

Praktikum angewandte Systemsoftwaretechnik

Organisatorisches

Alexander Würstlein

Lehrstuhl Informatik 4

2017-10-19

Beteiligte Personen, Lehrstuhl Informatik 4

Organisatoren des Praktikums



Alexander Würstlein

Ziele in PASST

- Erlernen und Vertiefen von Kenntnissen der Softwareentwicklung
 - Im Umfeld des Linux-Kerns
 - Innerhalb des Linux-Kerns
- Vermittlung des notwendigen Werkzeugwissens
 - Kernel konfigurieren und bauen
 - Verwendung von virtuellen Maschinen als Testplattform
 - Versionsverwaltung mit **git**
 - Verwendung von GDB innerhalb des Linux-Kerns
- Hintergrundwissen
 - Open-Source-Entwicklungsprozesse kennenlernen
 - Finden und Beheben von Bugs

Aufbau der Veranstaltung

Vorlesungszeit

Tafelübung (2,5 ECTS)

- Vorstellung der Aufgaben
- Hilfestellung zur Lösung
- Präsentationen und Diskussionen

Aufbau der Veranstaltung

Vorlesungszeit

Tafelübung (2,5 ECTS)

- Vorstellung der Aufgaben
- Hilfestellung zur Lösung
- Präsentationen und Diskussionen

Programmierteil (2,5 ECTS)

- Bearbeitung der Aufgaben
- Vorbereitung auf Blockpraktikum
- Rechnerübungen mit konkreter Hilfestellung

Aufbau der Veranstaltung

Vorlesungszeit

Tafelübung (2,5 ECTS)

- Vorstellung der Aufgaben
- Hilfestellung zur Lösung
- Präsentationen und Diskussionen

Programmierteil (2,5 ECTS)

- Bearbeitung der Aufgaben
- Vorbereitung auf Blockpraktikum
- Rechnerübungen mit konkreter Hilfestellung

Semesterferien

Blockpraktikum (5 ECTS)

Gruppenweises Bearbeiten einer umfangreichen Aufgabe (Projektcharakter)

Möglicher Termin: 26. Februar bis 09. März 2018 (KW 09 & 10)
Festlegung *nächste Woche*

Programmierteil

Aufgaben

- Sechs *gemeinsame* Aufgaben
- Programmieren, Kurzvorträge und Praktikumsvorbereitung
- Bearbeitung in **Zwei**ergruppen
- Etwa zwei bis drei Wochen pro Aufgabe
- Abgabe durch Vortrag oder Vorführen in der Rechnerübung am Abgabetag

Programmierteil

Aufgaben

- Sechs *gemeinsame* Aufgaben
- Programmieren, Kurzvorträge und Praktikumsvorbereitung
- Bearbeitung in **Zwei**ergruppen
- Etwa zwei bis drei Wochen pro Aufgabe
- Abgabe durch Vortrag oder Vorführen in der Rechnerübung am Abgabetag

Rechnerübungen

- Termine:
 - Do 16 – 18 Uhr
 - Fr 14 – 16 Uhr
- Betreuer zu Beginn vor Ort
- Sonst auf Abruf in Büro (0.052)

Aufgaben in der Vorlesungszeit

- Aufgabe 1: Umgebung einrichten** Einrichten einer Debian-Installation in einer VM mit selbst kompiliertem Kern. Verwendung von GDB und KGDB.
- Aufgabe 2: Kernel-Vergleiche** Kernel, Werkzeuge und Entwicklungsprozesse verschiedener Linux-Distributionen vergleichen und dazu vortragen.
- Aufgabe 3: OOPS! Kernel-Bugs finden und reparieren** Einen komplexen Bug im Linux-Kern lokalisieren und reparieren.
- Aufgabe 4: Patches bauen und einsenden** Erzeugen eines Patches zur Behebung eines Kernel-Problems, "Einsenden" und akzeptiert bekommen und davon berichten.
- Aufgabe 5: Temperatursensor** Entwicklung eines Gerätetreibers für einen (selbstzulötenden!) USB-Sensor in den Kern.
- Aufgabe 6: Treiber für PCI Gerät** Einen komplizierteren Treiber für einen Logic Analyzer auf FPGA-Basis entwickeln.

Mögliche Themen für das Blockpraktikum

- 1 USB-over-IP (arw)
 - in den letzten Semestern erstellte Verbesserungen (IPv6, Crypto)
 - Erweiterung um z.B. komfortablere Userspace-Tools, ACLs und fein-granulare Authentifizierung, ...
 - Windows-Treiber (?!)
- 2 Erweiterungen von FAUmaschine (Volkmar)
 - virtuelle Maschine mit Fehlerinjektion und externer Steuer-/Skriptbarkeit
 - Erweiterung z.B. um
 - USB-IP-Integration oder USB-Geräte-Emulation

Mögliche Themen für das Blockpraktikum (Forts.)

- virtuelles Peripheriegerät (z.B. Balanciertisch), physikalisch simuliert und graphisch dargestellt, zum Entwickeln und Testen von Echtzeitsystemen
- ...

Mögliche Themen für das Blockpraktikum (Forts.)

- ③ Implementierung eines Schedulers für Jobcluster zum automatisierten Testen (Florian, Tobias)
 - Verteilung von Testfällen auf heterogene Hardware
 - Überwachung und dynamische Anpassung des Jobablaufs
 - Ressourcenplanung, -beschränkung und „einfrieren“ von Jobs

Mögliche Themen für das Blockpraktikum (Forts.)

- 4 automatisches modulares Testsystem für EZS-Übungsaufgaben (Florian, Tobias, Peter Ulbrich)
 - Hardware in the loop, generiert Eingaben und prüft Verhalten des zu testenden Systems
 - Entwicklung von Übungsaufgaben, Testfällen
 - Entwurf und Herstellung der Hardware, evtl. auch mit zu regelndem physikalischem System (Balanciertisch)
 - Teilaspekt davon, in Zusammenarbeit mit Mitarbeitern, weiteren Gruppenmitgliedern

Mögliche Themen für das Blockpraktikum (Forts.)

- 5 Gerrit um Abhängigkeitsgraph erweitern
 - bestehende Vorarbeit, inzwischen aktualisierungsbedürftig
 - aber von Upstream nicht angenommen, in Form bringen
 - neue Features

- 6 Logic Analyzer auf PCI Express (arw)
 - bestehende PCI-Logic-Analyzer-Karte auf PCI Express portieren
 - evtl. weitere Features implementieren

Mögliche Themen für das Blockpraktikum (Forts.)

- 7 USB-Serial in Userspace (Rainer)
 - Serielle USB-Geräte (FTDI, USB RS232) im Kernel (z.B. Linux: `/dev/ttyUSB*`)
 - auf Betriebssystemen wie Mac OS X schlecht gewartet
 - mit libusb aber auch als Benutzerprogramm umsetzbar
 - Aufgabe: pty-Unterstützung im Kernel mit mehr ioctls, Erweiterung von socat um USB-Serial
- 8 Entwicklung auf Intel Xeon Phi (Rainer)
 - PCIe-Erweiterungskarte mit vielen Kernen, Betriebssystem Linux
 - gedachter Anwendungsbereich: HPC
 - Implementierung von 512-bit SIMD für GLM (OpenGL Mathematics)
 - Eigene Ideen?

Mögliche Themen für das Blockpraktikum (Forts.)

- 9 fanotify() erweitern (arw)
 - Schnittstelle zur Beobachtung von Dateisystemereignissen fuer Virens Scanner
 - kann momentan (fast) nur Ereignisse bei open()
 - unsere Anwendungsfälle Backup und Quotacheck brauchen aber noch (un)link, chown, ...

Mögliche Themen für das Blockpraktikum (Forts.)

- 10 SDDM erweitern: Login-Manager für grosse Multiuser-Umgebungen (arw)
 - Nutzerfreundlichkeit verbessern: Tastenbelegung, Timeouts, Konfigurierbarkeit, Knöpfe
 - PAM-Einbindung: benutzerspezifische Meldungen beim Login
 - administrative globale Hinweise
- 11 Zwei-Faktor-Authentifizierung per U2F im Display-Manager (arw)
 - U2F: Challenge-Response ECDSA mit einem USB-Stick
 - Einbindung und Anpassung best. libs in Display-Manager (KDM, SDDM)
 - PAM-Modul (?)

Mögliche Themen für das Blockpraktikum (Forts.)

- 12 Eigene Hardware bauen
- 13 Entwicklung eines Gerätetreibers
 - Ihr kennt/habt Hardware, die nicht unter Linux funktioniert?
 - Entwickelt einfach euren eigenen Treiber
- 14 Eigene Ideen und Vorschläge

Benotung der Veranstaltung

- Punkte auf Aufgaben und Vorträge
 - 40 % Punkte Semesterteil
 - Punkte pro Aufgabe (15 bis 30 Punkte)
 - Gewichtungsfaktor: Aufwand, Schwierigkeit
 - 60 % Punkte Blockpraktikumsteil
 - Vorbereitung und Anfangsvortrag
 - Projektdurchführung
 - Abschlussvortrag
- Punkte in Note nach Notenschlüssel

Notenfindung

Teilnote		A1	A2	A3	A4	A5	A6	Blockpraktikum
Gewichtung		1	1	2	2	2	2	15

Bewertungskriterien

- Aufgabenerfüllung
- Codequalität: Funktion, Lesbarkeit, Kommentare, Coding Style
- interne Interaktion: Zusammenarbeit und Aufgabenteilung innerhalb der (Zweier-)Gruppe
- externe Interaktion: Kommunikation und Zusammenarbeit mit Betreuern und Externen (Upstream)
- Vortrag und Vorstellung: Gliederung, Verständlichkeit, Inhalt, Stil, Behandlung von Fragen
- **sofern anwendbar** und **unterschiedlich gewichtet** je nach Aufgabe

Kommunikation

- Mailingliste Übungsleiter: i4passt@lists.cs.fau.de
- Mailingliste Teilnehmer: i4passt_all@lists.cs.fau.de
- IRC-Channel im IRCnet: #i4passt
- Rechnerübung: arw's Telefonnummer im Buero
- Webseite: https://www4.cs.fau.de/Lehre/WS17/P_PASST/