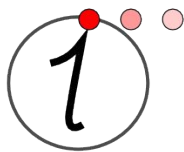


UMLMON

Ein Monitor für User Mode Linux

Produktmerkmale und Einsatzmöglichkeiten

FÜR VERSION 1.0



Informatikbüro
Dipl.-Inform. Gerd Stolpmann

Viktoriastr. 45 • 64293 Darmstadt
<http://www.gerd-stolpmann.de>

Inhaltsverzeichnis

Kurzfassung.....	4
Was ist Virtualisierung?.....	4
User Mode Linux und UMLMON.....	5
Einsatzmöglichkeiten.....	6
Illustration und Screenshots.....	7
Hardware-Emulation bei UML.....	11
Isolierung der Instanzen.....	11
Performance.....	11
Weitere Informationen.....	12
Daten UMLMON.....	13
Daten UMLMON-Web.....	14

Kurzfassung

Durch Virtualisierung können mehrere Betriebssysteme auf derselben Hardware ausgeführt werden. Ziele sind Kosteneinsparung durch bessere Ausnutzung der Hardware sowie Vereinfachung der Rechenzentrums-Administration. User Mode Linux ist eine freie Virtualisierungslösung für Linux-in-Linux, die durch Einspielen eines Patches aktiviert werden kann.

User Mode Linux ist kein fertiges Produkt, sondern eine Art Bausatz, der einzelne, nicht in die Umgebung integrierte Funktionen enthält. UMLMON ist ein kommerzielles Zusatzprodukt für User Mode Linux, das diese Schwachstellen ausgleicht, und die benötigte Integration liefert. UMLMON erlaubt die Administration der virtuellen Instanzen durch unprivilegierte Benutzer des Host-Systems. Desweiteren enthält UMLMON unverzichtbare Sicherheitsmerkmale, nämlich die Privilegien-Separation sowie die Einrichtung eines chroot-Jails.

Die Instanzen von UMLMON werden durch Kommandozeilen-Tools administriert. Der Zusatz UMLMON-Web erlaubt darüber hinaus die Web-Administration.

Was ist Virtualisierung?

Unter Virtualisierung versteht man die gleichzeitige Ausführung von mehreren Betriebssystemen auf derselben Hardware. Dies bedeutet, dass diese Systeme, auch Gast-Systeme oder virtuelle Instanzen genannt, jeweils eigenständig konfiguriert werden können und auch als eigenständige Systeme im Netzwerk auftreten.

In Rechenzentren möchte man häufig aus logischen und administrativen Gründen, dass bestimmte Software-Systeme auf je eigenen Rechnern ablaufen, beispielsweise, weil die Systeme unterschiedlichen Kunden zugeordnet sind oder weil die Systeme aus Sicherheitsgründen getrennt werden müssen. Gleichzeitig wird die verfügbare Hardware immer leistungsfähiger, und es wird immer weniger vertretbar, jeweils eigene Hardware anzuschaffen, da diese nur zu einem Bruchteil ausgenutzt werden würde. Die Virtualisierung ermöglicht nun, mehrere Systeme auf derselben Hardware ablaufen zu lassen.

Im Großrechner-Bereich wurde die Virtualisierung von IBM bereits in den 60er Jahren eingeführt. Auf Intel-Hardware war Virtualisierung bis vor wenigen Jahren



Virtualisierung: Aus Eins mach Viele

undenkbar, vor allem, weil die CPU diese Technik denkbar schlecht unterstützt, und der Performance-Verlust nicht vertretbar gewesen wäre. Bei der Leistungsfähigkeit heutiger Intel-Hardware ist der Einwand aber hinfällig – zwar gibt es immer noch einen deutlichen Performance-Verlust, dieser ist durch die hohe Grundgeschwindigkeit aber häufig akzeptabel.

Neben der besseren Ausnutzung von Hardware werden häufig auch administrative Gründe für die Einführung von Virtualisierung genannt. Es ist beispielsweise viel einfacher und weniger aufwändig, eine neue virtuelle Instanz aufzusetzen als ein echtes System zu installieren. Ersteres dauert nur wenige Minuten und benötigt nur wenige Handgriffe, während letzteres die Administratoren auch heute immer noch Stunden beschäftigt. Virtuelle Instanzen können außerdem problemlos von einer Hardware auf eine andere transportiert werden.

User Mode Linux und UMLMON

User Mode Linux (UML) ist, wie der Name schon sagt, eine Linux-spezifische Lösung. Sie ist darauf beschränkt, dass ein gastgebendes Linux-System, auch Host-System genannt, eine oder mehrere virtuelle Linux-Instanzen ausführen kann. UML ist damit eine reine Linux-in-Linux-Lösung.

UML beruht darauf, dass Linux im Quelltext vorliegt und daher verändert werden kann. Die an sich schwierige Virtualisierung auf Intel-Hardware wird auf diese Weise deutlich vereinfacht. Die Veränderung von Linux wird durch Patches bewirkt, die der System-Administrator sowohl für das Gast- als auch für das Host-System einspielen muss (diese Patches sind als „UML-Patch“ und als „SKAS-Patch“ bekannt). Die Patches sind, wie auch Linux, freie Software und jedermann zugänglich.

Der Nachteil von UML ist, dass es nur eine geringe Integrationstiefe aufweist. Der Administrator wird weit gehend alleine gelassen, was die Installation und die Betriebsintegration angeht. Das Produkt UMLMON liefert nach, was in der frei erhältlichen UML-Grundausstattung fehlt, und ergänzt wichtige Administrationsmöglichkeiten, die in einer professionellen Umgebung unverzichtbar sind:

- Zentrale Administration (Konfiguration, vorgegebenes Verzeichnis-Layout)
- Einbindung in die Boot-Skripte des Host-Systems
- Schreiben von Log-Dateien und Log-Rotation
- Konsolen-Konzentration des Gast-Systems
- Delegierung der Administration an einfache Benutzer durch Privilegien-Separation
- Administration durch Kommandozeile, durch Kommando-Shell oder durch das

im Quelltext erhältliche Zusatzprodukt UMLMON-Web (Web-Administration)

- Remote-Administration durch RPC-Protokoll
- Sichere Ablaufumgebung für UML durch Einrichtung eines chroot-Jails

Trotz der zahlreichen Funktionen ist UMLMON eine schnelle, leichtgewichtige und stabile Lösung. Ermöglicht wurde dies durch die Entwicklung in einer High-Tech-Programmiersprache: Objective Caml. Diese Sprache wurde seit Ende der 80er Jahre in den Labors des Forschungsinstituts INRIA entworfen und gilt in der akademischen Welt als große Hoffnung. Die Potenzial besteht vor allem darin, dass mit Algorithmen der künstlichen Intelligenz Programme vorab auf klassische Programmierfehler geprüft werden und damit viele Fehlerquellen von vorneherein ausgemerzt werden.

Einsatzmöglichkeiten

Die Kombination UML + UMLMON wurde primär für den Rechenzentrumsbetrieb ausgelegt. Daneben sind aber auch andere Einsatzgebiete im Bereich von Labor-Umgebungen und Schulungs-Umgebungen denkbar.

Server-Konsolidierung: Server-Hardware wird häufig nicht ausgelastet. Durch Virtualisierung können mehrere Systeme auf derselben Hardware ablaufen, um damit den Auslastungsgrad zu steigern. Desweiteren vereinfachen sich einige Administrationsarbeiten, z.B. kann mit wenigen Handgriffen ein neues System hochgezogen werden. Die Server-Konsolidierung per UML eignet sich vor allem für Services, die geringen bis mittleren Leistungshunger haben, z.B. DNS-Server, statische Web-Server, Print-Server, Mail-Server in kleinen Netzen.

Virtuelle Sicherheitszonen: Die sichere Anbindung von Firmennetzen an das Internet benötigt sogenannte demilitarisierte Zonen (DMZ). Für kleinere und mittlere Unternehmen ist der Aufbau von DMZ häufig zu kostspielig, da ein ganzer Park von Hardware benötigt wird. Gleichzeitig ist der Leistungshunger von DMZ meist gering. Eine Alternative zu einer physikalisch aufgebauten DMZ ist eine rein virtuelle DMZ, die mit UML leicht realisiert werden kann.

Virtuelles Hosting: Eine kostengünstige Version des Server-Hostings ist das virtuelle Hosting: Statt eines kompletten Systems mietet der Kunde nur eine virtuelle Instanz. Die Administrationsmöglichkeiten von UMLMON und UMLMON-Web eignen sich vorzüglich, den Kunden erstklassige Management-Schnittstellen zur Verfügung zu stellen und sich von der Hosting-Konkurrenz abzusetzen.

Labornetze: Software wird heutzutage häufig für ganze Netzwerke entwickelt. Um diese auch in Netzwerken zu testen, müssen in Labors ganze Server-Farmen aufgesetzt werden. Mit UML vereinfacht sich der Aufbau dieser Umgebungen erheblich: Statt echte Server-Farmen werden virtuelle Farmen bereitgestellt, die erstens weniger Hardware benötigen und zweitens wesentlich leichter aufzusetzen sind.

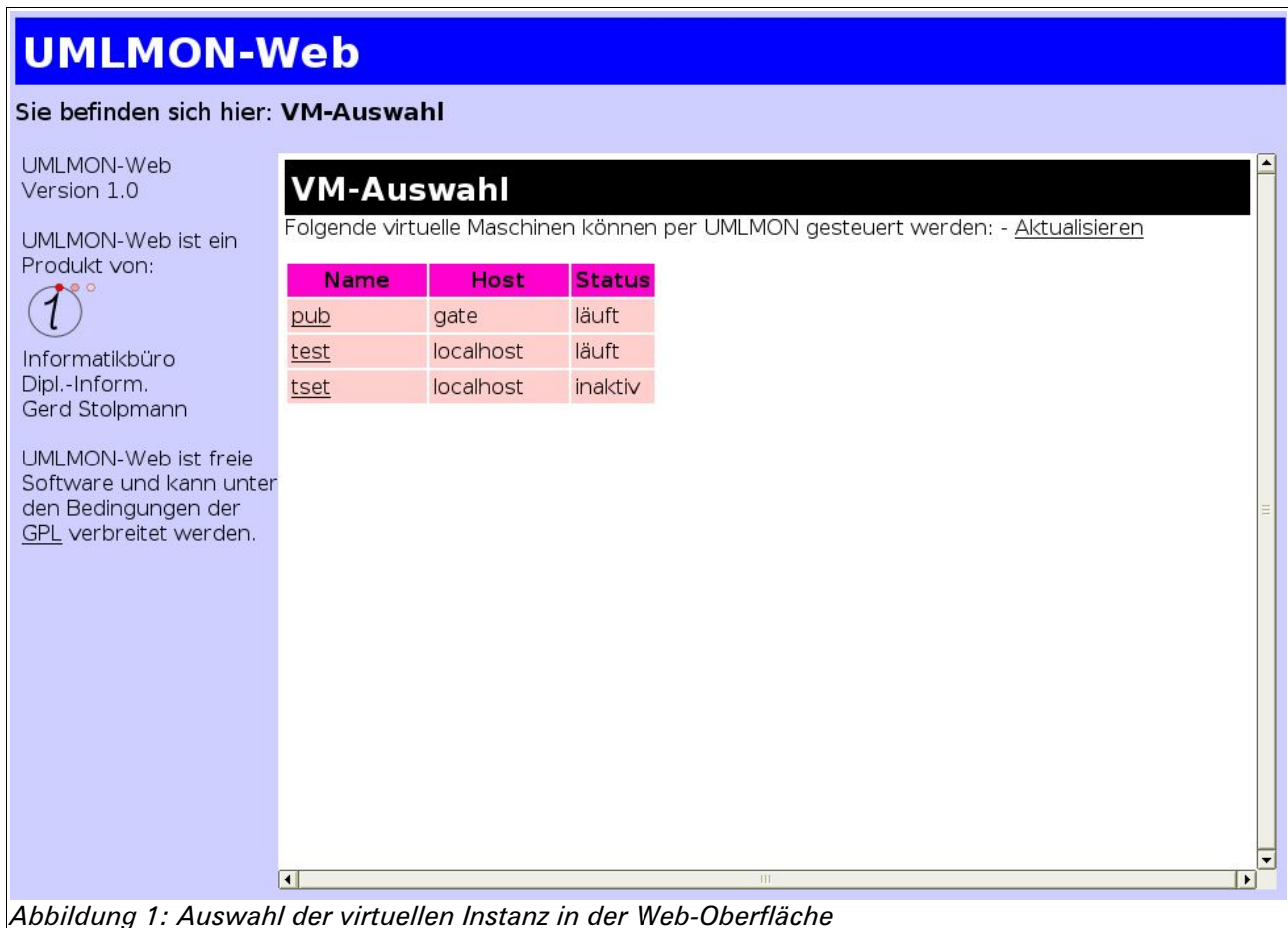


Abbildung 1: Auswahl der virtuellen Instanz in der Web-Oberfläche

Schulungs-Umgebungen: Bei Linux-Schulungen scheuen sich Anbieter häufig, den Teilnehmern echte Administrator-Rechte zu geben. Immerhin hat der Anbieter viel Zeit damit verbracht, die Schulungsrechner einheitlich aufzusetzen und möchte seine Arbeit sichern. Mit virtuellen Instanzen, die per UML und UMLMON aufgebaut werden, kann der Anbieter den Teilnehmern sorgenfrei volle Administrator-Rechte geben. Falls ein Teilnehmer die Installation beschädigt (z.B. übereifrig Dateien löscht), kann die Instanz per Mausklick wiederhergestellt werden. Desweiteren erlaubt UMLMON die Beobachtung der Eingaben, die Teilnehmer auf Textkonsolen machen, und gibt dem Schulungsleiter damit eine wichtige Kommunikationsmöglichkeit.

Illustration und Screenshots

UMLMON an sich ist ein reiner Netzwerk-Dienst, also eine abstrakte Angelegenheit. Mitgeliefert wird das Kommandozeilen-Tool `umlwatch`, das die textbasierte Administration ermöglicht. Fast genauso mächtig wie `umlwatch` ist die Web-Administrations-Software UMLMON-Web. Sie wird sogar im Quelltext zur Verfügung gestellt (mit GPL-Lizenz) und kann daher vom Kunden selber angepasst werden (z.B. sind Layout-Veränderungen sehr einfach zu bewerkstelligen).

UMLMON-Web illustriert die Möglichkeiten von UMLMON. In Abbildung 1 wird die

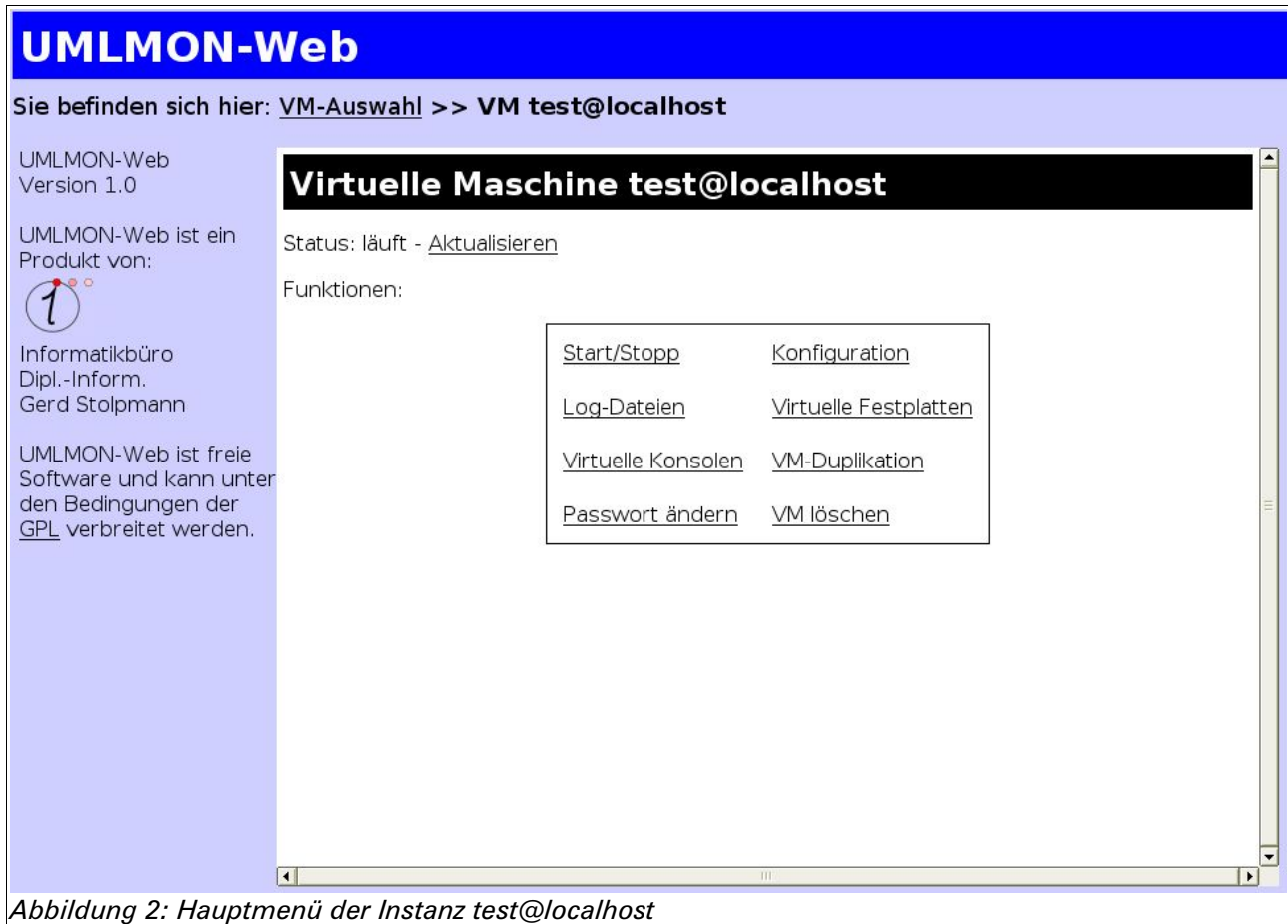


Abbildung 2: Hauptmenü der Instanz test@localhost

Einstiegsseite gezeigt. Der Benutzer kann hier die virtuelle Instanz auswählen, die er administrieren möchte. In diesem Fall werden eine Instanz auf dem Rechner *gate* und zwei Instanzen auf *localhost* angeboten.

Die Administrations-Funktionen für die Instanzen zeigt das Hauptmenü (Abbildung 2). Die angebotenen Funktionen erlauben die nahezu vollständige Administration der Instanzen durch den Web-Benutzer:

- **Start/Stop:** Ermöglicht das Starten der Instanz (d.h. das virtuelle Booten) sowie das Herunterfahren.
- **Log-Dateien:** Alle Log-Dateien können über das Web-Interface eingesehen werden.
- **Virtuelle Konsolen:** Der in UMLMON eingebaute Konsolen-Konzentrator ermöglicht es, den direkten Textzugang zum Gast-System auf ein Terminal zu schalten. Beispielsweise kann ein ssh-Zugang konfiguriert werden, so dass der Textzugang über ssh-Software ermöglicht wird (z.B. auf eine Linux- oder Windows-Workstation). Für den umstandslosen Zugang wird in der der Web-Oberfläche ein Terminal emuliert, so dass der Benutzer keine Zusatzsoftware benötigt. Allerdings ist die Emulation, bedingt durch die Web-Umgebung, nicht ganz so flüssig zu bedienen wie ein echtes Terminal. Dennoch ist dies

UMLMON-Web

Sie befinden sich hier: [VM-Auswahl](#) >> [VM test@localhost](#) >> **Virtuelle Konsolen**

UMLMON-Web
Version 1.0

UMLMON-Web ist ein Produkt von:

Informatikbüro
Dipl.-Inform.
Gerd Stolpmann

UMLMON-Web ist freie Software und kann unter den Bedingungen der [GPL](#) verbreitet werden.

Virtuelle Maschine test@localhost

Konsole con1
[Andere Konsole auswählen](#)

Zeile: |q

Spezialtasten

← → ↑ ↓ Pos1 Ende Bild↑ Bild↓ ←C Entf Einfg Strg C

```
09:32:00 up 46 min, 1 user, load average: 0.16, 0.04, 0.01
21 processes: 20 sleeping, 1 running, 0 zombie, 0 stopped
CPU states: 0.0% user, 2.0% system, 0.0% nice, 98.0% idle
Mem: 28272K total, 9192K used, 19080K free, 716K buffers
Swap: 0K total, 0K used, 0K free, 4740K cached
```

PID	USER	PRI	NI	SIZE	RSS	SHARE	STAT	%CPU	%MEM	TIME	COMMAND
172	root	14	0	924	924	748	R	3.3	3.2	0:05	top
1	root	9	0	484	484	424	S	0.0	1.7	0:00	init
2	root	9	0	0	0	0	SW	0.0	0.0	0:00	keventd
3	root	19	19	0	0	0	SWN	0.0	0.0	0:00	ksoftirqd_CPU0
4	root	9	0	0	0	0	SW	0.0	0.0	0:00	kswapd
5	root	9	0	0	0	0	SW	0.0	0.0	0:00	bdflush
6	root	9	0	0	0	0	SW	0.0	0.0	0:00	kupdated
7	root	9	0	0	0	0	SW	0.0	0.0	0:00	xfsbufd
8	root	9	0	0	0	0	SW	0.0	0.0	0:00	xfslogd/0
9	root	9	0	0	0	0	SW	0.0	0.0	0:00	xfsdatad/0
134	root	9	0	660	660	544	S	0.0	2.3	0:00	syslogd
137	root	9	0	528	528	380	S	0.0	1.8	0:00	klogd
145	root	9	0	508	508	444	S	0.0	1.7	0:00	inetd
149	daemon	9	0	580	580	504	S	0.0	2.0	0:00	atd
152	root	8	0	684	684	564	S	0.0	2.4	0:00	cron
155	root	9	0	1208	1208	992	S	0.0	4.2	0:00	bash
156	root	9	0	468	468	408	S	0.0	1.6	0:00	getty

Abbildung 3: Konsolenzugriff per Web-Terminal-Emulation

eine eindrucksvolle Funktion von UMLMON-Web, und auf den folgenden Seiten stellen wir sie vor.

- **Passwort ändern:** Die Instanzen können durch Passwörter geschützt werden.
- **Konfiguration:** Hinter diesem Menüpunkt verbirgt sich die Einstellung genereller Konfigurationsoptionen, z.B. die Größe des virtuellen Hauptspeichers.
- **Virtuelle Festplatten:** Eine virtuelle Festplatte ist in Wirklichkeit eine große Datei auf dem Host-System, die der Instanz als Festplatte vorgespiegelt wird. Mit diesem Menüpunkt können die Festplatten administriert werden. Auch in diesem Punkt zeigt sich die Integrationstiefe von UMLMON, und daher werden wir diese Funktion ebenfalls etwas detaillierter vorstellen.
- **VM-Duplikation:** Hiermit wird es dem Benutzer ermöglicht, per Mausklick eine existierende virtuelle Instanz zu duplizieren.
- **VM löschen:** Diese Funktion löscht eine Instanz. Wie die VM-Duplikation ist diese Funktion übrigens durch Benutzerskripte weit reichend an die Bedürfnisse des Rechenzentrums anpassbar.

Die Konsolen eines Linux-Systems sind der primäre textbasierte Zugang. Auf echter

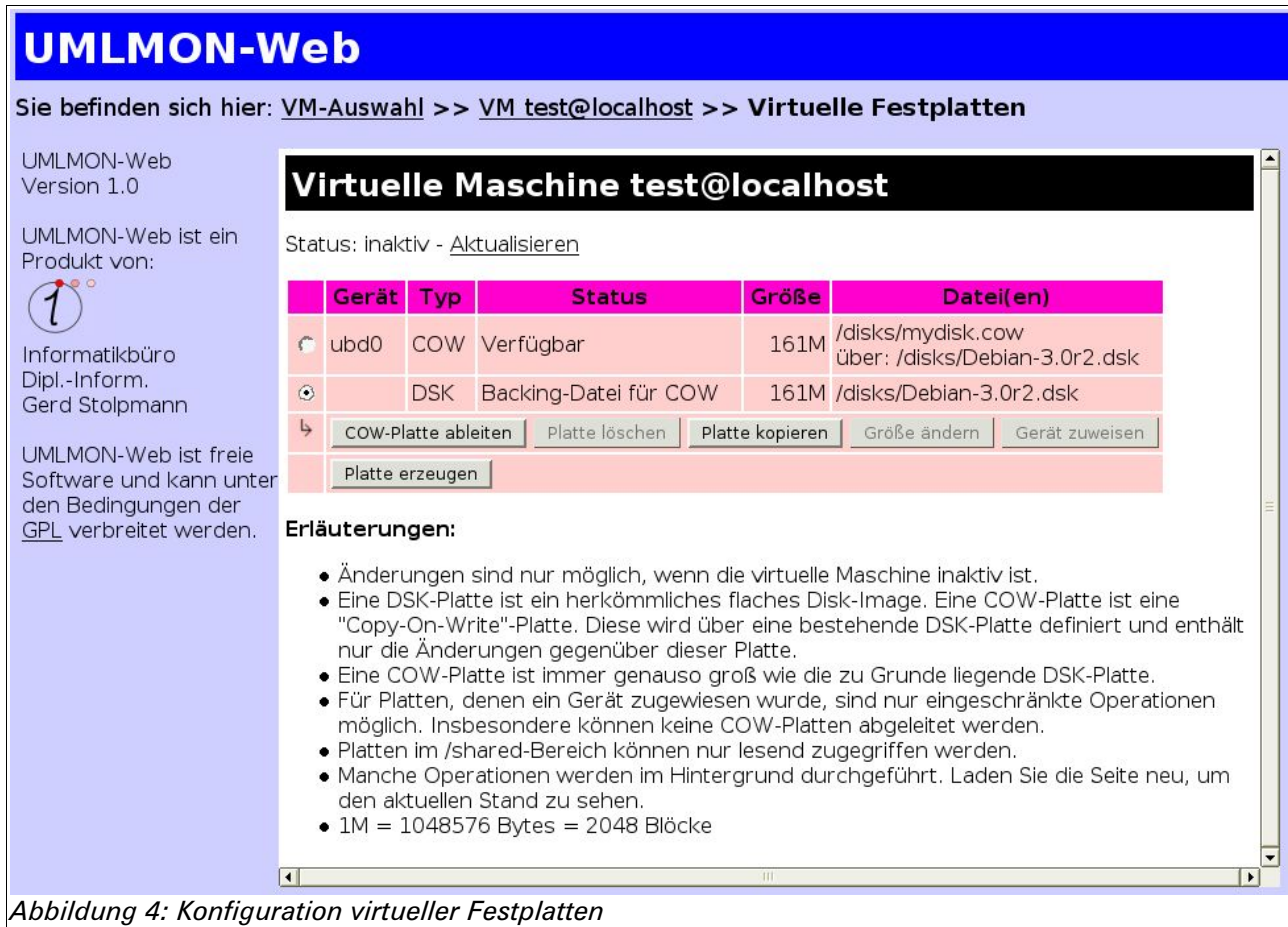


Abbildung 4: Konfiguration virtueller Festplatten

Hardware werden Konsolen in speziellen Bildschirmen angezeigt, zwischen denen hin- und hergewechselt werden kann. Konsolen werden benötigt, um das System zu administrieren, wenn noch keine anderen Zugänge (Netzwerk, Graphik) konfiguriert oder möglich sind. Konsolen können zudem für Logging-Funktionen verwendet werden.

Abbildung 3 zeigt den Zugriff auf eine Konsole per Web-Schnittstelle. Zu diesem Zweck enthält UMLMON einen Terminal-Emulator, der die Steuercodes von Terminals versteht und in eine ordentliche Bildausgabe umsetzt. Hier dargestellt ist der Aufruf des Tools *top*, mit dem Administratoren einen schnellen Überblick über die Augenblickssituation eines Linux-Systems bekommen. Der Terminal-Emulator stellt das Konsolenbild in nahezu Echtzeit dar. Tastatureingaben können ebenfalls emuliert werden. Es ist sogar die Steuerung von maskenorientierten Programmen und Texteditoren möglich (wenn auch etwas umständlich).

Die Konfiguration virtueller Festplatten ist besonders wichtig für die Administration von UML-Instanzen. In Abbildung 4 wird die Schnittstelle gezeigt, die UMLMON-Web hierfür zur Verfügung stellt. In UML gibt es zwei Arten von virtuellen Festplatten: Normale „flache“ Images, sowie Copy-On-Write-Images. Beide Arten lassen sich administrieren. UMLMON achtet dabei darauf, dass nur Operationen durchgeführt werden, die die Integrität der Platten gewährleisten. Beispielsweise

ist es nicht möglich, die Backing-Datei eines Copy-On-Write-Images versehentlich zu löschen. Die Administrationsfunktion ist in leicht verständliche Dialoge aufgeteilt.

Hardware-Emulation bei UML

Das Ideal der Virtualisierung ist, dass sich die virtuelle Instanz wie ein echtes, nativ installiertes System verhält. Natürlich ist das nicht möglich – ein natives System hat zu einem guten Teil damit zu tun, die Hardware zu steuern, die in einer virtuellen Instanz gar nicht vorhanden ist. Da die Hardware aber z.T. schlicht vorausgesetzt wird – was ist ein Betriebssystem ohne Festplatten? – muss das Host-System dem Gast-System das Vorhandensein von Hardware wie Platten, Netzwerk-Interfaces usw. vorspiegeln. Man spricht hier von Hardware-Emulation.

Das besondere an User Mode Linux ist, dass die Hardware-Emulation durch eigenständige Abstraktionen erfolgen kann, und nicht die Bits und Bytes echter Hardware nachgebildet werden muss. Beispielsweise wird in User Mode Linux der Zugriff auf Festplatten durch die speziellen ubd-Treiber realisiert (user block device). User Mode Linux ist daher besonders zuverlässig und flexibel konfigurierbar.

Isolierung der Instanzen

Das Host-System und die darauf ablaufenden Gast-Systeme sollen von einander isoliert sein. Dies bedeutet, dass potenziell schädliche Programme, die auf einem der Gast-Systeme ablaufen, nicht die anderen Gast-Systeme oder gar das Host-System beeinträchtigen können (höchstens darf dadurch Rechenzeit weggenommen werden). User Mode Linux implementiert die Isolierung dadurch, dass die Gast-Systeme unprivilegierte Prozesse des Host-Systems sind. Damit wird die Isolierung bereits durch die normale Prozess-Isolierung von Linux gewährleistet.

UMLMON verbessert die Isolierung noch einmal durch sogenannte chroot-Jails. Hierbei handelt es sich um eine spezielle Sicherheitsfunktion von Linux, die einzelne Bereiche des Systems einkapselt. Diese Funktion sichert die Instanzen gegen Konfigurations- und Kernelfehler ab.

Performance

Natürlich kostet die Virtualisierung erhebliche Rechenzeit. Man kann davon ausgehen, dass eine virtuelle Instanz zwischen 30% (bei I/O-intensiven Anwendungen wie Datenbanken) und 90% (bei CPU-intensiven Anwendungen) der Geschwindigkeit eines nativen Systems erreicht. Die Verlangsamung der I/O-Funktionen ist vor allem darauf zurückzuführen, dass sich der Datenpfad von der Anwendung bis zur echten Festplatte auf die doppelte Länge vergrößert.

Da sich die Performance heutiger Intel-Systeme alle paar Jahre verdoppelt, sind die Einbußen durch die Virtualisierung in vielen Fällen hinnehmbar. Ein heutiges

UML-System fühlt sich ungefähr an wie ein drei Jahre altes echtes System. Dies zeigt, wie akzeptabel die Performance in Wirklichkeit ist.

Weitere Informationen

Unter <http://www.gerd-stolpmann.de/umlmon> erhalten Sie:

- Die technische Produktbeschreibung von UMLMON
- Die Bedienungsanleitung von UMLMON
- Eine funktionsbeschränkte Demo-Version von UMLMON
- Die Vollversion von UMLMON-Web inkl. Quelltext

Sie finden dort auch die **Bestellunterlagen**. UMLMON ist ausschließlich direkt bei Gerd Stolpmann erhältlich.

Für weitere Informationen wenden Sie sich bitten an folgende Mail-Adresse:

- anfrage@gerd-stolpmann.de

Gerd Stolpmann unterstützt Sie gerne auch persönlich bei der Einrichtung Ihrer UML-Architektur.

Daten UMLMON

Systemvoraussetzungen:	
	<ul style="list-style-type: none"> ● Linux-System für x86, Kernel 2.4 oder 2.6 ● 256 MB Hauptspeicher oder besser ● glibc Version 2.3 oder besser
Skill-Voraussetzungen:	
	Kenntnisse in Linux-Systemadministration sind unabdingbar
Lieferumfang:	
	<ul style="list-style-type: none"> ● UMLMON wird nur elektronisch geliefert (Download), der Anwender muss einen Lizenzschlüssel einspielen. ● Enthalten sind der umlmon-Dämon, der umldir-Dämon sowie die Administrationswerkzeuge umladmin und umlwatch. Diese Programme werden nur als Binary geliefert. ● Enthalten ist außerdem die IDL-Definition der RPC-Steuerung im Quelltext. ● Die Lieferung erfolgt als tar-Datei mit Installationskript. Eine Anleitung kann separat heruntergeladen werden. ● Es erfolgt keine Lieferung von Linux-Kerneln mit UML- oder SKAS-Patch. (D.h. die UMLMON-Lieferung enthält nicht UML! Anwender müssen sich gepatchte Kernel selber übersetzen oder aus geeigneter Quelle besorgen.) ● Es erfolgt keine Lieferung von bootfähigen Images.
Lizenzbedingungen	
	UMLMON wird generell mit einer Lizenz geliefert, die den unternehmensweiten Einsatz erlaubt.
Funktionen:	
	<ul style="list-style-type: none"> ● Start und Stopp von virtuellen Instanzen, sowohl manuell als auch per Boot-Skript ● Logging des Monitors sowie der virtuellen Konsolen, inkl. Log-Rotation ● Konsolen-Konzentrator ● Privilegien-Separation: Die virtuelle Instanz kann als unprivilegierter Benutzer ablaufen. Der Monitor läuft allerdings immer als root-Benutzer. ● Ablauf der virtuellen Instanz in einem chroot-Jail ● Die Administration der virtuellen Instanz durch einen unprivilegierten Benutzer ist weitgehend möglich. ● Die Steuerung des Monitors per RPC ist möglich (Netzwerk-Steuerung). Das RPC-Protokoll ist dokumentiert.
Limitierungen:	
	<ul style="list-style-type: none"> ● Die Größe des virtuellen Hauptspeichers ist auf ca. 1GB pro Instanz beschränkt.
Ausführung:	
	<ul style="list-style-type: none"> ● UMLMON ist in der Programmiersprache Objective Caml entwickelt worden. ● Die parallele Abarbeitung von Anforderungen erfolgt durch klassisches Unix-Multiplexing.

Daten UMLMON-Web

Systemvoraussetzungen:	
	<ul style="list-style-type: none"> ● Linux-System für x86, Kernel 2.4 oder 2.6 ● 256 MB Hauptspeicher oder besser ● glibc Version 2.3 oder besser
Skill-Voraussetzungen:	
	Kenntnisse in Linux-Systemadministration und Web-Administration sind unabdingbar
Lieferumfang:	
	<ul style="list-style-type: none"> ● UMLMON-Web wird nur elektronisch geliefert (Download) ● Enthalten sind übersetzte Fassungen von UMLMON-Web für cgi, sowie der Quelltext. Zur eigenständigen Übersetzung sind weitere Tools und Libraries erforderlich, die nicht Bestandteil der Lieferung sind. ● Die Lieferung erfolgt als tar-Datei mit Installationskript. Eine Anleitung ist enthalten.
Lizenzbedingungen	
	UMLMON-Web wird generell unter den Bedingungen der General Public License geliefert.
Funktionen:	
	<ul style="list-style-type: none"> ● Administration von UMLMON-Instanzen, die auf beliebigen im LAN erreichbaren Rechnern ablaufen ● Einbindung in einen Web-Server per CGI. ● Das Web-Layout kann durch Editieren von XML-Dateien an Kundenbedürfnisse angepasst werden.
Ausführung:	
	<ul style="list-style-type: none"> ● UMLMON-Web ist in der Programmiersprache Objective Caml entwickelt worden.