20.1 CRYPTOGRAPHY CONCEPTS

- Data States
- Cryptography Components
- Cryptography Types
- XOR
- One Time Pad
- GAK

DATA STATES

- Data at Rest
  - Stored on a hard drive, USB stick, CD/DVD, or any other type of electronic storage medium
- Data in Transit
  - Data is actively being transmitted on a network
- Data in Use
  - Data is loaded into memory
  - Is, or will shortly be, processed by the CPU

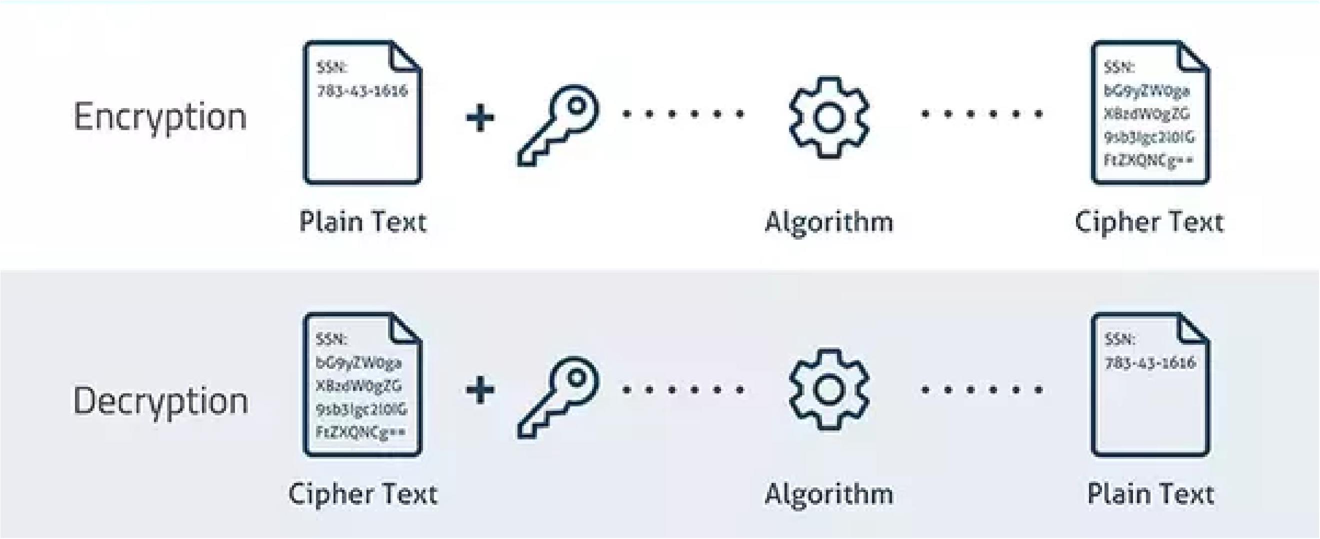
You can encrypt data in any of these states to increase confidentiality and trust

CRYPTOGRAPHY

- The process of converting ordinary plain text into unintelligible text and vice-versa
- When encrypted, the data can be safely stored, used, or transmitted across a network
- Even if it is stolen or intercepted, the attacker cannot read it
- Used to protect data confidentiality

COMPONENTS OF CRYPTOGRAPHY

- Unencrypted data (plain text)
- Algorithm (cipher)
- Key
- Ciphertext (encrypted text)



CIPHERS

- AKA algorithm
- A mathematical formula for scrambling data
- Block cipher
  - Data is encrypted in fixed-size blocks (typically 64 bits)
  - Plain text is converted into cipher text one block at a time
  - Often some output from one encrypted block is added to the encryption of the next block
  - Good for large amounts of data
    - E.g. files, data at rest
- Stream cipher
  - Data encrypted in a continuous stream
  - Uses XOR to encrypt data one bit, byte, or character at a time
  - Typically faster than block ciphers
  - Requires fewer resources and less complex circuitry
  - Good for real-time communications

TYPES OF CRYPTOGRAPHY

- Symmetric Encryption
  - Uses the same key for both encryption and decryption
- Asymmetric Encryption
  - Uses one key for encryption and a different key for decryption
- Hashing
  - One way encryption
  - Fixed length output for any length input
  - No key
  - Meant for data integrity
  - Data is not encrypted
  - Hashed output accompanies the data for anyone to verify

EXCLUSIVE OR (XOR)

- A boolean logic operation that is widely used in cryptography
- Used in generating parity bits for error checking and fault tolerance
- Also used by stream ciphers such as RC4 to encrypt a bytestream
- The output is True (or 1) if and only if the two inputs are different
- The output is false (or 0) if the two inputs have the same value
- Example:
  - What will be the result if you apply XOR to the following binary values:

```
11001100
01101010
10100110
```

Polymorphic shellcode encrypts its code using XORing.
The shellcode is then later decrypted and executed.

ONE TIME PAD

- An encryption technique that cannot be cracked
- Every message is encrypted with a different pre-shared key
  - Only the involved parties know the keys
- Ensures that there is no pattern in the key for an attacker to guess or find
  - Even if one message is decrypted, all other messages remain secure
- Requires two identical copies of the pad be produced and distributed securely before use
- Was popular during World War II

Do not confuse a One Time Pad with the modern One Time Password (OTP).
The One Time Pad is for encryption, using a different key for each message.
The One Time Password is time-limited, and used to authenticate the user or device for a single session. It is typically sent to a user's mobile phone via SMS.

GOVERNMENT ACCESS TO KEYS (GAK)

- GAK requires software companies to provide the government with enough copies of their keys that the remaining keys could be deciphered
- The government guarantees they will keep the keys secure
- The government guarantees the keys will only be used if there is a court-issued warrant
- Similar to the government's right to wiretap phones

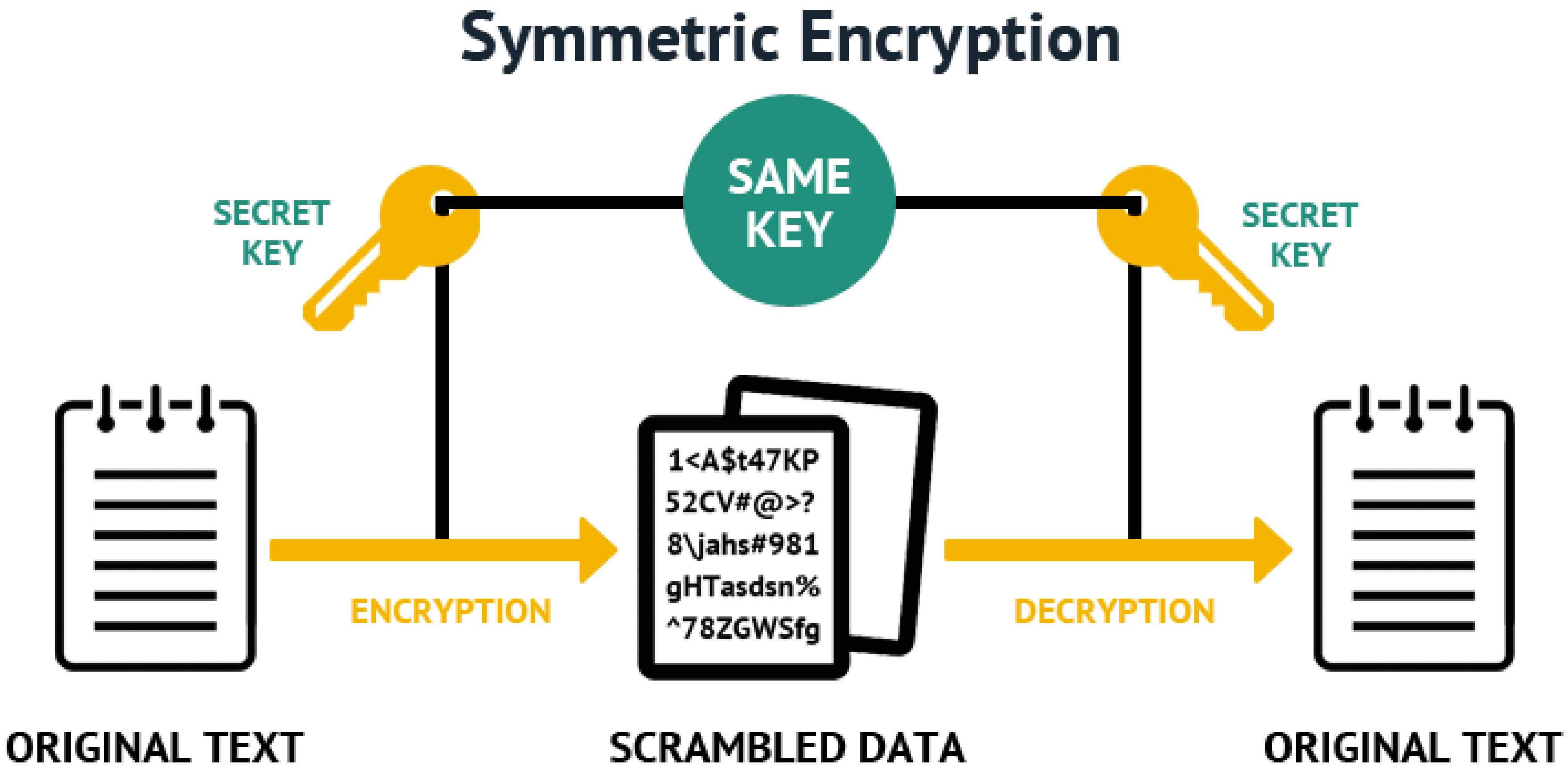
20.2 SYMMETRIC ENCRYPTION

- Symmetric Encryption Types
- Block Cipher
- Stream Cipher

SYMMETRIC ENCRYPTION

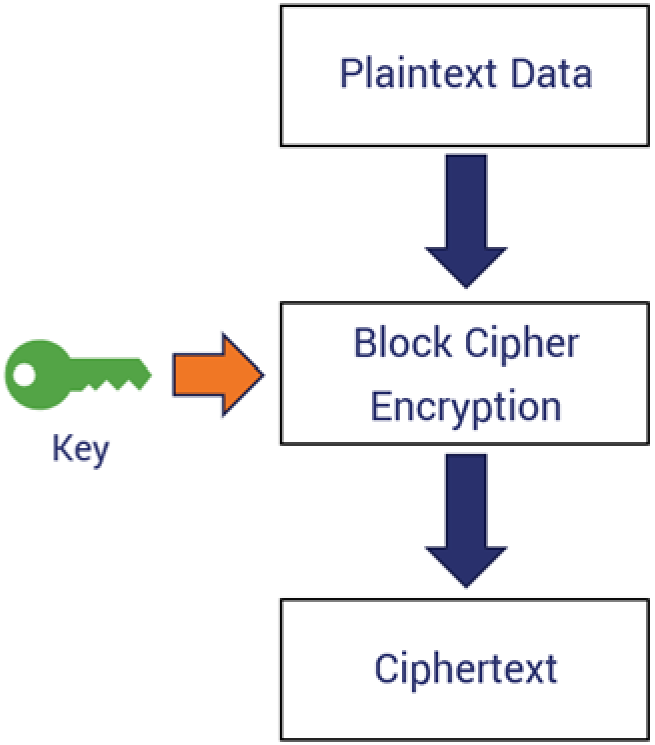
- The same key is used to encrypt and decrypt
- Used extensively to protect data at rest
- Provides confidentiality
- Excellent for bulk data encryption
- Is fast with good performance
- Less resource intensive than asymmetric encryption – easier on smaller devices!
- Uses the same key to encrypt and decrypt
  - Key is at risk
  - You must share the key in advance
  - If the key is compromised, all files are at risk of loss of confidentiality

SYMMETRIC ENCRYPTION EXAMPLE



BLOCK CIPHER SYMMETRIC ALGORITHM

- Block cipher
  - Takes a block of plaintext bits
  - Generates a block of ciphertext bits
    - Generally the same size
  - The size of block is fixed in the given scheme
  - The choice of block size does not directly affect to the strength of encryption scheme
  - The strength of cipher depends up on the key length



BLOCK CIPHER SYMMETRIC ALGORITHMS

- DES
  - Archetypal block cipher
  - Transforms fixed-length blocks of plaintext into ciphertext bit strings of equal length
  - Inherently weak with current technology
  - Has already been broken
- 3DES
  - DES process repeated 3 times to increase encryption strength
- AES (the current US government standard)
  - Symmetric-key algorithm designed to secure unclassified, sensitive U.S. government documents
  - Iterated block cipher designed to keep doing the same operation repeatedly
  - Block size of 128 bits
  - AES key sizes:
    - 128 for AES-128
    - 192 for AES-192
    - 256 for AES-256

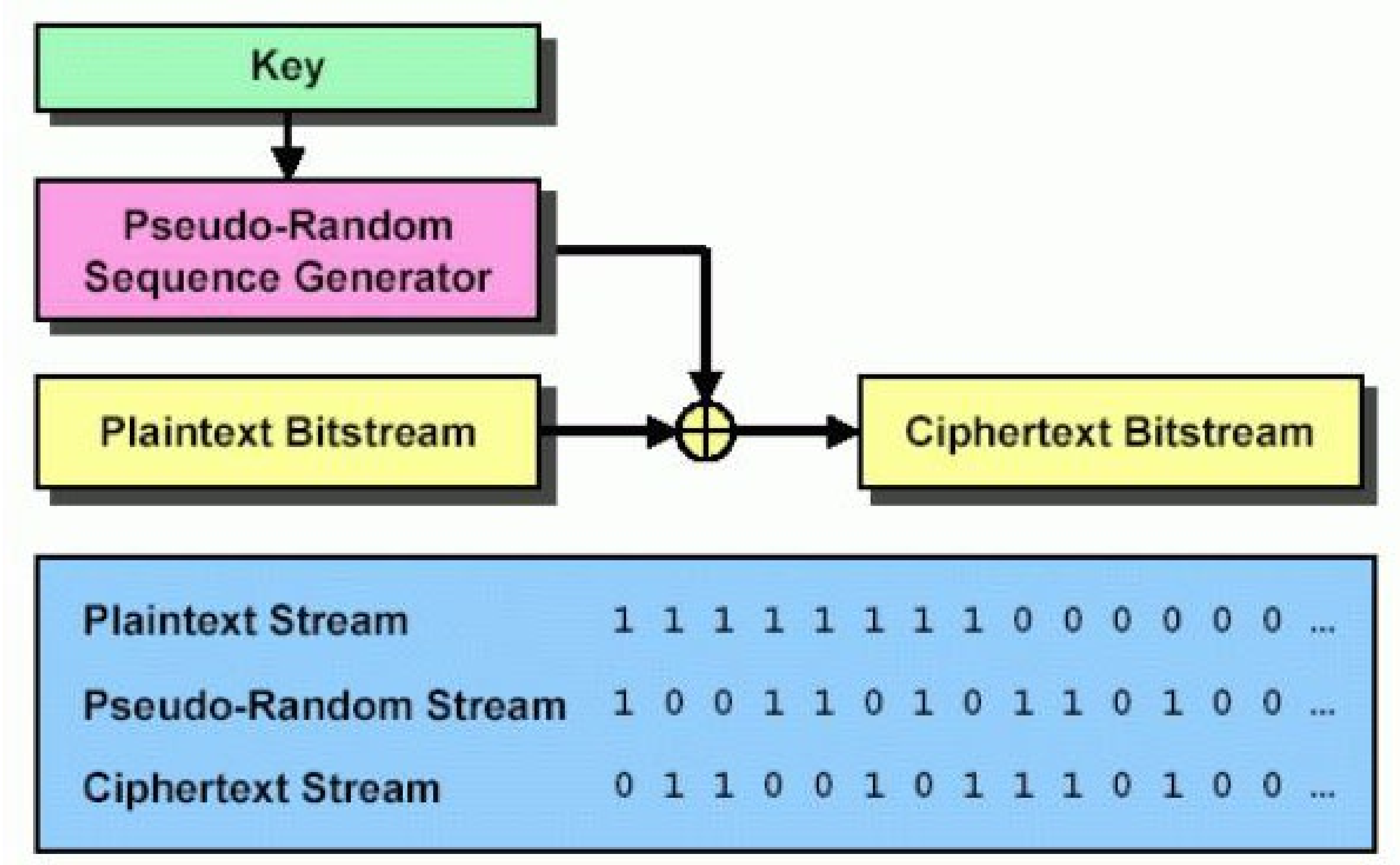
BLOCK CIPHER SYMMETRIC ALGORITHMS (CONT'D)

- Blowfish
  - 64 bit block cipher
  - 32 – 448 bit key length
  - Faster than DES
- Twofish
  - 128 bit block cipher
  - 128 – 256 bit key length
- RC2, RC5, RC6
  - 64 – 128 bit block cipher
  - Each iteration has increased the key size
  - RC6 supports 2040 bit keys

STREAM CIPHER SYMMETRIC ALGORITHM

- Processes an individual bit, byte, or character of plaintext at a time
  - Do not divide the data into discrete blocks
- At the transmitting end, XOR each bit of:
  - your plaintext continuous stream + a pseudo-random sequence
- At the receiving end, use the same symmetric key and XOR to decrypt
- Often faster than block ciphers
- Also useful when transmission errors are likely to occur
  - They have little or no error propagation

STREAM CIPHER EXAMPLE



Plaintext Stream	1	1	1	1	1	1	1	1	0	0	0	0	0	0	...
Pseudo-Random Stream	1	0	0	1	1	0	1	0	1	1	0	1	0	0	...
Ciphertext Stream	0	1	1	0	0	1	0	1	1	1	0	1	0	0	...

STREAM CIPHER SYMMETRIC ALGORITHMS

- RC4
  - Popular stream cipher
  - Used in Wi-Fi WEP
  - Key length 40 – 2048 bits
- PKZIP
  - File archive/compression program that uses a stream cipher to encrypt files

DATA AT REST SCENARIO

- You regularly perform backups of your critical servers
- You can’t afford to send the backup tapes to an off-site vendor for long-term storage and archiving
- Instead, you store the backup tapes in a safe in your office
- Security auditors tell you it’s safer to store the backup tapes off-site
- Your manager wants to take the tapes home in her briefcase every night
- What can she do to secure those tapes while in transit?
- Encrypt the backup tapes
- For good measure, have her carry them in a lockbox and not just her briefcase

In this scenario, the data is still considered to be “at rest”.
Even though someone is physically carrying the storage media to another location, the data itself is not being transmitted across a network where it can be intercepted by a sniffer

20.3 ASYMMETRIC ENCRYPTION

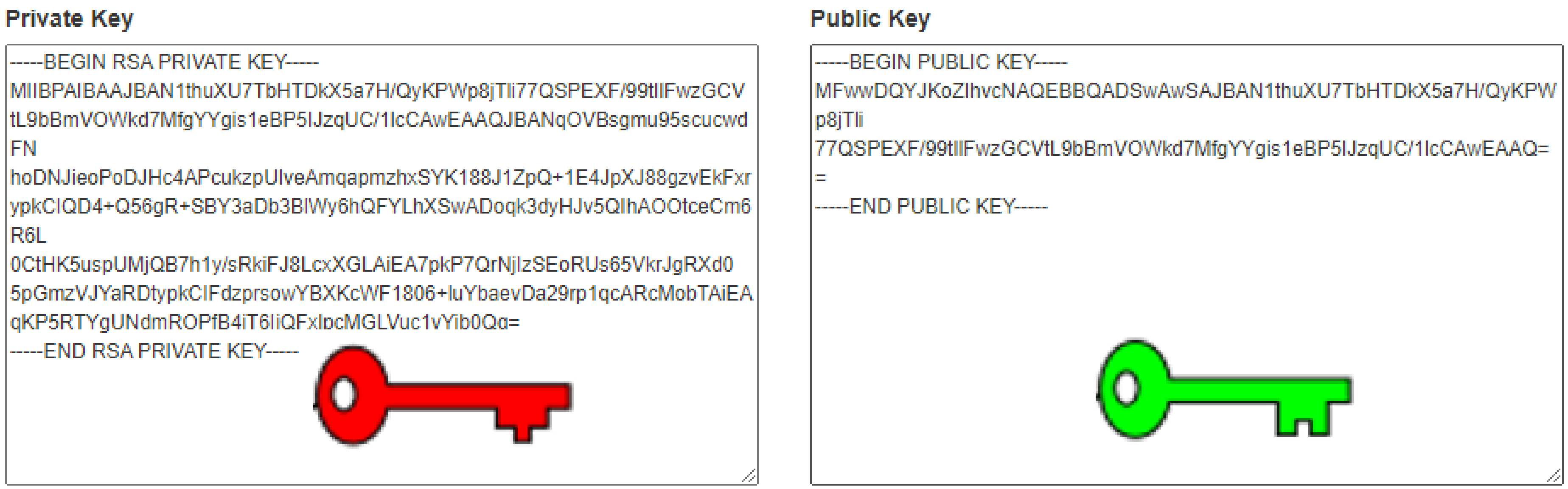
- Asymmetric Encryption Types
- Key Pairs
- Algorithms
- Key Exchange
- Protocols that Use Asymmetric Encryption

ASYMMETRIC ENCRYPTION

- Also known as Public Key Cryptography
- You have a pair of keys
  - Public key to encrypt
  - Private key to decrypt
  - Keys are mathematically related
- Excellent for protecting the symmetric encryption key
  - Asymmetric encryption is slow
  - Use symmetric encryption to encrypt the data
  - Then protect the symmetric encryption key with an asymmetric key pair
- Provides confidentiality and integrity
- You request (or create your own) public/private key pair
- You can freely give away your public key to anyone
- You must carefully guard the private key
  - Never let anyone else have access to it

ASYMMETRIC KEY PAIR

- Two keys that are mathematically related
- **Encrypt** with **public** key
  - Decrypt with related private key
- **Digitally sign** with the **private** key
  - Verify with the public key



Private Key

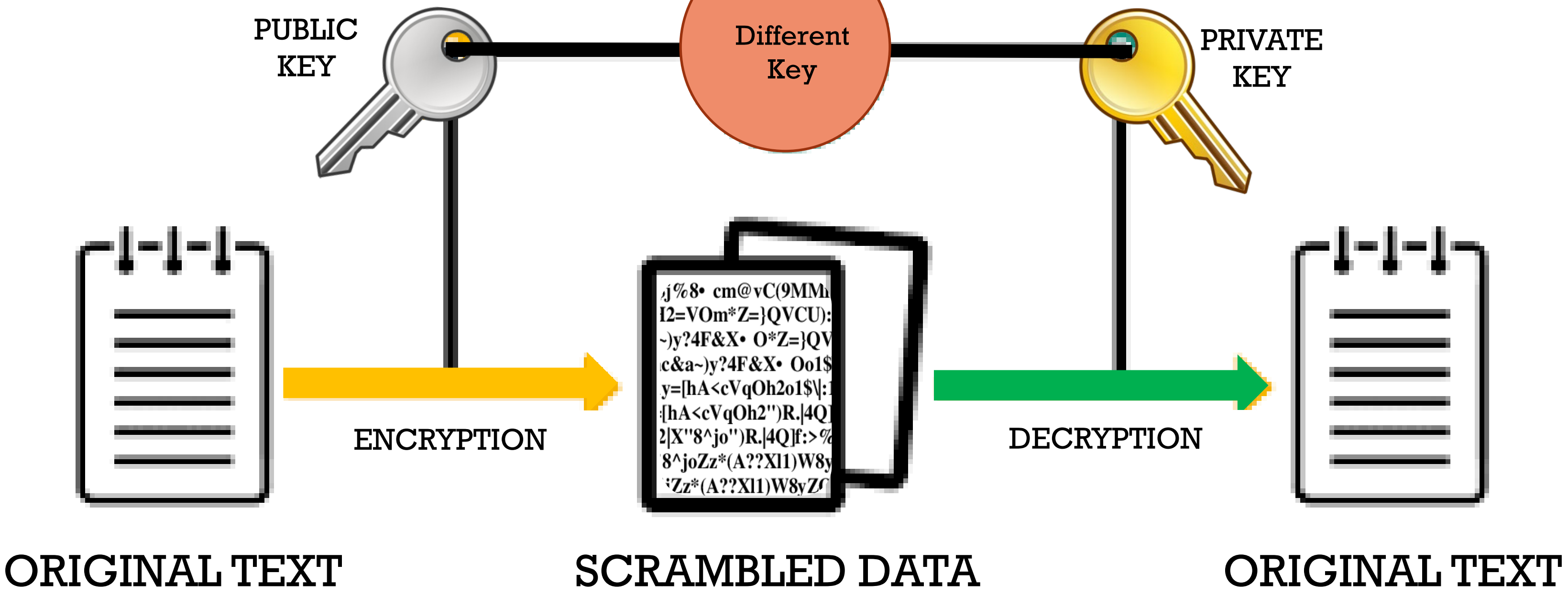
```
-----BEGIN RSA PRIVATE KEY-----
MIIBPAIBAAJBAN1thuXU7TbHTDkX5a7H/QyKPWp8jTli77QSPEXF/99tlIFwzGCV
tL9bBmVOWkd7MfgYYgis1eBP5lJzqUC/1lcCAwEAAQJBANqOVBsgmu95scucwd
FN
hoDNJieoPoDJHc4APcukzpUlveAmqapmzhxSYK188J1ZpQ+1E4JpXJ88gzvEkFxr
ypkClQD4+Q56gR+SBY3aDb3BlWy6hQFYLhXSwADoqk3dyHJv5QIhAOOtceCm6
R6L
0CtHK5uspUMjQB7h1y/sRkiFJ8LcxXGLAiEA7pkP7QrNjlzSEoRUs65VkrJgRXd0
5pGmzVJYaRDtypkCIFdzprsowYBXKcWF1806+luYbaevDa29rp1qcARcMobTAiEA
qKP5RTYgUNdmROPfB4iT6liQFxlpcMGLVuc1vYib0Qq=
-----END RSA PRIVATE KEY-----
```

Public Key

```
-----BEGIN PUBLIC KEY-----
MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAN1thuXU7TbHTDkX5a7H/QyKPW
p8jTli
77QSPEXF/99tlIFwzGCVtL9bBmVOWkd7MfgYYgis1eBP5lJzqUC/1lcCAwEAAQ=
=
-----END PUBLIC KEY-----
```

ASYMMETRIC ENCRYPTION EXAMPLE

Asymmetric Encryption

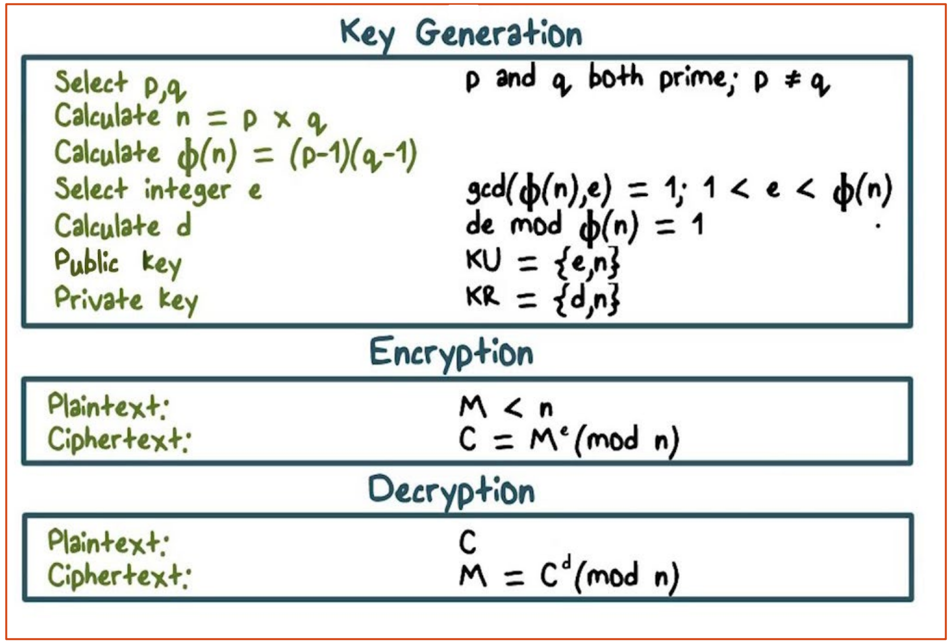


ASYMMETRIC ALGORITHMS

- RSA
  - De facto Internet encryption standard
  - Based on the practical difficulty of factoring the product of two large prime numbers
    - The factoring problem
- Diffie-Hellmann
  - Used for exchanging asymmetric keys
  - Diffie-Hellman (DH) groups determine the strength of the key used in the key exchange process
- ECC
  - Based on the algebraic structure of elliptic curves over finite fields
  - Can achieve the same level of security provided while using a shorter key length.
    - An ECC 256 = RSA 3072
    - Good for devices that have lower computing power
    - Smart cards
    - Mobile devices

ASYMMETRIC ALGORITHM EXAMPLES

RSA



Key Generation

Select p,q	p and q both prime; $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n),e) = 1;\ 1 < e < \phi(n)$
Calculate d	$de \bmod \phi(n) = 1$
Public key	$KU = \{e,n\}$
Private key	$KR = \{d,n\}$

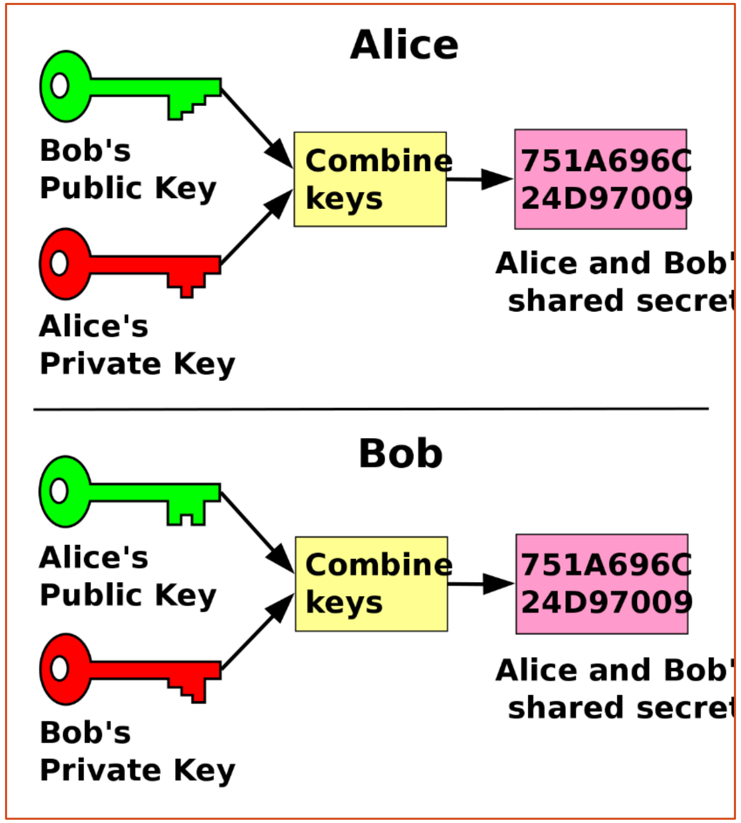
Encryption

Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod{n}$

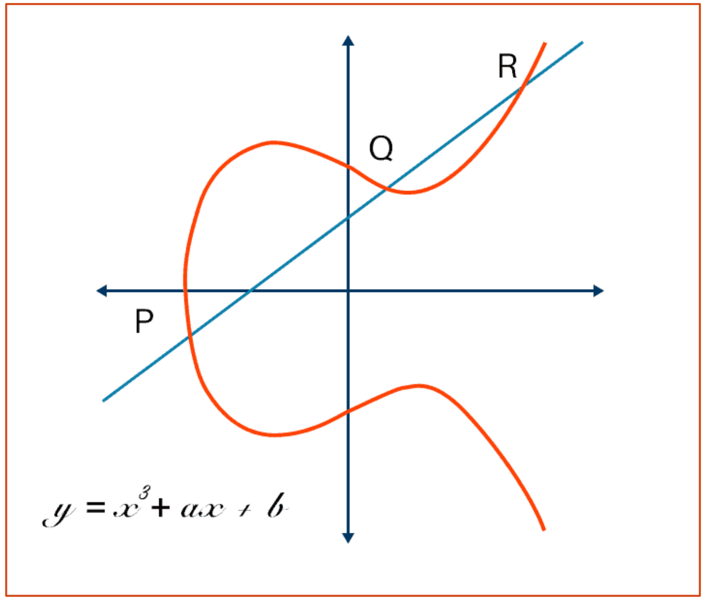
Decryption

Plaintext:	C
Ciphertext:	$M = C^d \pmod{n}$

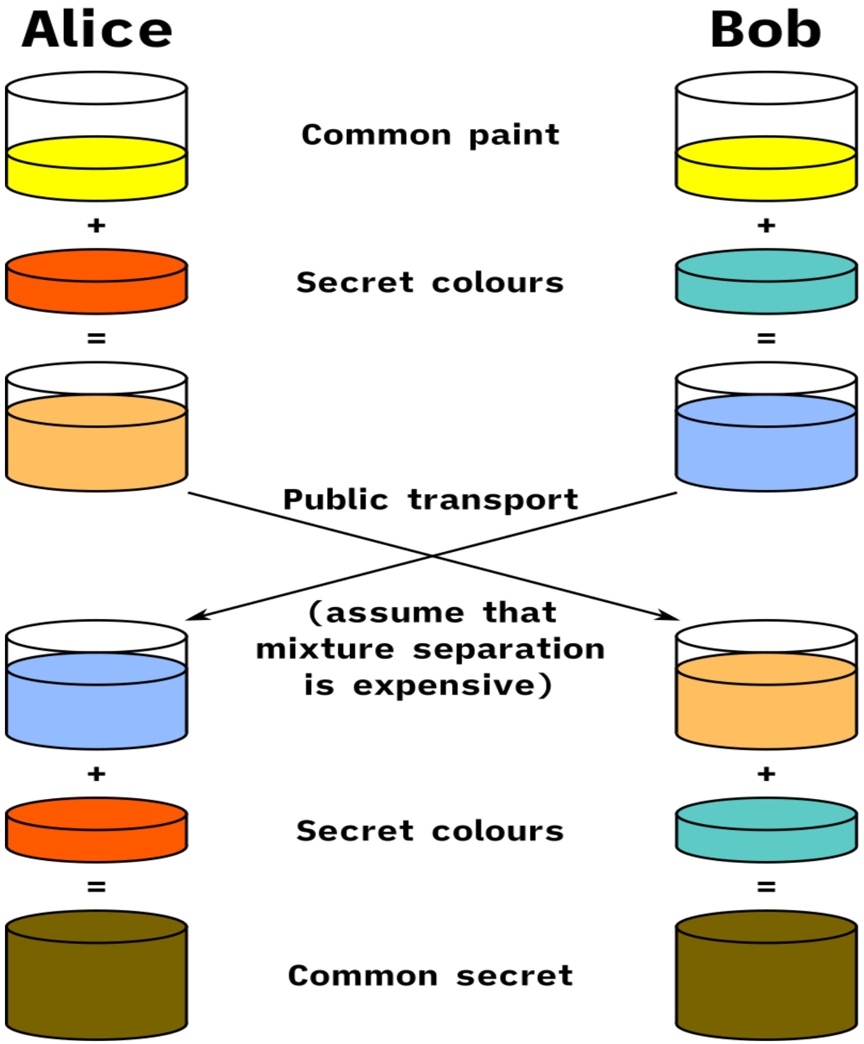
Diffie-Hellman



Elliptic Curve



DIFFIE-HELLMANN KEY EXCHANGE EXAMPLE



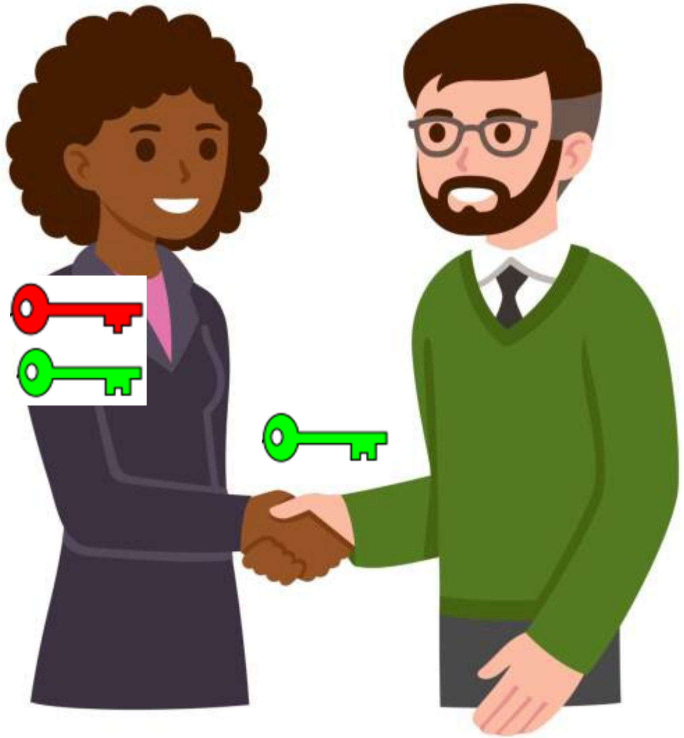
PROTOCOLS THAT USE ASYMMETRIC CRYPTOGRAPHY

- PGP/GPG
- SSL/TLS
- S/MIME
- SSH
- Internet Key Exchange (IKE) for IPSEC

20.4 PUBLIC KEY EXCHANGE

- Trading Keys
- Diffie-Hellmann
- PGP
- SSH Key Generation

TRADING PUBLIC KEYS



- Alice has an asymmetric key pair
- She can give Bob a copy of her public key
- Bob can then use her public key to send her an encrypted message
  - Alice will then use her private key to decrypt
- Alice can also use her private key to digitally sign messages
  - Bob can use her public key to verify the signature

DIFFIE-HELLMANN KEY EXCHANGE

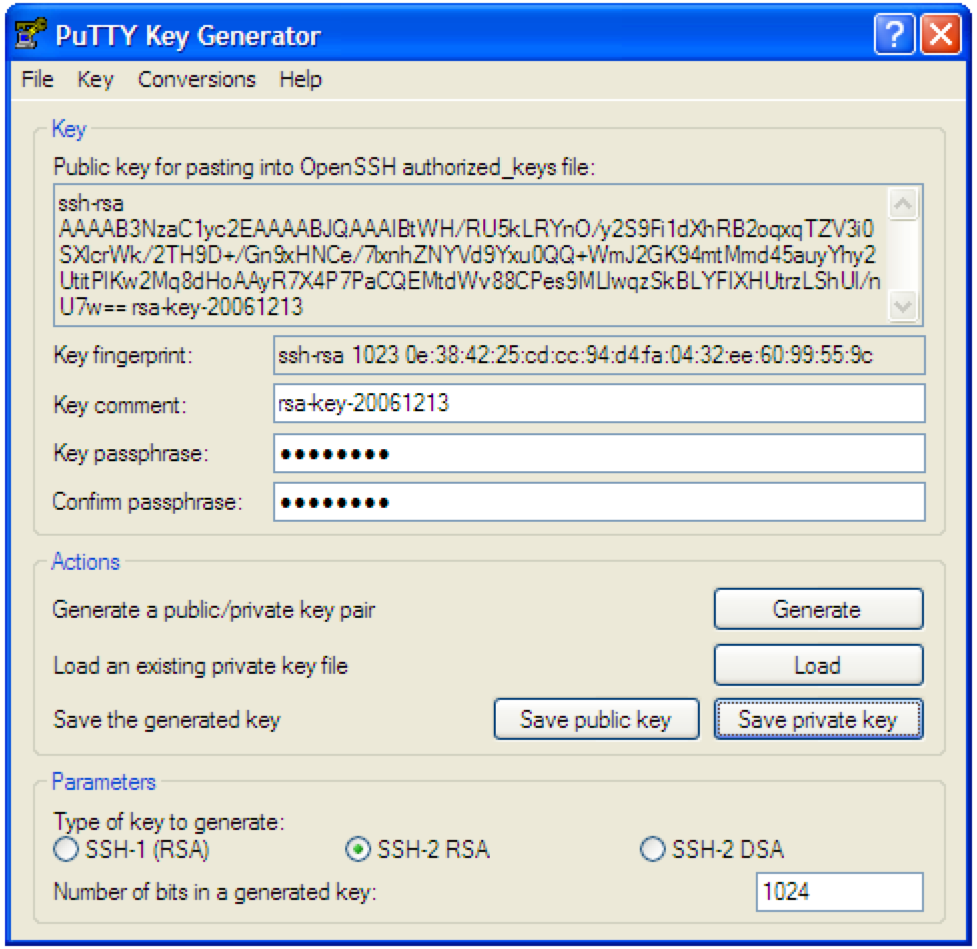
- Protocol for automatically exchanging public keys
- The first widely used method of safely developing and exchanging keys over an insecure channel
- Largely replaced by RSA, which has its own key exchange algorithm and can digitally sign certificates
- Diffie-Hellman Groups are used to determine the strength of the key used in the Diffie-Hellman key exchange process
  - Higher Diffie-Hellman Group numbers are more secure
  - But higher groups also require additional cpu power
- Commonly used DH Groups:
  - DH Group 1: 768-bit group
  - DH Group 2: 1024-bit group
  - DH Group 5: 1536-bit group
  - DH Group 14: 2048-bit group
  - DH Group 15: 3072-bit group

PRETTY GOOD PRIVACY (PGP)

- System for creating asymmetric key pairs and trading public keys
- Provides authentication and cryptographic privacy
- Used for digital signing, data compression, and to encrypt/decrypt emails, messages, files, and directories
- You can search MIT's PGP Public Key Server
  - Use information about the person such as their email address
  - If someone's public key is found, you can download it and put it on your key ring
- PGP was sold to Symantec in 2010
- Open source replacement is GPG

SSH KEY GENERATION

- Tools such as PuTTY can create a key pair
- You can then use the generated public key to establish an SSH session



Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAABJQAAAIBtWH/RU5kLRYnO/y2S9Fi1dXhRB2oqxqTZV3i0
SXlcrWk/2TH9D+/Gn9xHNCe/7lxnhZNYVd9Yxu0QQ+WmJ2GK94mtMmd45auyYhy2
UtitPIKw2Mq8dHoAAyR7X4P7PaCQEMtdWv88CPes9MLlwqzSkBLYFIXHUtrzLShUl/n
U7w== rsa-key-20061213
```

Key fingerprint: ssh-rsa 1023 0e:38:42:25:cd:cc:94:d4:fa:04:32:ee:60:99:55:9c

Key comment: rsa-key-20061213

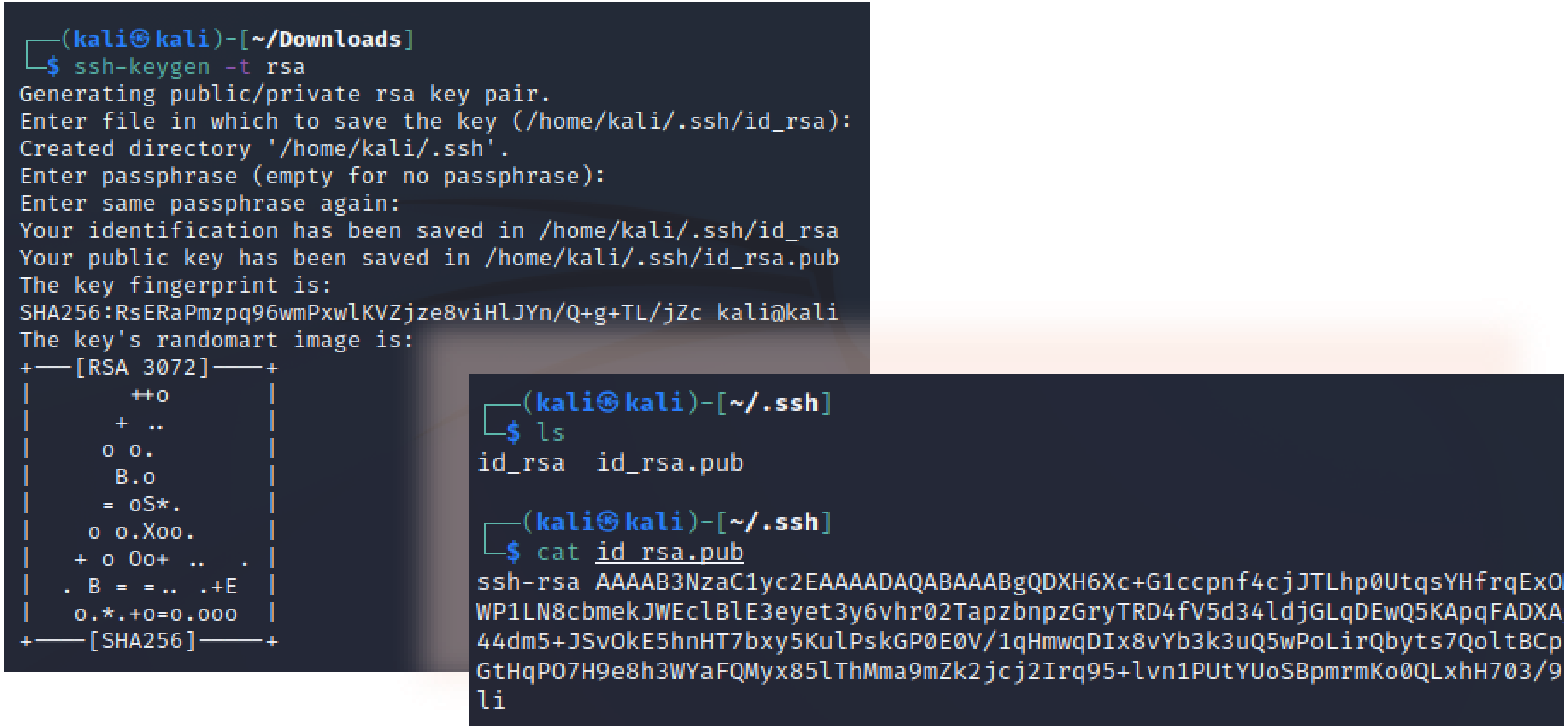
Key passphrase: ●●●●●●●●●

Confirm passphrase: ●●●●●●●●●

Type of key to generate:
○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 1024

GENERATING YOUR OWN KEY PAIR EXAMPLE



```
┌──(kali㉿kali)-[~/Downloads]
└─$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kali/.ssh/id_rsa):
Created directory '/home/kali/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/kali/.ssh/id_rsa
Your public key has been saved in /home/kali/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:RsERaPmzpq96wmPxwlKVZjze8viHlJYn/Q+g+TL/jZc kali@kali
The key's randomart image is:
+───[RSA 3072]────+
|       ++o       |
|      + ..       |
|     o o.        |
|      B.o        |
|     = oS*.      |
|    o o.Xoo.     |
|   + o Oo+ ..  . |
|  . B = =.. .+E  |
|   o.*.+o=o.ooo  |
+────[SHA256]─────+
```

```
┌──(kali㉿kali)-[~/.ssh]
└─$ ls
id_rsa  id_rsa.pub

┌──(kali㉿kali)-[~/.ssh]
└─$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDXH6Xc+G1ccpnf4cjJTLhp0UtqsYHfrqExO
WP1LN8cbmekJWEclBlE3eyet3y6vhr02TapzbnpzGryTRD4fV5d34ldjGLqDEwQ5KApqFADXA
44dm5+JSvOkE5hnHT7bxy5KulPskGP0E0V/1qHmwqDIx8vYb3k3uQ5wPoLirQbyts7QoltBCp
GtHqPO7H9e8h3WYaFQMyx85lThMma9mZk2jcj2Irq95+lvn1PUtYUoSBpmrmKo0QLxhH703/9
li
```

20.5 PKI

- Public Key Infrastructure
- PKI Components
- PKI Process
- Certificate Authorities
- Key Escrow

PUBLIC KEY INFRASTRUCTURE (PKI)

- PKI is an arrangement that ”binds” public keys with respective identities of entities
  - Such as people, organizations, devices, services
- PKI is a set of roles, policies, hardware, software and procedures
  - Used to create, manage, distribute, use, store and revoke digital certificates and manage public-key encryption
- Used to facilitate the secure electronic transfer of information for a range of network activities including:
  - e-commerce, internet banking, confidential email
- PKI is required for activities where:
  - Simple passwords are an inadequate authentication method
  - More rigorous proof is required to confirm the identity of the parties involved in the communication
  - The information being transferred needs to be validated

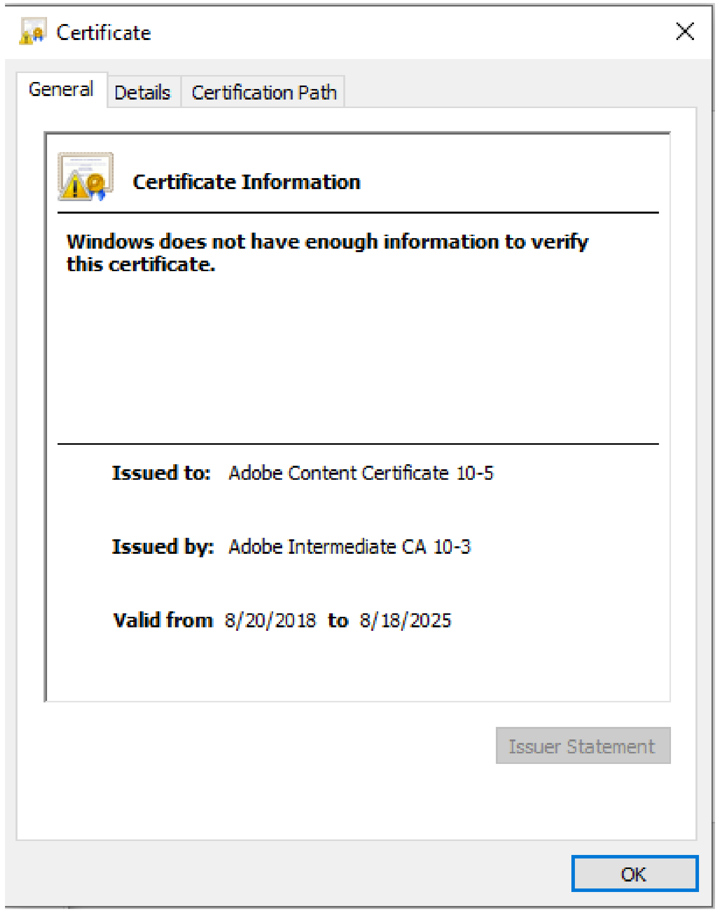
PKI COMPONENTS

- Certificate Authority (CA)
  - AKA Certification Authority
  - A service that registers and issues certificates
  - May be automated or manual
- Registration Authority
  - A role that may be delegated by a CA to assure valid and correct registration
  - Responsible for accepting requests for digital certificates and authenticating the entity making the request
- Validation Authority
  - Validates the identity of an entity bearing a certificate
- Certificates
  - A document issued by the CA
  - Contains the issued public key
  - Is accompanied by a private key

DIGITAL CERTIFICATES

- A public key on a document
  - Includes some metadata about the key
- Issued to the user, device, or service by a certification authority
- When initially issued to the user/device the certificate is accompanied by an encrypted private key
- The user/device downloads the certificate
- When they install the certificate on their device, it installs both keys in the device’s keystore
- Apps that need to use asymmetric encryption can then obtain access to the keys

DIGITAL CERTIFICATE EXAMPLE



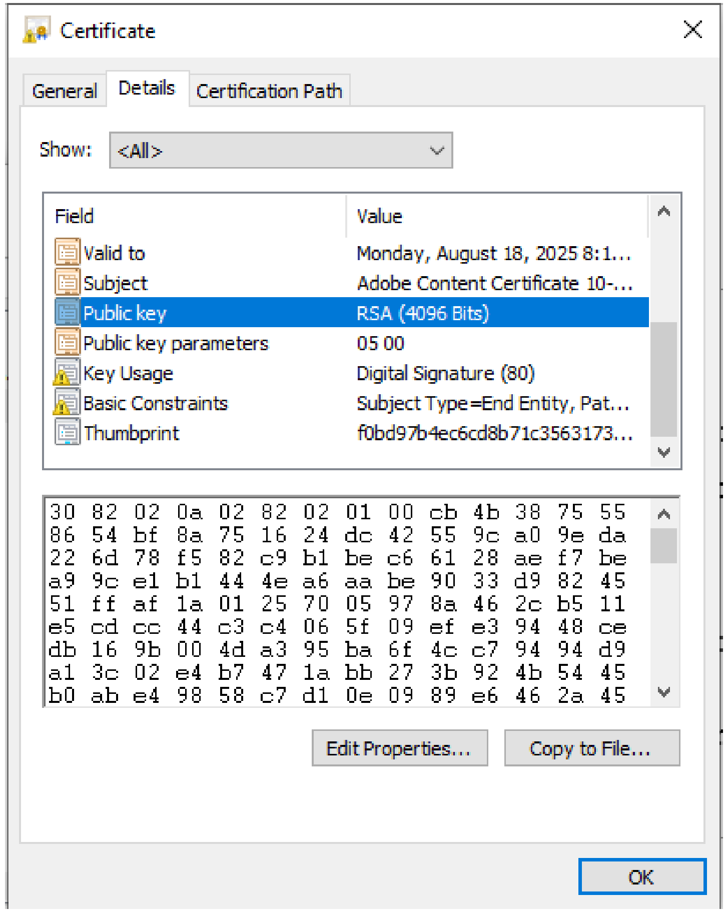
Certificate Information

Windows does not have enough information to verify this certificate.

Issued to: Adobe Content Certificate 10-5

Issued by: Adobe Intermediate CA 10-3

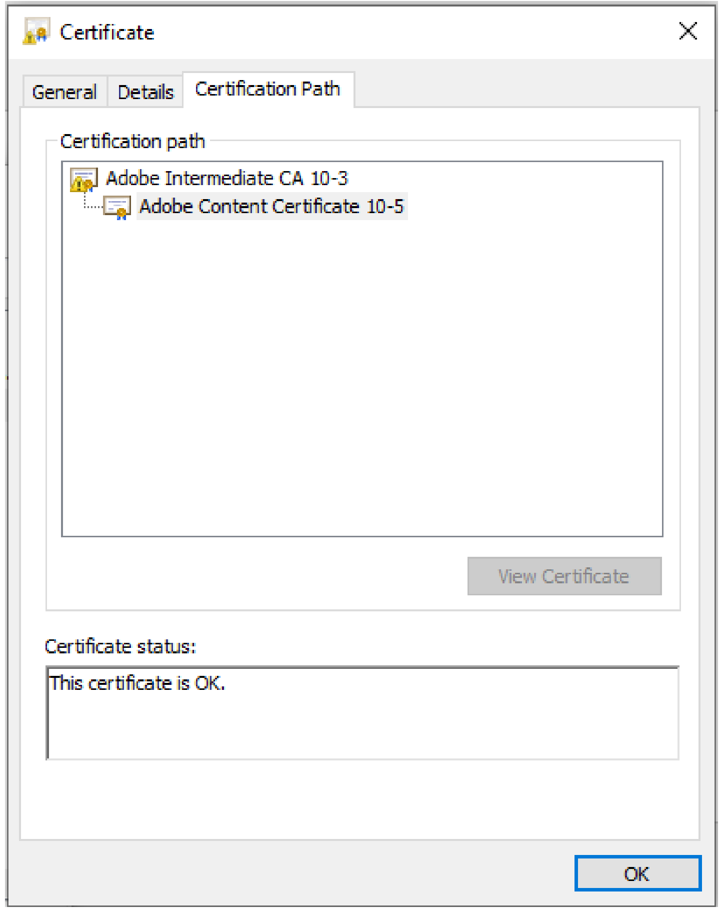
Valid from 8/20/2018 **to** 8/18/2025



Show: <All>

Field	Value
Valid to	Monday, August 18, 2025 8:1...
Subject	Adobe Content Certificate 10-...
Public key	RSA (4096 Bits)
Public key parameters	05 00
Key Usage	Digital Signature (80)
Basic Constraints	Subject Type=End Entity, Pat...
Thumbprint	f0bd97b4ec6cd8b71c3563173...

```
30 82 02 0a 02 82 02 01 00 cb 4b 38 75 55
86 54 bf 8a 75 16 24 dc 42 55 9c a0 9e da
22 6d 78 f5 82 c9 b1 be c6 61 28 ae f7 be
a9 9c e1 b1 44 4e a6 aa be 90 33 d9 82 45
51 ff af 1a 01 25 70 05 97 8a 46 2c b5 11
e5 cd cc 44 c3 c4 06 5f 09 ef e3 94 48 ce
db 16 9b 00 4d a3 95 ba 6f 4c c7 94 94 d9
a1 3c 02 e4 b7 47 1a bb 27 3b 92 4b 54 45
b0 ab e4 98 58 c7 d1 0e 09 89 e6 46 2a 45
```



Certification path

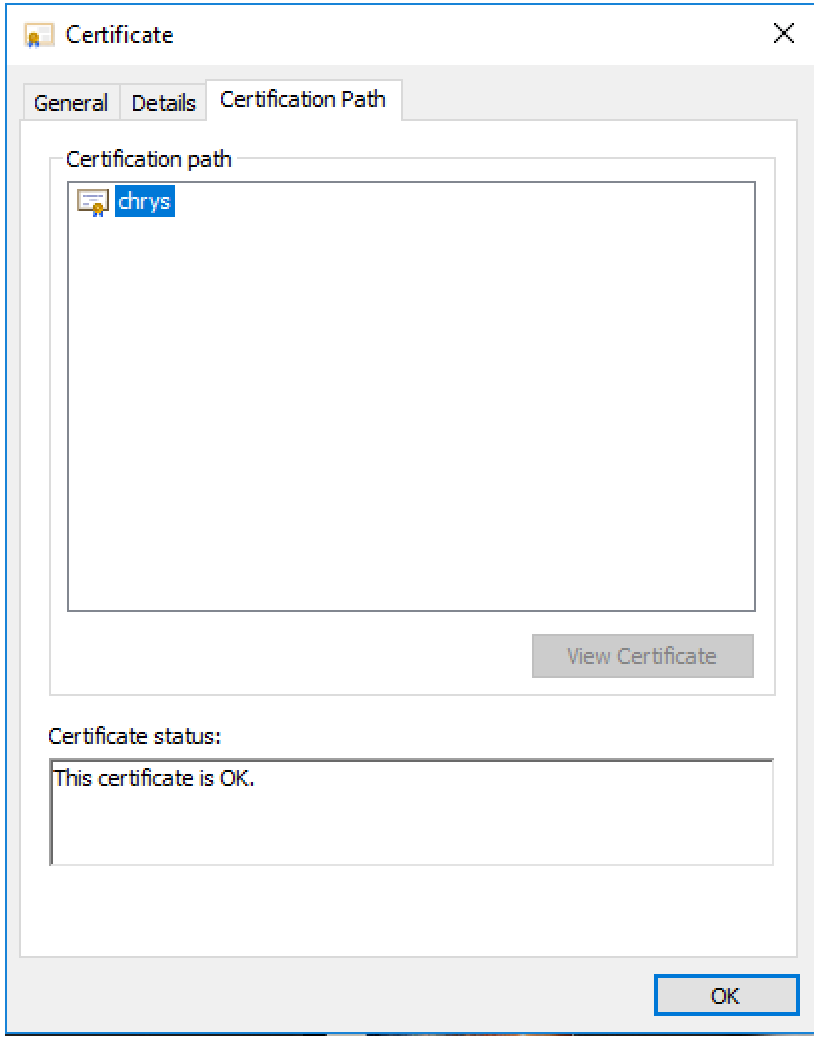
- Adobe Intermediate CA 10-3
  - Adobe Content Certificate 10-5

Certificate status:

This certificate is OK.

SELF-SIGNED CERTIFICATE

- User creates private and public keys using any available tool
- User self-signs document with public key
- Document delivered to receiver
- Public keys are traded
- A temporary symmetric session key is created
- The session key is protected by our public keys, which can only be decrypted by our private keys

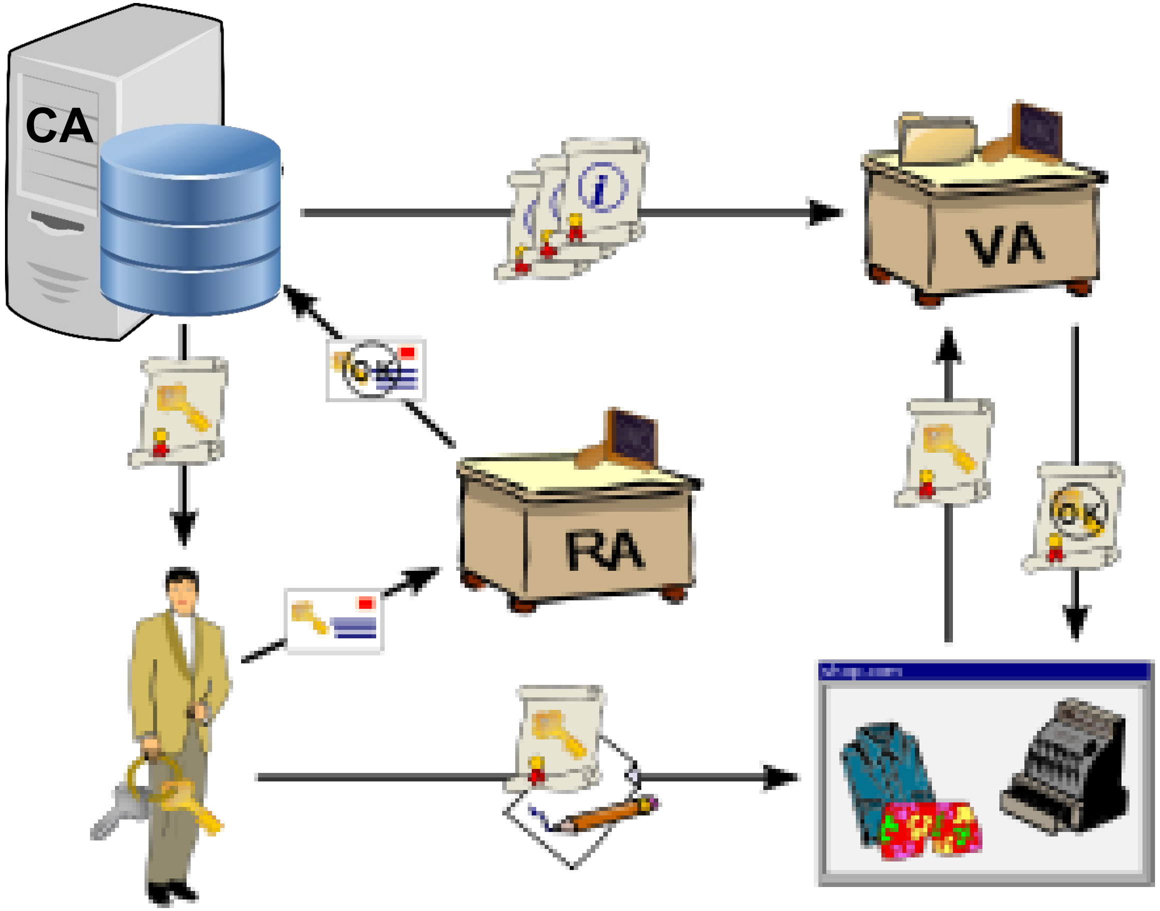


chrys

Certificate status:

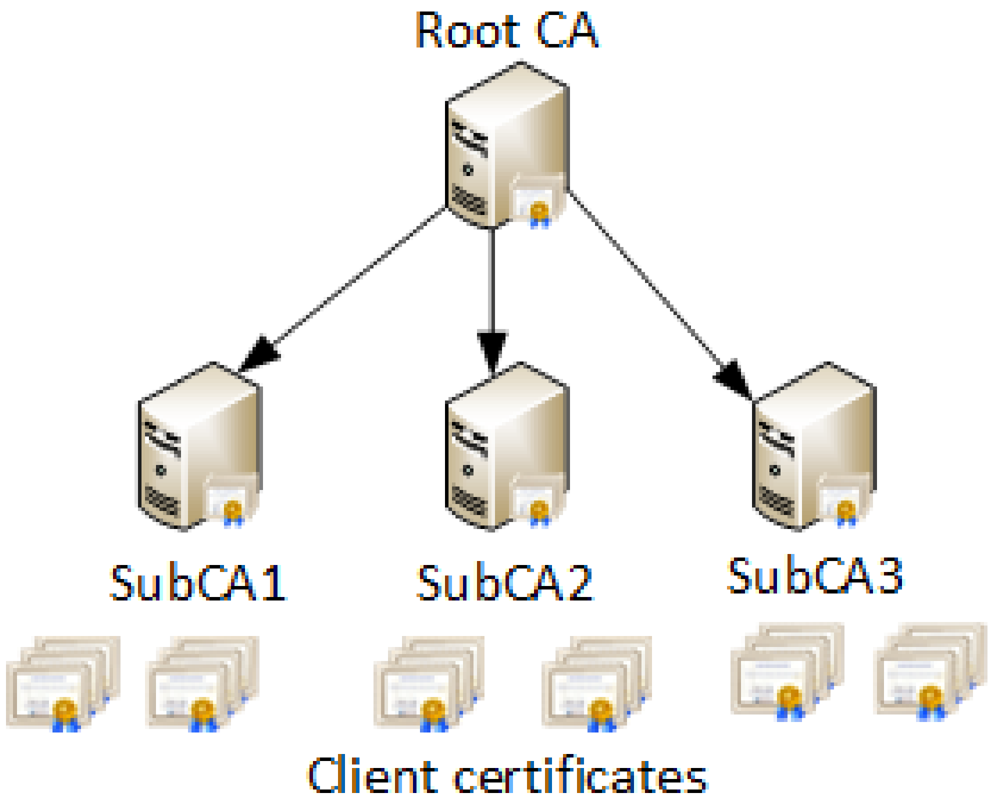
This certificate is OK.

PKI PROCESS

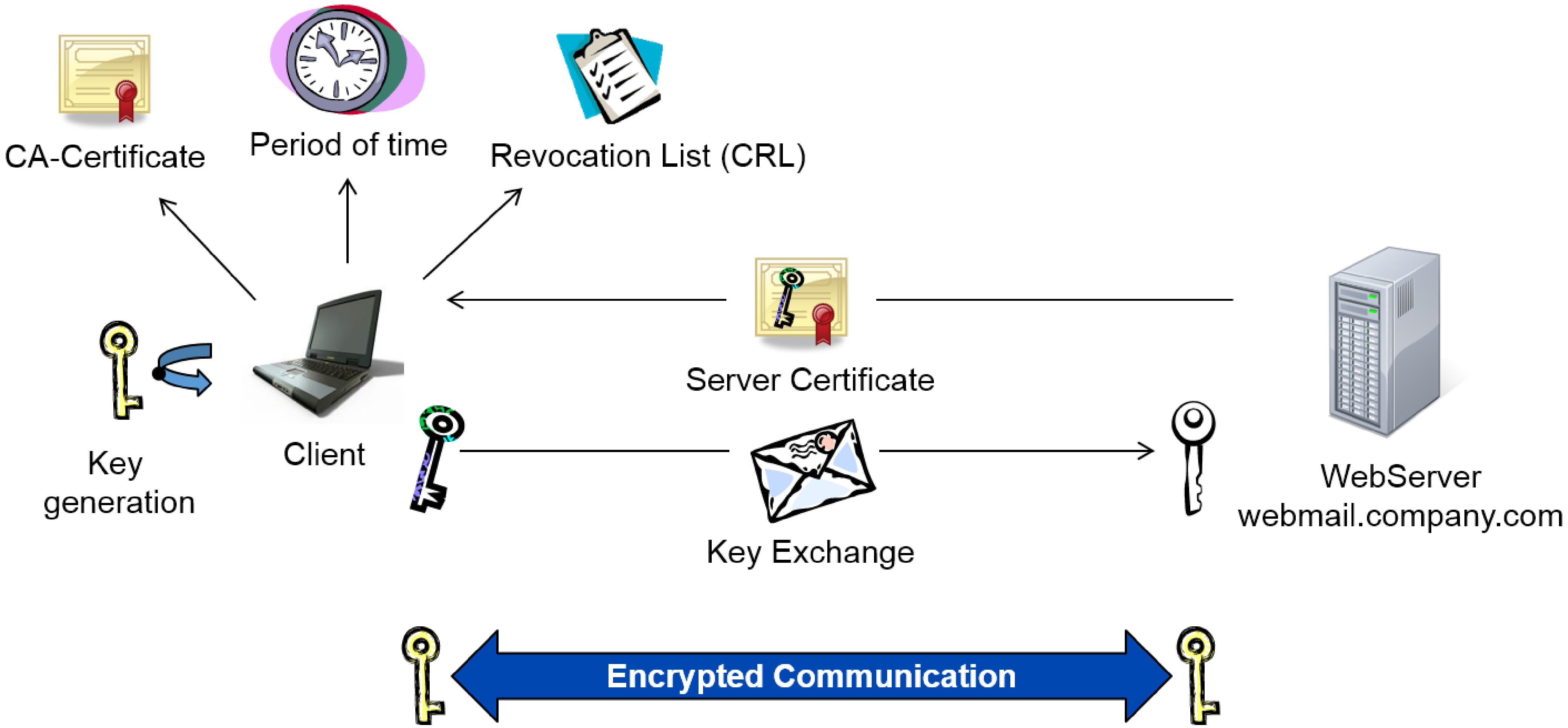


CERTIFICATE AUTHORITY HIERARCHY

- A Root CA is the highest authority
- It issues certificates to digitally sign subordinate CAs
- The subordinate CAs issue certificates to users and clients

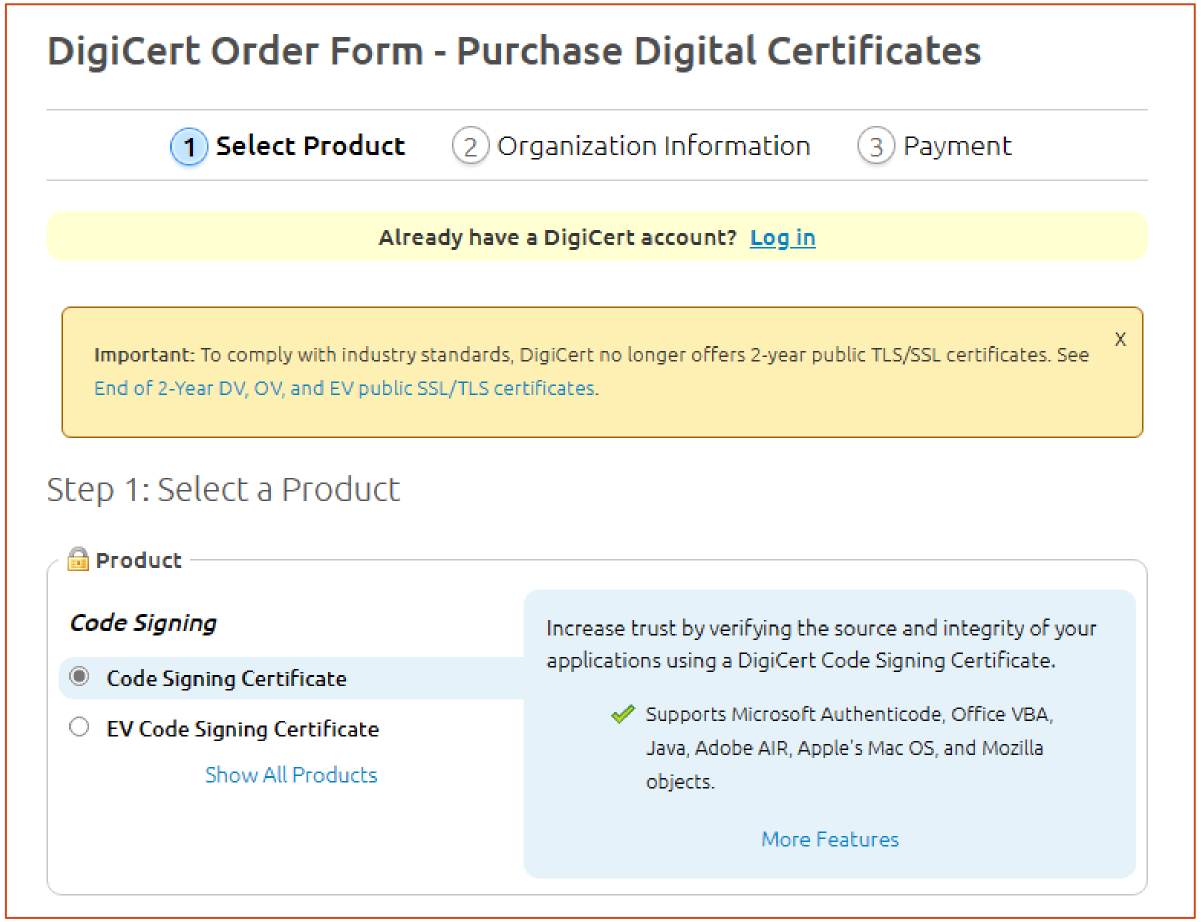


USING A DIGITAL CERTIFICATE



POPULAR CERTIFICATION AUTHORITIES

- VeriSign
- Digicert
- Godaddy
- Microsoft
- COMODO
- Norton Symantec
- Thawte
- Entrust



DigiCert Order Form - Purchase Digital Certificates

Already have a DigiCert account? Log in

Important: To comply with industry standards, DigiCert no longer offers 2-year public TLS/SSL certificates. See End of 2-Year DV, OV, and EV public SSL/TLS certificates.

Step 1: Select a Product

Product

Code Signing

● Code Signing Certificate

○ EV Code Signing Certificate

Show All Products

Increase trust by verifying the source and integrity of your applications using a DigiCert Code Signing Certificate.

✓ Supports Microsoft Authenticode, Office VBA, Java, Adobe AIR, Apple's Mac OS, and Mozilla objects.

More Features

KEY ESCROW

- A special component of PKI
- A copy of a private key is stored to provide third-party access and to facilitate recovery operations
- The private key is held in escrow, or stored, by a third party
- A key that is lost or compromised by its original user(s) may be used to decrypt encrypted material
- Allows restoration of the original material to its unencrypted state
- Keys held in escrow can also be divided into parts
  - Each part is stored by a different entity
  - All parts must be retrieved and put together to recreate the private key
  - This reduces the risk of fraud and collusion

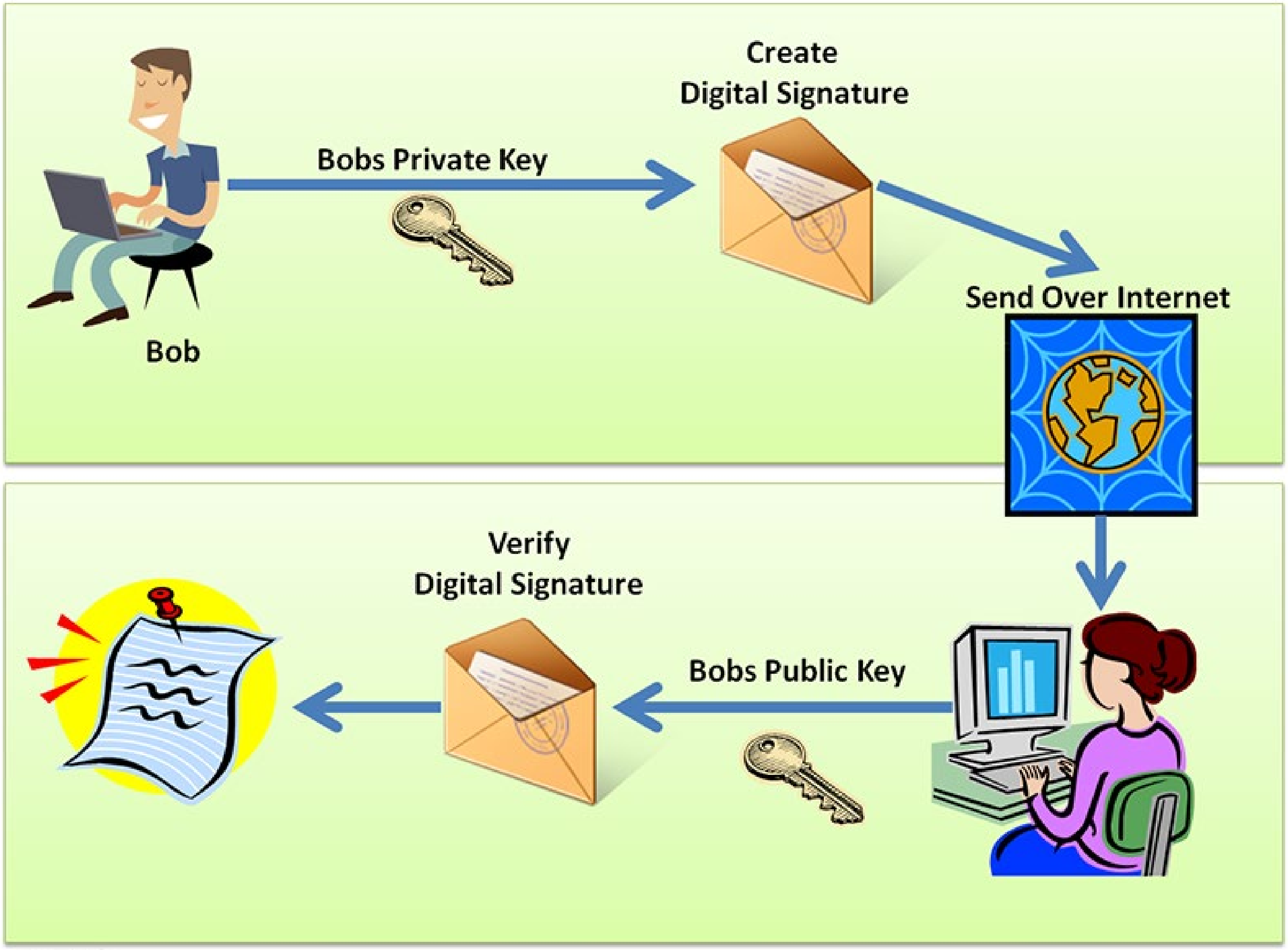
20.6 DIGITAL SIGNATURES

- Digital Signature
- Digital Signature Process
- Digital Signature Schemes

DIGITAL SIGNATURE

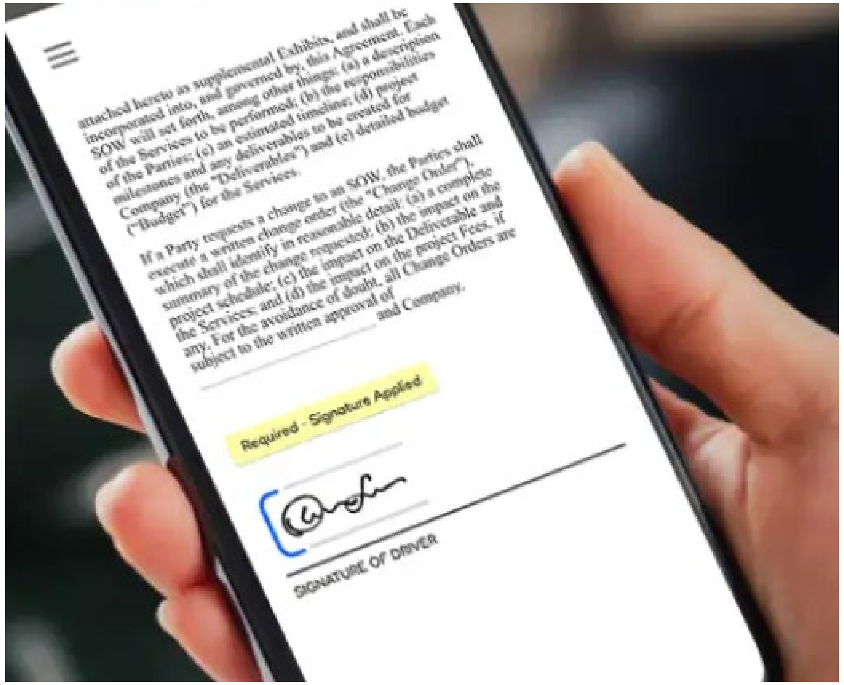
- Uses asymmetric cryptography
- Simulates security properties of a written signature in digital form
- Created with the user’s private key
- Accompanies the file/network packet/code
- Proves the integrity and identity of the files/network packets/code it signs

DIGITAL SIGNATURE PROCESS

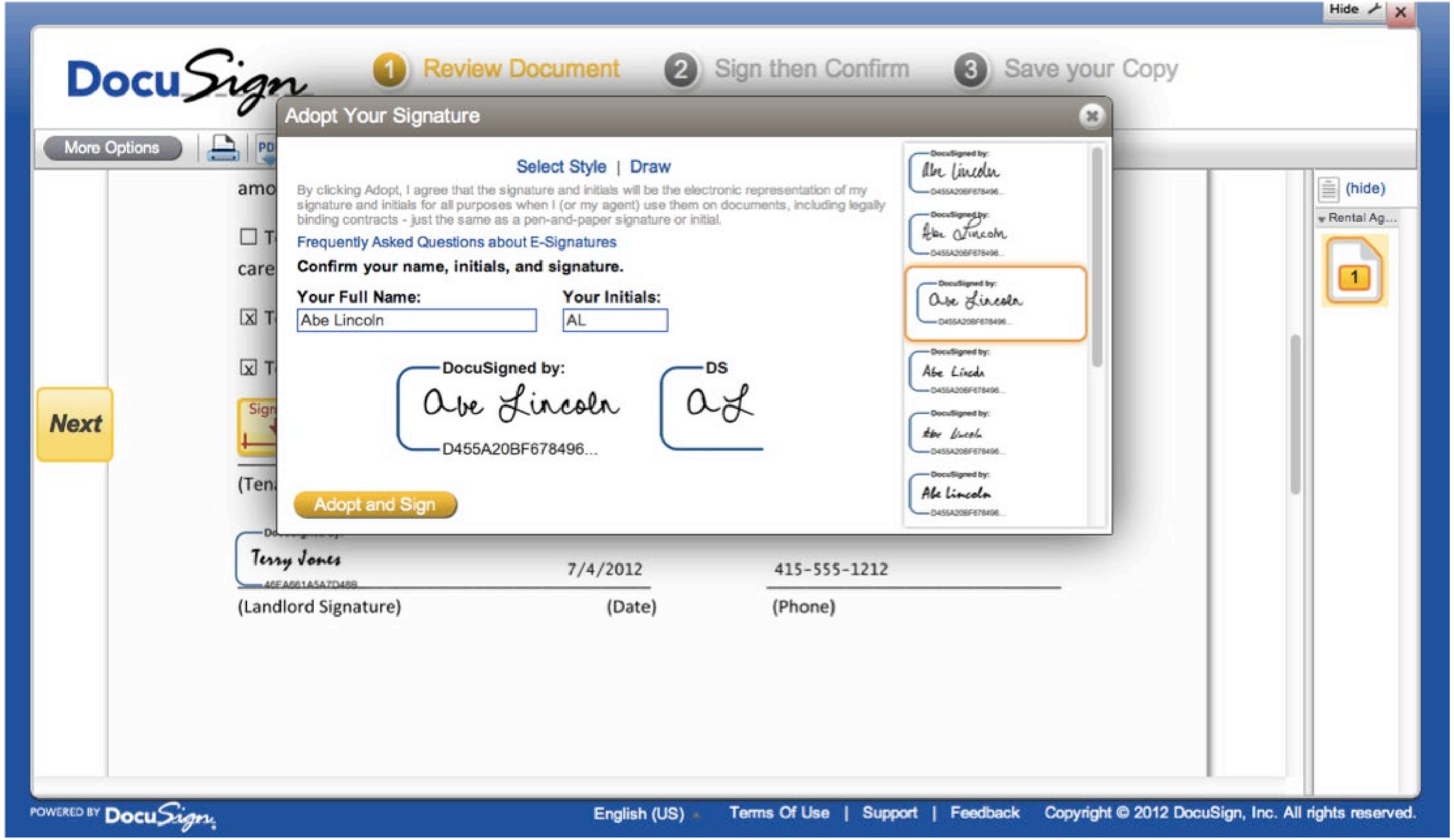


DIGITAL SIGNATURE SCHEMES

- RSA
  - Used by various apps including:
    - MS Office
    - Adobe Acrobat Pro
    - DNS Servers and clients using DNSSEC
    - Online services like DocuSign
- Digital Signature Algorithm (DSA)
  - Specific by FIPS 186-2
  - Used to generate and verify digital signatures
    - For unclassified, sensitive applications



DOCUSIGN EXAMPLE



Adopt Your Signature

Select Style | Draw

By clicking Adopt, I agree that the signature and initials will be the electronic representation of my signature and initials for all purposes when I (or my agent) use them on documents, including legally binding contracts - just the same as a pen-and-paper signature or initial.

Frequently Asked Questions about E-Signatures

Confirm your name, initials, and signature.

Your Full Name: Abe Lincoln

Your Initials: AL

DocuSigned by: Abe Lincoln D455A20BF678496...

DS AL

Terry Jones 7/4/2012 415-555-1212

(Landlord Signature) (Date) (Phone)

Copyright © 2012 DocuSign, Inc. All rights reserved.

DIGITAL SIGNATURE CONSIDERATIONS

- You cannot move or copy a digital signature from one document to another
  - Each document/packet/file must have its own signature
  - The signature is a hash of the original document encrypted with the private key of the signing party
- The digital signature must be unforgeable and authentic
- You can be legally liable for documents that contain your digital signature
- Both the sender and receiver must have the ability to use the digital signatures
  - For example: DNSSEC is a specification that allows a DNS server to attach digital signatures to DNS records
  - In reality, since DNSSEC is an add-on capability, most Internet clients are not configured to use it

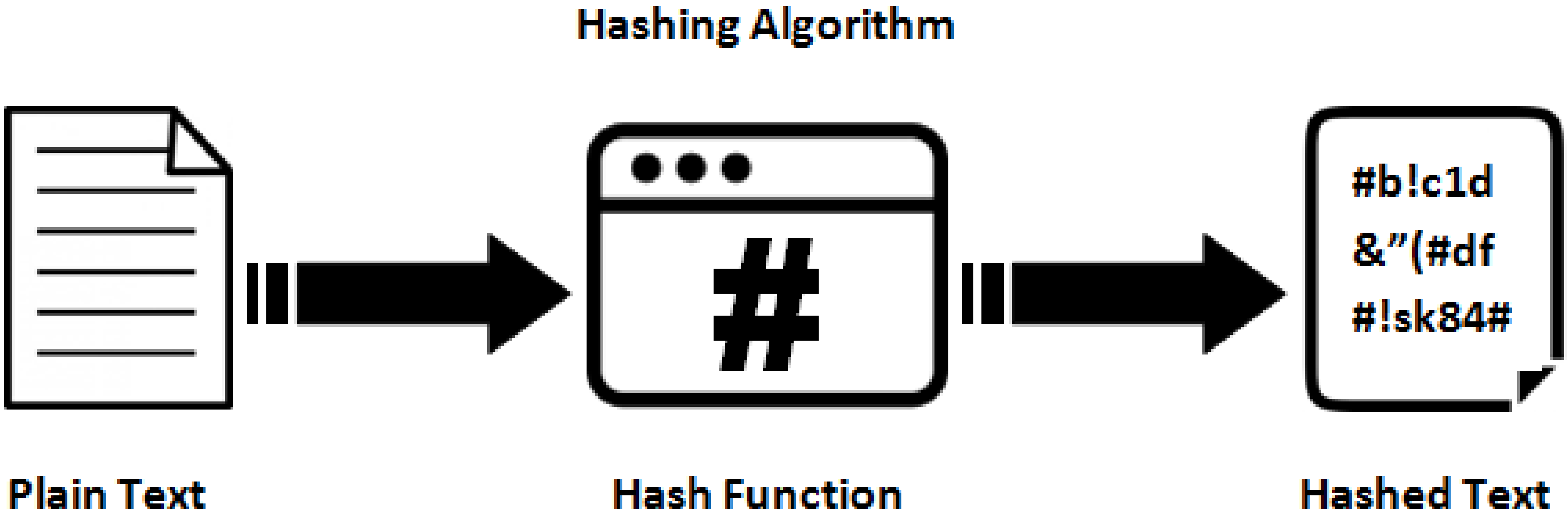
20.7 HASHING

- Hash
- Algorithms
- Hashing in Cyber Forensics
- Pass-the-Hash

HASHING

- Any function that can be used to map data of arbitrary size to data of fixed size
- Used to assure integrity of a file, packet, or any other stored or transmitted data
- Creates a one-way "encryption"
- Does not require a key
- Does not modify the original file/data
- Produces a fixed-length output, regardless of the size of the input
- The values returned by a hash function are called hash values, hash codes, digests, or simply hashes
- Any slight change to the input dramatically changes the output
- Used to securely store passwords

HASHING EXAMPLE



REQUIREMENTS FOR AN EFFECTIVE HASHING ALGORITHM

- Computationally infeasible to decrypt
- Resistant to collisions
  - Two different inputs must not create the same output

A collision attack is an attempt to find two input strings of a hash function that produce the same hash result.

POPULAR HASHING ALGORITHMS

- Original message: hello
- Message Digest MD2/MD4/MD5 – 128 bit
  - MD5 32 hex numbers - 5d41402abc4b2a76b9719d911017c592
- Secure Hash Algorithm
  - SHA-1 – 160 bit - 40 hex numbers - aaf4c61ddcc5e8a2dabede0f3b482cd9aea9434d
  - SHA-2:
    - SHA-256 - 64 hex numbers 2cf24dba5fb0a30e26e83b2ac5b9e29e1b161e5c1fa7425e73043362938b9824
    - SHA-384 - 96 hex numbers 59e1748777448c69de6b800d7a33bbfb9ff1b463e44354c3553bcdb9c666fa90125a3c79f90397bdf5f6a13de828684f
    - SHA-512 - 128 hex numbers 75d527c368f2efe848ecf6b073a36767800805e9eef2b1857d5f984f036eb6df891d75f72d9b154518c1cd58835286d1da9a38deba3de98b5a53e5ed78a84976
  - SHA-3
    - The latest version of SHA
    - Same hash lengths as SHA-2
    - Internal structure is significantly different
    - Currently the strongest hashing algorithm
- RIPEMD – 160 bit - 40 hex numbers 108f07b8382412612c048d07d13f814118445acd

MICROSOFT HASHING ALGORITHMS

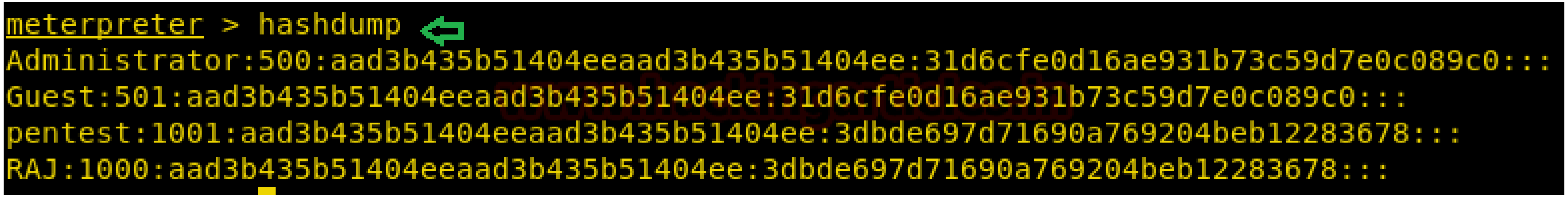
- LAN Manager (LM)
- A weak implementation of DES
  - Password is restricted to a maximum of 14 characters
  - Converts passwords to uppercase
  - Any password less than 14 characters is "NULL padded" to bring it to 14 characters
  - The 14 characters are then split into two 7-byte halves
  - Each half is used to create a 56-bit DES key
    - The DES keys are used to encrypt their respective half of the password
    - The two password halves are concatenated to create a 14-byte LM hash
  - The NULL padding is easy to identify, even when encrypted
  - Hashes are sent in clear text over the network.
  - Still used for backward compatibility
- NT Hash
  - Unicode characters
  - 128 bit
  - Unsalted MD4

THE ROLE OF HASHING IN CYBER FORENSICS

- The first thing that must be done after acquiring a forensic disk image is to:
  - Create a hash digest of the source drive and destination image file
  - Ensure they are identical
- A critical step in the presentation of evidence will be to prove:
  - Analysis has been performed on an identical image to the data present on the physical media
  - Neither data set has been tampered with
- The standard means of proving this is to create a cryptographic hash (fingerprint) of the disk contents and any derivative images made from it
- When comparing hash values, you need to use the same algorithm used to create the reference value

PASS-THE-HASH ATTACK

- A hacking technique that allows an attacker to authenticate without the password
  - The username and password are not entered normally at a login screen
  - Instead, the password *hash* is provided over the network using a special app
- Used when a password is too difficult to crack
- Requires the attacker to obtain the password hash ahead of time
- Hashes can be dumped from memory using tools such as:
  - Mimikatz, psexec, Metasploit meterpreter, fgdump, pwdump, cachedump, etc.



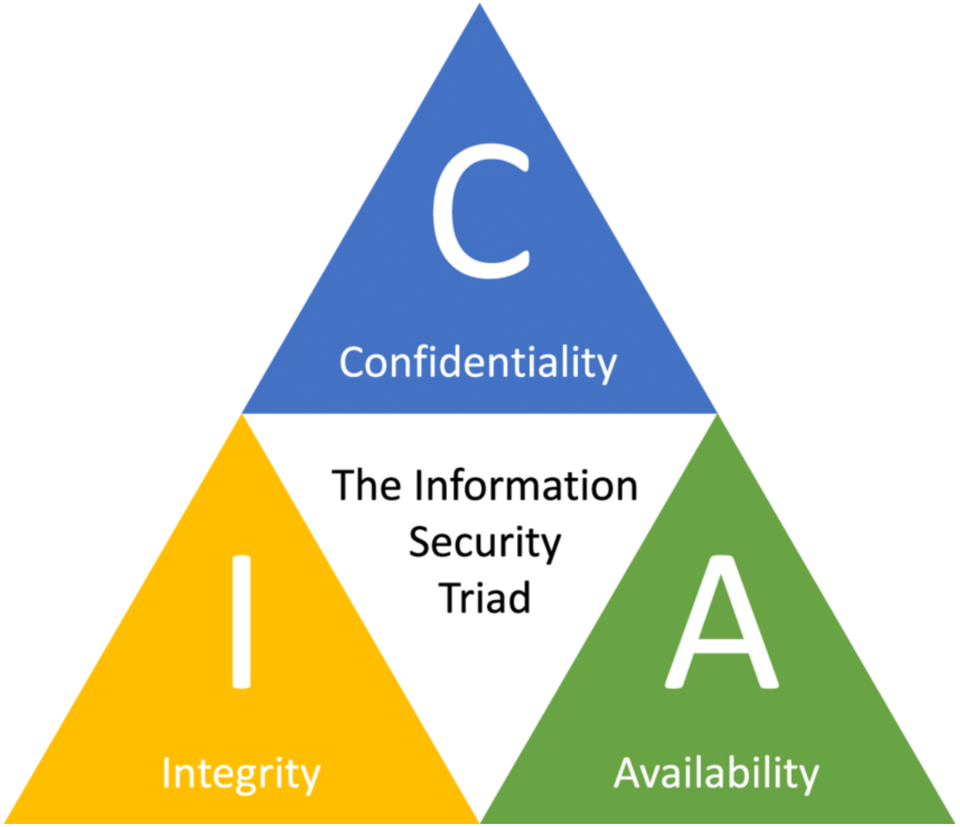
```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
pentest:1001:aad3b435b51404eeaad3b435b51404ee:3dbde697d71690a769204beb12283678:::
RAJ:1000:aad3b435b51404eeaad3b435b51404ee:3dbde697d71690a769204beb12283678:::
```

20.8 COMMON CRYPTOGRAPHY USE CASES

- Primary Use Cases
- Disk
- Email
- Network Communications
- VPN

PRIMARY USE CASES

- Encryption
  - Protect Confidentiality
- Hashing
  - Protect Integrity
- Digital Signatures
  - Authenticate
  - Protect Authenticity
  - Non-repudiation



DISK ENCRYPTION TYPES

Disk encryption protects data at rest

- File system encryption
  - Encrypt file system pointers that tell the OS where to find a file
- File encryption
  - Specific files or folders are themselves encrypted
- Full disk encryption
  - Secures all data stored on your hard drives
    - automatically and transparently
  - Includes swap files and hidden files
  - Does not require any user intervention
  - Does not protect data in transit
    - Data is unencrypted before it is:
      - attached to an email
      - transmitted over the network
      - copied to a USB stick

POPULAR DISK ENCRYPTION PRODUCTS

- Microsoft BitLocker
- Broadcom Symantec Endpoint Encryption
- Apple FileVault
- Check Point Harmony Endpoint
- ESET PROTECT
- McAfee Complete Data Protection
- Trend Micro Endpoint Encryption
- Micro Focus ZENworks Full Disk Encryption
- Rohde And Schwarz (R&S) Trusted Disk
- Sophos Central Device Encryption

DISK ENCRYPTION SCENARIO

- Moo travels a lot
- He worries that his laptop containing confidential documents might be stolen
- What do you suggest to address his concerns?
- Use full disk encryption on his laptop to protect his data

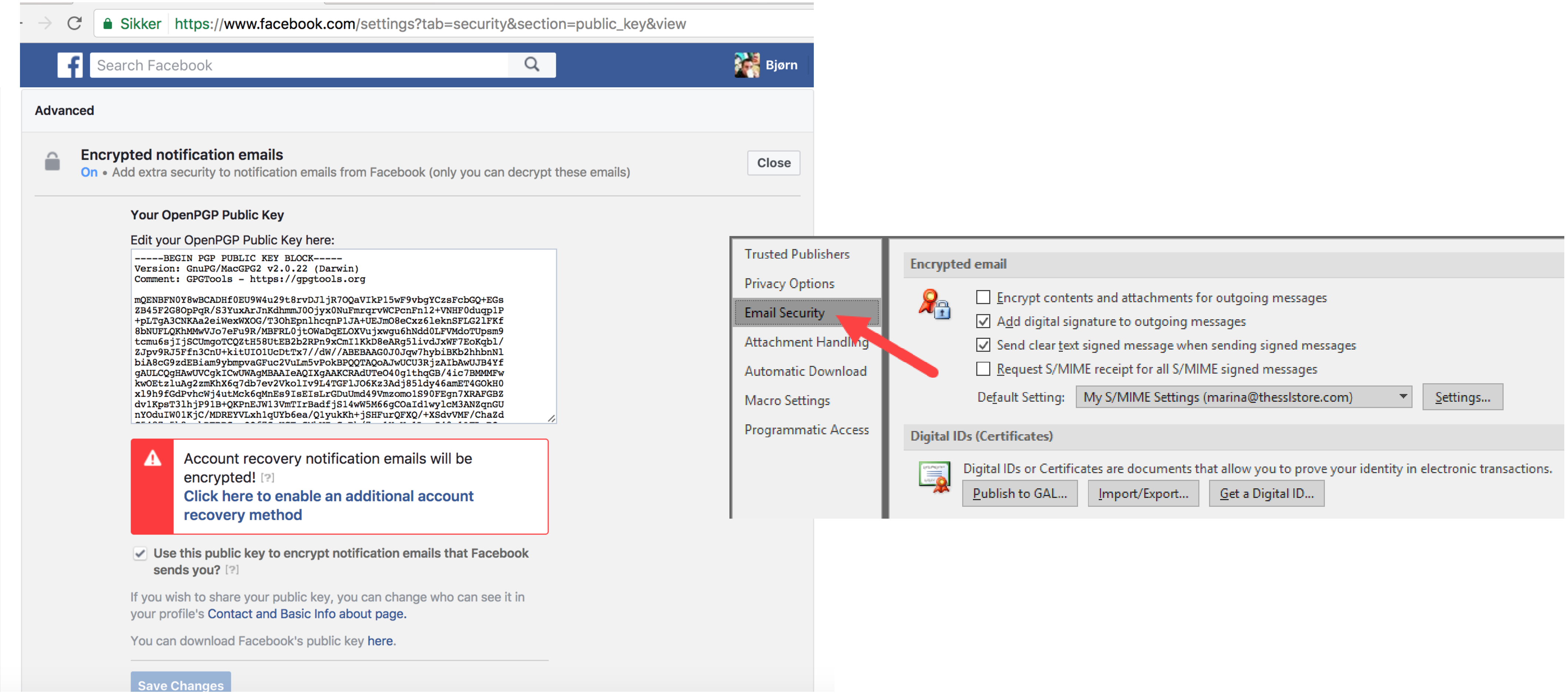
EMAIL ENCRYPTION

- Encrypting Email

ENCRYPTING EMAIL

- You can use an online secure email provider or your local email client
- Obtain or create a certificate (public key)
- Select the certificate in the email client
  - Alternatively, upload the certificate to the email provider
- In an enterprise environment, users' certificates are distributed and managed by the email server and/or directory service
- SMTP does not encrypt by default
- `STARTTLS` is the SMTP command to transmit email over TLS

SECURING EMAIL EXAMPLES



Sikker https://www.facebook.com/settings?tab=security&section=public_key&view

Advanced

Encrypted notification emails
On • Add extra security to notification emails from Facebook (only you can decrypt these emails)

Your OpenPGP Public Key

Edit your OpenPGP Public Key here:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG/MacGPG2 v2.0.22 (Darwin)
Comment: GPGTools - https://gpgtools.org

mQENBFN0Y8wBCADHf0EU9W4u29t8rvDJljR7OQaVIkPl5wF9vbgYCzsFcbGQ+EGs
ZB45F2G8OpPqR/S3YuxArJnKdhmmJ0Ojyx0NuFmrqrvWCPcnFn12+VNHF0duqp1P
+pLTgA3CNKAa2eiWexWXOG/T3OhEpnlhcqnP1JA+UEJmO8eCxz6leknSFLG2lFKf
8bNUFLQKhMMwVJo7eFu9R/MBFRL0jtOWaDqELOXVujxwgu6hNdd0LFVMdoTUpsm9
tcmu6sjIjSCUmgoTCQZtH58UtEB2b2RPn9xCmIlKkD8eARg5livdJxWF7EoKqbl/
ZJpv9RJ5Ffn3CnU+kitUIO1UcDtTx7//dW//ABEBAAG0J0Jqw7hybiBKb2hhbnNl
biA8cG9zdEBiam9ybmpvaGFuc2VuLm5vPokBPQQTAQoAJwUCU3RjzAIbAwUJB4Yf
gAULCQgHAwUVCgkICwUWAgMBAAIeAQIXgAAKCRAdUTeO40g1thqGB/4ic7BMMMFw
kwOEtzluAg2zmKhX6q7db7ev2VkolIv9L4TGFlJO6Kz3Adj85ldy46amET4GOkH0
xl9h9fGdPvhcWj4utMck6qMnEs9IsEIsLrGDuUmd49Vmzomo1S90FEgn7XRAFGBZ
dv1KpsT3lhjP91B+QKPnEJW13VmTIrBadfjS14wW5M66gCOaId1wylcM3ANZqnGU
nYOduIW01KjC/MDREYVLxh1qUYb6ea/Q1yukKh+jSHFurQFXQ/+XSdvVMF/ChaZd
```

Account recovery notification emails will be encrypted! [?]
Click here to enable an additional account recovery method

☑ Use this public key to encrypt notification emails that Facebook sends you? [?]

If you wish to share your public key, you can change who can see it in your profile's Contact and Basic Info about page.

You can download Facebook's public key here.

Encrypted email

☐ Encrypt contents and attachments for outgoing messages
☑ Add digital signature to outgoing messages
☑ Send clear text signed message when sending signed messages
☐ Request S/MIME receipt for all S/MIME signed messages

Default Setting: My S/MIME Settings (marina@thesslstore.com)

Digital IDs (Certificates)

Digital IDs or Certificates are documents that allow you to prove your identity in electronic transactions.

NETWORK COMMUNICATION ENCRYPTION

- SSH
- SSL/TLS
- OpenSSL

SECURE SHELL (SSH)

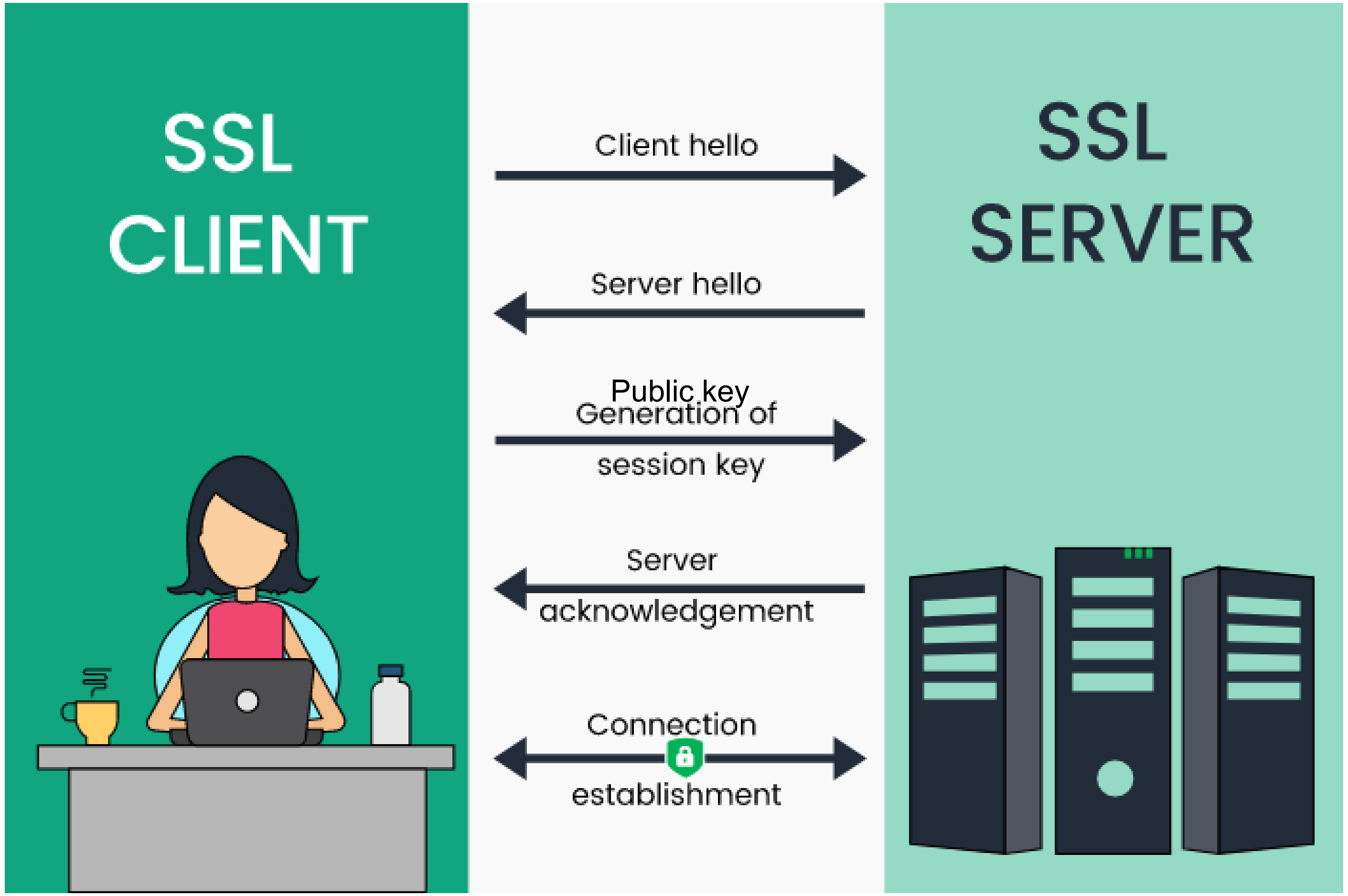
- Layer 7 protocol for secure remote logins and data transfer
- TCP 22
- Replacement for telnet and Berkeley remote-utilities
- Includes Secure Copy (SCP) and Secure FTP (SFTP) for data transfer
- Provides encrypted channel to be use for remote login, file transfers, and command execution
- Provides very strong user and host-to-host authentication
- Provides secure communication over the internet

SECURE SOCKETS LAYER (SSL)

- Layer 6 Protocol that establishes a secure connection between a client and server
- Used to secure confidentiality and integrity of data transmissions over the Internet
  - Particularly used by HTTPS to encrypt web traffic
  - Server proves its identity to the client
  - Server provides its public key to client
- Allows a client and server to:
  - Authenticate each other
  - Choose an encryption algorithm
  - Exchange public keys
  - Create a temporary session key
- Uses RSA asymmetric encryption
- Last version was SSL 3.0
- Has been replaced by TLS
- No longer considered secure
- Most modern browsers no longer support SSL

https:// www.website.com
http:// www.website.com

SSL EXAMPLE


SSL
CLIENT
SSL
SERVER
Client hello
Server hello
Public key
Generation of
session key
Server
acknowledgement
Connection
establishment

TRANSPORT LAYER SECURITY (TLS)

- The successor to SSL
- Fixes SSL security vulnerabilities
- Uses stronger encryption algorithms
- Can work over different ports
- More standardized
  - Can support emerging encryption algorithms
- Currently at version 1.3

OPENSSL

- A general purpose cryptography library
- Open-source implementation of the SSL and TLS protocols
  - Performs encryption/decryption
- Includes tools for generating:
  - Generating RSA private keys
  - Certificate Signing Requests (CSRs)
  - Checksums
- Can manage certificates
- Widely used by Internet servers and the majority of HTTPS websites

VPN ENCRYPTION

- IPSEC
- L2TP
- PPTP
- SSL

IPSEC

- AKA IP Security
- The strongest of the VPN protocols
  - Most widely used
  - Works at Layer 3 (IP only)
- Encrypts and authenticates data sent over a network
- Provides:
  - Origin authenticity through source authentication
  - Data integrity through hash functions
  - Confidentiality through encryption

IPSEC

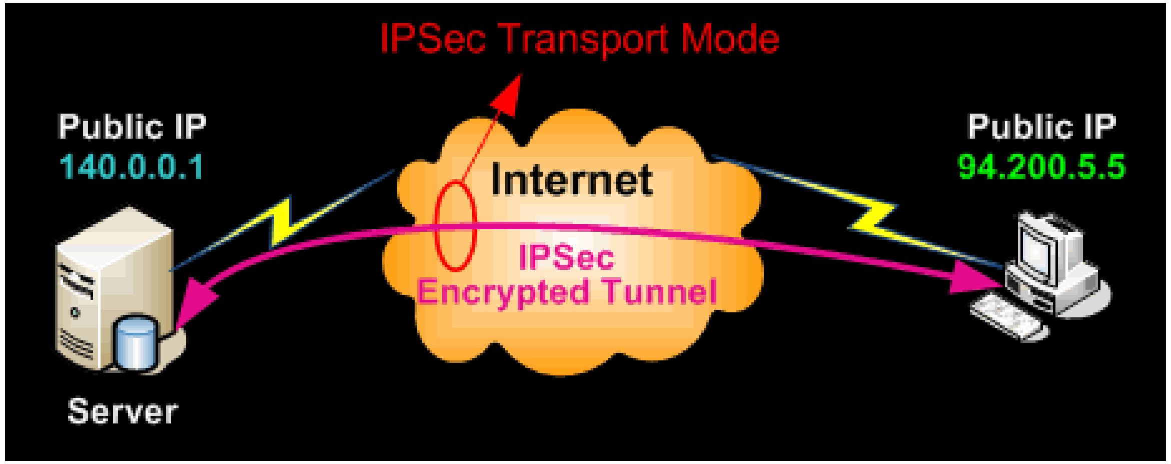
- Has two Layer 3 protocols:
  - Authentication Header (AH)
    - Digitally signs IP header to guarantee packet integrity
    - No payload encryption
    - MD5+HMAC, SHA+HMAC
    - Protocol ID 51
  - Encapsulating Security Payload (ESP)
    - Encrypts the payload using DES, 3DES, or AES
    - Also adds digitally signed UDP header to the payload to guarantee payload integrity
    - Protocol ID 50
  - You can use either or both protocols
- Includes a key exchange protocol:
  - ISAKMP
    - Used to secure the IPSEC key exchange process
    - UDP 500

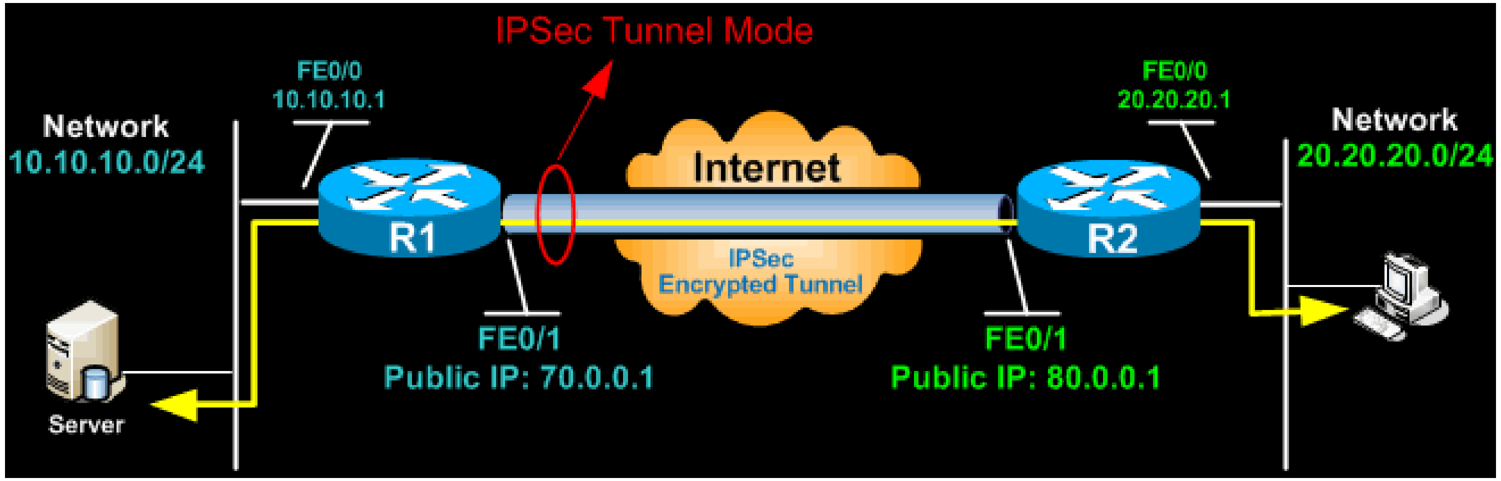
HMAC includes the private key in the message digest to prove identity

IPSEC MODES

- Transport mode
  - End-to-end encryption
  - VPN created between hosts
  - Good for:
    - Protecting clear text protocols
    - Client-server connections across the Internet
    - Server-server connections in the LAN, DMZ, or between the DMZ and LAN
- Tunnel mode
  - Gateway-gateway encryption
    - Routers / Firewalls
  - The entire original IP packet is protected by IPSec
  - IPSec wraps the original packet, encrypts it, adds a new IP header and sends it to the other side of the VPN tunnel (IPSec peer)
  - Hosts have no knowledge that their traffic is being sent through the tunnel
  - Good for connecting sites across the Internet

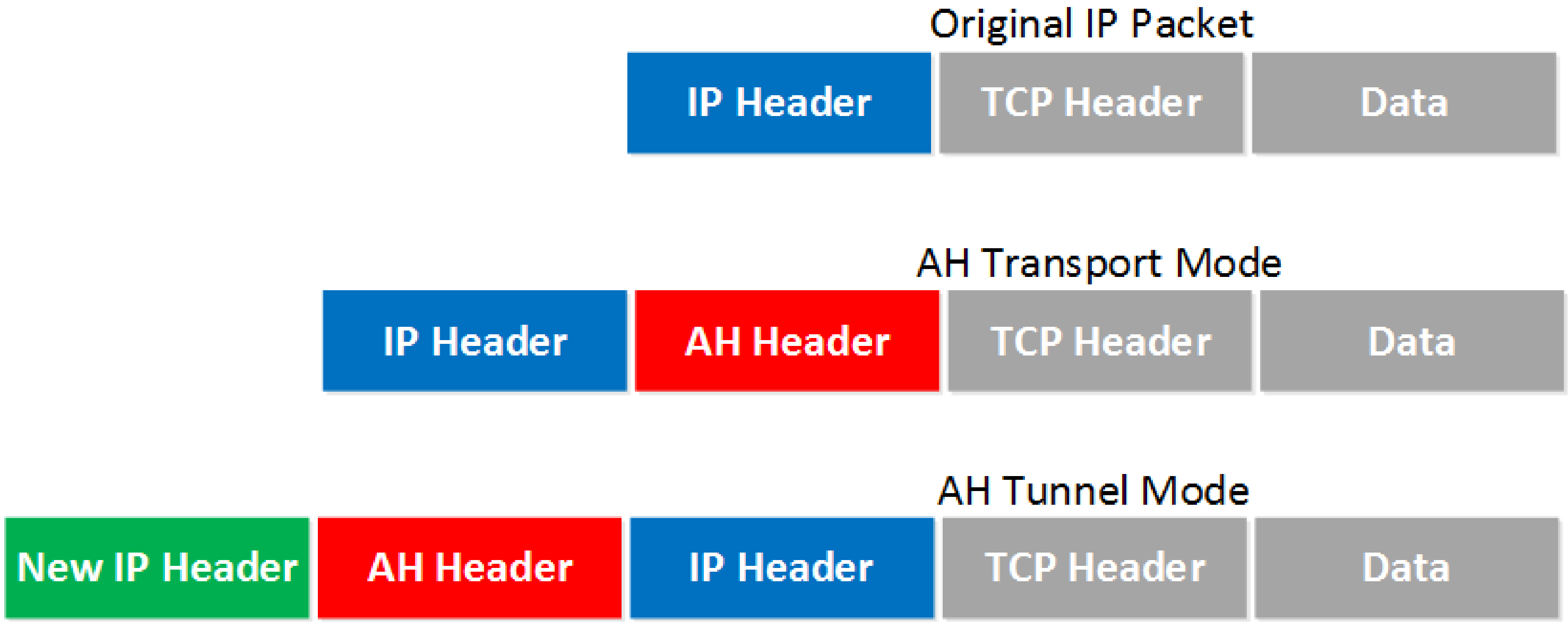
IPSEC MODE EXAMPLES





IPSEC AH TRANSPORT AND TUNNEL MODES

AH digital signature only -- no encryption



Original IP Packet

IP Header | TCP Header | Data

AH Transport Mode

IP Header | AH Header | TCP Header | Data

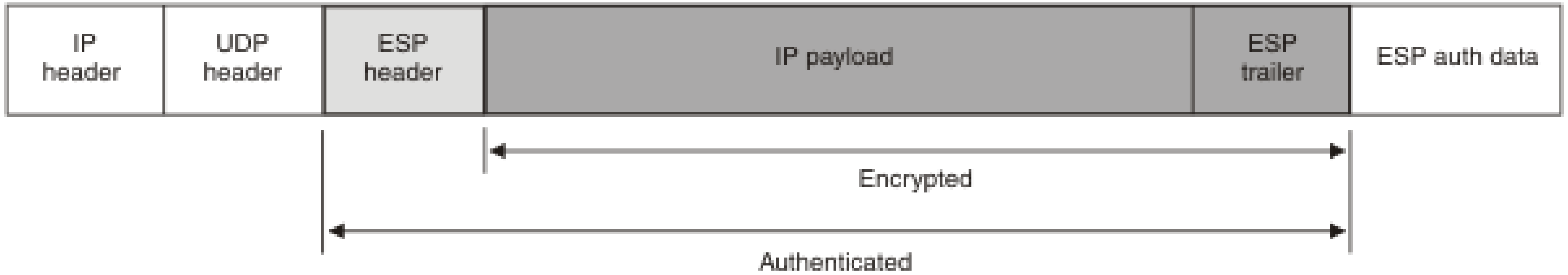
AH Tunnel Mode

New IP Header | AH Header | IP Header | TCP Header | Data

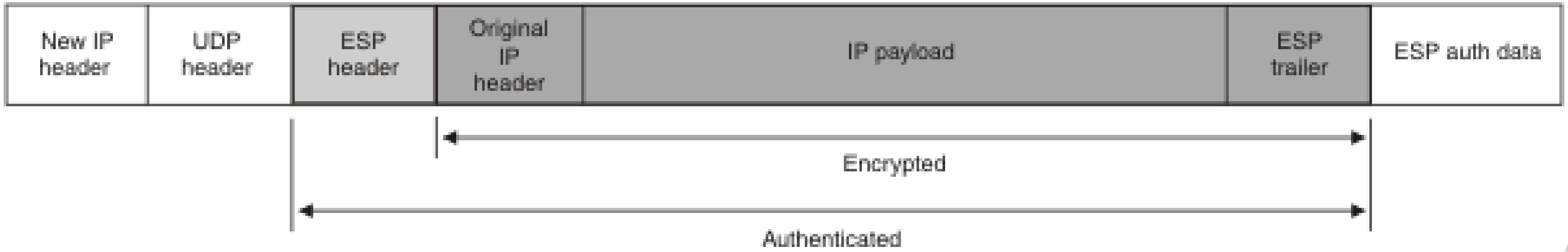
IPSEC ESP TRANSPORT AND TUNNEL MODES

Encryption and digital signature

Transport mode



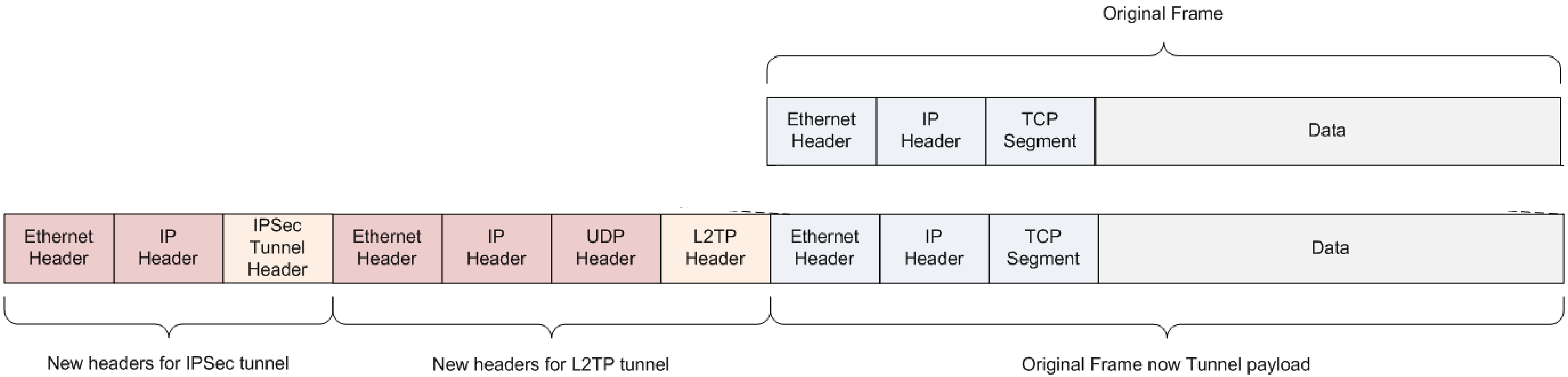
Tunnel mode



L2TP

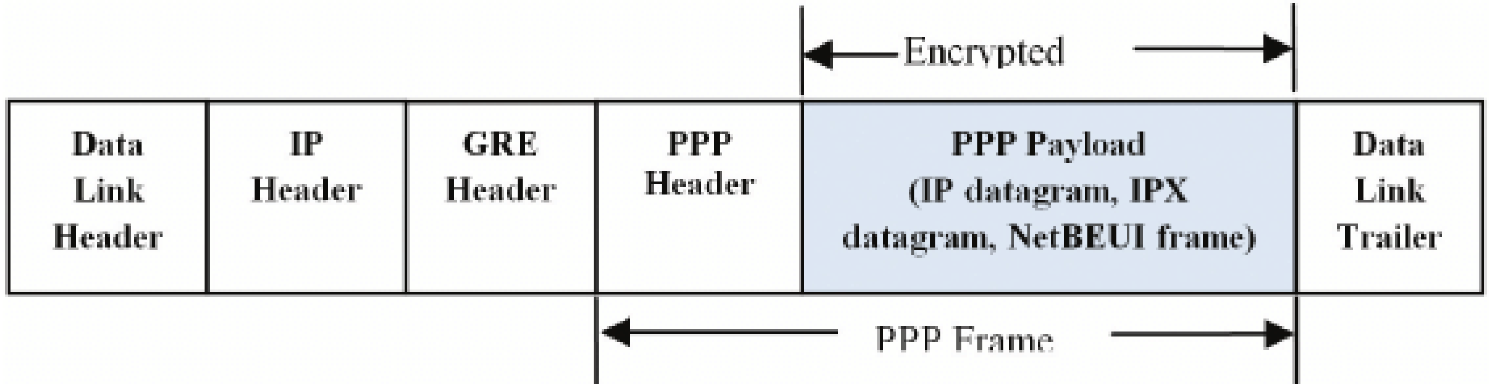
- Layer 2 Tunneling Protocol
  - TCP 1701
- Encapsulates but does not encrypt
- Can carry any payload: IP, IPX, NetBEUI
- Depends on IPSEC ESP for IP encryption
  - IPSEC over L2TP
  - UDP 500 (IKE)
- Can encapsulate but not encrypt other protocols

L2TP EXAMPLE



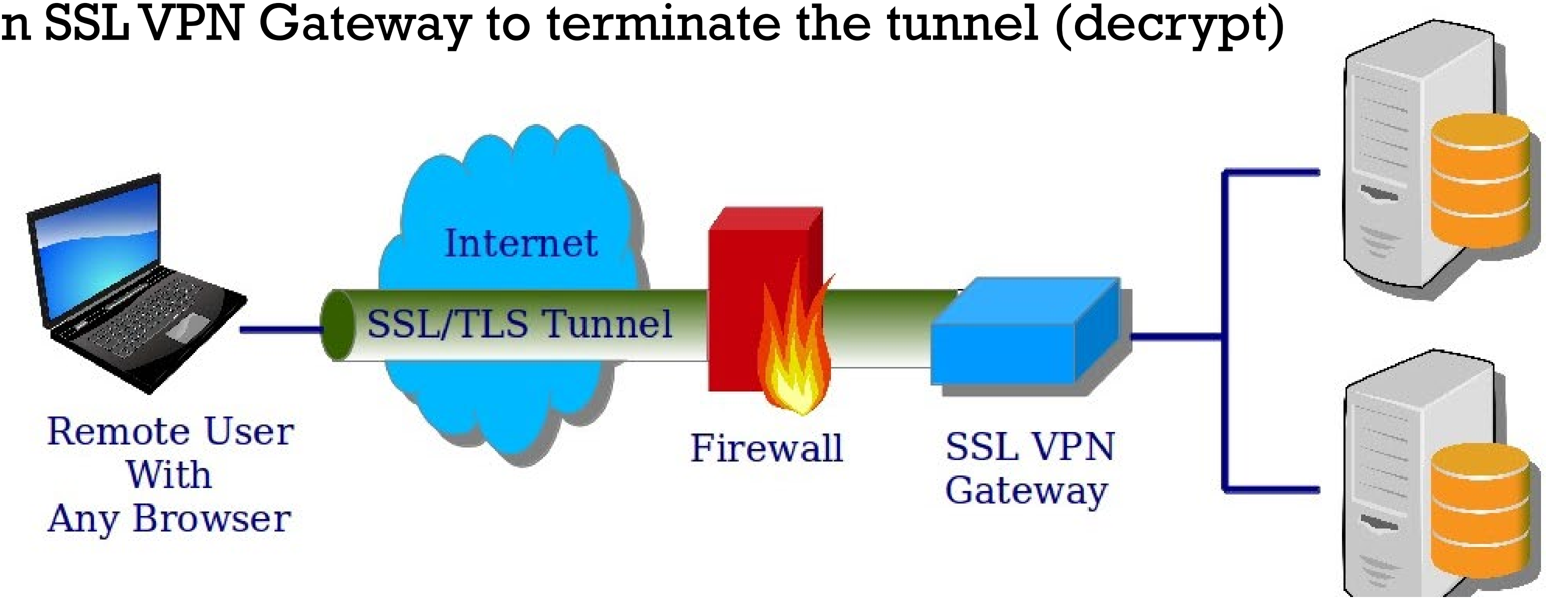
PPTP

- Point-to-Point Tunneling Protocol
- TCP port 1723
- Protocol ID 47 (GRE)
- Combination of Generic Routing Encapsulation (GRE) and PPP
- Can carry various payloads (IP, IPX, NetBEUI)
- Weak encryption
- No digital signatures
- Very easy to implement



SSL VPN

- Not a traditional VPN
  - No tunneling/encapsulation
- Uses SSL/TLS to encrypt the payload only
- Firewall friendly
- Requires an SSL VPN Gateway to terminate the tunnel (decrypt)

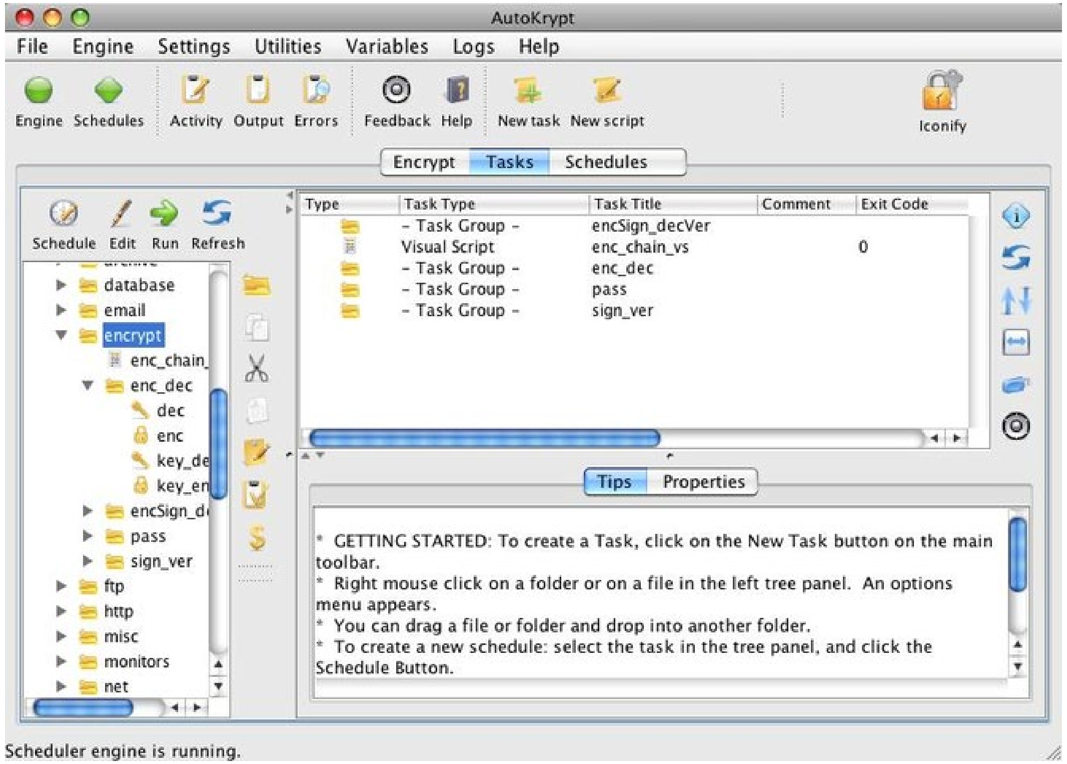


20.9 CRYPTOGRAPHY TOOLS

- Encryption Tools
- Tools for Mobile
- PGP
- Hashing Tools

CRYPTOGRAPHY TOOLS

- AutoKrypt
- Cryptainer LE Free Encryption Software
- Steganos LockNote
- AxCrypt
- CryptoForge
- Ncrypt XL
- ccrypt
- WinAES
- EncryptOnClick
- GNU Privacy Guard (GPG)



Type	Task Type	Task Title	Comment	Exit Code
	- Task Group -	encSign_decVer		
	Visual Script	enc_chain_vs		0
	- Task Group -	enc_dec		
	- Task Group -	pass		
	- Task Group -	sign_ver		

* GETTING STARTED: To create a Task, click on the New Task button on the main toolbar.
* Right mouse click on a folder or on a file in the left tree panel. An options menu appears.
* You can drag a file or folder and drop into another folder.
* To create a new schedule: select the task in the tree panel, and click the Schedule Button.

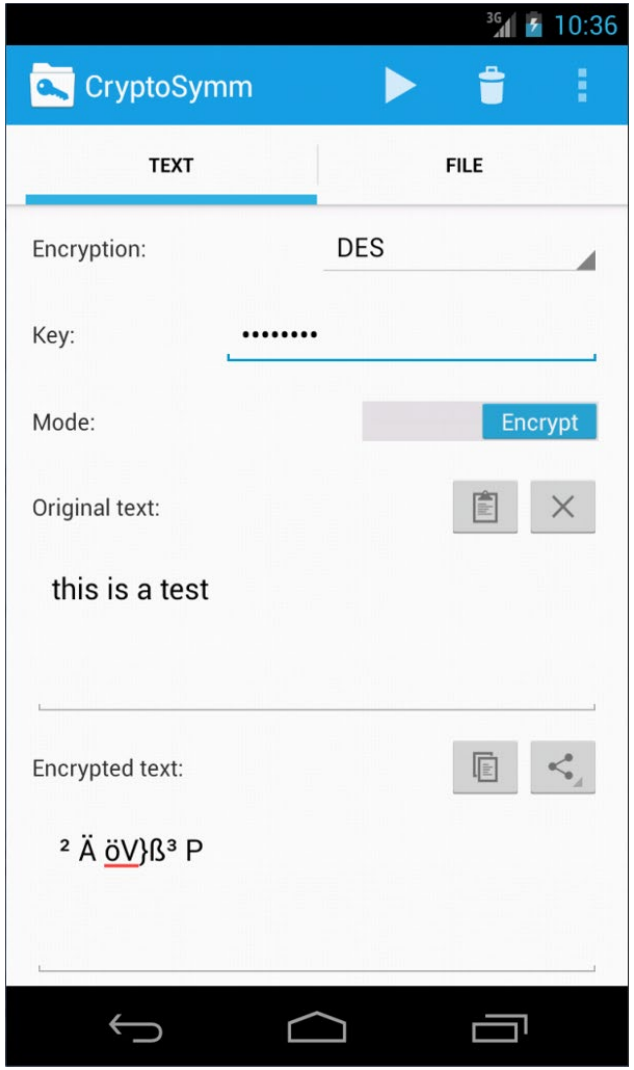
Scheduler engine is running.
]

CRYPTOGRAPHY TOOLKIT

- A command line tool to use various OpenSSL cryptography functions
- Uses SSL v2/v3 and TLS v1
- Key features:
  - Key rotation and versioning
  - Safe default algorithms, key lengths, and modes
  - Automated generation of ciphertext signatures and initialization vectors
  - Python, Java, and C++ implementations
  - Java international support

CRYPTOGRAPHY TOOLS FOR MOBILE

- Secret Space Encryptor
- CryptoSymm
- Cipher Sender



Encryption: DES

Key: •••••••••

Mode: Encrypt

Original text:

this is a test

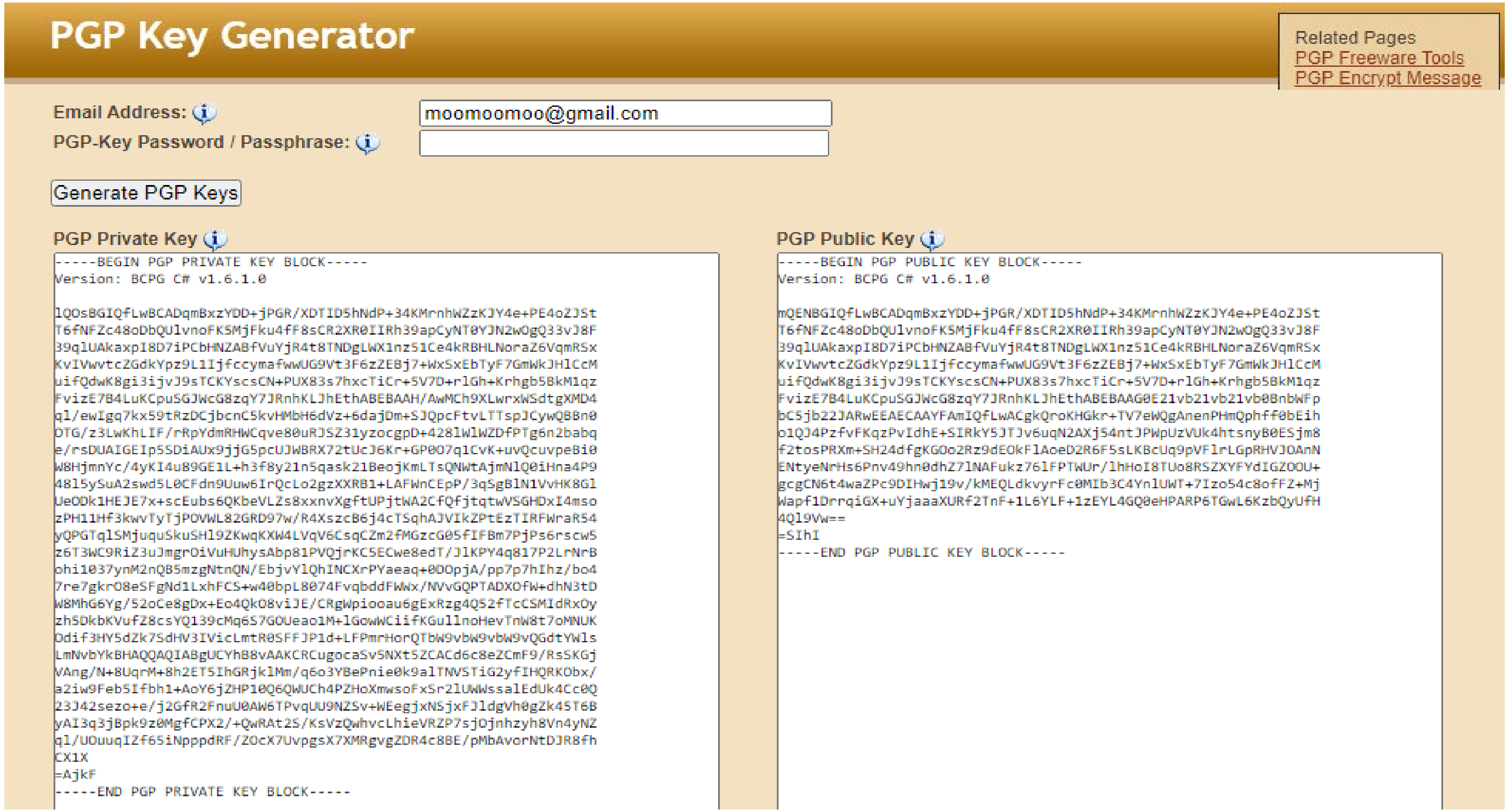
Encrypted text:

² Ä öV}ß³ P

PRETTY GOOD PRIVACY (PGP)

- System for creating asymmetric key pairs and trading public keys
- Provides authentication and cryptographic privacy
- Used for digital signing, data compression, and to encrypt/decrypt emails, messages, files, and directories
- You can download someone's public key and put it on your key ring
- Was sold to Symantec in 2010
- Open source replacement is GPG
- There are various online or downloadable PGP/GPG apps you can use

PGP KEY GENERATION EXAMPLE



PGP Key Generator

Related Pages
PGP Freeware Tools
PGP Encrypt Message

Email Address: moomoomoo@gmail.com

PGP-Key Password / Passphrase:

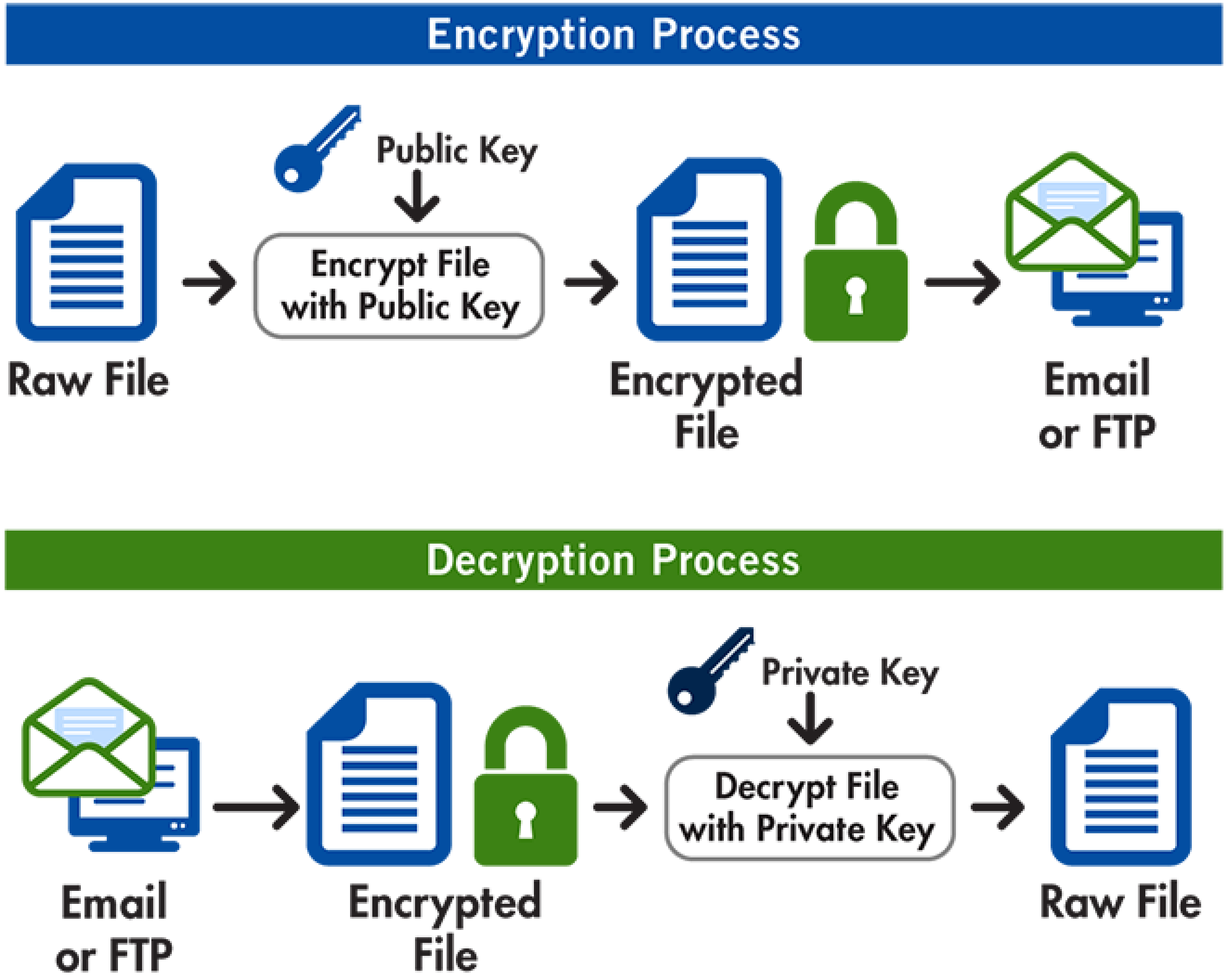
PGP Private Key

```
-----BEGIN PGP PRIVATE KEY BLOCK-----
Version: BCPG C# v1.6.1.0

lQOsBGIQfLwBCADqmBxzYDD+jPGR/XDTID5hNdP+34KMrnhWZzKJY4e+PE4oZJSt
T6fNFZc48oDbQUlvnoFK5MjFku4fF8sCR2XR0IIRh39apCyNT0YJN2wOgQ33vJ8F
39qlUAkaxpI8D7iPCbHNZABfVuYjR4t8TNDgLWX1nz51Ce4kRBHLNoraZ6VqmRSx
KvIVwvtcZGdkYpz9L1IjfccymafwwUG9Vt3F6zZEBj7+WxSxEbTyF7GmWkJHlCcM
uifQdwK8gi3ijvJ9sTCKYscsCN+PUX83s7hxcTiCr+5V7D+rlGh+Krhgb5BkM1qz
FvizE7B4LuKCpuSGJWcG8zqY7JRnhKLJhEthABEBAAH/AwMCh9XLwrxWSdtgXMD4
ql/ewIgq7kx59tRzDCjbcnC5kvHMbH6dVz+6dajDm+SJQpcFtvLTTspJCywQBBn0
OTG/z3LwKhLIF/rRpYdmRHWCqve80uRJSZ31yzocgpD+428lWlWZDfPTg6n2babq
e/rsDUAIGEIp5SDiAUx9jjG5pcUJWBRX72tUcJ6Kr+GP0O7q1CvK+uvQcuvpeBi0
W8HjmnYc/4yKI4u89GE1L+h3f8y21n5qask21BeojKmLTsQNWtAjmNlQ0iHna4P9
48l5ySuA2swd5L0CFdn9Uuw6IrQcLo2gzXXRB1+LAFWnCEpP/3qSgBlN1VvHK8Gl
UeODk1HEJE7x+scEubs6QKbeVLZs8xxnvXgftUPjtWA2CfQfjtqtwVSGHDxI4mso
zPH11Hf3kwvTyTjPOVWL82GRD97w/R4XszcB6j4cTSqhAJVIkZPtEzTIRFWraR54
yQPGTqlSMjuquSkuSHl9ZKwqKXW4LVqV6CsqCZm2fMGzcG05fIFBm7PjPs6rscw5
z6T3WC9RiZ3uJmgrOiVuHUhysAbp81PVQjrKC5ECwe8edT/JlKPY4q817P2LrNrB
ohi1037ynM2nQB5mzgNtnQN/EbjvYlQhINCXrPYaeaq+0DOpjA/pp7p7hIhz/bo4
7re7gkrO8eSFgNd1LxhFCS+w40bpL8074FvqbddFWWx/NVvGQPTADXOfW+dhN3tD
W8MhG6Yg/52oCe8gDx+Eo4QkO8viJE/CRgWpiooau6gExRzg4Q52fTcCSMIdRxOy
zh5DkbKVufZ8csYQ139cMq6S7GOUeao1M+lGowWCiifKGullnoHevTnW8t7oMNUK
Odif3HY5dZk7SdHV3IVicLmtR0SFFJP1d+LFPmrHorQTbW9vbW9vbW9vQGdtYWls
LmNvbYkBHAQQAQIABgUCYhB8vAAKCRCugocaSv5NXt5ZCACd6c8eZCmF9/RsSKGj
VAng/N+8UqrM+8h2ET5IhGRjklMm/q6o3YBePnie0k9alTNVSTiG2yfIHQRKObx/
a2iw9Feb5Ifbh1+AoY6jZHP10Q6QWUCh4PZHoXmwsoFxSr2lUWWssalEdUk4Cc0Q
23J42sezo+e/j2GfR2FnuU0AW6TPvqUU9NZSv+WEegjxNSjxFJldgVh0gZk45T6B
yAI3q3jBpk9z0MgfCPX2/+QwRAt2S/KsVzQwhvcLhieVRZP7sjOjnhzyh8Vn4yNZ
ql/UOuuqIZf65iNpppdRF/ZOcX7UvpgsX7XMRgvgZDR4c8BE/pMbAvorNtDJR8fh
CX1X
=AjkF
-----END PGP PRIVATE KEY BLOCK-----
```

PGP Public Key

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: BCPG C# v1.6.1.0

mQENBGIQfLwBCADqmBxzYDD+jPGR/XDTID5hNdP+34KMrnhWZzKJY4e+PE4oZJSt
T6fNFZc48oDbQUlvnoFK5MjFku4fF8sCR2XR0IIRh39apCyNT0YJN2wOgQ33vJ8F
39qlUAkaxpI8D7iPCbHNZABfVuYjR4t8TNDgLWX1nz51Ce4kRBHLNoraZ6VqmRSx
KvIVwvtcZGdkYpz9L1IjfccymafwwUG9Vt3F6zZEBj7+WxSxEbTyF7GmWkJHlCcM
uifQdwK8gi3ijvJ9sTCKYscsCN+PUX83s7hxcTiCr+5V7D+rlGh+Krhgb5BkM1qz
FvizE7B4LuKCpuSGJWcG8zqY7JRnhKLJhEthABEBAAG0E21vb21vb21vb0BnbWFp
bC5jb22JARwEEAECAAYFAmIQfLwACgkQroKHGkr+TV7eWQgAnenPHmQphff0bEih
o1QJ4PzfvFKqzPvIdhE+SIRkY5JTJv6uqN2AXj54ntJPWpUzVUk4htsnyB0ESjm8
f2tosPRXm+SH24dfgKGOo2Rz9dEOkFlAoeD2R6F5sLKBcUq9pVFlrLGpRHVJOAnN
ENtyeNrHs6Pnv49hn0dhZ7lNAFukz76lFPTWUr/lhHoI8TUo8RSZXYFYdIGZOOU+
gcgCN6t4waZPc9DIHwj19v/kMEQLdkvyrFc0MIb3C4YnlUWT+7Izo54c8offZ+Mj
Wapf1DrrqiGX+uYjaaaXURf2TnF+1L6YLF+1zEYL4GQ0eHPARP6TGwL6KzbQyUfH
4Ql9Vw==
=SIhI
-----END PGP PUBLIC KEY BLOCK-----
```

PGP/GPG EXAMPLE

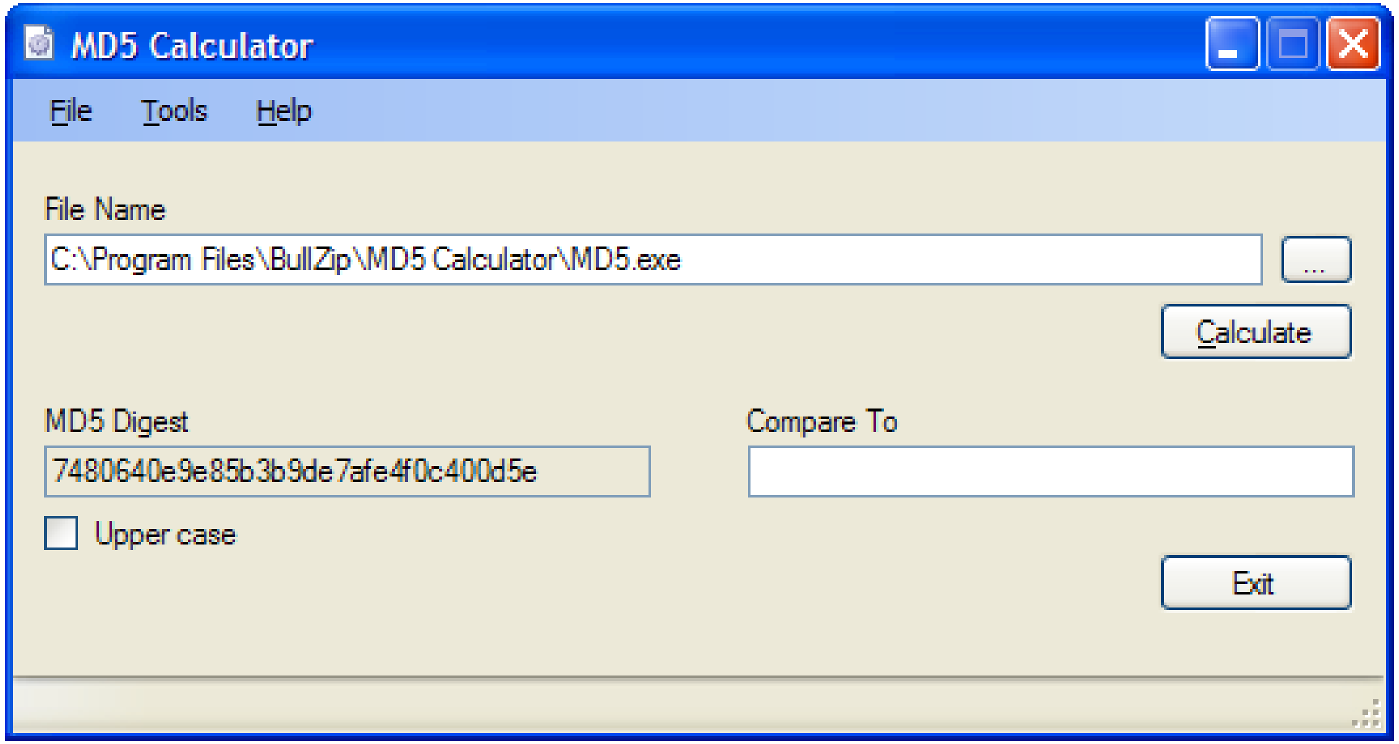


HASHING TOOL EXAMPLES

- Microsoft Hash Tool
- md5sum
- sha256sum
- CRC Calculator
- SHA Calculator
- MD2 Calculator
- MD4 Calculator
- MD5 Calculator
- MD6 Hash Generator
- Adler-32 Calculator
- RIPEMD Calculator
- Whirlpool Calculator
- NTLM Calculator
- CrackStation
- HashCalc
- MD5 Calculator
- HashMyFiles
- MD5 Hash Calculator
- Hash Droid
- Hash Calculator

There are also any number of online sites that will perform hashing for you

MD5 HASH CALCULATOR EXAMPLE



File Name

C:\Program Files\BullZip\MD5 Calculator\MD5.exe

MD5 Digest

7480640e9e85b3b9de7afe4f0c400d5e

□ Upper case

Compare To

MD5SUM AND SHA1SUM

- Command line hashing calculators for Linux
- Windows version can be downloaded

MD5 output:
32 hex numbers
128 bit

```
md5sum somefile.txt
c6779ec2960296ed9a04f08d67f64422  somefile.txt
```

SHA1 output:
40 hex numbers
160 bit

```
sha1sum somefile.txt
da39a3ee5e6b4b0d3255bfef95601890afd80709 somefile.txt
```

```
sha1sum somefile.txt > somefile.txt.sha1
cat somefile.txt.sha1
da39a3ee5e6b4b0d3255bfef95601890afd80709 somefile.txt
```

20.10 CRYPTOGRAPHY ATTACKS

- Code Breaking Methodologies
- Computational Resources
- Hash Collisions
- Crypto Attacks
- Cryptanalysis Countermeasures

CODE BREAKING METHODOLOGIES

- Trickery and Deceit
  - Social Engineering
- Brute Force
  - Try combinations until you crack it
- Frequency Analysis
  - Look for repeat patterns
- Meet-in-the-Middle
  - Examine encrypted and unencrypted text to figure out the key
- Side Channel
  - Examine emissions from electronic circuitry to determine corresponding algorithm activity

COMPUTATIONAL RESOURCES FOR CRYPTANALYSIS

- Attacks can be characterized by the resources they require
- Time:
  - The number of computation steps (e.g., test encryptions) that must be performed
- Memory:
  - The amount of storage required to perform the attack
- Data:
  - The quantity and type of plaintexts and ciphertexts required for an approach

HIGH-PERFORMANCE COMPUTING (HPC)

- One of the most essential tools in cryptanalysis
- Leverages GPU-powered parallel processing across multiple compute nodes
  - A Graphical Processing Unit (GPU) is a built-in CPU on a video card
  - The GPU offloads computationally-intensive tasks such as video rendering from the CPU
  - It can also be used in cryptanalysis
- You can also use the cloud to provide extensive compute resources
- You can even distribute your cracking across a bot army!

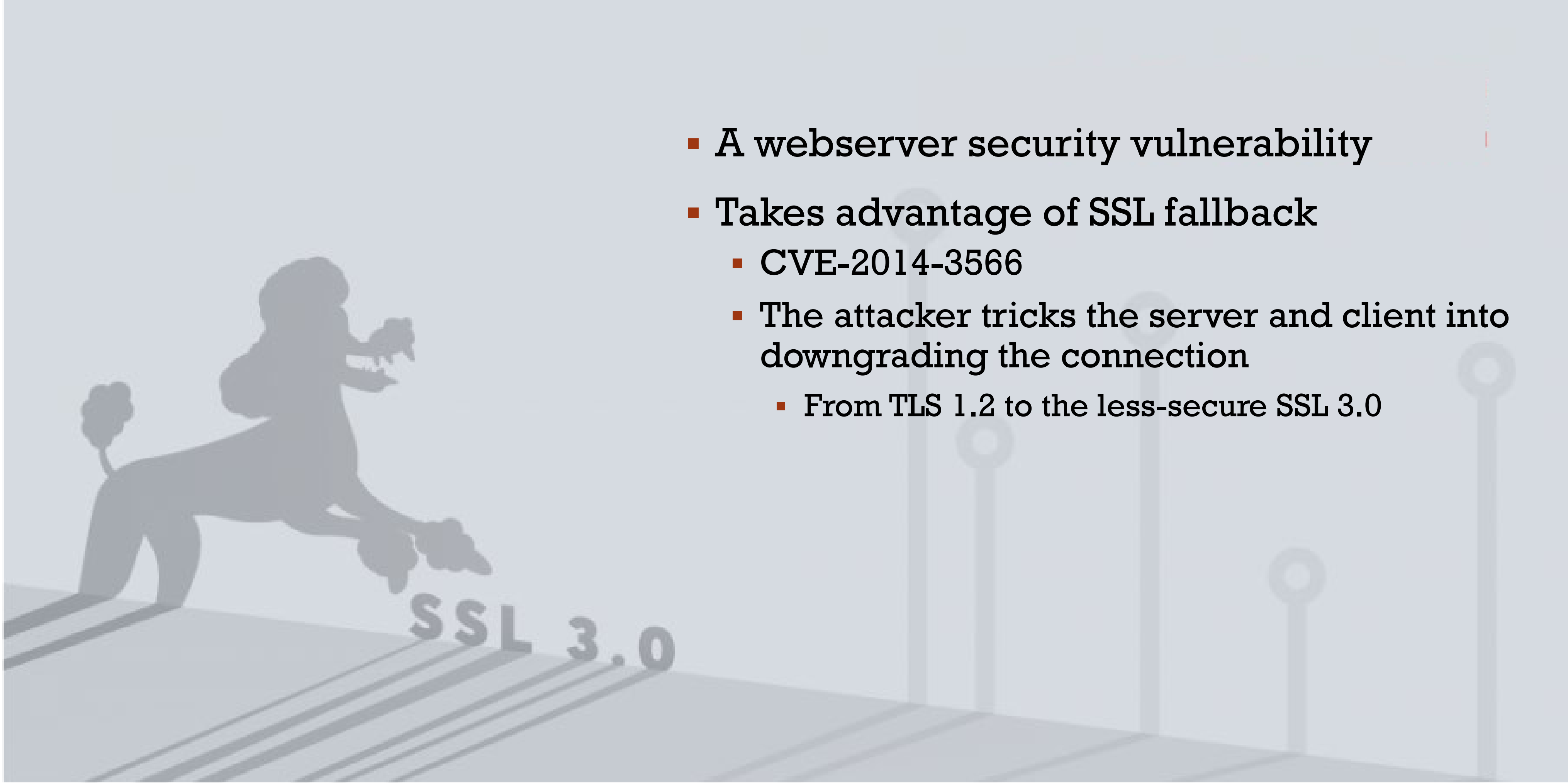
HASH COLLISION ATTACK

- An attempt to find two input strings of a hash function that produce the same hash result
- Because hash functions have infinite input length and a predefined output length
  - There is inevitably going to be the possibility of two different inputs that produce the same output hash
- A strong hashing algorithm is resistant to collisions

HEARTBLEED

- A severe memory handling bug
- Affects OpenSSL versions 1.0.1 through 1.0.1f
- Exists in the implementation of the TLS Heartbeat Extension
  - Heartbeats are used to keep the TLS session alive
- Could be used to reveal up to 64 KB of the application's memory with every heartbeat
- By reading the memory of the web server, attackers could access sensitive data, including the server's private key
- CVE-2014-0160

POODLE



- A webserver security vulnerability
- Takes advantage of SSL fallback
  - CVE-2014-3566
  - The attacker tricks the server and client into downgrading the connection
    - From TLS 1.2 to the less-secure SSL 3.0

POODLE ATTACK STAGES

1. The attacker inserts themselves as man-in-the-middle between client and server
2. The attacker falsely drops connections, tricking the server into assuming that the client does not support TLS 1.2
3. As the client and the server communicate using SSL 3.0, the attacker can use the POODLE attack to decrypt selected parts of the communication and steal confidential information
   - To make sure that the POODLE attack succeeds, the attacker uses social engineering to trick the user into running a Java script in their browser

CRYPTANALYSIS TECHNIQUES

- **Ciphertext Only**
  - The cryptanalyst has access only to a collection of ciphertexts or code texts
- **Known plaintext attack**
  - The analyst may have access to some or all the plaintext of the ciphertext
  - The goal is to discover the key used to encrypt the message and decrypt the message
  - Once the key is identified, an attacker can decode all messages that had been encrypted by utilizing that key
- **Chosen plaintext attack**
  - The analyst either knows the encryption algorithm or has access to the device used to do the encryption
  - The analyst can encrypt the ‘chosen plaintext’ with the targeted algorithm to obtain data about the key
- **Adaptive Chosen Plaintext**
  - Like a chosen-plaintext attack, except the attacker can choose subsequent plaintexts based on information learned from previous encryptions

CRYPTANALYSIS TECHNIQUES (CONT'D)

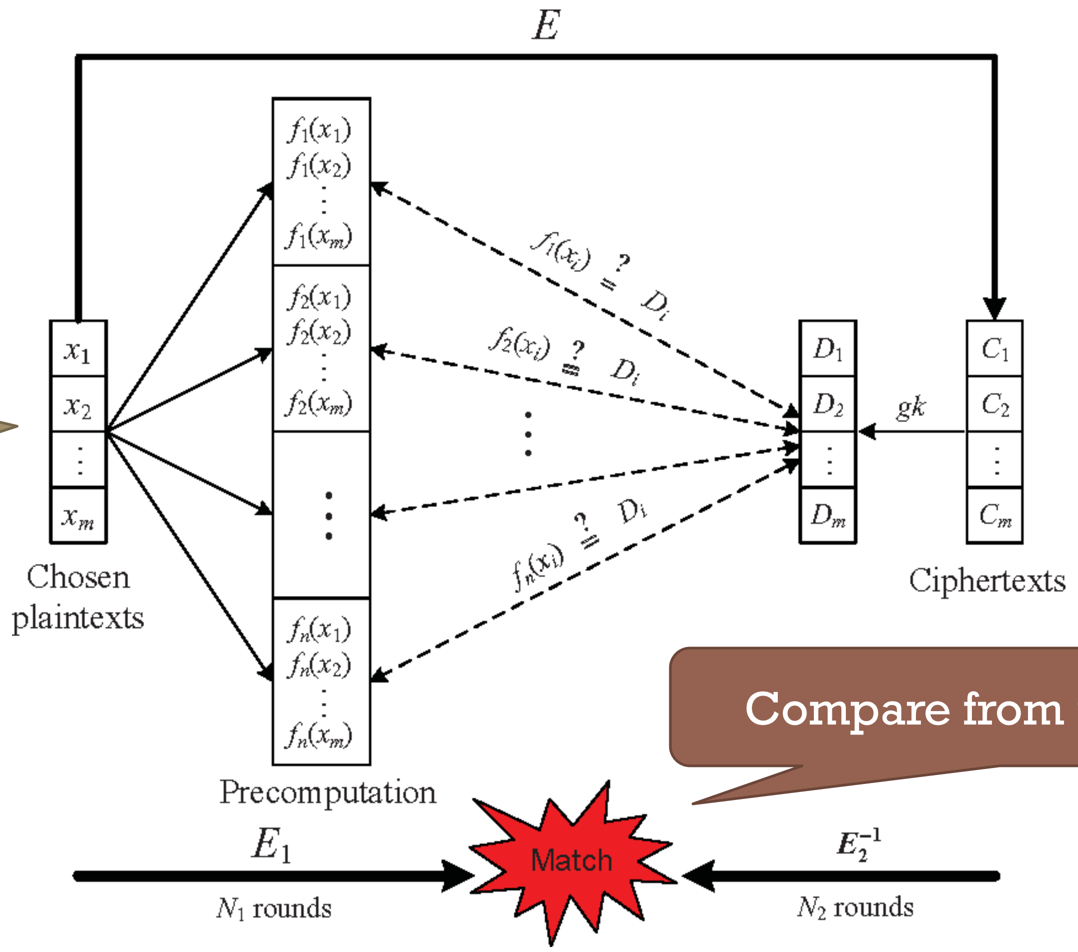
- **Related-key attack**
  - Like a chosen-plaintext attack
  - Except the attacker can obtain ciphertexts encrypted under two different keys
  - The keys are unknown, but the relationship between them is known
    - For example, two keys differ by one bit
- **Man-in-the-middle attack**
  - The attacker finds a way to insert themselves into the communication channel between two parties who wish to exchange public keys
  - The attacker then performs a key exchange with each party
    - The original parties believe they are exchanging keys with each other
    - The two parties end up utilizing keys that are familiar to the attacker
- **Integral cryptanalysis attack**
  - Uses sets of plaintexts
  - Part of the plaintext is kept constant
  - The rest of the plaintext is modified
  - This attack can be especially useful when applied to block ciphers that are based on substitution-permutation networks

MEET-IN-THE-MIDDLE ATTACK

- A type of known plaintext attack
- Uses two known assets:
  - a plaintext block
  - an associated ciphertext block
- The attacker uses both assets to decipher the key
- The attack involves working from either end of the encryption chain toward the middle
  - As opposed to trying brute-force permutations from one end of the encryption process to the other.
- Common attack against Data Encryption Standard (DES)
  - Can break ciphers that use two or more keys for multiple encryption using the same algorithm (2DES, 3DES)

MEET-IN-THE-MIDDLE ATTACK EXAMPLE

Compute and store mappings



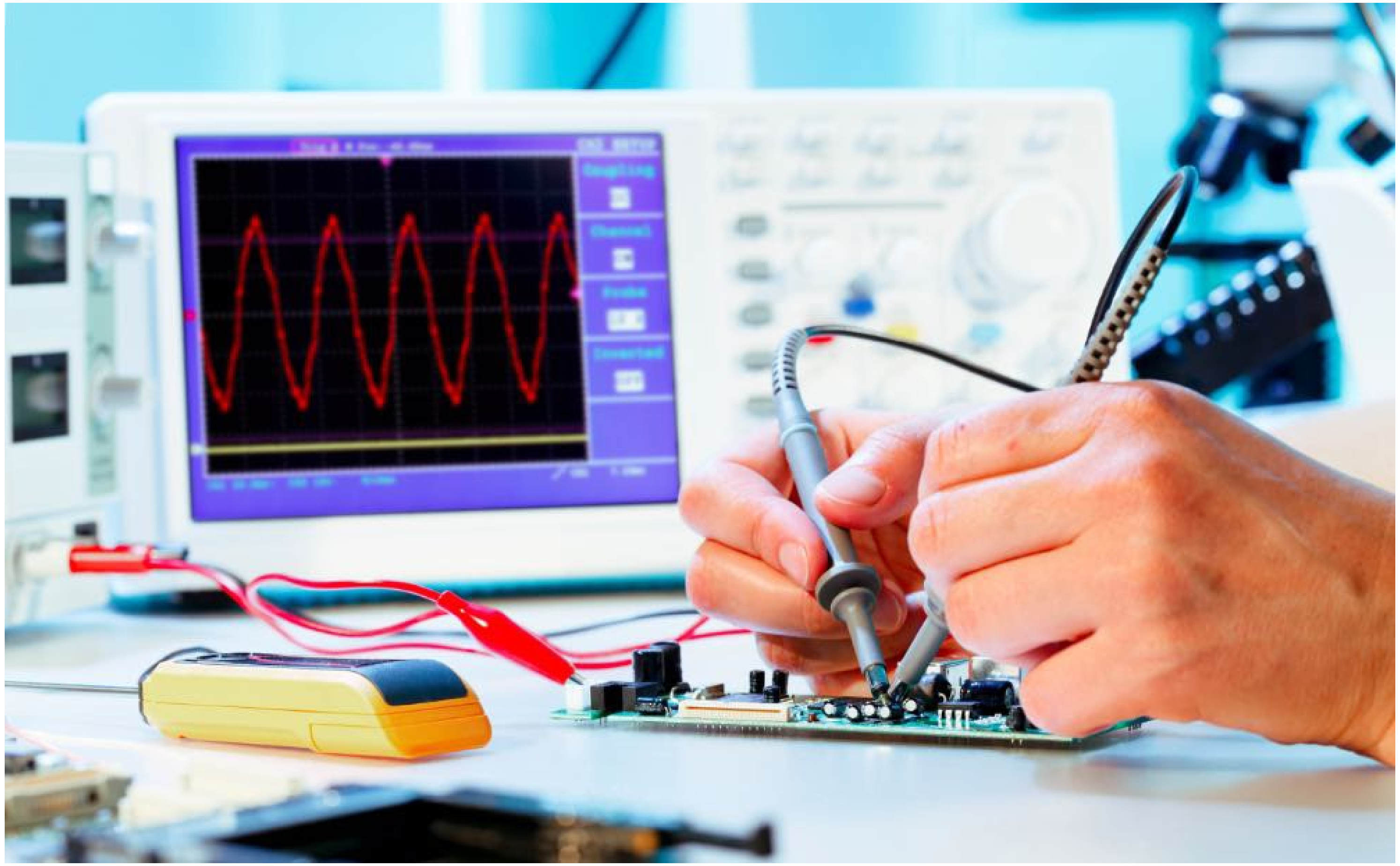
Compute decryptions

Compare from two directions

SIDE-CHANNEL ATTACK

- Electronic circuitry always “leaks” various forms of radiant energy as it processes signals and executes commands
- A side-channel attack takes advantage of observable external changes (side-channel properties) in the circuitry during processing:
  - Heat generated, power consumed, execution time
  - These changes happen at different times during algorithm execution
- If an attacker can run their own code on the encryption/decryption hardware
  - They can more quickly figure out what the different physical changes indicate

SIDE-CHANNEL ATTACK EXAMPLE



CRYPTANALYSIS TOOL EXAMPLES

- CrypTool
  - An open-source project that produces e-learning programs and a web portal for learning about cryptanalysis and cryptographic algorithms.
- Cryptol
  - Analyzes algorithms and implementations
  - Initially designed for the NSA
  - Is also widely used by private firms
- EverCrack
  - A GPL open-source software that mainly deals with monoalphabetic substitution and transposition ciphers
  - Its cryptanalysis engine supports multiple languages
- Ganzúa
  - An open-source cryptanalysis tool used for classical polyalphabetic and monoalphabetic ciphers
  - Lets users outline nearly complete arbitrary cipher and plain alphabets

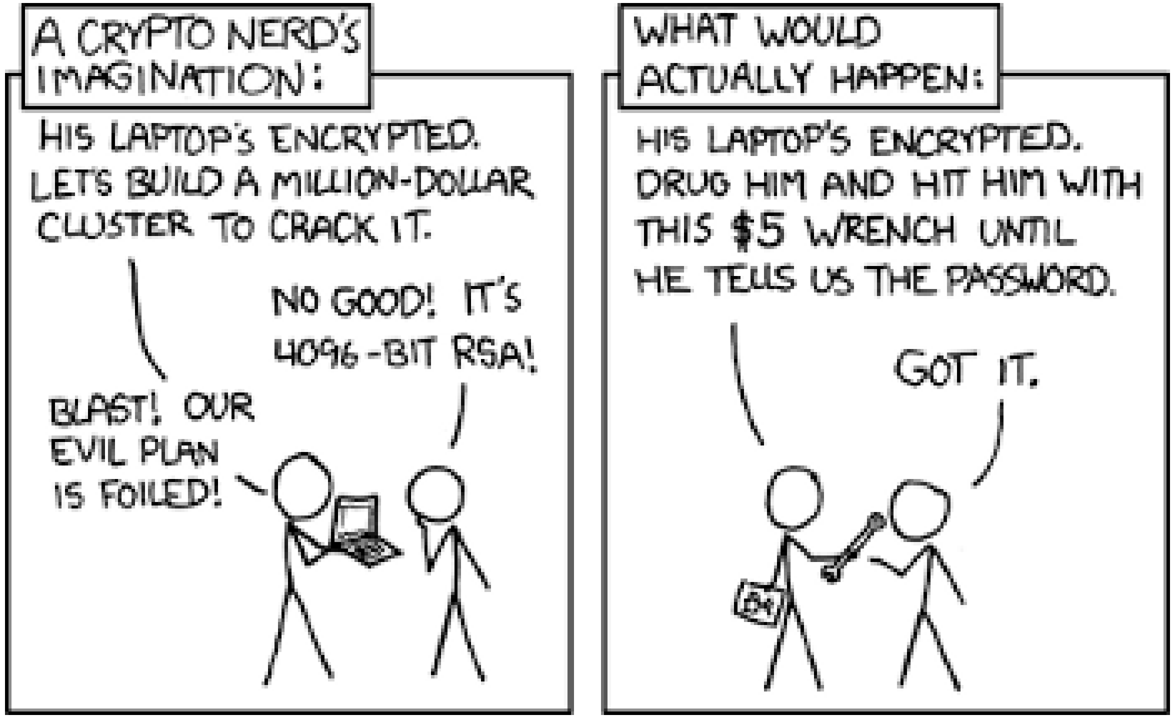
Cryptanalysis is the process of deciphering encrypted messages without being told the key

PASSWORD CRACKING TOOLS

- John-the-Ripper
  - Supports hundreds of hash and cipher types
  - Can use large word lists
- Hashcat
  - Performs dictionary and brute force password attacks
  - Utilizes both a computer's GPU as well as CPU for high performance
- Rainbow Tables
  - Specialized dictionary list
  - Pre-computed hashes
- There are various online password cracking services you can use
- You can also try social engineering to trick the user into divulging their password

RUBBER HOSE ATTACK

- Extraction of cryptographic secrets from a person by coercion or torture


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

CRYPTANALYSIS COUNTERMEASURES

There are a number of strategies that you can employ to protect your cryptosystem

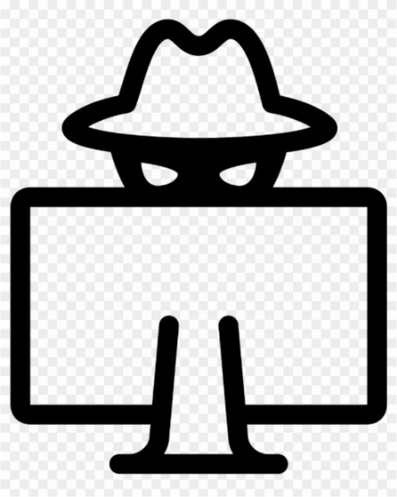
- Choose stronger cryptographic algorithms where practical
- Use longer keys or key stretching to counter a brute force attack
- Carefully protect private keys
  - Encrypt the keys and store locally
  - Do not store in the cloud
  - Never hard-code a cryptographic key in an application
- If the computer system has limited resources, consider using algorithms that provide comparable protection while using less compute power
  - E.g., Elliptic Curve Cryptography (ECC) over RSA
- Ensure application developers use well-vetted crypto frameworks
  - Do not attempt to "roll your own" encryption in application development
- Use bug bounties and public challenges to help vet your algorithm
  - Having thousands of security researchers enthusiastically trying to break your cryptosystem will reveal its weaknesses more quickly than any other method
  - A publicly known algorithm that no one has been able to crack is likely to be stronger than a secret algorithm that has been minimally tested
- Use compensating controls to reduce the risk of side-channel attacks
  - Example: use TEMPEST shielding prevent electrical emanations from being intercepted

20.11 CRYPTOGRAPHY REVIEW

- Review

CRYPTOGRAPHY REVIEW

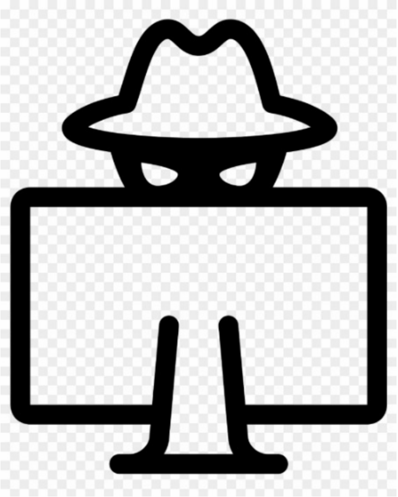
- Encryption happens at OSI Layer 6 (Presentation Layer)
- Data has three possible states:
  - at rest (stored on storage media), in transit (being transmitted across a network), in use (in RAM)
- Cryptography is the conversion of data into jumbled code to keep it safe
- Cryptography components are:
  - Plain text + key + cipher (algorithm) = ciphertext



- "Plain text" is a generic term often used to describe any unencrypted data
- A key is anything that can be reduced to a number
  - Also called a secret
  - The longer the key, the stronger the encryption
  - A key can be made longer by adding a salt or Initialization Vector to it
- A cipher is a mathematical formula that uses the key to encrypt the data
- Ciphertext is data that has been encrypted

CRYPTOGRAPHY REVIEW (CONT'D)

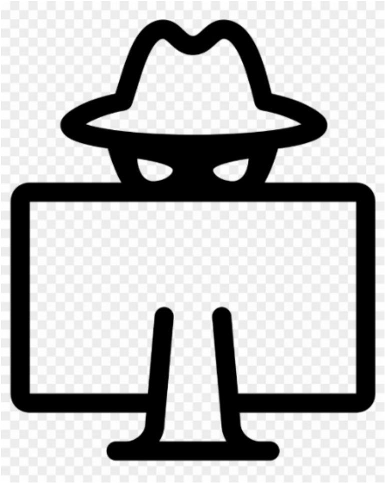
- Symmetric encryption uses same key for encryption and decryption
  - It must be known to both parties and agreed upon in advance
  - If it becomes compromised, everything encrypted with it is also considered to be compromised
- Symmetric algorithms include DES, 3DES, AES
  - DES and 3DES are no longer considered secure
  - AES is the current standard
- Symmetric encryption has relatively good performance, and is used to encrypt large amounts of data



- A block cipher divides the data into chunks
  - Encrypts each chunk one at a time
  - It is well suited for encrypting large amounts of data
- A stream cipher uses a key that is being continuously, randomly generated
  - It XOR's the key bits against the data bits, producing a stream of encrypted bits
  - It is well suited to encrypt realtime data such as realtime voice/video or network (Wi-Fi) transmissions

CRYPTOGRAPHY REVIEW (CONT'D)

- Asymmetric encryption uses a public/private key pair to encrypt/decrypt
  - The two keys are mathematically related
  - You freely give away the public key
  - You carefully guard the private key from unauthorized disclosure
- In asymmetric encryption, you encrypt with one key (typically the public key)
  - Then decrypt with the other (typically the private key)
- In order to send someone data that only they can read, you must use THEIR public key to encrypt it
  - They will then use their private key to decrypt the data
- Diffie-Hellmann or RSA are two popular key exchange algorithms used to securely trade public keys across the network



- The most popular asymmetric algorithm in use today is RSA
  - It is based on large factors (prime numbers)
- ECC is another popular asymmetric algorithm
  - It is based on the algebraic structure of elliptic curves over finite fields
  - It provides the same level of protection as RSA while consuming considerably fewer resources
  - It is the preferred choice for small devices such as smart cards and mobile/wireless devices
- Because RSA encryption is computationally expensive, a client and server will trade public keys
  - They will then use those keys to jointly create a temporary symmetric session key
  - Even if the transmission is intercepted, without one of the private keys an attacker cannot decrypt the message

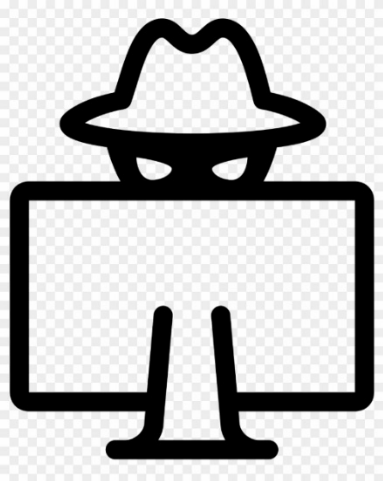
CRYPTOGRAPHY REVIEW (CONT'D)

- A certificate is a public key on a document
  - It is accompanied by a protected private key
- You can use your private key to digitally sign data
  - This proves authenticity
  - Others can verify the signature by using the public key from your certificate
  - You can be legally held liable if others use your private key to impersonate you

- You can generate your own public/private key pairs or certificates
- Public Key Infrastructure uses well-known certificate authorities (CA) to issue certificates to the general public
  - These certificates are trusted by everyone because operating systems ship with certificates from the well-known Root CAs
    - Thus the chain of authenticity can be proven all the way up to the issuing CA

CRYPTOGRAPHY REVIEW (CONT'D)

- Hashing creates a fixed-length output from a variable input
  - It proves data integrity
  - In general, hashing does not use a key in the hashing process
  - A hash is computationally infeasible to decrypt
  - User passwords are typically stored as hashes in an operating system file
- Hashing algorithms should be resistant to collisions
  - A collision is where two different inputs produce the same output
- Popular hashing algorithms include MD5, SHA1, SHA256, LM, NTLM



- HMAC is another hashing algorithm that adds the user's private key to the data before it is hashed
  - This proves both authenticity and integrity
- There are many practical uses for cryptography in data storage, network transmission, e-commerce, VPNs, email, etc.
- There are many ways to try to break encryption
- If you cannot break the encryption, try social engineering or coercion