

Mehr-Faktor-Authentifizierung

Open Source MFA mit privacyIDEA

Cornelius Kölbel
@cornelinux

cornelius.koelbel@netknights.it
<https://privacyidea.org>

tübix
11. Juni 2016

Identität – Authentifizierung Alltag

- **Zwischenmenschlich**
 - **Gesicht**
 - **Stimme**
 - **Verhalten**



Identität – Authentifizierung Auto



- Jedem Deutschen sein Auto
 - Schlüssel

Identität – Authentifizierung

Schöne Neue Welt?



Identität – Authentifizierung Computer

- Benutzerpasswort (123456)



Begriffsklärung

- Authentisierung vs. Authentifizierung
- Verschiedene Authentisierungsarten



Wissen



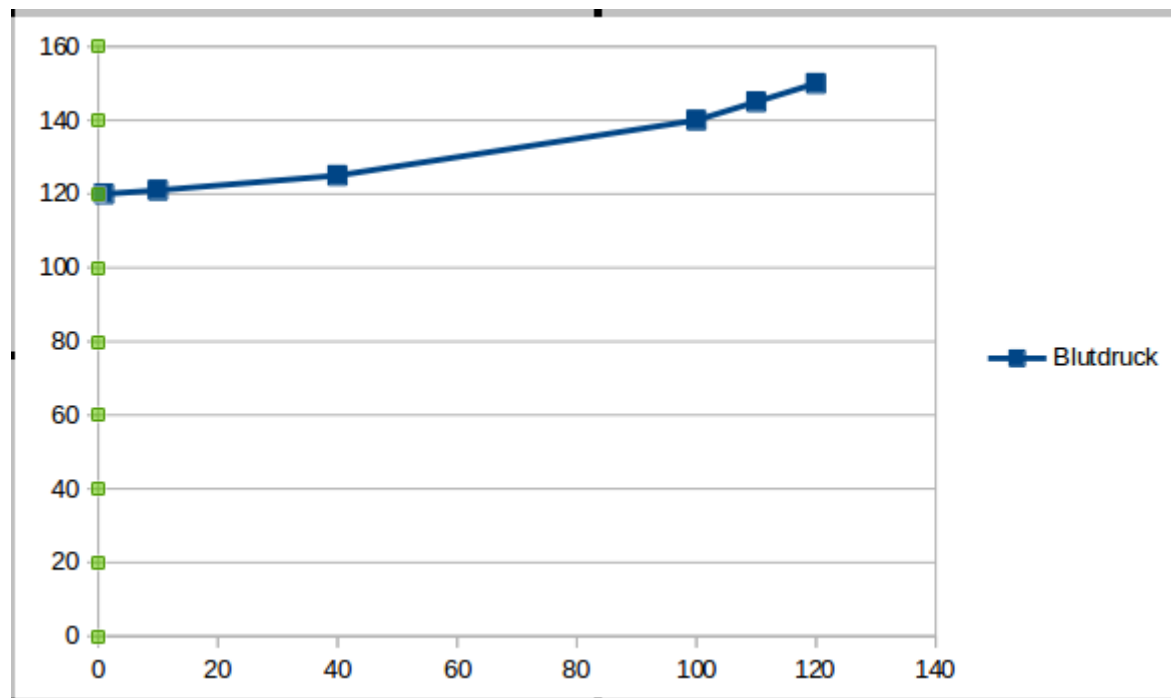
Besitz



Eigenschaft

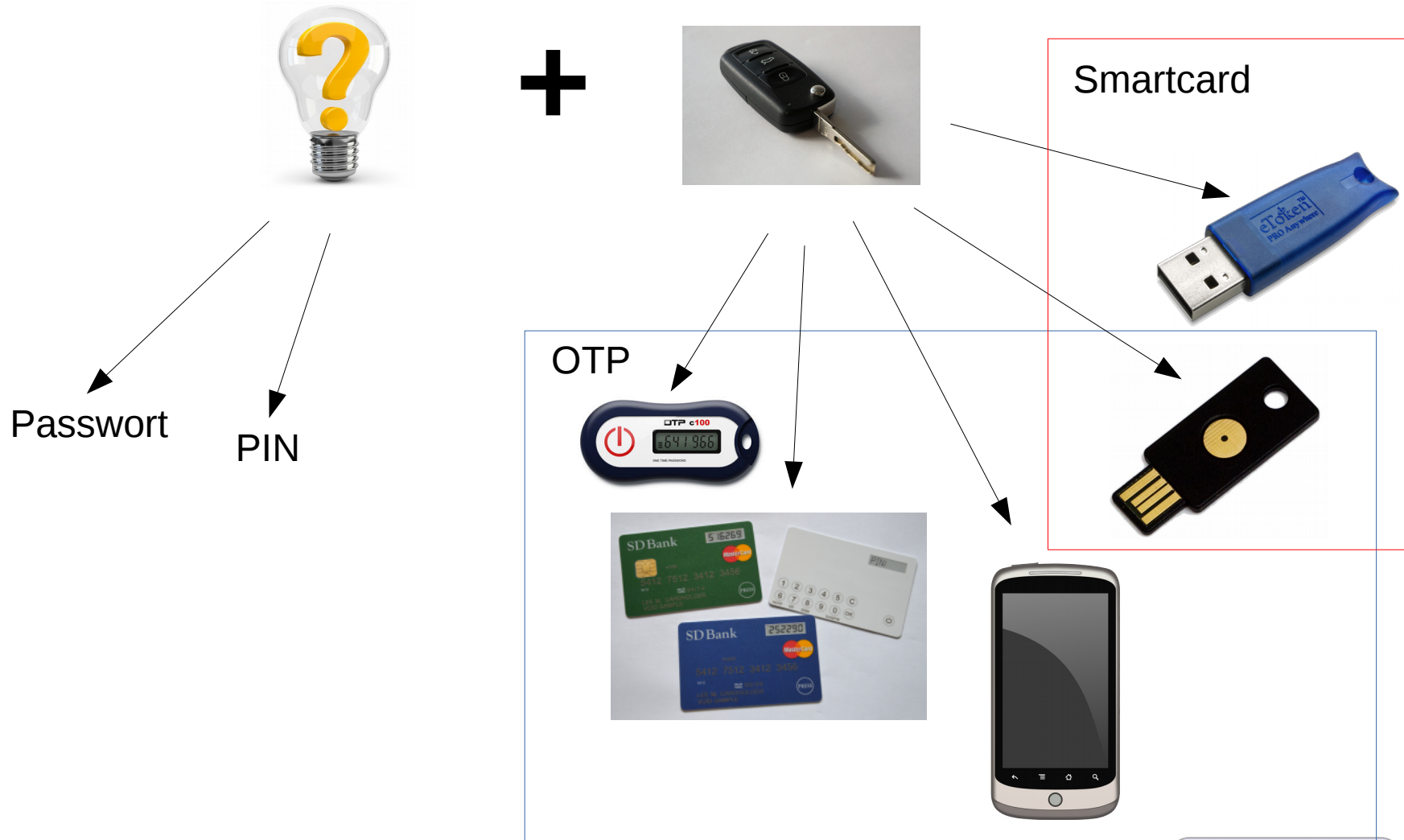
Passwörter: Komplex und viel

- Tein:ohwenoong7Ao|Sho



- Nur das Passwort reicht nicht aus.

Mehr-Faktor-Authentifizierung: Kombination



Warum jetzt Mehr-Faktor-Authentifizierung?

- Angriffsszenarien und Skill-Profile des Angreifers



- Phishing / Social Engineering
- SQL-Injection
- Cracker / Skript-Kiddie



- Physikalischer Diebstahl
- Zugriff zum Firmengebäude



- Körperkontakt
- Partybesucher / Einwohnermeldeamt

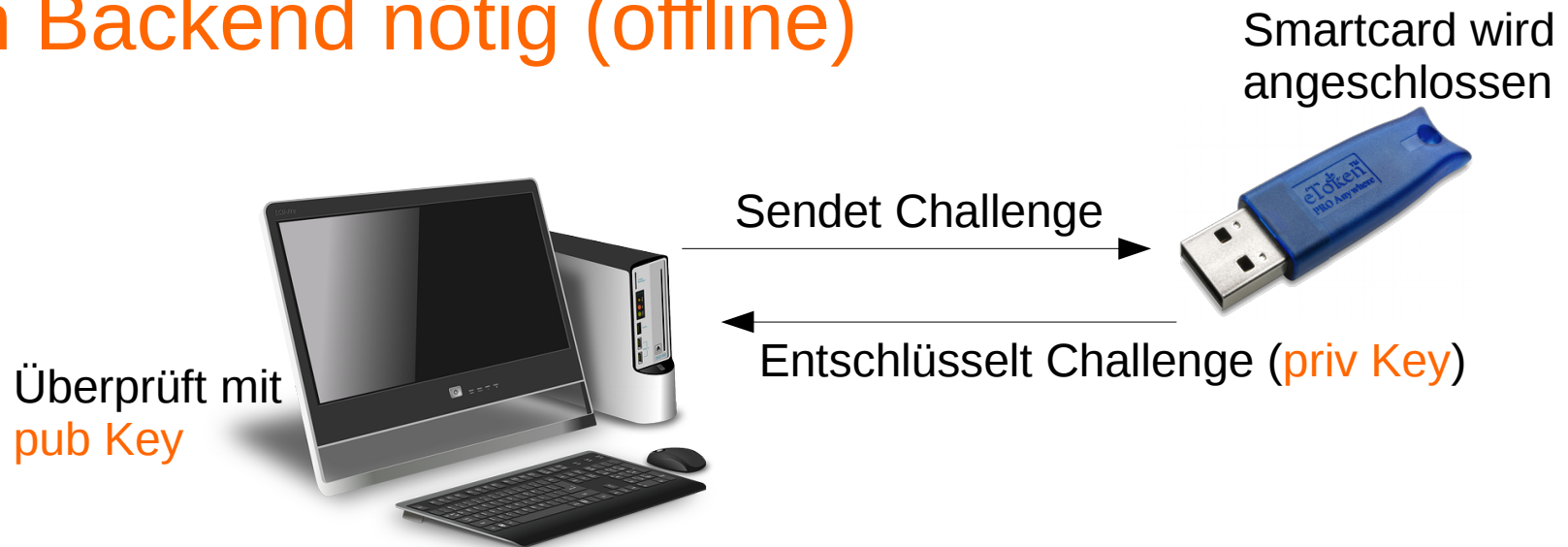
Anforderungen an 2. Faktor

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



Smartcard

- Assymmetrischer Algorithmus (RSA bis 2048/4096 bit)
- Treiber erforderlich
- Kein Backend nötig (offline)



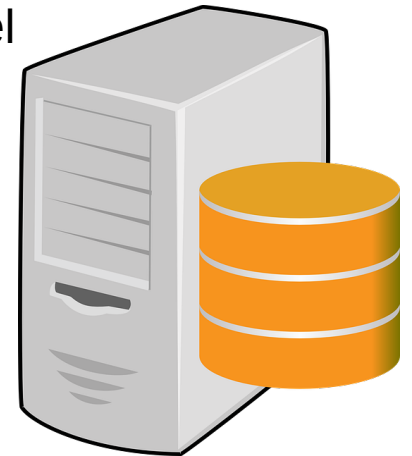
OTP: Einmal-Passwörter

- Symmetrischer Algorithmus (RFC 4226, 6238...)
- Keine Treiber
- Backend erforderlich

Rechnet nach RFC
mit symmetrischem
Schlüssel

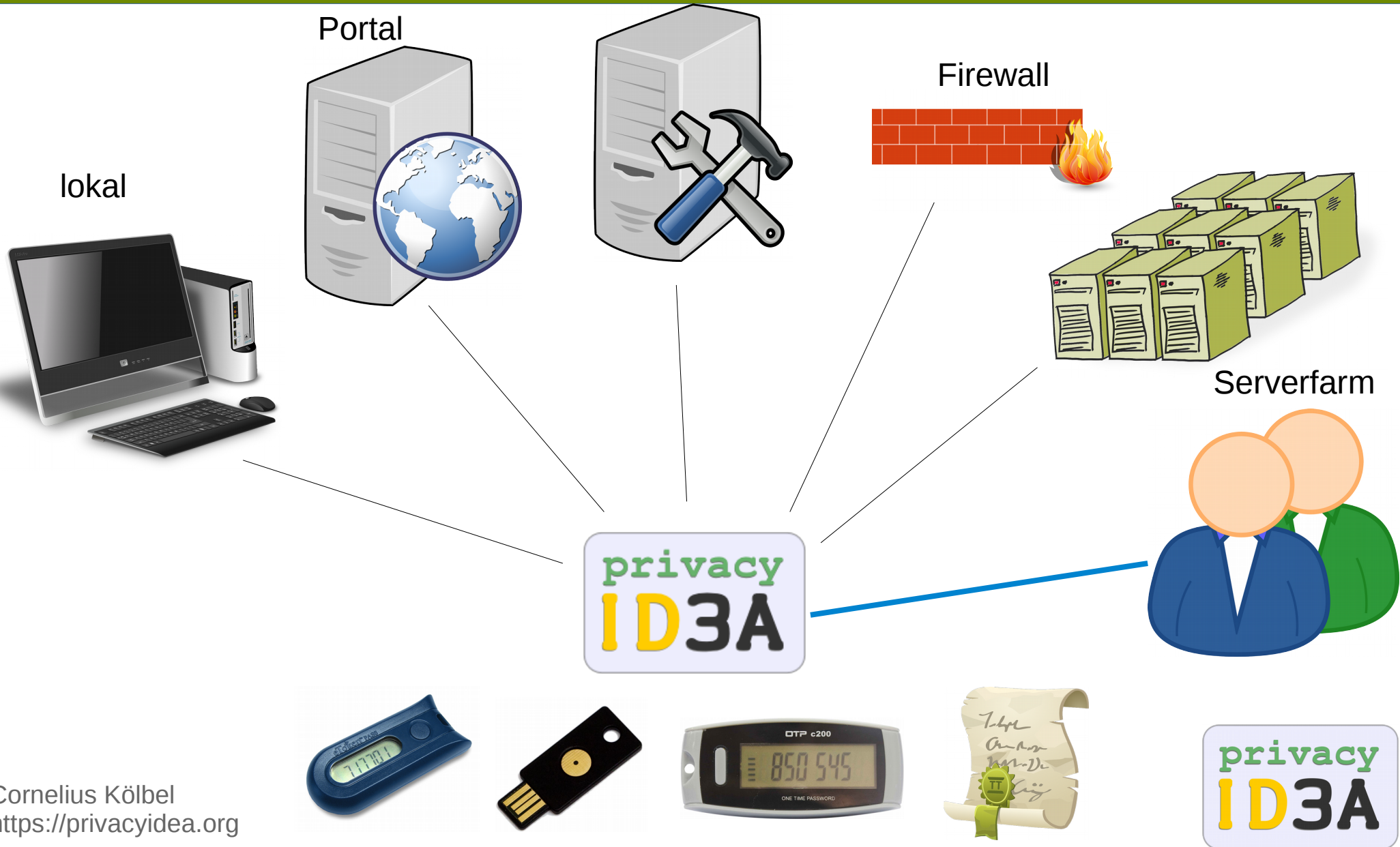


Rechnet nach RFC
mit symmetrischem
Schlüssel



privacyIDEA

Überblick



privacyIDEA

Features

- Offen, Open Source, Kein Vendor-Lock-In,
- Auth-Devices: Yubikey, U2F, HOTP/TOTP, TiQR, SMS, Email, Google Auth, SSH-Key, X.509 u.v.m.
 - Zertifizierungsstelle
- Policies → *Migration (2.11)*
- Event Handler Framework → *Email Notification (2.12)*
- API → Automatisierbar (*Token Enrollment*)
- Audit
- Benutzer lokal, LDAP, AD, SQL, SCIM...
- Anbindung von Linux und Windows Desktop, PAM, RADIUS, SAML, Wordpress, OTRS, Dokuwiki, TYPO3, Contao, ownCloud/Nextcloud u.v.m.

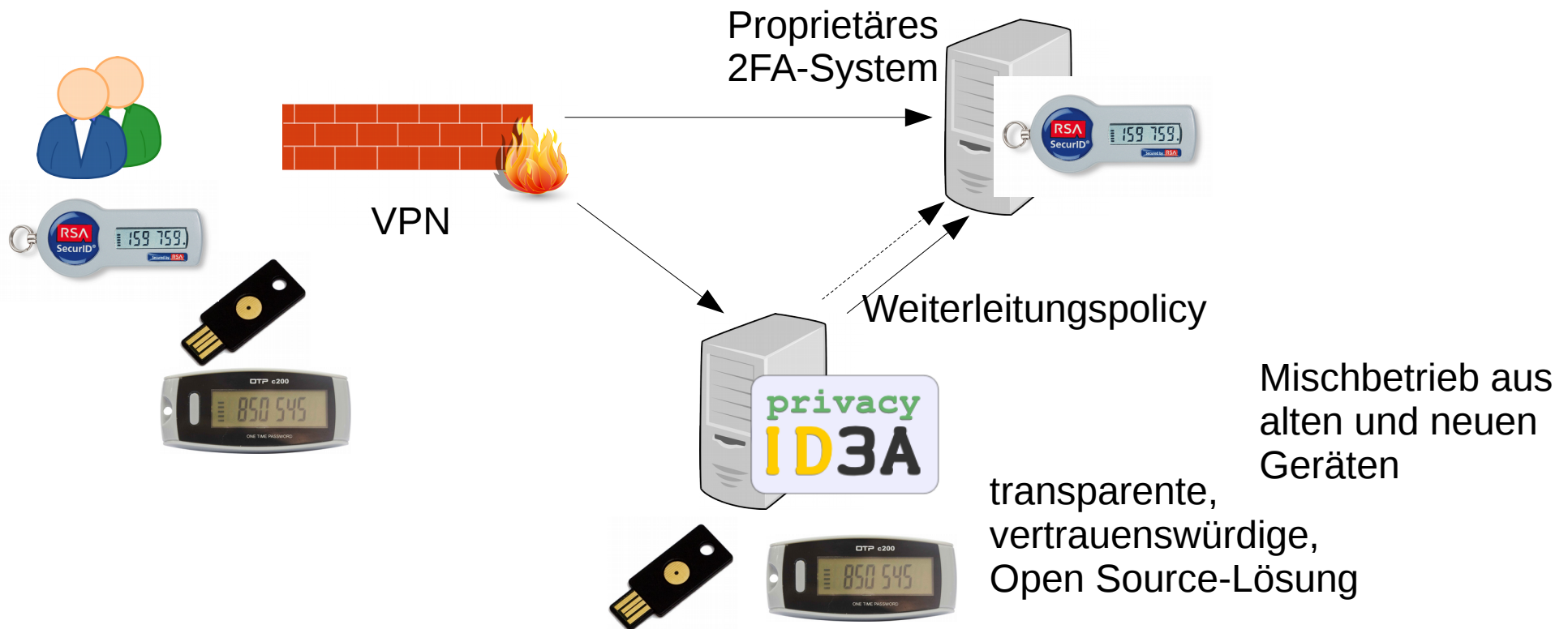
privacyIDEA

Features

- Beispiele:
 - Migration / Policies
 - Event Handler
 - Token Enrollment mittels API

Migration bestehender OTP-Lösungen

- Sanfte Migration



Migration im UI

- Definition von zentralen RADIUS Servern.
 - Für Migration oder auch für
 - RADIUS-Token

privacyIDEA Tokens Users Machines **Config** Audit admin @ (admin)

System Policies Events Tokens Machines Users Realms CAs

List RADIUS server definitions

New RADIUS server

Identifier	IP/FQDN	Dictionary	Description	
2ndRadius	10.0.1.3:1812	/etc/privacyidea/dictionary		Delete
SecurID	10.0.1.2:1812	/etc/privacyidea/dictionary		Delete
localhost	127.0.0.1:1812	/etc/privacyidea/dictionary		Delete

Migration im UI

privacyIDEA

Tokens

Users

Machines

Config

Audit

admin @ (admin)

System

Policies

Events

Tokens

Machines

Users

Realms

CAs

All Policies

Create new Policy

Edit Policy migration

Disable

Delete

Policy Name

migration

If you change the name of the policy, it will create a new policy with the new name!

+ Create Policy

Scope

authentication

Action

passthru

SecurID

If set, the user in this realm will be authenticated against the userstore or against the given RADIUS config, if the user has no tokens assigned.

smstext

text...

The text that will be send via SMS for an SMS token. Use <otp> and <serial> as parameters.

smsautosend

If set, a new SMS OTP will be sent after successful authentication with one SMS OTP.

emailsubject

text...

The subject of the EMail for an EMail token. Use <otp> and <serial> as parameters.

- Definition einer Richtlinie.

Event Handler - Notification

- Jeder Aktion im UI ist ein API-Call
- Admins und Helpdesk-MA können Token verwalten.

The screenshot displays two panels from the FreeIPA web interface. The left panel shows the 'Details for user cornelius in realm localsql', including fields for Username, Email, Given name, Phone, Surname, and Mobile. Below this is a table of 'Tokens for user cornelius' with columns for serial, type, Active status, window, description, failcounter, maxfail, and otpen. The right panel shows the 'Edit Policy superuser' configuration, including fields for Policy Name, Scope, and Admin-Realm, and a list of actions like set, revoke, adduser, enrollSMS, and policydelete.

System Policies Events Tokens Machines Users Realms CAs

Details for user cornelius in realm localsql

View user in Audit log

Username
cornelius

Email
cornelius.koelbel@netknights.it

Given name
Cornelius

Phone

Surname

Description

Mobile

Tokens for user cornelius

serial	type	Active	window	description	failcounter	maxfail	otplen
OATH0014AB4D	hotp	active	10		0	10	6
TOTP00112860	totp	active	10		0	10	6

Enroll New Token

Assign a new token

Serial

start typing a serial number of a token that is not assigned, yet.

Edit Policy superuser

Disable Delete

Policy Name superuser
If you change the name of the policy, it will create a new policy with the new name!

+ Create Policy

Scope admin

Admin-Realm None Selected

Action

- set** Admin is allowed to set properties.
- revoke** Admin is allowed to revoke tokens.
- adduser** Admin is allowed to add users to userstore/UserIdResolver.
- enrollSMS** Admin is allowed to enroll SMS tokens.
- policydelete** Admin is allowed to delete policies.

Event Handler - Notification

- Im Rahmen ihrer Rechte (Policies) können Admins neue Token ausrollen
→ Missbrauchspotential (trotz sign. Audit)
- An Token-Events */token/init* können Aktionen (Notification gebunden werden)
- Aktionen können an alle events (API-calls) gebunden werden.

Event Handler - UI

System Policies **Events** Tokens Machines Users Realms CAs

All Event Handlers
Create new Event Handler

Create a new Event Handler

Events token_init, token_assign, token_unassign, token_revoke, token_enable, token_disable ▾

Handlermodule UserNotification ▾

Condition

Action sendmail ▾

Options

body

```
Hallo {user},  
  
Der admin {admin} hat an Deinem Token {serial} rumgefummelt!
```

The body of the mail that is sent.

emailconfig themis ▾
Send notification email via this email server.

subject
The subject of the mail that is sent.

+ Create Event Handler Definition

Handler Module

- Definiert mögliche Aktionen
 - und dazugehörigen Parameter
 - <http://privacyidea.readthedocs.io/en/latest/modules/lib/eventhandler.html>
Methode *actions* liefert entsprechendes *dict* zurück
- Führt Aktionen entsprechend der zentralen Definitionen aus
 - Methode *do(action, options)* führt Aktion aus.
- UserNotification:
 - 89 Zeilen Python Code!

Denkbare Event Handler

- Neue Python Klasse von *BaseEventHandler*
 - PIN Notification
 - Clean Up
 - Datenbank
 - Abgelaufene/Tote Token
 - Enrollment/Issuing

- You Choose! → Github

API – Token Enrollment

- Vollständige REST API (readthedocs.io)
- Authentisierung an API auch gegen privacyIDEA selber
- JSON Web Tokens

POST /token/set

This API is only to be used by the admin! This can be used to set token specific attributes like

- description
- count_window
- sync_window
- count_auth_max
- count_auth_success_max
- hashlib,
- max_failcount

The token is identified by the unique serial number or by the token owner. In the later case all tokens of the owner will be modified.

JSON Parameters:

- **serial** (*basestring*) – the serial number of the single token to reset
- **user** (*basestring*) – The username of the token owner
- **realm** (*basestring*) – The realm name of the token owner

Return: returns the number of attributes set in “value”

Rtype: json object

GET /token/

Display the list of tokens. Using different parameters you can choose, which tokens you want to get and also in which format you want to get the information (*outform*).

Query Parameters:

- **serial** – Display the token data of this single token. You can do a not strict matching by specifying a serial like “OATH”.
- **type** – Display only token of type. You can do a non strict matching by specifying a tokentype like “otp”, to filter hotp and totp tokens.
- **user** – display tokens of this user
- **viewrealm** – takes a realm, only the tokens in this realm will be displayed
- **description** (*basestring*) – Display token with this kind of description
- **sortby** – sort the output by column

API – Token Enrollment

- JWT holen:

```
POST /auth
```

```
username=admin
```

```
password=topsecret290374
```

- Response: "token": "eyJGciOiJIUz...GejmbFbM"

- Token ausrollen:

```
POST /token/init
```

```
Authorization: eyGciOiJIUz...GejmbFbM (Header)
```

```
<diverse Parameter>
```

Token Enrollment – python module

- Token in python code ausrollen:

```
client = privacyideaclient(admin, password, URL)
client.inittoken(parameter)
```

- Token löschen

```
client.remove_token({"serial": <seriennummer>})
```

Ausblick - Roadmap

- Siehe Github / milestones / issues:
 - PIN-Handling → Eventhandling
 - SMS und Email-Token vereinen
 - Apps wie pushover
 - ownCloud 9.1/Nextcloud
 - Weitere plugins für Webapplikationen

Schau mal rein

- <http://github.com/privacyidea/>
 - Travis CI, codecov.io
- <http://privacyidea.org>
- <http://privacyidea.readthedocs.io>
- <https://launchpad.net/~privacyidea>
- <https://groups.google.com/forum/#!forum/privacyidea>

Ihre nächsten Schritte

- Noobs:
 - An welchen Stellen brauchen Sie 2FA?
- Experts:
 - Richtige Technologie?
Sowohl bei der Management-Software als auch bei den Token?
→ Migration.
- Workshop **HEUTE, 16:00 Uhr W3**

