

CBC-Mode

Nachricht in n Teile gliedern.

Schlüssel \Rightarrow Permutationsmatrix mit Einheitsvektoren

$$\pi = \begin{pmatrix} 1 & \cdots & n \\ a & \cdots & z \end{pmatrix} \Rightarrow P_\pi = \begin{pmatrix} e_a \\ \vdots \\ e_z \end{pmatrix}$$

Verschlüsselung:

$$c_0 = IV$$

$$c_i = E(\pi, c_{i-1} \oplus m_i) = P_\pi \cdot (c_{i-1} \oplus m_i)$$

$$\Rightarrow c_1, \dots, c_n$$

Entschlüsselung:

$$m_i = D(\pi^{-1}, c_i) \oplus c_{i-1} = (P_{\pi^{-1}} \cdot c_i) \oplus c_{i-1}$$

mit $\pi^{-1} = \pi'$ (transponiert)

CFB-Mode

Verschlüsselung:

$$c_i = E(\pi, c_{i-1}) \oplus m_i$$

Entschlüsselung:

$$m_i = E(\pi, c_{i-1}) \oplus c_i$$

CTR-Mode

Verschlüsselung:

$$c_i = E(\pi, IV + (i - 1)) \oplus m_i$$

(binäre Addition, Überträge verwerfen)

Entschlüsselung:

$$m_i = E(\pi, IV + (i - 1)) \oplus c_i$$

Hashfunktion

Nicht injektive Abbildung, die Urbildbereich auf erheblich kleineren Bildbereich abbildet. Einwegfunktion.

Speicherung von Passwörtern Dateivalidierung

Message Authentication Code (MAC)

Hashfunktion mit geheimen Schlüssel zur Integritätsprüfung von Nachrichten. Ermöglicht kein non-repudiation, daher nicht als digitale Unterschrift geeignet.

Kollisionsresistenz

Es ist schwierig zwei Werte x und y mit $H(x) = H(y)$ zu bestimmen

Schwache Kollisionsresistenz

Es ist schwierig zu geg. Wert x ein x' mit $H(x) = H(x')$ zu bestimmen

Shamir Secret-Sharing

t von n Stakeholdern sind nötig, um Geheimnis $k = f(0)$ zu entschlüsseln. Außerdem gegeben: Primzahl $p > n, k$ und vom Dealer gewähltes Polynom $f(x)$ vom Grad $t - 1$

Schlüssel $s_i = f(i) \bmod p$

Secret Recovery:

$$k = f(0) = \sum s_i \cdot l_i(0) \quad l_i(0) := \left[\prod_{j=1, j \neq i} \frac{j}{j-i} \right] \bmod p \text{ Beim}$$

Berechnen von $l_i(0)$ die Nenner zu $(a)^{-1}$ zusammenfassen und als inverses Element berechnen.

Erweiterter Euklidischer Algorithmus (EEA)

$$ggT(a, b) \text{ und } x \cdot a + y \cdot b = d$$

q	r	x	y	a	b	x_2	x_1	y_2	y_1
X	X	X	X	a	b	1	0	0	1
a/b	$a \bmod b$	$x_2 - qx_1$	$y_2 - qy_1$	b	r	x_1	x	y_1	y
\dots									
$?$	0	$?$	$?$	=d	0	=x	$?$	=y	$?$

Inverses Element $(a)^{-1}$ berechnen

Es gilt: $[a \cdot (a)^{-1}] \bmod k = 1$

$ggT(a, k)$ mit EEA durchführen.

$$(a)^{-1} = \begin{cases} (x + k) \bmod k & \text{wenn } x < 0 \\ x \bmod k & \text{sonst} \end{cases}$$

iptables

Chains: Pakete von...

INPUT Außen an System mit Firewall

FORWARD Außen an System innerhalb des geschützten Bereichs

OUTPUT Innen nach Außen

Targets:

ACCEPT Akzeptieren und Weiterleiten

DROP Verwerfen ohne Info an Absender

REJECT Verwerfen mit Info

Parameter:

Param	Argumente	Erklärung
-P	Chain Target	Policy für Chain
-A	Chain	Append Regel
-p	Protokoll	z.B. tcp, icmp
-j	Target	Gibt Target an
-F	Chain	Flush, löscht alle Regeln für Chain außer Standardregeln mit -P
-i/-o	Interface	in-interface bzw. out-interface
-dport/sport	port	Destination/Source port

Reliability

Wahrscheinlichkeit, dass ein System über gewissen Zeitraum korrekt funktioniert.

Reihenschaltung: $R_{ges}(t) = \prod_{i=1}^n R_i$

Falls gleiches Modell $R_i(t) = e^{-\lambda_i t}$ gilt: $R_{ges}(t) = e^{-\lambda t}$ mit $\lambda = \sum_{i=1}^n \lambda_i$

Parallelschaltung: $R_{ges}(t) = 1 - \prod_{i=1}^n (1 - R_i(t))$

Availability

Verfügbarkeit eines Systems in %:

$$A = \text{Uptime} / (\text{Downtime} + \text{Uptime}) = \text{MTTF} / (\text{MTTF} + \text{MTTR})$$

MTTF (Mean Time to Failure): $MTTF = \int_0^\infty R(t) dt \stackrel{\text{exp}}{=} \frac{1}{\lambda}$

MTTR (Mean Time to Recovery):

Wahrscheinlichkeit $F(t)$, dass System bis t fehlerhaft wird.

Exp. Modell: $F(t) = 1 - e^{-\lambda t}$

Reliability Funktion: $R(t) = 1 - F(t)$ exp. Modell: $R(t) = e^{-\lambda t}$

Sicherheitsziele

Safety	Betriebssicherheit, Ablauf- und Ausfallsicherheit
Security	Angriffssicherheit
Authenticity	Authentizität, Echtheit, Glaubwürdigkeit
Integrity	Integrität der Daten
Confidentiality	Vertraulichkeit, keine unautorisierte Datengewinnung
Availability	Verfügbarkeit, keine Funktionsbeeinträchtigungen
Non repudiation	Verbindlichkeit und Zuordenbarkeit

Kasiski-Test

1. Doppelt vorkommende N-Gramme und deren Abstand bestimmen
2. Primfaktorzerlegung
3. Gemeinsame Faktoren entsprechen Schlüssellänge

OTP – Beweis perfekte Sicherheit

zu zeigen: $Pr[Enc(k, m_1) = c | k \xleftarrow{\$} \mathcal{K}] = Pr[Enc(k, m_2) = c | k \xleftarrow{\$} \mathcal{K}]$

$$Pr[Enc(k, m) = c | k \xleftarrow{\$} \mathcal{K}] = Pr[m \oplus k = c | k \xleftarrow{\$} \mathcal{K}] \quad (1)$$

$$= Pr[k = c \oplus m | k \xleftarrow{\$} \mathcal{K}] \quad (2)$$

$$= Pr[k = k^* | k \xleftarrow{\$} \mathcal{K}] = \frac{1}{2^l} \quad (3)$$

Satz von Euler

$$n > 1 \Rightarrow a^{\phi(n)} \equiv 1 \pmod{n} \forall a \in \mathbb{Z}_n^*$$

Spezialfall: kleiner Satz von Fermat

Primzahl $p > 1$ so gilt für jede Zahl x mit $ggT(x, p) = 1$:

$$x^{p-1} \equiv 1 \pmod{p}$$

In jeder endl. Gruppe M gilt: $x^{\text{ord}(M)} = 1 \forall x \in M$

Korrektheit von ElGamal

Bekannt: $E(g, h, m) = (c_1, c_2) := (g^k, mh^k)$

Zu zeigen: $D(E(m)) = m \quad \forall m \in \mathbb{Z}_p^*$

Beweis:

$$D(E(m)) = D(g^k, mh^k) \quad (4)$$

$$\equiv mh^k \cdot (g^k)^{-a} \pmod{p} \quad (5)$$

$$\equiv mh^k \cdot g^{k(p-1-a)} \pmod{p} \quad (6)$$

$$\equiv mg^{ak} \cdot g^{k(p-1)-ak} \pmod{p} \quad (7)$$

$$\equiv m \cdot g^{k(p-1)} \pmod{p} \quad (8)$$

$$\equiv m \pmod{p} \quad (9)$$

RSA

Allgemein:

1. Wähle zufällig große Primzahlen p, q
2. Setze $n = p \cdot q$
3. Wähle zufällig (e, d) mit $ed \equiv 1 \pmod{\varphi(n)}$, mit $\varphi(n) = (p-1)(q-1)$

Public key: $pk = (e, n)$

Private key: $sk = (d, n)$

(Public und private key sind invers zueinander) Ver-/Entschlüsselung:

$$E(pk, m) = m^e \pmod{n}$$

$$D(sk, c) = c^d \pmod{n}$$

RSA Signatur:

$$\text{sign}(sk, m) = h(m)^d \pmod{n}$$

$$\text{verify}(pk, m, s) : [s^e = h(m)?]$$

Zeigen, dass Chiffriersystem definiert ist

Zu zeigen ist: $D[E(m_1, \dots, m_n)] = m_1, \dots, m_n$

Ansätze zur Userauthentifizierung

by knowledge (PIN, Passwort)

- free recall-based (Freies Erinnern ohne Abrufhilfe)

Draw a Secret (DAS), Android

- cued recall-based (Mit Abrufhilfe, Sicherheitsfrage)

PassPoints

- Recognition-based (Wiedererkennen)

by ownership (smart card, token, SIM)

by inherence (Biometrie, Verhalten)

Features of authentication schemes

Security

- Guessability $|P| = \log_2 \sum_{l \in L} |A|^l$

Angreifbar über:

Brute Force (Offline), Dictionary Attack (On/Offline)

- Observability

Angreifbar über:

Shoulder Surfing (Human Observer), Spyware (Technology)

- Recordability

Angreifbar über:

Social Engineering (Deception), Theft (Unsecured Record)

Usability

- Compatibility

- Costs

- Maturity

- Proprietary

Mandatory Access Control (MAC)

Systembestimmte (regelbasierte) Festlegung von Sicherheitseigenschaften, die benutzerdefinierte Rechte dominieren. Bell-La Padula Modell erweitert Matrixbasiertes Zugriffskontrollsystem.

Rechte $R = \{read, append, read - write\}$

Sicherheitsklassen $SC = (A, B)$ mit A total geordnete Sicherheitsmarken, mit B Menge von Kategorien/Personen.

Jedes Subjekt hat Clearance: $SC(s) \in SC$

Jedes Objekt hat Classification: $SC(o) \in SC$

Regeln:

- Simple-security-Property/no read up-Regel. Lesen nur erlauben, wenn

$$read \in M(s, o) \wedge SC(s) \geq SC(o)$$

- *-Property (no write down-Regel):

$$A \text{ nur erlauben, falls } append \in M(s, o) \wedge SC(s) \leq SC(o)$$

$$RW \text{ nur erlauben, falls } read - write \in M(s, o) \wedge SC(s) = SC(o)$$

- Während Laufzeit keine Änderungen

Usability aspects of knowledge based authentication

Effectiveness	Objective	Success-/Errorrate
Efficiency	Objective	Time needed per attempts, attempts needed
Satisfaction	Subjective	Questionnaires (SUS), Recommendations

Discretionary Access Control (DAC) - Fahrschein

Eigentümer bestimmt Rechte für einzelne Objekte in Matrix M

Menge von Objekten O , Subjekten S , Rechten R .

$$M : S \times O \rightarrow 2^R$$

Vorteile: Einfach in Nutzung & Implementierung, flexibel

Nachteile: dynamische Rechte schlecht abbildbar, Vergabe/Rücknahme von Rechten relativ komplex

Spaltenweise Speicherung: $ACL(Datei1) = \{(Prozess1), \{read, write\}\}$

Role-Based Access Control (RBAC)

Berechtigungen für Rollen statt Nutzer, intuitiv, flexibel

Menge von Subjekten S , Rollen R , Zugriffsrechten P .

Zwei Abbildungen/Zuordnungen:

Benutzer - Rollen: $sr : S \rightarrow 2^R$

Rolle - Rechte: $pr : R \rightarrow 2^P$ (Rolle bestimmt Rechte)

Sitzung: $session \subseteq S \times 2^R$

Für alle Paare $(s, rl) \in session : rl \subseteq sr(s)$

rl : aktive Rollen von s

Invarianten:

$$\forall s \in S : R_j \in session(s) \Rightarrow sr(s):$$

Subjekt darf nur Rollen haben, in denen es Mitglied ist

$$\forall s \in S : exec(s, p)_{\text{hat } s \text{ p?}} \Rightarrow R_j \in R : R_j \in session(s) \wedge p \in pr(R_j):$$

Subjekt hat nur Rechte der aktiven Rolle

$$\forall s \in S : exec(s, p) \Rightarrow \exists R_i \in R, R_i \in session(s) \wedge (p \in pr(R_i) \vee \exists R_j : (\exists R_j : (R_j < R_i \wedge p \in pr(R_j))))$$

Hierarchisches Modell

Statischer Ausschluss von Rollenpaaren: $SSD \subseteq R \times R$

$$\forall R_i, R_j \in R, \forall s \in S : (s \in member(R_i) \wedge s \in member(R_j)) \Rightarrow (R_i, R_j) \notin SSD$$

Dynamischer Ausschluss von Rollenpaaren: $DSD \subseteq R \times R$

$$\forall R_i, R_j \in R, \forall s \in S :$$

$$(s \in member(R_i) \wedge s \in member(R_j) \wedge \{R_i, R_j\} \subseteq active(s))$$

$$\Rightarrow (R_i, R_j) \notin SSD$$

$$active(s) = \{R_i | \exists R_s \in R \wedge (s, R_s) \in session \wedge R_i \in R_s\}$$