

GIS

Grundlagen der IT-Sicherheit

P. Felke

Wintersemester 2018

Organisatorisches

- ▶ **Sprechstunde:** Mi, 14.00-15.00 oder n.V.
- ▶ **E-Mail:** patrick.felke@hs-emden-leer.de
- ▶ Die Vorlesungsfolien und Übungsblätter werden in Moodle eingestellt.
- ▶ Bestehen des Praktikums: 3 Testate zu einzelnen Themenkomplexen.
- ▶ Prüfung: Klausur oder mündliche Prüfung.

Weitere Literatur

- ▶ IT-Sicherheit: Konzepte - Verfahren - Protokolle, C. Eckert, De Gruyter, Oldenburg
- ▶ Network Hacking, P. Kraft , A. Weyert, FRANZIS 2017
- ▶ Die Kunst des Human Hacking (Social Engineering), C. Hadnagy, mitp 2011.
- ▶ Paar, C., Pelzl, J.: Kryptografie verständlich, Springer 2016

Interessante Übersichten zu Angriffen oder Sicherheitslücken:

- ▶ Karten mit visueller Aufbereitung von Echtzeitangriffen:

Weitere Literatur

- ▶ IT-Sicherheit: Konzepte - Verfahren - Protokolle, C. Eckert, De Gruyter, Oldenburg
- ▶ Network Hacking, P. Kraft , A. Weyert, FRANZIS 2017
- ▶ Die Kunst des Human Hacking (Social Engineering), C. Hadnagy, mitp 2011.
- ▶ Paar, C., Pelzl, J.: Kryptografie verständlich, Springer 2016

Interessante Übersichten zu Angriffen oder Sicherheitslücken:

- ▶ Karten mit visueller Aufbereitung von Echtzeitangriffen:
 - ▶ von Heise-Security
 - ▶ von Kaspersky
 - ▶ von Norse
 - ▶ von Google und Arbor Networks mit Schwerpunkt auf DDOS-Attacken

Weitere Literatur

- ▶ IT-Sicherheit: Konzepte - Verfahren - Protokolle, C. Eckert, De Gruyter, Oldenburg
- ▶ Network Hacking, P. Kraft , A. Weyert, FRANZIS 2017
- ▶ Die Kunst des Human Hacking (Social Engineering), C. Hadnagy, mitp 2011.
- ▶ Paar, C., Pelzl, J.: Kryptografie verständlich, Springer 2016

Interessante Übersichten zu Angriffen oder Sicherheitslücken:

- ▶ Karten mit visueller Aufbereitung von Echtzeitangriffen:
 - ▶ von Heise-Security
 - ▶ von Kaspersky
 - ▶ von Norse
 - ▶ von Google und Arbor Networks mit Schwerpunkt auf DDOS-Attacken
- ▶ Interaktive Karte mit den größten Sicherheitslücken von information is beautiful
- ▶ Das Forum Netztechnik/Netzbetrieb im VDE (FNN)
- ▶ The Repository of Industrial Security Incidents

Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundschutz

Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundsatz

Malvertising

Tausende WordPress-Sites als Malware-Schleudern missbraucht



Check Point hat rund 10.000 gehackte WordPress-Websites entdeckt, die Teil einer Strategie zur großflächigen Malware-Verbreitung über Werbenetzwerke sind.

70

Ransomware

Krimineller verdient mit Erpressungstrojaner knapp 5 Millionen Euro



Der Trojaner SamSam, der unter anderem die Verwaltung in Atlanta lahmlegte, soll eine einzelne Person reich gemacht haben. Diese ist immer noch nicht gefasst.

46

- ▶ Beinahe täglich werden sicherheitskritische Schwachstellen und Angriffe auf informationsverarbeitende Systeme veröffentlicht.

Warum greift man IT-Systeme an?

- ▶ Klassische Angreifer:
 - ▶ White-Hat: Aufdecken von Sicherheitslücken
 - ▶ Geheimdienste: Spionage, Sabotage und Überwachung
 - ▶ Unternehmen: Wirtschaftsspionage (Zusammenarbeit mit Geheimdiensten)

Warum greift man IT-Systeme an?

- ▶ Klassische Angreifer:
 - ▶ White-Hat: Aufdecken von Sicherheitslücken
 - ▶ Geheimdienste: Spionage, Sabotage und Überwachung
 - ▶ Unternehmen: Wirtschaftsspionage (Zusammenarbeit mit Geheimdiensten)
- ▶ Neuere Entwicklungen:
 - ▶ Etablierung einer stark professionalisierten Schattenwirtschaft (Black-Hat, Auftragshacking).
 - ▶ Veröffentlichung geheimer Informationen (Whistleblower).
 - ▶ Politisch motivierte Angriffe.

Warum greift man IT-Systeme an?

► Schattenwirtschaft:

- ▶ Mit erfolgreichen Angriffen lässt sich viel Geld verdienen.
- ▶ Fälschung von PayTV-Karten.
- ▶ Abgreifen von Kreditkarteninformationen (Warenkreditbetrug).
- ▶ Phishing-Angriffe im Bereich Online-Banking.

Warum greift man IT-Systeme an?

- ▶ Schattenwirtschaft:

- ▶ Mit erfolgreichen Angriffen lässt sich viel Geld verdienen.
- ▶ Fälschung von PayTV-Karten.
- ▶ Abgreifen von Kreditkarteninformationen (Warenkreditbetrug).
- ▶ Phishing-Angriffe im Bereich Online-Banking.

- ▶ Big Bucks:

- ▶ (Zero-Day)-Exploits

Angriffsarten - Beispiele

- ▶ Ungezielte Angriffe z.B. über Massen-eMails, mit denen Viren, Würmer und Trojaner versandt oder Phishing-Angriffe durchgeführt werden.
- ▶ Gezielte Angriffe, z.B. zur Sabotage und Spionage, die auf bestimmte Institutionen gerichtet sind (DDoS-Angriffe auf staatliche Infrastrukturen).
- ▶ Skalpelartige Angriffe, z.B. gezielte Sabotage auf bestimmte IT-Systeme (Beispiel Stuxnet) oder auf Zertifikatediensteanbieter (Beispiel Fälschung von SSL-Zertifikaten der niederländischen Firma DigiNotar im Juli 2011)

Informationssysteme

Informationssysteme sind offene und vernetzte Systeme

- ▶ Sie stehen über ihre Schnittstellen beliebigen zunächst unbekannten Prozeßpartnern zur Verfügung.
- ▶ Kommunikation erfolgt über Nachrichten und mittlerweile in der Regel über komplexe Netzwerke.

Das birgt Gefahrenquellen:

- ▶ Angriff auf Objekte und Ressourcen durch Sicherheitsverletzungen.
- ▶ Angriff auf Nachrichten durch Sicherheitsverletzungen und Kommunikationsfehler/Abänderungen.

☞ Gefahr des Verlustes oder der Beschädigung von Informationen

Ziel ist der Schutz bzw. rechtmäßige Beschränkung des Zugriffs auf Informationen und Ressourcen.

Grundlegende Aufgaben in der IT-Sicherheit

Um einen nutzbaren elektronischen Geschäftsverkehr betrieben zu können ist neben einer geeigneten Infrastruktur eine verlässliche elektronische Kommunikation erforderlich. Vor Ort wird die Verlässlichkeit durch die anwesenden Personen verkörpert, Identität ggf. durch Ausweise verifiziert. Aufgabe von IT-Sicherheit ist

Grundlegende Aufgaben in der IT-Sicherheit

Um einen nutzbaren elektronischen Geschäftsverkehr betrieben zu können ist neben einer geeigneten Infrastruktur eine verlässliche elektronische Kommunikation erforderlich. Vor Ort wird die Verlässlichkeit durch die anwesenden Personen verkörpert, Identität ggf. durch Ausweise verifiziert. Aufgabe von IT-Sicherheit ist

- ▶ **Authentizität:** Der Sender einer Nachricht ist derjenige der er vorgibt zu sein.
- ▶ **Integrität:** Der Inhalt einer Nachricht ist derjenige, den der Emittent tatsächlich verschickt hat.
- ▶ **Vertraulichkeit:** Der Inhalt einer Nachricht bleibt vor dritten verborgen, also nicht semantisch erkennbar.
- ▶ **Verbindlichkeit:** Das versenden einer Nachricht oder die Durchführung einer Aktion kann nachgewiesen werden.
- ▶ **Verfügbarkeit:** Erreichbarkeit und Nutzbarkeit von Informationssystemen
- ▶ **Privatsphäre:** Umgang mit personenbezogenen Daten nur für vorgegebene Zwecke.

Gefahrenabwehr

durch Beschränkung des Zugriffs auf Informationen und Ressourcen auf berechtigte Prinzipale. Prinzipal ist die Zuordnung von Berechtigungen zu Aufruf und Resultat (Prinzipale können also Benutzer oder Prozesse sein).

Gefahrenabwehr

durch Beschränkung des Zugriffs auf Informationen und Ressourcen auf berechtigte Prinzipale. Prinzipal ist die Zuordnung von Berechtigungen zu Aufruf und Resultat (Prinzipale können also Benutzer oder Prozesse sein). Klassifizierung von Sicherheitsbedrohungen:

- ▶ **Lecks:** Erlangung von Informationen durch unberechtigten Empfänger bzw. Dritte.
- ▶ **Intrigieren (Tampering):** Unberechtigte Veränderung von Informationen.
- ▶ **Vandalismus:** Anonyme Störung des regulären Betriebs eines Systems.

Bedrohung von Objekten

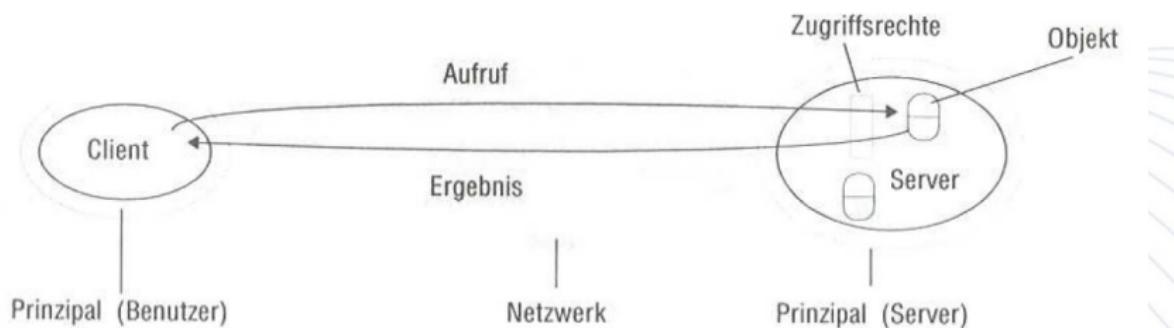
Schwachstelle: Clientseitiger entfernter (remote) Zugriff auf Objekte und Ressourcen von Servern.

Bedrohung von Objekten

Schwachstelle: Clientseitiger entfernter (remote) Zugriff auf Objekte und Ressourcen von Servern.

Behebung: Schutz gegen unerlaubten Zugriff durch (→ Vergabe von Zugriffsrechten):

- ▶ Server prüft (1) die Identität des Prinzipals (Benutzer) und (2) Berechtigung des Zugriffs auf angeforderte Objekte/Ressourcen
- ▶ Client prüft die Identität des Prinzipals (Server)



Bedrohung von Prozessen

Informationsverarbeitungsprozesse reagieren auf eingehende Nachrichten als Anforderungen (serverseitig) oder als geliefertes Resultat (clientseitig) eines Dienstes.

Bedrohung von Prozessen

Informationsverarbeitungsprozesse reagieren auf eingehende Nachrichten als Anforderungen (serverseitig) oder als geliefertes Resultat (clientseitig) eines Dienstes.

- ▶ Kommunikationsprotokolle wie das Internetprotokoll (IP) nehmen Adresse des Emittenten auf.
 - ☠ Ein Angreifer kann diese aber leicht korrumpern.

Bedrohung von Prozessen

Informationsverarbeitungsprozesse reagieren auf eingehende Nachrichten als Anforderungen (serverseitig) oder als geliefertes Resultat (clientseitig) eines Dienstes.

- ▶ Kommunikationsprotokolle wie das Internetprotokoll (IP) nehmen Adresse des Emittenten auf.
⚠ Ein Angreifer kann diese aber leicht korrumpern.
- ▶ **Schwachstelle:** Serverseitig kann bspw. ein großer Mailserver nicht für jede Anfrage die Emittentenauthentizität überprüfen (hohe Abfragefrequenz, viele Abfrager)
Behebung: Effiziente Sicherstellung, dass nur erlaubte Anfragen ausgeführt werden

Bedrohung von Prozessen

Informationsverarbeitungsprozesse reagieren auf eingehende Nachrichten als Anforderungen (serverseitig) oder als geliefertes Resultat (clientseitig) eines Dienstes.

- ▶ Kommunikationsprotokolle wie das Internetprotokoll (IP) nehmen Adresse des Emittenten auf.
⚠ Ein Angreifer kann diese aber leicht korrumpern.
- ▶ **Schwachstelle:** Serverseitig kann bspw. ein großer Mailserver nicht für jede Anfrage die Emittentenauthentizität überprüfen (hohe Abfragerfrequenz, viele Abfrager)
⚠ **Behebung:** Effiziente Sicherstellung, dass nur erlaubte Anfragen ausgeführt werden
- ▶ **Schwachstelle:** Clientseitig muß sich der Empfänger der Nachricht über den Ursprung als vom richtigen Sender versichern können.
⚠ **Behebung:** Sonst keine Verarbeitung der Antwort.

Bedrohung der Kommunikationskanäle

Nachrichten werden über Netzwerke und die vermittelnden Gateways geschickt, und können manipuliert werden, im wesentlichen durch kopieren, modifizieren oder wiederholtes Senden.

Bedrohung der Kommunikationskanäle

Nachrichten werden über Netzwerke und die vermittelnden Gateways geschickt, und können manipuliert werden, im wesentlichen durch kopieren, modifizieren oder wiederholtes Senden. Dies stellt eine Bedrohung dar, insbesondere für:

- ▶ Privatsphäre
- ▶ Integrität der Information
- ▶ Integrität des Systems

Bedrohung der Kommunikationskanäle

Nachrichten werden über Netzwerke und die vermittelnden Gateways geschickt, und können manipuliert werden, im wesentlichen durch kopieren, modifizieren oder wiederholtes Senden. Dies stellt eine Bedrohung dar, insbesondere für:

- ▶ Privatsphäre
- ▶ Integrität der Information
- ▶ Integrität des Systems

Dies schlägt sich beispielsweise wieder

- ▶ im unberechtigten lesen einer eMail.
- ▶ in der Veränderung des Inhalts einer eMail oder Protokolls durch den Angreifer.
- ▶ wiederholte Auftragsversendung nach temporärer Speicherung.

Bedrohung der Kommunikationskanäle

Nachrichten werden über Netzwerke und die vermittelnden Gateways geschickt, und können manipuliert werden, im wesentlichen durch kopieren, modifizieren oder wiederholtes Senden. Dies stellt eine Bedrohung dar, insbesondere für:

- ▶ Privatsphäre
- ▶ Integrität der Information
- ▶ Integrität des Systems

Dies schlägt sich beispielsweise wieder

- ▶ im unberechtigten lesen einer eMail.
- ▶ in der Veränderung des Inhalts einer eMail oder Protokolls durch den Angreifer.
- ▶ wiederholte Auftragsversendung nach temporärer Speicherung.

☞ **Behebung:** Abwehr aller Bedrohungsszenarien durch Installation sicherer Kanäle!

Bedrohung durch weitere Informationslecks (Information Leak)

Schwachstelle: Verwertbare Beobachtbarkeit von Nachrichten und Prozeßberechnungen.

Bedrohung durch weitere Informationslecks (Information Leak)

Schwachstelle: Verwertbare Beobachtbarkeit von Nachrichten und Prozeßberechnungen.

Inbesondere kann ein Nachrichtenfluß selbst bereits ein Leck darstellen

Auswertung von Art und Umfang der Nachrichten

- ▶ Ausspähen des Empfängers oder Senders
- ▶ Erfassung der Zugangshäufigkeit und Zugangs frequenz.

☞ **Behebung:** Als abwehrender Ansatz werden Sicherheitsprotokolle eingesetzt, die nach Sicherheitsbedürfnis der Informationen unterschiedliche Niveaus sicherer Kanäle auswählen.

Bedrohung durch Mobilien Code

Schwachstelle: Programmcode, der nicht originär auf dem auszuführendem System liegt, sondern remote zugeführt wird.

Bedrohung durch Mobilien Code

Schwachstelle: Programmcode, der nicht originär auf dem auszuführendem System liegt, sondern remote zugeführt wird.

- ▶ Im einfachsten Fall in Form eines eMail-Anhangs („Trojanisches Pferd“)
 - ☛ Maskierung schadhafter Codeelemente zum unberechtigtem Zugriff auf (berechtigte) Ressourcen
- ▶ Programmteile, die in Web-Systemen on demand geladen und ausgeführt werden (JavaScript, AJAX, Java-Applets, ActiveX-Controls...)
- ▶ Grundsätzlich bietet mobiler Code immer Basis potentieller Angriffsszenarien (gilt auch für mobile Apps)

Behebung: Isolierte Ausführung von obigen Programmcode.

Angriffstypen

- ▶  **Lauschangriff:** Unberechtigte Erstellung von Nachrichtenkopien.

Angriffstypen

- ▶ **Lauschangriff:** Unberechtigte Erstellung von Nachrichtenkopien.
- ▶ **Maskerade:** Unberechtigte Verwendung fremder Identitäten (ID).

Angriffstypen

- ▶ **Lauschangriff:** Unberechtigte Erstellung von Nachrichtenkopien.
- ▶ **Maskerade:** Unberechtigte Verwendung fremder Identitäten (ID).
- ▶ **Integrieren:** Auffangen, Veränderung und Weiterversendung von Nachrichten.

Angriffstypen

- ▶ **💀 Lauschangriff:** Unberechtigte Erstellung von Nachrichtenkopien.
- ▶ **💀 Maskerade:** Unberechtigte Verwendung fremder Identitäten (ID).
- ▶ **💀 Integrieren:** Auffangen, Veränderung und Weiterversendung von Nachrichten.
- ▶ **💀 Repetieren (Replaying):** Auffangen und wiederholtes Versenden von Nachrichten.
Funktioniert auch bei authentifizierten und verschlüsselten Nachrichten.

Angriffstypen

- ▶ **Lauschangriff:** Unberechtigte Erstellung von Nachrichtenkopien.
- ▶ **Maskerade:** Unberechtigte Verwendung fremder Identitäten (ID).
- ▶ **Integrieren:** Auffangen, Veränderung und Weiterversendung von Nachrichten.
- ▶ **Repetieren (Replaying):** Auffangen und wiederholtes Versenden von Nachrichten.
Funktioniert auch bei authentifizierten und verschlüsselten Nachrichten.
- ▶ **Verweigerung (Denial of Service):** Überflutung von Kanälen oder Ressourcen, bis diese die rechtmäßige Benutzung nicht mehr aufrechterhalten können.
- ▶ **Eindringen:** Zugriff auf Objekte und Ressourcen von Informationssystemen.

Sicherheitsmaßnahmen: Sand-Box Prinzip

Grundlegender Ansatz der isolierten Ausführung von Programmcode (vgl. mobiler Code) in kontrollierter Umgebung (→ JVM (Java Virtual Maschine), VM-Ware usw.) Eingeschränkter Zugriff (→ Sicherheitsmanager usw.) auf lokale Ressourcen, wie Dateien, Drucker, Netzwerk-Sockets...

Sicherheitsmaßnahmen: Sand-Box Prinzip

Grundlegender Ansatz der isolierten Ausführung von Programmcode (vgl. mobiler Code) in kontrollierter Umgebung (→ JVM (Java Virtual Maschine), VM-Ware usw.) Eingeschränkter Zugriff (→ Sicherheitsmanager usw.) auf lokale Ressourcen, wie Dateien, Drucker, Netzwerk-Sockets...

Darüber hinaus gehende Sicherheitsmechanismen:

- ▶ Separate Speicherung geladener Klassen von lokalen Klassen (sonst Gefahr der Verfälschung).
- ▶ Überprüfung des Bytecode auf Gültigkeit, sowie gleichermaßen der enthaltenen Anweisungen.
- ▶ Beweis der Vertrauenswürdigkeit mobilen Codes bzw. Codesigning

Problem: Aufbau der Umgebung einer Programmausführung inhärent unsicher. Somit erforderlich Ausschluß aller potentiellen Fehlerquellen.

Sicherheitsmaßnahmen: Sand-Box Prinzip

Grundlegender Ansatz der isolierten Ausführung von Programmcode (vgl. mobiler Code) in kontrollierter Umgebung (→ JVM (Java Virtual Maschine), VM-Ware usw.) Eingeschränkter Zugriff (→ Sicherheitsmanager usw.) auf lokale Ressourcen, wie Dateien, Drucker, Netzwerk-Sockets...

Darüber hinaus gehende Sicherheitsmechanismen:

- ▶ Separate Speicherung geladener Klassen von lokalen Klassen (sonst Gefahr der Verfälschung).
- ▶ Überprüfung des Bytecode auf Gültigkeit, sowie gleichermaßen der enthaltenen Anweisungen.
- ▶ Beweis der Vertrauenswürdigkeit mobilen Codes bzw. Codesigning

Problem: Aufbau der Umgebung einer Programmausführung inhärent unsicher. Somit erforderlich Ausschluß aller potentiellen Fehlerquellen.

Sicherheitsmaßnahme: Sicherer Kanal

Kommunikationsstrecke zur Verbindung eines Paares von Prozessen im Auftrag jeweils eines Prinzipals.

Verschlüsselung und Authentifizierung als Sicherheitstechniken in der Dienstschicht existierender Kommunikationsschichten (als Kanal wird hier jede Kommunikationsverbindung unabhängig ihrer technischen Realisation verstanden).

Sicherheitsmaßnahme: Sicherer Kanal

Kommunikationsstrecke zur Verbindung eines Paares von Prozessen im Auftrag jeweils eines Prinzipals.

Verschlüsselung und Authentifizierung als Sicherheitstechniken in der Dienstschicht existierender Kommunikationsschichten (als Kanal wird hier jede Kommunikationsverbindung unabhängig ihrer technischen Realisation verstanden).

- ▶ Jeder Prozeß kennt die Identität des Prinzipals des anderen Prozesses.
- ▶ Es wird die Vertraulichkeit und Integrität der übertragenen Nachrichten (Daten) sicher gestellt.
- ▶ Jede Nachricht enthält einen physischen oder logischen Zeitstempel.
→ Verhinderung wiederholter Übertragung oder erneuter Sortierung.

Sicherheitsmaßnahme: Sicherer Kanal

Kommunikationsstrecke zur Verbindung eines Paares von Prozessen im Auftrag jeweils eines Prinzipals.

Verschlüsselung und Authentifizierung als Sicherheitstechniken in der Dienstschicht existierender Kommunikationsschichten (als Kanal wird hier jede Kommunikationsverbindung unabhängig ihrer technischen Realisation verstanden).

- ▶ Jeder Prozeß kennt die Identität des Prinzipals des anderen Prozesses.
- ▶ Es wird die Vertraulichkeit und Integrität der übertragenen Nachrichten (Daten) sicher gestellt.
- ▶ Jede Nachricht enthält einen physischen oder logischen Zeitstempel.
→ Verhinderung wiederholter Übertragung oder erneuter Sortierung.

Typische Beispiele:

VPN (Virtual Private Network) oder TLS (Transport Layer Security, formerly known as SSL, Secure Socket Layer, SSLv3)

Einschub: Kryptologie

Kryptologie = Kryptographie + Kryptoanalyse

- ▶ Kryptographie ist die Disziplin, die sich mit dem Design von Verschlüsselungsverfahren beschäftigt.
Sicherheit beruht auf dem geheimen Schlüssel ohne dessen Kenntnis eine Entschlüsselung nicht möglich ist.

Einschub: Kryptologie

Kryptologie = Kryptographie + Kryptoanalyse

- ▶ Kryptographie ist die Disziplin, die sich mit dem Design von Verschlüsselungsverfahren beschäftigt.
Sicherheit beruht auf dem geheimen Schlüssel ohne dessen Kenntnis eine Entschlüsselung nicht möglich ist.
- ▶ Die Kryptoanalyse beschäftigt sich mit dem Brechen von Verschlüsselungsverfahren bzw. das Entschlüsseln von Nachrichten ohne vorherige Kenntnis des geheimen Schlüssels.

Zwei Typen von Verfahren

- ▶ Symmetrisches Verfahren: Verwendung eines gemeinsamen geheimen Schlüssels.
- ▶ Asymmetrisches Verfahren: Verwendung eines Paares aus öffentlichem und geheimen Schlüssel.

Symmetrische Verschlüsselung

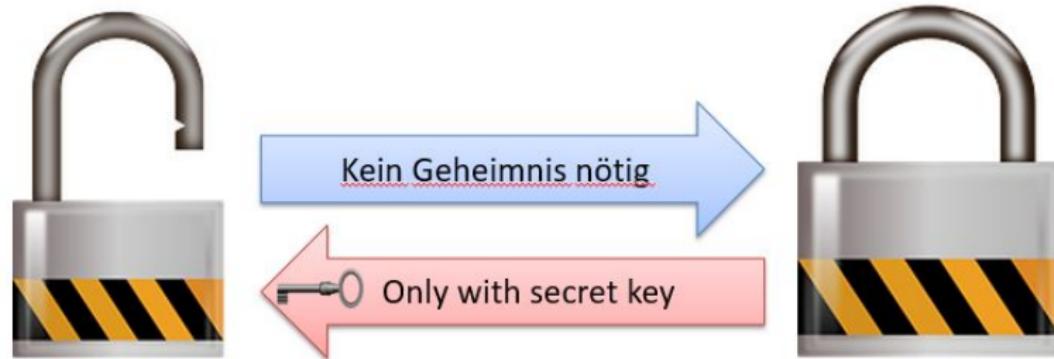


Symmetrische Verschlüsselung

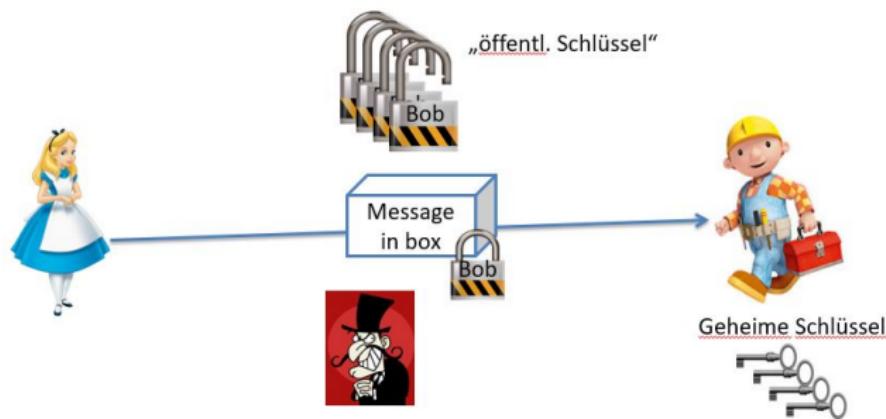


- ▶ Sender und Empfänger benutzen beiden denselben Schlüssel.
→ Chiffrierung und Dechiffrierung verwenden denselben Schlüssel.
- ▶ Notwendig und problematisch: Der Schlüssel muss sicher ausgetauscht und übertragen werden.
- ▶ Bei unterschiedlichem Schlüssel für jedes Kommunikationspaar, d.h. bei paarweiser vertraulicher Kommunikation, ergeben sich Komplexitätsprobleme beim Schlüsselaustausch.

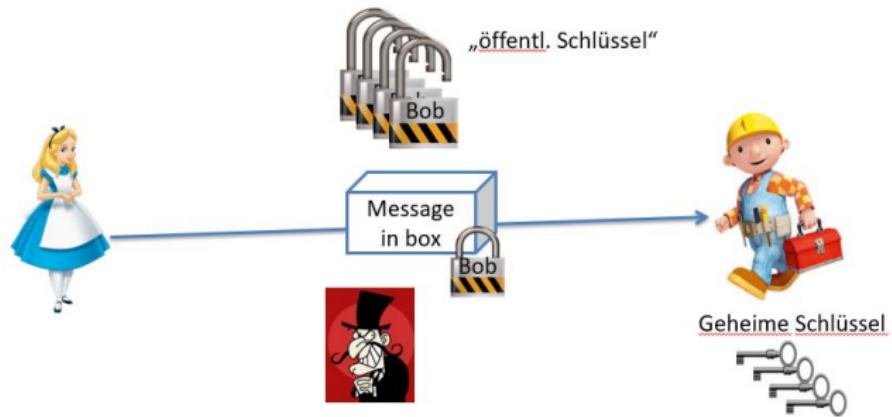
Grundprinzip der Public-Key Kryptographie (asymmetrische Verschlüsselung)



Asymmetrische Verschlüsselung



Asymmetrische Verschlüsselung



- ▶ Schlüssel besteht aus einem privaten (Private Key) und einem öffentlichen Teil (Public Key).
→ Es gibt also zwei Schlüssel.
- ▶ Der private Schlüssel ist geheim und verbleibt beim Empfänger.
- ▶ Der öffentliche Schlüssel kann über eine unsichere Leitung ausgetauscht werden.

Sicherheitsmaßnahme: Authentifikation (symmetrisch)

Ziel: Überprüfung der Identität des Emittenten

Grundlage: Verwendung gemeinsamer Geheimnisse und Verschlüsselung

Ansatz: Aufnahme eines (hinreichend großen) verschlüsselten Teils (Authentifizierungsteil) mit dem eigentlichen Inhalt einer Nachricht in die Gesamtnachricht. Voraussetzung ist ein beiden bekannter privater Schlüssel.

Sicherheitsmaßnahme: Authentifikation (symmetrisch)

Ziel: Überprüfung der Identität des Emittenten

Grundlage: Verwendung gemeinsamer Geheimnisse und Verschlüsselung

Ansatz: Aufnahme eines (hinreichend großen) verschlüsselten Teils (Authentifizierungsteil) mit dem eigentlichen Inhalt einer Nachricht in die Gesamtnachricht. Voraussetzung ist ein beiden bekannter privater Schlüssel.

- ▶ Anfrage an einen Datei-Server kann Identität des Prinzipals und der Datei sowie Datum und Uhrzeit der Anfrage enthalten.
- ▶ Anfrage bewirkt, dass der Server den Authentifizierungsteil liest und abgleicht. Dann erst folgt die Aufforderung zum eigentlichen Zugriff auf die Datei.
- ▶ Allgemein werden Prüfsummen oder andere erwartete Werte verwendet.

Sicherheitsmaßnahme: Digitale Signatur

Sicherstellung, daß ein Dokument eine unverfälschte Kopie eines vom unterzeichnenden Benutzers erzeugten Dokumentes ist.
→ Digitales Abbild analoger Unterschriften.

Sicherheitsmaßnahme: Digitale Signatur

Sicherstellung, daß ein Dokument eine unverfälschte Kopie eines vom unterzeichnenden Benutzers erzeugten Dokumentes ist.

→ Digitales Abbild analoger Unterschriften.

- ▶ Unwiderrufliches hinzufügen eines exklusiven Geheimnisses des Unterzeichnenden.
- ▶ Verschlüsselung der Nachricht oder einer komprimierten Form davon.

→ Hash-Funktion:

Funktion, die auf Basis des Inhalts einer Nachricht einen Kontrollwert (Hash-Wert) berechnet.

→ nicht injektive Abbildung die möglichst injektiv ist.

Sicher, wenn die Wahrscheinlichkeit eines identisches Bildpunktes hinreichend klein ist.

☞ Die digitale Signatur ermöglicht eine asymmetrische Authentifikation.

Sicherheitsmaßnahme: Zertifikate

- ▶ Stellen eine Art elektronischen Ausweises dar, mit dem Identität und Echtheit einer Person oder eines Dokumentes nachgewiesen werden können.
- ▶ Sie sind Teil der sog. Public Key Infrastruktur. Darunter werden desweiteren Trustcenter, Standards, Richtlinien, Software und technisches Know-How zusammengefaßt.
- ▶ Grundlage ist das Gesetz über Rahmenbedingungen für elektronische Signaturen (SigG).

Trustcenter

- ▶ Sie sind vertrauenswürdige Einrichtungen, welche eine personenbezogene Zertifikaterstellung in besonders gesicherten und vertrauenswürdigen Räumen durchführen (→ BSI).
- ▶ sind exklusiv autorisiert Zertifikate auszustellen.

Einsatz von Zertifikaten

Identifikation des Absenders. Das Zertifikat fungiert dabei als elektronischer Ausweis.

Exemplarisches Verwendungsprinzip:

Einsatz von Zertifikaten

Identifikation des Absenders. Das Zertifikat fungiert dabei als elektronischer Ausweis.

Exemplarisches Verwendungsprinzip:

1. Ein eCommerce-Anbieter schickt einem Konsumenten eine Nachricht zusammen mit einem Zertifikat.
2. Der Konsument nimmt den zugehörigen öffentlichen Schlüssel und überprüft das Zertifikat. Bei Erfolg ist:
 - ▶ Sender ist als solcher ausgewiesen.
 - ▶ Der öffentliche Schlüssel nicht korrumptiert bzw. passt zum Zertifikat.

Einsatz von Zertifikaten

Identifikation des Absenders. Das Zertifikat fungiert dabei als elektronischer Ausweis.

Exemplarisches Verwendungsprinzip:

1. Ein eCommerce-Anbieter schickt einem Konsumenten eine Nachricht zusammen mit einem Zertifikat.
2. Der Konsument nimmt den zugehörigen öffentlichen Schlüssel und überprüft das Zertifikat. Bei Erfolg ist:
 - ▶ Sender ist als solcher ausgewiesen.
 - ▶ Der öffentliche Schlüssel nicht korrumptiert bzw. passt zum Zertifikat.

Sicherheitsaspekte:

- ▶ Das Abfangen und Korrumpern des öffentlichen Schlüssels verhindert das Entschlüsseln des Zertifikates beim Konsumenten (→ keine Identifikation möglich)
- ▶ Das Abfangen einer Nachricht des Konsumenten mit nicht korrumpiertem öffentlichen Schlüssel ist nutzlos, da der private Schlüssel sich beim eCommerce-Anbieter befindet

Sicherheitsmaßnahmen: Webseitensicherheit

Transport Layer Security (TLS)

Client-Server-basiertes Protokoll, welches einen sicheren Datenaustausch im Internet zwischen Web-Server und Web-Client ermöglicht. Es nutzt symmetrische, asymmetrische kryptographische Mechanismen und Hash-Funktionen.

Sicherheitsmaßnahmen: Webseitensicherheit

Transport Layer Security (TLS)

Client-Server-basiertes Protokoll, welches einen sicheren Datenaustausch im Internet zwischen Web-Server und Web-Client ermöglicht. Es nutzt symmetrische, asymmetrische kryptographische Mechanismen und Hash-Funktionen.

Erkennungsmerkmal:

Die URL beginnt mit `https://...`

Prinzip:

1. Zu Beginn einer Kommunikation wird eine gesicherte Verbindung durch wechselseitiges Identifizieren aufgebaut.
2. Danach erfolgt der eigentliche Datenaustausch über das TLS Record Protocol.

Sicherheitsmaßnahmen: GNU Privacy Guard (GnuPG)

Neben anderen eines der weltweit besten und sichersten Verschlüsselungstools (**bei richtiger Versionswahl!!**).

Sicherheitsmaßnahmen: GNU Privacy Guard (GnuPG)

Neben anderen eines der weltweit besten und sichersten Verschlüsselungstools (**bei richtiger Versionswahl!!**).

Sicherheitsmaßnahmen: GNU Privacy Guard (GnuPG)

Neben anderen eines der weltweit besten und sichersten Verschlüsselungstools (**bei richtiger Versionswahl!!**).

Open-Source-System:

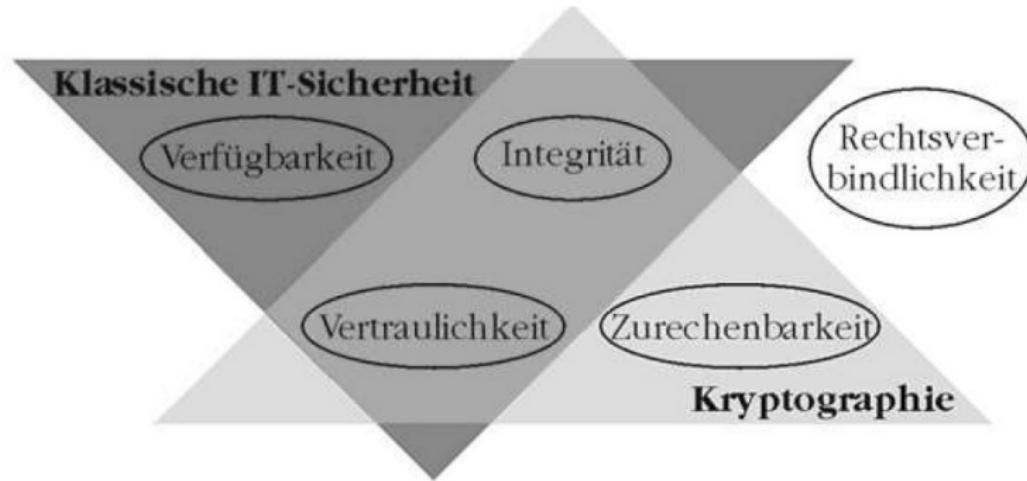
Quellcode liegt vollständig offen und kann somit im Prinzip auf Bugs und Back Doors (Hintertüren) untersucht werden.

- ▶ Leicht bedienbar und installierbar.
- ▶ In Taskleistentool für beliebige eMail-Clients integriert.
- ▶ Plug-In für Outlook ebenfalls verfügbar.
- ▶ Weitere Softwaremodule verfügbar. Softwaremodule und weiterführende Informationen unter:
<https://gnupg.org/>

Übersicht: Aufgaben der IT-Sicherheit

Verlässlichkeit := Verfügbarkeit + Integrität + Vertraulichkeit

Beherrschbarkeit := Zurechenbarkeit + Rechtsverbindlichkeit



Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundsatz

Safety und Security

- ▶ Im Deutschen bedeutet Sicherheit im Allgemeinen ein Zustand, der frei von Gefahren ist. Bei einem IT-System können sich die Gefahren sowohl auf die Betriebssicherheit als auch auf die Sicherheit vor beabsichtigten Angriffen beziehen.
- ▶ In der englischen Sprache gibt dazu die Begriffe Safety und Security.

Die Fachliteratur ist nicht immer einheitlich. Wir definieren hier:

Definition

Safety bezieht sich meist auf die Funktionssicherheit und Ausfallsicherheit von Systemen, z.B. ein Stahlgehäuse für Laptops bei Auslandseinsätzen der Bundeswehr.

Security ist Angriffssicherheit, also der Schutz vor beabsichtigten Angriffen. Dementsprechend definiert auch die International Standards Organisation (ISO) Security als die „Minimierung der Verwundbarkeit von Werten und Ressourcen“ (Eckert 2014). Z.B. Schutzmaßnahmen gegen Auslesen von Daten von Laptops.

Schwachstelle

Eine Systemeigenschaft, die Missbrauchsmöglichkeiten bietet.

Bedrohung

Schwachstelle

Eine Systemeigenschaft, die Missbrauchsmöglichkeiten bietet.

Bedrohung

Jedes mögliche Vorkommen (böswillig oder anderweitig) mit ungewünschten Effekten auf die Werte (Assets) und Ressourcen eines Computersystems.

Risiko

Schwachstelle

Eine Systemeigenschaft, die Missbrauchsmöglichkeiten bietet.

Bedrohung

Jedes mögliche Vorkommen (böswillig oder anderweitig) mit ungewünschten Effekten auf die Werte (Assets) und Ressourcen eines Computersystems.

Risiko

Produkt aus Schadenshöhe und Eintrittswahrscheinlichkeit eines Ereignisses (aus Bedrohungszenario).

Gefährdungskategorien:

- ▶ Höhere Gewalt (z.B. Hochwasser, Blitzeinschlag, globaler Stromausfall).
- ▶ Technische Fehler (z.B. defekte Datenträger, Ausfall einer Datenbank).
- ▶ Fahrlässigkeit (z.B. Nichtbeachtung von Sicherheitsmaßnahmen, ungeeigneter Umgang mit Passwörtern).
- ▶ Organisatorische Mängel (z.B. fehlende oder unzureichende Regelungen, nicht erkannte Sicherheitsvorfälle).
- ▶ Vorsätzliche Handlungen (z.B. Abhören und Manipulation von Leitungen, Schadprogramme, Diebstahl).

Schadenszenarien sind üblicherweise

- ▶ Verstoß gegen Gesetze, Vorschriften, Verträge.
- ▶ Beeinträchtigung des informationellen Selbstbestimmungsrechts.
- ▶ Beeinträchtigung der persönlichen Unversehrtheit.
- ▶ Beeinträchtigung der Aufgabenerfüllung.
- ▶ negative Innen- oder Außenwirkung.
- ▶ negative finanzielle Auswirkungen.

Eintrittswahrscheinlichkeit und Schadenshöhe lassen sich nur schwer quantifizieren:

- ▶ Für Eintrittswahrscheinlichkeit: Welche Mittel wird ein Angreifer einsetzen.
- ▶ Diese sind nicht nur abhängig von finanziellen Gewinnaussichten, sondern teilweise auch vom persönlichen Ehrgeiz (z.B. Whistleblower).
- ▶ Für Schadenshöhe: Abschätzung, welche Folgen ein Angriff hat.
- ▶ meist sind mehrere Schadensszenarien betroffen.
- ▶ und hängt von der konkreten Institution ab (für kleine Unternehmen kann ein Verlust von 100.000 Euro existenzgefährdend sein, für Konzerne nicht).

Vereinfachung Risiko

		Risiko		
		mittel	hoch	kritisch
Schadens- höhe	3			
	2	niedrig	mittel	hoch
	1	unbedeutend	niedrig	mittel
		1	2	3
Eintrittswahrscheinlichkeit				

- ▶ Risiko = Schadenshöhe * Eintrittswahrscheinlichkeit.

Die Sicherheitsaspekte/Schutzziele

- ▶ **Authentizität:** Der Sender einer Nachricht ist derjenige der er vorgibt zu sein.
- ▶ **Integrität:** Der Inhalt einer Nachricht ist derjenige, den der Emittent tatsächlich verschickt hat.
- ▶ **Vertraulichkeit:** Der Inhalt einer Nachricht bleibt vor dritten verborgen, also nicht semantisch erkennbar.
- ▶ **Verbindlichkeit:** Das versenden einer Nachricht oder die Durchführung einer Aktion kann nachgewiesen werden.
- ▶ **Verfügbarkeit:** Erreichbarkeit und Nutzbarkeit von Informationssystemen
- ▶ **Privatsphäre:** Umgang mit personenbezogenen Daten nur für vorgegebene Zwecke.

Privatsphäre wird nicht immer dazugezählt.

Relevanz von Sicherheitsaspekten

Welcher Sicherheitsaspekt ist am wichtigsten für ...

- ▶ ... die Liste der CIA-Agenten im Auslandseinsatz?
Vertraulichkeit.
- ▶ ... die Kontostände auf dem Server einer Bank?
Integrität.
- ▶ ... die Bestellung einer Ware per E-Mail?
 - ▶ Für den Verkäufer:
Authentizität der Nachricht.
Nicht-Abstreitbarkeit der Nachricht.
 - ▶ Für den Käufer:
Verfügbarkeit.

Relevanz von Sicherheitsaspekten

Welcher Sicherheitsaspekt ist am wichtigsten für ...

- ▶ ... die Liste der CIA-Agenten im Auslandseinsatz?
Vertraulichkeit.
- ▶ ... die Kontostände auf dem Server einer Bank?
Integrität.
- ▶ ... die Bestellung einer Ware per E-Mail?
 - ▶ Für den Verkäufer:
Authentizität der Nachricht.
Nicht-Abstreitbarkeit der Nachricht.
 - ▶ Für den Käufer:
Verfügbarkeit.

☞ Verschiedene Anwendungen und Parteien können unterschiedliche Sicherheitsziele haben!

Sicherheitsanalyse

Ziel der Sicherheitsanalyse ist

- ▶ die Ermittlung der Ursachen von Gefährdungen.
- ▶ die Abschätzung von deren Eintrittswahrscheinlichkeiten sowohl
 - ▶ bei der Erstellung als auch
 - ▶ beim Einsatz des Systems.

☞ Black Box Penetrationstest durch Auftraggeber bzw. White Box Penetrationstest durch Auftragnehmer.

Eine Sicherheitsanalyse umfasst in der Regel:

- ▶ **Die Validierung**, dabei wird die Verlässlichkeit, Benutzbarkeit (usability) und Regelkonformität analysiert.
- ▶ **Die Evaluierung**, dabei wird die festgestellte Qualität der technischen und organisatorischen Maßnahmen bewertet.
- ▶ **Die Zertifizierung**, dabei wird die Bewertung von Validierung oder Evaluierung bestätigt.

Anonymität und Pseudonymität

- ▶ Beispiel für Anonymität:
Geheime Abstimmung bei Wahlen: Zuordnung zwischen Wahlzettel und Wähler ist nicht möglich.
- ▶ Beispiel für Pseudonymität:
E-Mail Adresse: Kommunikationspartner kennen nicht die reale Identität (aber der Dienstanbieter).

Rechtliche Rahmen

IT-Sicherheit sollte nicht nur im Eigeninteresse einer Institution umgesetzt werden.

Vielfach fordern Gesetze die Umsetzung geeigneter IT-Sicherheitsmaßnahmen, z.B.:

- ▶ Bundesdatenschutzgesetz und Datenschutzgesetze der Bundesländer.
- ▶ Gesetz zur Kontrolle und Transparenz im Unternehmensbereich.
- ▶ Teledienstedatenschutzgesetz.
- ▶ Telekommunikationsgesetz.

Datenschutz

Mit Datenschutz wird der Schutz personenbezogener Daten vor Missbrauch durch Dritte bezeichnet.

- ▶ Die Privatsphäre jedes Einzelnen soll geschützt werden.
- ▶ Rechtlicher Ausgangspunkt ist das Grundrecht auf informationelle Selbstbestimmung.
- ▶ Grundidee ist, dass der Einzelne die Möglichkeit haben soll, selbst zu bestimmen, wer bei welcher Gelegenheit welche Informationen über ihn erhält.

Bundesdatenschutzgesetz |

Grundpfeiler des Bundesdatenschutzgesetzes (BDSG) sind die Prinzipien

- ▶ Datenvermeidung und Datensparsamkeit: Bei der Datenverarbeitung dürfen nur so viele personenbezogene Daten gesammelt werden, wie für die jeweilige Anwendung unbedingt notwendig sind (§ 3a BDSG).
- ▶ Das allgemeine Verbot der Verarbeitung personenbezogener Daten: Grundsätzlich dürfen personenbezogene Daten nur mit Einwilligung der Betroffenen oder aufgrund gesetzlicher Gestattung verarbeitet werden (§ 4 Abs. 1 BDSG).

Bundesdatenschutzgesetz II

Das Bundesdatenschutzgesetz (§ 9) verpflichtet Daten verarbeitende Stellen, geeignete technische und organisatorische Maßnahmen zum Schutz der erhobenen Daten zu treffen (Datensicherheit). Personenbezogene Daten dürfen

- ▶ nur Befugten zugänglich sein, da sie vertraulich sind.
- ▶ nicht unbemerkt verändert oder gelöscht werden.

Weitere Gesetze

Teledienstedatenschutzgesetz (Auszug):

- ▶ § 4 Abs. 4 Nr. 3 besagt: Der Diensteanbieter hat durch technische und organisatorische Vorkehrungen sicherzustellen, dass der Nutzer Teledienste gegen Kenntnisnahme Dritter geschützt in Anspruch nehmen kann.
- ▶ Anwendungsbereich:
Internet- und E-Mailprovider wie Deutsche Telekom, usw.

Telekommunikationsgesetz (Auszug):

- ▶ Anmeldepflicht: Das Erbringen von Telekommunikationsleistungen ist frei und lediglich anmeldepflichtig. Eine Genehmigung ist nicht erforderlich (§ 6 TKG).
- ▶ Abhören: Das unbefugte Abhören von Nachrichten über Telekommunikationswege wird nach § 148 Abs. 1 Satz 1 TKG mit Freiheitsstrafe bis zu zwei Jahren oder Geldstrafe bestraft. Ebenso wird bestraft, wer unzulässige Sendeanlagen besitzt, herstellt, vertreibt oder einführt (§ 148 Abs. 1 Satz 2 TKG).

Identity Management

Das Identity Management umfaßt folgende Aufgaben:

- ▶ die Repräsentation von Personen oder Objekten (allgemeiner: Entitäten) in einem System,
- ▶ die Prüfung, ob eine Entität auch genau jene ist, für die sie sich ausgibt, und
- ▶ die Prüfung, welche Rechte mit dieser Person (Entität) verknüpft sind.

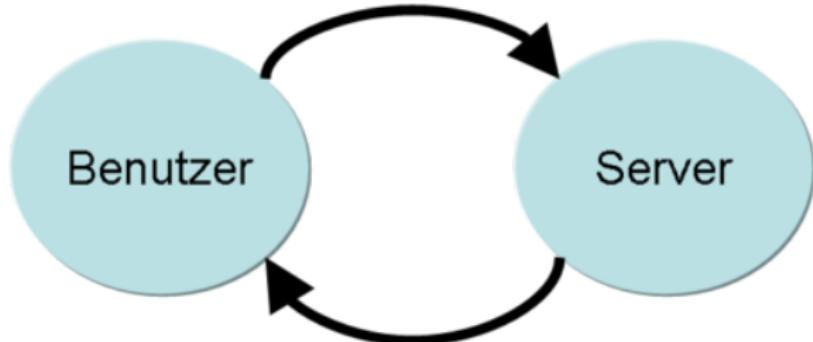
Identität

- ▶ Die Identität ist im Prinzip die (technische) Repräsentation einer Entität, wobei eine Entität eine Person, aber auch ein Objekt sein kann.

Authentifikationsverfahren

- ▶ Unter Authentisierung wird der Nachweis der behaupteten Identität durch ein Subjekt verstanden.
- ▶ Authentifizierung bedeutet die Prüfung des Nachweises. Dabei wird zwischen Verifikation und Identifikation unterschieden.
 - ▶ Verifikation bedeutet einen 1:1-Vergleich. (z.B. Passbild mit Person).
 - ▶ Identifikation bedeutet einen 1:n Vergleich (Person mit allen Bildern in einer DB)
- ▶ Den Prozeß, der sowohl die Authentifikation als auch die Authentisierung umfasst, bezeichnet man als Authentifikationsverfahren.

authentisiert sich am



authentifiziert den

Abbildung: Quelle: Wikipedia

Authentifikationstechniken



"This site wants a two-factor authentication.
A retina scan and a urine sample."



Authentifikation durch

Authentifikationstechniken



"This site wants a two-factor authentication.
A retinal scan and a urine sample."



Authentifikation durch

- ▶ Wissen (z.B. Passworte, PINs, kryptographische Schlüssel)
- ▶ Besitz (z.B. Smartcard, USB-Token, SIM-Karte im Mobiltelefon)
- ▶ Merkmale (z.B. Biometrie: Fingerabdruck, Iris etc.)

Mehrfaktor-Authentifikation:

Beispiel: 2-Faktor-Authentifikation beim Mobiltelefon

Authentifikationstechniken



"This site wants a two-factor authentication.
A retinal scan and a urine sample."



Authentifikation durch

- ▶ Wissen (z.B. Passworte, PINs, kryptographische Schlüssel)
- ▶ Besitz (z.B. Smartcard, USB-Token, SIM-Karte im Mobiltelefon)
- ▶ Merkmale (z.B. Biometrie: Fingerabdruck, Iris etc.)

Mehrfaktor-Authentifikation:

Beispiel: 2-Faktor-Authentifikation beim Mobiltelefon

1. Authentifikation durch Wissen (PIN) gegenüber SIM-Karte
2. und Besitz der SIM-Karte (geheimer Schlüssel Ki).
3. SIM-Karte authentifiziert sich gegenüber Netz mit Ki.

Authentifikationstechniken



"This site wants a two-factor authentication.
A retinas scan and a urine sample."



Authentifikation durch

- ▶ Wissen (z.B. Passworte, PINs, kryptographische Schlüssel)
- ▶ Besitz (z.B. Smartcard, USB-Token, SIM-Karte im Mobiltelefon)
- ▶ Merkmale (z.B. Biometrie: Fingerabdruck, Iris etc.)

Mehrfaktor-Authentifikation:

Beispiel: 2-Faktor-Authentifikation beim Mobiltelefon

1. Authentifikation durch Wissen (PIN) gegenüber SIM-Karte
2. und Besitz der SIM-Karte (geheimer Schlüssel Ki).
3. SIM-Karte authentifiziert sich gegenüber Netz mit Ki.

Beispiel: Online Dienste, Online Banking

- ▶ Authentifikation durch Wissen: Einloggen mittels PIN und durch Besitz: Hardware-Token, Personalausweis.

Autorisierung

- ▶ Die erfolgreiche Zuweisung solcher Rechte zu Identitäten bezeichnet man als Autorisierung (engl. authorization) und erfolgt normalerweise nach einer erfolgreichen Authentifizierung.
- ▶ Beispiel: Ein erfolgreich authentifizierter Nutzer eines Betriebssystems ist für den Zugriff auf sein Home-Verzeichnis und die Benutzung eines Netzwerkdruckers autorisiert, nicht aber für den Zugriff auf Systemdateien.

Einseitige vs. Wechselseitige Authentifikation



Einseitige Authentifikation

- ▶ Eine Partei identifiziert sich gegenüber einer anderen
- ▶ Beispiele:
 - ▶ Benutzer gegenüber PC.
 - ▶ Handy gegenüber Netz-Provider bei GSM (IMSI Catcher!).
 - ▶ WWW-Server gegenüber Nutzer.

Einseitige vs. Wechselseitige Authentifikation



Einseitige Authentifikation

- ▶ Eine Partei identifiziert sich gegenüber einer anderen
- ▶ Beispiele:
 - ▶ Benutzer gegenüber PC.
 - ▶ Handy gegenüber Netz-Provider bei GSM (IMSI Catcher!).
 - ▶ WWW-Server gegenüber Nutzer.

Wechselseitige Authentifikation

- ▶ Zwei Parteien authentifizieren sich gegenseitig.
- ▶ Beispiele:
 - ▶ Handy vs. Netz bei UMTS und LTE.
 - ▶ Webdienste bei entsprechender Verwendung von Zertifikaten.

Authentifikation durch Wissen: Passwort

Gängigste Methode: Passwörter

- ▶ Speicherung der Passwörter üblicherweise als Hash-Wert
- ▶ Beispiel: UNIX/Linux: /etc/passwd und /etc/shadow
- ▶ Shadow-Passwörter: Hash von Passwort und Salt

Authentifikation durch Wissen: Passwort

Gängigste Methode: Passwörter

- ▶ Speicherung der Passwörter üblicherweise als Hash-Wert
- ▶ Beispiel: UNIX/Linux: /etc/passwd und /etc/shadow
- ▶ Shadow-Passwörter: Hash von Passwort und Salt

```
cat etc/shadow
```

```
stud:$1$hVmMb2Ix$aAGVIJKYfBKavZt5Ee9ZL/:15050:0:99999:7:::
```

- ▶ 1. Feld: Benutzername,
- ▶ 2. Feld: gehashtes Passwort, ist hier ein ! oder ein * eingetragen, kann sich der Nutzer nicht am System anmelden.
- ▶ 3. Feld: Tag der letzten Passwortänderung In diesem Feld wird die Anzahl der Tage seit dem 1.1.1970 bis zum Tag der letzten Passwortänderung gespeichert.
- ▶ :

Probleme:

- ▶ „Schwache“ Passwörter, die erraten werden können (Wörterbuchangriff).
 - ▶ Benutzer verwenden gleiches Passwort bei verschiedenen Servern.
 - ▶ Manche UNIX-Tools (telnet, ftp) übertragen das Passwort im Klartext!
- ☞ Authentifikationsverfahren, die die Übertragung eines Passwortes nicht erfordern notwendig, z.B. Challenge-Response Verfahren.

Passwortcracker der Hochschule Emden/Leer



Abbildung: Cudabasierter Passwortcracker der HEL

Passwortcracker der Hochschule Emden/Leer

Eckdaten:

- ▶ Seit ca. 5 Jahren im Einsatz.
- ▶ 10 Nvidia GTX-590 Grafikkarten.
- ▶ 5 Knoten mit 8 Gb Speicher zur Lastenverteilung.
- ▶ Für Dictionaryattacks stehen ca. 3 Gb pro Grafikkarte zur Verfügung.

Passwortcracker der Hochschule Emden/Leer

Eckdaten:

- ▶ Seit ca. 5 Jahren im Einsatz.
- ▶ 10 Nvidia GTX-590 Grafikkarten.
- ▶ 5 Knoten mit 8 Gb Speicher zur Lastenverteilung.
- ▶ Für Dictionaryattacks stehen ca. 3 Gb pro Grafikkarte zur Verfügung.

💀 Mit diesem Cracker sind ca. 2^{44} Bitoperationen in der Sekunde möglich.

Ein Passwort mit 8 Zeichen bestehend aus [A-Za-z0-9] kann damit bei Anwendungen, bei denen der Check aufs korrekte Passwort sehr schnell durchführbar (immer noch durchaus der Fall) ist, in weniger als eine halbe Stunde gefunden werden.

Wie speichere ich ein Passwort?

Clientseitige populäre Negativbeispiele:

- ▶ Offen notiert.
- ▶ Klebezettel auf dem Monitor.
- ▶ Offene Speicherung auf Firmenrechnern.

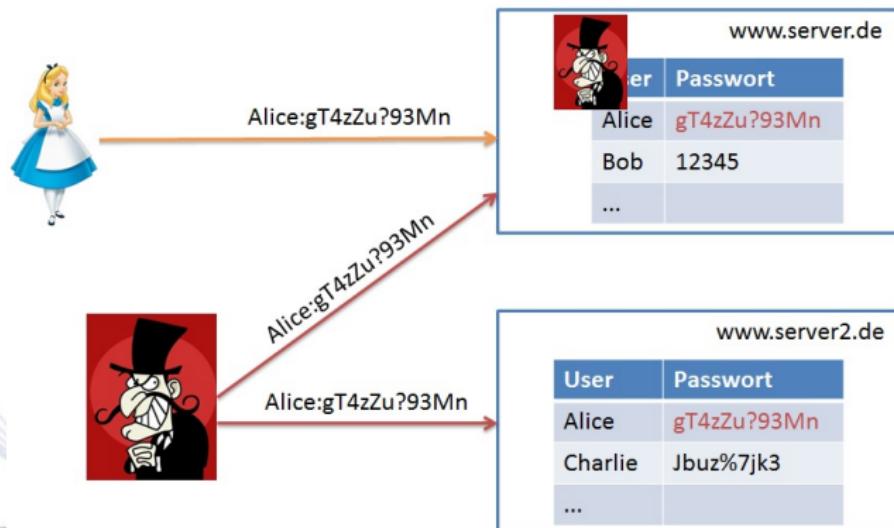
Wie speichere ich ein Passwort?

Clientseitige populäre Negativbeispiele:

- ▶ Offen notiert.
- ▶ Klebezettel auf dem Monitor.
- ▶ Offene Speicherung auf Firmenrechnern.

Das serverseitige populäre Negativbeispiel:

- ▶ Passwortleaks durch offene Speicherung auf Firmenservern.



Passwortleaks

Meldung vom 07.07.2017:

„Gestohlene Daten tauchen immer wieder im Internet auf. Jetzt hat das Bundeskriminalamt in einem Hacker-Forum einen riesigen Fund gemacht. 500 Millionen E-Mail-Adressen mit dazugehörigen Passwörtern von unterschiedlichen Onlineplattformen wurden entdeckt. Jeder Internetnutzer sollte jetzt aktiv werden....“

Ausweg:

- ▶ Vermeidung von offen notierten Passwörtern und Klebezetteln auf Monitoren.

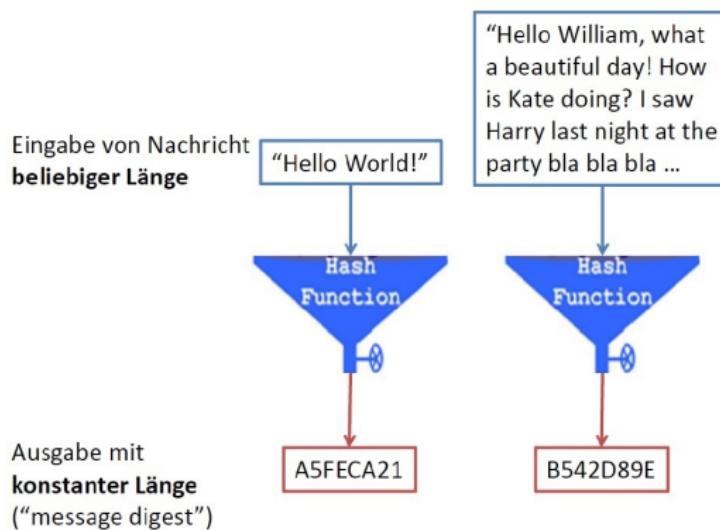
Ausweg:

- ▶ Vermeidung von offen notierten Passwörtern und Klebezetteln auf Monitoren.
- ▶ Speichere Passworte so, dass
 - ▶ geprüft werden kann, ob ein gegebenes Passwort „korrekt“ ist.
 - ▶ ein Angreifer, **der alle auf dem Server gespeicherten Daten lesen kann (inkl. Passwort-Datenbank)**, nicht in der Lage ist diese Passworte zum Login zu verwenden

Ausweg:

- ▶ Vermeidung von offen notierten Passwörtern und Klebezetteln auf Monitoren.
- ▶ Speichere Passworte so, dass
 - ▶ geprüft werden kann, ob ein gegebenes Passwort „korrekt“ ist.
 - ▶ ein Angreifer, **der alle auf dem Server gespeicherten Daten lesen kann (inkl. Passwort-Datenbank)**, nicht in der Lage ist diese Passworte zum Login zu verwenden
- ⌚ Werkzeug: Kryptographische Hashfunktionen.

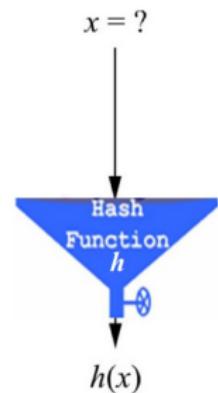
Kryptologische Grundlagen I: Kryptographische Hashfunktionen



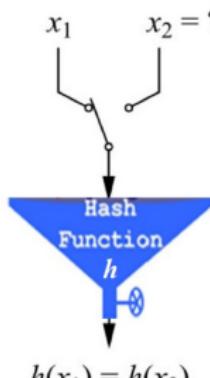
Definition

Sei n eine natürliche Zahl. Eine kryptographische Hashfunktion ist eine Funktion $H : \{0, 1\}^* \rightarrow \{0, 1\}^n$. Dabei bezeichnet $\{0, 1\}^*$ die Menge der Bitstrings beliebiger, aber endlicher Länge.

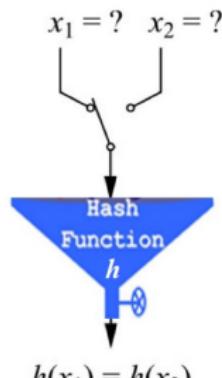
Sicherheitseigenschaften von Hashfunktionen



preimage resistance



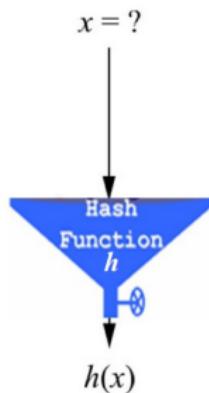
second preimage
resistance



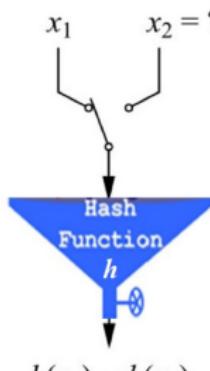
collision resistance

☞ Ideal zur sicheren Speicherung von Passwörtern.

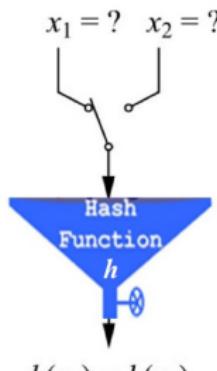
Sicherheitseigenschaften von Hashfunktionen



preimage resistance



second preimage
resistance



collision resistance

- ☞ Ideal zur sicheren Speicherung von Passwörtern.
- ☞ AuG/Krypto.

Beispiele für Hashfunktionen

Name	Ausgabe	Kommentar
MD5	128 b	<ul style="list-style-type: none">Gilt als “gebrochen” (*) (Kollisionen)Nicht mehr für neue Anwendungen geeignet
SHA-1	160 b	<ul style="list-style-type: none">Gilt als “gebrochen” (Kollisionen)Nicht mehr für neue Anwendungen geeignet
SHA-256	256 b	Keine nichttrivialen Angriffe
SHA-3	224 b, 256 b, 384 b, 512 b	<ul style="list-style-type: none">Gewinner der SHA-3 CompetitionGute Wahl für neue Anwendungen

Die Entropie

Definition (Entropie)

Ist $\{Z_1, \dots, Z_N\}$ der Passwort/Schlüsselraumes, so ist $\log_2(N)$ die Entropie des Passwortes/Schlüssels bei uniformverteilter zufälliger Auswahl aus dieser Menge.

Die Entropie

Definition (Entropie)

Ist $\{Z_1, \dots, Z_N\}$ der Passwort/Schlüsselraumes, so ist $\log_2(N)$ die Entropie des Passwortes/Schlüssels bei uniformverteilter zufälliger Auswahl aus dieser Menge. Ein Brute-Force Angriff besitzt für diese Entropie eine äußere Komplexität von $2^{\log_2(N)}$. Bei der äußeren Komplexität werden die bei jedem Passwort/Schlüsseltest anfallenden Operationen mit Aufwand 1 gezählt.

Die Entropie

Definition (Entropie)

Ist $\{Z_1, \dots, Z_N\}$ der Passwort/Schlüsselraumes, so ist $\log_2(N)$ die Entropie des Passwortes/Schlüssels bei uniformverteilter zufälliger Auswahl aus dieser Menge. Ein Brute-Force Angriff besitzt für diese Entropie eine äußere Komplexität von $2^{\log_2(N)}$. Bei der äußeren Komplexität werden die bei jedem Passwort/Schlüsseltest anfallenden Operationen mit Aufwand 1 gezählt.

Bem.: Die äußere Komplexität misst „die Größe der For-Schleife über den Passwort/Schlüsselraum“ (in der Regel der dominierende Faktor!). Für die Gesamtkomplexität müssen die zusätzlichen Operationen, wie z.B. Hashwertberechnungen/Entschlüsselungen, mitgerechnet werden (Operationen innerhalb der For-Schleife). Die Entropie bestimmt die Laufzeit vieler Angriffe.

Die Entropie

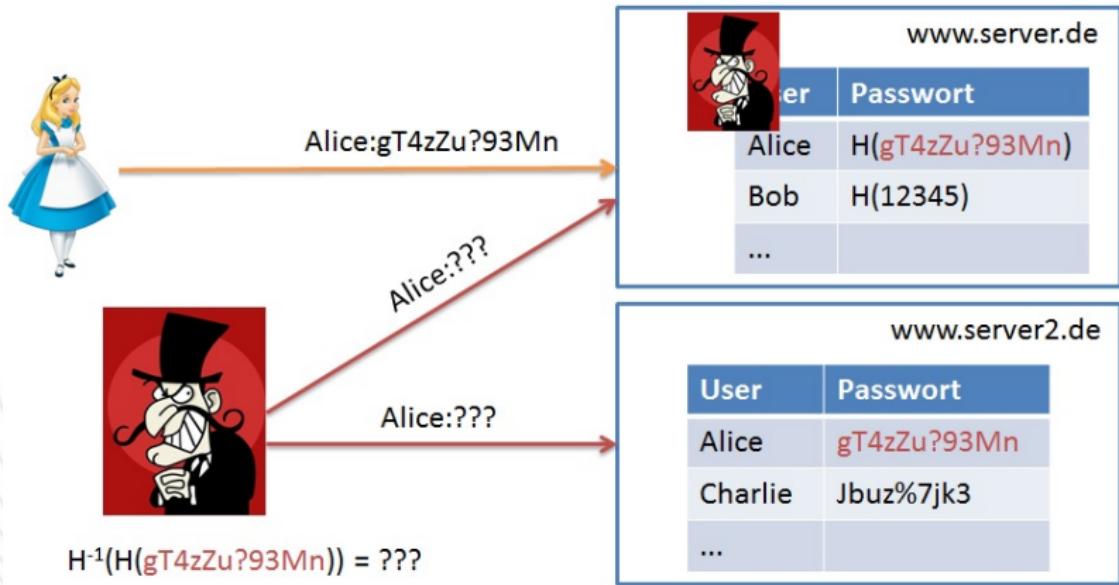
Definition (Entropie)

Ist $\{Z_1, \dots, Z_N\}$ der Passwort/Schlüsselraumes, so ist $\log_2(N)$ die Entropie des Passwortes/Schlüssels bei uniformverteilter zufälliger Auswahl aus dieser Menge. Ein Brute-Force Angriff besitzt für diese Entropie eine äußere Komplexität von $2^{\log_2(N)}$. Bei der äußeren Komplexität werden die bei jedem Passwort/Schlüsseltest anfallenden Operationen mit Aufwand 1 gezählt.

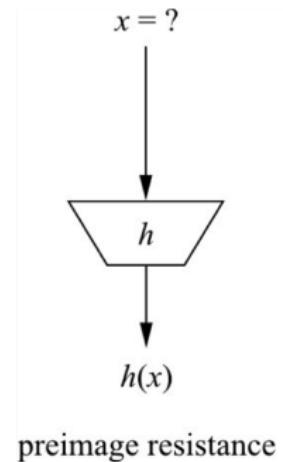
Bem.: Die äußere Komplexität misst „die Größe der For-Schleife über den Passwort/Schlüsselraum“ (in der Regel der dominierende Faktor!). Für die Gesamtkomplexität müssen die zusätzlichen Operationen, wie z.B. Hashwertberechnungen/Entschlüsselungen, mitgerechnet werden (Operationen innerhalb der For-Schleife). Die Entropie bestimmt die Laufzeit vieler Angriffe.

Bsp.: Werden 10-stellige Passwörter bestehend aus Groß- und Kleinbuchstaben verwendet, so beträgt die Entropie $52^{10} \approx 2^{57,004}$, d.h. die Entropie kann mit 57 abgeschätzt werden.

Password Hashing (Grundidee)

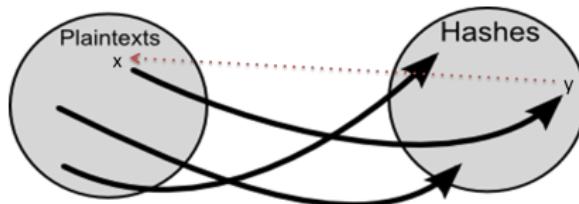


Angriffe auf die Urbildresistenz



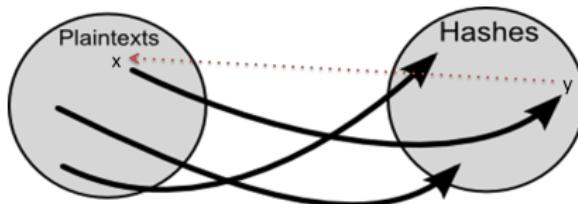
- ▶ z.B. SHA-1: Keine Urbild-Angriffe bekannt, die wesentlich besser sind als vollst. Suche („brute force“)
- ▶ Für Passwort Hashing: Wie finden wir möglichst effizient Urbilder, wenn wir wissen dass das Urbild ein Passwort ist?

Finde Urbild



- ▶ $H : X \rightarrow Y$
- ▶ $X =$ Menge der Passworte, z. B. 10-stellig aus [A-Za-z0-9!?]
- ▶ $Y =$ Menge der Passwort-Hashes (Y ist Teilmenge der Bildmenge $\{0, 1\}^n$ von H)

Finde Urbild



- ▶ $H : X \rightarrow Y$
 - ▶ $X =$ Menge der Passwörte, z. B. 10-stellig aus [A-Za-z0-9!?]
 - ▶ $Y =$ Menge der Passwort-Hashes (Y ist Teilmenge der Bildmenge $\{0, 1\}^n$ von H)
- ☞ Es reicht irgendein Urbild in X von $y = H(x)$ zu finden.
Es können bei Passworthashes auch Blockchiffren, wie z.B. AES oder DES, zum Einsatz kommen.

Einfachster Angriff: Brute-Force Angriff

- ▶ Bei einem Brute-Force-Angriff werden sämtliche Passwörter bis zu einer bestimmten Länge und einem vorgegebenen Zeichensatz z. B. [A-Za-z0-9!?] ¹⁰ durchprobiert und geprüft, ob der gesuchte Hashwert daraus erzeugt werden kann.

Einfachster Angriff: Brute-Force Angriff

- ▶ Bei einem Brute-Force-Angriff werden sämtliche Passwörter bis zu einer bestimmten Länge und einem vorgegebenen Zeichensatz z. B. $[A-Za-z0-9!?]^{10}$ ¹⁰ durchprobiert und geprüft, ob der gesuchte Hashwert daraus erzeugt werden kann.
- ▶ Die Komplexität hängt dabei von der Länge, dem Zeichensatz, d.h. der Entropie und der effizienten Berechenbarkeit der Hashfunktion ab.

Einfachster Angriff: Brute-Force Angriff

- ▶ Bei einem Brute-Force-Angriff werden sämtliche Passwörter bis zu einer bestimmten Länge und einem vorgegebenen Zeichensatz z. B. [A-Za-z0-9!?] ¹⁰ durchprobiert und geprüft, ob der gesuchte Hashwert daraus erzeugt werden kann.
- ▶ Die Komplexität hängt dabei von der Länge, dem Zeichensatz, d.h. der Entropie und der effizienten Berechenbarkeit der Hashfunktion ab.
- ▶ Ist n die Entropie der Passwörter, so ist die äußere Worst-Case Komplexität (Worst-Case = wirklich alle Passwörter müssen durchprobiert werden) 2^n und die äußere Average-Case Komplexität 2^{n-1} .

Aufgabe: Brute-Force-Angriff

Es steht Ihnen ein Rechner zur Verfügung mit:

- ▶ 10 Nvidia GTX-590 Grafikkarten.
Pro Karte 2 Grafikprozessoren mit 1024 Kernen und Taktfrequenz 1215 Mhz.
- ▶ 5 Knoten mit 8 Gb Speicher zur Lastenverteilung auf den Grafikkarten.

Gegeben sein ein Passworthash von dem Sie wissen, dass dieser aus einem Passwort mit 8 Zeichen aus dem Zeichensatz [A-Za-z0-9] generiert wurde.

Wie lange dauert es mit Ihrem Rechner das Passwort zu knacken?

Hinweis: Zur Vereinfachung dürfen Sie die Zeit zur Auswertung der Hashfunktion und die notwendigen Vergleiche mit 1 ansetzen (äußere Komplexität). Ferner dürfen Sie Prozessortakte pro Sekunde als Bitoperationen pro Sekunde rechnen.

Lösung

- ▶ $1215 \text{ MHz} = 1215 \cdot 10^6 \text{ Hz} \approx 2^{30}$ Schwingungen bzw. Bitoperationen pro Sekunde.

Lösung

- ▶ $1215 \text{ MHz} = 1215 \cdot 10^6 \text{ Hz} \approx 2^{30}$ Schwingungen bzw. Bitoperationen pro Sekunde.
- ▶ 2 Grafikprozessoren pro Karte, d.h. insgesamt 2^{11} Kerne, die jeweils 2^{30} Bitoperationen in der Sekunde schaffen.
☞ 2^{41} Bitoperationen pro Sekunde pro Karte und somit insgesamt $\approx 2^{44}$ Bitoperationen pro Sekunde.

Lösung

- ▶ $1215 \text{ MHz} = 1215 \cdot 10^6 \text{ Hz} \approx 2^{30}$ Schwingungen bzw. Bitoperationen pro Sekunde.
- ▶ 2 Grafikprozessoren pro Karte, d.h. insgesamt 2^{11} Kerne, die jeweils 2^{30} Bitoperationen in der Sekunde schaffen.
☞ 2^{41} Bitoperationen pro Sekunde pro Karte und somit insgesamt $\approx 2^{44}$ Bitoperationen pro Sekunde.
- ▶ Es gibt $62^8 \approx 2^{48}$ mögliche Passwörter zu dem Passworthash.
☞ Die Entropie des Passwortraumes ist 48.

Lösung

- ▶ $1215 \text{ MHz} = 1215 \cdot 10^6 \text{ Hz} \approx 2^{30}$ Schwingungen bzw. Bitoperationen pro Sekunde.
- ▶ 2 Grafikprozessoren pro Karte, d.h. insgesamt 2^{11} Kerne, die jeweils 2^{30} Bitoperationen in der Sekunde schaffen.
☞ 2^{41} Bitoperationen pro Sekunde pro Karte und somit insgesamt $\approx 2^{44}$ Bitoperationen pro Sekunde.
- ▶ Es gibt $62^8 \approx 2^{48}$ mögliche Passwörter zu dem Passworthash.
☞ Die Entropie des Passwortraumes ist 48.
- ▶ Somit benötigt der Rechner bei der vereinfachten Komplexitätsberechnung im Worst-Case ≈ 16 Sekunden.

Lösung

- ▶ $1215 \text{ MHz} = 1215 \cdot 10^6 \text{ Hz} \approx 2^{30}$ Schwingungen bzw. Bitoperationen pro Sekunde.
- ▶ 2 Grafikprozessoren pro Karte, d.h. insgesamt 2^{11} Kerne, die jeweils 2^{30} Bitoperationen in der Sekunde schaffen.
☞ 2^{41} Bitoperationen pro Sekunde pro Karte und somit insgesamt $\approx 2^{44}$ Bitoperationen pro Sekunde.
- ▶ Es gibt $62^8 \approx 2^{48}$ mögliche Passwörter zu dem Passworthash.
☞ Die Entropie des Passwortraumes ist 48.
- ▶ Somit benötigt der Rechner bei der vereinfachten Komplexitätsberechnung im Worst-Case ≈ 16 Sekunden.
- ▶ In der Praxis sind eher 30 Minuten zu erwarten.

Wie muss ein sicheres Passwort-basiertes Authentifikationsverfahren aussehen?

Beispiel:

Nimmt man noch 10 Sonderzeichen (z.B. die über den Zahlen auf einer typischen Computertastatur) hinzu, so beträgt die Entropie des Passwortraumes ≈ 62 , d.h. es gibt $\approx 2^{62}$ verschiedene Passwörter.

Wie muss ein sicheres Passwort-basiertes Authentifikationsverfahren aussehen?

Beispiel:

Nimmt man noch 10 Sonderzeichen (z.B. die über den Zahlen auf einer typischen Computertastatur) hinzu, so beträgt die Entropie des Passwortraumes ≈ 62 , d.h. es gibt $\approx 2^{62}$ verschiedene Passwörter.

- ▶ Der Cracker braucht in der Praxis ca. 30 Minuten für 2^{48} Passwörter und somit $2^{14} \cdot 30$ Min, d.h. ungefähr 1 Jahr für den erweiterten Passwortraum (im Worst-Case).
- ▶  Der Cracker ist veraltet und es wurde als Angriff nur Brute-Force betrachtet.

Wie muss ein sicheres Passwort-basiertes Authentifikationsverfahren aussehen?

Beispiel:

Nimmt man noch 10 Sonderzeichen (z.B. die über den Zahlen auf einer typischen Computertastatur) hinzu, so beträgt die Entropie des Passwortraumes ≈ 62 , d.h. es gibt $\approx 2^{62}$ verschiedene Passwörter.

- ▶ Der Cracker braucht in der Praxis ca. 30 Minuten für 2^{48} Passwörter und somit $2^{14} \cdot 30$ Min, d.h. ungefähr 1 Jahr für den erweiterten Passwortraum (im Worst-Case).
 - ▶  Der Cracker ist veraltet und es wurde als Angriff nur Brute-Force betrachtet.
- ☞ Zur Konstruktion sicherer Passwort-basierter Authentifikationsverfahren und Festlegung von Nutzerrichtlinien müssen sämtliche Angriffe betrachtet werden.

Wörterbuchangriffe (Dictionary Attacks)

- Bilde eine Liste mit wahrscheinlichen Passwörtern und gehe diese für jeden Passworthash durch.

Wörterbuchangriffe (Dictionary Attacks)

- ▶ Bilde eine Liste mit wahrscheinlichen Passwörtern und gehe diese für jeden Passworthash durch.
☞ Welche Passwörter als wahrscheinlich gelten hängt vom Nutzerkreis ab.
- ▶ Erfordert viel Speicherplatz.
- ▶ Im Internet findet man viele Beispillisten.

Wörterbuchangriffe (Dictionary Attacks)

- ▶ Bilde eine Liste mit wahrscheinlichen Passwörtern und gehe diese für jeden Passworthash durch.
☞ Welche Passwörter als wahrscheinlich gelten hängt vom Nutzerkreis ab.
- ▶ Erfordert viel Speicherplatz.
- ▶ Im Internet findet man viele Beispieldaten.

The screenshot shows a forum page for 'hashcat advanced password recovery'. The top navigation bar includes links for 'hashcat', 'Forums' (which is selected), 'Wiki', 'Tools', and 'Events'. A search bar and help link are also present. The main content area shows a post by a user named 'Hash-IT' (Moderator) from April 6, 2012, at 07:36 PM. The post discusses requests for word lists and provides several links to external resources. The post has 723 posts, 85 threads, and was joined in April 2011. The bottom of the page includes a 'Find' bar and a 'Go' button.

hashcat
advanced password recovery

hashcat **Forums** Wiki Tools Events

Search Help

Hello There, Guest | Login | Register |

Current time: 10-21-2016, 04:19 PM

hashcat Forum › Misc › User Contributions ▾
└ Word List Downloads

Pages (2): 1 2 Thread Modes

Word List Downloads

06-04-2012, 07:36 PM

Hash-IT Moderator

I have seen occasional requests on the forums for word lists so I thought I would post the best ones in one place.
If you know of a better site then please add it.

<http://cyberwarzone.com/cyberwarfare/pas...wordlists>
http://hashcrack.blogspot.de/p/wordlist....ds_29.html
<http://www.skullsecurity.org/wiki/index.php/Passwords>
<http://packetstormsecurity.org/Crackers/wordlists/>
<http://www.lsdpodcast.com/resources/62k...passwords>
<http://g0tm11k.blogspot.com/2011/06/dict...lists.html>
<http://www.mdsfthis.com/tools/wordlists.html>
<http://www.mdsfdecrypter.co.uk/downloads.aspx>

Posts: 723
Threads: 85
Joined: Apr 2011

Find Go

Weiterentwickelte Strategien: Time-Memory Trade-Off (TMTO)

Gundidee: Geschickte Vorberechnung die effizient genutzt werden kann.

- ▶ Philippe Oechslin, Making a Faster Cryptanalytic Time-Memory Trade-Off, CRYPTO 2003,
<http://lasecwww.epfl.ch/pub/lasec/doc/Oech03.pdf>
- ▶ <http://kestas.kuliukas.com/RainbowTables/>

Weiterentwickelte Strategien: Time-Memory Trade-Off (TMTO)

Gundidee: Geschickte Vorberechnung die effizient genutzt werden kann.

- ▶ Philippe Oechslin, Making a Faster Cryptanalytic Time-Memory Trade-Off, CRYPTO 2003,
<http://lasecwww.epfl.ch/pub/lasec/doc/Oech03.pdf>
 - ▶ <http://kestas.kuliukas.com/RainbowTables/>
- ☞ Diese extrem gute Technik wird in AuG untersucht.

Die wohl bekannteste TMTO-Anwendung: GSM's A5/1

💀 Das "GSM Cracking Project" hat im Jahr 2008 Tabellen für A5/1 fertiggestellt, die sie mit Forschern teilen, die drei TByte freien Speicherplatz haben. Mit den Tabellen lässt sich der geheime 64 bit Schlüssel mit über 95-prozentiger Wahrscheinlichkeit innerhalb von 30 Sekunden finden. Ist ein Schlüssel erst einmal gefunden, lässt sich die komplette Gesprächs- oder SMS-Verbindung mithören beziehungsweise -lesen.

Die wohl bekannteste TMTO-Anwendung: GSM's A5/1

💀 Das "GSM Cracking Project" hat im Jahr 2008 Tabellen für A5/1 fertiggestellt, die sie mit Forschern teilen, die drei TByte freien Speicherplatz haben. Mit den Tabellen lässt sich der geheime 64 bit Schlüssel mit über 95-prozentiger Wahrscheinlichkeit innerhalb von 30 Sekunden finden. Ist ein Schlüssel erst einmal gefunden, lässt sich die komplette Gesprächs- oder SMS-Verbindung mithören beziehungsweise -lesen.

Anmerkung: Ein reiner Brute-Force-Angriff war insbesondere im Jahre 2008 undenkbar (Auch die NSA wird wahrscheinlich einen TMTO-Angriff bevorzugt haben.)

Gegenmaßnahmen: Salt and Pepper



Alice:gT4zZu?93Mn

www.server.de

User	Salt	Passwort
Alice	a34D	H(a34D,gT4zZu?93Mn)
Bob	sv42	H(sv42,12345)
...		

- ▶ Salt: Vergrößere Dauer der Vorberechnung und Speicherbedarf von Time-Memory Trade-Offs
Z.B.: 16 bit Salt → Erhöhung der Komplexität um Faktor 2^{16} .
Warum?

Gegenmaßnahmen: Salt and Pepper



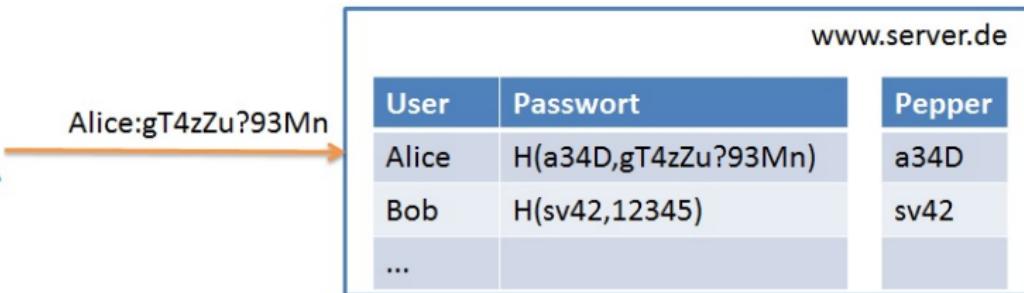
Alice:gT4zZu?93Mn

User	Salt	Passwort
Alice	a34D	H(a34D,gT4zZu?93Mn)
Bob	sv42	H(sv42,12345)
...		

- ▶ Salt: Vergrößere Dauer der Vorberechnung und Speicherbedarf von Time-Memory Trade-Offs
Z.B.: 16 bit Salt → Erhöhung der Komplexität um Faktor 2^{16} .
Warum?
- ▶ Moderne Betriebssysteme nutzen 128 bit = 16 byte Salt.
☞ Vorberechnung der für TMTO benötigten Tabellen nicht mehr praktisch durchführbar.

Nutzt Unix/Linux Salts und falls ja, wo findet man diese in einer /etc/shadow?

Gegenmaßnahmen: Salt and Pepper



- ▶ Pepper: Gleches Ziel wie Salting, aber Speicherung des Peppers außerhalb der Passwort-DB
- ▶ Sinnvoll z. B. wenn Angreifer **nur** Zugriff auf die Passwort-DB erhält (z.B. SQL-Injection)

Authentifikation durch Wissen: Challenge - Response

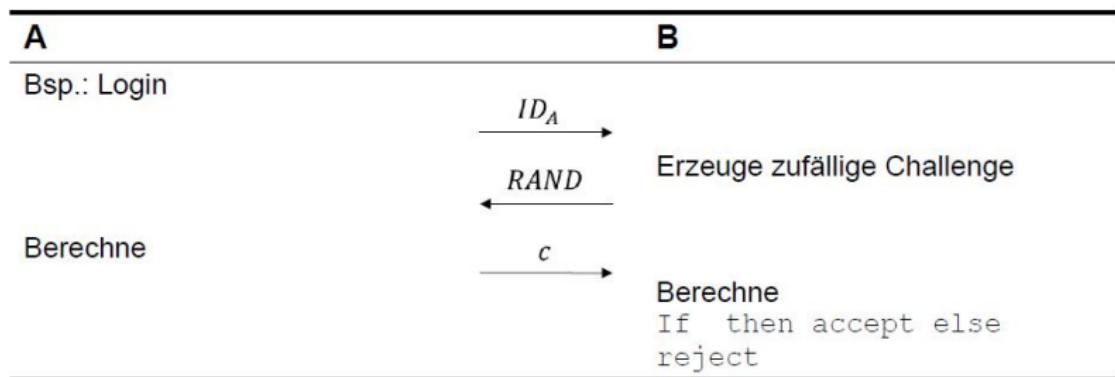
Challenge-Response-Verfahren (CR)

- ▶ Authentifikation eines Subjekts gegenüber einer Instanz
 - ▶ Subjekt (z.B. Mensch, Gerät, Dienst ...)
 - ▶ Instanz (z.B. Server, Gerät, Dienst, Mensch)
- ▶ Idee: Authentizitätsnachweis (z.B. bei jedem Login)
 - ▶ Subjekt gibt seine Identität an, z.B. Name, IMSI, MAC-Adresse
 - ▶ Instanz sendet eine Challenge (in der Regel Zufallszahl) zum Subjekt
 - ▶ Subjekt berechnet Response (z.B. mittels Verschlüsselung)
 - ▶ Instanz prüft Response, falls korrekt, dann hat Subjekt ein geheimes Wissen (z.B. Schlüssel) nachgewiesen

Authentifikation durch Wissen: Challenge - Response

Symmetrisches CR-Verfahren

- ▶ Ziel: Subjekt A authentifiziert sich gegenüber Instanz B.
- ▶ Basis:
 - ▶ Vorab geheimer Schlüssel (Pre-shared Secret)
 - ▶ Symmetrisches Verschlüsselungsverfahren



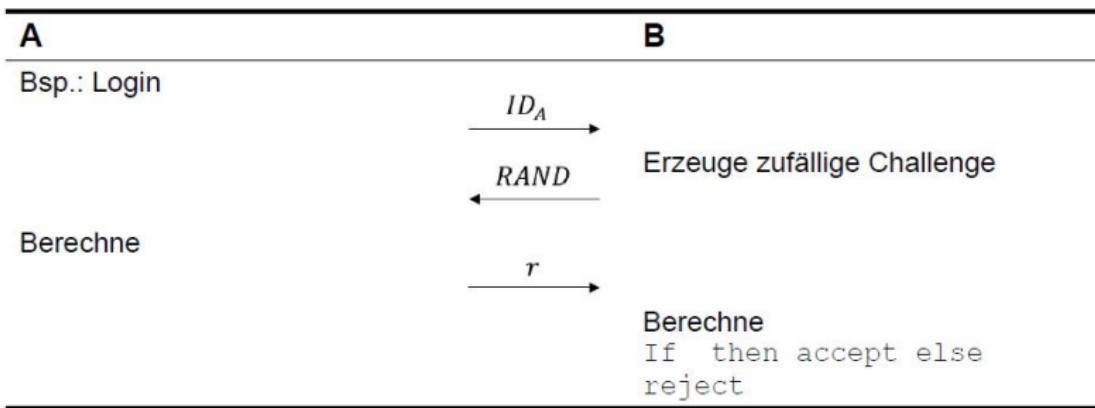
Probleme und Angriffe

- ▶ Klartextraum für Challenges muss groß sein. Ansonsten:
 - ▶ Angreifer hört alle Nachrichten ab und speichert sie.
 - ▶ Angreifer kann dann mehrfach benutzte Challenges korrekt beantworten!
- ▶ Verwendete Chiffre muss sicher gegenüber Known-Plaintext Angriffe sein
- ▶ Challenges haben oft Zeitstempel, um Gültigkeit einzuschränken und somit Replay Angriffe zu erschweren

Asymmetrisches CR-Verfahren

Basis:

- ▶ Schlüsselpaare eines asymmetrischen Verfahrens.



Probleme und Angriffe?

► Selber.

Authentifikation durch Wissen (Single-Sign-On) - Kerberos

- ▶ Ziel: Authentifikation von Subjekten bzw. Principals:
 - ▶ u.a. Benutzer, PC/Laptop, Server.
 - ▶ Austausch von Sitzungs-Schlüsseln und Authentifikation.
 - ▶ für Principals basierend auf sog. Needham-Schroeder-Protokoll.
 - ▶ Bietet Single-Sign-on (SSO) für Dienste in einer administrativen Domäne (auch realm genannt).
- ▶ Ziel eines Single-Sign-On (SSO) Konzepts.
- ▶ Benutzer authentifiziert sich einmal (zentral), keine separate Authentifikation bei Dienstenutzung mehr erforderlich.

Design von Kerberos

- ▶ Pro Domäne ein vertrauenswürdiger Server: Key Distribution Center: (Authentication Server AS + Ticket Granting Service TGS)
- ▶ Aufgabe des KDC:
 - ▶ Authentifizierung der Principals seiner Domäne.
 - ▶ Ausstellen von Tickets als Identitätsausweise.

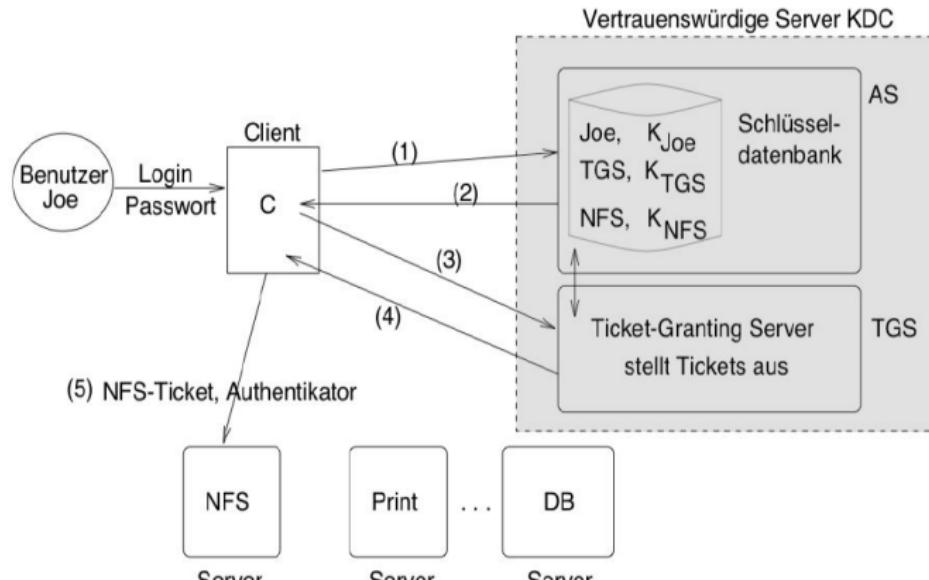
Authentifizierung eines Principals

- ▶ Basis: Pre-Shared Secrets zwischen KDC und Principal.
- ▶ Falls Principal ein Benutzer ist:
 - ▶ aus Benutzer-Passwort abgeleiteter Master-Key.
- ▶ Falls Principal ein Server ist:
 - ▶ KDC kennt gemeinsamen, geheimen Master-Key.

Das Ticket ist nur für einen Principal (z.B. Tom) und einen Server (z.B. Network file system) gültig. Inhalt eines Tickets

- ▶ Name des Servers,
- ▶ Name des anfordernden Clients,
- ▶ IP-Adresse des Clients,
- ▶ aktuelle Zeit,
- ▶ Lebenszeit des Tickets,
- ▶ Sitzungsschlüssel für Kommunikation zwischen Client und Server.

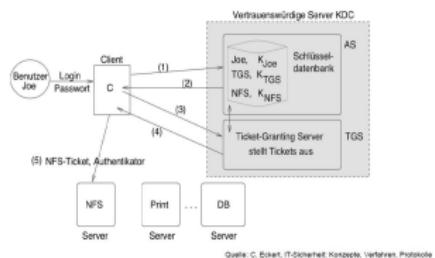
Authentifikation durch Wissen (Single-Sign-On) - Kerberos



Quelle: C. Eckert, IT-Sicherheit: Konzepte, Verfahren, Protokolle

Grober Protokoll-Ablauf:

- ▶ Benutzer Joe loggt sich mit Passwort auf lokalem Rechner (Client) ein.
- ▶ Lokaler Client fragt am AS ein Ticket für
 - 1) TGS an
 - 2) AS extrahiert Benutzer-Master-Key aus seiner Datenbank.
 - 3) und erstellt ein Ticket für die Nutzung des TGS.
 - 4) Client beantragt Ticket bei TGS zur Nutzung des NFS-Servers.
 - 5) TGS überprüft Ticket und sendet Joe ein Ticket für NFS Server.
 - 6) Joe benutzt das Ticket beim NFS-Server.



Quelle: C. Eckert, IT-Sicherheit: Konzepte, Verfahren, Protokole

Authentifikation durch Wissen (Single-Sign-On) - Kerberos

► Sicherheit von Kerberos.

- ▶ Clients verwalten Sitzungsschlüssel.
- ▶ Sichere Ablage von Tickets notwendig.
- ▶ Kritisch in Mehrbenutzersystemen (Unix Workstations).
- ▶ Verwendung von IP-Adressen problematisch, da fälschbar.
- ▶ Sicherheit basiert auf der Sicherheit des Nutzerpassworts.
- ▶ KDC stellt einen potentiellen Single Point of Failure dar.
- ▶ Vertrauenswürdige Zeit notwendig (Sicherheit gegen Replay-Angriffe).

☞ Trotz Problemen stellt Kerberos einen großen Fortschritt dar. Sicherheitsproblem Passwort (bzw. allgemein Wissen) erfordert weitere Maßnahmen, z.B. (zusätzlich) Smartcards, ID-Token etc.

Kerberos - Zusammenfassung

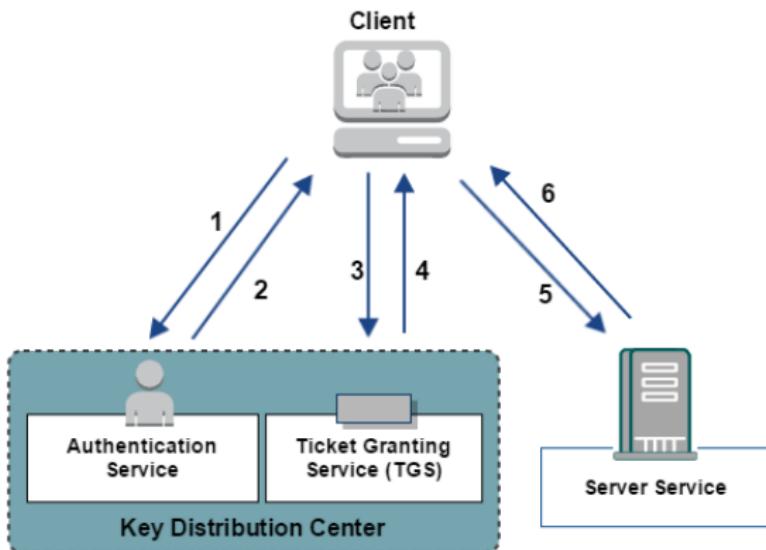


Abbildung: Quelle: <https://docops.ca.com/ca-single-sign-on/12-7/en/configuring/policy-server-configuration/authentication-schemes/configure-kerberos-authentication>

Authentifikation durch Besitz - ID Token

- ▶ Hardware-basierte One-Time-Passwort (OTP)-Verfahren:
ID-Token
 - ▶ OTP-Verfahren zur Authentifikation beim Server.
 - ▶ Benutzer erhält ein Hardware-Token.
 - ▶ Token besitzt eindeutige Nummer.
 - ▶ Server kennt diese Nummer.
- ▶ Beispiel RSA SecureID-Token:
 - ▶ Initialisierung.
 - ▶ Admin des Servers richtet Benutzer-Account ein, enthält:
 - ▶ Token-Nummer und 128 bit Seed.
 - ▶ Seed wird auch auf RSA-Token gespeichert.
 - ▶ Erzeugen von OTP's.
 - ▶ Alle 60 Sekunden generieren Token und Server neues OTP
 - ▶ OTP wird als Tokencode bezeichnet und findet seinen Ursprung im One-Time-Pad.



Authentifikation durch Besitz - ID Token



- ▶ Beispiel RSA SecureID-Token (Forts.)
 - ▶ Verwendung durch den Nutzer.
 - ▶ Tokencode wird auf Display des Tokens angezeigt.
 - ▶ Nutzer gibt den Code zur Authentifizierung ein.
Nachweis des Besitzes des Tokens.
 - ▶ Validierung eines Tokencodes durch den Server.
 - ▶ Hierbei werden die nächsten 3-5 Token zugelassen.
 - ▶ Anmerkungen:
 - ▶ Bei einem Hackerangriff auf Server von RSA im März 2011 sind ggf. Seeds und Seriennummern gestohlen worden, deshalb hat RSA ca. 40 Millionen SecureID-Tokens ausgetauscht.
 - ▶ RSA-Warnung vor eigenen Produkten im Sommer 2013:
 - ▶ Betrifft Produkt Bsafe für Entwickler, das einen ggf. unsicheren Zufallszahlengenerator umfasst.
- SecureID Token seien nicht betroffen.

Authentifikation durch Besitz - Universal 2-Factor (U2F)

- ▶ Offener Standard für 2-Faktor-Authentifizierung.
- ▶ Initial entwickelt von Google mit Yubico und NXP
 - ▶ <https://sites.google.com/site/oauthgoog/gnubby>
 - ▶ Weiterentwicklung durch die FIDO (Fast Identity Online) Alliance
<https://fidoalliance.org>
- ▶ Aktuell unterstützt im Google Chrome Browser (ab Version 38) zum
 - ▶ Einloggen bei Google-Accounts
 - ▶ als zweiter Faktor zum Zugriff auf Dropbox (seit 12.08.2015).
 - ▶ als zweiter Faktor zum Zugriff auf Github (seit 01.10.2015).

Authentifikation durch Besitz - Universal 2-Factor (U2F)



► Ansatz

- ▶ Für jeden Account (Username und ggf. Passwort) wird ein eigenes asymmetrisches Schlüsselpaar erzeugt und der öffentlichen Schlüssel wird an den Server übertragen
- ▶ Privater Schlüssel ist sicher auf USB- / NFC-Hardware-Token gespeichert
- ▶ Nutzer muss zur Nutzung explizit einen Knopf oder Touchsensor betätigen
- ▶ Challenge-Response zur Nutzerauthentifizierung

Authentifikation durch Besitz - Universal 2-Factor (U2F)

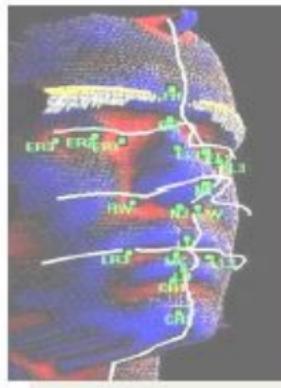
Sicherheit.

- ▶ U2F erschwert Man-in-the-Middle und Phishing.
- ▶ Integration von manipulationsresistentem Secure Element (SE) im Token kann zum Schutz des privaten Schlüssels genutzt werden.
 - ▶ Aus Kostengründen hat SE wenig Speicher, privater Schlüssel wird verschlüsselt abgelegt und symmetrischer Schlüssel im SE abgelegt.
 - ▶ SE schützt ebenfalls sogenannte Attestations-Schlüssel zum Nachweis der Echtheit des Tokens.
- ▶ Knopf bzw. Touchsensor zur Prüfung der physikalischen Präsenz
- ▶ Verschlüsselte Kommunikation zwischen Browser und U2F-Token (wichtig bei NFC (Near Field Communication oder Bluetooth-Kommunikation))

Authentifikation durch Merkmal - Biometrie

Biometrie altgriechisch bios „Leben“ und metron „Maß“

- ▶ Beobachtung und Messung von Merkmalen oder verhaltenstypischen Eigenschaften des menschlichen Körpers zur (Wieder-) Erkennung
- ▶ ISO/IEC Definition „Biometrics“
- ▶ „Automated recognition of individuals based on their behavioral and biological characteristics.“



Authentifikation durch Merkmal - Biometrie

Vorgehen bei biometrischer Authentifikation

- ▶ Enrolment / Registrierung eines Benutzers.
 - ▶ Messdatenerfassung durch biometrischen Sensor und Digitalisierung (Feature-Extraction).
 - ▶ Aufnahme, Auswahl und Speicherung der Referenzdaten, z.B. 5 bis 7 Fingerabdrücke.
- ▶ Authentifizierung
 - ▶ Erfassung der aktuellen Verifikationsdaten (mit Sensoren).
 - ▶ Verifikationsdaten digitalisieren (und ggf. normieren).
 - ▶ Vergleich mit gespeichertem Referenzwert (unter Berücksichtigung von Toleranzschwellen).
 - ▶ Ggf. zusätzlich: parallele Durchführung von Prozessen zur Erkennung von Angriffen / Fälschungen („Lebend-Erkennung“).

Authentifikation durch Merkmal - Biometrie

- ▶ Generelle Anforderungen an biometrischer Merkmale (Modalitäten)
 - ▶ Verbreitung: jede natürliche Person sollte die Charakteristik haben.
 - ▶ Einzigartigkeit: die Charakteristik ist unterschiedlich für jede Person.
 - ▶ Beständigkeit: die Charakteristik verändert sich nicht mit der Zeit.
 - ▶ Messbarkeit: die Charakteristik ist mit geringem Aufwand messbar.
- ▶ Anforderungen beim Einsatz in biometrischen Systemen
 - ▶ Performanz: in Bezug auf Erkennungsleistung und Geschwindigkeit.
 - ▶ Akzeptanz: wird das Verfahren von der Zielgruppe angenommen.
 - ▶ Fälschungssicherheit: es ist aufwändig das System zu umgehen.

Authentifikation durch Merkmal - Biometrie

Biometrische Merkmale (Modalitäten)

- ▶ Statische biologische Merkmale

- ▶ Fingerabdruck.
- ▶ Gesicht, Ohren.
- ▶ Retina und Iris.
- ▶ Handgeometrie.
- ▶ Venenmuster.
- ▶ DNA.
- ▶ :

- ▶ Dynamische biologische Merkmale

- ▶ Tippverhalten.
- ▶ Unterschriftendynamik.
- ▶ Stimme.
- ▶ Gang.
- ▶ EKG.
- ▶ :

Authentifikation durch Merkmal - Biometrie



Beispiel Gesichtserkennung

► Motivation

- ▶ Gesicht ist das Charakteristikum mit der größten Verbreitung.
- ▶ Potentiell hohe Benutzerakzeptanz (Bedienbarkeit).
- ▶ Erfassung erfolgt berührungslos.
- ▶ Kein Eingabegerät erforderlich, Kameras sind Massenware.
- ▶ Umfasst Stirn, Augen und Mundregion.

► Anatomischer Einfluss

- ▶ Knochengerüst.
- ▶ Gesichtsmuskulatur.
- ▶ Faltenwurf.
- ▶ Haut-Textur.
- ▶ Haarwuchs.
- ▶ Augen.

Beispiel Gesichtserkennung (Fortsetzung)

► Herausforderungen

► Pose.

- Orientation der Person zur Kamera.
- Unbekannter Abstand der Person zur Kamera.

► Beleuchtung.

- Sonnenlicht.
- Wechselnde Umweltbedingungen.
- Seitlicher Schattenwurf.

► Ausdruck und andere physikalische Variationen.

- Emotionale Ausdrücke.
- Haare.
- Alterung.
- :

Problem

- ▶ Abweichungen zwischen Referenz- und Verifikationsdaten.
- ▶ Erkennungsleistung wird in Fehlerwahrscheinlichkeiten (error rates) formuliert.
- ▶ Algorithmenfehler (false-positives, false-negatives).
 - ▶ Werden auf Basis einer existierenden Sample-Datenbank berechnet.
 - ▶ Messung der False-Match-Rate (FMR).
 - ▶ Messung der False-Non-Match-Rate (FNMR).
- ▶ Systemfehler, ergänzen die Algorithmenfehler
 - ▶ um Mensch-Sensor-Interaktionsfehler.
 - ▶ um Fehler in der Merkmalsextraktion.
- ▶ False-Accept-Rate (FAR)
 - ▶ Wahrscheinlichkeit, dass ein unberechtigter Nutzer akzeptiert wird (false positive).
- ▶ False-Reject-Rate (FRR)
 - ▶ Wahrscheinlichkeit, dass ein berechtigter Nutzer abgewiesen wird (false negative).

Authentifikation durch Merkmal - Biometrie



Angriffe

- ▶ Beispiel: Schäubles Fingerabdruck.
- ▶ Abdruck wurde durch CCC-Aktivisten von einem Wasserglas sichergestellt, das Dr. Schäuble bei einer öffentlichen Veranstaltung benutzt hatte.
- ▶
- ▶ Quelle und Bastelanleitung:
<http://www.ccc.de/updates/2008/schaubles-finger>.

Authentifikation durch Merkmal - Biometrie

Zusammenfassung

- ▶ Vorteile:
 - ▶ Einfache Benutzbarkeit.
 - ▶ Kein Verlust möglich (im Gegensatz z.B. zu Hardware-Token).
 - ▶ Kein Vergessen von PINs, Passwörtern etc. möglich.
 - ▶ Delegation nur schwer möglich.

Authentifikation durch Merkmal - Biometrie

Zusammenfassung

- ▶ Nachteile
 - ▶ Unscharfes Ergebnis.
 - ▶ Schwellwerte notwendig.
 - ▶ Angriffsmöglichkeiten:
 - ▶ Direkte Täuschung des biometrischen Sensors durch Attrappen, z.B. Gummifinger.
 - ▶ Einspielen von Daten unter Umgehung des biometrischen Sensors.
 - ▶ Wiedereinspielen abgehörter Daten (Replay-Angriffe).
 - ▶ Enge Kopplung zwischen Merkmal und Person.
 - ▶ Bedrohung der informationellen Selbstbestimmung / Datenschutzproblematik.
 - ▶ Gefahren durch gewaltsame Angriffe gegen Personen.
 - ▶ Unveränderlichkeit der Eigenschaften.
 - ▶ Problem der Öffentlichkeit der Daten und rechtliche Aspekte.

Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

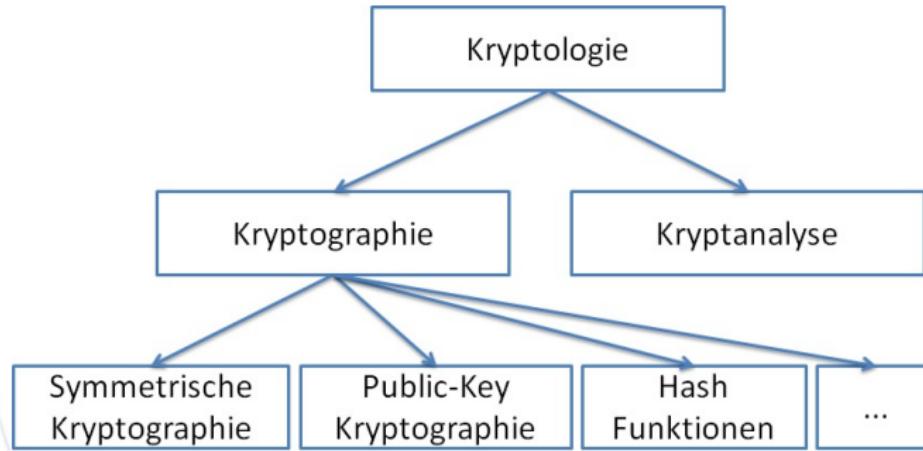
Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundsatz

Übersicht



Symmetrische Verschlüsselung



- ▶ Übertragung von Daten über unsicheren Kanal.
- ▶ Angreifer kann verschlüsselte Daten lesen.
- ▶ Angreifer soll die Nachricht inhaltlich nicht erschließen können.
- ▶ Voraussetzung: A und B teilen symm. Schlüssel k_{AB} .
Muss zuvor über „sicheren“ Kanal ausgetauscht werden!

Erreichbare Sicherheitsaspekte

- ▶ Vertraulichkeit von Daten.
- ▶ Integrität von Daten.
- ▶ Authentizität von Daten.
- ▶ ~~Nicht Abstreitbarkeit, dass eine gegebene Nachricht von einer bestimmten Partei gesendet wurde.~~

Kerckhoffs' Prinzip

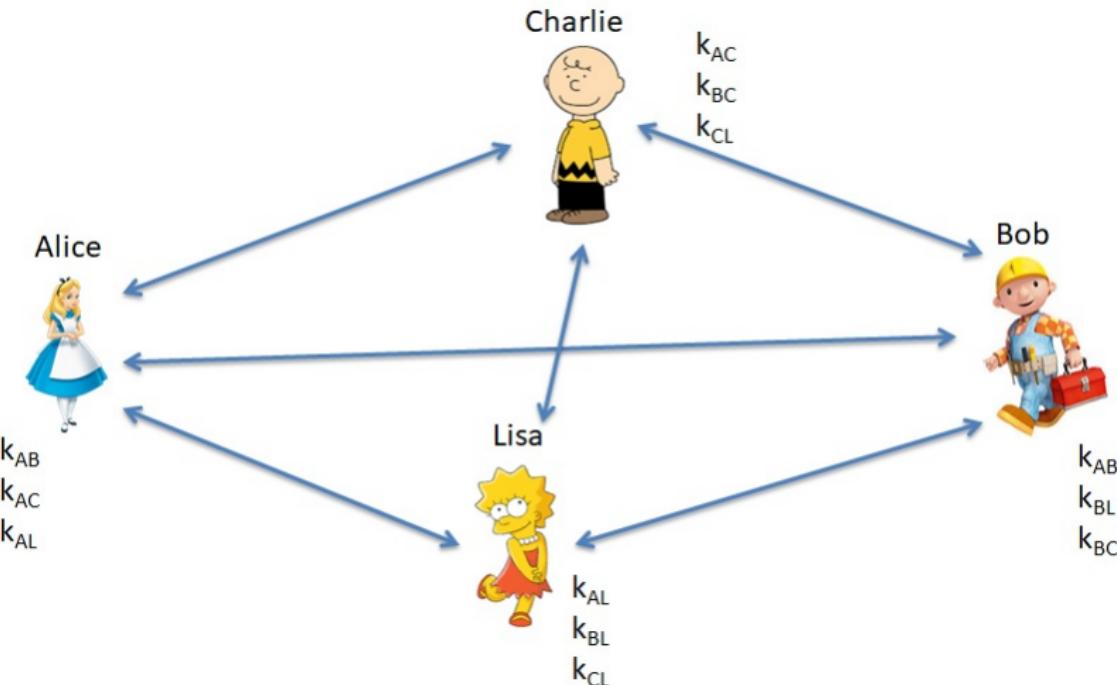
(Auguste Kerckhoffs, 19th century)

Ein Kryptosystem soll auch dann sicher sein, wenn alle Details des Systems öffentlich bekannt sind, bis auf den geheimen Schlüssel.

- ▶ Grundlegendes Prinzip der modernen Kryptographie.
- ▶ Öffentliche Analyse von Algorithmen.
- ▶ Keine „security by obscurity“.
  Oft schief gegangen: DVD CSS algorithm, KeeLoq, Enigma,...

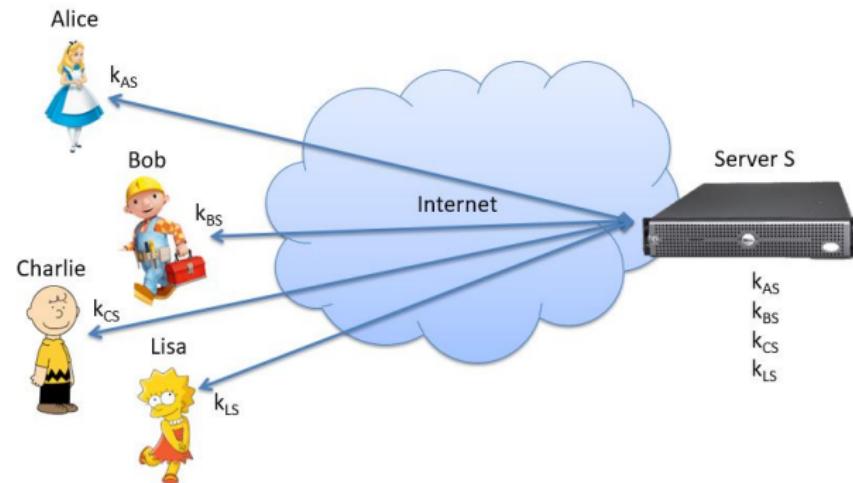
Ausnahme: Spezielle Chiffren entwickelt von Sicherheitsbehörden.

Schlüsselaustauschproblem 1



Schlüsselaustauschproblem 2

Key Distribution Center



- ▶ Alle müssen einen symmetrischen Schlüssel mit dem entfernten Server etablieren.
- ☞ Das bereits besprochene Kerberos verfolgt diesen Ansatz.

Weitere Beispiele für symmetrischen Schlüsselaustausch

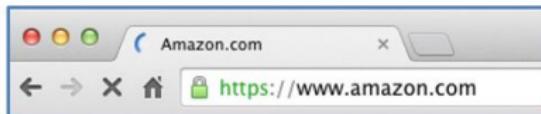
- Diplomatenpost bei hoheitlichen Verkehren.

Weitere Beispiele für symmetrischen Schlüsselaustausch

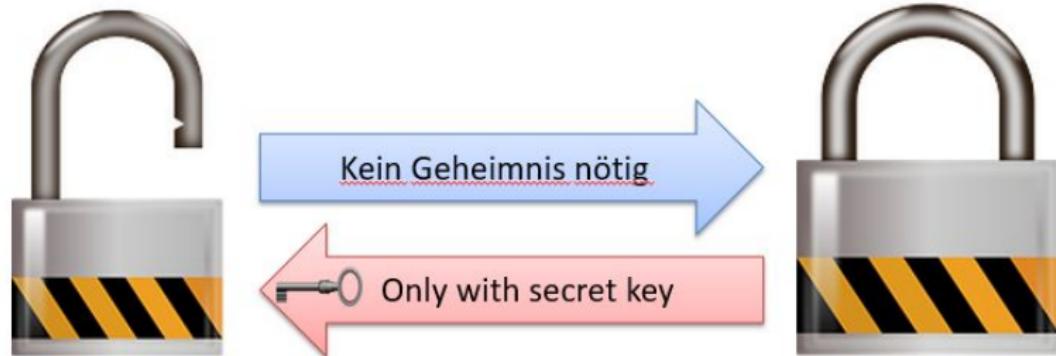
- ▶ Diplomatenpost bei hoheitlichen Verkehren.
- ▶ Sim-Karten bei mobilen Endgeräten.

Public-Key Kryptographie

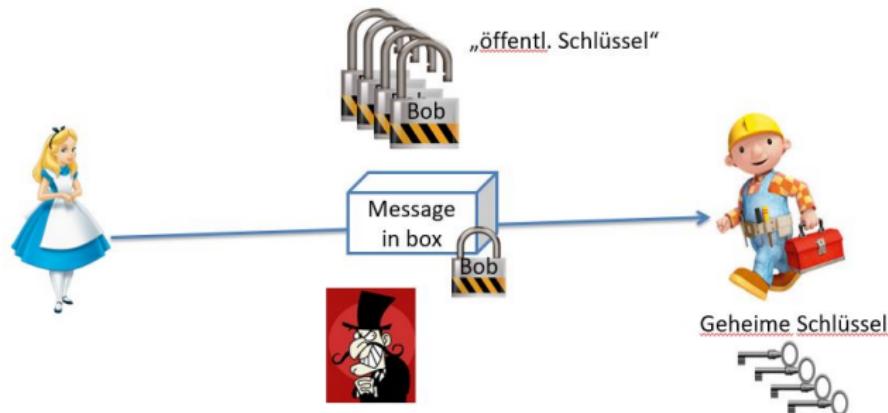
- ▶ Lösung für das Schlüsselverteilungsproblem
- ▶ „Enabling Technology“ für viele Anwendungen.
Sichere Datenübertragung im Internet
Authentication von Servern, Benutzern, Daten,...



Grundprinzip der Public-Key Kryptographie

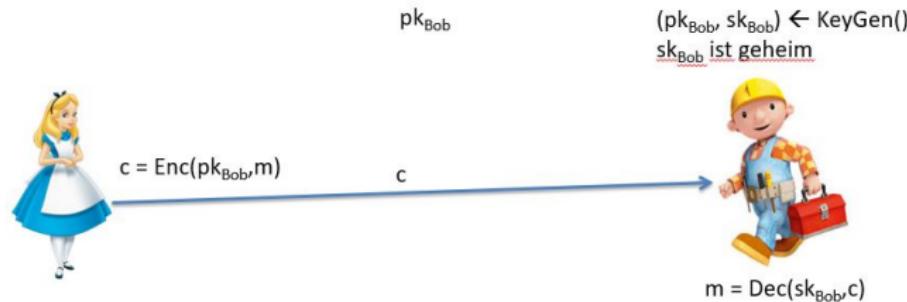


Grundprinzip Public-Key Kryptographie



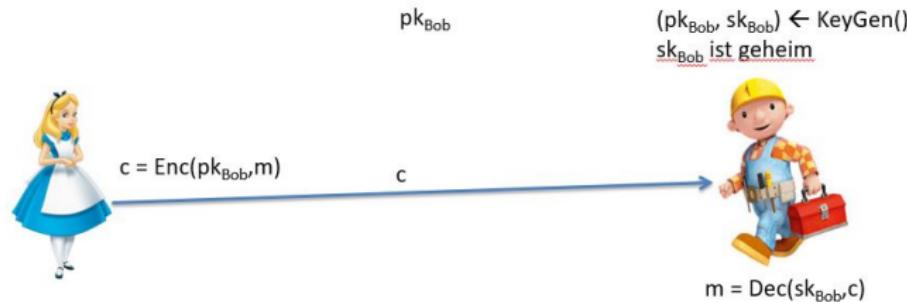
- ▶ Bob veröffentlicht offene Vorhängeschlösser.
- ▶ Alice legt Nachricht in eine Box, verschließt sie mit Bob's Schloß.
- ▶ Nur Bob kann die Box öffnen.

Public-Key Verschlüsselung



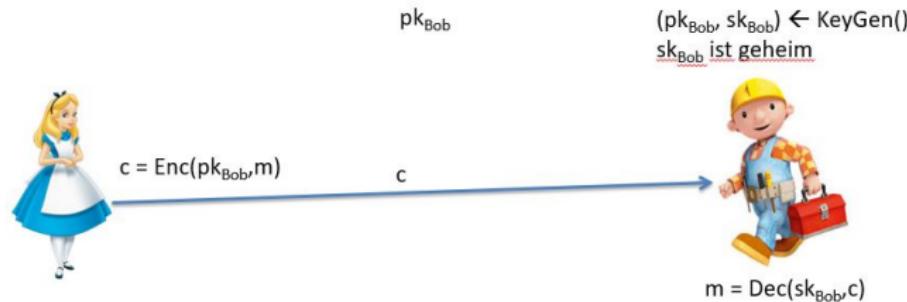
- ▶ Bob erzeugt Schlüsselpaar $(pk_{\text{Bob}}, sk_{\text{Bob}})$, veröffentlicht pk_{Bob} .

Public-Key Verschlüsselung



- ▶ Bob erzeugt Schlüsselpaar $(pk_{\text{Bob}}, sk_{\text{Bob}})$, veröffentlicht pk_{Bob} .
 - ▶ Public key pk_{Bob} : „Offenes Vorhängeschloss.“
 - ▶ Secret key sk_{Bob} : „Passender Schlüssel.“

Public-Key Verschlüsselung



- ▶ Bob erzeugt Schlüsselpaar $(\text{pk}_{\text{Bob}}, \text{sk}_{\text{Bob}})$, veröffentlicht pk_{Bob} .
 - ▶ Public key pk_{Bob} : „Offenes Vorhängeschloss.“
 - ▶ Secret key sk_{Bob} : „Passender Schlüssel.“
- ▶ Alice verschlüsselt Nachricht mit pk_{Bob} .
- ▶ Nur mit Bob's secret key sk_{Bob} kann die Nachricht entschlüsselt werden.

Mit Public-Key Verschlüsselung erreicht man

- ▶ Confidentiality of messages.
- ▶ ~~Integrity of messages.~~ In dem bisher dargestellten Einsatzszenario.

Mit Public-Key Verschlüsselung erreicht man

- ▶ Confidentiality of messages.
- ▶ ~~Integrity of messages~~. In dem bisher dargestellten Einsatzszenario.
- ▶ ~~Authenticity of messages~~. In dem bisher dargestellten Einsatzszenario
- ▶ ~~Non-repudiation that a given message was sent by a certain party~~. In dem bisher dargestellten Einsatzszenario.
- ▶ ~~Identification of users or devices~~. In dem bisher dargestellten Einsatzszenario.

Hybride Verschlüsselung

- ▶ Public key Verschlüsselung (PKE) ist i.A. viel ineffizienter als symmetrische Verschlüsselungsverfahren.

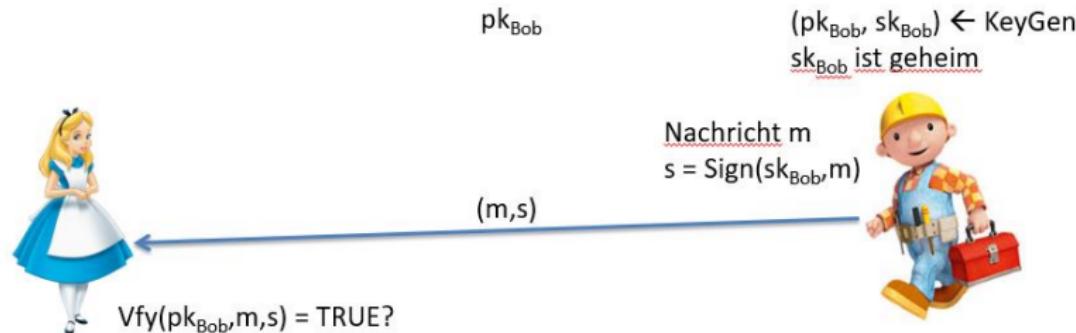
Hybride Verschlüsselung

- ▶ Public key Verschlüsselung (PKE) ist i.A. viel ineffizienter als symmetrische Verschlüsselungsverfahren.
 - ▶ Mehr Berechnungen nötig um die gleiche Menge an Daten zu ver-/entschlüsseln.
 - ▶ In der Praxis: Daten werden selten direkt mit PKE verschlüsselt.

Hybride Verschlüsselung

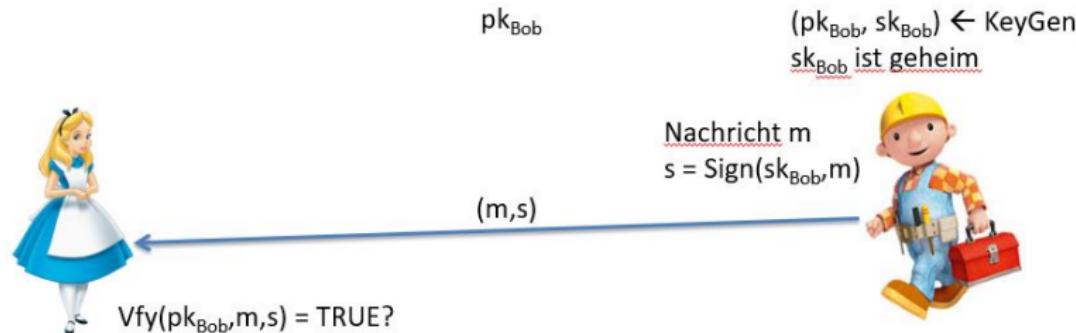
- ▶ Public key Verschlüsselung (PKE) ist i.A. viel ineffizienter als symmetrische Verschlüsselungsverfahren.
 - ▶ Mehr Berechnungen nötig um die gleiche Menge an Daten zu ver-/entschlüsseln.
 - ▶ In der Praxis: Daten werden selten direkt mit PKE verschlüsselt.
- ▶ In der Praxis: Hybride Public-Key Verschlüsselung
 - ▶ PKE um einen kurzen Schlüssel k zu verschlüsseln.
 - ▶ Symmetrisches Verfahren zur Verschlüsselung von Daten mit k .

Digitale Signaturen



- ▶ Bob möchte m an Alice senden, sodass Alice verifizieren kann dass die Nachricht von Bob stammt.
- ▶ Idee: „Nur Bob kennt sk_{Bob} “

Digitale Signaturen

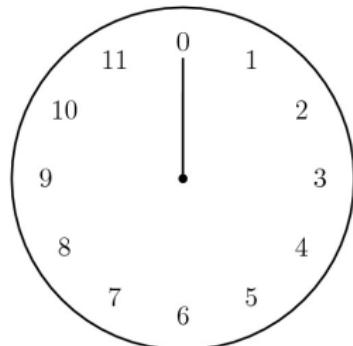


- ▶ Bob möchte m an Alice senden, sodass Alice verifizieren kann dass die Nachricht von Bob stammt.
- ▶ Idee: „Nur Bob kennt sk_{Bob} “
 - ▶ Bob berechnet Digitale Signatur s über m .
 - ▶ Alice nutzt pk_{Bob} zur Verifikation.

Digitale Signaturen ermöglichen

- ▶ ~~Vertraulichkeit von Daten.~~
- ▶ Integrität von Daten.
- ▶ Authentizität von Daten.
- ▶ Nicht-Abstreitbarkeit, dass eine gegebene Nachricht von einer bestimmten Partei gesendet wurde.
- ▶ Identifikation von Nutzern oder Geräten.

Modulare Arithmetik am Beispiel der klassischen Uhr

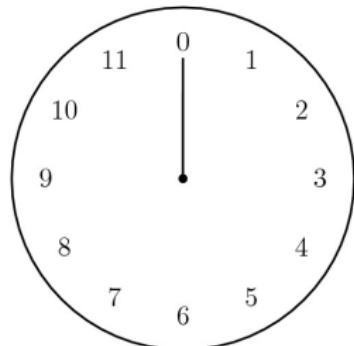


Bei einer Uhr werden immer wieder die Stunden $0, 1, 2, \dots, 11, 0, 1, 2, \dots, 11, \dots$ durchlaufen.

Insbesondere gilt:

Wenn es 10 Uhr ist und man sich in 5 Stunden trifft, dann trifft man sich

Modulare Arithmetik am Beispiel der klassischen Uhr

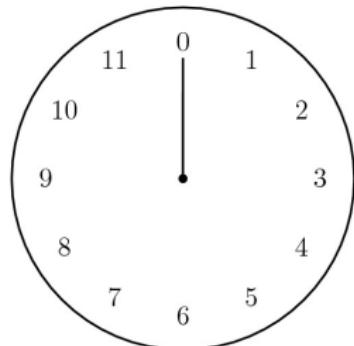


Bei einer Uhr werden immer wieder die Stunden $0, 1, 2, \dots, 11, 0, 1, 2, \dots, 11, \dots$ durchlaufen.

Insbesondere gilt:

Wenn es 10 Uhr ist und man sich in 5 Stunden trifft, dann trifft man sich um 3 Uhr.

Modulare Arithmetik am Beispiel der klassischen Uhr



Bei einer Uhr werden immer wieder die Stunden $0, 1, 2, \dots, 11, 0, 1, 2, \dots, 11, \dots$ durchlaufen.

Insbesondere gilt:

Wenn es 10 Uhr ist und man sich in 5 Stunden trifft, dann trifft man sich um 3 Uhr.

Abstrakt: $10 + 5 = 3$.

- Wir definieren nun eine Arithmetik $(+, \cdot)$ für diskrete, endlichen Mengen, wie die Menge $\{0, \dots, 11\}$ vom Beispiel der Uhr.

Der Kongruenzbegriff

Es seien $a, r, m \in \mathbb{Z}$ und $m > 0$. Dann ist
 $a \equiv r \pmod{m}$ genau dann, wenn
 $(r - a)$ von m geteilt wird.

- m heißt der Modulus.
- r heißt der Rest.

- Wir definieren nun eine Arithmetik $(+, \cdot)$ für diskrete, endlichen Mengen, wie die Menge $\{0, \dots, 11\}$ vom Beispiel der Uhr.

Der Kongruenzbegriff

Es seien $a, r, m \in \mathbb{Z}$ und $m > 0$. Dann ist
 $a \equiv r \pmod{m}$ genau dann, wenn
 $(r - a)$ von m geteilt wird.

- m heißt der Modulus.
- r heißt der Rest.

Beispiel:

- $m = 12$ und $a = 13$: $13 \equiv 1 \pmod{12}$ (Uhrenrechner)
- $m = 7$ und $a = 13$: $13 \equiv 6 \pmod{7}$

Der Rest r ist nicht eindeutig!



Der Rest r ist nicht eindeutig!

Beispiel:

- ▶ $12 \equiv 3 \pmod{9}$
- ▶ $12 \equiv 30 \pmod{9}$
- ▶ $12 \equiv -15 \pmod{9}$

Der Rest r ist nicht eindeutig!

Beispiel:

- ▶ $12 \equiv 3 \pmod{9}$
- ▶ $12 \equiv 30 \pmod{9}$
- ▶ $12 \equiv -15 \pmod{9}$

Der richtige Repräsentant:

- ▶ Für $a, m \in \mathbb{Z}$ gilt:

Es gibt eindeutige Zahlen $q, r \in \mathbb{Z}$ derart, dass $a = qm + r$ mit
 $0 \leq r \leq m - 1$

Beispiel: $a = 12, m = 9$

$$12 = 1 \cdot 9 + 3 \Rightarrow r = 3$$

Definition

Gegeben sei die Menge $\{0, \dots, m - 1\}$, $m \in \mathbb{Z}$. Für $a, b \in \{0, \dots, m - 1\}$ setzen wir

- ▶ $a + b := r_0$, wobei r_0 der Eindeutige Rest $a + b \equiv r_0 \pmod{m}$, $0 \leq r_0 \leq m - 1$ ist.
- ▶ $a \cdot b := r_1$, wobei r_1 der Eindeutige Rest $a \cdot b \equiv r_1 \pmod{m}$, $0 \leq r_1 \leq m - 1$ ist.

Die Menge $\{0, \dots, m - 1\}$, $m \in \mathbb{Z}$ zusammen mit diesen Operationen wird damit einem sogenannten Restklassenring, den wir mit $\mathbb{Z}/m\mathbb{Z}$ bezeichnen.

Wir schreiben häufig $a + b = r_0$ anstelle von $a + b \equiv r_0 \pmod{m}$, wenn klar ist, in welchem $\mathbb{Z}/m\mathbb{Z}$ gerechnet wird. Analog für '·'.

Reduktion bei Anwendung mehrerer arithmetischer Operationen

Betrachte $\mathbb{Z}/7\mathbb{Z}$.

Berechne $5 + 6 \cdot 5 \in \mathbb{Z}/7\mathbb{Z}$

Man kann erst $5 + 6 \cdot 5 = 35$ berechnen, dann modulo 7 reduzieren und erhält 0.

Genauso kann man zunächst $6 \cdot 5 = 30$ modulo 7 reduzieren und erhält 2. Anschließend berechnet man $2 + 5$ und erhält 0 modulo 7.

Man kann beweisen, dass es egal ist ab wann man reduziert!

Man sollte immer frühzeitig reduzieren, wie das Beispiel

$$6^{100} =$$

Reduktion bei Anwendung mehrerer arithmetischer Operationen

Betrachte $\mathbb{Z}/7\mathbb{Z}$.

Berechne $5 + 6 \cdot 5 \in \mathbb{Z}/7\mathbb{Z}$

Man kann erst $5 + 6 \cdot 5 = 35$ berechnen, dann modulo 7 reduzieren und erhält 0.

Genauso kann man zunächst $6 \cdot 5 = 30$ modulo 7 reduzieren und erhält 2. Anschließend berechnet man $2 + 5$ und erhält 0 modulo 7.

Man kann beweisen, dass es egal ist ab wann man reduziert!

Man sollte immer frühzeitig reduzieren, wie das Beispiel

$6^{100} = 36^{50} = 1^{50} = 1 \in \mathbb{Z}/7\mathbb{Z}$ zeigt.

Beispiel $m = 2, \{0, 1\}$

- ▶ $1 + 1 = 0, 1 + 0 = 1, \dots$
- ▶ $1 \cdot 1 = 1, 1 \cdot 0 = 0$

Die bekannte Binärarithmetik. $\mathbb{Z}/2\mathbb{Z}$ wird auch mit \mathbb{F}_2 bezeichnet.

Beispiel $m = 12, \mathbb{Z}/m\mathbb{Z} = \{0, \dots, 11\}$

► $10 + 3 = 1$

► $5 \cdot 5 =$

Beispiel $m = 12, \mathbb{Z}/m\mathbb{Z} = \{0, \dots, 11\}$

- ▶ $10 + 3 = 1$
- ▶ $5 \cdot 5 = 1$

$\mathbb{Z}/12\mathbb{Z}$ ist der Uhrenrechner.

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .



$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- ▶ Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- ▶ Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$
- ▶ Die Addition und Multiplikation ist assoziativ, d.h. für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$

$$a + (b + c) = (a + b) + c$$

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- ▶ Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$
- ▶ Die Addition und Multiplikation ist assoziativ, d.h. für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$
$$a + (b + c) = (a + b) + c$$
$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$
- ▶ Die Addition und Multiplikation ist kommutativ:

$$a + b = b + a, a \cdot b = b \cdot a$$

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- ▶ Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$
- ▶ Die Addition und Multiplikation ist assoziativ, d.h. für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$

$$a + (b + c) = (a + b) + c$$

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

- ▶ Die Addition und Multiplikation ist kommutativ:

$$a + b = b + a, a \cdot b = b \cdot a$$

- ▶ Das Distributivgesetz gilt: $a \cdot (b + c) = a \cdot b + a \cdot c$ für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$.

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- ▶ Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$
- ▶ Die Addition und Multiplikation ist assoziativ, d.h. für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$

$$a + (b + c) = (a + b) + c$$

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

- ▶ Die Addition und Multiplikation ist kommutativ:

$$a + b = b + a, a \cdot b = b \cdot a$$

- ▶ Das Distributivgesetz gilt: $a \cdot (b + c) = a \cdot b + a \cdot c$ für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$.
- ▶ Das neutrale Element bzgl. der Addition ist 0, d.h. für alle $a \in \mathbb{Z}/m\mathbb{Z}$ ist $a + 0 = a$.

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$

- Die Addition und Multiplikation ist assoziativ, d.h. für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$

$$a + (b + c) = (a + b) + c$$

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

- Die Addition und Multiplikation ist kommutativ:

$$a + b = b + a, a \cdot b = b \cdot a$$

- Das Distributivgesetz gilt: $a \cdot (b + c) = a \cdot b + a \cdot c$ für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$.

- Das neutrale Element bzgl. der Addition ist 0, d.h. für alle $a \in \mathbb{Z}/m\mathbb{Z}$ ist $a + 0 = a$.

- Für alle $a \in \mathbb{Z}/m\mathbb{Z}$ gibt es ein additives Inverses $-a$, d.h. ein Element $-a$ mit $a + (-a) = 0$.

$\mathbb{Z}/m\mathbb{Z}$ erbt viele schöne Eigenschaften von \mathbb{Z} .

- Die Addition und Multiplikation ist abgeschlossen, d.h. das Ergebnis liegt wieder in $\mathbb{Z}/m\mathbb{Z}$

- Die Addition und Multiplikation ist assoziativ, d.h. für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$

$$a + (b + c) = (a + b) + c$$

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c$$

- Die Addition und Multiplikation ist kommutativ:

$$a + b = b + a, a \cdot b = b \cdot a$$

- Das Distributivgesetz gilt: $a \cdot (b + c) = a \cdot b + a \cdot c$ für alle $a, b, c \in \mathbb{Z}/m\mathbb{Z}$.

- Das neutrale Element bzgl. der Addition ist 0, d.h. für alle $a \in \mathbb{Z}/m\mathbb{Z}$ ist $a + 0 = a$.

- Für alle $a \in \mathbb{Z}/m\mathbb{Z}$ gibt es ein additives Inverses $-a$, d.h. ein Element $-a$ mit $a + (-a) = 0$.

- 1 ist das neutrale Element bzgl. der Multiplikation, d.h. für jedes $a \in \mathbb{Z}/m\mathbb{Z}$ gilt $a \cdot 1 = a$

Multiplikative Inverse

In \mathbb{Z} ist $-1 \cdot -1 = 1$, wie sieht es mit Multiplikativen Inversen in $\mathbb{Z}/m\mathbb{Z}$ aus?

Multiplikative Inverse

In \mathbb{Z} ist $-1 \cdot -1 = 1$, wie sieht es mit Multiplikativen Inversen in $\mathbb{Z}/m\mathbb{Z}$ aus?

Definition

Es sei $a, b \in \mathbb{Z}/m\mathbb{Z}$.

- Ist $a \cdot b = 1$, so heißt b das Inverse und wird mit a^{-1} bezeichnet.

Multiplikative Inverse

In \mathbb{Z} ist $-1 \cdot -1 = 1$, wie sieht es mit Multiplikativen Inversen in $\mathbb{Z}/m\mathbb{Z}$ aus?

Definition

Es sei $a, b \in \mathbb{Z}/m\mathbb{Z}$.

- Ist $a \cdot b = 1$, so heißt b das Inverse und wird mit a^{-1} bezeichnet.

Beispiel: $5 \cdot 5 \equiv 1 \pmod{12}$.

4 besitzt kein Inverses modulo 12 bzw. in $\mathbb{Z}/12\mathbb{Z}$.

Multiplikative Inverse

In \mathbb{Z} ist $-1 \cdot -1 = 1$, wie sieht es mit Multiplikativen Inversen in $\mathbb{Z}/m\mathbb{Z}$ aus?

Definition

Es sei $a, b \in \mathbb{Z}/m\mathbb{Z}$.

- Ist $a \cdot b = 1$, so heißt b das Inverse und wird mit a^{-1} bezeichnet.

Beispiel: $5 \cdot 5 \equiv 1 \pmod{12}$.

4 besitzt kein Inverses modulo 12 bzw. in $\mathbb{Z}/12\mathbb{Z}$.

- Das Inverse von a existiert genau dann, wenn $\text{ggT}(a, m) = 1$. Man sagt auch, dass a teilerfremd, koprime oder relativ prim zu m ist.

Satz und Definition

- ▶ Für $\mathbb{Z}/m\mathbb{Z}$ wird die Menge der invertierbaren Elementen mit $\mathbb{Z}/m\mathbb{Z}^*$ bezeichnet und heißt prime Restklassengruppe.
- ▶ Es ist $\mathbb{Z}/m\mathbb{Z}^* = \{1, \dots, m-1\}$ genau dann, wenn $m = p$, p prim ist.
In diesem Fall ist $\mathbb{Z}/p\mathbb{Z}$ ein sogenannter Körper den wir auch mit \mathbb{F}_p bezeichnen.



- ▶ \mathbb{F}_2 ist die bekannte Binäarithmetik mit
 $1 + 0 = 1, 1 + 1 = 0, \dots, 1 \cdot 1 = 1, \dots$
- ▶ In diesem Fall wird statt $+$ oft \oplus verwendet.

- Beispiele für Körper mit unendlich vielen Elementen sind \mathbb{Q} oder \mathbb{R} .
- ☞ Für $+$ und \cdot gelten dieselben algebraischen Eigenschaften.

- ▶ Beispiele für Körper mit unendlich vielen Elementen sind \mathbb{Q} oder \mathbb{R} .
☞ Für $+$ und \cdot gelten dieselben algebraischen Eigenschaften.
- ▶ Die Anzahl der Elemente mit $\text{ggT}(a, m) = 1$ kann mittels der eulerschen ϕ -Funktion bestimmt werden (wichtig für RSA).
- ▶ Das Inverse kann mittels des effizienten euklidischen Algorithmus berechnet werden (wichtig für RSA, DSA).
- ▶ Die Inversenberechnung ist zentral für viele moderne Kryptosysteme.

Das Inverse kann man mittels des erweiterten euklidischen Algorithmus bestimmen.

Multiplikative Inverse

Betrachte $\mathbb{Z}/10\mathbb{Z} = \{0, 1, 2, \dots, 9\}$:

- ▶ Es ist z.B. $3 \cdot 7 = 1 \text{ mod } 10$, d.h. $3^{-1} = 7$ aber auch $7^{-1} = 3 \text{ mod } 10$. Es ist $\text{ggT}(7, 10) = \text{ggT}(3, 10) = 1$
- ▶ 5 besitzt kein multiplikatives Inverses. Es ist $\text{ggT}(5, 10) = 2$.

Betrachte $\mathbb{Z}/3\mathbb{Z} = \{0, 1, 2\}$:

- ▶ Es ist $2 \cdot 2 = 1$ und $1 \cdot 1 = 1 \text{ mod } 3$, d.h. $2^{-1} = 2$ und $1^{-1} = 1 \text{ mod } 3$.
Das sind alle koprime Elemente in $\{0, 1, 2\}$

Der Vektorraum \mathbb{F}_p^n

Definition

Die Menge

$$\left\{ \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \mid x_i \in \mathbb{F}_p, i = 1, \dots, n \right\}$$

wird mit der komponentenweisen Addition

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} + \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} x_1 + y_1 \\ \vdots \\ x_n + y_n \end{pmatrix}$$

und der Skalarmultiplikation $\lambda \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \lambda x_1 \\ \vdots \\ \lambda x_n \end{pmatrix}$, $\lambda \in \mathbb{F}_p$ zu

einem sogenannten \mathbb{F}_p -Vektorraum. Diesen bezeichnen wir mit \mathbb{F}_p^n .

Eigenschaften

1. $\forall x, y, z \in \mathbb{F}_p^n : (x + y) + z = x + (y + z)$

2. $\forall x, y \in \mathbb{F}_p^n : x + y = y + x$

3. $\forall x \in \mathbb{F}_p^n : \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} + x = x.$

4. $\forall x \in \mathbb{F}_p^n \exists y \in \mathbb{F}_p^n : x + y = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$, Bezeichnung $y := -x..$

5. $\forall x \in \mathbb{F}_p^n \forall \lambda, \nu \in \mathbb{F}_p : (\lambda\nu)x = \lambda(\nu x).$

6. $\forall x \in \mathbb{F}_p^n : 1 \cdot x = x.$

7. $\forall x, y \in \mathbb{F}_p^n \forall \lambda \in \mathbb{F}_p : \lambda(x + y) = \lambda x + \lambda y.$

8. $\forall x \in \mathbb{F}_p^n \forall \lambda, \nu \in \mathbb{F}_p : (\lambda + \nu) \cdot x = \lambda x + \nu x.$

☞ Das sind die Vektorraumaxiome.

Beispiel über \mathbb{F}_3

$$\blacktriangleright \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\blacktriangleright 2 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \end{pmatrix}$$

$$\blacktriangleright \begin{pmatrix} 0 \\ 1 \\ 2 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 2 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \text{ d.h. } - \begin{pmatrix} 0 \\ 1 \\ 2 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 1 \\ 0 \end{pmatrix}$$

Beispiel über \mathbb{F}_2

$$\blacktriangleright \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \text{ d.h. } - \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

- ▶ Vektorräume finden ihren Ursprung in der Geometrie und physikalischen Fragestellungen.
- ▶ Vektorräume und Matrizen (später) über endlichen Körpern gehören zu den wichtigsten, elementaren Handwerkszeugen in der Kryptologie und beim Entwickeln von Hacking-Tools.

Untervektorraum

Untervektorraum

Sei U eine nichtleere Teilmenge U des Vektorraumes \mathbb{F}_p^n . U heißt Untervektorraum von \mathbb{F}_p^n genau dann, wenn

1. $v + w \in U$ für alle $v, w \in U$.
2. $\lambda v \in U$ für alle $\lambda \in \mathbb{F}_p, v \in U$.

Bsp.: $U = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \right\}$ ist ein

Untervektorraum des \mathbb{F}_2^3 .

\mathbb{F}_2^3 selbst ist ein UVR.

Linearkombination

Für Vektoren v_1, \dots, v_m des \mathbb{F}_p^n und $\lambda_1, \dots, \lambda_m \in \mathbb{F}_p$ heißt die Summe

$$\lambda_1 v_1 + \dots + \lambda_m v_m$$

Linearkombination von v_1, \dots, v_m . Die Menge aller Linearkombinationen $\{\lambda_1 v_1 + \dots + \lambda_m v_m | \lambda_1, \dots, \lambda_m \in \mathbb{F}_p\}$ heißt Spann von v_1, \dots, v_m und wird mit $\langle v_1, \dots, v_m \rangle$ bezeichnet.

Bsp.: $U = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \right\} =$

$$\langle \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \rangle.$$

Definition (Basis und Dimension)

Gegeben sei ein Untervektorraum U des \mathbb{F}_p^n .

Die Vektoren u_1, \dots, u_m heißen Basis von U genau dann, wenn gilt:

1. $U = \langle u_1, \dots, u_m \rangle$, d.h. $u \in U$, dann ist $u = \lambda_1 u_1 + \dots + \lambda_m u_m$ mit geeigneten $\lambda_1, \dots, \lambda_m \in \mathbb{F}_p$.
2. Ist $0 = \lambda_1 u_1 + \dots + \lambda_m u_m$, so ist $0 = \lambda_1 = \dots = \lambda_m \in \mathbb{F}_p$, d.h. u_1, \dots, u_m sind linear unabhängig.
3. m heißt die Dimension von U .

Wichtig:

- Man kann zeigen, dass je zwei Basen gleich viele Vektoren besitzen (sonst Definition nicht wohldefiniert!).
- Ist $u = \lambda_1 u_1 + \dots + \lambda_m u_m$, so sind $\lambda_1, \dots, \lambda_m$ eindeutig bestimmt für eine Basis u_1, \dots, u_m .

Beispiel:

$U = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \right\}$ = besitzt die Basis

$u_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, u_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$. U hat also die Dimension 2.

Wie sieht die eindeutige Linearkombination von $\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ aus ? Kann

man $\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ linear kombinieren?

Beispiel:

$$U = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \right\} = \text{besitzt die Basis}$$

$$u_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, u_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}. U \text{ hat also die Dimension } 2.$$

Wie sieht die eindeutige Linearkombination von $\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$ aus? Kann

man $\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$ linear kombinieren?

$$\mathbb{F}_5^3 \text{ besitzt die Basis } e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, e_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

\mathbb{F}_5^3 hat also die Dimension 3.

Kann man dieses Beispiel auf \mathbb{F}_p^n verallgemeinern?

1. Ein rechteckiges Schema

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \text{ mit Elementen}$$

$a_{ij} \in \mathbb{F}_p$, $1 \leq i \leq m$, $1 \leq j \leq n$ heißt eine Matrix. Die Elemente a_{ij} heißen Koeffizienten oder Einträge der Matrix. Ist m die Anzahl der Zeilen und n die Anzahl der Spalten, so sagt man dafür auch, A ist eine $m \times n$ Matrix.

2. Die Menge aller Matrizen mit Einträgen aus \mathbb{F}_p wird mit $M(m \times n, \mathbb{F}_p)$ bezeichnet.

Matrizen und Abbildungen

Matrix-Vektor-Multiplikation und lineare Abbildungen

Gegeben sei eine Matrix $A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \vdots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$, $a_{ij} \in \mathbb{F}_p$.

1. Die Matrix-Vektor-Multiplikation wird definiert als

$$\begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \mapsto A \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} := \begin{pmatrix} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \\ \vdots \\ a_{m1}x_1 + a_{12}x_2 + \dots + a_{mn}x_n \end{pmatrix}.$$

2. Die Matrix-Vektor-Multiplikation liefert eine Abbildung von $\mathbb{F}_p^n \rightarrow \mathbb{F}_p^m$, die wir mit L_A bezeichnen.

Matrizen und Abbildungen

Lineare Abbildung

1. Die durch eine $m \times n$ -Matrix A definierte Abbildung

$L_A : \mathbb{F}_p^n \rightarrow \mathbb{F}_p^m, x \mapsto A \cdot x$ ist linear, d.h. es ist

$$A \cdot \left(\lambda_1 \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} + \lambda_2 \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \right) =$$
$$\lambda_1 A \cdot \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} + \lambda_2 A \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \text{ für alle}$$
$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}, \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \in \mathbb{F}_p^n, \lambda_1, \lambda_2 \in \mathbb{F}_p.$$

Beispiel

$$A = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, A \in M(3 \times 4, \mathbb{F}_3)$$

$$\blacktriangleright A \cdot \begin{pmatrix} 2 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

$$\blacktriangleright A \cdot \left(2 \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \right) =$$

$$2A \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + A \cdot \begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = 2 \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Verknüpfungen von Matrizen

Addition und Skalarmultiplikation

Gegeben seien $m \times n$ -Matrizen $A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \vdots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$ und

$B = \begin{pmatrix} b_{11} & \dots & b_{1n} \\ \vdots & \vdots & \vdots \\ b_{m1} & \dots & b_{mn} \end{pmatrix}$. Wir definieren

$A + B := \begin{pmatrix} a_{11} + b_{11} & \dots & a_{1n} + b_{1n} \\ \vdots & \vdots & \vdots \\ a_{m1} + b_{m1} & \dots & a_{mn} + b_{mn} \end{pmatrix}$ und

$\lambda A = \begin{pmatrix} \lambda a_{11} & \dots & \lambda a_{1n} \\ \vdots & \vdots & \vdots \\ \lambda a_{m1} & \dots & \lambda a_{mn} \end{pmatrix}$.

Bem.: Matrizen können als \mathbb{F}_p^{nm} -Vektorräume aufgefasst werden.

Somit sind obige Definitionen kanonisch.

Verknüpfungen von Matrizen

Matrizenmultiplikation

Gegeben sei eine $m \times n$ -Matrix $A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \vdots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}$ und
 $n \times r$ -Matrix $B = \begin{pmatrix} b_{11} & \dots & b_{1r} \\ \vdots & \vdots & \vdots \\ b_{n1} & \dots & b_{nr} \end{pmatrix}$. Wir definieren das Produkt
 $A \cdot B := \begin{pmatrix} c_{11} & \dots & c_{1r} \\ \vdots & \vdots & \vdots \\ c_{m1} & \dots & c_{mr} \end{pmatrix}$ durch $c_{ij} = a_{i1}b_{1j} + \dots + a_{in}b_{nj}$
für $i = 1, \dots, m$ und $j = 1, \dots, r$. Es gilt $A \cdot B = M(m \times r, \mathbb{F}_p)$.

Bem.: Die Matrix-Vektor-Mult. ist ein Spezialfall, wenn man den Vektor als $n \times 1$ -Matrix auffasst.

Einheitsmatrix

$$E_n := \begin{pmatrix} 1 & \dots & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & \dots & 1 \end{pmatrix} \text{ heißt Einheitsmatrix und}$$

$$0_n := \begin{pmatrix} 0 & \dots & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & \dots & 0 \end{pmatrix} \text{ die Nullmatrix } 0_n.$$

Für $A \in M(n \times n, \mathbb{F}_p)$ ist $E_n \cdot A = A \cdot E_n = A$ und
 $0_n \cdot A = A \cdot 0_n = 0_n$.

Einheitsmatrix

$$E_n := \begin{pmatrix} 1 & \dots & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & \dots & 1 \end{pmatrix} \text{ heißt Einheitsmatrix und}$$

$$0_n := \begin{pmatrix} 0 & \dots & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & \dots & 0 \end{pmatrix} \text{ die Nullmatrix } 0_n.$$

Für $A \in M(n \times n, \mathbb{F}_p)$ ist $E_n \cdot A = A \cdot E_n = A$ und
 $0_n \cdot A = A \cdot 0_n = 0_n$.

E_n und 0_n sind vergleichbar mit 0, 1 in der modularen Arithmetik bzw. in $\mathbb{R}, \mathbb{Q}, \mathbb{Z}$ usw.

Matrizenmultiplikation

Für eine $n \times n$ -Matrix A definieren wir $A^0 = \begin{pmatrix} 1 & \dots & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & \dots & 1 \end{pmatrix}$

und $A^i := \underbrace{A \cdot \dots \cdot A}_{i \text{ mal}}, i \geq 1, i \in \mathbb{N}.$

Beispiel

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, B = \begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} \text{ über } \mathbb{F}_5.$$

Beispiel

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, B = \begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} \text{ über } \mathbb{F}_5. B + B = \begin{pmatrix} 3 & 3 \\ 0 & 2 \end{pmatrix},$$
$$A + B?$$

Beispiel

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, B = \begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} \text{ über } \mathbb{F}_5. B + B = \begin{pmatrix} 3 & 3 \\ 0 & 2 \end{pmatrix},$$

$A + B?$

$$A \cdot B = \begin{pmatrix} 4 & 0 \\ 0 & 1 \\ 4 & 0 \end{pmatrix}, B \cdot A?$$

$$\begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} \text{ und}$$

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Komposition und Multiplikation

Komposition und Multiplikation

Sind L_A, L_B die zu $A \in M(n \times r, \mathbb{F}_p)$, $B \in M(r \times m, \mathbb{F}_p)$ gehörigen linearen Abbildungen, so ist $L_{A \cdot B} = L_A \circ L_B$.

Beispiel

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 1 \end{pmatrix}, B = \begin{pmatrix} 4 & 4 \\ 0 & 1 \end{pmatrix} \text{ über } \mathbb{F}_5.$$

$$L_B \left(\begin{pmatrix} 4 \\ 0 \end{pmatrix} \right) := B \cdot \begin{pmatrix} 4 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ und}$$

$$L_A \left(\begin{pmatrix} 1 \\ 0 \end{pmatrix} \right) = A \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}.$$

$$L_{A \cdot B} \left(\begin{pmatrix} 4 \\ 0 \end{pmatrix} \right) = A \cdot B \left(\begin{pmatrix} 4 \\ 0 \end{pmatrix} \right) = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}.$$

Anwendung bei symmetrischer Krypto: Stromchiffren

Definition

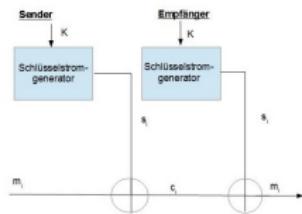
Mit $\{0, 1\}^*$ bezeichnen wir die Menge der Bitsequenzen beliebiger, endlicher Länge mit Einträgen aus \mathbb{F}_2 . Ist $m \in \{0, 1\}^*$, so ist $m = (m_0, \dots, m_{l-1})$, $m_i \in \mathbb{F}_2$, $0 \leq i \leq l - 1$, $l \in \mathbb{N}$, l geeignet gewählt.

Grundprinzip einer Stromchiffre.

Ansatz:

- ▶ Erzeuge aus einem geheimen symmetrischen Schlüssel k , z.B. der Länge 128 bit, einen langen Schlüsselstrom $s \in \{0, 1\}^*$.
- ▶ s wird via bitweiser binärer Addition zur Verschlüsselung genutzt.

Grundprinzip einer Stromchiffre

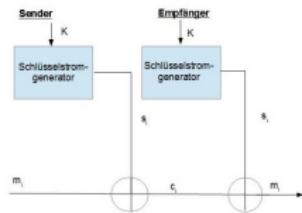


Nachrichten m und der Schlüsselstrom s aus $\{0, 1\}^*$, d.h.

$m = (m_0, \dots, m_{l-1}), s = (s_0, \dots, s_{l-1}), m_i, s_i \in \mathbb{F}_2, 0 \leq i \leq l-1, l \in \mathbb{N}$. Immer möglich!

- ▶ Verschlüsselung: $c_i = E_k(m_i) = m_i \oplus s_i$
- ▶ Entschlüsselung: $m_i = D_k(c_i) = c_i \oplus s_i,$
 $s_i, c_i, m_i \in \mathbb{F}_2$.

Grundprinzip einer Stromchiffre



Nachrichten m und der Schlüsselstrom s aus $\{0, 1\}^*$, d.h.

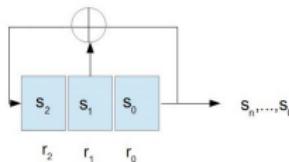
$m = (m_0, \dots, m_{l-1}), s = (s_0, \dots, s_{l-1}), m_i, s_i \in \mathbb{F}_2, 0 \leq i \leq l-1, l \in \mathbb{N}$. Immer möglich!

- ▶ Verschlüsselung: $c_i = E_k(m_i) = m_i \oplus s_i$
- ▶ Entschlüsselung: $m_i = D_k(c_i) = c_i \oplus s_i,$
 $s_i, c_i, m_i \in \mathbb{F}_2$.

Vorteil von \oplus :

- ▶ Einfach umzusetzen, da Addition=Subtraktion (z.B. $1 \oplus 1 = 0$).
 Verschlüsselungs- und Entschlüsselungsfunktion sind identisch.
- ▶ Das Design der Wahl bis spät in die 90er.
- ▶ Beliebt in Ransomware, da mittlerweile wieder „unorthodox“.

Ansatz: Linear Feedback Shift Register (LFSR)



- Die Rückkopplungsvorschrift ist wie folgt:

$$s_{i+3} = s_{i+1} + s_i \bmod 2, i \geq 0$$

Ausgabefolge s_i	r_2	r_1	r_0
0	1	0	0
0	0	1	0
1	1	0	1
0	1	1	0
1	1	1	1
1	0	1	1
1	0	0	1
0	1	0	0

Stromchiffren mittels LFSRs

Ausgabefolgen von LFSRs haben exzellente statische Eigenschaften

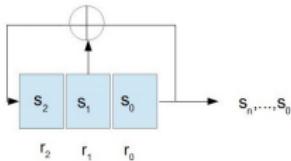
Mögliches Kryptosystem?

Stromchiffren mittels LFSRs

Ausgabefolgen von LFSRs haben exzellente statische Eigenschaften
Mögliches Kryptosystem?

- ▶ Geheimer Schlüssel k ist der Startvektor s_0, \dots, s_{n-1} , nicht alle $s_i = 0$ und a_0, \dots, a_{n-1} derart, dass die Rückkoppelungsvorschrift
$$s_{n+i} := \sum_{j=0}^{n-1} a_j s_{i+j}, i \geq 0, a_j \in \mathbb{F}_2,$$
stets m-Sequenzen (maximal periodisch) liefert.
- ▶ Verschlüsselung: $m_i \oplus s_i = c_i, i = 0, 1, \dots$
- ▶ Entschlüsselung: $c_i \oplus s_i = m_i, i = 0, 1, \dots$

Beispiel

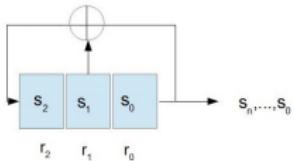


► $s_{i+3} = s_{i+1} + s_i \bmod 2, i \geq 0$

Ausgabefolge s_i	r_2	r_1	r_0
0	1	0	0
0	0	1	0
1	1	0	1
0	1	1	0
1	1	1	1
1	0	1	1
1	0	0	1
0	1	0	0

Die Verschlüsselung von 11111... ist 11010... (Warum?)

Beispiel



► $s_{i+3} = s_{i+1} + s_i \bmod 2, i \geq 0$

Ausgabefolge s_i	r_2	r_1	r_0
0	1	0	0
0	0	1	0
1	1	0	1
0	1	1	0
1	1	1	1
1	0	1	1
1	0	0	1
0	1	0	0

Die Verschlüsselung von 11111... ist 11010... (Warum?)
Wie entschlüsselt man 11010...?

Bezug zu Vektoren und Matrizen

Es sei s_0, s_1, \dots die Ausgabefolge und

$s_{i+n} = a_{n-1}s_{n-1+i} + \dots + a_0s_{0+i}, i \geq 0$. Es gilt:

$$\begin{pmatrix} s_{i+1} \\ s_{i+2} \\ \vdots \\ \vdots \\ s_{i+n} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \vdots \\ \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 \\ a_0 & a_1 & a_2 & \dots & a_{n-1} \end{pmatrix} \cdot \begin{pmatrix} s_i \\ s_{i+1} \\ \vdots \\ \vdots \\ s_{i+n-1} \end{pmatrix}$$

Bezug zu Vektoren und Matrizen

Es sei s_0, s_1, \dots die Ausgabefolge und

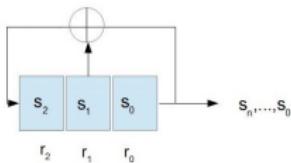
$s_{i+n} = a_{n-1}s_{n-1+i} + \dots + a_0s_{0+i}, i \geq 0$. Es gilt:

$$\begin{pmatrix} s_{i+1} \\ s_{i+2} \\ \vdots \\ \vdots \\ s_{i+n} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \vdots \\ \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 \\ a_0 & a_1 & a_2 & \dots & a_{n-1} \end{pmatrix} \cdot \begin{pmatrix} s_i \\ s_{i+1} \\ \vdots \\ \vdots \\ s_{i+n-1} \end{pmatrix}$$

Die Matrix $F = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \vdots \\ \vdots & \vdots & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & 1 \\ a_0 & a_1 & a_2 & \dots & a_{n-1} \end{pmatrix}$ heißt

Frobenius-Begleitmatrix.

Zurück zum Beispiel



- Die Rückkopplungsvorschrift ist wie folgt:

$$s_{i+3} = s_{i+1} + s_i \bmod 2, i \geq 0$$

$$F = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

Das i -te Bit $s_i, i \geq 0$ des Ausgabestroms erhält man durch die

Beziehung $\begin{pmatrix} s_i \\ s_{i+1} \\ s_{i+2} \end{pmatrix} = F^i \cdot \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$.

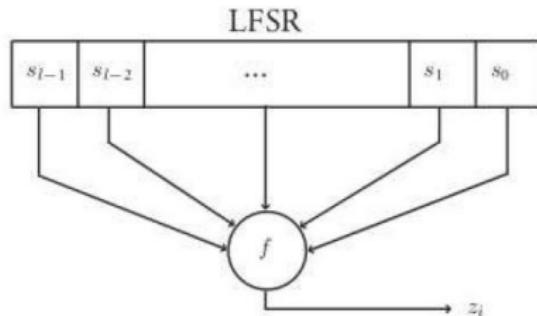
Warum dieser theoretische Aufwand

- ▶ Zentrale kryptoanalytische Eigenschaften von LFSRs werden durch die Frobeniusbegleitmatrix bestimmt bzw. gesteuert!
- ▶ Bei diesem Kryptosystem ergibt sich z.B. sofort, dass es total schwach ist! Es sollte in also auf keinen Fall in der Praxis eingesetzt werden.
 Gauß-Algorithmus.

Warum dieser theoretische Aufwand

- ▶ Zentrale kryptoanalytische Eigenschaften von LFSRs werden durch die Frobeniusbegleitmatrix bestimmt bzw. gesteuert!
- ▶ Bei diesem Kryptosystem ergibt sich z.B. sofort, dass es total schwach ist! Es sollte in also auf keinen Fall in der Praxis eingesetzt werden.
 Gauß-Algorithmus.
- ▶ Es gibt einen einfachen Angriff, wenn die Abgriffe a_i und n aufeinanderfolgende Ausgabebits s_i bekannt sind (Ü-Aufgabe).

Typisches Stromchiffrendesign: Gefilterte Schieberegister



- ▶ Vor der Ausgabe werden die Bits des LFSRs gefiltert.

Anwendung auf unser Beispiel

$$f(x_0, x_1, x_2) = x_0x_1 + x_1x_2 + x_0 + x_1 + x_2$$

- Die Rückkopplungsvorschrift ist wie folgt:

$$s_{i+3} = s_{i+1} + s_i \bmod 2, i \geq 0$$

Ausgabefolge $z_i = f(s_i, s_{i+1}, s_{i+2})$	r_2	r_1	r_0
1	1	0	0
1	0	1	0
0	1	0	1
1	1	1	0
1	1	1	1
1	0	1	1
1	0	0	1
1	1	0	0

Die Verschlüsselung von 11111... ist 00110... (Warum?).

Anwendung auf unser Beispiel

$$f(x_0, x_1, x_2) = x_0x_1 + x_1x_2 + x_0 + x_1 + x_2$$

- Die Rückkopplungsvorschrift ist wie folgt:

$$s_{i+3} = s_{i+1} + s_i \bmod 2, i \geq 0$$

Ausgabefolge $z_i = f(s_i, s_{i+1}, s_{i+2})$	r_2	r_1	r_0
1	1	0	0
1	0	1	0
0	1	0	1
1	1	1	0
1	1	1	1
1	0	1	1
1	0	0	1
1	1	0	0

Die Verschlüsselung von 11111... ist 00110... (Warum?).
Wie entschlüsselt man 00110...?

Kryptoanalyse von Stromchiffren

Die großen Angriffstechniken

- ▶ Brute-Force-Angriffe (TMTO, z.B. A5/1).
- ▶ Korrelationsangriffe (Siegenthaler, Meier, Staffelbach '85, '88).
- ▶ Angriffe via Convolutional Codes (T. Johannson, , '99).
- ▶ Algebraic attacks (Courtois, Meier, 2002).



„Good practice“:

- ▶ Entwickle keine Chiffren selbst
- ▶ Vertraue „selbstgemachten“ Chiffren nicht (Es sei denn man weiß genau was man tut, z.B. NSA, BND, GCHQ ...)

- ▶ Fortgeschrittene Krypto: Design und Kryptanalyse von Stromchiffren
- ▶ Computersicherheit: Richtiger Einsatz von Stromchiffren (immer noch schwer genug!)

Why bother?

- ▶ Gefilterte Schieberegister waren „en vogue“ bis in die späten 90er.
- ▶ Stromchiffren werden heutzutage in der mobilen Kommunikation eingesetzt, z.B. Snow3G oder ZUC (China).
- ▶  RC4 ist ebenfalls eine Stromchiffre, aber gilt als unsicher.
- ▶ Die Kryptoanalyse macht Spaß und die benötigte Mathematik ist moderat.

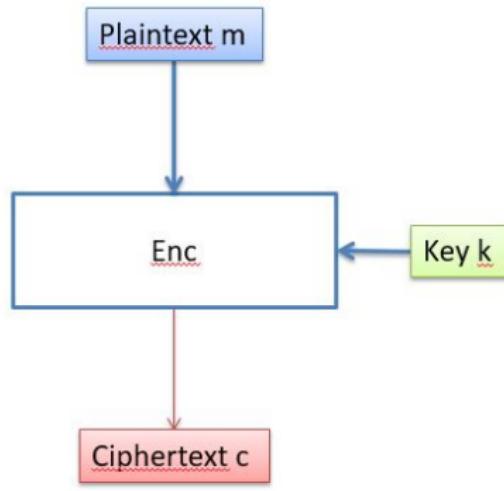
Why bother?

- ▶ Gefilterte Schieberegister waren „en vogue“ bis in die späten 90er.
- ▶ Stromchiffren werden heutzutage in der mobilen Kommunikation eingesetzt, z.B. Snow3G oder ZUC (China).
- ▶  RC4 ist ebenfalls eine Stromchiffre, aber gilt als unsicher.
- ▶ Die Kryptoanalyse macht Spaß und die benötigte Mathematik ist moderat.
- ▶  Lineare Abbildungen (Matrix-Vektor-Multiplikation) bzw. Stromchiffren sind wieder von essentieller Bedeutung für Exploit-Writing (z.B. Obfuscation), Cryptocracker, Hacking-Tools usw.

Why bother?

- ▶ Gefilterte Schieberegister waren „en vogue“ bis in die späten 90er.
- ▶ Stromchiffren werden heutzutage in der mobilen Kommunikation eingesetzt, z.B. Snow3G oder ZUC (China).
- ▶  RC4 ist ebenfalls eine Stromchiffre, aber gilt als unsicher.
- ▶ Die Kryptoanalyse macht Spaß und die benötigte Mathematik ist moderat.
- ▶  Lineare Abbildungen (Matrix-Vektor-Multiplikation) bzw. Stromchiffren sind wieder von essentieller Bedeutung für Exploit-Writing (z.B. Obfuscation), Cryptocracker, Hacking-Tools usw.
- ▶  Neue Forschungsansätze.

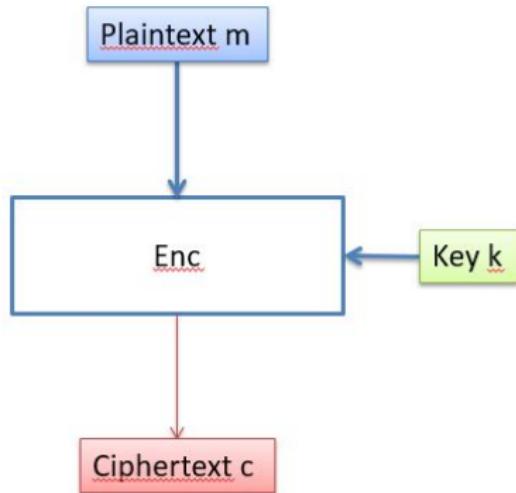
Blockchiffren



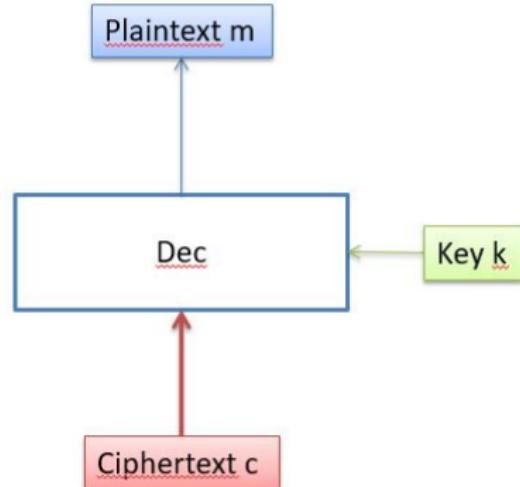
$$c = \text{Enc}(k, m)$$

- **Feste Eingabelänge**
 - Typisch:
 - $|m| = |c| = 64 \text{ bis } 128 \text{ bit}$
 - $|k| = 128 \text{ bis } 256 \text{ bit}$
 - $|m| = |c|$ ist die **Blocklänge**
- Ca. 10 kommerziell relevante Blockchiffren
 - z.B.: (3)DES, AES, Kasumi

Blockchiffren



$$c = \text{Enc}(k, m)$$

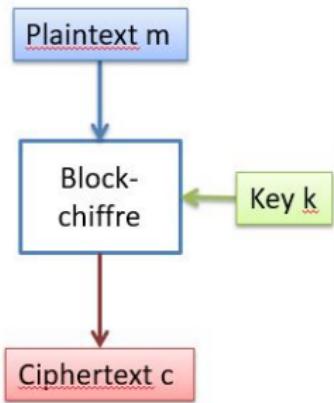


$$m = \text{Dec}(k, c)$$

- ▶ $m = \text{Dec}(k, \text{Enc}(k, m))$ für alle (m, k)

Beispiele für Blockchiffren

- Advanced Encryption Standard AES (2000)
 - Blocklänge = 128 bit
 - $|k| = 128$ oder 192 oder 256 bit
- Data Encryption Standard DES (1977)
 - Blocklänge = 64 bit
 - $|k| = 56$ bit
- 3DES-EDE
 - Encrypt-decrypt-encrypt with DES
 - Different keys for Enc / Dec
 - Blocklänge = 64 bit
 - $|k| = 112$ bit



Kryptoanalyse von Blockchiffren

Die großen Angriffstechniken

- ▶ Differential cryptanalysis (Biham, Shamir 1991).
- ▶ Linear cryptanalysis (Matsui 1992).
- ▶ Brute-Force über n bit Schlüssel. Im Schnitt 2^{n-1} . ~~mit~~ TMTD-Techniken.
- ▶ Algebraic attacks (Albrecht, Cid, Courtois, 2002).



„Good practice“:

- ▶ Entwickle keine Chiffren selbst.
- ▶ Vertraue „selbstgemachten“ Chiffren nicht. (Es sei denn man weiß genau was man tut, z.B. NSA, BND GCHQ ...).
- ▶ Fortgeschrittene Krypto: Design und Kryptanalyse von Blockchiffren.
- ▶ Computersicherheit: Richtiger Einsatz von Blockchiffren (immer noch schwer genug!).

Verschlüsselung mit Blockchiffren: Operationsmodi

- ▶ Electronic Code Book Modus (ECB).
- ▶ Cipher Block Chaining Modus (CBC).
- ▶ Output Feedback Modus (OFB).
- ▶ Cipher Feedback Modus (CFB).
- ▶ Counter Modus (CTR)/Galois Counter Modus (GCM).

Electronic Code Book Modus (ECB-Mode)

Es sei $E_K : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ eine Blockchiffre mit Blockbreite n bit , z.B. 128 bit.

- ▶ Der Klartext M , z.B. eine E-Mail, wird in l Blöcke m_i der Breite n aufgeteilt.
☞ Ggf. Padding, z.B. Sequenz von 0, falls der letzte Block kürzer ist, d.h. falls Textlänge mod $n = r \neq 0$ ist.
- ▶ Separate Verschlüsselung jedes Blocks m_i vermöge $E_K(m_i) = c_i$. Der geheime Schlüssel K ist stets derselbe.

Electronic Code Book Modus (ECB)

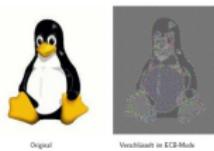
Vorteile:

- ▶ Keine blockweise Synchronisierung zwischen Sender und Empfänger notwendig.
- ▶ Übertragungsfehler wirken nur auf dem betreffenden Block.
- ▶ Parallelisierbar.

Electronic Code Book Modus (ECB)

Vorteile:

- ▶ Keine blockweise Synchronisierung zwischen Sender und Empfänger notwendig.
- ▶ Übertragungsfehler wirken nur auf dem betreffenden Block.
- ▶ Parallelisierbar.



Schwächen

Electronic Code Book Modus (ECB)

Vorteile:

- ▶ Keine blockweise Synchronisierung zwischen Sender und Empfänger notwendig.
- ▶ Übertragungsfehler wirken nur auf dem betreffenden Block.
- ▶ Parallelisierbar.



Schwächen

- ▶ Gleiche Klartextblöcke werden immer auf gleiche Chiphertextblöcke abgebildet. ↗ Man erhält aus dem Schlüsseltext Informationen über den Klartext ohne diesen entschlüsseln zu müssen.
- ▶ Angreifer kann durch einfügen von mit gleichem Schlüssel chiffrierten Text Nachricht ändern.
- ▶ Angreifer kann unbemerkt die Reihenfolge der Schlüsseltextblöcke verändern.

Electronic Code Book Modus (ECB)

Vorteile:

- ▶ Keine blockweise Synchronisierung zwischen Sender und Empfänger notwendig.
- ▶ Übertragungsfehler wirken nur auf dem betreffenden Block.
- ▶ Parallelisierbar.



Schwächen

- ▶ Gleiche Klartextblöcke werden immer auf gleiche Chiphertextblöcke abgebildet. Man erhält aus dem Schlüsseltext Informationen über den Klartext ohne diesen entschlüsseln zu müssen.
- ▶ Angreifer kann durch einfügen von mit gleichem Schlüssel chiffrierten Text Nachricht ändern.
- ▶ Angreifer kann unbemerkt die Reihenfolge der Schlüsseltextblöcke verändern.
- ▶ Der Einsatz des ECB-Modus sollte gut überlegt sein.

Electronic Code Book Modus (ECB)

Vorteile:

- ▶ Keine blockweise Synchronisierung zwischen Sender und Empfänger notwendig.
- ▶ Übertragungsfehler wirken nur auf dem betreffenden Block.
- ▶ Parallelisierbar.



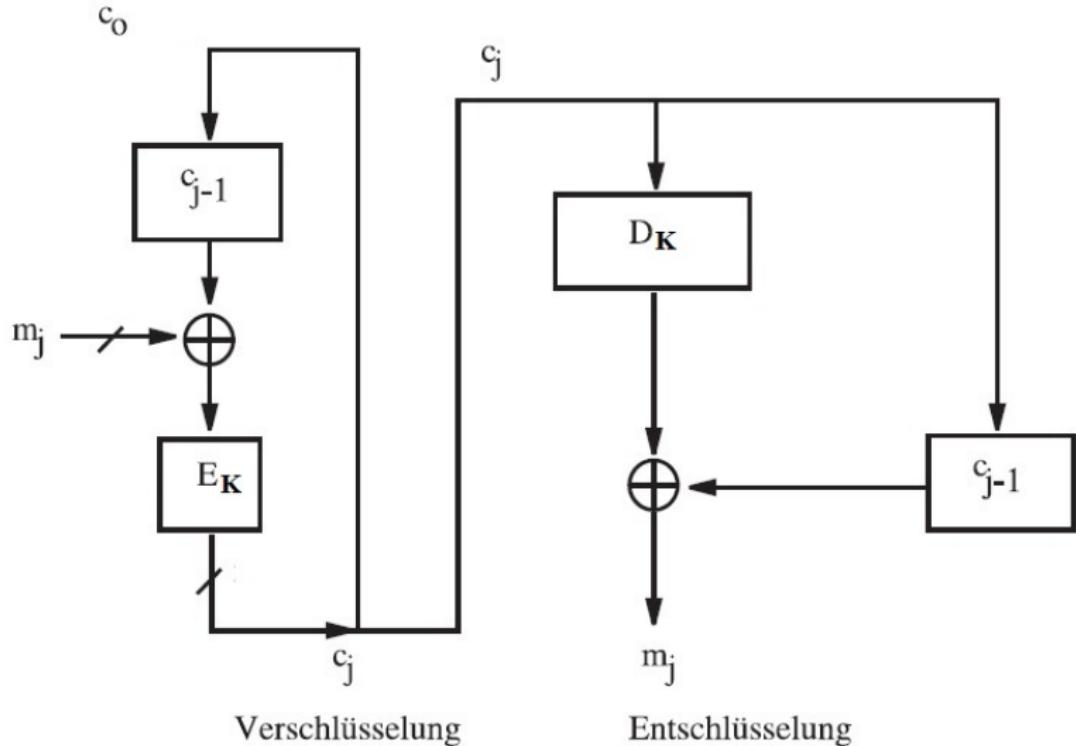
Schwächen

- ▶ Gleiche Klartextblöcke werden immer auf gleiche Chiphertextblöcke abgebildet. Man erhält aus dem Schlüsseltext Informationen über den Klartext ohne diesen entschlüsseln zu müssen.
- ▶ Angreifer kann durch einfügen von mit gleichem Schlüssel chiffrierten Text Nachricht ändern.
- ▶ Angreifer kann unbemerkt die Reihenfolge der Schlüsseltextblöcke verändern.
- ▶ Der Einsatz des ECB-Modus sollte gut überlegt sein.

Cipherblock Chaining Mode (CBC-Mode)

- ▶ Ein Verschlüsselungsverfahren für beliebig lange Texte.
- ▶ Wieder Aufteilung des Textes in l Blöcke der Breite n
- ▶ Verschlüsselung: Man wählt einen zufälligen Initialisierungsvektor $c_0 \in \mathbb{F}_2^n$ der offen übertragen wird und verschlüsselt via
$$c_j = E_K(c_{j-1} \oplus m_j), 1 \leq j \leq l.$$
- ▶ Entschlüsselung: $m_j = c_{j-1} + D_K(c_j), 1 \leq j \leq l.$
☞ c_0 ist dem Empfänger bekannt.

Cipherblock Chaining Mode (CBC-Mode)



Vorteile

- ▶ Erster Chitextblock c_1 hängt vom Klartext m_1 und dem $IV = c_0$ ab.
Zweite Chitextblock m_2 hängt vom IV und c_1, c_2 ab, usw.
- ▶ Durch die Wahl eines neuen $IV = c_0$ bei jeder Verschlüsselung wird der CBC-Modus zu einem probabilistischen Verschlüsselungsverfahren, d.h. die Chiffrate zweier Klartexte sind vollkommen unterschiedlich.
- ▶ Der IV kann und wird offen übertragen.

Nachteile:

- ▶ Bei fehlerhafter Übertragung (bereits bei 1-Bit-Fehler) des j -ten Blocks sind die Blöcke j und $j + 1$ nicht dechiffrierbar.
- ▶ Padding Oracle Angriffe.

Praktische Schlüsselgrößen

- ▶ Block- und Stromchiffren sollten, allein schon auf Grund von Brute-Force-Angriffen, mit mindestens 128 bit Schlüssellänge betrieben werden.

MAC = Message Authentication Code

- ▶ Authentifikationsverfahren: MACs werden zwecks Authentisierung an eine Nachricht gehängt. Der Empfänger kann diese dann authentifizieren und dabei die Integrität (Nachricht wurde nicht manipuliert) und den Ursprung der Nachricht bis zu einem gewissen Grad (s.u.) prüfen.
- ▶ MACs basieren auf symmetrischer Verschlüsselung.
- ▶ MACs können für beliebig lange Nachrichten gebildet werden.
- ▶ MACs selber haben eine feste Länge.
- ▶ Im Gegensatz zu Signaturen gewährleisten MACs nicht die „nicht Abstreitbarkeit“, da sie auf symmetrischer Kryptographie basieren.

MACs mittels Blockchiffren

- ▶ Verwende einfach z.B. AES im CBC-Mode mit $IV = 0$.

MACs mittels Blockchiffren

- Verwende einfach z.B. AES im CBC-Mode mit $IV = 0$.

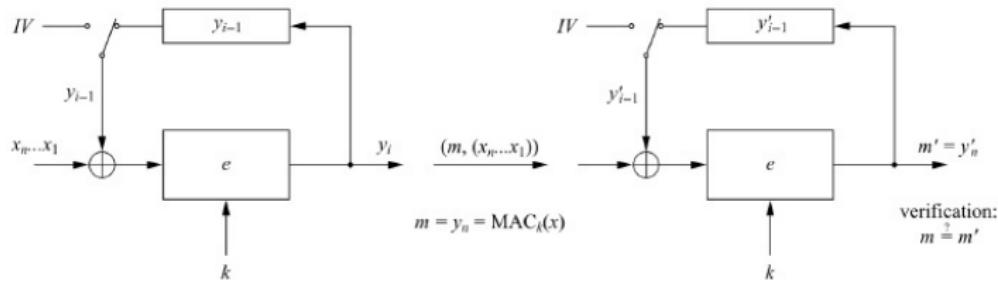
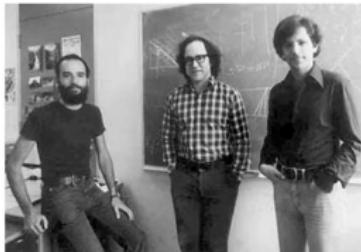


Abbildung: Kryptografie verständlich, C. Paar und J. Pelzl

Das RSA Kryptosystem

- ▶ Ron Rivest, Adi Shamir, Leonard Adleman
Clifford Cocks from GCHQ hat bereits 1973 ein äquivalentes System entwickelt (geheim bis 1997).
- ▶ Turing Award 2002.



The Story Behind RSA

They tried many approaches including „knapsack-based“ and „permutation polynomials“. For a time they thought it was impossible for what they wanted to achieve due to contradictory requirements. In April 1977, they spent Passover at the house of a student and drank a good deal of Manischewitz wine before returning to their home at around midnight. Rivest, unable to sleep, lay on the couch with a math textbook and started thinking about their one-way function. He spent the rest of the night formalizing his idea and had much of the paper ready by daybreak. The algorithm is now known as RSA.

Quelle: https://en.wikipedia.org/wiki/RSA_%28cryptosystem%29
and references therein

Das RSA-Verfahren

Zwei wesentliche Anwendungen für RSA:

- ▶ Transport von (symmetrischen) Schlüsseln.
- ▶ Digitale Signaturen.

Der Satz von Euler

Ein für das RSA-Verfahren wichtiger Satz aus der Mathematik ist der folgende

Satz von Euler

Gegeben sei n eine natürliche Zahl. Es bezeichne $\phi(n) := |\mathbb{Z}/n\mathbb{Z}^*|$ die Anzahl der Elemente der primen Restklassengruppe. ϕ heißt die Euler'sche ϕ -Funktion. Dann gilt für jedes $a \in \mathbb{Z}/n\mathbb{Z}^*$
 $a^{\phi(n)} = 1 \text{ mod } n.$

Der Satz von Euler

Ein für das RSA-Verfahren wichtiger Satz aus der Mathematik ist der folgende

Satz von Euler

Gegeben sei n eine natürliche Zahl. Es bezeichne $\phi(n) := |\mathbb{Z}/n\mathbb{Z}^*|$ die Anzahl der Elemente der primen Restklassengruppe. ϕ heißt die Euler'sche ϕ -Funktion. Dann gilt für jedes $a \in \mathbb{Z}/n\mathbb{Z}^*$ $a^{\phi(n)} = 1 \bmod n$.

Bsp: $n = 10, a = 3$. $3^{\phi(10)} = 3^{(2-1)(5-1)} = 3^4 = 81 = 1 \bmod 10$.

Der Satz von Euler

Ein für das RSA-Verfahren wichtiger Satz aus der Mathematik ist der folgende

Satz von Euler

Gegeben sei n eine natürliche Zahl. Es bezeichne $\phi(n) := |\mathbb{Z}/n\mathbb{Z}^*|$ die Anzahl der Elemente der primen Restklassengruppe. ϕ heißt die Euler'sche ϕ -Funktion. Dann gilt für jedes $a \in \mathbb{Z}/n\mathbb{Z}^*$ $a^{\phi(n)} = 1 \bmod n$.

Bsp: $n = 10, a = 3$. $3^{\phi(10)} = 3^{(2-1)(5-1)} = 3^4 = 81 = 1 \bmod 10$.

Korollar

Ist $n = pq$, p, q prim, so ist $\phi(n) = (p-1)(q-1)$ und für $a \in \mathbb{Z}/n\mathbb{Z}^*$ gilt $a^{(p-1)(q-1)} = 1 \bmod n$.

Schlüsselerzeugung:

$P, Q \leftarrow$ Primes

$$N = PQ$$

Wähle e mit $\text{ggT}(e, (P-1)(Q-1))=1$

$$d = e^{-1} \bmod (P-1)(Q-1)$$

$$\text{sk}_B = (N, d)$$

$$\text{pk}_B = (N, e)$$



Encryption(pk_B, m):
 $c = m^e \bmod N$

$$\text{pk}_B$$



$$c$$

Decryption($\text{sk}_B, (c_0, c_1)$):
 $m = c^d \bmod N$

$$\blacktriangleright c^d = (m^e)^d = m^{ed} \bmod (P-1)(Q-1) \quad \underbrace{\equiv m^1}_{\text{Satz von Euler}} = m \bmod N$$

Beispiel

1. Wähle als Primzahlen $p = 2$ und $q = 11$, d.h. $n = 22$.
2. Es ist $\phi(22) = 10$.
3. Aus $\mathbb{Z}/10\mathbb{Z}^*$ wähle den öffentlichen Schlüssel, hier $e = 3$.
4. Bestimme das multiplikative Inverse $d = 7$ von e modulo 10 ($3 \cdot 7 = 21 = 1 \bmod 10$).

Beispiel

1. Wähle als Primzahlen $p = 2$ und $q = 11$, d.h. $n = 22$.
 2. Es ist $\phi(22) = 10$.
 3. Aus $\mathbb{Z}/10\mathbb{Z}^*$ wähle den öffentlichen Schlüssel, hier $e = 3$.
 4. Bestimme das multiplikative Inverse $d = 7$ von e modulo 10 ($3 \cdot 7 = 21 = 1 \bmod 10$).
- Öffentlicher Schlüssel $(e, n) \rightarrow (3, 22)$.
- Privater Schlüssel $(d, n) \rightarrow 7$

Klartext zur Verschlüsselung: $m = 2$

- Mit dem öffentlichen Schlüssel $(3, 22)$ berechne c durch $c = 2^3 = 8 \bmod 22$.
- Mit dem privaten Schlüssel 7 berechne m durch $c^7 = 8^7 = 2 \bmod 22 = m$.

Das RSA Problem

- ▶ RSA Problem 1: Geg. N, e, y bestimme Klartext x mit $x^e = y \pmod{N}$.
- ▶ Bester bekannter Algorithmus: Berechne P, Q mit $N = PQ$ (Faktorisierungsproblem) und daraus d mittels Berechne $d = e^{-1} \pmod{(P-1)(Q-1)}$
Beispiel (Faktorisierungsproblem RSA):
Berechne P, Q so dass $N = PQ$ für $N = 91$

- ▶ Zunehmend schwierig für größere Zahlen
- ▶ In der Praxis: $N \approx 2^{2048}$

„Lehrbuch“-Signaturen mit RSA

Schlüsselerzeugung:

$P, Q \leftarrow$ Primes

$$N = PQ$$

Wähle e mit $\text{ggT}(e, (P-1)(Q-1)) = 1$

$$d = e^{-1} \bmod (P-1)(Q-1)$$

$$\text{sk}_B = (N, d)$$

$$\text{pk}_B = (N, e)$$

$$\text{pk}_{\text{Bob}}$$



Message m
 $s = m^d \bmod N$



$$(m, s)$$

Accept signature, if
 $s^e \bmod N = m$

$$\blacktriangleright s^e = (m^d)^e = m^{ed} \bmod (P-1)(Q-1) = m^1 = m \bmod N$$

„Lehrbuch“-Signaturen mit RSA

Schlüsselerzeugung:

$P, Q \leftarrow$ Primes

$$N = PQ$$

Wähle e mit $\text{ggT}(e, (P-1)(Q-1)) = 1$

$$d = e^{-1} \bmod (P-1)(Q-1)$$

$$\text{sk}_B = (N, d)$$

$$\text{pk}_B = (N, e)$$

$$\text{pk}_{\text{Bob}}$$



Message m
 $s = m^d \bmod N$



$$(m, s)$$

Accept signature, if
 $s^e \bmod N = m$

$$\blacktriangleright s^e = (m^d)^e = m^{ed} \bmod (P-1)(Q-1) = m^1 = m \bmod N$$

Praktische Schlüsselgrößen für RSA

- ▶ Für einige Zeit durfte daher eine Schlüssellänge von 2048 bit eine völlig ausreichende Sicherheit bieten, allerdings gibt es...

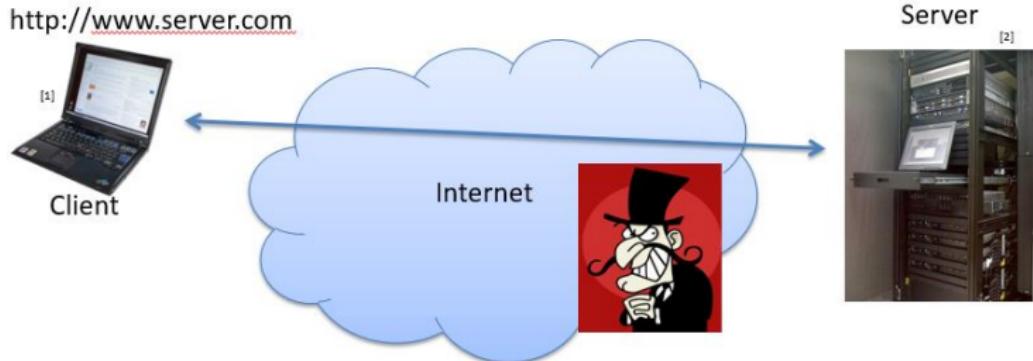
Praktische Schlüsselgrößen für RSA

- ▶ Für einige Zeit dürfte daher eine Schlüssellänge von 2048 bit eine völlig ausreichende Sicherheit bieten, allerdings gibt es...
- ▶ ... Quantencomputer!?
 - 💀 Diese können das Faktorisierungsproblem effizient lösen. Theoretische Komplexität $\mathcal{O}(\log^3 n)$.

Offen: Es ist unklar, ob es Quantencomputer im annähernd nennenswertem Leistungsumfang geben wird. Es ist nicht einmal klar, ob es sie in relevanter Zukunft geben wird.

💀 Eine sorgfältige Kryptoanalyse ist nach wie vor notwendig.

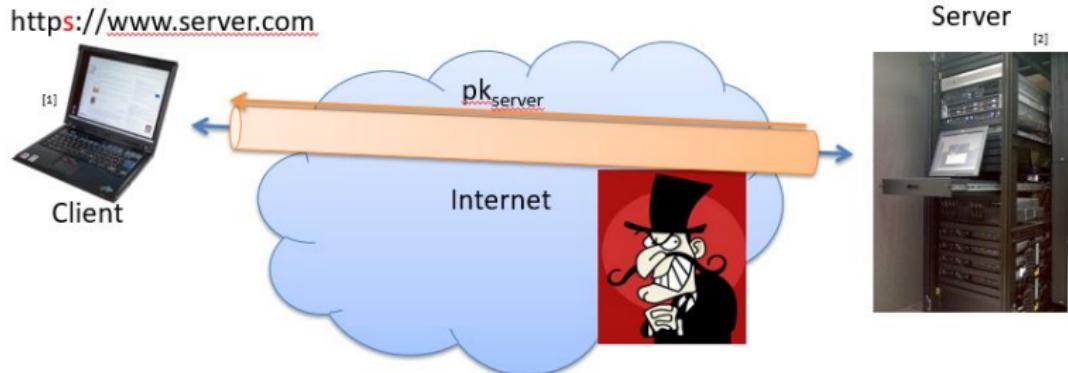
Anwendung: Sichere Kommunikation im Internet



Ein Angreifer kann prinzipiell alle Daten lesen und ändern

[1] Quelle: <http://de.wikipedia.org/wiki/Notebook>
[2] Quelle: <http://de.wikipedia.org/wiki/19%22>

Anwendung: Sichere Kommunikation im Internet

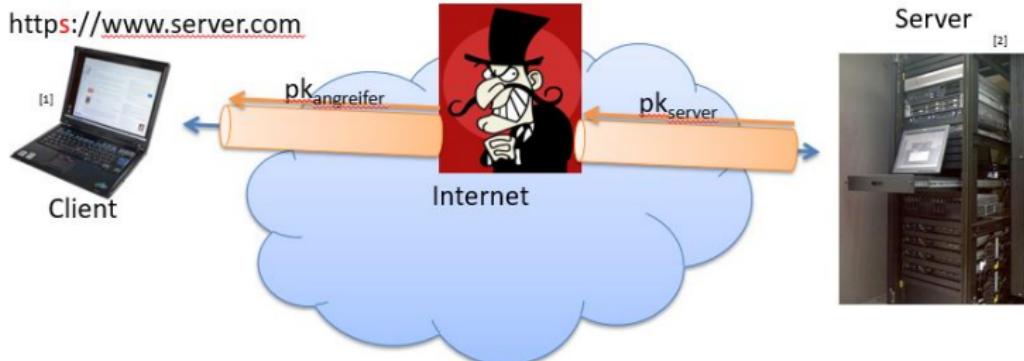


- Lösung: Verschlüsselung und Authentifikation
 - Zum Beispiel mit TLS
 - Server sendet pk zum Client

[1] Quelle: <http://de.wikipedia.org/wiki/Notebook>

[2] Quelle: <http://de.wikipedia.org/wiki/19%22>

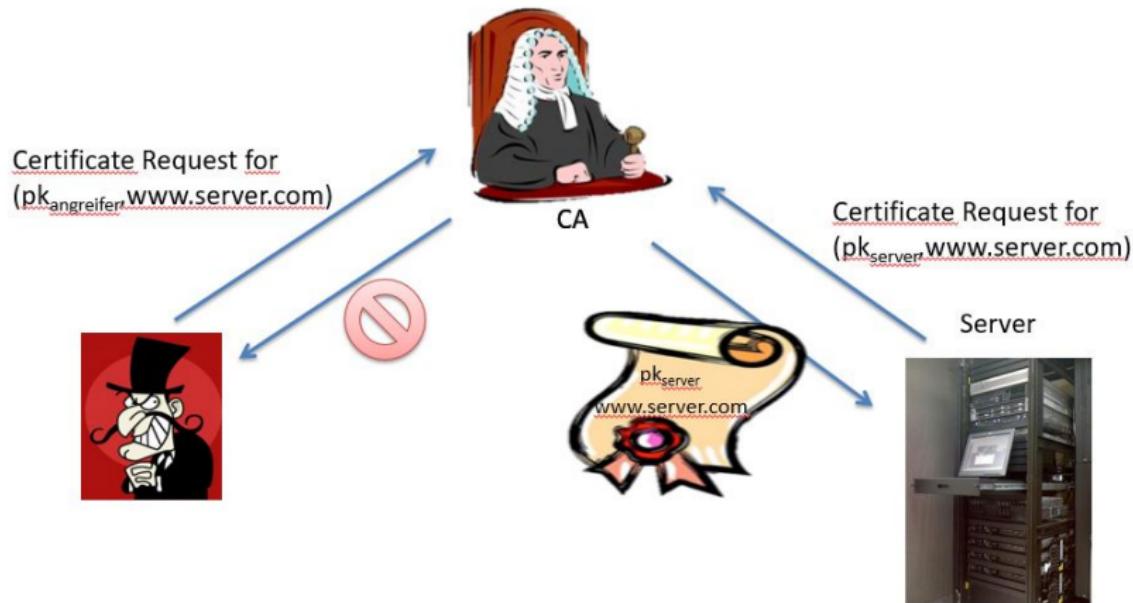
Man-in-the-Middle Angriff



Authentizität von Public Keys!

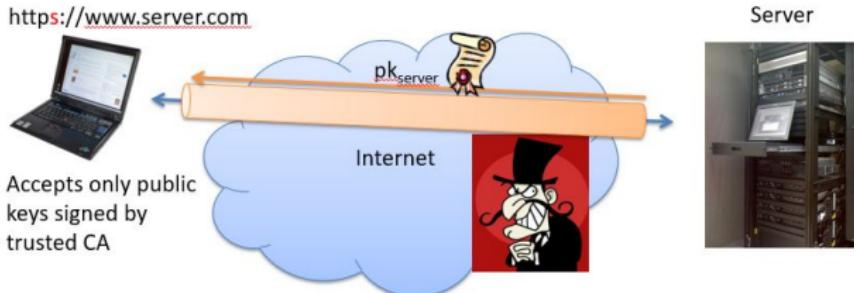
- [1] Quelle: <http://de.wikipedia.org/wiki>Notebook>
[2] Quelle: <http://de.wikipedia.org/wiki/19%22>

Certification Authorities



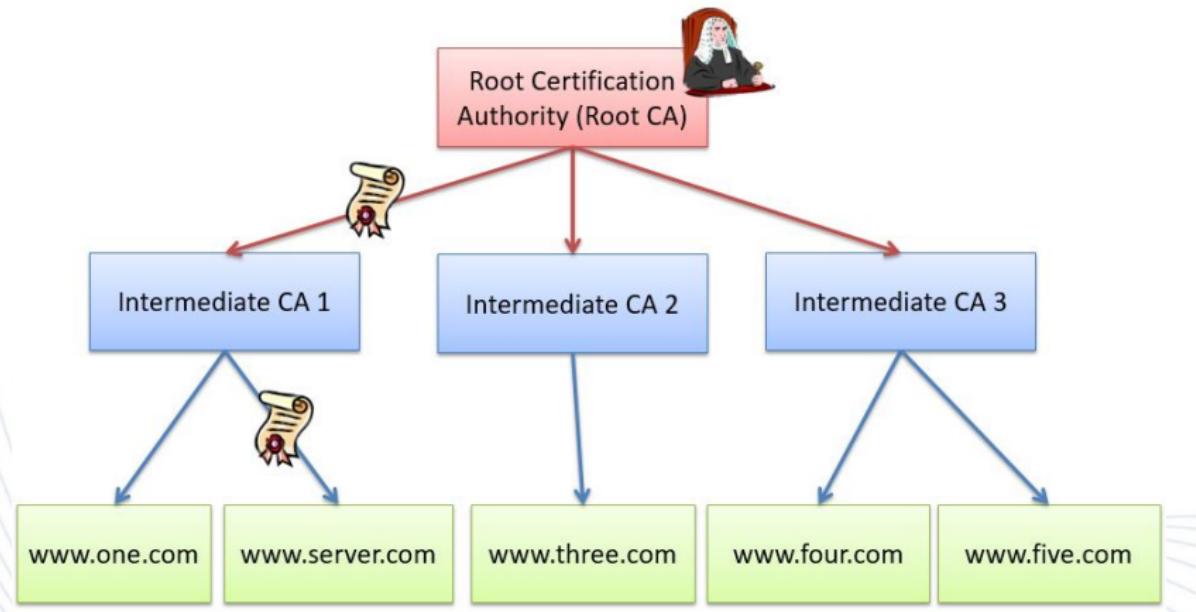
☞ Unter <https://letsencrypt.org/> kann man kostenlose Zertifikate ausstellen lassen.

Certification Authorities

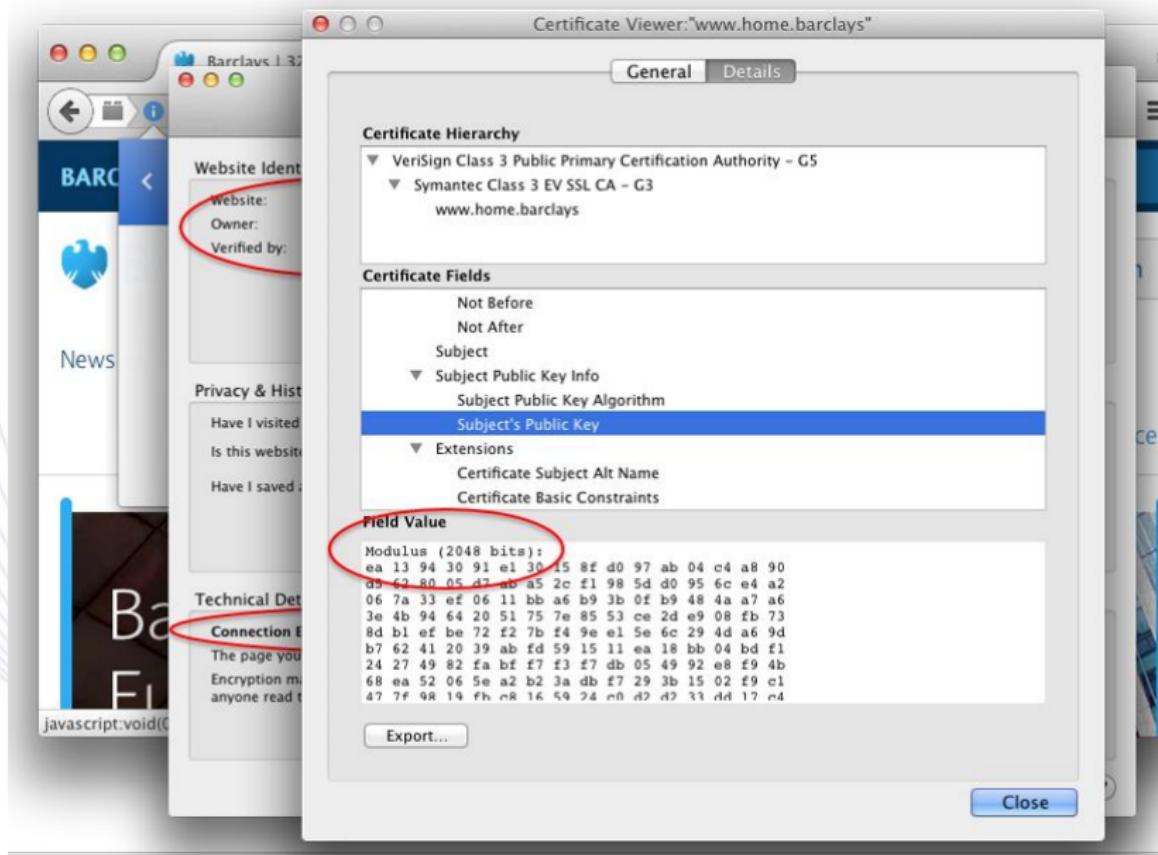


- ▶ Server hat einen zertifizierten pk (signiert von einer vertrauenswürdigen CA)
- ▶ Der Angreifer erhält keinen zertifizierten pk von der CA
💀 Könnte sich durch Let's encrypt ändern.

Zertifikatshierarchien



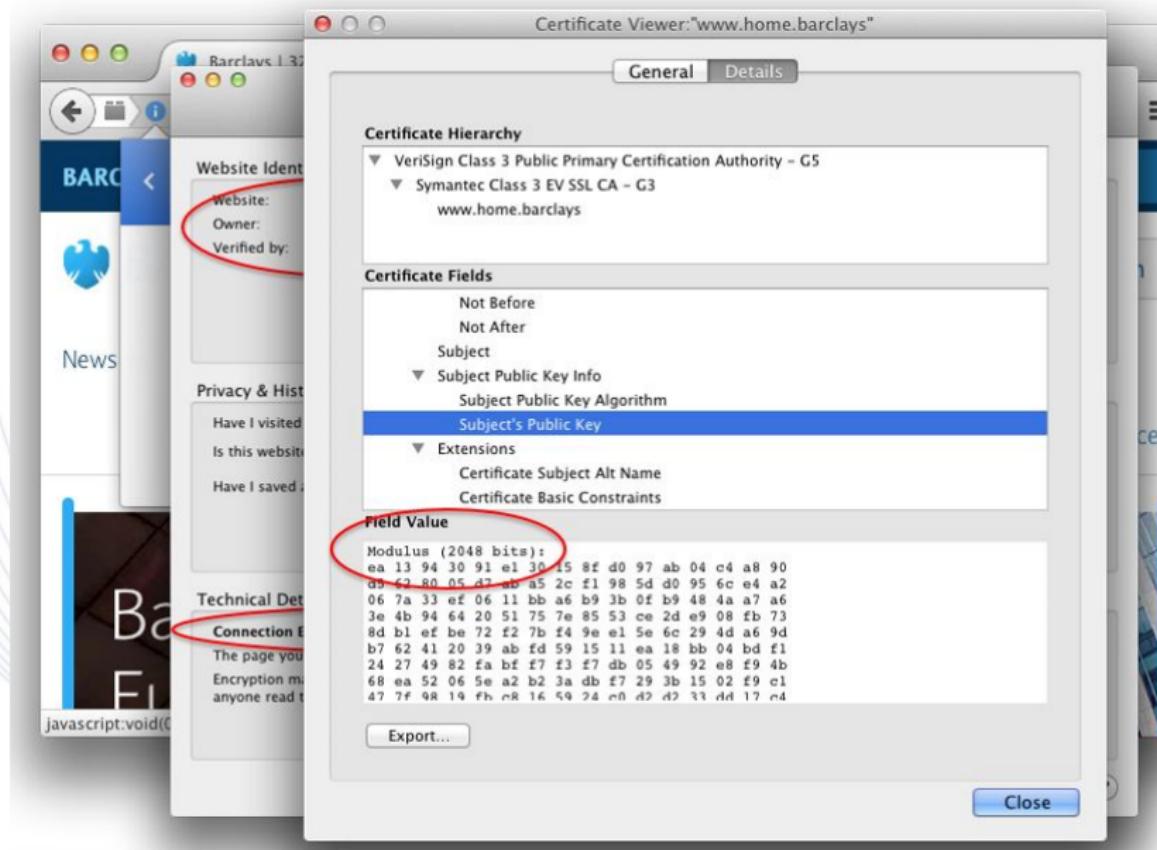
Beispiel: Zertifikate im Browser



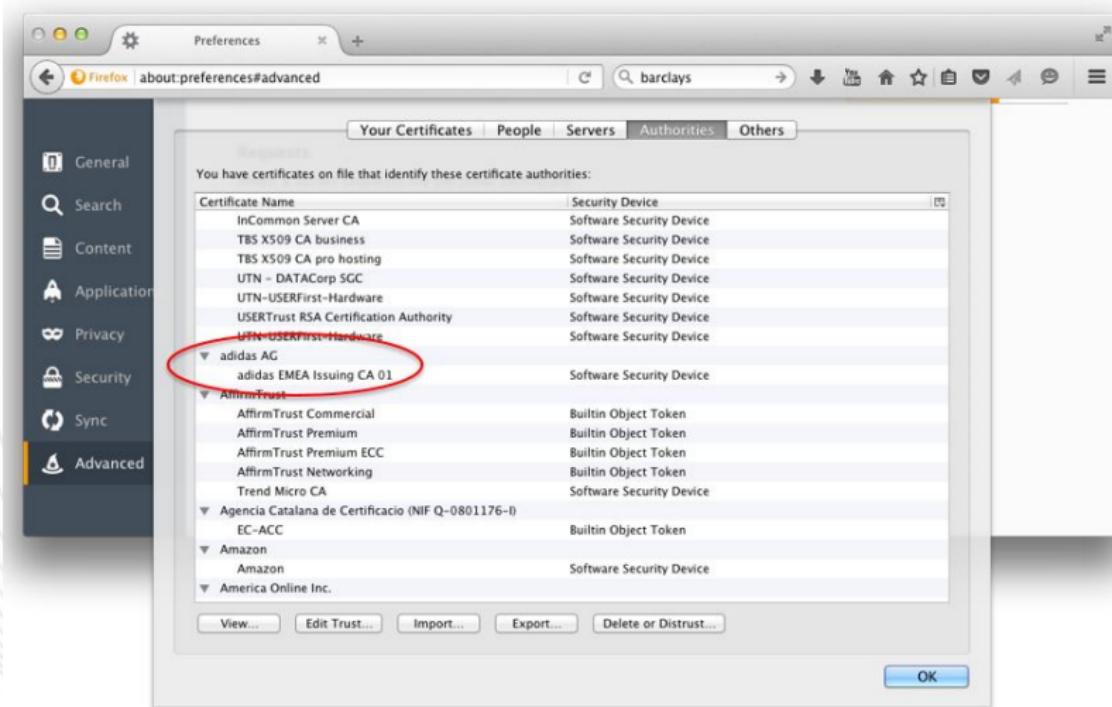
Probleme

- ▶ Jede CA kann Zertifikate für jede Domain ausstellen.
- ▶ Jede CA kann unerkannte MITM-Angriffe auf TLS-Verbindungen durchführen.
- ▶ Ein Nutzer muss allen CAs bzw. Root-Zertifikaten vertrauen, die in seinem Browser installiert sind Hunderte in modernen Browsern.
- ▶ Alle CAs müssen absolut vertrauenswürdig sein.

Beispiel: Root-Zertifikate im Browser



Beispiel: Root-Zertifikate im Browser





- ▶ Zertifikate von CAs kosten Geld.
- ▶ Eigene Zertifikate werden von Browsern nicht akzeptiert.



- ▶ Zertifikate von CAs kosten Geld.
- ▶ Eigene Zertifikate werden von Browsern nicht akzeptiert.
- ▶ Let's Encrypt stellt kostenlose, flächendeckend akzeptierte Zertifikate aus.
- ☠ Mißbrauch.

Probleme

- ▶ CAs machen Fehler.
- ▶ Zertifikate, die „unsicher“ werden bevor sie ihre Gültigkeit verlieren, müssen umständlich zurückgezogen werden



Turktrust Certificate Authority Errors Demonstrate The Risk of “Subordinate” Certificates

JANUARY 3, 2013 BY STEVE SCHULTZE

Update: More details have continued to come out, and I think that they generally support the less-paranoid version of events. There continues to be discussion on the [mozilla.dev.security.policy](#) list, Turktrust has [given more details](#), and Mozilla has just opened up for public viewing their own [detailed internal response documentation](#) (including copies of all of the certs in question). None of this changes the fundamental riskiness of subordinate certificates, or the improvements that should be made to the CA system. It just means that in this case, the failure didn't progress to a full-blown meltdown.

Today, the public learned of another failure by a Certificate Authority—one of the companies that certifies SSL-encryption for our internet communications. (See the end of this post for a catalogue of our past writing on problems with this “CA” system.) This time, the company **Turktrust** was revealed to have [issued two subordinate certificates](#) (also known as “intermediate” certificates) to entities that should not have had them. [Subordinate certificates are very powerful. They give the holder the ability to issue SSL certificates for any domain name as though they have control of the parent CA’s “root” certificate. In this case, Google discovered that one of Turktrust’s previously undisclosed subordinate certificates had issued SSL certificates for the domain gmail.com, and that these certificates had been used to intercept Gmail users’ traffic... somewhere.](#) This is where the details get foggy, but [Turktrust has begun to describe their version of events.](#)

- ▶ Die Art und Weise, wie PKIs im Internet benutzt werden hat zahlreiche Schwächen.
☠ Änderungen sind schwierig. (Backwards Compatibility)

- ▶ Die Art und Weise, wie PKIs im Internet benutzt werden hat zahlreiche Schwächen.
💀 Änderungen sind schwierig. (Backwards Compatibility)
- ▶ Trotzdem: Insgesamt funktioniert es überraschend gut in der Praxis... (mit Ausnahmen)



Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundsatz

Der „Hackerparagraph“

„Mit der Einführung des § 202 c (sog. Hackerparagraph) und weiterer IT-spezifischer Regelungen in das Strafgesetzbuch (StGB) im August 2007 wurden Vorgaben der Europäischen Union zur Bekämpfung von Computerkriminalität (Cybercrime) in deutsches Recht umgesetzt. Bezogen auf den § 202c StGB bedeutete dies, dass nicht nur das unberechtigte Beschaffen bzw. Manipulieren von Daten Dritter sondern nunmehr bereits die reine Vorbereitungshandlung hierzu unter Strafe gestellt war. Der Gesetzgeber zielte damit insbesondere auf die Herstellung, Beschaffung oder Verbreitung von Software ab, die dem Anwender der Software auf strafbare Weise Zugang zu Daten verschafft, die nicht für ihn bestimmt waren. Erfuhren die Strafverfolgungsbehörden von der Existenz derartiger Programme, müssten Sie ermitteln. StGB, d.h. §202c war ein Offizialdelikt!“

Quelle: Heise Verlag

Dual-Use-Problematik

- ▶ Auch rechtsschaffende Softwareanbieter und Anwender können u. U. in die Nähe der Kriminalität geraten.
- ▶ Diese Tatbestandsvoraussetzung (StGB) erfüllen auch Sicherheitsforscher, Penetrationstester, Netzwerk-Administratoren und Software-Qualitätsprüfer!
- ▶ Problem: Nur durch die Nutzung solcher Programme (z.B. Metasploit) lassen sich z.B. Exploits und Sicherheitslücken in den Programmen erkennen!
Ein Exploit (englisch, to exploit = ausnutzen) ist in der IT eine systematische Möglichkeit, Schwachstellen, die bisher nicht gefunden wurden, auszunutzen.

☞ Hier gibt es rechtliche Regelungen.

Wichtig: Nicht einfach „loshacken“!!!

Allgemeine Angriffsarten

Passive Angriffe: Keine Veränderungen an Systemen oder Daten werden vorgenommen.

- ▶ Mitlesen des Datenverkehrs, z.B. E-Mails, Konfigurationseinstellungen von Software oder Hardware, Authentifizierungsdaten.
- ▶ Analysieren des Kommunikationsverhaltens, z. B. über Tracking, Metadatenanalyse
- ▶ :

Aktive Angriffe: Hier werden aktiv Veränderungen vorgenommen.

- ▶ Absender oder Inhalte von E-Mails verfälschen oder Daten umleiten.
- ▶ Unterbrechen der Systemerreichbarkeit, z. B. wenn ein Webserver aktiv überlastet wird.
- ▶ Zusenden bzw. Download von Schadsoftware.
- ▶ Entfernen von Daten, z. B. während der Datenübertragung.

Allgemeine Angriffsarten

Angriffe auf die IT-Sicherheit können aber auch danach unterschieden werden, auf welche Systemkomponenten sie sich beziehen, entweder als Mittel zum Zweck, oder als Ziel selbst.

Allgemeine Angriffsarten

Angriffe auf die IT-Sicherheit können aber auch danach unterschieden werden, auf welche Systemkomponenten sie sich beziehen, entweder als Mittel zum Zweck, oder als Ziel selbst.

- ▶ Software: z. B. Manipulation von Anwendungssoftware oder des Betriebssystems, Kompromittierung von Verschlüsselungssystemen.
- ▶ Netze: z. B. ausnutzen von Schwachstellen, um über Netze in andere Systeme einzudringen.
- ▶ Daten und Informationen: z. B. abgreifen, verändern, löschen, hinzufügen von Informationen (Meta- oder Systemdaten).
- ▶ Hardware: Manipulation einzelner Hardware-Komponenten, z.B. Kameras in Rechnern oder Manipulation von Steuerungssystemen wie bei Stuxnet.
- ▶ Identität: Angriffe können sich auch auf die Identität beziehen, z. B. Aushebeln von Authentifizierungsmechanismen.

 Vgl. Schutzziele.

Typen von Angreifern

Böswillige Angreifer: Eigenen Vorteil auch unter Schaden durchsetzen, einfach nur Schaden zufügen, illegaler Informationshandel...

Typen von Angreifern

Böswillige Angreifer: Eigenen Vorteil auch unter Schaden durchsetzen, einfach nur Schaden zufügen, illegaler Informationshandel...

- ▶ Böswillige Angreifer werden häufig als Hacker bezeichnet. Ursprünglich bezeichnete dieser Begriff aber Menschen, die kreativ mit technischen Systemen umgehen, diese verstehen oder auf neuartige Weise (um)nutzen wollen. Dabei geht es ihnen nicht darum, anderen zum eigenen Vorteil Schaden zuzufügen, sondern im Gegenteil neue technische Möglichkeiten nutzbar zu machen.

Gute Angreifer greifen Systeme an, um sie besser zu machen, nicht um zu schaden.

- ▶ Man spricht von „ethischen Angreifern“ bzw. „ethischen Hackern“.

Typen von Angreifern

Analysen haben gezeigt:

- ▶ Vor der Jahrtausendwende erfolgten ca. 30% der Angriffe auf Firmen von außen und ca 70% von innen, z.B. von Personen, die auf nicht-technischer Ebene Informationen weitergaben oder technisches Insiderwissen hatten.
- ▶ Seit der Jahrtausendwende hat sich das Verhältnis nahezu umgekehrt.

Weitere Angreifer

- ▶ Als Cracker werden solche Menschen bezeichnet, die zwar ähnliches tun wie („ethical“) Hacker, allerdings mit kriminellen Absichten und zum eigenen Vorteil.
  Begriff „gecrackte Software“, bezeichnet z.B. solche, deren Kopier- oder Installationsschutz ausgehebelt wurde.
- ▶ Script Kiddies sind Angreifer, die mit wenigen eigenen Programmierkenntnissen vorgefertigte Angriffscode (Scripts) nutzen, um Schaden anzurichten.
- ▶ Hacktivisten sind Personen, die Computer und Computernetzwerke als politisches Protestmittel verwenden.

Grundprinzipien des ethischen Hackens

Ethische Hacker greifen Systeme an, um sie besser zu machen, nicht um Schaden anzurichten. Folgende Prinzipien werden eingehalten:

- ▶ Die Privatsphäre respektieren: z.B. sensible Daten im Zuge eines Angriffs mit allergrößtem Respekt behandeln.
- ▶ Keine Systeme zum Absturz bringen: Es ist sehr wichtig, dass man soweit möglich die Systeme weder schädigt, noch deren Betrieb stört.
- ▶ Der Angriff hört an der Stelle auf, an der eine Schädigung möglich wäre, und weist den Betreiber lediglich auf das Problem hin.
- ▶ Sorgfältige Planung, um Pannen zu vermeiden.

Hackerethik (vgl. Chaos Computer Clubs)

- ▶ Diese Ethik geht über einen Verhaltenskodex für Systemangriffe hinaus.
- ▶ Diese Ethik beschreibt allgemeine Werte im Umgang mit Technologien.

Angriffe auf Rechner: Schadsoftware

- ▶ Malware = „Malicious Software“ umschreibt eine Software, die die Funktionalität eines Computers/IT-Systems beeinträchtigt bzw. verändert.
 - ▶ „beeinträchtigt“/„verändert“ aus Sicht eines rechtmäßigen Anwenders
- ▶ Malware lässt sich i.A. nicht genau definieren.

„Kurioses“ Beispiel:

Angriffe auf Rechner: Schadsoftware

- ▶ Malware = „Malicious Software“ umschreibt eine Software, die die Funktionalität eines Computers/IT-Systems beeinträchtigt bzw. verändert.
 - ▶ „beeinträchtigt“/„verändert“ aus Sicht eines rechtmäßigen Anwenders
- ▶ Malware lässt sich i.A. nicht genau definieren.

„Kurioses“ Beispiel:

- ▶ CD Kopierschutz (digital rights management) von Sony BMG.
http://www.theregister.co.uk/2005/11/10/sony_drm_trojan/
Die Firma bestückte ihre Musik-CDs mit einem sog. Rootkit, mit dessen Hilfe das Kopieren der Musik verhindert werden sollte.

Angriffe auf Rechner: Schadsoftware

- ▶ Malware = „Malicious Software“ umschreibt eine Software, die die Funktionalität eines Computers/IT-Systems beeinträchtigt bzw. verändert.
 - ▶ „beeinträchtigt“/„verändert“ aus Sicht eines rechtmäßigen Anwenders
- ▶ Malware lässt sich i.A. nicht genau definieren.

„Kurioses“ Beispiel:

- ▶ CD Kopierschutz (digital rights management) von Sony BMG.
http://www.theregister.co.uk/2005/11/10/sony_drm_trojan/
Die Firma bestückte ihre Musik-CDs mit einem sog. Rootkit, mit dessen Hilfe das Kopieren der Musik verhindert werden sollte.
- ▶ Aufgabe des Sony-Rootkits war nach bekannten Kopierprogrammen zu suchen und ggf. den Nutzer dazu aufzufordern, diese zu beenden; ansonsten Abspielen der CD durch Software gestört.

Wie versteckte sich das Rootkit?

- ▶ Unter Windows wurde die Voreinstellung, Software auf einer CD automatisch auszuführen, genutzt. Die Musik- CDs besaßen ein automatisch startendes Installationsprogramm, um den Nutzer darauf hinzuweisen, dass die Musik nur mit dem mitgelieferten Player abgespielt werden könne und um diesen Player zu installieren.

Das Rootkit wurde ungefragt mitinstalliert.

Wie versteckte sich das Rootkit?

- ▶ Unter Windows wurde die Voreinstellung, Software auf einer CD automatisch auszuführen, genutzt. Die Musik- CDs besaßen ein automatisch startendes Installationsprogramm, um den Nutzer darauf hinzuweisen, dass die Musik nur mit dem mitgelieferten Player abgespielt werden könne und um diesen Player zu installieren.
Das Rootkit wurde ungefragt mitinstalliert.
- ▶ Die Namen der Dateien, die zu dem Rootkit gehörten, begannen mit \$sys\$.
- ▶ Es wurden Systemfunktionen so manipuliert, dass alle Dateien oder Prozesse, deren Namen mit \$sys\$ beginnen, nicht angezeigt wurden.



Ist das ein Problem?

Ja, da ...

- ☠ durch das Verstecken sämtlicher Dateien, deren Namen mit \$sys\$ beginnen, konnten und haben sich weitere Schadprogramme mit entsprechender Namensgebung unerkannt eingenistet.
- ☠ das Rootkit unsauber programmiert war, sodass es die normalen Systemfunktionen der betroffenen Rechner beeinträchtigt hat. Eine Folge war Datenverlust.
- ☠ das Rootkit nutzte selbst nicht-lizenzierte Software, sodass es Urheberrechte verletzte - zur Bekämpfung von Raubkopien?!

Rechtlich problematisch:

- ☠ Das Rootkit wurde installiert, auch wenn der Nutzer der Lizenzvereinbarung nicht zustimmte und den Installationsvorgang abbrach.

Rechtlich problematisch:

- ☠ Das Rootkit wurde installiert, auch wenn der Nutzer der Lizenzvereinbarung nicht zustimmte und den Installationsvorgang abbrach.
- ☠ Es wurden Hörgewohnheiten an Sony übermittelt. Damit enthielt die Software sogar Spyware.

Rechtlich problematisch:

- ☠ Das Rootkit wurde installiert, auch wenn der Nutzer der Lizenzvereinbarung nicht zustimmte und den Installationsvorgang abbrach.
- ☠ Es wurden Hörgewohnheiten an Sony übermittelt. Damit enthielt die Software sogar Spyware.
- ☠ Die Schadensbegrenzung seitens Sony selbst war problematisch: Updates gab es nur gegen persönliche Daten, das Rootkit selbst wurde nicht entfernt.
- ☠ Ironischerweise hatten dadurch Personen, die eine legal gekaufte Musik-CD abspielen wollten, mehr Ärger als jene, die sich die Musik als kopierte Files besorgten.

Weiteres Beispiel: Bundestrojaner

- ▶ Software zur Quellentelekomunikationsüberwachung des BKA [http://www.heise.de/newsticker/meldung/
BKA-Chef-Bundestrojaner-im-
Herbst-einsatzbereit-2621280.html](http://www.heise.de/newsticker/meldung/BKA-Chef-Bundestrojaner-im-Herbst-einsatzbereit-2621280.html)
- ▶ Sie soll in Fällen schwerer Kriminalität sowie nach richterlichem Beschluss im Rahmen der Strafverfolgung, zur Gefahrenabwehr oder auch zur nachrichtendienstlichen Informationsbeschaffung eingesetzt werden.
- ▶ Es wird die laufende Kommunikation der Zielpersonen direkt am Endgerät mittels Spionagesoftware überwacht.

Im folgenden wird Schadsoftware klassifiziert.

Gibt es auch etwas mit bösartiger Krypto? Ja, z.B. Locky

Gibt es auch etwas mit bösartiger Krypto? Ja, z.B. Locky

- ▶ Locky zählt zur Ransomware.
- ▶ Erpressungstrojaner gibt es seit 1989.
- ▶ Richtig bekannt wurden sie Anfang des Jahres 2016, als Locky und TeslaCrypt eine große Infektionswelle weltweit hervorriefen. Unter anderem hat Locky 60 Arbeitsplätze in einem Fraunhofer-Institut in Bayreuth sowie ein Krankenhaus in Los Angeles lahmgelegt.
- ▶ Die Verbreitung von Locky erfolgte - wie häufig bei Trojanern - mittels Anhängen von Spam-Mails.
- ▶ Es gab sowohl Script-Dateien, die unter bestimmten Umständen den Trojaner aus dem Netz geladen haben, als auch Anhänge, die sich als Scans aus dem eigenen Netzwerk tarnten.

- ▶ Kurze Zeit später tauchte eine Variante dieses Schädlings auf, der nicht Rechner von Endnutzern angriff, sondern Webserver.
- ▶ Kam der Schädling zur Ausführung, wurden die Daten hunderter Webseiten verschlüsselt (je nach dem, wieviele Websites auf dem Server gehostet werden).
- ▶ Da Backups bei Webservern häufig durch das Spiegeln der Daten in Echtzeit auf andere Server realisiert sind, sind bei Schädlingsbefall auch leicht die entsprechenden Ausfallsysteme betroffen.

Wesentliche Charakteristika von Malware

Verbreitungsweg bzw. Mechanismus

- ▶ Selbstreplizierend im Netz.
- ▶ E-Mail.
- ▶ Physikalische Weitergabe (z.B. via USB-Stick).

Bösartiges Verhalten (sog. Schadroutine) z.B.

- ▶ Löschen der Festplatte.
- ▶ Verschlüsseln der Festplatte
(zwecks Erpressung  Ransomware).
- ▶ Mailer für SPAM.
- ▶ Abgreifen von Zugangsdaten.

„Berühmte“ Vertreter: Viren

Angelehnt an biologische Viren (ältestes Malware-Konzept)

- ▶ Benötigen einen Wirt (Programm, Datei), den sie infizieren können.
- ▶ Verbreiten sich unkontrolliert auf einem System.

Typen (Auswahl)

- ▶ Bootviren (befallen Computer beim Start eines Systems).
- ▶ Dateiviren (befallen ausführbare Dateien).
- ▶ Makroviren (befallen Dokumente mit Makrofunktionalität).
- ▶ Mischformen existieren.

Viruskennung	<pre>PROCEDURE Virus; BEGIN 4711</pre>	Hilft erkennen, ob Programm schon infiziert ist
Infektionsteil	<pre>Suche eine nicht infizierte Programmdatei; IF (gesundes Programm gefunden) THEN kopiere Virus in das Programm</pre>	
Schadensteil	<pre>IF (Datum = Freitag der 13.) THEN formatiere Festplatte;</pre>	
Sprung	<pre>Springe an den Anfang des Wirtsprogramms; END</pre>	Falls Wirtsprogramm noch gestartet werden soll

Was an diesem Aufbau ist hilfreich für Antiviren-Programme?

Beispiele - Michelangelo (Virus)

- ▶ 1991/1992.
- ▶ Boot-Sektor-Virus für MS-DOS Systeme (Vorgänger von Windows).
- ▶ Aktiviert sich nur am 6. März eines Jahres (Geburtstag des Malers Michelangelo).
- ▶ Schadroutine: Löscht die ersten 100 Sektoren einer Festplatte. Danach ist der Zugriff auf die Daten nicht mehr (ohne Weiteres) möglich.

☞ IT-Forensik

Beispiele - Melissa

1999

- ▶ Makrovirus für MS Word.
- ▶ Verbreitet sich als Anhang einer E-Mail. Nutzer muss Anhang öffnen.
- ▶ Schadroutine:
 - ▶ Verändert Sicherheitseinstellungen.
 - ▶ Infiziert Normal.dot (Vorlage für MS Word).
 - ▶ Fügt unter gewissen Voraussetzungen einen kurzen Text in ein Word Dokument.

„Berühmte“ Vertreter: Würmer

Autonomes Programm, d.h. startet selbstständig ohne Wirt.

- ▶ Verbreitet sich über Computernetze oder Wechselmedien.

Verbreitungswege über Hilfsprogramme (nicht Wirtprogramme!).

Abhängig davon ergeben sich folgende Typen:

- ▶ E-Mail-Würmer

werden über E-Mail-Anhänge verbreitet. Bekannt geworden ist z. B. der ILOVEYOU-Wurm, der sich im Jahre 2000 als .vbs-Datei (Visual Basic Script) in E-Mails mit dem Betreff ILOVEYOU versendete. Der Wurm zerstörte auf dem Zielrechner alle Dateien mit bestimmten Dateiendungen. Der Schaden, den er verursachte, bestand nicht nur in den zerstörten Dateien und dem Aufwand, den Rechner zu desinfizieren, sondern auch in durch die rasante Verbreitung überlasteten Mailservern.

► IM-Würmer

verbreiten sich über Nachrichten von sog. Instant Messengern (IM). Dabei sorgt der Wurm dafür, dass Nachrichten mit einem bestimmten Link an die Kontakte eines Messengers selbstständig verschickt werden. Hält der Empfänger den Link für vertrauenswürdig und klickt auf ihn, kommt er auf eine Website, von der der Wurm automatisch heruntergeladen und aktiviert wird (Drive-by-Exploits). Daraufhin kann er sich erneut über den Messenger weiterverbreiten.

► IM-Würmer

verbreiten sich über Nachrichten von sog. Instant Messengern (IM). Dabei sorgt der Wurm dafür, dass Nachrichten mit einem bestimmten Link an die Kontakte eines Messengers selbstständig verschickt werden. Hält der Empfänger den Link für vertrauenswürdig und klickt auf ihn, kommt er auf eine Website, von der der Wurm automatisch heruntergeladen und aktiviert wird (Drive-by-Exploits). Daraufhin kann er sich erneut über den Messenger weiterverbreiten.

► P2P-Würmer

verbreiten sich über Filesharing-Plattformen. Dafür kopiert er sich, erst einmal auf einem System mit einer P2P-Software vorhanden, in das Verzeichnis, in dem die untereinander geteilten Dateien liegen. Liegt er unentdeckt zwischen anderen interessanten Dateien, wird er von anderen Nutzern unwissentlich mit heruntergeladen. Auf dem Zielsystem liegt der Wurm wieder in einem Ordner, von dem er von anderen Beteiligten heruntergeladen wird etc.

► Netzwerk-Würmer

verbreiten sich direkt über das LAN, indem sie Sicherheitslücken in dem vorhandenen Betriebssystem oder der zugrundeliegenden Netzwerktechnik ausnutzen und angeschlossene Zielsysteme finden.



Je nach Verbreitungsweg u.U. (ungewollte) Mithilfe durch den Nutzer von Nöten.

- ▶ Netzwerk-Würmer
 - verbreiten sich direkt über das LAN, indem sie Sicherheitslücken in dem vorhandenen Betriebssystem oder der zugrundeliegenden Netzwerktechnik ausnutzen und angeschlossene Zielsysteme finden.

☠ Je nach Verbreitungsweg u.U. (ungewollte) Mithilfe durch den Nutzer von Nöten.

☞ Ursprünglich wurden Würmer nicht als Schadprogramme entwickelt, sondern als eine Art „verteiltes Betriebssystem“, um z. B. Berechnungen verteilt ausführen zu können und hohe Fehlertoleranz zu haben.

Beispiel Smart-Home-Bereich:

- ▶ Würmer haben auch das IoT erreicht. Glühlampen, deren Sicherheitslücken es einem Wurm ermöglichten, per Funk von einer Glühlampe zur nächsten zu hüpfen (Forschungsarbeit Ronen et al. 2016).
- ▶ Smarte Glühbirnen besitzen eine Firmware, über die die Birne mit anderen Geräten per Funk insbesondere zu ihrer Steuerung kommunizieren kann.
- ▶ Für die Kommunikation wurde bei den bei diesem Angriff genutzten Geräten ZigBee benutzt. ZigBee ist eines der für SMART-Home-Geräte entwickelten Funkprotokolle. ZigBee soll dafür sorgen, dass in einem Netz von Geräten, z. B. für eine Wohnung, einzelne Geräte nicht so einfach von außerhalb dieses Netzes gesteuert werden können.

- ▶ Der von Forschern vorgestellte Wurm nutzte zwei Sicherheitslücken:
 - ▶ Das ZigBee-Protokoll wurde für die betroffenen Glühbirnen falsch implementiert, sodass es dem Wurm möglich war, die befallene Glühbirne aus dem Smart-Home-Netz herauszulösen und selbst zu steuern.
 - ▶ Die Firmware musste für den Wurm verändert werden und konnte dann mit einem Schlüssel vom Hersteller Philips signiert werden.

- ▶ Der von Forschern vorgestellte Wurm nutzte zwei Sicherheitslücken:
 - ▶ Das ZigBee-Protokoll wurde für die betroffenen Glühbirnen falsch implementiert, sodass es dem Wurm möglich war, die befallene Glühbirne aus dem Smart-Home-Netz herauszulösen und selbst zu steuern.
 - ▶ Die Firmware musste für den Wurm verändert werden und konnte dann mit einem Schlüssel vom Hersteller Philips signiert werden.
- ▶ Die kompromittierte Firmware konnte durch die gültige Signatur auf andere Glühlampen geladen werden.
- ▶ Die Forscher konnten Glühbirnen auch von Drohnen im Vorbeifliegen infizieren lassen.
- ▶ Der Wurm konnte nicht nur die Glühbirne selbst schädigen (z.B. durch flackern), sondern auch WLAN-Verkehr in der Umgebung stören und Daten verdeckt übertragen.

☞ Die konkrete Sicherheitslücke ist inzwischen geschlossen.

Beispiele - Morris worm

Erster Computerwurm des Internets.

- ▶ 1988 von Robert T. Morris entwickelt und freigesetzt.
- ▶ Verbreitete sich Mithilfe der Unix Netzdiene
sendmail, finger, rexec.
- ▶ Keine Schadroutine, jedoch waren befallene Rechner stark ausgelastet (DoS).

THE RSH/REXEC ATTACK:

The third way it tried to get into systems was via the .rhosts and /etc/hosts.equiv files to determine 'trusted' hosts where it might be able to migrate to. To use the .rhosts feature, it needed to actually get into people's accounts - since the worm was not running as root (it was running as daemon) it had to figure out people's passwords. To do this, it went through the /etc/passwd file, trying to guess passwords. It tried combinations of: the username, the last, first, last+first, nick names (from the GECOS field), and a list of special "popular" passwords:

aaa	cornelius	guntis	noxious	simon
academia	couscous	hacker	nutrition	simple
aerobics	creation	hamlet	nyquist	singer
airplane	creosote	handily	oceanography	single
albany	cretin	happening	ocelot	smile

Abbildung:

Quelle:<http://www.foo.be/docs-free/morris-worm/worm.paper>

„Berühmte“ Vertreter: Trojaner

Software, die auf den ersten Blick eine nützliche, gewollte Funktionalität bietet.

- ▶ Im Hintergrund aber zusätzlich eine dem Nutzer verborgene/ungewollte Funktionalität.
- ▶ Name in Anlehnung an die griechische Mythologie (vgl. Trojanischer Krieg).
- ▶ Benötigen Mitarbeit des Anwenders (z.B. um Trojaner initial zu installieren/starten).

„Berühmte“ Vertreter: Trojaner

Krypto-Trojaner oder Ransomware

- ▶ Verschlüsselt die gesamten Dateien auf dem infizierten Rechner und zeigt dann einen Hinweis an, auf welches Konto der Nutzer wie viel Lösegeld (per Bitcoin) zu überweisen hat, um den Rechner wieder zu entschlüsseln.
- ▶ Oft lässt sich Verschlüsselung nicht knacken.
- ▶ Datenwiederherstellung dann nur mit Backup möglich.
- ▶ Liegt das Backup allerdings nicht separat vom Rechner, sondern wird z. B. laufend dynamisch über das Netz erstellt, so wird es ggf. vom Trojaner gleich mitverschlüsselt.

Beispiel: Jigsaw, Locky, Sony(?)

„Berühmte“ Vertreter: Rootkits /Backdoors

root (Systemadministrator, super user) unter Unix. Zwei typische Eigenschaften:

- ▶ Sichert einem Angreifer die Kontrolle über ein System (über eine Hintertür/Backdoor) nach einem erfolgreichen Angriff.
- ▶ Manipuliert diverse Teile eines Betriebssystems, um nicht entdeckt zu werden (Logdateien, Prozesstabellen etc.) und um das System überwachen zu können (Tastaturlogger, Netzwerkverkehr etc.).
- ▶ Damit weisen Rootkits gewisse Ähnlichkeiten zu Trojanern auf, da auch sie sich tarnen und ggf. weitere Schädlinge einschleusen. Allerdings sind sie einerseits spezieller (Erlangung von Zugriffsrechten als Ziel) und verbreiten sich auch anders. Aber auch hier gilt, dass die Grenzen fließend sind.

- ▶ Ziel ist also eine Art Generalschlüssel bzw. Fernzugriff für ein System zu haben.

- ▶ Ziel ist also eine Art Generalschlüssel bzw. Fernzugriff für ein System zu haben.
 - ☛ Diese Funktionalität ist übrigens auch für Remote-Desktop-Programme notwendig, was zeigt, dass es nicht der Code selbst ist, der böswillig ist, sondern dass es eine Frage der Einsatzziele ist (vgl. Sony-Rootkit).
- ▶ Rootkits können danach unterschieden werden, wie sie sich im System verstecken und entsprechend auch entdeckt werden.
- ▶ Möglichkeiten sind hier, Log-Dateien zu löschen, Systemprogramme zu modifizieren bzw. auszutauschen oder gar dort verwendete Systemaufrufe zu manipulieren.

„Berühmte“ Vertreter: Botnetze



Abbildung: Quelle: BSI

Botnetze sind Netzwerke, die aus kompromittierten Endnutzer- PCs bestehen, die für eine gewisse Funktionalität ferngesteuert werden.

- ▶ Verschleiern der Herkunft.
- ▶ Versand von E-Mails.
- ▶ Ausführen von DDoS Angriffen.
- ▶ Klickbetrug.
- ▶ Verteilter Speicher (illegaler Inhalte).
- ▶ Verteiltes Rechnen.

Betreiber eines Botnetzes kontrollieren die Bots und greifen Informationen ab über sog. Command-and-Control Kanäle. Bspw. IRC (Internet Relay Chat), P2P, HTTP.

„Berühmte“ Vertreter: Adware/Spyware und Co.

Software, die gerne im Bundle mit anderen Programmen installiert wird und das Verhalten des Nutzers auswertet und/oder ungefragt Werbung einblendet.

Adware

- ▶ ist eigentlich kostenfreie Software (nicht unbedingt Schadsoftware), die Werbung einblendet, um sich zu finanzieren. Allerdings wird auch Malware, die zu Werbezwecken dient, als Adware bezeichnet ( Malvertising). Hierbei werden Werbeflächen auf Webseiten dazu ausgenutzt, Malware zu verbreiten - dazu braucht ein Besucher der Website nicht einmal auf das Werbebanner selbst klicken.
- ▶ Auswertung geschieht zu Werbezwecken (Verlust der Privatsphäre).
- ▶ Strittig: Hat ein Nutzer seine Einwilligung bei Installation gegeben oder nicht?

Spyware

- ▶ ist Software, die Daten auf dem infizierten Rechner sammelt und zu einem bestimmten System übermittelt.
- ▶ Besonders interessant sind im Klartext gespeicherte Passwörter bzw. Zugangsdaten, sensible Dokumente (z. B. Geschäftsgeheimnisse) oder E-Mails. Auch Cookies werden gern ausgelesen, da sie ggf. Zugangsdaten für Webdienste enthalten.
- ▶ Spyware findet häufig über Trojaner ihren Weg auf den Zielrechner bzw. kann Teil eines Trojaners sein.

Backdoors

- ▶ sind Hintertüren innerhalb von Software. Sie sind damit keine eigenständigen Programme, sondern heimlich integrierte Programmteile, die z. B. Sicherheitsvorkehrungen umgehen, und insofern eigentlich auch nicht Schadprogramme im eigentlichen Sinne.
- ▶ Rootkits enthalten Hintertüren.
- ▶ Es ist nicht immer eindeutig, ob es sich bei entsprechenden Sicherheitslücken tatsächlich um eine Hintertür oder lediglich um einen (unabsichtlichen) Bug handelt. So gab es z. B. in dem Content Management System Joomla eine Sicherheitslücke, die zunächst aufgrund eines Bugs nicht ausgenutzt werden konnte; der Bug wurde gefixt, aber die Lücke blieb. Ob dies Absicht war oder nicht, konnte nicht geklärt werden.

Unstrittig: Keylogger

- ▶ sind kleine Programme, die die Tastaturanschläge der Nutzer protokollieren. Das kann sowohl permanent und umfassend geschehen (alle Anschläge werden aufgezeichnet) oder selektiv (z. B. bei Passwort-Feldern).
- ▶ Sie schalten sich zwischen Tastatur und Betriebssystem, um die Eingaben zu speichern oder über das Internet zu versenden, und leiten die Tastatureingaben dann an das Betriebssystem weiter.
- ▶ Da Keylogger meist dazu verwendet werden, vertrauliche Informationen an den Angreifer weiterzuleiten, kann man Keylogger auch zu Spyware zählen.

Scareware

- ▶ (engl. to scare, „erschrecken“) bezeichnet Programme, die mit Angst einflößenden Informationen Nutzer dazu bewegen wollen, unbedacht Webseiten zu besuchen, weitere Software zu installieren oder Informationen weiterzugeben.
- ▶ Beispiel sind Pop-ups auf Webseiten mit der Mitteilung, es seien Schadprogramme auf dem Rechner gefunden worden, die mit einer (teuren) angepriesene Antivirensoftware entfernt werden könnten - wobei das zu kaufende Programm allerdings ggf. diese Funktionalität gar nicht besitzt und schlechtestenfalls ein Trojaner ist, der Schadsoftware auf dem Rechner installiert.

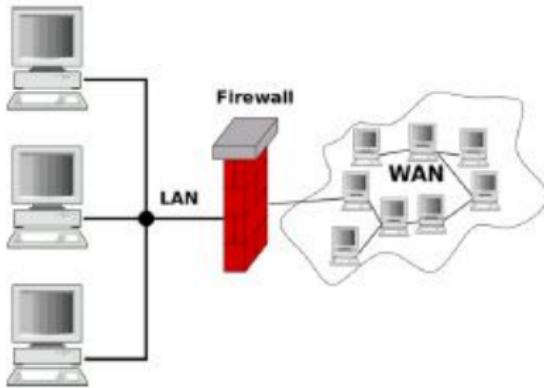
Weiteres Beispiel - Stuxnet (vereint einige Typen)

- ▶ 2010
- ▶ Sehr komplexe Malware.
- ▶ Nutzte unbekannte Sicherheitslücken, sog. Zero-Day-Exploits, aus.
- ▶ Installierte ein Rootkit.
- ▶ Greift nur ausgewählte Steuerungssysteme von Industrieanlagen an (vermutlich Teile des iranischen Atomprogramms).
- ▶ Nachfolger Duqu (2011).
 - ▶ Keine Schadroutine. Sammelt nur Informationen.

Schutzmaßnahmen - VirensScanner

- ▶ Softwaretool zur Erkennung, Deaktivierung und u.U. Löschung von Malware.
- ▶ Betriebsmodi:
 - ▶ Überwachung in Echtzeit.
 - ▶ VirensScanner überwacht jeden Schreib- und Lesezugriff eines Betriebssystem.
 - ▶ Manuelle Überwachung.
Anwender muss Überprüfung der Dateien anstoßen.
 - ▶ Erkennen nur bekannte Schadsoftware (Erfolgsrate bei ca. 45%).
 - ▶ Verlangsamen u.U. deutlich das System.

Schutzmaßnahmen - Firewall



- ▶ Netzwerkverkehr (Traffic) zwischen dem Internet und einem Firmennetz wird untersucht.
- ▶ Nur erlaubter Traffic darf eine sog. Firewall passieren.
- ▶ Typen:
 - ▶ Paketfilter.
 - ▶ Zustandsgesteuerte Paketfilter.
 - ▶ Application Level Firewalls.

Schutzmaßnahmen - Honeypot

- ▶ Honeypots sind IT-Systeme, die eigens dafür aufgesetzt sind, um einen Angreifer vom eigentlichen Ziel abzulenken.
- ▶ Honeypots erlauben es, einen Angreifer zu beobachten und Angriffsmuster zu studieren.
- ▶ Beispiele/Projekte:
 - ▶ <http://glastopf.org/>
 - ▶ <https://www.honeynet.org/>
 - ▶ <http://honeytrap.carnivore.it/>

Schutzmaßnahmen - Pentesting

- ▶ Penetrationstests sind Sicherheitstests, die helfen, Schwachstellen in einem System zu finden.
- ▶ Pentest muss vorab vertraglich vereinbart werden, sonst u.U. Strafbestand (vgl. Ausspähen von Daten, §202a StGB).
- ▶ Hierzu werden Mittel und Methoden eines Angreifers angewendet.
- ▶ Wichtige Werkzeuge (Auswahl):
 - ▶ Metasploit
 - ▶ nmap
 - ▶ OpenVAS (www.openvas.org)

Schutzmaßnahmen - Intrusion Detection/Prevention

- ▶ Snort: Freies Network Intrusion Detection System (NIDS) und ein Network Intrusion Prevention System (NIPS), um Angriffe unmittelbar ereignisgesteuert automatisch zu blockieren.
<https://www.snort.org/>
- ▶ Funktionsweise beruht auf Stringmatching, eines von mehreren bekannten Mustern (Pattern) oder ein Text (PCAP Strom), Aho-Corasick-Algorithmus
- ▶ Snort wurde von Martin Roesch programmiert und von der Firma Sourcefire weiterentwickelt (in 2013 von Cisco übernommen).
- ▶ In 2009 wurde Snort in die „Open Source Hall of Fame“aufgenommen.

Sensibilisierung der Mitarbeiter

- ▶ Einspielen der neuesten Updates.
- ▶ Keine verdächtigen Mails oder deren Anhänge öffnen.
- ▶ Sorgsamer Umgang mit Links (vor allem in E-Mails).
- ▶ Genau überlegen, ob eine Software installiert werden muss (kann nicht immer ganz verboten werden).
- ▶ Nutzen von Antivirensoftware.
- ▶ :

☞ Diese Tipps sind auch privat nützlich.

Steuerungssysteme: SCADA (Supervisory Control and Data Acquisition)

Setzt z.B. Enercon ein.

- ▶ SCADA-Steuerungssysteme nutzen heutzutage gewöhnliche Netzwerkprotokolle wie TCP/IP; damit sind sie zum Beispiel per IP-Adressen ansprechbar oder mit Büronetzen gekoppelt.
- ▶ Außerdem laufen sie immer häufiger auf handelsüblichen Betriebssystemen wie Windows oder Linux, deren Konfiguration nicht speziell auf die Bedarfe von Industrieanlagen zugeschnitten sind.
- ▶ Damit sind sie dann im Prinzip den gleichen Attacken ausgesetzt sind wie andere (vernetzte) IT-Systeme .

- ▶ Die Steuerungssysteme von Industrieanlagen weisen besondere Charakteristika auf, die eine Absicherung mit herkömmlichen Methoden, wie sie bereits erwähnt wurden, erschweren.
 - ▶ SCADA-Systeme arbeiten im Echtzeitbetrieb, d. h. sie müssen nahezu verzögerungsfrei in die Steuerungssysteme von Industrieanlagen eingreifen. VirensScanner reduzieren die Performance und werden deshalb nicht eingesetzt.
-  Schadsoftware kann sich daher einfacher ausbreiten.
- ▶ Das Patchmanagement ist lückenhaft: SCADA-Systeme arbeiten typischerweise ohne Unterbrechungen (24/7 pro Woche) und häufig ohne Wartungsfenster für die Software.
 - ▶ Softwareaktualisierungen werden seltener oder gar nicht durchgeführt. Entsprechend alt und mit öffentlich bekannten Sicherheitslücken behaftet sind die eingesetzten Betriebssysteme und Applikationen.

- ▶ Penetrationstests werden sporadisch durchgeführt, da diese bei SCADA Systemen das Risiko massiver Fehlfunktionen mit schwer abzuschätzenden Konsequenzen.
 - ▶ Authentifizierung ist oft schwach, da Netzwerzugriffe auf Systemkomponenten möglich sind. Fest einprogrammierte Passwörter sind bei Maschinezu- Maschine-Kommunikation ein nicht unübliches Mittel.
 - 💀 Wurde auch von Stuxnet zur Infektion ausgenutzt.
 - ▶ Daten werden unverschlüsselt übertragen.
 - 💀 SCADA-Systeme nutzen Verschlüsselung oft nicht einmal bei der Übertragung von Passwörtern.
- ☞ Bei Enercon wird großer Wert auf Sicherheit gelegt.

Angriffe im und auf das Netz

Zur Erinnerung: Computernetzwerk

Angriffe im und auf das Netz

Zur Erinnerung: Computernetzwerk

- ▶ Zwei oder mehrere Computer, die durch ein Übertragungsmedium miteinander verbunden (vernetzt) sind, bilden ein Computernetzwerk.
- ▶ Ein Computernetz besteht daher mindestens aus zwei Knoten (Rechner) und einer (physikalischen + logischen) Verbindung.

Es gibt viele verschiedene Netzwerke

- ▶ Das Internet
- ▶ Intranet
- ▶ Computernetzwerke
- ▶ Telefonnetzwerke
- ▶ Straßenverkehrsnetze
- ▶ Mobilfunknetzwerke
- ▶ Optische Netzwerke
- ▶ Post- und Paketnetzwerke
- ▶ Smart-Home-Netzwerke

☞ Diese wachsen derzeit zu einem großen zusammen, Stichwort IoT.

Hier: Angriffe auf Datennetzwerk

- ▶ Starken Fokus auf Internet-Protokoll (IP) Netzwerken.
- ▶ Das Internet als Anwendung eines Computernetzwerkes.
- ▶ Lokale Netzwerke.

Typische miteinander vernetzte IoT-Geräte

- ▶ Steuerung der Klimaanlage.
- ▶ Drucker.
- ▶ Energieversorgung.
- ▶ Audio/Video-Geräte.
- ▶ Haussteuerung.
- ▶ Transportmittel.

Wie werden diese vernetzt?

- ▶ Wireless-LAN.
- ▶ Ethernetkabel.
- ▶ Bluetooth.
- ▶ Mobilfunk (GPRS, UMTS, HSDPA, LTE).
- ▶ Optische-Links (Glasfaser).

Sinn und Zweck von Computernetzwerken

- ▶ Zugriffe auf Daten, Programme, Ressourcen anderer Rechner ist möglich, z.B.
 - ▶ Zugriff auf Hochleistungsrechner.
 - ▶ Abruf von Filmen aus einem Archiv.
- ▶ Verteilung von Rechenleistung und Datenhaltung auf unterschiedliche Rechner.
 - ▶ Erhöhte Flexibilität und Ausfallsicherheit.
- ▶ Rechnergestützte Aufgaben können arbeitsteilig ausgeführt werden.
 - ▶ Verteilung von Rechenaufgaben auf unterschiedliche Computer.

Typische Bausteine eines Computernetzwerkes

- ▶ Endgeräte / „Computer“.
 - ▶ Laptop, PC, Smartphone, Großrechner, Kaffeemaschine, Webserver.
- ▶ Hardware für die physikalische Übertragung.
 - ▶ Verkabelung der Netzwerkgeräte.
 - ▶ Netzwerkgeräte (Router, Switches, Netzwerkkarten, ...)
- ▶ Netzwerk-Software.
 - ▶ Implementierung von Protokollen.
- ▶ Netzwerk-Applikationen.
 - ▶ Web, Email, etc.

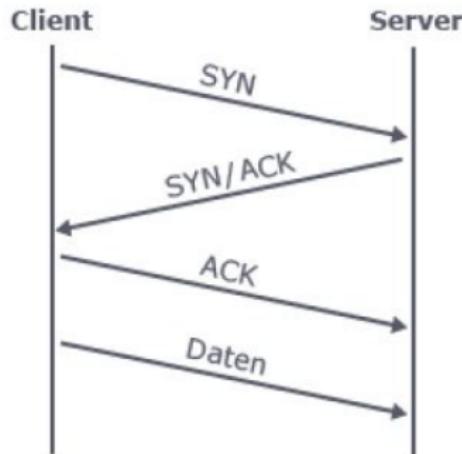
Mögliche Vorgehensweise

- ▶ Angreifer will herausfinden, welche Hosts (Rechner oder andere vernetzte Systeme) in einem Netzwerk aktiv sind.
- ▶ Dazu gibt es z.B. das Diagnosewerkzeug Ping (in Analogie zum Geräusch eines Sonars).
- ▶ Ping sendet sog. Echo-Requests des Internet Control Message Protocols (ICMP) an einen Host (dessen Adresse) und wartet, ob der Host antwortet oder nicht.
☞ Mit Ping Sweep kann man Adressbereiche ansprechen.

Hat man aktive Hosts gefunden, ist die nächste Frage, ob man in sie hineinkommt, man sucht also nach Türen - in Netzwerken Ports genannt.

- ▶ Damit Datenpakete, z. B. einer ausgelieferten Website, nicht nur den Client sondern auch die richtige Anwendung finden werden sog. Ports genutzt.
- ▶ Für bestimmte Daten werden bestimmte Ports benutzt, z.B. Port 80 bei Webseiten, Port 25 bei ausgehenden Mails.
- ▶ Ports dienen der Unterscheidung mehrerer Verbindungen zwischen kommunizierenden Endpunkten und der Identifizierung verschiedener Netzwerkdienste und -protokolle. Ein Rechner hat 2^{16} Ports zur Verfügung.
- ▶ Deshalb führt ein Angreifer sog. Port Scans durch: Eine systematische Prüfung offener Ports.
Solche Tests erfolgen über die Netzwerkprotokolle TCP und UDP.
Es werden Datenpakete an diese Ports gesendet und die Antworten oder Fehlermeldungen ausgewertet.
- ▶ Entsprechende Werkzeuge zur Analyse verbinden meist das Scannen von Hosts und Ports.

Portscans im Detail



Ist der Port allerdings geschlossen oder gibt es ein technisches Problem, sendet der Server ein RST-Paket (von eng. reset: zurücksetzen) zurück an den Client.

Es gibt verschiedene Typen von Port Scans, die sich auf diese speziellen Datenpakete beziehen. Mögliche Fälle sind:

- ▶ Zielrechner sendet ein sog. „ACK“-Paket. Der Port ist offen.
- ▶ Zielrechner sendet ein sog. „RST“-Paket. Der Port ist geschlossen.
- ▶ Zielrechner antwortet gar nicht. Der Port ist versteckt und es ist unklar, ob der Port offen oder geschlossen ist.
☞ Wie können erwünschte Rechner auf den Port bzw. damit verbundenen Dienst zugreifen?

Über Portscans lassen sich zum Beispiel auch Verzeichnisse finden, die ein Nutzer auf seinem System freigegeben hat, z. B. für Tauschbörsen.

☠ Angriffsmöglichkeiten.

Wardriving

Wardriving ist das Suchen nach ungeschützten Geräten, z.B. WLANs in der Umgebung.

💀 Im Falle von Wlan muss der Angreifer sich dann doch mal auf den Weg machen, um mehr Netze als jene in unmittelbarer Umgebung zu finden. Er muss aufgrund der relativ hohen Reichweite heutiger WLANs aber z. B. nicht in die Gebäude eindringen, aus denen heraus sie betrieben werden.

Betriebssysteme erkennen

- ▶ Mit Portscans lassen sich Betriebssysteme (das sog. Fingerprinting) erkennen.  Eingrenzung von Schwachstellen.
- ▶ Trick: Die Spezifikation der Protokolle TCP/IP legt nicht für jeden Fehlerfall ein exaktes Verhalten fest.
 Implementierungen zwischen Betriebssystemen sind unterschiedlich. Das ergibt den Fingerabdruck.
- ▶ Beim Fingerprinting gibt es passive und aktive Methoden.
 - ▶ Passive Methoden: Beobachten des Netzwerkverkehrs und Analysieren der Datenpakete.
 - ▶ Aktive Methoden: Bestimmte Datenpakete werden an das Zielsystem geschickt, um das spezifische Antwortverhalten zu analysieren.
Z.B. sieht das TCP Protokoll vor, dass ein Host, der nur ein FIN-Paket erhält, keine Antwort zurücksendet. Windows NT sendet aber ein FIN/ACK-Paket.

 Teil des sog. „Banner Grabbing“.

Netzwerkverkehr abhören

- ▶ Es gibt Werkzeuge, um den Datenverkehr mitzuschneiden und anschließend analysieren zu können (sog. Sniffer).
- ▶ Das bekannteste Werkzeug für das Mitschneiden und Analysieren des Datenverkehrs ist Wireshark.
- ▶ Ein Mitschnitt des Datenverkehrs zeigt zum Beispiel, zwischen welchen IP-Adressen zu welcher Uhrzeit was für Datenpakete ausgetauscht wurden.
- ▶ Werden entsprechend ungesicherte Protokolle verwendet, können natürlich auch übertragene Passwörter mitgehört werden.

Mitschnitt Wireshark

Time	Source	Destination	Protocol	Length	Info
1 2012/296 22:38:53.206855	24.6.173.2...	75.75.75.75	DNS	74	Standard query 0xae0b A w
2 2012/296 22:38:53.220092	75.75.75.75	24.6.173.220	DNS	154	Standard query response 0:
3 2012/296 22:38:53.220826	24.6.173.2...	75.75.75.75	DNS	74	Standard query 0x4553 AAA
4 2012/296 22:38:53.234550	75.75.75.75	24.6.173.220	DNS	102	Standard query response 0:
5 2012/296 22:38:53.235554	24.6.173.2...	74.125.224...	TCP	66	35145 → 80 [SYN] Seq=0 Win
6 2012/296 22:38:53.252926	74.125.224...	24.6.173.220	TCP	66	80 → 35145 [SYN, ACK] Seq=1
7 2012/296 22:38:53.253113	24.6.173.2...	74.125.224...	TCP	54	35145 → 80 [ACK] Seq=1 Ack
8 2012/296 22:38:53.253853	24.6.173.2...	74.125.224...	HTTP	342	GET / HTTP/1.1
9 2012/296 22:38:53.272556	74.125.224...	24.6.173.220	TCP	60	80 → 35145 [ACK] Seq=1 Ack
10 2012/296 22:38:53.327329	74.125.224...	24.6.173.220	TCP	1484	[TCP segment of a reassem]
11 2012/296 22:38:53.329529	74.125.224...	24.6.173.220	TCP	1484	[TCP segment of a reassem]
12 2012/296 22:38:53.329535	74.125.224...	24.6.173.220	TCP	863	[TCP segment of a reassem]
13 2012/296 22:38:53.329537	74.125.224...	24.6.173.220	TCP	1484	[TCP segment of a reassem]
14 2012/296 22:38:53.329540	74.125.224...	24.6.173.220	TCP	1484	[TCP segment of a reassem]
15 2012/296 22:38:53.329542	74.125.224...	24.6.173.220	TCP	1290	[TCP segment of a reassem]
16 2012/296 22:38:53.329544	74.125.224...	24.6.173.220	TCP	1484	[TCP segment of a reassem]
17 2012/296 22:38:53.329547	74.125.224...	24.6.173.220	TCP	358	[TCP segment of a reassem]

Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 0
Ethernet II, Src: HewlettP_a7:bf:a3 (d4:85:64:a7:bf:a3), Dst: Cadant_31:bb:c1 (00:01:5c:31:bb:c1)
Internet Protocol Version 4, Src: 24.6.173.220, Dst: 75.75.75.75
User Datagram Protocol, Src Port: 51724 (51724), Dst Port: 53 (53)
Domain Name System (query)

http (unverschlüsselt) vs. https (verschlüsselt)

The screenshot shows two separate network captures side-by-side in Wireshark.

Left Window (http traffic):

- Frame 282: 771 bytes on wire (6168 bits), 771 bytes captured (6168 bits) on interface 0
- Ethernet II, Src: 192.168.178.1 (c8:25:06:06:23:34), Dst: 192.168.178.69 (60:f8:1d:c9:28:10)
- Internet Protocol Version 4, Src: 212.201.22.203, Dst: 192.168.178.69
- Transmission Control Protocol, Src Port: 80 (80), Dst Port: 51521 (51521), Seq: 1, Ack: 478, Len: 705
- HyperText Transfer Protocol
- HTTP/1.1 200 OK
- Content-Type: text/html
- <!DOCTYPE html>
- <html lang="en">
- <head>
- <meta charset="UTF-8">
- <title>Onlinelabor FH-L\383\274beck</title>
- </head>
- <body>
- <h1>Online-Labor FH-L\383\274beck</h1>
- </body>

Right Window (https traffic):

- Frame 0040: 60 88 1d c9 28 18 00 25 00 06 23 34 08 00 45 00,..#.,..E.
- Frame 0050: 48 80 00 44 61 74 65 30 20 53 75 66 2c 20 32 35 KODI-Daten 1 25
- Frame 0060: 48 80 00 44 61 74 65 30 20 53 75 66 2c 20 32 35 KODI-Daten 1 25
- Frame 0070: 34 33 29 47 44 54 04 00 53 65 72 76 65 72 38 20 43 GHT.. Server:
- Frame 0080: 41 70 63 63 68 65 2f 32 2e 34 2e 37 20 28 55 62 Apache/2.4.7 (Ubuntu)
- Frame 0090: 75 6e 74 75 29 00 80 4c 61 73 74 2d 4d 6f 64 69 untu..L ast-Modi
- Frame 00a0: 65 62 3a 20 54 65 2c 20 53 75 66 2c 20 32 35 1.2816 Tu 21.07.05
- Frame 00b0: 74 20 65 62 3a 20 54 65 2c 20 53 75 66 2c 20 32 35 1.2816 Tu 21.07.05
- Frame 00c0: 47 4d 54 04 8d 45 54 61 67 34 2a 22 32 34 2d 43 GHT.. Eta g: "224-
- Frame 00d0: 35 33 65 38 39 37 35 33 35 39 32 37 61 2a 67 7a 53e09753 5927a-gz

Both windows show a list of captured frames with their details and hex dump. The https traffic shows standard TLS handshake and application data exchange, while the http traffic shows raw HTML content.

Absender- oder Ziel-Adressen fälschen

- ▶ Man gibt sich einfach als der richtige Empfänger aus.
- ▶ Diese Art des Angriffs nennt man Spoofing (oder auch Masquerading, da der Angreifer sich als jemand Vertrauenswürdiger maskiert), und die verschiedenen Varianten beziehen sich auf unterschiedliche Protokolle bzw. unterschiedliche Schichten des OSI Modells.
- ▶ Auf der Anwendungsschicht ist zum Beispiel das Web-Spoofing bekannt, bei dem die URL eines Webservers gefälscht wird.
- ▶ Ein Nutzer wird durch typografisch sehr ähnliche Namen ausgetrickst, wenn er statt auf `paypal.com` auf `paypai.com` geleitet wird. Bei einigen Fonts sind die Namen visuell kaum zu unterscheiden.
 Gibt der Nutzer dort seine Zugangsdaten ein, so landen diese eben nicht bei dem vermeintlichen Anbieter, sondern beim Angreifer.

Absender- oder Ziel-Adressen fälschen

- ▶ Mit diesem Szenario häufig verknüpft ist das E-Mail-Spoofing. Hier wird die Absender- Adresse in E-Mails gefälscht. Relativ einfach, da das Mail-Protokoll SMTP die Absender-Adresse nicht von sich aus prüft.
- ▶ E-Mail-Spoofing wird gern für Spam- und Phishing-Mails verwendet. Bei letzterem wird mittels gefälschtem Absender und vom Layout passender E-Mail vorgegeben, dass die Mail von einem seriösen Unternehmen kommt; im Text sind dann z. B. Links entsprechend dem Web-Spoofing enthalten.
- ▶ Eine weitere Möglichkeit ist das IP-Spoofing, bei dem IP-Adressen gefälscht werden. Der Angreifer benutzt nicht Ähnlichkeiten (Webseite), um das Opfer zu täuschen, sondern die echte IP-Adresse des anderen Kommunikationspartners. Das funktioniert, weil in vielen Protokollen der TCP/IP-Familie vorgesehen ist, dass sich Geräte lediglich über ihre IP-Adresse authentifizieren.

Das geht auch in Mobilfunknetzen

- ▶ Auch in Mobilfunknetzen kann die Sprachkommunikation oft noch mitgehört werden, da diese derzeit noch über GSM erfolgt. Deshalb rät das BSI davon ab, Mobilfunk bei sicherheitsrelevanter Kommunikation zu verwenden (BSI 2008).
- ▶ Die Ursache liegt zum einen in Schwächen der Verschlüsselungsimplementierung. Zum anderen ist das Problem, dass die Authentifizierung nur einseitig erfolgt: Das Endgerät authentifiziert sich gegenüber Basisstation.
☞ Bei LTE behoben.
- ▶ Das hat zur Folge, dass ein Angreifer eine Basisstation fälschen kann (sog. IMSI-Catcher). Meldet sich das Endgerät an einer Basisstation eines Angreifers an, kann der Angreifer die Kommunikation von und zu diesem Endgerät mithören. Dem Endgerät bzw. dessen Nutzer fällt nicht auf, dass es mit einer gefälschten Basisstation kommuniziert (MITM).

Kommunikation umleiten

- ▶ Neben dem Spoofing, bei dem es darum geht, sich als ein anderer Kommunikationspartner zu maskieren, kann man Kommunikation auch umleiten.
- ▶ Eine Möglichkeit, dies zu erreichen, ist, Router dazu zu bringen, Datenverkehr falsch weiterzuleiten. Router nutzen sog. Routingtabellen mit Informationen darüber, welches Netz über welche eigene Schnittstelle oder über andere Router im Netz erreichbar ist. Angreifer können nun gefälschte Routing-Informationen an Router senden und damit diese Tabellen so verändern, dass die Datenpakete an eine „Wunschadresse“ geleitet werden.

- ▶ Sobald die Internetkonnektivität für Organisationen aber sehr wichtig wird, verfügen diese oft über mehrere unabhängige Netzanbindungen. Dadurch müssen sie eigene Routing-Konfigurationen vornehmen, um den Datenverkehr zu lenken. Da die entsprechenden Protokolle nicht vorsehen, dass Routing-Informationen überprüft werden, liegt hier eine gewisse Anfälligkeit für Angriffe vor.
- ▶ Umleitungen funktionieren auch auf der Ebene der IP-Adressen und Domain-Namen: Ein DHCP-Server sorgt in einem lokalen Netz dafür, den Geräten dynamisch IP-Adressen zuzuweisen.
- ▶ Schafft es ein Angreifer, ein eigenes Gerät als DHCP-Server (Rogue DHCP Server) einzubinden, stehen ihm verschiedene Möglichkeiten zur Verfügung, Daten zu sich umzuleiten, z. B. indem er seine eigene Adresse als Default- Gateway angibt, oder indem gefälschte DNS-Server angegeben werden, die falsche Zuordnungen von IP-Adressen zu Domain-Namen vornehmen.

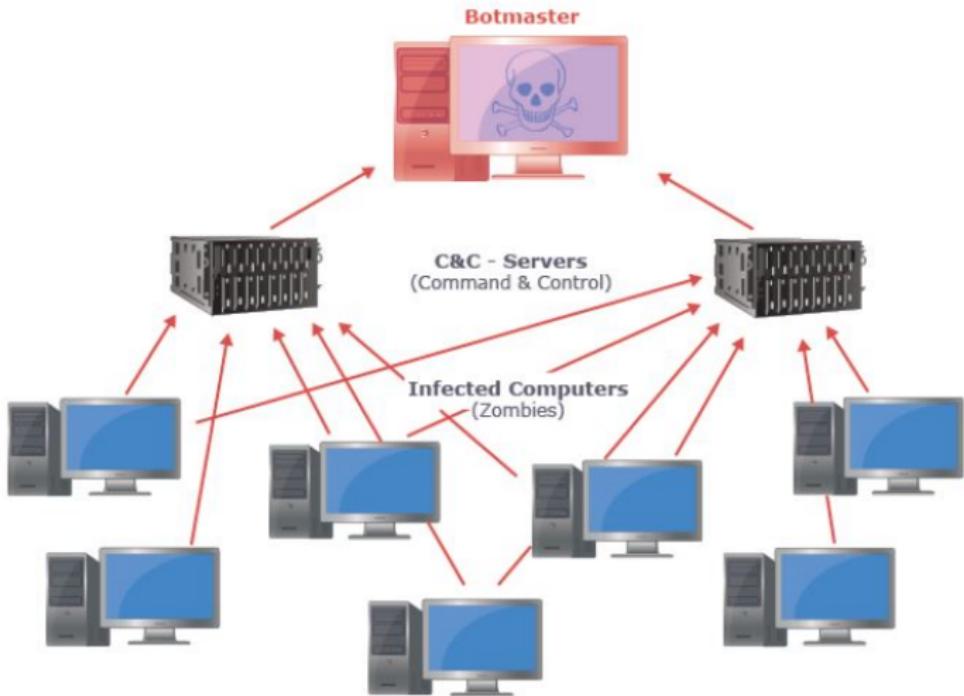
Geräte kommunizieren miteinander (IoT)

- ▶ Vernetzte Geräte sind überall und werden immer mehr. Es sind nicht nur Geräte, um z. B. ein SmartHome zu realisieren, sondern auch in der Gebäudeautomatisierung mit vernetzten Aufzügen, Klimaanlagen oder Fenstersystemen.
- ▶ Es ist zwar relativ schwer, Smart Meters selbst anzugreifen, weil ihre physikalische Sicherheit stark ist, aber es sind u.a. viel genutzte ungesicherte Protokolle, über die Smart Meters auch schon massenweise angegriffen wurden.
- ▶ 2009 wurden in Puerto Rico Smart Meters angegriffen. Der Angriff hatte einen großen Abrechnungsbetrug zur Folge.

- ▶ Auch SIM-Karten werden zur Vernetzung eingesetzt; bekanntestes Beispiel ist vermutlich das Auto, das über eine SIM-Karte Fahrzeugdaten automatisch an Service-Center übermittelt oder bei einem Unfall das Callcenter mit den Fahrzeuginsassen Kontakt aufnehmen lässt.
- ▶ Es ist wichtig bei der Systemgestaltung auch diese Angriffsszenarien zu berücksichtigen und darauf zu achten, dass elementare Sicherheitsmaßnahmen wie verschlüsselte Datenübertragung umgesetzt werden.

Botnetze re-visited

- ▶ Das Bot-Programm gelangt durch herkömmliche Schadsoftware wie Viren, Würmer und Trojaner auf den einzelnen Rechner.
- ▶ Um die einzelnen Rechner zu einer „Armee“ zusammenzuschließen, die gemeinsam den Angriff durchführt, gibt es in dem Botnetz meist einen zentralen Server, der Command-and-Control-Server (C&C-Server).
- ▶ Neuere Botnetz-Strukturen basieren allerdings auf sog. Peer-to-Peer-Netzwerken, die es ermöglichen, dass ein Bot sowohl Befehle erhalten als auch erteilen kann.
- ▶ Peer-to-Peer-Botnetze sind wesentlich robuster gegen Teilausfälle des Netzverbundes.
- ▶ Der initiale Drahtzieher des Angriffs ist dadurch auch viel schwieriger zu identifizieren, da verschiedene Rechner des Botnetzes als „Befehlshaber“ auftreten können.



Einsatzzwecke

- ▶ Bots selbst (also die automatisiert laufenden Programme) werden sowohl für nützliche als auch für zweifelhafte Zwecke genutzt.
- ▶ Eingesetzt werden sie für wiederkehrende Aufgaben, die vor allem keine Interaktion mit einem Benutzer erfordern sollen. Z.B. Webcrawler, die selbständig das Web durchsuchen und indizieren.
- ⚠ Das nutzen auch Bots, die gezielt auf Webseiten veröffentlichte E-Mail-Adressen für Werbezwecke sammeln. Bots werden auch in Computerspielen eingesetzt, um Avatare zu steuern.

- ▶ Botnetze können aus beliebig vielen Bots bestehen und sie werden für Aufgaben eingesetzt, die viel Rechenleistung oder Bandbreite erfordern, oder bei denen viele Informationen zusammengetragen werden sollen.
 - ▶ Durch den Zusammenschluss vieler Bots auf verschiedenen Rechnern kann sowohl die Rechenleistung des infizierten Rechners als auch dessen Internetanbindung genutzt werden.
-  2016 erfolgte ein Angriff auf einen französischen Webhoster mit 1.1 terabit pro Sekunde.
- 

Für Botnetze gibt es ganz unterschiedliche Einsatzzwecke, z.B.

- ▶ die Verbreitung von Spam-Mails (die wiederum Schadsoftware verbreiten können).
- ▶ Überlastungsangriffe (DDOS) auf Onlinedienste.
- ▶ um auf mit Bot-Programmen infizierten Rechnern selbst Schaden anzurichten, wie Antivirensoftware deaktivieren, Schadprogramme herunterladen oder Passwörter ausleiten.
- ▶ Mining zur Generierung von Währungseinheiten einer „Krypto-Währung“.

- ▶ Ein Botnetz wird dabei heutzutage nicht mehr für eine Aktion aufgesetzt oder von einem Angreifer betrieben.
- ▶ Botnetze werden sogar vermietet, um die mit dem Botnet angezapfte Rechenleistung und Bandbreite für beliebige Aktivitäten zu nutzen.
- ▶ Die „Armee“ ist also sehr flexibel einsetz- und vermietbar, sodass sie mit relativ geringem Aufwand neue Aufgaben ausführen kann.

Botnetze sind eine der größten Bedrohungen für IT-Infrastrukturen:

- ▶ Laut BSI ist davon auszugehen, dass im Durchschnitt mehrere Hundert C&C-Server pro Tag aktiv sind.
- ▶ Täglich bis zu 60.000 Infektionen deutscher Systeme sind nicht ungewöhnlich.
- ▶ Botnetze machen die unzähligen kontrollierten Opfer-Rechner selbst zu unwissenden Tätern.
- ▶ Die Abwehr von Angriffen ist umso aufwändiger und teurer, je größer die Botnetze sind (was insbesondere für DDOS-Angriffe relevant ist), und damit möglicherweise ein Privileg von großen Plattformbetreibern.

Gesellschaftliche Fragen:

- ▶ Wem gehören zum Beispiel Bitcoins, die auf Basis eines Botnetzes errechnet wurden, das unfreiwillig kontrollierte Rechner benutzt?
- ▶ Im politischen Wahlkampf werden schon Social Bots zur Meinungsbildung eingesetzt, und immer mehr Parteien werden solche Möglichkeiten nutzen. Mit Social Bots, werden Meinungen automatisiert im Netz zu verbreitet. Welche Relevanz hat die „öffentliche politische Meinung“ dann noch?

Denial of Service

- ▶ Angriffe mit dem Ziel, dass Dienste ihren Dienst verweigern und damit Services oder Daten nicht mehr zugänglich sind, nennt man Denial-of-Service- oder Überlastungs- Angriffe.
- ▶ Bei einem verteilten DoS-Angriff (Distributed DoS, DDoS) erfolgt der Angriff nicht von einem Rechner aus, sondern verschiedene Systeme führen gleichzeitig einen DoS-Angriff auf das Opfer-System durch. Je mehr Systeme beim Angriff mitmachen, desto effektiver wird er: je überlasteter das Opfer, desto schlechter kann es den Angriff abwehren.
- ▶ Meistens wird für den Angriff nicht eine eigene Infrastruktur aufgebaut, sondern man kompromittiert die Systeme anderer und nutzt sie für den eigenen Angriff. Hier ist der Zusammenhang mit Botnetzen: Es wird ein Botnetz aufgebaut oder gar gemietet, um eine DDoS-Attacke durchzuführen.

- ▶ Denial of Service-Angriffe sind im Grunde nichts Neues, haben aber in jüngster Zeit auch in der Allgemeinheit einen hohen Bekanntheitsgrad erlangt.
- ▶ Neuerdings wird das DoS-Prinzip auch dazu verwendet, vorhandene IT-Sicherheitssysteme auszuhebeln. Statt einen DoS-Angriff so lange zu fahren, bis das Opfer den Dienst versagt, erfolgen hier die Angriffe sehr häufig und nur kurzzeitig. Das kann dazu führen, dass Logs, die Angriffe protokollieren, überfüllt werden, sodass andere Angriffe nicht mehr registriert werden.

Wie überlastet man Systeme?

- ▶ Beim sogenannten SYN-Flooding-Angriff werden zunächst ganz reguläre Datenpakete zum Verbindungsaufbau, die SYN-Segmente, an den Server geschickt.
- ▶ Der Server sendet sein SYN/ACK-Segment zurück und wartet auf die Bestätigung des Clients, hier also des Angreifers.
- ▶ Da der Server ständig eine ganze Reihe solcher Anfragen hat, merkt er sich diese Anfragen, die noch nicht bestätigt wurden, in einer speziellen Warteschlange für halb-offene Verbindungen, bis die Bestätigung vom Client bekommt. Der Angreifer antwortet nicht wie vorgesehen mit einem ACKSegment.
- ▶ Beim Server existieren nun eine halb-offene Verbindung in der Warteschlange und diese bleibt bis zu einem Timeout (i.d.R. im Minuten-Bereich).
- ▶ Wird ein Server mit solchen halb-offenen Verbindungsanfragen überschwemmt, läuft die Warteschlange schnell voll, sodass andere Verbindungsanfragen nicht mehr verarbeitet werden können. Damit ist der Dienst für andere Nutzer nicht mehr erreichbar.

Webanwendungen angreifen

- ▶ Die wesentlichen Risiken, die sich durch ungesicherte Webanwendungen (Application Layer) ergeben können, sind
 - ▶ das Umleiten der Nutzer auf nicht-originale Webseiten.
 - ▶ das unbefugte Abgreifen von Daten (insb. Authentifizierungsdaten).
 - ▶ das Einfangen von Schadsoftware. Diese Risiken gehen meist Hand in Hand.
- ▶ Webanwendungen sind sehr häufig nicht gegen solche Risiken geschützt.
- ▶ Das ist deshalbbrisant, weil Webanwendungen auch zunehmend Bestandteil vernetzter Steuerungssysteme und dem „Internet der Dinge“ sind, über die Geräte wie Lampen, Fernseher, Küchengeräte oder auch größere Systeme gesteuert oder überwacht werden.
- ▶ Die Zusammenstellung der Sicherheitsrisiken bei Webanwendungen des Open Web Application Security Project (OWASP) sind sehr lehrreich.

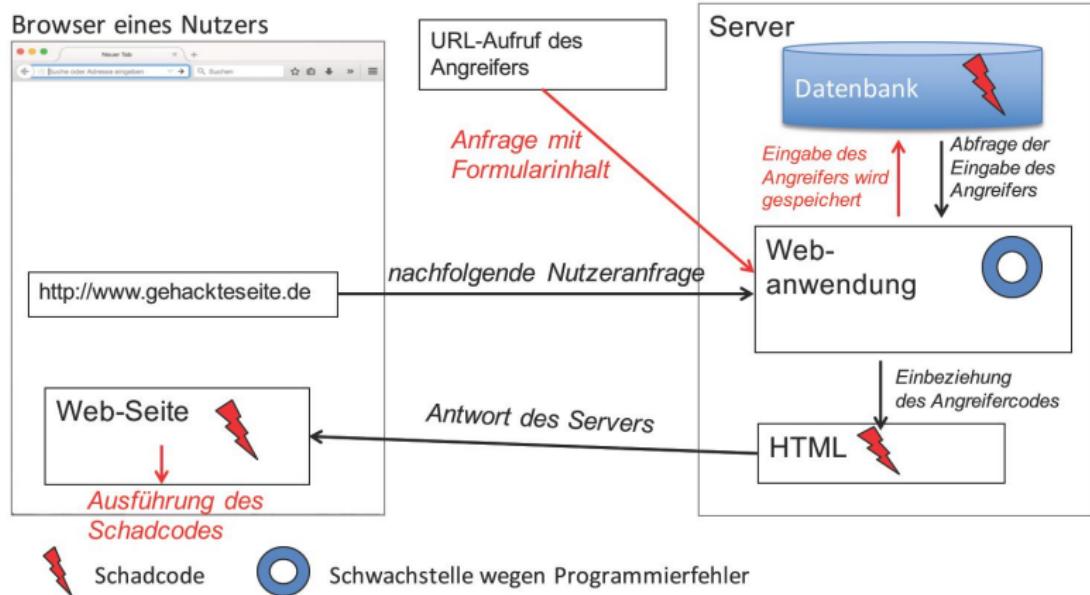
Cross-Site-Scripting (XSS)

- ▶ Es dabei darum, dass im Browser eines Webseiten-Besuchers Scripte ausgeführt werden, die vom Webseitenbetreiber nicht vorgesehen wurden. Ziel eines solchen Angriffs ist meist, Zugangsdaten des Opfers für Webanwendungen zu erhalten, um so Identitätsdiebstahl zu betreiben. Das kann zum Beispiel dadurch erreicht werden, dass Cookies vom Browser ausgelesen werden.
- ▶ Die Schwachstelle, die für XSS ausgenutzt wird, ist also die Fähigkeit der Browser, Script- Code auszuführen (meist handelt es sich um JavaScript). Das Funktionsprinzip von XSS ist, dass ein Angreifer seinen in JavaScript geschriebenen Schadcode so platziert, dass dieser an den Nutzer, der eine entsprechende Website aufruft, weitergeleitet wird. Wenn das geschehen ist, wird der Schadcode vom Browser des Benutzers automatisch ausgeführt.

Persistent Attack

- ▶ Bei der persistenten Variante wird JavaScript-Schadcode persistent in die Webanwendung (genauer: in die Datenbank) eingeschleust, sodass dieser immer wieder an Browser ausgeliefert und vom Browser ausgeführt wird.
- ▶ Dies ist möglich, wenn die Anwendung Benutzereingaben in einem Webformular nicht darauf prüft, ob es sich um Code handelt.

Persistent Attack



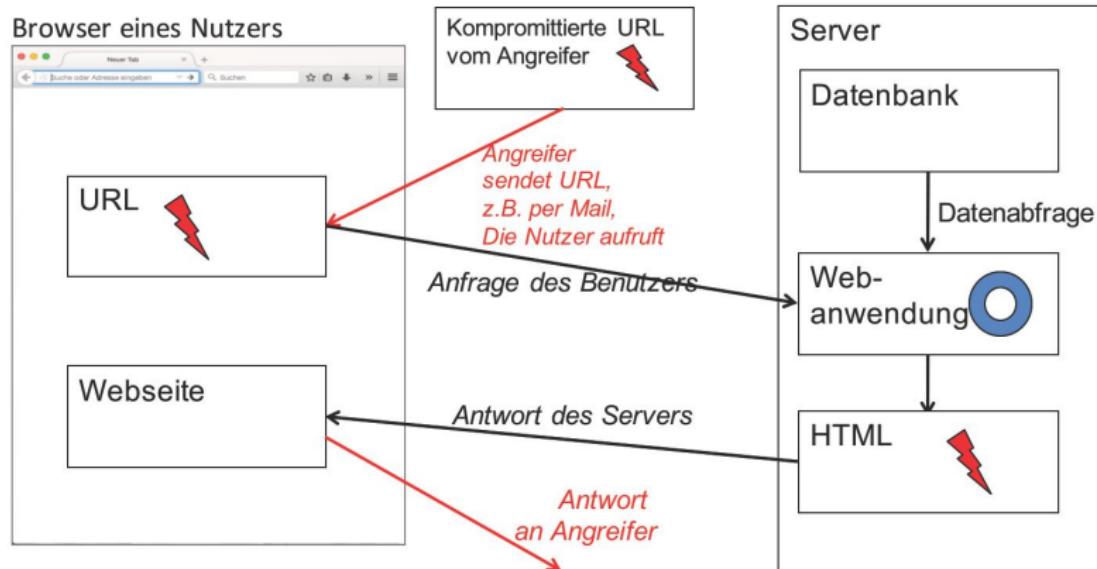
Reflected Attack

- ▶ Bei der reflected Attack wird der Schadcode nicht von der Anwendung gespeichert, sondern nur zurückgespiegelt, also reflektiert. Das heißt, der Schadcode wird nicht zu jedem Webseitenbesucher ausgeliefert, sondern nur zu jenen, die ihn vorher selbst an die Anwendung geliefert haben.
- ▶ Aber wie kommt es dazu, dass Anwender selbst Schadcode senden?

Reflected Attack

- ▶ Bei der reflected Attack wird der Schadcode nicht von der Anwendung gespeichert, sondern nur zurückgespiegelt, also reflektiert. Das heißt, der Schadcode wird nicht zu jedem Webseitenbesucher ausgeliefert, sondern nur zu jenen, die ihn vorher selbst an die Anwendung geliefert haben.
- ▶ Aber wie kommt es dazu, dass Anwender selbst Schadcode senden? Die Reflexion kommt zustande über kompromittierte Links: Der Link selbst enthält den Schadcode.

Reflected Attack



SQL-Injection

- ▶ SQL-Injections gehören ebenfalls zu den schon sehr lange bekannten und weit verbreiteten Angriffsmöglichkeiten auf Webanwendungen.
- ▶ SQL-Injections zielen direkt auf die Datenbank von Webanwendungen.
- ▶ SQL (Structured Query Language) ist die Sprache, mit der für viele Datenbanksysteme Befehle zum Auslesen und Schreiben von Daten aus der/in die Datenbank geschrieben werden.
Injections (Injizierungen, Einschleusungen) sind Angriffe, bei denen nicht vorgesehener Code eingeschleust wird.

SQL-Injection

Mitarbeitersuche

1

Altersbereich

von 25

bis 45; SELECT name, adress, salary FROM personal

3

```
SELECT name, age FROM personal  
WHERE alter >= 25 AND <= 45;
```

```
SELECT name, adress, salary  
FROM personal;
```

Suchergebnisse

2

Müller, 26
Meier, 33

Müller, Spielstraße 15, 21.000
Meier, Probenweg 3, 54.630
Muster, Zielallee 133, 43.800
Schulze, Testtwiete 45, 67.000
Ullmann, Liekstraße 29, 44.360
Werner, Kohlenweg 8, 120.000

Verbreitung von Schadsoftware

Dies kann prinzipiell auf drei Arten geschehen:

1. Ein Angreifer betreibt eine entsprechende Website selbst und bringt die Nutzer über attraktive Angebote dazu, seine Seite zu besuchen.
2. Ein Angreifer kompromittiert Webanwendungen, z. B. über XSS oder SQLi, um dort seine Schadsoftware zu platzieren.
3. Ein Angreifer benutzt Werbebanner, die über die unterschiedlichsten Webseiten ausgeliefert werden können.
☞ In der Regel laufen die Angriffe auf sog. Drive-by-Exploits hinaus, d.h. über Webseiten verteilte und oft in Werbebanner versteckte Schadsoftware, die eine Kontrolle des Nutzerrechners durch einen Angreifer ermöglicht.

Gegenmaßnahmen

- ▶ Die sog. Same-Origin-Policy. Sie untersagt z.B. clientseitigen Scriptssprachen, dass sie auf Objekte zugreifen dürfen, die von anderen Webseiten stammen als der, von der der Script-Code ursprünglich stammt. Dieses Konzept wird standardmäßig von Browsern implementiert.
- ▶ Es obliegt den Entwicklern der Webanwendungen, dafür zu sorgen, dass XSS und SQL-Injection verhindert werden.
- ▶ Webentwickler können entweder selbst entsprechende Sicherheitskonzepte implementieren, oder sie nutzen Frameworks, die diese schon mitbringen.
- ▶ Das Prinzip ist sehr einfach, nämlich, dass bei der Übernahme von Parameter-Werten diese auf gewisse Kriterien hin geprüft werden. Das Einschleusen von SQL-Statements kann durch sog. Prepared Statements verhindert werden. Diese sorgen dafür, dass SQL-Statement nur auf ganz bestimmte Weise und nicht beliebig dynamisch ergänzt werden können.

Zusammenfassung Sicherheitsmängel

- ▶ Standard-Passwörter: Geräte nutzen zur gegenseitigen Authentifizierung Passwörter, die von den Betreibern nicht geändert werden oder sogar nicht geändert werden können.
- ▶ Passwörter oder Schlüssel sind sehr kurz (oft nur 6 Zeichen lang) und ergeben sich aus Gerätenamen.
- ▶ Es erfolgt keine sichere Authentifizierung der Geräte vor einer Kommunikation.
- ▶ Keine verschlüsselte Datenübertragung: Es werden Standard-Technologien benutzt, die keine Verschlüsselung vorsehen, und keine zusätzliche Verschlüsselung konfiguriert.

- ▶ Offene Ports: Ports, die für die Funktionalität des Geräts gar nicht notwendig sind, werden nicht geschlossen.
- ▶ Mangelnde Updates für Firmware: Geräte-Hersteller produzieren Geräte, für die sie keine Updates bereitstellen. Für manche Geräte ist ein Update-Prozess, der von Nutzern durchgeführt wird, nicht einmal vorgesehen.
- ▶ Mit vernetzten Geräten werden eine Menge persönlicher Daten gesammelt, die nicht unbedingt für die Funktionalität gebraucht werden; neben der absichtlichen Weitergabe besteht hier das Risiko, dass sie durch mangelnde Absicherung gegen Angriffe im Netz in falsche Hände geraten.

Schutzmaßnahmen gegen Angriffe im Netz

- ▶ Setzen Sie sich dafür ein, dass Sicherheitsaspekte in Ihrem Kontext berücksichtigt und nicht als lediglich kostenverursachende Last wahrgenommen werden.
- ▶ Messen Sie Webschnittstellen für Smart Devices eine hohe Bedeutung zu. Web-Schnittstellen zur Steuerung von Geräten sind schnell implementiert, liefern Endnutzern großen Mehrwert, werden aber häufig als technisch nicht so relevant eingestuft.
- ▶ Verschlüsseln Sie die Kommunikation im Netz, auch bei Smart Devices.
- ▶ Sichern Sie Ihr internes Netz und Ihre dort eingebundenen Geräte mit Firewalls.
- ▶ Sorgen Sie für Updates.

Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundsatz

Social Engineering

- ▶ Social Engineering wird gemeinhin definiert als die geschickte Beeinflussung anderer Menschen mit dem Ziel, dass sie bestimmte Handlungen ausführen.
- ▶ Hadnagy definiert Social Engineering als „die Wissenschaft, wie man Menschen hinsichtlich bestimmter Aspekte ihres Lebens geschickt und umsichtig in Aktion bringt.“(nicht nur negativ)
- ▶ In der IT-Sicherheit ist der Fall von Interesse, bei dem Personen zum Beispiel dazu gebracht werden, vertrauliche Informationen preiszugeben oder auch Software aus zweifelhafter Quelle zu installieren.
- ▶ Das Social Engineering wird in der Studie der Allianz für Cybersicherheit (ACS 2015) als sechsthäufigste Ursache genannt durch die Sicherheitsvorkehrungen umgangen werden.

- ▶ Eine wichtige und zentrale Aufgabe beim Social Engineering ist das Sammeln von Informationen. Hierbei kann es sowohl um technische Informationen gehen als auch um solche über Mitarbeiter.
- ▶ Möchte ein Angreifer in ein Unternehmensnetzwerk einbrechen, so ist es für ihn wichtig zu wissen, welche Software benutzt wird, welche Schutzmechanismen implementiert sind, welche Datenbanken genutzt werden und welche Unternehmensdaten in welchen Systemen gespeichert und verarbeitet werden.
- ▶ Genauso nützlich ist es zu wissen, welche Mitarbeiter Zugang zu welchen Systemen haben, welche Befugnisse sie haben, mit wem sie im Unternehmen gut vernetzt sind und mit wem nicht.

- ▶ Auch private Informationen über Personen können einem Angreifer gut weiterhelfen, z. B. um Passwörter zu erraten oder um Vertrauen aufzubauen.
- ▶ Einem Angreifer stehen dafür eine ganze Reihe potentieller Quellen zur Verfügung; z.B. sind Websites von Unternehmen oder Personen ersten Anlaufstellen, um Informationen über sein Ziel herauszufinden. Aber auch klassische Suchmaschinen werden eingesetzt, um weiterreichende Hintergrundinformationen zu bekommen.
- ▶ Interessant ist zum Beispiel Whois (ändert sich durch Datenschutz), ein Abfragedienst für Domains.
- ▶ Das Internet kann man auch nach öffentlich erreichbaren Servern von Firmen durchsuchen.
- ▶ Ein Tool zur Informationssuche in diesem Kontext ist Maltego. Dieses Tool hat inzwischen auch Angriffskomponenten integriert.

Dumpster Diving

- ▶ Bei dieser Methode wird Müll durchsucht, z.B. nach Einladungskarten, Telefonnummern, Zugangsdaten. In wichtigen Fällen machen sich Angreifer auch die Mühe, geschredderte Dokumente wieder zusammenzusetzen.
- ▶ Diese Methode hat zunächst nur wenig mit IT-Sicherheit zu tun, aber dort gefundene Informationen helfen auf Umwegen dem Angreifer, in ITSysteme einzudringen. Daher ist es so wichtig, sensible Informationen verlässlich zu vernichten.
- ▶ Oft geht es bei diesen Strategien darum mit diesen Informationen den eigentlichen Social-Engineering-Angriff vorzubereiten und Vertrauen zu Personen aufzubauen.

Vertrauen aufbauen - Pretexting

- ▶ Die Idee ist Menschen „geschickt in Aktion“ zu setzen.
- ▶ Für das Pretexting nutzt der Angreifer u.a. vorher gesammelte Informationen. Er gibt sich eine Rolle (z. B. die eines Administrators, eines Mitglieds der Unternehmensführung o.ä.), und dafür nutzt er z. B. sein Wissen über die Unternehmensstruktur, das soziale Netzwerk der Zielperson und die Unternehmenskultur.
- ▶ Der Angreifer nutzt u.U. auch Requisiten wie passende Werkzeuge, um den Pretext noch überzeugender zu gestalten.
- ▶ Beim Kontakt muss der Angreifer es schaffen, sein Opfer so zu beeinflussen, dass er die gewünschten Reaktionen hervorruft. Dabei unterstützt der Pretext. Hier steht im Prinzip das ganze Arsenal psychologischer Tricks zur Verfügung.

- ▶ Der Angreifer versucht dabei Abhängigkeiten zu schaffen.
- ▶ Ein Angreifer hilft seinem Opfer, z.B. durch technische Hilfe.
- ▶ Bei der „Chef-Masche“ arbeitet man mit Dringlichkeit, um Druck aufzubauen. Zum Beispiel könnte sich ein Angreifer als Systemadministrator ausgeben und behaupten, dass ein Problem gelöst werden müsse, bevor der Chef wieder im Hause ist; oder dass das Unternehmensnetz in die Knie gehen werde, wenn die Zielperson nicht sofort handelte.

Beispiel: Computerbasiertes Social Engineering - Phishing re-visited

- ▶ Beim Phishing wird versucht, mit gefälschten E-Mails, Webseiten oder neuerdings auch Kurznachrichten bei Messengerdiensten Personen dazu zu bringen, ihre Zugangsdaten preiszugeben, also Identitätsdiebstahl zu betreiben.
- ▶ Ein Benutzer erhält eine Mail, die vermeintlich von seiner eigenen Bank kommt. Als Absender steht dort die Bank, das Layout der Mail entspricht der Corporate Identity des Unternehmens. Der Text der Mail enthält Hinweise auf dringenden Handlungsbedarf.
- ▶ Eine Abbuchung müsse bestätigt, persönliche Daten aktualisiert oder ein neues Sicherheitsmerkmal aktiviert werden. Der Text enthält einen Link zur Website, wo sich der Kunde direkt anmelden kann. Folgt der unbedarfte Leser diesem Link, landet er auf einer Website, die ebenfalls nach seiner Bank aussieht.

The screenshot shows an email from 'Sparda-Bank' with the subject 'Sparda-Bank - Aktueller Sicherheitshinweis'. The body of the email contains several redacted sections. A red circle highlights the URL 'http://www.sparda-bank.de/...'. Another red circle highlights the text 'Ihr Konto ist sicher!'. A third red circle highlights the button 'Jetzt anmelden'.

Abbildung: Quelle: BSI

Das Social-Engineering besteht hier aus:

- Das sog. Pretexting (Vertrauen aufb.), d.h. hier sich als Bank auszugeben und E-Mail als auch Website diesem Pretext anzupassen.
 - In der Vertrauensbildung durch das passende Layout, wenn Empfänger der Mail ein Konto bei der besagten Bank hat. (Vorher prüfen oder „viel bringt viel“).
 - Aufbau von Handlungsdruck, um den Mailempfänger dazu zu bringen, auf den Link zu klicken und seine Zugangsdaten preiszugeben.

Profiling and Tracking

- ▶ Weitgehend bekannt ist, dass große Web-Konzerne wie Google und Facebook Daten ihrer Nutzer sammeln, um diesen dann auf ihren Plattformen möglichst passgenaue Werbung anzuzeigen.
- ▶ Sie verdienen einen Großteil ihres Geldes damit, an Werbetreibende Werbeplatz zu vermieten und die Werbung möglichst nur dann auszuspielen, wenn der jeweilige Nutzer zur Zielgruppe passt.
- ▶ Ein soziales Netzwerk wie Facebook kann dabei auf viele Informationen zurückgreifen, die einzelne Nutzer selbst eingestellt haben: Was sie gerade tun, was sie gut finden, mit wem sie welche Nachrichten austauschen etc. Aus diesen Profilen lassen sich schon recht verlässliche Persönlichkeitsprofile erstellen.

- ▶ Informationen und Daten fallen allerdings nicht nur dort an, wo wir sie willentlich und wissentlich teilen, und auch nicht nur im Web.
- ▶ Sie entstehen und werden auch gesammelt beim Einkaufen (Bezahlung per Karte; Treue-Systeme), beim Reisen (Fluggastdaten, Hotelbuchungen), ...
- ▶ In zunehmenden Maße fallen Informationen in der Wohnung z.B. durch smarte Stromzähler, Thermostate oder Beleuchtungsanlagen. Die Daten sind nicht so isoliert, sondern werden verknüpft, z.B. wenn Versandhändler Daten von anderen Unternehmen dazukaufen oder gleich ganze Unternehmen mit Kundendaten übernehmen.

Auswertung der Daten

- ▶ Man spricht auch von Data Mining, d.h. die systematische Anwendung statistischer Methoden auf große Datenbestände (insbesondere „Big Data“ bzw. Massendaten) mit dem Ziel, neue Querverbindungen und Trends zu erkennen.
- ▶ Beim Clustering wird nach Ähnlichkeitsstrukturen gesucht. Zum Beispiel könnte untersucht werden, ob Menschen, die gleiche Produkte kaufen, ähnliche Einkommen oder ähnliche andere Interessen haben. Bei der Klassifikation werden Objekte (hier: Nutzer) vordefinierten Klassen zugeordnet. Zum Beispiel könnten Nutzer in die Gruppen „kreditwürdig“ und „nicht kreditwürdig“ eingeteilt werden.

- ▶ Die Netzwerkanalyse dient dazu, das persönliche Netzwerk eines Nutzers, also dessen soziales Umfeld, auszuleuchten. Zum Beispiel könnte untersucht werden, zu wem der Nutzer enge oder eher lose Beziehungen führt, ob er ein Meinungsmacher ist oder eher mitläuft.
- ▶ Die Bewegungsprofilanalyse untersucht, wo sich ein Nutzer zu welcher Tageszeit meistens aufhält, um z. B. Wohnsitz oder Arbeitsplatz herauszufinden.

Tracking: Automatisierte Verfolgung im Web

- ▶ Mit Web-Tracking werden im Prinzip drei Ziele verfolgt Den Nutzer wiedererkennen: zum Beispiel soll ein Webshop einen Benutzer wiedererkennen können, um ihm bisher ausgewählte Produkte auch nach einer Weile noch im Warenkorb anzuzeigen und um sicherzustellen, dass im Warenkorb nicht Waren, die andere ausgewählt haben, landen.
- ▶ Das Nutzerverhalten beobachten: ein Webshop möchte zum Beispiel herausfinden, wie oft und wie lange sich Nutzer bestimmte Seiten oder Produkte anschauen, um die Angebote bzw. deren Darstellung optimieren zu können.
- ▶ Den Nutzer kennenlernen: Ein Shop möchte zum Beispiel wissen, welche Interessen ein Nutzer hat, welche anderen Angebote im Netz er anschaut, wie viel er einkauft u.v.m. Hierfür ist das erste Ziel, den Nutzer wiederzuerkennen, natürlich eine Voraussetzung.

- ▶ Beim Web-Tracking geht um das Sammeln von Daten, die der Nutzer nicht unbedingt selbst aktiv mitteilt (Facebook-Profile, Mails, Bestellungen), sondern die durch seine Aktivitäten im Web automatisch entstehen. Ursprünglich wurden technische Methoden zur Wiedererkennung entwickelt (z. B. Cookies), um sinnvolle Funktionalitäten im Web überhaupt implementieren zu können.
- ▶ Heutzutage gehen die Tracking-Methoden über solche auf Funktionalität fokussierte Anwendungsszenarien hinaus. Ziel ist es, Nutzerverhalten über eine längere Zeit und über verschiedene Webangebote hinweg analysieren zu können.
- ▶ Web-Tracking ist eine automatisierte Form der Beobachtung.



Abbildung: Webtracking

Beim Webtracking finden im Prinzip folgende Informationsflüsse statt:

- ▶ Zwischen Seitenbetreiber und Nutzer-Browser
 - ▶ Ein Nutzer ruft eine Webseite auf. Damit sendet der Browser schon automatisch viele Daten an den Seitenbetreiber.
 - ▶ Der Seitenbetreiber liefert Website mit Tracking-Code an Browser aus.
- ▶ Zwischen Nutzer-Browser und Tracker
 - ▶ Der Tracking-Code wird im Browser des Nutzers aktiv und nimmt Verbindung zum Tracking-Anbieter auf.
 - ▶ Der Tracker sendet an den Browser ggf. weiteren Code, der zur Ausführung kommt.

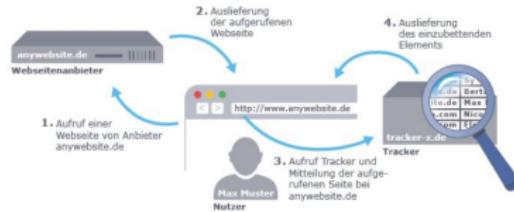


Abbildung: Webtracking

- ▶ Zwischen Seitenbetreiber und Tracker
 - ▶ Tracker stellt ggf. Code zur Einbettung für den Seitenbetreiber bereit.
 - ▶ Tracker liefert dem Seitenbetreiber Analyse-Ergebnisse.

- ▶ Ein besonders einfacher Fall liegt vor, wenn der Seitenbetreiber selbst Nutzeraktivitäten analysiert. In den Anfangsjahren der Analyse war das auch meist der Fall. Der Seitenbetreiber nutzte z.B. Logdaten über Seitenbesuche, Seitenwechsel sowie Browserdaten, um auf diesen Daten sein Angebot zu optimieren.
- ▶ Drittanbieter für das Tracking kamen erst später auf, um Seitenbetreibern ohne Zugang zu oder Kenntnisse über Server-Logdaten auch Analysemöglichkeiten an die Hand zu geben.
- ▶ Mittlerweile sind oft mehrere Tracker und Seitenbetreiber im Spiel. Das dritte Ziele möglichst viel über einen Nutzer zu erfahren, erreicht nämlich umso besser, je mehr Daten aus verschiedenen Quellen benutzt werden. Daher sind Tracker effektiver, wenn sie in möglichst vielen Websites eingebunden sind.

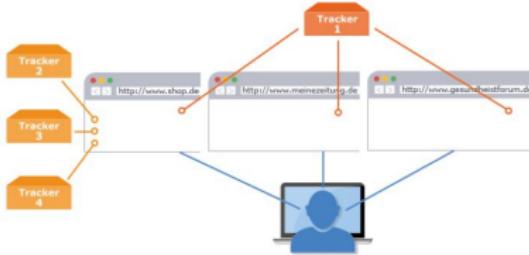


Abbildung: Webtracking-Netz

- ▶ Durch das Einbinden mehrerer Tracker in Websites entsteht breites Netz entsteht, über das das Nutzerverhalten beobachtet wird.
- ▶ Das bedeutet bezüglich der Datenflüsse, dass Informationen, die über einen Shop A von einem Nutzer gesammelt wurden, von Shop B genutzt werden können (wobei hier normalerweise keine konkreten, auf einzelne Nutzer bezogene, sondern lediglich statistische Informationen ausgetauscht werden.)

Beim Tracking spielen häufig Drittanbieter eine Rolle.

Sicherheitstechnisch ist das relevant, denn

- ▶ Drittanbieter haben einen sehr großen Datenbestand, dessen Sicherheit und dessen Nutzungskontexte vor allem für die Nutzer intransparent sind.
- ▶ Seitenbetreiber müssen Fremdcode einbinden, dessen Sicherheit sie oft nicht einschätzen können.
- ▶ Tracker werden meist in Verbindung mit Werbeanzeigen benutzt; auch diese können Schadcode enthalten.

- ▶ Tracking-Methoden müssen eine Verbindung zwischen sich und dem Nutzer initiieren, sie müssen Informationen über die aufgerufene Seite erhalten und sie müssen den Nutzer identifizieren.
- ▶ Manche Methoden sind auf eine dieser Aufgaben spezialisiert, andere erfüllen gleich mehrere Aufgaben.
 - ▶ Das Browser-Fingerprinting zeigt z.B. auf, wie viel über einen Nutzer allein durch seinen Browser in Erfahrung gebracht werden kann.
 - ▶ Mit mehreren Arten von Cookies lassen sich Informationen zur Wiedererkennung auf dem Rechner des Nutzers speichern und wieder auslesen.
 - ▶ Mit Referern erfährt der Tracker, auf welcher Website ein Nutzer sich gerade befindet.

- ▶ Bei einem klassischen Aufruf einer Webseite, wofür das HTTP-Protokoll zur Übertragung von Webseiten verwendet wird, ruft ein Nutzer eine Website auf, dann sendet der Browser eine Anfrage (Request) an den Server.
- ▶ Diese sieht mehrere Parameter vor, u.a. die HTTP-Methode (die bekanntesten sind GET und POST) mit der angeforderten Seite (z. B. index.html), der Host mit der Domain-Adresse., z. B.
`GET index.html Host: www.seite.de`
- ▶ Enthält `www.seite.de` enthält einen Link zur Seite der HS-EL und klickt der Nutzer darauf, sendet sein Browser wiederum eine ähnliche Anfrage, diesmal eben an `www.hs-el.de`. Die Anfrage kann neben vielen weiteren Parametern auch einen Referer enthalten, der angibt, von welcher Seite aus die aktuelle aufgerufen wurde, also von wo sie referenziert wurde.

- ▶ Solche Referer kann der Server der Zielseite nutzen, um zum Beispiel Inhalte der angefragten Seite dynamisch anzupassen (Beispiel: kommt der Nutzer gerade von einem Forum für Fahrräder, könnte ihm auf der neuen Seite die Fahrradwerbung angezeigt werden).
- ▶ Meist dienen sie aber eher dem Protokollieren von Nutzeraktivitäten. Referer werden aber nicht nur bei einer expliziten Anfrage eines Nutzers mitgeliefert, sondern z. B. auch, wenn Webseiten weitere Elemente nachladen, die in die Seite eingebunden werden; klassischerweise sind das Bilder, Scripte und Stylesheets.

- ▶ Ein Request mit Referer kann z. B. so aussehen:
GET https://www.hs-el.de/aktuelles/neuigkeiten/
Host: www.hs-el.de
Referer: http://www.seite.de
- ▶ Bindet www.seite.de einen Tracker ein, dazu wird dann z.B. ein Script nachgeladen, das aber nicht vom eigenen Server, sondern von www.tracker.de geladen wird.
- ▶ Der Browser schickt dabei automatisch seine Anfrage an den entsprechenden Server, die z. B. so aussehen kann:

GET /trackingcode.js

Host: www.tracker.de

Referer: http://www.seite.de/forum/bikes

Der Tracker weiss jetzt, dass der Nutzer sich im Forum zu Bikes auf meineegeliebteseite.de befunden hat. Je mehr Seiten den Tracker www.tracker.de einbinden, umso mehr sieht der Tracker, welche Seiten sich die Nutzer so anschauen.

Abbildung: Netzwerkanalyse

- ▶ In Firefox kann man unter dem Reiter Webentwickler Netzwerkanalysen betreiben, um Trackingaktivitäten festzustellen.
- ▶ Um weitere als Nutzer Analysen durchführen zu können, hat Mozilla das Add-On Lightbeam entwickelt, zu finden unter [Lightbeam for Firefox](#). Einmal installiert, zeichnet es Ereignisse auf, die beim aktiven Besuch von Websites sowie durch Anfragen an Drittanbieter im Hintergrund entstehen.

- ▶ Mit der Einbindung des Tracker-Codes wird automatisch eine Verbindung zwischen Browser und Tracker aufgebaut.
- ▶ Zusätzlich erfährt der Tracker, welche Seite der Nutzer gelesen hat.
- ▶ Die Verbindung kann nicht nur über eingebundene Scripte sondern auch mit im Webseiten-Code eingebauten GEToder POST-Requests zu veranlasst werden. Außerdem wird auch die Einbindung von Medien dazu verwendet (z. B. Bilder, Videos). Auch mit Pop-ups und iFrames ist dies möglich.

Wie erkennt der Tracker aber, dass verschiedene Verbindungen zu ihm vom gleichen Nutzer kommen?

Nutzer wiedererkennen: Cookies

- ▶ Cookies sind kleine Textdateien, die Webanwendungen auf dem Rechner des Nutzers ablegen und wieder auslesen können.
- ▶ Sie wurden ursprünglich erfunden, damit Webanwendungen konsistent über mehrere Seiten hinweg funktionieren können.
- ▶ Sie werden inzwischen allerdings auch zum Tracken benutzt, da in Cookies Informationen zum Nutzer gespeichert werden können.
- ▶ Das HTTP-Protokoll sieht vor, dass Cookies nur von den Servern ausgelesen werden dürfen, die sie auch gesetzt haben. Allerdings gibt es in Webanwendungen auch sog. Third-Party-Cookies, die zum Beispiel von eingeblendeten Werbebannern gesetzt werden.

- ▶ Nutzer können im Browser einstellen, ob Cookies gespeichert werden dürfen, wann sie gelöscht werden sollen oder sie gar selbst zu einem beliebigen Zeitpunkt löschen.
- ▶ Wurden Cookies gelöscht, setzen die Webanwendungen neue, die auch neue Identifikatoren für den Nutzer beinhalten.
- ▶ Es gibt inzwischen eine Vielzahl weiterer Cookie-Varianten, die es Nutzern viel schwerer bis unmöglich machen, die Kontrolle über Cookies zu behalten.

Die sog. Evercookies sind eine Applikation, mit der Cookies geschrieben werden, die quasi unlösbar sind bzw. sich nach Löschtätigkeiten selbst wieder herstellen. Daher werden sie auch Zombie-Cookies genannt.

Nutzer wiedererkennen: Browser Fingerprinting

- ▶ Browser senden einem Server zur Auslieferung einer Seite die IP-Adresse, den Browertyp (Firefox, Chrome etc.), das Betriebssystem (Windows, Mac, Linux etc.) und die eingestellte Sprache.
- ▶ Mit entsprechenden Scripten verrät der Browser auch, welche Add-Ons/Erweiterungen und Schriftarten installiert sind, die Bildschirmauflösung oder die Zeitzone.
- ▶ Die Gesamtheit dieser Daten nennt man den Browser-Fingerprint, den Fingerabdruck, den Browser hinterlassen.
- ▶ Über diesen können Nutzer von Servern wiedererkannt werden.

☞ Panopticlick ist ein Projekt der Electronic Frontier Foundation (EFF). Auf der Seite Panopticlick können Sie Ihren Browser bzw. dessen Konfiguration testen lassen. Als Ergebnis erhalten Sie eine Liste mit Daten über Ihren Browser sowie eine Einschätzung, wie einzigartig Ihr Browser unter den bisher getesteten ist.

Grundlegende Schutzmaßnahmen gegen Angriffe auf persönliche Daten

- ▶ Nutzen Sie sichere Passwörter.
- ▶ Gehen Sie sorgsam mit Zugangsdaten um.
- ▶ Gehen Sie generell sorgsam mit der Preisgabe von Informationen um.
- ▶ Machen Sie Ihre Kollegen und Mitarbeiter auf die Gefahr des Social Engineerings aufmerksam.
- ▶ Reduzieren Sie die Tracking-Möglichkeiten.
- ▶ Seien Sie achtsam bei der Nutzung von Tracking-Tools und entsprechendem Code.

Sollten Sie selbst Tracking-Tools einsetzen müssen, prüfen Sie genau, zu welchem Anlass und welchen Tracker Sie nutzen. Schützen Sie Ihre eigenen Kunden vor Schadcode oder unnötigem Tracking.

Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundsatz

Security by Default

Security by Default

Bei diesem Gestaltungsprinzip geht es darum, in konfigurierbaren Systemen die Standardeinstellungen mit möglichst strengen Sicherheitsvorkehrungen zu versehen. Damit soll gewährleistet werden, dass ein größtmöglicher Schutz bei Inbetriebnahme des Systems gewährleistet wird. Typische Beispiele hierfür sind:

- ▶ Ein Gerät hat kein Standardpasswort. Stattdessen muss das Passwort bei Inbetriebnahme vom Nutzer gesetzt werden.
- ▶ Falls vom Werk vorgegebene Passwörter erforderlich sind, so sind diese nicht für alle Geräte gleich, sondern Zufallspasswörter.
- ▶ Sicherheitskritische Features sind per default deaktiviert.
- ▶ Standardzugriffsrechte für Nutzer oder Geräte ist „none“.

Security by Design

Security by Design bedeutet, dass Bedrohungsszenarien für das zu gestaltende System im Vorwege analysiert, daraus Anforderungen zum Schutz an das System abgeleitet und entsprechende Mechanismen implementiert werden.

Das Prinzip Security by Design ist damit ein Gegenentwurf zur Security by Obscurity, bei der Sicherheit dadurch hergestellt werden soll, dass Systeme bzw. deren Funktionsweise geheim gehalten werden.

☞ Das BSI veröffentlicht daher verschiedene Broschüren und Reports zu den Mindeststandards, die für die Sicherheit bei z. B. eCommerce-Anbietern, Web-Browsern, Cloud Computing oder auch zu Smart Metern umgesetzt werden sollten. Hier findet man noch weitere Prinzipien wie Principle of psychological acceptability oder Principle of economy of mechanism.

Privacy by Default

Privacy by Default

Das Prinzip fordert - analog zur Security by Default -, dass in konfigurierbaren Systemen die Standardeinstellungen einen möglichst hohen Datenschutz vorsehen.

💀 Beispiel Facebooks Privatsphäre-Einstellungen:
Nach Anlegen eines persönlichen Profils auf dieser Plattform war dieses per Default für die ganze Welt sichtbar.

Privacy by Design

Bei diesem Prinzip wird im Vorwege geklärt, wie sicher Daten gespeichert und übertragen werden, welche Daten überhaupt erhoben und verarbeitet und wie die Zugriffe darauf geregelt werden. Dem Prinzip zufolge sollten

- ▶ Sicherheitsaspekte schon bei der Planung der Anwendung berücksichtigt,
- ▶ der Umfang von zu verarbeitenden Daten minimiert werden,
- ▶ bestimmte Datenerhebungen auch ausgeschaltet werden können.

- ▶ Die Prinzipien Privacy by Default und Privacy by Design fokussieren auf die Sicherheitsanforderung Vertraulichkeit und stärken den Datenschutz bzw. die Rechte von Anwendern.
- ▶ Das Bundesdatenschutzgesetz macht Vorgaben, wie der Datenschutz technisch und organisatorisch umzusetzen sei.
- ▶ Das wird auch im Grundschutzkatalog des BSI aufgegriffen. Deren Umsetzung im Gestaltungsprozess folgt dem Prinzip Privacy by Design.

Beispiel Smart Metering Gateway

- ▶ Die Entwicklung einer intelligenten Stromversorgung erfordert Datensicherheit und Datenschutz. Hier kommt das sog. Smart Metering Gateway ins Spiel.
- ▶ Entsprechend ist der Datenschutz auch beim Entwurf des „Gesetzes zur Digitalisierung der Energiewende“ berücksichtigt worden: in Form von Datenschutzanforderungen für die Smart Meter Gateways (BMWi und BSI 2015), die sich in den Technische Richtlinien und Schutzprofilen wiederfinden.
- ▶ Das bedeutet, dass diese Datenschutzanforderungen auch Bestandteil der im Gesetzesentwurf vorgeschriebenen Zertifizierung und somit ein gutes Beispiel sind, wie Privacy by Design gefördert werden kann.

Auszug aus den Anforderungen (gekürzt wiedergegeben):

- ▶ Die Erhebung und Nutzung der Daten ist ohne Zustimmung des Verbrauchers nur soweit erlaubt, wie es für energiewirtschaftliche Zwecke erforderlich ist.
- ▶ Die Ableseintervalle sind möglichst datensparsam vorgegeben, sodass keine Rückschlüsse auf das Verhalten der Nutzer gezogen werden können.
- ▶ Die Daten werden nur anonymisiert, pseudonymisiert oder aggregiert übermittelt.
- ▶ Die Daten werden nicht extern verarbeitet, sondern lokal, direkt beim Verbraucher.
- ▶ Die Energiedaten werden an möglichst wenige Stellen übermittelt.
- ▶ Es sind strenge Löschfristen für die Daten vorgegeben.
- ▶ Kommunikations- und Verarbeitungsschritte sind zu jeder Zeit für den Verbraucher sichtbar und nachweisbar.

Standards, Richtlinien und Vorgaben

- ▶ Es gibt auch Standards und Regularien für Informationssysteme und hier speziell für sicherheitsrelevante Themen, die von internationalen und nationalen Gremien und Arbeitsgruppen entwickelt werden.
- ▶ Die internationale ISO-Norm 27001 Anforderungen für die Implementierung von geeigneten Sicherheitsmechanismen.
- ▶ Dabei wird die Herstellung, Einführung, der Betrieb, die Überwachung, Wartung und Verbesserung eines Informationssicherheitssystems berücksichtigt.
- ▶ Die Norm ist so formuliert, dass sie von verschiedenen Organisationen gut adaptiert werden kann (also z. B. privatwirtschaftliche oder auch staatliche Organisationen).
- ▶ Die ISO 27001 wird auch als Grundlage für Qualitätssiegel herangezogen und dient als Grundlage für den IT-Grundschutz des BSI.

Das neue IT-Sicherheitsgesetz

- ▶ Der Bundestag Mitte Juni 2015 nach mehreren Jahren Vorlauf das „Gesetz zur Erhöhung der Sicherheit informationstechnischer Systeme“(IT-Sicherheitsgesetz) verabschiedet.
- ▶ Er will dadurch eine verbindliche Grundlage dafür schaffen, die Sicherheit von IT-Systemen in Deutschland zu erhöhen.
- ▶ Gründe:
 - ▶ Weiterhin angespannte IT-Sicherheitslage in Deutschland, sowie Defizite im Bereich der IT-Sicherheit genannt, die es abzubauen gelte.
 - ▶ Der Schutz der Bürger im Internet müsse verbessert werden.
 - ▶ Die Rolle des Bundeskriminalamts (BKA) und des Bundesamts für Sicherheit in der Informationstechnik (BSI) sei zu stärken.

Der BSI Grundschatz

Das BSI bietet mit seinem IT-Grundschatz (BSI 2016) ein ganzheitliches Konzept für die Informationssicherheit.

- ▶ Der IT-Grundschatz stellt einerseits eine Vorgehensweise und Methodik bereit, um Informationen und Informationssysteme zu schützen.
- ▶ Andererseits bieten die Grundschatz-Kataloge strukturierte Maßnahmen für die Risikobewertung, Analyse vorhandener und Planung zukünftiger Systeme zur Verfügung.

Für wen gilt das neue Gesetz?

Unternehmen aus Bereichen, die als von zentraler Bedeutung für die Gesellschaft angesehen werden, wie z.B.

- ▶ Energie (Elektrizität, Gas, Öl, alternative Energien, ...).
- ▶ Informationstechnik und Telekommunikation.
- ▶ Transport und Verkehr.
- ▶ Gesundheit (Krankenhäuser, Pharmahersteller, Labore, ...).
- ▶ Wasser (Wasserversorgung und Abwasserentsorgung).
- ▶ Ernährung.
- ▶ Finanz- und Versicherungswesen.
- ▶ Bundesbehörden.

Welche Unternehmen allerdings genau als „Kritische Infrastrukturen“(KRITIS) gelten, wird noch per Rechtsverordnung festgelegt.

Welche Auswirkungen hat das neue Gesetz?

- ▶ KRITIS-Unternehmen sind durch das IT-Sicherheitsgesetz verpflichtet, bestimmte Mindest-Sicherheitsstandards für ihre IT-Systeme einzuhalten.
- ▶ Diese Anforderung hat die Entwicklung eines allgemein gültigen Informationssicherheits-Managementsystems (ISMS) im Blick, das konform zu den ISO 27001 Standards sein soll.
☞ BSI-Grundschutz.
- ▶ Die betroffenen Unternehmen werden also verpflichtet, organisatorische und technische Vorkehrungen zur Vermeidung von Störungen ihrer informationstechnischen Systeme, Komponenten oder Prozesse zu treffen, die für die Funktionsfähigkeit der von ihnen betriebenen kritischen Infrastrukturen maßgeblich sind.

Wie erfolgt die Überprüfung der betroffenen Unternehmen?

- ▶ Ein Sicherheitskonzept ist nie etwas statisches und KRITIS Unternehmen sind mit dem neuen Gesetz verpflichtet, die Umsetzung der geforderten Sicherheitsmaßnahmen mindestens alle zwei Jahre beispielsweise durch Audits oder Zertifizierungen nachzuweisen.
- ▶ Die Ergebnisse müssen an das BSI gemeldet werden - einschließlich möglicherweise aufgedeckter Sicherheitsmängel, deren Beseitigung das BSI im Anschluss verlangen kann.
- ▶ Die genaue Ausgestaltung der Audits wird dabei nicht im Gesetz geregelt, sondern soll u.a. bereits bestehende branchenspezifische Sicherheitsstandards berücksichtigen, wie beispielsweise die ISO 27001 Norm für Rechenzentren und technische Dienstleister.

Wie erfolgt die Überprüfung der betroffenen Unternehmen und welche Kosten entstehen?

- ▶ Die oben bereits erwähnten Meldepflichten im Fall der Aufdeckung von Sicherheitsproblemen im Rahmen von Audits oder Zertifizierungen erfolgen in der Regel anonym. Für den Fall, dass ein vollständiger Systemausfall droht, muss allerdings der Name des Unternehmens an das BSI gemeldet werden.
- ▶ Die Nicht-Meldung solcher Sicherheitsvorfälle kann mit Strafen von bis zu 100.000 Euro geahndet werden.
- ▶ Es entstehen Kosten für die Überprüfung und Einhaltung des Sicherheitsniveaus durch Audits und die Zertifizierung.
- ▶ Zusätzlich fallen durch die Meldepflichten bei Sicherheitsvorfällen weitere Kosten an.

Einführung in die IT-Sicherheit

Grundbegriffe

Mathematisch-kryptologische Grundlagen der IT-Sicherheit

Angriffe auf IT-Infrastrukturen

Angriffe auf den Menschen

Sicheres Systemdesign/Sichere Organisation

Management für Informationssicherheit - BSI Grundschutz

- ▶ Technologie kann nicht alle IT-Sicherheitsprobleme lösen,
 - ▶ z.B. verschlüsselte Dateien sind nicht sicher, wenn das Passwort unter der Tastatur klebt.
- ▶ Gesamtheit der Maßnahmen wichtig:
 - ▶ technische, z.B. verschlüsselte Festplatten,
 - ▶ organisatorische, z.B. Festlegung von Verantwortlichkeiten,
 - ▶ personelle, z.B. Schulung, Einweisung, Sensibilisierung von Mitarbeitern,
 - ▶ und infrastrukturelle, z.B. Gebäudesicherung.

Gesamtheit der Maßnahmen werden in einem Managementsystem zusammengefasst, dem Informationssicherheitsmanagementsystem (ISMS), dabei ist hier der Begriff System hier nicht als technisches System zu verstehen.

Informationssicherheitsmanagementsystem (ISMS)



Quelle :BSI

- ▶ Umfasst alle Regelungen, die für die Steuerung und Lenkung der Aufgaben und Aktivitäten nötig sind, um Informationssicherheit in einem Unternehmen zu etablieren, kontrollieren und zu verbessern.
- ▶ Hierzu gehören
 - ▶ Management-Prinzipien.
 - ▶ Ressourcen.
 - ▶ Mitarbeiter.
 - ▶ Sicherheitsprozess.

Management-Prinzipien

► Leitungsebene

- ▶ übernimmt Gesamtverantwortung für Informationssicherheit.
 - ▶ sieht Informationssicherheit als integralen Bestandteil.
 - ▶ steuert und hält Informationssicherheit aufrecht.
 - ▶ setzt erreichbare Ziele.
 - ▶ wägt Kosten gegen Nutzen ab.
 - ▶ übernimmt Vorbildfunktion.
-
- ▶ Informationssicherheit als kontinuierlichen Prozess leben.
 - ▶ Auf lückenlose Kommunikation und Wissensaustausch achten.

Ressourcen

- ▶ Leitungsebene muss
 - ▶ finanzielle,
 - ▶ personelle,
 - ▶ zeitliche, Ressourcen in ausreichendem Maße bereitstellen.

Mitarbeiter

- ▶ Mitarbeiter müssen eingebunden sein.
- ▶ Schulungen und Sensibilisierung sind Grundvoraussetzung für funktionierenden Sicherheitsprozess.
- ▶ Motivation für das Erkennen und Melden von Sicherheitsvorfällen schaffen.

Sicherheitsprozess

- ▶ Der ISMS Sicherheitsprozess umfasst
 - ▶ IT-Sicherheitsziele.
 - ▶ IT-Sicherheitsstrategie.
 - ▶ Sicherheitskonzept.
 - ▶ Informationssicherheitsorganisation.
- ▶ IT-Sicherheitsziele bestimmen die IT-Sicherheitsstrategie.
- ▶ Sicherheitskonzept und IS-Organisation unterstützen bei der IT-Sicherheitsstrategie.

Zur Umsetzung einer Sicherheitsstrategie



Abbildung: Quelle:BSI

Lebenszyklus des Sicherheitskonzeptes

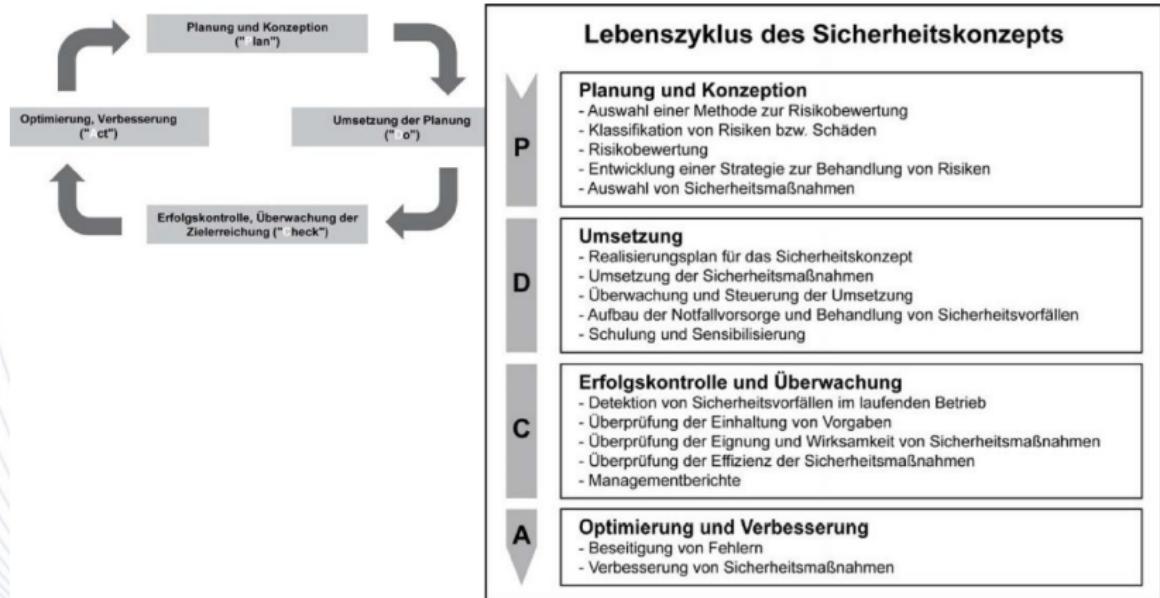


Abbildung: Quelle: BSI

ISMS des BSI: IT-Grundschutz

- ▶ Prozessbeschreibung des Bundesamtes für Sicherheit in der Informationstechnik (BSI, <https://www.bsi.bund.de>).
- ▶ Für Standardkomponenten eines IT-Systems sind zusätzlich sog. IT-Grundschutzkataloge veröffentlicht, dort sind Maßnahmen beschrieben, die einen normalen Schutzbedarf abdecken.
 - ▶ Übersicht unter https://www.bsi.bund.de/DE/Themen/ITGrundschutz/ITGrundschutzKataloge/itgrundschutzkataloge_node.html
 - ▶ Standardkomponenten werden Bausteine genannt.
- ▶ IT-Systeme, die einen hohen bis sehr hohen Schutzbedarf haben, müssen zusätzlich Gefährdungs- und Risikoanalysen durchgeführt werden.

BSI: Bundesamt für Sicherheit in der Informationstechnik



LEICHTESPRACHE GEBÄRDENSPRACHE ENGLISH KONTAKT LOGIN

Suchbegriff

Themen Das BSI Presse Publikationen Service

IT-Grundschutz

IT-Grundschutz - die Basis für Informationssicherheit - Der vom BSI entwickelte IT-Grundschutz ermöglicht es, notwendige Sicherheitsmaßnahmen zu identifizieren und umzusetzen. Viele Arbeitsprozesse werden elektronisch gesteuert und große Mengen von Informationen sind digital gespeichert, werden verarbeitet und in Netzen übermittelt. Damit sind die Institutionen in Wirtschaft und Verwaltung und jeder Bürger von dem einwandfreien Funktionieren der eingesetzten IT abhängig.



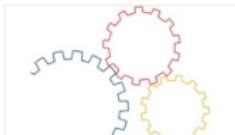
IT-Grundschutz-Kataloge

Die IT-Grundschutz-Kataloge enthalten organisatorische, technische, personelle und infrastrukturelle Empfehlungen.



IT-Grundschutz-Schulung

Ressourcenplanung, Technik, Sicherheitskonzept: Lernen Sie den IT-Grundschutz in unserer Schulung kennen.



IT-Grundschutz-Standards

Die BSI-Standards enthalten Empfehlungen zu Methoden, Prozessen, Vorgehensweisen und Maßnahmen zur Informationssicherheit.

BSI: Bundesamt für Sicherheit in der Informationstechnik



LEICHTESPRACHE GEBÄRDENSPRACHE ENGLISH KONTAKT LOGIN

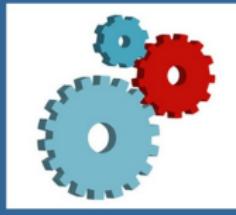
Suchbegriff



Themen Das BSI Presse Publikationen Service

IT-Grundschutz

IT-Grundschutz - die Basis für Informationssicherheit - Der vom BSI entwickelte IT-Grundschutz ermöglicht es, notwendige Sicherheitsmaßnahmen zu identifizieren und umzusetzen. Viele Arbeitsprozesse werden elektronisch gesteuert und große Mengen von Informationen sind digital gespeichert, werden verarbeitet und in Netzen übermittelt. Damit sind die Institutionen in Wirtschaft und Verwaltung und jeder Bürger von dem einwandfreien Funktionieren der eingesetzten IT abhängig.



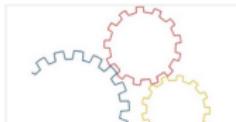
IT-Grundschutz-Kataloge

Die IT-Grundschutz-Kataloge enthalten organisatorische, technische, personelle und infrastrukturelle Empfehlungen.



IT-Grundschutz-Schulung

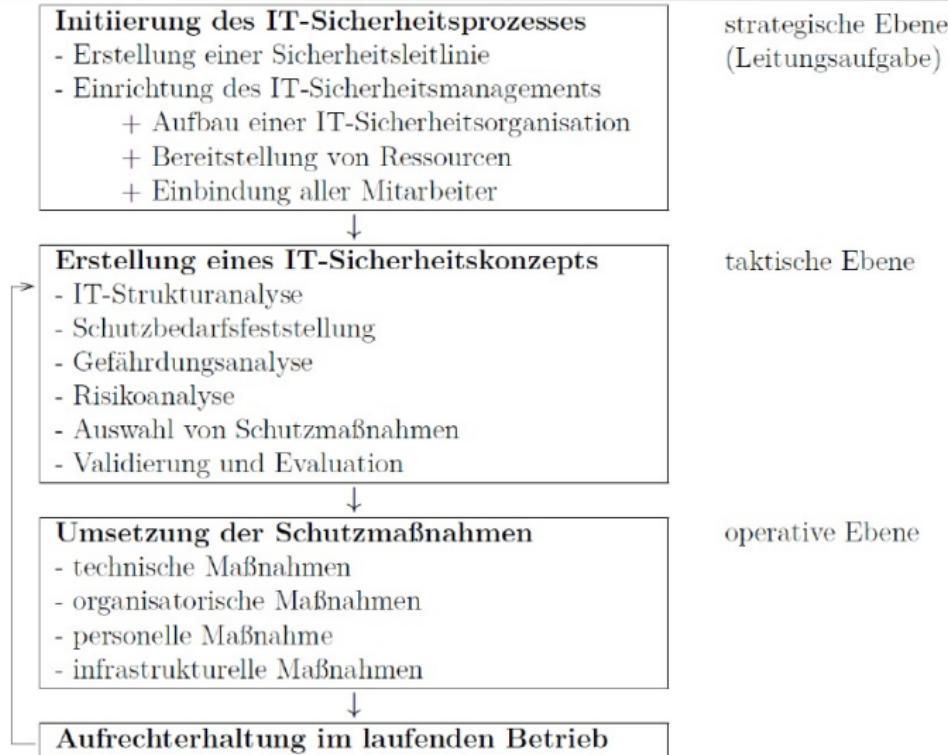
Ressourcenplanung, Technik, Sicherheitskonzept: Lernen Sie den IT-Grundschutz in unserer Schulung kennen.



IT-Grundschutz-Standards

Die BSI-Standards enthalten Empfehlungen zu Methoden, Prozessen, Vorgehensweisen und Maßnahmen zur Informationssicherheit.

Managementsysteme für Informationssicherheit des IT-Grundschutz



Erstellung eines Sicherheitskonzeptes

1. Strukturanalyse.
2. Schutzbedarfsfeststellung.
3. Auswahl und Anpassung von Maßnahmen.
4. Basis-Sicherheitschecks.
5. Ergänzende Sicherheitsanalyse.

IT-Strukturanalyse

- Erfassung der Räumlichkeiten, Netze, IT-Systeme und IT-Anwendungen
- Gruppenbildung



Schutzbedarfsfeststellung

normal ↙

↘ hoch, sehr hoch

IT-Grundschutzzanalyse

- Modellierung
- Auswahl von Maßnahmen
- Basis-Sicherheitscheck

Gefährdungsanalyse

- Gefährdungsübersicht
- zusätzliche Gefährdungen



Risikoanalyse

- Gefährdungsbewertung



Maßnahmen

- Auswahl von Maßnahmen
- Restrisikoanalyse

Realisierungsplanung

- Konsolidierung der Maßnahmen
- Umsetzungsplan

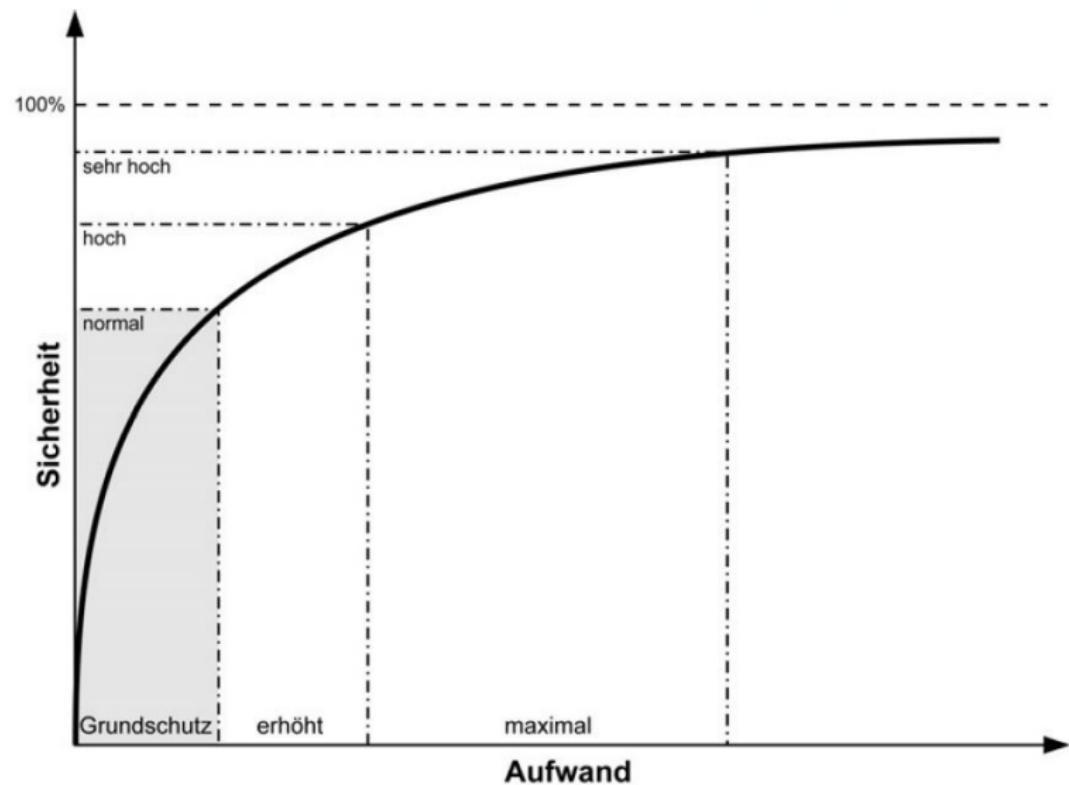


IT-Sicherheit – IT-Sicherheitsmanagement

Bei der Umsetzung der Maßnahmen zu beachten

- ▶ Gefundene Schutzmaßnahmen werden umgesetzt.
- ▶ Schutzmaßnahmen fortlaufend aufrechterhalten.
- ▶ Aufrechterhaltung erfordert
 - ▶ eine kontinuierliche Überwachung der Einhaltung der Schutzmaßnahmen.
 - ▶ eine Anpassung der Maßnahmen bei Sicherheitsvorfällen oder Änderungen der Bewertung.
- ▶ Wichtig
 - ▶ Ressourcen müssen vorhanden sein.
 - ▶ Zeit, Geld und Personal.
 - ▶ Aufwand gemeinsam mit Nutzen betrachten.
- ▶ Klare Verantwortlichkeiten.

Kosten/Nutzenbetrachtung



IT-Strukturanalyse

Ziel: Darstellung aller Bestandteile des IT-Verbundes und ihrer Beziehungen untereinander.

IT-Strukturanalyse

Ziel: Darstellung aller Bestandteile des IT-Verbundes und ihrer Beziehungen untereinander.

- ▶ Bestandteile:
 - ▶ Geschäftsprozesse (z.B. Personalverwaltung, Entgegennahme von Bestellungen),
 - ▶ Daten/Informationen (z.B. Personaldaten, Verträge, aber auch technische Informationen wie Konfigurationsdateien),
 - ▶ Anwendungen (z.B. Betriebssysteme, Office-, E-Mail-, Backup-Programme),
 - ▶ IT-Systeme (z.B. Computer, Server, Router, USB-Sticks, Smartphones),
 - ▶ Kommunikationsnetze (z.B. Intranet, Internet),
 - ▶ Räumlichkeiten (z.B. Büros, Standorte).

☞ Zur Verringerung der Komplexität der Strukturanalyse sollten ähnliche Objekte zu Gruppen zusammenfassen, z.B. Objekte die vom gleichen Typ sind oder ähnlich konfiguriert sind.

Güte der IT-Sicherheitsmaßnahmen

Stärke der eingesetzten IT-Sicherheitsmaßnahmen hängt von der Höhe des Schutzbedarfs der

- ▶ Geschäftsprozesse,
- ▶ Informationen,
- ▶ IT-Systeme,
- ▶ Kommunikationsnetze und
- ▶ Räumlichkeiten

im Hinblick auf die Ziele Vertraulichkeit, Integrität, Authentizität, Nichtabstrebbarkeit und Verfügbarkeit ab bzw.

Güte der IT-Sicherheitsmaßnahmen

Stärke der eingesetzten IT-Sicherheitsmaßnahmen hängt von der Höhe des Schutzbedarfs der

- ▶ Geschäftsprozesse,
- ▶ Informationen,
- ▶ IT-Systeme,
- ▶ Kommunikationsnetze und
- ▶ Räumlichkeiten

im Hinblick auf die Ziele Vertraulichkeit, Integrität, Authentizität, Nichtabstrebbarkeit und Verfügbarkeit ab bzw. genauer, von den Schäden, die bei Verlust der aufgeführten Ziele entstehen können.

Schadenszenarien

Verstoß gegen Gesetze/Vorschriften/Verträge.

- ▶ Beispiele für relevante Gesetze, Vorschriften und Verträge (D):
 - ▶ Gesetze: Grundgesetz, Bürgerliches Gesetzbuch.
 - ▶ Bundesdatenschutzgesetz und Datenschutzgesetze der Länder, Informations- und Kommunikationsdienstgesetz, Gesetz zur Kontrolle und Transparenz im Unternehmen.
 - ▶ Vorschriften: Verschluss Sachen Anweisung. Verwaltungsvorschriften, Verordnungen und Dienstvorschriften.
 - ▶ Verträge zur Wahrung von Betriebsgeheimnissen, Dienstleistungsverträge im Bereich Datenverarbeitung.

Schadenszenarien

- ▶ Beeinträchtigung des informationellen Selbstbestimmungsrechts
 - ▶ unbefugte Weitergabe personenbezogener Daten.
 - ▶ Unzulässige Erhebung personenbezogener Daten ohne Rechtsgrundlage.
- ▶ Beeinträchtigung der persönlichen Unversehrtheit Fehlfunktionen von IT-Systemen können unmittelbar zu gesundheitlichen Schäden (Verletzungen, Invalidität oder Tod von Personen) führen. Z.B. bei
 - ▶ medizinische Überwachungsrechner,
 - ▶ Flugkontrollrechner und Verkehrsleitsysteme.
- ▶ Beeinträchtigung der Aufgabenerfüllung durch Verlust der Ziele Verfügbarkeit oder Integrität von Daten.
Beispiele hierfür sind:
 - ▶ Fristversäumnisse durch verzögerte Bearbeitung von Verwaltungsvorgängen.

Schadenszenarien

- ▶ Negative Innen- oder Außenwirkung
 - ▶ durch den Verlust einer der Ziele Vertraulichkeit, Integrität oder Verfügbarkeit einer IT-Anwendung können verschiedenartige negative Innen- oder Außenwirkungen entstehen.
 - ▶ Beispiele:
 - ▶ Demoralisierung der Mitarbeiter.
 - ▶ Beeinträchtigung der wirtschaftlichen Beziehungen zusammenarbeitender Institutionen.
- ▶ Finanzielle Auswirkungen
 - ▶ Finanzielle Schäden können durch den Verlust der Vertraulichkeit schutzbedürftiger Daten, die Veränderung von Daten oder den Ausfall von IT-Anwendungen entstehen.
 - ▶ Beispiele:
 - ▶ Ausfall eines IT-gesteuerten Produktionssystems.
 - ▶ Weitergabe von Forschungs- und Entwicklungsergebnissen.

Schutzbedarfsanalyse

- ▶ Schutzbedarf der Informationen.
 - ▶ Ausgehend von der Strukturanalyse wird mittels der Schadens-Szenarien der Schutzbedarf der Daten bestimmt.
 - ▶ Schutzbedarf hinsichtlich Schutzzieilen (Vertraulichkeit, Integrität, Authentizität, Nichtabstreichbarkeit und Verfügbarkeit)
 - ▶ Einstufung jeweils begründen.
- ▶ Schutzbedarf der IT-Systeme (inklusive Netze)
 - ▶ richtet sich im Wesentlichen nach den Schutzbedarfen der Daten, die in diesen verarbeitet werden.
 - ▶ Dabei sollte ein möglicher Kumulationseffekt mit berücksichtigt werden. Beispiel: Werden auf einen Computer viele Daten der Schutzkategorie hoch verarbeitet, so sollte dieses System mit der Schutzkategorie sehr hoch bewertet werden.

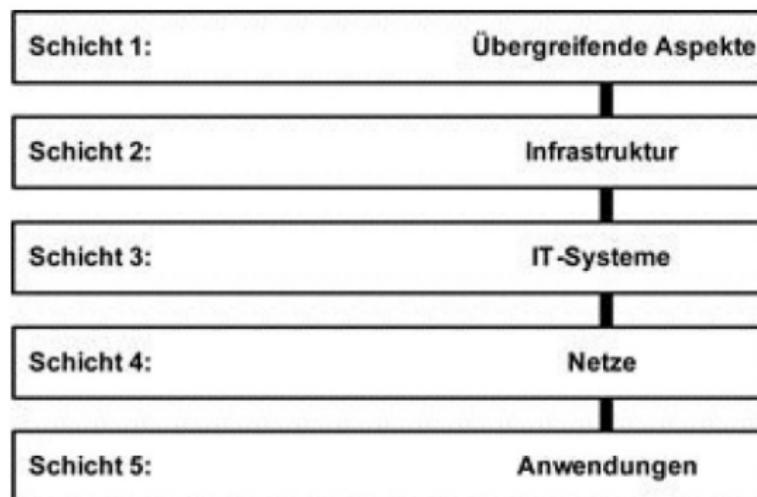
- ▶ Schutzbedarf der Räumlichkeiten:
 - ▶ Ausgehend von den Ergebnissen der Schutzbedarfsanalyse der IT-Systeme wird abgeleitet, welcher Schutzbedarf für die jeweiligen Liegenschaften bzw. Räume daraus resultiert.

► Schutzbedarf der Räumlichkeiten:

- Ausgehend von den Ergebnissen der Schutzbedarfsanalyse der IT-Systeme wird abgeleitet, welcher Schutzbedarf für die jeweiligen Liegenschaften bzw. Räume daraus resultiert.
Z.B. durch Betrachtung der im jeweiligen Raum installierten IT-Systeme, verarbeiteten Informationen oder beherbergten Datenträger, wie schon bei der Schutzbedarfsermittlung für IT-Systeme.
- Auch hier sollte zusätzlich ein möglicher Kumulationseffekt berücksichtigt werden, wenn sich in einem Raum eine größere Anzahl von IT- Systemen befindet, wie typischerweise bei Serverräumen.

Schichtenmodell

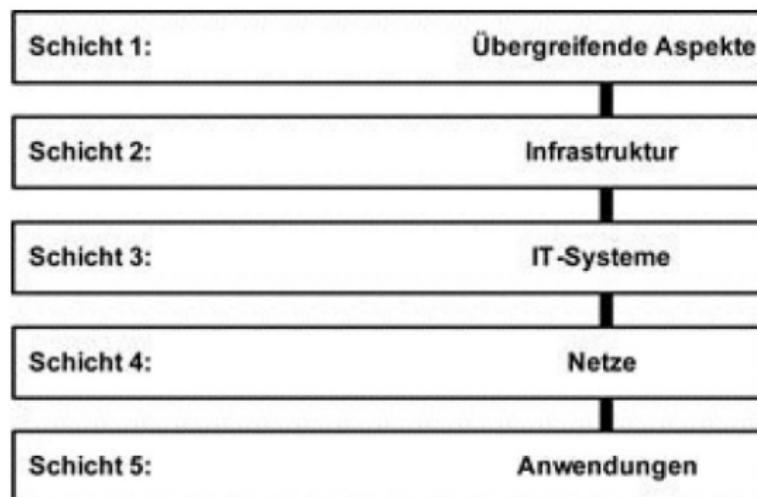
- ▶ IT-Verbund, IT-Systeme i.d.R. komplex
- ▶ Sicherheitsaspekte werden gruppiert und Schichten zugeordnet.
☞ Pauschalisiertes Sicherheitsniveau.



IT-Sicherheit – IT-Sicherheitsmanagement

Schichtenmodell

- ▶ IT-Verbund, IT-Systeme i.d.R. komplex
- ▶ Sicherheitsaspekte werden gruppiert und Schichten zugeordnet.
☞ Pauschalisiertes Sicherheitsniveau.



IT-Sicherheit – IT-Sicherheitsmanagement

Das Schichtenmodell

- ▶ Schicht 1 umfasst die übergreifenden Aspekte.
 - ▶ d.h. Aspekte, die sich auf den gesamten IT-Verbund oder große Teile hiervon beziehen (z.B. Geschäftsprozesse, Organisation des IT-Sicherheitsmanagementprozesses, Datensicherheitskonzept).
- ▶ Schicht 2 baulich-technischen Infrastruktur z.B. Gebäude, Büro- und Serverräume.
- ▶ Schicht 3 betrachtet die IT-Systeme, z.B. Client unter Mac OS X, Server unter Unix.
- ▶ Schicht 4 erfasst die Kommunikationsnetze, z.B. WLAN, heterogene Netze.
- ▶ Schicht 5 beschäftigt sich mit den Anwendungen, z.B. E-Mail, Datenbanken.

Schichtenmodell und Modellierung

- ▶ Ziel: alle Komponenten des IT-Verbunds als Bausteine zu beschreiben.
- ▶ Zu jeder Schicht gibt es im IT-Grundschutz geeignete Bausteine.

Zu jedem **Baustein** gibt es dann **Gefährdungskataloge** und **Maßnahmenkataloge**.

Beispiel: Baustein Gefährdung Maßnahme

- ▶ Baustein: 3.101 Allgemeiner Server
- ▶ Gefährdung: G.4.1 Ausfall der Stromversorgung
- ▶ Maßname: M 1.28 Lokale unterbrechungsfreie Stromversorgung

Auswahl und Anpassung von Maßnahmen

IT-Grundschutzkataloge bieten für jeden Baustein Gefährdungen und entsprechende Maßnahmen (inkl. Vorschläge für die Verantwortlichen der Maßnahme).

- ▶ Gefährdungskategorien
 - ▶ Elementare Gefährdungen
 - ▶ Höhere Gewalt
 - ▶ Organisatorische Mängel
 - ▶ Menschliche Fehlhandlungen
 - ▶ Technisches Versagen
 - ▶ Vorsätzliche Handlungen
- ▶ Schutzmaßnahmen-Kategorien
 - ▶ Infrastruktur
 - ▶ Organisation
 - ▶ Personal
 - ▶ Hardware und Software
 - ▶ Kommunikation
 - ▶ Notfallvorsorge

Beispiel: Baustein Server unter Unix (Auszug)

- ▶ Annahme: Firma betreibt einen Server mit dem Betriebssystem UNIX.
- ▶ IT-Grundschutz definiert Gefährdungen und Schutzmaßnahmen

- ▶ Organisatorische Mängel:
G 2.15 Vertraulichkeitsverlust schutzbedürftiger Daten im Unix-System
- ▶ Menschliche Fehlhandlungen:
G 3.10 Falsches Exportieren von Dateisystemen unter Unix
G 3.11 Fehlerhafte Konfiguration von sendmail
- ▶ Technisches Versagen
G 4.11 Fehlende Authentisierungsmöglichkeit zwischen NIS-Server und NIS-Client
G 4.12 Fehlende Authentisierungsmöglichkeit zwischen X-Server und X-Client
- ▶ Vorsätzliche Handlungen
G 5.41 Missbräuchliche Nutzung eines Unix-Systems mit Hilfe von UUCP
G 5.89 Hijacking von Netz-Verbindungen

Schutzmaßnahmen: Server unter Unix

Planung und Konzeption (Auszug):

- ▶ Die generelle Planung der Netzarchitektur wird im Baustein B 3.101 Allgemeiner Server festgelegt, in denen insbesondere die generelle Netzarchitektur und netzweite Regelungen festgelegt werden.
- ▶ Es ist sinnvoll, den Server in einem separaten Serverraum aufzustellen. Zu realisierende Maßnahmen sind im Baustein B 2.4 Serverraum beschrieben. Steht kein Serverraum zur Verfügung, sollte ein Serverschrank verwendet werden, vergleiche dazu den Baustein B 2.7 Schutzschränke .
- ▶ Es ist ein Verfahren für die Vergabe von Benutzerkennungen festzulegen, durch das gewährleistet wird, dass privilegierte und unprivilegierte Benutzerkennungen klar getrennt sind.
- ▶ Weiterhin ist sicherzustellen, dass kein unkontrollierter Zugang zum Single-User-Modus möglich ist, da sonst alle für die Laufzeit des Systems festgelegten Sicherheitsmaßnahmen unterlaufen werden können.

Umsetzung (Auszug):

- ▶ Bei der Konfigurierung eines Unix-Servers ist nach der Installation mit der Maßnahme M 4.105 „Erste Maßnahmen nach einer Unix-Standardinstallation“ zu beginnen.
- ▶ Hierbei sind, je nach Einsatzszenario (vergleiche B 3.101 Allgemeiner Server), Grundeinstellungen so vorzunehmen, dass nur benötigte Dienste aktiv sind bzw. die beschriebenen Vorkehrungen getroffen werden und die Systemprotokollierung aktiviert wird.

Betrieb (Auszug):

- ▶ Um die Sicherheit eines Servers unter Unix im laufenden Betrieb zuverlässig aufrecht zu erhalten, ist es unabdingbar, durch regelmäßige Überprüfungen festzustellen, ob irgendwelche Lücken aufgetreten sind, und diese so schnell wie möglich zu schließen. Dabei sind auch die vom System erzeugten Protokolle auf eventuelle Unregelmäßigkeiten hin zu betrachten.

Notfallvorsorge (Auszug):

- ▶ Da Unix-Systeme aufgrund ihrer Komplexität nach einem erfolgreichen Angriff oft auf schwer durchschaubare Weise kompromittiert sind, ist es wichtig, schon im Vorfeld Regeln festzulegen, nach denen bei einem echten oder vermuteten Verlust der Systemintegrität zu verfahren ist.
- ☞ Diese Maßnahmenempfehlungen werden durch ein Maßnahmenbündel für den Bereich „Server unter Unix“ mit Verlinkung zu den einzelnen Abschnitten in den Katalogen zur Unterstützung der konkreten Umsetzung begleitet.

Wesentliche Aspekte für den Erfolg dieser Methodik

- ▶ Wirksamkeit: Maßnahmen müssen vor den möglichen Gefährdungen wirksam schützen.
- ▶ Eignung: Maßnahmen müssen in der Praxis einsetzbar sein, d.h. keine Organisationsabläufe behindern oder andere Schutzmaßnahmen aushebeln.
- ▶ Praktikabilität: Maßnahmen sollten leicht verständlich, einfach anwendbar und wenig fehleranfällig sein.
- ▶ Akzeptanz: Maßnahmen sollten barrierefrei sein und niemanden diskriminieren.
- ▶ Wirtschaftlichkeit: Maßnahmen sollten das Risiko bestmöglich minimieren aber auch in einem geeigneten Verhältnis zu den zu schützenden Werten stehen.

Beispiel: Baustein Mobile Endgeräte (Auszug)

B 3.405 Smartphones, Tablets und PDAs:

Dieser Baustein beschäftigt sich mit mobilen Endgeräten zur Datenerfassung, -bearbeitung und -kommunikation. Diese gibt es in verschiedenen Geräteklassen, die sich nach Abmessungen und Leistungsmerkmalen unterscheiden. Dazu gehören unter anderem:

- ▶ Organizer, um Adressen und Termine zu verwalten.
- ▶ :

Gefährdungen: Auszug

- ▶ Höhere Gewalt:
 - ▶ G 1.15 Beeinträchtigung durch wechselnde Einsatzumgebung
- ▶ Organisatorische Mängel:
 - ▶ G 2.2 Unzureichende Kenntnis über Regelungen
 - ▶ G 2.4 Unzureichende Kontrolle der Sicherheitsmaßnahmen
 - ▶ ...

Maßnahmen (Auszug)

- ▶ Planung und Konzeption
 - ▶ M 2.218 (C) Regelung der Mitnahme von Datenträgern und IT-Komponenten.
 - ▶ M 2.303 (A) Festlegung einer Strategie für den Einsatz von Smartphones, Tablets oder PDAs.
- ▶ Beschaffung
 - ▶ :
- ▶ Umsetzung
 - ▶ :

Weitere Standards/Rahmenwerke:

- ▶ ISO 2700x Normenreihe (eng verzahnt mit IT-Grundschutz)
- ▶ ITIL (IT-Infrastructure Library)
- ▶ COBIT (Control Objectives for Information and Related Technology)