# Datenschutz & Kryptografie

## Daten

### Dateien verschlüsseln

GnuPG = GNU Privacy Guard
PGP = Pretty Good Privacy

Quellen:

- Projekt GnuPG
- GnuPG Befehle
- GnuPG bei Ubuntuusers
- GnuPG-Software (auch für Windows-Systeme)

Installation[1)]

```
sudo apt install gnupg2
```

#### Verschlüsseln mit Passwort

Das Passwort bei u.a. Befehlen wird jeweils abgefragt.

Einzelne Datei mit Passwort verschlüsseln

```
gpg -c <Original_DATEINAME>
```

Neuer Dateiname automatisch = <Original_DATEINAME> + „.gpg“

Einzelne mit Passwort verschlüsselte Datei entschlüsseln

```
gpg -d <Krypt_DATEINAME>.gpg > <DATEINAME>
```

#### Verwaltung von Schlüsseln

Für das verschlüsseln wird ein Schlüsselpaar benötigt. Mit dem öffentlichen Schlüssel wird verschlüsselt, den kann jeder haben. So ist es kein Problem diesen auf der eigenen Webseite zu veröffentlichen. Mit dem privaten Schlüssel wird entschlüsselt, der muss geheim bleiben. Verschlüsselt ein Dritter mit dem öffentlichen Schlüssel, kann auch dieser die Datei nicht mehr ohne den privaten Schlüssel entschlüsseln. da er i.d.R. das Original besitzt, ist das auch nicht notwendig.

**Der private Schlüssel muss unbedingt vor fremdem Zugriff geschützt bleiben!**
Das gilt auch für den wie u.a. exportierten Schlüssel für die E-Mail-Nutzung.
Nach Aussen gegeben wird NUR der öffentliche Schlüssel.

Vorhandene **private Schlüssel** auflisten

```
gpg2 --list-secret-keys
```

```
gpg2 -K
```

Vorhandene **öffentliche Schlüssel** auflisten

```
gpg2 --list-keys
```

```
gpg2 -k
```

Schlüsselpaar erzeugen

```
gpg2 --full-gen-key
```

Der Schlüssel befindet sich dann im Home-Verzeichnis des ausführenden Users unter ~/.gnupg

Privaten Schlüssel exportieren (um ihn z.B. im E-Mailprogramm zu verwenden)

```
gpg --export-secret-keys --armor > my-secret-key.asc
```

Dateiname **my-secret-key.asc** nach Wunsch benennen (Erweiterung .asc sollte bleiben). Diese durch das System neu erstellte Datei befindet sich anschließend in dem Verzeichnis, in dem der Befehl ausgeführt wurde.

Öffentlichen Schlüssel exportieren

```
gpg -a --output gpg-key --export <Schlüssel-ID oder Name>
```

```
gpg -a --export <Schlüssel-ID oder Name> | tee gpg-key
```

### Verschlüsseln mit Schlüsseln

Datei mit Schlüssel verschlüsseln

```
gpg2 --encrypt -a --recipient <Name oder Key_Id> test.txt
```

Datei mit Schlüssel entschlüsseln

```
gpg2 --decrypt --output entschluesselt.txt test.txt.asc
```

### Fingerprint und Signatur

siehe Wikipedia/Elektronische_Signatur
siehe Wikipedia/Fingerprint

Fingerprint anzeigen

```
gpg2 --fingerprint <ID oder Name des Schlüssels>
```

Datei Signieren

```
gpg2 --detach-sig -a test.txt
```

Datei Signatur überprüfen

```
gpg2 --verify test.txt.asc
```

## E-Mails verschlüsseln

E-Mails digital signieren und verschlüsseln

In diversen E-Mail-Programmen ist OpenPGP bereits enthalten und neue Schlüssel können direkt damit erstellt werden.
Der öffentliche Schlüssel wird an den Kommunikationspartner geschickt, der die von ihm verfassten E-Mails damit verschlüsselt. (Nur) der Besitzer des privaten Schlüssels kann die E-Mail dann wieder entschlüsseln.
Lesen klappt nur mit einem System, auf dem der Schlüssel hinterlegt ist. Ist der private Schlüssel nur auf dem Notebook hinterlegt, kann die selbe E-Mail z.B. auf dem Handy nicht entschlüsselt werden. Der private Schlüssel muss dann auch dort hinterlegt werden.

### Thunderbird

Schlüsselpaar erzeugen:
Extras → OpenPGP-Schlüssel verwalten → erzeugen → neues Schlüsselpaar.

Beim Einrichten von OpenPGP muss bei Thunderbird die Anwendung des privaten Schlüssels in den Konten-Einstellungen unter Ende-zu-Ende-Verschlüsselung aktiviert werden.

### E-Mail Schlüssel veröffentlichen

Der öffentliche Schlüssel kann auf der Webseite von openpgp.org veröffentlicht werden. Dann hat jeder Zugriff auf den öffentlichen Schlüssel und kann danmit E-Mails an den dort hinterlegten Empfänger verschlüsseln. Der Schlüssel kann direkt über Thunderbird (Konto-Einstellungen → Ende-zu-Ende-Verschlüsselung→Veröffentlichen (Buttom im Feld des Schlüssels) veröffentlicht oder auf der Seite von openpgp.org hochgeladen werden. Die abschließende Veröffentlichung erfolgt nach Bestätigung der eingehenden E-Mail.

### E-Mail Anhang verschlüsseln

Wenn die Entschlüsselung nach dem Verfahren mit Schlüssel für den Empfänger nicht geeignet/möglich ist, kann stattdessen ein E-Mail-Anhang verschlüsselt werden, bevor er der E-Mail angehängt wird. Siehe einzelne Datei mit Passwort verschlüsseln. Der Empfänger benötigt dann natürlich das Passwort und das sollte ihm im günstigsten Fall über einen 2. Kanal (z.B. per Telefon) zur Verfügung gestellt werden. In keinem Fall darf es in derselben E-Mail enthalten sein!
Abhängig von der notwendigen Sicherheitsstufe sollte auch die Komplexität bzw. die Länge des Passwortes gewählt werden. Einfache Passwörter sind ungeeignet, letztlich nur für unwichtige Dokumente richtig und dann kann man auch Unverschlüsselt senden.

## Partitionen verschlüsseln

Handelt es sich um die Systempartition, muss dies bereits während der Installation des OS erfolgen.
Achtung: Muss ein Server mit verschlüsseltem System neu gestartet werden, dann kann man dies nicht aus der Ferne durchführen. Das Passwort muss lokal, vor Ort eingegeben werden!!!
Siehe auch Installation Debian, Partitionierung.
Option: System und Daten trennen und nur Datenbereiche verschlüsseln, dann lässt sich das auch aus der Ferne realiseren. Hierzu kann z.B. Veracrypt genutzt werden.

Partitionen anzeigen

```
lsblk
```

Speicherplätze anzeigen

```
df -h
```

https://wiki.ubuntuusers.de/LUKS/

Mit cryptsetup (LVM-LUKS)
Anzeige der Header-Information[2]:

```
sudo cryptsetup luksDump /dev/nvme0n1p3
```

Anzeige Status:

```
sudo cryptsetup status /dev/nvme0n1p3_crypt
```

Passwort: Bis zu 8 Passwörter/Slots möglich (Keyslot 0 bis 7)
Neues Passwort/Slot anlegen

```
sudo cryptsetup luksAddKey /dev/nvme0n1p3
```

Passwort/Slot löschen <SLOT> = 0 bis 7[3]

```
sudo cryptsetup luksKillSlot /dev/nvme0n1p3 <SLOT>
```

Passwort ändern

```
sudo cryptsetup luksChangeKey /dev/nvme0n1p3
```

## Container & Laufwerke verschlüsseln

### LUKS

https://www.tuxedocomputers.com/de/Infos/Hilfe-Support/Anleitungen/LUKS-Verschluesselungskennwort-aendern.tuxedo

### Veracrypt

**Wenn es wirklich sicher sein muss:**
a) VeraCrypt kann ein **verstecktes Laufwerk** (hidden volume) erstellen. Dabei werden zwei verschiedene Passwörter für den selben Container vergeben. Und je nach Passworteingabe wird das sichtbare oder das versteckte Volumen geöffnet. Veracrypt empfiehlt, während der Installation in das sichtbare Laufwerk einige Daten zu legen, die vermeintlich wichtig sind - falls man **ein** Passwort, z.B. bei Entführung, verraten muss. Erst anschließend wird das versteckte Volumen erstellt. Das sichtbare Volumen sollte man dann nicht mehr verändern (keine neuen Daten aufspielen), weil dadurch das versteckte Volumen zerstört werden könnte.

b) Liegt der verschlüsselte Container auf einem entfernten Server, dann sollte dieser nicht auf dem Server „geöffnet“ werden, sondern auf dem lokalen System. Dadurch wird nur der verschlüsselte Container Online übertragen und nicht die darin enthaltenen Dateien bzw. Daten, wodurch die Datenübertragung zusätzlich gesichert ist. Die räumliche Trennung von Containerablage (auf dem Server) und Passworteingabe (auf dem lokalen System) macht zusätzlich Sinn, damit das PW nicht in irgendeinem Zwischenspeicher auf dem Server erhalten bleiben kann.

c) In der Standardeinstellung von Veracrypt bleibt die **Änderungszeit** der Containerdateien erhalten, auch wenn deren Inhalt verändert wurde. Dadurch ist auch der Zeitpunkt der letzten Änderung von außen nicht sichtbar.
Einstellung über die GUI-Anwendung: Einstellungen → Voreinstelllungen → „Änderungszeiten von Containerdateien erhalten“.
Hinweis: Soll jedoch von einer Containerdatei regelmäßig ein Backup erstellt oder diese synchronisiert werden (Nextcloud, etc.), kann ggf. nicht erkannt werden, ob sich innerhalb des Containers Änderungen ergeben haben und der Vorgang kann scheitern.
Lösung: Auf betroffenen Systemen o.a. Änderungszeiten … erhalten → **abschalten**. Die Änderung bezieht sich auf das System und nicht auf den Container!

Hier Installation und Befehle über Konsole.
Beispielhaft für ein verschlüsseltes und am Server angeschlossenes USB-Speichermedium.

https://veracrypt.fr/en/

https://github.com/arcanecode/VeraCrypt-CommandLine-Examples/blob/main/Linux/post.md

Download Programm über Konsole mit

```
wget https://launchpad.net/veracrypt/trunk.....deb
```

Pfad & Dateiname holen unter https://www.veracrypt.fr/en/Downloads.html

Installation

```
sudo apt install ./veracrypt-console-.....deb
```

Anzeige der Speichermedien

```
lsblk
```

```
df -h
```

Einhängen

```
veracrypt --text --mount /dev/<SDB> /<ZIELPFAD>/ --password <PASSWORD> --pim
0 --keyfiles "" --protect-hidden no --slot 1 --verbose
```

Anpassen:
<SDB> Ort der Festplatte → Abfrage über „lsblk“
<ZIELPFAD> (existierendes) Verzeichnis
<PASSWORD>

Aushängen

```
sudo veracrypt --text --dismount --slot 1
```

Anzeige gemounteter Volumen

```
sudo veracrypt --text --list
```

### Veracrypt Auto-Mount

Soll eine Partition oder eine Container-Datei automatisch nach dem Einloggen eingebunden werden, dann muss hierfür auch das Passwort auf dem ungeschützten Teil des PCs hinterlegt werden. Das ist grundsätzlich problematisch, da auch ein potentieller Angreifer, der im Besitz des Gerätes ist, das Passwort auf der Festplatte finden kann. Trotzdem: Variante 1.
Ist die Systemplatte geschützt, Passworteingabe bereits beim bzw. zum Booten des Rechners erforderlich, kann das Passwort für weitere Festplatten in dem dadurch geschützten Bereich abgelegt werden. Siehe Variante 2.

#### Variante 1

https://www.computercorrect.com/2018/operating-systems/linux/ubuntu/auto-mounting-a-veracrypt-volume-under-ubuntu-debian-linux/
https://www.freedesktop.org/software/systemd/man/latest/crypttab.html
Block (Volumen) beim Booten anlegen

```
sudo nano /etc/crypttab
```

```
volume_name /dev/sda password tcrypt-veracrypt,tcrypt-
keyfile=/path/to/keyfile
```

- volume_name kann frei gewählt werden und wird als bekanntes Volumen im Pfad /dev/mapper/ hinzugefügt
- /dev/sda = Pfad des Volumens oder des Containiers, das verschlüsselt ist
- password
  - none = Abfrage des Passwortes beim booten.
  - Alternative: Pfad zu einer Datei, die ausschließlich das Passwort enthält.
  - Alternative: Wird ein keyfile genutzt, dann wird hier eingetragen /dev/null
- Wird eine keyfile benutzt, dann wird der Pfad hier eingetragen. Wenn keiner verwendet wird, dann ist der Text „,tcrypt-keyfile=/path/to/keyfile“ inkl. Komma zu entfernen.

Ansicht der bestehenden Blocks

```
lsblk
```

Block/Volumen beim Booten einbinden (mount)

```
sudo nano /etc/fstab
```

```
/dev/mapper/volume_name /mnt/point auto nosuid,nodev,nofail 0 0
```

- volume_name, wie oben ausgewählt (Pfad /dev/mapper gehört trotzdem dazu)
- /mnt/pont = wie gewünscht

#### Variante 2

```
sudo nano /secureplace/decrypt.sh
```

```
#!/bin/bash
sudo veracrypt -t /dev/sda /mnt/xyz --non-interactive -p <PASSWORD>
```

<PASSWORD> ersetzen mit dem echten Passwort

Datei /etc/fuse.conf editieren:

```
sudo nano /etc/fuse.conf
```

```
mount_max = 1000        | das # davor entfernen
user_allow_other        | das # davor entfernen
```

<USER> (Name ändern) zur Gruppe users hinzufügen

```
sudo usermod -a -G users <USER>
```

Datei /etc/sudoers editieren (am Ende einfügen):

```
sudo visudo
```

```
%users  ALL=(root) NOPASSWD: /usr/bin/veracrypt
```

Automatisch beim Starten über /etc/crontab ausführen

```
sudo nano /etc/crontab
```

```
@reboot <USER>      /secureplace/decrypt.sh
```

am Ende einfügen, Pfad s.o., und <USER> anpassen

# Datenübertragung

Siehe auch Netzwerktraffic analysieren

## SSH-Verbindungen

Secure Shell oder SSH bezeichnet ein kryptographisches Netzwerkprotokoll für den sicheren Betrieb von Netzwerkdiensten über ungesicherte Netzwerke.
Siehe SSH-Verbindungen.

## VPN-Tunnel

Virtual Private Network (deutsch „virtuelles privates Netzwerk“; kurz: VPN) bezeichnet eine Netzwerkverbindung, die von Unbeteiligten nicht einsehbar ist.
Siehe WireGuard - VPN-Server

## Webseiten mit TLS/SSL

Transport Layer Security (TLS, englisch für „Transportschichtsicherheit“), auch bekannt unter der Vorgängerbezeichnung Secure Sockets Layer (SSL), ist ein Verschlüsselungsprotokoll zur sicheren Datenübertragung im Internet.
Siehe SSL-Verbindung mit Apache Web-Server

## Datenübertragung mit SFTP/FTP

**SFTP**: SSH File Transfer Protocol bzw. Secure File Transfer Protocol, ein Netzwerkprotokoll
**FTP**: File Transfer Protocol ohne Verschlüsselung & Signatur.

Siehe FTP-Zugang

### E-Mails

Die Verbindungssicherheit erfolgt i.d.R. über den E-Mail-Provider in Kombination mit der Einrichtung des E-Mail-Programms.
Im Falle des Betriebs eines eigenen E-Mail-Servers siehe: Mail-Server.
Ist die Übertagungssicherheit nicht gewährleistet, kann entweder die gesamte E-Mail verschlüsselt oder verschlüsselte Dateien mit der E-Mail gesandt werden.

## Identität schützen

### Internet Surfen mit Tor

Mit dem Tor Browser anonym bleiben. Gegen Verfolgung und Überwachung verteidigen, Zensur umgehen.

Tor läuft über verschlüsselte End-to-End-Verbindungen und ist vernetzt über diverse Server in diversen Ländern. Mit Tor surft man anonym und es ist nicht möglich, eine Verbindung zwischen den abgerufenen Inhalten und dem Abrufenden herzustellen – Verfolgung auch nicht möglich für: Netzanbieter, Abhörende, Seitenbetreiber oder Google-Analytics.
Die Route der Daten verläuft über mehrere Länder. Die Start- und Endpunkte der Route werden von Tor frei, vermutlich zufällig, ausgewählt.

Die Route kann auch manuell eingestellt werden.
Das hat allerdings zwei Nachteile:

1. kann zu Verlusten bei der Performance führen. Das Suchen nach passenden Servern dauert ggf. länger.
2. es ist weniger sicher, da sich wiederholende Pfade und Muster eher nachvollziehen lassen.

Das sollte also nur gemacht werden, wenn es triftige Gründe dafür gibt. Sinniger weise sollte es später, wenn der Grund weggefallen ist, wieder rückgängig gemacht werden.
Möchte man simulieren, sich in einem bestimmten Land aufzuhalten, z.B. weil bestimmte Internet-Angebote auf ein Land begrenzt sind, oder gerade eben nicht verboten sind (ohne Zensur), genügt es den ExitNode einzutragen.

Datei torrc aufrufen. Pfad entsprechend ergänzen:

```
nano .../tor-browser/Browser/TorBrowser/Data/Tor/torrc
```

Am Ende Einfügen:

```
EntryNodes {gb} StrictNodes 1
ExitNodes {ca} StrictNodes 1
```

{gb} = Ländercodes. Beispiele DE = Deutschland, GB = United Kingdom, CA = Kanada.
Groß-Klein-Schreibung nicht relevant.

**StrictNodes**:
1 = nur dieses Land darf verwendet werden. Suche nach passenden Servern kann ggf. lange dauern.
0 = dieses Land bevorzugt verwenden.

Auf dem Debian 10 habe ich Tor zunächst über Discover installiert. Mit dieser Installation hat die Länderkonfiguration nicht geklappt. Download über die Internetseite torprojekt.org, entpacken und starten hat problemlos geklappt.

Quellen:

- wikihow.de
- ubuntuusers
- Ländercodes

1)
gnuPG2 is an encryption tool that includes digital signatures and certificates.
2)
**/dev/nvme0n1p3** = die hier gewählte verschlüsselte Partition
3)
Vor dem Löschen ein neues anlegen!!!

From:
https://wiki.kauris.de/ - **wiki**

Permanent link:
**https://wiki.kauris.de/doku.php/open:it:crypto**

Last update: **2025/10/05 17:27**