

B. e t t e b s i s s y m - s e t t e n t W / c u k n G l u m i i t t . L i t e r a t e P r o G r a m m i n G



TECHNISCHE HOCHSCHULE NÜRNBERG
GEORG SIMON OHM

Winter semester 2015/16

Dr. Hans-Georg Eßer

h.g.esser@gmx.de

<http://ohm.hgesser.de/>

v1.2, 10/04/2015

Hans-Georg Eßer

- Dipl.-Math. (RWTH Aachen, 1997)
Dipl.-Inform. (RWTH Aachen, 2005) Dr.-Ing.
(FAU Erlangen-Nuremberg, 2015)
- Editor-in-chief of a Linux magazine (since 2000) and author of various Linux books
- Since 2006 teaching assignments at various universities: operating systems, computer architecture, IT basics, system programming, operating system development, etc.
- 2010–2015 PhD student at the Univ. Erlangen-Nuremberg - Topic: "Design, Implementation and Evaluation of the ULIX Operating System"
- ~~Supervision of theses (no longer for LBAs ...)~~

To the lecture (1)

BS development dates

Lecture: Thu 3: 45–5: 15 p.m. HQ 104 Thu 5:

Internship: 30–7: 00 p.m. HQ 104

website

<http://ohm.hgesser.de/>

(Slides, videos, appointments,

Other Information)

To the lecture (2)

Helpful previous knowledge:

- **Lecture Operating Systems**
- **C.** - Basics of programming in C (or C ++, C #, Java)
 - C crash course (in the practical part)
- **Lecture system programming Linux**
- **Unix shell** - Use of the standard shell *bash*
under Linux → Bash crash course

To the lecture (3)

Dates in the winter semester 2015/16

4 SWS, ie

- Lecture / internship: 13 x 180 min. Room HQ
104

Service / web page

- Slides and internship assignments
- Lecture videos

To the lecture (4)

Effort: 5 ECTS (1 ECTS point \approx 25–30 h)

5 x 25 = 125 hours, of which:

- 52 hours of attendance (lecture + internship)
- \leq 13 hours: Working through the exercises and videos on the topics of C programming and bash shell (not applicable for those who listen to "system programming")
- \leq 15 hours: Independent familiarization with LaTeX
- \leq 20 hours: reading articles and specialist books
- \leq 25 hours: final processing of the exercises with completion of the documentation (after the end of the lecture)
- Event a bit more complex than system programming

Contact

Consultation hour: -

Ask:

- directly in the lecture
- or after
- or via email:

h.g.esser@gmx.de

Sep 19 14:20:18 amd64 sshd [20494]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 61557 Sep 19 14:27:41 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 / usr / sbin / cron [29278]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 20 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 / usr / sbin / cron [30103]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 20 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sshd [6516]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62004 Sep 20 12:46:44 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 12:48:41 amd64 sshd [6609]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62105 Sep 20 12:54:44 amd64 sshd [6694]: Accepted rsa for esser from :: ffff: 87.234 .201.207 port 62514 Sep 20 15:27:35 amd64 sshd [9077]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64242 Sep 20 15:27:35 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sshd [10102]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63375 Sep 20 16:37:11 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd [10140]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63546 Sep 21 01:00:01 amd64 / usr / sbin / cron [17055]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG")
Sep 21 01:00:01 amd64 s syslog-ng [7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 / usr / sbin / cron [17878]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 21 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd [31088]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63397 Sep 21 17:43:26 amd64 s syslog-ng [. 7653]: ST ATS: dropped 0
Sep 21 17:53:39 amd64 sshd [31269]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64391
Sep 21 18:43:26 amd64 syslog-ng [7653]: STATS: dropped 0 Sep 21 19:43:26 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 / usr / sbin / cron [4674]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 22 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 / usr / sbin / cron [5499]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 22 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 22 20:23:21 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 / usr / sbin / cron [24739]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 23 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 / usr / sbin / cron [25555]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 23 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sshd [6554]: Accepted publickey for esser from :: ffff: 192.168.1.5 port 59771 ssh2 Sep 23 18:04:05 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 18:04:34 amd64 sshd [6606]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62093 Sep 24 01:00:01 amd64 / usr / sbin / cron [12436]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 24 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 / usr / sbin / cron [13253]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 24 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sshd [20998]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64456 Sep 24 11:15:48 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sshd [23197]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 61330 Sep 24 13:49:08 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_midi_event: unsupported module, tainting kernel. Sep 24 15:42:07 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd [29399]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62566 Sep 24 20:25:31 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 / usr / sbin / cron [662]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 25 01:00:02 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 / usr / sbin / cron [1484]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 25 02:00:02 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd [8889]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64183 Sep 25 10:59:25 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd [8921]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64253 Sep 25 11:30:02 amd64 sshd [9372]: Accepted rsa for esser from :: ffff: 87.234 .201.207 port 62029 Sep 25 11:59:25 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sshd [11554]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62822 Sep 25 14:05:37 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sshd [11586]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62951 Sep 25 14:07:17 amd64 sshd [11608]: Accepted rsa for esser from :: ffff: 87.234 .201.207 port 63392 Sep 25 14:08:33 amd64 sshd [11630]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63709 Sep 25 15:25:33 amd64 sshd [12930]: Accepted rsa for esser from : : ffff: 87.234.201.207 port 62778

1 F.i guide / Motivation

Why BS Development?

- logical continuation of the events operating systems and system programming
- understand,
 - how programs, libraries and operating systems work together
 - how tasks are solved in the BS
 - how the OS enables (multiple) processes to access the hardware
- and that on the basis of specific BS code

Why Unix?

- long tradition (Unix: 1969; C: 1972)
- Unix internals (original Unix or more modern variants such as Linux, * BSD etc.) are well documented and code is freely available
- Clear set of BS features (in the form of system calls)
- small Unix systems can be implemented with little code
 - z. E.g .: no graphical user interface

Properties of ULIX

- 32-bit system (x86)
- Virtual memory for OS and processes, separate address spaces (paging)
- System Call Interface with associated library for UserMode applications (Ulix programs)
- virtual file system (floppy disks and hard drives, drivers for Minix v2)
- Mini shell with built-in tools (pwd, cd, ls, cat, head, cp, mv, rm, ln, diff, hexdump, ps, kill among others)
- Implementation with → Literate programming

Why Literate Programming

- special form of code development and code documentation
- Literate programs are well suited for presenting complex code
 - supports bottom-up and top-down development
 - can also document how the code was created (creative process)
- Getting started with literate programming is easy

Development environment

- Virtual machine with Debian Linux 6.0.1
- Installed tools:
 - C compiler gcc and x86 assembler nasm
 - Left ld, Object file analysis with objdump
 - PC emulators qemu, bochs
 - Lit. Programming: NoWeb (notangle, noweave)
 - Ulix update script: update-ulix.sh
- Register: ulix / ulix
- become root: sudo su (no password)

Sep 19 14:20:18 amd64 sshd [20494]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 61557 Sep 19 14:27:41 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 01:00:01 amd64 / usr / sbin / cron [29278]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 20 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 02:00:01 amd64 / usr / sbin / cron [30103]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 20 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 12:46:44 amd64 sshd [6516]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62004 Sep 20 12:46:44 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 12:48:41 amd64 sshd [6609]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62105 Sep 20 12:54:44 amd64 sshd [6694]: Accepted rsa for esser from :: ffff: 87.234 .201.207 port 62514 Sep 20 15:27:35 amd64 sshd [9077]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64242 Sep 20 15:27:35 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 16:37:11 amd64 sshd [10102]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63375 Sep 20 16:37:11 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 20 16:38:10 amd64 sshd [10140]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63546 Sep 21 01:00:01 amd64 / usr / sbin / cron [17055]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 21 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 21 02:00:01 amd64 / usr / sbin / cron [17878]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 21 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 21 17:43:26 amd64 sshd [31088]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63397 Sep 21 17:43:26 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 21 17:53:39 amd64 sshd [31269]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64391 Sep 21 18:43:26 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 21 19:43:26 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 22 01:00:01 amd64 / usr / sbin / cron [4674]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 22 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 22 02:00:01 amd64 / usr / sbin / cron [5499]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 22 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0 Sep 22 20:23:21 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 01:00:01 amd64 / usr / sbin / cron [24739]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 23 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 02:00:01 amd64 / usr / sbin / cron [25555]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 23 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 18:04:05 amd64 sshd [6554]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 61557 Sep 23 18:04:05 amd64 sshd [6592]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 61597
Sep 23 18:04:05 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 23 18:04:34 amd64 sshd [6606]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62093 Sep 24 01:00:01 amd64 / usr / sbin / cron [12436]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 24 01:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 02:00:01 amd64 / usr / sbin / cron [13253]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 24 02:00:01 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 11:15:48 amd64 sshd [20998]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64456 Sep 24 11:15:48 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 13:49:08 amd64 sshd [23197]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 61330 Sep 24 13:49:08 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_midi_event: unsupported module, tainting kernel. Sep 24 15:42:07 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 24 15:42:07 amd64 kernel: snd_seq_oss: unsupported module, tainting kernel.
Sep 24 20:25:31 amd64 sshd [29399]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62566 Sep 24 20:25:31 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 01:00:02 amd64 / usr / sbin / cron [662]: (root) CMD (/ sbin / evlogmgr -c "severity = DEBUG") Sep 25 01:00:02 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 02:00:01 amd64 / usr / sbin / cron [1484]: (root) CMD (/ sbin / evlogmgr -c 'age> "30d"') Sep 25 02:00:02 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 10:59:25 amd64 sshd [8889]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64183 Sep 25 10:59:25 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 10:59:47 amd64 sshd [8921]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 64253 Sep 25 11:30:02 amd64 sshd [9372]: Accepted rsa for esser from :: ffff: 87.234 .201.207 port 62029 Sep 25 11:59:25 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 14:05:37 amd64 sshd [11554]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62822 Sep 25 14:05:37 amd64 syslog-ng [7653]: STATS: dropped 0
Sep 25 14:06:10 amd64 sshd [11586]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62951 Sep 25 14:07:17 amd64 sshd [11608]: Accepted rsa for esser from :: ffff: 87.234 .201.207 port 63392 Sep 25 14:08:33 amd64 sshd [11630]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 63709 Sep 25 15:25:33 amd64 sshd [12930]: Accepted rsa for esser from :: ffff: 87.234.201.207 port 62778

Guideline

Outline (1)

1. introduction
2. Introduction to C and the Linux shell Bash (*Reading / exercises outside of the lecture*)
3. Features of a Unix system (overview) LaTeX and
- 4th Literate Programming
5. Booting, protected mode, processes, memory interrupts
- 6th and system calls in Ulix
- 7th fork () - Implementation in the kernel
- 8th. File systems (*with reading outside d. Template*)
9. Implementation project

Outline (2)

- For some topics first an overview of the basics (theory of the operating systems)
- Presentation of the features that UNIX offers here
- Discussion of code snippets
- Programming tasks in addition

2. Crash courses C and Bash

structure

- 1 Introduction
2. C and Bash
3. Unix features
4. LaTeX / LitProg
5. Booting, processes,
Storage
6. Interr., Syscalls
7. fork ()
8. File systems
- 9th project

- If you have no knowledge of C and / or the Unix shell (bash) use (outside of the classroom appointments) the ones offered Crash courses and videos.
- I answer (by mail) questions about C / bash, that occur when processing the tasks.

3. Features of a Unix system

structure

- 1 Introduction
2. C and Bash
3. Unix features
4. LaTeX / LitProg
5. Booting, processes,
Storage
6. Interr., Syscalls
7. fork ()
8. File systems
- 9th project

- Overview of Unix / UNIX
- processes
- Storage (for OS and processes) file
- system, VFS
- Interrupt handling
- System calls
- Shell
- Scheduler

4. LaTeX and Literate Programming

structure

- 1 Introduction
2. C and Bash
3. Unix features
4. LaTeX / LitProg
5. Booting, processes,
Storage
6. Interr., Syscalls
7. fork ()
8. File systems
- 9th project

- (Ultra) short introduction to LaTeX
- Principle of literate programming
- Small examples with LaTeX and HTML
- Exercises with NoWeb
(notangle, noweave)

5. Booting, protected mode, processes, memory

structure

- 1 Introduction
- 2. C and Bash
- 3. Unix features
- 4. LaTeX / LitProg
- 5. Booting, processes, Storage
- 6. Interr., Syscalls
- 7. fork ()
- 8. File systems
- 9th project

- How does a (x86) PC boot?
- x86: Real Mode vs. Protected mode
- Processes
 - Process control block
 - Process creation
 - Scheduling / context switch
- Storage
 - Paging: address ranges, Side tables

6. System calls in ULIX

structure

- 1 Introduction
2. C and Bash
3. Unix features
4. LaTeX / LitProg
5. Booting, processes,
Storage
6. Interr., Syscalls
7. fork ()
8. File systems
- 9th project

- Interrupts on x86 (in Prot. Mode)
- ULIX: Interrupt Handler
- Example: timer handler
- Software interrupt (int 0x80)
- ULIX: System Call Interface
- Example: Syscall for read ()

7. Process duplication with fork ()

structure

- 1 Introduction
- 2. C and Bash
- 3. Unix features
- 4. LaTeX / LitProg
- 5. Booting, processes,
Storage
- 6. Interr., Syscalls
- 7. fork ()
- 8. File systems
- 9th project

- fork () at all levels:
 - fork () - Implementation in the ULIX Kernel
 - What is being copied?
 - What will be changed?
 - How does ULIX activate the new process?
 - fork () as a library function
 - fork () in program

8. File systems

structure

- 1 Introduction
2. C and Bash
3. Unix features
4. LaTeX / LitProg
5. Booting, processes,
Storage
6. Interr., Syscalls
7. fork ()
8. File systems
- 9th project

- Multiple layers
 - Hardware: Access to floppy disks and hard drives (readblock, writeblock)
 - Logical: Minix v2 file system
 - Superblock, maps, inodes, files, directories (with content from the lecture system programming; Reading)
 - Indirection blocks
 - Virtual file system
 - User mode functions (open, read, write Etc.)

9th project

structure

- 1 Introduction
2. C and Bash
3. Unix features
4. LaTeX / LitProg
5. Booting, processes,
Storage
6. Interr., Syscalls
7. fork ()
8. File systems
- 9th project

- Depending on the pace with which we get through the previous topics:

Start in the current semester or as a final assignment

- Medium-sized implementation project
- Documentation in literary
Programming style

literature

- H.-G. Eßer, FC Freiling: The Design and Implementation of the ULIX Operating System, 2015 (available on course website)
- F. Mittelbach, M. Goossens: The LaTeX Companion, 2010, ISBN: 386894088X
- DE Knuth: Literate Programming, The Computer Journal, 27 (2), pp. 97-111, 1984, <http://literateprogramming.com/knuthweb.pdf>
- N. Ramsey: Literate Programming Simplified, IEEE Software, 11 (5): 97-105, 1994, <http://literateprogramming.com/lpsimp.pdf>
- N. Ramsey: A One-Page Guide to Using noweb with LaTeX, <http://literateprogramming.com/nowebpg1.pdf>
- AS Tanenbaum, AS Woodhull: Operating Systems Design and Implementation, 3rd edition, ISBN 0131429388 or 0135053765 (TB)
- WR Stevens, SA Rago: Advanced Programming in the Unix Environment, 2013, ISBN 0321637739
- Helmut Herold: Linux / Unix Systemprogrammierung, 2004, ISBN: 3827321603
- Elias Fischer: The C-Tutorial, <http://www.c-howto.de/>
- Ted Jensen: A Tutorial on Pointers and Arrays in C, 2003, <http://home.earthlink.net/~momotuk/pointers.pdf>