

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.

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:

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

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

Erweiterter Euklidischer Algorithmus (EEA)

$ggT(a, b)$ 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

Verschwiegenheit, keine unautorisierte Datengewinnung

Availability

Verfügbarkeit, keine Funktionsbeeinträchtigungen

Non repudiation

Verbindlichkeit und Zuordenbarkeit