

# Informationssicherheit und IT-Forensik



1-05 – Authentifikation und digitale Identitäten



## Lernziele dieser Einheit

- Grundlegende Begriffe und deren Zusammenhang verstehen und merken
- Verstehen, warum Menschen die Wahl von und der Umgang mit „guten“ Passwörtern schwer fällt
- Verstehen, was ein gutes Passwort ausmacht und wie dessen Güte gemessen werden kann
- Grundzüge und Vorteile/Nachteile verschiedener Authentifikationsverfahren kennen (z.B. Authentifikation durch Wissen oder durch Besitz)

## Motivation

- Beispiel Türsteher, Einlass z.B. nach
  - „Nasenfaktor“
  - Persönlicher, direkter Bekanntheit
  - Gästeliste
  - Codewort
- Beispiel Notar, Überprüfung z.B. nach
  - Persönlicher, direkter Bekanntheit
  - Abgleich mit vorgelegtem Personalausweis
- Diese Maßnahmen sind mehr oder weniger Zuverlässig und hängen auch vom Input anderer ab (Personalausweis gefälscht?)

## Motivation

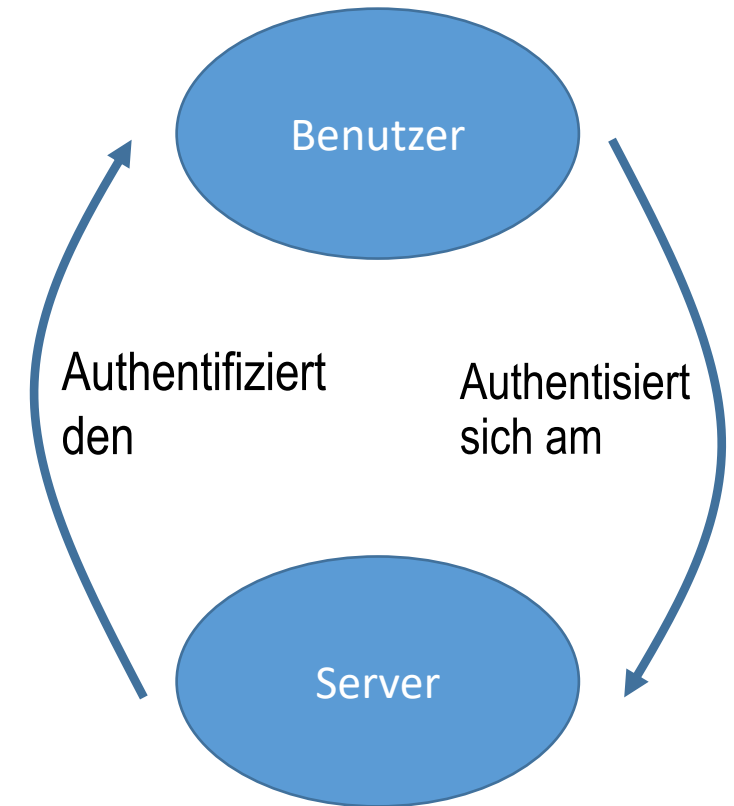
- Definierte Informationssicherheit kann nur erreicht werden, wenn interagierende Objekte/Subjekte mit ausreichender Gewissheit identifiziert werden können
- Identität
  - Setzt sich zusammen aus Identitätsattributen (charakteristische Eigenschaften)
  - Z.B. ein Benutzername oder eine Kundennummer
- Identifikation
  - A sagt zu B: „Ich bin A“
  - Zunächst nur eine Behauptung
  - Es gibt Situationen, in denen das allein bereits reicht
  - Typischerweise folgt jedoch die Frage von B: „Stimmt das auch?“
- Ziel: ausreichend hohe Fälschungssicherheit einer Identität

## Definition: Authentizität

- *„Unter der Authentizität eines Objekts bzw. Subjekts (engl. authenticity) verstehen wir die Echtheit und Glaubwürdigkeit des Objekts bzw. Subjekts, die anhand einer eindeutigen Identität und charakteristischen Eigenschaften überprüfbar ist.“*  
Claudia Eckert, IT-Sicherheit, 9. Auflage, S. 8
- Überprüfung von Eigenschaften einer Identität => authentisch

## Begriffe: Authentifikation

- Mittels Maßnahmen der Authentifikation wird die Authentizität eines Objekts oder Subjekts überprüft
- Nachweis, dass behauptete Identität mit den charakterisierenden Eigenschaften eines Objekts/Subjekts übereinstimmt
- Granulare Unterscheidung im Deutschen
  - Authentifikation: Prüfung der Echtheit/Übereinstimmung
  - Authentifizierung: Bezeugung der Echtheit/Übereinstimmung -> ein Verfahren
- Authentisierung: der Benutzer weist seine Identität nach



## Begriffe: Authentifikation

- **Subjekt-Authentizität**
  - Benutzer -> erhält Kennung
  - Authentisiert sich z.B. mittels Kennwort
- **Objekt-Authentizität**
  - Server -> weist sich gegenüber Client/Benutzer aus
  - Quellcode -> wird als unverändert überprüft
  - Nachweis z.B. mittels digitaler Signatur oder Hash (HMAC)

## Begriffe: Autorisierung

- Lateinisch: auctorizare -> bestätigen, beglaubigen
- Prüfung und Kontrolle von Rechten
- Subjekt + Zugriffsberechtigung = autorisiert
- In Bezug auf Authentifikation und digitale Identitäten ist damit meist gemeint: System S autorisiert Benutzer B mit Zugriffsrechten Z



Begriffe: Verwendung im Alltag

- Identifikation
  - Authentifikation
  - Authentifizierung
  - Authentisierung
  - Autorisierung
- 
- Im Alltag/Praxis: Nicht immer exakt verwendet, gelegentlich sogar synonym

## Authentifikation mittels Wissen

- „Der Klassiker“
- Authentifikation erfolgt mittels Austausch eines Geheimnis, das beiden Seiten bekannt ist
- Beispiel: Passwort/PIN
- Vorteile:
  - Erfordert keine spezielle Hardware auf Seiten des Benutzers (Passwort/PIN wird über Tastatur eingegeben)
  - Vergleichsweise günstig
  - Einfach zu verwenden
  - Kann einfach geändert werden
- Nachteile:
  - Jeder, der das Passwort kennt, kann die zugehörige Identität impersonifizieren
  - Benutzer neigen dazu, schwache Passwörter zu vergeben
  - Shoulder Surfing
  - Kann z.B. in einer Stress-Situation vergessen werden

Authentifikation mittels Wissen: wenn Benutzer selbst Passwörter vergeben

- Quelle: *Password Security: A Case History, Morris und Thompson, Communications of the ACM, 1979*
- Anwender neigen offenbar von sich aus zur Wahl schwacher Passwörter

The authors have conducted experiments to try to determine typical users' habits in the choice of passwords when no constraint is put on their choice. The results were disappointing, except to the bad guy. In a collection of 3,289 passwords gathered from many users over a long period of time,

15 were a single ASCII character;  
72 were strings of two ASCII characters;  
464 were strings of three ASCII characters;  
477 were strings of four alphanumerics;  
706 were five letters, all upper-case or all lower-case;  
605 were six letters, all lower-case.

An additional 492 passwords appeared in various available dictionaries, name lists, and the like. A total of 2,831 or 86 percent of this sample of passwords fell into one of these classes.

Authentifikation mittels Wissen: wenn Entwickler Passwörter vorgeben



Authentifikation mittels Wissen: Empfehlungen für gute Passwörter

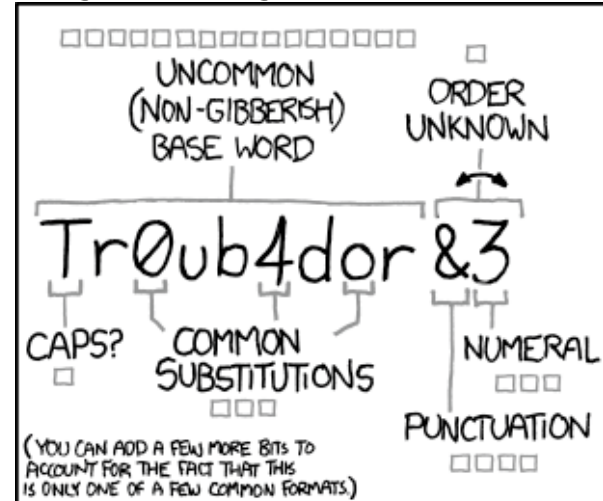
- Je nach Wissenstand, Meinung/Konsens/Experte:
  - Passwörter regelmäßig ändern
  - Passwörter nicht ändern
  - Großbuchstaben erzwingen (mindestens 1/2/3, genau 1/2/3)
  - Keine speziellen Zeichen erzwingen
  - ...
- Gegenstand einer gewissen Kontroverse
- Hängt auch von den Anforderungen und Gegebenheiten ab, sowie Einsatzzweck
- Zwei grundsätzlich geeignete Kriterien:
  - Hohe Entropie des verwendeten Passwortes
  - Dem Benutzer ein abstraktes Feedback geben (schlecht/mittel/gutes Passwort) + Hinweise für gute Passwörter

Authentifikation mittels Wissen: Empfehlungen für gute Passwörter

- <https://pages.nist.gov/800-63-3/>
- [https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/\\_content/m/m02/m02011.html](https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/Inhalt/_content/m/m02/m02011.html)
- [https://en.wikipedia.org/wiki/Password\\_strength](https://en.wikipedia.org/wiki/Password_strength)
- <https://en.wikipedia.org/wiki/Diceware>
- <https://queue.acm.org/detail.cfm?id=2422416>

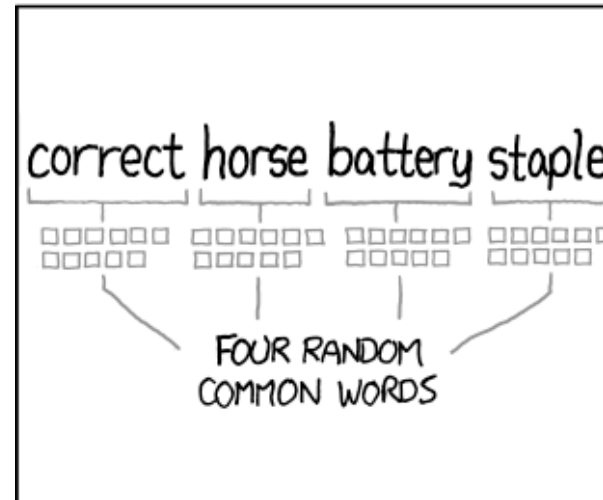
## Authentifikation mittels Wissen: Empfehlungen für gute Passwörter

- Quelle: <https://xkcd.com/936/>
- [https://www.explainxkcd.com/wiki/index.php/936:\\_Password\\_Strength](https://www.explainxkcd.com/wiki/index.php/936:_Password_Strength)
- $\log_2(2048) = 11 \rightarrow$  Auswahl aus 2048 verbreiteten Wörtern
- Überlegung:
  - $\log_2 1024 = 10$
  - $\log_2 4096 = 12$
- Frage: Wie hoch ist die Entropie der hier gezeigten, konkreten Passwörter?
- Frage: muss es komplett random sein?



~28 BITS OF ENTROPY  
 $2^{28} = 3 \text{ DAYS AT } 1000 \text{ GUESSES/SEC}$   
 (PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)  
 DIFFICULTY TO GUESS: **EASY**

WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE 0s WAS A ZERO?  
 AND THERE WAS SOME SYMBOL...  
 DIFFICULTY TO REMEMBER: **HARD**



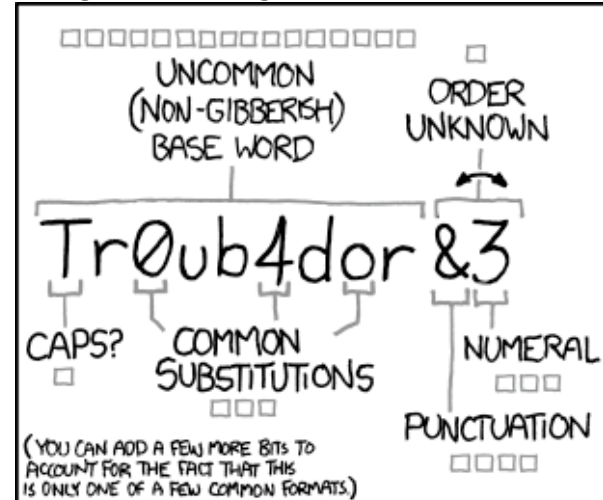
~44 BITS OF ENTROPY  
 $2^{44} = 550 \text{ YEARS AT } 1000 \text{ GUESSES/SEC}$   
 DIFFICULTY TO GUESS: **HARD**

THAT'S A BATTERY STAPLE.  
 CORRECT!  
 DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT

THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

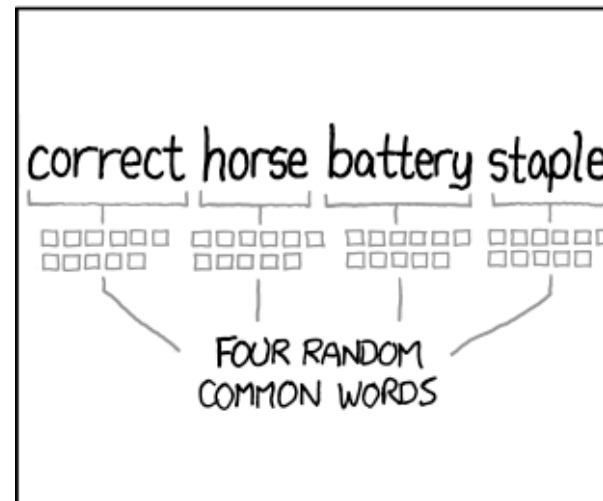
## Authentifikation mittels Wissen: Empfehlungen für gute Passwörter

- [https://www.schneier.com/blog/archives/2014/03/choosing\\_secure\\_1.html#!s!xkcd](https://www.schneier.com/blog/archives/2014/03/choosing_secure_1.html#!s!xkcd)
- Kritische Debatte zu verschiedenen Aspekten und Überlegungen in Bezug auf Entropie



~28 BITS OF ENTROPY  
 $2^{28} = 3 \text{ DAYS AT } 1000 \text{ GUESSES/SEC}$   
 (PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)  
 DIFFICULTY TO GUESS: **EASY**

WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE O's WAS A ZERO?  
 AND THERE WAS SOME SYMBOL...  
 DIFFICULTY TO REMEMBER: **HARD**



~44 BITS OF ENTROPY  
 $2^{44} = 550 \text{ YEARS AT } 1000 \text{ GUESSES/SEC}$   
 DIFFICULTY TO GUESS: **HARD**

THAT'S A BATTERY STAPLE.  
 CORRECT!  
 DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT

THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.



Authentifikation mittels Wissen: Einmal-Passwörter

- one-time password (OTP)
- Passwort, das nur einmal verwendet werden kann -> wird nach Verwendung invalidiert
- Beispiele
  - TAN-Liste bei Online-Banking
  - S/Key-Verfahren
  - Google Authenticator
  - RSA SecureID
  - Allgemein: Seed-basierte Verfahren

Authentifikation mittels Wissen: Einmal-Passwörter -> Beispiel eines einfachen Seed-Verfahrens

## Authentifikation mittels Besitz

- Authentifikation erfolgt mittels Verwendung eines „greifbaren Objekts“
- Beispiel: elektronisches Zertifikat, Smartcard, physischer Schlüssel
- Unterscheidet sich (wenn digital) nicht grundlegend von „Wissen“
- Vorteile:
  - Smartcard: verfügt über aktive und passive Schutzmaßnahmen -> nur sehr schwer zu kopieren
  - Für den Benutzer etwas „greifbares“, muss nur „zur Hand sein“
  - Shoulder Surfing nicht möglich
  - Kann getauscht werden
- Nachteile:
  - Kann verloren/kaputt gehen
  - Einfache Systeme sind (zu leicht) kopierbar
  - Muss mitgeführt werden
  - Kann an (zu leicht) an jemand anderen weitergegeben werden

## Authentifikation mittels biometrischem Merkmal

- Authentifikation erfolgt mittels körpereigenem biometrischem Merkmal
- Beispiel: Stimme, Gesicht, Fingerabdruck, Venen, DNA...
- Idee: besonders schwer zu kopieren/imitieren, aber...
- Vorteile:
  - Für den Benutzer sehr intuitiv und „immer dabei“
  - Shoulder Surfing nicht möglich
  - Kann nicht an andere weitergegeben werden
- Nachteile:
  - Kann „kaputt“ gehen
  - Kann nicht geändert werden
  - Erfordert physische Anwesenheit oder vertrauenswürdige Lesestation
  - Unscharf: false acceptance und false rejection



## Mehrfaktor-Authentifikation

- Kombination von mindestens zwei Verfahren
- Auch möglich: Passwort+Passwort
- Beispiel: Debit-Karte + PIN zum Geldabheben
- Ziel: erhöhte Sicherheit

## Vier-Augen-Prinzip

- Zur Authentifizierung einer Identität müssen sich zwei (oder mehr) Personen authentisieren
  - Z.B. kennt je eine Person einen Teil des Passwortes, oder
  - Eine Person kennt das Passwort, die zweite Person verfügt über das biometrische Merkmal

## Challenge-Response-Verfahren

- Verfahren aus dem Bereich „Authentifikation mittels Wissen“ mit dem Ziel, erhöhter Sicherheit
- Problem bei „Authentifikation mittels Wissen“:
  - Wie kann das Geheimnis im Transit geschützt werden?
  - Wie kann ein Schutz vor Replay-Angriffen bei Man-In-The-Middle-Angriffen erreicht werden?
- Aufgabe: nicht mehr das invariante Geheimnis (Passwort) selbst übertragen, sondern eine davon abhängige, veränderliche Information
- One-time-Passwörter und TAN-Banking sind eine Form des Challenge-Response-Verfahrens

## Challenge-Response-Verfahren

- **Verwendung symmetrischer Kryptographie**
  - Client und Server verwenden den gleichen Algorithmus und den gleichen Key
  - Beispiel einer einfachen Implementierung:
    - Server sendet Client im Klartext eine Zufallszahl (NONCE)
    - Client verschlüsselt diese und sendet das Ergebnis zurück zum Server
    - Server entschlüsselt mit dem Schlüssel. Wenn das Ergebnis nun der NONCE entspricht -> OK
  - Gefahr z.B. durch Known-Plaintext-Angriffe
- **Verwendung asymmetrischer Kryptographie**
  - Verwendung einer Public-Key-Infrastruktur
  - Client signiert die NONCE mit seinem private key
  - Server prüft mit dem ihm bekannten public key des Clients

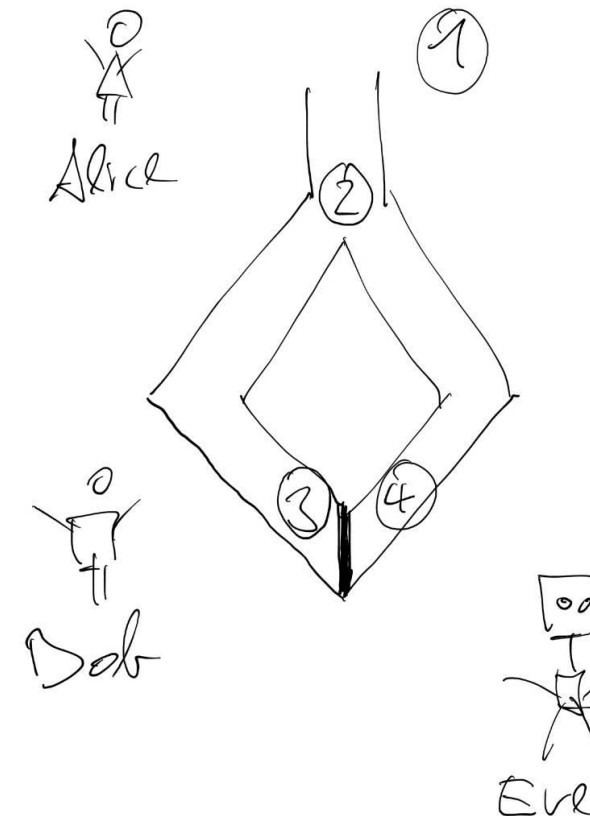


## Exkurs: Zero-Knowledge-Verfahren (Idee und Prinzip)

- Challenge-Response-Verfahren
- Alice überzeugt Bob von der Kenntnis des Geheimnisses
- Bob kann über das Geheimnis selbst nichts erfahren und muss dazu auch nichts speichern
- Eve kann nicht verifizieren, ob Alice und Bob sich abgesprochen haben, oder nicht
- Abstrakte

### Beispielimplementierung/Idee:

Jean-Jacques Quisquater, Louis Guillou: *How to explain zero-knowledge protocols to your children*. Advances in Cryptology - CRYPTO '89, Lecture Notes in Computer Science 435, pp. 628–631, 1990



Zuletzt geändert: 14:49

## Authentifizierung in verteilten Systemen

- Bedarf: Authentifizierung z.B. in (großen) Computernetzwerken
  - Muss wirksam/sicher sein
  - Muss effizient sein
- Typische Einsatzzwecke
  - Clients, die sich an Servern authentisieren
  - Single-Sign-On
  - Zentrale Verwaltung von Benutzerkonten und Zugriffsrechten + Konfigurationsoptionen

## Authentifizierung in verteilten Systemen: RADIUS

- Remote Authentication Dial In User Service
- Ermöglicht mehrstufige, verteilte Authentifizierungssysteme
- Benutzer baut Verbindung zum Netz auf, übergibt Benutzername und Passwort an einen RADIUS-Client
- Dieser übergibt die Credentials an einen RADIUS-Server, der wiederum prüft und das Ergebnis + evtl. Konfigurationsoptionen dem Client mitteilt.
- Der Client gewährt daraufhin dem Benutzer z.B. Zugriff auf den gewünschten Dienst
- Mehrstufig, Proxy-Konzept, ermöglicht auch Roaming durch Vertrauen zwischen RADIUS-Servern
- Kommt z.B. bei DSL-Providern und großen WLAN-Infrastrukturen zum Einsatz

## Authentifizierung in verteilten Systemen: Kerberos

- Ermöglicht mehrstufige, verteilte Authentifizierungssysteme
- Funktioniert auch in „unsicheren“ Netzen, etwa Internet
- Verwendet symmetrische Kryptographie
- Ermöglicht Single-Sign-On
- Token-basiertes System: „Tickets“
- (Sehr) grobe, vereinfachte Architektur
  - Client: möchte einen Dienst/Server nutzen
  - Server: möchte wissen, ob der Client autorisiert ist
  - Kerberos-Server: übernimmt die Authentifizierung und stellt ein Token aus
  - Der Server erhält vom Client das Token und kann dessen Gültigkeit überprüfen
- Kommt z.B. bei Windows ActiveDirectory zum Einsatz
- Ist sowohl für Server als auch Client in Open-Source-Implementierungen verfügbar

Serverseitig: Klartextspeicherung, Salting und verfügbare Techniken wie PBKDF2

- Wenn Passwörter im Klartext auf einem System gespeichert sind -> „fette Beute“
- Seit langem verwendete Lösung: den Hash eines Passwortes speichern
- Probleme:
  - Hashes gleicher Passwörter ebenfalls gleich
  - Hashes von alten Verfahren heute schnell zu berechnen („knacken“)

Serverseitig: Klartextspeicherung, Salting und verfügbare Techniken wie PBKDF2

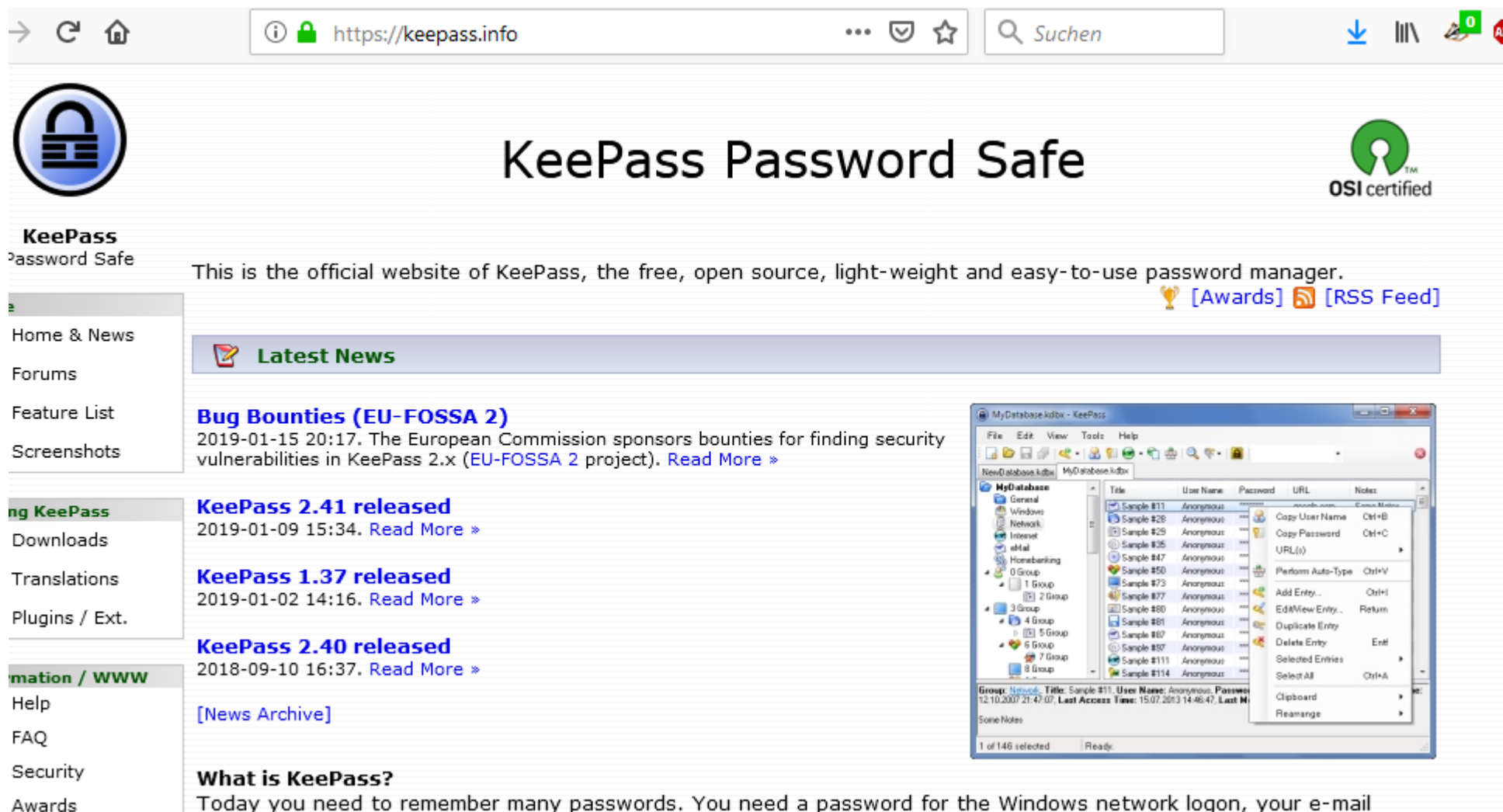
- Lösung „Salting“ - Grundidee:  
an jedes serverseitig gespeicherte Passwort einen zufällig gewählten Wert anhängen
- Dazu nötig:
  - Cryptographically Secure Pseudo-Random Number Generator
  - oder auf deutsch:  
Etwas, das wirklich sichere zufällig aussehende Zahlen erzeugt
  - Speichern von Salt: als zusätzliche Spalte in Datenbank
  - Nicht schlimm, wenn Angreifer diese beim Angriff auch ausliest  
→ er weiß trotzdem nicht, ob  $\text{password}(\text{Alice}) = \text{password}(\text{Bob})$



Serverseitig: Klartextspeicherung, Salting und verfügbare Techniken wie PBKDF2

- PBKDF2 (Password-Based Key Derivation Function 2)
- ist ein standardisiertes Verfahren und Bestandteil der Public-Key Cryptography Standards von RSA-Laboratories (PKCS #5)
- Aufgabe: Schlüssel von einem Passwort ableiten und für symmetrische Verfahren verwenden
- Implementiert Salting und hohe Anzahl an Iterationen
- Vorteil:
  - Verlangsamt so Brute-Force-Angriffe

## Clientseitig: Verwendung von Passwort-Safes



The image shows the official KeePass website and a screenshot of the KeePass application. The website is for 'KeePass Password Safe' and describes it as a free, open source, light-weight and easy-to-use password manager. It features a sidebar with links to Home & News, Forums, Feature List, Screenshots, Downloads, Translations, Plugins / Ext., Help, FAQ, Security, and Awards. The main content area includes 'Latest News' with entries for 'Bug Bounties (EU-FOSSA 2)', 'KeePass 2.41 released', 'KeePass 1.37 released', and 'KeePass 2.40 released'. A 'What is KeePass?' section is also visible. The application screenshot shows a database named 'MyDatabase.kdbx' with a list of entries, including 'Sample #111' through 'Sample #114', each with a title, user name, password, URL, and notes. A context menu is open over the 'Sample #111' entry, showing options like 'Copy User Name', 'Copy Password', 'Perform Auto-Type', 'Add Entry...', 'Edit/View Entry...', 'Duplicate Entry', 'Delete Entry', 'Selected Entries', 'Select All', 'Clipboard', and 'Rearrange'.

**KeePass Password Safe**

This is the official website of KeePass, the free, open source, light-weight and easy-to-use password manager. [\[Awards\]](#) [\[RSS Feed\]](#)

**Latest News**

**Bug Bounties (EU-FOSSA 2)**  
2019-01-15 20:17. The European Commission sponsors bounties for finding security vulnerabilities in KeePass 2.x (EU-FOSSA 2 project). [Read More »](#)

**KeePass 2.41 released**  
2019-01-09 15:34. [Read More »](#)

**KeePass 1.37 released**  
2019-01-02 14:16. [Read More »](#)

**KeePass 2.40 released**  
2018-09-10 16:37. [Read More »](#)

[\[News Archive\]](#)

**What is KeePass?**  
Today you need to remember many passwords. You need a password for the Windows network login, your e-mail

## Universal Second Factor Authentication (U2F)

- Offenes Protokoll, ursprünglich von Google entwickelt, jetzt von FIDO-Alliance
- Ermöglicht Zwei-Faktor-Authentifizierung im Web
- Unterstützt z.B. von Chrome, Firefox, Safari
- Kann als zweiten Faktor z.B. USB-Token verwenden, aber auch reine Software-Lösung (z.B. lokal gespeichertes Zertifikat)
- Ein Benutzer muss sich vor der ersten Verwendung bei der Webanwendung mit seinem U2F-Client registrieren
- Anschließend erfolgt Eingabe von Benutzername/Passwort und Hinzuziehung des zweiten Faktors
- Anonymes Login kann auch realisiert werden
- Pro Webanwendung ein eigenes Schlüsselpaar

## Smart Cards



## Smart Cards: Auswahl an Standards

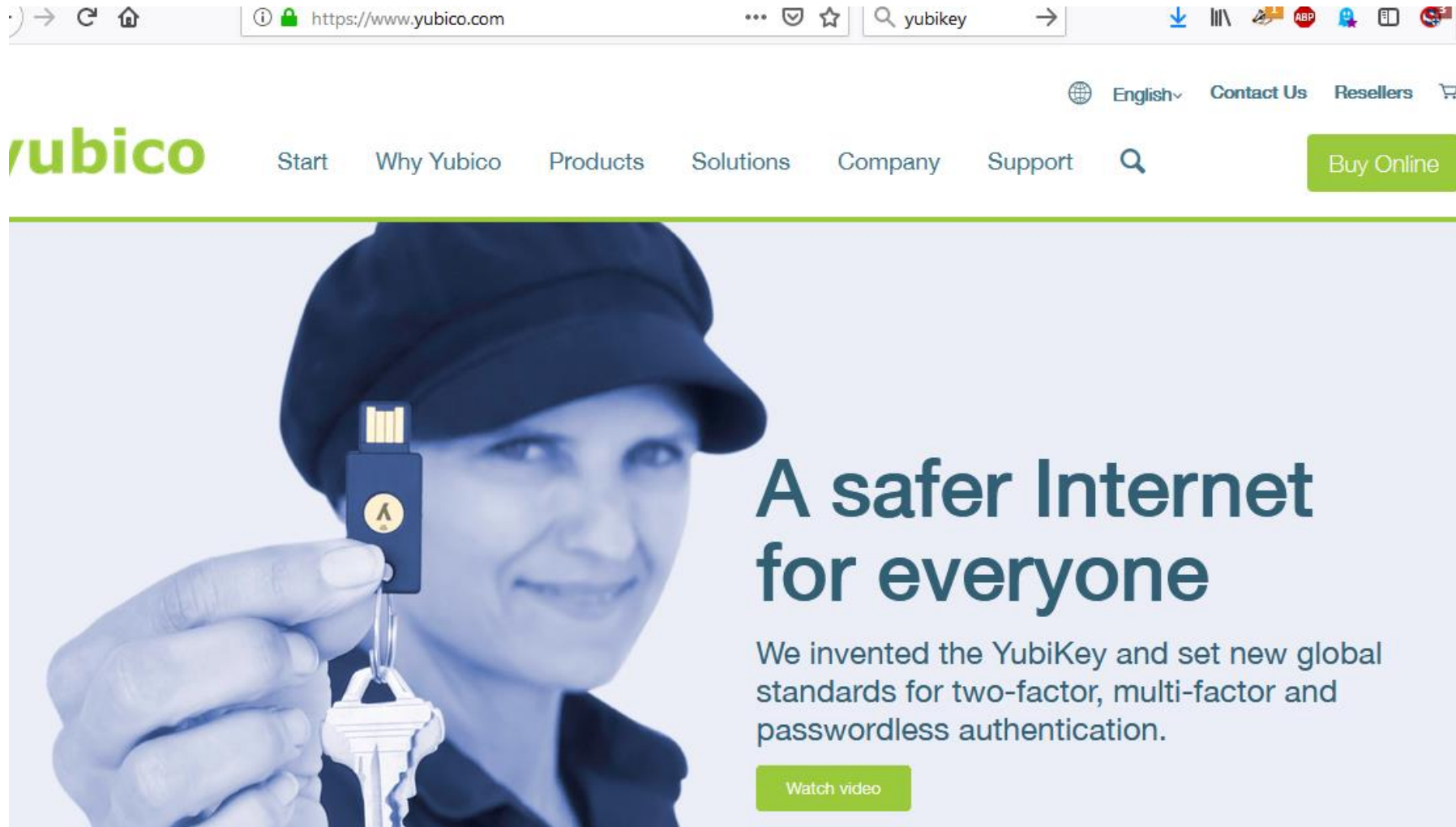
- ISO 7816
  - ISO/IEC 14443-4
  - ISO/IEC 15693
  - ISO/IEC 7501
  - ISO/IEC 18092
  - ...
- 
- Ursprung in den 1970er Jahren

## Smart Cards: Beispiele für Implementierungen in Deutschland

- SIM-Karte
- Hotel-Zutrittskarten
- GeldKarte
- EC-/Debit-Karte
- Elektronische Gesundheitskarte
- Elektronischer Reisepass (ePass)
- Elektronischer Personalausweis (nPA)
  - Qualifizierte Signatur
- ...
- Kommunikation
  - Drahtlos (RFID/NFC)
  - Über Chip-Kontakte



## Smart Cards: YubiKey (USB Token)



The screenshot shows the Yubico website homepage. At the top, there is a browser address bar with the URL 'https://www.yubico.com' and a search bar containing 'yubikey'. Below the browser bar, the Yubico logo is on the left, and navigation links for 'Start', 'Why Yubico', 'Products', 'Solutions', 'Company', and 'Support' are in the center. On the right, there are links for 'English', 'Contact Us', 'Resellers', and a 'Buy Online' button. The main content area features a large image of a person wearing a dark cap and holding a YubiKey USB token. To the right of the image, the headline reads 'A safer Internet for everyone'. Below the headline, a paragraph states: 'We invented the YubiKey and set new global standards for two-factor, multi-factor and passwordless authentication.' At the bottom right of the main content area, there is a 'Watch video' button.

## Smart Cards: Unsicher (eine Schwachstelle) trotz CC-EAL5+-Zertifizierung



**Infineon**

## Unsichere Verschlüsselung – trotz Zertifikat vom Bundesamt

Seite 2/2: Das BSI schweigt

### INHALT

**Seite 1** — Unsichere  
Verschlüsselung – trotz Zertifikat  
vom Bundesamt

**Seite 2** — Das BSI schweigt

**Auf einer Seite lesen** ›

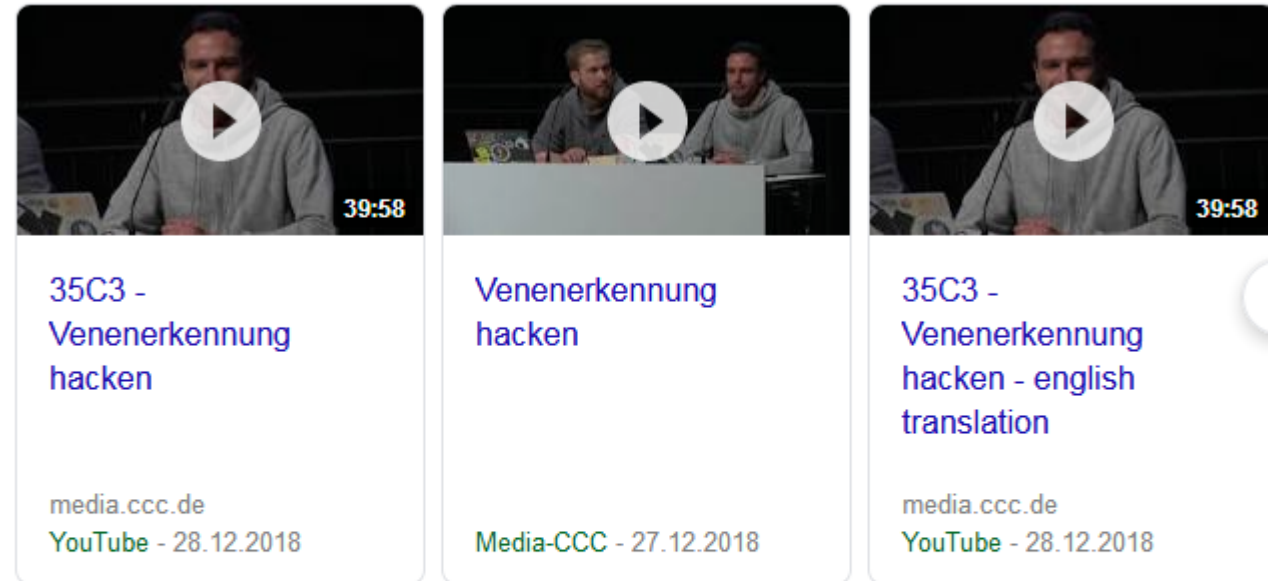
Infineon hat also einen unverzeihlichen Fehler begangen. Eine eiserne Regel bei der Entwicklung von Verschlüsselungssystemen ist es, nie auf Eigenentwicklungen zu setzen. Kryptografische Algorithmen gelten nur dann als sicher, wenn sie über einen längeren Zeitraum bekannt waren und wenn gleichzeitig viele Wissenschaftler versucht haben, sie zu brechen.

Wiederkehrende, vorlesungs- und übungsbegleitende Übungsaufgabe

- Überarbeiten Sie (als Hausaufgabe) Ihre Planung für die Errichtung einer „smarten“ Einbruchmeldeanlage 2.0 für das Handwerksunternehmen Ihrer Eltern
- Beantworten und begründen Sie unter eigenen Annahmen z.B. folgende Fragen:
  - Ändern Sie Ihre bisherige Planung?
  - Welche Authentifizierungstechniken wollen Sie implementieren (Wissen, Besitz, Biometrie)? Warum? Welche nicht? Warum?
  - Skizzieren Sie mögliche Problemstellen. Wo könnten Probleme lauern?

## CCC-Vortrag 2018 zu Biometrie

- Überwindung von Venenerkennungs-Systemen durch Erstellung physischer Kopien
- ACHTUNG:
  - Verwendung ist bei fremden Systemen grundsätzlich illegal
  - Z.B. Verwendung der Merkmale einer anderen Person um diese zu impersonifizieren



**35C3: Mit Venenbild auf Handatrappe Geld abheben oder beim BND ...**

<https://www.heise.de/.../35C3-Mit-Venenbild-auf-Handatrappe-Geld-abheben-oder-b...> ▼

28.12.2018 - Chaos Communication Congress (35C3) in Leipzig vor, dass sie mit einer aus ... Zudem werde die Venenerkennung in Geldautomaten etwa in ...