Philipps Universität Marburg

Mehr-Faktor-Authentifizierung für 35.000 Benutzer

Am Beispiel Philipps-Universität Marburg

Referenten:
Bernd Nicklas, HRZ der Philipps-Universität Marburg
Cornelius Kölbel, Entwickler von privacyIDEA

Vorstellung

- Bernd Nicklas,
  Hochschulrechenzentrum der Philipps-Universität Marburg,
  nicklas@hrz.uni-marburg.de

- 2FA seit 2015

Philipps-Universität Marburg

- Etwa 35.000 Benutzer
  - ~ 29000 Studenten
  - ~ 4500 Mitarbeiter

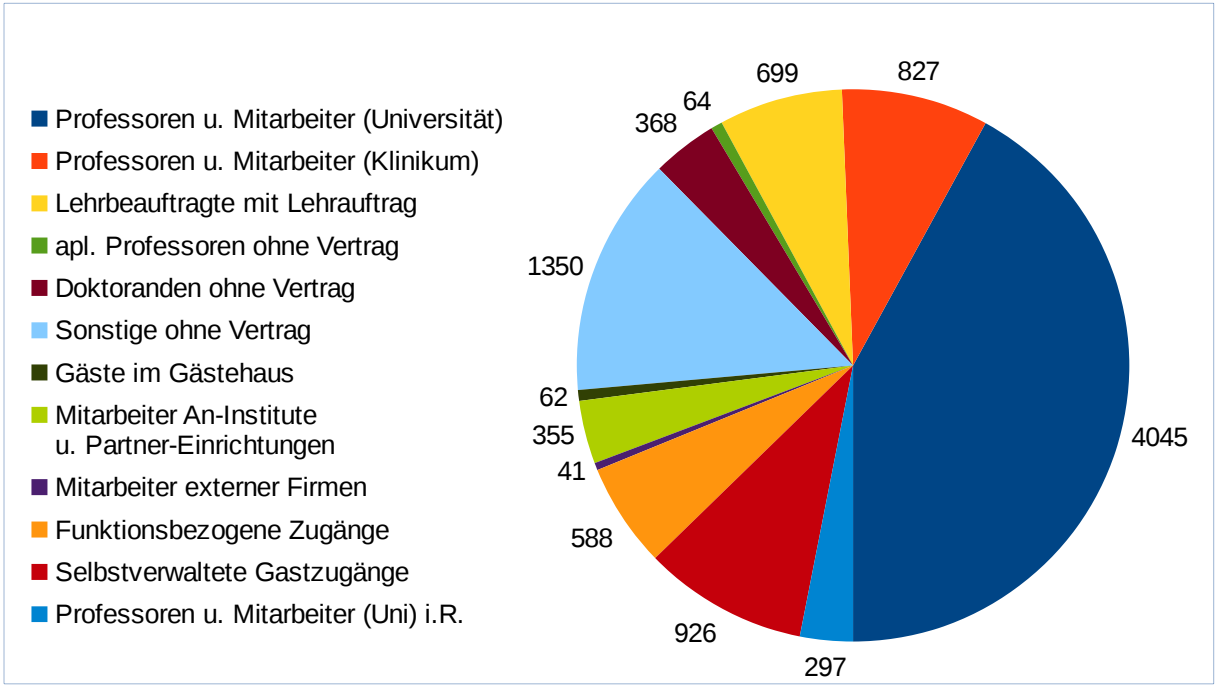
- Etwa 150 Gebäude verteilt über das gesamte Stadtgebiet

- Typische Einteilung in Universitätsverwaltung mit Studierendensekretariat und getrennter Verwaltungs-IT, sowie Fachbereiche und Institute, Einrichtungen, Hochschulrechenzentrum (HRZ), Studierendenvertretung, Personalvertretung

Philipps-Universität Marburg

„Mitarbeiter“

- ca. 9.622 Identitäten
  (davon 1.514 Nicht-Personen)
- nur ca. 8.559 Accounts
  (nicht alle Pers. haben Acc.)

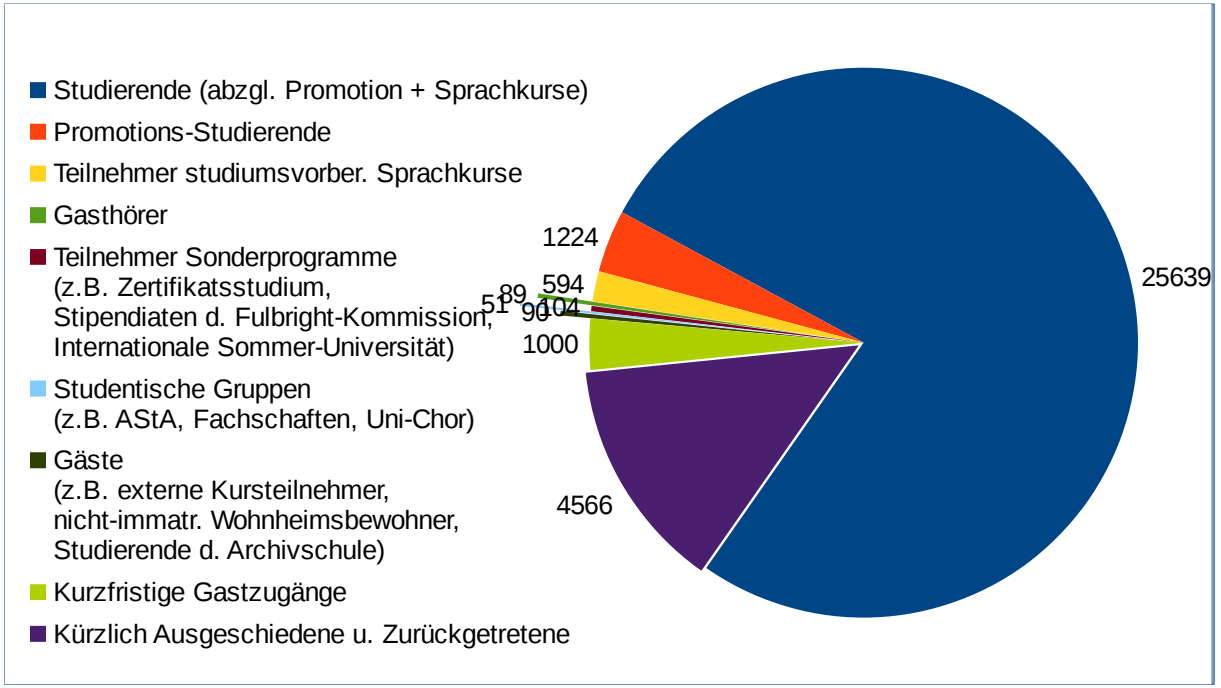


Statistik und Grafik: Manuel Haim, Identity Management, Hochschulrechenzentrum, Philipps-Universität Marburg

Philipps-Universität Marburg

„Studierende“

- ca. 33.357 Identitäten
  (davon 1.051 Nicht-Personen)
- ebensoviele Accounts



Statistik und Grafik: Manuel Haim, Identity Management, Hochschulrechenzentrum, Philipps-Universität Marburg

Philipps-Universität Marburg

- Äußerst in-homogene Benutzer-Struktur: Benutzer verfügen in der Regel über unterschiedlichste Tätigkeitsbereiche und unterschiedlichstes Spezialwissen, – aber in der Regel nicht über IT-Affinität

- Mitarbeiter z. T. ohne Büro, z. T. keine oder keine tagesaktuellen Büro- und Adressdaten im HRZ vorhanden.

- Daten der Studierenden werden im Campus-Managment-System gepflegt, Adressen sind aufgrund der halbjährlich benötigten Studienbescheinigung weitgehend aktuell

Hochschulrechenzentrum I

- Etwa 200 km Glasfaserleitungen vom HRZ verwaltet
- Etwa 25.000 Hosts, vom zentral verwalteten Arbeitsplatz-Rechner, über IP-Telefone bis hin zum Hardware-basierten oder virtualisierten Server zur Bereitstellung zentraler Dienste

Hochschulrechenzentrum II

- Große Anzahl unterschiedlicher Dienste für Professoren, Mitarbeiter und Studierende, u. a.
  - Identity Management (Accounts/Berechtigungen/Personeneinträge)
  - Internet Zugang (LAN/WLAN/VPN/DHCP/DNS), Telefon, Fax
  - E-Mail-Dienst und Webmail-Plattform
  - Plone Content Management System für Website
  - HISinOne Campus Management System
  - ILIAS E-Learning Plattform
  - High Performance Computing
  - Windows Arbeitsplatz-Rechner / Active Directory

Hochschulrechenzentrum III

- Große Anzahl unterschiedlicher Dienste für die Organisation des HRZ, u. a.
  - *Virtualisierungs-Dienste*: VMware; Ganeti (KVM); OpenStack (KVM)
  - *Installations und Konfigurations-Dienste*: FAI für die Installation von Linux-Servern, Puppet für das Management von Linux-Servern, Opsi für die Installation von Windows-Clients
  - *Verzeichnis-Dienste*: LDAP-Verzeichnis mit LDAP-Web-Portal für die Verwaltung von Accounts, Berechtigungen, Personeneinträgen, Host-, Telefon- und Netzwerkkonfigurationen; Active Directory für die Verwaltung von Windows-Rechnern, Windows-Accounts und Berechtigungen
  - *Monitoring-Dienste*: Icinga, Munin
  - *Source-Code-Management-Dienst*: GitLab (Instanz), Git
  - *Support-Ticket-System*: Request Tracker

Hochschulrechenzentrum IV

- Eingesetzte Server-Betriebssysteme überwiegend basierend auf Linux und anderen UNIX-artigen Betriebssystemen
- Eingesetzte Server-Softwares überwiegend basierend auf Open Source Softwares

Vorstellung

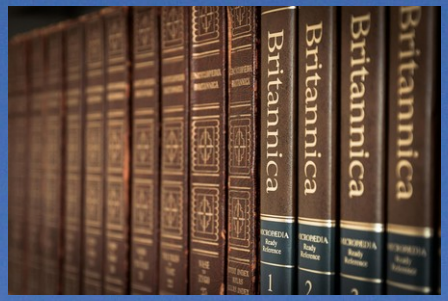
- Cornelius Kölbel
- 2FA seit 2005
  - Smartcards, Aladdin eToken, 2005 (HOTP) < 2007 (iPhone 1), privacyIDEA seit 2014
- Cornelius.koelbel@netknights.it
- @cornelinux


Das Passwort ist tot
Nur mit einem geeigneten Partner
kann es einen zweiten Frühling erleben.

Partnerwahl

- Der ideale Partner sollte :
- die gleichen Ziele verfolgen
- eine Ergänzung sein
- Schwächen des anderen ertragen und ausgleichen!

Zwei-Faktor-Authentifizierung



Passwort

Besitz

Eigenschaf

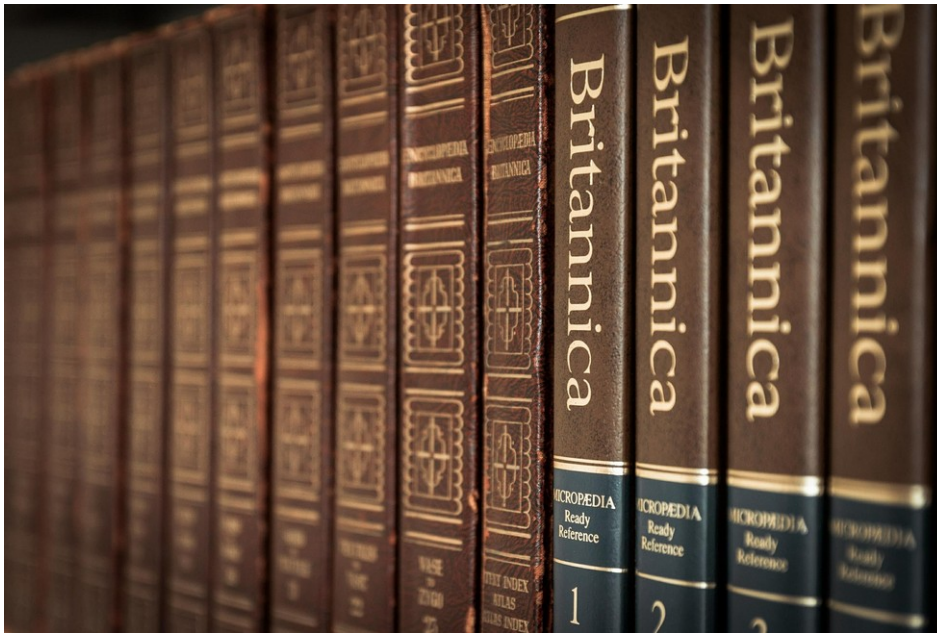
Sinn von 2FA

- Foto mit Skillprofil

Skillprofil!

Sinn von 2FA
Angriffe auf Wissen

- Angriffszenarien und Skill-Profile
  - Phishing / Social Engineering
  - SQL-Injection
  - Cracker / Skript-Kiddie
- Skaliert gut



Sinn von 2FA
Angriffe auf Besitz

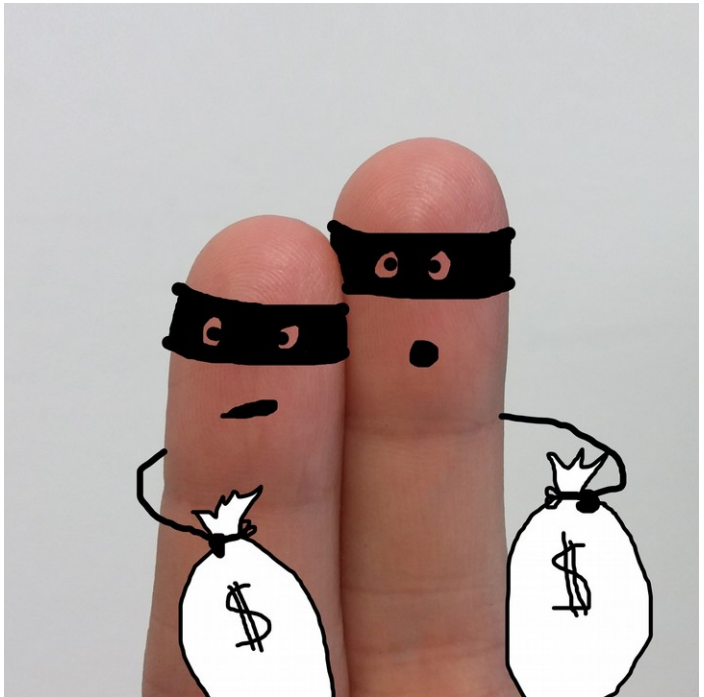
- Angriffszenarien und Skill-Profile
  - Physikalischer Diebstahl
  - Social Engineering
  - Zugriff auf Firmengebäude
  - Putzfrau,Besucher
- Skaliert schlecht

Sinn von 2FA
Angriffe auf Eigenschaft

- Angriffszenarien und Skill-Profile
  - Körperkontakt
  - Partybesucher / Einwohnermeldeamt
- Kann gut skalieren

Anforderungen an Faktoren

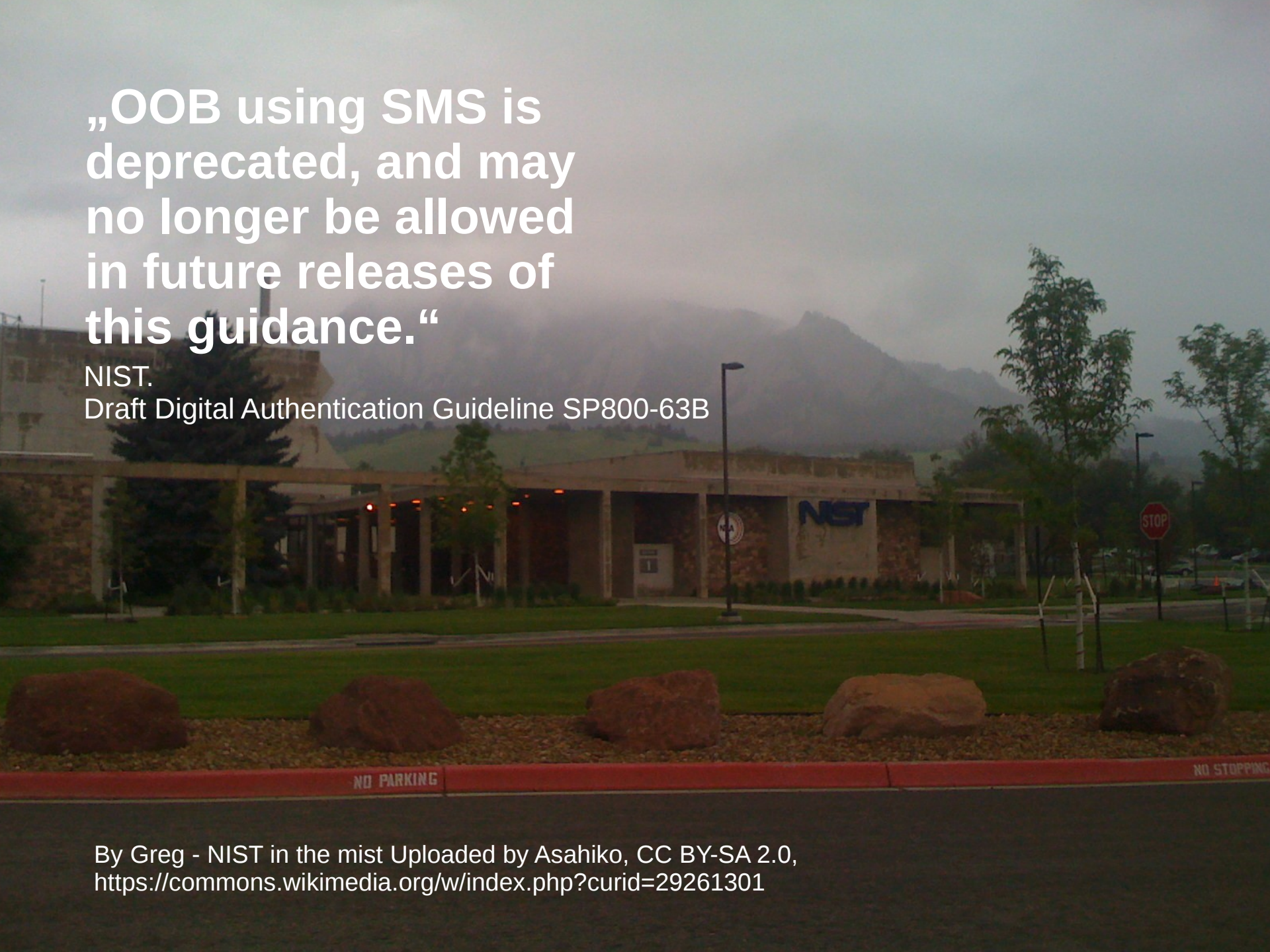
- Eindeutig → Nicht kopierbar
- Verlust sollte bemerkbar sein
- Revozierbar / Neu ausstellbar



Einzigartigkeit
des zweiten Besitz-Faktors

„OOB using SMS is deprecated, and may no longer be allowed in future releases of this guidance.“

NIST.
Draft Digital Authentication Guideline SP800-63B



Token-Typen

Eine Auswahl unterschiedlicher Token-Typen. Alle angezeigten Tokens sind Test-Tokens.

Token-Typen I

- Mögliche Kriterien zur Auswahl
  - Kosten für Beschaffung, Betrieb, Wartung, Support und Austausch, inkl. Kosten für zusätzlich nachzurüstende Schnittstellen und Treiber
  - Kosten für die Integration der mit dem Token verbundenen Authentisierungs-Technologie in die eigenen Dienste („Müssen alle Login-Dialoge und Authentisierungs-APIs der Dienste angepasst werden?“)

Token-Typen II

- Mögliche Kriterien zur Auswahl
  - Hardware-, Betriebssystem- und Anwendungs-Kompatibilität
  - Sicherheit, insbesondere (1) „Welche Angriffs-Szenarien werden unterbunden“ und (2) „wer initialisiert die Tokens (Hersteller/Kunde)“?
  - Werden offene Schnittstellen zum Initialisieren angeboten?
  - Formfaktor und Usability

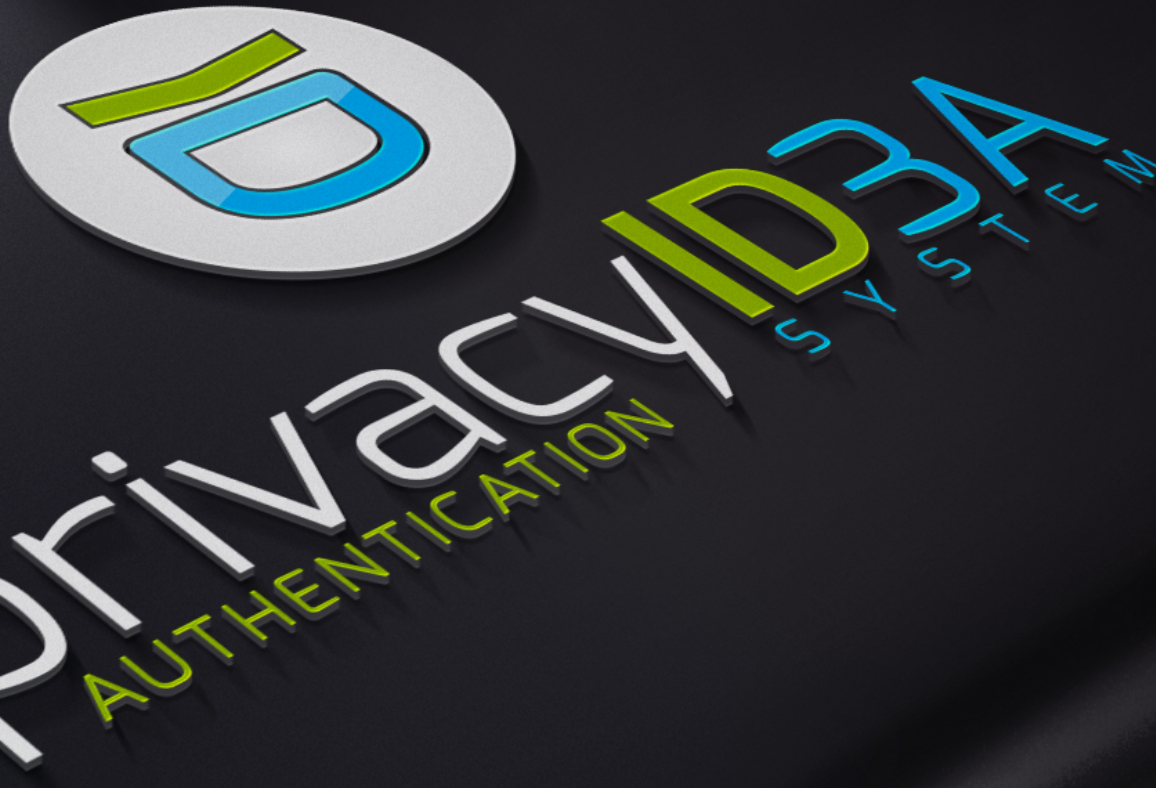
Authentifizierungs-Server I

- Mögliche Kriterien zur Auswahl
  - Cloud-Service vs. In-House Service (Stichworte: Nachvollziehbarkeit, Migrationsmöglichkeiten)
  - Closed Source vs. Open Source (Stichworte: Nachvollziehbarkeit, Modifikationsmöglichkeiten, Migrationsmöglichkeiten)

Authentifizierungs-Server II

- Mögliche Kriterien zur Auswahl
  - Unterstützung der favorisierten Token-Typen
  - Schnittstellen für Management von Tokens und Konfiguration
  - Kosten für Beschaffung, Betrieb, Wartung, Support und Austausch, inkl. Kosten für zusätzlich nachzurüstende Schnittstellen (z. B. eventuell Programmierung gesonderter Authentisierungs-Plug-Ins bei allen anzubindenden Diensten)
  - Passend zur bestehenden IT-Landschaft (Personal, Systeme)

privacyIDEA
Authentication System



- Das aktivste Open Source 2FA Projekt auf Github
  - 20 Contributors
  - 640 Issues
  - 1923 Commits
  - 77 Pull Requests
  - **208 Stars :-(**

privacyIDEA

Ein eigenes 2FA System

- On Premise
- Schlüsselmaterial kann erzeugt werden
- Unterstützte Algorithmen (HOTP, TOTP, mOTP, TiQR/OCRA, RSA)
- Webservice mit gut dokumentierter “REST” API
  - Flask / Python / Token-DB

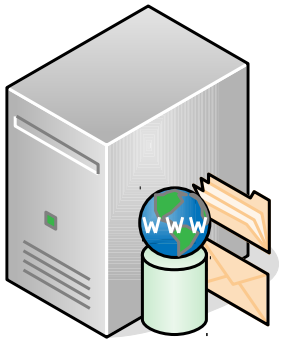
privacyIDEA

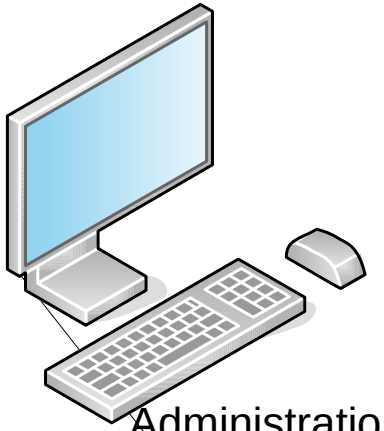
unterstützte Authentifizierungsobjekte

- Key-fob Tokens
- OTP Karten
- SMS, Email, Smartphone
- Yubikey
- U2F
- eToken NG/OTP
- SSH Keys
- x.509-Zertifikate
- ...

Beispiel-Netzwerk
Anbindung / Protokolle

Pre-privacyIDEA

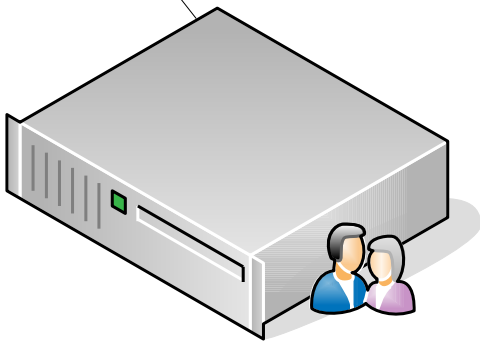




Administration

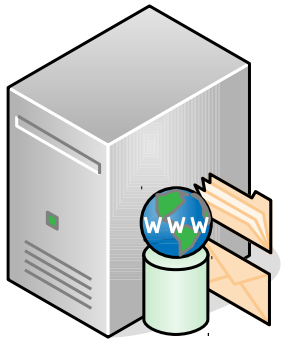
Authentisierung/
Applikation

LDAP



privacyIDEA

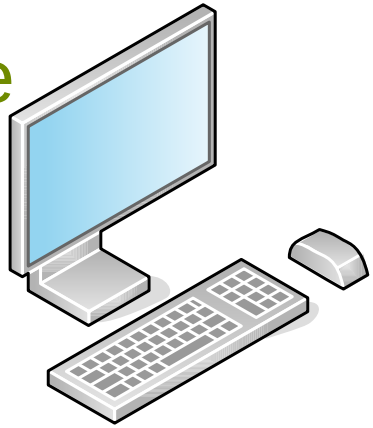
Anbindung / Protokolle



REST API, PAM,
RADIUS, SAML,
Win Cred Prov

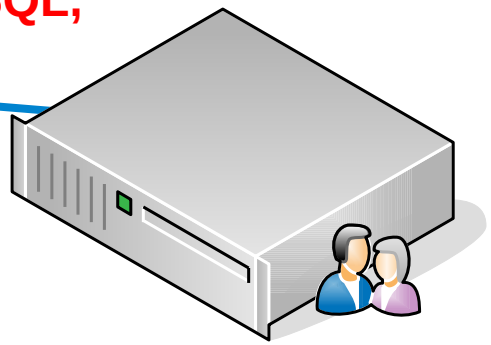
Authentisierung/
Applikation

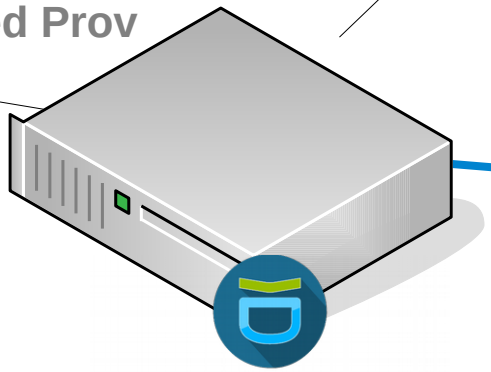
REST API,
Web UI,
CLI,
DB



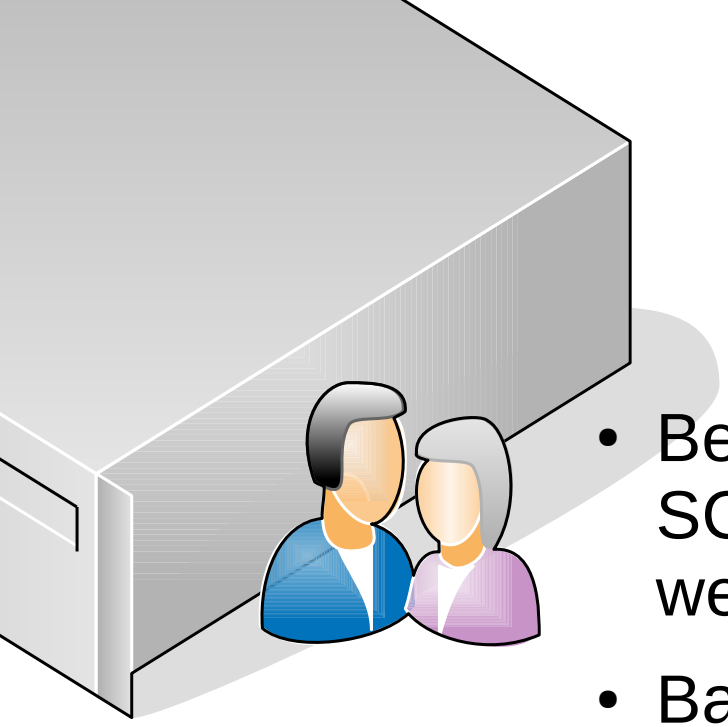
Administration

LDAP, AD, SQL,
File, SCIM





privacyIDEA
AUTHENTICATION SYSTEM

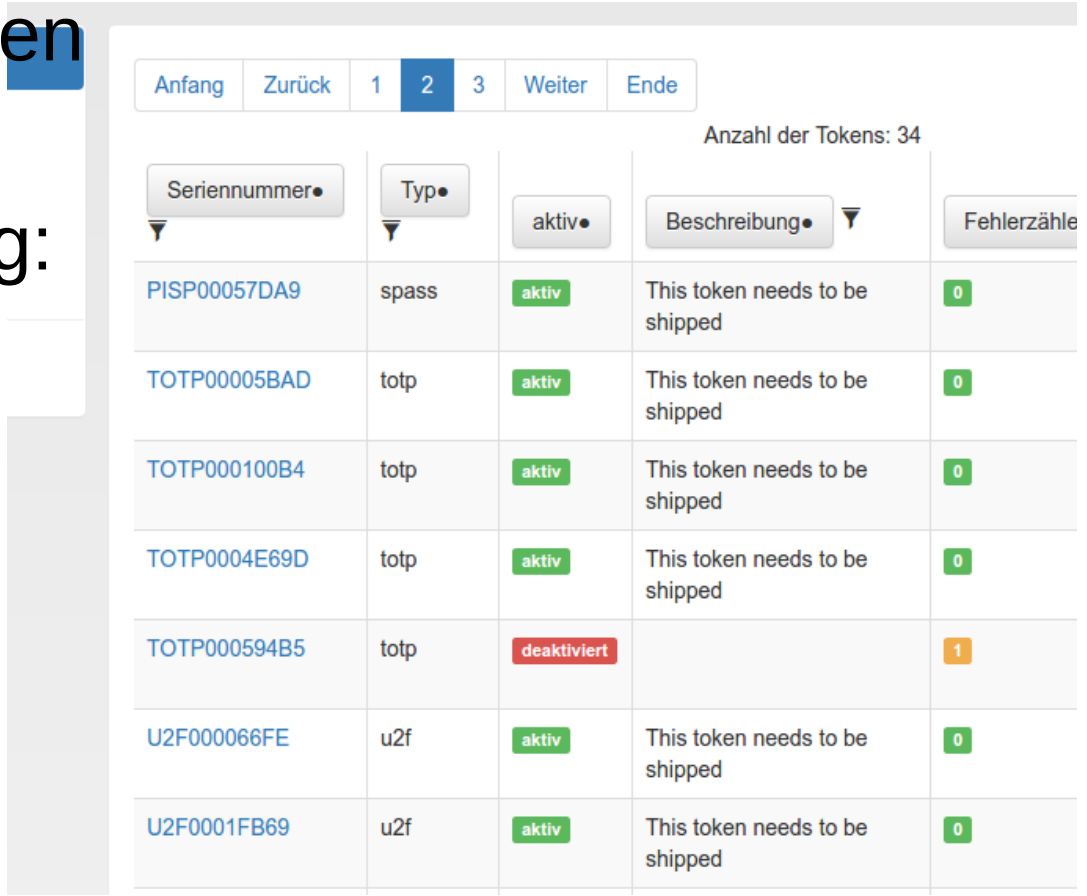


privacyIDEA
Benutzer

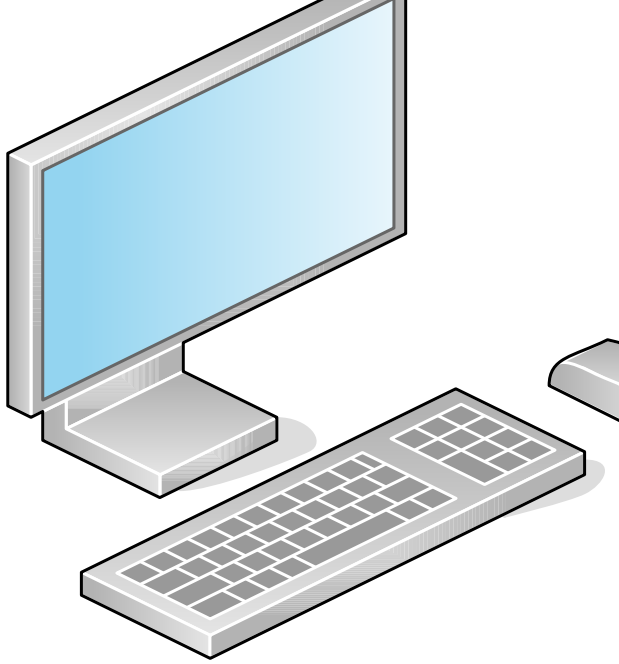
- Benutzerquellen LDAP, SQL, Flatfile, SCIM können beliebig kombiniert werden.
- Basierend auf diesen Benutzerquellen können unterschiedliche Richtlinien umgesetzt werden.
- Das Verhalten für Helpdesk, Administratoren, Studenten, Verwaltungsmitarbeiter… kann unterschiedlich definiert werden.

privacyIDEA
Tokentypen

- Unterschiedliche Tokentypen können gleichzeitig betrieben werden.
- Richtlinienabhängig:
  - Authentifizierung kann an Token-Typen gebunden werden.



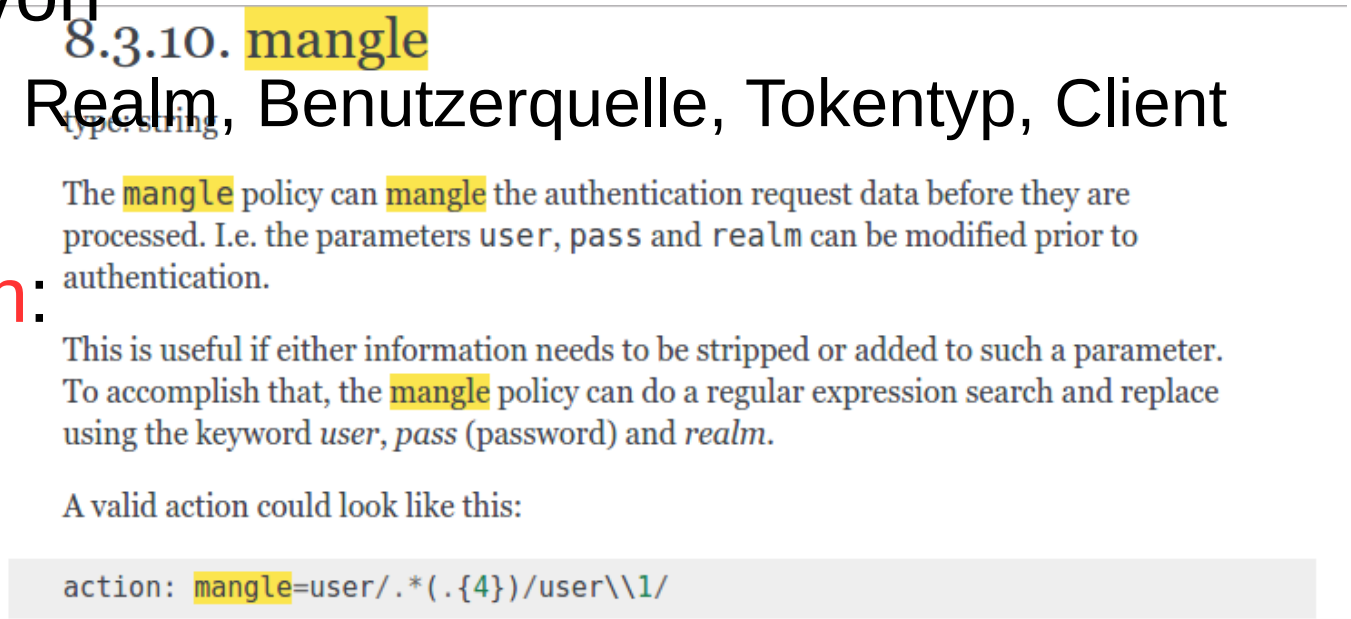
privacyIDEA
API



- Alles ist eine API
  - `POST /validate/check`
  - `POST /token/init`
  - `GET /token/`
  - `DELETE /token/OATH12344`
- Flask, JWT
  - `POST /auth` liefert einen JWT entweder als Rolle **user** oder Rolle **admin**.
- http://privacyidea.readthedocs.io

privacyIDEA
Richtlinien

- Richtlinien sind Python Decorator und ändern das Verhalten von privacyIDEA
  - Scopes: admin, user, enrollment, auth, authz, webui
- Abhängig von
  - Benutzer, Realm, Benutzerquelle, Tokentyp, Client IP, Zeit
- Abgefahren:



8.3.10. mangle

type: string

The `mangle` policy can mangle the authentication request data before they are processed. I.e. the parameters `user`, `pass` and `realm` can be modified prior to authentication.

This is useful if either information needs to be stripped or added to such a parameter. To accomplish that, the mangle policy can do a regular expression search and replace using the keyword *user*, *pass* (password) and *realm*.

A valid action could look like this:

```
action: mangle=user/.*(.{4})/user\\1/
```

privacyIDEA
Eventhandler

- Ändert nicht das Verhalten sondern startet neue Aktionen abhängig von eintretenden Ereignissen:
  - UserNotification Handler
  - Token Handler
  - Script Handler

API-event → zusätzliche Aktion (Email, Token deakieren, Skript)

privacyIDEA
Eventhandler

Aktiv	ID	Beschreibung	Ereignisse	Handlermodul	Bedingungen	Aktion	Op
✔	1	User can only enroll disabled tokens	["token_init"]	Token	{"logged_in_user":"user"}	disable	{}
✔	2	help desk enrollment	["token_init", "token_assign"]	Token	{"logged_in_user":"admin"}	set description	{ "de "Th nee shi
✔	3	delete token	["validate_check"]	Token	{"result_value":"False","tokentype":"hotp"}	delete	{}

privacyIDEA
Automatisierung

- Dokumentiertes Datenbank Model
- pi-manage
- REST API
- Event Handler

Table Of Contents

Related Topics

Documentation overview
- **Previous:** 13. Application Plugins
- **Next:** 14.1.1. REST API

privacyIDEA
Get Involved

- https://github.com/privacyidea/privacyidea
- https://pootle.privacyidea.org

Konkrete Umsetzung an der Philipps-Universität I

- Die Zwei-Faktor-Authentisierung an der Philipps-Universität erfolgt über den Einsatz von *Einmalpasswörten*.

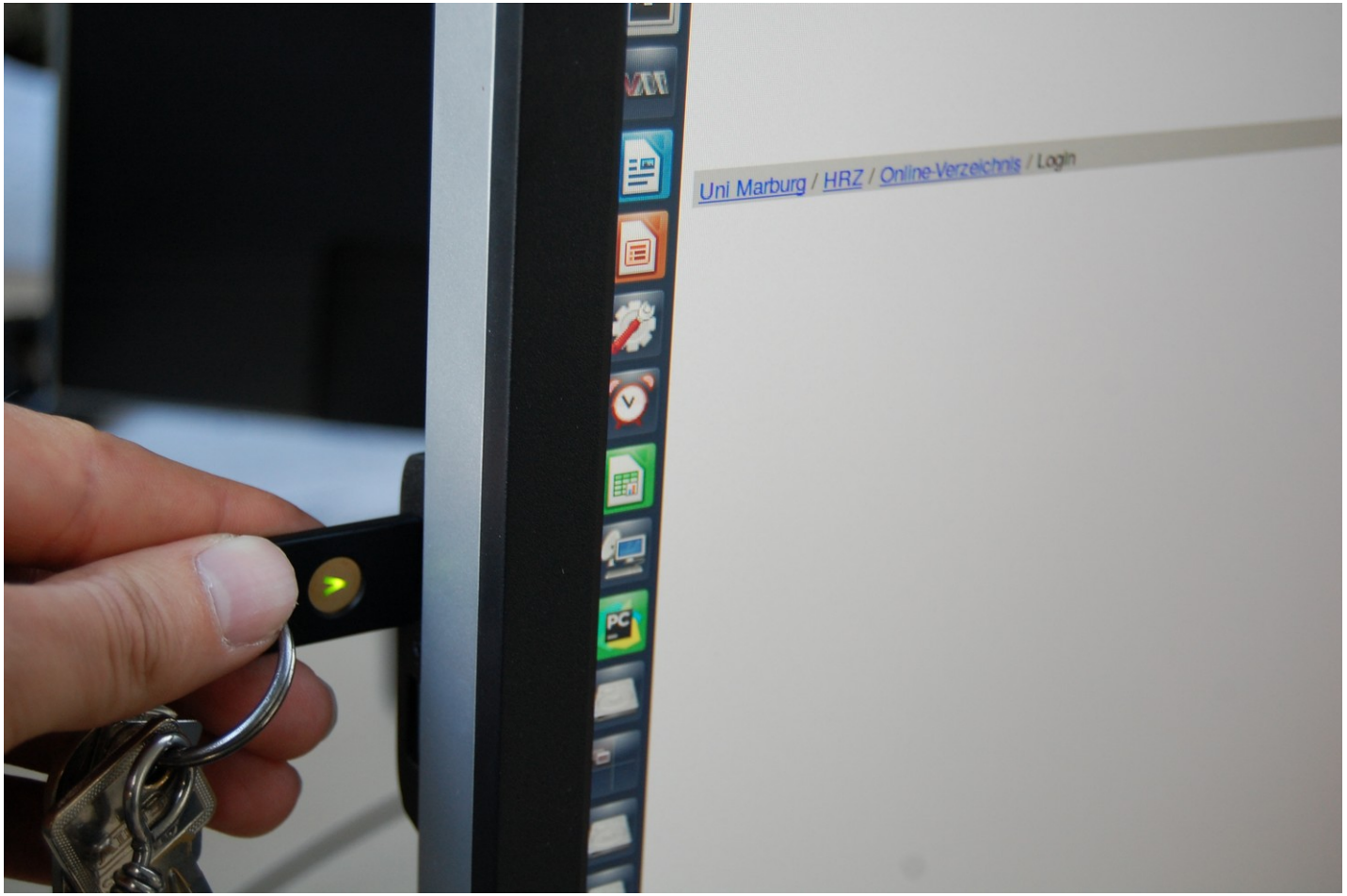
- Warum *Einmalpasswörte*?
  - Kompatibilität mit allen bestehenden Web-Diensten, die Passworte benutzen, bisher über LDAP authentisieren und damit über einen LDAP-2FA-Proxy-Server angebunden werden können
  - Geringe Hardware- und Software-Anforderungen auf Seiten der Benutzer
  - Das Verfahren von Einmalpassworten ist Benutzern i. d. R. bereits bekannt (z. B. TAN-Liste/Einmalpasswort-Liste und Mobile-TAN für das Bank-Konto, Registrierungs-Code)

Konkrete Umsetzung an der Philipps-Universität II

- Tokens für Mitarbeiter
  - YubiKey USB-Hardware-Tokens per Post (Standard)
  - Software-Tokens für das Smartphone (Optional) via Web-Portal
  - Einmalpasswort-Listen (Optional) via Web-Portal

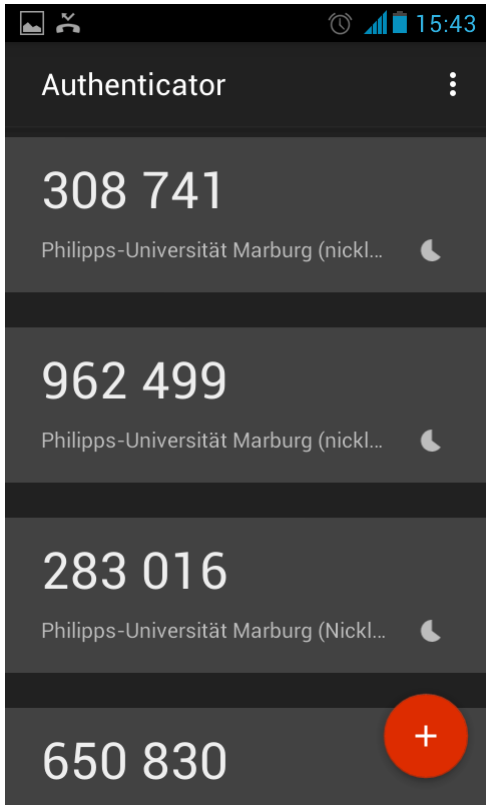
- Tokens für Studenten
  - Registrierungs-Token per Post für den einmaligen Zugang zum Web-Portal (Standard, jedes Semester neu)
  - Software-Tokens für das Smartphone (Standard) via Web-Portal
  - Einmalpasswort-Listen (Optional) via Web-Portal

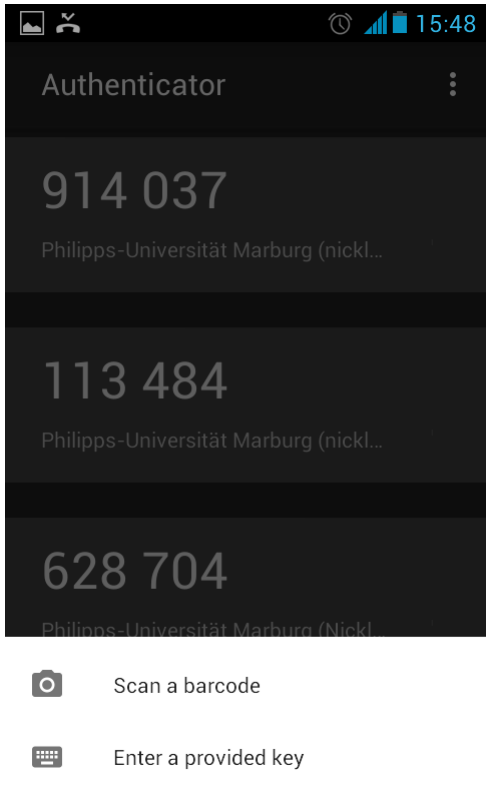
Konkrete Umsetzung an der Philipps-Universität III

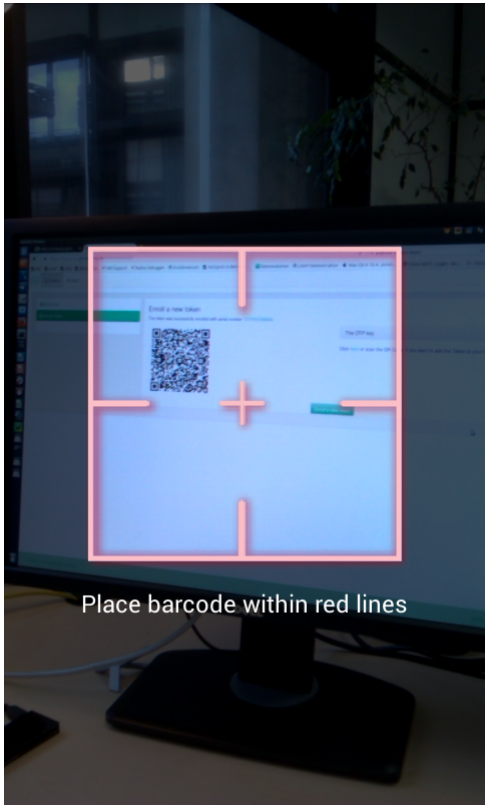


Einstecken des YubiKey-Tokens in eines freien USB-Anschluss des Arbeitsplatz-PCs.

Konkrete Umsetzung an der Philipps-Universität IV







Ausrollen eines App-Tokens für die Authenticator-App.
Alle gezeigten Tokens sind Test-Tokens.

Konkrete Umsetzung an der Philipps-Universität V

- Authentifizierungs-Server: privacyIDEA
  - Open Source Software
  - Große Anzahl unterstützter Token-Typen, inkl. YubiKey-Token, Paper-Token, OATH-TOTP- und OATH-HOTP-Token
  - Python API / REST-API / WebUI für Self-Service und Administation
  - Authentifizierungs-Policies
  - Event-Framework zur Automatisierung

Konkrete Umsetzung an der Philipps-Universität VI

- Server-Komponenten
  - Web-Portal für Self-Service
  - Web-Portal für Administration
  - Mehrere privacyIDEA-Worker
  - PostgreSQL-Datenbank-Server

Konkrete Umsetzung an der Philipps-Universität VII

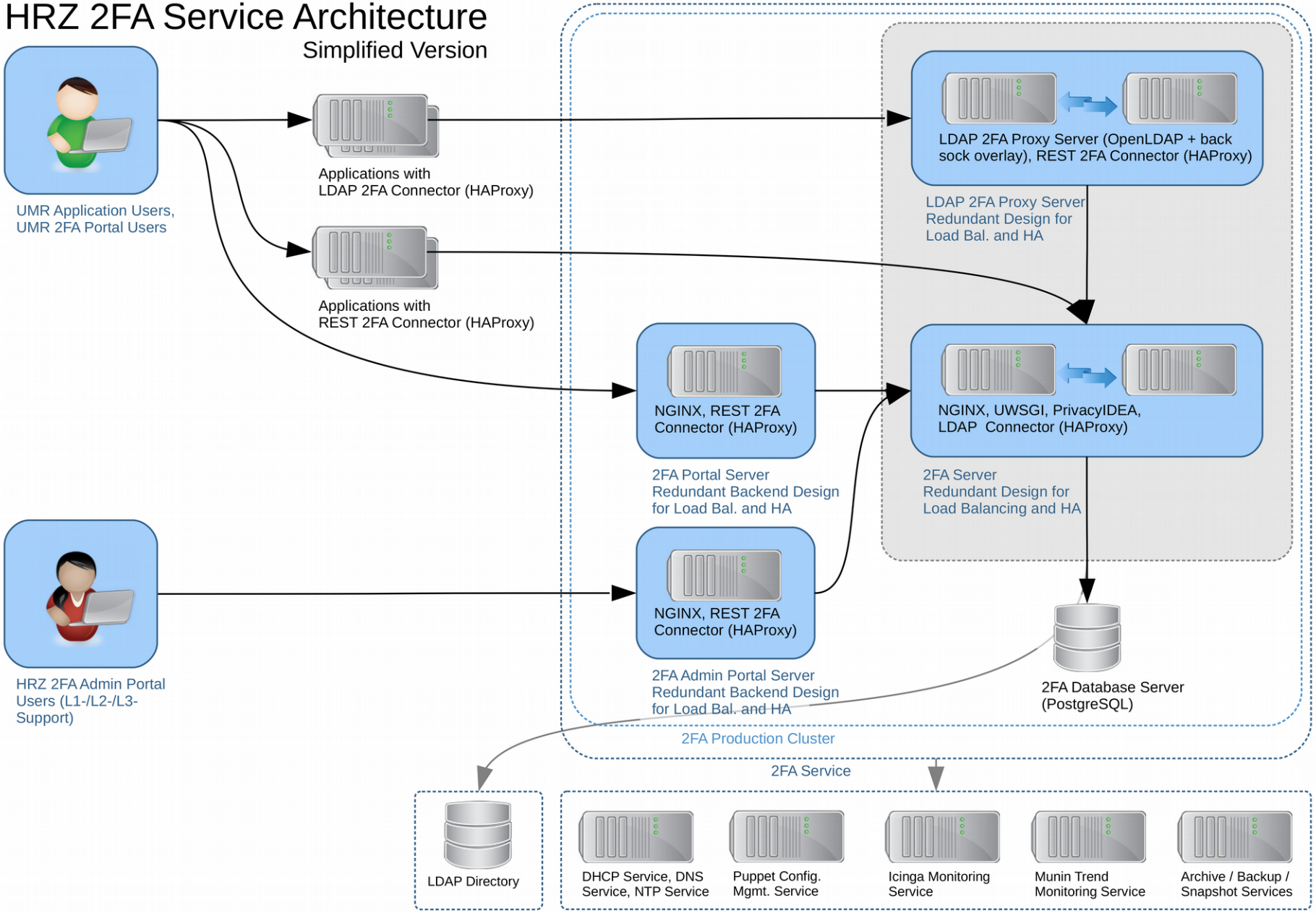
- Server-Komponenten
  - LDAP-2FA-Proxy-Server
  - LDAP-Server der Benutzerverwaltung

Konkrete Umsetzung an der Philipps-Universität VIII

- Client-Komponenten
  - YubiKey-Initialisation-Client
    - Mini-PC mit USB- und Ethernet-Schnittstelle
    - SSH-Zugang mittels Public-Key und 2FA
    - Initialisieren der YubiKeys über den privacyIDEA-Command-Line-Client
    - Verschlüsselte-Registrierung der YubiKeys im privacyIDEA-Server
  - YubiKey-Rollout-Client
    - Vorreservierung freier Tokens an Benutzer mittels python-privacyidea
    - Erstellen von Token-Anschreiben mittels python-privacyidea, python-ldap3, python-pystache, Latex und offiziellen Latex-Brief-Vorlagen

HRZ 2FA Service Architecture

Simplified Version



Zwei-Faktor-authentifizierte Dienste

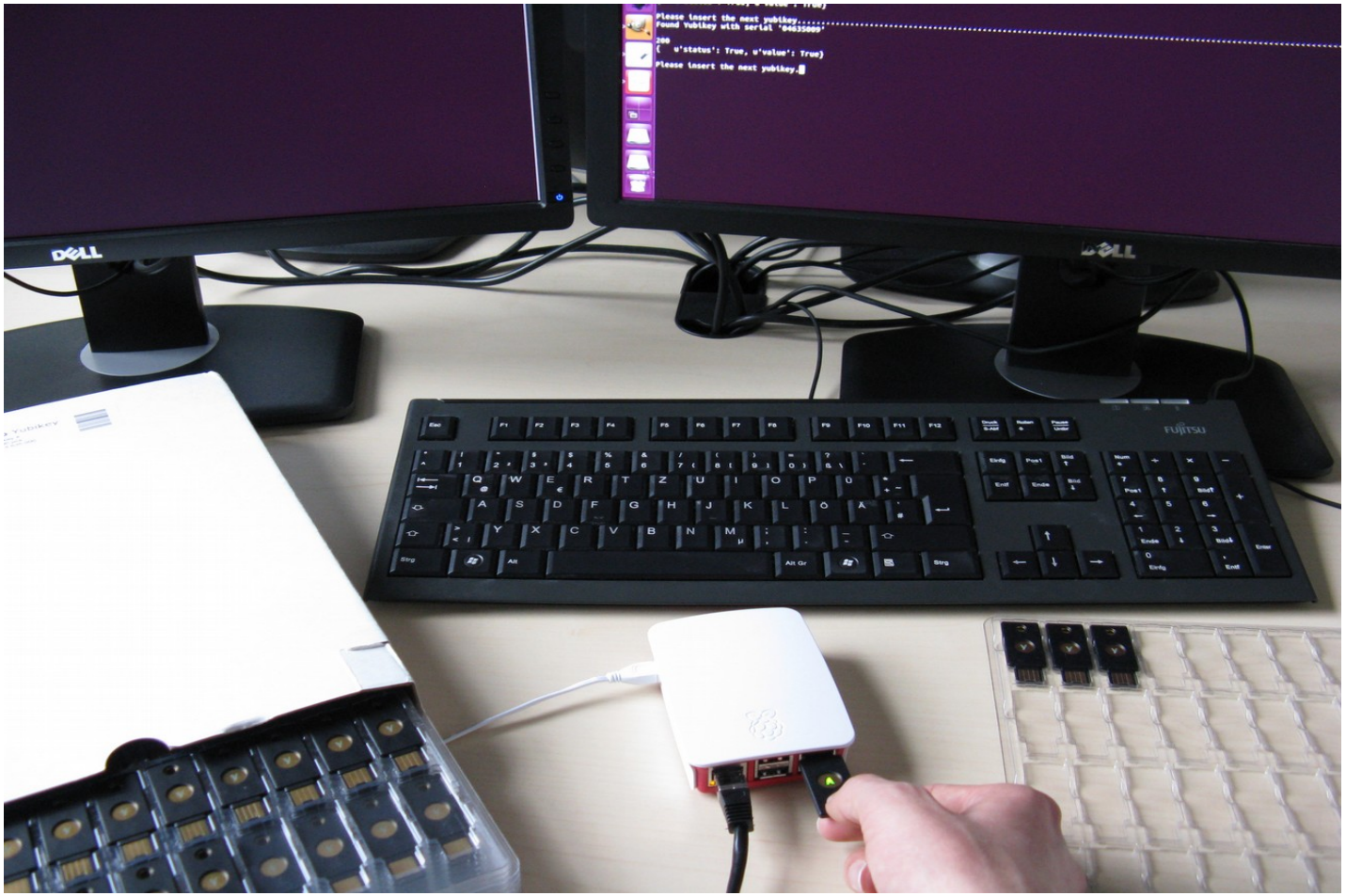
- In Betrieb
  - LDAP-Web-Portal (aktuell ~ 260 Nutzer)
  - Diverse HRZ-interne Dienste: GitLab, Jenkins, VPN für HRZ-Mitarbeiter, i-doit Datenbank für technische und organisatorische Dokumentation, SSH-Login auf YubiKey-Initialisation-Clients

- In Planung / Realisierung
  - HISinOne Campus Management System (in Marburg: *Marvin*)
    - Transaktionsbasierte Einmalpassworte für Mitarbeiter (Noteneintragung) und Studierende (Prüfungsanmeldung)
    - ~1500 Mitarbeiter („Noteneintragungsberechtigte“)
    - ~30.000 Studierende

Zwei-Faktor-authentifizierte Dienste

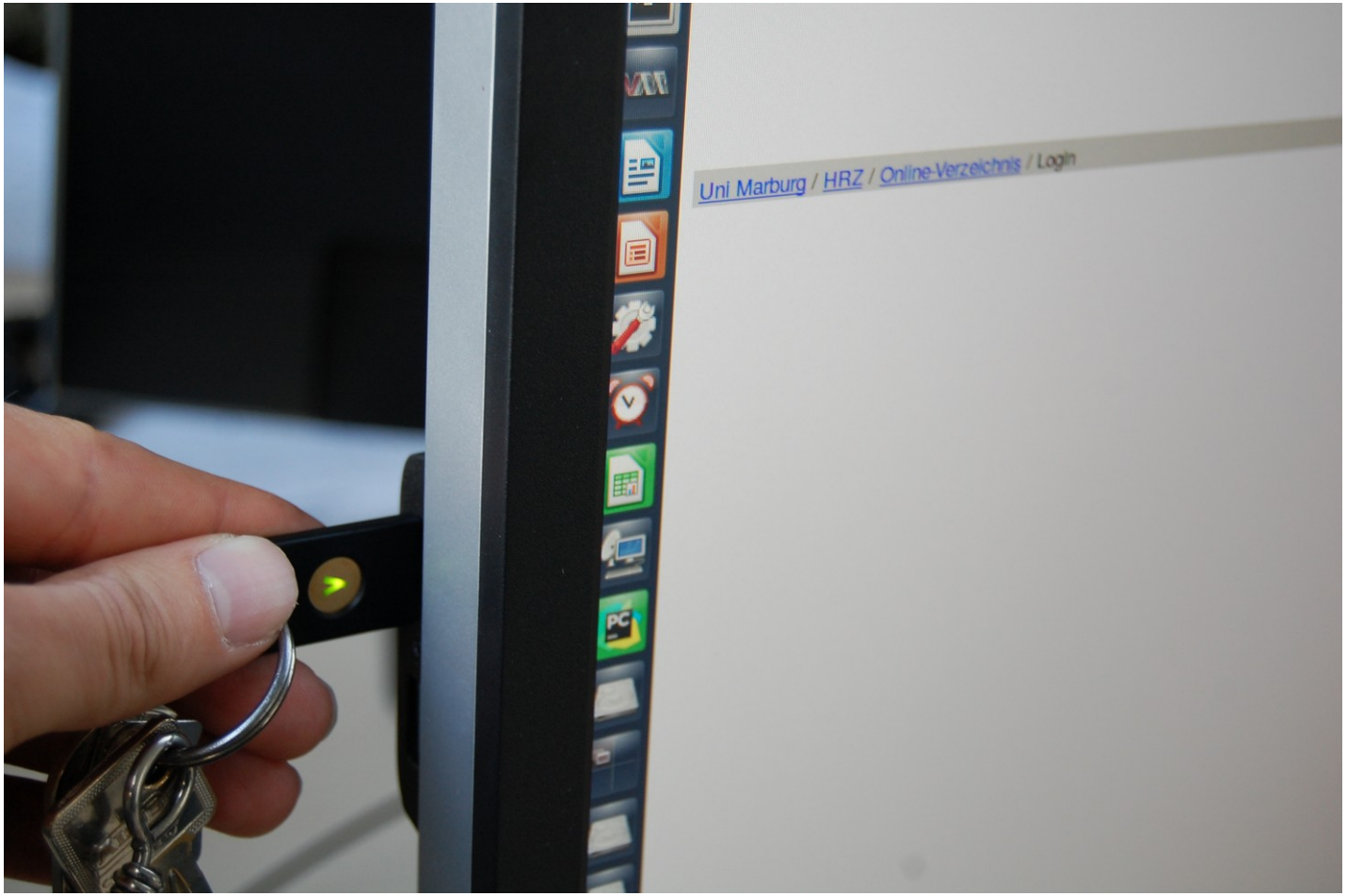
- Weitere mögliche Kandidaten
  - SSH
    - Sinnvoll für persönliche Admin-Accounts und einzelne Logins (Public Key + Benutzerpasswort + Einmalpasswort)
    - Nicht sinnvoll für multiple oder automatisierte Logins
    - Mögliche Lösung: 2FA nur für Hop-Hosts, die anschließend den exklusiven Zugriff auf weitere Server ohne 2FA ermöglichen
  - Shibboleth Single-Sign-On-Dienst
  - Content Management System, ILIAS E-Learning-Portal
  - Windows-Login / AD-Login
    - Alles was über die Modifikation des lokalen Login-Dialogs hinausgeht ist *eine große Herausforderung*

Rollout von YubiKey-Tokens an Mitarbeiter



Einstecken eines YubiKey-Tokens in einen Initialisation-Client.

Rollout von YubiKey-Tokens an Mitarbeiter



Einstecken des YubiKey-Tokens in eines freien USB-Anschluss des Arbeitsplatz-PCs.

Rollout von YubiKey-Tokens an Mitarbeiter

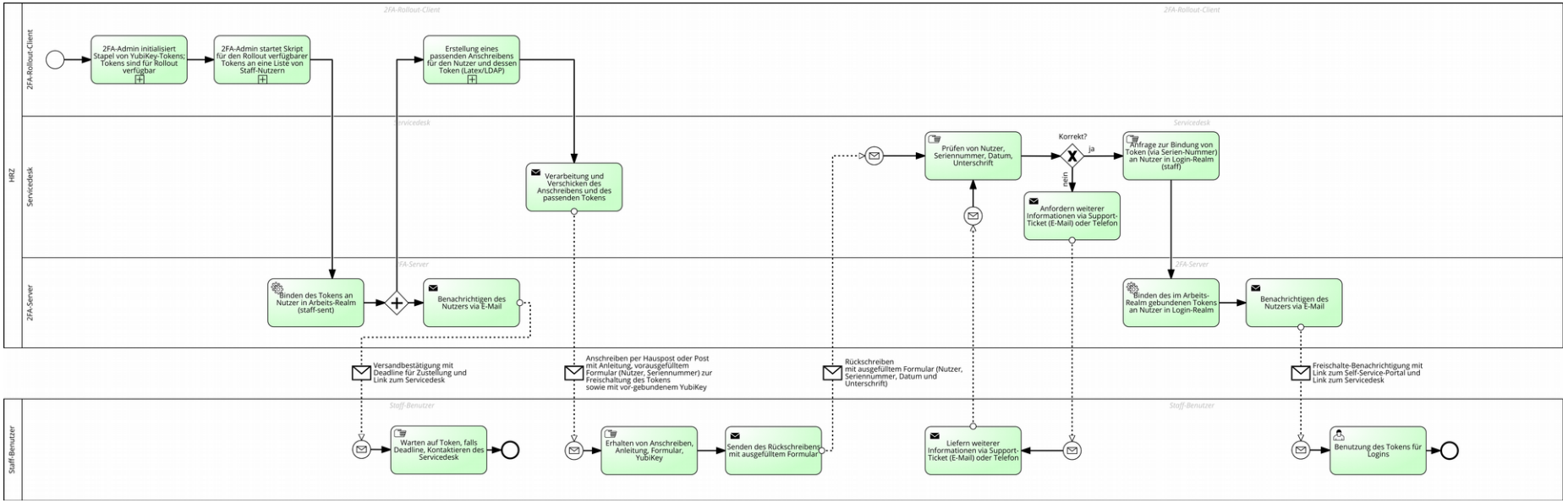
- Was ein Mitarbeiter von uns benötigt
  - Token
  - Informationen
    - Was ist 2FA und warum brauche ich bald ein Token?
    - Für welche Dienste wird 2FA zukünftig benötigt?
    - Wann erfolgt die verpflichtende Umstellung?
    - Was sind die weiteren Schritte zum Erhalt des Tokens?
    - Wann wurde das Token versendet? Wann sollte es ankommen?
    - Was sind die weiteren Schritte zur Freischaltung des Tokens?
    - Wann wurde das Token freigeschaltet, d.h. ab wann *muss* es verwendet werden?

Rollout von YubiKey-Tokens an Mitarbeiter

- Was wir von einem Mitarbeiter benötigen
  - Rückschreiben (Schriftform)
    - Freischalte-Antrag mit Bestätigung des Erhalts des Tokens
    - Zustimmung zu den Nutzungsbedingungen
  - Ob Schriftform benötigt wird, ist durchaus diskutabel.

Rollout von YubiKey-Tokens an Mitarbeiter

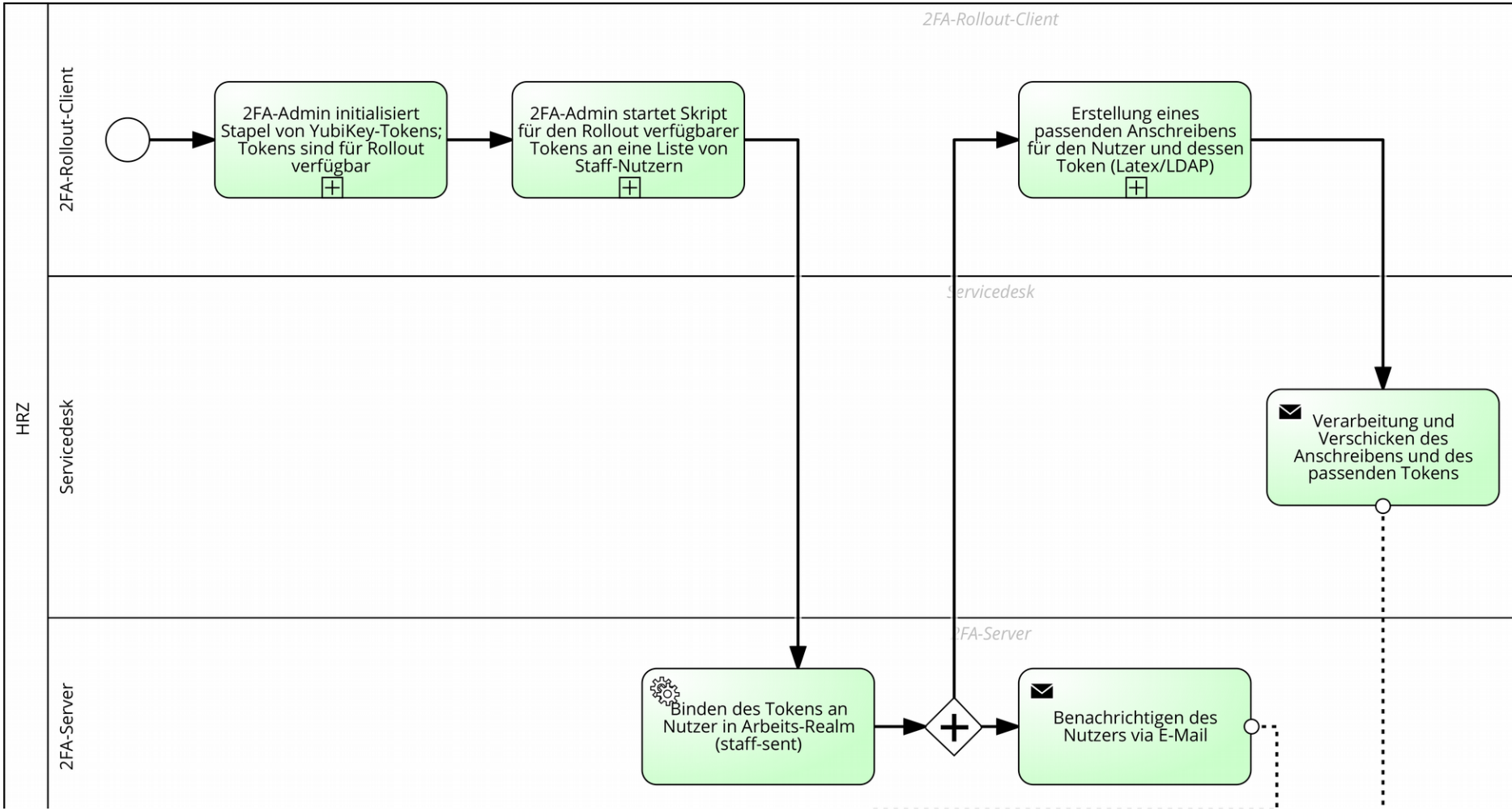
Gesamtprozess Rollout



Oh, das kann ja keiner erkennen… Einen Moment bitte...

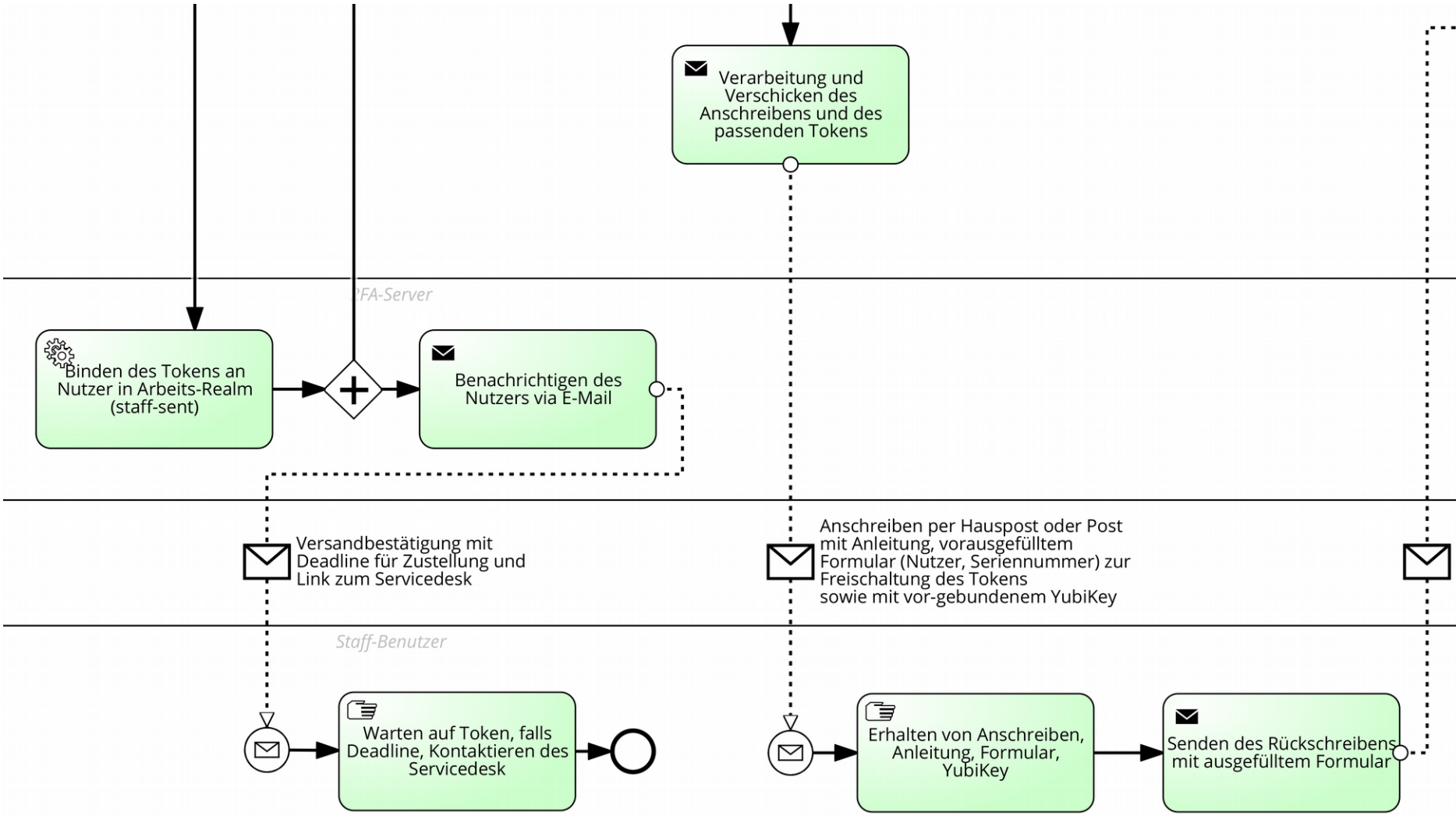
Rollout von YubiKey-Tokens an Mitarbeiter

Reservieren des Tokens, Versenden des Anschreibens, Versandbestätigung



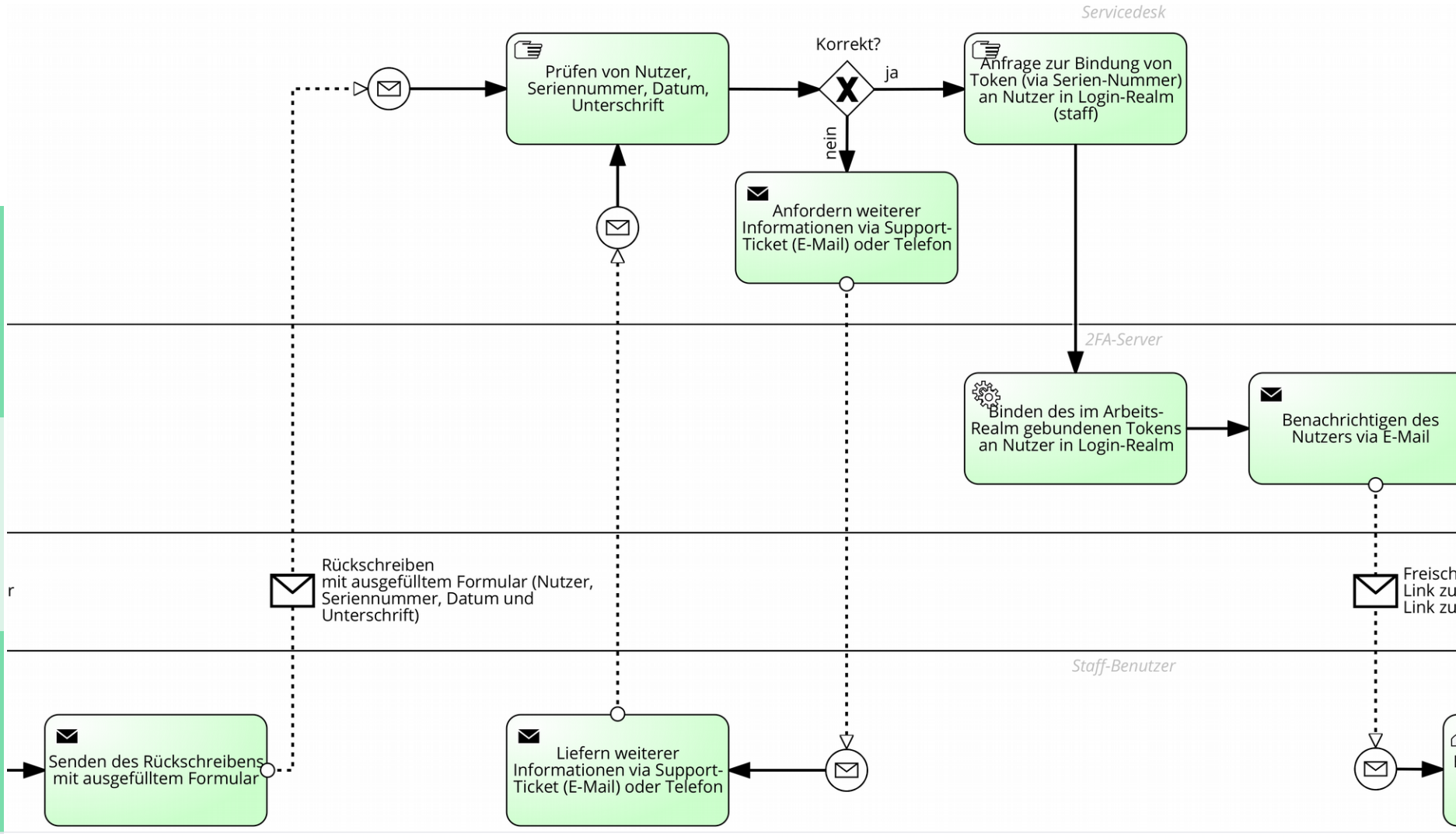
Rollout von YubiKey-Tokens an Mitarbeiter

Versandbestätigung, Versenden des Anschreibens, Erhalten des Tokens



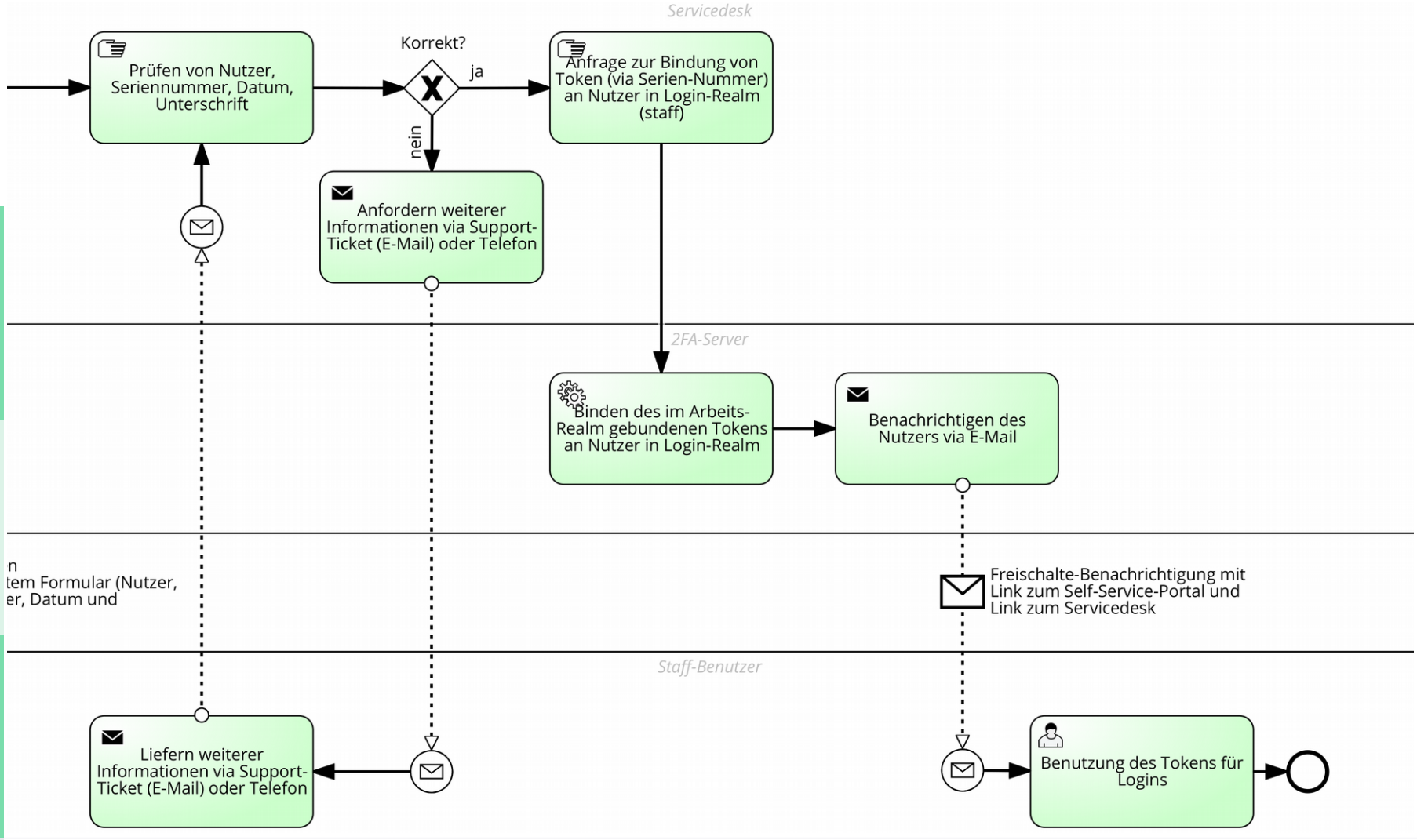
Rollout von YubiKey-Tokens an Mitarbeiter

Versenden des Rückschreibens, Prüfen des Rückschreibens, Freischaltung



Rollout von YubiKey-Tokens an Mitarbeiter

Prüfen des Rückschreibens, Freischalten des Tokens, Benachrichtigung



Rollout von YubiKey-Tokens an Mitarbeiter

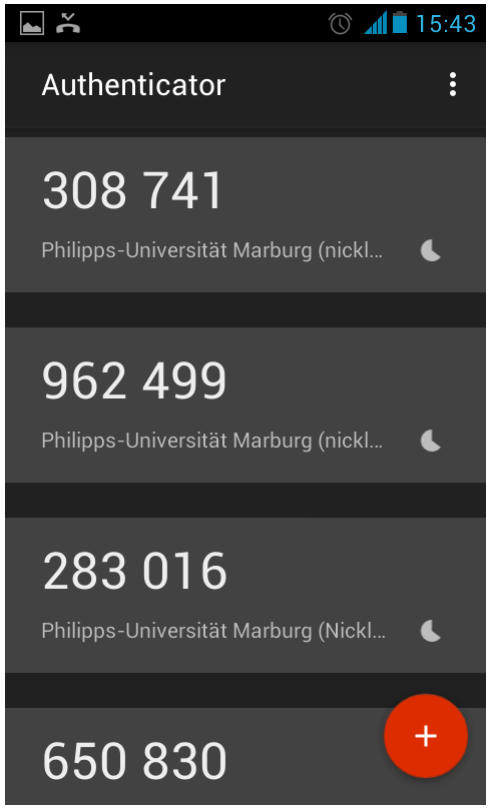
- PrivacyIDEA-WebUI-Beispiel
  - Automatischer E-Mail-Versand bei Versand eines YubiKey (bei Binden in realm staff-sent)
  - Automatischer E-Mail-Versand bei Freischaltung eines YubiKey (bei Binden in realm staff)

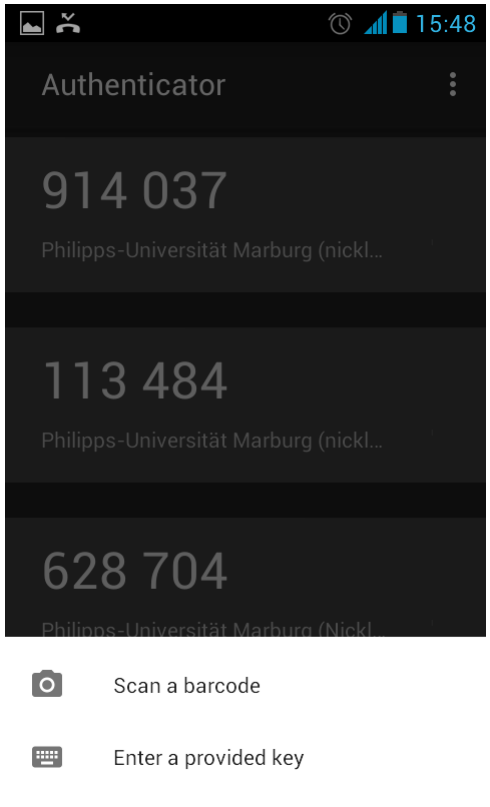
- Das alles geht natürlich auch direkt über die REST-API!

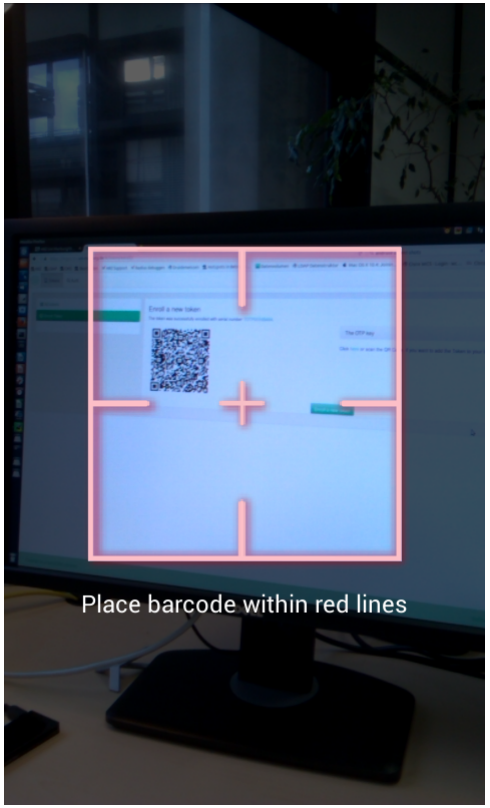
Rollout von App-Tokens an Studierende

Erhalt und Eingabe des Registrierungscodes zum erstmaligen Ausrollen eines App-Tokens im 2FA-Portal (Self-Service-Portal)

Rollout von App-Tokens an Studierende







Ausrollen eines App-Tokens für die Authenticator-App.
Alle gezeigten Tokens sind Test-Tokens.

Rollout von App-Tokens an Studierende

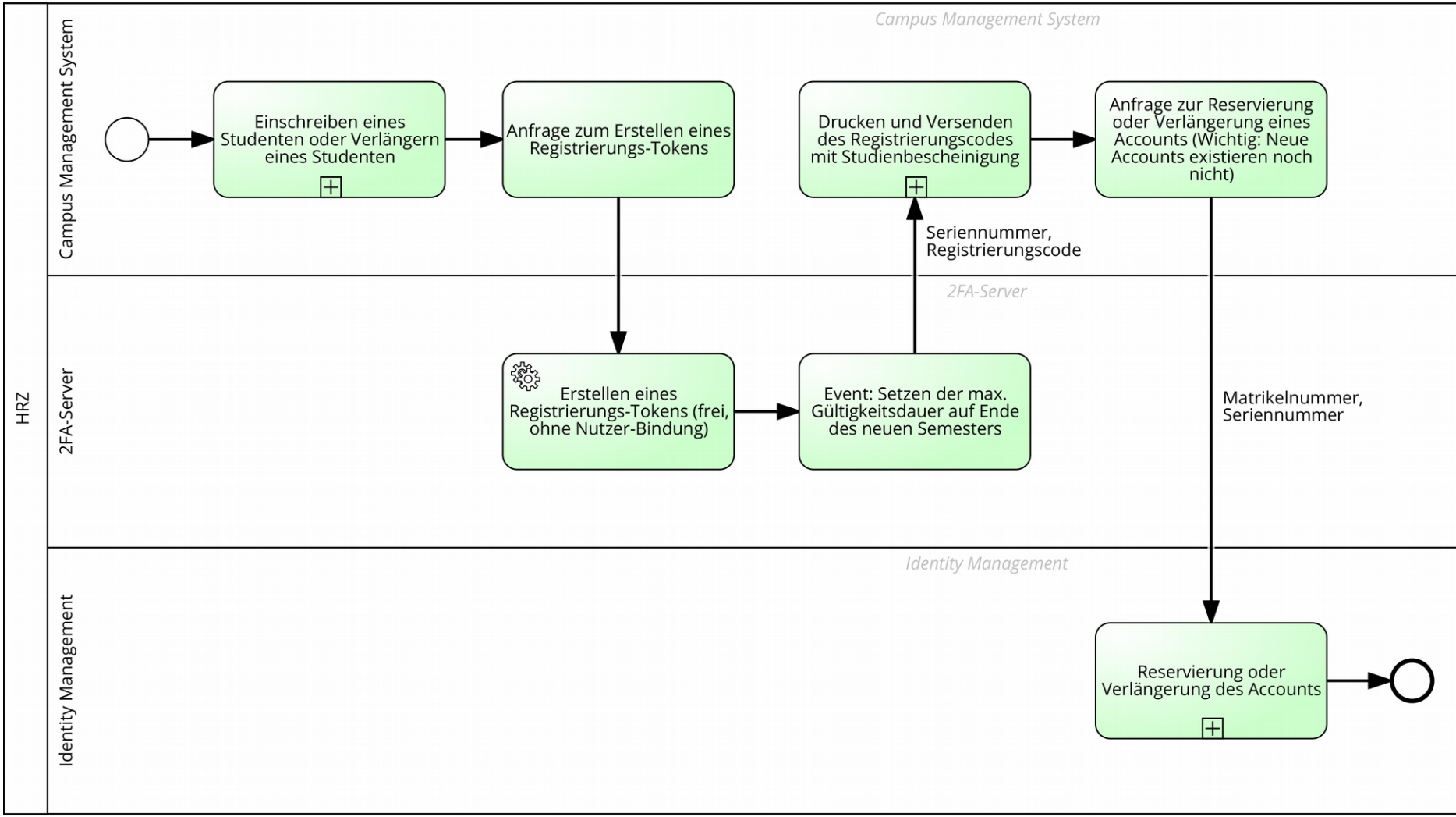
- Was wir von einem Studierenden benötigen
  - Ein initiales 2FA-Token zur Freischaltung weiterer App-Tokens
  - Zustimmung zu den Nutzungsbedingungen

Rollout von App-Tokens an Studierende

- Was ein Studierender von uns benötigt
  - Ein initiales Einmalpasswort-Token zur Freischaltung weiterer App-Tokens (z. B. mit der Studienbescheinigung ein zusätzlicher Registrierungscode per Post)
  - Informationen
    - Warum?
    - Wie logge ich mich im 2FA-Portal ein?
    - Wie erstelle ich mir ein App-Token?

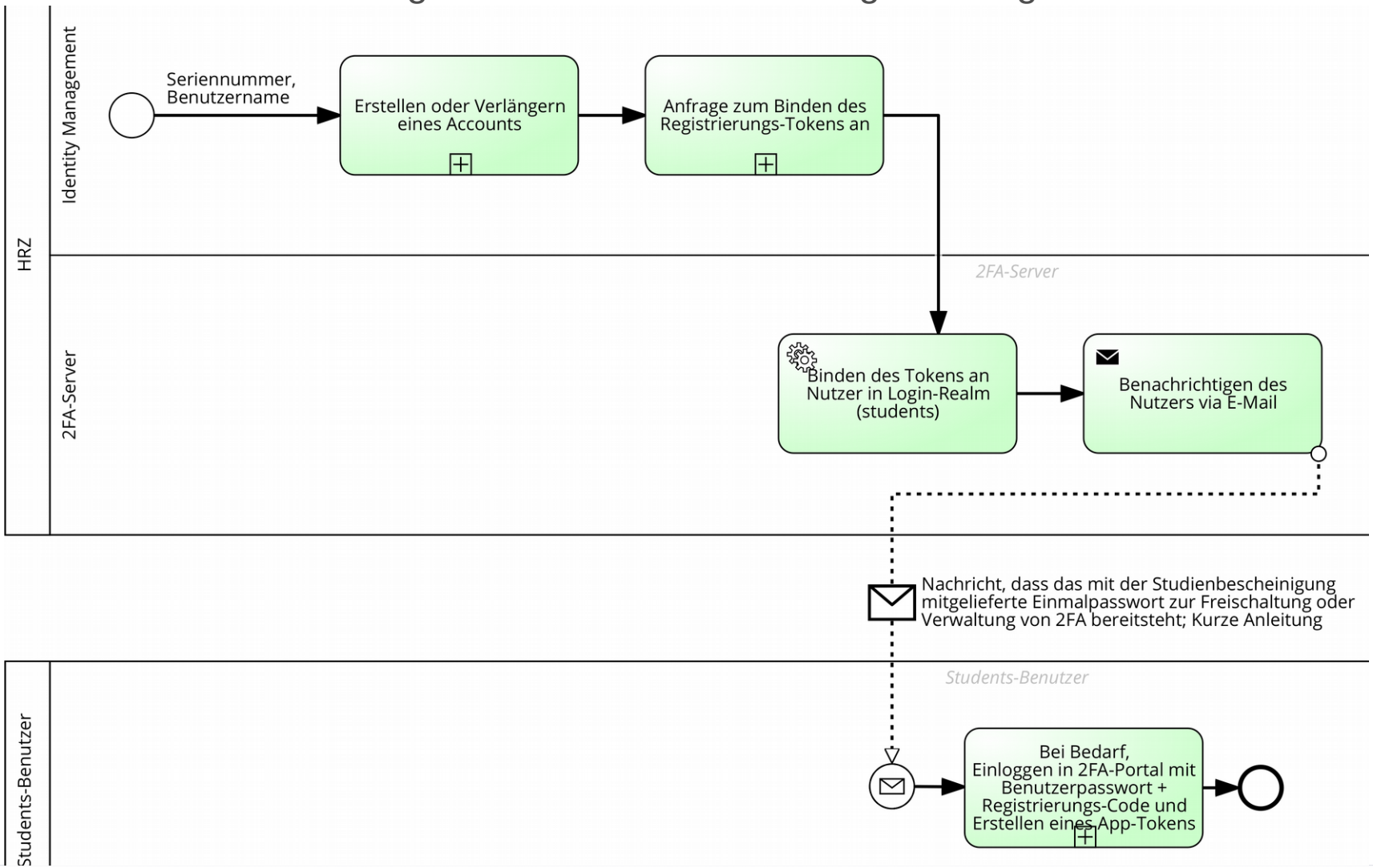
Rollout von App-Tokens an Studierende

Einschreibung/Verlängerung → Erstellung Registrierungs-Token (ungebunden)



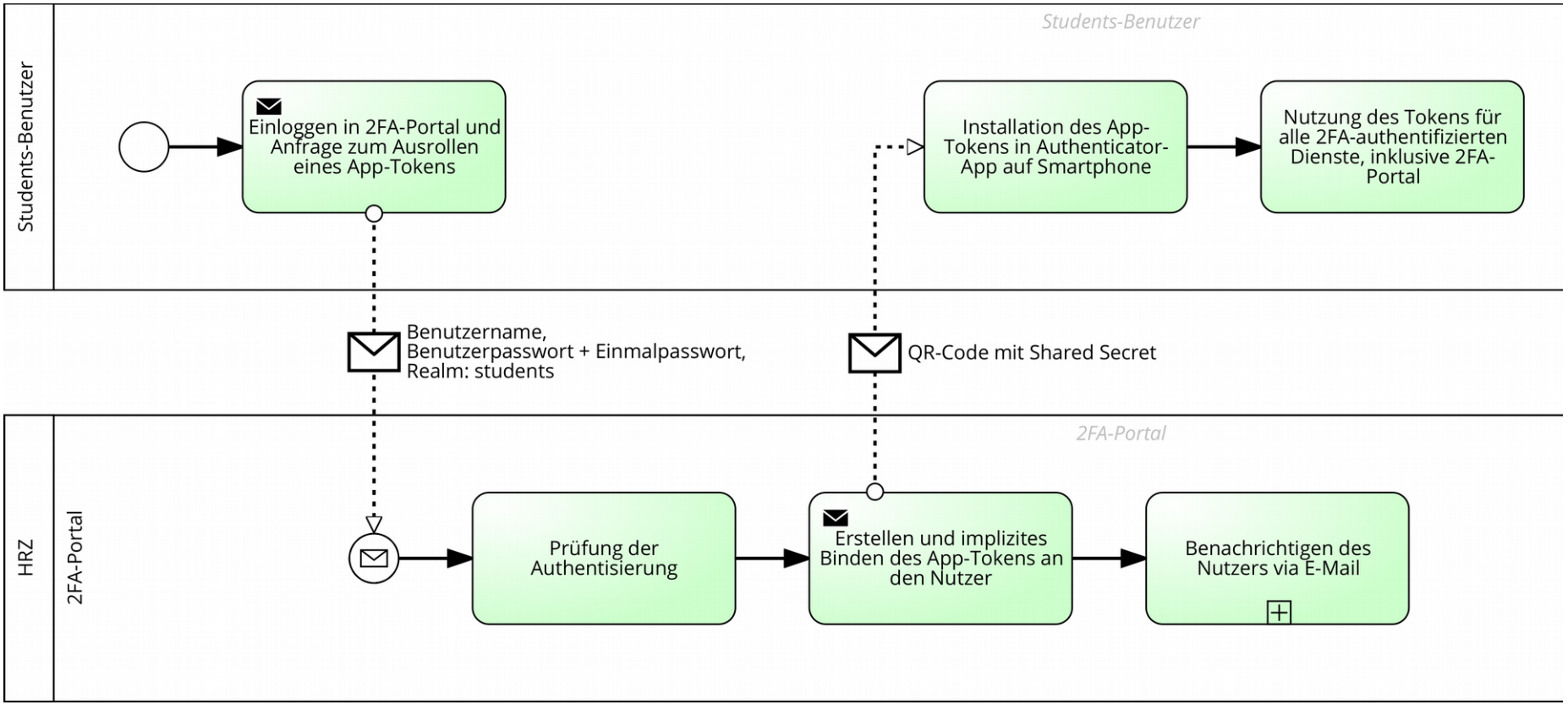
Rollout von App-Tokens an Studierende

Erstellen oder Verlängern Account → Binden Registrierungs-Token an Nutzer



Rollout von App-Tokens an Studierende

Login in 2FA-Portal via Registrierungscode → Ausrollen eines App-Tokens



HISinOne Campus Management System

- Closed Source, Quellcode jedoch für Lizenznehmer einsehbar und modifizierbar
- Zusammenarbeit mit Hersteller, um die eingebaute TAN-Funktionalität für bestimmte kritische Aktionen wie Noteneintragung, Prüfungsanmeldung und Adressänderung um die Nutzung der privacyIDEA-REST-API zu erweitern.

Tipps

- Sich im klaren sein, dass ein sehr sicheres System für *ganz gewöhnliche Benutzer* sehr wenig benutzbar ist.
- Einen sinnvollen Kompromiss zwischen Sicherheit und Benutzerfreundlichkeit finden.
- Prozesse mit allen relevanten Abteilungen und Akteuren abstimmen.
- Den Nutzern nicht alle Informationen auf Einmal liefern, sondern immer nur die für jeden Teilprozess des Rollouts notwendigen Informationen.

Tipps

- Ausroll-Prozess mit gehörigem Vorlauf an ernstzunehmender Anzahl *ganz gewöhnlicher Nutzer* testen. Mögliche Probleme können z. B. sein:
  - Anschreiben für (1) die Ankündigung des Rollouts, (2) den Erhalt des Tokens und den Freischalte-Prozess sowie (3) die Benutzung des Tokens werden von Nutzern nicht oder falsch verstanden.
  - Es werden gegen alle Erwartungen Einwände gegen die Nutzungsbedingungen laut, da diese möglicherweise unerwünscht in Arbeitsalltag und Arbeitsrecht eingreifen, und es ist eine entsprechende Klärung mit der Rechts- und Personalabteilung sowie der Personalvertretung notwendig.

Vielen Dank für die Aufmerksamkeit

- Weitere Informationen
  - https://www.privacyidea.org
  - https://www.uni-marburg.de/2fa

Fragen

Anhang

- Technische Details zum LDAP-2FA-Proxy-Server
  - OpenLDAP (Consumer, angebunden an zentralen Provider)
  - OpenLDAP Back-Sock-Overlay für alle LDAP-Bind-Requests
  - Python Socket Listener mit Anbindung zu den privacyIDEA-Workern), ermöglicht die einfach 2FA-Anbindung aller bisher über LDAP authentisierender Dienste
  - Eingehende Bind-Requests
    - Anwendungs-generische Anonymous- und Proxy-Benutzer authentisieren wie gewohnt direkt gegen LDAP (kein 2FA)
    - Staff- und Students-Benutzer authentisieren gegen privacyIDEA (Je nach hinterlegten Policies mit 2FA für alle verpflichtend oder mit passthru für Nutzer, die noch kein Token besitzen)