

# FERN-E-PRÜFUNGEN

rechnergestützte Klausuren aufwandsarm  
und ausfallsicher durchführen

Prof. Dr.

**Dominik Herrmann**

@herdom · herdom.net

Otto-Friedrich-Universität Bamberg



en!). Bitte das Testimage oder Vulnerable keinesfalls für Angriffe

Bearbeiten

www-sec.uni-regensburg.de

Terminal Shell Bearbeiten Ansicht Fenster Hilfe Manual Pages: ssh-add(1)

http://www.openbsd.org/cgi-bin/man.cgi?query=ssh-add&sektion=1

Terminal -- ssh-add -- 80x24

```
ssh-agent add a ssh ke... r not known
moia:ssh sm$ ssh root@10.19.1.5
Permission denied (publickey).
moia:ssh sm$ cd vulnerable/
moia:vulnerable sm$ ls
vulnerable  vulnerable.ppk  vulnerable.pub
moia:vulnerable sm$ chmod 550 *
moia:vulnerable sm$ ls
vulnerable  vulnerable.ppk  vulnerable.pub
moia:vulnerable sm$ cd ..
moia:ssh sm$ ssh-add vulnerable/vulnerable.pub
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@ WARNING: UNPROTECTED PRIVATE KEY FILE!
@
Permissions 0550 for 'vulnerable/vulnerable.pub'.
It is recommended that your private key files are NOT accessible by others.
This private key will be ignored.
moia:ssh sm$ chmod 700 vulnerable/*
moia:ssh sm$ ssh-add vulnerable/vulnerable.pub
Enter passphrase for vulnerable/vulnerable.pub:
moia:ssh sm$ ssh root@10.19.1.5
Permission denied (publickey).
moia:ssh sm$ ssh-add vulnerable/vulnerable.pub
Enter passphrase for vulnerable/vulnerable.pub:
```

**ENVIRONMENT**

**DISPLAY and SSH\_ASKPASS**

If `ssh-add` needs to display a password prompt, it will use the `SSH_ASKPASS` program if it is set in the `ENVIRONMENT`. If `SSH_ASKPASS` is not set, it will use the `ssh-add` program's built-in prompt.

**SSH\_AUTH\_SOCK**

Identifies the location of the `ssh-agent` socket to use for authentication.

**FILES**

- `~/.ssh/identity`  
Contains the protocol version 1 authentication identity of the user.
- `~/.ssh/id_dsa`  
Contains the protocol version 2 DSA authentication identity of the user.
- `~/.ssh/id_rsa`  
Contains the protocol version 2 RSA authentication identity of the user.

Identity files should not be readable by anyone but the user. Note that `ssh-add` ignores identity files if they are accessible by others.

**DIAGNOSTICS**

Exit status is 0 on success, 1 if the specified command fails, and 2 if `ssh-add` is unable to contact the authentication agent.

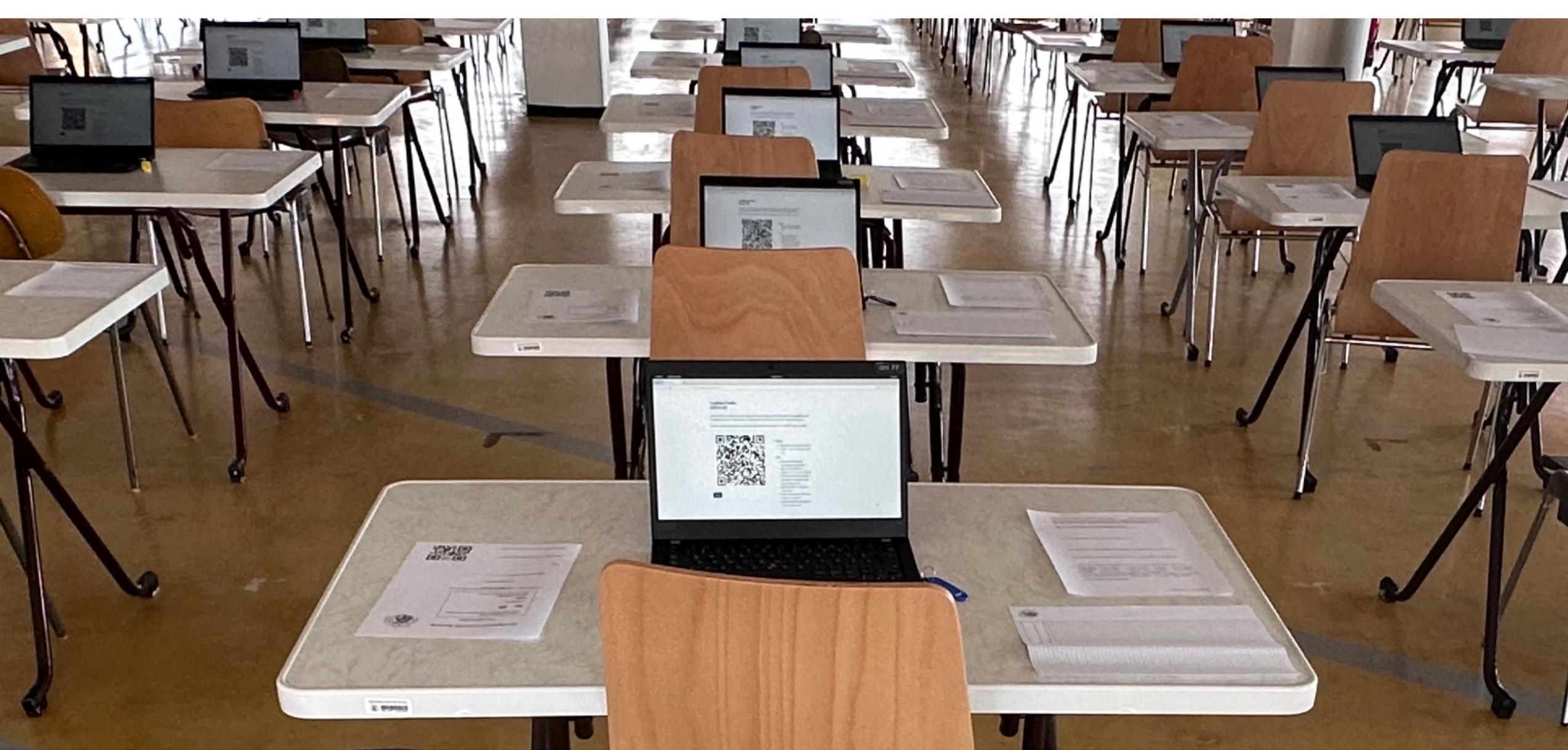
**SEE ALSO**

[ssh\(1\)](#), [ssh-agent\(1\)](#), [ssh-keygen\(1\)](#), [sshd\(8\)](#)

**AUTHORS**

OpenSSH is a derivative of the original and free ssh 1.2.12 release by Tatu Ylonen. Aaron Campbell, Bob Beck, Markus Friedl, Niels Provos, Theo de Raadt and Dug Song removed many bugs, re-added newer features and created the OpenSSH project. Markus Friedl contributed the support for SSH protocol

# E-Prüfung Uni Regensburg vor 18 Jahren



**Vision: einfacher e-prüfen in Präsenz und zu Hause**

# AUSGANGSLAGE



**Wunsch:** (Fern-)E-Prüfungen  
direkt am Rechner

vorhanden: Moodle, EvaExam

**Kompetenzorientiert prüfen** mit  
beliebigen Fachanwendungen –  
**ohne Eingriff in private Rechner**

„**termingleiche Präsenzprüfung**“ ...  
„unter strenger Beachtung der  
Grundsätze der **Chancengleichheit**“

### Stream Chat

93+

NixenSays: bounty!!!!

Roasee009: LOOOOOL

Roasee009: I think hes gettin off early 2day

MarkenwithME: I mean he's not wrong?

claytontheRAD: he already ran back

Asam001: hulk is marrying his niece

anotherwiseguy: I remember my first time reading too

lukone53: Faxxxxx!!!

sakquasm01: anyone above me in TP

clovethefridge: It's like opening kinquer eggs!

maly2200: Who was your 6th?

fari\_ubername60: Message from future mah mam!

dumgyle64: Pete has enough time to comeback

haxk123: Only if you added up and averaged it out...

Silktherick: Cant wait to watch you sleep

Claythefridge: the realest game changer... for real

jpianopuy91: Jst got 88 Abuino & 88 Marta

Send



### YOUR COMMUNITY

Write something you're proud of

#### VIP members

- JackieTheMonster
- Mikesparta
- youcantseeme
- alberon332
- 123KreamX321
- Dilimanko999

#### Members

- babyto
- shelton
- plowch
- calliana
- clayton68
- breves12
- shelton
- theBoltz
- malwarex13
- Junkyle1994
- markenwithME
- markenwithME

Your chat has been slowed down as there are too many messages being sent at once

# AUSGANGSLAGE



CIP-Pools ungeeignet

kein **Blickschutz**, zu geringer **Sitzabstand**,  
Nutzung **schwer einschränkbar**, hoher  
**Koordinationsaufwand** und Aufsichtenbedarf

Einschränkungen in Prüfungsräumen

kein zuverlässiges **Netz**, kein **Strom**, keine  
**dauerhafte Installation** möglich

**Wunsch:** flexible Klausuren mit  
neuen didaktischen Konzepten

# ZIEL UND IDEE

Lösung für Fern-E-Prüfungen

gleiche Hilfsmittel und Werkzeuge  
zu Hause und in Präsenz

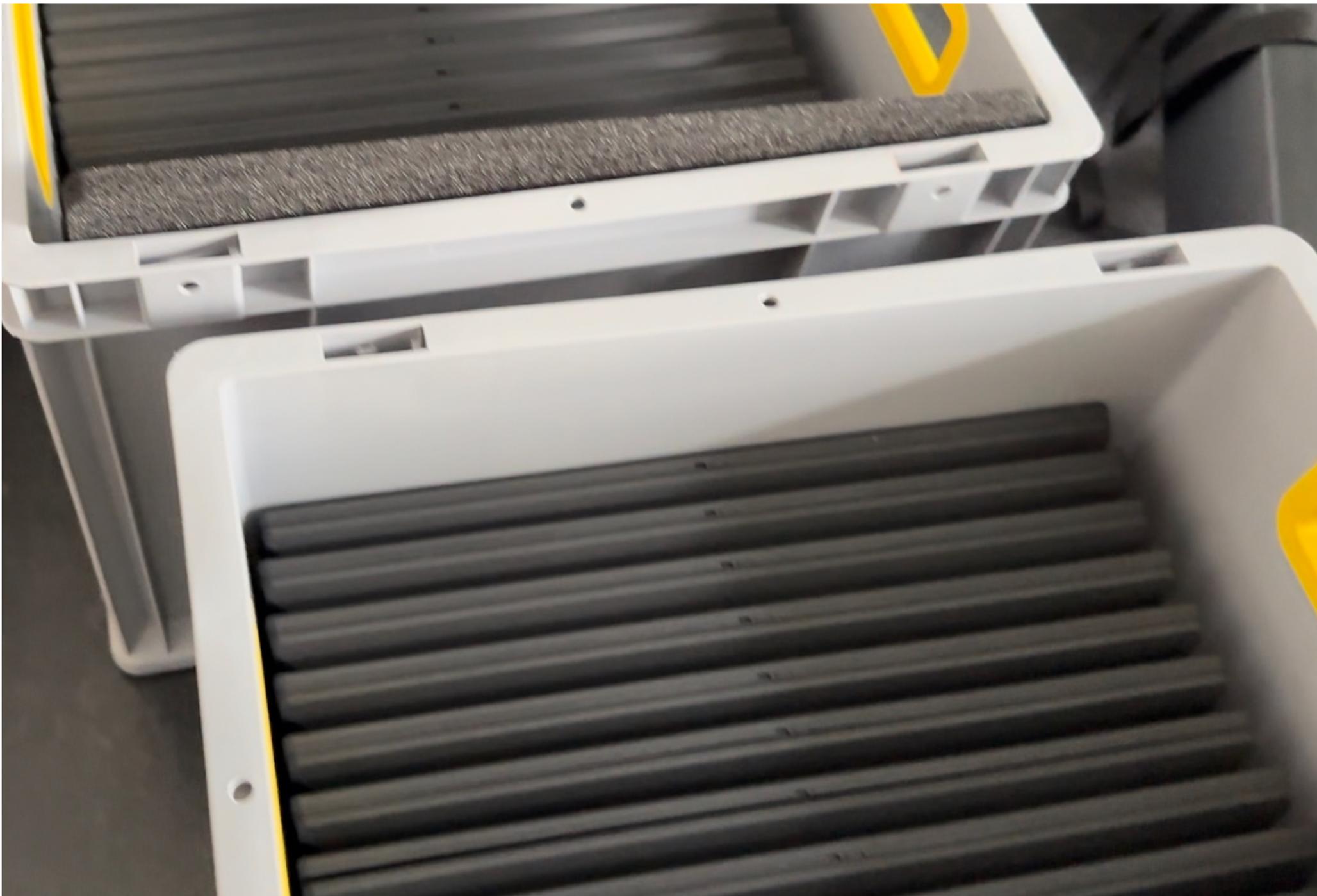
Präsenzprüfung

**ohne stabiles Netz und Strom**

gleichzeitige Fernprüfung

Verwendung speziell präparierter  
verliehener **Laptops**

UMSE





# UMSETZUNG



Finanzierung von **40 Stk. Lenovo T14**  
14"-Display mit 400 nits, Core i5 2,4 GHz,  
50 Wh Akku, Laden über USB-C

**WLAN:** Aruba AP-505; upscreens Spy  
Shield, gepolsterte Euroboxen 40x30cm

**Software:** Debian, Gnome 42, Firefox

**Softwareverteilung** via Multicast mit  
Clonezilla (USB-Stick bzw. PXE-Boot)  
an 48-Port-Switch (Dauer: 90 min)

**Laden:** alle auf einmal an IP-Steckdose  
(GUDE 8041-1) mit Einschaltverzögerung  
( $40 \cdot 65 \text{ W} = 2600 \text{ W}$  bzw. 12 A).

**<section>**

**<p>**a) Der fiktive Prozessor P22 arbeitet mit einer Taktrate von 2,2 MHz. Er verwendet keine Pipeline. Zur Abarbeitung von Instruktionen benötigt der P22 entweder einen, zwei, vier oder sechs Clock Cycles. Instruktionen der Gruppe IG-1 werden in **<em>einem</em>** Clock Cycle abgearbeitet. Instruktionen der Gruppe IG-2 benötigen hingegen **<em>zwei</em>** Clock Cycles, Instruktionen der Gruppe IG-4 **<em>vier</em>** Clock Cycles und Instruktionen der Gruppe IG-6 **<em>sechs</em>** Clock Cycles. Zu jedem Zeitpunkt wird höchstens eine Instruktion auf dem P22 ausgeführt.

**</p>**

**<p>**Ein Programm bestehe aus den folgenden Instruktionen, die alle abgearbeitet werden müssen:  
erstens  $8,1 \cdot 10^7$  Instruktionen aus IG-2 und  
zweitens  $3,7 \cdot 10^6$  Instruktionen aus Gruppe IG-4.

**</p>**

**<p>**Wie viele Minuten benötigt ein P22 im Idealfall für die Ausführung des Programms?  
**<em>Rechenweg erforderlich. Runden Sie das Ergebnis auf zwei Nachkommastellen.</em>**

**</p>**

**<p>**`{{ task_input("2a_calculation", multiline=True, cols=50, rows=8) }}`

**</p>**

**</section>**

**<aside>**  
**<strong>**4 Punkte**</strong>**  
`{{ i_dont_know_box("task2a") }}`  
**</aside>**

**AUFGABENERSTELLUNG  
VORERST MIT HTML/CSS/JS**

# CHANCEN- GLEICHHEIT



Laptops werden ohne Prüfungsaufgaben ausgegeben.

**Kein hoher Schutzbedarf.**

Aufgaben werden vor Beginn der Bearbeitungszeit **verschlüsselt** übermittelt.

Voraussetzung: **(instabiles) WLAN in der Präsenzprüfung.**

Einheitlicher Bearbeitungsbeginn durch Ansage und Eingabe eines **Startcodes.**

# BEARBEITUNG IM BROWSER

Prüfung wird von Webserver, der auf Laptops läuft (localhost), bereitgestellt. Kein **stabiles** Netz erforderlich.

**Präsenz:** Laptops in isoliertem WLAN (Aufsichtenlaptop)

**Fernprüfung:** Firewall-Regeln schränken Internetnutzung ein.

Lösungen werden auf SSD und USB-Stick gesichert und im Hintergrund kontinuierlich an Aufsichtenrechner übermittelt. In Präsenz kann Rechner schnell getauscht werden.

## Studierendendaten

Sie haben folgende Daten eingegeben. Bitte wählen Sie den nächsten Menüpunkt um die Prüfungsaufgaben freizuschalten.

Sie haben sich vertippt? Das ist kein Problem, solange ihr Name noch erkennbar ist (z.B. Sopia anstelle von Sophie).

Matrikelnummer: **1518212**

Vorname: **Theresa**

Nachname: **Testmeier**

# ENTSCHLÜSSELN

## Prüfung freischalten

Geben Sie den Entschlüsselungscode der Prüfung ein. Dieser Code wird vor dem Beginn der Bearbeitungszeit von den Aufsichten ausgeteilt. Sie können den Code bereits *vor* dem Beginn der Bearbeitungszeit eingeben.

Entschlüsselungscode der Prüfung

Sobald Sie den korrekten Entschlüsselungscode eingeben haben und die Aufgaben entschlüsselt wurden, können Sie [hier](#) die Prüfsumme der Aufgaben überprüfen.

# BEARBEITUNG BEGINNT

## Startcode eingeben

Sie bekommen den Startcode von der Aufsicht, sobald die Bearbeitungszeit beginnt.

Startcode der Prüfung

# KONTROLLE DER AUFGABEN

## Deckblatt

### PSI-EiRBS-B: Einführung in Rechner und Betriebssysteme (Sommersemester 2023)

Bearbeitungszeit der Probeklausur: **60 Minuten**, max. Punktzahl: **84 Punkte**, Bestehensgrenze: **42 Punkte**. Anzahl der Aufgaben: **4**. Bitte antworten Sie stichpunktartig und präzise.

Bei jeder Teilaufgabe gibt es ein „Weiß ich nicht“-Kästchen. Wenn dieses Kästchen angekreuzt ist , erhalten Sie 1 Punkt für die Teilaufgabe, unabhängig von Ihrer/n Antwort(en) in dieser Teilaufgabe (Antworten werden bei der Korrektur ignoriert). Sie können das Kästchen bis zu zehn Mal ankreuzen. Wenn mehr als zehn Kästchen angekreuzt sind, wird die oben erwähnte Regel nur auf die ersten zehn Teilaufgaben mit angekreuzten Kästchen angewendet.

**Zugelassene Hilfsmittel:** das für die Prüfung bereitgestellte Laptop mit den darauf befindlichen Programmen, nichtprogrammierbarer Taschenrechner und das selbst mitgebrachte eigenständig handschriftlich erstellte Booklet.

### Prüfsumme der Klausuraufgaben

Prüfsumme der Klausuraufgaben: **M7U**.

Vergleichen Sie diese Prüfsumme mit der Angabe auf dem ausgeteilten Zettel. Sollte die Prüfsumme nicht übereinstimmen, melden Sie sich bitte unverzüglich bei einer Aufsicht.

# BEDIENUNG

## Anleitung

Sie schreiben alle Ihre Antworten in Eingabefelder. Jedes Eingabefeld hat einen eigenen Knopf, um den aktuellen eingegeben und angezeigten Wert zu speichern. Die Umrandung der Eingabefelder haben folgende Bedeutungen.

1. Antwort ist **nicht** gespeichert (blauer Rahmen), aktiver Speichern-Knopf.

Beispiel 1

2. Antwort wird gerade gespeichert (nach Druck auf Speichern-Knopf), beide Elemente sind deaktiviert. Dies sollte nur wenige Sekunden dauern. Falls dies nicht der Fall ist, melden Sie sich bei der Aufsicht.

Beispiel 2

3. Antwort ist gespeichert (grüner Rahmen), deaktivierter Speichern-Knopf.

Beispiel 3

4. Während des Speicherns ist ein Fehler aufgetreten (roter Rahmen). Versuchen Sie erneut zu speichern. Falls der Fehler wieder auftritt, melden Sie sich bei der Aufsicht.

Beispiel 4

**Hinweis:** Falls Sie leer abgeben möchten und Ihre Antworten **nicht** bewertet werden sollen, können Sie dies nach Beginn der Bearbeitungszeit [hier](#) tun.

# FEEDBACK WÄHREND EINGABE MÖGLICH

Wird als hilfreich  
empfunden.

## Aufgabe 1: Daten und Repräsentation (16 Punkte)

a) Berechnen Sie „ $-5 - 7$ “ im 2-Komplement. Interpretieren Sie auch das Ergebnis im 2-Komplement. Für die Darstellung einer Zahl stehen 4 Bit zur Verfügung. *Hinweis: Bitte geben Sie für diese Aufgabe den Rechenweg an.*

„ $-5$ “ im 2-Komplement

Ihre aktuelle Eingabe sieht gut aus! Weiter so!

„ $-7$ “ im 2-Komplement

Ergebnis in Binärdarstellung

Ergebnis als Dezimalzahl

2  
Punkte



Weiß  
ich  
nicht.

# SCREENSHOTS

Speichern

Sie möchten nun ausgeben, wie viele verdächtige Prozesse es gibt. Wie lautet ein geeigneter Befehl?

Speichern

f) Abschließend möchten Sie alle verdächtigen Prozesse beenden. Zum Beenden soll für jeden Prozess der Befehl `kill` aufgerufen werden, um dem Prozess das Signal `KILL` zu senden. Verwenden Sie dazu eine *while*-Schleife. Die PIDs der Prozesse stehen zeilenweise in der Datei `~/prozesse.txt`. *Im Folgenden sehen Sie wie eine while-Schleife (für einen anderen Zweck) verwendet wird. Wandeln Sie das Beispiel geeignet ab.*

```
$ cat datei.txt
DiesIst
EinTest
$ while read file; do
    echo $file | xxd -c 8 | head
done < datei.txt
00000000: 4469 6573 4973 740a  DiesIst.
00000000: 4569 6e54 6573 740a  EinTest.
```

Speichern

Ende von Aufgabe 3

# SCREENSHOTS

## Abgabe

### Prüfsumme aller Antworten

Bitte notieren Sie nun auf Ihrem Prüfungsumschlag die folgende Prüfsumme im dafür vorgesehenen Feld (**Großbuchstaben und Ziffern**).

Prüfsumme für Klausurumschlag: **WUZ6A**

### Prüfung nicht werten (leer abgeben)

**Aktueller Status:** Ihre Antworten werden gewertet.

Geben Sie in das untenstehende Feld den Text „**nicht werten**“ ein und klicken Sie *Bestätigen*, wenn Sie leer abgeben möchten. Dies entspricht dem Durchstreichen aller Antworten. Ihre Abgaben werden folglich **nicht** gewertet. Sie können den Inhalt des Feldes bis zur Abgabe ändern, falls Sie Ihre Entscheidung überdenken.

# LESSONS LEARNED

Probleme mit **Debian, Touchpad und Tastatur**: daher Maus als erlaubtes Hilfsmittel. **Videoanleitung** und Möglichkeit Laptops **vorab zu testen**

**Rechenweg** ist mühsam einzutippen. Bearbeitung langsamer als auf Papier.

Viel **Scrollen** nötig – und Aufgabenstellungen nicht so übersichtlich wie auf A3-Bögen.

## Aufgabe 4: Assembly-Programmierung (21 Punkte)

Links ist das Assembly-Programm CIE angegeben, das auf dem Ihnen bekannten D-CORE (16-Bit Wortbreite, Big Endian) ausgeführt wird. Der rechts abgebildete Speicherinhalt zeigt die Situation vor der Ausführung des Programms. *Hinweis: 256<sub>10</sub> entspricht 100<sub>16</sub>.*

Am Ende dieser Aufgaben finden Sie die [Befehle](#) für den D-CORE-Prozessor.

```

1 movi r0, 1
2 lsli r0, 8
3 movi r1, 0
4 movi r2, 0
5 ldw r3, r0

6 loop:
7 cmpeq r3, r1
8 bt end

9 addi r0, 2
10 ldw r4, r0
11 subi r3, 1
12 addu r2, r4
13 br loop

14 end:
15 mov r4, r2
16 bseti r2, 0
17 subu r2, r4
18 halt
    
```

| OFFSET  | SPEICHERINHALT          |
|---------|-------------------------|
| [...]   |                         |
| 0x00f0: | 00 00 00 00 00 00 00 00 |
| 0x00f8: | 00 00 00 00 00 00 00 00 |
| 0x0100: | 00 02 00 0e 00 1f 00 00 |
| 0x0108: | 00 00 00 00 00 00 00 00 |
| 0x0110: | 00 00 00 00 00 00 00 00 |
| [...]   |                         |

a) Ein anderes Programm enthält die Instruktion 0x3550. Verändern Sie diese Instruktion, sodass anstelle der ursprünglichen Operation – bei sonst gleichen Parametern – nun eine Subtraktion durchgeführt wird.

 Speichern

Geben Sie die Zeilennummern-Paare eines Use-After-Load-Hazards in CIE an.

 Speichern

Zeigen Sie an einem Use-After-Load-Hazard, wie durch Umordnen der betroffenen Instruktionen der Hazard vermieden wird.

 Speichern

Gibt es in CIE Control-Hazards? Geben Sie *nein* an, falls es *keine* Control-Hazards gibt. Andernfalls geben Sie *ein* Zeilennummer-Paar für einen Control-Hazard in CIE an.

 Speichern

b) Welche Werte haben folgende Register, wenn CIE bis *einschließlich* Zeile 15 ausgeführt worden ist?

Wert für Register **r0**:  Speichern

Wert für Register **r1**:  Speichern

Wert für Register **r2**:  Speichern

8 Punkte  
 Weiß ich nicht.

7 Punkte  
 Weiß ich nicht.

Wert für Register **r1**:  Speichern

Wert für Register **r2**:  Speichern

Wert für Register **r3**:  Speichern

c) In Zeile 18 enthält r2 entweder den Wert 0 oder 1. Welche Eigenschaft von r4 wird damit kodiert?

Speichern

CIE soll nun auf einer Rechnerarchitektur ausgeführt werden, in der es die Instruktion bseti nicht gibt. Daher muss die Funktionalität der Zeilen 15–17 ohne bseti realisiert werden. Vervollständigen Sie dazu das folgende Gerüst von CIE mit geeigneten Instruktionen und den dazugehörigen Parametern.

[...] + r2 enthält das Zwischenergebnis end:

Geben Sie hier Ihren Code ein ...

Speichern

halt

6 Punkte  
 Weiß ich nicht.

### Assembly-Referenz: Befehle des D-CORE-Prozessors

Hier finden Sie alle Befehle des D-Core-Prozessors. R[x] steht für den Wert im Register x, C für den Wert des 1-Bit-Carry-Registers. MEM[R[y]] referenziert ein Wort an einer Speicheradresse, die in R[y] steht. PC ist der Program Counter.

| Mnemonic  | Kodierung               | Hex  | Bedeutung              |
|---|-------------------------|------|------------------------|
| <b>ALU-Operationen: Mnemonic x, y</b><br>Werte werden vorzeichenlos interpretiert.<br>Nur addc ändert C. bei Überlauf C=1, sonst C=0. |                         |      |                        |
| mov   | 0010 0000 <yyyy> <xxxx> | 20yx | R[x] = R[y]            |
| addu  | 0010 0001 <yyyy> <xxxx> | 21yx | R[x] = R[x] + R[y]     |
| addc  | 0010 0010 <yyyy> <xxxx> | 22yx | R[x] = R[x] + R[y] + C |
| subu  | 0010 0011 <yyyy> <xxxx> | 23yx | R[x] = R[x] - R[y]     |
| and   | 0010 0100 <yyyy> <xxxx> | 24yx | R[x] = R[x] AND R[y]   |
| or  | 0010 0101 <yyyy> <xxxx> | 25yx | R[x] = R[x] OR R[y]    |
| xor   | 0010 0110 <yyyy> <xxxx> | 26yx | R[x] = R[x] XOR R[y]   |

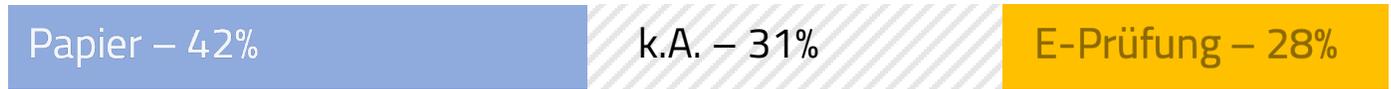
|  |                           |      |   |
|--|---------------------------|------|---|
|  | <xxxx>                    |      |   |
| or   | 0010 0101 <yyyy> <xxxx>   | 25yx | R[x] = R[x] OR R[y]                       |
| xor  | 0010 0110 <yyyy> <xxxx>   | 26yx | R[x] = R[x] XOR R[y]                      |
| not  | 0010 0111 <****> <xxxx>   | 27*x | R[x] = NOT R[x]                           |
| <b>Vergleichs-Operationen: Mnemonic x, y</b>   |                           |      |   |
| cmpeq  | 0011 0000 <yyyy> <xxxx>   | 30yx | C = (R[x] == R[y])                        |
| cmpne  | 0011 0001 <yyyy> <xxxx>   | 31yx | C = (R[x] != R[y])                        |
| cmpgt  | 0011 0010 <yyyy> <xxxx>   | 32yx | C = (R[x] > R[y])                         |
| cmplt  | 0011 0011 <yyyy> <xxxx>   | 33yx | C = (R[x] < R[y])                         |
| <b>Immediate-Operationen: Mnemonic x, c</b><br>Diese Instruktionen verändern C nicht.  |                           |      |   |
| movi   | 0011 0100 <cccc> <xxxx>   | 34cx | R[X] = 0x000<c>                           |
| addi   | 0011 0101 <cccc> <xxxx>   | 35cx | R[X] = R[X] + 0x000<c>                    |
| subi   | 0011 0110 <cccc> <xxxx>   | 36cx | R[X] = R[X] - 0x000<c>                    |
| andi   | 0011 0111 <cccc> <xxxx>   | 37cx | R[X] = R[X] AND 0x000<c>                  |
| lsli   | 0011 1000 <cccc> <xxxx>   | 38cx | R[X] = R[X] << 0x000<c>                   |
| lsri   | 0011 1001 <cccc> <xxxx>   | 39cx | R[X] = R[X] >> 0x000<c>                   |
| bseti  | 0011 1010 <cccc> <xxxx>   | 3Acx | R[X] = R[X] OR (1 << 0x000<c>)            |
| bclri  | 0011 1010 <cccc> <xxxx>   | 3Bcx | R[X] = R[X] AND NOT(1 << 0x000<c>)        |
| <b>Speicher-Operationen: Mnemonic x, y, c</b><br>Diese Instruktionen verändern C nicht.  |                           |      |   |
| ldw  | 0100 <cccc> <yyyy> <xxxx> | 4cyx | R[X] = MEM[R[Y] + (0x000<c> << 1)]        |
| stw  | 0101 <cccc> <yyyy> <xxxx> | 5cyx | MEM[R[Y] + (0x000<c> << 1)] = R[X]        |
| <b>Kontrollfluss: Mnemonic label/jimm12</b><br>PC bezieht sich auf PC vor dem Inkrementieren für den nächsten Befehl.<br>Diese Instruktionen verändern C nicht.<br>jimm12 wird als vorzeichenbehafteter 12-Bit-Immediate-Wert im 2-Komplement interpretiert. |                           |      |   |
| br   | 1000 <iiii> <iiii> <iiii> | 8iii | PC = PC+2+<jimm12>                        |
| call   | 1001 <iiii> <iiii> <iiii> | 9iii | R[15] = PC+2; PC = PC+2+<jimm12>          |
| bt   | 1010 <iiii> <iiii> <iiii> | Aiii | (C == 1) ? PC = PC+2+<jimm12> : PC = PC+2 |
| bf   | 1011 <iiii> <iiii> <iiii> | Biii | (C == 0) ? PC = PC+2+<jimm12> : PC = PC+2 |
| jmp  | 1100 <****> <****> <xxxx> | C**x | PC = R[X]                                 |



# BEFRAGUNG

$N = 120$  Studierende

Welche Variante bevorzugen Sie?



Linux-Shell benutzt?



Linux-Shell hilfreich: 55%

Schon einmal an einer Fernklausur teilgenommen?



Womit würden Sie eine Fernklausur schreiben wollen. Fertig eingerichteter ausgeliehener Laptop oder lieber privater Laptop, wenn auf Ihrem Gerät vor der Prüfung eine Prüfungssoftware installiert werden müsste?



geliehener – 20%

# WEITERES VORGEHEN

**Modulprüfung** im August 2023.

**Unterstützungssoftware für digitale Korrektur** (gleiche Antworten gleich bewerten, ähnliche Antworten schnell finden)

Lösung für **digitale Einsicht** und effiziente **Archivierung**

# FERN-E-PRÜFUNGEN

rechnergestützte Klausuren aufwandsarm  
und ausfallsicher durchführen

Prof. Dr.

**Dominik Herrmann**

@herdom · herdom.net

Otto-Friedrich-Universität Bamberg

