

# Praktikum Rechnernetze

Protokoll zu Versuch 5 (Paketfilter-Firewall unter Linux) von  
Gruppe 1

---

Jakob Waibel   Daniel Hiller   Elia Wüstner   Felix Pojtinger

2021-11-16

# Einführung

---

Diese Materialien basieren auf Professor Kiefers “Praktikum Rechnernetze”-Vorlesung der HdM Stuttgart.

**Sie haben einen Fehler gefunden oder haben einen Verbesserungsvorschlag?** Bitte eröffnen Sie ein Issue auf GitHub ([github.com/poijntfx/uni-netpractice-notes](https://github.com/poijntfx/uni-netpractice-notes)):



**Figure 1:** QR-Code zum Quelltext auf GitHub

Dieses Dokument und der enthaltene Quelltext ist freie Kultur bzw. freie Software.



**Figure 2:** Badge der AGPL-3.0-Lizenz

Uni Network Practice Notes (c) 2021 Jakob Waibel, Daniel Hiller,  
Elia Wüstner, Felix Pojtinger

SPDX-License-Identifier: AGPL-3.0

# Wordpress Konfigurieren

---

# Wordpress Konfigurieren

**Auf ihrem Server ist Wordpress vorinstalliert /  
vorkonfiguriert. Lediglich die abschließende Einrichtung ist  
noch nicht erfolgt... Führen Sie die Einrichtung durch und  
stellen Sie die Funktion sicher. Rufen Sie dazu die IP der  
Servers in einem Web-Browser auf.**

Zur Fertigstellung der Konfiguration muss zuerst folgender Dialog  
ausgefüllt werden:



The image shows the WordPress installation 'Welcome' screen. At the top is the WordPress logo. Below it, the heading 'Welcome' is followed by a paragraph: 'Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.' The next section is 'Information needed', with a subtext: 'Please provide the following information. Don't worry, you can always change these settings later.' There are four input fields: 'Site Title' with 'Ubunt', 'Username' with 'Ubongo', 'Password' with '6w^7agD\*(SxXnNpIoM' (marked as 'Strong'), and 'Your Email' with 'jakwa01@gmail.com'. A 'Hide' button is next to the password field. Below the password field, an 'Important!' note states: 'You will need this password to log in. Please store it in a secure location.' At the bottom, a small instruction says: 'Double-check your email address before continuing.'

WordPress logo

## Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

### Information needed

Please provide the following information. Don't worry, you can always change these settings later.

**Site Title**

**Username**

Username can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

**Password**  [Hide](#)

**Strong**

**Important!** You will need this password to log in. Please store it in a secure location.

**Your Email**

Double-check your email address before continuing.

# Portscan durchführen

---

# Portscan durchführen

Überprüfen Sie mit einem Portscanner welche Ports an Ihrem Server öffentlich erreichbar sind. Welche Ports/Services sind das? Müssen diese Services öffentlich erreichbar sein?

Zur Sicherheit starten wir bevor wir mit dem Portscanning beginnen den VPN unseres Vertrauens.





# Blockieren von Services

---

# Blockieren von Services

**Sie haben in Aufgabe 2 mindestens einen Service identifiziert, der nicht öffentlich verfügbar sein muss. Blockieren Sie den externen Zugriff auf diesen Service in Ihrer Firewall (Blacklist-Ansatz).**

Der “Blacklist-Ansatz” bedeutet, dass mit einer ACCEPT policy und negativen Regeln gearbeitet wird, sodass alles erlaubt ist, sofern es nicht durch eine Regel explizit verboten wird.

Blocken aller Ports neben 22 und 80:

```
$ sudo iptables -F INPUT
$ sudo iptables -A INPUT -p tcp --dport 25 -j DROP
$ sudo iptables -A INPUT -p tcp --dport 53 -j DROP
$ sudo iptables -A INPUT -p tcp --dport 139 -j DROP
$ sudo iptables -A INPUT -p tcp --dport 445 -j DROP
$ sudo iptables -A INPUT -p tcp --dport 1900 -j DRO
```

## **Whitelist-Ansatz per Shell-Skript**

---

# Whitelist-Ansatz per Shell-Skript

**Stellen Sie den gleichen Zustand der Firewall (Damit meine ich, dass die gleichen Services erreichbar sind) her wie in Aufgabe 3, allerdings verfolgen Sie jetzt den Whitelist-Ansatz.**

Der “Whitelist-Ansatz” bedeutet, dass die default policy DROP verwendet wird und dass alles verboten ist, was nicht explizit durch eine Regel erlaubt wurde.

Inhalt von `iptables —rules.sh`:

```
# $HOME/iptables—rules.sh
```

```
#!/usr/bin/env bash
```

```
sudo iptables —F INPUT
```

# ICMP und Prometheus Node-Exporter

---

# ICMP und Prometheus Node-Exporter

Der Prometheus Node-Exporter liefert Metriken für Prometheus (<https://prometheus.io/>). Konfigurieren Sie ihre Firewall so, dass diese Metriken nur von Ihren IP-Adressen aus erreichbar sind (Nutzen Sie <https://ifconfig.co/> um Ihre öffentliche IP-Adresse in Erfahrung zu bringen). Das selbe gilt für ICMP. Die Angriffsvektoren für ICMP sind zwar ziemlich eingeschränkt, trotzdem reicht es, wenn Sie in der Lage sind Probes an den Server zu senden.

Unsere IP-Adresse:

```
$ curl https://ifconfig.io  
193.27.14.134
```

Nun müssen zwei ACCEPT-Rules erstellt werden; zuerst für den Prometheus Node-Exporter:

# **Besprechung, Musterlösung und Einbindung als System-Service**

---

# Besprechung, Musterlösung und Einbindung als System-Service

**Diese Aufgabe führen wir zusammen durch, dokumentieren Sie trotzdem die Schritte und Ergebnisse!**

```
$ systemctl cat iptables
# /etc/systemd/system/iptables.service
[Unit]
Description=firewall service
Before=network.target
AssertPathExists=/root/iptables-rules.sh

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/root/iptables-rules.sh
StandardOutput=syslog
```