sandstorm

Sandstorm Sicherheitskonzept

Unser IT-Sicherheitskonzept beschreibt, wie wir bei Sandstorm mit den Themen IT-Sicherheit und Datenschutz umgehen, welche Absicherungen wir vornehmen und wie wir unsere eigenen und die Daten unserer Kunden schützen.

Transparenz:
Unsere Kunden und Partner sehen, wie wir mit Datenschutz und Datensicherheit umgehen.

Interne Konsistenz:
Dieses Dokument ist gleichzeitig das Handbuch für alle Teammitglieder bei Sandstorm.

Wir freuen uns über Feedback jeglicher Art zu diesem Dokument!

Schutzziele

Vertraulichkeit:
Uns anvertraute Daten sollten nie in die Hände nicht autorisierter Personen gelangen.

Verfügbarkeit:
Unsere Teammitglieder müssen mit allen für sie relevanten Daten arbeiten können.

Integrität:
Daten dürfen nicht unbemerkt verändert werden - insbesondere unsere Hardware (Laptops) dürfen keine Viren, Backdoors, etc. aufweisen.

Bedrohungsmodell (Threat Model)

Wir möchten uns insbesondere vor folgenden Bedrohungen schützen:

- Nicht-Teammitglieder besuchen das Büro
- Geräte (Handy, Laptop, Yubikey, Backupfestplatte) gehen verloren, werden gestohlen oder gehen kaputt
- Viren und Backdoors auf Laptops
- Einbruchsversuche auf Servern bspw. über HTTP und/oder SSH

Technische und organisatorische Maßnahmen für IT-Sicherheit

1. Zutrittskontrolle

Firmen-Räume müssen grundsätzlich abgeschlossen und nicht öffentlich zugänglich sein.

- In Dresden erfolgt die Zutrittskontrolle mittels personalisierter Key Cards. Die Teammitglieder sind unterwiesen, wie im Falle von Kartenverlust vorzugehen ist (Karte wird gesperrt). Mitarbeiter des Gebäudeservices haben in Dresden außerdem Zutritt zu den Räumlichkeiten (über personalisierte Key Cards).
- In Darmstadt arbeiten wir in CoWorking-Spaces; d.h. nur die Mieter des CoWorking-Spaces haben Zutritt zu den Räumen.
- Arbeitsgeräte (z.B. Laptops, Backup-Festplatten) werden in allen Standorten bei Nichtbenutzung in abgeschlossenen Schränken verwahrt.
- Personenbezogene Daten in gedruckter Form werden in allen Standorten in abgeschlossenen Schränken verwahrt und nur bei Nutzung herausgeholt.

Unsere Server mieten wir von professionellen Hostern, deren Rechenzentren zertifizierte Sicherheitskonzepte einsetzen.

2. Zugangskontrolle

Ein kompromittiertes System darf nicht dazu führen, dass man auf alle anderen Systeme Zugriff hat. Sensible Daten werden nur auf Firmeneigentum gespeichert (Firmen-Laptops, Backup-Festplatten, Firmen-USB-Sticks).

Wir verwenden kein iCloud Backup für unsere Macs.
Alle Firmengeräte sind verschlüsselt.

- Mac OS X: Verschlüsselung per File Vault. Der Wiederherstellungsschlüssel ist im Recovery Vault in unserem Passwort-Manager abgelegt.
- Backupfestplatte: Verschlüsselung per File Vault. Der Wiederherstellungsschlüssel ist im Recovery Vault in unserem Passwort-Manager abgelegt, um im Falle eines System-Crashes die Daten wieder einlesen zu können.

- Firmen-Handy: (Sicherheitscode, Entsperrmuster, ...)
- USB-Sticks: Verwendung von FileVault oder Veracrypt

Firmen-Laptops und -PCs sperren sich bei Nichtbenutzung automatisch innerhalb von wenigen Minuten. Wenn ein Teammitglied seinen Laptop verlässt, muss dieses den Laptop sperren. Das Passwort muss sofort nach der Sperrung eingegeben werden. Zugriff auf Systeme ist nur mit personalisierten Accounts möglich.

Wir setzen für wichtige Domänen Multi-Factor Authentication (MFA) ein:

- Es gibt ein Passwort zur Absicherung der Services.
- Zusätzlich gibt es ein One Time Password; entweder per Handy (Google Authenticator) oder per Yubikey (U2F)
- Es müssen mindestens zwei Geräte bei der MFA beteiligt sein (bspw. Laptop, Yubikey). MFA-Funktionalitäten von Passwort-Managern (bspw. 1Password) dürfen nicht verwendet werden.
- Der zweite Faktor (Yubikey) sollte immer am Schlüsselbund getragen werden, insbesondere darf er nicht dauerhaft im/am Laptop verbleiben.
- Der Yubikey darf nicht als einziger Authentisierungsfaktor verwendet werden; also darf er nicht für den Laptop-Login verwendet werden.

Wir verwenden einen Passwort-Manager:

- es muss für jeden Dienst ein neues Passwort vergeben werden, sodass im Falle eines Leaks bei einem Dienst die anderen Dienste nicht betroffen sind.
- Bestimmte Passwörter (insbesondere zu nicht-kritischen Tools) können mit einzelnen oder allen Sandstormern geteilt werden.
- Idealerweise werden Passwörter nicht geteilt. Je sensibler geteilte Passwörter sind, umso weniger Sandstormer haben darauf Zugriff. Zum Beispiel auf den Recovery Vault haben kaum Sandstormer Zugriff.
- Das Passwort für den Passwort-Manager darf an keiner Stelle abgelegt werden.
- Die Passwörter werden mit einem Zero-Knowledge-Server gespeichert und übertragen. Die Ver- und Entschlüsselung erfolgt lokal. Ein Admin-Nutzer auf diesem Server ist nicht in der Lage, auf Passwörter zuzugreifen.

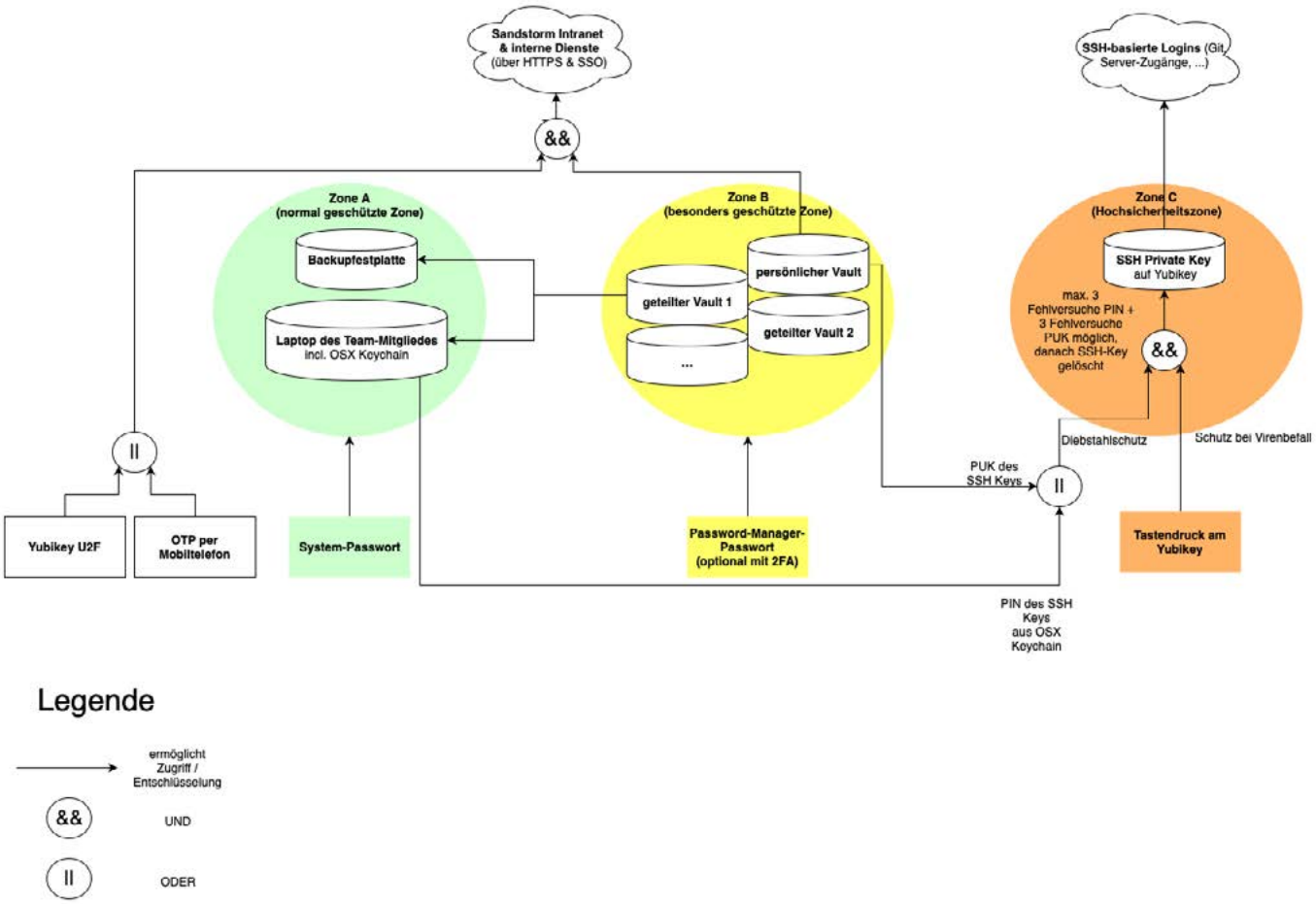
- Ein Passwort-Manager Admin ist ebenfalls nicht in der Lage, sich Zugriff auf Passwörter zu verschaffen, die nicht vom Ersteller geteilt werden.

Passwort-Richtlinien: Es sind grundsätzlich starke Passwörter zu verwenden.
Passwörter werden niemals mehrfach verwendet.
System-Updates werden regelmäßig eingespielt.

2.1 Sicherheitszonen

Es gibt verschiedene Sicherheitszonen, welche jeweils mit einem unterschiedlichen Passwort geschützt sein müssen:

- Systembenutzer auf dem Laptop
- Passwort Vaults des Nutzers
- Mail/Slack-Zugangsdaten (dürfen im persönlichen Passwort Vault abgelegt werden)
- Intranet-Login (darf im persönlichen Passwort Vault abgelegt werden)



Sicherheitszone A – Laptop-Systembenutzer
Diese Zone enthält alle Daten des Nutzers auf dem Laptop, verschlüsselt abgelegt. Dies ist der Basis-Schutz für all unsere Daten.
Sobald der Nutzer angemeldet ist, ist diese Sicherheitszone entsperrt.

Sicherheitszone B – Password Vault
Diese Zone enthält die Password Vaults des Nutzers, gesondert verschlüsselt abgelegt. Der Password-Manager ist so eingestellt, dass nach einiger Zeit ohne Passworteingabe der Vault wieder gesperrt wird.
Diese Zone bietet etwas mehr Sicherheit als Zone A, da der Vault nur dann entsperrt ist, wenn tatsächlich ein Passwort benötigt wird (und sich automatisch wieder sperrt). Zusätzlich ist ein Login nur mit 2nd Factor (i.d.R. Yubikey) möglich.

Sicherheitszone C – SSH Private Key

- Da der SSH-Key auf dem Yubikey gespeichert ist, kann verhindert werden, dass bspw. durch Viren der private SSH-Key ausgelesen wird.
- Wenn der Yubikey verloren oder gestohlen wurde, muss sichergestellt sein, dass der SSH Key nicht durch andere Personen nutzbar ist.
- aus diesem Grund gibt es eine PIN und eine PUK - bei 3-maliger Falscheingabe der PIN + 3-maliger Falscheingabe der PUK wird der SSH-Key unwiederbringlich zerstört. (Diebstahlschutz)
- Die PIN wird bei uns in der OSX Keychain gespeichert, damit sie nicht jedes mal eingegeben werden muss. Somit müsste ein Angreifer 1) die OSX-Keychain stehlen, 2) den Yubikey (in Hardware) stehlen. Dies ist aus unserer Sicht ein praktikabler Mittelweg, der Usability und Sicherheit balanciert.
- Der Yubikey ist so konfiguriert, dass jede Nutzung des Private Keys mit einem Tastendruck auf dem Yubikey bestätigt werden muss. Dies schützt vor Viren/Backdoors – diese könnten zwar einen Verschlüsselungsbefehl an den Yubikey schicken, der Nutzer müsste aber gesondert die Taste betätigen.
- Wenn der Yubikey verloren/gestohlen wurde, müssen die entsprechenden Public Keys auf den Servern ausgetauscht werden von einem Sandstormer, dessen Yubikey einen Zugriff ermöglichen.

3. Zugriffskontrolle

Wir nutzen individualisierte Logins, wo immer möglich, um gezielt Nutzerrechte zu vergeben und Audit Logging zu ermöglichen:

- System-Benutzer auf Laptops - Persönlicher Login; kein Gast-Account und kein gemeinsamer "Superadmin"-Account
- Intranet-Benutzer (HTTPS)
- Server-SSH-Logins mit individuellen Nutzern und jeweils individuellen SSH Keys
- Benutzer in Fremdsystemen (bspw. bei Kunden und Partnern)

Um auf unseren Servern per SSH Root-Rechte zu erhalten, erfolgt zuerst ein personalisierter User-Login per SSH (Key auf Yubikey), danach können mit sudo und einem systemspezifischen Nutzer-Passwort Root-Rechte erlangt werden (Password-Manager). Somit kann gesteuert werden, welche Nutzer sich als Root anmelden dürfen. Außerdem kann nachvollzogen werden, wann sich ein individueller Nutzer als Root angemeldet hat.

Für besonders sensible Daten soll ein zusätzlicher Veracrypt-Container eingesetzt werden.

SSH Keys sind für uns besonders sensibel, da wir mit ihnen nicht nur Zugriff auf unsere eigene Server-Infrastruktur erhalten, sondern auch Quellcode über Git austauschen sowie oftmals Zugriff auf Systeme unserer Kunden erhalten.

- Wir nutzen ausschließlich key-basierte Authentisierung.
- Wir möchten verhindern, dass eine Backdoor den privaten SSH-Key ausliest. Aus diesem Grund wird der persönliche SSH Key auf dem Yubikey generiert und kann dieses Gerät nicht verlassen (Secure Enclave). Yubikey im PIV-Mode unterstützt 2048 bit RSA keys, welche wir verwenden.
- Die Nutzung der Yubikeys wird unten im Detail beschrieben.

Die Server sind mit einer Firewall ausgestattet; nur die notwendigen Ports sind freigeschaltet.

4. Weitergabekontrolle

Hier erklären wir, wie wir mit Aspekten der Weitergabe personenbezogener Daten umgehen.

- Für personenbezogene Daten von Kunden werden keinerlei Cloud-Dienste verwendet.
- Slack ist die einzige Cloud-Applikation, welche wir beim täglichen Arbeiten einsetzen. Über Slack dürfen keine Passwörter, Zugänge, oder personenbezogenen Daten von Kunden kommuniziert werden. Personenbezogene Daten unserer Teammitglieder werden über Slack kommuniziert.
- Alle internen Services (Mail, Intranet, ...) sind ausschließlich per HTTPS und/oder SSH erreichbar. Mails von externen Dritten gehen manchmal über unverschlüsselte SMTP-Verbindungen ein, wenn der sendende Server diese Übertragung wählt.
- Wenn wir auf Kundensysteme zugreifen, erfolgt dies ausschließlich auf verschlüsseltem Wege (per SSH; teilweise auch mit VPN).
- Die Mac OS Firewall wird von allen Teammitgliedern benutzt. Diese verhindert die unkontrollierte Kommunikation der Außenwelt mit dem Laptop.
- Teammitglieder können eine Personal Firewall nutzen, um ausgehende Verbindungen zu beschränken; hier kommt Little Snitch zum Einsatz.

5. Eingabekontrolle

Für die Speicherung von personenbezogenen Daten innerhalb der Firma nutzen wir Git. Dies ermöglicht eine detaillierte Nachvollziehbarkeit, wann wer welche Daten eingegeben oder geändert hat.

In Systemen, die kundenbezogene Daten verarbeiten, führen wir ein Audit Log, Auch hier ist somit ersichtlich, welche Daten zu welchem Zeitpunkt von wem eingegeben / geändert wurden.

6. Auftragskontrolle

Wenn wir Auftraggeber sind:

- ist die Verarbeitung von Daten vertraglich geregelt
- verpflichten wir den Auftragnehmer auf die Einhaltung der gesetzlichen Bestimmungen, insbesondere der Datenschutzbestimmungen
- und wählen auch nur solche Dienstleister aus, welche diese Standards erfüllen.

Wenn wir Auftragnehmer sind:

- ist die Verarbeitung von Daten vertraglich geregelt
- verpflichten wir uns auf die Einhaltung der gesetzlichen Bestimmungen, insbesondere der Datenschutzbestimmungen
- schulen wir die mit der Umsetzung betrauten Teammitglieder nochmals gesondert über die Verarbeitung personenbezogener Daten.
- Im Falle eines Incidents wird innerhalb von 3 Werktagen der Ansprechpartner auf Auftraggeber-Seite kontaktiert.

7. Verfügbarkeitskontrolle

Für die Arbeits-Laptops der Teammitglieder erfolgt ein regelmäßiges (mindestens wöchentliches) Backup auf externe verschlüsselte Festplatten (s.o.). Serverseitig nutzen wir RAID1-Systeme zur Spiegelung der Festplatten, um Daten vor zufälliger Zerstörung oder Verlust zu schützen.

Unsere Server mieten wir von professionellen Hostern, deren Rechenzentren etablierte Paradigmen verwenden, um Hardware-Ausfälle mit damit einhergehendem Datenverlust zu verhinden (bspw. USVs, RAID, Backups).

Serverseitige Backups erfolgen auf einem der folgenden Wege:

- Es erfolgt ein Backup auf Backup-Server des Hosters. Dieses Backup ist mit symmetrischem Schlüssel per GPG verschlüsselt und wird nur in verschlüsselter Form übertragen. Die Schlüssel werden getrennt aufbewahrt.
- Es erfolgt ein Backup zu Cloud-Storage-Dienstleistern in Europa. Dieses Backup ist gleichermaßen mit symmetrischem Schlüssel verschlüsselt und wird nur in verschlüsselter Form übertragen. Die Schlüssel werden getrennt aufbewahrt.

8. Gewährleistung der Zweckbindung

Wir ergreifen unter anderem folgende Maßnahmen, um zu gewährleisten, dass zu unterschiedlichen Zwecken erhobene Daten getrennt verarbeitet werden:

- Nutzung verschiedener (teilweise physischer, teilweise virtualisierter) Systeme für die verschiedenen Datenerhebungszwecke.
- Kompartmentalisierung der verschiedenen Systeme, sodass diese jeweils nur auf die ihnen zugeordneten Daten zugreifen können (bspw. durch Nutzung von Virtualisierung, getrennten Service-Accounts in Datenbanken, Filesystem-Rechte).

Konkrete Angriffsszenarien

Handy gestohlen: ein Teil des second factor bei Authentisierung fällt weg; stattdessen kann Yubikey verwendet werden.

- Vertraulichkeit: nicht verletzt
- Verfügbarkeit: gewährleistet (durch Yubikey als anderen second factor)
- Integrität: nicht relevant

Laptop gestohlen / geht kaputt

- Vertraulichkeit: bei sicherem Systempasswort nicht verletzt, da Festplatte komplett verschlüsselt ist
- Verfügbarkeit: eingeschränkt gewährleistet. Es muss ein neuer Laptop eingerichtet und die Daten von der Backupfestplatte wieder hergestellt werden. Hierbei muss von einem autorisierten Teammitglied aus dem Founders Vault der Backup FileVault Key gelesen werden.
- Integrität: nicht relevant

Backupfestplatte gestohlen / geht kaputt

- Vertraulichkeit: nicht verletzt, da Festplatte komplett verschlüsselt ist
- Verfügbarkeit: gewährleistet
- Integrität: nicht relevant

Yubikey gestohlen / geht kaputt

- Vertraulichkeit: nicht verletzt, da SSH Key durch PIN + PUK geschützt sind, und Brute Force-Attacken verhindert werden.
- Verfügbarkeit: nicht gewährleistet. SSH-Zugänge müssen komplett neu eingerichtet werden.
- Integrität: nicht relevant

Yubikey + Laptop gestohlen

- Vertraulichkeit: bei sicherem Systempasswort nicht verletzt. Bei unsicherem Systempasswort VERLETZT – SSH-Key nutzbar!
- Verfügbarkeit: nicht gewährleistet. (siehe oben)
- Integrität: nicht relevant

Backdoor auf Laptop

- Vertraulichkeit: nicht gewährleistet für Daten auf Laptop, gewährleistet für SSH Key (da dieser physisch im Yubikey steht).
- Verfügbarkeit: gewährleistet
- Integrität: nicht gewährleistet (da Backdoor)

Einbruchsversuche auf Servern über HTTP und SSH

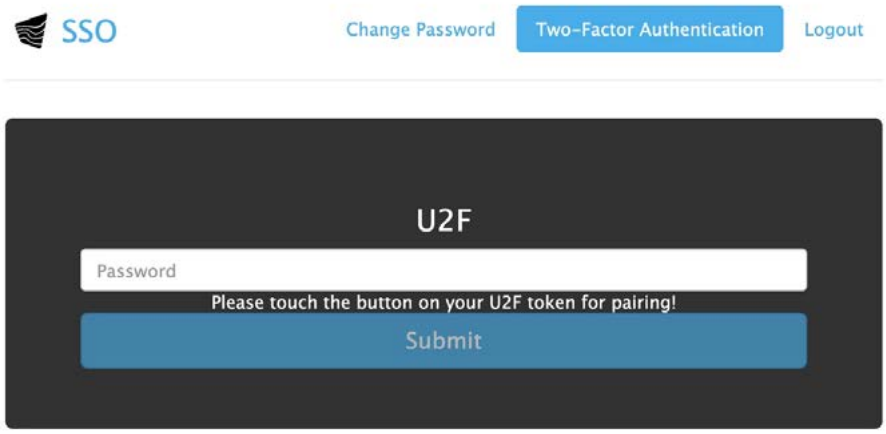
- SSH Keys werden spezifisch abgesichert
- für eingeloggte Bereiche verwenden wir HTTPS, idealerweise in Kombination mit U2F
- Logging/Auditing passiert per Root Cause Analyse-System
- Vertraulichkeit / Verfügbarkeit / Integrität: keine konkrete Aussage machbar; abhängig von der Art des Angriffes.

Konfiguration des YubiKeys für SSH

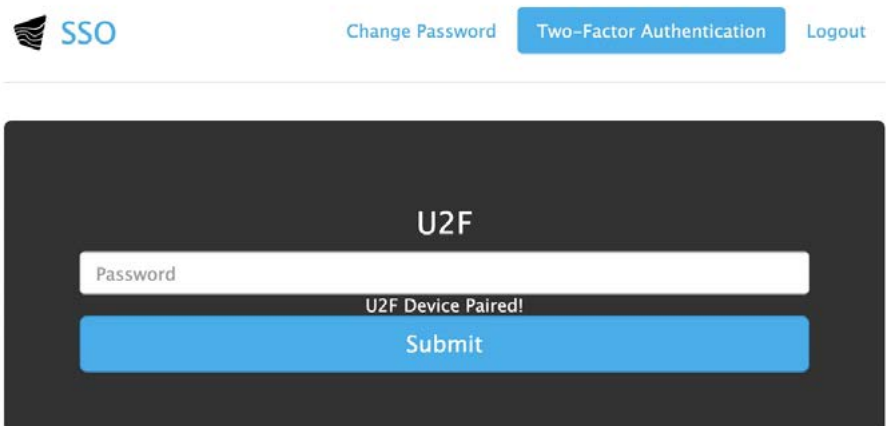
Vorbemerkung: Man kann entweder SSH-über-PKCS11 (PIV) einrichten; oder aber SSH-über-GPG. Wir haben uns für ersteren Weg entschieden, da dies deutlich einfacher in der Firma deploybar ist. Nachteil: Somit sind nur 2048 bit RSA Keys möglich; bei SSH-über-GPG wären es 4096 bit.

Für das konkrete Einrichten stellen wir Shell-Scripte zur Verfügung. Diese haben wir auf Github veröffentlicht > Sandstorm.YubiKeyProvisioner

Nach dem Einrichten wird der YubiKey in unserem SSO als zweiter Faktor hinterlegt.
Der öffentliche SSH-Schlüssel wird auf den entsprechenden System hinterlegt.
Eine SSH Verbindung kann nur mit verbundenem YubiKey aufgebaut werden.



Touch your YubiKey to start pairing.



Enter your auth.sandstorm.de login password and click on Submit to finish pairing.