



Kryptering



Repetition
to slide 29

- Kryptering
- Hashar
- Digitala signaturer, PKI
- Krypto FS
- Lösenord

<http://en.wikipedia.org/wiki/Portal:Cryptography>

Kryptering och dekryptering

- Lösenordsskyddad vs. lösenordskrypterad
- 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

Enigma

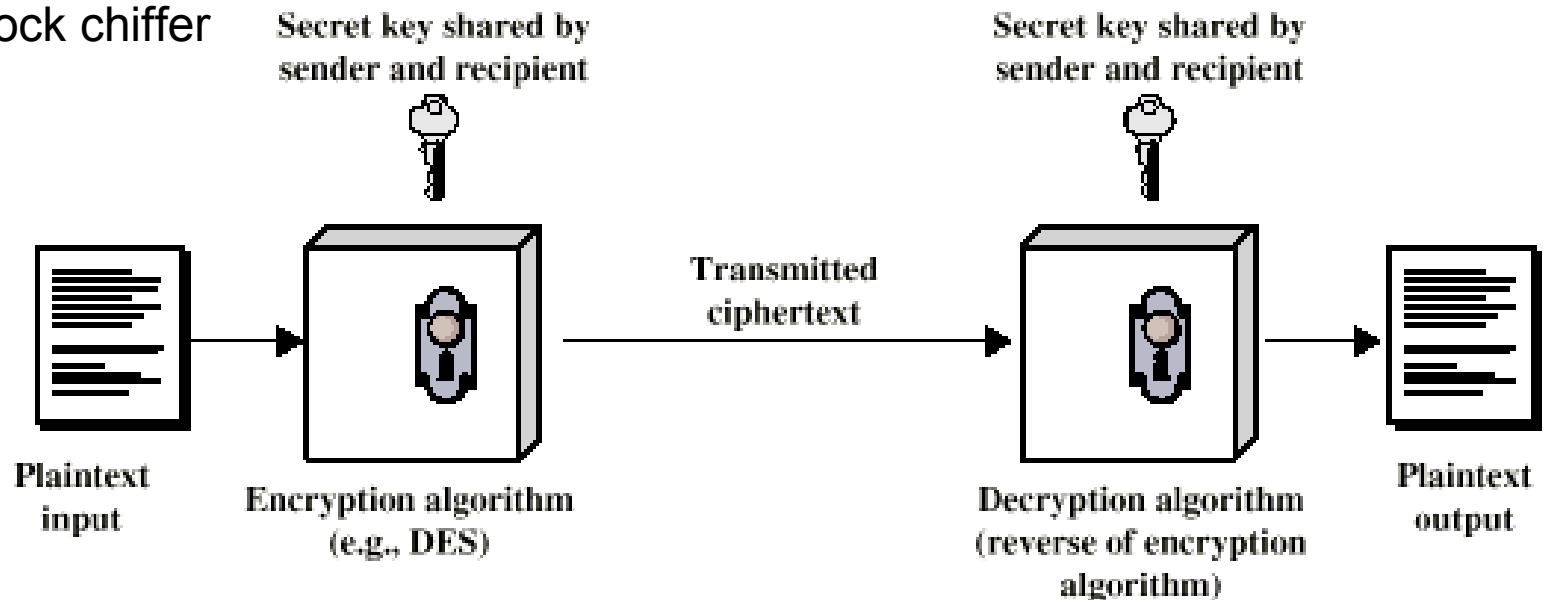
- Tyskarna, WW II
- Elektromekanisk
- Tekniska museet i Stockholm har ett exemplar!
- Alan Turing



- http://sv.wikipedia.org/wiki/Enigma_%28krypteringsmaskin%29

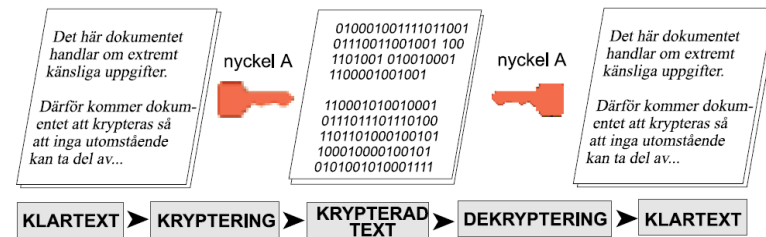
Symmetrisk kryptering och dekryptering

- 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 (triple DES)
 - RC5, RC6
 - Blowfish, Twofish
 - IDEA (International Data Encryption Algorithm)



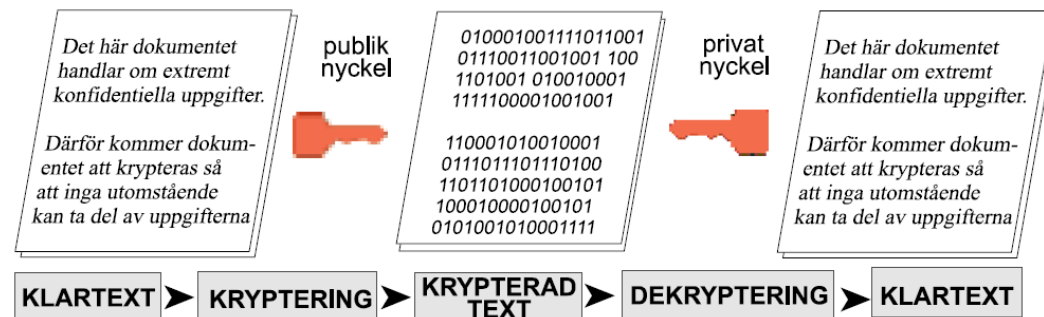
Symmetrisk kryptering och dekryptering

A	B	---	A	XOR	B
0	0	----	0		
0	1	----	1		
1	0	----	1		
1	1	----	0		

- Ström chiffer
 - Behandlar meddelandet som en ström av bitar/bytes och gör matematiska funktioner för dem individuellt t.ex. XOR
 - Används när längden på sändningen av data inte är känd, t.ex. i trådlösa tillämpningar
 - Implementeras oftast i hårdvara men även i mjukvara
 - Exempel, RC4 (WEP-WLAN, MS Office, RDP), A5/1 och A5/2 (GSM)
- 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 - i bästa fall)
 - Om man knäckt nyckeln kan man läsa alla meddelanden

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 symmetrisk kryptering brukar man oftast endast kryptera en symmetrisk nyckel som kallas för "sessions nyckel", vilken sedan används för att dekryptera själva meddelandet
- En symmetrisk nyckel är ca: 4-10 ggr. så "stark" som en asymmetrisk
- Attacken mot publik kryptering är matematisk - faktorerering 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 tal (kallas NP-kompleta i matematiken – finns ingen lösning utom att prova alla kombinationer)
- Militärt värde
 - Exportrestriktioner har gällt tex. från USA. Max 40 bits symmetrisk och 512 bitar asymmetrisk

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, t.ex. e-mail

Alice kan kryptera ett meddelande till Bob med den publika nyckeln hon fått från Bob

Vemsomhelst kan få tillgång till Alices krypterade meddelande, men meddelandet är värdelöst (oläsbart) 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#1)
 - Välj två stora primtal (bara delbara med 1 och sig själv) P och Q
 - Hitta deras produkt $n = PQ$ **Formel: en/de-crypt(m) = m^(e/d) mod n**
 - Beräkna $x = (P-1)(Q-1)$
 - Välj ett "coprime" - relativt prima (endast en gemensam nämnare med x som är 1) tal e mindre än x och större än 1
 - Formeln för d blir då: $d = 1/e \text{ mod } x$ där e är den publika och d den privata exponenten
 - Den publika nyckeln är paret e och n och den privata nyckeln är paret d och n
 - Faktorerna P och Q måste hållas hemliga eller förstöras
 - <http://williamstallings.com/Crypt-Tut/Crypto%20Tutorial%20-%20JERIC.swf>
- DSA algoritmen (Digital Signature Algorithm, 1991)
 - US Federal Government standard
 - Fungerar som RSA (primtal, exponenter, modulo etc.)

RSA exempel

modulo ger resten vid en heltalsdivision, ex: $5 \bmod 4 = 1$

- Den publika nyckeln är (e, n) och den privata nyckeln är (d, n)
Krypteringsfunktionen är:
 - $\text{encrypt}(m) = m^e \bmod n \Rightarrow m^7 \bmod 55$ - där m är "plaintext" och $m < n$
- Dekrypteringsfunktionen är:
 - $\text{decrypt}(c) = c^d \bmod n \Rightarrow c^{23} \bmod 55$ - där c är "ciphertext"

$P = 5$

- Första primtalet (förvaras säkert eller raderas)

$Q = 11$

- Andra primtalet (förvaras säkert eller raderas)

$n = P \cdot Q = 55$

- Modulo (blir publik och privat)

$x = (P-1) \cdot (Q-1) = 40$

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

- e är publik exponent (publik åtkomst)

Vi beräknar $d = 1/e \bmod x \Rightarrow d \cdot e \bmod 40 = 1$

$d \cdot 7 \bmod 40 = 1 \Rightarrow d = 23$ ($23 \cdot 7 \bmod 40 = 1$)

- d är privat exponent (förvaras hemligt)

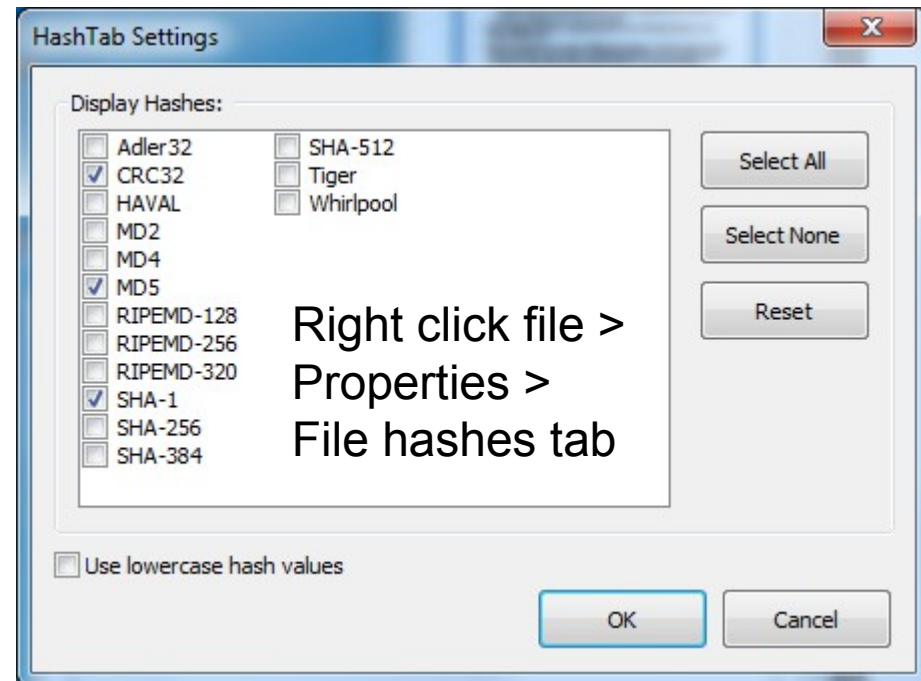
- För att kryptera plain-text värdet $m = 8$, beräknar vi
 - $\text{encrypt}(8) \Rightarrow$
 $8^7 \bmod 55 \Rightarrow$
 $2097152 \bmod 55 = 2$
- För att dekryptera chiffrertext värdet $c = 2$, beräknar vi
 - $\text{decrypt}(2) \Rightarrow$
 $2^{23} \bmod 55 \Rightarrow$
 $8388608 \bmod 55 = 8$

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 och DSA
- Digitala signaturer och certifikat
 - Är ett stort användningsområde för asymmetrisk kryptering i samband med autentiseringstjänster
 - Motsvarar en underskrift på papper

Hashes and control sums

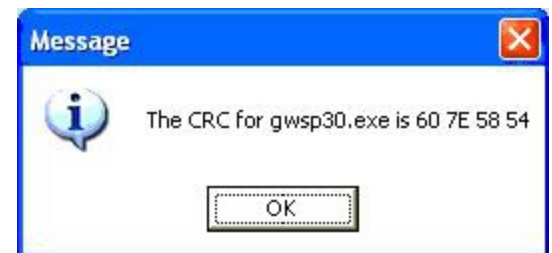
- The demands for an algorithm calculating hash sums are
 - It should not be possible to change or create another bit stream (a text for example) which give the same hash sum
 - It should not be possible to get hold of the text by knowing the hash sum
 - The algorithm must be sensitive for small changes in the original information so a different hash sum will be generated
 - The calculation algorithm should be fast
- The first demand is essential for the usability of control sums
- It should not be possible to create for example two documents with different text which give the same hash by for example inserting blank characters etc.
- Can you guess any other usage for hashes?



CRC (Cyclic Redundancy Check)

- A kind of simple hash function (bit length is to short and algorithm is to simple)
- CRC is popular since it is simple to implement in hardware and simple to analyze mathematically. But good enough to discover errors
- CRC-1 is used for parity bit checks in memory
- CRC-32 is used to control the integrity in network packets etc.

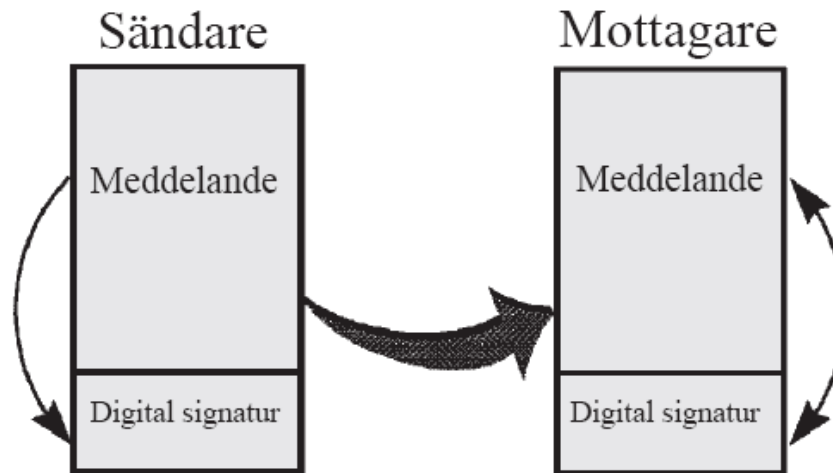
— http://en.wikipedia.org/wiki/Cyclic_redundancy_check



Digitala signaturer

- Principen är grovt att om ett känt värde hashas och krypteras av avsändaren (M, meddelandet) och tolkas likadant hos mottagare efter dekryptering anses avsändarens identitet och meddelande vara styrkt

1. Räkna ut en kontrollsumma på meddelandet med hjälp av en "hash"-algoritm.
2. Kryptera kontrollsumman med avsändarens hemliga nyckel. Det är nu en digital signatur.



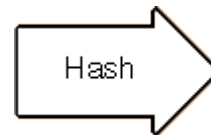
3. Räkna ut en ny kontrollsumma på meddelandet
4. Dekryptera signaturen, kontrollsumman, med avsändarens publika nyckel.
5. Jämför de två kontrollsummorna. De ska vara identiska!

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?

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



Bob

Message
Digest



Signature



Try the creator of PGP (Pretty Good Privacy), a public-key encryption software package for the protection of electronic mail. Since PGP was published December 19 at the end of 1991, it has spread organically all over the world, and has since become the de facto worldwide standard for encryption of e-mail, winning numerous industry awards along the way. For three years I was the target of a criminal investigation by the US Customs Service, who suspected that letters were broken when PGP opened outside the US. That investigation was closed without indication in January 1995.

Computers were developed in secret back in World War II mainly to break codes. Ordinary people did not have access to computers because they were still in their infancy and too expensive. Some people panicked that there would never be a need for more than half a dozen computers in the country, and assumed that ordinary people would never have a need for computers. Some of the government's officials looked cryptographically today were focused in that period, and across the old structure looked at computers. Why would ordinary people need to have access to good cryptography?

Signature

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...



Ove

This is the creator of PGP (Pretty Good Privacy), a public-key encryption software package for the protection of electronic mail. Since PGP was published, documentation as between in June of 1991, it has spread organically all over the world, and has since become the de facto standard for encryption of e-mail, winning numerous industry awards along the way. For those who I see the suggestion of a criminal investigation by the US Customs Service, who contacted that letter was broken when PGP passed outside the US. That investigation was closed without incident in January 1995.

Computers were developed in secret back in World War II solely to break codes. Ordinary people did not have access to computers, because they were not in vantage and too expensive. Some people pointed out that there would never be a need for mass mail in those computers in the country, and assumed that ordinary people would never have need for computers. Some of the participants of a slide show of cryptography today were located in that period, and someone the old attitude took a computer. "Why would ordinary people need to have access to good cryptography?"

Signature



Message Digest



Message Digest

Först dekrypterar Oves mjukvara signaturen (med Bobs publika nyckel) tillbaka till en 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 den dekrypterade signaturen har inte det signerade datat ändrats sedan signaturen blev krypterad, dvs. dokumentets innehåll har inte ändrats efter det lämnat Bob

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



Sven



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

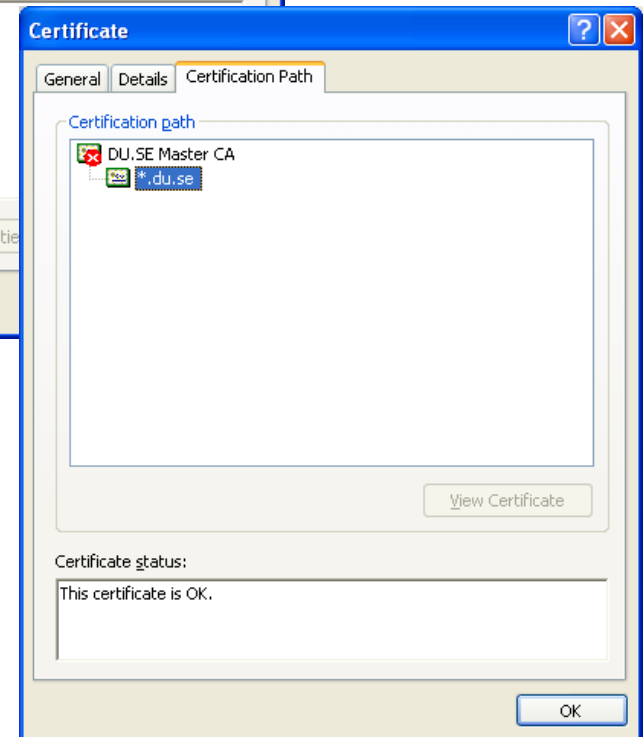
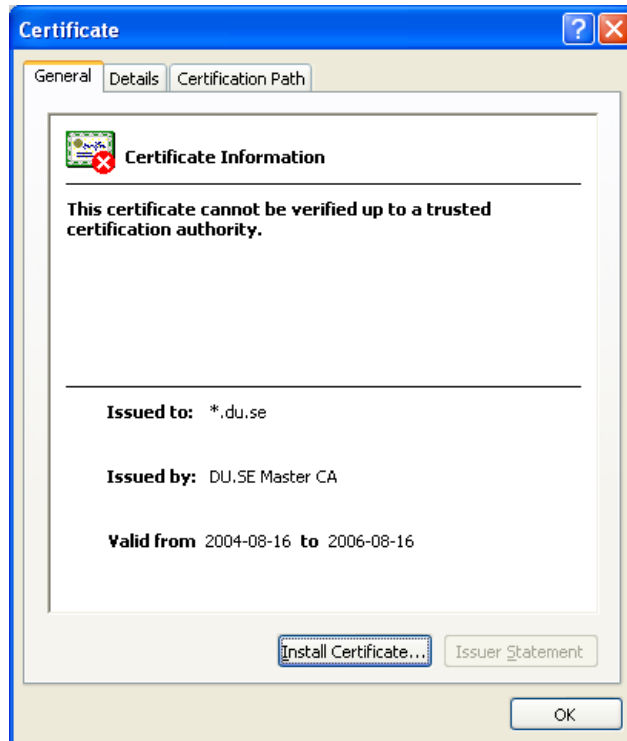
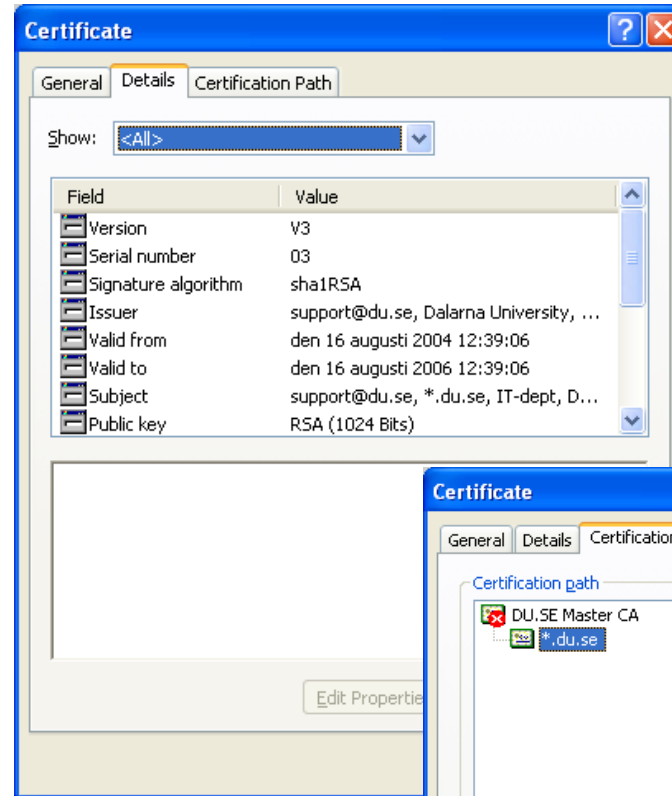
Bobs bekanta accepterar dessutom inte andra signaturer än de som har ett certifikat utfärdat av Svens certifikat-utfärdningscenter

Läs mera: http://en.wikipedia.org/wiki/Digital_signature

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
- IE > Tools > Internet Options > Content > Certificates

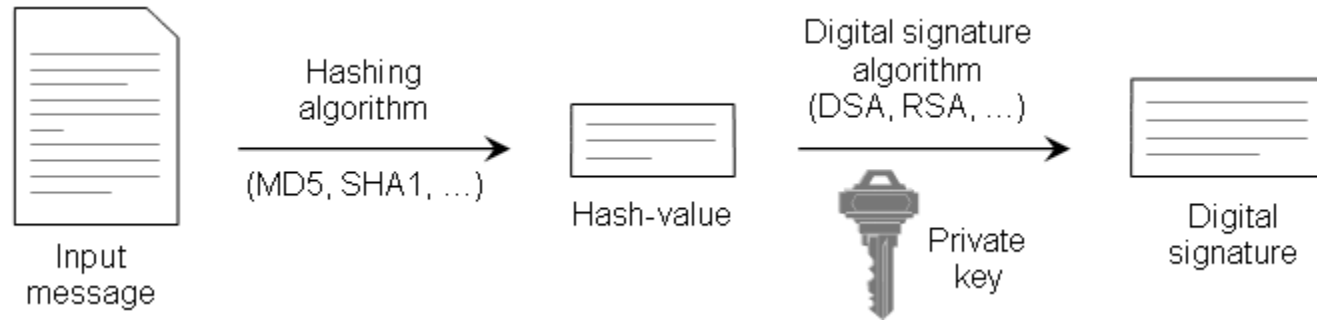
Ex. att logga på fronter (HTTPS)



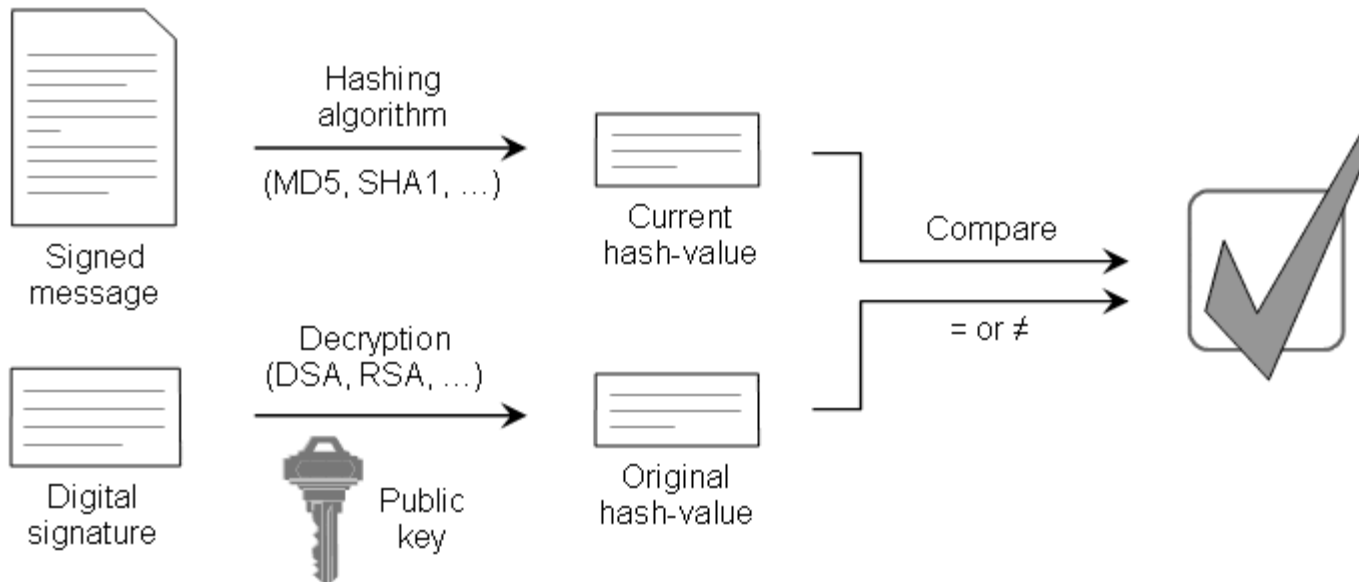
Exempel MS File Signature Verification (sigverif.exe)

http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sig_verification_tool.mspx?mfr=true

- Kontrollerar integriteten av systemets alla kritiska filer



- Varje fil har en digital signatur genererad med privata nyckeln
- Kontroll sker med den publika nyckeln mot filen och dess signatur



Verifiering av filer GNU/Linux

- Debian secure apt-get : NO_PUBKEY / GPG error
 - W: GPG error: <http://security.debian.org> etch/updates Release: The following signatures couldn't be verified because the public key is not available: NO_PUBKEY 010908312D230C5F
 - gpg --keyserver pgpkeys.mit.edu --recv-key 010908312D230C5F
 - gpg -a --export 010908312D230C5F | sudo apt-key add -
- Jämför med NIST's (National Institute of Standards and Technology) NSRL (National Software Reference Library) hashar
 - Reference Data Set version 2.16 was available march 2007, providing 11,778,827 unique SHA-1, MD5 and CRC32 values for 42,077,728 files.

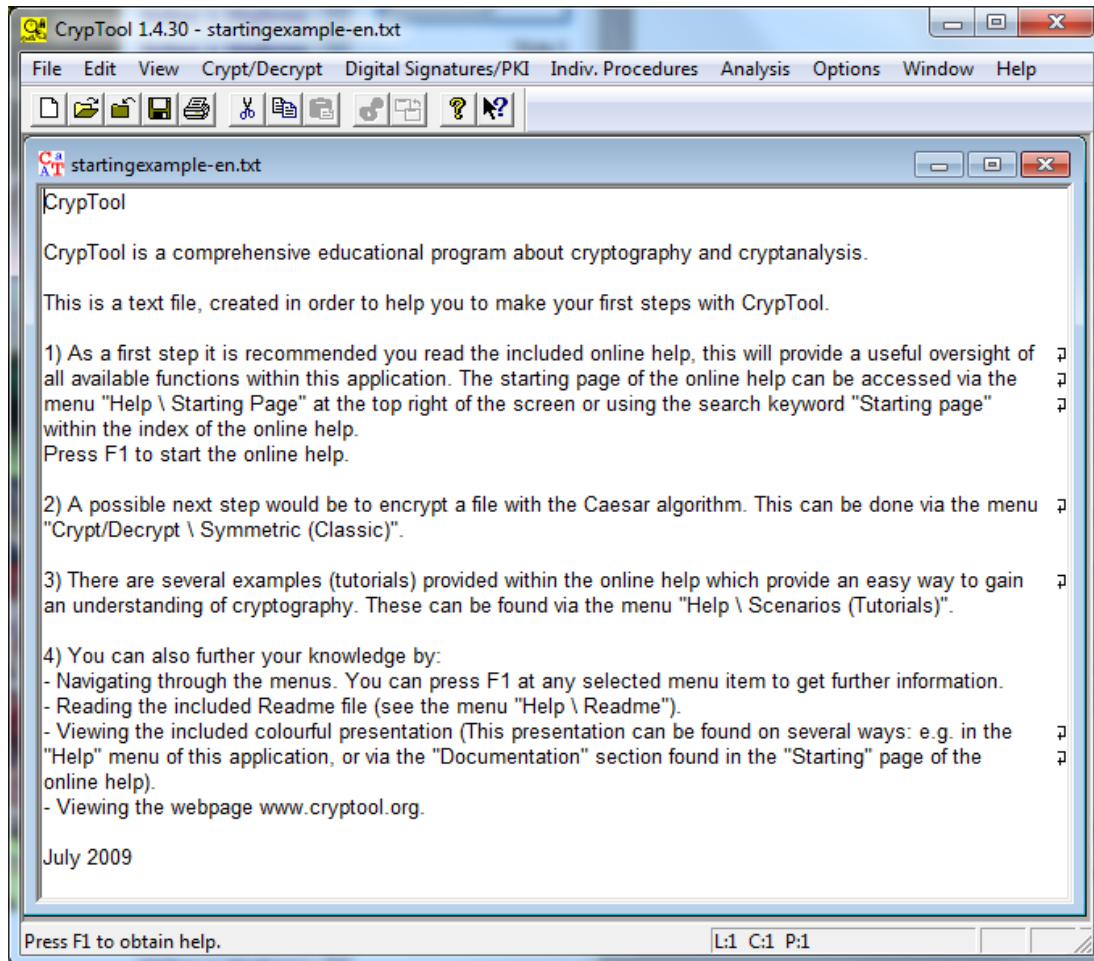
PKI (Public Key Infrastructure)

- 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"

CrypTool

Cryptography for the masses

Rekommenderas!



CrypTool (Freeware)

Crypt methods

Analysis

Visualisations

Etc. etc...

<http://www.cryptool.org/>

PGP/MIME och OpenPGP/GPG

<http://www.bretschneider.net.de/tips/secmua.html>



- PGP (Pretty Good Privacy)
 - Phil Zimmerman 1991
 - Privat och publik 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, köpt av Symantec 2010
- OpenPGP Alliance och 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 4880, OpenPGP Message Format / GnuPG
 - RFC 3156, MIME Security with OpenPGP

PGP/MIME och Secure (S/MIME) Vx

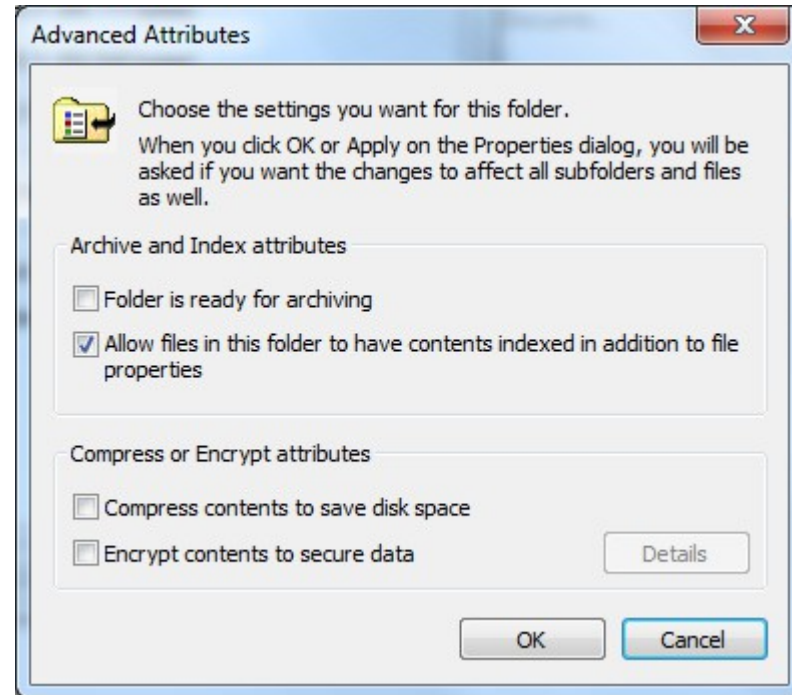
- Är en standard baserad på MIME och utvecklad av RSA Data Security Inc. för att sända säker e-post
- Certifikatbaserad = behöver PKI (denna kan dock vara enkel)
- S/MIME är likt OpenPGP och det äldre PGP/MIME (i stort sett samma funktioner som nedan) men inkompatibelt med dessa
- Signerad e-post i S/MIME 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)

Kryptografiska filsystem typer

- FDE(Full Disk Encryption)
 - Program/hårdvara i disken - kräver ATA/HDD password i BIOS (transparent)
- Volym kryptering
 - Använder en drivrutin och krypterar en hel (det mesta) eller en delmängd av fysisk disk (transparent)
 - TrueCrypt, PGPDisk, Secure File System (SFS), Linux CryptoAPI, ScramDisk, MS BitLocker
- Fil krypterare (en container)
 - Opererar via applikationslagret eller presentationslagret för end-to-end kryptering
 - PGP, GnuPG, Mac OS X Disk Utility, etc.
- Filsystem kryptering
 - Medger per fil eller per folder kryptering (transparent)
 - MS EFS, TCFS, CFS, CryptFS, EncFS, Mac OS X FileVault

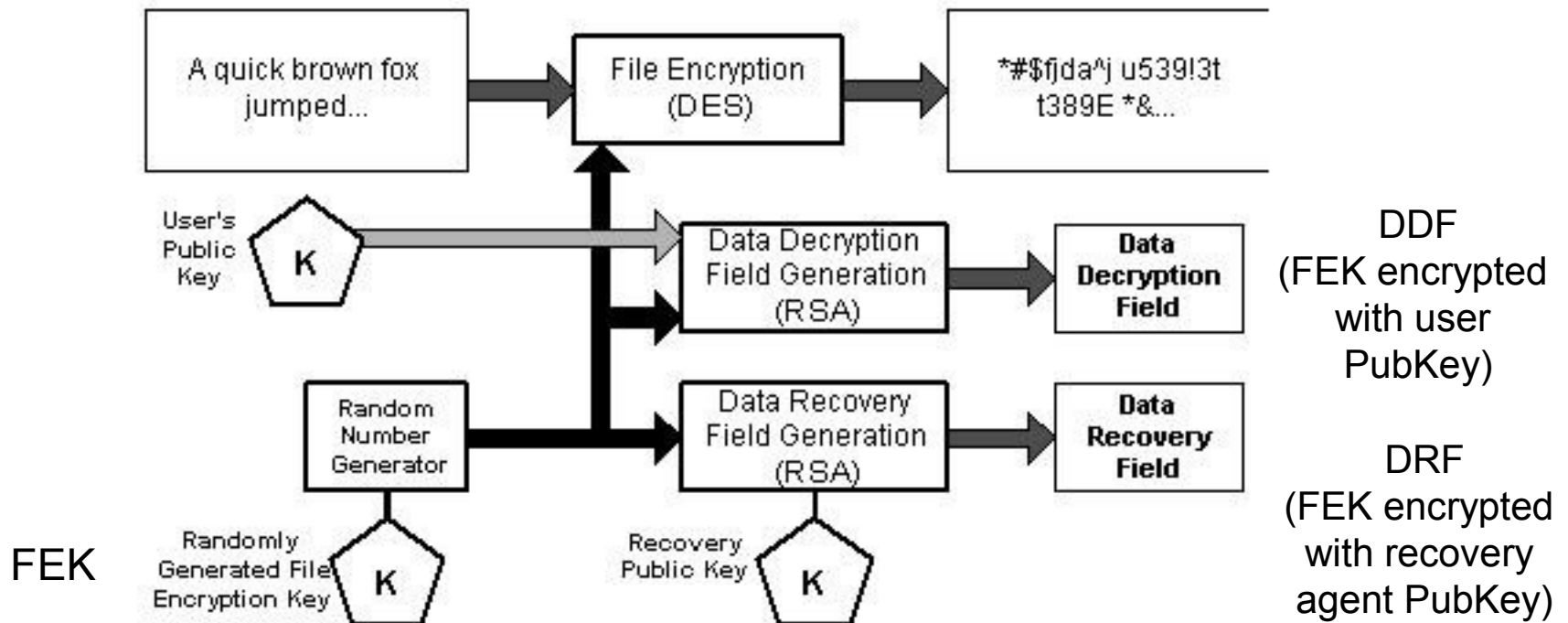
Windows EFS and Mac OS X

- För att kryptera en fil/mapp i Windows med EFS
 - Öppna Windows explorer
 - Högerklicka på fil/mapp och välj egenskaper
 - Klicka sedan på avancerat knappen
 - Kryptera fil/mapp genom att markera **Encrypt contents to secure data**
- Mac OS X
 - Disk Utility - krypterad arkivfil (.dmg) skyddad med ett lösenord
 - FileVault, OS X v10.3 och nyare, krypterar användarens /home mapp med användarens login password, ett master password kan sättas
- Se artiklar här
 - http://en.wikipedia.org/wiki/Encrypting_File_System
 - <http://en.wikipedia.org/wiki/FileVault>



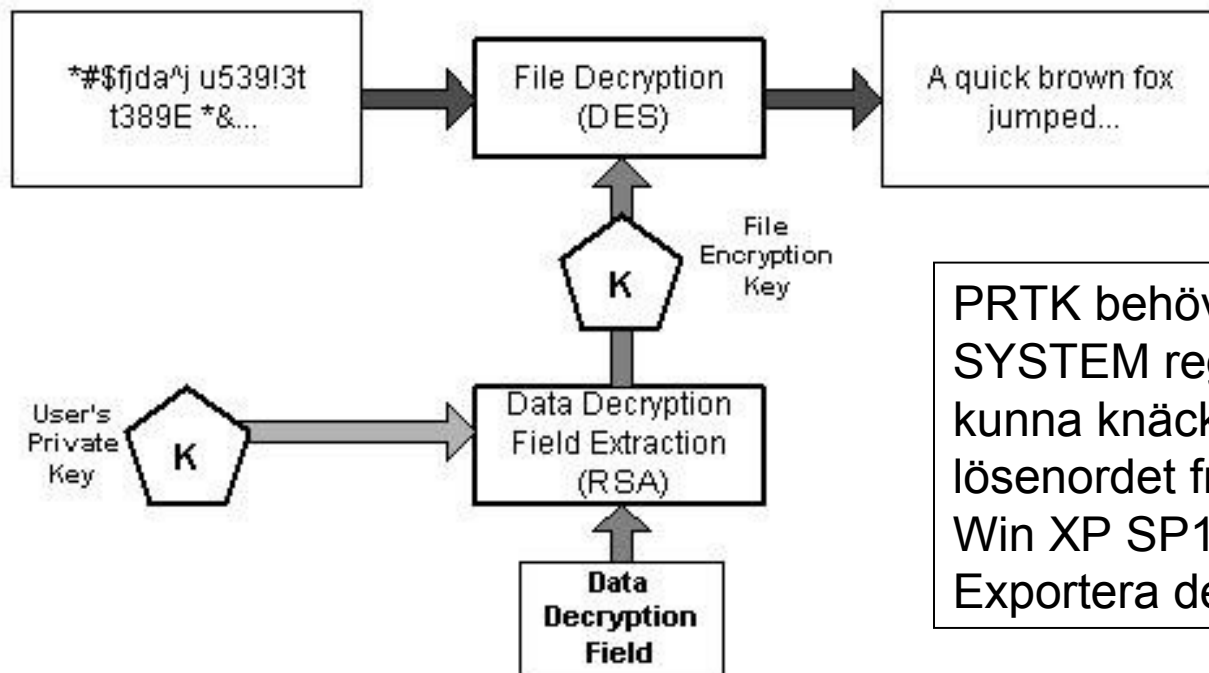
Microsoft's Encrypted File System (EFS)

- Public Key Infrastructure Schema
- Filer krypteras med en snabb symmetrisk algoritm och nyckel som är slumpmässigt genererad för varje ny fil
- Nyckeln (FEK) krypteras med användarens publika nyckel från dennes X.509 certifikat



Microsoft's Encrypted File System (EFS)

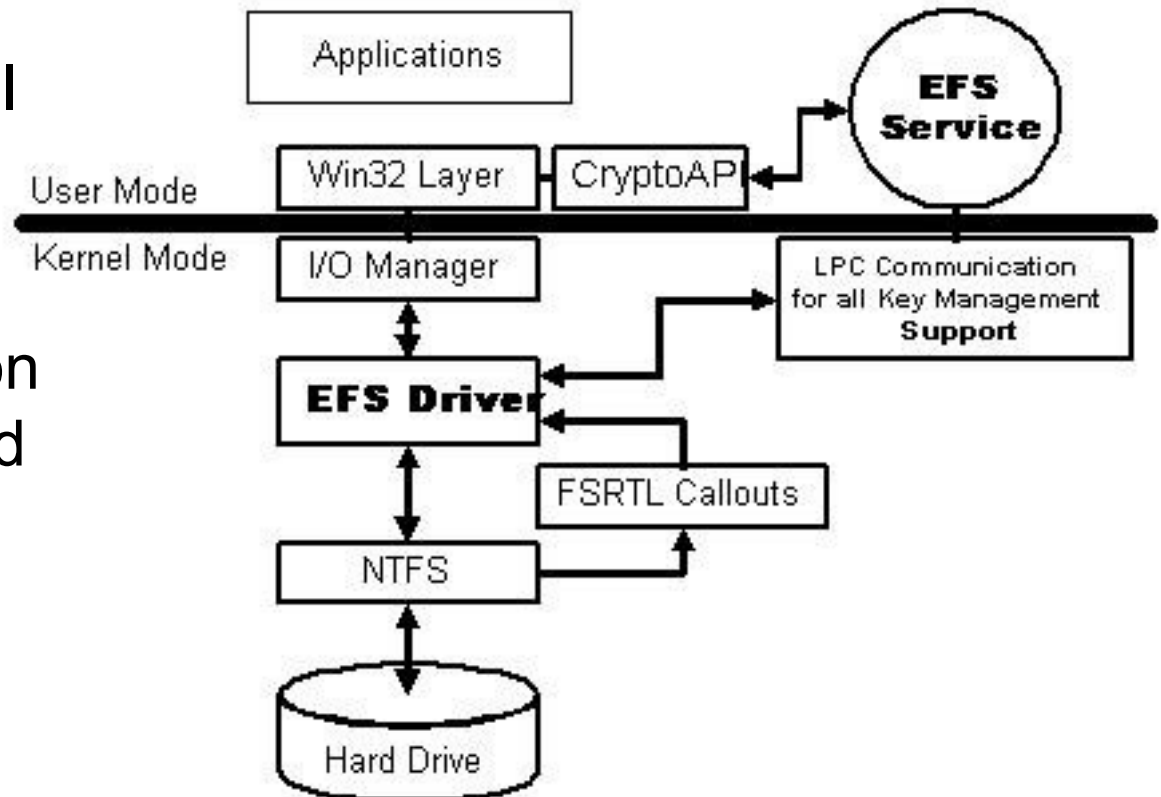
- Användarens privata nyckel används för att dekryptera filkrypteringsnyckeln (FEK)
- EFS tillåter att filkrypteringsnyckeln kan dekrypteras av recovery agents (DRF nycklar) om så konfigurerat (optional)
- EFS är designat för transparens



PRTK behöver SAM och SYSTEM reg. filerna för att kunna knäcka Windows login lösenordet från och med Win XP SP1
Exportera dem med FTK *

Microsoft's Encrypted File System (EFS)

- EFS arkitekturen kan delas upp i 4 kategorier
 - EFS drivrutin
 - Fil system run-time bibliotek (FSRTL)
 - EFS service
 - Olika Win32 API
- DESX encryption algorithm, based on a 128-bit encryption key



Sammanfattning EFS

- Kryptering av filer på hårddisken (Windows EFS)
 - Varje fil får en unik EFS-nyckel kallad, FEK (File Encryption Key)
 - FEK nyckeln krypteras i sin tur och skyddas av användarens publika nyckel som matchar användarens motsvarande EFS-certifikat
 - Krypterade nycklar sparas i filens EFS ADS \$Logged_UTILITY_Stream
 - FEK + DDF (Data Decryption Field) / DRF (Data Recovery Field)
 - FTK visar strömmen som \$EFS
 - Transparent för användaren
 - Lämnar filen/mappen hårddisken (NTFS) dekrypteras den!
- Data återställning (dekryptering)
 - På något sätt bör skyddade/krypterade filer/diskar vara möjliga att återfå i läsbart skick, eller..?
 - Borttappad nyckel, slutat arbetet, rättsliga krav
 - Användare som har access till denna process kallas för "recovery agents", vanligen superuser plus ett till konto (admin gruppen)

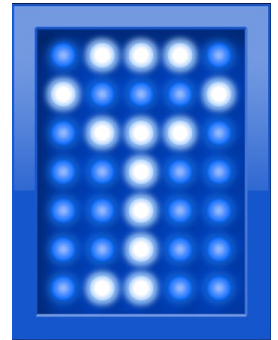
MS Vista/7 BitLocker



- OS Logical Volume Encryption
 - Encrypts the entire OS drive including the page and hibernation files
 - Requires a separate “system” partition to store boot files unencrypted
 - Bitlocker in Windows 7/2008 R2 allows encryption of removable drives
- Authentication mechanisms (may be combined)
 - User authentication mode – Pre boot PIN
 - Trusted Platform Module (TPM chip) – with built-in encrypted key
 - USB Key Mode which contains a startup key in order to boot OS
- Encryption Algorithm
 - AES-CBC (Cipher Block Chaining) with a specialized diffuser that improves the security against manipulation attacks, 128 bits
- Bra artiklar
 - <http://en.wikipedia.org/wiki/Bitlocker>
 - http://blogs.msdn.com/si_team/
 - <http://www.securityfocus.com/infocus/1889>

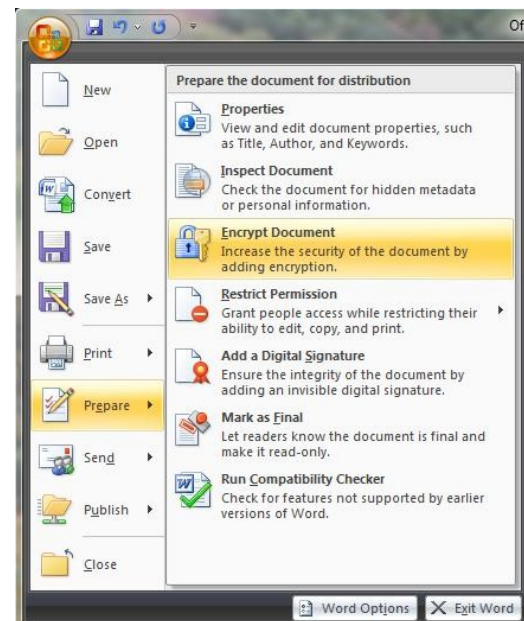
Andra FS krypteringsverktyg etc.

- Övriga OS
 - FreeBSD och GNU/Linux – Crypto API, TCFS (Transparent Cryptographic File System), CryptFS, EncFS
 - Andra UNIX – CFS (Cryptographic File System)
- TrueCrypt – flexiblaste och bästa krypteringsmjukvaran?
 - Multi platform, OS transparent, krypterar boot partition
 - Gömda containrar (plausible deniability)
 - <http://www.truecrypt.org>
- Några 3dje-parts krypteringsalternativ
 - Proffs
 - Jetico Bestcrypt, Pointsec, Compusec, Safeboot, PGP enterprise etc.
 - Enklare
 - PGP Desktop, Winguard, Secure File System (SFS), Folder Lock etc.
- Lista - http://en.wikipedia.org/wiki/Disk_encryption_software



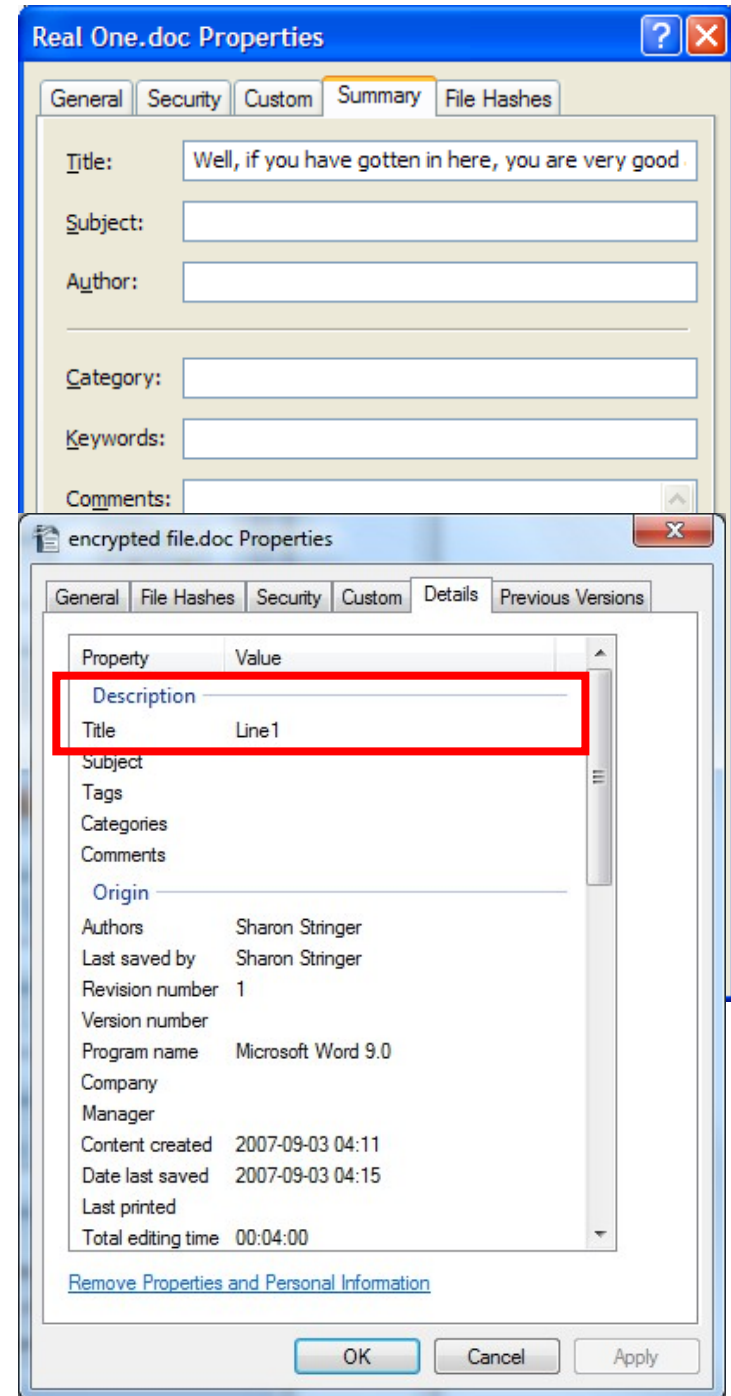
Andra krypteringsverktyg

- Skydd finns inbyggt i vissa applikationer som .zip, .rar, .7z, etc.
 - Enbart lösenordsskyddad fil (komprimerad)
 - Lösenordsskyddad fil och krypterat innehåll
- Olika officeprogramvaror kan även kryptera dokument
 - Office -97/2000 använde 40 bitar
 - Office XP/2003, default 40 bitar men kan ha stark kryptering
 - Office 2007, default 128 bit AES, 256 bit även möjligt
- Tidiga office ej tillräckligt skydd mot seriösare attacker
- Russian Password Crackers har mycket info om olika filtyper, algoritmer och svårigheter
 - <http://www.password-crackers.com>



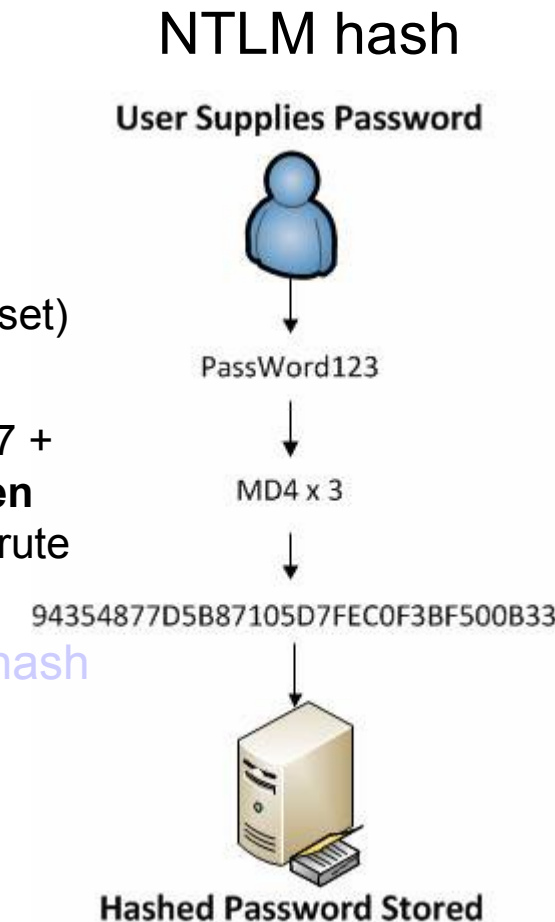
Krypterade Wordfiler

- Ett antal tecken i början av filen är inte krypterade!
 - Första raden/meningen i dokumentet
- Kolla summary > title under properties för dokumentet
- Details fliken för nyare OS än Windows XP
- Gäller det nyare OS även nyare office (>= 2007)?



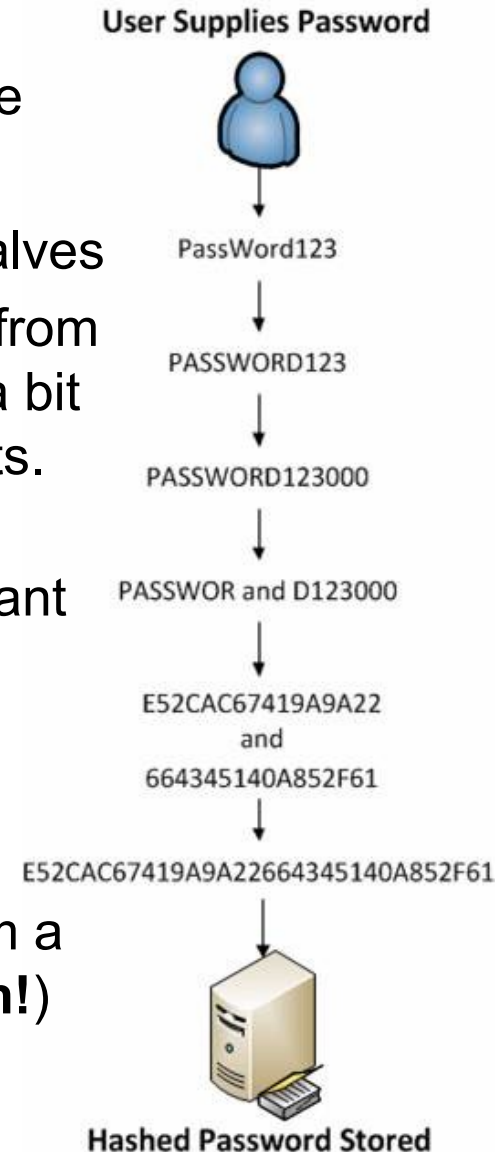
Passwords

- Effektiv lösenordshantering – en bra grund för gott 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 (charset)
 - Ej namn, vardagliga ord/fraser, ordsammansättningar etc.
 - Med tex. 7 tecken från alfabetet (AaBb...Zz) så får man $52^7 + 52^6 \dots = 1048,229,971,169$ miljarder kombinationer, vilket **en** normal single CPU dator knäcker på max ett par dar med "brute force"
- LM hash vs NTLM hash (http://en.wikipedia.org/wiki/Lm_hash)
 - <http://support.microsoft.com/kb/299656> (hur förhindra LM hashar)



The LM hash is computed as follows

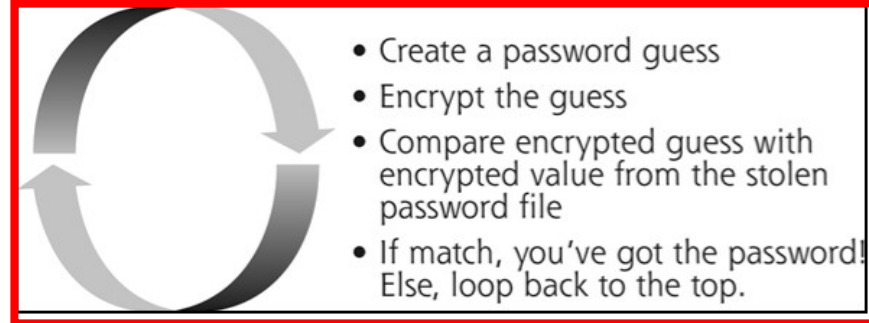
1. The user's ASCII password is converted to uppercase
2. This password is null-padded to 14 bytes
3. The "fixed-length" password is split into two 7-byte halves
4. These values are used to create two DES keys, one from each 7-byte half, by converting the seven bytes into a bit stream, and inserting a parity bit after every seven bits. This generates the 64 bits needed for the DES key
5. Each of these keys is used to DES-encrypt the constant ASCII string "KGS!@#\$\$%", resulting in two 8-byte ciphertext values
 - The DES CipherMode should Set to ECB, and PaddingMode should set to NONE
 - These two ciphertext values are concatenated to form a 16-byte value, which is the LM hash (**not a true hash!**)



Password attacks

http://en.wikipedia.org/wiki/Password_cracking

- Gissa lösenordet (nättjänster etc.)
- Lösenord är vanligen hashade eller krypterade
 - Svagheter i hash/krypterings-algoritm
- Dictionary eller word list
 - Eget eller standard med permutationer, kan ge mycket snabbt resultat!
 - Ledtrådar från brottsplatsen/arbetsplatsen/misstänkte
- Brute Force (keyspace)
 - Allt över "äkta" 64 bitars längd kräver speciella metoder
 - GPGPU, speciell hårdvara, distribuerad attack – botnets etc.
- Reset Attack (är även en DoS attack)
- Rainbow tables - http://en.wikipedia.org/wiki/Rainbow_table
 - Pre-computed keys i en speciell lookup table, knäcker det som tar veckor på minuter! (Accessdata har 40 bits - Office, PDF och LM/NT = 2,7 TB vardera)
 - Salt (en liten förlängning, 2-8 bytes) används för att försvåra rainbow attacker
 - hash = MD5 (password + salt)
- Social engineering



Entropi (lösenords entropi)

- Mäter ett lösenords styrka (oordningen)
- I ett helt slumpmässigt är varje tecken värt ca: 6,56 bitar
- Vid användarvalt ger första tecknet 4 bitar, tecken 2-8 ger 2 bitar, tecken 9-20 ger 1.5 och 21- ger 1 bit per tecken
- Entropi (bitar) tabell på olika längder av lösenord

Längd (tecken)	8	20	63
Användarvalt (helt fritt valt)	18	36	79
Användarvalt (enligt regler)	30	42	85
Slumpmässigt	52	131	413

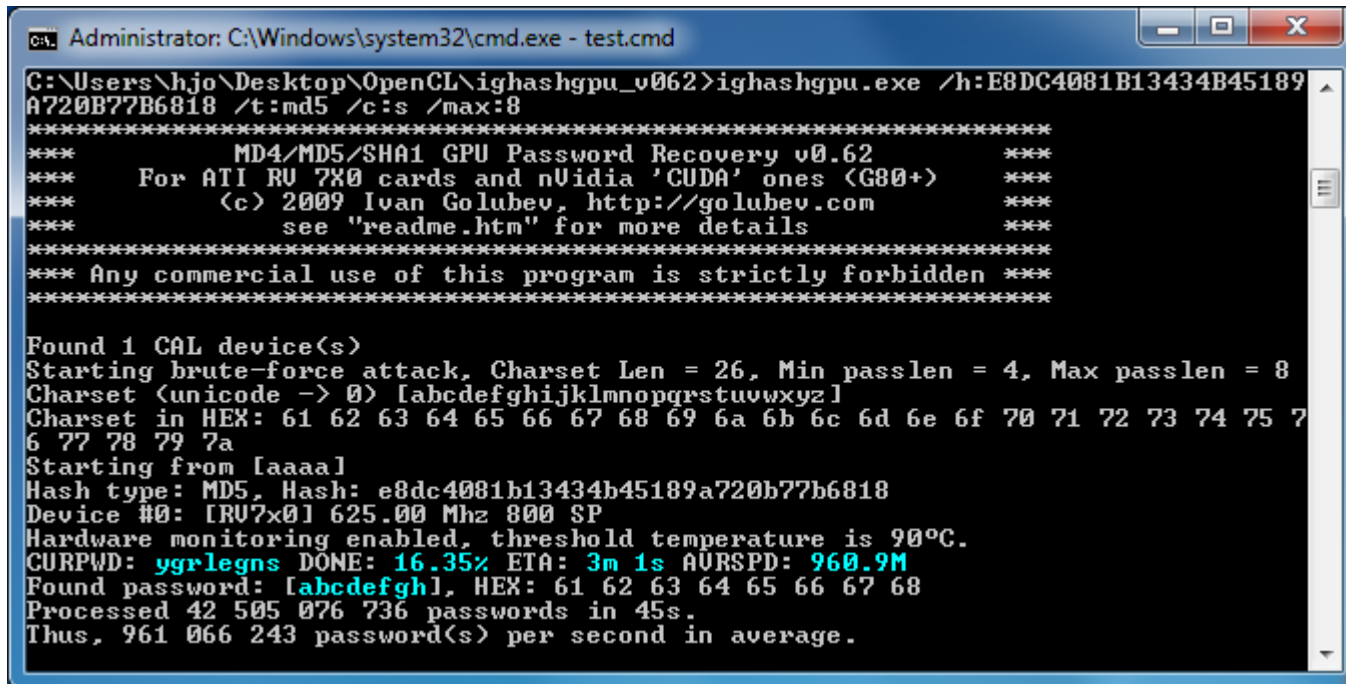
- Antalet variationer det finns av ett lösenord $2^{(\text{antal bitar})}$
- $(\text{antal möjliga tecken})^{(\text{antal tecken i lösen})}$ är i princip felaktigt eftersom ett kort lösenord kan vara komplext och ett långt lösenord kan vara av en lätt gissbar karaktär

http://en.wikipedia.org/wiki/Password_strength

GPGPU testing 1

- IGHASHGPU (Brook+/CUDA), recover/crack SHA1, MD5 & MD4 hashes
 - Supports salted hashes, NTLM, MySQL*, Oracle 11g, ..., etc.
 - Plain MD5, 8 chars, lowercase
 - Windows 7 x64, AMD Phenom x4 @ 2.2GHz
 - ATI 4850 - 800SP, Catalyst 9.12, Stream SDK v2.0
 - Count down time (ETA) started at almost 4 minutes

<http://golubev.com>
Intresting discussion
RAR GPU as well



```
Administrator: C:\Windows\system32\cmd.exe - test.cmd
C:\Users\hjo\Desktop\OpenCL\ighashgpu_v062>ighashgpu.exe /h:E8DC4081B13434B45189A720B77B6818 /t:md5 /c:s /max:8
*****
***      MD4/MD5/SHA1 GPU Password Recovery v0.62      ***
***      For ATI RV 7X0 cards and nVidia 'CUDA' ones (G80+) ***
***      (c) 2009 Ivan Golubev, http://golubev.com      ***
***      see "readme.htm" for more details              ***
*****
*** Any commercial use of this program is strictly forbidden ***
*****
Found 1 CAL device(s)
Starting brute-force attack, Charset Len = 26, Min passlen = 4, Max passlen = 8
Charset (unicode -> 0) [abcdefghijklmnopqrstuvwxyz]
Charset in HEX: 61 62 63 64 65 66 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76 77 78 79 7a
Starting from [aaaa]
Hash type: MD5, Hash: e8dc4081b13434b45189a720b77b6818
Device #0: [RV7x0] 625.00 Mhz 800 SP
Hardware monitoring enabled, threshold temperature is 90°C.
CURPWD: ygrlegns DONE: 16.35% ETA: 3m 1s AURSPD: 960.9M
Found password: [ygrlegns], HEX: 61 62 63 64 65 66 67 68
Processed 42 505 076 736 passwords in 45s.
Thus, 961 066 243 password(s) per second in average.
```

BF NT hash crack
7 char pass a-z,0-9
ETA:
Cain, 3.5h
IGHASHGPU, 1 min

**2012 GPU
generation is
around 10
times faster!**

Field-programmable gate array (FPGA)

- Tableau TACC1441 Hardware Accelerator \$3900
 - <http://www.tableau.com/>
 - AccessData PRTK support - TACC_Install.pdf
 - <http://www.digitalintelligence.com/products/rack-a-tacc/>
- Bruce Schneier - Secure Passwords Keep You Safer
 - <http://www.schneier.com/essay-148.html>
- NSA (at) Home
 - Breaks 800 hashes concurrently
 - <http://nsa.unaligned.org/>



Rack-A-TACK 2U module, \$20000

Rack-A-TACC™

Performance Data

