Information Security

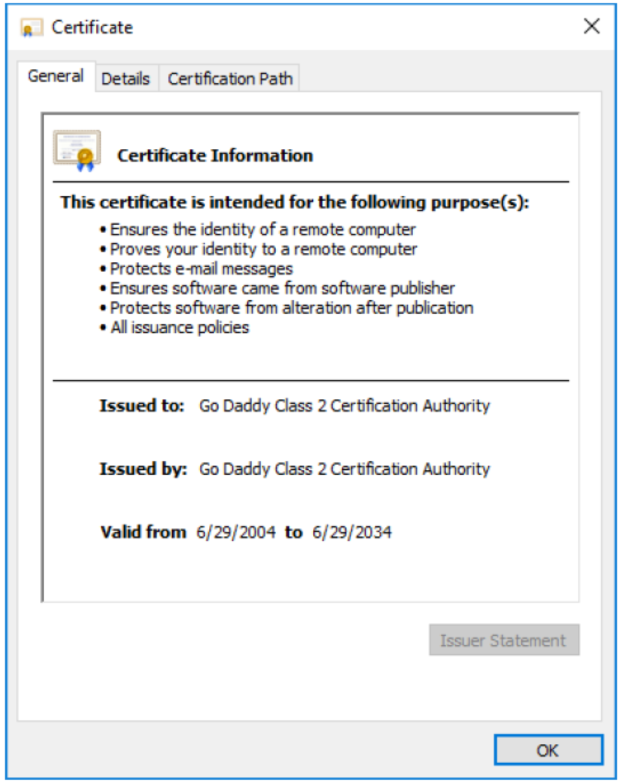
Lecture 4: Network Security Protocols

Mona Taghavi

LaSalle College
Montréal

What Is a Digital Certificate?

- A **digital certificate** is an electronic document used to identify an individual, a server, an organization, or some other entity and associate that entity with a **public key**.
- Digital certificates are used in **public key infrastructure (PKI)** encryption.
- We can think of a digital certificate as our “online” **digital credential** that verifies our identity.



Certificate Information

This certificate is intended for the following purpose(s):

- Ensures the identity of a remote computer
- Proves your identity to a remote computer
- Protects e-mail messages
- Ensures software came from software publisher
- Protects software from alteration after publication
- All issuance policies

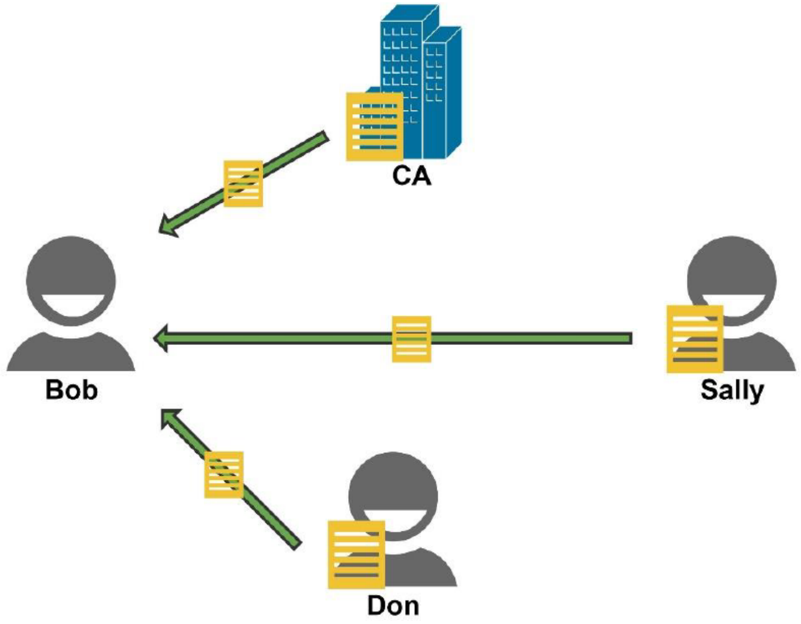
Issued to: Go Daddy Class 2 Certification Authority

Issued by: Go Daddy Class 2 Certification Authority

Valid from 6/29/2004 **to** 6/29/2034

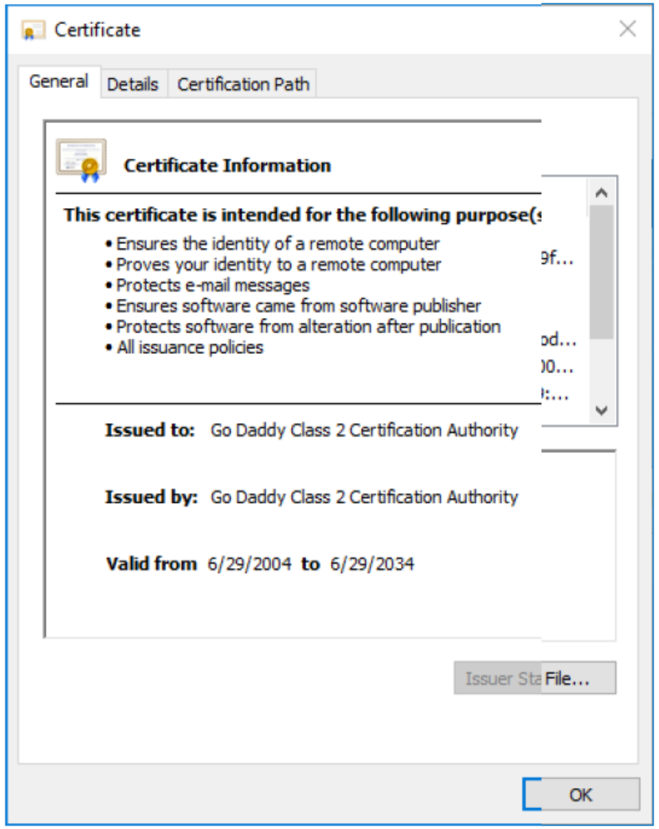
The Role of Certificate Authorities

- Digital certificates are issued by a **Certificate Authority (CA)**.
- **Certificate Authorities** are a trusted entity, typically an organization such as VeriSign, that verifies an entity's identity, issues, manages, and signs that entity's digital certificate.
- Just like we trusted SAAQ to issue driver's licenses, we trust CAs to issue digital certificates.



What's Included in a Digital Certificate?

- **Serial Number**: Used to uniquely identify the certificate.
- **Signature Algorithm**: The algorithm used to create the signature.
- **Issuer**: The entity that verified the information and issued the certificate.
- **Valid-From**: The date the certificate is first valid from.
- **Valid-To**: The expiration date.
- **Public Key**: The public key.
- Plus Additional Information.



This certificate is intended for the following purpose(s

- Ensures the identity of a remote computer
- Proves your identity to a remote computer
- Protects e-mail messages
- Ensures software came from software publisher
- Protects software from alteration after publication
- All issuance policies

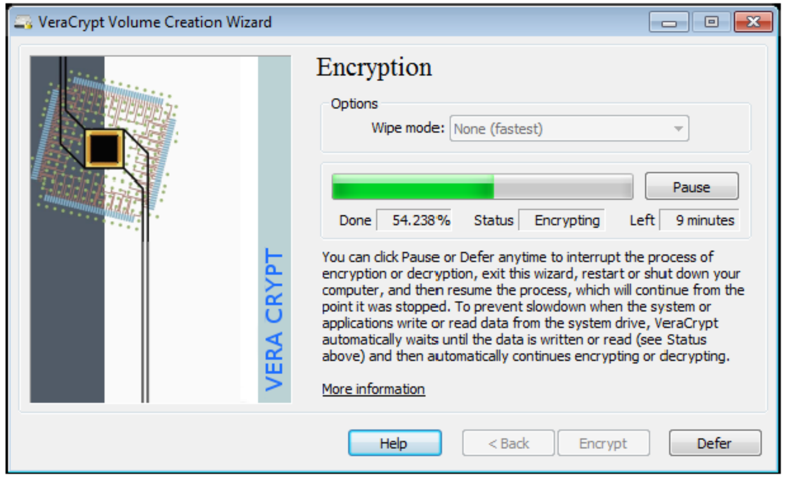
Issued to: Go Daddy Class 2 Certification Authority

Issued by: Go Daddy Class 2 Certification Authority

Valid from 6/29/2004 **to** 6/29/2034

Software-Based Encryption

- Uses software tools to encrypt your data:
  - BitLocker, Windows EFS, VeraCrypt, 7zip
- Typically as secure as the Operating System.
- A vulnerability in the Operating System can compromise the encryption software.



VeraCrypt Volume Creation Wizard

Encryption

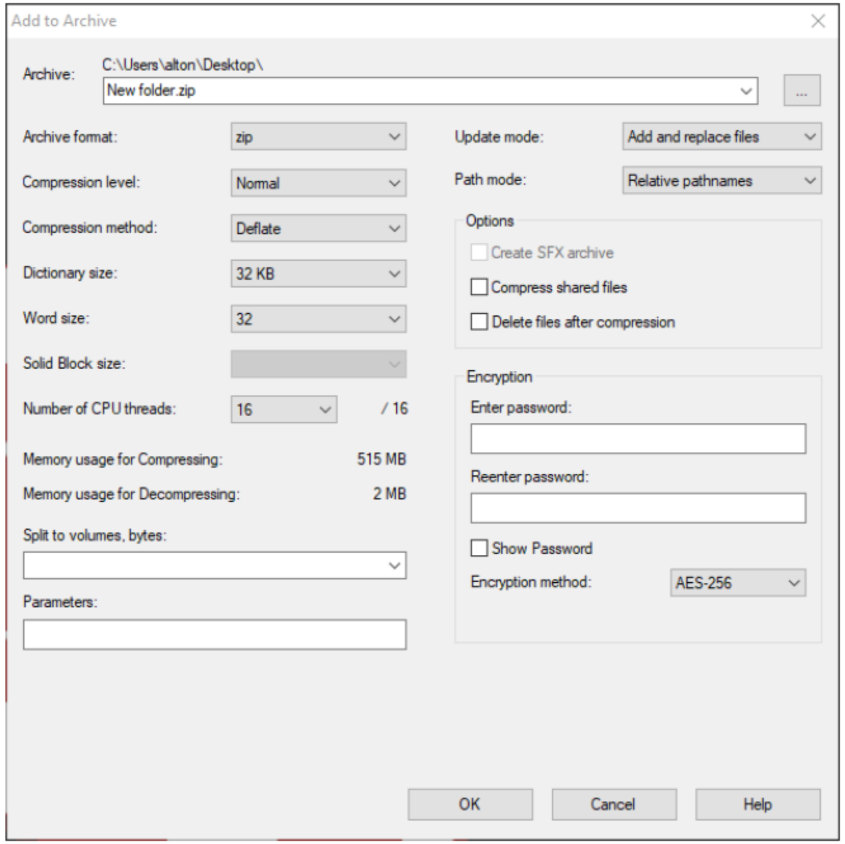
Options

Wipe mode: None (fastest)

Done 54.238% Status Encrypting Left 9 minutes

You can click Pause or Defer anytime to interrupt the process of encryption or decryption, exit this wizard, restart or shut down your computer, and then resume the process, which will continue from the point it was stopped. To prevent slowdown when the system or applications write or read data from the system drive, VeraCrypt automatically waits until the data is written or read (see Status above) and then automatically continues encrypting or decrypting.

More information



Add to Archive

Archive: C:\Users\alton\Desktop\
New folder.zip

Archive format: zip

Compression level: Normal

Compression method: Deflate

Dictionary size: 32 KB

Word size: 32

Solid Block size:

Number of CPU threads: 16 / 16

Memory usage for Compressing: 515 MB

Memory usage for Decompressing: 2 MB

Split to volumes, bytes:

Parameters:

Update mode: Add and replace files

Path mode: Relative pathnames

Options

□ Create SFX archive

□ Compress shared files

□ Delete files after compression

Encryption

Enter password:

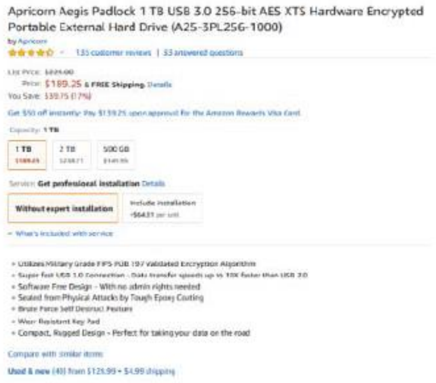
Reenter password:

□ Show Password

Encryption method: AES-256

Hardware-Based Encryption

- Uses hardware to perform encryption:
  - TPM (Crypto Processor)
  - Processors with x86 Instruction Set (AES Encryption)
- Many times, stand alone USB hard drives.



Apricorn Aegis Padlock 1 TB USB 3.0 256-bit AES XTS Hardware Encrypted Portable External Hard Drive (A25-3PL256-1000)

by Apricorn

135 customer reviews | 33 answered questions

List Price: $229.00

Price: $189.25 & FREE Shipping. Details

You Save: $39.75 (17%)

Get $50 off instantly: Pay $139.25 upon approval for the Amazon Rewards Visa Card

Capacity: 1 TB

1 TB $189.25 | 2 TB $258.71 | 500 GB $141.95

Service: Get professional installation Details

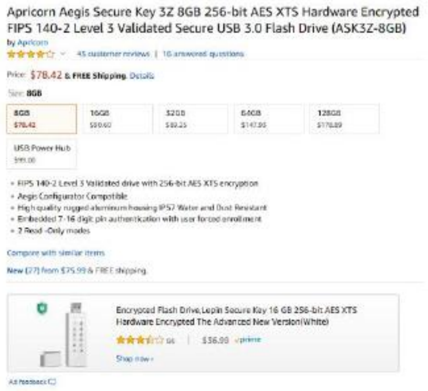
Without expert installation | Include installation +$64.31 per unit

› What's included with service

- Utilizes Military Grade FIPS PUB 197 Validated Encryption Algorithm
- Super fast USB 3.0 Connection - Data transfer speeds up to 10X faster than USB 2.0
- Software Free Design - With no admin rights needed
- Sealed from Physical Attacks by Tough Epoxy Coating
- Brute Force Self Destruct Feature
- Wear Resistant Key Pad
- Compact, Rugged Design - Perfect for taking your data on the road

Compare with similar items

Used & new (40) from $125.99 + $4.99 shipping



Apricorn Aegis Secure Key 3Z 8GB 256-bit AES XTS Hardware Encrypted FIPS 140-2 Level 3 Validated Secure USB 3.0 Flash Drive (ASK3Z-8GB)

by Apricorn

45 customer reviews | 16 answered questions

Price: $78.42 & FREE Shipping. Details

Size: 8GB

8GB $78.42 | 16GB $90.60 | 32GB $89.25 | 64GB $147.95 | 128GB $178.89

USB Power Hub $99.00

- FIPS 140-2 Level 3 Validated drive with 256-bit AES XTS encryption
- Aegis Configurator Compatible
- High quality rugged aluminum housing IP57 Water and Dust Resistant
- Embedded 7-16 digit pin authentication with user forced enrollment
- 2 Read -Only modes

Compare with similar items

New (27) from $75.99 & FREE shipping.

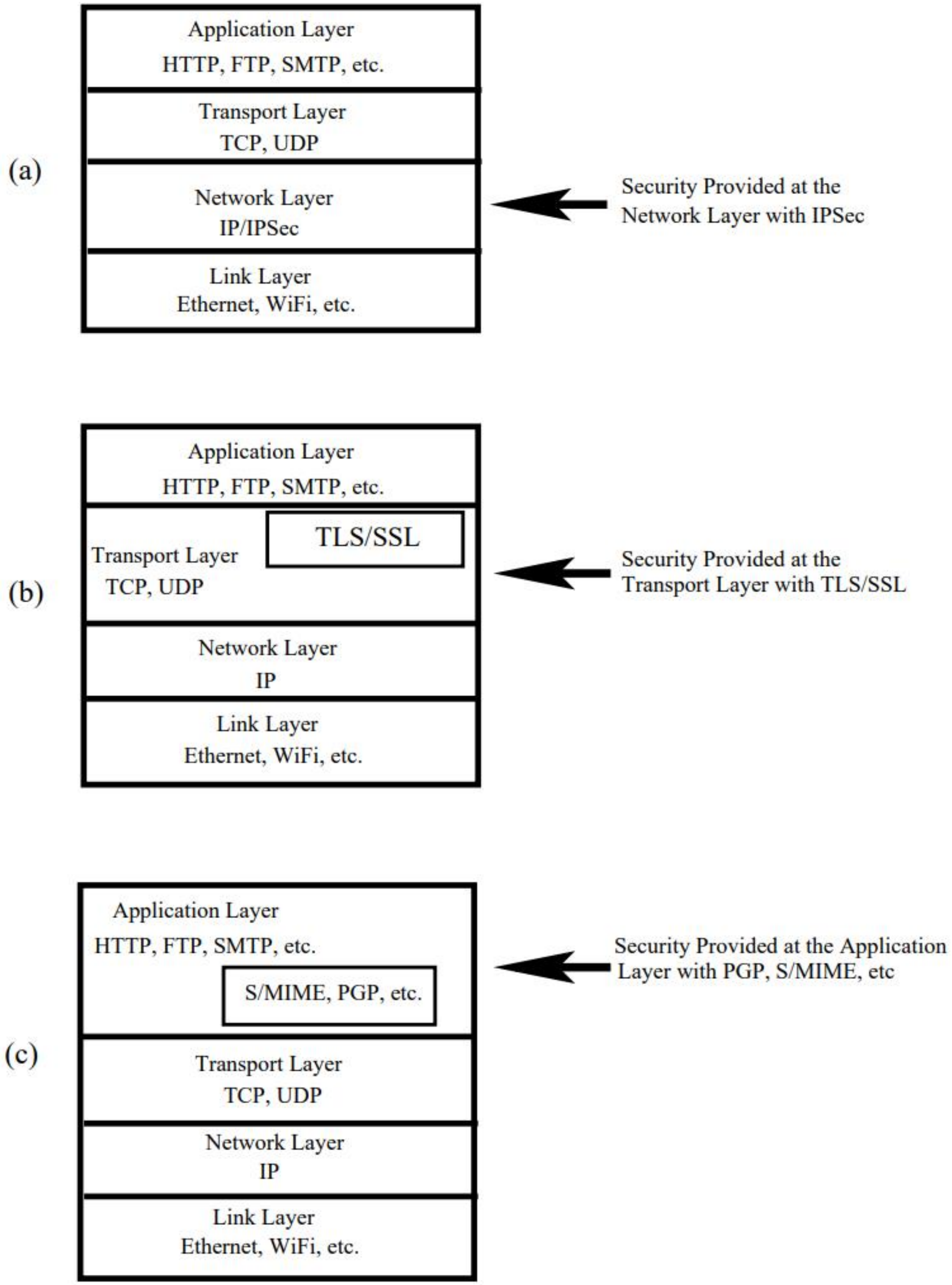
Encrypted Flash Drive,Lepin Secure Key 16 GB 256-bit AES XTS Hardware Encrypted The Advanced New Version(White)

96 | $36.99 ✓prime

Shop now ›

Ad feedback

Four Layer Representation of the TCP/IP Protocol Stack



(a)

Application Layer
HTTP, FTP, SMTP, etc.

Transport Layer
TCP, UDP

Network Layer
IP/IPSec

Link Layer
Ethernet, WiFi, etc.

Security Provided at the Network Layer with IPSec

(b)

Application Layer
HTTP, FTP, SMTP, etc.

Transport Layer
TCP, UDP

TLS/SSL

Network Layer
IP

Link Layer
Ethernet, WiFi, etc.

Security Provided at the Transport Layer with TLS/SSL

(c)

Application Layer
HTTP, FTP, SMTP, etc.

S/MIME, PGP, etc.

Transport Layer
TCP, UDP

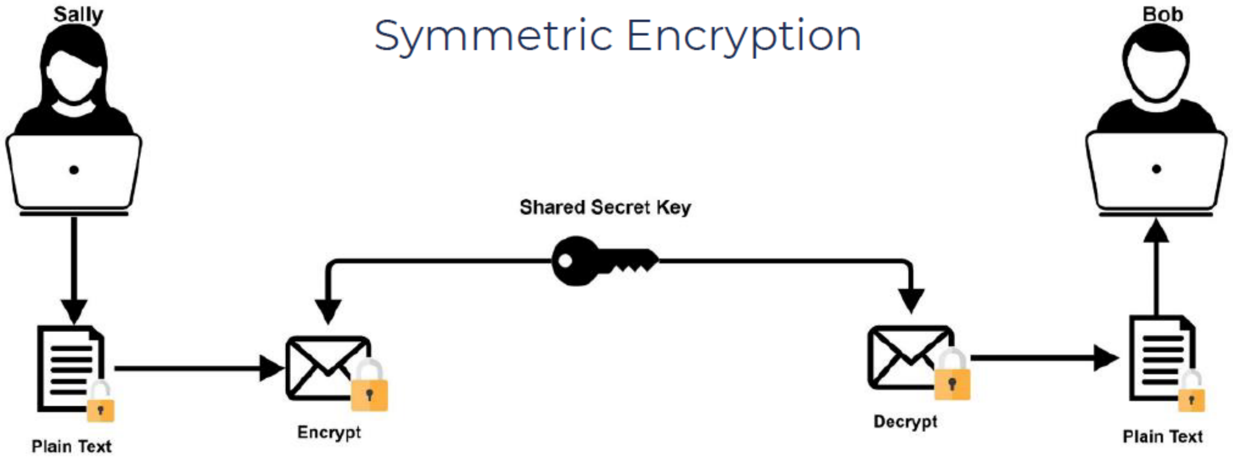
Network Layer
IP

Link Layer
Ethernet, WiFi, etc.

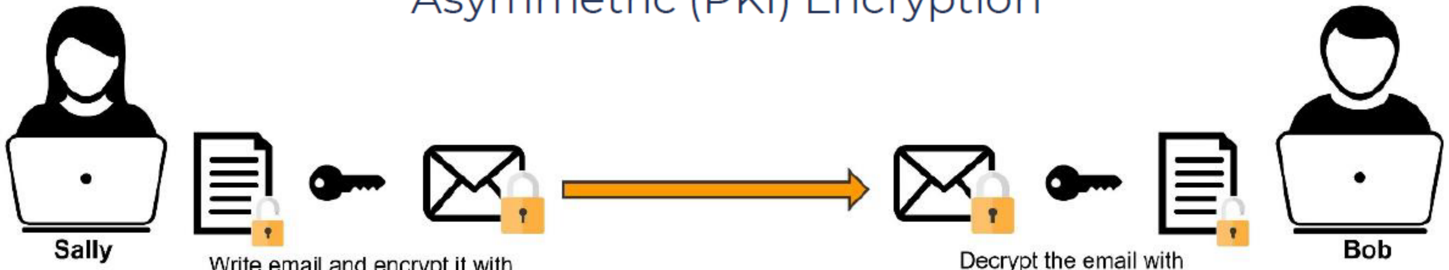
Security Provided at the Application Layer with PGP, S/MIME, etc

Email Confidentiality

Symmetric Encryption



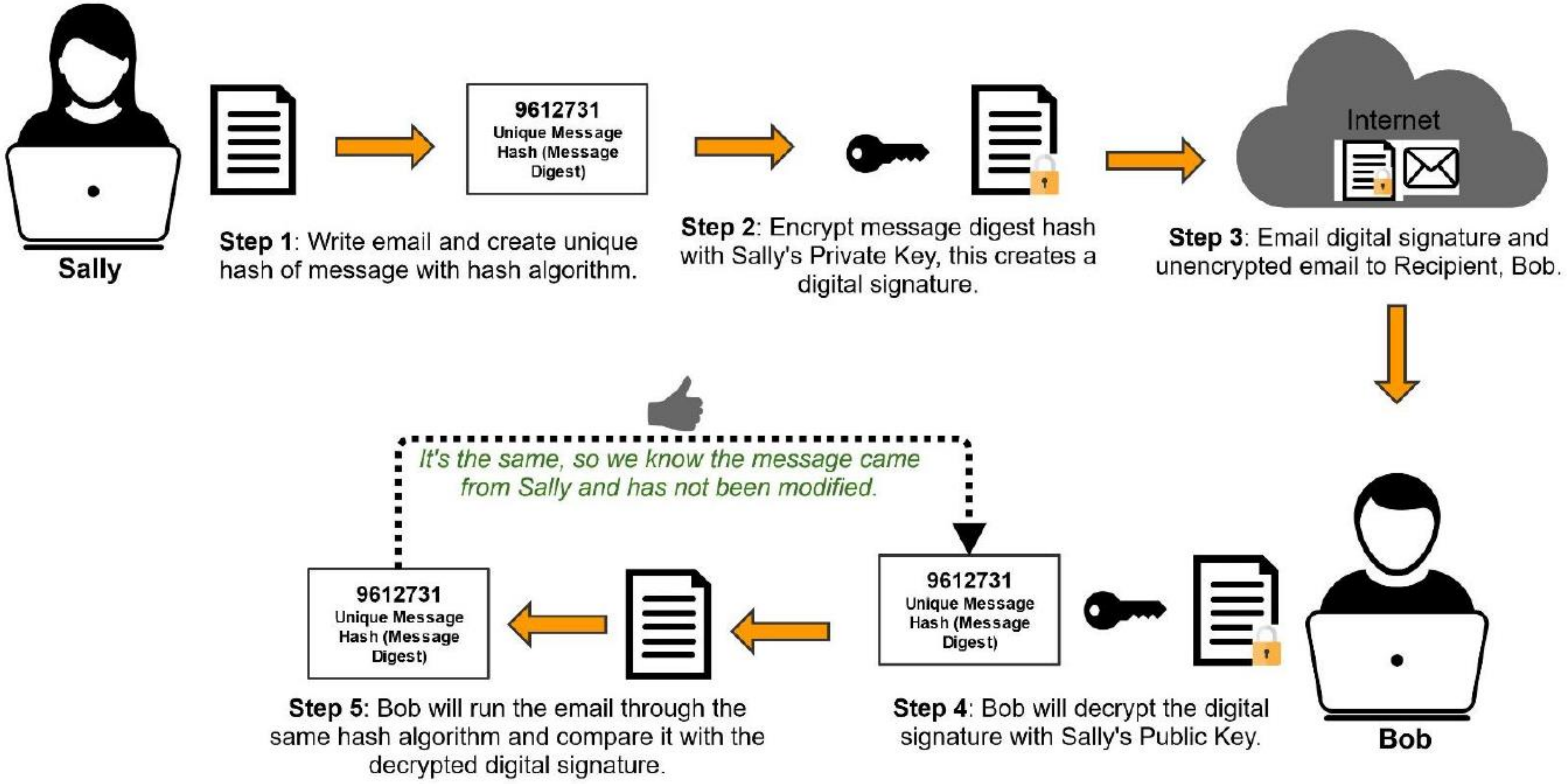
Asymmetric (PKI) Encryption



Write email and encrypt it with **Bob's public key**.

Decrypt the email with **Bob's private key**.

Email Integrity, Authentication & Non-Repudiation



Sally
9612731
Unique Message Hash (Message Digest)
Internet
Bob
9612731
Unique Message Hash (Message Digest)
9612731
Unique Message Hash (Message Digest)

Step 1: Write email and create unique hash of message with hash algorithm.

Step 2: Encrypt message digest hash with Sally's Private Key, this creates a digital signature.

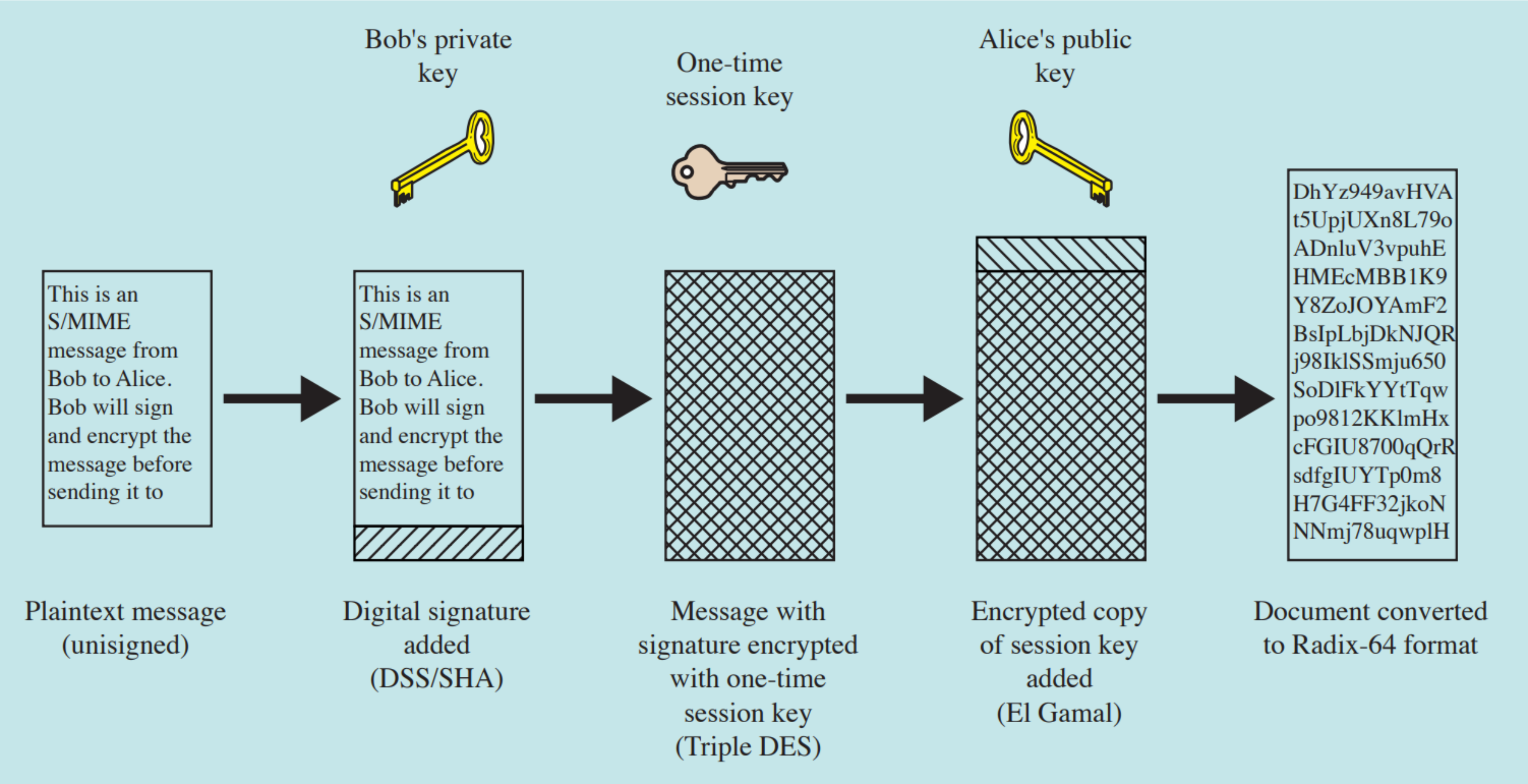
Step 3: Email digital signature and unencrypted email to Recipient, Bob.

Step 4: Bob will decrypt the digital signature with Sally's Public Key.

Step 5: Bob will run the email through the same hash algorithm and compare it with the decrypted digital signature.

It's the same, so we know the message came from Sally and has not been modified.

Typical S/MIME Process



Bob's private key

One-time session key

Alice's public key

This is an S/MIME message from Bob to Alice. Bob will sign and encrypt the message before sending it to

This is an S/MIME message from Bob to Alice. Bob will sign and encrypt the message before sending it to

DhYz949avHVA
t5UpjUXn8L79o
ADnluV3vpuhE
HMEcMBB1K9
Y8ZoJOYAmF2
BsIpLbjDkNJQR
j98IklSSmju650
SoDlFkYYtTqw
po9812KKlmHx
cFGIU8700qQrR
sdfgIUYTp0m8
H7G4FF32jkoN
NNmj78uqwplH

Plaintext message (unisigned)

Digital signature added (DSS/SHA)

Message with signature encrypted with one-time session key (Triple DES)

Encrypted copy of session key added (El Gamal)

Document converted to Radix-64 format

PGP FOR EMAIL SECURITY

- PGP stands for Pretty Good Privacy. It was developed originally by Phil Zimmerman. However, in its incarnation as OpenPGP, it has now become an open-source standard. The standard is described in the document RFC 4880.
- What makes PGP particularly important is that it is now widely used for protecting data in long-term storage.
- Since encryption, even when it is limited to the signature, results in arbitrary binary strings, and since network message transmission is character oriented, we must represent binary data with ASCII strings. PGP uses Base64 encoding for this purpose.

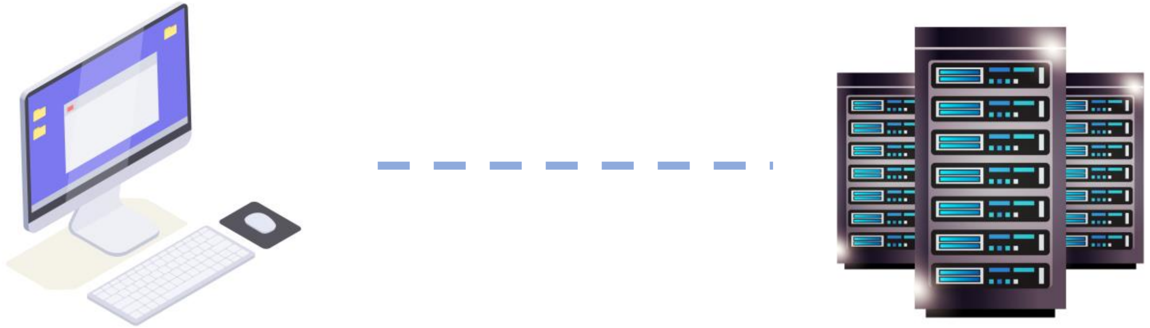
PGP FOR EMAIL SECURITY

- Base64 encoding is referred to as Radix 64 encoding in the PGP documentation. It has emerged as probably the most common way to transmit binary data over a network. To briefly review Base64, it first segments the bytes of the object that needs to be encoded into 6-bit words. The 2^6 = 64 different possible 6-bit words are represented by printable characters as follows: The first 26 are mapped to the uppercase letters A through Z, the next 26 to the lowercase a through z, the next 10 to the digits 0 through 9, and the last two to the characters ’/’ and ’+’. This causes each triple of adjoining bytes to be mapped into four ASCII characters. The Base64 character set includes a 65th character, ‘=’, to indicate how many characters the binary string is short of being an exact multiple of 3 bytes. When the binary string is short one byte, that is indicated by terminating the Base64 string with a single ‘=’. And when it is short two bytes, the termination becomes ‘==’

HTTP and TLS

HTTP and TLS Protocols

*if the **client visits a website** then the data will be fetched from the server with*



***Hypertext Trasfer Protocol (HTTP)** is used for viewing web pages. It is a **request-response protocol** in the client-server computing model*

*every single website is stored on a so-called **server** (images, texts and all the data)*

HTTP and TLS Protocols

- The problem with standard **HTTP** is that it does not use any cryptographic related encryption algorithm
- All information is sent in plain text format (without encryption)
- Usernames, passwords and credit card related details are public
- **THIS IS WHY WE NEED MORE SECURE APPROACHES** and this is exactly why **HTTPS** came to be.

HTTP and TLS Protocols

- **HTTPS** is **Secure Hypertext Transfer Protocol**
- It encrypts the data that is being retrieved by **HTTP**
- There are several **public key** and **private key cryptography** realted approaches it supports (**RSA**, **ECC**, **AES**, **DES** etc.)
- **Https** uses protocols to ensure data security: **ssl** and **tls**
- **SSL** stands for **secure socket layer**
- It is no longer secure this is why back in **1999** it was updated to become **Transport Layer Security (TLS)**

Transport Layer Security

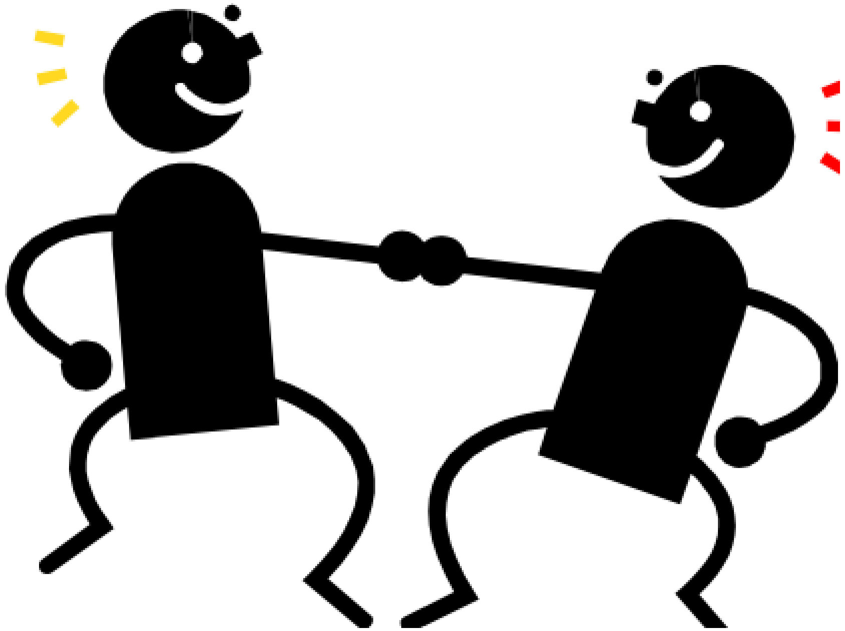
- 1994: Netscape Communications developed the network authentication protocol Secure Sockets Layer, SSLv2.
  - Badly broken, officially deprecated 2011
- 1995: Netscape release their own improvements SSLv3.
  - Broken, officially deprecated 2015
- In January 1999, RFC 2246 was issued by the IETF,
- Transport Layer Security Protocol: TLS 1.0
  - Similar to, but incompatible with SSLv3
  - Followed by TLS 1.1 (2006) and TLS 1.2 (2008)
  - Current version: TLS 1.3 (2018), removes all old/insecure features/algorithms

TLS: Overview

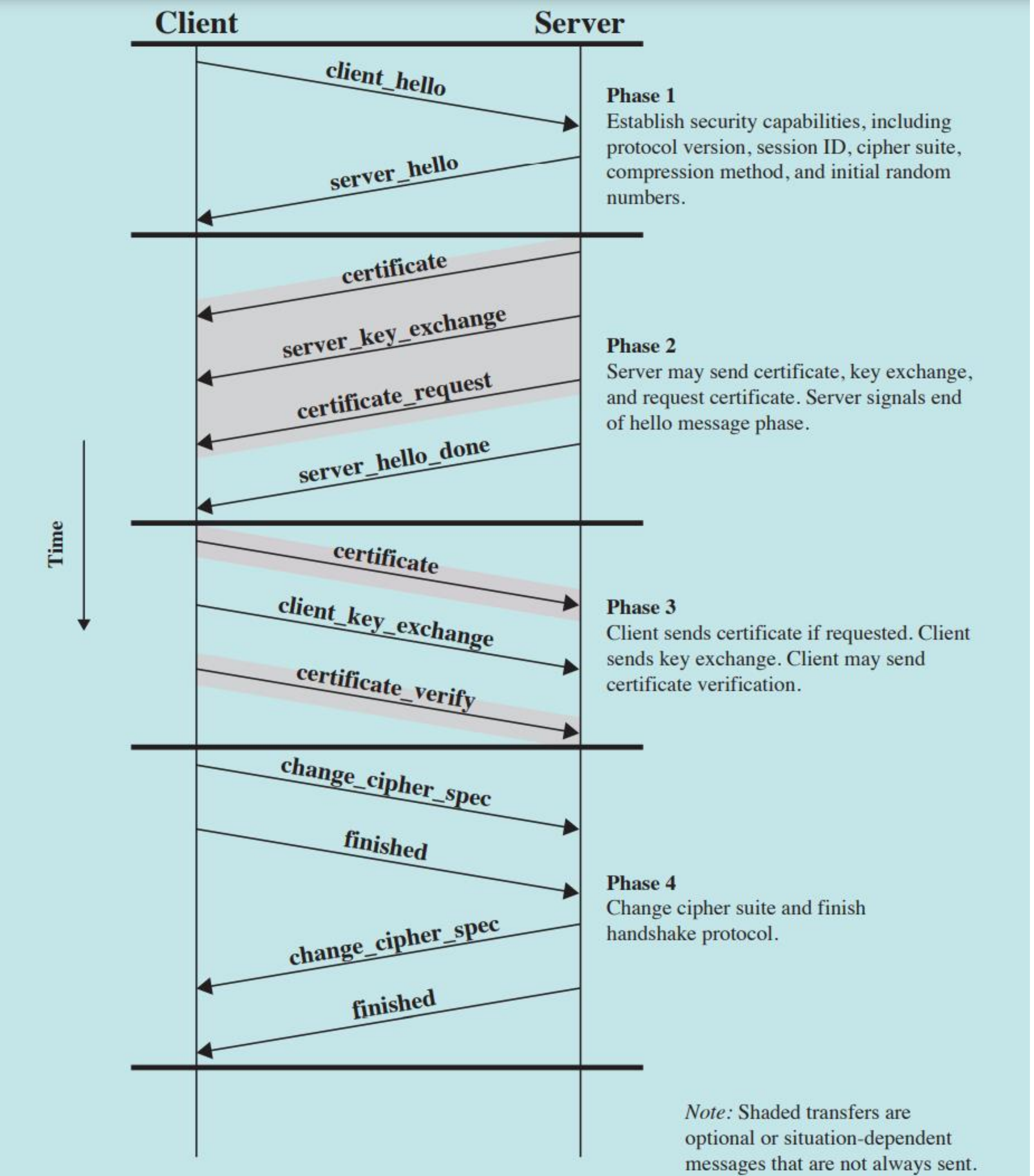
- TLS is a cryptographic services protocol based on the Browser PKI and is commonly used on the Internet.
  - Each server has a server certificate and private key installed
  - Allows browsers to establish secure sessions with web servers.
- Port 443 is reserved for HTTP over TLS/SSL and the protocol https is used with this port.
  - http://www.xxx.com implies using standard HTTP using port 80.
  - https://www.xxx.com implies HTTP over TLS/SSL with port 443.

Handshake Protocol

- The handshake protocol
  - Negotiates the encryption to be used
  - Establishes a shared session key
  - Authenticates the server
  - Authenticates the client (optional)
- After the handshake, application data is transmitted securely (encrypted + integrity protected)



SSL Handshake Protocol



Phase 1
Establish security capabilities, including protocol version, session ID, cipher suite, compression method, and initial random numbers.

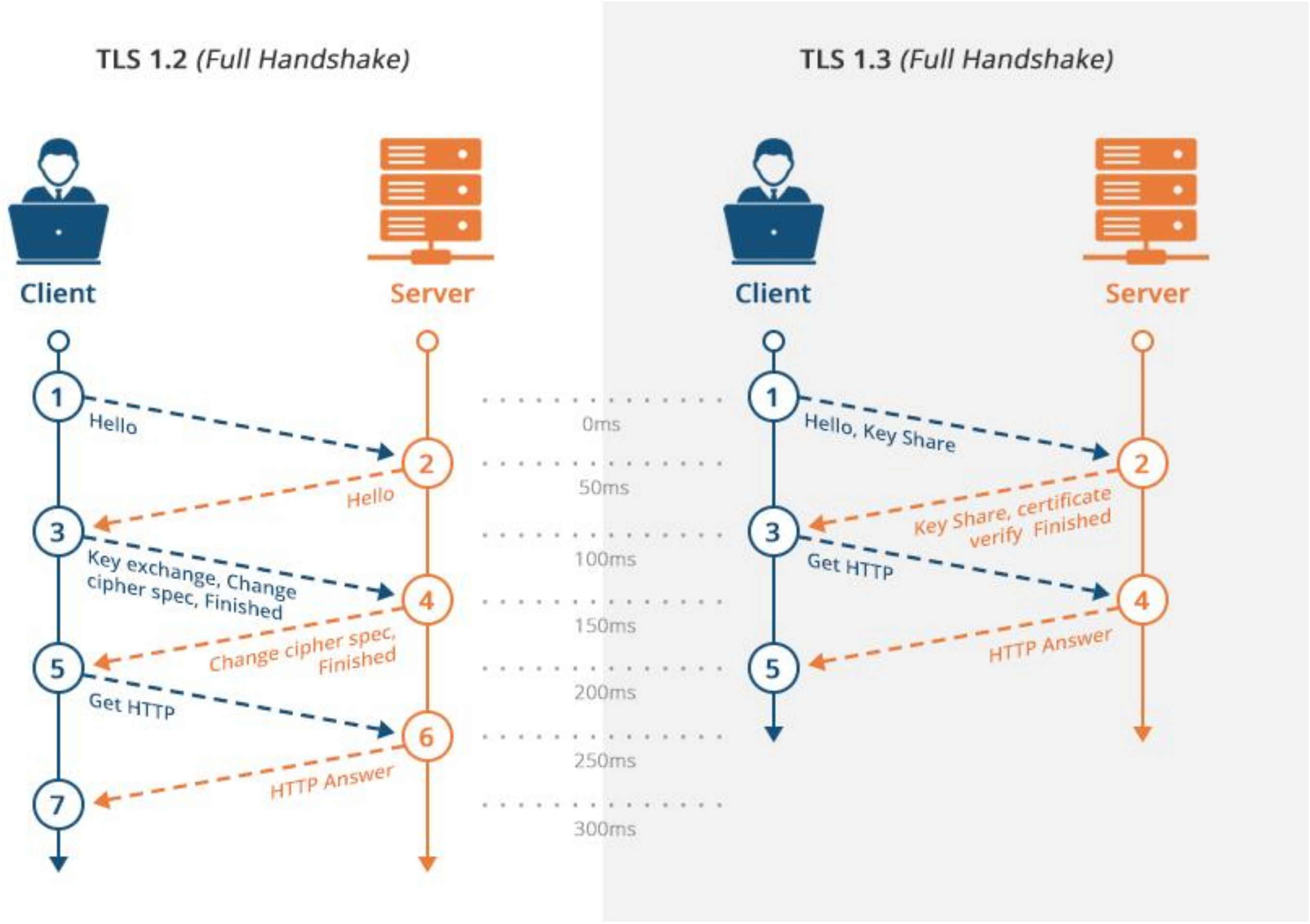
Phase 2
Server may send certificate, key exchange, and request certificate. Server signals end of hello message phase.

Phase 3
Client sends certificate if requested. Client sends key exchange. Client may send certificate verification.

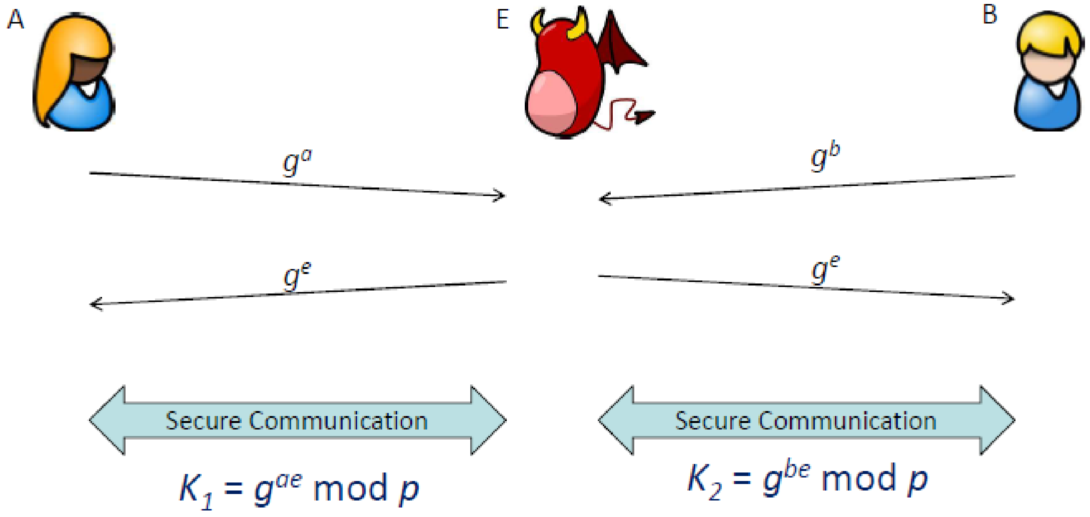
Phase 4
Change cipher suite and finish handshake protocol.

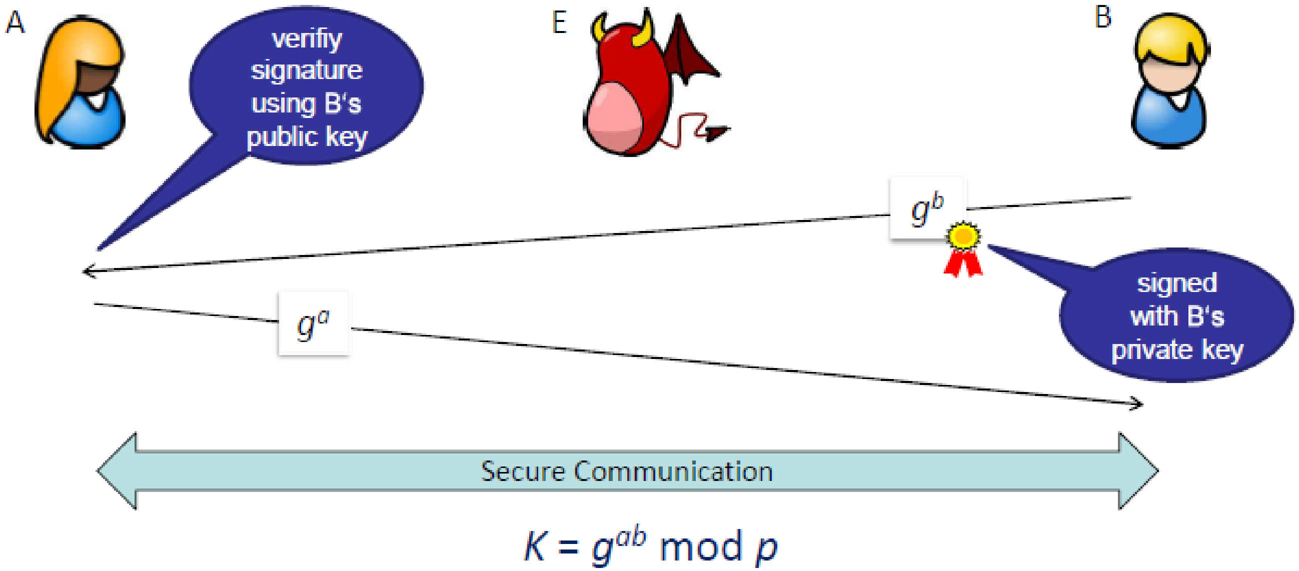
Note: Shaded transfers are optional or situation-dependent messages that are not always sent.

TLS Protocols



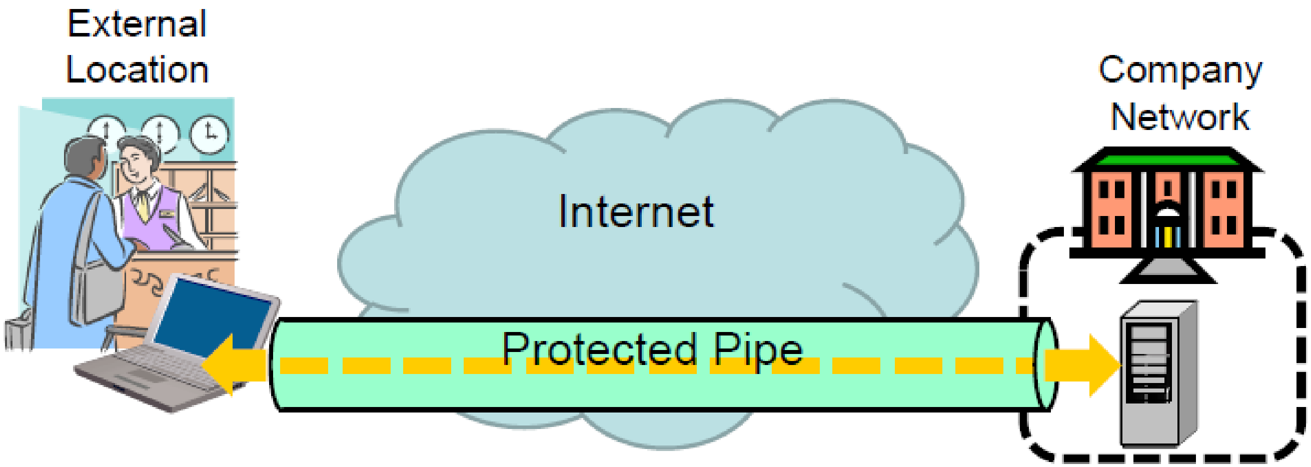
Weakness of DH Key Exchange




A
verifiy signature using B's public key
E
B
g^b
g^a
signed with B's private key
Secure Communication
$K = g^{ab} \bmod p$

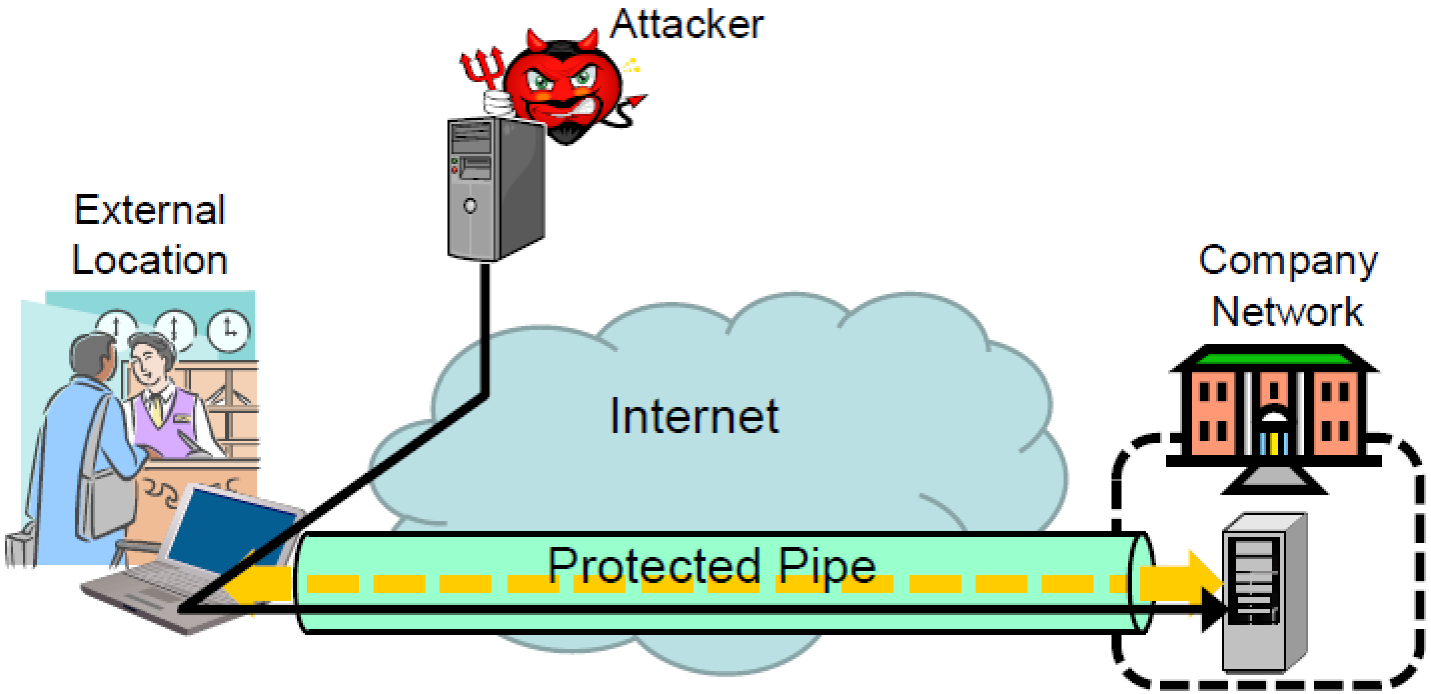
Virtual Private Networks & Firewall

VPN



- TLS secures only a single TCP connection
- Sometimes:
  - all communication from a computer shall be secured
  - also non-TCP communication shall be secured
- Typical application:
  - VPN tunnel into a company network
  - Tunnel can only be established after authentication
  - All communication is routed (and secured) through the tunnel
  - Client is virtually part of the local company network
  - Client gets access to internal services

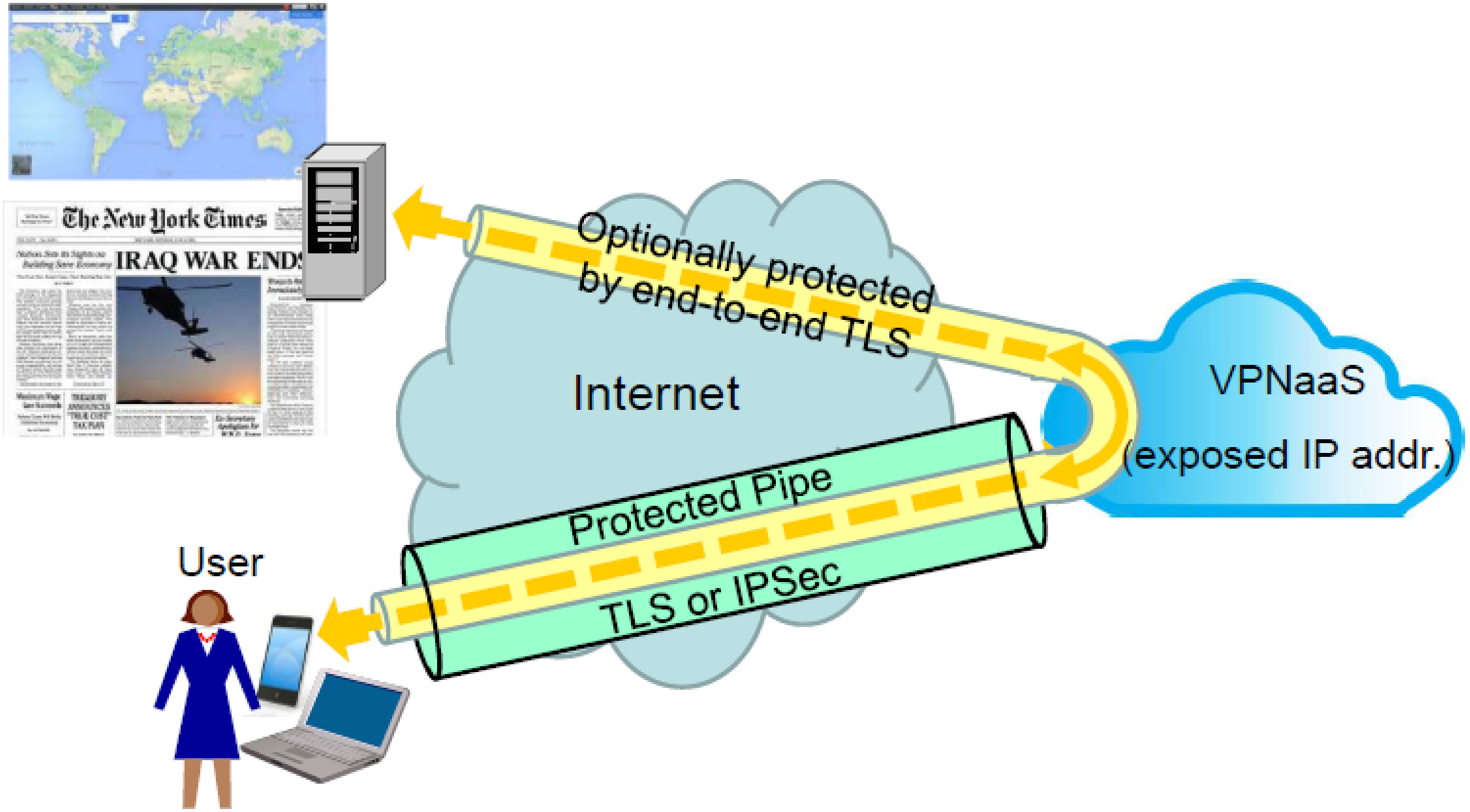
Risk of using VPN



Secure pipe can be attack channel to company network !

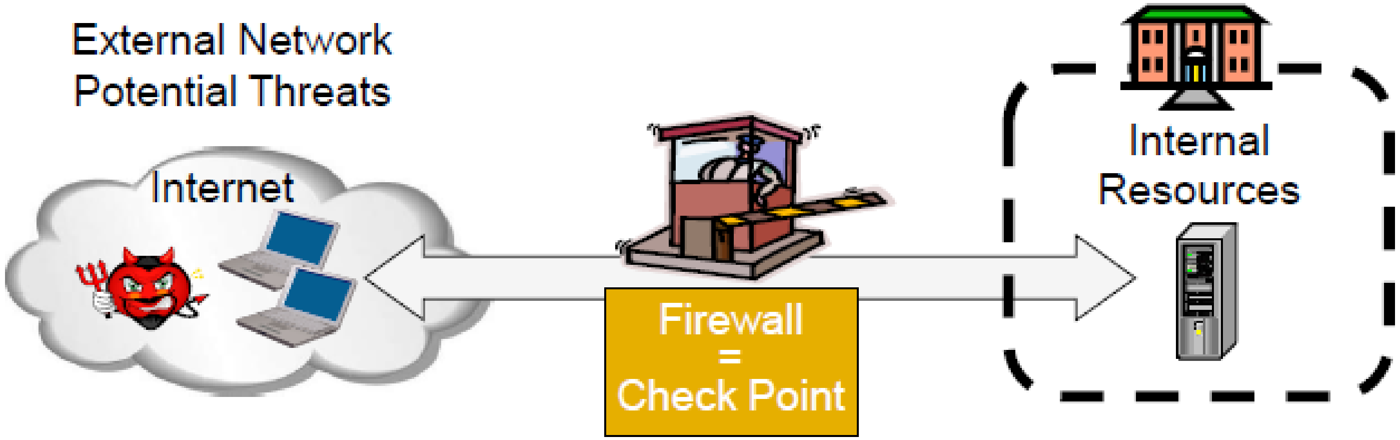
VPN Browsing via VPN Proxy

- Usage Examples:
  - Access to services subscribed by own organization
  - Hide users true location



Firewalls

- A firewall is a check point that protects the internal networks against attack from outside networks
- The check point decides which traffic can pass in & out based on rules



Firewall

- If the risk of having a connection to the Internet is unacceptable, the most effective way of treating the risk is to avoid the risk altogether and disconnect completely.
- If disconnection from the Internet is not practical, then firewalls may provide an effective level of protection that can reduce the risk to an acceptable level.
- Firewalls are often the first line of defense against external attacks but should not be the only defense.

Firewall

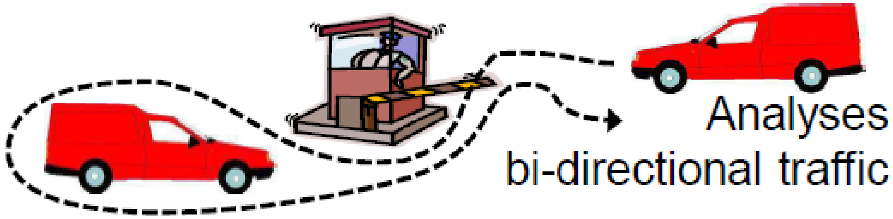
- Firewalls are the foundation of a defense-in-depth network security strategy.
- They’re designed to protect organizations from network-based attacks. Firewalls do this by filtering data packets that go through them.
- The network owner must define criteria for what is (un)authorized
- The effectiveness of firewalls depends on specifying authorized traffic in terms of rules
    - The rules defines what to let pass through;
    - The rules defines what to block.
- They can be a standalone network device or software on a computer system, meaning **network-based** (hardware) or **host-based**(software).

3 Common Types of Firewalls

Packet Filters

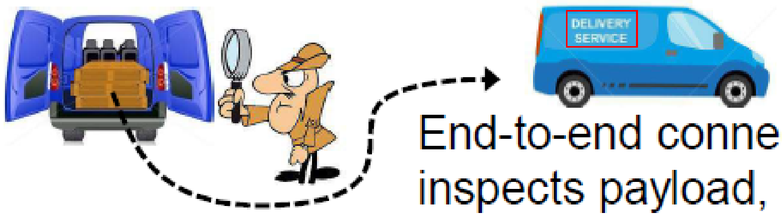
Inspects packet
headers only

Stateful Packet Filters



Analyses
bi-directional traffic

Application Level Gateway/
Next Generation Firewall



End-to-end connection
inspects payload, and
analyses traffic

1st Gen: Packet Filtering Firewalls

- 1st generation and most basic type of firewall.
- They inspect all data packets that attempt to traverse it, and based on predefined rules, packets are either allowed or denied.
- These predefined rules are commonly called an Access Control List (ACL).
- Considered Stateless Firewalls.
- Packet filtering rules are common TCP/IP packet attributes:
  - **IP Address**
    - Source IP Address
    - Destination IP Address
  - **TCP/UDP Port**
    - Source TCP/UDP Port
    - Destination TCP/UDP Port
  - **Inbound or Outbound**
    - Inbound Firewall Network Interface
    - Outbound Firewall Network Interface

1st Gen: Packet Filtering Firewalls

- Widespread packet filter software (Linux):
  - iptables / netfilter
  - nft / nttables
- Examples (iptables)

❑ iptables -A FORWARD -s 131.234.142.33 -j ACCEPT

  - All packets from source IP Address 131.234.142.33 are accepted

❑ iptables -A FORWARD -p tcp d 10.0.0.56 --dport 22 -j ACCEPT

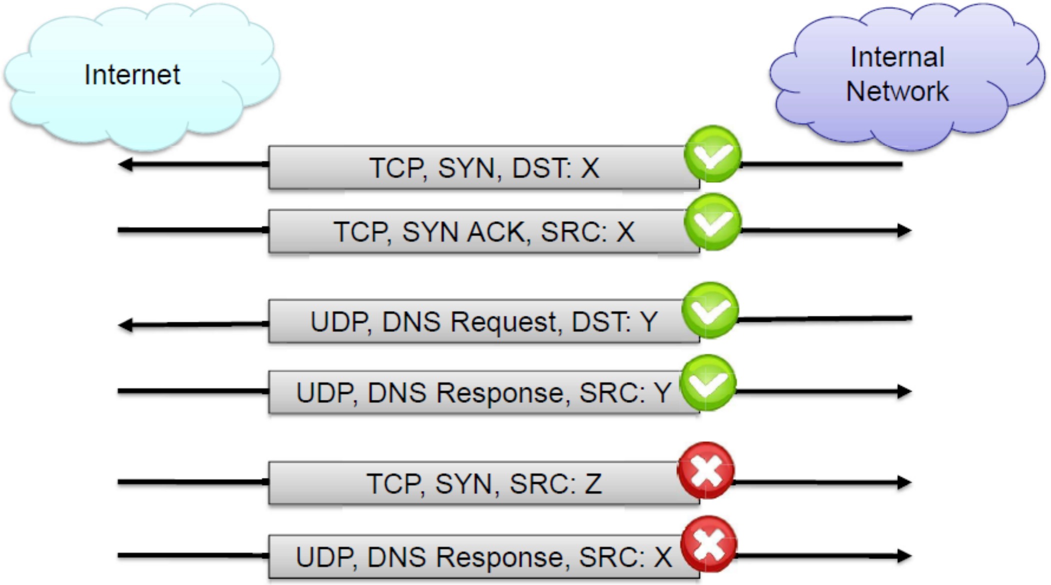
  - All packets using transport protocol and destination address 10.0.0.56 and destination port 22 are accepted

Problems with Stateless Filtering

- Assume a typical security policy
  - Access from internal to external allowed
  - Access from external to internal prohibited
- Example application: home network
- Naive packet filter configuration:
  - outgoing packet forward
  - incoming packet reject

2nd Gen: Stateful Inspection Firewalls

- Operate at the Transport Layer of the OSI Model (Layer 4) and monitor TCP sessions.
- Determine the legitimacy of a requested session by monitoring the 3-way handshake between packets.
- Valid TCP sessions are allowed to pass, while invalid and terminated sessions are not.



(Stateful) Packet Filter: Evaluation

- Strengths:
  - Low overhead and high throughput
  - Supports almost any application
- Weaknesses:
  - Unable to interpret application layer data/commands
  - may allow insecure operations to occur
  - Allows direct connection between hosts inside & outside firewall

3rd Gen: Application-Level Firewalls

- Also known as proxy servers, these firewalls operate at the Application Layer of the OSI Model (Layer 7).
- Can be configured to filter specific user applications
  - E.g. Facebook, Youtube, LinkedIn
  - Can filter detailed elements in each specific user application
- Can provide intrusion detection and intrusion prevention
- Very high processing load in firewall
  - High volume needs high performance hardware, or else will be slow

3rd Gen: Application-Level Firewalls

- Specifically, proxy servers can provide the following services:
  - **Filter**: Filters packets based on an application or service (FTP, SMTP, etc.).
  - **Caching**: Provides caching services, for example:
    - When you request a page from a website, the proxy server will retrieve it and then cache it in its memory.
    - The next time someone requests that website, the proxy server can retrieve it from its cache.
    - This saves Internet bandwidth.
  - **Logging**: Has the ability to log user activity for auditing purposes.

3rd Gen: Application-Level Firewalls

- Strengths:
  - Easy logging and audit of all incoming traffic
  - Provides potential for best security through control of application layer data/commands
- Weaknesses:
  - May require some time for adapting to new applications
  - Much slower than packet filters
  - Much more expensive than packet filters

Intrusion Detection and Prevention System

Intrusion

- Intrusion
  - Actions aimed at compromising the security of a target network (confidentiality, integrity, availability of resources)
- Intrusions have many causes, such as malware (worms, spyware, etc…), attackers gaining unauthorized access to systems from the Internet, and authorized users of systems who misuse their privileges or attempt to gain additional privileges for which they are not authorized.
- Although many intrusions are malicious in nature, many others are not; for example: a person might mistype the address of a computer and accidentally attempt to connect to a different system without authorization.

Intrusion Detection and Prevention

- Intrusion detection
  - The identification of possible intrusion through intrusion signatures and network activity analysis
  - IDS: Intrusion Detection Systems
- Intrusion prevention
  - The process of both detecting intrusion activities and managing automatic responsive actions throughout the network
  - IPS: Intrusion Prevention Systems
  - IDPS: Intrusion Detection and Prevention Systems
- IPSs combine IDSs and improved firewall technologies, they make access control decisions based on application content, rather than IP address or ports as traditional firewalls had done.

Intrusion Detection and Prevention

- IDPSs are primarily focused on:
  - Identifying possible incidents, logging information about them, attempting to stop them, and reporting them to security administrators.
  - Identifying problems with security policies
  - Documenting existing threats
  - Deterring individuals from violating security policies.

IPS

- IPSs respond to a detected threat by attempting to prevent it from succeeding. They use several response techniques:
- The IPS stops the attack itself. Examples:
  - Terminate the network connection or user session that is being used for the attack. Block access to the target (or possibly other likely targets) from the offending user account, IP address, or other attacker attribute. Block all access to the targeted host, service, application, or other resource.

IPS

- The IPS changes the security environment. The IPS could change the configuration of other security controls to disrupt an attack. Such as reconfiguring a network device (e.g., firewall, router, switch) to block access from the attacker or to the target, and altering a host-based firewall on a target to block incoming attacks. Some IPSs can even cause patches to be applied to a host if the IPS detects that the host has vulnerabilities.
- The IPS changes the attack's content. Some IPS technologies can remove or replace malicious portions of an attack to make it benign. An example is an IPS removing an infected file attachment from an e-mail and then permitting the cleaned email to reach its recipient.

Classes of detection methodologies:

- **Signature-based:** compares known threat signatures to observed events to identify incidents.
- This is very effective at detecting known threats but largely ineffective at detecting unknown threats and many variants on known threats. It has fewer false positive alarms.
- Signature-based detection cannot track and understand the state of complex communications, so it cannot detect most attacks that comprise multiple events.

Example:

  - An e-mail with a subject of “Free pictures!” and an attachment filename of “freepics.exe”, which are characteristics of a known form of malware

Classes of detection methodologies:

- **Anomaly-based detection**: Using a model of normal system behavior, try to detect deviations and abnormalities

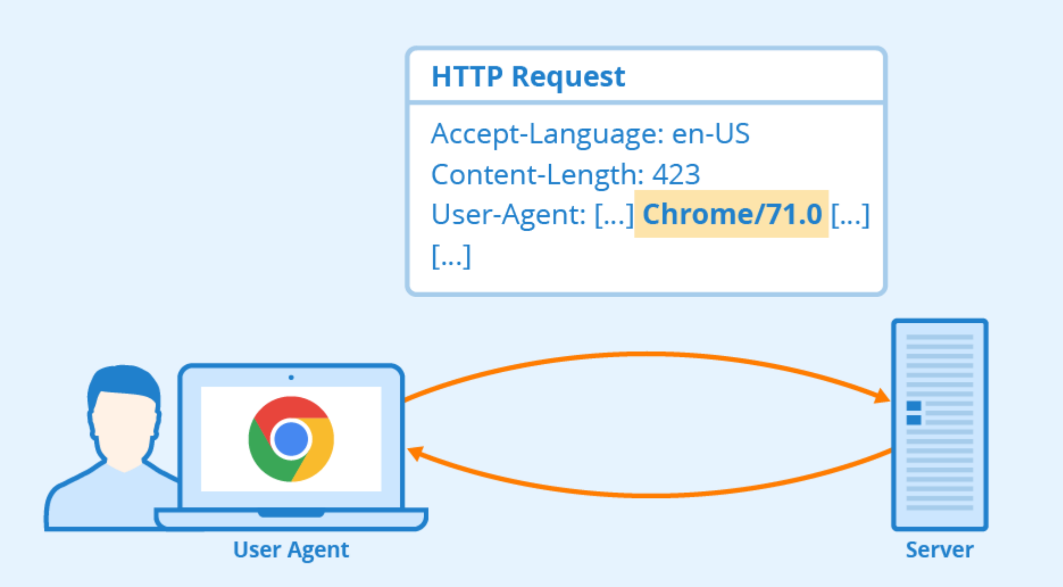
- Anomaly-based detection can detect new types of attacks.
- Requires much more overhead and processing capacity than signature-based .
- May generate many false positives.

Classes of detection methodologies:

- **Anomaly detection example:** a profile for a network might show that Web activity comprises an average of 13% of network bandwidth at the Internet border during typical workday hours. The IDPS then uses statistical methods to compare the characteristics of current activity to thresholds related to the profile, such as detecting when Web activity comprises significantly more bandwidth than expected and alerting an administrator of the anomaly. Profiles can be developed for many behavioral attributes, such as the number of e-mails sent by a user, the number of failed login attempts for a host, and the level of processor usage for a host in a given period of time.

- **Stateful protocol analysis:** A key development in IDPS technologies was the use of protocol analyzers.
- Protocol analyzers can natively decode application-layer network protocols, like HTTP or FTP. Once the protocols are fully decoded, the IPS analysis engine can evaluate different parts of the protocol for anomalous behavior or exploits against predetermined profiles of generally accepted definitions of benign protocol activity for each protocol state.
- Problems with this type include that it is often very difficult or impossible to develop completely accurate models of protocols, it is very resource-intensive, and it cannot detect attacks that do not violate the characteristics of generally acceptable protocol behavior.

- For example: the existence of a large binary file in the User-Agent field of an HTTP request would be very unusual and likely an intrusion. A protocol analyzer could detect this anomalous behavior and instruct the IPS engine to drop the offending packets.



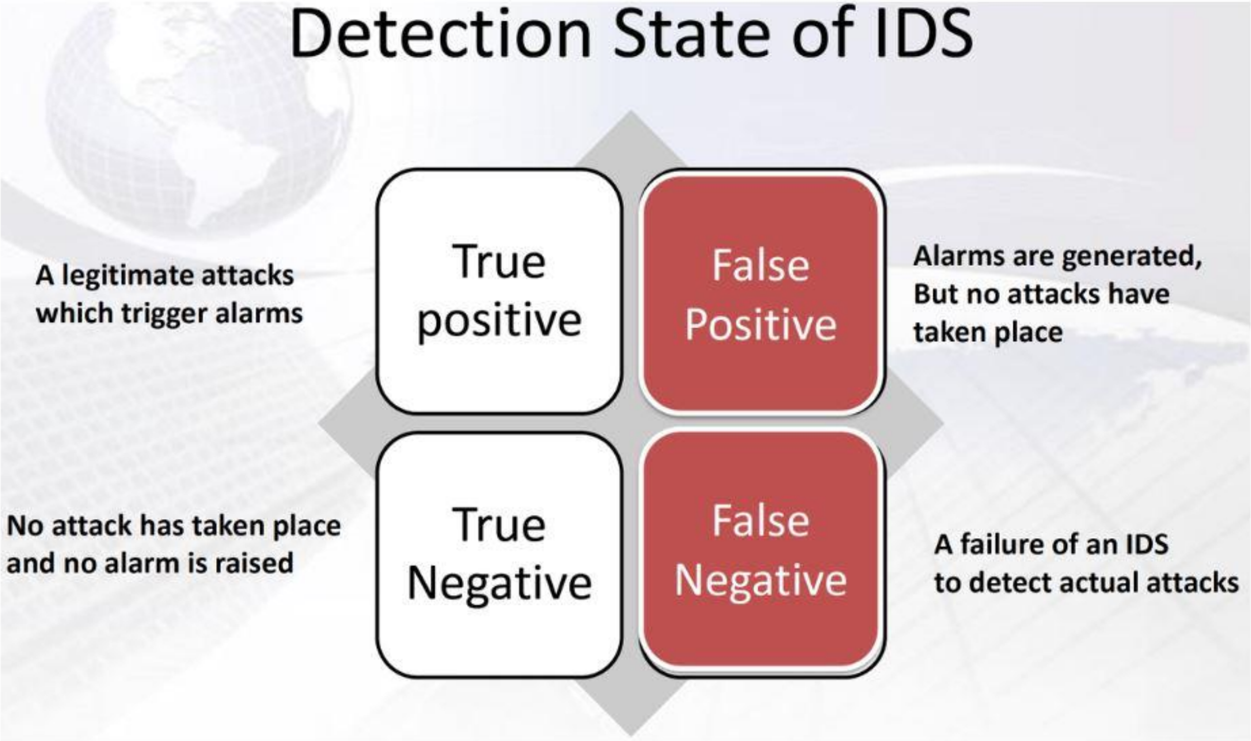
HTTP Request

Accept-Language: en-US
Content-Length: 423
User-Agent: [...] **Chrome/71.0** [...]
[...]

IDPS technologies cannot provide completely accurate detection. When an IDPS incorrectly identifies benign activity as being malicious, a false positive has occurred. When an IDPS fails to identify malicious activity, a false negative has occurred. It is not possible to eliminate all false positives and negatives; in most cases, reducing the occurrences of one increases the occurrences of the other.

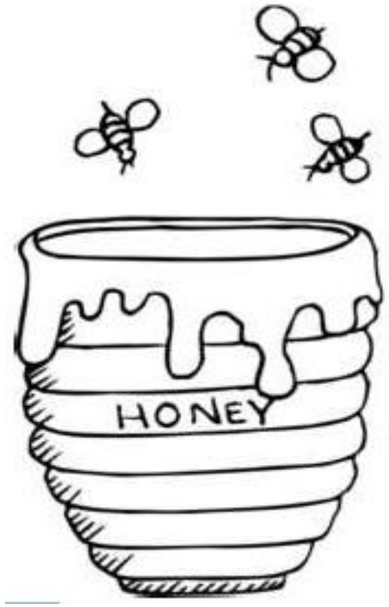
Intrusion Detection Errors

- False negatives: attack is not detected
    - Big problem in signature-based misuse detection
- False positives: harmless behavior is classified as attack
    - Big problem in statistical anomaly detection
- Both types of IDS suffer from both error types


Detection State of IDS
A legitimate attacks which trigger alarms
True positive
False Positive
Alarms are generated, But no attacks have taken place
No attack has taken place and no alarm is raised
True Negative
False Negative
A failure of an IDS to detect actual attacks

- Many organizations choose to decrease false negatives at the cost of increasing false positives, which means that more malicious events are detected but more analysis resources are needed to differentiate false positives from true malicious events. Altering the configuration of an IDPS to improve its detection accuracy is known as tuning.
- Most alarms are false positives
  - Requires automated screening and filtering of alarms
- Most true positives are trivial incidents
  - can be ignored,
  - the attacks will never be able to penetrate any system

Honeypots



- A honeypot:
  - is a computer configured to detect network attacks or malicious behavior,
  - appears to be part of a network, and seems to contain information or a resource of value to attackers.
- But honeypots are isolated, are never advertised and are continuously monitored
- All connections to honeypots are per definition malicious
- Can be used to extract attack signatures
- Honeynet is an international security club, see next slide