Mängel und Hintertüren in Krypto-Lösungen für's Internet

Worauf man achten sollte

NaSenbAer

Matthias Brüstle

Nürnberg 2013-11-24

Edward Snowden

Laura Poitras / Praxis Films

Edward Snowden

National Intelligence
Distinguished Service

Laura Poitras / Praxis Films

Milipol – 18th worldwide exhibition of internal State security

- Körperpanzerungen
- Taser (zum Ausprobieren)
- Waffen bis Maschinengewehre (zum Anfassen)
- Minen
- Panzerfahrzeuge (zum Probesitzen)
- Drohnen
- Überwachungsausrüstung
- ...

Milipol – Impressionen (1)

Milipol – Impressionen (2)

Milipol – Impressionen (3)

Milipol – Impressionen (4)

Zweifelhafte Sicherheit (Teil 1): Stream cipher RC4

- On The Security of RC4 in TLS and WPA: http://www.isg.rhul.ac.uk/tls/
- NSA entschlüsselt Webserver-Daten angeblich in Echtzeit: http://www.heise.de/security/meldung/NSA-entschluesselt-Webserver-Daten-angeblich-in-Echtzeit-2041383.html
- **RC4 möglichst überall abschalten!**
- nmap --script ssl-cert,ssl-enum-ciphers -p 443,465,993,995 www.example.com

Zweifelhafte Sicherheit - Teil 1: Stream cipher RC4 (1)

10. November:

```
Nmap scan report for www.allianz-fuer-cybersicherheit.de (77.87.229.82)
...
ssl-cert: Subject: commonName=www.allianz-fuer-cybersicherheit.de/...
Issuer: commonName=TC TrustCenter Class 2 L1 CA XI/...
Public Key type: rsa
Public Key bits: 2048
...
ssl-enum-ciphers:
  SSLv3
    Ciphers (2)
      TLS_RSA_WITH_RC4_128_MD5 - unknown strength
      TLS_RSA_WITH_RC4_128_SHA - strong
    Compressors (1)
      NULL
  TLSv1.0
    Ciphers (2)
      TLS_RSA_WITH_RC4_128_MD5 - unknown strength
      TLS_RSA_WITH_RC4_128_SHA - strong
    Compressors (1)
      NULL
```

Zweifelhafte Sicherheit - Teil 1: Stream cipher RC4 (2)

23. November:

```
Nmap scan report for www.allianz-fuer-cybersicherheit.de (77.87.229.82)
...
ssl-cert: Subject: commonName=www.allianz-fuer-cybersicherheit.de/...
Issuer: commonName=TC TrustCenter Class 2 L1 CA XI/...
Public Key type: rsa
Public Key bits: 2048
...
ssl-enum-ciphers:
  SSLv3/TLSv1.0/TLSv1.1/TLSv1.2
    Ciphers (7)
      TLS_DHE_RSA_WITH_AES_128_CBC_SHA - strong
      TLS_DHE_RSA_WITH_AES_256_CBC_SHA - unknown strength
      TLS_RSA_WITH_AES_128_CBC_SHA - strong
      TLS_RSA_WITH_AES_128_CBC_SHA256 - strong
      TLS_RSA_WITH_AES_256_CBC_SHA - unknown strength
      TLS_RSA_WITH_AES_256_CBC_SHA256 - unknown strength
      TLS_RSA_WITH_RC4_128_SHA - strong
    Compressors (1)
      NULL
```

Zweifelhafte Sicherheit - Teil 2: Zufallsgenerator Dual_EC_DRBG

- NIST SP 800-90A
- z.B.: Shumow/Ferguson: On the Possibility of a Back Door in the NIST SP800-90 Dual EC PRNG (http://rump2007.cr.yp.to/15-shumow.pdf)
- Wer den geheimen „Schlüssel“ zum Algorithmen-Parameter Q kennt, kennt den internen Zustand
- http://www.nytimes.com/2013/09/06/us/nsa-foils-much-internet-encryption.html?pagewanted=3:

  According to an intelligence budget document leaked by Mr. Snowden, the N.S.A. spends more than $250 million a year on its Sigint Enabling Project, which “actively engages the U.S. and foreign IT industries to covertly influence and/or overtly leverage their commercial products’ designs” to make them “exploitable.”

Zweifelhafte Sicherheit - Teil 3: Elliptische Kurven NIST P-*

- z.B. NIST P-192:
  p: **FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFF**
  a: **FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFFFFFFFFFFFFFFFC**
  b: 64210519E59C80E70FA7E9AB72243049FEB8DEECC146B9B1
  G.x: 188DA80EB03090F67CBF20EB43A18800F4FF0AFD82FF1012
  G.y: 7192B95FFC8DA78631011ED6B24CDD573F977A11E794811
  r: FFFFFFFFFFFFFFFFFFFFFFFF99DEF836146BC9B1B4D22831
  h: 1
- Spezielle Werte wegen Berechnungsgeschwindigkeit und nicht wegen Sicherheit!
- Viele kritische Publikationen zur Sicherheit.
- Schwierig zu implementieren
- Geheim gehaltene Eigenschaften?
- Bessere Kurvenwahl: Brainpool-Kurven www.ecc-brainpool.org

Zweifelhafte Sicherheit - Teil 4: Signatur-Algorithmus (EC)DSA

- Signatur für die gleichen Daten immer anders
- Grund: für jede Signatur ein zufälliger Wert k
- Problem: Ist k bekannt kann man den privaten Schlüssel berechnen
- Woher guten Zufall nehmen und nicht stehlen?

Zweifelhafte Sicherheit - Teil 4: Signatur-Algorithmus (EC)DSA (2)

- Vgl. Dual_EC_DRBG oben
- https://lists.openwrt.org/pipermail/openwrt-devel/2013-September/021318.html
  get_cycles auf MIPS liefert immer 0 und wird für Zufallszahlen verwendet
- http://emboss.github.io/blog/2013/08/21/openssl-prng-is-not-really-fork-safe/
  Android OpenSSL PRNG nicht fork-sicher
- Welche Zufallszahlengeneratorprobleme noch bekannt?
- „de-randomized“ (EC)DSA mit k als Hash(Daten + privater Schlüssel)
- Hash häufig nicht sicher gegen Seitenkanalangriffe

Zweifelhafte Sicherheit - Teil 5: Neuer Hash-Algorithmus SHA-3

- https://www.cdt.org/blogs/joseph-lorenzo-hall/2409-nist-sha-3

  NSA modifiziert nachträglich Gewinneralgorithmus wegen Geschwindigkeit

- http://www.heise.de/security/meldung/Kryptographie-NIST-rudert-bei-SHA-3-Aenderungen-zurueck-2038327.html

  NIST beläßt Gewinner

Meine aktuellen Empfehlungen: OpenPGP

- DSA oder RSA?
- DSA: Besser nicht
  (siehe Zweifelhafte Sicherheit Teil 4)
- RSA: 4096 bits (Maximum) nehmen
  Vermutlich das sicherste und trotzdem schnell.
- gpg.conf:
  cipher-algo AES256, AES192, *AES, TWOFISH, 3DES*
  digest-algo SHA512, SHA384, SHA256, SHA224, SHA1
- CAST5? CAMELLIA*? Wozu?

Meine aktuellen Empfehlungen: OpenSSH

- DSA, ECDSA oder RSA?
- DSA: Besser nicht
  (siehe Zweifelhafte Sicherheit Teil 4)
- ECDSA: Nur NIST-Kurven. Besser nicht
  (siehe Zweifelhafte Sicherheit Teil 3 und 4)
- RSA: 8192 bits (Maximum) nehmen.
  Vermutlich das sicherste und trotzdem schnell genug.
- ConnectBot mit RSA nur 4096 bits

Meine aktuellen Empfehlungen: https

- RC4 abschalten, immer, überall
- Problem: manche Websites gehen nicht mehr
- Browser-Addon zur Benachrichtigung bei CA-Zertifikatswechsel
-

Meine aktuellen Empfehlungen: Festplattenverschlüsselung

- Wenn man noch nicht verschlüsselt:
  - Braucht man es?
  - Bei Notebooks kann es nie schaden
  - Fuer MS Win: Truecrypt oder Diskcryptor?
- http://istruecryptauditedyet.com/

Algorithmenempfehlung des BSI TR-02102(-1)

- Symmetrische Algorithmen: AES-128, AES-192, AES-256 (3DES nur für Altanwendungen)
- Blockmodi: GCM (IV muß einmalig sein!), CBC, CTR (Zähler muß einmalig sein!)
- Asymmetrische Algorithmen: EC Brainpool min. 224 bits (250 ab 2015), DL (DH) und RSA min. 2000 bits (3000 ab 2015)
- Hash Algorithmen: SHA-224 (bis 2015), SHA-256, SHA-384, SHA-512
- Hinweise auch zu IPSec, SSH

Algorithmenempfehlung des BSI TR-02102-2

- Empfehlungen für TLS
- TLS 1.1 oder 1.2
- Algorithmenempfehlungen wie in Teil 1
- Mindestanforderungen für Interoperabilität
  - TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
  - TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
- EC-Kurven secp224r1, secp256r1, secp384r1, wenn Brainpool verfügbar ist migrieren

European Network of Excellence in Cryptology II

- http://www.ecrypt.eu.org/
- Nötige Sicherheit gegen Geheimdienste:
  - für 10 Jahre: 96 bits (DLP/RSA ca. 1600 bits)
  - für 20 Jahre: 112 bits (DLP/RSA 2432 bits)
  - für 30 Jahre: 128 bits (DLP/RSA 3248 bits)
- Ähnliche Algorithmenangaben wie BSI
- Keine Probleme mit NIST-Kurven

Die Zukunft: Quantencomputer

- Viel besser beim Brechen von Public Key-Systemen (RSA, ECDSA, DH) als aktuelle Computer
- http://esciencenews.com/articles/2013/06/28/large.scale.quantum.chip.validated

  128 qubits
- Quantenphysiker für US Regierungsorganisation gesucht
- Wie kann man dann noch verschlüsseln?

Symmetrische Verschlüsselung

- Effektive Schlüssellänge halbiert sich
- Es sind jetzt Algorithmen mit 256bit-Schlüsseln verfügbar und implementiert
- ECRYPT II empfiehlt 256bit

Asymmetrische Algorithmen (PKI, Signatur, ...)

- Tja, blöd. Sowas wie das DLP ist dann nur noch DL.
- „Postquantum Cryptography“ nötig
- Im Angebot:
  - Hash-based Digital Signature Schemes
  - Code-based cryptography
  - Latice-based cryprography
  - Multivariate Public Key Cryptography
- Unbequemer als bisherige Systeme

Asymmetrische Algorithmen (PKI, Signatur, ...)

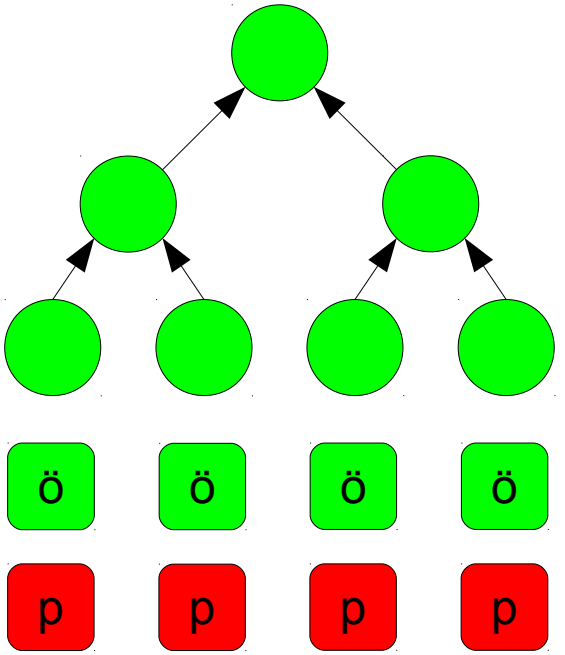
- Tja, blöd. Sowas wie das DLP ist dann nur noch DL.
- „Postquantum Cryptography“ nötig
- Im Angebot:
  - Hash-based Digital Signature Schemes
  - ~~Code-based cryptography~~
  - ~~Latice-based cryprography~~
  - ~~Multivariate Public Key Cryptography~~
- Unbequemer als bisherige Systeme
- Nicht alle sicher sicher

Hash-based Digital Signature Systems: Lamport-Diffie One-Time Signature Scheme

- Pro bit 2 geheime Zahlen (priv. Schl.) und 2 Hashwerte davon (öff. Schl.)
  ö0 = H(p0), ö1 = H(p1)
- Signatur: private Zahlen der Bitwerte
  Bitwert = 1 → s = p1, p0 wird entsorgt
- Problem:
  - Große Datenmengen (10 - 100 kB)
  - Nur einmal zu gebrauchen
- Winternitz OTSS: Mehre bits auf einmal

Hash-based Digital Signature Systems: Merkle's Tree Authentication Scheme

- Viele OTSS-Schlüssel auf eine Wurzel zurückführbar
- Damit mehrere Signaturen möglich
- Aber immer noch nur begrenzt viele Signaturen
- Über Verkettung von Bäumen erweiterbar
- Ob das jemand korrekt implementiert?



XMSS – A Practical Forward Secure Signature Scheme based on Minimal Security Assumptions

- eXtended Merkle Signature Scheme
- eprint.iacr.org/2011/484.pdf
- Das praktikabelste PQ-Signaturverfahren, das ich gefunden habe
- Minimiert die Datenhaltung
  - durch deterministische Erzeugung der Schlüsselpaare von einem Startwert und
  - durch optimierte Baum-Datenverwaltung.
- In BSI TR-02102 referenziert
- Schwierig alles zu verstehen und zu implementieren

Saenks!

NaSenbAer

Matthias Brüstle

Nürnberg 2013-11-24