Asymmetrisk kryptografi och meddelandeautentisering

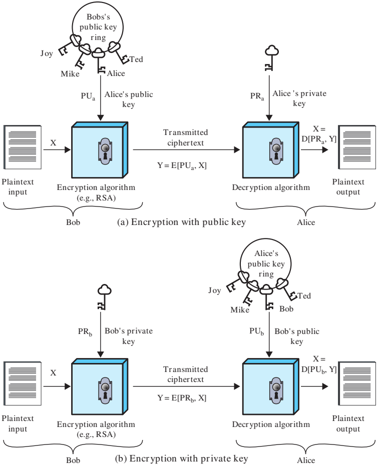
Daniel Bosk[1]

Avdelningen för informations- och kommunikationssytem (IKS), Mittuniversitetet, SE-851 70 Sundsvall.

pubkey.tex 2195 2015-02-10 15:39:00Z danbos

Översikt

Introduktion



Figur: Översikt av asymmetrisk kryptering. Bild: [Sta11].

Introduktion

Sats (Fermat–Eulers sats)

1. *Om n och a är heltal sådana att* $\gcd(n, a) = 1$,
2. *då gäller att* $a^{\phi(n)} \equiv 1 \pmod{n}$,
   där $\phi(n) = p_1^{e_1-1}(p_1 - 1) \cdots p_m^{e_m-1}(p_m - 1)$,
   p_i *är alla primtalsfaktorer och* e_i *respektive exponent för* $n = p_1^{e_1} p_2^{e_2} \cdots p_m^{e_m}$.

Exempel

1. Låt $n = 10$ och $a = 3$ (a får inte vara på formen $a = 2k$ eller $a = 5k$ för $k \in \mathbb{N}$). Då är $\gcd(n, a) = \gcd(10, 3) = 1$.
2. Vi har också $a^{\phi(n)} \equiv 3^{(2-1)(5-1)} \equiv 3^4 \equiv 81 \equiv 1 \pmod{n}$.

Mittuniversitetet
MID SWEDEN UNIVERSITY

Kryptosystem

Rivest, Shamir, Adleman (RSA)

Definition

Låt $n = pq$, där p och q är primtal. Låt $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$ och

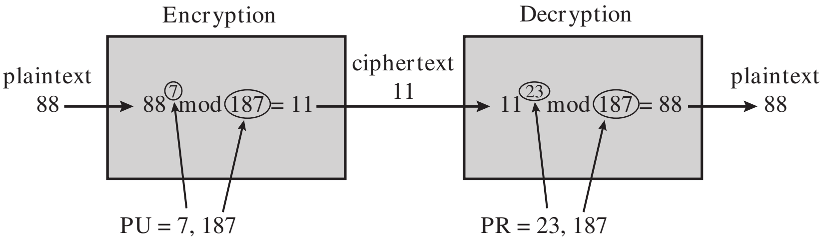
$$\mathcal{K} = \{(n, p, q, e, d) \colon ed \equiv 1 \pmod{\phi(n)}\}.$$

För $k = (n, p, q, e, d)$ definiera

$$\begin{aligned} e_k(p) &= p^e \mod n \text{ och} \\ d_k(c) &= c^d \mod n, \end{aligned}$$

där $p \in \mathcal{P}$ och $c \in \mathcal{C}$. Tupeln (n, e) utgör den *publika nyckeln* och (p, q, d) den *privata nyckeln*.

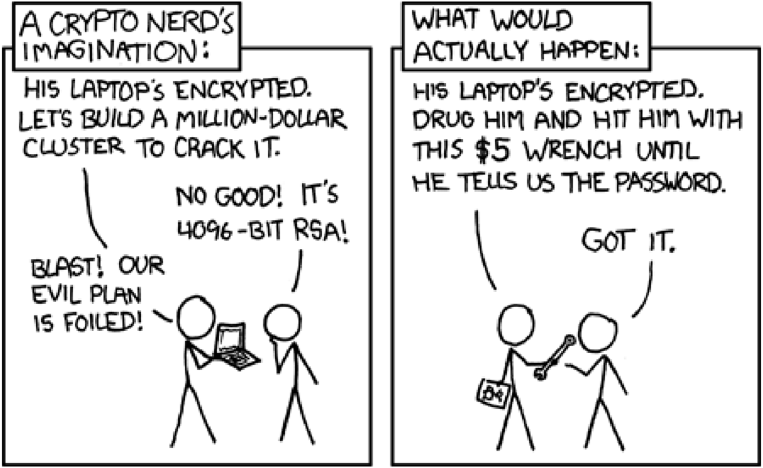
Kryptosystem



Figur: Ett exempel på kryptering med RSA. Bild: [Sta11].

Kryptosystem

Men glöm ej . . .

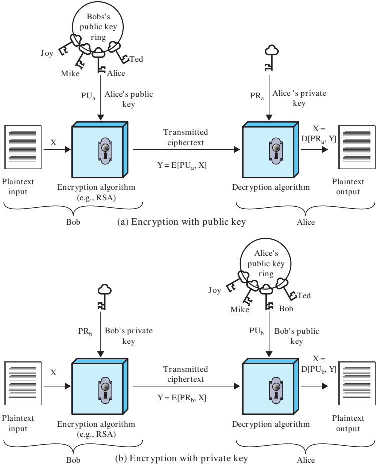


Figur: En serieruta som illustrerar vikten av den sociala aspekten på säkerhet. Bild: [xkc].

Kryptosystem

- Det finns även andra asymmetriska chiffer.
- Ett exempel är El Gamal.
- Detta system bygger på diskreta logaritmproblemet (DLP) snarare än faktoriseringsproblemet som i RSA:s fall.

Digitala signaturer



Figur: Översikt av asymmetrisk kryptering. Bild: [Sta11].

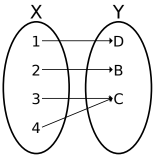
Mittuniversitetet
MID SWEDEN UNIVERSITY

Digitala signaturer

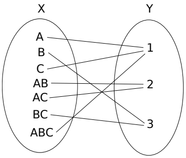
- Kan ha digitala signaturer med symmetriska chiffer, men i mycket begränsad utsträckning.
- Det är inte jag som skapat detta meddelande, då måste det vara den andre.
  - A och B delar nyckeln k.
  - A tar emot $n, E_k(n, m)$.
  - A vet att A inte skapat meddelandet, alltså måste någon annan med tillgång till nyckeln k gjort det.
  - Eftersom att B är den enda utöver A som känner till nyckeln måste meddelandet m vara från B.

Introduktion till hashfunktioner

- En hashfunktion är en funktion $h\colon X \to Y$, där X är en möjligen oändlig mängd och Y är en ändlig mängd.
- Den är således en icke-injektiv surjektiv funktion och saknar invers $h^{-1}\colon Y \to X$ sådan att $h^{-1}(h(x)) = x$ för alla $x \in X$.



(a) $h\colon X \to Y$



(b) $h'\colon X \to Y$

Figur: Två icke-injektiva surjektiva funktioner h respektive h'.

Introduktion till hashfunktioner

- Finns många olika hashfunktioner:
  - MD5,
  - SHA1,
  - SHA256,
  - SHA512.
- Tillämpningsområdet är stort:
  - verifiera integritet hos filer,
  - snabb sökning i datastrukturer,
  - digitala signaturer,
  - skydda lösenord.

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

Definition

En *hashfamilj* är en tupel $(\mathcal{X}, \mathcal{Y}, \mathcal{K}, \mathcal{H})$, där

- $\mathcal{X}$ är mängden av möjliga *meddelanden*.
- $\mathcal{Y}$ är en ändlig mängd av möjliga *meddelandesammandrag*.
- $\mathcal{K}$ är en ändlig mängd av möjliga nycklar.
- För varje nyckel $k \in \mathcal{K}$ finns en hashfunktion $h_k \in \mathcal{H}$ sådan att $h_k \colon \mathcal{X} \to \mathcal{Y}$.

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

- $\mathcal{X}$ kan vara ändlig eller oändlig, men alltid $|\mathcal{X}| \geq |\mathcal{Y}|$.
- Vissa hashfunktioner saknar nycklar, då är $|\mathcal{K}| = 1$.
- Låt $\mathcal{Y}^{\mathcal{X}}$ beteckna mängden av alla funktioner från $\mathcal{X}$ till $\mathcal{Y}$, då är $|\mathcal{Y}^{\mathcal{X}}| = |\mathcal{Y}|^{|\mathcal{X}|}$.

Formell behandling av hashfunktioner

Preimage resistant eller *one-way*

Inversa bilden (*preimage*)

1. Given hashfunktionen $h\colon \mathcal{X} \to \mathcal{Y}$ och element $y \in \mathcal{Y}$.
2. Hitta $x \in \mathcal{X}$ sådant att $h(x) = y$.

Formell behandling av hashfunktioner

Second preimage resistant

Andra inversa förbilden (*second preimage*)

1. Given hashfunktionen $h\colon \mathcal{X} \to \mathcal{Y}$ och element $x \in \mathcal{X}$.
2. Hitta $x' \in \mathcal{X}$ sådant att $x' \neq x$ och $h(x') = h(x)$.

Formell behandling av hashfunktioner

Collision resistant

Kollision

1. Given hashfunktionen $h\colon \mathcal{X} \to \mathcal{Y}$.
2. Hitta $x, x' \in \mathcal{X}$ sådana att $x' \neq x$ och $h(x') = h(x)$.

Formell behandling av hashfunktioner

Random Oracle Model

- Idealisering av en hashfunktion.
- Kan liknas vid ett orakel som ger slumpmässiga svar på frågor.
- Men vid upprepningar ska samma svar ges.
- En funktion $h \in \mathcal{Y}^{\mathcal{X}}$ väljs slumpmässigt, vi får enbart ställa frågor som ”vad är $h(x)$?”
- Innan vi ställer frågan $h(x)$ vet vi ingenting om h.
- Efter att vi ställt frågan $h(x)$ och erhållit svaret y, då vet vi enbart att $h(x) = y$.

Formell behandling av hashfunktioner

- Det går att visa att om man kan hitta en andra invers avbildning, då kan man hitta en kollision.
- Det går även att visa att om man kan hitta en invers avbildning, då kan man hitta en kollision.
- Följaktligen, om en hashfunktion är *collision resistant*, då är den även *preimage* och *second preimage resistant*.

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

Sats (Oberoendesatsen)

Antag att $h \in \mathcal{Y}^{\mathcal{X}}$ *väljs slumpmässigt. Låt* $\mathcal{X}_0 \subseteq \mathcal{X}$*. Antag att värdet* $h(x)$ *bestämts genom att fråga oraklet om och endast om* $x \in \mathcal{X}_0$*. Då gäller att*

$$\Pr(h(x) = y) = \frac{1}{|\mathcal{Y}|}$$

för alla $x \in \mathcal{X} \setminus \mathcal{X}_0$ *och alla* $y \in \mathcal{Y}$*.*

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

Algoritm (Hitta invers avbild)

input $h \in \mathcal{Y}^{\mathcal{X}}, y \in \mathcal{Y}, Q \in \mathbb{N}$
output x sådant att $h(x) = y$

Välj någon mängd $\mathcal{X}_0 \subseteq \mathcal{X}$ sådan att $|\mathcal{X}_0| = Q$.
for all $x \in \mathcal{X}_0$ **do**
  if $h(x) = y$ **then**
    return x
  end if
end for
return misslyckande

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

Sats

För någon mängd $\mathcal{X}_0 \subseteq \mathcal{X}$ med $|\mathcal{X}_0| = Q$ är sannolikheten ϵ att algoritmen för att finna en inverterad avbildning lyckas $\epsilon = 1 - \left(1 - \frac{1}{|\mathcal{Y}|}\right)^Q$.

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

Algoritm (Hitta kollision)

input $h \in \mathcal{Y}^{\mathcal{X}}, Q \in \mathbb{N}$
output $x, x' \in \mathcal{X}$ sådana att $x \neq x', h(x) = h(x')$

Välj någon mängd $\mathcal{X}_0 \subseteq \mathcal{X}$ sådan att $|\mathcal{X}_0| = Q$.
for all $x \in \mathcal{X}$ **do**
  Låt $y_x = h(x)$.
end for
if $y_x = y_{x'}$ för något $x \neq x'$ **then**
  return (x, x')
end if
return misslyckande

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

Sats

För någon mängd $\mathcal{X}_0 \subseteq \mathcal{X}$ med $|\,\mathcal{X}_0\,| = Q$ är sannolikheten ϵ för att kollisionsalgoritmen lyckas följande:

$$\epsilon = 1 - \left(\frac{|\,\mathcal{Y}\,| - 1}{|Y|}\right)\left(\frac{|\,\mathcal{Y}\,| - 2}{|\,\mathcal{Y}\,|}\right)\cdots\left(\frac{|\,\mathcal{Y}\,| - Q + 1}{|\,\mathcal{Y}\,|}\right).$$

Formell behandling av hashfunktioner

- Vi hade att sannolikheten för ingen kollision är $\prod_{i=1}^{Q-1}(1 - 1/|\mathcal{Y}|)$.
- För små x gäller att $1 - x \approx e^{-x}$.
- Då får vi

$$\prod_{i=1}^{Q-1}(1 - 1/|\mathcal{Y}|) \approx \prod_{i=1}^{Q-1} e^{-i/|\mathcal{Y}|} = e^{\sum_{i=1}^{Q-1} i/|\mathcal{Y}|}.$$

- Följaktligen gäller $e^{\sum_{i=1}^{Q-1} i/|\mathcal{Y}|} \approx 1 - \epsilon$.
- Med lite omskrivningar får vi $Q \approx \sqrt{2|\mathcal{Y}|\log\frac{1}{1-\epsilon}}$.
- För $\epsilon = 1/2$ får vi då $Q \approx 1.17\sqrt{|\mathcal{Y}|}$.

Mittuniversitetet
MID SWEDEN UNIVERSITY

Formell behandling av hashfunktioner

- Detta kallas födelsedagsparadoxen.
- Detta betyder att om $|\mathcal{Y}| = 365$, då är den 50 % sannolikhet att kollisionsalgoritmen finner en kollision då $Q = 23$.
- Om en fingeravtrycksläsare lagrar fingeravtryck som 20 bitar långa bitsträngar, då är det 50 % sannolikhet att två personer kan identifiera sig som varandra vid 1000 användare.
- Vi kan finna kollisioner med 50 % sannolikhet för en hashfunktion som har 256 bitars meddelandesammandrag med 2^{128} gissningar.

Formell behandling av hashfunktioner

MD5 Fullständigt knäckt; kan finna godtyckliga kollisioner, snabb att beräkna [se LD05].

SHA1 Finns attacker som antyder att det går att finna kollisioner med $Q = 2^{69}$, borde vara $Q = 2^{80}$.

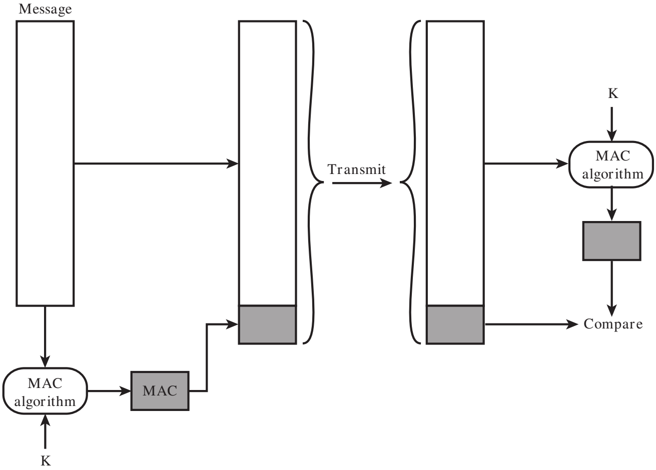
SHA256 Inga attacker som är märkbart lägre än $Q = 2^{128}$.

SHA512 Inga attacker som är märkbart lägre än $Q = 2^{256}$.

Message Authentication Code (MAC)

- Vi har sett att både symmetrisk och asymmetrisk kryptering kan användas för att signera kod.
- Dock uppstår problem om vi använder exempelvis ECB som mode of operation.
  - Byt ordning på blocken.
  - Ta bort vissa block.
- För detta ändamål skapar vi MAC.

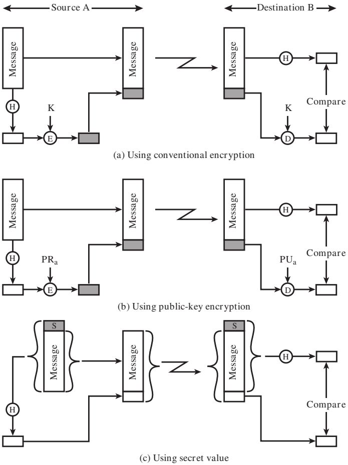
Hashfunktionsbaserade MAC



Figur: En översikt av en enkel MAC. Bild: [Sta13].

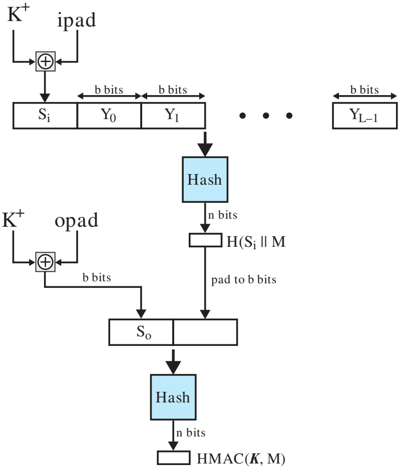
Mittuniversitetet
MID SWEDEN UNIVERSITY

Hashfunktionsbaserade MAC



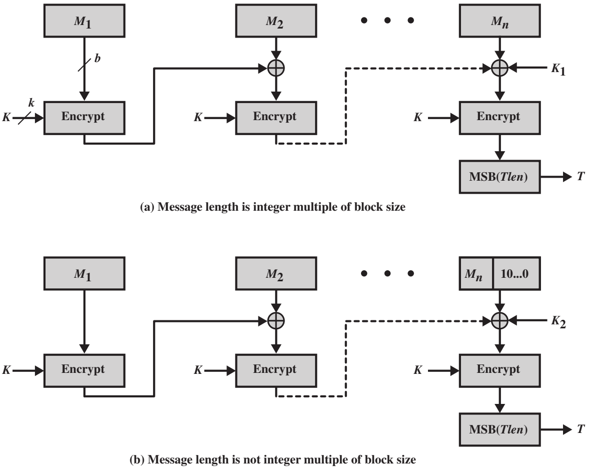
Figur: Exempel på olika former av MAC. Bild: [Sta13].

Hashfunktionsbaserade MAC



Figur: Hashbaserad MAC kallad HMAC,
$\mathrm{HMAC}(K, M) = h[(K^+ \oplus \mathit{opad}) \,||\, h[(K^+ \oplus \mathit{ipad}) \,||\, M]]$. Bild: [Sta13].

MAC baserade på blockchiffer

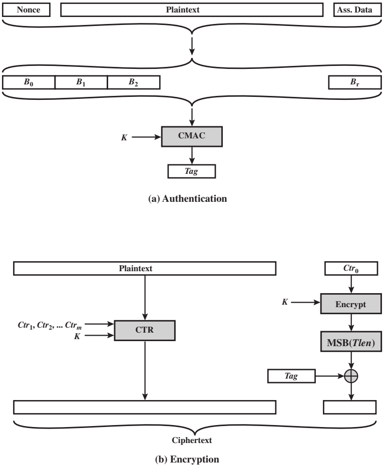


Figur: En schematisk översikt av CMAC. Bild: [Sta13].

Chiffer med autentisering

- Counter with Cipher Block Chaining Message Authentication Code (CCM).
- Är ett mode of operation för kryptering med autentisering.

Chiffer med autentisering



Figur: En schematisk översikt av CCM. Bild: [Sta13].

Referenser I

[LD05] Stefan Lucks och Magnus Daum. *Hash Collisions (The Poisoned Message Attack)*. 2005. URL: `http://th.informatik.uni-mannheim.de/people/lucks/HashCollisions/`.

[Sta11] William Stallings. *Cryptography and network security : principles and practice*. 5. ed., International ed. Upper Saddle River: Prentice Hall, 2011. ISBN: 0-13-705632-X (pbk).

[Sta13] William Stallings. *Network security essentials : applications and standards*. 5. utg. International Edition. Pearson Education, 2013. ISBN: 978-0-273-79336-6.

[xkc] xkcd. *Security*. URL: `https://xkcd.com/538/`.

Mittuniversitetet
MID SWEDEN UNIVERSITY