NORWICH UNIVERSITY™
Expect Challenge. Achieve Distinction.

Introduction to Cryptography

Gary C. Kessler

Vermont Internet Crimes Against Children Task Force
Burlington, VT

Norwich University
Northfield, VT

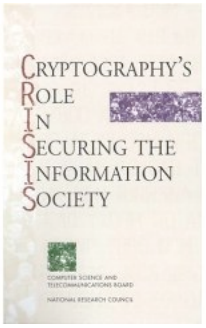
January 2012

http://www.garykessler.net/presentations

Overview

- The role of cryptography
- Types of cryptographic algorithms
  - » Hash functions
  - » Secret key cryptography
  - » Public key cryptography
- Putting them altogether... case studies in cryptography
- Trust models
  - » Certificates
  - » Sample applications: SSL, personal certificates

The Role of Cryptography



Cryptography

- The science of writing in secret codes
  - » Dates back to 1900 B.C. in Egypt (non-standard hieroglyphics); probably appears spontaneously soon after writing is developed
- Historically, two types of cryptography:
  - » Substitution
  - » Transposition/Permutation

Substitution Ciphers

- Most famous: *Caesar's Cipher*
  - » Shift each letter to the right by 3
- Today: *Rotation 13 (ROT13)* still found on Unix systems and Usenet to hide offensive text, puzzle solutions, passwords, etc.

```
PLAIN: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
rot13: n o p q r s t u v w x y z a b c d e f g h i j k l m
```


Transposition Cipher

Columnar transposition cipher

```
PLAINTEXT: CRYPTO TODAY IS A LOT MORE COMPLEX THAN IT USED TO BE

      CRYPTOTODA
      YISALOTMOR
      ECOMPLEXTH
      ANITUSEDTO
      BENSIVEGFL

ciphertext:   cyea bric neys oinp amts tlpu iool svtt
              eeeo mxdg dott farh olxy
```

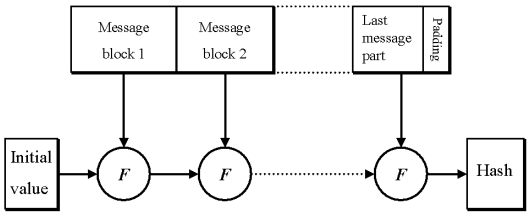
Cryptography Today

- Cryptography is necessary today in telecommunications when communicating over any untrusted medium
- Digital cryptography basically comes in three varieties:
  - » Hash functions (no key)
  - » Secret key cryptography (one key)
  - » Public key cryptography (two keys)

Secure Communications

- Secure communications requires:
  - » Authentication
  - » Message integrity
  - » Non-repudiation
  - » Privacy/confidentiality
  - » Key exchange

Hash Functions



Hash Functions

hash function

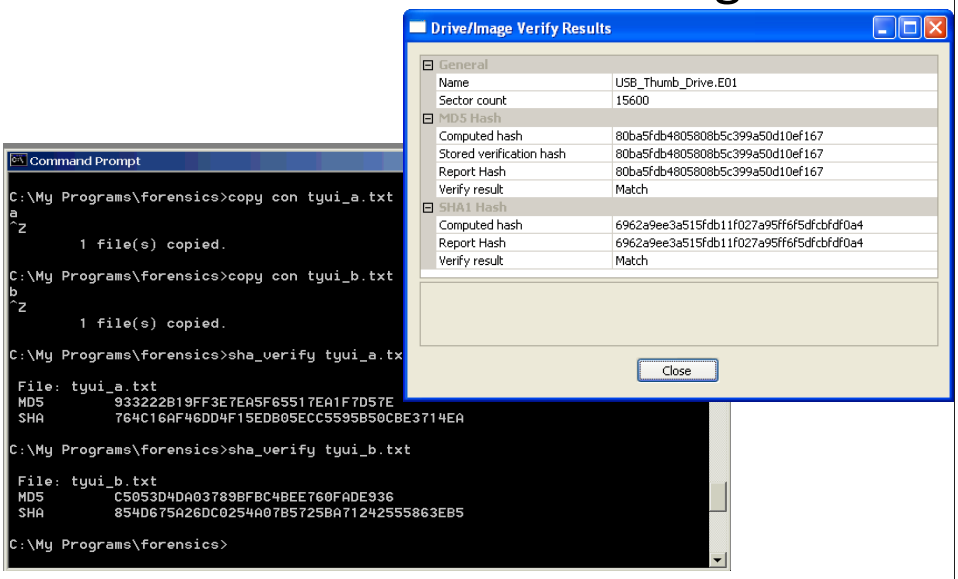
plaintext ⟶ ciphertext

- No key
  - » Plaintext (and length of plaintext) is not recoverable from the ciphertext
  - » Examples: HMAC, MD2, MD4, MD5, RIPEMD-160, SHA
  - » Also called *message digests* or *one-way encryption*
- Primary use: Message integrity

Hashing: UNIX Password File

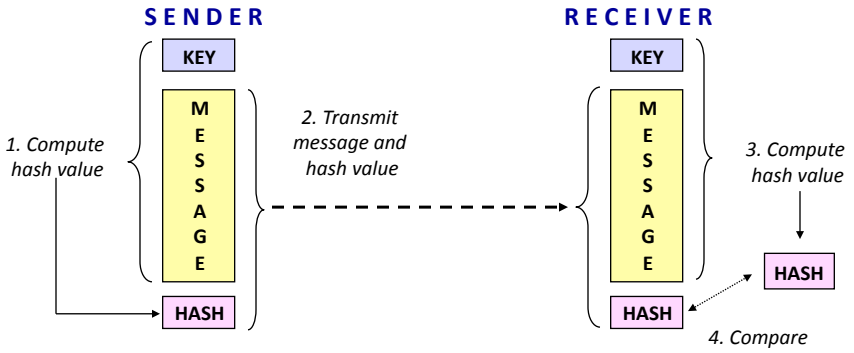
```
carol:FM5ikbQt1K052:502:100:Carol Monaghan:/home/carol:/bin/bash
alex:LqAi7Mdyg/HcQ:503:100:Alex Insley:/home/alex:/bin/bash
gary:FkJXupRyFqY4s:501:100:Gary Kessler:/home/gary:/bin/bash
todd:edGqQUAaGv7g6:506:101:Todd Pritsky:/home/todd:/bin/bash
sarah:Jbw6BwE4XoUHo:504:101:Sarah Antone:/home/schedule:/bin/bash
josh:FiH0ONcjPut1g:505:101:Joshua Kessler:/home/webroot:/bin/bash
```


SHA and MD5 Hashing



Keyed-Hash MAC (HMAC)

- RFC 2104
- Mutual authentication without sending key on the network



Hash Collisions

- There are 2^K possible hash values (where K = hash length) while there are an infinite number of files
  - » Since $\infty >> 2^K$, there *will* be hash collisions
    - In fact, an infinite number of files will have the same hash!
- The problem: *Can hash collisions be forced?*
  - » What is the impact on information security?
  - » What is the impact on digital forensics?
- Solutions to collisions
  - » Use longer hashes (e.g., SHA-256)
  - » Use multiple hashes (e.g., MD5 and SHA-1)

Ref: ***http://www.garykessler.net/library/crypto.html#hash***

What About Hashed Passwords?

md5crack

Using Google to crack passwords.

Your Results

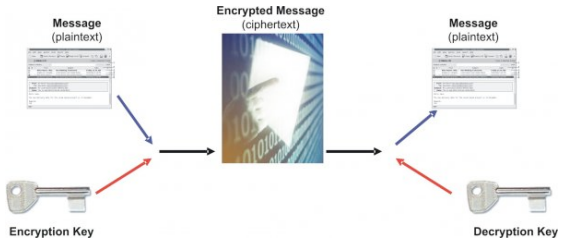
Found: c87d19bfa5f5a0c8dc75379411af75a6 = md5("kumquat")

Your Results

Sorry! Guess we couldn't find it.

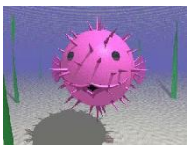
Using 2a9e402f3b2a4db8826606d527a27609, the MD5 hash of a disk drive.

Secret Key Cryptography



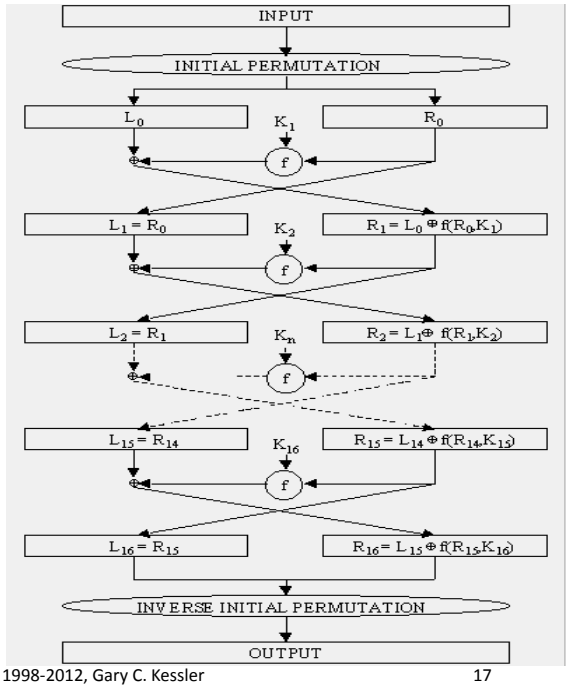
Secret Key Cryptography

- Single key *(symmetric cryptography)*
  - » Same key is used for encryption and decryption
  - » Examples: AES, DES, IDEA, 3DES, RC4, RC5, CAST, Blowfish, Twofish
- Primary use: Privacy



DES

- Designed to be fast in hardware, slow in software, resistant to various attacks
- Block cipher using 56-bit key and 64-bit blocks
- 56-bit key expanded to 64 bits using parity
- K_i is a 48-bit value derived from 64-bit key
- FIPS 46-2/ANS X3.92 describes entire process

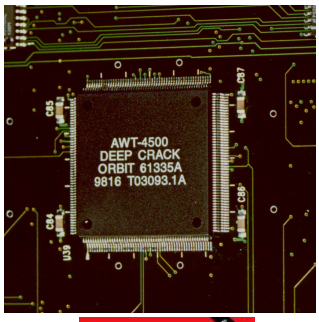


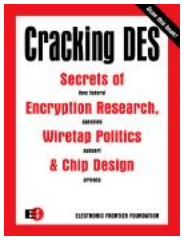
A Few Words About DES...

- DES introduced in 1977
  - » Proposed by IBM with 56- or 128-bit key; NSA adopted 56-bit key
- March 1998, U.S. Gov't. still claims that DES is safe from attack...
  - » July 1998, EFF introduces DES cracker designed for $220K; can break keys in average 4.5 days
  - » For $1M, could break DES keys in average <22 hours
- We care because DES is the most widely used crypto scheme in the financial industry!!

Breaking DES

- DES Challenge I (3/97)
  - » 84 days using thousands of computers
- DES Challenge II (1998)
  - » *distributed.net* (40 days)
  - » EFF Deep Crack (3 days)
- DES Challenge III (1/99)
  - » *distributed.net* and Deep Crack (<1 day)





Interim DES Replacements

- Triple-DES (3DES)
  - » Uses two or three keys, and three crypto steps

    $C = E_{K3}(D_{K2}(E_{K1}(P)))$ $\quad P = D_{K1}(E_{K2}(D_{K3}(C)))$
  - » Effective key length is 112 or 168 bits against brute force attacks
    - If two keys, K1=K3
  - » Recommended for all new implementations
- DESX
  - » XOR 64 additional key bits to plaintext
  - » Effective key length is 120 bits against brute force attacks
  - » Design by Rivest to protect against brute-force attacks

Advanced Encryption Standard

- NIST's next-generation SKC
  - » Open process
  - » International "competition"
    - Process started 1997, decision 2001
- Rijndael
  - » Employs 128-, 192-, or 256-bit key on a 128-, 192-, or 256-bit block
    - AES only uses a 128-bit block size
  - » Selection criteria included general security features, security implementation, software performance, smart card performance, hardware performance, and design features

AES Characteristics

- AES is an iterated block cipher
  - » Uses a 128-bit block (**Nb** = 4 words [32-bits])
  - » AES-128, AES-192, and AES-256 use a 128-, 192-, or 256-bit key (**Nk** = 4, 6, or 8 words), respectively
  - » AES-128, AES-192, and AES-256 employ 10, 12, or 14 encryption rounds (**Nr**), respectively

s_{00}	s_{01}	s_{02}	s_{03}
s_{10}	s_{11}	s_{12}	s_{13}
s_{20}	s_{21}	s_{22}	s_{23}
s_{30}	s_{31}	s_{32}	s_{33}

128-bit block (*State*)

k_{00}	k_{01}	k_{02}	k_{03}	k_{04}	k_{05}
k_{10}	k_{11}	k_{12}	k_{13}	k_{14}	k_{15}
k_{20}	k_{21}	k_{22}	k_{23}	k_{24}	k_{25}
k_{30}	k_{31}	k_{32}	k_{33}	k_{34}	k_{35}

192-bit Cipher Key

AES Cipher Keys

- AES uses a 128-, 192-, or 256-bit Cipher Key but needs a different key for every cipher round
- Key Expansion function creates enough *Round Keys*
  - » Each Round Key is the same length as the block (**Nb**)
  - » Length of Expanded Key = **Nb** * (**Nr** + 1)
  - » Cipher Key occupies first **Nk** words of Expanded Key and is the basis for deriving rest of key material
- Example using 192-bit Cipher Key (**Nb**=4, **Nk**=6, **Nr**=12)

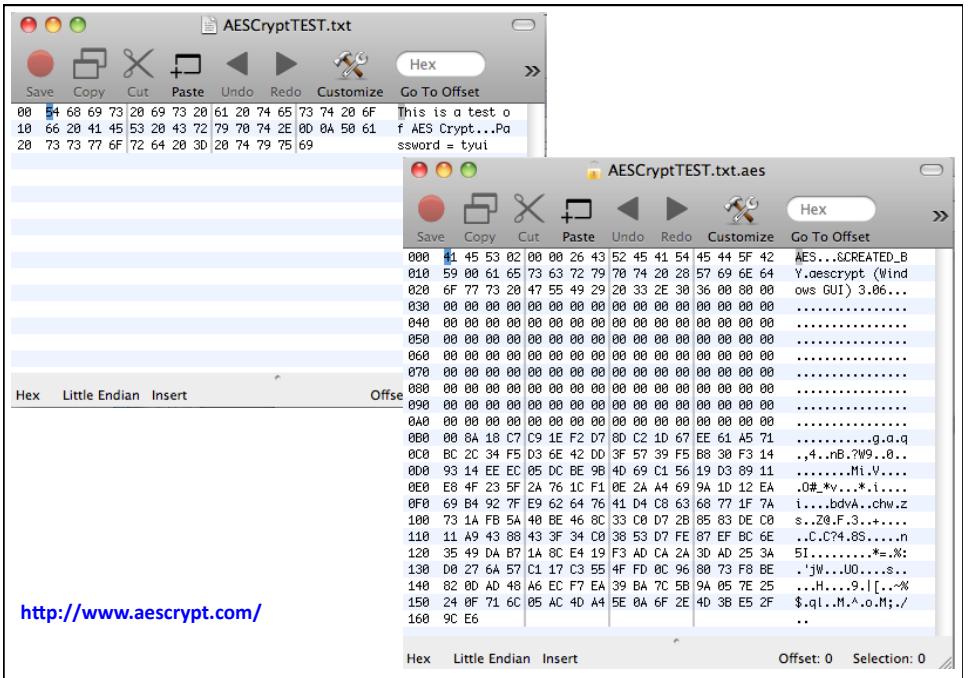
Expanded Key:	W_0	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	W_9	W_{10}	W_{11}	…	W_{48}	W_{49}	W_{50}	W_{51}
Round Keys:	**Round key 0**				**Round key 1**				**Round key 2**				…	**Round key 12**			

AES Algorithm

```
Cipher (byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
  byte state[4,Nb]

  state = in

  AddRoundKey(state, w)

  for round = 1 step 1 to Nr-1
    SubBytes(state)
    ShiftRows(state)
    MixColumns(state)
    AddRoundKey(state, w+round*Nb)
  end for

  SubBytes(state)
  ShiftRows(state)
  AddRoundKey(state, w+Nr*Nb)

  out = state
end
```


AES Basic Functions

- AES algorithm employs four basic transformations:
  - » *AddRoundKey:* XOR Round Key with State
  - » *SubBytes:* Substitute bytes in State s to form State s' on a byte-for-byte basis using S-box
  - » *ShiftRows:* Left circular shift of rows 1-3 in State s by 1, 2, and 3 bytes, respectively
  - » *MixColumns:* Apply mathematical transformation to each column in State s to form State s'



http://www.aescrypt.com/

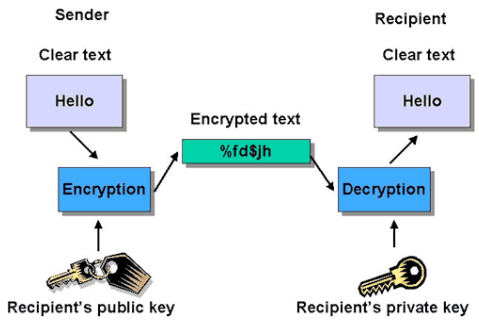
Key Length and SKC

Attacker	Budget	Tool	Time Per Recovered Key 40-bit	56-bit	Key Length For Protection In Late-1995
Pedestrian hacker	Tiny $400	PC FPGA	1 week 5 hours	Never 38 years	45 50
Small business	$10K	FPGA	12 min.	18 mon.	55
Corporate Dept.	$300K	FPGA ASIC	24 sec. 0.18 sec.	19 days 3 hours	60
Big Company	$10M	FPGA ASIC	7 sec. 5 ms	13 hours 6 min.	70
Government	$300M	ASIC	0.2 ms	12 sec.	75

ASIC = Application-specific integrated circuit
FPGA = Field programmable gate array

Source: Blaze, et al., 1996

Public Key Cryptography



Public Key Cryptography

- Two keys *(asymmetric cryptography)*
  - » One key is used for encryption, the other for decryption
  - » The two keys are related mathematically but knowledge of one key does not easily yield knowledge of the other key
  - » Examples: RSA, DSA, Diffie-Hellman, ECC, ElGamal
- Primary uses: Authentication, non-repudiation, key exchange (but invented for bulk encryption)

PKC

- All PKC based on some mathematical function that is easy but where the inverse is hard
  - » E.g., exponentiation vs. logarithms, multiplication vs. factorization
- Actual *invention* is unclear...
  - » NSA, 1966 (no proof)
  - » U.K. Gov't. Communication Headquarters, 1969 (classified until 1990s)
  - » Merkle's Puzzles (claim 1974, pub. 1978)
  - » Diffie & Hellman (pub. 1976)

The Knapsack Algorithm

Problem: The knapsack is filled with a subset of the items shown (with weights in grams). Given that the knapsack weighs 3064 grams, can you determine which set of items is contained in the knapsack?

This problem becomes computationally infeasible when there are 100 items. However, if the set of weights has nice properties known only to someone with special "trapdoor" information, then that person can quickly decipher the secret information; i.e., a 100-bit word that specifies which of the items are in the knapsack.

After "An Overview of Public Key Cryptography," M.E. Hellman, IEEE Communications Magazine, *Nov. 1978*

Diffie-Hellman Key Exchange

Alice and Bob agree on the value of a large prime number, N and a generator, G. Each calculates a private key (X) and public key (Y). The secret key (K) is derived from X and the other person's Y.

Select N=7, G=4

Choose X_A=2
$Y_A = G^{X_A} \bmod N$
$= 4^2 \bmod 7$
$= 2$

Choose X_B=3
$Y_B = G^{X_B} \bmod N$
$= 4^3 \bmod 7$
$= 1$

$K_A = Y_B^{X_A} \bmod N$
$= 1^2 \bmod 7$
$= 1$

$K_B = Y_A^{X_B} \bmod N$
$= 2^3 \bmod 7$
$= 1$

Secret Key

RSA Mathematics

- Create private/public key pair:
  - » Choose 2 primes, p & q
  - » Modulus n = pq
  - » Select public exponent e, relatively prime to (p-1)(q-1)
  - » Calculate private exponent d = (ed-1)/[(p-1)(q-1)]
- To encrypt message M with public key:
  - » $C = M^e \bmod n$
- To decrypt ciphertext C with private key:
  - » $M = C^d \bmod n$
- Of course, either key can be used first...

RSA Example

- Select p=3, q=5
- n = pq = 15
- Choose e=11, relatively prime to (p-1)(q-1) = 8
- (11d-1)/8 must be an integer; choose d=3
- M = 8384 (0x8384)
- Encrypt
  - Public key value is (e,n) = (11,15)
  - $C_i = M_i^{11} \bmod 15$
  - C = 0x2c24
- Decrypt
  - Private key value is (d,n) = (3,15)
  - $M_i = C_i^3 \bmod 15$
  - M = 0x8384

RSA Application

ciphertext = PVT_{ALICE} (message)

Alice can sign messages by encrypting with her own private key; this authenticates that she sent the message

ciphertext = PUB_{BOB} (message)

Alice can ensure that only Bob can read a message by encrypting with his public key; this provides privacy *and* proves that Bob was the intended receiver.

Elliptic Curve Cryptography

- First described in 1985 by two independent teams
- Uses logarithms and hard-to-solve problems that fall on an elliptic curve
  - » Because problems are harder to solve than factoring, smaller keys yield better protection and faster processing than RSA
- Current uses: Smart cards, mobile devices, PDAs
  - » Primary vendor is Certicom

The Elliptic Curve

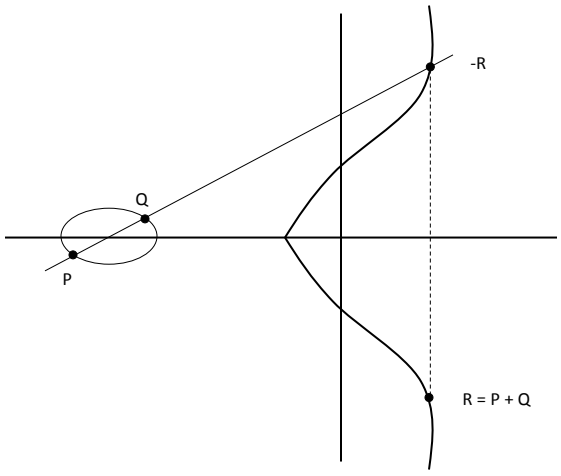
The problem: Given two points, P and Q, on an elliptic curve, find integer i ∋ P=iQ.

• Public key = iQ
• Private key = i

Elliptic curve consists of the set of real numbers (x,y) that satisfy:

$$y^2 = x^3 + ax + b$$

Small changes in a and b can make major changes in the shape of the curve and, therefore, the set of (x,y) points that satisfy the equation.



Public Key Sizes Compared

ECC Key Size	RSA Key Size	Key Size Ratio	AES Key Size
163	1,024	1:6	--
256	3,072	1:12	128
384	7,680	1:20	192
512	15,360	1:30	256

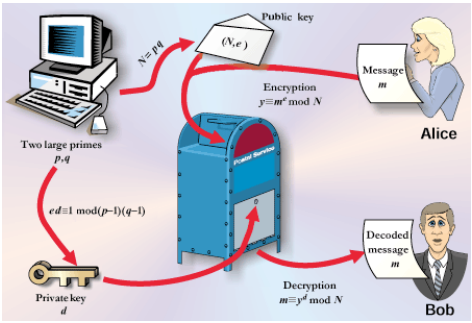
All key sizes are in bits

Source: Certicom, NIST

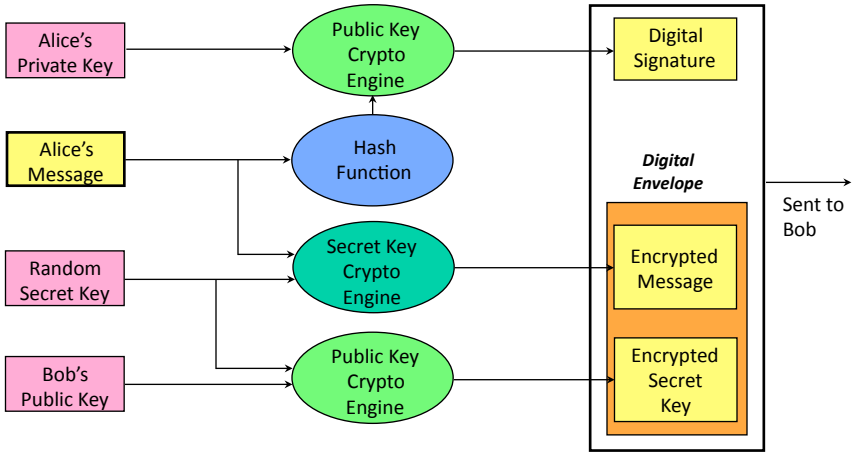
Signing and Key Exchange

- PKC is used for authentication and key exchange
  - » But not all PKC algorithms do both
- Common PKC schemes
  - » Authentication: RSA, Digital Signature Standard
  - » Key exchange: RSA, Diffie-Hellman
- PKC keys tend to be very long (e.g., 2,048 to 4,096 bits) because they have relatively long lifetimes

Sample Cryptosystems



Sample Hybrid Cryptosystem



Case Study: PGP Signatures

```
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

Hi Carol.

What was that pithy Groucho Marx quote?

/kess

-----BEGIN PGP SIGNATURE-----
Version: PGP for Personal Privacy 5.0
Charset: noconv

iQA/AwUBNFUdO5WOcz5SFtuEEQJx/ACaAgR97+vvDU6XWELV/GANjAAgBtUAnjG3
Sdfw2JgmZIOLNjFe7jP0Y8/M
=jUAU
-----END PGP SIGNATURE-----
```

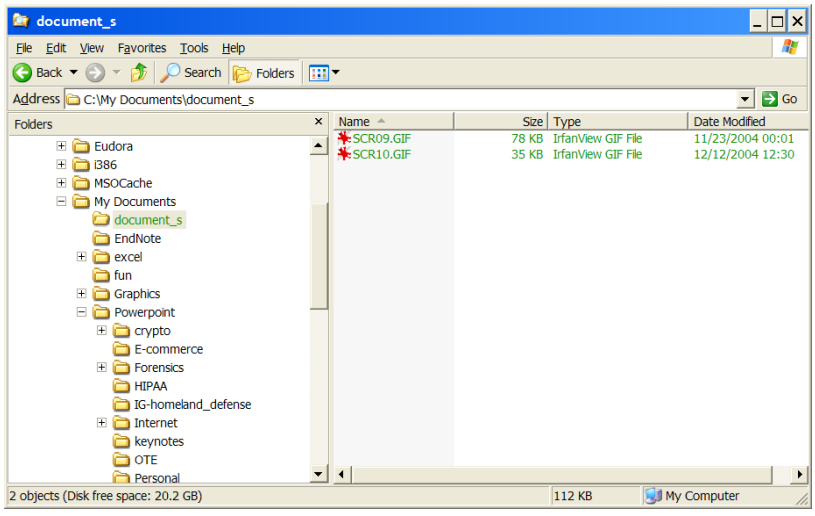

Case Study: PGP Encryption

```
-----BEGIN PGP MESSAGE-----
Version: PGP for Personal Privacy 5.0
MessageID: DAdVB3wzpBr3YRunZwYvhK5gBKBXOb/m

qANQR1DBwU4D/TlT68XXuiUQCADfj2o4b4aFYBcWumA7hR1Wvz9rbv2BR6WbEUsy
ZBIEFtjyqCd96qF38sp9IQiJIKlNaZfx2GLRWikPZwchUXxB+AA5+lqsG/ELBvRa
c9XefaYpbbAZ6z6LkOQ+eE0XASe7aEEPfdxvZZT37dVyiyxuBBRYNLN8Bphdr2zv
z/9Ak4/OLnLiJRk05/2UNE5Z0a+3lcvITMmfGajvRhkXqocavPOKiin3hv7+Vx88
uLLem2/fQHZhGcQvkqZVqXx8SmNw5gzuvwjV1WHj9muDGBY0MkjiZIRI7azWnoU9
3KCnmpR60VO4rDRAS5uGl9fioSvze+q8XqxubaNsgdKkoD+tB/4u4c4tznLfw1L2
YBS+dzFDw5desMFSo7JkecAS4NB9jAu9K+f7PTAsesCBNETDd49BTOFFTWWavAfE
gLYcPrcn4s3EriUgvL3OzPR4P1chNu6sa3ZJkTBbriDoA3VpnqG3hxqfNyOlqAka
mJJuQ53Ob9ThaFH8YcE/VqUFdw+bQtrAJ6NpjIxi/x0FfOInhC/bBw7pDLXBFNaX
HdlLQRPQdrmnWskKznOSarxq4GjpRTQo4hpCRJJ5aU7tZO9HPTZXFG6iRIT0wa47
AR5nvkEKoIAjW5HaDKiJriuWLdtN4OXecWvxFsjR32ebz76U8aLpAK87GZEyTzBx
dV+lH0hwyT/y1cZQ/E5USePP4oKWF4uqquPee1OPeFMBo4CvuGyhZXD/18Ft/53Y
WIebvdiCqsOoabK3jEfdGExce63zDI0=
=MpRf
-----END PGP MESSAGE-----
```

Case Study: Windows 2000/XP EFS

- Encrypting File System (EFS)
  - » When the file is saved to disk:
    - Random file encryption key (FEK) is created
    - File contents encrypted using FEK and encryption algorithm
      - Windows 2000 and XP default to DESX; XP also supports 3DES
      - Windows XP SP1, Server 2003, Vista, and Server 2008 default to AES; also support DESX and 3DES
      - Windows 7 and Server 2008 R2 default to AES, SHA, and ECC; also support DESX and 3DES
    - FEK stored with file, encrypted with user's RSA public key (and, optionally, recovery agent's RSA public key)
  - » When the file is opened:
    - FEK recovered for decryption using RSA private key, which can be stored on external floppy disk or smart card
    - If private key lost, files may be accessed using RA's private key
    - Key tied to username prior to Win XP SP2; now uses user password
  - » A pre-encryption backup file is *deleted* after encryption

EFS



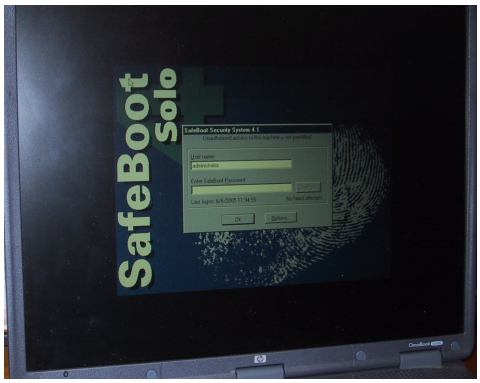
The `cipher` command (Windows)

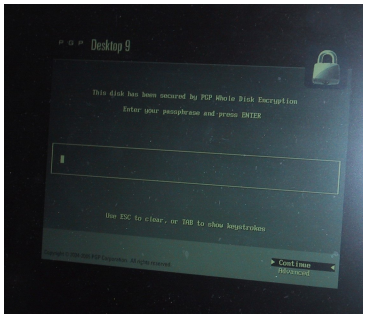
```
C:\> cipher /u /n
Encrypted File(s) on your system:

C:\My Documents\document_s\SCR09.GIF
C:\My Documents\document_s\SCR10.GIF
C:\My Documents\Word\GKS\phish\1_real1.png
C:\My Documents\Word\GKS\phish\2_real2.png
C:\My Documents\Word\GKS\phish\3_bogus1.png
C:\My Documents\Word\GKS\phish\4_bogus2.png
C:\My Documents\Word\GKS\phish\5_has_data.png
C:\My Documents\Word\GKS\phish\62.193.219.166.html.txt
C:\My Documents\Word\GKS\phish\6_after_submit.png
C:\My Documents\Word\GKS\phish\7_tcp_stream.png
C:\My Documents\Word\GKS\phish\commnatlbank.acp
C:\My Documents\Word\GKS\phish\Re Help with investigation.txt

C:\>
```

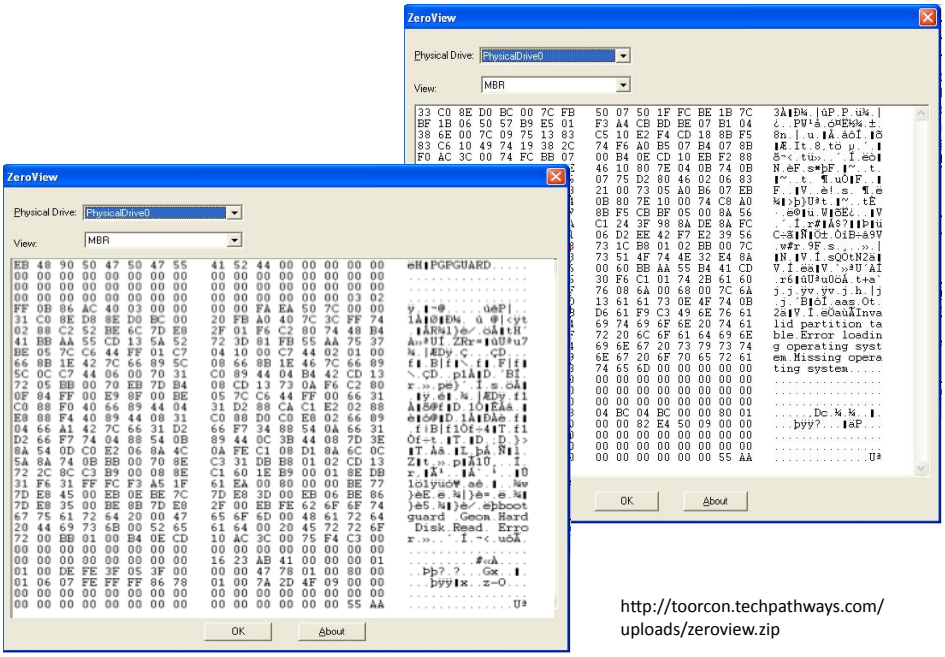

Whole Disk Encryption (WDE)

- WDE encrypts entire drive
  - » WDE modifies boot sector 0 to go to alternate loader
  - » Files are available to user after a password is entered when logging on
    - Files are encrypted when viewed by forensics software if disk drive is powered down
- Some disk encryption software
  - » Windows Vista
    - Full volume encryption; Vista encrypts data partition, but not boot partition
  - » PGP 9.0
  - » Pointsec
  - » Safeboot
  - » Utimaco
- A case for live forensics...

Pre-Boot Logon







http://toorcon.techpathways.com/uploads/zeroview.zip

Sector 0 Strings

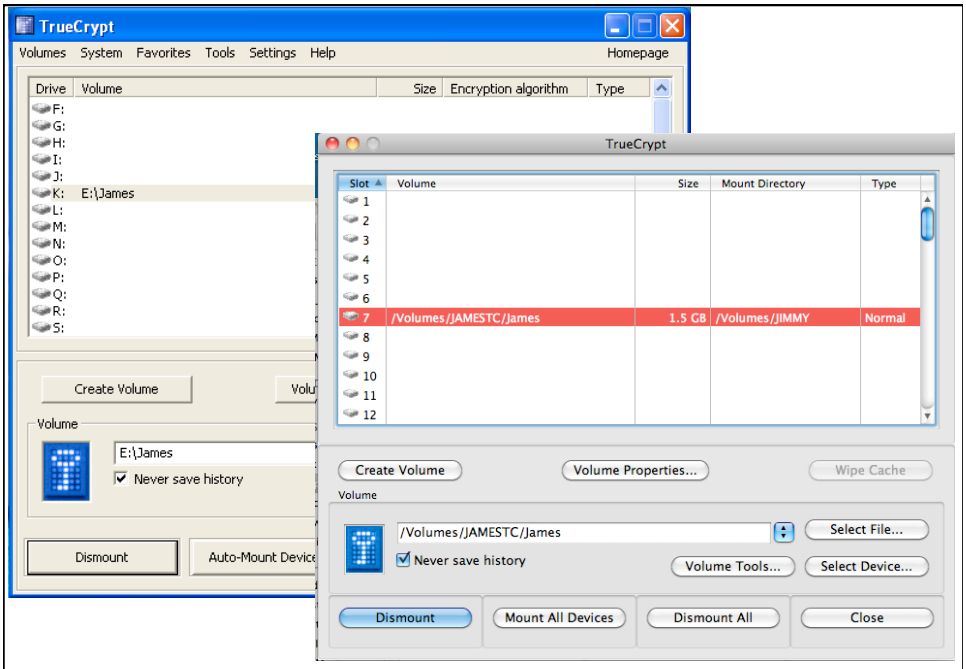
- PGP 9
  - » "PGPGUARD" at offset 0x3
  - » "bootguard" at offset 0x16C
- SafeBoot 4.13
  - » "SafeBoot" at offsets 0x3, 0x168, 0x183

Windows Vista

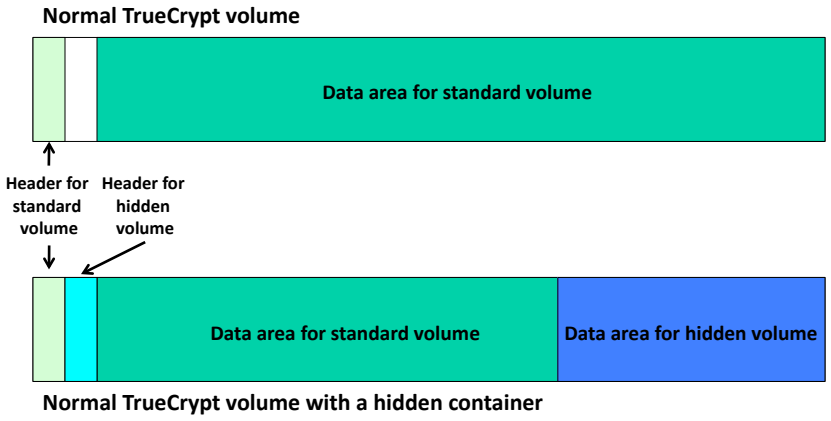
- Vista encryption will cause forensics examiners some problems...
  - » Hardware-enabled full-volume encryption
  - » Windows Vista Enterprise and Ultimate
    - Encryption **not** on by default
  - » Use BitLocker drive encryption tied to Trusted Platform Module (TPM) chip or USB flash drive for key storage

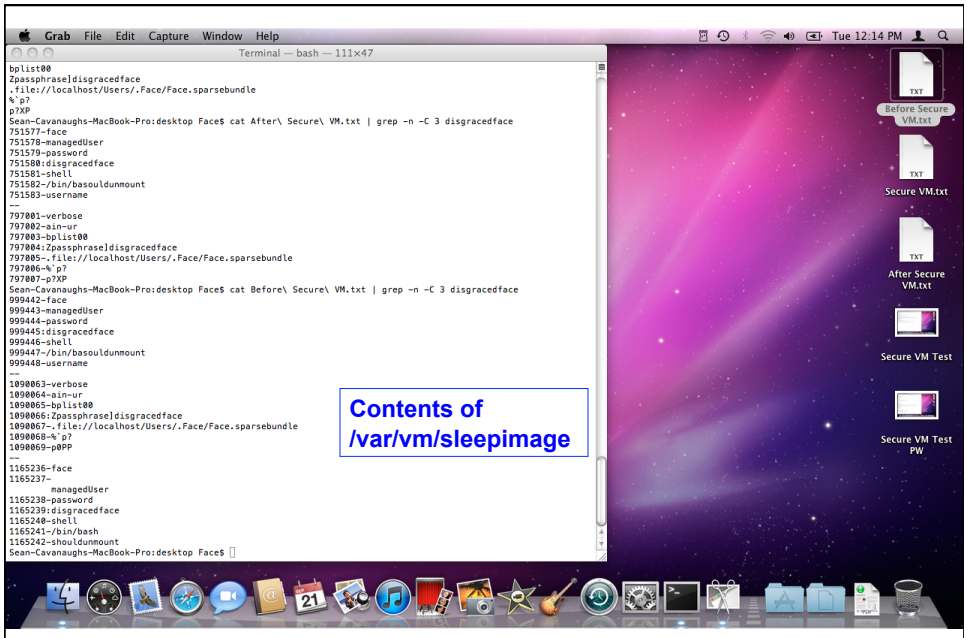
Other Crypto Schemes

- TrueCrypt
  - » Open source encryption for Windows, MacOS, or Linux
  - » Virtual encrypted disk using AES, Serpent, or Twofish
  - » Can create hidden encrypted volume
  - » First released in 2004
- FileVault
  - » File encryption for Macs, using AES
  - » Password derived from user's login password



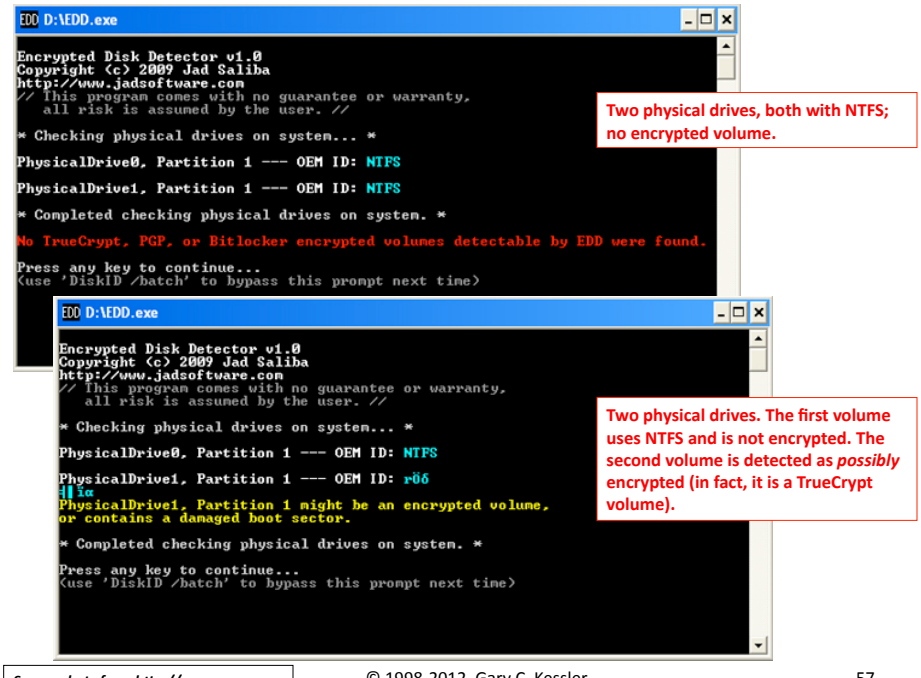
Plausible Deniability



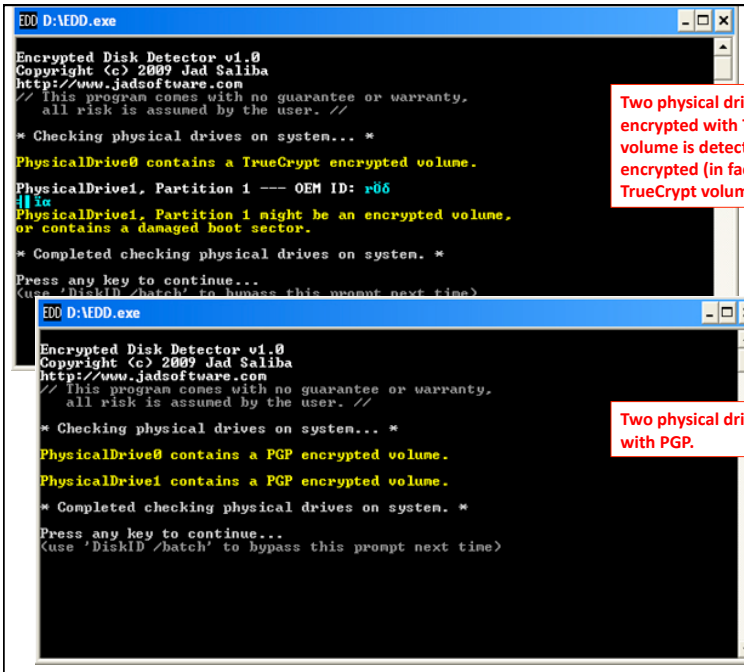


EDD

- JADsoftware's Encrypted Disk Detector
  - » Tests for BitLocker, PGP, and TrueCrypt encrypted drives and volumes (partitions)
  - » *http://www.jadsoftware.com/go/?page_id=167*



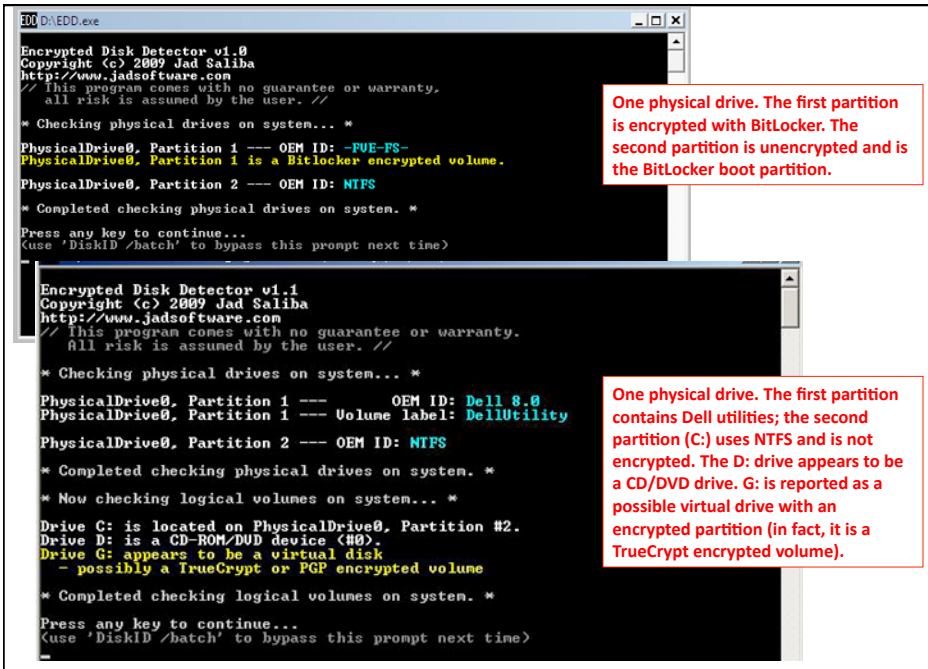
Screen shots from http://www.jadsoftware.com/home/edd.htm



Two physical drives. The first volume is encrypted with TrueCrypt. The second volume is detected as *possibly* encrypted (in fact, it is also a TrueCrypt volume).

Two physical drives, both encrypted with PGP.

Screen shots from http://www.jadsoftware.com/home/edd.htm



One physical drive. The first partition is encrypted with BitLocker. The second partition is unencrypted and is the BitLocker boot partition.

One physical drive. The first partition contains Dell utilities; the second partition (C:) uses NTFS and is not encrypted. The D: drive appears to be a CD/DVD drive. G: is reported as a possible virtual drive with an encrypted partition (in fact, it is a TrueCrypt encrypted volume).

Screen shots from http://www.jadsoftware.com/home/edd.htm

Secure Communication Protocols

- Secure MIME (S/MIME)
- Secure Sockets Layer (SSL)
  - » https, ftps, pops, smtps, ...
- Secure Electronic Transactions (SET)
- Secure HTTP (S-HTTP)
- Transaction Internet Protocol (TIP)
- Simple Authentication and Security Layer (SASL)
- Pretty Good Privacy (PGP)
- IP Security Protocol (IPsec)
- Kerberos
- Server Gated Cryptography (SGC)
- Transport Layer Security (TLS)
- Secure Shell (SSH)
- Authenticated POP (APOP)

Do *not* trust "secret" cryptographic protocols (e.g., Skipjack!). The safety is in the choice (and length) of the *key*, not the secrecy of the *algorithm (*Kerckhoffs' Principle, 1883).

NSA Suite B Cryptography

- NSA guideline to protect government documents at SECRET level (2005)
- Expected lifetime: 2031+

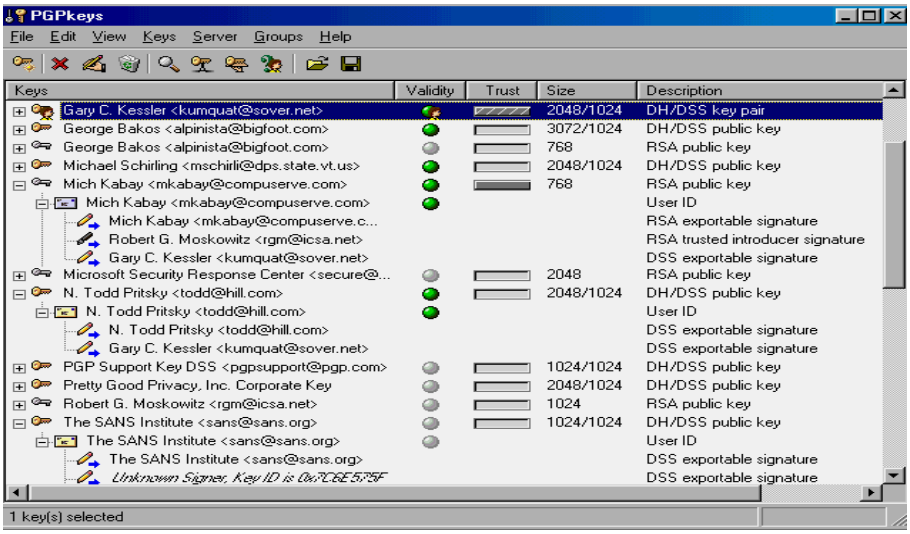
Cryptographic Function	Sensitive But Unclassified	Classified
Secret Key Algorithm	AES-128	AES-256
Hash Algorithm	SHA-256	SHA-384
Public Key Algorithm	ECC w/ Diffie-Hellman (key exchange) and ECC w/ DSA (digital signatures); 256 and 384 bits	

Trust in Cryptosystems

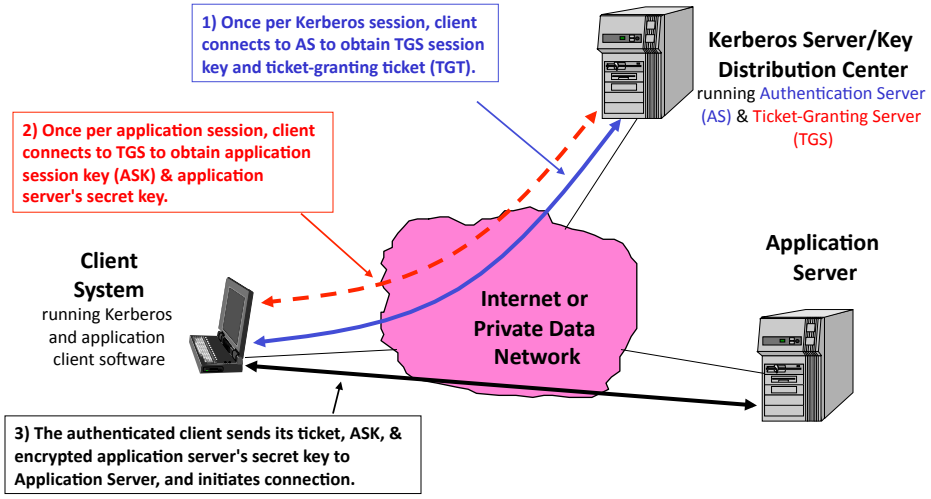
Trust Models

- When using cryptography, how can you trust the entity that gives you a key?
  - » PGP Web of trust
  - » Kerberos trusted server and SKC for *a priori* relationships
  - » PKI trusted third parties and PKC for anyone-to-anyone communication

PGP Web of Trust



Kerberos

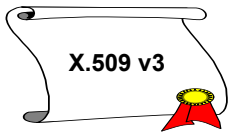


Public Key Infrastructure

- How can a party's public key be found if not known locally? Where is the key stored?
- How does a recipient verify that a public key really belongs to the sender *and* that it is being used for a legitimate purpose?
- When does a public key expire?
- How can a key be revoked in case of loss or compromise?

Certificates

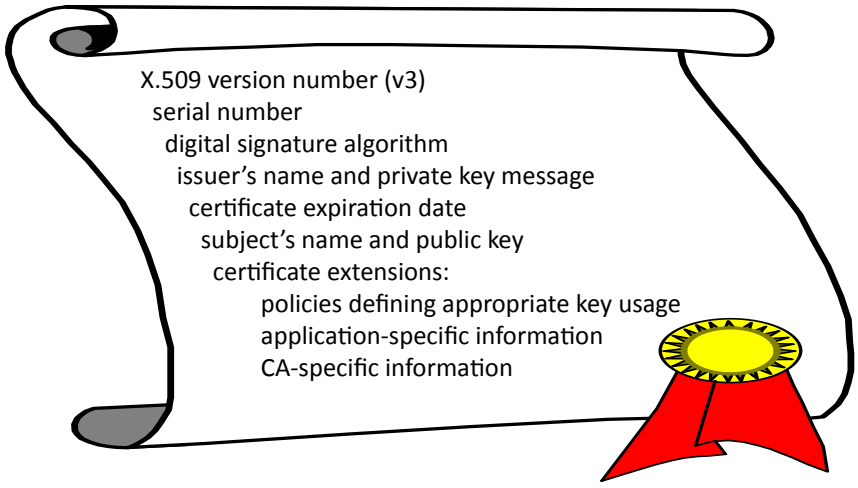
- *Certificates* bind a public key to an individual, position, or other entity, and provide
  - » Identification
  - » Expiration date
  - » Issuing authority
  - » Serial number
  - » Policies about how the user was identified
  - » Limitations on how the key may be used



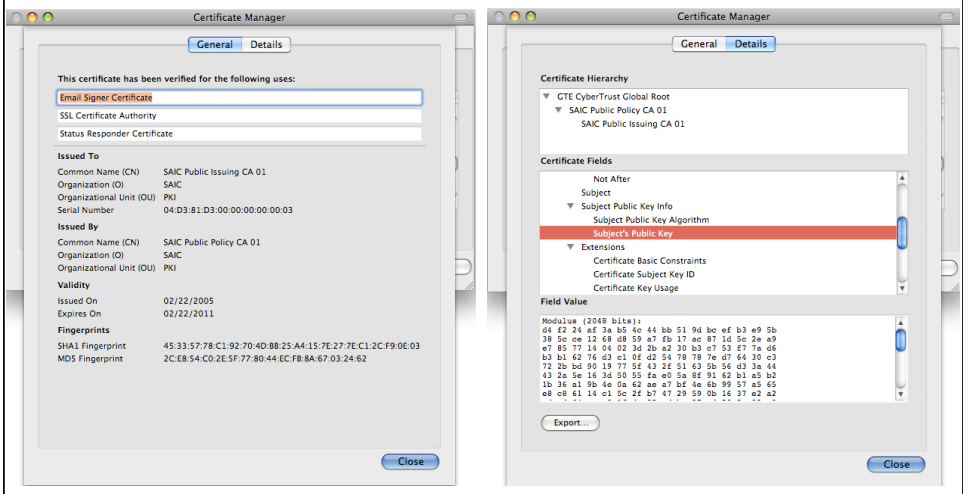
Certificates in Real-Life...

- Certificates identify us, what we are allowed to do, issuer, validity period, etc.
  - » Driver's license: Name, DOB, address, type of vehicle, issuing state, valid period, serial number, photo(?), organ donation(?)
  - » Credit card: Name, serial number, valid period, issuer
  - » SCUBA certification: Name, DOB, serial number, level of training, certification date, instructor, issuing agency, photo(?)

ITU-T X.509 Digital Certificate



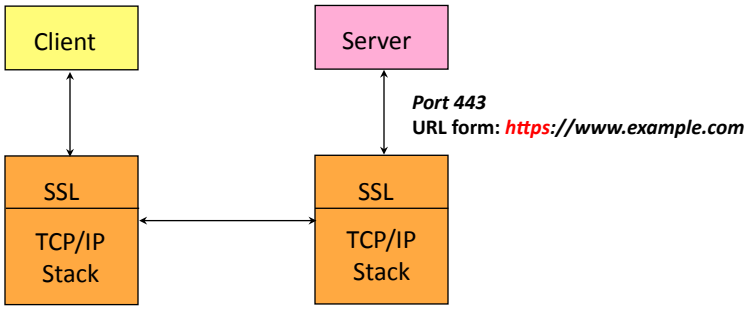
Sample Browser Certificate



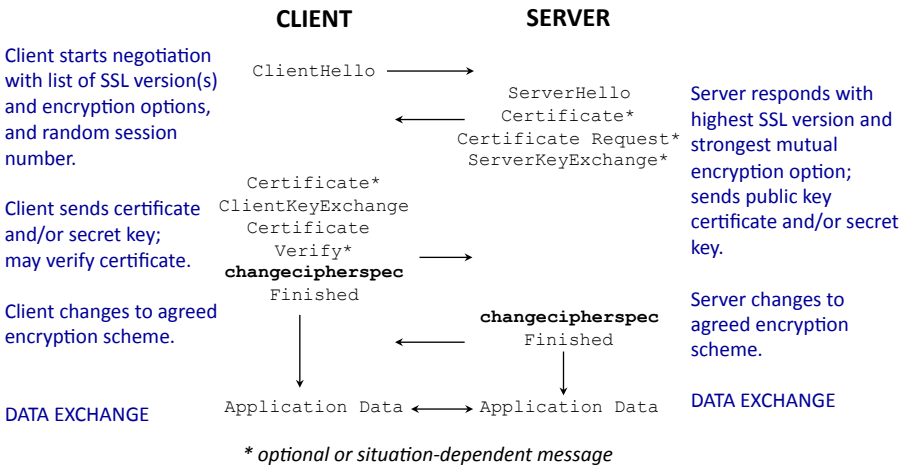
Secure Sockets Layer

- Originally developed by Netscape Communications
  - » SSL v2.0 (deprecated) and v3.0 (weak)
  - » Transport Layer Security (TLS) v1.0 ~ "SSL v3.1" (RFC 2246)
    - Theoretical vulnerabilty described in 2002 made practical in 2011!
    - TLS v1.1 (RFC 4346) and TLS v1.2 (RFC 5246)
- Provides privacy, integrity, client/server authentication
- Application-independent
  - » Can be used with HTTP, Telnet, FTP, NNTP, IMAP, POP3 over TCP
  - » Datagram TLS (v1.2, RFC 6347) operates over UDP
- Two main protocols
  - » SSL Handshake Protocol (parameter negotiation)
  - » SSL Record Protocol (data transfer)

HTTP Over SSL



SSL Handshake Protocol



** optional or situation-dependent message*

Limitations of PKI

- A digital signature does ***not*** prove that Alice signed a message, but that her private key did
  - » Good cryptographic algorithms can be bypassed by viruses, malicious code, abuse/misuse by users, and other real-world events
- Users do not generally check the source or validity of received certificates

Summary and Closure!

Detecting Encryption

- Cryptography provides *secret* communication but not necessarily *hidden*
  - » Use of crypto does not form a covert communications channel
  - » Encrypted messages and files can be detected by a third party

Example Detection Statistics (FTK)

- Most encryption detection schemes are testing for randomness; high randomness suggests use of encryption
  - » *Arithmetic Mean:* Calculated by summing all of the bytes in a file and dividing by the file length; if random, the value should be ~1.75.
  - » *Chi-Squared Error Percent:* This distribution is calculated for a byte stream in a file; the value indicates how frequently a truly random number would exceed the calculated value.
  - » *Entropy:* Describes the information density (per Shannon) of a file in bits/character; as entropy-->8, there is more randomness.
  - » *MCPI Error Percent:* The Monte Carlo algorithm uses statistical techniques to approximate the value of π; A high error rate implies more randomness.
  - » *Serial Correlation Coefficient:* Indicates the amount to which each byte is an e-mail relies on the previous byte. A value close to 0 indicates randomness.

Cryto Attack Methods

- Password guessing
- Known plaintext
- Chosen plaintext
- Known ciphertext
- Dictionary attack
- Brute force attack
- Side channel attacks

Crypto Breaking Tools

- Rainbow tables
- *distributed.net* 
- Rack attack

- Passware
- ElcomSoft
  - » Password Recovery Bundle

- AccessData
  - » Password Recovery Toolkit (PRTK)
  - » Distributed Network Attack (DNA)
  - » Portable Office Rainbow Table (PORT)

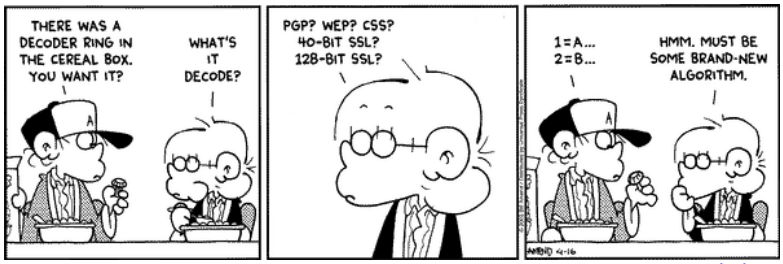
Some Parting Notes

- Authentication and proof of identity ≠ trust
- Goodness of a crypto scheme is *time in use*, not *secrecy of the algorithm*
  - » Kerckhoffs' Principle, 1883
- Good crypto can be implemented poorly
  - » Think WEP, CSS
- Theoretical ways to break a crypto scheme may not be practical
  - » E.g., almost *every* crypto scheme can be brute-forced...

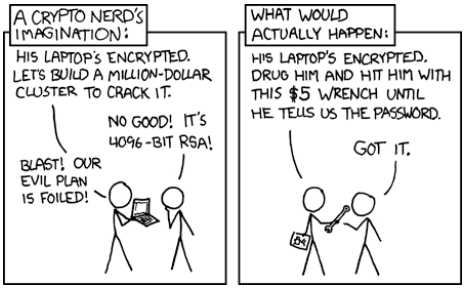
A Caveat About Cryptography

- Crypto is *necessary* but not *sufficient* for trusted, secure communication
  - » Crypto can provide perfect security... but humans provide an insurmountable weak link!
  - » Crypto can be used to prove that I am who I say I am but it can't tell you whether to trust me or not
- Ref: *Secrets and Lies*, B. Schneier

A Little Crypto Humor...



Foxtrot, 4/16/2003



http://xkcd.com/538/, Feb. 2009

Additional References

- *Cryptography Engineering: Design Principles and Practical Applications*, Ferguson, Schneier, & Kohno
- *The Code Book: The Evolution of Secrecy from Mary Queen of Scots to Quantum Cryptography*, Singh
- *Classical and Contemporary Cryptology*, Spillman
- *Malicious Cryptography*, Young & Yung

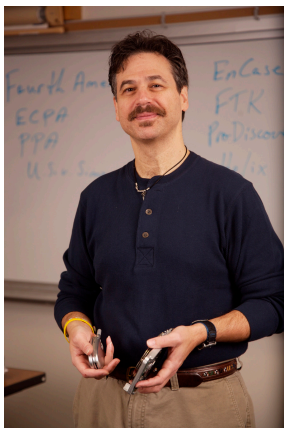
- Counterpane (*www.counterpane.com)*
- Cryptography Research (*www.cryptography.com*)
- RSA's Crypto FAQ (*www.rsa.com/rsalabs/node.asp?id=2152*)
- GCK's crypto overview paper (*www.garykessler.net/library/crypto.html*) and crypto links (*www.garykessler.net/library/securityurl.html#crypto*)

Author Contact Information

Gary C. Kessler, Ph.D., CCE, CISSP
GARY KESSLER ASSOCIATES
2 Southwind Drive
Burlington, VT 05401

mobile: +1 802-238-8913
e-mail: gck@gary*kessler.net*
gkessler@bpdvt.org
kesslerg@norwich.edu
Skype: *gary.c.kessler*

http://www.garykessler.net
http://www.vtinternetcrimes.org
http://infoassurance.norwich.edu/



Acronyms and Abbreviations

3DES	Triple DES
AES	Advanced Encryption Standard (NIST)
ANS	American National Standard
CA	Certificate authority
CPS	Certification practice statement
CRL	Certificate Revocation List
DES	Data Encryption Standard
DOB	Date of birth
DoS	Denial of service
DSA	Digital Signature Algorithm (NIST)
ECC	Elliptic Curve Cryptography
EFF	Electronic Frontier Foundation
EFS	Encrypting File System (W2K)
FIPS	Federal Information Processing Standard
FTP	File Transfer Protocol (IETF)
HMAC	Hashed message authentication code
HTTP	Hypertext Transfer Protocol (IETF)
https	HTTP over SSL
IDEA	International Data Encryption Algorithm
IE	Internet Explorer (MS)
IETF	Internet Engineering Task Force
IMAP	Internet Message Access Protocol (IETF)
ISP	Internet service provider
ITU-T	International Telecommunication Union, Telecommunication Standardization Sector
IV	Initial vector
MAC	Message authentication code
MD2/4/5	Message Digest 2, 4, & 5
MIPS	Millions of instructions per second
MS	Microsoft
NIST	National Institute of Standards and Technology
NNTP	Network News Transport Protocol (IETF)
NSA	National Security Agency
OS	Operating system
PDA	Personal digital assistant
PGP	Pretty Good Privacy
PKC	Public key cryptography
PKI	Public key infrastructure
POP	Post Office Protocol (IETF)
RA	Registration Authority
RC2/4/5	Rivest Cipher (or Ron's Code) 2, 4, and 5
RFC	Request for Comments (IETF)
RSA	Rivest, Shamir, Adleman

Acronyms and Abbreviations (cont.)

SCUBA	Self-contained underwater breathing apparatus
SHA	Secure Hash Algorithm (NIST)
SKC	Secret-key cryptography
SSL	Secure Sockets Layer (Netscape)
TCP	Transmission Control Protocol (IETF)
TLS	Transport Layer Security (IETF)
URL	Uniform Resource Locator
WDE	Whole disk encryption
W2K	Windows 2000 (MS)
XOR	Exclusive OR

Questions? Comments? Queries?