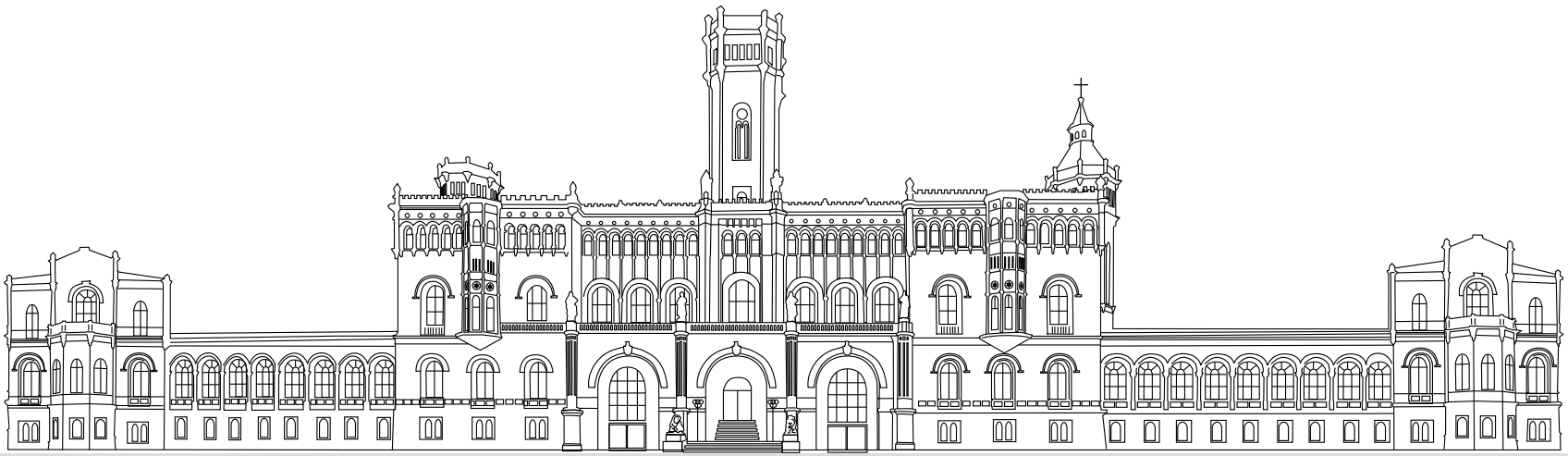
Labor: Linux Systemadministration

Dr. Hans Georg Krojanski

krojanski@chi.uni-hannover.de



$ whoami

- Experimentalphysiker
- PhD Quantum Computing
- Medizinphysiker (bildgebende Verfahren)

$ whoami

- Experimentalphysiker
- PhD Quantum Computing
- Medizinphysiker (bildgebende Verfahren)
- LUIS (RRZN)
  - Begonnen im Suchmaschinenlabor (metaGer, ...)
  - Leitung des IT Security Teams
  - Abteilungsleitung Compute and Storage Systems (CSS)
- FEI: Computational Health Informatics (CHI)
  - Prof. Dr.-Ing. Gabriele von Voigt

$ whoami

- Experimentalphysiker
- PhD Quantum Computing
- Medizinphysiker (bildgebende Verfahren)
- LUIS (RRZN)
  - Begonnen im Suchmaschinenlabor (metaGer, ...)
  - Leitung des IT Security Teams
  - Abteilungsleitung Compute and Storage Systems (CSS)
- FEI: Computational Health Informatics (CHI)
  - Prof. Dr.-Ing. Gabriele von Voigt
- Unix/Linux Systemadministrator

LV IT-Infrastrukturen in der Medizin

- Rechenzentren-Infrastruktur
- Physische IT-Sicherheit
- Informationssicherheit und Datenschutz
- Medizinische Einrichtungen und Gesundheitswesen
- Smart Hospitals und IoT
- IT-Sicherheit medizinischer Geräte
- Speichersysteme und ihre Anwendungen
- Datensicherung & Langzeitarchivierung
- Ausfallsicherheit und Business Continuity
- Virtualisierung, {Applikations-,System-}Container

Labor: Containervirtualisierung

- Grundlagen Linux-Systemadministration
- Namespaces & cgroups
- Deployment von Webanwendungen
- Applikationscontainer (Docker)
- Systemcontainer (LXC/LXD bzw. Incus, systemd-nspawn)
- Anwendungen in der Medizin

Lernziele (Modkat)

Modkat 1/2

Das Labor vermittelt Kenntnisse und Fähigkeiten im Bereich der Administration und Absicherung von Linux-Servern. Am Beispiel von Open Source Software aus dem Bereich der Medizinischen Informatik werden Linux-Server in einer virtuellen Umgebung aufgesetzt und konfiguriert.

Lernziele (Modkat)

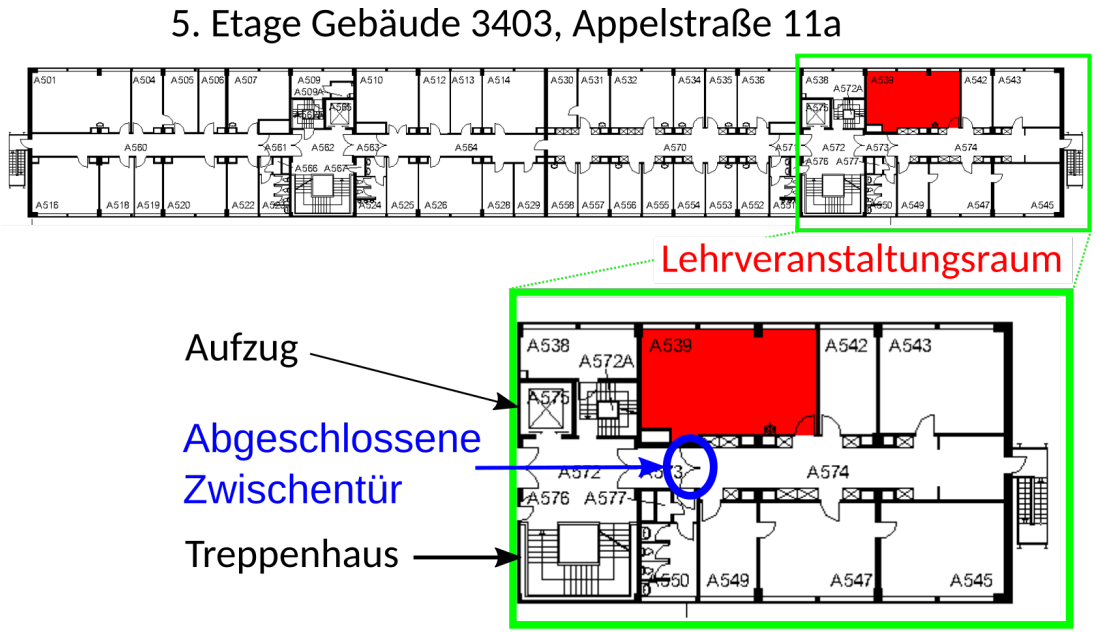
Modkat 2/2

Die Studierenden entwickeln selbstständig Lösungen für die Aufteilung einer Anwendung in mehrere Komponenten auf unterschiedlichen Servern sowie deren Verwaltung und Dokumentation. Nach erfolgreichem Abschluss des Moduls können die Studierenden Konzepte für mehrkomponentige Systeme unter Berücksichtigung gängiger Sicherheitsstandards erstellen und implementieren.

Stoffplan/Modulinhalte

- Grundlagen Linux & Shell
- Server Sicherheitsmaßnahmen (nftables, sshd, ...)
- Webserver, HTTPS & Zertifikate (eigene CA)
- Deployment einer Anwendung in 3 VMs: Datenbank-, Applikationsserver & Reverse Proxy
- Backup & Restore (Dateien und Datenbank)
- Konfigurationsmanagement mit Ansible
- Ergänzende Themen (Monitoring, Disaster Recovery, ...)

- **Vorlesungszeit: 7.4.2025 bis 19.7.2025**
  - 7.4.2025 frei (LUH Einführungsveranstaltungen)
  - 21.4.2025 Ostermontag
  - 9.6.2025 Pfingsmontag
- **Arbeiten außerhalb der 12 Präsenztermine möglich**
- **Laborraum A539, Gebäude 3403**



Vorkenntnisse

- Grundlagen der Betriebssysteme
- Linux Grundkenntnisse (empfohlen)

Vorkenntnisse, didaktisches Konzept

- Grundlagen der Betriebssysteme
- Linux Grundkenntnisse (empfohlen)

- I.d.R. nur grobe Einführung bzw. Vorstellung der Grundlagen
- Teilweise offen gestellte Aufgaben
  - Keine detaillierten Einzelschritte vorgegeben
  - Selbstständige Lösungsfindung
  - Diskussion und Austausch untereinander

Prüfungsleistung

- Erfolgreiche Teilname durch regelmäßiges Erbringen von Einzelleistungen (d. h. jede Person macht alle Übungen selbst), Dokumentation des Vorgehens (Markdown)

Prüfungsleistung

- Erfolgreiche Teilname durch regelmäßiges Erbringen von Einzelleistungen (d. h. jede Person macht alle Übungen selbst), Dokumentation des Vorgehens (Markdown)
- Leistungen werden definiert in Form von zu erreichenden Zielen (Fragen, Aufgabenstellungen)

Prüfungsleistung

- Erfolgreiche Teilname durch regelmäßiges Erbringen von Einzelleistungen (d. h. jede Person macht alle Übungen selbst), Dokumentation des Vorgehens (Markdown)
- Leistungen werden definiert in Form von zu erreichenden Zielen (Fragen, Aufgabenstellungen)
- Abschlussbericht schreiben und abgeben bis 19.7.2025 (23:59 Uhr)

Übungsaufgaben/-fragen

- **Lest euch die Aufgaben erst einmal in Ruhe durch!**

Übungsaufgaben/-fragen

- **Lest euch die Aufgaben erst einmal in Ruhe durch!**
- Übungsaufgaben und -fragen werden im Folgenden mit dem "LUH-Blau" geschrieben
- Optionale und/oder schwierigere Aufgaben für Fortgeschrittene sind in Hellblau geschrieben

Übungsaufgaben/-fragen

- **Lest euch die Aufgaben erst einmal in Ruhe durch!**
- Übungsaufgaben und -fragen werden im Folgenden mit dem "LUH-Blau" geschrieben
- Optionale und/oder schwierigere Aufgaben für Fortgeschrittene sind in Hellblau geschrieben
- Die Server für die Übungen werden 24/7 zur Verfügung stehen (bis auf Wartungsfenster), es kann also auch zwischen den Präsenztreffen gearbeitet werden

Laufende Dokumentation

- **Während der Aufgabenbearbeitung Notizen machen und Fragen beantworten in Markdown-Vorlage** (copy & paste, Stichworte, sinnvolle Teile von Ausgaben auf der Kommandozeile, ...)

Laufende Dokumentation

- Während der Aufgabenbearbeitung Notizen machen und Fragen beantworten in Markdown-Vorlage (copy & paste, Stichworte, sinnvolle Teile von Ausgaben auf der Kommandozeile, ...)
- Jede Woche nach der Präsenzveranstaltung per E-Mail an den Dozenten schicken
- Gerne auch schon Feedback, Rückfragen, etc.

Abschlussbericht

- Format selbst wählbar (übliche Fontgrößen, Ränder, … verwenden)
- Abgabe als PDF, nicht mehr als 10 Seiten
- Prägnant, eindeutig und präzise formulieren

Abschlussbericht

- Format selbst wählbar (übliche Fontgrößen, Ränder, … verwenden)
- Abgabe als PDF, nicht mehr als 10 Seiten
- Prägnant, eindeutig und präzise formulieren
- Grafik und kurzer Begleittext zur Systemübersicht
  - Welche Dienste laufen wo? Welche Ports sind erreichbar?
  - Datenflüsse inkl. der verwendeten Protokolle/Formate einzeichnen (extern & zwischen den Teilsystemen)
  - Datenspeicherung (nicht nur Datenbank)
  - …

Abschlussbericht

- Welche Pakete wurden wie (apt, pip, git, ...) und warum installiert?
- Angepasste und relevante Konfigurationsdateien dokumentieren, ggf. erklären
- Implementierte Sicherheitsmaßnahmen beschreiben

Abschlussbericht

- Welche Pakete wurden wie (apt, pip, git, ...) und warum installiert?
- Angepasste und relevante Konfigurationsdateien dokumentieren, ggf. erklären
- Implementierte Sicherheitsmaßnahmen beschreiben
- Wie startet man die Anwendung bzw. Teildienste neu? (Reihenfolge, Voraussetzungen, ...)
- Reaktion auf Ereignisse: Mögliche auftretende Fehler, Restore von Backups, Disaster Recovery und Neuinstallation, ...

Abschlussbericht

- Welche Pakete wurden wie (apt, pip, git, ...) und warum installiert?
- Angepasste und relevante Konfigurationsdateien dokumentieren, ggf. erklären
- Implementierte Sicherheitsmaßnahmen beschreiben
- Wie startet man die Anwendung bzw. Teildienste neu? (Reihenfolge, Voraussetzungen, ...)
- Reaktion auf Ereignisse: Mögliche auftretende Fehler, Restore von Backups, Disaster Recovery und Neuinstallation, ...
- Tipp: Dokumentiert für eure Urlaubsvertretung

Arbeitsmittel

- Stud.IP: Hochladen der Lehrmittel
- Ggf. BigBlueButton: Gruppenbesprechung
- Matrix Messenger (Chatraum)

Arbeitsmittel

- Stud.IP: Hochladen der Lehrmittel
- Ggf. BigBlueButton: Gruppenbesprechung
- Matrix Messenger (Chatraum)
- 4 eigene virtuelle Maschinen (VMs):
  - Server für Webanwendung: `ll-db-X.incus`, `ll-app-X.incus`, `ll-web-X.incus`
  - Administrationsserver: `ll-admin-X.incus`
- 1 vCPU, 2 GiB RAM, 25 GiB Partition
- Debian 12 (Minimalinstallation)

Arbeitsmittel

- Stud.IP: Hochladen der Lehrmittel
- Ggf. BigBlueButton: Gruppenbesprechung
- Matrix Messenger (Chatraum)
- 4 eigene virtuelle Maschinen (VMs):
  - Server für Webanwendung: `ll-db-X.incus`, `ll-app-X.incus`, `ll-web-X.incus`
  - Administrationsserver: `ll-admin-X.incus`
- 1 vCPU, 2 GiB RAM, 25 GiB Partition
- Debian 12 (Minimalinstallation)
- Netz ausgehend nur über LUH-Proxy! (LUIS-Konfigurationsbeispiele, nicht vollständig)
- Tägliche Snapshots (≈ 3 Uhr)

Einige Verhaltensregeln

- Volle root-Rechte → "With great power ..."
- Die zur Verfügung gestellten Ressourcen (insb. die virtuellen Maschinen) dürfen ausschließlich für das Labor verwendet werden

Einige Verhaltensregeln

- Volle root-Rechte → "With great power ..."
- Die zur Verfügung gestellten Ressourcen (insb. die virtuellen Maschinen) dürfen ausschließlich für das Labor verwendet werden
- Die übergebenen Passwörter und Geheimnisse dürfen nicht weitergegeben werden und müssen angemessen geschützt werden (keine Übermittlung im Klartext; Speicherung im Passwortmanager o.ä. empfohlen)

Einige Verhaltensregeln

- Volle root-Rechte → "With great power ..."
- Die zur Verfügung gestellten Ressourcen (insb. die virtuellen Maschinen) dürfen ausschließlich für das Labor verwendet werden
- Die übergebenen Passwörter und Geheimnisse dürfen nicht weitergegeben werden und müssen angemessen geschützt werden (keine Übermittlung im Klartext; Speicherung im Passwortmanager o.ä. empfohlen)
- Wer sich nicht an diese Regeln hält, muss das Labor mit "nicht bestanden" verlassen

Technische Voraussetzungen

- Terminal/SSH-Client; Empfehlung: Linux (z.B. als VM auf eigenem Laptop) nutzen
- `ssh(1)`: OpenSSH remote login client
- ...program for logging into a remote machine
- ...program for executing commands on a remote machine
- ...provides secure encrypted communicaton over an insecure network

Technische Voraussetzungen

- Anwendung für die Generierung von TOTP-Token
  - Time-based One-time Password Algorithm für 2-Faktor-Authentisierung
  - Mobile Apps (Android, iOS) oder Desktop-Anwendungen
- Per E-Mail wird u.a. ein TOTP secret übergeben

Technische Voraussetzungen

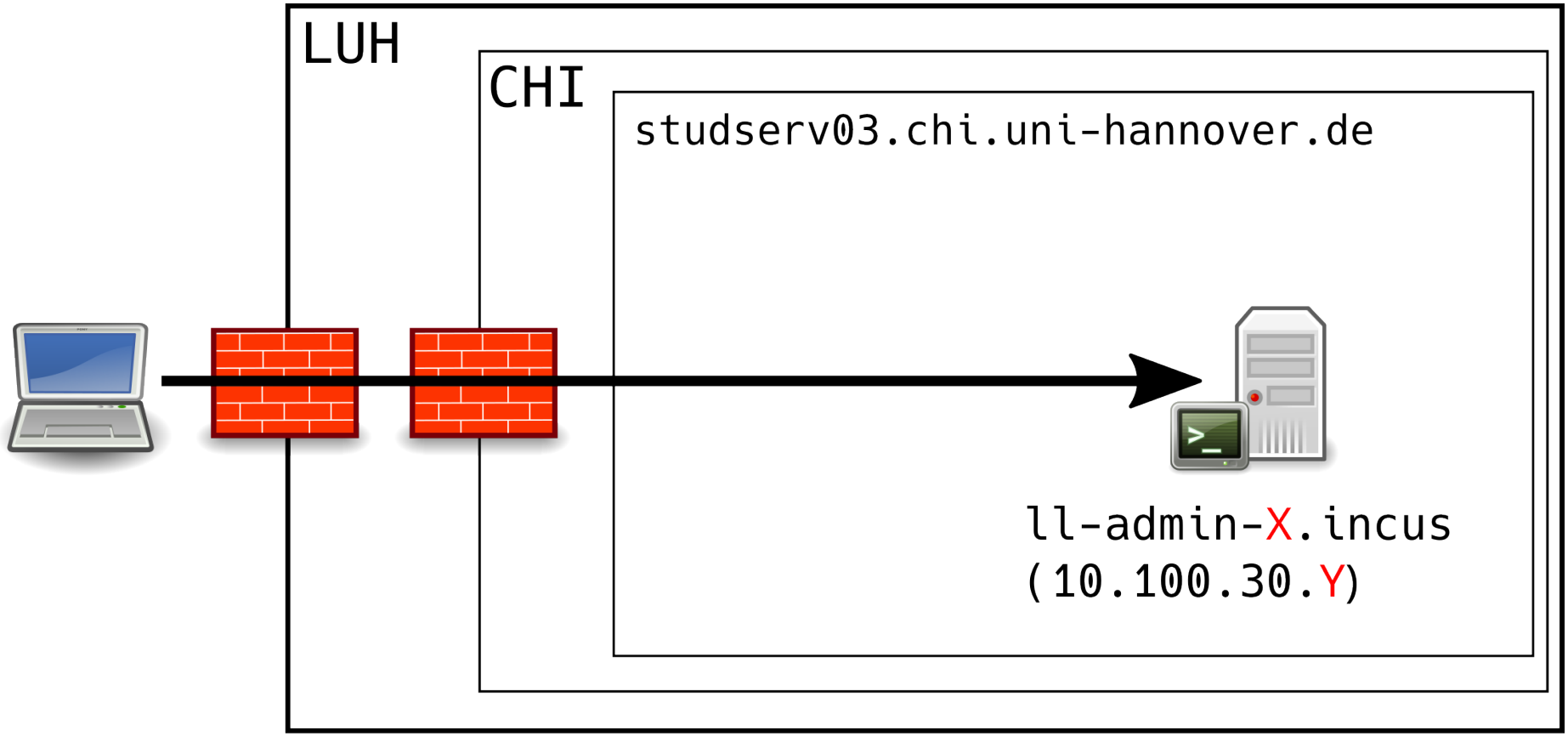
- Anwendung für die Generierung von TOTP-Token
  - Time-based One-time Password Algorithm für 2-Faktor-Authentisierung
  - Mobile Apps (Android, iOS) oder Desktop-Anwendungen
- Per E-Mail wird u.a. ein TOTP secret übergeben
- Ein QR-Code kann bspw. in der Shell angezeigt werden durch:

```
qrencode -t ansiutf8 -o - 'otpauth://totp/
linuxlabX@ll-jumphost?secret=...'
```

- Anzeigen eines TOTP-Codes in der Shell:

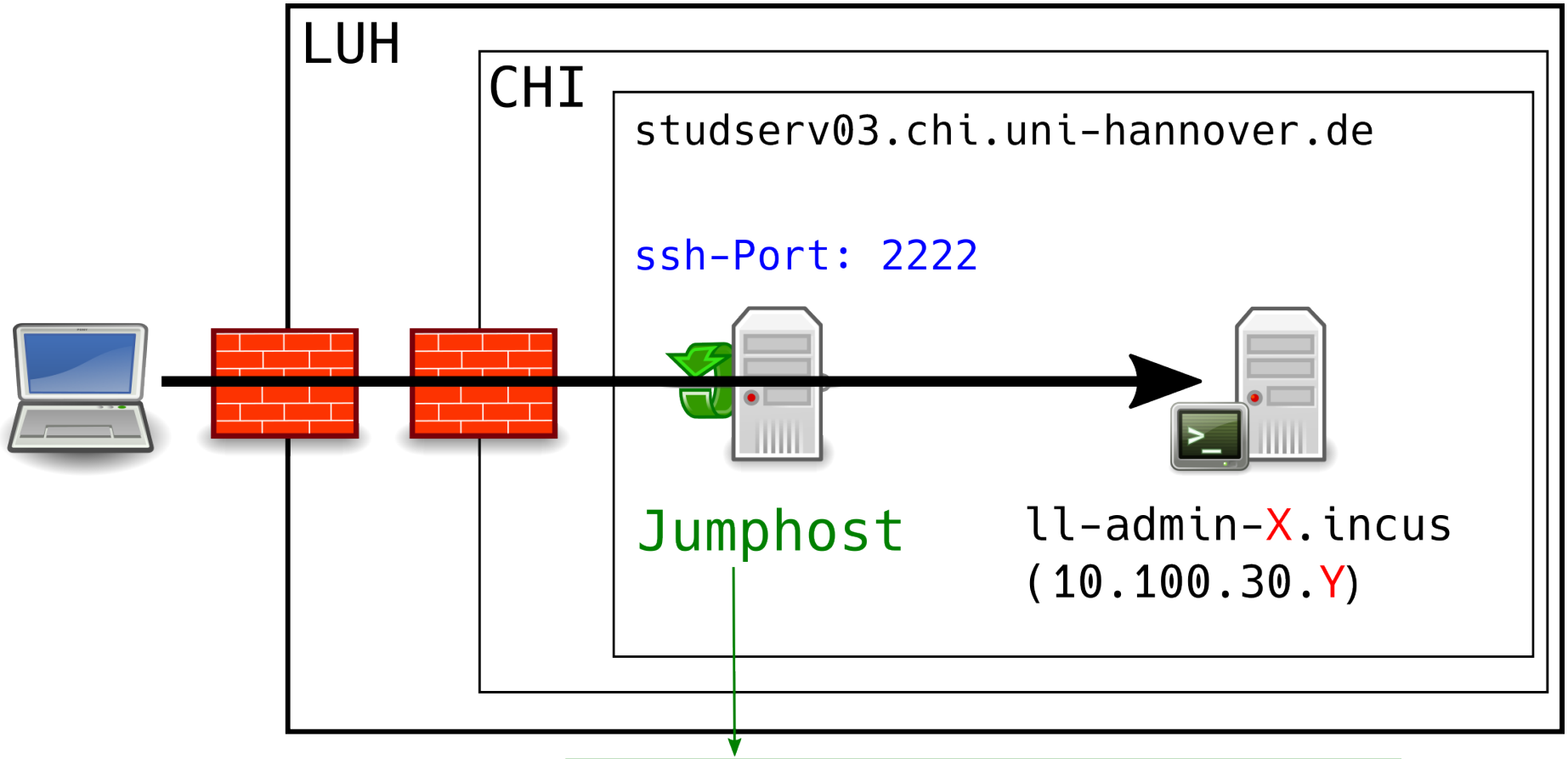
```
oathtool -b --totp <secret>
```

Login bei direkter Erreichbarkeit




```
ssh debian@ll-admin-X.incus
```

Login via ssh-Jumphost




```
ssh debian@ll-admin-X.incus -J linuxlabX@studserv03.chi.uni-hannover.de:2222
```

Arbeiten mit den VMs

- VMs sind via SSH-Jumphost weltweit erreichbar
- `ll-db-X` für Datenbank
- `ll-app-X` für Applikation (Python, uwsgi)
- `ll-web-X` für Webserver (Reverse Proxy)
- `ll-admin-X` als Admin-Server
- Login per SSH (via Jumphost) nur in `ll-admin-X`
  - Session absichern durch Nutzung von `tmux(1)`
  - Von hier Login zu den anderen VMs
  - Später: Ansible Controller
  - Ort der eigenen CA (privater CA-Key!)
  - Ort der Backups (Dateien, Datenbank)

Arbeiten mit den VMs

- VMs sind via SSH-Jumphost weltweit erreichbar
- `ll-db-X` für Datenbank
- `ll-app-X` für Applikation (Python, uwsgi)
- `ll-web-X` für Webserver (Reverse Proxy)
- `ll-admin-X` als Admin-Server
- Login per SSH (via Jumphost) nur in `ll-admin-X`
  - Session absichern durch Nutzung von `tmux(1)`
  - Von hier Login zu den anderen VMs
  - Später: Ansible Controller
  - Ort der eigenen CA (privater CA-Key!)
  - Ort der Backups (Dateien, Datenbank)
- Nur benötigte Verbindungen zulassen!

Test einer Webanwendung

Eine Webanwendung kann lokal auf dem Server (`ll-web-X`) getestet werden, von einem anderen Server im gleichen Netzwerk (z. B. `ll-admin-X`) aus oder vom eigenen Rechner. In den ersten beiden Fällen bedient man sich unterschiedlicher Kommandozeilen–Tools, im letzten Fall kann ein Webbrowser genutzt werden. Dieser Webbrowser kann aber die Webanwendung nicht direkt erreichen, sondern muss einen vorher aufgebauten SSH–Tunnel zu einem Server verwenden. Auf der nächsten Seite stehen Fragen für später, wenn ein Test vom eigenen Rechner aus benötigt wird.

Dynamic port forwarding, SOCKS v5

- Establish a tunnel to your admin server and use the `ssh(1)` option which activates dynamic port forwarding, so that `ssh(1)` will act as a SOCKS v5 server

Dynamic port forwarding, SOCKS v5

- Establish a tunnel to your admin server and use the `ssh(1)` option which activates dynamic port forwarding, so that `ssh(1)` will act as a SOCKS v5 server
- Configure your web browser to use the SOCKS v5 server/port on your local machine
- Be careful to use this web browser only for testing purposes in this lab! Make sure that no other tabs are open. Remember that the outgoing traffic flows through the LUH web proxy and this traffic is monitored by LUIS...

Dynamic port forwarding, SOCKS v5

- Ensure in your browser configuration that the name resolution for internal `.incus` domains works
- Which URL do you have to use to connect to your web application?
- Use a fully qualified domain name in the URL, so that the certificate of the web server can be matched and checked
- Did your browser accept the certificate of the web server? If not, what prerequisite is needed for successful certificate checking on your machine?

Übergabe der Login-Daten

- Bitte schickt mir einen ssh-pubkey per E-Mail zu. Erzeugen kann man diesen mit `ssh-keygen(1)`. Mehr Infos dazu bspw. im Debian Wiki oder Archlinux Wiki.
- Ich antworte darauf und hänge eine verschlüsselte Datei mit allen Zugangsdaten an
- Entschlüsselung erfolgt mit eurem privaten ssh-key und der Software age. Infos zur Installation und Entschlüsselung: github-Seite von age.

Erste Schritte

- Richtet euren TOTP-Generator ein und testet den Login in euren Admin-Server

Erste Schritte

- Richtet euren TOTP-Generator ein und testet den Login in euren Admin-Server
- Aktualisiert alle Pakete mit

  ```
  sudo apt update && sudo apt upgrade
  ```

  und startet den Server ggf. neu: `reboot`
- Installiert wichtige Pakete

  ```
  sudo apt install less nano tmux
  ```

  Empfohlen wird zusätzlich

  ```
  sudo apt install bash-completion
  ```

Erste Schritte

- Richtet euren TOTP-Generator ein und testet den Login in euren Admin-Server
- Aktualisiert alle Pakete mit
  `sudo apt update && sudo apt upgrade`
  und startet den Server ggf. neu: `reboot`
- Installiert wichtige Pakete
  `sudo apt install less nano tmux`
  Empfohlen wird zusätzlich
  `sudo apt install bash-completion`
- Startet `tmux(1)`, macht euch mit dem Server vertraut und beginnt dort mit den Linux-Übungen

- **Guckt euch den Foliensatz `02_linux.pdf` des Containerlabors an. Die Kenntnis der dort besprochenen Programme und Techniken werden im Folgenden vorausgesetzt.**
  - **Studierende, die ausreichend Erfahrung mit Linux und der Shell haben, können die Beantwortung der Fragen überspringen, Unerfahrene sollten diesen Foliensatz (weitestgehend) durcharbeiten.**
  - **Am 28.4.2025 kann ich gerne auf Fragen eingehen und ggf. Lösungen von Aufgaben besprechen**

- **Guckt euch den Foliensatz `02_linux.pdf` des Containerlabors an. Die Kenntnis der dort besprochenen Programme und Techniken werden im Folgenden vorausgesetzt.**
  - **Studierende, die ausreichend Erfahrung mit Linux und der Shell haben, können die Beantwortung der Fragen überspringen, Unerfahrene sollten diesen Foliensatz (weitestgehend) durcharbeiten.**
  - **Am 28.4.2025 kann ich gerne auf Fragen eingehen und ggf. Lösungen von Aufgaben besprechen**
- **Nach Ostern werde ich die ersten Übungen dieses Labors auf Stud.IP veröffentlichen. Dann beginnt auch die oben erwähnte Dokumentation der Antworten.**