem die im Kasten 1 beschriebenen Software-Pakete installiert worden sind, kann mit deren Konfiguration fortgefahren werden. Die dazu erforderlichen Schritte sind in der der sehr ausführlichen Installationsdokumentation des debian-Projektes [9] beschrieben und werden deshalb an dieser Stelle nicht weiter vertieft. Die Anpassung des vom System genutzten Window-Managers Fluxbox wird in Kasten 3 erläutert. Weiterführende Informationen hierzu findet man auf der Homepage des Projektes [10].

Datenaustausch

Für den Datenaustausch mit dem Host-System bieten sich beim VMware Player die Nutzung von lokalen Netzwerkfreigaben (sogenannte Shared Folders) oder für die Nutzung von Flash-Speichern wie USB-Sticks oder Smartphones die Installation von usbmount an. Auch dies erfolgt nach bekanntem Muster:

```
% sudo aptitude install usbmount
```

Nach der Installation kann der Administrator mit Bordmitteln des installierten Thin Clients auf die Medien zugreifen, da das Mounten der Partitionen automatisch erfolgt, sofern es sich um ein unterstütztes Dateisystem handelt. Die Nutzung der Shared Folders erfordert die Installation der VMware Tools, einer Zusatzfunktion des Virtualisierers.

Der Window-Manager Fluxbox

Im Home-Verzeichnis des jeweiligen Nutzers legt Fluxbox beim ersten Aufruf ein verstecktes Verzeichnis `.fluxbox` an. In diesem befinden sich u. a. die Dateien `init`, `menu`, und `startup`. Hierbei handelt es sich, wie bei UNIX-Systemen üblich, um einfache Text-Dateien, deren Inhalt sich einem schnell erschließt. Die Konfiguration des Fluxbox-Menüs erfolgt in der gleichnamigen Datei `menu`. Das Fluxbox-Menü ist in einzelne Gruppen untergliedert, die dann entweder Subgruppen oder die Aufrufe der einzelnen Applikationen selbst enthalten. Zum Zweck der Anpassung kann der Administrator die aktuelle Version der Datei `menu` sichern, die in der entsprechenden include-Anweisung enthaltene Datei in das Verzeichnis `.fluxbox` kopieren und editieren.

Kasten 3: Konfiguration des Window-Managers Fluxbox

Fazit

Das hier vorgestellte System wird durch den Autor produktiv zur Administration von UNIX-Systemen und in Test-Umgebungen verwendet (Abbildung 3). Die Leistungsfähigkeit der virtuellen Appliance ergibt zusätzlich vorstellbare Einsatzszenarien, wie z. B. die sichere Fernadministration von Windows-Servern mit RDP Clients über einen SSH- oder (Open)VPN-Tunnel aus einem offenen Netzwerkbereich heraus. Die geringen Hardware-Anforderungen ermöglichen auf einem aktuellen Host-System eine Vielzahl von virtuellen Maschinen, die komplexe Systeme, welche in voneinander abgegrenzten Sicherheitskontexten agieren, realisierbar erscheinen lassen. So existiert derzeit die Vorstellung, die in diesem Heft an anderer Stelle vorgestellte VPN-Box zu virtualisieren, um die erprobte Sicherheitsinfrastruktur auf mobile Endgeräte zu erweitern und Administratoren und Endnutzern den Zugang zu geschützten Netzwerkbereichen zu ermöglichen.

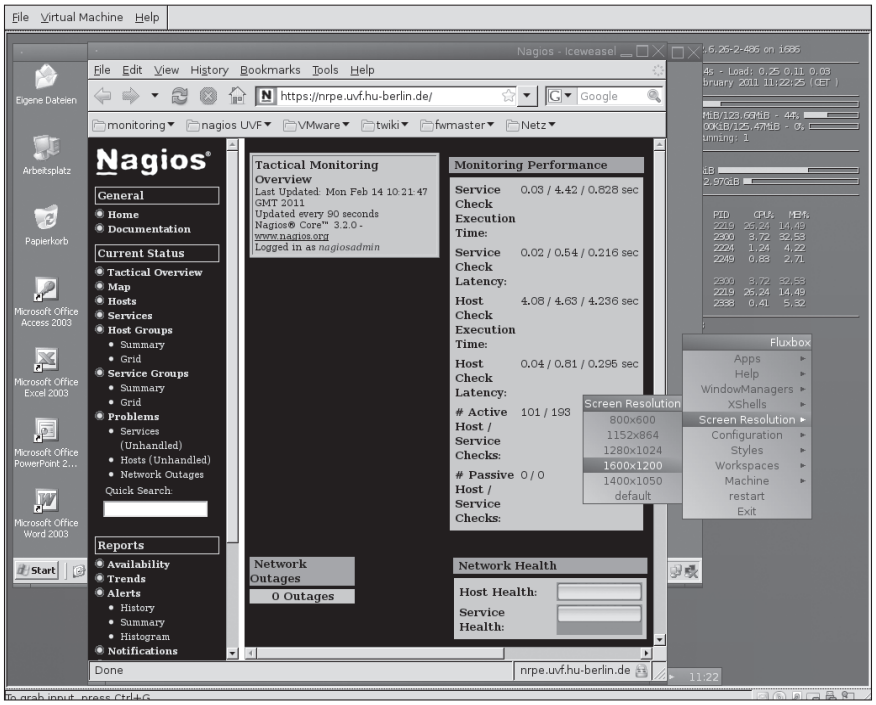

Abb. 3: Der Virtuelle Thin Client in Aktion

Literatur

[1] Herbst, R.: *Der Computer als Applikation*. cms-journal Nr. 28, 27.02. 2006

[2] *VirtualBox*. http://www.virtualbox.org/

[3] *Xen*. http://www.xen.org/

[4] *KVM*. http://www.linux-kvm.org/

[5] *vmware*. http://www.vmware.com/

[6] *Download VMware Player*. http://downloads.vmware.com/d/info/desktop_downloads/vmware_player/3_0

[7] http://cdimage.debian.org/debian-cd/6.0.1a/multi-arch/iso-cd/debian-6.0.1a-amd64-i386-netinst.iso

[8] *cryptsetup – Setup virtual encryption devices under dm-crypt Linux*. http://code.google.com/p/cryptsetup/

[9] *Debian Squeeze – Installationsanleitung*. http://www.debian.org/releases/squeeze/installmanual

[10] *fluxbox.org. Home of the Fluxbox windowmanager*. http://www.fluxbox.org/

[11] *OpenSSL Project*. http://www.openssl.org/

[12] *OpenVPN Community Software*. http://www.openvpn.net/index.php/open-source.html