Workshop IT-Sicherheit

Jan Schejbal

22. September 2008

1. Arten von Verschlüsselung
   - Hashes
   - Symmetrisch
   - Asymmetrisch
   - Diffie-Hellman
2. Anwendungen
   - PGP/GPG
   - SSL/TLS
   - Off-the-Record
   - TOR
   - SSH
3. Angriffe
   - Klassische Angriffe
   - Hässliche Angriffe
4. Fragen?

1. Arten von Verschlüsselung
   - Hashes
   - Symmetrisch
   - Asymmetrisch
   - Diffie-Hellman
2. Anwendungen
   - PGP/GPG
   - SSL/TLS
   - Off-the-Record
   - TOR
   - SSH
3. Angriffe
   - Klassische Angriffe
   - Hässliche Angriffe
4. Fragen?

Eigenschaften

- Eingabe beliebiger Länge
- Ausgabe fester Länge
- sehr schnell
- Bei kryptographisch sicheren Hashes:
  - Kleine Veränderungen der Eingabe ändern Ausgabe völlig
  - Gleichmäßige Verteilung der Ausgabe
  - Nichtumkehrbar (Einwegfunktion)
  - Kollisionsresistenz
    - zu einem Hash kein „passender“ Eingabewert findbar
    - keine zwei Eingaben mit gleichem Hash findbar

Beispiele

- Verwendung
  - Passwörter
  - Prüfsummen
- Beispiele
  - MD5
  - SHA-1
  - Whirlpool

1. Arten von Verschlüsselung
   - Hashes
   - Symmetrisch
   - Asymmetrisch
   - Diffie-Hellman
2. Anwendungen
   - PGP/GPG
   - SSL/TLS
   - Off-the-Record
   - TOR
   - SSH
3. Angriffe
   - Klassische Angriffe
   - Hässliche Angriffe
4. Fragen?

Überblick

- schnell
- Gleicher Schlüssel (key) zum Ver- und Entschlüsseln
- Unterteilt in Block- und Stromchiffren
- Beispiele
  - Blockchiffren:
    - AES/Rijndael
    - Blowfish/Twofish
    - DES/3DES
  - Stromchiffren:
    - RC4

Anwendung

- Zur Verschlüsselung mit einem Passwort:
  oft key = hash(pw)
- Wird z. B. verwendet in:
  - RAR (AES)
  - TrueCrypt (verschiedene)
  - SSL (verschiedene)
  - WEP (RC4)

Funktionsweise

- Stromchiffren
  - Erstellen eines „Zufalls“-Datenstroms von einem „Seed“ (key) ausgehend
    - Datenstrom für gleichen Seed immer gleich
  - Verschlüsselung per XOR mit Daten
  - Key darf nicht für mehrere Nachrichten verwendet werden
    - Verwendung von Initialisierungsvektoren

1. Arten von Verschlüsselung
   - Hashes
   - Symmetrisch
   - Asymmetrisch
   - Diffie-Hellman
2. Anwendungen
   - PGP/GPG
   - SSL/TLS
   - Off-the-Record
   - TOR
   - SSH
3. Angriffe
   - Klassische Angriffe
   - Hässliche Angriffe
4. Fragen?

Überblick

- Unterschiedliche Schlüssel zum Ver- und Entschlüsseln (private key, public key)
- Auch als „public key cryptography“ bekannt
- sehr langsam
- Algorithmen:
  - RSA
  - DSA (nur Signatur)
- Nutzung: PGP, SSL, ...
- private key nicht aus public key ermittelbar, umgekehrt geht es

Verschlüsselung und Signatur

- Mit dem private key können Daten entschlüsselt oder signiert werden
- Mit dem public key können Daten verschlüsselt oder Signaturen geprüft werden
- wird näher bei PGP erläutert

Funktionsweise

- am Beispiel von RSA:
  - zwei Primzahlen werden erstellt (private key)
  - Produkt der zwei Primzahlen ist public key
  - Faktorisierung nicht möglich
    - Zahlen sind 1024 bis 4096 bit lang

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Verwendungszweck

- Schlüsselaustausch über passiv abgehörte Leitung
- verwendet zum Aushandeln von Session-Keys
- aktiver Angriff durch MitM immer noch möglich
  - oft über Signaturen etc. gesichert

Funktionsweise

- Beide erstellen je eine Zahl
- Beide berechnen eine Funktion dieser Zahl und teilen das Ergebnis dem Partner mit
- Beide berechnen aus der eigenen (geheimgehaltenen) Zahl und dem Funktionsergebnis des anderen einen Wert
- Dieser Wert ist der Schlüssel
- Komplizierte Mathematik, bei Interesse: Wikipedia

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Verwendung

- Verschlüsselungssoftware (hauptsächlich) für Mails
- hauptsächlich asymmetrische Verschlüsselung
- jeder erzeugt ein Schlüsselpaar (private/public)
- er gibt den public Key seinen Kommunikationspartnern
  - Keyserver
  - private key muss geheim bleiben!
- Nachrichten werden mit dem public key des Empfängers **verschlüsselt**
- Nachrichten werden mit dem eigenen private key **signiert**
- symmetrische Verschlüsselung möglich

Arbeitsweise

- Signatur: Hash der Nachricht asymmetrisch signieren
  - Hash schnell, asymmetrische Signatur langsam
- Verschlüsselung:
  - Nachricht symmmetrisch mit zufälligem Key verschlüsseln
  - symmetrischen Key aysmmetrisch verschlüsseln
  - dadurch Versand an mehrere Empfänger möglich
- ASCII-Armor um Nachricht übertragungsfähig zu machen
- private key wird mit passphrase symmetrisch verschlüsselt

Problem: Richtiger Public Key?

- Angriffsszenario:
  - Jemand gibt mir seinen public key und behauptet es sei der des Empfängers
    - z. B. indem er den falschen Key an einen Keyserver schickt
  - Ich verschlüssele die Mail mit dem falschen key
    - weil ich zu faul/doof war den Key zu verifizieren
  - Er fängt sie ab, liest sie, verschlüsselt sie mit dem richtigen key und leitet sie weiter

Schutz

- Möglichkeit 1: Key persönlich überbringen
  - Aufwändig, oft nicht möglich
- Möglichkeit 2: telefonische Prüfung
  - Fingerprint eines Keys reicht zur Prüfung
  - Nicht 100% sicher (Stimme kann gefälscht werden!)
- Möglichkeit 3: Keysigning
  - Wir signieren unsere Keys gegenseitig
  - „Ich bestätige, dass dieser Key tatsächlich der Person X gehört"
  - Web of Trust

Web of Trust

- Empfänger hat den Key von jemandem, der den Key des Absenders geprüft und signiert hat
- Keysigning parties
  - Viele Leute treffen sich und signieren gegenseitig die Schlüssel
  - Aufbau des Web of Trust
  - Typischerweise auf Messen, Veranstaltungen etc.
  - Oft vertrauenswürdige „Großsignierer“ dabei (c't)
- Sauber arbeiten!
  - Sorgfältig Fingerprint prüfen
  - Identität des Gegenübers mittels Perso etc. prüfen (!)

Praxis

- Viele verschiedene Programme
  - GPG für Kommandozeile
  - PGP Suite
  - GnuPT für Windows
  - Plugins für Mailprogramme
    - Enigmail
  - Verschlüsselung manuell (Copy&Paste) möglich
    - darüber: webmail, theoretisch auch ICQ etc. (dazu später)

Praxis, konkret

- Was ist zu tun?
  - Software nach Wahl besorgen
  - Key erstellen und sichern
    - Gute Passphrase wählen
    - System muss sicher sein
  - public key verbreiten, signieren lassen
  - public keys von Kommunikationspartnern geben lassen und verifizieren
  - Verschlüsselung nutzen

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Was ist das?

- TLS = Nachfolger von SSL
- z. B. https, aber für alle protokolle nutzbar (pop3s, jabber, ...)
- Server beweist dass er wirklich er ist, indem er das Vorhandensein eines private Key beweist
  - signiert beim Verbindungsaufbau bestimmte Daten
- symmetrischer Verschlüsselungsschlüssel wird ausgehandelt
- Stellt zwei wichtige Merkmale sicher:
  - Vertraulichkeit (keiner kann abhören)
  - Authentizität (keiner kann fälschen)
- verwendet alle genannten Verschlüsselungsarten!

Gefahr: MitM

- Man-in-the-Middle-Angriffe drohen
  - wie bei PGP: Angreifer täuscht vor der Kommunikationspartner zu sein
- Schutz: Zertifikate
- Vertrauenswürdige CA signiert Key des Servers
  - Browser/Betriebssystem kennt eine Liste von CAs inkl. Keys
- CA prüft Identität auf verschiedene Arten, will meist viel Geld dafür

Selbstsignierte Zertifikate

- Ohne CA-Signatur: „Selbstsigniertes“ Zertifikat
  - Kein direkter Schutz gegen MitM
  - Manuelle Prüfung möglich
  - Besser als nichts (passives Abhören unmöglich)
  - Browser zeigt Warnung
- Besser: Kostenlose CA nutzen

Kostenlose CAs

- StartSSL
  - Prüfung per E-Mail
  - Anerkannt in Firefox
  - Nicht anerkannt in IE
- CAcert
  - Prüfung per Mail und Assurer
  - In Browsern nicht vorinstalliert
  - Besser als selbstsigniertes
    - Viele installieren die CA manuell
- Nachteil: Kostenlose CAs oft nicht in Browsern
  - Nicht für Seiten für Allgemeinheit nutzbar!

Debian-Debakel

- Private Keys waren nicht zufällig generiert
  - Nur ca. 32.000 verschiedene keys
  - Diese lassen sich leicht alle ausrechnen
- Verbindungen zu betroffenen Servern unsicher
  - Zertifikate oft immer noch gültig
  - Mit dem Key kann man MitM-Angriffe machen
  - ELSTER-Download betroffen!
- Schutz: Schwache keys ablehnen
  - SSL-Blacklists für Linux
  - Extensions für Firefox
  - c't-SSL-Wächter für Windows

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Was ist das?

- Verschlüsselungssystem für Instant Messaging
- Plugins für die meisten Clients
- Sehr leicht zu bedienen, fast alles automatisch
- Bietet:
  - Verschlüsselung
  - Authentifizierung
  - perfect forward secrecy
    - Alte Nachrichten selbst bei Schlüsselverlust sicher
  - deniability (Abstreitbarkeit)

Funktionsweise

- Ähnlich PGP
  - erstellt keys automatisch
  - erkennt ob Gesprächspartner OTR hat und verschlüsselt automatisch
- Zusatzfunktionen:
  - perfect forward secrecy
    - temporäre Schlüssel per Diffie-Hellman
  - deniability
    - veröffentlicht Signaturschlüssel nach Sitzungsende
    - Gesprächsprotokoll kann gefälscht werden
    - während des Gesprächs trotzdem sicher!

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Was ist das?

- Anonymisierungsdienst
- Leitet Verkehr über mehrere Rechner
  - langsam, instabile Verbindungen
- verschlüsselt (Schalenprinzip)
  - häufiges Missverständnis: kein Schutz der Daten ab Exitnode!
- Bruch der Anonymität erfordert Kooperation aller Nodes

Sicherheit

- offiziell „beta”
  - „This is experimental software. Do not rely on it for strong anonymity.“
- dennoch relativ sicher
- Andere Angriffe möglich
  - Java
  - Trojaner
  - Prozessortemperatur-Trick
  - Timingangriffe
  - Man sollte wissen was man tut und aktive Inhalte filtern

Bitte

- Baut bitte keinen Scheiß damit
  - nichts allzu illegales
    - gibt Ärger für Exitnodes
    - Hausdurchsuchungen will keiner
    - Weniger Exitnodes - Netz geht kaputt
  - kein P2P
    - ist eh langsam
    - schadet dem Netz durch den vielen Traffic
- Viele Leute brauchen das Netz wirklich (China!)

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Kurzer Überblick

- verschlüsselte Verbindung
- verschiedene Datenkanäle (shell, tunnel, sftp)
- Authentifizierung des Servers über den „Host Key”
  - Nutzer muss Fingerprint des host key manuell prüfen
- Authentifizierung des Clients über Passwort oder Key
  - Client-Keys werden auf dem Server hinterlegt

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Bruteforce

- „durchprobieren”
- Aufwand steigt exponentiell mit Schlüssellänge
  - Schlüssel ein Bit länger verdoppelt Aufwand!
- meist nicht praktikabel
- außer bei schwachem Passwort
- Rechenbeispiel:
  - 1.000.000 Keys/s prüfbar
  - Zufallsschlüssel 128 bit $= 2^{128}$ Möglichkeiten
    - 10782897524556318080696079 Jahre
  - Passwort, 6 Stellen, a-zA-Z0-9 $= 62^6$ Möglichkeiten
    - 15 Stunden
    - 8 Stellen: $15 * 62 * 62 = 6$ Jahre

Implementierungsfehler

- Debian-Lücke: nur 32000 private keys
- Sehr beliebt: Passwort steht nochmal irgendwo (fast) im Klartext
- Selbst gebastelt statt anerkanntem Algorithmus

1 Arten von Verschlüsselung
- Hashes
- Symmetrisch
- Asymmetrisch
- Diffie-Hellman

2 Anwendungen
- PGP/GPG
- SSL/TLS
- Off-the-Record
- TOR
- SSH

3 Angriffe
- Klassische Angriffe
- Hässliche Angriffe

4 Fragen?

Physikalische Angriff

- Rechner/Speichermedium mit entschlüsselten Daten oder Key mitnehmen
- TEMPEST
  - Abfangen von unerwünschter Abstrahlung
  - Funktioniert auf große Entfernungen
  - Bei Röhrenbildschirmen trivial, aber auch bei Tastaturen, Netzwerkkablen, LCDs etc. möglich
  - Wahlcomputer
  - Schutz: Abschirmung
- Reflektionen von Brillen, Teekannen etc.
  - bis zu 10 Meter mit billiger Ausrüstung
- Tastaturgeräusche

Seitenkanalangriffe

- Timing

Codebeispiel

```
for (int i=0;i<20;i++) {
  if ( storedhash[i] != currenthash[i] )
    return false;
}
```

- Stromverbrauch

Soziale Angriffe

- Größte Schwachstelle: Mensch
- Social Engineering

  Guten Tag, mein Name ist Müller von der IT-Abteilung, uns ist die Datenbank abgestürzt und wir müssen Ihr Passwort neu setzen

- rubber hose cryptanalysis

  a rubber hose is applied forcefully and frequently to the soles of the feet until the key to the cryptosystem is discovered, a process that can take a surprisingly short time and is quite computationally inexpensive

Schlussfolgerung

- Schutz gegen alle Angriffe nicht möglich
- Eine Kette ist so stark wie ihr schwächstes Glied!
  - Verschlüsselungsverfahren selbst: meist stärkstes Glied
- Kosten-Nutzen-Abwägung
- Wie sicher soll es sein?

Noch Fragen?

- Folien gibt es im Netz
  - inklusive Quellcode
  - erstellt mit LaTeX Beamer Class
  - und eigenem Programm outlinebeamer
    - http://outlinebeamer.sourceforge.net