

Praktikum angewandte Systemsoftwaretechnik (PASST)

Arbeitsumgebung / Aufgabe 2

26. April 2018

Stefan Reif, Peter Wägemann, Florian Schmaus, Michael Eischer,
Andreas Ziegler, Bernhard Heinloth und Benedikt Herzog

Lehrstuhl für Informatik 4
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme
und Betriebssysteme



FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT

Arbeiten mit der VM

SSH-Authentifizierung mit einem Schlüsselpaar ohne Passwort

- ssh-keygen erzeugt **privaten** und **öffentlichen** Schlüssel

```
01 $ ssh-keygen -f <gruppen_name> -N ""
02 Generating public/private rsa key pair.
03 Your identification has been saved in <gruppen_name>.
04 Your public key has been saved in <gruppen_name>.pub.
05 (...)
```

- Generierte Schlüssel (gruppen_name = gruppe0)

```
01 $ ls -l
02 -rw----- 1 thoenig users 1675  8. Mai 11:29 gruppe0
03 -rw-r--r-- 1 thoenig users  394  8. Mai 11:29 gruppe0.pub
```

- Port-Weiterleitung für SSH-Verbindungen (QEMU)

```
01 $ qemu <...> -net user,hostfwd=tcp:127.0.0.1:5022-:22
```

In der Betriebsumgebung des Host-Rechners ausführen!

SSH-Authentifizierung mit Schlüsseln (VM)

- Installation des SSH-Servers in der virtuellen Maschine

```
01 $ apt-get install ssh openssh-server
```

- Zugriffe auf die virtuelle Maschine unter Zuhilfenahme des generierten **öffentlichen** Schlüssels
- Hinterlegen des **öffentlichen** Schlüssels

```
01 $ su - <vm_user>  
02 $ mkdir .ssh  
03 $ scp <user>@<host_ip>:~/<gruppen_name>.pub \  
04 /home/<vm_user>/.ssh/authorized_keys
```

Alternative: `ssh-copy-id(1)`

In der Betriebsumgebung der virtuellen Maschine ausführen!

Verbindungsaufbau vom Host zur virtuellen Maschine

■ Kontrollverbindung zur virtuellen Maschine aufbauen

```
01 $ ssh -p 5022 -i <gruppen_name> \  
02     <vm_user>@localhost
```

■ Datenverbindung

```
01 $ scp -P 5022 <datei1> [<datei2>] \  
02     <vm_user>@localhost:~/<vm_path>
```

■ Alternative: SSHFS (benötigt root-Rechte!)

```
01 $ sshfs -p 5022                               \  
02     -o IdentityFile=<absolute_path>/<gruppe_name> \  
03     <vm_user>@localhost:~/<vm_path>          \  
04     <mount_point>
```

Hinweis bei Verwendung von `sshfs(1)`: Absoluter Pfad zum Schlüssel zwingend notwendig

- *Plan 9 Folder Sharing* erlaubt transparentes Einhängen freigegebener Verzeichnisse des Hosts
- Konfiguration des Gast-Kernels

```
01 CONFIG_NET_9P=y
02 CONFIG_NET_9P_VIRTIO=y
03 #CONFIG_NET_9P_DEBUG=y
04 CONFIG_9P_FS=y
05 CONFIG_9P_FS_POSIX_ACL=y
```

Zugriff auf freigegebene Verzeichnisse des Hosts (2/2)

- Freigabe von Verzeichnissen mit QEMU

```
01 (host) $ mkdir shared
02 (host) $ qemu <...> -virtfs local,path=shared,\
03     security_model=none,mount_tag=shared
```

Optional: ,readonly

- Einhängen der freigegebenen Verzeichnisse mit mount

```
01 (guest) $ mkdir /mnt/shared
02 (guest) $ mount -t 9p -o trans=virtio,version=9p2000.L \
03     shared /mnt/shared
```

- Automatisches Einhängen über Konfiguration in /etc/fstab im Gast-System:

```
01 shared /mnt/shared 9p trans=virtio,version=9p2000.L 0 0
```

Unterschiede zwischen Quelltexten

Grundlegendes Werkzeug um Unterschiede zweier Programm-Code-Revisionen zu erhalten: `diff(1)`

- zeilenorientiert
- bevorzugter Modus: „unified diff“

Beispiel: „Normaler“ Diff

```
01 $ diff -r a b
02 diff -r a/linux/drivers/staging/crystalhd/crystalhd_hw.c
03         b/linux/drivers/staging/crystalhd/crystalhd_hw.c
04 87c87
05 <     * (63MHz * 40us = 0x9D8)
06 ---
07 >     * (63MHz * 20us = 0x4EC)
08 89c89
09 <     crystalhd_reg_wr(adp, GISB_ARBITER_TIMER, 0x9D8);
10 ---
11 >     crystalhd_reg_wr(adp, GISB_ARBITER_TIMER, 0x4EC);
```

Beispiel: „Vereinheitlichter“ Diff (unified diff)

```
01 $ diff -ur a b
02 --- a/linux/drivers/staging/crystalhd/crystalhd_hw.c 16:55
03 +++ b/linux/drivers/staging/crystalhd/crystalhd_hw.c 16:57
04 @@ -84,9 +84,9 @@
05     /*
06      * Bus Arbiter Timeout: GISB_ARBITER_TIMER
07 - * (63MHz * 40us = 0x9D8)
08 + * (63MHz * 20us = 0x4EC)
09     */
10 - crystalhd_reg_wr(adp, GISB_ARBITER_TIMER, 0x9D8);
11 + crystalhd_reg_wr(adp, GISB_ARBITER_TIMER, 0x4EC);
12
13     /*
14      * Decoder clocks: MISC_PERST_DECODER_CTRL
15 Only in b/linux/drivers/staging/crystalhd: crystalhd-fix.msg
```

Beispiel: unified diff mit -N und -p

```
01 $ diff -urNp a b
02 --- a/linux/drivers/staging/crystalhd/crystalhd_hw.c 16:55
03 +++ b/linux/drivers/staging/crystalhd/crystalhd_hw.c 16:57
04 @@ -84,9 +84,9 @@ static bool crystalhd_bring_out_of_rst(s
05 /*
06  * Bus Arbiter Timeout: GISB_ARBITER_TIMER
07 - * (63MHz * 40us = 0x9D8)
08 + * (63MHz * 20us = 0x4EC)
09 */
10 - crystalhd_reg_wr(adp, GISB_ARBITER_TIMER, 0x9D8);
11 + crystalhd_reg_wr(adp, GISB_ARBITER_TIMER, 0x4EC);
12
13 /*
14  * Decoder clocks: MISC_PERST_DECODER_CTRL
15 diff -urNp a/linux/drivers/staging/crystalhd/crystalhd-fix.msg
16         b/linux/drivers/staging/crystalhd/crystalhd-fix.msg
17 --- a/linux/drivers/staging/crystalhd/crystalhd-fix.msg 01:00
18 +++ b/linux/drivers/staging/crystalhd/crystalhd-fix.msg 17:10
19 @@ -0,0 +1 @@
20 +Initial patch description.
```

Diffs gezielt durchsuchen (1/2)

- Funktioniert auch mit gepackten Diffs
- Kann führende Unterverzeichnisse überspringen

```
01 $ lsdiff -z --strip=1 big.gz | grep ^tools
02 tools/perf/.gitignore
03 tools/perf/CREDITS
04 tools/perf/Documentation/Makefile
05 tools/perf/design.txt
06 tools/perf/perf.c
07 tools/perf/perf.h
08 [...]
```

Änderungen auf Unterverzeichnisse beschränken:

```
01 $ filterdiff -z --strip=1 -i '*/tools/perf/*' big.gz | lsdiff
02 tools/perf/.gitignore
03 tools/perf/CREDITS
04 tools/perf/Documentation/Makefile
05 tools/perf/Documentation/asciidoc.conf
06 [...]
```

Größe von Diffs ermitteln

Ausgabe aus dem Kommando `diff` kann direkt weiterverwendet werden:

```
01 $ diff -Nurp linux-2.6.18/ linux-2.6.32/ | diffstat
02 .gitignore | 36
03 .mailmap | 107
04 CREDITS | 367
05 Documentation/00-INDEX | 240
06 [...]
07 virt/kvm/ioapic.c | 362
08 virt/kvm/ioapic.h | 76
09 virt/kvm/iodev.h | 70
10 virt/kvm/iommu.c | 233
11 virt/kvm/irq_comm.c | 422
12 virt/kvm/kvm_main.c | 2830 +
13 36994 files changed, 7819725 insertions(+), 3043036 deletions(-)
```

Aufgabe 2

- Zusammenstellung von Linux-Kernel mit Userland
- Paketverwaltung zur Installation von Software
- Für bestimmte Aufgaben zugeschnitten

- Zusammenstellung von Linux-Kernel mit Userland
- Paketverwaltung zur Installation von Software
- Für bestimmte Aufgaben zugeschnitten

Aufgabe 2

- Veränderungen des Linux-Kernel gegenüber dem „Standard-Kernel“
- Art, Umfang und Gründe der Änderungen
- technische Verfahren diese Änderungen einzubauen
- organisatorische Verfahren drumrum

Aufgabe 2

- Mainline: Linux **3.2, 3.5, ...** (passend je nach Distribution)
- Distributionen / Projekte:
 - Arch
 - Clear Linux
 - Gentoo
 - Grml
 - OpenELEC
 - openSUSE / SLES
 - Puppy Linux
 - Redhat / CentOS / Fedora
 - Tiny Core Linux
 - Yocto Linux

Präsentation der Ergebnisse als Vortrag am **18. Mai**

Aufgabe 2

- Auswahl eines Vortragsthemas
- Recherche zu den vorgestellten Themen
- Präsentation der Ergebnisse als Vortrag
 - Folien vorher an die Liste schicken
 - Aufbereitung mit diffstat, Grafiken, ...
 - pro Gruppe maximal bis zu 15 Minuten

Worauf wir achten

- Vorbereitung und Vortrag durch jeweils Beide
- zeitlicher und inhaltlicher Umfang
- verständliche und übersichtliche Darstellung und Sprache
- sorgfältige Vorbereitung
- inhaltliche Korrektheit und Vollständigkeit
- Quellen und Zitierweise

Allgemeine Hinweise zu Vorträgen

- Faustregel: pro Folie 2 Minuten einplanen
- man braucht immer länger, als man denkt
- Probehalten hilft sehr
- Üben beseitigt Nervosität
- Vorführungen, Videos o.Ä. vermeiden, geht zu oft schief
- keine „special effects“, aber visuell ansprechend (Diagramme, Grafiken)

L^AT_EX

- L^AT_EX vereinfacht die Benutzung von T_EX
- TeX entwickelt von Donald E. Knuth (1970er Jahre)
- LaTeX entwickelt von Leslie Lamport (**Lamport TeX**, 1980er Jahre)
- Erweiterbar durch zahllose Pakete

■ Fettschrift

```
01 Hallo \textbf{Welt}
```

→ Hallo **Welt**

■ Kursiv

```
01 Hallo \textit{Welt}
```

→ Hallo *Welt*

■ Festbreite

```
01 Hallo \texttt{Welt}
```

→ Hallo Welt

■ Text hervorheben

```
01 Hallo \emph{Welt}
```

→ Hallo Welt

- Neues Kapitel erstellen (u.U. optional)

```
01 \chapter{KapitelÃ½berschrift}
```

- Neue Unterkapitel erstellen

```
01 \section{Ãberschrift}
02 \subsection{Ãberschrift}
03 \subsubsection{Ãberschrift}
```

- Automatisch Inhaltsverzeichnis erstellen

```
01 \tableofcontents
```

- Paket: `graphicx`
- Einbinden von Bildern (z.B. Abbildung 1)

```
01 \begin{figure}  
02   \includegraphics[width=0.3\textwidth]{fig/git-logo}  
03   \caption{Git Logo}  
04   \label{fig:git-logo}  
05 \end{figure}
```



Abbildung 1: Git Logo

■ ohne Nummerierung

```
01 \begin{itemize}
02   \item Erster Punkt
03   \item Zweiter Punkt
04 \end{itemize}
```

- Erster Punkt
- Zweiter Punkt

■ mit Nummerierung

```
01 \begin{enumerate}
02   \item Erster Punkt
03   \item Zweiter Punkt
04 \end{enumerate}
```

1. Erster Punkt
2. Zweiter Punkt

■ Verwaltung in einer extra Datei

```
01 @inproceedings{kahn1999next,  
02   author = {Kahn, J. M. and Katz, R. H. and  
03             Pister, K. S. J.},  
04   title = {Next Century Challenges: Mobile Networking  
05             for 'Smart Dust'},  
06   booktitle = {Proceedings of the 5th Annual International  
07                 Conference on Mobile Computing and  
08                 Networking (MobiCom)},  
09   year = {1999},  
10   pages = {271--278},  
11 }
```

■ Compilieren per bibtex references.bib

■ Referenzieren im Text

```
01 Kahn et al.~forged the term \emph{Smart Dust}~\cite{  
    ↪ kahn1999next}.
```

Kahn et al. forged the term Smart Dust [1].

- Einbinden per \bibliography



J. M. Kahn, R. H. Katz, and K. S. J. Pister.

Next century challenges: Mobile networking for 'smart dust'.

In Proceedings of the 5th Annual International Conference on Mobile Computing and Networking (MobiCom), pages 271–278, 1999.

LaTeX Beamer

■ Neue Folie

```
01 \begin{frame}{Folienüberschrift}  
02     Hallo Welt  
03 \end{ frame}
```

■ Inhaltsverzeichnis

```
01 \begin{frame}{Folienüberschrift}  
02     \tableofcontents  
03 \end{ frame}
```

Arbeiten mit der VM

Unterschiede zwischen Quelltexten

Aufgabe 2

LaTeX

LaTeXBeamer

■ Zwei benachbarte Boxen

```
01 \begin{minipage}[t]{0.45\textwidth}
02 Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam
    ↳ nonumy eirmod tempor invidunt ut labore et dolore magna
    ↳ aliquyam erat, sed diam voluptua.
03 \end{minipage}
04
05 \begin{minipage}[t]{0.45\textwidth}
06   \centering
07   \begin{itemize}
08     \item Punkt 1
09     \item Punkt 2
10   \end{itemize}
11
12   \includegraphics[width=0.8\textwidth]{fig/git-logo}
13 \end{minipage}
```

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua.

- Punkt 1
- Punkt 2



Animationen

```
01 \begin{itemize}
02   \item Punkt 1
03   \visible<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 3

```
01 \begin{itemize}
02   \item Punkt 1
03   \only<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 3

Animationen

```
01 \begin{itemize}
02   \item Punkt 1
03   \visible<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 2
- Punkt 3

```
01 \begin{itemize}
02   \item Punkt 1
03   \only<2-> {
04     \item Punkt 2
05   }
06   \item Punkt 3
07 \end{itemize}
```

- Punkt 1
- Punkt 2
- Punkt 3

Fragen?