$J \int Skripte$

Kryptologie ICS.KRYPTO

Folien zur Präsenz 13, Teil 2
«Aktuelle Kennzahlen, Weiterentwicklungen sowie ein paar (repetitive) Kurzthemen», FS 24, V6.2

Der Dozent wird eine Auswahl der Themen treffen. Der Rest muss selbstständig nachbearbeitet werden!!

Die Schlüssellänge ist nicht das einzige Kriterium!!

Eingangsfrage:
Warum?
Finden Sie Antworten!!

©Josef Schuler, dipl. math., dipl. Ing. NDS ETHZ, MSc Applied IT-Security, Feldhof 25, 6300 Zug, j.schuler@bluewin.ch resp. josef.schuler@hslu.ch

Einleitung

Agenda

- Kennzahlen für heutige und zukünftige Anwendungen sowie weitere Entwicklungen, cf. Kap. 16 im JS Skript.
- Ein paar weitere Kurzthemen und Schlussgedanken
  - Wie gut ist ein Passwort? Repetition von Kap. 5.1 im JS Skript «Einführung in die Kryptologie».
  - Die kryptographische Stärke und ein Vergleich zu Passwörter, cf. Kap. 10.7 im JS Skript «Einführung in die Kryptologie».
  - Denkanstösse und Irrmeinungen zur Kryptographie.
  - Schlussdiskussion
  - Der Dozent wird eine ***Auswahl*** aus diesen Themen treffen!!
- ***Wichtig:*** In dieser Präsenz werden gewisse schon behandelte Aspekte nochmals angesprochen!
- Danksagung
  - Ich habe wiederum einige Folien von Prof. Dr. A. Steffen, HSR übernommen. An dieser Stelle nochmals herzlichen Dank.
  - Das Titelbild ist aus Folien von Prof. Dr. Rolf Oppliger, vielen Dank.

Verweise zur Literatur

- JS Skript „Einführung in die Kryptologie“
  - Kap. 5.1 Schlüssel und Passwort
  - Kap. 7.6.2 Verlauf beim Brechen des RSA, DH und EC
  - Kap. 8.2.4 Diskussion um die Reihenfolge von MAC und Verschlüsselung.
  - Kap. 10.7 Weitere Zahlenbeispiele (= Zahlen zur kryptographischen Stärke).
  - Kap. 11.1 Diskussion um die Sicherheit von EC.
  - Kap. 16 Kennzahlen und Weiterentwicklung.

Lernziele

- Ich kenne die aktuellen Kennzahlen und die weitere Entwicklung der Kryptographischen Algorithmen.
- Ich kenne weitere (Sicherheits-)Diskussionen.
- Ich kann die Stärke eines Passwortes und weitere Berechnungen zur kryptographischen Stärke berechnen.
- Ich kann Irrmeinungen der Kryptologie erkennen.

Die Slogan zu dieser Präsenz

Der Lumpensammler am Schluss versucht einen Bogen zu spannen.

Kap. 16

Kennzahlen für heutige und zukünftige Anwendungen sowie weitere Entwicklungen

Algorithmen und Schlüsselgrössen

- Algorithmen
  - Blockchiffren: AES
  - Lightweight Block Ciphers (wird in Chipkarten – besonders bei RFID – verwendet): PERSENT
  - Stromchiffren: keine Empfehlung, da sehr unterschiedliche Schlüsselgeneratoren. Z.B. Einige Modi bei den Blockchiffren sind so, dass die Blockchiffre «nur» für den Schlüsselstrom benötigt wird. Die Verschlüsselung ist eine Stromchiffre. Bekanntes aktuelles Beispiel ist der CTR Modus.
  - Hashfunktionen: SHA-2 oder SHA-3
  - Asymmetrische Verfahren: RSA, ECC, Elgamal
- Schlüsselgrössen
  - Blockchiffren wie AES, mind. 128 Bit, besser 192 resp. 256 Bit.
  - PERSENT: 128 Bit (es gibt auch eine 80 Bit Version)
  - Hashfunktionen wie SHA-2 o. SHA-3, mind. 256, besser 384 o. 512 Bit
  - Asymmetrische Verfahren
    - RSA oder disk. Log Verfahren wie Elgamal, mind. 2048, besser 3072 Bit
    - Elliptische Kurven, mind. 256, besser 384 o. 512 Bit

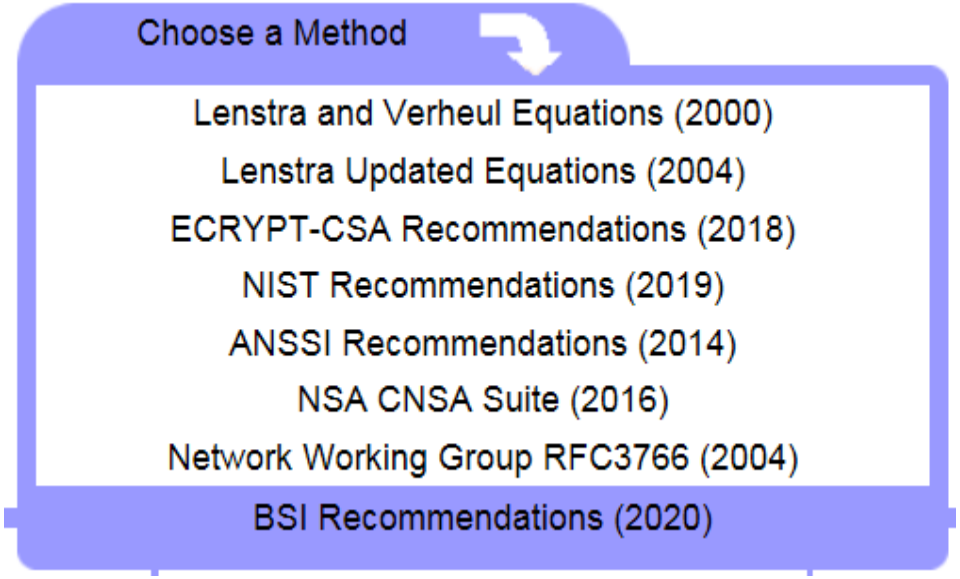
Wie viele Bits sind geknackt?

Bevor wir die Kennzahlen anschauen, an dieser Stelle nochmals kurz die Grössenordnung der Anzahl Bits beim RSA (Faktorisierungsproblem) und klassischem DH sowie Elliptischen Kurven (diskretes Logarithmen Problem). Siehe auch Kap. 7.6.2. in «Einführung in die Kryptologie» und [DB].

- RSA: Februar 2020, 829 Bit, resp. ca. 250-stellige Dezimalzahl.
- DH: Dezember 2019, 795 Bit, resp. ca. 240-stellige Dezimalzahl.
- EC: Juni 2020, 114 Bit, ca. 35-stellige Dezimalzahl.

www.keylength.com

In www.keylength.com sind verschiedene Empfehlungen aufgeführt, ausgewählt wurde BSI Recommendations (2020).



Date	Symmetric	Factoring Modulus	Discrete Logarithm Key	Discrete Logarithm Group	Elliptic Curve	Hash	
2020 - 2022	128	2000	250	2000	250	SHA-256 SHA-512/256 SHA-384 SHA-512	SHA3-256 SHA3-384 SHA3-512
2023 - 2026	128	3000	250	3000	250	SHA-256 SHA-512/256 SHA-384 SHA-512	SHA3-256 SHA3-384 SHA3-512

Zusammenfassung der Empfehlungen

Anbei eine möglichst einfache Empfehlung von Algorithmen und Schlüsselgrössen, die den unterschiedlichen Empfehlungen nicht widersprechen und höchstwahrscheinlich bis ins Jahr 2025 überdauern.

Beschreibung des krypt. Mechanismus	Empfehlung Algorithmus	Empfehlung der Schlüsselgrösse/ und Bemerkungen
Symmetrische Verschlüsselung	AES	Wenn möglich 256 Bit, 128 Bit sind aber nach wie vor OK
Symmetrische Verschlüsselung. Z.B. bei ePassport z.Z. ein MUSS.	3-DES	Mit 3 versch. Schlüssel noch bis 2030 erlaubt. In Bankapplikationen noch oft verwendet!!
Symmetrische Verschlüsselung in Massenprodukten → standard. Lightweight Blockcipher	PERSENT	128 Bit (gibt ihn auch mit nur 80 Bit)
CBC-MAC mit Zusatz im letzten Block	AES	Wenn möglich 256 Bit, 128 Bit sind aber nach wie vor OK
HMAC	SHA-2, SHA-3	Mindestens 256 Bit.
Asymmetrische Verschlüsselung Resp. Signatur	RSA	2048 Bit, ev. 3072 Bit und grösser
Asymmetrische Verschlüsselung Resp. Signatur	ECC	Mind. 256 Bit, ev. 384 o. 512 Bit
Hash für Signatur	SHA-2, SHA-3	Mindestens 256 Bit (384, 512).

Weitere Entwicklung, symmetrische Algo.

- AES ist so designed, dass in den nächsten (über) 20 Jahren kaum ein Bedarf an neuen Algorithmen besteht.
  - Der dem AES zugrunde liegende Algorithmus (Rijndael) ist voll parametrisierbar in Schlüssel- und Inputgrösse sowie Rundenzahl.
  - Bei prinzipiellen Schwächen (z.B. 256 Bit Schlüssel würden nicht mehr genügen), könnte man die Schlüsselgrösse „beliebig“ vergrössern.
    - Die Anpassung der Anwendungen wäre das viel grössere Problem, als die Änderungen im Algorithmus selber.
  - Ausnutzbare Schwächen des AES sind nicht bekannt. Obwohl schon einiges in „gutartige“ Kryptoanalyse investiert wurde. Es ist aber bekannt, dass der AES 256 «nur» eine Sicherheit von 224 Bit hat.
  - Z.Z. gilt: AES 256 ist auch sicher beim Einsatz von Quantencomputer (siehe auch weiter hinten).
- Komi Modi (Verschlüsseln und MAC’en) als Ersatz des Galois Counter Mode (GCM); er enthält Schwächen. Der Wettbewerb CAESAR (Competition for Authenticated Encryption: Security, Applicability and Robustness) ist beendet. Die Sieger heissen ACRON & AEGIS erkürt und werden nach und nach eingesetzt werden.

Weitere Entwicklung, asymmetrische Algo.

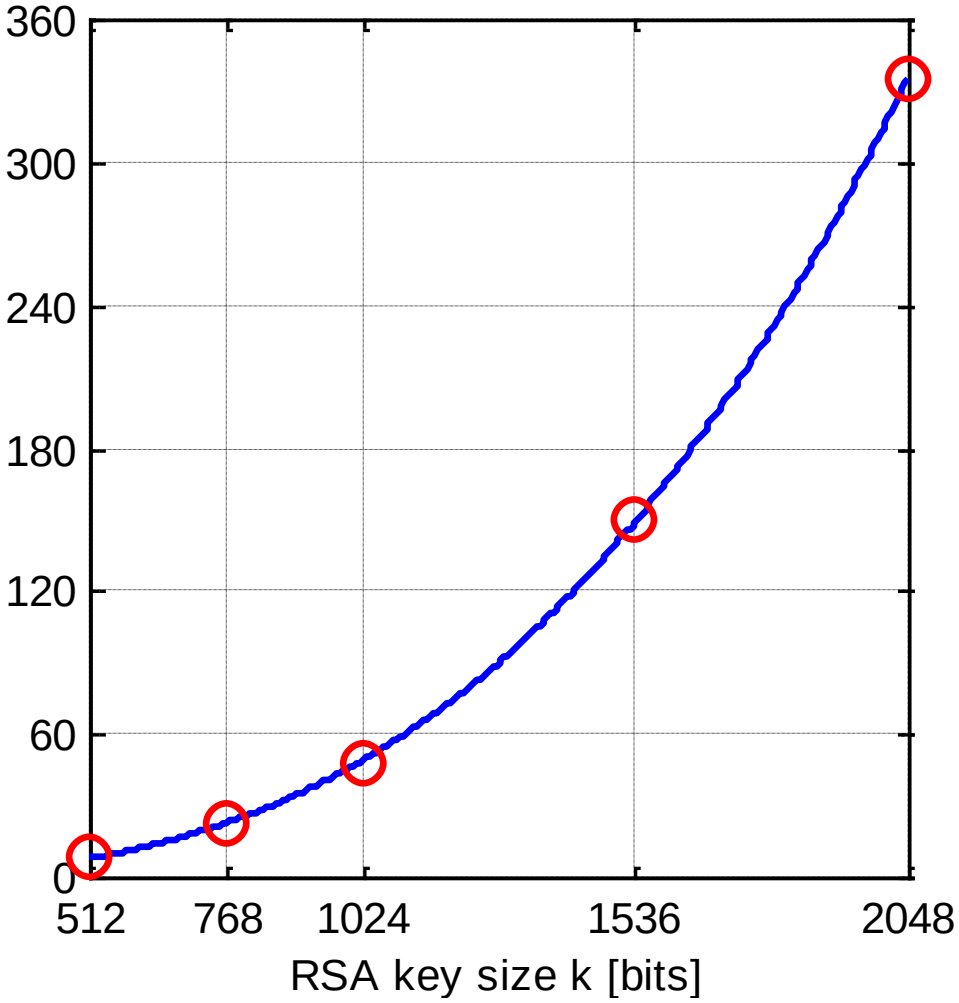
- ECC werden RSA ablösen
  - Die Schlüssellänge ist bei ECC nur 1/10 und weniger (cf. Nächste Folie) der Schlüssellänge bei RSA, bei gleicher Sicherheit.
  - Somit sind die Signaturen auch nur 1/10 so gross (Vorteil insbesondere für Zertifikate in Chipkarten).
  - Bei Verdoppelung der Schlüsselgrösse wird bei RSA der Rechenaufwand 6 – 8 Mal vergrössert (cf. übernächste Folie).
- Man rechnet, dass in 10 Jahren für die Signierung von archivierten Dokumenten bis 7kBit RSA gebraucht werden müssen.
- Sicherheitsdiskussionen werden zunehmen (siehe weiter hinten).
- Seit 2009 werden asymmetrische Algorithmen diskutiert, die resistent gegenüber Quantencomputer sind. Diese werden PQC = «Post Quantum Cryptography Algorithm» genannt. So um 2025 sollten die ersten Standards bereit sein.
  https://www.nist.pov/pqcrypto

Die Kryptografische Stärke im Vergleich (Quantencomputer nicht berücksichtigt!)

Symmetric	**56**	**80**	**112**	**128**	**192**	**256**
RSA n	512	1024	2048	3072	7680	15360
ECC p	112	160	224	256	384	512
Key size ratio	5:1	6:1	9:1	12:1	20:1	30:1

(*) Analoge Zahlen wie für den RSA gelten für alle diskreten Log.systeme wie Diffie-Hellman, Elgamal, Schnorr, Nyberg-Rueppel, DSA usw.

RSA Modular Exponentiation $y = x^e \bmod n$



RSA key size k [bits]	Processing time t [s]
512	8
768	22
1024	48
1536	150
2048	335

Weitere Entwicklung, Schlüsselverteilung

- Quantenkryptografie, besser Quantenschlüsselverteilung.
  - Es ist nun möglich zu entdecken, ob man abgehört worden ist.
  - Seit 2. Mai 2007 muss dies ev. relativiert werden. Es scheint so zu sein, dass es gelungen ist, bis zu 40% der Schlüsselbits abzuhören, ohne dass das entdeckt wurde.
    - Cf. www.heise.de/newsticker/meldung/print/89151
  - Im Bereich Teleportation sind ab 2011 neue Forschungsergebnisse herausgekommen. Diese könnten allenfalls dem Quantencomputer oder –verschlüsselung Auftrieb geben.
  - Aber Achtung, resp. ungeachtet allen Fortschritten:
    - Das Problem „der erste Schritt ist ein sicherer Schlüsselaustausch und/oder beim ersten Schritt muss eine Authentifizierung stattfinden“ ist damit nicht gelöst.
    - Dieses Problem ist z.Z. nicht gelöst, resp. ich wage zu sagen, nicht lösbar.
  - Es war aber so ca. in den letzten 5 Jahren – zumindest aus meiner Wahrnehmung – ziemlich ruhig gewesen in diesem Bereich.

Weitere Entwicklung, Hashfunktionen

- Der Wettbewerb zur Evaluierung einer neuen (besser: neuartigen) Hashfunktion ist (schon länger) abgeschlossen.
  - Sieger: SHA-3 = Keccak = SHS (siehe [CP-D], Kap. 11.3.1.9
  - Fundamental andere Struktur als SHA1- und SHA-2 Familie.
  - Gültige Hashgrössen und Kollisionsresistenz resp. Sicherheit wie:

224 Bit (**)	3DES	ECC-224 (***)	RSA-2048 (*)
256 Bit	AES-128	ECC-256	RSA-3096
384 Bit	AES-192	ECC-384	RSA-7680
512 Bit	AES-256	ECC-512	RSA-15360

  - SHA-2 Familie ist nach wie vor ohne Einschränkung einsetzbar.
  - (*) Analoge Zahlen gelten für diskrete Log.systeme wie Diffie-Hellman, Elgamal, Schnorr, Nyberg-Rueppel, DSA usw.
  - (**) Ab 2023 ist die Mindestgrösse des Hashes auf 240 Bit festgelegt. Damit ist der 224 Bit nicht mehr erlaubt und damit ist faktisch 256 Bit das Minimum.
  - (***) Mit ECC-224 kann nur ein 224 Bit Hash signiert werden. Wegen (**) wird damit ECC-224 ausgeschlossen.

Weitere (Sicherheits-)Diskussionen

- Sicherheitsdiskussionen werden zunehmen
  - Die Diskussion über die Reihenfolge von Integritätsschutz und Verschlüsselung. Siehe dazu Kap. 8.2.4 in «Einführung in die Kryptologie».
  - Die Diskussion über die Sicherheit und Verwendung von ECC im Allgemeinen und über die von NIST definierten ECC-Kurven im Speziellen. Siehe dazu Kap. 11.1 in «Einführung in die Kryptologie», resp. Kap. 2.3.6. in „Elliptische Kurven …“
  - Diskussion um „Cryptographic Engineering in a Post-Quantum World“.
- Einsatz von Künstlicher Intelligenz in der Kryptoanalyse. Das ist tatsächlich ein Thema, dem immer mehr Gewicht zuteil wird. Im deutschsprachigen Raum ist Werner Schindler vom Bundesamt für Sicherheit in der Informationstechnik (BSI) so der aktuelle Spezialist: https://omnisecure.berlin/wp-content/uploads/os20_Schindler_Werner.pdf

Aufgabe 1

Aufgabe 16.1
Sie müssen signierte Daten für eine lange Zeit archivieren (mehrere Jahrzehnte). Was müssen Sie in kryptologischer Hinsicht beachten, um sicher zu sein?

Aufgabe 16.2
Welche Schlüssellängen genügen voraussichtlich in den nächsten 10 Jahren den höchsten Sicherheitsansprüchen bei der symmetrischen Verschlüsselung:

☐ 40-56 Bit
☐ 64-80 Bit
☐ 128-256 Bit
☐ 512-1024 Bit
☐ 1024-2048 Bit
☐ Keine der Antworten ist richtig.

Aufgabe 16.3
a) Welcher der folgenden Schlüssel besitzt die grösste kryptographische Stärke?

☐ 120 Bit ECC
☐ 240 Bit ECC
☐ 1024 Bit RSA
☐ 2048 Bit RSA
☐ ECC und RSA können nicht miteinander verglichen werden, deshalb ist keine eindeutige Angabe möglich.

b) Begründen Sie Ihren Entscheid

Aufgabe 2

- In einem Security Journal wurde publiziert, dass ein 112 Bit ECC geknackt wurde. Man kann nun davon ausgehen, dass die 112 Bit ECC „unsicher“ sind. (Cf. https://bluray-disc.de/blu-ray-news/playstation3/8002-playstation_3_schweizer_forscher_knacken_112bitverschluesselung_mit_200_playstation_3_konsolen?cpage=2

a) Stellen Sie diesen 112 Bit ECC in Relation zu einem symmetrischen Verfahren.

b) Stellen Sie diesen 112 Bit ECC in Relation zum RSA.

c) Im Weiteren wurde erwähnt, dass “… the effort is equivalent to about 14 full 56-bit DES key searches”.
Kommentieren und beurteilen Sie das in wenigen Sätzen.

d) Ihr Chef kommt nun zu Ihnen und will wissen, wie schlimm diese Meldung ist. Erörtern Sie das in max. 2 – 3 Sätzen.

Repetition Kap. 5.1

Wie gut ist ein Passwort?

Entropy of the English Language

- Single character statistics
  - Entropy H = 4 bits / character
- Written English taking into account the full context
  - Shannon (1950): Entropy H = 0.6 ... 1.3 bits / character
  - Simulations (1999): Entropy H = 1.1 bits / character
- Compression before encryption increases security
  - Good data compression algorithms (e.g. Lempel-Ziv) remove all redundancy and come very close to the entropy of the plaintext.
  - Compression is with algorithms like AES not longer necessary.

Passwort resp. Schlüssel der Grösse von 128 Bit

Passwort und Schlüssel sind im Wesentlichen synonym. Ein kryptographischer Schlüssel wird aber in aller Regel in HEX-Zeichen dargestellt, ein Passwort oft in anderen Codierungen.

Codierungen, die Anzahl Bit pro Zeichen und je ein Beispiel

1. Digits (0...9): 39 digits ∗ 3.32 bits/digits (*)
   39475 10485 98021 43380 05872 49759 70291 2634

2. Hexadecimal (0...F): 32 nibbles ∗ 4 bits/nibble (**)
   3F8A 84D1 EA7B 5092 C64F 8EA6 73BD F01B

3. Alphabet (A...Z): 28 characters ∗ 4.64 bits/character (*)
   AWORH GHJBP IUCMX MLZFQ TZDOP ZJV

4. Alphabet & Digits (A...Z, 0...9): 25 symbols ∗ 5.12 bits/symbol (*)
   E5RGL UPQ7A 8F3ZP NWTIC 22JBM

5. Base64 (A...Z, a...z, 0...9, /, +): 22 symbols ∗ 6 bits/symbol (*)
   y5GNa Riq92 VCm4Q 1BOKl x0

(*) Der Wert ist leicht über 128 Bit.
(**) Nibble = Halbbyte (in der HEX-Codierung)

Beispiel:

Für die Berechnung, dass die 10 Digits 3,32 Bit Information brauchen, muss die Gleichung $2^x = 10$ gelöst werden. Die Lösung lautet: $x = \log_2 10 = \frac{lg10}{lg2} = 3{,}32$. Und somit braucht es für 128 Bit $\frac{128}{3{,}32} = 38{,}55$, also 39 Zeichen.

Aufgabe 3

a) Wie gross ist die Entropie (= kryptographische Stärke) in Bits einer 6-stelligen PIN, die Sie auf einer Zahlentastatur am Bancomaten eingeben?

b) Wie gross ist die kryptografische Stärke des hexadezimalen Passworts EB832A10B5A8221D6E7E gemessen in Bit?

c) Schreiben Sie ein Passwort hin, das eine kryptografische Stärke von ca. 120 Bits besitzt und erklären Sie kurz wie Sie es gebildet haben:

d) Welche kryptographische Stärke in Bit hat in etwa folgendes, in Anführungszeichen gesetztes Passwort bestehend aus 20 Base64 Zeichen: „6+R2z7BOQ4GW5TJxhF14“

Kap. 10.6

Zahlen zur kryptographischen Stärke

Ein paar Zahlen und Annahmen

- Zahlen
  - Der Planet Erde hat 10^{51} Atome
  - Der Mensch besteht aus 10^{28} Atomen
  - Ein Jahr hat $3{,}3 \cdot 10^{7}$ Sekunden
  - Der Schlüsselraum eines 256 Bit Schlüssel beträgt $2^{256} \approx 10^{77}$
  - Wie berechnen? $1000 = 10^{3} \approx 2^{10}$, und dann die Potenzgesetze anwenden.

- Computerannahmen
  - Ein 1 TeraHz Computer macht 10^{12} Taktzyklen/sec.
  - Resp. $10^{12} \cdot 3{,}3 \cdot 10^{7} = 3{,}3 \cdot 10^{19}$ Taktzyklen im Jahr
  - Eine 6,5 TeraByte HD kann 100 Mia resp. 10^{11} Schlüssel à 256 Bit speichern.

Brute-Force Attacken von 256 Bit

- Annahmen und erste Berechnungen
  - Jedes der 10^{51} Atome der Erde bestünde aus einem 1 T Hz PC
  - Pro Takt kann ein Schlüssel ausprobiert werden.
  - Dieser Riesencluster würde demnach $10^{51} \cdot 10^{12} = 10^{63}$ Taktzyklen/sec resp. $10^{63} \cdot 3{,}3 \cdot 10^{7} = 3{,}3 \cdot 10^{70}$ Takte/Jahr machen.

- Vollständige Schlüsselsuche von 256 Bit
  - Für die 10^{77} Schlüssel bräuchte es demnach $\frac{10^{77}}{3{,}3 \cdot 10^{70}} = 3{,}3 \cdot 10^{6}$ Jahre, d.h. 3 Mio. Jahre.

- Speicherung aller 256 Bit Schlüssel
  - Jedes Atom müsste $\frac{10^{77}}{10^{51}} = 10^{26}$ Schlüssel tragen. D.h. jeder Mensch würde ca. $10^{26} \cdot 10^{28} = 10^{54}$ Schlüssel tragen. M.a.W. 1000-mal so viele Schlüssel wie die Erde Atome hat!!

Folgerung

- Brute Force Attacken sind keinen Gedanken wert!!!!
- Geschweige denn einer Argumentation $\rightarrow$ obwohl das immer wieder gemacht wird.
- Die Grösse einer Brute Force Attacke ergibt nicht den kleinsten Hinweis auf die Sicherheit eines Algorithmus.
  - Beispiel 104 Bit RC4.
    - Mit Brute Force absolut keine Chance.
    - Kann trotzdem mit Home Equipment geknackt werden.

Warum in aller Welt ein 256 Bit AES?

- Weil man vermutet, dass es kombinierte Attacken gibt (sog. Time-Memory-Tradeoff) die nur 1/3 der Bit brauchen. Genauer: Sie brauchen nur die 3-te Wurzel des Aufwandes.
- Ansätze (Merkle-Hellman) die nur 2/3 der Bits brauchen gibt es schon 20 Jahre. Die Erfolgswahrscheinlichkeit ist ca. 70% [FU].
- Im Weiteren vermutet man, dass Blockchiffren robuster gegenüber Attacken mit Quantencomputer als es asymmetrische Verfahren sind. Man schätzt, dass eine mit Quantencomputer eine vollständige Schlüsselsuche nicht mit 2^n sondern nur mit $2^{n/2} = \sqrt{2^n}$ durchgeführt werden kann. Der Aufwand von $2^{256/2} = \sqrt{2^{256}} = 2^{128}$ wäre also auch für Quantencomputer nicht durchführbar (cf. [CP-D], S. 104 & S. 166).

Also rechnen wir mit 256/3 = ca. 85

- Bedingungen
  - $2^{85} = 10^{26}$
  - 10 Mia ($= 10^{10}$) Menschen und jeder besitzt ein 1 T Hz PC.
  - Pro Schlüssel 1000 Taktzyklen („realistischer“ als 1 Zyklus)
- Vollständige Schlüsselsuche
  - $\frac{10^{26}}{10^{10} \cdot 10^{9}} = 10^{7}$ Sekunden = 4 Monate.
- Vollständige Speicherung
  - Pro PC wiederum 10^{11} Schlüssel, somit könnten die 10^{10} PC‘s nur 10^{21} Schlüssel speichern. Es bräuchte aber 10^{5} oder 100‘000-mal mehr Speicherplatz.

- Übersicht für 256 Bit AES

Brute-Force	**Merkle-Hellman mit $k^{2/3}$**	**Mit Quantencomputer mit $\sqrt{k} = k^{1/2}$**	**Merkle-Hellman vermutet mit $\sqrt[3]{k} = k^{1/3}$**
2^{256}	$(2^{256})^{2/3} \approx 2^{170}$	$(2^{256})^{1/2} = 2^{128}$	$(2^{256})^{1/3} \approx 2^{85}$

Wie sieht das bei 128 Bit Schlüsseln aus?

- Brute-Force
  - Keine Chance, da $2^{128} = 2^{43} \cdot 2^{85} = 10^{13} \cdot 2^{85}$
  - 10^{13} = 10‘000 Milliarden
  - Also 10‘000 Milliarden mal so viel Aufwand wie in der vorherigen Folie aufgezeigt wurde.
- Stärke bei Ansatz von Merkle-Hellman $\sqrt[3]{k^2} = k^{2/3}$
- Stärke beim Einsatz von Quantencomputer $\sqrt{k} = k^{1/2}$
- Stärke bei „worst-case“ $\sqrt[3]{k} = k^{1/3}$
  - Sollte das wirklich wahr werden, dann müsste man sehr schnell reagieren. Denn 2^{43} ist in einem Hackerverbund zu bewältigen. Das ist ca. die Grösse, die vermutet wird, dass grosse Geheimdienste in «Echtzeit» an Rechnerkapazität haben.

Aufgabe 4: Füllen Sie die Tabelle für k = 128 Bit aus.

Brute-Force	**Merkle-Hellman mit $k^{2/3}$**	**Mit Quantencomputer mit $\sqrt{k} = k^{1/2}$**	**Merkle-Hellman vermutet mit $\sqrt[3]{k} = k^{1/3}$**

Nachbearbeitung: Repetieren Sie das Kap. 10.5 „Time-Memory Tradeoff und lösen Sie bitte die Aufgabe 16.1 im Kap. 16.8.1 im JS Skript «Einführung in die Kryptologie».

Der Vergleich zum Passwort

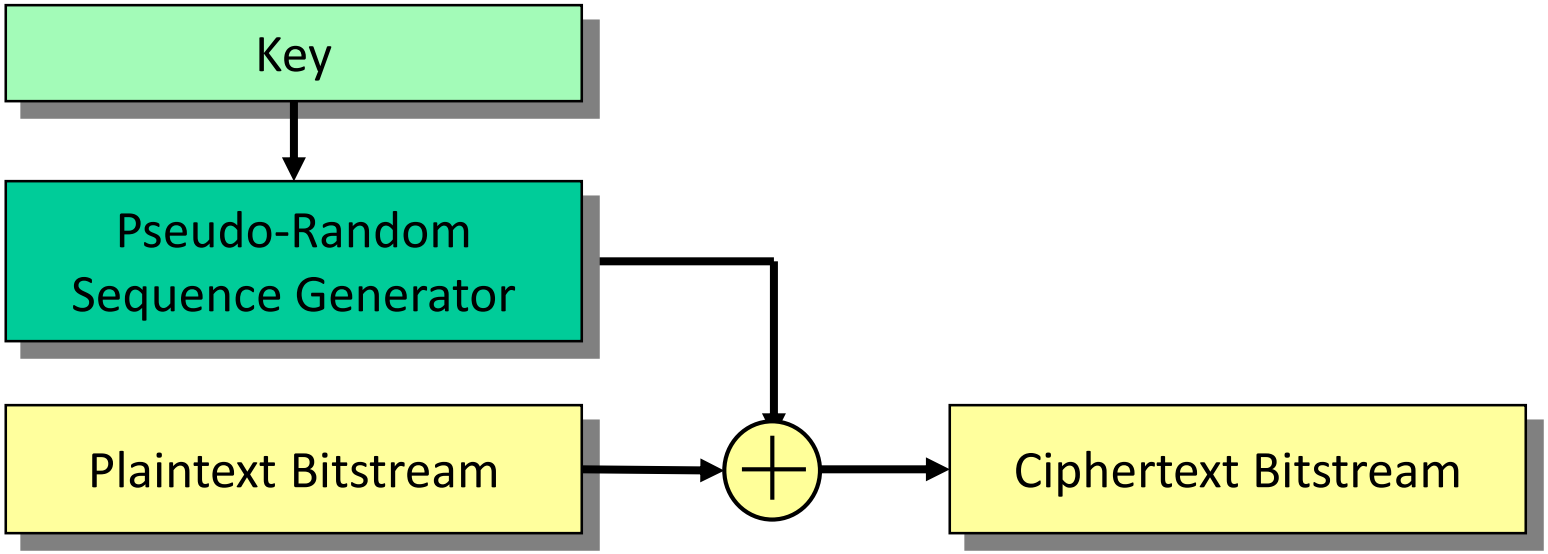
- Ein 12-stelliges Passwort mit allen Sonderzeichen, Gross- und Kleinbuchstaben entspricht 72 Bit.
- D.h. dieses Passwort ist...
  - ... $\frac{2^{256}}{2^{72}} = 2^{184} = 10^{55}$ mal schwächer als ein 256 Bit Schlüssel gegenüber einer Brute-Force. (Im Gegensatz zu krypt. Schlüssel sind Brute-Force Attacken gegen Passwörter gang und gäbe).
- Ein bisschen „humaner“: Dieses Passwort ist...
  - ... $\frac{2^{128}}{2^{72}} = 2^{56} \approx 10^{17}$ mal schwächer als ein 128 Bit Schlüssel.
  - Resp. das Verhältnis ist 1 Bit zu einem 56 Bit DES Schlüssel.
- Nur: wer merkt sich schon ein solches Passwort?
- Beachten Sie nun meine Aussagen in der PR 12 zu sogenannten «starken Authentisierverfahren».

Denkanstösse und Irrmeinungen zur Kryptographie

Verschlüsselung und Integrität

- Mit Verschlüsselung kann i.a. keine Integrität gewährleistet werden.
  - Beispiel: Stromchiffre
  - Nicht Mechanismen und Einsatz vermischen
- Also entweder Integritätsschutzmechanismen HAMC, C-MAC usw. …
- ... oder integrierte Modi verwenden
  - Z.Z. ist GCM (Galois Counter Mode) der am meist verbreitete Modus.
  - Wie weiter vorne schon erwähnt, wird in nächster Zeit ein neu zu bestimmender Kombi Modus standardisiert.
  - Hie und da ging es bei den Combi Modi auch schief: PCBC (cf. Bruce Schneier) $\rightarrow$ z.B. in Kerberos Version 4 implementiert $\rightarrow$ Version 5 ging wieder auf CBC zurück, nachdem eine Schwäche im PCBC Mode entdeckt wurde.
  - Wie schnell sich die neuen Modi – ACRON & AEGIS – durchsetzen werden, wird sich noch zeigen.

Stromchiffrierer

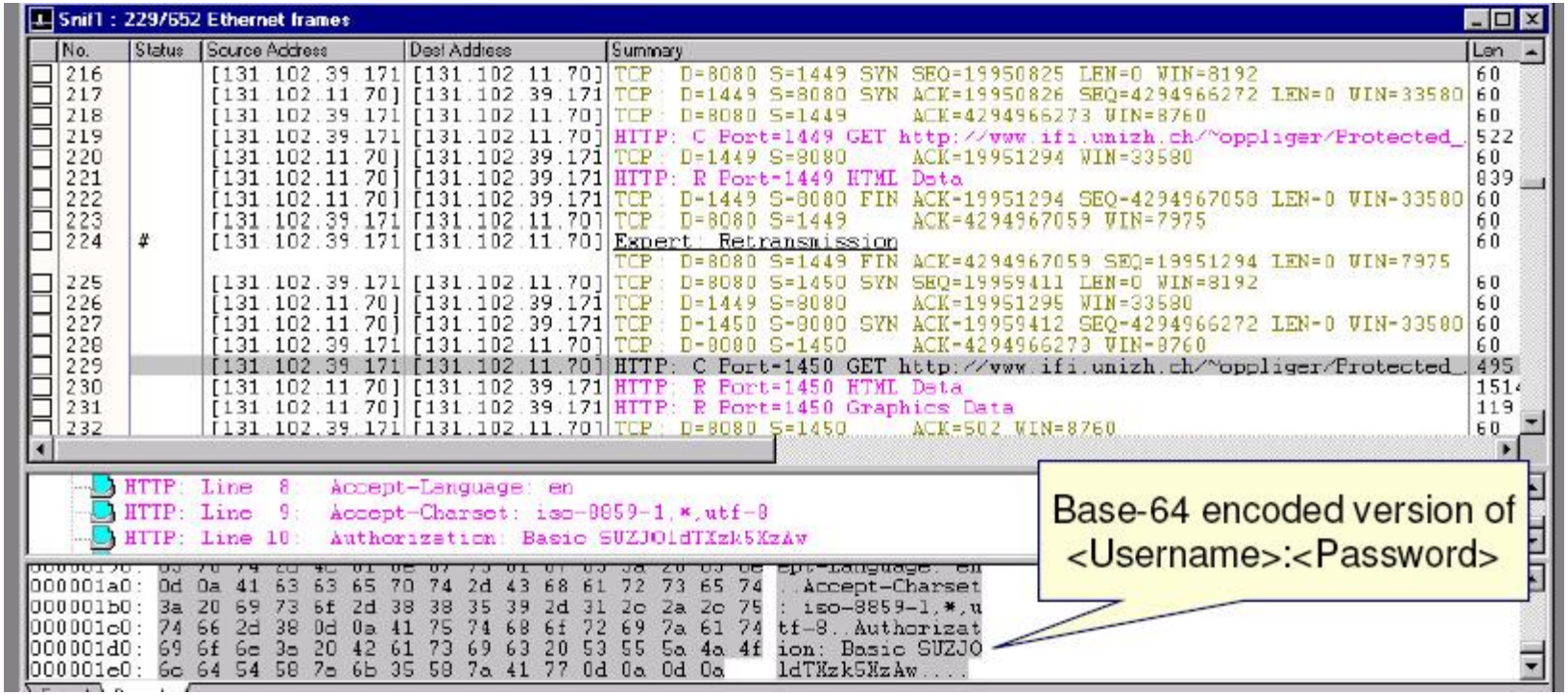


Plaintext Stream	**1 1 1 1 1 1 1 1 0 0 0 0 0 0 …**
Pseudo-Random Stream	**1 0 0 1 1 0 1 0 1 1 0 1 0 0 …**
Ciphertext Stream	**0 1 1 0 0 1 0 1 1 1 0 1 0 0 …**

RSA

- RSA-Verschlüsselung: $C = m^e \; mod \; N$ & $m = C^d \; mod \; N$ (e, N) sind der öffentliche Schlüssel (z. B. e = 23, N = 55)
  - d ist der geheime Exponent → in diesem Fall ist d = 7
- Berechnungen:
  - Wählen Sie nun ein m zwischen 2 und 9, z.B. m = 9.
  - Berechnen Sie $C = 9^{23} \; mod \; 55 = 14.$
  - Multiplizieren Sie nun in der Rolle von Eve das Chiffrat mit einer Zahl, z.B. mit $2^{23} \; mod \; 55 = 8.$
  - Also $C' = 8 \cdot C \; mod \; 55 = 8 \cdot 14 \; mod \; 55 = 2.$
  - Was erhält nun der Empfänger? ***Behauptung:*** Er erhält $2 \cdot 9 = 18.$
  - Beweis: $2^7 \; mod \; 55 = 18.$
- Also: RSA hat keine Integrität.

The first step is the most important one



- **Wenn man so anfängt (Username und Password unverschlüsselt – nur Base64 codiert – mitschicken), nützen nachher die stärksten Krypto-algorithmen nichts mehr!**
- **Das Titelbild lässt grüssen!**
- **Alter Hut, das war einmal → leider NEIN, z.B. WhatsApp macht etwas Ähnliches. Man geht einfach davon aus, dass die erste Meldung nicht ab. gehört wird. Danach arbeitet man dann mit starken Kryptoalgorithmen.**

Das Krypto-Puzzle

Das Krypto-Puzzle nochmals anschauen

Symmetrisch
(Abhören verhindern)
Algorithmen: DES, IDEA, AES usw. in versch. Modi)

Symmetrisch
(Insertion verhindern/ Integrität gewähren)
Algorithmen: MAC/MIC (DES, IDEA, AES usw. in speziellen Modi)

Vertraulichkeit, Geheimhaltung	**Authentizität/ Integrität**
Mechanismus: **Verschlüsselung**	Mechanismus: **Anhängen eines „Authentizierwertes“**

Asymmetrisch
(Abhören verhindern)
Algorithmen: RSA u.a.
Vorteil: Vereinfachung des Key-Managements
Verschlüsseln mit dem Public Key des Empfängers
Entschlüsseln mit dem Privat Key des Empfängers

Asymmetrisch
(Insertion verhindern/ Integrität gewähren/ Non repudiation of origin ⇨ digitale Signatur)
Algorithmen: RSA, DSA u.a. ⇨ es gibt auch reine Signierer (z.B. DSA), welche nicht verschlüsseln können. Solche reinen Signierer heissen Signaturalgorithmen ohne message recovery

Vorteil: Vereinfachung des Key-Managements

Signieren mit Secret Key des Senders
Verifizieren mit Public Key des Senders

Wichtig:

- Die Integrität ist eine Teilmenge der Authentizität.
- Mit Verschlüsselung alleine kann keine absolute Integrität/Authentizität erreicht werden.
- MAC bietet lediglich Integrität, jedoch keine Authentizität.

Nachschlagewerke & Links

Nachschlagewerke & Links for future life

Sollten Sie einmal (z.B. im Berufsumfeld) etwas Spezifisches in Bezug auf Kryptologie suchen, dann helfen Ihnen ggf. die folgenden Angaben etwas.

Schlüssellängen: https://www.keylength.com/

Vorgaben vom BSI (Bundesamt für Sicherheit in IT, DE):
https://www.bsi.bund.de/DE/Themen/Unternehmen-und-Organisationen/Standards-und-Zertifizierung/Technische-Richtlinien/TR-nach-Thema-sortiert/tr02102/tr02102_node.html
Dieses Paper ist in Ilias hochgeladen, es hat aber noch div. andere auf der Homepage.

ENSI (European Union Agency for Network and Information Security:
https://www.enisa.europa.eu

European Payments Council: https://ww.epc-cep.eu
Das Paper **"Guidelines on cryptographic algorithms usage and key management**" ist in Ilias hochgeladen.

Diverse Standards zur Kryptologie mit Fokus auf Zahlsysteme:
Eine Zusammenstellung solcher Standards ist im oben erwähnten Paper **"Guidelines on cryptographic algorithms usage and key management"** im Annex II, ab p. 64 enthalten.

Schlussdiskussion

Diskutieren Sie untereinander im Forum…

Ein CH-Sicherheitsexperte hat in TagesAnzeiger vom 30.01.2012 u.a. den folgenden Ratschlag zur Verschlüsselung von Bankdaten formuliert:

„… die Bankdaten sollten in einzelne Gruppen zusammengefasst sein, die jede mit einem anderen Verfahren und einem unterschiedlichen digitalen Schlüssel verschlüsselt werden. Diese verschlüsselten Gruppen sollten dann komprimiert und nochmals verschlüsselt werden, wiederum mit einem anderen Verschlüsselungsverfahren und einem weiteren digitalen Schlüssel…“.
Kommentieren Sie diese Ratschläge.

Das aus dem TA Archiv herausgeholte Dokument (Zitat oben = grün unterstrichener Text im Original) ist ebenfalls hochgeladen.

Basis-Test PR 13_2

Aussage	Richtig oder falsch?	Begründung
Die Schlüssellänge eines Algorithmus ist das einzige wirkliche Kriterium für die Sicherheit eines Algorithmus.	☐ richtig ☒ falsch	
Z.B. unter www.keylength.com können aktuelle Empfehlungen zur Schlüssellänge nachgeschlagen werden.	☒ richtig ☐ falsch	
In der nächsten Dekade werden hauptsächlich Weiterentwicklungen in Post-Quanten-Algorithmen und Kombi-Mode (Verschlüsseln und MAC'en) geschehen.	☒ richtig ☐ falsch	
In der „Kryptoszene" sind sich die Experten immer einig.	☐ richtig ☒ falsch	
Der Einsatz von Künstlicher Intelligenz ist noch kein Thema in „Kryptoszene".	☐ richtig ☒ falsch	
Es gibt in etwa gleich viele 256 Bit Schlüssel wie Atome im Weltall.	☒ richtig ☒ falsch	
Zum Glück gibt es keine Irrmeinungen zur Kryptographie.	☐ richtig ☒ falsch	
Die Ratschläge des CH-Sicherheitsexperte müssen nicht diskutiert werden, es sind sehr gute Ratschläge.	☐ richtig ☒ falsch	

Lösungen

Lösung der Eingangsfrage

- Side-Channel Attacken
- "schlecht" geschützte Schlüssel (2048 RSA Secret key mit 6 stelligem PW geschützt)
- "schlecht" geschützte Schlüssel und Algorithmen --> Kryptoalgorithmen laufen im PC und nicht im Krypto HW (Security Modul)
- Schlechter (erster Schlüsselaustausch) z.B. TOFU (Trust of the first use) oder ftp Protokoll
- Falsches/schlechtes Protokoll
- Angriff nicht auf der Linie (klassisches Alice und Bob) Model
- Weitere Antworten sind möglich.

Aufgabe 1

Aufgabe 16.1

- Es muss eine genügende Schlüssellänge des Signierers gewählt werden (z.B. Bei RSA 3072 Bit, ECC mit 512 Bit)
- Es muss ein guter Hash verwendet werden z.B. SHA-512

Aufgabe 16.2

☐ 40-56 Bit
☐ 64-80 Bit
☑ 128-256 Bit
☐ 512-1024 Bit
☐ 1024-2048 Bit
☐ Keine der Antworten ist richtig.

Aufgabe 16.3

a) Welcher der folgenden Schlüssel besitzt die grösste kryptographische Stärke?

☐ 120 Bit ECC
☑ 240 Bit ECC
☐ 1024 Bit RSA
☐ 2048 Bit RSA
☐ ECC und RSA können nicht miteinander verglichen werden, deshalb ist keine eindeutige Angabe möglich.

b) Begründen Sie Ihren Entscheid.

Lösung: 2048 Bit RSA entsprechen ca. 224 Bit ECC.

Aufgabe 2

a) Das entspricht in etwa der Stärke eines 56 Bit (z.B. Single-DES) symmetrischen Verfahrens.

b) Das entspricht in etwa der Stärke eines 512 Bit RSA.

c) Die Stärke des 112 Bit ECC ist ca. 56 Bit, das Brechen braucht aber immer noch 14 mal so viel wie die vollständige Schlüsselsuche eines Single-DES. Somit ist die Sicherheit des 112 Bit ECC im Wesentlich immer noch gleich gut (resp. besser) als sein symmetrisches Äquivalent.

d) Aus c) ist ersichtlich, dass das nicht so schlimm ist, resp. dass man dies in etwa erwarten muss (ansonsten wäre der 112 Bit ECC klar stärker als ein 56 Bit DES). Wichtig ist, dass man die entsprechenden ECC nimmt, also mit 256 und mehr Bit.

Aufgabe 3

a) 6*3,3 Bit = 20 Bit

b) 20*4 Bit = 80 Bit

c) Keine Musterlösung

d) 20*6 Bit = 120 Bit

Aufgabe 4

Brute-Force	**Merkle-Hellman mit $k^{2/3}$**	**Mit Quantencomputer mit $\sqrt{k} = k^{1/2}$**	**Merkle-Hellman vermutet mit $\sqrt[3]{k} = k^{1/3}$**
2^{128}	$(2^{128})^{2/3} \approx 2^{85}$	$(2^{128})^{1/2} = 2^{64}$	$(2^{128})^{1/3} \approx 2^{43}$

Basis-Test PR 13_2

Aussage	Richtig oder falsch?	Begründung
Die Schlüssellänge eines Algorithmus ist das einzige wirkliche Kriterium für die Sicherheit eines Algorithmus.	☐ richtig ☑ falsch	Siehe Lösungen der Titelfrage in den Folien.
Z.B. unter www.keylength.com können aktuelle Empfehlungen zur Schlüssellänge nachgeschlagen werden.	☑ richtig ☐ falsch	
In der nächsten Dekade werden hauptsächlich Weiterentwicklungen in Post-Quanten-Algorithmen und Kombi-Mode (Verschlüsseln und MAC'en) geschehen.	☑ richtig ☐ falsch	
In der „Kryptoszene" sind sich die Experten immer einig.	☐ richtig ☑ falsch	Cf. Diskussion um ECC oder Reihenfolge MAC'en und Verschlüsseln.
Der Einsatz von Künstlicher Intelligenz ist noch kein Thema in „Kryptoszene".	☐ richtig ☑ falsch	Das Thema wird je länger je wichtiger, cf. Werner Schindler vom BSI.
Es gibt in etwa gleich viele 256 Bit Schlüssel wie Atome im Weltall.	☑ richtig ☐ falsch	$2^{256} \approx 10^{77}$, man schätzt etwa auf 10^{78} *bis* 10^{80}
Zum Glück gibt es keine Irrmeinungen zur Kryptographie.	☐ richtig ☑ falsch	Z.B. Integrität mit Verschlüsseln gewähren.
Die Ratschläge des CH-Sicherheitsexperte müssen nicht diskutiert werden, es sind sehr gute Ratschläge.	☐ richtig ☐ falsch	Keine Antwort, diskutieren Sie das untereinander oder im Forum.