

# Wie man zu[**mr**] **System Ingenieur(in)?** **wird**

Grosse Teile der globalen Web/Cloud/IoT Infrastruktur basieren auf Free/OpenSource Software.

Dies macht es möglich -und notwendig-, sich die für die Bedienung dieser Infrastruktur erforderlichen Kenntnisse selber durch Basteln am Computer anzueignen.

2020-12-26 Folkert



# Inhalt

1. Warum man zum System Ingenieur wird
2. Wie man zum System Ingenieur wird

# System Ingenieur(in)?

## Begriffsklärung

Der Systemingenieur, auch Systems Engineer, ist für die Analyse der Kundenanforderungen, der Architektur und des Designs von komplexen integrierten Systemen verantwortlich.

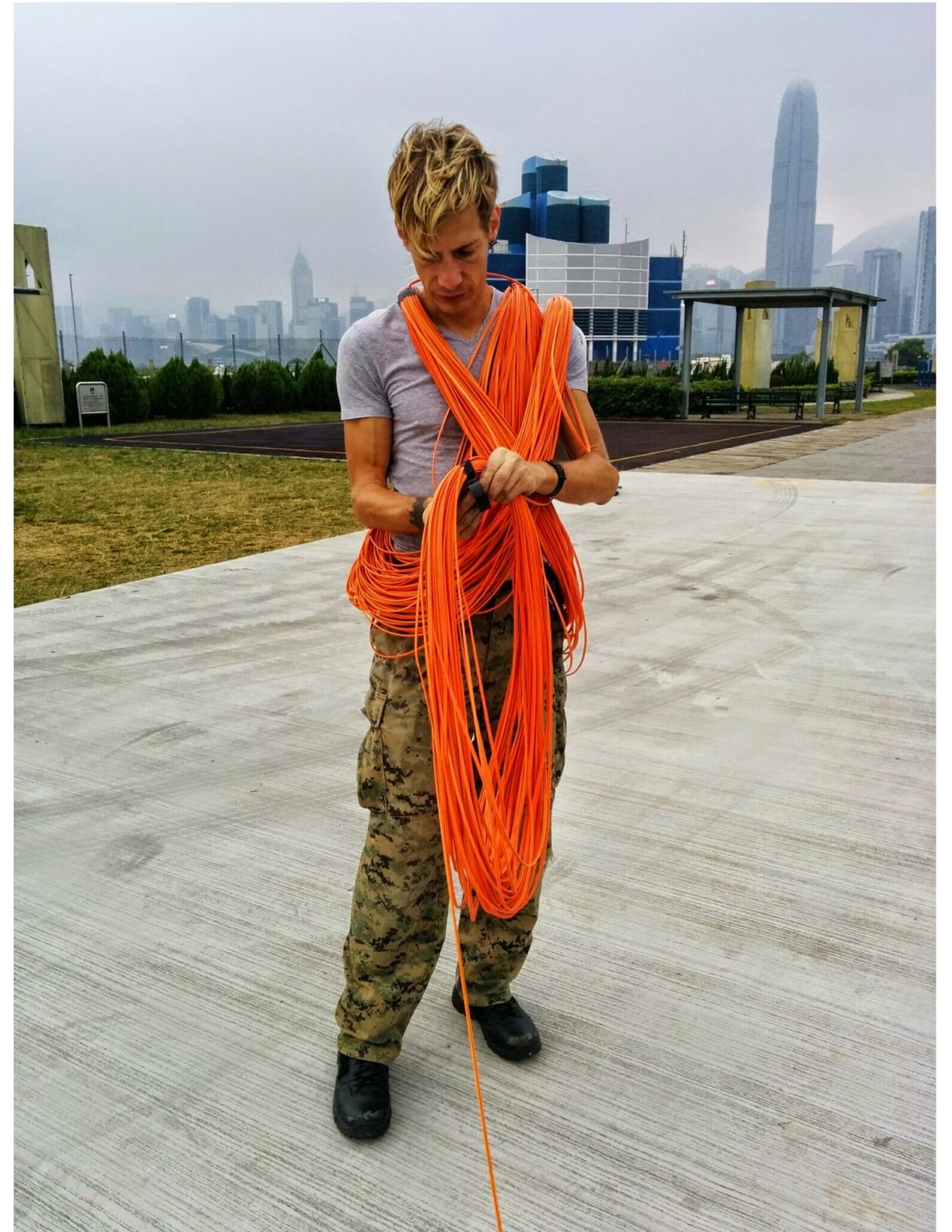
<https://de.wikipedia.org/wiki/Systemingenieur>

- Ein System Ingenieur baut und betreibt Infrastruktur
  - Netzwerke (TCP/IP, WiFi, VPN, ... )
  - Server (Web, Datenbank, Mail, Chat, ... )
- aka *System Administrator / System Operator / System Architect*

# Folkert

Systems Engineer, Hong Kong / Berlin

- Web Portal Infra
- Payment Services Infra
- Festival Network Infra
- Genomics Research Infra



# WARUM man System Ingenieur wird

- Jobsicherheit
- Autonomie
- Datenschutz
- Ethik



# Warum man System Ingenieur wird

Jobsicherheit :D

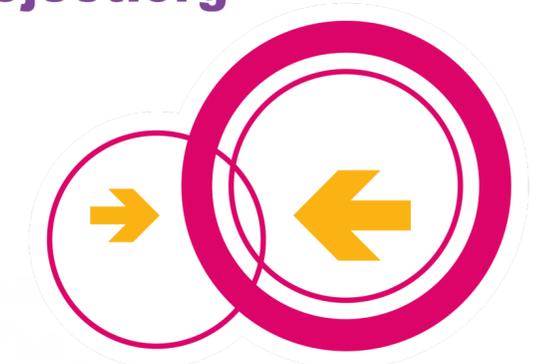
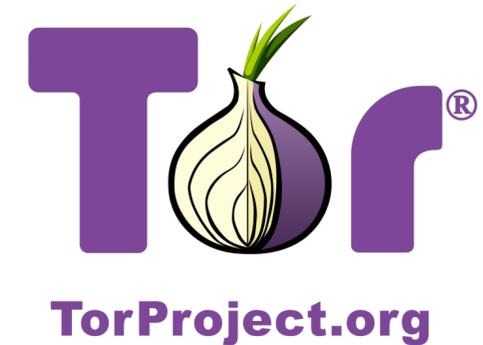
- Solange es Computer gibt, werden System Ingenieure gebraucht
- Überall wo es Computer gibt, werden System Ingenieure gebraucht



# Warum man System Ingenieur wird

## Autonomie

- Es ist möglich, durch Netzwerke und Dienste Veränderung in der Welt zu schaffen
- Netzwerke und Dienste erfordern Infrastruktur
- Infrastruktur erzeugt Abhängigkeiten:
  - Google / Amazon / Cloudflare / PayPal
- Eigene Infrastruktur schafft Unabhängigkeit!



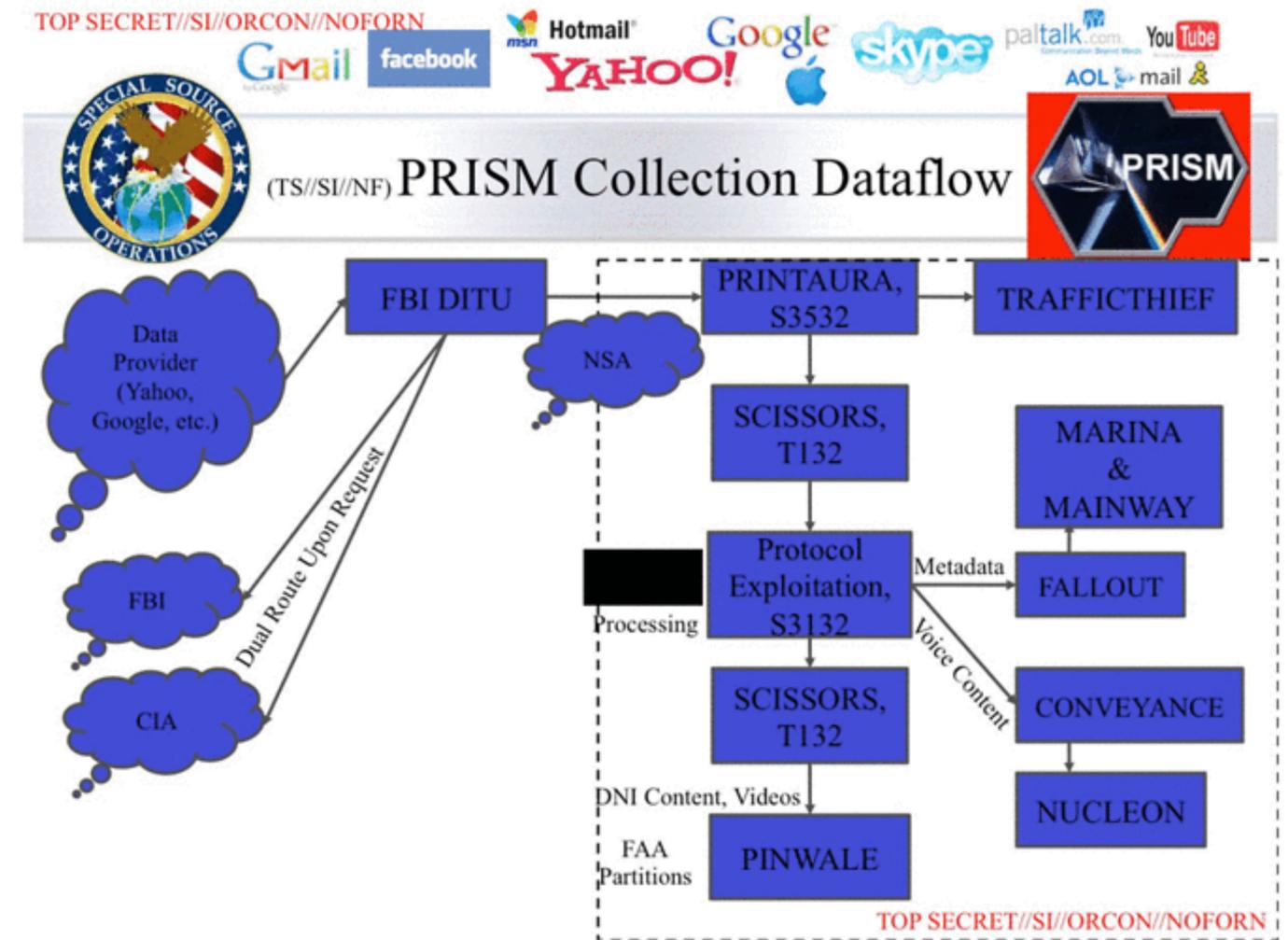
**freifunk.net**



# Warum man System Ingenieur wird

## Datenschutz

- Netzwerkdienste erzeugen Nutzerdaten
- Schutz dieser Daten erfordert Vertrauen in die Infrastruktur
- Vertrauen in die Infrastruktur erfordert exklusiven Zugang zu den Maschinen
- Traue keinem System, das du nicht anfassen kannst.



# Warum man System Ingenieur wird

## Ethik

- Jedes System betrifft Menschen, direkt oder indirekt
- Kein System ist ethisch neutral
- Systeme machen, was System Ingenieure ihnen sagen
- System Ingenieure machen (hoffentlich) nicht alles, was man ihnen sagt



# WARUM man System Ingenieur wird

- ✓ Jobsicherheit
- ✓ Autonomie
- ✓ Datenschutz
- ✓ Ethik



# WIE man System Ingenieur wird

- Mach soviel wie möglich selbst!
- Mach es dir nicht zu einfach!
- Lerne die Kultur!



# Mach soviel wie möglich selbst!

- Betreibe die von dir genutzten Dienste selbst:
  - Home Router
  - Website
  - git Repositories
  - DNS
  - ...
- Nutze Single-Board Computers:
  - PC Engines APU2
  - Raspberry Pi



# Mach soviel wie möglich selbst!

zB: Website

Du Nutzt:

Du Lernst:

---

Wordpress

Wordpress

---

Cloud Server  
(vorkonfiguriert)

ssh  
nginx.conf (evtl)

---

Raspberry Pi  
+ Raspbian

Linux  
ssh  
vi  
sudo  
apt  
nginx.conf  
certbot  
...



# Mach soviel wie möglich selbst!

zB: Home Router / Access Point

Du Nutzt:

Fritz!Box

Du Lernst:

Fritz!Box

OpenWRT  
auf LinkSys

OpenWRT GUI  
TCP/IP

PC-Engines APU2  
+ FreeBSD

Serial Console  
BSD  
ifconfig  
hostapd.conf  
pf.conf  
dhcpcd.conf  
unbound.conf  
TCP/IP  
...



# Mach es dir nicht zu einfach!

Text ist besser

- Command Line statt Graphical User Interface!
  - erleichtert Automatisierung
  - CLIs bieten meist mehr Optionen
  - Fehlermeldungen lassen sich leichter suchen
  - (Web)GUIs gehen häufig kaputt
- Betreibe deinen Raspberry Pi ohne Tastatur / Maus / Monitor!

ssh  
vim  
bash  
git  
gpg  
curl  
nc

# Mach es dir nicht zu einfach!

Text ist besser

- Config Files statt Konfigurationsassistenten!
  - erleichtert Wiederverwendung
  - Config Files bieten meist mehr Optionen
  - Beispiele sind leichter zu finden

sshd\_config

nginx.conf

pf.conf

git

# Mach es dir nicht zu einfach!

- Automatisiere alles, was zuviel Zeit kostet!
  - Scripting
  - Tooling (ansible, saltstack)
- Baue alles so, dass du es skalieren kannst!
  - egal ob nur 1 Computer, oder viele
  - egal ob nur 1 Nutzer, oder viele

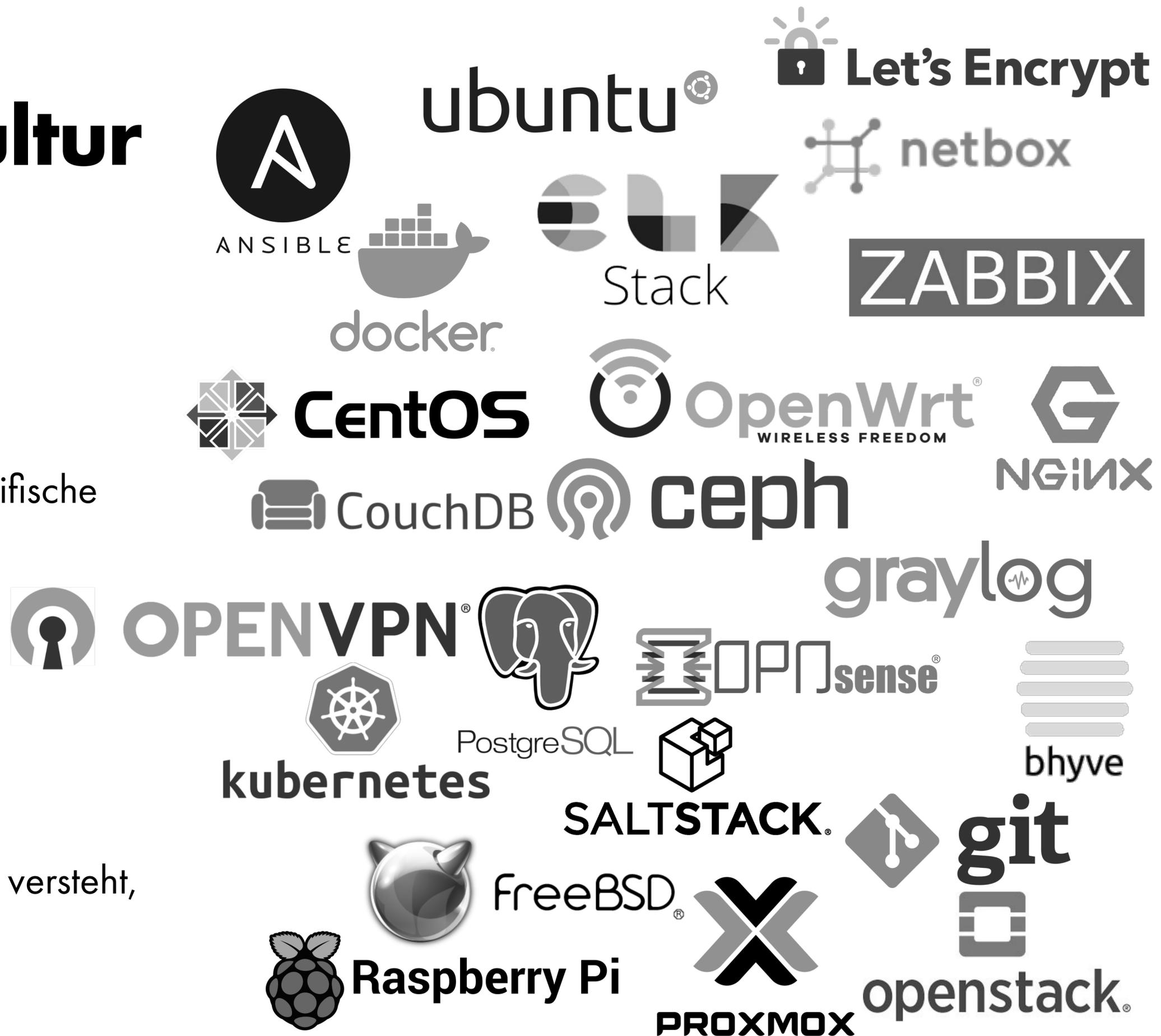
ansible

saltstack

terraform

# Software ist Kultur

- Alle Software ist von Gruppen von Menschen gemacht
- Jede dieser Gruppen hat ihre spezifische eigene Kultur und Sprache:
  - Wahl von Abstraktionen
  - Wahl von Namen
  - Syntax
- Wenn man die Kultur und Sprache versteht, versteht man die Software



# Software ist Kultur

- Alle Software ist von Gruppen von Menschen gemacht
- Jede dieser Gruppen hat ihre spezifische eigene Kultur und Sprache:
  - Wahl von Abstraktionen
  - Wahl von Namen
  - Syntax
- Wenn man die Kultur und Sprache versteht, versteht man die Software



OpenBSD

iked.conf, ikectl, pfctl, pf.conf, ikerd, rcctl, rc.conf, doas, doas.conf, httpd.conf, httpd, relayd, relayd.conf



Ubuntu

lsb\_release, service, systemd, apt, /var/lib/\*, /etc/sudoers, sudo, /etc/ufw/, iptables, dpkg



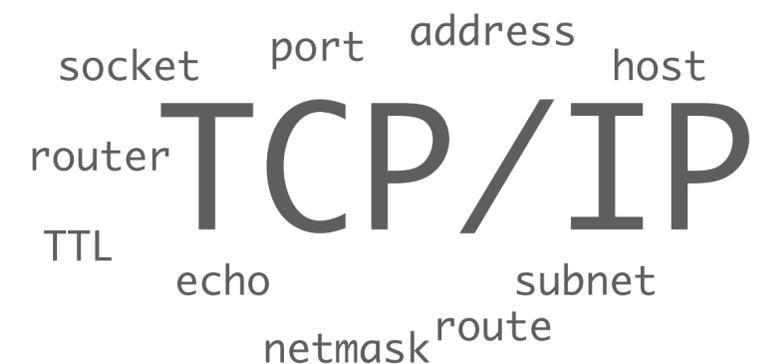
git

commit, reset, tag, remote, stash, add, push, pull, clone, module



docker

registry, run, node, swarm, ps, pull, run, image, service, container, push, commit, tag



TCP/IP

socket, port, address, host, router, TTL, echo, subnet, netmask, route



ansible

handler, task, playbook, group, tag, host, play, template, module, fact

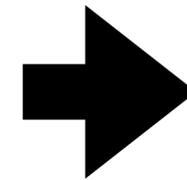
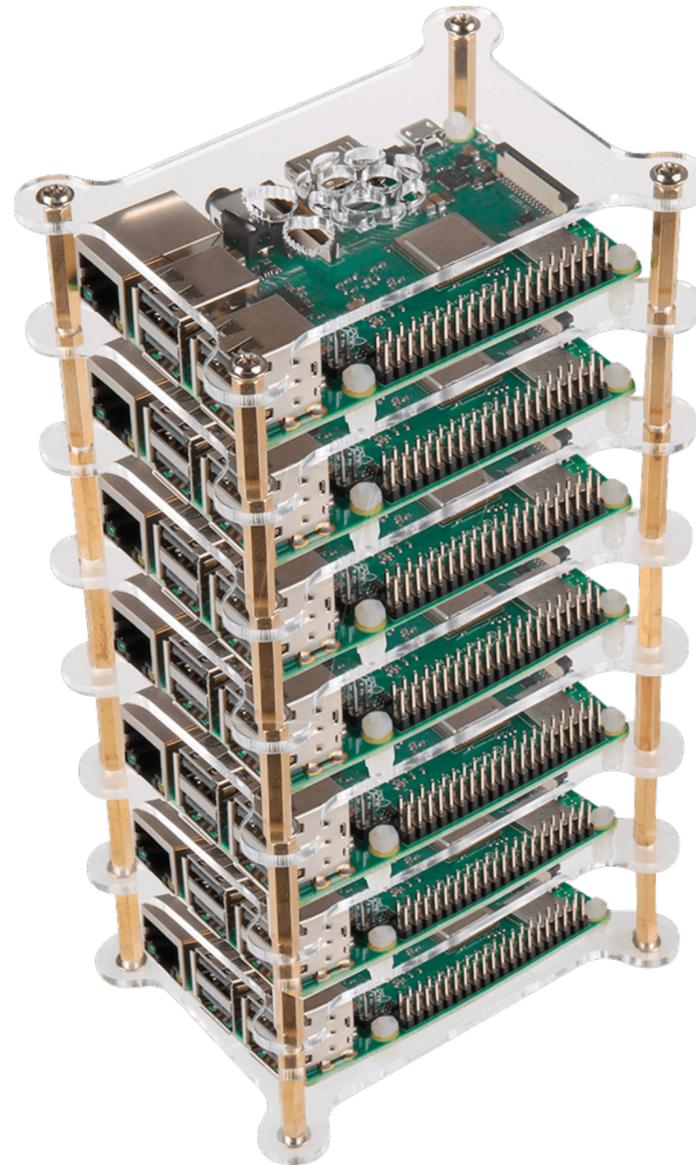
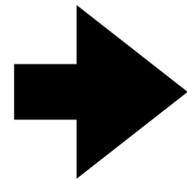
# WIE man System Ingenieur wird

- ✓ Mach soviel wie möglich selbst!
- ✓ Mach es dir nicht zu einfach!
- ✓ Lerne die Kultur!



# Ist alles nur UNIX!

- Deine Website läuft auf Raspberry Pi?
- Du kannst automatisch auf viele Raspberry Pi ausrollen?
- Im Datacenter läuft das nicht anders!



**EOF**