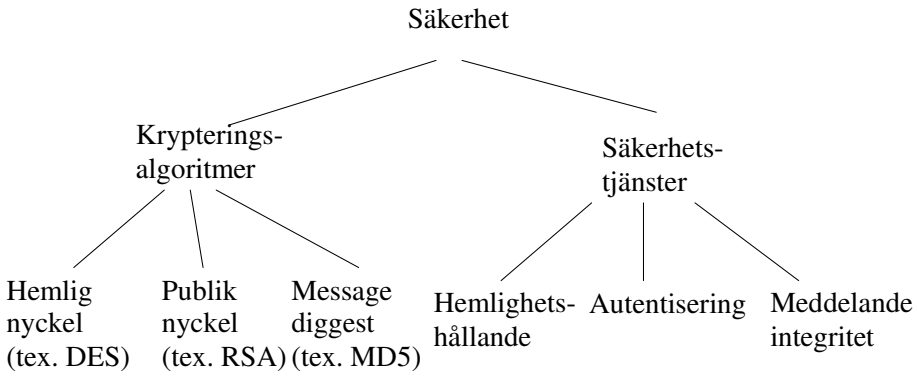
Datasäkerhet och integritet

OH-2 v1

- Kryptering
- Hashing
- Digitala signaturer
- Certifikat & PKI

Taxonomi



Hur säkra nätet?

- Effektiv lösenordshantering – grunden för allt säkerhetsarbete
- En fördel med lösenord är att de flesta system har det inbyggt
- Lösenord är ett bra skydd om:
  - Användarna väljer bra lösenord
  - Skyddar sitt lösenord och inte avslöjar det för någon
  - Ändrar lösenordet periodiskt
- Bra lösenord? Minst 8 tecken långa, helst över 12, gärna 15
  - Varierade tecken som bokstav, nummer och symboler
  - Ej personnamn eller vardagliga fraser
  - Med t.ex. 7 tecken från alfabetet (AaBb…Zz) så får man 52^7 + 52^6… = 1048,229,971,169 miljarder kombinationer, vilket **en** snabb dator knäcker på ungefär en vecka med ”brute force”
  - Det finns botnets eller programvaror (http://ophcrack.sourceforge.net) som knäcker detta på några minuter/timmar!

Behov av säker e-post?

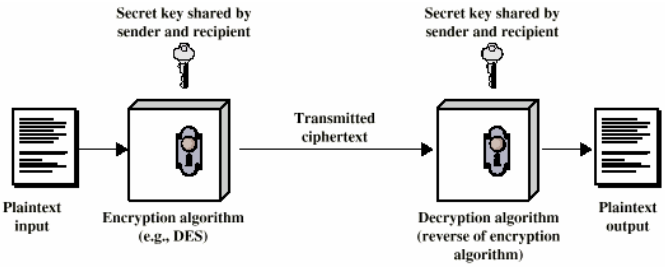
- Vykort!?
- E-post används till all typ av information
  - Reseräkningar, internbeställningar
  - Kärlekstjafs
  - Viktig kommunikation med kunder/leverantörer
- Problem
  - E-post är store-and-forward, och går underliga vägar (hoppar mellan servrar)
  - Forwardering av mail från företagskonto till Hotmail
  - Feladressering
  - Medveten förfalskning
  - Massutskick, bilagor...

Kryptering och dekryptering

- Meddelandet i ”cleartext” omvandlas till oläsbar ”ciphertext” med hjälp av krypteringssystem och nyckel
- Dekryptering reverserar förloppet med rätt nyckel
- Exempel på enkla krypteringsalgoritmer
  - Rövarspråket
    - Efter varje konsonant lägger man till ett ”o” och samma konsonant igen, exempel: Hej -> Hohejoj
  - Ceasarrullning eller ROT(n)
    - Varje bokstav i meddelandet flyttas ett fixt antal steg (n) framåt, exempel med fyrstegsrullning: HEMLIG -> LIQPMK
- Det finns i huvudsak två olika krypteringsmetoder
  - Symmetrisk (privat/hemlig) eller asymmetrisk (publik) kryptering

Symmetrisk kryptering och dekryptering

- Vid symmetrisk kryptering används samma krypteringsalgoritm och (privata/hemliga) nyckel av sändare och mottagare
- Används oftast då säkerhetskraven är mycket höga
- Nackdelen är säker skötsel av nycklar i ett distribuerat system
- Det finns två typer av symmetrisk kryptering
  - Ström chiffer
  - Block chiffer



Symmetrisk kryptering och dekryptering

- Block chiffer:
  - Meddelandet delas upp i fixerade block av bitar som behandlas i olika substitutions boxar
  - Om längden på data är mindre än blocklängden måste man ”padda” (fylla i) data
  - Implementeras mest i mjukvara
    - AES, Advanced Encryption Standard
    - DES, Digital Encryption Standard
    - 3DES (tripple DES)
    - RC5, RC6
    - Blowfish, Twofish
    - IDEA (International Data Encryption Algorithm)

Symmetrisk kryptering och dekryptering

- Ström chiffer
  - Behandlar meddelandet som en ström av bitar/bytes och gör matematiska funktioner för dem individuellt
  - Används när längden på sändningen av data inte är känd, t.ex. i trådlösa tillämpningar
  - Implementeras oftast i hårdvara
    - Exempel är t.ex. RC4, A5/1
- Attacken för att knäcka båda metoderna är ”brute force” (om algoritmen är bra)
  - Pröva dekrypteringsnycklar tills output är läsbar
  - Generellt gäller att en längre nyckel gör det svårare att knäcka krypteringen (precis som för ett lösenord)
  - Om man knäckt nyckeln kan man läsa alla meddelanden

Symmetrisk kryptering och dekryptering
RC4 Ström chiffer – Ron Rivest

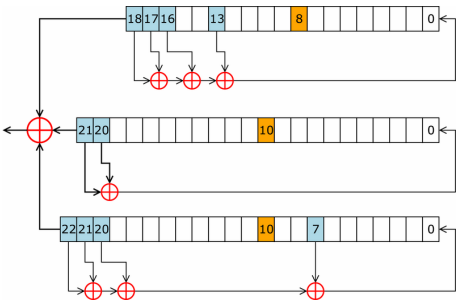
- Mycket använd i trådlösa nätverk
  - TLS/SSL
- Populärt därför att det är
  - Snabbt
  - Enkelt att implementera
  - Svårt att attackera
- Algoritmen använder nyckeln för att generera ett slumptal, vilket är exklusivt OR:at med cleartexten för att producera ciphertexten

XOR	Input 1	
Input 2	0	1
0	0	1
1	1	0

- ClearT = 1011, key = 1001
- ClearT XOR key = 0010
- Decrypt
- CipherT XOR key = 1011

Symmetrisk kryptering och dekryptering
http://en.wikipedia.org/wiki/Stream_cipher

- Ström chiffer
- A5/1 som ofta används för mobiltelefoner
- Var hemligt men ”reverse enginerades” 1999
- http://en.wikipedia.org/wiki/A5/1

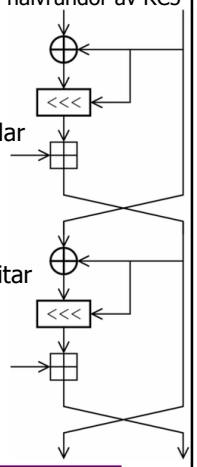


Symmetrisk kryptering och dekryptering

http://www.eff.org/Privacy/Crypto/Crypto_misc/DESCracker/

- DES och triple DES
  - 64 bitars blockstorlek och 56 bitars nyckel, 16 rundor
  - Det finns teoretiskt 2^56 kombinationer vilket är för klent idag, därför används istället triple DES
  - 3DES meddelanden krypteras 3 ggr, helst med 3 olika nycklar
- AES (Rijndael algoritmen)
  - Syftet var att skapa motstånd mot alla kända attacker
  - Enkel design och arbeta snabbt på de flesta plattformar
  - 128 bitars block, variabel nyckellängd, 128, 192 eller 256 bitar
- RC5 (RSA Data Security patent – Ron Rivest)
  - Variabel blocklängd, 32, 64 och 128 bitar
  - Krypterar i två ordblock (halva blocklängden)
  - Variabelt antal möjliga förändringar (rounds) 0 - 255
  - Variabelt antal bitar i nyckeln 0 – 2040 bitar

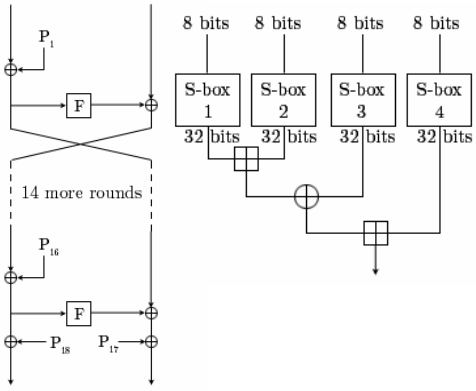
2 halvrundor av RC5



Symmetrisk kryptering och dekryptering – Blowfish

- Helt fri öppen standard, tänkt som efterträdare till DES
- 64 bitars blockstorlek (32 – 448 bitars nyckel, default 128 bitar)
- Baserad på Feistel chiffer (identisk kod för kryptering och dekryptering)
- Använder stora nyckelberoende substitutions boxar
- Innehåller en uppslagstabell som ger instruktioner hur bitar skall förändras eller flyttas runt.
- Nyckeln som används i dekrypteringsprocessen dikterar vilka S-boxar som används och vilken ordning.

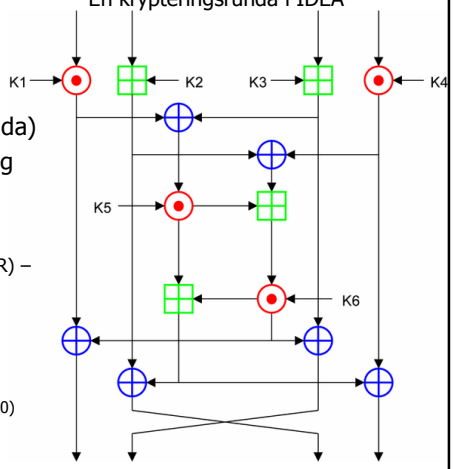
En krypteringsrunda i Blowfish



Symmetrisk kryptering och dekryptering - IDEA

- Tänkt som efterträdare till DES, fritt för icke-kommersiell anv.
- 64 bitars blockstorlek (128 bitars nyckel)
- Använder 8 identiska transformer (en runda) och en output transform (halvrunda)
- Processen för kryptering och dekryptering är liknande
- IDEA får mycket av sin säkerhet genom att väva samman operationer från olika grupper – modulo additioner och multiplikation, bitvis exklusiv OR (XOR) – som är algebraiskt ”inkompatibelt” i viss mån
- I detalj är det dessa operatorer som opererar på 16 bitar åt gången
  - Bitvis exklusiv OR (blått cirkelkors)
  - Addition modulo 216 (grönt multikvadrat)
  - Multiplikation modulo 216+1, där alla null word (0x0000) tolkas som 216 (röd cikelpunkt).

En krypteringsrunda i IDEA



Asymmetrisk (publik) kryptering och dekryptering

- ***Två olika*** nycklar används för kryptering och dekryptering
  - Den **privata** nyckeln är hemlig
  - Den **publika** nyckeln kan läsas av vem som helst
- Funktionen är möjlig tack vare att det alltid finns ett ***nyckelpar*** som matchar varandra
  - Irrelevant i vilken ordning eller med vilken nyckel kryptering och dekryptering sker
  - Samma nyckel kan ***inte*** kryptera och dekryptera i nyckelparet
  - Meddelanden som t.ex. krypterats med en publik nyckel kan endast dekrypteras med den matchande privata nyckeln
- Svagheter är ett lättare forcerat skydd och att mer datakraft krävs vid kryptering/dekryptering

Asymmetrisk (publik) kryptering och dekryptering

- Iom. att publik kryptering är långsamt i jämförelse mot privat kryptering brukar man ofta endast kryptera en symmetrisk nyckel som kallas för ”sessions nyckel”, vilken används för att dekryptera meddelandet
- Attacken är matematisk - faktorering av stora tal
  - Bryt ner ett heltal i dess faktorer, t.ex. 15 = 5 och 3
  - Faktorerna måste vara primtal, enkelt för små tal men otroligt svårt för stora (kallas NP-kompletta i matematiken, ingen lösning utom att prova alla kombinationer)
- Militärt värde
  - Exportrestriktioner råder t.ex. från USA. Max 40 bits symmetrisk och 512 bitar assymetrisk
  - http://www.enges.org/anders.enges/html/ecomm4.html

Asymmetrisk (publik) kryptering och dekryptering

Bob

Bob har fått två nycklar. En kallas för Public Key (grön), den andra Private Key (röd)

Vemsomhelst kan få Bobs publika nyckel, men den privata behåller han själv

Ove Sven Alice

Båda Bobs nycklar kan kryptera data och den ena nyckeln kan dekryptera vad den andra nyckeln skrev för data och vice versa

Asymmetrisk (publik) kryptering och dekryptering

Alice kan kryptera ett meddelande till Bob med sin publika nyckel hon fått från Bob

Vemsomhelst kan få tillgång till Alices krypterade meddelande, men meddelandet är värdelöst utan Bobs privata nyckel

Alice

Hej Bob! Vad sägs om en öl på krogen i kväll?

Det är happy hour hela natten!

HNFmsEm6Un
BejhhyCGKOK
JUxhiygSBCEiC
0QYIh/Hn3xgiK

Bob

HNFmsEm6Un
BejhhyCGKOK
JUxhiygSBCEiC
0QYIh/Hn3xgiK

Hej Bob! Vad sägs om en öl på krogen i kväll?

Det är happy hour hela natten!

Asymmetrisk kryptering och dekryptering, algoritmer

- RSA algoritmen (Rivest, Shamir, Adleman, 1978) PKCS#3
  - Välj två stora primtal (bara delbara med 1) P och Q. Hitta deras produkt n = PQ
  - Beräkna x = (P-1)(Q-1)
  - Välj ett relativt prima (gemensam nämnare med x som är 1) tal **e** mindre än **x** och större än 1. Formeln blir då: d = 1/e mod x där **e** är den publika och **d** den privata exponenten.
  - Den publika nyckeln är paret **n** och **e** och den privata nyckeln är paret **n** och **d**. Faktorerna P och Q måste hållas hemliga eller förstöras
- DSA algoritmen (Digital Signature Algorithm, 1991)
  - US Federal Goverment standard
  - Fungerar som RSA (primtal, exponenter, modulo etc.)

RSA exempel

modulo ger resten vid en heltalsdivision, ex 5 mod 4 = 1

- **Den publika nyckeln är (*e, n*) och den privata nyckeln är (*d, n*). Krypteringsfunktionen är:**
  - **encrypt(*m*) = *m^e* mod *n* => *m^7* mod 55 - där *m* är ”plaintext” och m < n.**
- **Dekrypteringsfunktionen är:**
  - **decrypt(*c*) = *c^d* mod *n* => *c^23* mod 55 - där *c* är ”ciphertext”.**

***p* = 5**

- Första primtalet (förvaras säkert eller raderas)

***q* = 11**

- Andra primtalet (förvaras säkert eller raderas)

***n* = *p*q* = 55**

- modulo (blir publik & privat)

***x* = (*p-1)*(q-1)* = 40**

***e* = 7, talet måste ha endast en gemensam nämnare med x som är 1**

- publik exponent (blir publik)

Vi beräknar *d* = 1/*e* mod *x*

***7*d* mod *40* = 1 => *d* = 23 (7*23/40=1)**

- privat exponent (förvaras hemligt)

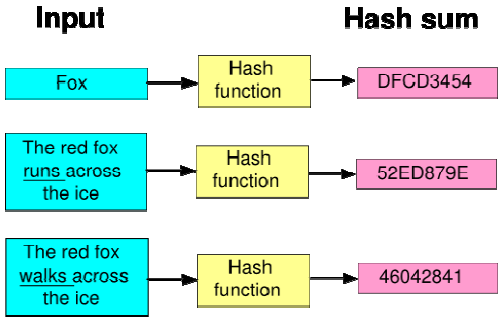
- För att kryptera plain-text värdet 8, beräknar vi
  - encrypt(8) = 8^7 mod 55 = 2097152 mod 55 = 2
- För att dekryptera chiffertext värdet 2, beräknar vi
  - decrypt(2) = 2^23 mod 55 = 8388608 mod 55 = 8

Asymmetrisk kryptering och dekryptering, algoritmer

- Diffie-Hellman, PKCS#3 (krypterar bara i en riktning)
  - Ett protokoll utvecklat för att utväxla nycklar över osäkert medium
  - Känsligt för ”man-in-the-middle”-attacker (avlyssning av datatrafiken) eftersom ingen autentisering krävs av parterna.
- ElGamal
  - Kan användas både för kryptering och signering
  - Anses vara något långsamt, liknar annars Diffie-Hellman & DSA
- Digitala signaturer och certifikat
  - Är ett stort användningsområde för asymmetrisk kryptering i samband med autentiseringstjänster
  - Motsvarar en underskrift på papper
  - Principen är grovt att om ett känt värde krypteras av avsändaren (M, meddelandet) och tolkas likadant hos mottagare efter dekryptering anses avsändarens identitet vara styrkt

Hashalgoritm eller hashfunktion

- Används till att undersöka indata i ett meddelande eller fil enligt en viss algoritm och ge ett tal som utdata, denna process kallas för ”hashing”
- Det är mycket osannolikt att två olika input skall ge samma output (hash summa)



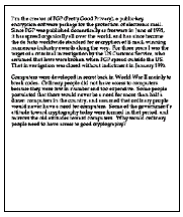
MD (Message Digest) hashalgoritmer

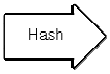
- Ex: MD5, SHA-X (Secure Hash Algorithm)
- Används främst för att kontrollera integriteten med en checksumma av ett meddelande M
  - Ej modifierad eller förvanskat på resan
- Message digest 5 implementeras som en 128 bitars **en-vägs** hash-funktion H(M)
- Enkelt att beräkna H(M) men mycket svårt att bestämma M så att H(M) = h för given kod h
- MD5 representeras oftast med 32 hexadecimala tal (128 bitar)
- MD5("The quick brown fox jumps over the lazy dog") = 9e107d9d372bb6826bd81d3542a419d6

Vad är en digital signatur?

Med en privat nyckel och den rätta mjukvaran kan Bob sätta digitala signaturer på dokument och annan data. Den digitala signaturen ”stämpeln” är väldigt svår att förfalska, minsta lilla förändring i datat kommer att märkas

Bob





För att signera dokumentet bearbetar Bobs mjukvara datat till några få rader med hexadecimala tal i en process som kallas hashing.

Dessa tal kallas för message digest

Vad är en digital signatur?

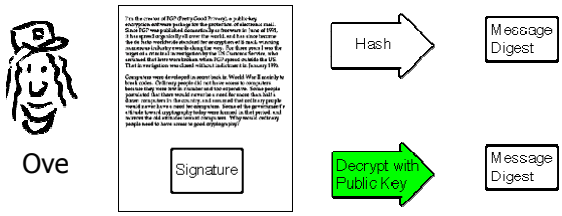
Bob mjukvara krypterar sedan message digest med sin privata nyckel och resultatet är den digitala signaturen

Bob

Slutligen så lägger Bobs mjukvara till den digitala signaturen i dokumentet. All data som blivit hashat har nu signerats

Vad är en digital signatur?

Bob skickar nu dokumentet till Ove…

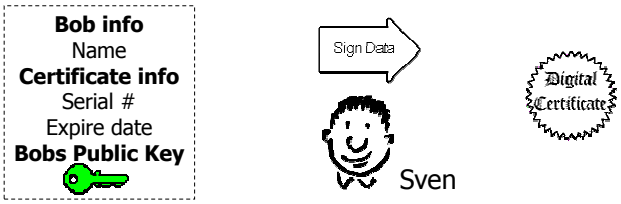


Först så dekrypterar Oves mjukvara signaturen (med Bobs publika nyckel) tillbaka till message digest. Om det fungerade så bevisar det att det var Bob som signerade dokumentet (Bobs privata nyckel)

Oves mjukvara hashar sedan dokumentet till message digest, om denna message digest är samma message digest som när signaturen dekrypterades har inte det signerade datat ändrats sedan det blev krypterat

Vad är en digital signatur?

För andra verkligen skall veta att Bob är Bob så kan Bob be ett certifikat-utfärdningscenter (Sven) låta tillverka ett digitalt certifikat åt honom



Bobs bekanta kan nu kontrollera Bobs betrodda certifikat så att det verkligen är Bobs publika nyckel som tillhör Bob

Bobs bekanta accepterar dessutom inte andra signaturer än de som har ett certifikat utfärdat av Svens certifikat-utfärdningscenter

Läs mera: http://www.youdzone.com/signature.html

MIME (Multipurpose Internet Mail Extensions)

- Internet Standard för formatet på e-post, kallas ibland för SMTP/MIME
- e-post sänds med protokollet SMTP som endast stödjer 7-bits ASCII
- MIME definierar mekanismer för att sända e-post med andra språk och inkluderade bilagor av alla slag
- Används också för att kunna sända viss data inbäddat i andra protokoll, t.ex. av HTTP (MIME types)
- ”Transfer encodings” definierar hur man representerar 8-bits binär data med 7-bits ASCII
- ”Content-type” ger meddelandets typ, t.ex. text/plain
- ”Multipart messages” hanterar flera content types

PGP/MIME & OpenPGP

http://www.bretschneidernet.de/tips/secmua.html

- PGP (Pretty Good Privacy)
  - Phil Zimmerman 1991
  - Privat nyckel på hemlig nyckelring
  - Bygger på ”web of trust”, personliga förhållanden (tillit)
    - www.thawte.com har kommersiell tjänst för detta
  - PGP blev kommersiellt kring 1996
- OpenPGP & GnuPG (Gnu Privacy Guard)
- PGP/MIME för att kryptera **hela** meddelandet byggde på
  - RFC 1991, PGP Message Exchange Formats
  - RFC 2015, MIME Security with Pretty Good Privacy
- Nu gäller
  - RFC 2440, OpenPGP Message Format
  - RFC 3156, MIME Security with OpenPGP

PGP/MIME & Secure (S/MIME) Vx

- Är en standard baserad på MIME och utvecklad av RSA Data Security Inc. för att sända säker e-post
- Certifikatbaserad = behöver PKI (denna kan dock vara enkel)
- S/MIME är likt OpenPGP och det äldre PGP/MIME (i stort sett samma funktioner som nedan) men inkompatibelt med dessa
- Signerad e-post i S/MIME formatet innehåller en signaturbilaga i PKCS#7-formatet samt en hash från originalmeddelandet signerat med sändarens privata nyckel och sändarens certifikat
- Krypterad e-post genereras med mottagarens publika nyckel
  - Meddelandet krypteras dock först med en symmetrisk nyckel, denna nyckel krypteras också i sin tur med mottagarens publika nyckel och sänds med meddelandet
  - Om meddelandet sänds till flera mottagare så krypteras den symmetriska nyckeln separat av alla mottagares publika nyckel
- S/MIME stödjer att meddelanden först signeras med sändarens privata nyckel och sedan krypteras med mottagarnas publika nycklar (signering och kryptering).

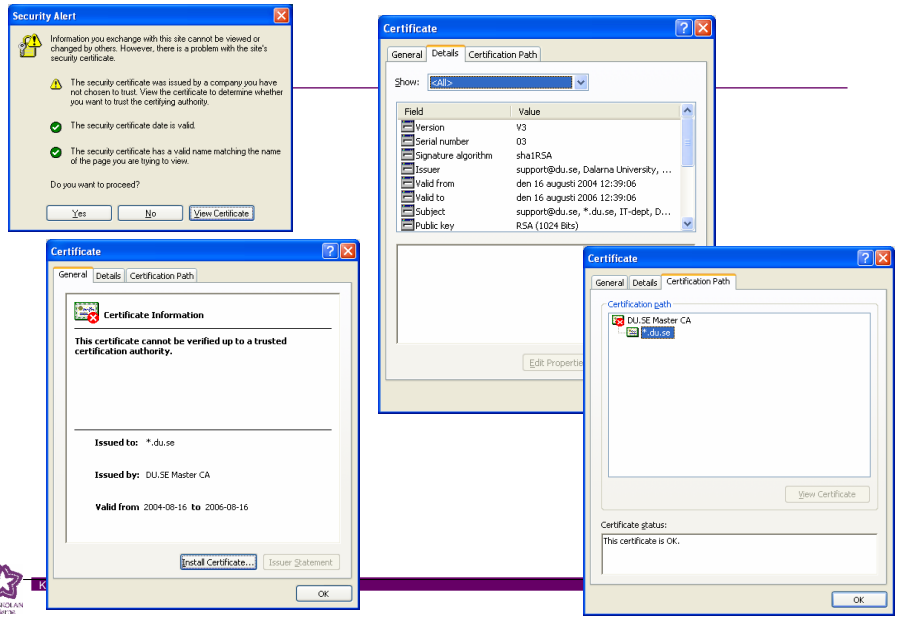
Certifikathanteringen

- Hur kan man vara säker på att en viss nyckel hör ihop med en viss person?
- Certifieringsinstansens (CA) roll är att knyta innehavares identitet till en uppsättning nycklar
- Certifikatet innehåller bl.a. följande information
  - Ett objekt (X.500 format)
  - Vem som utfärdat certifikatet
  - Objektets publika nyckel
  - Certifikatets giltighetstid
  - Hashat värde av hela innehållet
- Certifikatet lagras av innehavaren efter att certifikatet offentliggjorts

Asymmetrisk kryptering server - klient exempel

- Detta system bygger på två krypteringsnycklar
  - Public key. Krypteringsnyckeln som används för kryptering
  - Private key. Krypteringsnyckeln som används för dekryptering
- De används på följande sätt:
  1. Klienten tar kontakt med en server.
  2. Under en handskakningsprocess enas dessa om krypteringsmetod.
  3. Klienten får tillgång till serverns *Public key* som måste vara verifierad av en oberoende instans (t.ex. Verisign).
     Detta gör att klienten kan vara säker på att servern är den den utger sig för att vara.
  4. Klienten genererar ett tillfälligt Public/Private par och ger servern till gång till dess *Public key*
  5. Om klienten har ett **certifikat** installerat används detta för att autentisera klienten.
  6. Allt som klienten sänder till servern (inklusive URL paths) krypteras med serverns *Public key* som bara servern kan dekryptera.
  7. Allt som servern sänder till klienten krypteras med klientens *Public key* som bara klienten kan dekryptera.
- Denna kommunikation anses som säker…

Ex. att logga på fronter (HTTPS)



Certifikathanteringen

- CRL (Certifikation Revocation List)
- 3 ledande PKI-standards
  - ITU-T X.509
  - RSA Security, PKCS#x
  - PKIX Working Group (X.509), RFC xxx
- Version
  - Tre olika versioner 1-3
- Serial number
  - Endast unikt för CA
- Signature
  - Information om krypto och hash

Certifikathanteringen

- Issuer
  - Information om utfärdaren
- Validity
  - Certifikatets giltighetstid
- Subject
  - Identifiering av certifikatets innehavare
- Subject public key info
  - Info om den publika nyckeln
- Issuer unique identifier
- Subject unique identifier
- Extensions
  - Tillägg endera i fördefinierad form eller egna

PKI (Public Key Infrastructure)

- Lösning eller falsk säkerhet?
- PKI är en infrastruktur som möjliggör
  - Stark autentisering – lösenord + certifikat
  - Säker e-post – signering/kryptering + certifikat
  - Digitala signaturer
  - Säker fjärråtkomst
  - Filkryptering
  - Kodsignering – säkra tjänster på www
  - Informationsintegritet och Upphovsrättskydd
  - Säkra webbserveruppkopplingar
  - Aktiva kort - lagring av certifikat
- Syftar till att efterlikna den ”vanliga världen”

PKI grundsten och komponter

- Baseras på förtroende/tillit mellan kund och utgivare CA (Certification Authority)
  - Risker med detta?
- PKI-komponeter
  - CA (Certification Authority)
  - RA (Registration Authority)
  - Certifikat/PKI-server
  - Certifikatlager (Certificate Repository)
  - Certifikatvalidering (Certificate Validation)
  - Nyckel-återställning/arkivering

Upprätthålla PKI, aktiva kort

- Hur länge skall ett certifikat gälla?
- När skall certifikat återkallas?
- CRL (Certificate Revocation List)
- Idag är aktiva kort mycket vanliga
  - SIM-kort, bankautomatkort etc.
- PC/SC (Personal Computer/Smart Card)
  - PKI på det aktiva kortet
  - Samverka med applikationer via cryptoAPI
- Omfattande arbete innan PKI kan införas i en organisation